MITSUBISHI DATA BOOK 1982

LSI

MITSUBISHI
ELECTRIC

MITSUBISHI DATA BOOK 1982

LSI

MITSUBISHI
ELECTRIC

INDEXES 1

RANDOM-ACCESS MEMORIES 2

READ-ONLY MEMORIES 3

MELPS 4 MICROCOMPUTERS 4

MELPS 41/42 MICROCOMPUTERS 5

MELPS 8-48 MICROCOMPUTERS 6

MELPS 8/85 MICROPROCESSORS 7

LSIs FOR PERIPHERAL CIRCUITS 8

MELPS 86 MICROPROCESSORS 9

SPEECH SYNTHESIS LSIs (PARCOR SYSTEM) 10

GENERAL-PURPOSE MOS LSIs 11

MICROCOMPUTER SYSTEMS 12

MICROCOMPUTER SUPPORT SYSTEMS 13

MICROCOMPUTER SOFTWARE 14

APPLICATIONS 15

PREFACE

Thank you for your continued patronage of Mitsubishi Electric and our semiconductor products.

Semiconductor devices are a mainstay of the burgeoning electronics industry, where they are finding more and more applications, and meeting demands for increased sophistication and diversification of performance and function.

This data book has been compiled to be as complete as possible, including data on large-scale IC memories, single-chip microcomputers, peripheral LSIs for 16-bit parallel processing CPUs, speech synthesis LSIs and microcomputer development support equipment, with the addition of a variety of originally developed MOS LSI devices.

We hope you will let us know of any mistakes or omissions that come to your attention, and any suggestions you might have on improving the usefulness of this data book.

January, 1982

Kimio Sato, General Manager
Semiconductors Division
Mitsubishi Electric Corporation

CONTENTS

1 INDEXES

Page

Index by Function .... 1—2
Index by Type Designation .... 1—7
Guide to Interchangeability .... 1—10
Guide to Selection of RAMs, PROMs, EPROMs and ROMs .... 1—14
Ordering Information .... 1—15
Package Outlines .... 1—17
Terminology .... 1—28
Letter Symbols for The Dynamic Parameters .... 1—33
Symbology .... 1—36
Quality Assurance and Reliability Testing .... 1—39
Precautions in Handling MOS ICs .... 1—44

2 RANDOM-ACCESS MEMORIES

M58725P, P-15 16384-Bit (2048-Word by 8-Bit) Static RAM .... 2—3
M58981P-30, P-45 4096-Bit (1024-Word by 4-Bit) CMOS Static RAM .... 2—9
M5K4116P-2, P-3 16384-Bit (16384-Word by 1-Bit) Dynamic RAM .... 2—13
M5K4164P-15, P-20 65536-Bit (65536-Word by 1-Bit) Dynamic RAM .... 2—25
M5K4164NP-15, NP-20 65536-Bit (65536-Word by 1-Bit) Dynamic RAM .... 2—41
M5K4164S-15, S-20 65536-Bit (65536-Word by 1-Bit) Dynamic RAM .... 2—55
M5K4164NS-15, NS-20 65536-Bit (65536-Word by 1-Bit) Dynamic RAM .... 2—71
M5L2114LP, P-2, P-3 4096-Bit (1024-Word by 4-Bit) Static RAM .... 2—85
M5L5101LP-1 1024-Bit (256-Word by 4-Bit) CMOS Static RAM .... 2—89
M5T4044P-20, P-30, P-45 4096-Bit (4096-Word by 1-Bit) Static RAM .... 2—93

3 READ-ONLY MEMORIES

Development of Mask-Programmable ROMs .... 3—3
M54700P, S 1024-Bit (256-Word by 4-Bit) Field-Programmable ROM with Open-Collector Outputs .... 3—5
M54730P, S 256-Bit (32-Word by 8-Bit) Field-Programmable ROM with Open-Collector Outputs .... 3—10
M54740AP, S/M54741AP, S 4096-Bit (1024-Word by 4-Bit) Field-Programmable ROM .... 3—14
M58653P 700-Bit (50-Word by 14-Bit) Electrically Alterable ROM .... 3—18
M58735-XXXP 32768-Bit (4096-Word by 8-Bit) Mask-Programmable ROM .... 3—22
M5G1400P 1400-Bit (100-Word by 14-Bit) Electrically Alterable ROM .... 3—24
M5L2716K, K-65 16384-Bit (2048-Word by 8-Bit) Erasable and Electrically Reprogrammable ROM .... 3—28
M5L2732K, K-6 32768-Bit (4096-Word by 8-Bit) Erasable and Electrically Reprogrammable ROM .... 3—32
M5L2764K, K-2, K-3 65536-Bit (8192-Word by 8-Bit) Erasable and Electrically Reprogrammable ROM .... 3—36

4 MELPS 4 MICROCOMPUTERS

M58840-XXXP, M58841-XXXSP Single-Chip 4-Bit Microcomputer with 8-Bit A/D Converter .... 4—2
M58842S MELPS 4 System Evaluation Device .... 4—13
M58843-XXXP, M58844-XXXSP Single-Chip 4-Bit Microcomputer with 8-Bit A/D Converter .... 4—18
M58845-XXXSP Single-Chip 4-Bit Microcomputer with 8-Bit A/D Conveter and Two Timer/Event Counter .... 4—29
M58846-XXXSP Single-Chip 4-Bit Microcmputer with Two Timer/Event Counter .... 4—41
M58847-XXXSP Single-Chip 4-Bit Microcomputer .... 4—53

5 MELPS 41/42 MICROCOMPUTERS

M58494-XXXP Single-Chip 4-Bit CMOS Microcomputer .... 5—3
M58496-XXXP Single-Chip 4-Bit CMOS Microcomputer .... 5—17
M58497-XXXP Single-Chip 4-Bit CMOS Microcomputer .... 5—32

6 MELPS 8-48 MICROCOMPUTERS

MELPS 8-48 Microcomputers Function of MELPS 8-48 Microcomputers .... 6—3
M5L8048-XXXP, M5L8035LP Single-Chip 8-Bit Microcomputer .... 6—21
M5L8049-XXXP, P-8, P-6, M5L8039P-11, P-8, P-6 Single-Chip 8-Bit Microcomputer .... 6—25
M5L8748S Single-Chip 8-Bit Microcomputer with EPROM .... 6—29
M5L8243P Input/Output Expander .... 6—37

CONTENTS

Page

7 MELPS 8/85 MICROPROCESSORS

M5L8085AP, S 8-Bit Parallel Microprocessor .... 7—3
M5L8212P 8-Bit Input/Output Port with 3-State Output .... 7—17
M5L8216P, M5L8226P 4-Bit Parallel Bidirectional Bus Drivers .... 7—21
M5L8155P 2048-Bit Static RAM with I/O Ports and Timer .... 7—25
M5L8156P 2048-Bit Static RAM with I/O Ports and Timer .... 7—33

8 LSIs FOR PERIPHERAL CIRCUITS

M58990P 8-Bit 8-Channel A-D Converter .... 8—3
M5C6847P-1 Video Display Generator .... 8—7
M5L8041A-XXXP Universal Peripheral Interface .... 8—17
M5L8251AP Programmable Communication Interface .... 8—41
M5L8253P-5 Programmable Interval Timer .... 8—57
M5L8255AP-5 Programmable Peripheral Interface .... 8—65
M5L8257P-5 Programmable DMA Controller .... 8—81
M5L8259AP Programmable Interrupt Controller .... 8—91
M5L8279P-5 Programmable Keyboard/Display Interface .... 8—105
M5W1791-02P Floppy Disk Formatter/Controller .... 8—117

9 MELPS 86 MICROPROCESSORS

M5L8086S 16-Bit Parallel Microprocessor .... 9—3
M5L8282P, M5L8283P Octal Latch .... 9—35
M5L8284P Clock Generator and Driver for 8086, 8088, 8089 Processors .... 9—39
M5L8286P, M5L8287P Octal Bus Transceiver .... 9—46
M5L8288P Bus Controller for 8086, 8088, 8089 Processors .... 9—50

10 SPEECH SYNTHESIS LSIs (PARCOR SYSTEM)

M58817AP Speech Synthesizer .... 10—3
M58818-XXXP 128K-Bit Phrase ROM .... 10—13
M58819S EPROM Interface .... 10—19

11 GENERAL-PURPOSE MOS LSIs

M50110XP, M50115XP 30~ OR 120-Function Remote-Control Transmitters .... 11—3
M50111XP, M50116XP, M50117XP 30~120-Function Remote-Control Receiver .... 11—9
M50250P Refrigerator Controller .... 11—15
M50401P, M50402P, M50403P, M50404P, M50405P CMOS Analog Clock Circuits .... 11—19
M58412P, M58413P CMOS LCD Digital Alarm Clock Circuits .... 11—23
M58435P, M58437-001P CMOS Analog Clock Circuits .... 11—31
M58478P, M50121P, M50122P 17-Stage Oscillator/Divider .... 11—35
M58479P, M58482P CMOS Counter/Timers .... 11—39
M58480P, M58484P 30-Function Remote-Control Transmitters .... 11—43
M58481P 30-Function Remote-Control Receiver .... 11—47
M58485P 29-Function Remote-Control Receiver .... 11—51
M58486AP Voltage Synthesizer .... 11—55
M58487AP 24-Function Remote-Control Receiver .... 11—65

12 MICROCOMPUTER SYSTEMS

PCA8501G01, G02 MELCS 85/2 Single-Board Computer .... 12—3
PCA8506 MELCS 85/2 Memory and Parallel I/O Expansion Board .... 12—7
PCA8507 MELCS 85/2 Memory and Serial I/O Expansion Board .... 12—11
PCA8520G01, G02 MELCS 85/3 Voice Generating Single-Board Computer .... 12—15
PCA8540G01, G02 MELCS 85/2 Color TV Display Single-Board Computer .... 12—19
PCA7002G01, G02 MELCS 70/2 Speech Synthesizer Single-Board Computer .... 12—25

CONTENTS

Page

13 MICROCOMPUTER SUPPORT SYSTEMS

PCA0803 MELCS 8/2 Program Checker .... 13—3
PC4000 Debugging Machine .... 13—5
PC7000 Speech Synthesis Evaluation Unit .... 13—9
PC8500, PCA8503 MELCS 85/1 Portable Microcomputer Console .... 13—11
PC9000 Cross Assembler Machine .... 13—17
PCA4001 MELPS 4 Dedicated Board .... 13—20
PCA4003 MELPS 4 Dedicated Board .... 13—22
PCA4004 MELPS 4 Dedicated Board .... 13—24
PCA4005 MELPS 4 Dedicated Board .... 13—26
PCA4011 MELPS 41 Dedicated Board .... 13—28
PCA4012 MELPS 42 Dedicated Board .... 13—30
PCA4014 MELPS 42 Dedicated Board .... 13—32
PCA8400 MELPS 8-48 Dedicated Board .... 13—34
PC4100 M5L8748S Programming Adaptor .... 13—36
PCA4301 MELPS 4 Evaluation Board .... 13—37
PCA4303 MELPS 4 Evaluation Board .... 13—38
PCA4304 MELPS 4 Evaluation Board .... 13—39
PCA4305 MELPS 4 Evaluation Board .... 13—40
PCA4101 MELPS 41 Evaluation Board .... 13—42
PCA4201 MELPS 42 Evaluation Board .... 13—44
PCA4202 MELPS 42 Evaluation Board .... 13—46
PCA8402 MELPS 8-48 Evaluation Board .... 13—48

14 MICROCOMPUTER SOFTWARE

Software Codes .... 14—3
MELPS 4/41 Software Available Materials .... 14—4
MELPS 4/41 Software General Description .... 14—6
MELPS 4/41 Software Development of Application Programs .... 14—7
MELPS 8/85 Software Available Materials .... 14—8
MELPS 8/85 Software General Description .... 14—9
MELPS 8/85 Software Development of Application Programs .... 14—10
MELPS 4 Software Cross Assembler .... 14—11
MELPS 4 Software Simulator .... 14—15
MELPS 4 Software Paper-Tape Generation Program for PROM Writers .... 14—19
MELPS 41 Software Cross Assembler .... 14—21
MELPS 41 Software Simulator .... 14—25
MELPS 41 Software Paper-Tape Generation Program for PROM Writers .... 14—29
MELPS 42 Software Cross Assembler .... 14—31
MELPS 42 Software Paper-Tape Generation Program for PROM Writers .... 14—35
MELPS 8-48 Software Cross Assembler .... 14—37
MELPS 8-48 Software Paper-Tape Generation Program for PROM Writers .... 14—41
MELPS 8/85 Software PL/1μ Cross Compiler .... 14—43
MELPS 8/85 Software Cross Assembler .... 14—47
MELPS 8/85 Software Simulator .... 14—51
MELPS 8/85 Software Paper-Tape Generation Program for PROM Writers .... 14—55
MELPS 8/85 Software Self Assembler .... 14—57
MELPS Software Editor .... 14—61
MELPS 8 BOM-PTS Basic Operating Monitor—Paper-Tape System .... 14—63
MELPS 8 BOM-B Basic Operating Monitor—Basic System .... 14—65

CONTENTS

15 APPLICATIONS

Page

Memory Development Approaches ...... 15—3
16K-Bit Dynamic RAM (M5K4116P, S) ...... 15—9
64K-Bit Dynamic RAM (M5K4164S, M5K4164NS) ...... 15—21
Static RAM (M58725P, M5L2114LP) ...... 15—51
CMOS Static RAM (M58981P, M5L5101LP-1) ...... 15—56
EPROM (M5L2716K, M5L2732K, M5L2764K) ...... 15—67
Error Detecting and Correcting ...... 15—79
MELPS 4 Program Library Subroutines ...... 15—86
Application of MELPS 4 Single-Chip 4-Bit Microcomputer (M58840-×××P) in a Microwave Oven ...... 15—98
MELPS 8/85 Program Library Subroutines ...... 15—102
Application of MELCS 8/2 Single-Board Computer (PCA0801) in Data Transmission through a Master-Slave Multicomputer System ...... 15—106

Contact Address for Further Information

INDEXES

1

INDEX BY FUNCTION

■Static RAMs

Type	Circuit function and organization	Structure (Note 1)	Supply voltage (V)	Electrical characteristics: Typ pwr dissipasion (mW)	Max. access time (ns)	Min. cycle time (ns)	Max. frequency (MHz)	Package (Note 2)	Interchangeable products	Page
M5L2114LP-2	4096-Bit (1024×4) Static RAM	N, Si, ED	5±10%	300	200	200	–	18P4	i 2114L-2 TMS4045-20	2—85
M5L2114LP-3				250	300	300	–		i 2114L-3 TMS4045-30	2—85
M5L2114LP				200	450	450	–		i 2114L TMS4045-45	2—85
M5T4044P-20	4096-Bit (4096×1) Static RAM	N, Si, ED	5±10%	300	200	200	–	18P4	TMS4044-20	2—93
M5T4044P-30				250	300	300	–		TMS4044-30	2—93
M5T4044P-45				200	450	450	–		TMS4044-45	2—93
M58725P-15	16384-Bit (2048×8) Static RAM	N, Si, ED	5±10%	200	150	150	–	24P1	TMS4016-15	2—3
M58725P				200	200	200	–		TMS4016	2—3

■Dynamic RAMs

Type	Circuit function and organization	Structure (Note 1)	Supply voltage (V)	Electrical characteristics: Typ pwr dissipasion (mW)	Max. access time (ns)	Min. cycle time (ns)	Max. frequency (MHz)	Package (Note 2)	Interchangeable products	Page
M5K4116P-2	16384-Bit (16384×1) Dynamic RAM	N, Si	12±10% 5±10% −4.5~−5.7	280	150	375	–	16P4	MK4116-2	2—13
M5K4116P-3				280	200	375	–		MK4116-3	2—13
M5K4164P-15	65536-Bit (65536×1) Dynamic RAM Pin 1 (RFE) function	N, Si	5±10%	200	150	260	–	16P4	—	2—25
M5K4164P-20				170	200	330	–		—	2—25
M5K4164NP-15	65536-Bit (65536×1) Dynamic RAM Pin 1 no connection	N, Si	5±10%	200	150	260	–	16P4	—	2—41
M5K4164NP-20				170	200	330	–		—	2—41
M5K4164S-15	65536-Bit (65536×1) Dynamic RAM Pin 1 (REF) function	N, Si	5±10%	200	150	260	–	16S1	MK4164	2—55
M5K4164S-20				170	200	330	–		MCM6664	2—55
M5K4164NS-15	65536-Bit (65536×1) Dynamic RAM Pin 1 no connection	N, Si	5±10%	200	150	260	–	16S1	i 2164	2—71
M5K4164NS-20				170	200	330	–		MCM6665	2—71

■CMOS Static RAMs

Type	Circuit function and organization	Structure (Note 1)	Supply voltage (V)	Electrical characteristics: Typ pwr dissipasion (mW)	Max. access time (ns)	Min. cycle time (ns)	Max. frequency (MHz)	Package (Note 2)	Interchangeable products	Page
M5L5101LP-1	1024-Bit (256×4) CMOS Static RAM	C, Si	5±10%	75	450	450	–	22P1	i 5101L-1	2—89
M58981P-30	4096-Bit (1024×4) CMOS Static RAM	C, Si	5±10%	75	300	300	–	18P4	—	2—9
M58981P-45				75	450	450	–		—	2—9

■Mask ROM

Type	Circuit function and organization	Structure (Note 1)	Supply voltage (V)	Electrical characteristics: Typ pwr dissipasion (mW)	Max. access time (ns)	Min. cycle time (ns)	Max. frequency (MHz)	Package (Note 2)	Interchangeable products	Page
M58735-XXXP	32768-Bit (4096×8) Mask Programmable ROM	N, Si	5±10%	300	450	–	–	24P1	—	3—22

■Field-Programmable ROMs

Type	Circuit function and organization	Structure (Note 1)	Supply voltage (V)	Electrical characteristics: Typ pwr dissipasion (mW)	Max. access time (ns)	Min. cycle time (ns)	Max. frequency (MHz)	Package (Note 2)	Interchangeable products	Page
M58653P	700-Bit (50×14) Electrically Alterable ROM	P, Al	5±5%	200	20μs	–	16.8kHz	14P4	—	3—18
M5G1400P	1400-Bit (100×14) Electrically Alterable ROM	P, Al	5±5%	200	20μs	–	16.8kHz	14P4	GI 1400	3—24
M5L2716K	16384-Bit (2048×8) Erasable and Electrically Reprogrammable ROM	N, Si, FA	5±5%	300	450	–	–	24K10	i 2716	3—28
M5L2716K-65				300	650	–	–		i 2716-6	3—28
M5L2732K	32768-Bit (4096×8) Erasable and Electrically Reprogrammable ROM	N, Si, FA	5±5%	400	450	–	–	24K10	i 2732	3—32
M5L2732K-6				400	550	–	–		i 2732-6	3—32
M5L2764K-2	65536-Bit (8192×8) Erasable and Electrically Reprogrammable ROM	N, Si, FA	5±5%	500	200	–	–	28K10	i 2764-2	3—36
M5L2764K					250	–	–		i 2764	3—36
M5L2764K-3					300	–	–		i 2764-3	3—36
M54700P, S	1024-Bit (256×4) Field-Programmable ROM with Open-Collector	B	5±5%	430	60	60	–	16P4 16S1	MM6300	3—5
M54730P, S	256-Bit (32×8) Field-Programmable ROM with Open-Collector	B	5±5%	430	50	60	–	16P4 16S1	MM6330	3—10
M54740AP, S	4096-Bit (1024×4) Field-Programmable ROM with Open-Collector	B, S	5±5%	600	55	55	–	18P4 18S1	93452	3—14
M54741AP, S	4096-Bit (1024×4) Field-Programmable ROM with 3-State Outputs	B, S	5±5%	600	55	55	–	18P4 18S1	93453	3—14

Type	Circuit function and organization	Structure (Note 1)	Supply voltage (V)	Electrical characteristics: Typ pwr dissipation (mW)	Max. access time (ns)	Min. cycle time (ns)	Max. frequency (MHz)	Package (Note 2)	Interchangeable products	Page

■ Single-Chip Microcomputers

M58840-XXXP	Single-Chip 4-Bit Microcomputer with 8-Bit A/D Converter	P, Al, ED	−15±10%	500	−	10μs	0.6	42P1	−	**4—2**
M58841-XXXSP	Single-Chip 4-Bit Microcomputer with 8-Bit A/D Converter	P, Al, ED	−15±10%	500	−	10μs	0.6	42P4B	−	**4—2**
M58842S	MELPS 4 System Evaluation Device	P, Al, ED	−15±10%	500	−	10μs	0.6	64S1	−	**4—13**
M58843-XXXP	Single-Chip 4-Bit Microcomputer with 8-Bit A/D Converter	P, Al, ED	−15±10%	400	−	10μs	0.6	28P4	−	**4—18**
M58844-XXXSP	Single-Chip 4-Bit Microcomputer with 8-Bit A/D Converter	P, Al, ED	−15±10%	400	−	10μs	0.6	40P4B	−	**4—18**
M58845-XXXSP	Single-Chip 4-Bit Microcomputer with 8-Bit A/D Converter	P, Al, ED	−15±10%	350	−	10μs	0.6	40P4B	−	**4—29**
M58846-XXXSP	Single-Chip 4-Bit Microcomputer	P, Al, ED	−12±10%	280	−	10μs	0.6	40P4B	−	**4—41**
M58847-XXXSP	Single-Chip 4-Bit Microcomputer	P, Al, ED	−12±10%	10	−	15μs	0.4	40P4B	−	**4—53**
M58494-XXXP	Single-Chip 4-Bit CMOS Microcomputer	C, Al	5± 5%	5	−	8.8μs	0.455	72P2	−	**5—3**
M58496-XXXP	Single-Chip 4-Bit CMOS Microcomputer	C, Al	5± 5%	5	−	7.7μs	4.2	72P2	−	**5—17**
M58497-XXXP	Single-Chip 4-Bit CMOS Microcomputer	C, Al	3~5.5%	2	−	15.4μs	0.455	72P2	−	**5—32**
M5L8048-XXXP	Single-Chip 8-Bit Microcomputer	N, Si, ED	5±10%	325	−	2500	6	40P1	i 8048	**6—21**
M5L8035LP	Single-Chip 8-Bit Microcomputer	N, Si, ED	5±10%	325	−	2500	6	40P1	i 8035L	**6—21**
M5L8049-XXXP	Single-Chip 8-Bit Microcomputer	N, Si, ED	5±10%	500	−	1360	11	40P1	i 8049	6—25
M5L8049-XXXP-8				500	−	1875	8		−	
M5L8049-XXXP-6				500	−	2500	6		−	
M5L8039P-11	Single-Chip 8-Bit Microcomputer	N, Si, ED	5±10%	500		1360	11	40P1	i 8039	6—25
M5L8039P-8				500	−	1875	8		−	
M5L8039P-6				500		2500	6		i 8039-6	
M5L8748S	Single-Chip 8-Bit Microcomputer with EPROM	N, Si, ED	5±10%	500	−	2500	6	40S10	i 8748	6—29

■ Microprocessors

M5L8085AP, S	8-Bit Parallel Microprocessor	N, Si, ED	5±5%	600	−	−	3	40P1 40S1	i 8085A	7—3
M5L8086S	16-Bit Parallel Microprocessor	N, Si, ED	5±10%	1375	−	−	5	40S1	i 8086	9—3

■ LSIs for Peripheral Circuits

M58990P	8-Bit 8-Channel A-D Converter	C, Si	5±10%	−	−	−	−	28P4	ADC0808	**8—3**
M5C6847P-1	Video Display Generator	N, Si, ED	5±5%	500	−	−	3.58	40P1	MC6847-1	**8—7**
M5L8041A-XXXP	Universal Peripheral Interface	N, Si, ED	5±10%	300	−	−	6	40P1	i 8041A	**8—17**
M5L8155P	2048-Bit Static RAM with I/O Ports and Timer ($\overline{CE}$="L"active)	N, Si, ED	5±5%	500	−	−	−	40P1	i 8155	**7—25**
M5L8156P	2048-Bit Static RAM with I/O Ports and Timer (CE="H"active)	N, Si, ED	5±5%	500	−	−	−	40P1	i 8156	**7—33**
M5L8212P	8-Bit Input/Output Port	B, S	5±5%	450	35☆	−	−	24P1	i 8212	**7—17**
M5L8216P	4-Bit Parallel Bidirectional Bus Driver (Non Inverting)	B, S	5±5%	475	25☆	−	−	16P4	i 8216	**7—21**
M5L8226P	4-Bit Parallel Bidirectional Bus Driver (Inverting)	B, S	5±5%	425	25☆	−	−	16P4	i 8226	**7—21**
M5L8243P	Input/Output Expander	N, Si, ED	5±10%	50	−	−	−	24P1	i 8243	**6—37**
M5L8251AP	Programmable Communication Interface	N, Si, ED	5±5%	300	−	−	3	28P4	i 8251A	**8—41**
M5L8253P-5	Programmable Interval Timer	N, Si, ED	5±5%	300	−	−	2	24P1	i 8253-5	**8—57**
M5L8255AP-5	Programmable Peripheral Interface	N, Si, ED	5±5%	250	−	−	−	40P1	i 8255A-5	**8—65**

Type	Circuit function and organization	Structure (Note 1)	Supply voltage (V)	Electrical characteristics: Typ pwr dissi-pasion (mW)	Max. access time (ns)	Min cycle time (ns)	Max. fre-quency (MHz)	Package (Note 2)	Interchangeable products	Page
■LSIs for Peripheral Circuits (Continued)										
M5L8257P-5	Programmable DMA Controller	N, Si, ED	5±5%	300	–	–	3	40P1	i 8257-5	8—81
M5L8259AP	Programmable Interrupt Controller	N, Si, ED	5±10%	275	–	–	–	28P4	i 8259A	8—91
M5L8279P-5	Programmable Keyboard/Display Interface	N, Si, ED	5±10%	650	–	–	3	40P1	i 8279-5	8—105
M5L8282P	8-Bit Latch (Non Inverting)	B, S	5±10%	500	–	–	–	20P4	i 8282	9—35
M5L8283P	8-Bit Latch (Inverting)	B, S	5±10%	500	–	–	–	20P4	i 8283	9—35
M5L8284P	Clock Generator and Driver for M5L8086S CPU	B, S	5±10%	490	–	–	–	18P4	i 8284	9—39
M5L8286P	Octal Bus Transceiver (Non Inverting)	B, S	5±10%	560	–	–	–	20P4	i 8286	9—46
M5L8287P	Octal Bus Transceiver (Inverting)	B, S	5±10%	90	–	–	–	20P4	i 8287	9—46
M5L8288P	Bus Controller for M5L8086S CPU	B, S	5±10%	800	–	–	–	20P4	i 8288	9—50
M5W1791-02	Floppy Disk Formatter/Controller	N, Si, ED	5±5%	300	–	–	–	40P1	FD1791-02B	8—117
■Speech Synthesis (PARCOR SYSTEM)										
M58817AP	Speech Synthesizer	P, Al, ED	−10±10%	300	–	–	0.66	28P4	–	10—3
M58818-XXXP	128K-Bit Phrase ROM	P, Al, ED	−10±10%	80	–	–	0.17	24P1	–	10—13
M58819S	EPROM Interface	P, Al, ED	−10±10% −5±5%	150	–	–	0.17	40S1	–	10—19
■LSIs for Remote-Control Receiver and Transmitter										
M50110XP	30-Function Remote-Control Transmitter	C, Al	2.2~8	–	–	–	–	16P4	–	11—3
M50115XP	120-Function Remote-Control Transmitter	C, Al	2.2~8	–	–	–	–	18P4	–	11—3
M50111XP	120-Function Remote-Control Receiver	C, Al	4.5~8	–	–	–	–	16P4	–	11—9
M50116XP	120-Function Remote-Control Receiver	C, Al	4.5~8	–	–	–	–	18P4	–	11—9
M50117XP	120-Function Remote-Control Receiver	C, Al	4.5~8	–	–	–	–	18P4	–	11—9
M58480P	30-Function Remote-Control Transmitter	C, Al	2.2~8	–	–	–	–	16P4	–	11—43
M58484P	30-Function Remote-Control Transmitter	C, Al	2.2~8	–	–	–	–	16P4	–	11—43
M58481P	30-Function Remote-Control Receiver	C, Al	4.5~8	–	–	–	–	28P4	–	11—47
M58485P	29-Function Remote-Control Receiver	C, Al	8~14	–	–	–	–	28P4	–	11—51
M58487AP	24-Function Remote-Control Receiver	C, Al	8~14	–	–	–	–	28P4	–	11—65
■LSIs for Clock Circuits										
M50401P	CMOS Analog Clock Circuit	C, Si	1.1~1.8	–	–	–	–	8P4	–	11—19
M50402P	CMOS Analog Clock Circuit	C, Si	1.1~1.8	–	–	–	–	8P4	–	11—19
M50403P	CMOS Analog Clock Circuit	C, Si	1.1~1.8	–	–	–	–	8P4	–	11—19
M50404P	CMOS Analog Clock Circuit	C, Si	1.1~1.8	–	–	–	–	8P4	–	11—19
M50405P	CMOS Analog Clock Circuit	C, Si	1.1~1.8	–	–	–	–	8P4	–	11—19
M58412P	CMOS LCD Digital Alarm Clock Circuit	C, Al	−1.2 ~−1.9	–	–	–	–	60P2	–	11—23
M58413P	CMOS LCD Digital Alarm Clock Circuit	C, Al	−1.1 ~−2	–	–	–	–	60P2	–	11—23
M58435P	CMOS Analog Clock Circuit	C, Si	1.2~1.9	–	–	–	–	8P4	–	11—31
M58437-001P	CMOS Analog Clock Circuit	C, Al	1.1~1.9	–	–	–	–	8P4	–	11—31

1

Type	Circuit function and organization	Structure (Note 1)	Supply voltage (V)	Electrical characteristics: Typ pwr dissi-pasion (mW)	Max. access time (ns)	Min. cycle time (ns)	Max. fre-quency (MHz)	Package (Note 2)	Interchangeable products	Page

■General-Purpose MOS LSIs

M50121P	17-Stage Oscillator/Divider	C, Al	4.75~8.5	–	–	–	–	8P4	–	**11—35**
M50122P	17-Stage Oscillator/Divider	C, Al	4.75~8.5	–	–	–	–	8P4	–	**11—35**
M50250P	Refrigerator Controller	C, Al	7~9	–	–	–	–	16P4	–	**11—15**
M58478P	17-Stage Oscillator/Divider	C, Al	4.75~8.5	–	–	–	–	8P4	–	**11—35**
M58479P	CMOS Counter/Timer	C, Al	7.9~9	–	–	–	–	14P4	–	**11—39**
M58482P	CMOS Counter/Timer	C, Al	3~9	–	–	–	–	14P4	–	**11—39**
M58486AP	Voltage Synthesizer	C, Al	11~13	–	–	–	–	42P1	–	**11—55**

Note 1: Al=Aluminum gate B=Bipolar. C=CMOS. ED=Enhancement depletion mode. FA=FAMOS.
N=N-channel. P=P-channel. S=Schottkey. Si=Silicon gate.

2: Package code 24 S 1
- 24 — Number of pins
- S — Package structure: K=Glass-sealed ceramic: P=Molded plastic: S=Metal-sealed ceramic
- 1 — Package outline
  1=DIL without fin. 2=Flat without fin.
  4=DIL without fin(improved): 10=DIL w/o fin. and w/quartz lid
  4B=Shrink DIL without fin

3: ☆ Indicates propagation time

Type	Circuit function and organization	Memory capacity RAM (bytes)	Memory capacity ROM (bytes)	I/O port (bits)	Ambient operating temp Ta (°C)	Supply voltage (V)	Dimensions (lxwxh) (mm)	Page

■Micromputer Systems

Type	Circuit function and organization	RAM (bytes)	ROM (bytes)	I/O port (bits)	Ambient operating temp Ta (°C)	Supply voltage (V)	Dimensions (lxwxh) (mm)	Page
PCA8501 G01 PCA8501 G02	MELCS 85/2 Single-Board Computer	1K	4K	48	0~55	5	125×145×17	**12—3**
PCA8506	MELCS 85/2 Memory and Parallel I/O Expansion Board	12K		48	0~55	5	125×145×17	**12—7**
PCA8507	MELCS 85/2 Memory and Serial I/O Expansion Board	12K		1 (serial)	0~55	12, 5, −12	125×145×17	**12—11**
PCA8540 G01 PCA8540 G02	MELCS 82/2 Video Display Single-Board Computer	256	4K	22	5~40	5, −5	125×145×20	**12—19**

■Speech Synthesize Single-Board Computers

Type	Circuit function and organization	RAM (bytes)	ROM (bytes)	I/O port (bits)	Ambient operating temp Ta (°C)	Supply voltage (V)	Dimensions (lxwxh) (mm)	Page
PCA7002 G01 PCA7002 G02	MELCS 70/2 Speech Synthesizer Board	−	8K or 16K	−	0~55	5, −5	125×145×25	**12—25**
PCA8520 G01 PCA8520 G02	MELCS 85/3 Voice Generating Single-Board Computer	256	16K	24	0~55	5, −5	125×145×20	**12—15**

■Microcomputer Support Systems

Type	Circuit function and organization	RAM (bytes)	ROM (bytes)	I/O port (bits)	Ambient operating temp Ta (°C)	Supply voltage (V)	Dimensions (lxwxh) (mm)	Page
PCA0803	MELCS 8/2 Program Checker	−	−	−	0~55	5	170×200×27	**13—3**
PC4000	Debugging Machine	−	−	−	10~40	AC100	364×257×85	**13—5**
PC7000	Speech Synthesis Evaluation Unit	−	−	−	10~40	AC100	390×212×73	**13—9**
PC8500	MELCS 85/1 Portable Microcomputer Console	−	−	−	10~40	AC100	350×370×140	**13—11**
PC9000	Cross Assemble Machine	−	−	−	10~40	AC100	500×470×287	**13—17**

■Dedicated Board

Type	Circuit function and organization	RAM (bytes)	ROM (bytes)	I/O port (bits)	Ambient operating temp Ta (°C)	Supply voltage (V)	Dimensions (lxwxh) (mm)	Page
PCA4001	Emulator Boad for M58840, M58841	−	−	−	10~40	Supplied from PC4000	210×230×20	**13—20**
PCA4003	Emulator Board for M58843	−	−	−	10~40			**13—22**
PCA4004	Emulator Board for M58844	−	−	−	10~40			**13—24**
PCA4005	Emulator Board for M58845	−	−	−	10~40			**13—26**
PCA4011	Emulator Board for M58494	−	−	−	10~40			**13—28**
PCA4012	Emulator Board for M58496	−	−	−	10~40			**13—30**
PCA4014	Emulator Board for M58497	−	−	−	10~40			**13—32**
PCA8400	Emulator Board for MELPS 8-48	−	−	−	10~40			**13—34**
PC4100	M5L8748S Programming Adaptor	−	−	−	10~40		165×105×37	**13—36**

■Evaluation Board

Type	Circuit function and organization	RAM (bytes)	ROM (bytes)	I/O port (bits)	Ambient operating temp Ta (°C)	Supply voltage (V)	Dimensions (lxwxh) (mm)	Page
PCA4301	Evaluation Board for M58840, M58841	−	−	−	0~55	−15	125×110×20	**13—37**
PCA4303	Evaluation Board for M58843	−	−	−	0~55	−15	125×110×20	**13—38**
PCA4304	Evaluation Board for M58844	−	−	−	0~55	−15	125×110×20	**13—39**
PCA4305	Evaluation Board for M58845	−	−	−	0~55	−15	210×230×20	**13—40**
PCA4101	Evaluation Board for M58494	−	−	−	0~55	5	150×200×20	**13—42**
PCA4201	Evaluation Board for M58496	−	−	−	0~55	5	150×200×20	**13—44**
PCA4202	Evaluation Board for M58497	−	−	−	0~55	5	150×200×20	**13—46**
PCA8402	Evaluation Board for MELPS 8-48	−	−	−	0~55	5	150× 58 ×27	**13—48**

INDEX BY TYPE DESIGNATION

Type	Structure	Function	Circuit function	Page
M50110XP	C, Al	Remo-con	30-function remote-control transmitter	11—3
M50111XP	C, Al	Remo-con	120-function remote-control receiver	11—9
M50115XP	C, Al	Remo-con	120-function remote-control transmitter	11—3
M50116XP	C, Al	Remo-con	120-function remote-control receiver	11—9
M50117XP	C, Al	Remo-con	120-function remote-control receiver	11—9
M50121P	C, Al	Counter	17-stage oscillator/divider	11—35
M50122P	C, Al	Counter	17-stage oscillator/divider	11—35
M50250P	C, Al	Counter	Refrigerator counter	11—15
M50401P	C, Al	Clock	CMOS analog clock circuit	11—19
M50402P	C, Al	Clock	CMOS analog clock circuit	11—19
M50403P	C, Al	Clock	CMOS analog clock circuit	11—19
M50404P	C, Al	Clock	CMOS analog clock circuit	11—19
M50405P	C, Al	Clock	CMOS analog clock circuit	11—19
M54700P **M54700S**	B	PROM	1024-bit (256-word×4-bit) field-programmable ROM with open-collector	3—5
M54730P **M54730S**	B	PROM	256-bit (32-word×8-bit) field-programmable ROM with open-collector	3—10
M54740AP **M54740AS**	B, S	PROM	4096-bit (1024-word×4-bit) field-programmable ROM with open-collector	3—14
M54741AP **M54741AS**	B, S	PROM	4096-bit (1024-word×4-bit) field-programmable ROM with 3-state	3—14
M58412P	C, Al	Clock	CMOS LCD degital alarm clock circuit	11—23
M58413P	C, Al	Clock	CMOS LCD digital alarm clock circuit	11—23
M58435P	C, Si	Clock	CMOS analog clock circuit	11—31
M58437-001P	C, Al	Clock	CMOS analog clock circuit	11—31
M58478P	C, Al	Counter	17-stage oscillator/divider	11—35
M58479P	C, Al	Counter	CMOS counter/timer	11—39
M58480P	C, Al	Remo-con	30-function remote-control transmitter	11—43
M58481P	C, Al	Remo-con	30-function remote-control receiver	11—47
M58482P	C, Al	Counter	CMOS counter/timer	11—39
M58484P	C, Al	Remo-con	30-function remote-control transmitter	11—43
M58485P	C, Al	Remo-con	29-function remote-control receiver	11—51
M58486AP	C, Al	Counter	Voltage synthesizer	11—55
M58487AP	C, Al	Remo-con	22-function remote-control receiver	11—65
M58494-XXXP	C, Al	CPU	Single-chip 4-bit CMOS microcomputer	5—3
M58496-XXXP	C, Al	CPU	Single-chip 4-bit CMOS microcomputer	5—17
M58497-XXXP	C, Al	CPU	Single-chip 4-bit CMOS microcomputer	5—32
M58653P	P, Al	EEPROM	700-bit (50-word by 14-bit) electrically alterable ROM	3—18
M58725P **M58725P-15**	N, Si ED	RAM	16384-bit (2048-word×8-bit) static RAM	2—3
M58735-XXXP	N, Si	ROM	32768-bit (4096-word×8-bit) mask-programmable ROM	3—22
M58817AP	P, Al, ED	Speech	Speech synthesizer	10—3
M58818-XXXP	P, Al, ED	Speech	131072-bit phrase ROM	10—13
M58819S	P, Al, ED	Speech	EPROM interface	10—19
M58840-XXXP	P, Al, ED	CPU	Single-chip 4-bit microcomputer with 8-bit A/D converter	4—2
M58841-XXXSP	P, Al, ED	CPU	Single-chip 4-bit microcomputer with 8-bit A/D converter	4—2
M58842S	P, Al, ED	CPU	MELPS 4-system evaluation device	4—13
M58843-XXXP	P, Al, ED	CPU	Single-chip 4-bit microcomputer with 8-bit A/D converter	4—18
M58844-XXXSP	P, Al, ED	CPU	Single-chip 4-bit microcomputer with 8-bit A/D converter	4—18
M58845-XXXSP	P, Al, ED	CPU	Single-chip 4-bit microcomputer with 8-bit A/D converter	4—29
M58846-XXXSP	P, Al, ED	CPU	Single-chip 4-bit microcomputer	4—41
M58847-XXXSP	P, Al, ED	CPU	Single-chip 4-bit microcomputer	4—53

INDEX BY TYPE DESIGNATION

Type	Structure	Function	Circuit function	Page
M58981P-30	C, Si	RAM	4096-bit (1024-word×4-bit) CMOS static RAM	2—9
M58981P-45				
M58990P	C, Si	I/O	8-bit 8-channel A/D converter	8—3
M5C6847P-1	N, Si, ED	I/O	Video display generator	8—7
M5G1400P	P, Al	EEPROM	1400-bit (100-word×14-bit) electrically alterable ROM	3—24
M5K4116P-2	N, Si	RAM	16384-bit (16384-word×1-bit) dynamic RAM	2—13
M5K4116P-3				
M5K4164P-15	N, Si	RAM	65536-bit (65536-word×1-bit) dynamic RAM Pin 1 ($\overline{RFE}$) function	2—25
M5K4164P-20				
M5K4164NP-15	N, Si	RAM	65536-bit (65536-word×1-bit) dynamic RAM Pin 1 no connection	2—41
M5K4164NP-20				
M5K4164S-15	N, Si	RAM	65536-bit (65536-word×1-bit) dynamic RAM Pin 1 ($\overline{REF}$) function	2—55
M5K4164S-20				
M5K4164NS-15	N, Si	RAM	65536-bit (65536-word×1-bit) dynamic RAM Pin 1 no connection	2—71
M5K4164NS-20				
M5L2114LP	N, Si, ED	RAM	4096-bit (1024-word×4-bit) static RAM	2—85
M5L2114LP-2				
M5L2114LP-3				
M5L2716K	N, Si, FA	EPROM	16384-bit (2048-word×8-bit) erasable and electrically reprogrammable ROM	3—28
M5L2716K-65				
M5L2732K	N, Si, FA	EPROM	32768-bit (4096-word×8-bit) erasable and electrically reprogrammable ROM	3—32
M5L2732K-6				
M5L2764K	N, Si, FA	EPROM	65536-bit (8192-word×8-bit) erasable and electrically reprogrammable ROM	3—36
M5L2764K-2				
M5L2764K-3				
M5L8039P-6	N, Si, ED	CPU	Single-chip 8-bit microcomputer	6—25
M5L8039P-8				
M5L8039P-11				
M5L8041A-XXXP	N, Si, ED	I/O	Universal peripheral interface	8—17
M5L8048-XXXP	N, Si, ED	CPU	Single-chip 8-bit microcomputer	6—21
M5L8049-XXXP	N, Si, ED	CPU	Single-chip 8-bit microcomputer	6—25
M5L8049-XXXP-6				
M5L8049-XXXP-8				
M5L8085AP	N, Si, ED	CPU	8-bit parallel CPU	7—3
M5L8085AS				
M5L8086S	N, Si, ED	CPU	16-bit parallel microprocessor	9—3
M5L8155P	N, Si, ED	I/O	2048-bit static RAM with I/O ports and timer (CE=low active)	7—25
M5L8156P	N, Si, ED	I/O	2048-bit static RAM with I/O ports and timer (CE=high active)	7—33
M5L8212P	B, S	I/O	8-bit input/output port	7—17
M5L8216P	B, S	I/O	4-bit parallel bidirectional bus driver (non invert outputs)	7—21
M5L8226P	B, S	I/O	4-bit parallel bidirectional bus driver (invert outputs)	7—21
M5L8243P	N, Si, ED	I/O	Input/output expander	6—37
M5L8251AP	N, Si, ED	I/O	Programmable communication interface	8—41
M5L8253P-5	N, Si, ED	I/O	Programmable interval timer	8—57
M5L8255AP-5	N, Si, ED	I/O	Programmable peripheral interface	8—65
M5L8257P-5	N, Si, ED	I/O	Programmable DMA controller	8—81
M5L8259AP	N, Si, ED	I/O	Programmable interrupt controller	8—91
M5L8279P-5	N, Si, ED	I/O	Programmable keyboard/display interface	8—105
M5L8282P	B, S	I/O	8-bit latch (non inverting)	9—35
M5L8283P	B, S	I/O	8-bit latch (inverting)	9—35
M5L8284P	B, S	I/O	Clock generator and driver for 8086, 8088, 8089 processors	9—39
M5L8286P	B, S	I/O	Octal bus transceiver (non inverting)	9—46

INDEX BY TYPE DESIGNATION

Type	Structure	Function	Circuit function	Page
M5L8287P	B, S	I/O	Octal bus transceiver (inverting)	9—46
M5L8288P	B, S	I/O	Bus controller for 8086, 8088, 8089 processors	9—50
M5L8748S	N, Si, ED	I/O	Single-chip 8-bit microcomputer with EPROM	6—29
M5T4044P-20	N, Si, ED	RAM	4096-bit (4096-word×1-bit) static RAM	2—93
M5T4044P-30				
M5T4044P-45				
M5W1791-02P	N, Si, ED	I/O	Floppy disk formatter/controller	8—117

Note 1. Al=Aluminum gate, B=Bipolar, C=CMOS, ED=Enhancement depletion mode, FA=FAMOS, N=N-channel, P=P-channel, S=Schottkey, Si=Silicon gate
2. CPU=Central processing unit, I/O=input/output device, PROM=Programmable read-only memory, RAM=Random-access memory, Remo-con=Remote controller, ROM=Read-only memory, Speech=Speech Synthesizer

Type	Function	Page
PC4000	Debugging Machine	13—5
PC4100	M5L8748S programming adaptor	13—36
PC7000	Speech synthesis evaluation unit	13—9
PC8500	MELCS 85/1 portable microcomputer console	13—11
PC9000	Cross assembler machine	13—17
PCA4001	MELPS 4 dedicated board	13—20
PCA4003	MELPS 4 dedicated board	13—22
PCA4004	MELPS 4 dedicated board	13—24
PCA4005	MELPS 4 dedicated board	13—26
PCA4011	MELPS 41 dedicated board	13—28
PCA4012	MELPS 42 dedicated board	13—30
PCA4014	MELPS 42 dedicated board	13—32
PCA4101	MELPS 41 evaluation board	13—42
PCA4201	MELPS 42 evaluation board	13—44
PCA4202	MELPS 42 evaluation board	13—46
PCA4301	MELPS 4 dedicated board	13—37
PCA4303	MELPS 4 evaluation board	13—38
PCA4304	MELPS 4 evaluation board	13—39
PCA4305	MELPS 4 evaluation board	13—40
PCA7002G01	MELCS 70/2 speech synthesizer single-board computer	12—25
PCA7002G02		
PCA8400	MELPS 8-48 dedicated board	13—34
PCA8402	MELPS 8-48 evaluation board	13—48
PCA8501G01	MELCS 85/2 single-board computer	12—3
PCA8501G02		
PCA8506	MELCS 85/2 memory and parallel I/O expansion board	12—7
PCA8507	MELCS 85/2 memory and serial I/O expansion board	12—11
PCA8520G01	MELCS 85/3 voice generating single-board computer	12—15
PCA8520G02		
PCA8540G01	MELCS 85/2 color TV display single-board computer	12—19
PCA8540G02		

GUIDE TO INTERCHANGEABILITY

Function	Mitsubishi Electric	Advanced Micro Devices	American Microsystems	Fairchild Semiconductor	Fujitsu	Hitachi	Intel	Intersil
Static RAMs	**M5L2114LP-2**				MB2114A-20L	HM472114A-2	P2114L-2	
	M5L2114LP-3					HM472114A-3	P2114L-3	
	M5L2114LP					HM472114A-4	P2114L	
	M5T4044P-20				MB8144EL			
	M5T4044P-30				MB8144NL			
	M5T4044P-45							
	M58725P						21U812-20	
	M58725P-15				MB8128-15		21U812-15	
Dynamic RAMs	**M5K4116P-2**							
	M5K4116P-3	AM9016E						
	M5K4164P-15							
	M5K4164P-20							
	M5K4164NP-15							
	M5K4164NP-20							
	M5K4164S-15				MB8265-15			
	M5K4164S-20				MB8265-20			
	M5K4164NS-15				MB8264-15	HM4864-2	C2164-15	
	M5K4164NS-20				MB8264-20	HM4864-3	C2164-20	
CMOS Static RAMs	**M5L5101LP-1**		S5101L-1			HM435101-1	P5101L-1	IM6551
	M58981P-30					HM4334-3		
	M58981P-45				MB8414E	HM4334-4		IM6514
Mask ROM	**M58735-XXXP**							
EPROMs	**M58653P**							
	M5G1400P							
	M5L2716K				MB8516	HN2716	D2716	
	M5L2716K-65						D2716-6	
	M5L2732K				MB8532-45	HN462732	D2732	
	M5L2732K-6						D2732-6	
	M5L2764K				MBM2764-25	HN482764	D2764	
	M5L2764K-2				MBM2764-20		D2764-2	
	M5L2764K-3				MBM2764-30	HN482764-3	D2764-3	
PROMs	**M54700P**			93417P				
	M54700S			93417D				
	M54730P							
	M54730S							
	M54740AP			93452P				
	M54740AS			93452D				
	M54741AP			93453P				
	M54741AS			93453D				

1

Monolithic Memories	Mostek	Motorola Semiconductor products	National Semiconductor	Nippon Electric	Texas Instruments	Toshiba	Signetic
		MCM21L14-20P		μPD2114LC-3	TMS4045-20NL	TMM314APL-1	
		MCM21L14-30P		μPD2114LC-1	TMS4045-30NL	TMM314APL-3	
		MCM21L14-45P		μPD2114LC	TMS4045-45NL	TMM314APL	
		MCM66L41-20P			TMS4044-20NL		
		MCM66L41-30P			TMS4044-30NL		
		MCM66L41-45P			TMS4044-45NL		
				μPD4016C-2	TMS4016	TMM2016P-2	
				μPD4016C-3		TMM2016P	
	MK4116-2						2690-2-N
	MK4116-3						2690-3-N
	MK-4164-15	MCM6664-15JL					
	MK4164-20	MCM6664-20JL					
	MK4564-15	MCM6665-15JL		μPD4164D-3	TMS4164-15	TMM4164C-3	
	MK4564-20	MCM6665-20JL		μPD4164D-2	TMS4164-20	TMM4164C-4	
		MCM145101-1P	NMC6551	μPD5101LC-1		TC5501P	
				μPD444C-1			
			NMC6514	μPD444C		TC5514P	
				μPD2716D		TMM323D	
				μPD2732D			
						TMM2764D	
						TMM2764D-2	
6300			DM74S387N				
6300D			DM74S387J				
6330			DM74S188N				
6330D			DM74S188J				

GUIDE TO INTERCHANGEABILITY

Function	Mitsubishi Electric	Advanced Micro Devices	American Microsystems	Fairchild Semiconductor	Fujitsu	Hitachi	Intel
CPUs	M5L8048-XXXP						P8048
	M5L8035LP						P8035L
	M5L8049-XXXP						P8049
	M5L8049-XXXP-8						
	M5L8049-XXXP-6						
	M5L8039P-11						P8039
	M5L8039P-8						
	M5L8039P-6						P8039-6
	M5L8748S						C8748
	M5L8085AP						P8085A
	M5L8085AS	AM8085A					C8085A
	M5L8086S						C8086
Peripheral circuits	M58990P						
	M5C6847P-1						
	M5L8041A-XXXP						P8041A
	M5L8155P						P8155
	M5L8156P						P8156
	M5L8212P	AM8212			MB471		P8212
	M5L8216P	AM8216					P8216
	M5L8226P	AM8226C					P8226
	M5L8243P						P8243
	M5L8251AP						P8251A
	M5L8253P-5						P8253-5
	M5L8255AP-5						P8255A-5
	M5L8257P-5						P8257-5
	M5L8259AP						P8259A
	M5L8279P-5						P8279P-5
	M5L8282P						P8282
	M5L8283P						P8283
	M5L8284P						P8284
	M5L8286P						P8286
	M5L8287P						P8287
	M5L8288P						P8288
	M5W1791-02P				MB8866		

GUIDE TO INTERCHANGEABILITY

Intersil	Monolithic Memories	Mostek	Motorola Semiconductor products	National Semiconductor	Nippon Electric	Texas Instruments	Toshiba	Signetics
				ADC0808CCN				
				DP8212	μPB8212			
				DP8216				
				DP8226				

GUIDE TO SELECTION OF RAMs, PROMs, EPROMs AND ROMs

Words	Bits/Word		
	1	4	8
32			PROM M54730P, S
256		RAM M5L5101LP-1 PROM M54700P, S	
1024		RAM M58981P-30 M58981P-45 M5L2114LP M5L2114LP-2 M5L2114LP-3 PROM M54740AP, S M54741AP, S	
2048			RAM M58725P M58725P-15 EPROM M5L2716K M5L2716K-65
4096	RAM M5T4044P-20 M5T4044P-30 M5T4044P-45		ROM M58735-XXXP EPROM M5L2732K M5L2732K-6
8192			EPROM M5L2764K M5L2764K-2 M5L2764K-3
16384	RAM M5K4116P-2 M5K4116P-3		
65536	RAM M5K4164P-15 M5K4164P-20 M5K4164NP-15 M5K4164NP-20 M5K4164S-15 M5K4164S-20 M5K4164NS-15 M5K4164NS-20		

ORDERING INFORMATION

FUNCTION CODE

Mitsubishi integrated circuits may be ordered using the following simplified alphanumeric type-codes which define the function of the ICs and the package style.

For Mitsubishi Original Products

Example: M 5 89 81 S – 45

- **M**: Mitsubishi integrated circuit prefix
- Temperature range
  - **5**: Standard industrial/commercial (0 to 70/75°C or –20 to 85°C).
  - **9**: High reliability
- Series designation using 1 or 2 alphanumeric characters.
  - **0**: CMOS
  - **1**: Linear circuit
  - **3**: TTL
  - **10～19**: Linear circuit
  - **32～33**: TTL (equivalent to Texas Instruments' SN74 series)
  - **41～47**: TTL
  - **84**: CMOS
  - **85**: P-channel silicon-gate MOS
  - **86**: P-channel aluminum-gate MOS
  - **87**: N-channel silicon-gate MOS
  - **88**: P-channel aluminum-gate ED-MOS
  - **89**: CMOS
  - **9**: DTL
  - **S0～S2**: Schottky TTL (equivalent to Texas Instruments' SN74S series)
- Circuit function identification code using 2 digits.
- Package style
  - **K**: Glass-sealed ceramic
  - **P**: Molded plastic
  - **S**: Metal-sealed ceramic
- Electrical characteristic identification code using 1 or 2 digits.

For Second Source Products

Example: M 5 K 4116 S – 2

- **M**: Mitsubishi integrated circuit prefix
- Temperature range
  - **5**: Standard industrial/commercial (0 to 70/75°C or –20 to 85°C)
  - **9**: High reliability
- Series designation of original source using 1 or 2 alphabetical characters.
  - **C**: Motorola's MC series
  - **G**: General Instrument's series
  - **K**: Mostek's MK series
  - **L**: Intel's series
  - **T**: Texas Instruments' TMS series
  - **W**: Western Digital's series
- Circuit function identification code of the original source type name
- Package style
  - **K**: Glass-sealed ceramic
  - **P**: Molded plastic
  - **S**: Metal-sealed ceramic
- Electrical characteristic identification code using 1 or 2 digits.

PACKAGE CODE

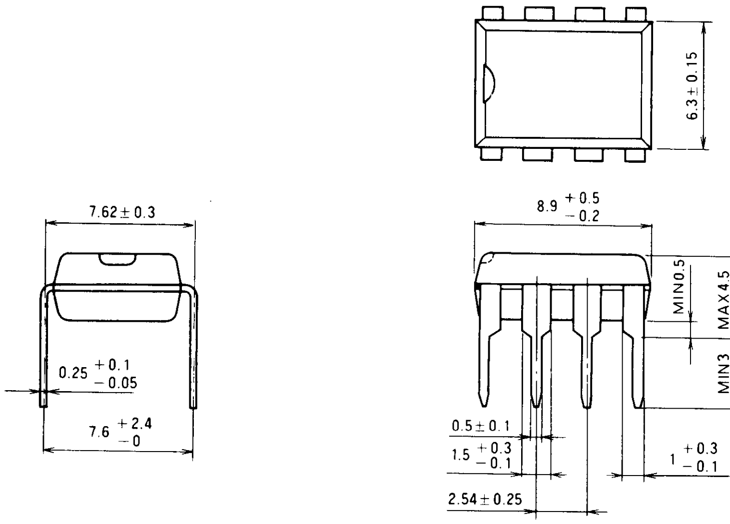
Package style may be specified by using the following simplified alphanumeric code.

Example: 24 P 1

- Number of pins
- Package structure
  - **K**: Glass-sealed ceramic
  - **P**: Molded plastic
  - **S**: Metal-sealed ceramic
- Package outline
  - **1**: DIL without fin
  - **2**: Flat without fin
  - **4**: DIL without fin (improved)
  - **4B**: Shrink DIL without fin
  - **10**: DIL without fin and with quartz lid

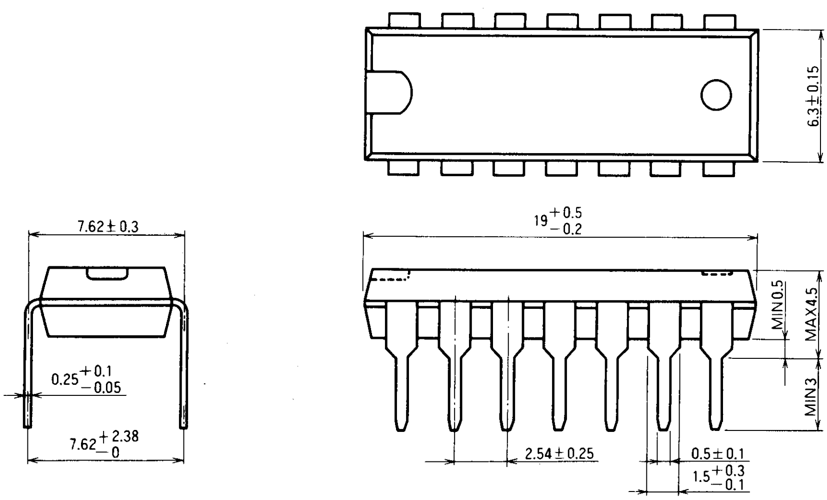
TYPE 8P4 8-PIN MOLDED PLASTIC DIL

UNIT: mm



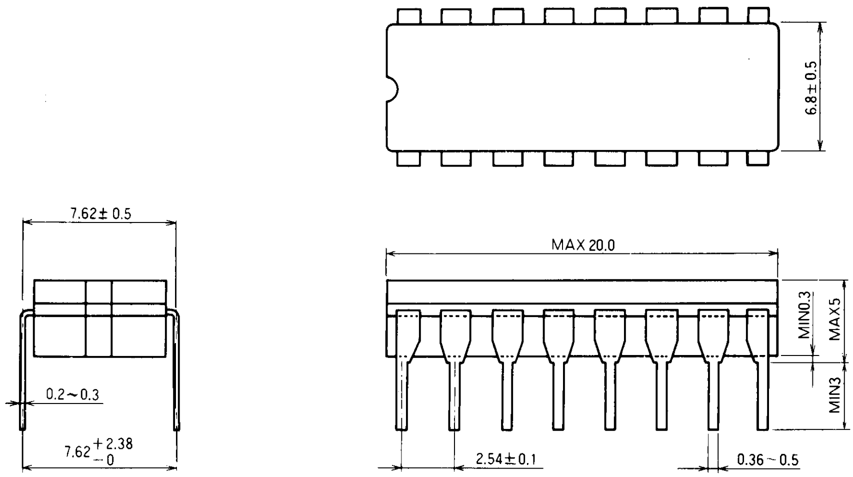
TYPE 14P4 14-PIN MOLDED PLASTIC DIL

UNIT: mm



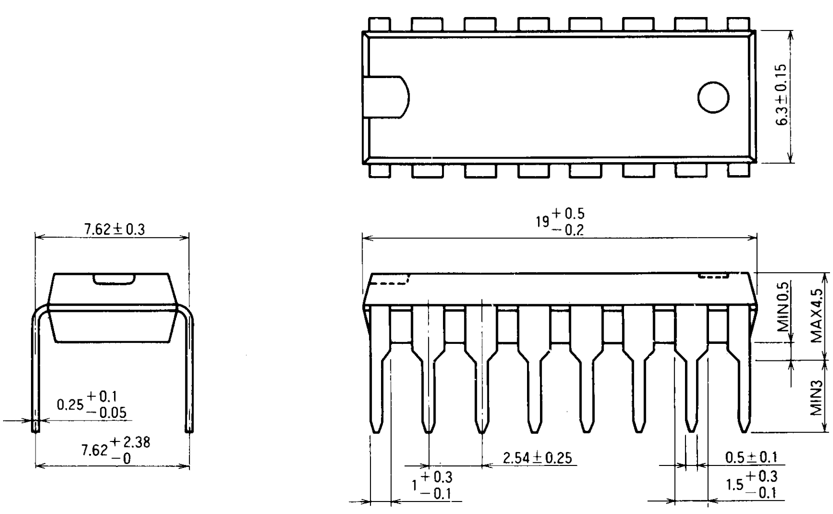
TYPE 16K1 16-PIN GLASS-SEALED CERAMIC DIL

UNIT: mm



TYPE 16P4 16-PIN MOLDED PLASTIC DIL

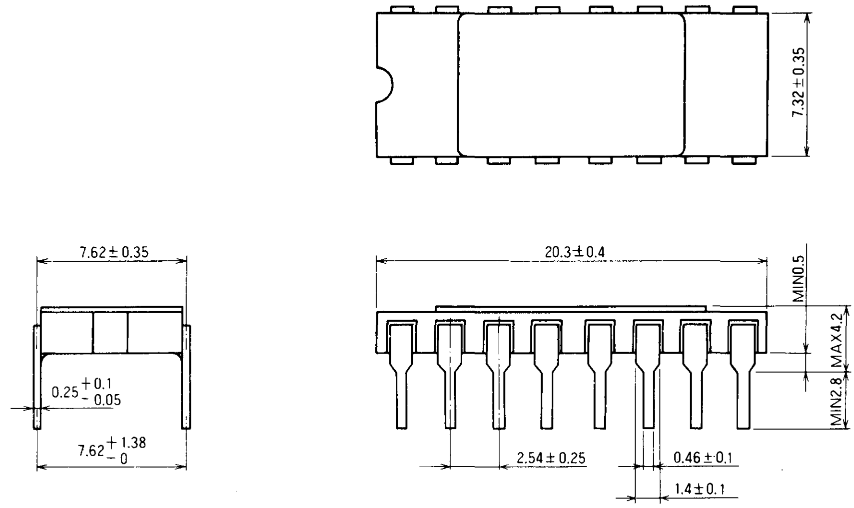
UNIT: mm



1

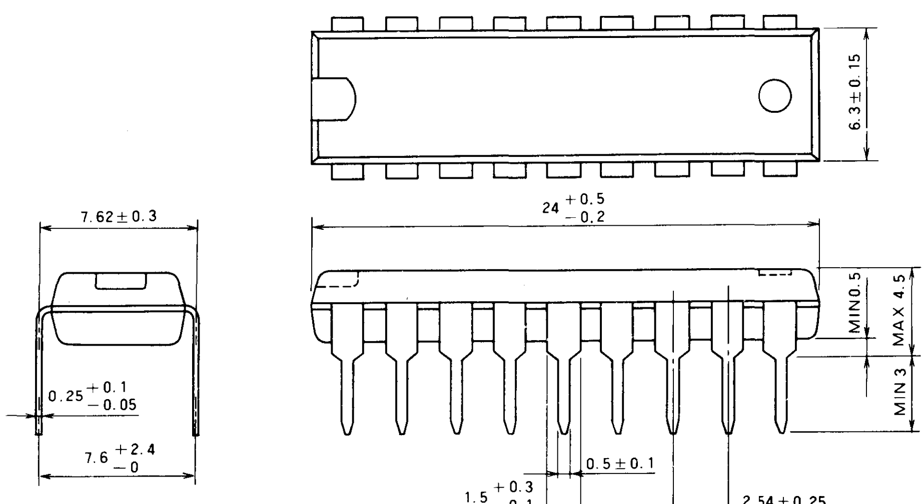
TYPE 16S1 16-PIN METAL-SEALED CERAMIC DIL

UNIT: mm



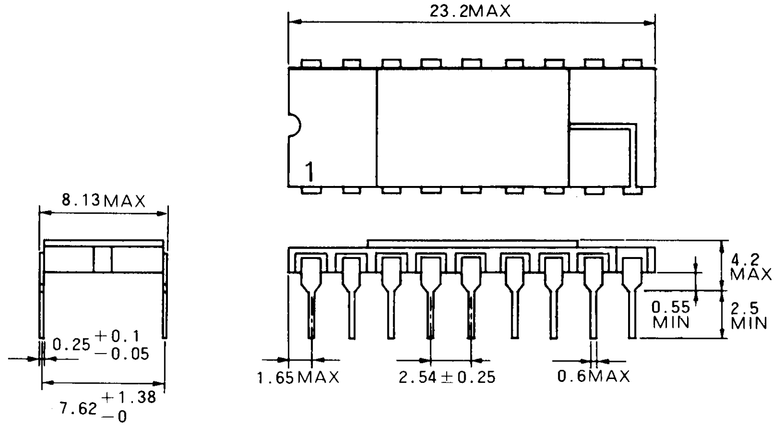
TYPE 18P4 18-PIN MOLDED PLASTIC DIL

UNIT: mm



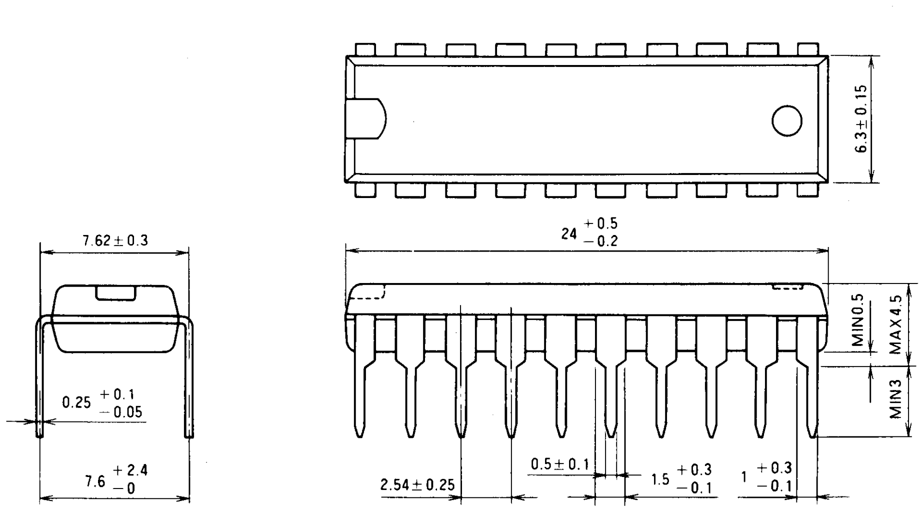
TYPE 18S1 18-PIN METAL-SEALED CERAMIC DIL

UNIT: mm



TYPE 20P4 20-PIN MOLDED PLASTIC DIL

UNIT: mm

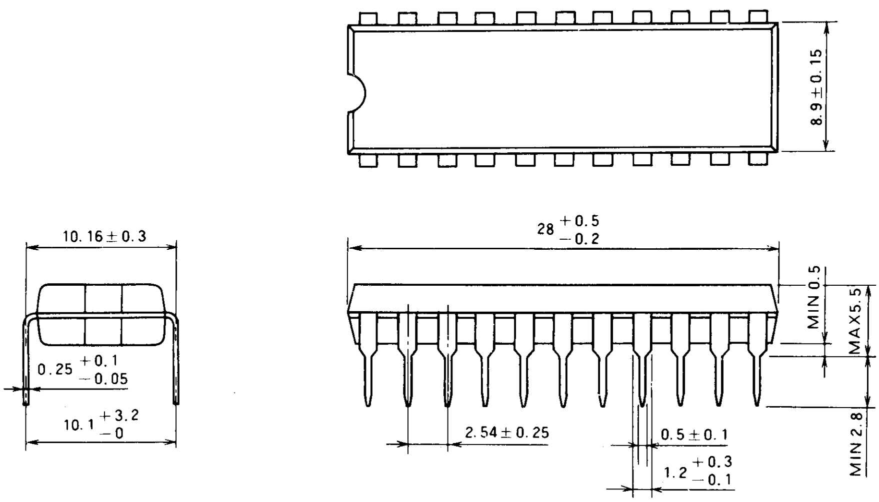


MITSUBISHI
ELECTRIC

1

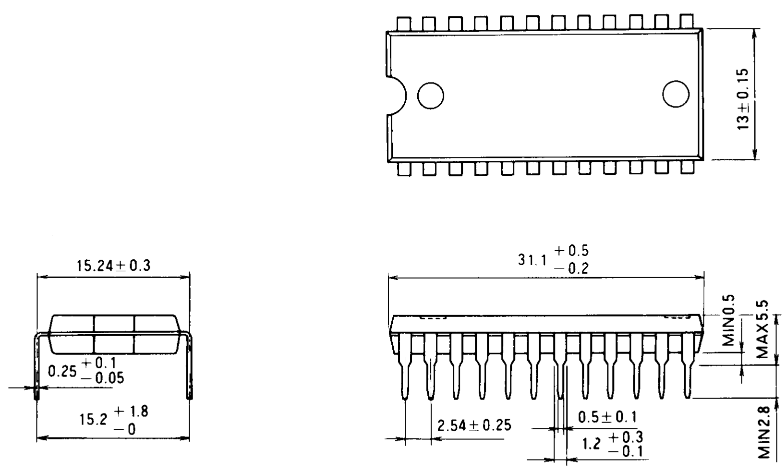
TYPE 22P1 22-PIN MOLDED PLASTIC DIL

UNIT: mm

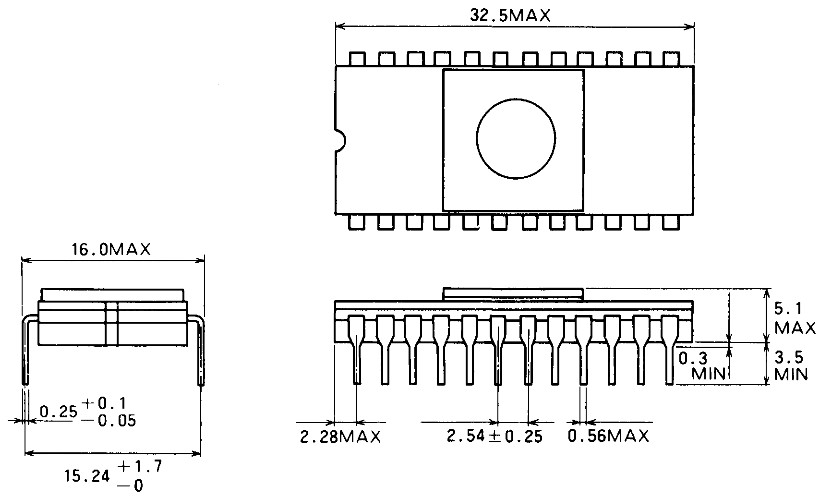


TYPE 24P1 24-PIN MOLDED PLASTIC DIL

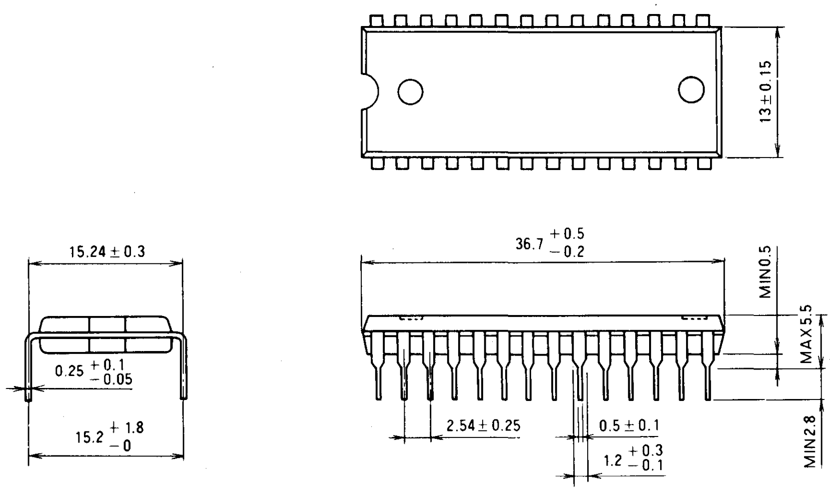
UNIT: mm



TYPE 24K10 24-PIN METAL-SEALED CERAMIC DIL WITH QUARTZ LID UNIT: mm



TYPE 28P4 28-PIN MOLDED PLASTIC DIL UNIT: mm

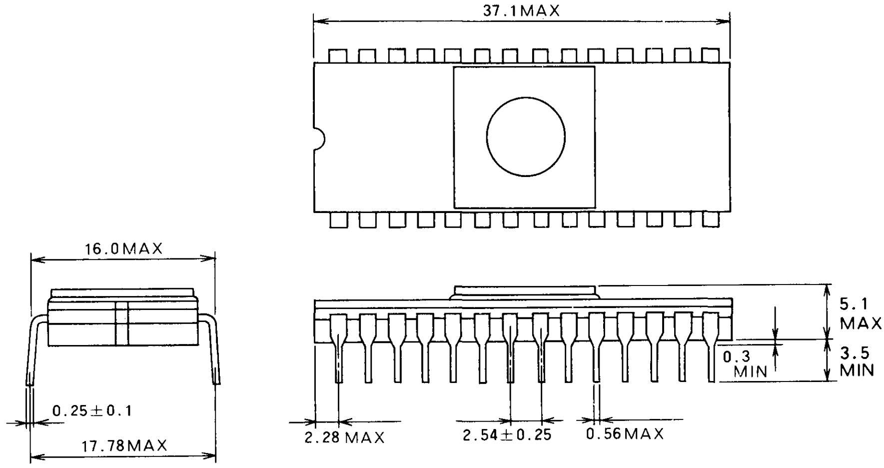


MITSUBISHI ELECTRIC

1

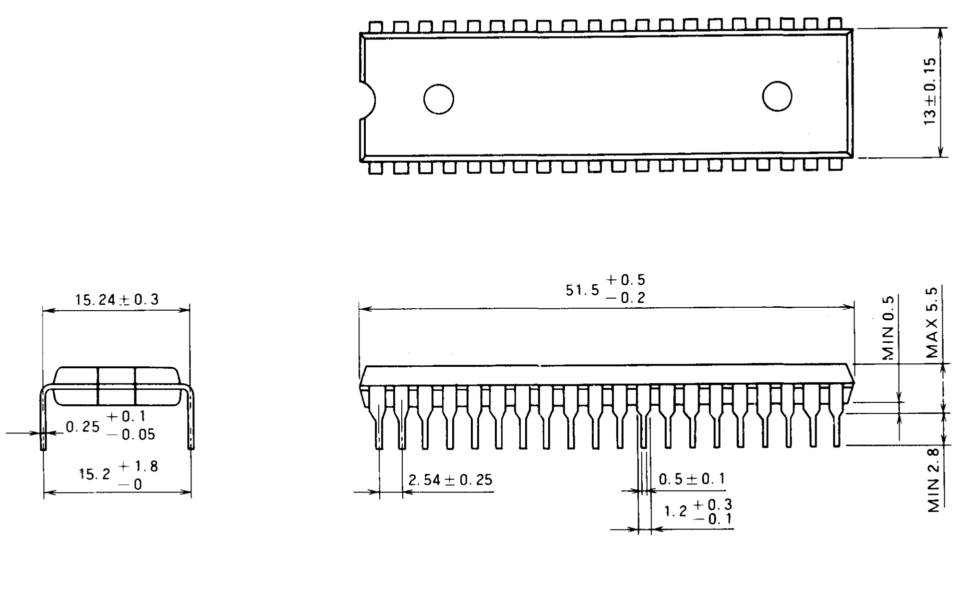
TYPE 28K10 28-PIN METAL-SEALED CERAMIC DIL WITH QUARTZ LID

UNIT: mm



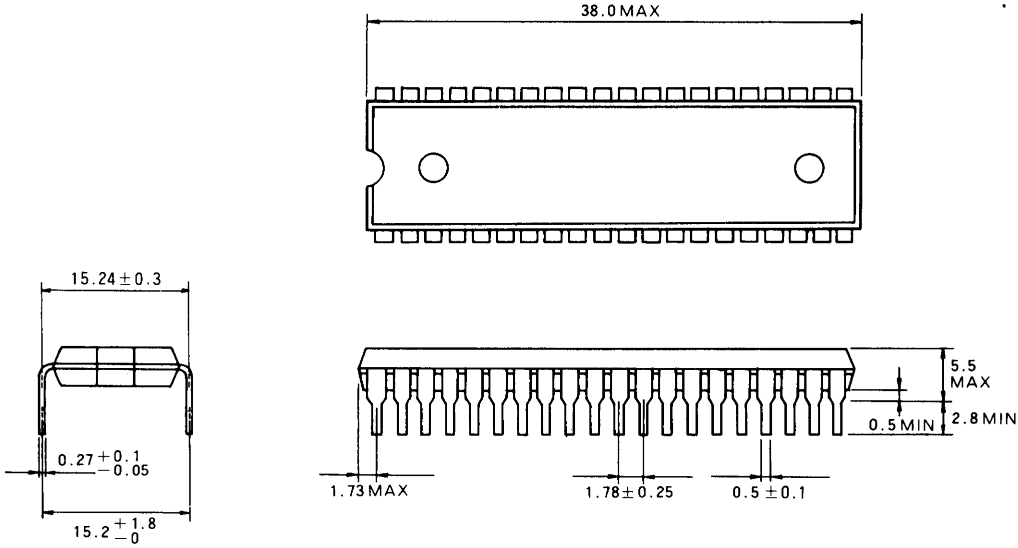
TYPE 40P1 40-PIN MOLDED PLASTIC DIL

UNIT: mm



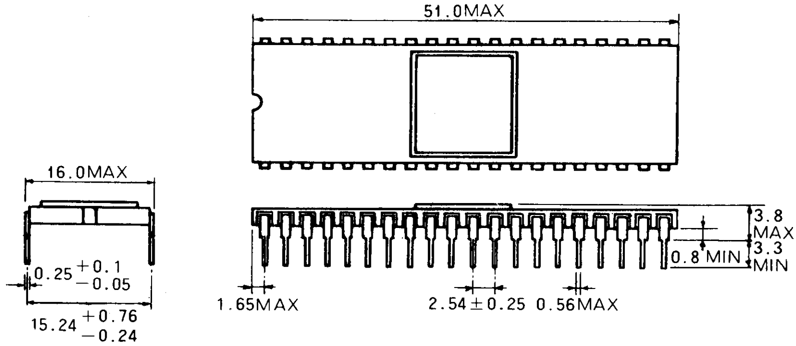
TYPE 40P4B 40-PIN SHRINK MOLDED PLASTIC DIL

UNIT: mm



TYPE 40S1 40-PIN METAL-SEALED CERAMIC DIL

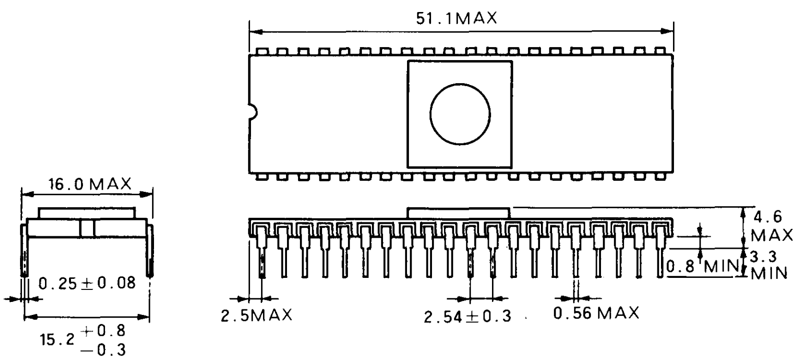
UNIT: mm



MITSUBISHI ELECTRIC

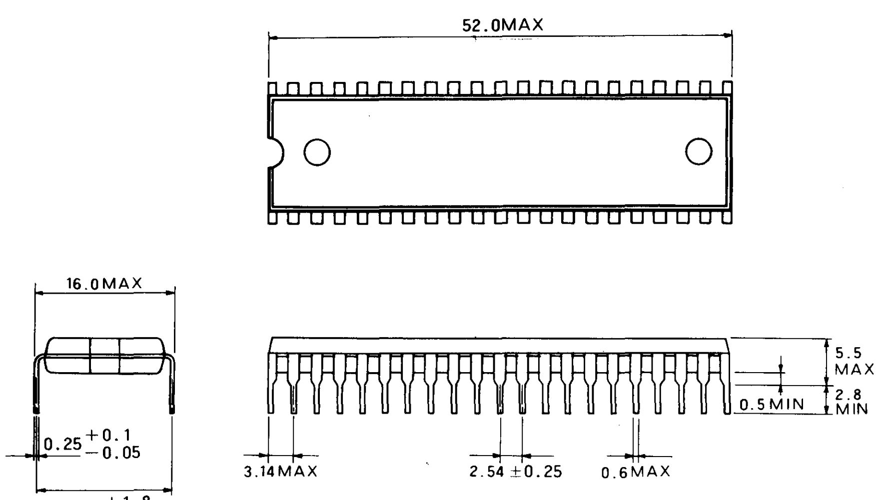
TYPE 40S10 40-PIN METAL-SEALED CERAMIC DIL WITH QUARTZ LID

UNIT: mm



TYPE 42P1 42-PIN MOLDED PLASTIC DIL

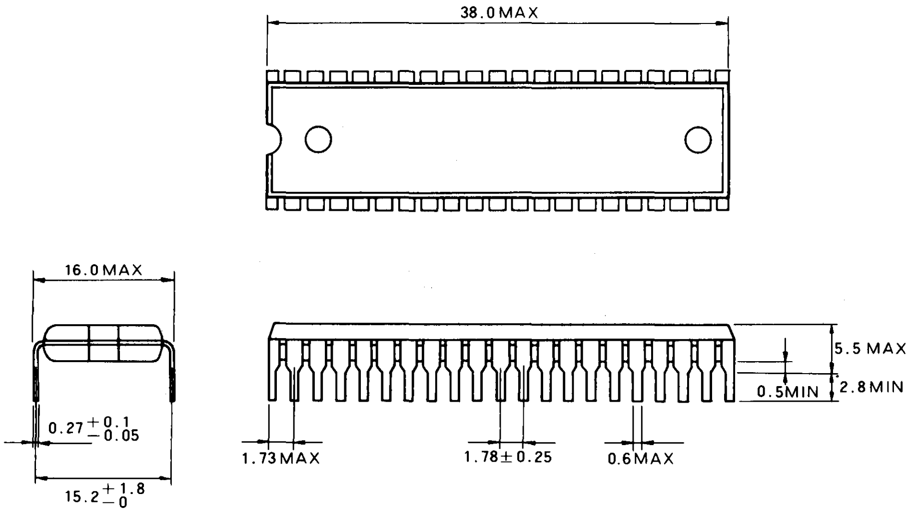
UNIT: mm



PACKAGE OUTLINES

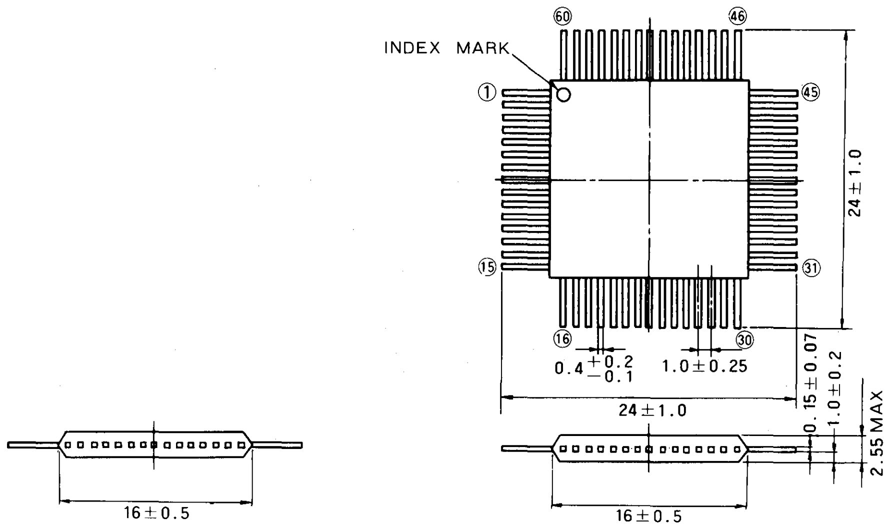
TYPE 42P4B 42-PIN SHRINK MOLDED PLASTIC DIL

UNIT: mm



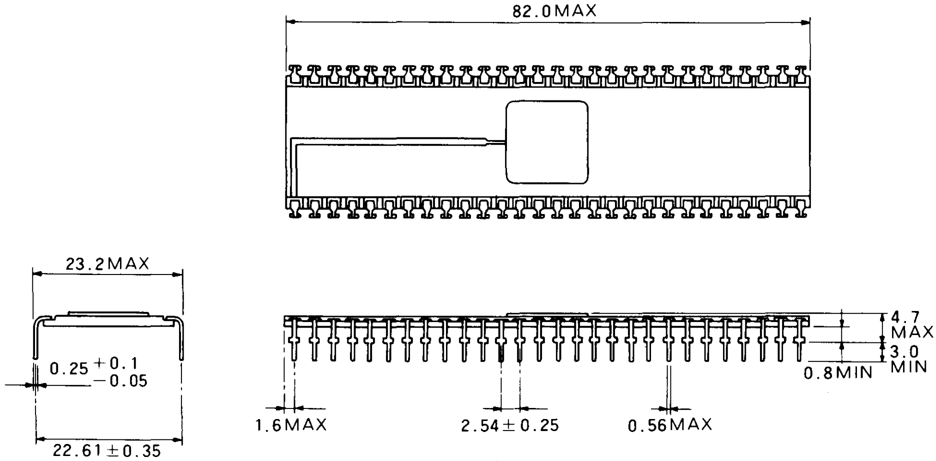
TYPE 60P2 60-PIN MOLDED PLASTIC FLAT

UNIT: mm



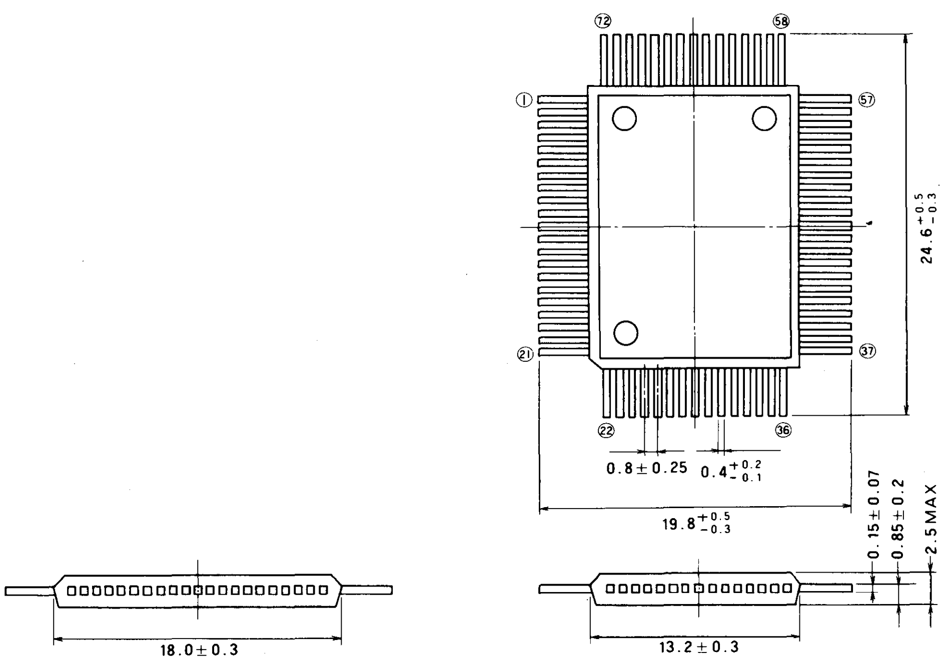
TYPE 64S1 64-PIN METAL-SEALED CERAMIC DIL

UNIT: mm



TYPE 72P2 72-PIN MOLDED PLASTIC FLAT

UNIT: mm



TERMINOLOGY

GENERAL

Semiconductor A material with resistivity usually in the range between metals and insulators, in which the electrical charge carrier concentration increases with increasing temperature range.

Extrinsic semiconductor A semiconductor with charge carrier concentration dependent upon impurities or other imperfections.

N-type semiconductor An extrinsic semiconductor in which the conduction electron density exceeds the mobile hole density.

P-type semiconductor An extrinsic semiconductor in which the mobile hole density exceeds the conduction electron density.

Junction A region of transition between semiconducting regions of different electrical properties.

PN junction A junction between P- and N-type semiconductor materials.

Depletion layer A region in which the mobile charge carrier density is insufficient to neutralize the net fixed charge density of donors and acceptors.

Breakdown (of a reverse-biased PN junction) A phenomenon, the initiation of which is observed as a transition from a state of dynamic resistance to a state of substantially lower dynamic resistance, for increasing the magnitude of a reverse current.

Semiconductor device A device whose essential characteristics are due to the flow of charge carriers within a semiconductor.

Reverse voltage The voltage across a junction or a diode when biased in the direction corresponding to the higher resistance.

Breakdown voltage The reverse voltage at which the reverse current through a junction becomes greater than a specified value.

Case temperature The temperature measured at a specified point on the case of a semiconductor device.

Storage temperature The temperature at which a semiconductor device is stored without any voltage applied.

INTEGRATED CIRCUITS

Microelectronics The concept of the construction and use of highly miniaturized electronic circuits.

Microcircuit A microelectronic device, having a high equivalent circuit-element and/or component density, which is considered as a single unit.

Note: A microcircuit may be a microassembly or an integrated (micro) circuit.

Integrated circuit A circuit in which a number of circuit elements are inseparably associated and electrically interconnected, so that, for the purpose of specification, testing, commerce and maintenance, it is considered indivisible.

Note: For this definition, a circuit element does not have an envelope or external connection and is not specified or sold as a separate item.

Integrated microcircuit A microcircuit in which a number of circuit elements are inseparably associated and electrically interconnected, so that, for the purpose of specification, testing, commerce and maintenance, it is considered indivisible.

Note 1: For this definition, a circuit element does not have an envelope or external connection and is not specified or sold as a separate item.

2: Where no misunderstanding is possible, the term 'integrated microcircuit' may be abbreviated to 'integrated circuit.

3: Further qualifying terms may be used to describe the technique used in the manufacture of a specific integrated microcircuit.

Examples of the use of qualifying terms are:

semiconductor monolithic integrated circuit
semiconductor multichip integrated circuit
thin film integrated circuit
thick film integrated circuit
hybrid integrated circuit

Microassembly A microcircuit consisting of various components and/or integrated microcircuits which are constructed separately and which can be tested before being assembled and packaged.

Note 1: For this definition, a component has external connections and possibly an envelope as well, and it also can be specified and sold as a separate item.

2: Further qualifying terms may be used to describe the form of the components and/or the assembly techniques used in the construction of a specific microassembly

Examples of use of qualifying terms are:

semiconductor multichip microassembly
discrete component microassembly

Integrated electronics The art and technology of the design, fabrication and use of integrated circuits.

Worst-case conditions (for a single characteristic) The values of the applied conditions which are individually chosen from within a specified range and together produce the most unfavorable value for a considered characteristic.

Note: Worst-case conditions for different characteristics may be different.

DIGITAL INTEGRATED CIRCUITS

Digital signal The variation with time of a physical quantity that is used for the transmission of information or for information processing, and that has a finite number of nonoverlapping ranges of values.

Note 1: The physical quantity may be voltage, or current, or impedance, etc.

2: For convenience, each range of values can be represented by a single value—e.g., the nominal value.

1

Binary signal A digital signal with only two possible ranges of values.

Note: For convenience, each range of values can be represented by a single value—e.g., the nominal value.

Low range (of a binary signal) The range of least positive (most negative) levels of a binary signal.

Note: This range is often denoted by 'L-range,' and any level in the range by 'L-level.

High range (of a binary signal) The range of most positive (least negative) levels of a binary signal.

Note: This range is often denoted by 'H-range,' and any level in the range by 'H-level.

Digital circuit A circuit which is designed to operate by means of digital signals at the input(s) and at the output(s).

Note 1: In this definition, it is understood that 'inputs' and 'outputs' exclude static power supplies.
2: In some digital circuits—e.g., certain types of a stable circuits—the inputs need not exist.

Binary circuit A digital circuit designed to operate with binary signals.

Note: The pairs of ranges of values of the binary signals may be different at different terminals.

Input configuration (input pattern) (of a binary circuit) A combination of the L-levels and H-levels at the input terminals at a given instant.

Output configuration (output pattern) (of a binary circuit) A combination of the L-levels and H-levels at the output terminals at a given instant.

Note: When there is no possibility of ambiguity, the output configuration (output pattern) may be represented by the level (expressed as L-level or H-level) of the signal at a stated output terminal of the circuit (the reference output terminal).

Input terminal A terminal by means of which an applied signal may modify the output configuration (output pattern) of the circuit—either directly or indirectly—by modifying the ways in which the circuit reacts to signals at other terminals.

Combinatorial (digital) circuit A digital circuit in which there exists one, and only one, combination of the digital signals at the outputs for each possible combination of digital signals at the inputs.

Sequential (digital) circuit A digital circuit in which there exists at least one combination of the digital signals at the inputs for which there is more than one corresponding combination of the digital signals at the outputs.

Note: These combinations at the outputs are determined by previous history—e.g., as a result of internal memory or delay.

Elementary combinatorial circuit A binary combinatorial (digital) circuit which has only one output terminal, and in which the output signal accepts the value occurring only once in the function if, and only if, the signals applied to all the input terminals are either all in the H-range or all in the L-range.

Note 1: Because the output signal value (occurring only once in the function table) can lie either in the H-range or in the L-range, there are four types of elementary combinatorial circuits.
According to the assignment of the signal values L and H to the binary values 0 and 1 of Boolean algebra, the following logic operations can be realized by means of the four types of elementary combinatorial circuits: AND, OR, NAND, NOR.
2: Nonelementary combinatorial circuits can be formed by combining elementary combinational circuits or by combining elementary combinatorial circuits with inverters.

Function table A representation of the necessary or possible relations between the values of the digital signals at the inputs and the outputs of a digital circuit, these values of the digital signals being indicated either by using electrical values directly or by stating the electrical significance of the symbols—e.g., L and H for binary circuits. Generally, every column indicates the values of the digital signals at an input or at an output of the digital circuit; every row indicates the combination of values of the digital signals at the input(s) and the resulting values of the digital signals at the output(s); whenever the value of the digital signal at an output is not determined, it should be indicated by a question mark; whenever the value of a digital signal at an input has no influence, it should be indicated by the symbol L/H or X.

Truth table (for a relation between digital variables) A representation of the logic relationship between one or more independent digital variables and one or more dependent digital variables, by means of a table which, for each possible combination of the values of the independent variables, gives the appropriate values of the dependent variables.

Note: The distinction between 'function table' and 'truth table' is fundamentally necessary, because the same digital circuit may fulfill several different logic operations, according to the arbitrary assignment of the values of the digital variables to the values of the digital electrical quantities.

Input loading factor (of a bipolar digital circuit) A factor which indicates the ratio of the input current of a specified input terminal of a digital circuit to the input current of a particular circuit which is chosen as a reference load.

Note: The reference load should preferably be chosen in such a way that the input loading factor becomes an integer.

Output loading capability (of a bipolar digital circuit) A factor which indicates the ratio of the maximum output concurrent of a specified output terminal of a digital circuit to the input current of a particular circuit which is chosen as a reference load.

Note: The reference load should preferably be chosen in such a way that the output loading capability becomes an integer.

Excitation An input configuration (input pattern), or change in input configuration (input pattern), that can cause the circuit to change its output configuration (output pattern), either directly, or in conjunction with an already existing state of preparedness; or put the circuit in a state of preparedness; or either cancel or modify an already existing state of preparedness.

Note 1: The repetition or reiteration of a given excitation will not necessarily produce the same effect:

2: In some cases, an excitation can also maintain an output configuration (output pattern) which it could have produced.

Expander circuit An auxiliary circuit which can be used to expand the number of inputs of equal influence of an associated circuit without modifying the function of the associated circuit.

Binary inverter A binary circuit which has only one input terminal and one output terminal, and in which a signal value L (or H) at the input produces a signal value H (or L) at the output.

Function (sequential) matrix A table having several inputs which gives the possible output configurations for each input configuration(s) and from which the output configuration(s) resulting from a transition from each individual input configuration to any other input configuration can be read directly.

Note: Where appropriate, a function (sequential) matrix may be completed by additional data or details concerning time conditions—e.g., transition times for the input levels, delay time, duration of the input configuration to produce a desired new output configuration.

SEQUENTIAL CIRCUITS

Master-slave arrangement An arrangement of two bistable circuits such that one of them, called the 'slave,' reproduces the output configuration of the other circuit, called the 'master.' The transfer of information from the master to the slave is produced by means of an appropriate signal.

Register An arrangement of bistable circuits by means of which information may be accepted, stored and restituted.

Note: The register may form part of another memory and is of a specified capacity.

Shift Register A register that, by means of an appropriate control signal, can transfer information between consecutive bistable circuits with the sequence being preserved.

Counter A sequential circuit for storing numbers that permits such numbers to be incremented or decremented by a defined constant, including unity.

TIME INTERVALS BETWEEN INPUT SIGNALS

Setup time (t_{su}) (of a digital circuit) The time interval between the application of a signal which is maintained at a specified input terminal and a consecutive active transition at another specified input terminal.

Note 1: The setup time is measured between the instants at which the magnitudes of the two signals pass through specified values within the transition of the signal levels.

2: The setup time is the actual time between two events and may be insufficient to accomplish the setup. A minimum value is specified which is the shortest interval for which correct operation of the digital circuit is guaranteed.

3: The setup time may have a negative value, in which case the minimum limit for which correct operation of the digital circuit is guaranteed defines the longest interval between the active transition and the application of the other signal.

Hold time (t_h) (of a digital circuit) The time interval during which a signal is retained at a specified input terminal after an active transition occurs at another specified input terminal.

Note 1: The hold time is measured between the instants at which the magnitudes of the two signals pass through specified values within the transitions of the signal levels.

2: The hold time is the actual time between two events and may be insufficient to accomplish the intended result.
A minimum value is specified which is the shortest interval for which correct operation of the digital circuit is guaranteed.

3: The hold time may have a negative value, in which case the minimum limit for which correct operation of the digital circuit is guaranteed defines the longest interval between the change of the signal and the active transition.

Resolution time (t_{res}) (of a digital circuit) The time interval between the cessation of one input pulse and the commencement of the next input pulse applied to the same input terminal.

Note 1: The resolution time is measured between the instants at which the magnitude of the input signal passes through specified values within the transitions of the signal levels.

2: The resolution time is the actual time between two pulses and may be insufficient to ensure that both pulses are recognized. A minimum value is specified which is the shortest interval for which correct operation of the digital circuit is guaranteed.

1

SWITCHING TIMES OF BINARY CIRCUITS

High-level to low-level (low-level to high-level) propagation time (t_{PHL} and t_{PLH}) The time interval between specified reference points on the input and on the output pulses, when the output is going to the low (high) level and when the device is driven and loaded by typical devices of stated type.

Note: The mean value between the upper limit of the input low range and the lower limit of the input high range is generally used as the specified reference level.

High-level to low-level (low-level to high-level) delay time (t_{DHL} and t_{DLH}) The time interval between specified reference points on the input and on the output pulses, when the output is going to the low (high) level and when the device is driven and loaded by specified networks.

High-level to low-level (low-level to high-level) transition time (t_{THL} and t_{TLH}) The time interval between specified reference points on the edge of the output pulse when the output is going to the low (high) level and when a specified input signal is applied through a specified network and the output is loaded by another specified network.

INTEGRATED CIRCUIT MEMORIES

Memory cell (memory element) The smallest subdivision of a memory into which a unit of data has been or can be entered, in which it is or can be stored, and from which it can be retrieved.

Integrated circuit memory An integrated circuit consisting of memory cells (elements) and usually including associated circuits such as those for address selection, amplifiers, etc.

Read-only memory (ROM) A memory intended to be read only.

Note: Unless otherwise specified, the term 'read-only memory implies that the content is unalterable, and defined by its structure.

Fixed-programmed read-only memory A read-only memory in which the data contents of each cell (element) are determined during manufacture and are thereafter unalterable.

Mask-programmed read-only memory A fixed-programmed read-only memory in which the data contents of each cell (element) are determined during manufacture by the use of a mask.

Field-programmable read-only memory A read-only memory that, after being manufactured, can have the data content of each memory cell (element) altered.

Programmable read-only memory (PROM) A read-only memory that can have the data content of each memory cell (element) altered once only.

Reprogrammable read-only memory A read-only memory that can have the data content of each memory cell (element) altered more than once.

Read/write memory A memory in which each cell (element) may be selected by applying appropriate electrical input signals, and in which the stored data may be either: a) sensed at appropriate output terminals; or b) changed in response to other similar electrical input signals.

Static read/write memory A memory in which the data is retained in the absence of control signals.

Note 1: The words 'read/write' may be omitted from the term when no misunderstanding will result.
2: A static memory may use dynamic addressing or sensing circuits.

Dynamic read/write memory A memory in which the cells (elements) require the repetitive application of control signals in order to retain the data stored.

Note 1: The words 'read/write may be omitted from the term when no misunderstanding will result.
2: Such repetitive application of the control signals is normally called a refresh operation.
3: A dynamic memory may use static addressing or sensing circuits.
4: This definition applies whether the control signals are generated inside or outside the integrated circuit.

Volatile memory A memory whose data content is lost when the power supply is disconnected.

Random-access memory (RAM) A memory that permits access to any of its address locations in any desired sequence.

MICROPROCESSOR INTEGRATED CIRCUITS

Microprocessor integrated circuit An integrated circuit capable of:

1. Accepting coded instructions at one or more terminals.
2. Carrying out, in accordance with the instructions received, all of:
   a. the acceptance of coded data for processing and/or storage;
   b. arithmetic and logical operations on the input data together with any relevant data stored in the microprocessor integrated circuit;
   c. the delivery of coded data.
3. Accepting and/or delivering signals controlling and/or describing the operation or state of the microprocessor integrated circuit.

Note: The instructions may be fed in, built in, or held in an internal store.

Note: The definitions of terms described here are extracted from IEC publication 147-0. Some of the terms for integrated circuit memories and microprocessors are under consideration.

LETTER SYMBOLS FOR THE DYNAMIC PARAMETERS

1. INTRODUCTION

A system of letter symbols to be used to represent the dynamic parameters of intergrated circuit memories and other sequential circuits especially for single-chip microcomputers, microprocessors and LSIs for peripheral circuits has been discussed internationally in the TC47 of the International Electrotechnical Committee (IEC). Finally the IEC has decided on the meeting of TC47 in February 1980 that this system of letter symbols will be a Central Office document and circulated to all countries to vote which means this system of letter symbols will be a international standard.

The system is applied in this LSI data book for the new products only. Future editions of this data book will be applied this system. The IEC document which describes "Letter symbols for dynamic parameters of sequential integrated circuits, including memories" is introduced below. In this data book, the dynamic parameters in the IEC document are applied to timing requirements and switching characteristics.

2. LETTER SYMBOLS

The system of letter symbols outlined in this document enables symbols to be generated for the dynamic parameters of complex sequential circuits, including memories, and also allows these symbols to be abbreviated to simple mnemonic symbols when no ambiguity is likely to arise.

2.1. General Form

The dynamic parameters are represented by a general symbol of the form:-

$t_{A(BC-DC)F}$ .... (1)

where :

Subscript A indicates the type of dynamic parameter being represented, for example; cycle time, setup time, enable time, etc.

Subscript B indicates the name of the signal or terminal for which a change of state or level (or establishment of a state or level) constitutes a signal event assumed to occur first, that is, at the beginning of the time interval. If this event actually occurs last, that is, at the end of the time interval, the value of the time interval is negative.

Subscript C indicates the direction of the transition and/or the final state or level of the signal represented by B. When two letters are used, the initial state or level is also indicated.

Subscript D indicates the name of the signal or terminal for which a change of state or level (or establishment of a state or level) constitutes a signal event assumed to occur last, that is, at the end of the time interval. If this event actually occurs first, that is, at the beginning of the time interval, the value of the time interval is negative.

Subscript E indicates the direction of the transition and/or the final state or level of the signal represented by D. When two letters are used, the initial state or level is also indicated.

Subscript F indicates additional information such as mode of operation, test conditions, etc.

Note 1: Subscripts A to F may each consists of one or more letters.
2: Subscripts D and E are not used for transition times.
3: The "–" in the symbol (1) above is used to indicate "to"; hence the symbol represents the time interval from signal event B occuring to signal event D occuring, and it is important to note that this convention is used for all dynamic parameters including hold times. Where no misunderstanding can occur the hyphen may be omitted.

2.2. Abbreviated Form

The general symbol given above may be abbreviated when no misunderstanding is likely to arise. For example to :

$t_{A(B-D)}$

or $t_{A(B)}$

or $t_{A(D)}$ – often used for hold times

or t_{AF} – no brackets are used in this case

or t_{A}

or t_{BC-DE} – often used for unclassified time intervals

2.3. Allocation of Subscripts

In allocating letter symbols for the subscripts, the most commonly used subscripts are given single letters where practicable and those less commonly used are designated by up to three letters. As far as possible, some form of mnemonic representation is used. Longer letter symbols may be used for specialised signals or terminals if this aids understanding.

3. SUBSCRIPT A (For Type of Dynamic Parameter)

The subscript A represents the type of dynamic parameter to be designated by the symbol and, for memories, the parameters may be divided into two classes :

a) those that are timing requirements for the memory and

b) those that are characteristics of the memory. The letter symbols so far proposed for memory circuits are listed in sub-clauses 3.1 and 3.2 below.
All subscripts A should be in lower-case.

3.1. Timing Requirements

The letter symbols for the timing requirements of semiconductor memories are as follows :

Term	Subscript
Cycle time	c
Time interval between two signal events	d
Fall time	f
Hold time	h
Precharging time	pc
Rise time	r
Recovery time	rec
Refresh time interval	rf
Setup time	su
Transition time	t
Pulse duration (width)	w

3.2. Characteristics

The letter symbols for the dynamic characteristics of semiconductor memories are as follows :

Characteristic	Subscript
Access time	a
Disable time	dis
Enable time	en
Propagation time	p
Recovery time	rec
Transition time	t
Valid time	v

Note: Recovery time for use as a characteristic is limited to sense recovery time.

4. SUBSCRIPTS B AND D (For Signal Name or Terminal Name)

The letter symbols for the signal name or the name of the terminal are as given below.
All subscripts B and D should be in upper-case.

Signal or terminal	Subscript
Address	A
Clock	C
Column address	CA
Column address strobe	CAS
Data input	D
Data input/output	DQ
Chip enable	E
Erasure	ER
Output enable	G
Program	PR
Data output	Q
Read	R
Row address	RA
Row address strobe	RAS
Refresh	RF
Read/Write	RW
Chip select	S
Write (write enable)	W

Note 1: In the letter symbols for time intervals, bars over the subscripts, for example CAS, should not be used.
2: It should be noted, when further letter symbols are chosen, that the subscript should not end with H, K, V, X, or Z. (See clause 5)
3: If the same terminal, or signal, can be used for two functions (for example Data input/output, Read/Write) the waveform should be labelled with the dual function, if appropriate, but the symbols for the dynamic parameters should include only that part of the subscript relevant to the parameter.

5. SUBSCRIPTS C AND E (For Transition of Signal)

The following symbols are used to represent the level or state of a signal :

Transition of signal	Subscript
High logic level	H
Low logic level	L
Valid steady-state level (either low or high)	V
Unknown, changing, or 'don't care' level	X
High-impedance state of three-state output	Z

The direction of transition is expressed by two letters, the direction being from the state represented by the first letter to that represented by the second letter, with the letters being as given above.
When no misunderstanding can occur, the first letter may be omitted to give an abbreviated symbol for subscripts C and E as indicated below.
All subscripts C and E should be in upper-case.

Examples	Subscript Full	Subscript Abbreviated
Transition from high level to low level	HL	L
Transition from low level to high level	LH	H
Transition from unknown or changing state to valid state	XV	V
Transition from valid state to unknown or changing state	VX	X
Transition from high-impedance state to valid state	ZV	V

Note: Since subscripts C and E may be abbreviated, and since subscripts B and D may contain an indeterminate number of letters, it is necessary to put the restriction on the subscripts B and D that they should not end with H, L, V, X, or Z, so as to avoid possible confusion.

6. SUBSCRIPT F (For Additional Information)

If necessary, subscript F is used to represent any additional qualification of the parameter such as mode of operation, test conditions, etc. The letter symbols for subscript F are given below.
Subscript F should be in upper-case.

Modes of operation	Subscript
Power-down	PD
Page-mode read	PGR
Page-mode write	PGW
Read	R
Refresh	RF
Read-modify-write	RMW
Read-write	RW
Write	W

FOR DIGITAL INTEGRATED CIRCUITS

New symbol	Former symbol	Parameter—definition
C_i		Input capacitance
C_o		Output capacitance
$C_{i/o}$		Input/output terminal capacitance
$C_{i(\phi)}$		Input capacitance of clock input
f		Frequency
$f_{(\phi)}$		Clock frequency
I		Current—the current into an integrated circuit terminal is defined as a positive value and the current out of a terminal is defined as a negative value
I_{BB}		Supply current from V_{BB}
$I_{BB(AV)}$		Average supply current from V_{BB}
I_{CC}		Supply current from Vcc
$I_{CC(AV)}$		Avarage supply current from Vcc
$I_{CC(PD)}$		Power-down supply current from Vcc
I_{DD}		Supply current from V_{DD}
$I_{DD(AV)}$		Average supply current from V_{DD}
I_{GG}		Supply current from V_{GG}
$I_{GG(AV)}$		Average supply current from V_{GG}
I_I		Input current
I_{IH}		High-level input current—the value of the input current when V_{OH} is applied to the input considered
I_{IL}		Low-level input current—the value of the input current when V_{OL} is applied to the input considered
I_{OH}		High-level output current—the value of the output current when V_{OH} is applied to the output considered
I_{OL}		Low-level output current—the value of the output current when V_{OL} is applied to the output considered
I_{OZ}		Off-state (high-impedance state) output current—the current into an output having a three-state capability with input condition so applied that it will establish according to the product specification, the off (high-impedance) state at the output
I_{OZH}		.Off-state (high-impedance state) output current, with high-level voltage applied to the output
I_{OZL}		Off-state (high-impedance state) output current, with low-level voltage applied to the output
I_{OS}		Short-circuit output current
I_{SS}		Supply current from V_{SS}
Pd		Power dissipation
N_{EW}		Number of erase/write cycles
N_{RA}		Number of read access unrefreshed
R_i		Input resistance
R_L		External load resistance
R_{OFF}		Off-state output resistance
R_{ON}		On-state output resistance
t_a		Access time—the time interval between the application of a specified input pulse during a read cycle and the availability of valid data signal at an output
$t_{a(A)}$	$t_{a(AD)}$	Address access time—the time interval between the application of an address input pulse and the availability of valid data signals at an output
$t_{a(CAS)}$		Column address strobe access time
$t_{a(E)}$	$t_{a(CE)}$	Chip enable access time
$t_{a(G)}$	$t_{a(OE)}$	Output enable access time
$t_{a(PR)}$		Data access time after program
$t_{a(RAS)}$		Row address strobe access time
$t_{a(S)}$	$t_{a(CS)}$	Chip select access time
t_C		Cycle time
t_{CR}	$t_{C(RD)}$	Read cycle time—the time interval between the start of a read cylce and the start of the next cycle
t_{CRF}	$t_{C(REF)}$	Refresh cycle time—the time interval between successive signals that are intended to restore the level in a dynamic memory cell to its original level
t_{CPG}	$t_{C(PG)}$	Page-mode cycle time
t_{CRMW}	$t_{C(RMR)}$	Read-modify-write cycle time—the time interval between teh start of a cycle in which the memory is read and new data is entered, and the start of the next cycle
t_{CW}	$t_{C(WR)}$	Write cycle time—the time interval between the start of a write cycle and the start of the next cycle

New symbol	Former symbol	Parameter—definition
t_d		Delay time—the time between the specified reference points on two pulses
$t_{d(\phi)}$		Delay time between clock pulses—e.g., symbology, delay time, clock 1 to clock 2 or clock 2 to clock 1
$t_{d(CAS-RAS)}$		Delay time, column address strobe to row address strobe
$t_{d(CAS-W)}$	$t_{d(CAS-WR)}$	Delay time, column address strobe to write
$t_{d(RAS-CAS)}$		Delay time, row address strobe to column address strobe
$t_{d(RAS-W)}$	$t_{d(RAS-WR)}$	Delay time, row address strobe to write
$t_{dis(R-Q)}$	$t_{dis(R-DA)}$	Output disable time after read
$t_{dis(S)}$	$t_{PXZ(CS)}$	Output disable time after chip select
$t_{dis(W)}$	$t_{PXZ(WR)}$	Output disable time after write
t_{DHL}		High-level to low-level delay time } the time interval between specified reference points on the input and on the output pulses, when the output is going to the low (high) level and when the device is driven and loaded by specified networks.
t_{DLH}		Low-level to high-level delay time
$t_{en(A-Q)}$	$t_{PZV(A-DQ)}$	Output enable time after address
$t_{en(R-Q)}$	$t_{PZV(R-DQ)}$	Output enable time after read
$t_{en(S-Q)}$	$t_{PZX(CS-DQ)}$	Output enable time after chip select
t_f		Fall time
t_h		Hold time—the interval time during which a signal at a specified input terminal after an active transition occurs at another specified input terminal
$t_{h(A)}$	$t_{h(AD)}$	Address hold time
$t_{h(A-E)}$	$t_{h(AD-CE)}$	Chip enable hold time after address
$t_{h(A-PR)}$	$t_{h(AD-PRO)}$	Program hold time after address
$t_{h(CAS-CA)}$		Column address hold time after column address strobe
$t_{h(CAS-D)}$	$t_{h(CAS-DA)}$	Data-in hold time after column address strobe
$t_{h(CAS-Q)}$	$t_{h(CAS-OUT)}$	Data-out hold time after column address strobe
$t_{h(CAS-RAS)}$		Row address strobe hold time after column address strobe
$t_{h(CAS-W)}$	$t_{h(CAS-WR)}$	Write hold time after column address strobe
$t_{h(D)}$	$t_{h(DA)}$	Data-in hold time
$t_{h(D-PR)}$	$t_{h(DA-PRO)}$	Program hold time after data-in
$t_{h(E)}$	$t_{h(CE)}$	Chip enable hold time
$t_{h(E-D)}$	$t_{h(CE-DA)}$	Data-in hold time after chip enable
$t_{h(E-G)}$	$t_{h(CE-OE)}$	Output enable hold time after chip enable
$t_{h(R)}$	$t_{h(RD)}$	Read hold time
$t_{h(RAS-CA)}$		Column address hold time after row address strobe
$t_{h(RAS-CAS)}$		Column address strobe hold time after row address strobe
$t_{h(RAS-D)}$	$t_{h(RAS-DA)}$	Data-in hold time after row address strobe
$t_{h(RAS-W)}$	$t_{h(RAS-WR)}$	Write hold time after row address strobe
$t_{h(S)}$	$t_{h(CS)}$	Chip select hold time
$t_{h(W)}$	$t_{h(WR)}$	Write hold time
$t_{h(W-CAS)}$	$t_{h(WR-CAS)}$	Column address strobe hold time after write
$t_{h(W-D)}$	$t_{h(WR-DA)}$	Data-in hold time after write
$t_{h(W-RAS)}$	$t_{h(WR-RAS)}$	Row address hold time after write
t_{PHL}		High-level to low-level propagation time } the time interval between specified reference points on the input and on the output pulses when the output is going to the low (high) level and when the device is driven and loaded by typical devices of stated type
t_{PLH}		Low-level to high-level propagation time
t_r		Rise time
$t_{rec(W)}$	t_{wr}	Write recovery time—the time interval between the termination of a write pulse and the initiation of a new cycle
$t_{rec(PD)}$	$t_{R(PD)}$	Power-down recovery time
t_{su}		Setup time—the time interval between the application of a signal which is maintained at a speciifed input terminal and a consecutive active tarnsition at another specified input terminal
$t_{su(A)}$	$t_{su(AD)}$	Address setup time
$t_{su(A-E)}$	$t_{su(AD-CE)}$	Chip enable setup time before address
$t_{su(A-W)}$	$t_{su(AD-WR)}$	Write setup time before address
$t_{su(CA-RAS)}$		Row address strobe setup time before column address

New symbol	Former symbol	Parameter—definition
$t_{su(D)}$	$t_{su(DA)}$	Data-in setup time
$t_{su(D-E)}$	$t_{su(DA-CE)}$	Chip enable setup time before data-in
$t_{su(D-W)}$	$t_{su(DA-WR)}$	Write setup time before data-in
$t_{su(E)}$	$t_{su(CE)}$	Chip enable setup time
$t_{su(E-P)}$	$t_{su(CE-P)}$	Precharge setup time before chip enable
$t_{su(G-E)}$	$t_{su(OE-CE)}$	Chip enable setup time before output enable
$t_{su(P-E)}$	$t_{su(P-CE)}$	Chip enable setup time before precharge
$t_{su(PD)}$		Power-down setup time
$t_{su(R)}$	$t_{su(RD)}$	Read setup time
$t_{su(R-CAS)}$	$t_{su(RA-CAS)}$	Column address strobe setup time before read
$t_{su(RA-CAS)}$		Column address strobe setup time before row address
$t_{su(S)}$	$t_{su(CS)}$	Chip select setup time
$t_{su(S-W)}$	$t_{su(CS-WR)}$	Write setup time before chip select
$t_{su(W)}$	$t_{su(WR)}$	Write setup time
t_{THL}		High-level to low-level transition time } the time interval between specified reference points on the edge of the output pulse when the output is going to the low (high) level and when a specified input signal is applied through a specified network and the output is loaded by another specified network
t_{TLH}		Low-level- to high-level transition time
$t_{v(A)}$	$t_{dv(AD)}$	Data valid time after address
$t_{v(E)}$	$t_{dv(CE)}$	Data valid time after chip enable
$t_{v(E)PR}$	$t_{v(CE)PR}$	Data valid time after chip enable in program mode
$t_{v(G)}$	$t_{v(OE)}$	Data valid time after output enable
$t_{v(PR)}$		Data valid time after program
$t_{v(S)}$	$t_{v(CS)}$	Data valid time after chip select
t_w		Pulse width (pulse duration) the time interval between specified reference points on the leading and training edges of the waveforms
$t_{w(E)}$	$t_{w(CE)}$	Chip enable pulse width
$t_{w(EH)}$	$t_{w(CEH)}$	Chip enable high pulse width
$t_{w(EL)}$	$t_{w(EL)}$	Chip enable low pulse width
$t_{w(PR)}$		Program pulse width
$t_{w(R)}$	$t_{w(RD)}$	Read pulse width
$t_{w(S)}$	$t_{w(CS)}$	Chip select pulse width
$t_{w(W)}$	$t_{w(WR)}$	Wrtie pulse width
$t_{w(\phi)}$		Clock pulse width
Ta		Ambient temperature
Topr		Operating temperature
Tstg		Storage temperature
V_{BB}		V_{BB} supply voltage
V_{CC}		V_{CC} supply voltage
V_{DD}		V_{DD} supply voltage
V_{GG}		V_{GG} supply voltage
V_I		Input voltage
V_{IH}		High-level input voltage—the value of the permitted high-state voltage at the input
V_{IL}		Low-level input voltage—the value of the permitted low-state voltage at the input
V_O		Output voltage
V_{OH}		High-level output voltage—the value of the guaranteed high-state voltage range at the output
V_{OL}		Low-level output voltage—the value of the guaranteed low-state voltage range at the output
V_{SS}		V_{SS} supply voltage

QUALITY ASSURANCE AND RELIABILITY TESTING

1. PLANNING

In recent years, advances in integrated circuits have been rapid, with increasing density and speed accompanied by decreasing cost. Because of these advances, it is now practical and economically justifiable to use these devices in systems of greater complexity and in which they were previously considered too expensive. All of these advances add up to increased demand.

We at Mitsubishi foresaw this increased demand and organized our production facilities to meet it. We also realized that simply increasing production to meet the demand was not enough and that positive steps would have to be taken to assure the reliability of our products.

This realization resulted in development of our Quality Assurance System. The system has resulted in improved products, and Mitsubishi is able to supply its customers' needs with ICs of high reliability and stable quality. This system is the key to future planning for improved design, production and quality assurance.

2. QUALITY ASSURANCE SYSTEM

The Quality Assurance System imposes quality controls on Mitsubishi products from the initial conception of a new product to the final delivery of the product to the customer. A diagram of the total system is shown in Fig. 1. For ease of understanding, the system is divided into three stages.

2.1 Quality Assurance in the Design Stage

The characteristics of the breadboard devices are carefully checked to assure that all specifications are met. Standard integrated circuits and high-quality discrete components are used. During the design stage, extensive use is made of a sophisticated CAD program, which is updated to always include the latest state-of-the-art techniques.

2.2 Quality Assurance in the Limited-Manufacturing Stage

Rigid controls are maintained on the environment, incoming material and manufacturing equipment such as tools and test equipment. The products and materials used are subjected to stringent tests and inspections as they are manufactured. Wafer production is closely monitored.

Finally, a tough quality assurance test and inspection is made before the product is released for delivery to a customer. This final test includes a complete visual inspection and electrical characteristics tests. A sampling technique is used to conduct tests under severe operating conditions to assure that the products meet reliability specifications.

2.3 Quality Assurance in the Full Production Stage

Full production of a product is not started until it has been confirmed that it can be manufactured to meet quality and reliability specifications. The controls, tests and inspection procedures developed in §2.2 are continued. The closest monitoring assures that they are complied with.

3. RELIABILITY CONTROL

3.1 Reliability Tests

The newly established Reliability Center for Electronic Components of Japan has established a qualification system for electronic components. Reliability test methods and procedures are developed to mainly meet MIL-STD-883 and JIS C 7022 specifications. Details of typical tests used on Mitsubishi ICs are shown in Table 1.

Table 1 Typical reliability test items and conditions

Group	Item	Test condition
1	High temperature operating life	Maximum operating ambient temperature 1000h
	High temperature storage life	Maximum storage temperature 1000h
	Humidity (steady state) life	65°C 95%RH 500h
2	Soldering heat	260°C 10s
	Thermal shock	0～100°C 15 cycles, 10min/cycle
	Temperature cycle	Minimum to maximum storage temperature, 10 cycles of 1h/cycle
3	Soldering	230°C, 5s, use rosin flux
	Lead integrity	Tension: 340g 30s Bending stress: 225g, ±30°, 3 times
	Vibration	20G, X, Y, Z each direction, 4 times 100～2000Hz—4 min/cycle
	Shock	1500G, 0.5ms in X_1, Y_1 and Z_1 direction, 5 times.
	Constant acceleration	20000G, Y_1 direction, 1 min

3.2 Failure Analysis

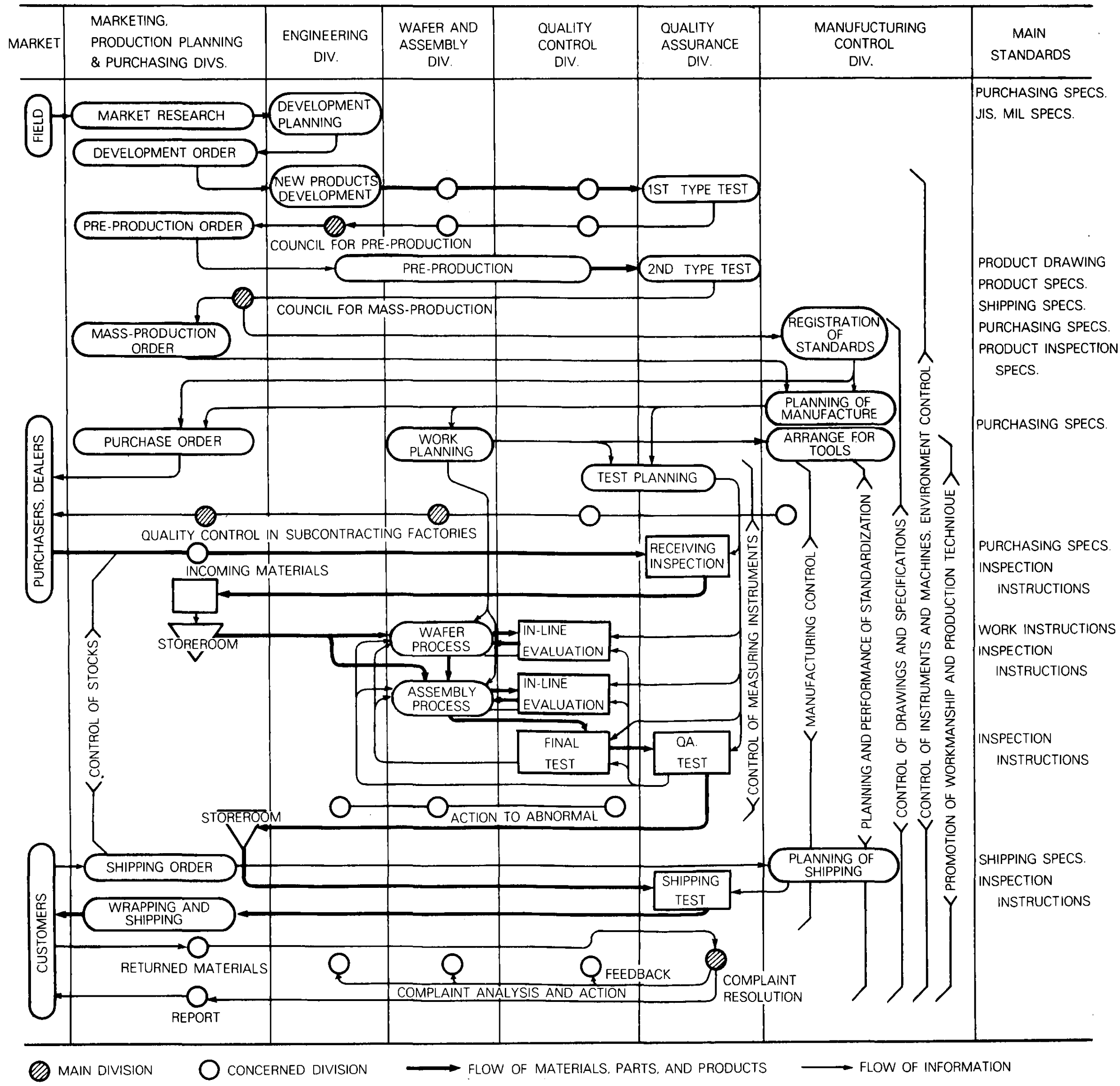
Devices that have failed during reliability or acceleration tests are analyzed to determine the cause of failure. This information is fed back to the process engineering section and manufacturing section so that improvements can be made to increase reliability. A summary of failure analysis procedures is shown in Table 2.

Table 2 Summary of failure analysis procedures

Step	Description
1. External examination	O Inspection of leads, plating, soldering and welding O Inspection of materials, sealing, package and marking O Visual inspection of other items of the specifications O Use of stereo microscopes, metallurgical microscopes, X-ray photographic equipment, fine leakage and gross leakage testers in the examination
2. Electrical tests	O Checking for open circuits, short circuits and parametric degradation by electrical parameter measurement O Observation of characteristics by a synchroscope or a curve tracer and checking of important physical characteristics by electrical characteristics O Stress tests such as environmental or life tests, if required
3. Internal examination	O Removal of the cover of the device, the optical inspection of the internal structure of the device O Checking of the silicon chip surface O Measurement of electrical characteristics by probes, if applicable O Use of SEM, XMA and infrared microscanner if required
4. Chip analysis	O Use of metallurgical analysis techniques to supplement analysis of the internal examination O Slicing for cross-sectional inspection O Analysis of oxide film defects O Analysis of diffusion defects

QUALITY ASSURANCE AND RELIABILITY TESTING

Fig. 1 Quality assurance system



4. TYPICAL RESULTS OF RELIABILITY TESTS AND FAILURE ANALYSES

4.1 Results of Reliability Test

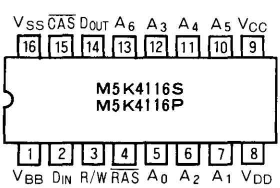
Formerly, sufficient reliability for memory MOS LSIs was obtained by using metal-sealed ceramic packages, but with the development of high-reliability plastic molding technology, production has been shifted to plastic molded memory MOS LSIs.

The following tests are performed:

1. Operating life test: Durability is tested at high temperature under operating state conditions by applying clock pulse inputs as shown in Fig. 2.
2. DC biased test: Durability is tested at high temperature biasing DC voltage, as shown in Fig. 3.
3. High temperature storage: The durability of devices stored at high temperatures is tested.

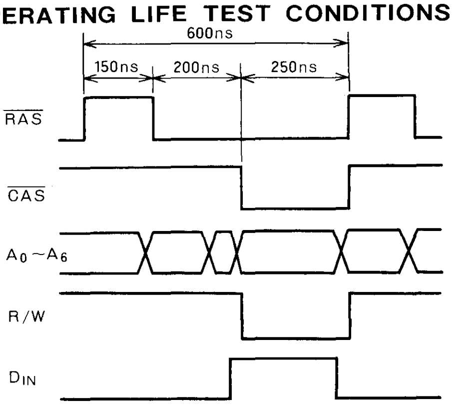
Typical results of memory MOS LSI life tests are shown in Table 3. The failure rate computed from this reliability data using an appropriate acceleration factor is 0.1 FIT or less (1 FIT=10^{-9}/hour) per bit, about the same as, or less than, for core memories.

PIN CONFIGURATION (TOP VIEW)



D_{IN} =DATA IN
$\overline{RAS}$ =ROW ADDRESS STROBE
$\overline{CAS}$ =COLUMN ADDRESS STROBE
R/W =READ/WRITE
D_{OUT} =DATA OUT

OPERATING LIFE TEST CONDITIONS



All addresses including the A_0 to A_6 row addresses and column addresses are cycled through in binary sequence such that the entire 16K bits are accessed.

$V_{BB}=-5.5V$, $V_{SS}=V_{IL}=0V$, $V_{DD}=13.2V$
$V_{CC}=5.5V$, $V_{IH}=5.0V$

Fig. 2 Operating life test procedure (for M5K4116 P, S 16K-bit dynamic RAM)

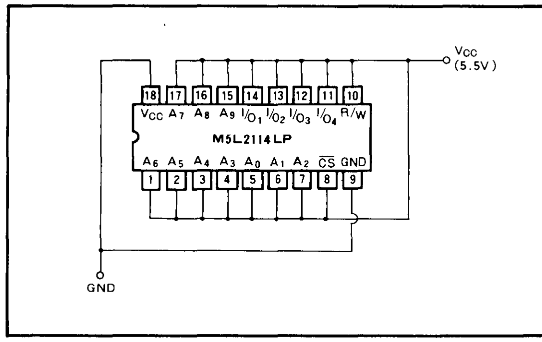


Fig. 3 DC biased test procedure (for M5L2114 LP 4K-bit static RAM)

QUALITY ASSURANCE AND RELIABILTY TESTING

Table 3 Examples of Endurance Test Results

Type No.	Package	Test category		Number of samples	Component hours	Number of failures	Remarks
M5K 4164S	16-pin metal-seald ceramic DIL	Operating life	125°C	350	350,000	1	Functional failure
		High-temperature storage	150°C	150	150,000	0	
M5K 4116P	16-pin plastic-molded DIL	Operating life	125°C	334	334,000	0	
		DC biased	125°C	88	132,000	0	
		High-temperature storage	150°C	132	132,000	0	
M58725P	24-pin plastic-molded DIL	Operating life	125°C	114	114,000	0	
		High-temperature storage	150°C	38	38,000	0	
M5L 2114LP	18-pin plastic-molded DIL	Operating life	125°C	176	198,000	0	
		DC biased	125°C	22	22,000	0	
		High-temperature storage	150°C	88	132,000	0	
M58981P	18-pin plastic-molded DIL	Operating life	125°C	110	110,000	0	
		DC biased	125°C	22	22,000	0	
		High-temperature storage	150°C	44	66,000	0	
M5L 5101LP	22-pin plastic-molded DIL	Operating life	125°C	444	544,000	1	Functional failure
		DC biased	125°C	94	94,000	0	
		High-temperature storage	150°C	94	94,000	0	
M5L 2716K	24-pin metal-sealed ceramic DIL with quartz lid	Operating life	125°C	274	362,000	0	
		High-temperature storage	150°C	66	88,000	0	
M5L 2732K	24-pin metal-sealed ceramic DIL with quartz lid	Operating life	125°C	264	308,000	0	
		High-temperature storage	150°C	44	66,000	0	

Table 4 Examples of Environmental Test Results

Test category		Test conditions	Type No.	Number of samples	Number of failures	Remarks
Thermal environment	Soldering heat	260°C, 10s	M5K4116P M5L2114LP M5L5101LP	330	0	
	Thermal shock	−40°C~125°C, 10min/cycle, 15 cycles				
	Temperature cycling	−65°C~150°C, 1h/cycle, 100 cycles				
Thermal cycling	Soldering heat	260°C, 10s	M5K4164S	1,000	0	
	Thermal shock	−55°C~125°C, 10min/cycle, 15 cycles				
	Temperature cycling	−65°C~150°C, 1h/cycle, 100 cycles				
Temperature cycling		−65°C~150°C 1h/cycle, 10 cycles	M5K4116P M5L2114LP M5L5101LP	1,500	0	
Mechanical environment	Shock	1,500G, 0.5ms in X_1, Y_1, and Z_1 directions, 3 times	M5K4164S	1,000	0	
	Vibration	20G, 20~2000Hz, in X, Y, and Z directions				
	Constant acceleration	30,000G, Y_1 direction for 1min				

5. CONCLUSION

Mitsubishi Electric's Quality Assurance System is being expanded to provide stronger emphasis on the following points:

1. Establishment of quality and reliability levels that satisfy customers' requirements.
2. Expansion of the reliability tests of wafers and assembly processes for better evaluation, and standardization of circuit and design rules.
3. Establishment of procedures for speeding up the introduction of new technology and improved methods that raise reliability and to improve the accelerated life tests for better failure analysis.
4. Establishment of a system for collecting data on failures in the field, which will then be analyzed to develop improved methods for increasing reliability.

We welcome and appreciate the cooperation of our customers in developing design specifications, establishing quality levels, controlling incoming inspections, developing assembly and adjusting processes and collecting field data. Mitsubishi is anxious to work with its customers to develop ICs of increased reliability that meet their requirements.

PRECAUTIONS IN HANDLING MOS ICs

A MOS transistor has a very thin oxide insulator under the gate electrode on the silicon substrate. It is operated by altering the conductance (g_m) between source and drain to control mobile charges in the channel formed by the applied gate voltage.

If a high voltage were applied to a gate terminal, the insulator-film under the gate electrode could be destroyed, and all Mitsubishi MOS IC/LSIs contain internal protection circuits at each input terminal to prevent this. It is inherently necessary to apply reverse bias to the P-N junctions of a MOS IC/LSI.

Under certain conditions, however, it may be impossible to completely avoid destruction of the thin insulator-film due to the application of unexpectedly high voltage or thermal destruction due to excessive current from a forward biased P-N junction. The following recommendations should be followed in handling MOS devices.

1. KEEPING VOLTAGE AND CURRENT TO EACH TERMINAL BELOW MAXIMUM RATINGS

1. The recommended ranges of operating conditions provide adequate safety margins. Operating within these limits will assure maximum equipment performance and quality.
2. Forward bias should not be applied to any terminal since excessive current may cause thermal destruction.
3. Output terminals should not be connected directly to the power supply. Short-circuiting of a terminal to a power supply having low impedance may cause burn-out of the internal leads or thermal destruction due to excessive current.

2. KEEPING ALL TERMINALS AT THE SAME POTENTIAL DURING TRANSPORT AND STORAGE

When MOS IC/LSIs are not in use, both input and output terminals can be in a very high impedance state so that they are easily subjected to electrostatic induction from AC fields of the surrounding space or from charged objects in their vicinity. For this reason, MOS IC/LSIs should be protected from electrostatic charges while being transported and stored by conductive rubber foam, aluminum foil, shielded boxes or other protective precautions.

3. KEEPING ELECTRICAL EQUIPMENT, WORK TABLES AND OPERATING PERSONNEL AT THE SAME POTENTIAL

1. All electric equipment, work table surfaces and operating personnel should be grounded. Work tables should be covered with copper or aluminum plates of good conductivity, and grounded. One method of grounding personnel, after making sure that there is no potential difference with electrical equipment, is by the use of a wristwatch metallic ring, etc. attached around the wrist and grounded in series with a 1M Ω resistor. Be sure that the grounding meets national regulations on personnel safety.
2. Current leakage from electric equipment must be prevented not only for personnel safety, but also to avert the destruction of MOS IC/LSIs, as described above. Items such as testers, curve-tracers and synchroscopes must be checked for current leakage before being grounded.

4. PRECAUTIONS FOR MOUNTING OF MOS IC/LSIs

1. The printed wiring lines to input and output terminals of MOS IC/LSIs should not be close to or parallel to high-voltage or high-power signal lines. Turning power on while the device is short-circuited, either by a solder bridge made during assembly or by a probe during adjusting and testing, may cause maximum ratings to be exceeded, which may result in the destruction of the device.
2. When input/output, or input and/or output, terminals of MOS IC/LSIs (now open-circuits) are connected, we must consider the possibility of current leakage and take precautions similar to §2 above. To reduce such undesirable trouble, it is recommended that an interface circuit be inserted at the input or output terminal, or a resistor with a resistance that does not exceed the output driving capability of the MOS IC/LSI be inserted between the power supply and the ground.
3. A filter circuit should be inserted in the AC power supply line to absorb surges which can frequently be strong enough to destroy a MOS IC/LSI.
4. Terminal connections should be made as described in the catalog while being careful to meet specifications.
5. Ungrounded metal plates should not be placed near input or output terminals of any MOS IC/LSIs, since destruction of the insulation may result if they become electrostatically charged.
6. Equipment cases should provide shielding from electrostatic charges for more reliable operation. When a plastic case is used, it is desirable to coat the inside of the case with conductive paint and to ground it. This is considered necessary even for battery-operated equipment.

RANDOM-ACCESS MEMORIES

2

16 384-BIT (2048-WORD BY 8-BIT) STATIC RAM

DESCRIPTION

This is a family of 2048-word by 8-bit static RAMs, fabricated with the N-channel silicon-gate MOS process and designed for simple interfacing. These devices operate on a single 5V supply, as does TTL, and are directly TTL-compatible.

The input and output terminals are common, and an $\overline{OE}$ terminal is provided. $\overline{S}$ controls the power-down feature.

FEATURES

- Fast access time:

M58725P	200ns (max)
M58725 P-15	150ns (max)

- Low power dissipation:

Active:	250mW (typ)
Stand by:	25mW (typ)

- Power down by $\overline{S}$
- Single 5V supply voltage (±10% tolerance)
- Requires neither external clock nor refreshing
- All inputs and outputs are directly TTL compatible
- All outputs are three-state, with OR-tie capability
- Easy memory expansion by chip-select ($\overline{S}$) input
- Common data DQ terminals.
- Same pin configuration as M5L2716K 16 384-bit EPROM

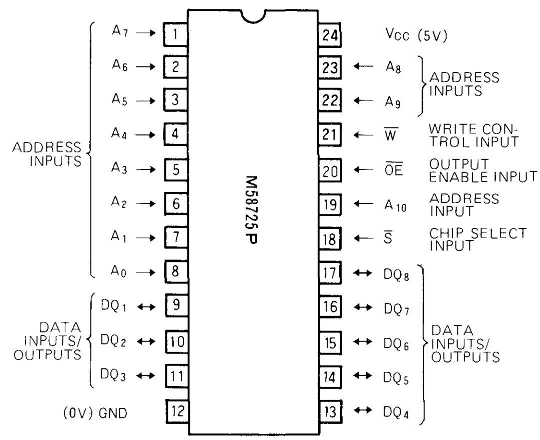
APPLICATION

- Small-capacity memory units

FUNCTION

These devices provide common data input and output terminals. During a write cycle, when a location is designated by address signals $A_0 \sim A_{10}$ the $\overline{OE}$ signal is kept high to keep the DQ terminals in the input mode, signal $\overline{W}$ goes low, and the data of the DQ signal at that time is written.

PIN CONFIGURATION (TOP VIEW)



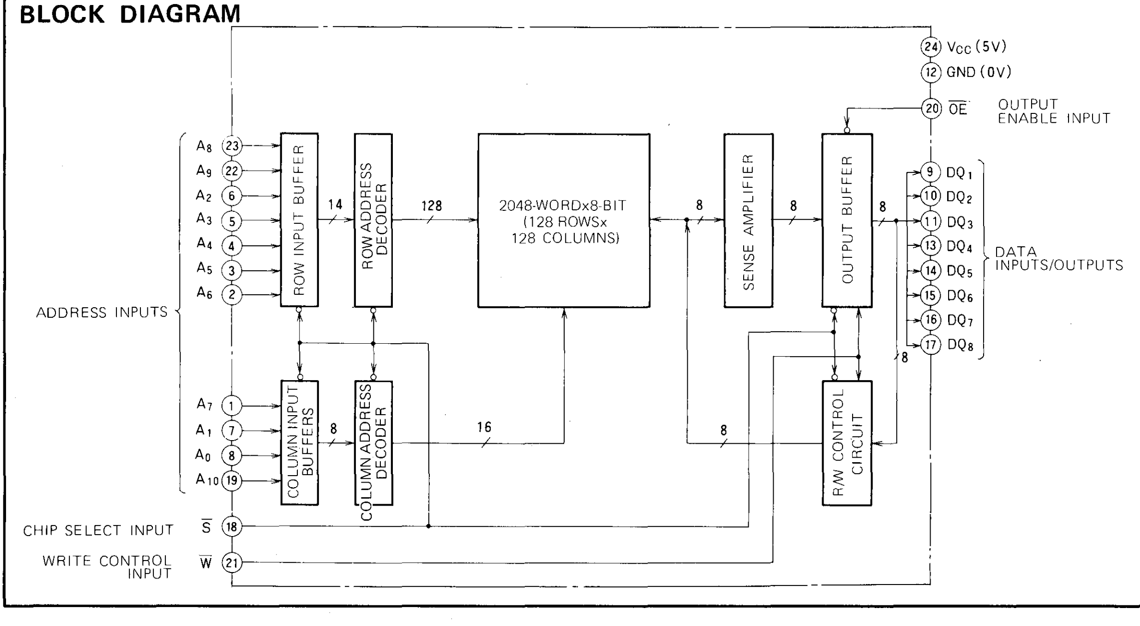
Outline 24P1

During a read cycle, when a location is designated by address signals $A_0 \sim A_{10}$ the $\overline{OE}$ signal is kept low to keep the DQ terminals in the output mode, signal $\overline{W}$ goes high, and the data of the designated address is available at the I/O terminals.

When signal $\overline{S}$ is high, the chip is in the non-selectable state, disabling both reading and writing. In this case the output is in the floating (high-impedance) state, useful for OR-ties with other output terminals.

Signal $\overline{S}$ controls the power down feature. When $\overline{S}$ goes high power dissipation is reduced to 1/10 of active power. The access time from $\overline{S}$ is equivalent to the address access time.

BLOCK DIAGRAM



FUNCTION TABLE

$\overline{S}$	$\overline{OE}$	$\overline{W}$	$DQ_1 \sim DQ_8$	Mode
H	X	X	Hi-Z	Deselect
L	X	L	D_{IN}	Write
L	L	H	D_{OUT}	Read
L	H	H	Hi-Z	—

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Test conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	$-0.5 \sim 7$	V
V_I	Input voltage		$-0.5 \sim 7$	V
V_O	Output voltage		$-0.5 \sim 7$	V
P_d	Maximum power dissipation	Ta = 25°C	1000	mW
Topr	Operating free-air ambient temperature range		$0 \sim 70$	°C
Tstg	Storage temperature range		$-65 \sim 150$	°C

RECOMMENDED OPERATING CONDITIONS (Ta = 0 ~ 70°C, unless otherwise noted.)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{IL}	Low-level input voltage	−1		0.8	V
V_{IH}	High-level input voltage	2		6	V

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 70°C, V_{CC} = 5V ± 10%, unless otherwise noted.)

Symbol	Parameter	Test conditions		Limits			Unit
				Min	Typ	Max	
V_{IH}	High-level input voltage			2		6	V
V_{IL}	Low-level input voltage			−1		0.8	V
V_{OH}	High-level output voltage	$I_{OH} = -1mA$, $V_{CC} = 4.5V$		2.4			V
V_{OL}	Low-level output voltage	$I_{OL} = 3.2mA$				0.4	V
I_I	Input current	$V_I = 0 \sim 5.5V$				10	μA
I_{OZH}	Off-state high-level output current	$V_{I(\overline{S})} = 2V$, $V_O = 2.4V \sim V_{CC}$				10	μA
I_{OZL}	Off-state low-level output current	$V_{I(\overline{S})} = 2V$, $V_O = 0.4V$				−10	μA
I_{CC1}	Supply current from V_{CC}	$V_I = 5.5V$, $V_{I(\overline{S})} = 0.8V$, outputs open	Ta = 25°C		50	80	mA
			Ta = 0°C			90	mA
I_{CC2}	Stand by current	$V_I = 5.5V$, $V_{I(\overline{S})} = 2V$ outputs open	Ta = 25°C		5	10	mA
			Ta = 70°C		7	15	mA
Ci	Input capacitance, all inputs	V_I = GND, Vi = 25mVrms, f = 1MHz			3	5	pF
Co	Output capacitance	V_O = GND, V_O = 25mVrms, f = 1MHz			5	8	pF

Note 1: Current flowing into an IC is positive, out is negative.

SWITCHING CHARACTERISTICS (For Read Cycle) (Ta=0~70°C, V_{CC}=5V ± 10%, unless otherwise noted.)

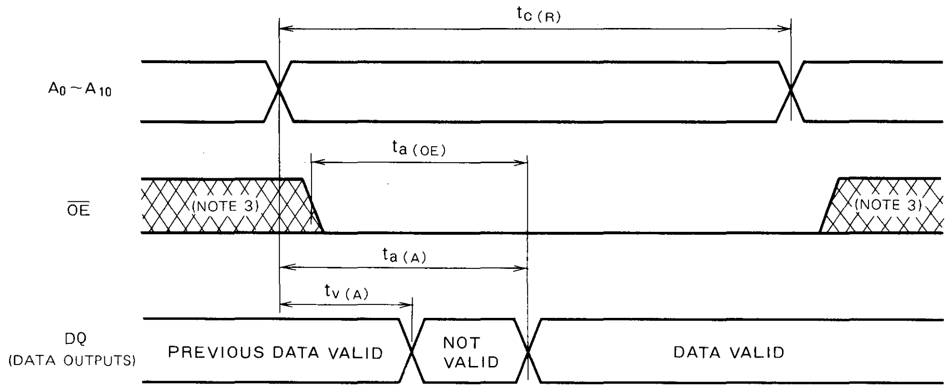
Symbol	Parameter	M58725P-15			M58725P			Unit
		Limits			Limits			
		Min	Typ	Max	Min	Typ	Max	
$t_{c(R)}$	Read cycle time	150			200			ns
$t_{a(A)}$	Address access time			150			200	ns
$t_{a(S)}$	Chip select access time			150			200	ns
$t_{a(OE)}$	Output enable access time			50			60	ns
$t_{v(A)}$	Data valid time after address	20			20			ns
$t_{PXZ(S)}$	Output disable time after chip select			50			60	ns
$t_{PZX(S)}$	Output active time after chip select	10			20			ns
t_{PU}	Power up time after chip selection	0			0			ns
t_{PD}	Power down time after chip deselection			60			80	ns

TIMING REQUIREMENTS (For Write Cycle) (Ta=0~70°C, V_{CC}=5V ± 10%, unless otherwise noted.)

Symbol	Parameter	M58725P-15			M58725P			Unit
		Limits			Limits			
		Min	Typ	Max	Min	Typ	Max	
$t_{c(W)}$	Write cycle time	150			200			ns
$t_{su(S)}$	Chip select setup time	100			120			ns
$t_{su(A)}$	Address setup time	20			20			ns
$t_{w(W)}$	Write pulse width	80			100			ns
t_{wr}	Write recovery time	10			10			ns
$t_{su(OE)}$	Output enable setup time	40			40			ns
$t_{su(D)}$	Data setup time	60			60			ns
$t_{h(D)}$	Data hold time	10			10			ns
$t_{PXZ(OE)}$	Output disable time after output enable			40			40	ns
$t_{PXZ(W)}$	Output disable time after write enable			40			40	ns

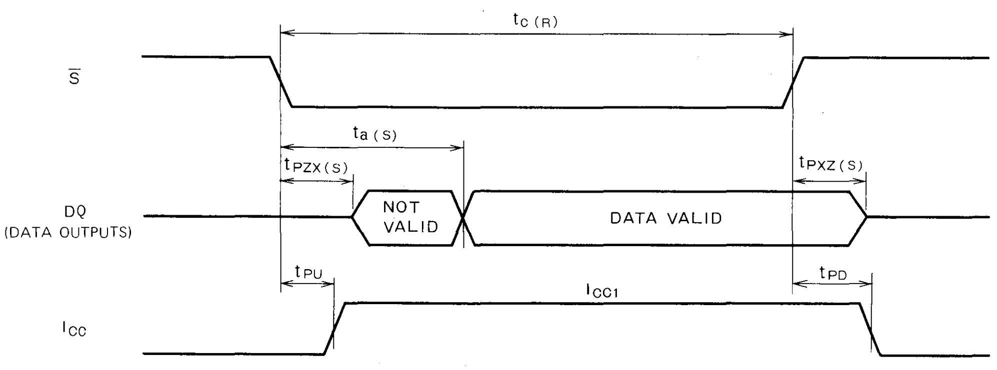
TIMING DIAGRAMS (Note 2)

Read Cycle 1



$\overline{W}$ = high level
$\overline{S}$ = low level

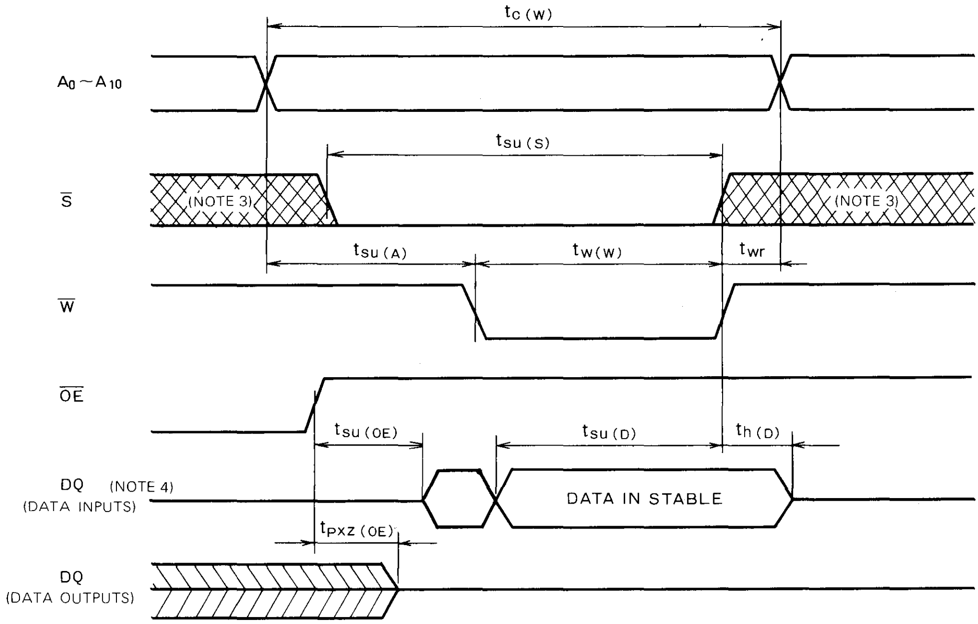
Read Cycle 2



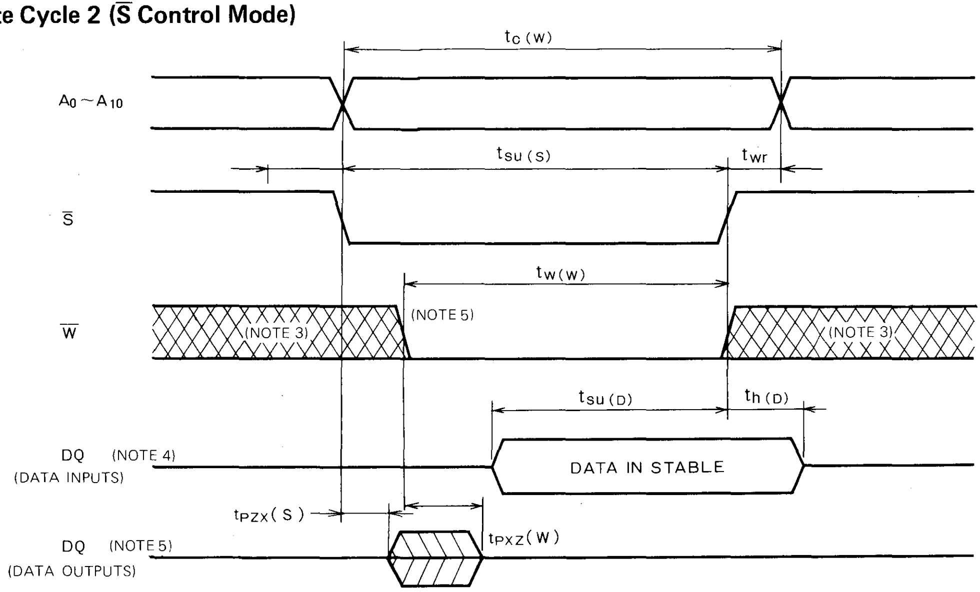
$\overline{W}$ = high level
$\overline{OE}$ = low level

2

Write Cycle ($\overline{W}$ Control Mode)



Write Cycle 2 ($\overline{S}$ Control Mode)



$\overline{OE}$ = low level

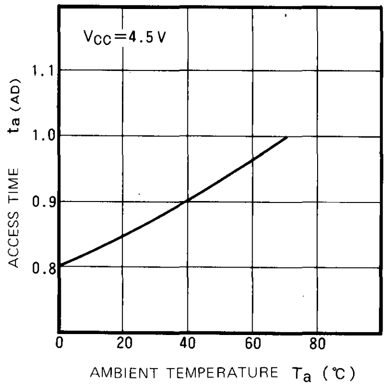
Note 2. Testconditions

Input pulse level	0.4 ~ 2.4V
Input pulse rise time	10ns
Input pulse fall time	10ns
Reference level	1.5 V
Load	1TTL, C_L = 100pF

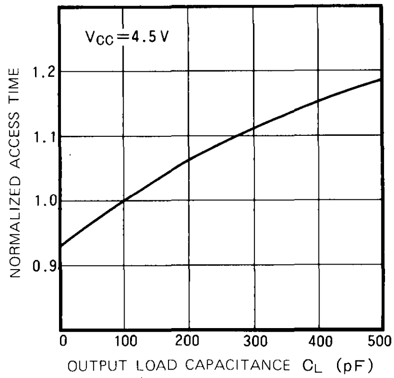
Note 3. Either the high or low state is possible.
4. When the DQ pin is in the output state, a reverse phase signal should not be applied externally.
5. When the falling edge of $\overline{W}$ is simultaneous to or prior to the falling edge of $\overline{S}$ the output is maintained in the high-impedance state.

TYPICAL CHARACTERISTICS

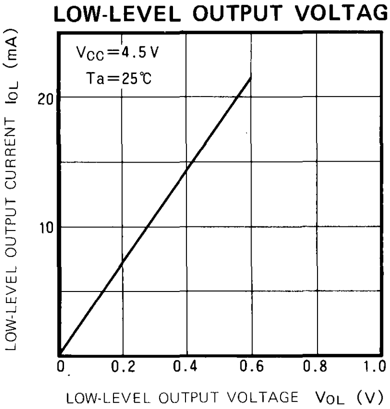
NORMALIZED ACCESS TIME VS. AMBIENT TEMPERATURE



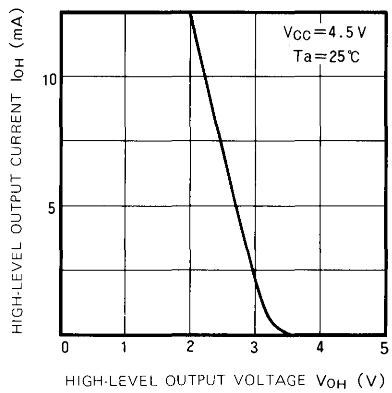
NORMALIZED ACCESS TIME VS. OUTPUT LOAD CAPACITANCE



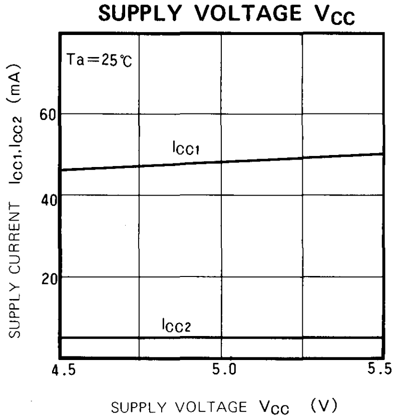
LOW-LEVEL OUTPUT CURRENT VS. LOW-LEVEL OUTPUT VOLTAGE



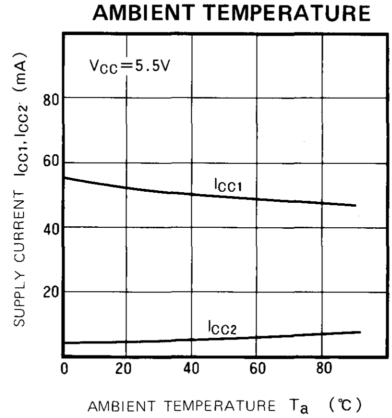
HIGH-LEVEL OUTPUT CURRENT VS. HIGH-LEVEL OUTPUT VOLTAGE



SUPPLY CURRENT VS. SUPPLY VOLTAGE V_{CC}



SUPPLY CURRENT VS. AMBIENT TEMPERATURE



M58981P-30, P-45

4096-BIT (1024-WORD BY 4-BIT) CMOS STATIC RAM

DESCRIPTION

This is a 1024-word by 4-bit static RAM fabricated with the silicon-gate CMOS process and designed for low power dissipation and easy application of battery back-up.

While maintained in the chip non-select state by the chip-select signal $\overline{CS}$, it consumes power only at the low value of 15μA (max) standby current and accordingly is especially suitable as a memory system for battery-operated applications and for battery back-up.

It operates on a single 5V supply, as does TTL, and inputs and outputs are directly TTL-compatible and are provided with common I/O terminals.

FEATURES

- Access time: M58981P-30: 300 ns (max); M58981P-45: 450 ns (max)
- Low power dissipation in the standby mode: 15μA (max)
- Single 5V power supply
- Data holding at 2V supply voltage
- No external clock or refreshing operation required
- Both inputs and outputs are directly TTL-compatible
- Outputs are three-state, with OR-tie capability
- Simple memory expansion by chip-select signal
- Data terminals are common for both inputs and outputs
- Pin configuration is identical with that of Mitsubishi's M5L 2114LP N-channel 4K static RAM, Intel's 2114, and TI's TMS4045

APPLICATION

- Battery-driven or battery back-up small-capacity memory units

FUNCTION

This device provides common data input and output terminals.

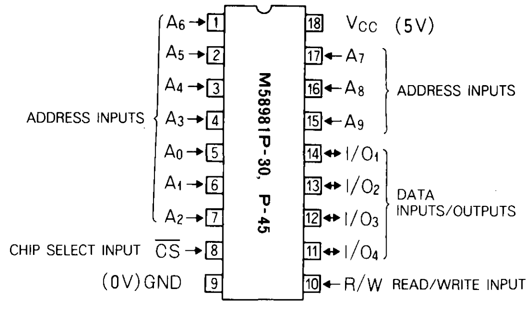
During a write cycle, when a location is designated by address signals $A_0 \sim A_9$ and signal R/W goes low, the data of the I/O at that time is written.

During a read cycle, when a location is designated by address signals $A_0 \sim A_9$, and signal R/W goes high, the data of the designated address is available at the I/O terminals.

When signal $\overline{CS}$ is high, the chip is in the non-selectable state, disabling both reading and writing. In this case, the output is in the floating (high-impedance state) useful for OR-ties with the output terminals of other chips.

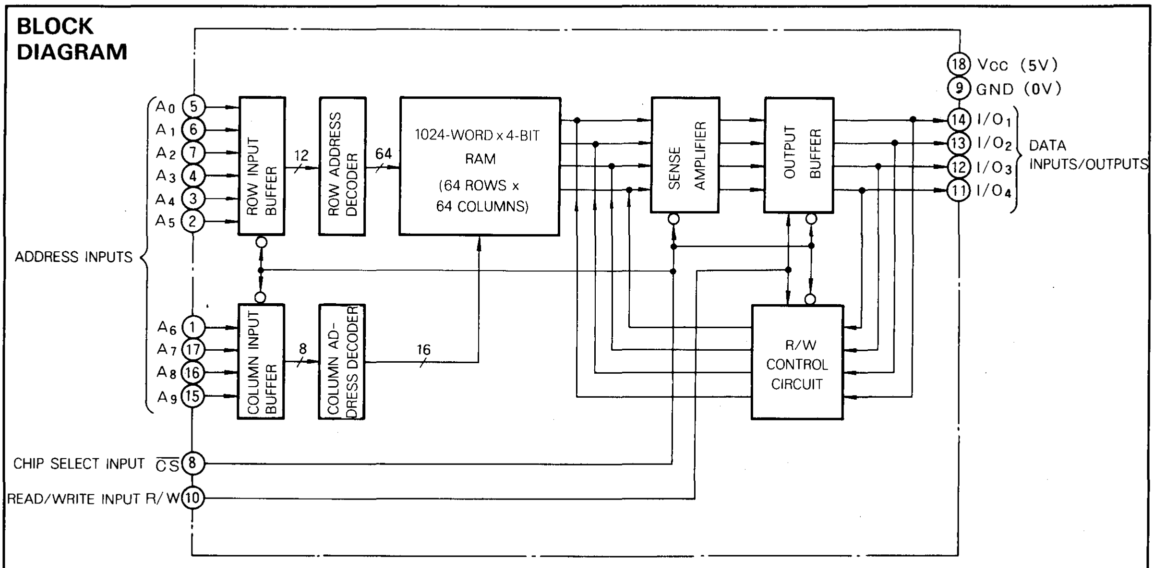
Also in the chip non-select state, the device operates with a low power dissipation, having a standby current of 15μA (max), so that the memory data can be held at a supply voltage of 2V, enabling battery back-up operation during power failure and power-down operation in the standby mode.

PIN CONFIGURATION (TOP VIEW)



Outline 18P4

BLOCK DIAGRAM



4096-BIT (1024-WORD BY 4-BIT) CMOS STATIC RAM

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	−0.3～7	V
V_I	Input voltage		−0.3～V_{CC}+0.3	V
V_O	Output voltage		0～V_{CC}	V
P_d	Maximum power dissipation	Ta＝25℃	700	mW
Topr	Operating free-air ambient temperature range		0～70	℃
Tstg	Storage temperature range		−65～150	℃

RECOMMENDED OPERATING CONDITIONS (Ta＝0～70℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{IL}	Low-level input voltage	−0.3		0.65	V
V_{IH}	High-level input voltage	2.2		V_{CC}	V

ELECTRICAL CHARACTERISTICS (Ta＝0～70℃, V_{CC}＝5V ± 10%, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
V_{IH}	High-level input voltage			2.2		V_{CC}	V
V_{IL}	Low-level input voltage			−0.3		0.65	V
V_{OH}	High-level output voltage		I_{OH}＝−1mA	2.4			V
V_{OL}	Low-level output voltage		I_{OL}＝2mA			0.4	V
I_I	Input current		V_I＝0～5.5V			±1	μA
I_{OZH}	Off-state high-level output current		$V_{I(\overline{CS})}$＝2.2V, V_O＝2.4V～V_{CC}			1	μA
I_{OZL}	Off-state low-level output current		$V_{I(\overline{CS})}$＝2.2V, V_O＝0.4V			−1	μA
I_{CC1}	Supply current from V_{CC}	M58981P-45	$\overline{CS}$≦0.65V, other inputs＝V_{CC},		9	25	mA
		M58981P-30	Output open		12	25	mA
I_{CC2}	Supply current from V_{CC}	M58981P-45	$\overline{CS}$≦0.65V, other inputs＝2.2V,		20	40	mA
		M58981P-30	Output open		25	40	mA
I_{CC3}	Supply current from V_{CC}		$V_{I(\overline{CS})}$＝V_{CC}			15	μA
C_i	Input capacitance, all inputs		V_I＝GND, V_i＝25mVrms, f＝1MHz		4	8	pF
C_o	Output capacitance		V_O＝GND, V_o＝25mVrms, f＝1MHz		8	12	pF

Note 1 : Current flowing into an IC is positive; out is negative.

TIMING REQUIREMENTS (For Write Cycle) (Ta＝0～70℃, V_{CC}＝5V ± 10%, unless otherwise noted)

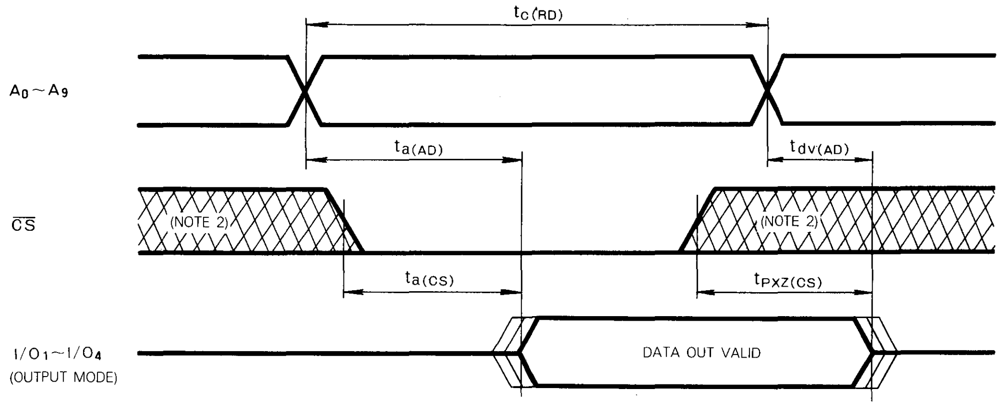
Symbol	Parameter	Test conditions	M58981P-30			M58981P-45			Unit
			Limits			Limits			
			Min	Typ	Max	Min	Typ	Max	
$t_{C(WR)}$	Write cycle time	Input pulse	300			450			ns
$t_{SU(AD)}$	Address setup time with respect to write pulse	V_{IH}＝2.2V	60			80			ns
$t_{W(WR)}$	Write pulse width	V_{IL}＝0.65V	210			250			ns
t_{WR}	Write recovery time	t_r＝t_f＝20ns	30			50			ns
$t_{SU(DA)}$	Data setup time	Reference level＝1.5V	130			150			ns
$t_{h(DA)}$	Data hold time	Load＝1TTL,	30			50			ns
$t_{SU(CS)}$	Chip select setup time	C_L＝100pF	250			300			ns
$t_{PXZ(WR)}$	Output disable time with respect to write pulse				80			100	ns

SWITCHING CHARACTERISTICS (For Read Cycle) (Ta＝0～70℃, V_{CC}＝5V ± 10%, unless otherwise noted)

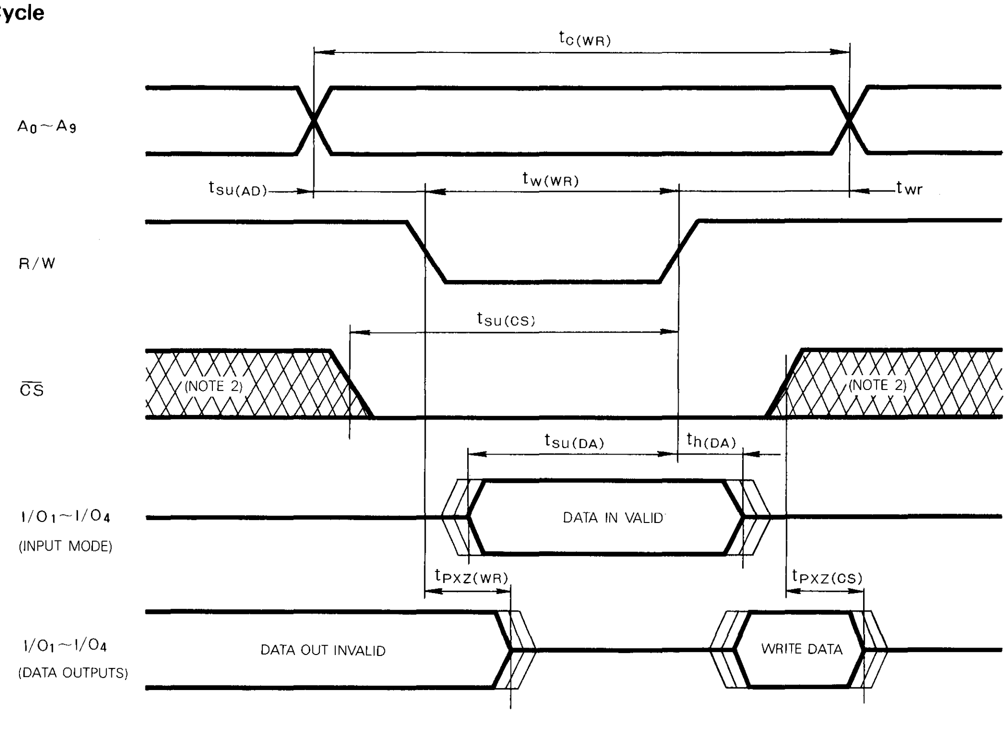
Symbol	Parameter	Test conditions	M58981P-30			M58981P-45			Unit
			Limits			Limits			
			Min	Typ	Max	Min	Typ	Max	
$t_{C(RD)}$	Read cycle time	Input pulse	300			450			ns
$t_{a(AD)}$	Address access time	V_{IH}＝2.2V,			300			450	ns
$t_{a(CS)}$	Chip select access time	V_{IL}＝0.65V, t_r＝t_f＝20ns			300			450	ns
$t_{PXZ(CS)}$	Output disable time with respect to chip select	Reference level＝1.5V, Load＝1TTL,			80			100	ns
$t_{dv(AD)}$	Data valid time with respect to address	C_L＝100pF	20			20			ns

TIMING DIAGRAMS

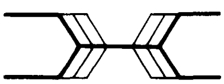
Read Cycle



Write Cycle



Note 2: Hatching indicates the state is unknown.



The center line indicates a floating (high-impedance) state.

POWER-DOWN OPERATION

Electrical Characteristics (Ta = 0 ~ 70°C. unless otherwise noted)

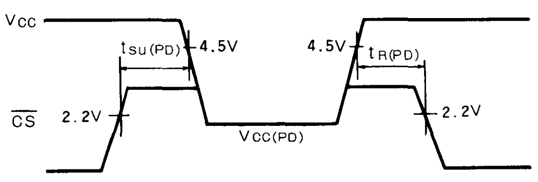
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$V_{CC(PD)}$	Power-down supply voltage	$V_{I(CS)} = V_{CC}$	2			V
$V_{I(\overline{CS})}$	Power-down chip select input voltage	$2.2V \leqq V_{CC(PD)} \leqq V_{CC}$	2.2			V
		$2V \leqq V_{CC(PD)} \leqq 2.2V$		$V_{CC(PD)}$		V
$I_{CC(PD)}$	Power-down supply current from V_{CC}	$V_{CC} = 2V$, all inputs $= 2V$			15	μA

Note 3 : Current flowing into an IC is positive; out is negative.

Timing Requirements (Ta = 0 ~ 70°C, $V_{CC} = 5V \pm 10\%$, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Typ	Max	
$t_{su(PD)}$	Power-down setup time	0			ns
$t_{R(PD)}$	Power-down recovery time	$t_{C(RD)}$			ns

Timing Diagram



16 384-BIT (16 384-WORD BY 1-BIT) DYNAMIC RAM

DESCRIPTION

This is a family of 16 384-word by 1-bit dynamic RAMs, fabricated with the N-channel silicon-gate MOS process, and is ideal for large-capacity memory systems where high speed, low power dissipation, and low costs are essential. The use of double-layer poly-silicon process technology and a single-transistor dynamic storage cell provide high circuit density at reduced costs, and the use of dynamic circuitry including sense amplifiers assures low power dissipation. Multiplexed address inputs permit both a reduction in pins to the standard 16-pin package configuration and an increase in system densities.

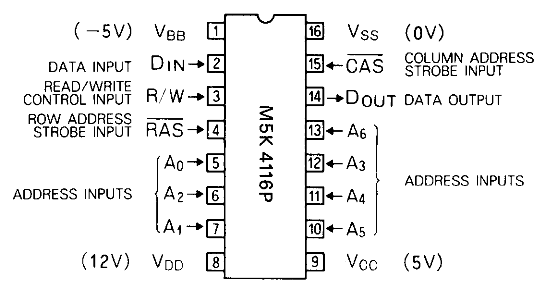
FEATURES

- Performance ranges

Type name	Access time (max) (ns)	Cycle time (min) (ns)	Power dissipation (typ) (mW)
M5K 4116 P-2	150	320	330
M5K 4116 P-3	200	375	280

- Standard 16-pin package
- Voltage range on all power supplies (V_{DD}, V_{CC}, V_{BB}): ±10%
- Low standby power dissipation: 19.8mW (max)
- Low operating power dissipation: 462mW (max)
- Unlatched output enables two-dimensional chip selection and extended page boundary
- Early-write operation gives common I/O capability
- Read-modify-write, $\overline{RAS}$-only refresh, and page-mode capabilities
- All input terminals have low input capacitance and are directly TTL-compatible
- Output is three-state and directly TTL-compatible
- 128 refresh cycles

PIN CONFIGURATION (TOP VIEW)



Outline 16P1

- Interchangeable with Mostek's MK4116 in both electrical characteristics and pin configuration

APPLICATION

- Main memory unit for computers

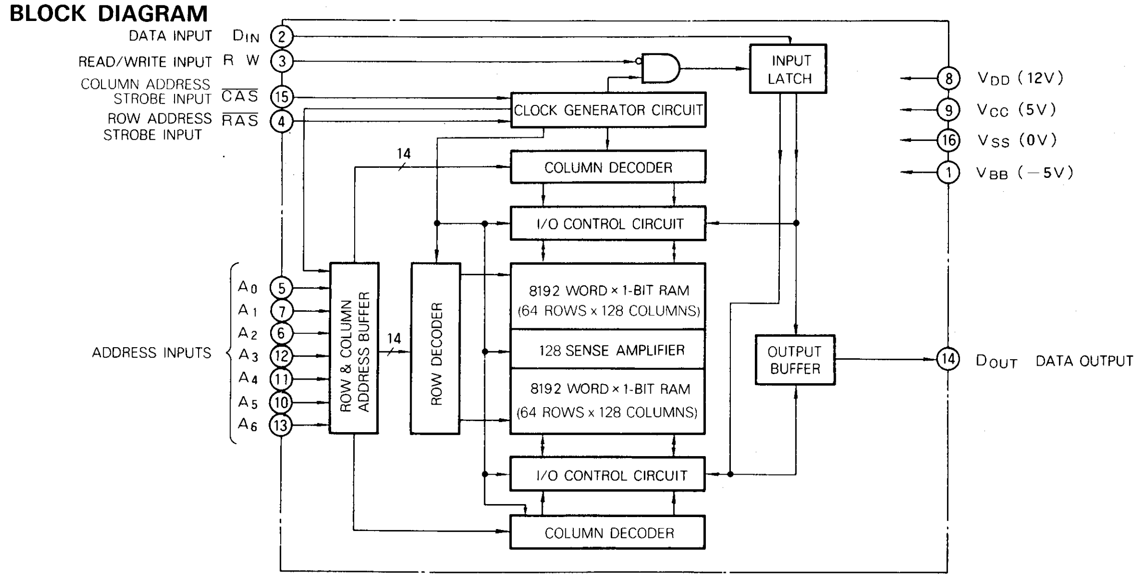
FUNCTION

The M5K4116P provide, in addition to normal read, write, and read-modify-write operations, a number of other functions, e.g., page mode, $\overline{RAS}$-only refresh, and delayed-write. The input conditions for each are shown below.

Operation	Inputs						Output	Refresh	Remarks
	$\overline{RAS}$	$\overline{CAS}$	R/W	D_{IN}	Row address	Column address	D_{OUT}		
Read	ACT	ACT	NAC	DNC	APD	APD	VLD	YES	Page mode identical except refresh is NO.
Write	ACT	ACT	ACT	VLD	APD	APD	OPN	YES	
Read-modify-write	ACT	ACT	ACT	VLD	APD	APD	VLD	YES	
$\overline{RAS}$-only refresh	ACT	NAC	DNC	DNC	APD	DNC	OPN	YES	
Standby	NAC	DNC	DNC	DNC	DNC	DNC	OPN	NO	

Note : ACT : active ; NAC : nonactive ; DNC : don't care ; VLD : valid ; APD : applied ; OPN : open

BLOCK DIAGRAM



16 384-BIT (16 384-WORD BY 1-BIT) DYNAMIC RAM

SUMMARY OF OPERATIONS

Addressing

To select one of the 16 384 memory cells in the M5K 4116 P the 14-bit address signal must be multiplexed into 7 address signals, which are then latched into the on-chip latch by two externally-applied clock pulses. First, the negative-going edge of the row-address-strobe pulse ($\overline{RAS}$) latches the 7 row-address bits; next, the negative-going edge of the column-address-strobe pulse ($\overline{CAS}$) latches the 7 column-address bits. Timing of the $\overline{RAS}$ and $\overline{CAS}$ clocks can be selected by either of the following two methods:

1. The delay time from $\overline{RAS}$ to $\overline{CAS}$ $t_{d(\overline{RAS}-\overline{CAS})}$ is set between the minimum and maximum values of the limits. In this case, the internal $\overline{CAS}$ control signals are inhibited almost until $t_{d(\overline{RAS}-\overline{CAS})max}$ ('gated $\overline{CAS}$' operation). The external $\overline{CAS}$ signal can be applied with a margin not affecting the on-chip circuit operations, e.g. access time, and the address inputs can be easily changed from row address to column address.
2. The delay time $t_{d(\overline{RAS}-\overline{CAS})}$ is set larger than the maximum value of the limits. In this case the internal inhibition of $\overline{CAS}$ has already been released, so that the internal $\overline{CAS}$ control signals are controlled by the externally applied $\overline{CAS}$, which also controls the access time.

Data Input

Data to be written into a selected cell is strobed by the later of the two negative transitions of R/W input and $\overline{CAS}$ input. Thus when the R/W input makes its negative transition prior to $\overline{CAS}$ input (early write), the data input is strobed by $\overline{CAS}$, and the negative transition of $\overline{CAS}$ is set as the reference point for set-up and hold times. In the read-write or read-modify-write cycles, however, when the R/W input makes its negative transition after $\overline{CAS}$, the R/W negative transition is set as the reference point for set-up and hold times.

Data Output Control

The output of the M5K 4116P is in the high-impedance state when $\overline{CAS}$ is high. When the memory cycle in progress is a read, read-modify-write, or a delayed-write cycle, the data output will go from the high-impedance state to the active condition, and the data in the selected cell will be read. This data output will have the same polarity as the input data. Once the output has entered the active condition, this condition will be maintained until $\overline{CAS}$ goes high, irrespective of the condition of $\overline{RAS}$ (for a maximum of 10μs).

The output will remain in the high-impedance state throughout the entire cycle in an early-write cycle.

These output conditions, of the M5K 4116P which can readily be changed by controlling the timing of the write pulse in a write cycle, and the width of the $\overline{CAS}$ pulse in a read cycle, offer capabilities for a number of applications, as follows.

1. Common I/O Operation

If all write operations are performed in the early-write mode, input and output can be connected directly to give a common I/O data bus.

2. Data Output Hold

The data output can be held between read cycles, without lengthening the cycle time, until the next cycle commences. This enables extremely flexible clock-timing settings for $\overline{RAS}$ and $\overline{CAS}$.

3. Two Methods of Chip Selection

Since the output is not latched, $\overline{CAS}$ is not required to keep the outputs of selected chips in the matrix in a high-impedance state. This means that $\overline{CAS}$ and/or $\overline{RAS}$ can both be decoded for chip selection.

4. Extended-Page Boundary

By decoding $\overline{CAS}$, the page boundary can be extended beyond the 128 column locations in a single chip. In this case, $\overline{RAS}$ must be applied to all devices.

Page-Mode Operation

This operation allows for multiple-column addressing at the same row address, and eliminates the power dissipation associated with the negative-going edge of $\overline{RAS}$, because once the row address has been strobed, $\overline{RAS}$ is maintained. Also, the time required to strobe in the row address for the second and subsequent cycles is eliminated, thereby decreasing the access and cycle times.

Refresh

The refreshing of the dynamic cell matrix is accomplished by performing a memory operation at each of the 128 row-address locations within a 2ms time interval. Any normal memory cycle will perform the refreshing, and $\overline{RAS}$-only refresh offers a significant reduction in operating power.

Power Dissipation

Most of the circuitry in the M5K 4116P is dynamic, and most of the power is dissipated when addresses are strobed. Both $\overline{RAS}$ and $\overline{CAS}$ are decoded and applied to the M5K 4116P as chip-select in the memory system, but if RAS is decoded, all unselected devices go into stand-by independent of the $\overline{CAS}$ condition, minimizing system power dissipation.

Power Supplies

Although the M5K 4116P require no particular power-supply sequencing so long as the devices are used within the limits of the absolute maximum ratings, it is recommended that the V_{BB} supply be applied first and removed last. V_{BB} should never be more positive than V_{SS} when power supply is applied to V_{DD}.

Some eight dummy cycles are necessary after power is applied to the device before memory operation is achieved.

16 384-BIT (16 384-WORD BY 1-BIT) DYNAMIC RAM

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to	$-0.5 \sim 20$	V
V_{CC}	Supply voltage		$-0.5 \sim 20$	V
V_{SS}	Supply voltage		$-0.5 \sim 20$	V
V_I	Input voltage		$-0.5 \sim 20$	V
V_O	Output voltage		$-0.5 \sim 20$	V
V_{DD}	Supply voltage	With respect to V_{SS}	$-1 \sim 15$	V
V_{CC}	Supply voltage		$-1 \sim 15$	V
$V_{BB}-V_{SS}$	Supply voltage	$V_{DD}-V_{SS}>0$	0	V
I_O	Output current		50	mA
P_d	Power dissipation	$T_a=25$ ℃	700	mW
Topr	Operating free-air temperature range		$0 \sim 70$	℃
Tstg	Storage temperature range		$-65 \sim 150$	℃

RECOMMENDED OPERATING CONDITIONS (Ta = 0 ~ 70 ℃, unless otherwise noted. Note 1)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	10.8	12	13.2	V
V_{CC}	Supply voltage (Note 2)	4.5	5	5.5	V
V_{SS}	Supply voltage	0	0	0	V
V_{BB}	Supply voltage	-4.5	-5	-5.7	V
V_{IH1}	High-level input voltage, $\overline{RAS}$, $\overline{CAS}$, R W	2.7		7	V
V_{IH2}	High-level input voltage, $A_0 \sim A_6$, D_{IN}	2.4		7	V
V_{IL}	Low-level input voltage, all inputs	-1		0.8	V

Note 1 All voltages with respect to V_{SS}. Apply V_{BB} power supply first, prior to other power supplies, and remove last.
2 The output voltage will swing from V_{SS} to V_{CC} when output loading current is zero. In standby mode V_{CC} may be reduced to V_{SS} without affecting refresh operations or data retention, but the V_{OH}min specification is not guaranteed in this mode.

ELECTRICAL CHARACTERISTICS

($T_a=0 \sim 70$ ℃, $V_{DD}=12V \pm 10\%$, $V_{CC}=5V \pm 10\%$, $V_{SS}=0V$, $-5.7V \leq V_{BB} \leq -4.5V$, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage (Note 2)	$I_{OH}=-5$ mA	2.4		V_{CC}	V
V_{OL}	Low-level output voltage (Note 2)	$I_{OL}=4.2$ mA	0		0.4	V
I_{OZ}	Off-state output current	D_{OUT} floating $0V \leq V_{OUT} \leq 5.5V$	-10		10	μA
I_I	Input current	$V_{BB}=-5V$, $0V \leq V_{IN} \leq 7V$ All other pins = 0V	-10		10	μA
$I_{DD1(AV)}$	Average supply current from V_{DD}, operating	$\overline{RAS}$, $\overline{CAS}$ cycling $t_{C(RD)}=t_{C(WR)}=$ min			35	mA
$I_{CC1(AV)}$	Average supply current from , operating (Note 4)					—
$I_{BB1(AV)}$	Average supply current from V_{BB}, operating				200	μA
I_{DD2}	Supply current from V_{DD}, standby	$\overline{RAS}=V_{IH}$ $D_{OUT}=$ floating			1.5	mA
I_{CC2}	Supply current from V_{CC}, standby		-10		10	μA
I_{BB2}	Supply current from V_{BB}, standby				100	μA
$I_{DD3(AV)}$	Average supply current from V_{DD}, refreshing	$\overline{RAS}$ cycling $\overline{CAS}=V_{IH}$ $t_{C(REF)}=$ min			27	mA
$I_{CC3(AV)}$	Average supply current from V_{CC}, refreshing		-10		10	μA
$I_{BB3(AV)}$	Average supply current from V_{BB}, refreshing				200	μA
$I_{DD4(AV)}$	Average supply current from V_{DD}, page mode	$\overline{RAS}=V_{IL}$, $\overline{CAS}$ $t_{C(PG)}=$ min			27	mA
$I_{CC4(AV)}$	Average supply current from V_{CC}, page mode (Note 4)					—
$I_{BB4(AV)}$	Average supply current from V_{BB}, page mode				200	μA
$C_{i(AD)}$	Input capacitance, address inputs	$V_I=V_{SS}$ $f=1MHz$ $V_i=25mVrms$			5	pF
$C_{i(DA)}$	Input capacitance, data input				5	pF
$C_{i(R W)}$	Input capacitance, read/write control input				7	pF
$C_{i(\overline{RAS})}$	Input capacitance, $\overline{RAS}$ input				10	pF
$C_{i(\overline{CAS})}$	Input capacitance, $\overline{CAS}$ input				10	pF
C_o	Output capacitance	$V_O=V_{SS}$, $f=1MHz$, $V_i=25mVrms$			7	pF

Note 3 Except for I_{BB}, current flowing into an IC is positive ; out is negative.
4 V_{CC} is connected only to the output buffer, so that I_{CC1} and I_{CC4} depend upon output loading.

TIMING REQUIREMENTS (For Read, Write, Read-Modify-Write, Refresh, and Page-Mode Cycles)

(Ta = 0 ~ 70°C, V_{DD} = 12V ± 10%, V_{CC} = 5V ± 10%, V_{SS} = 0V, −5.7V ≤ V_{BB} ≤ −4.5V, unless otherwise noted. See notes 5, 6, and 7.)

Symbol	Parameter	Alternative Symbol	M5K4116P-2 Limits		M5K4116P-3 Limits		Unit
			Min	Max	Min	Max	
$t_{c(REF)}$	Refresh cycle time	t_{REF}		2		2	ms
$t_{w(\overline{RASH})}$	$\overline{RAS}$ high pulse width	t_{RP}	100		120		ns
$t_{w(\overline{RASL})}$	$\overline{RAS}$ low pulse width	t_{RAS}	150	10000	200	10000	ns
$t_{w(\overline{CASL})}$	$\overline{CAS}$ low pulse width (Note 8)	t_{CAS}	100	10000	135	10000	ns
$t_{h(\overline{RAS}-\overline{CAS})}$	$\overline{CAS}$ hold time with respect to $\overline{RAS}$	t_{CSH}	150		200		ns
$t_{h(\overline{CAS}-\overline{RAS})}$	$\overline{RAS}$ hold time with respect to $\overline{CAS}$	t_{RSH}	100		135		ns
$t_{d(\overline{RAS}-\overline{CAS})}$	Delay time, $\overline{RAS}$ to $\overline{CAS}$ (Note 9)	t_{RCD}	20	50	25	65	ns
$t_{d(\overline{CAS}-\overline{RAS})}$	Delay time, $\overline{CAS}$ to $\overline{RAS}$	t_{CRP}	−20		−20		ns
$t_{su(RA-\overline{RAS})}$	Row address setup time with respect to $\overline{RAS}$	t_{ASR}	0		0		ns
$t_{su(CA-\overline{CAS})}$	Column address setup time with respect to $\overline{CAS}$	t_{ASC}	−10		−10		ns
$t_{h(\overline{RAS}-RA)}$	Row address hold time with respect to $\overline{RAS}$	t_{RAH}	20		25		ns
$t_{h(\overline{CAS}-CA)}$	Column address hold time with respect to $\overline{CAS}$	t_{CAH}	45		55		ns
$t_{h(\overline{RAS}-CA)}$	Column address hold time with respect to $\overline{RAS}$	t_{AR}	95		120		ns
t_{THL} t_{TLH}	Transition time	t_{T}	3	35	3	50	ns

Note 5 After power supply is applied, some eight dummy cycles are required before memory operation is achieved. $\overline{RAS}/\overline{CAS}$ refresh cycles or $\overline{RAS}$ read-only cycles are suitable as dummy cycles. Once power is applied, it is also recommended to keep the $\overline{RAS}$ at high-level for more than 3μs before the dummy cycles, or to keep the $\overline{RAS}$ high pulse width $t_{w(\overline{RASH})}$ more than 3μs for a minimum of one dummy cycle.

6 The switching characteristics are defined as $t_{THL} = t_{TLH} = 5ns$.

7 Reference levels of input signals are V_{IH1} min, V_{IH2} min and V_{IL} max. Reference levels for transition time are also between V_{IH1} or V_{IH2} and V_{IL}.

8 Assumes that $t_{d(\overline{RAS}-\overline{CAS})} \geqq t_{d(\overline{RAS}-\overline{CAS})}$ max. If $t_{d(\overline{RAS}-\overline{CAS})} < t_{d(\overline{RAS}-\overline{CAS})}$ max, $t_{w(\overline{CASL})}$ will be increased by the amount that $t_{d(\overline{RAS}-\overline{CAS})}$ has decreased.

9 The maximum value of $t_{d(\overline{RAS}-\overline{CAS})}$ does not define the limit of operation, but is specified as a reference point only; if $t_{d(\overline{RAS}-\overline{CAS})}$ is greater than the specified $t_{d(\overline{RAS}-\overline{CAS})}$ max limit, then access time is controlled exclusively by $t_{a(\overline{CAS})}$.

SWITCHING CHARACTERISTICS (Ta = 0 ~ 70°C, V_{DD} = 12V ± 10%, V_{CC} = 5V ± 10%, V_{SS} = 0V, −5.7V ≤ V_{BB} ≤ −4.5V, unless otherwise noted

Read Cycle

Symbol	Parameter	Alternative Symbol	M5K4116P-2 Limits		M5K4116P-3 Limits		Unit
			Min	Max	Min	Max	
$t_{c(RD)}$	Read cycle time	t_{RC}	320		375		ns
$t_{su(RD-\overline{CAS})}$	Read set-up time with respect to $\overline{CAS}$	t_{RCS}	0		0		ns
$t_{h(\overline{CAS}-RD)}$	Read hold time with respect to $\overline{CAS}$	t_{RCH}	0		0		ns
$t_{h(\overline{CAS}-OUT)}$	Data-out hold time	t_{OFF}	0	40	0	50	ns
$t_{a(\overline{CAS})}$	$\overline{CAS}$ access time (Note 10)	t_{CAC}		100		135	ns
$t_{a(\overline{RAS})}$	$\overline{RAS}$ access time (Note 11)	t_{RAC}		150		200	ns

Note 10 This is the value when $t_{d(\overline{RAS}-\overline{CAS})} \geqq t_{d(\overline{RAS}-\overline{CAS})}$max. Test conditions ; Load = 2TTL, C_L=100pF

11 This is the value when $t_{d(\overline{RAS}-\overline{CAS})} < t_{d(\overline{RAS}-\overline{CAS})}$max. When $t_{d(\overline{RAS}-\overline{CAS})} \geqq t_{d(\overline{RAS}-\overline{CAS})}$max $t_{a(\overline{RAS})}$ increases by the amount of increase of $t_{d(\overline{RAS}-\overline{CAS})}$. Test conditions ; Load = 2TTL, C_L=100pF

Write Cycle

Symbol	Parameter	Alternative Symbol	M5K4116P-2 Limits		M5K4116P-3 Limits		Unit
			Min	Max	Min	Max	
$t_{c(WR)}$	Write cycle time	t_{RC}	320		375		ns
$t_{su(WR-\overline{CAS})}$	Write set-up time with respect to $\overline{CAS}$ (Note 12)	t_{WCS}	−20		−20		ns
$t_{h(\overline{CAS}-WR)}$	Write hold time with respect to $\overline{CAS}$	t_{WCH}	45		55		ns
$t_{h(\overline{RAS}-WR)}$	Write hold time with respect to $\overline{RAS}$	t_{WCR}	95		120		ns
$t_{h(WR-\overline{RAS})}$	$\overline{RAS}$ hold time with respect to write	t_{RWL}	50		70		ns
$t_{h(WR-\overline{CAS})}$	$\overline{CAS}$ hold time with respect to write	t_{CWL}	50		70		ns
$t_{w(WR)}$	Write pulse width	t_{WP}	45		55		ns
$t_{su(DA-\overline{CAS})}$	Data-in setup time with respect to $\overline{CAS}$	t_{DS}	0		0		ns
$t_{h(\overline{CAS}-DA)}$	Data-in hold time with respect to $\overline{CAS}$	t_{DH}	45		55		ns
$t_{h(\overline{RAS}-DA)}$	Data-in hold time with respect to $\overline{RAS}$	t_{DHR}	95		120		ns

Read-Write and Read-Modify-Write Cycles

Symbol	Parameter	Alternative symbol	M5K4116P-2 Limits		M5K4116P-3 Limits		Unit
			Min	Max	Min	Max	
$t_{c(RMW)}$	Read-modify-write cycle time	t_{RWC}	320		405		ns
$t_{c(RW)}$	Read-write cycle time	t_{RWC}	320		375		ns
$t_{h(WR-\overline{RAS})}$	$\overline{RAS}$ hold time with respect to write	t_{RWL}	50		70		ns
$t_{h(WR-\overline{CAS})}$	$\overline{CAS}$ hold time with respect to write	t_{CWL}	50		70		ns
$t_{w(WR)}$	Write pulse width	t_{WP}	45		55		ns
$t_{su(RD-\overline{CAS})}$	Read setup time with respect to $\overline{CAS}$	t_{RCS}	0		0		ns
$t_{d(\overline{RAS}-WR)}$	Delay time, $\overline{RAS}$ to write (Note 12)	t_{RWD}	110		145		ns
$t_{d(\overline{CAS}-WR)}$	Delay time, $\overline{CAS}$ to write (Note 12)	t_{CWD}	60		80		ns
$t_{su(DA-WR)}$	Data-in set-up time with respect to write	t_{DS}	0		0		ns
$t_{h(WR-DA)}$	Data-in hold time with respect to write	t_{DH}	45		55		ns
$t_{h(\overline{CAS}-OUT)}$	Data-out hold time with respect to $\overline{CAS}$	t_{OFF}	0	40	0	50	ns
$t_{a(\overline{CAS})}$	$\overline{CAS}$ access time (Note 10)	t_{CAC}		100		135	ns
$t_{a(\overline{RAS})}$	$\overline{RAS}$ access time (Note 11)	t_{RAC}		150		200	ns

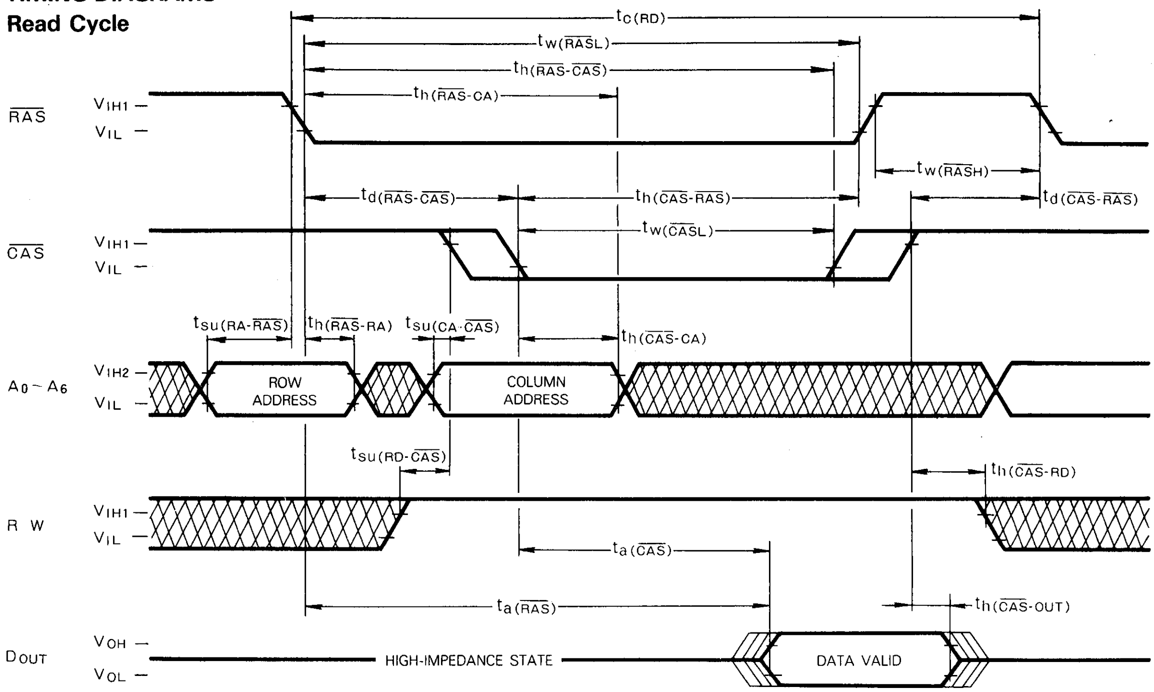
Note 12 : $t_{su(WR-\overline{CAS})}$, $t_{d(\overline{RAS}-WR)}$, and $t_{d(\overline{CAS}-WR)}$ do not define the limits of operation, but are included as electrical characteristics only.
When $t_{su(WR-\overline{CAS})} \geqq t_{su(WR-\overline{CAS})min}$, an early-write cycle is performed, and the data output keeps the high-impedance state.
When $t_{d(\overline{RAS}-WR)} \geqq t_{d(\overline{RAS}-WR)min}$ and $t_{d(\overline{CAS}-WR)} \geqq t_{d(\overline{CAS}-WR)min}$, a read-modify-write cycle is performed, and the data of the selected address will be read out on the data outputs.
For all conditions other than those described above the condition of data output is not defined.

Page-Mode Cycle

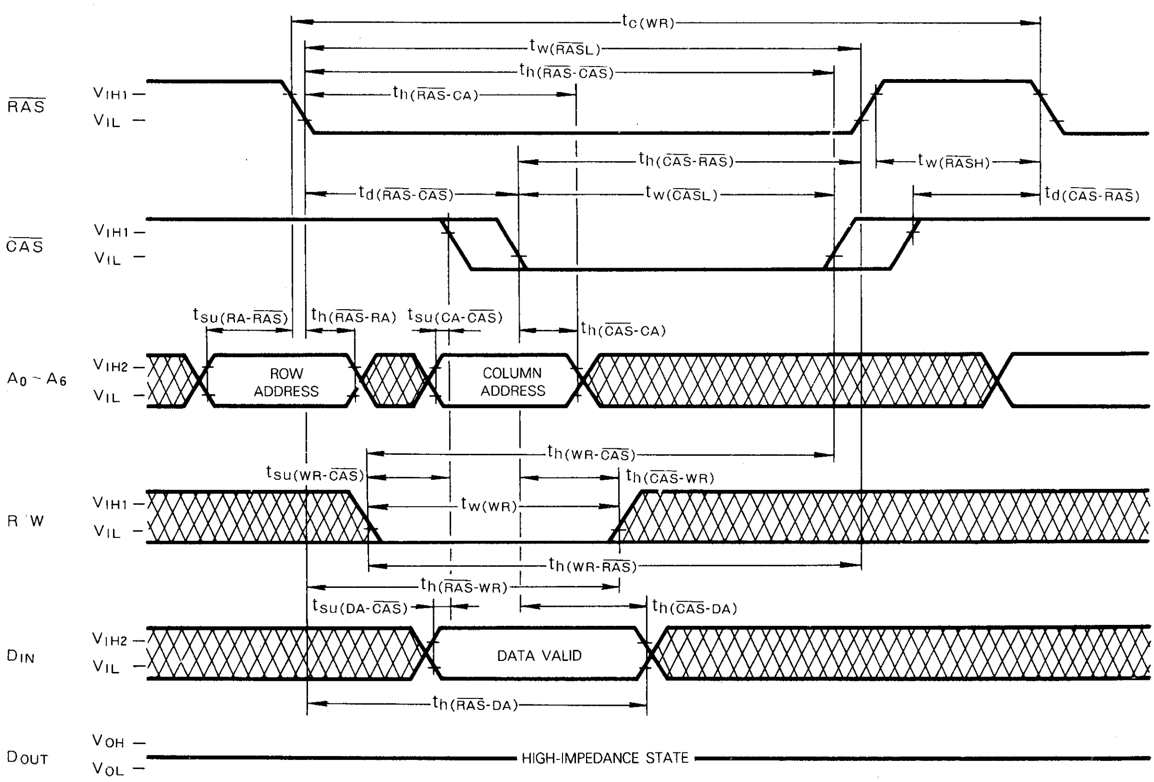
Symbol	Parameter	Alternative symbol	M5K4116P-2 Limits		M5K4116P-3 Limits		Unit
			Min	Max	Min	Max	
$t_{c(PG)}$	Page-mode cycle time	t_{PC}	170		225		ns
$t_{w(\overline{CASH})}$	$\overline{CAS}$ high pulse width	t_{CP}	60		80		ns

TIMING DIAGRAMS

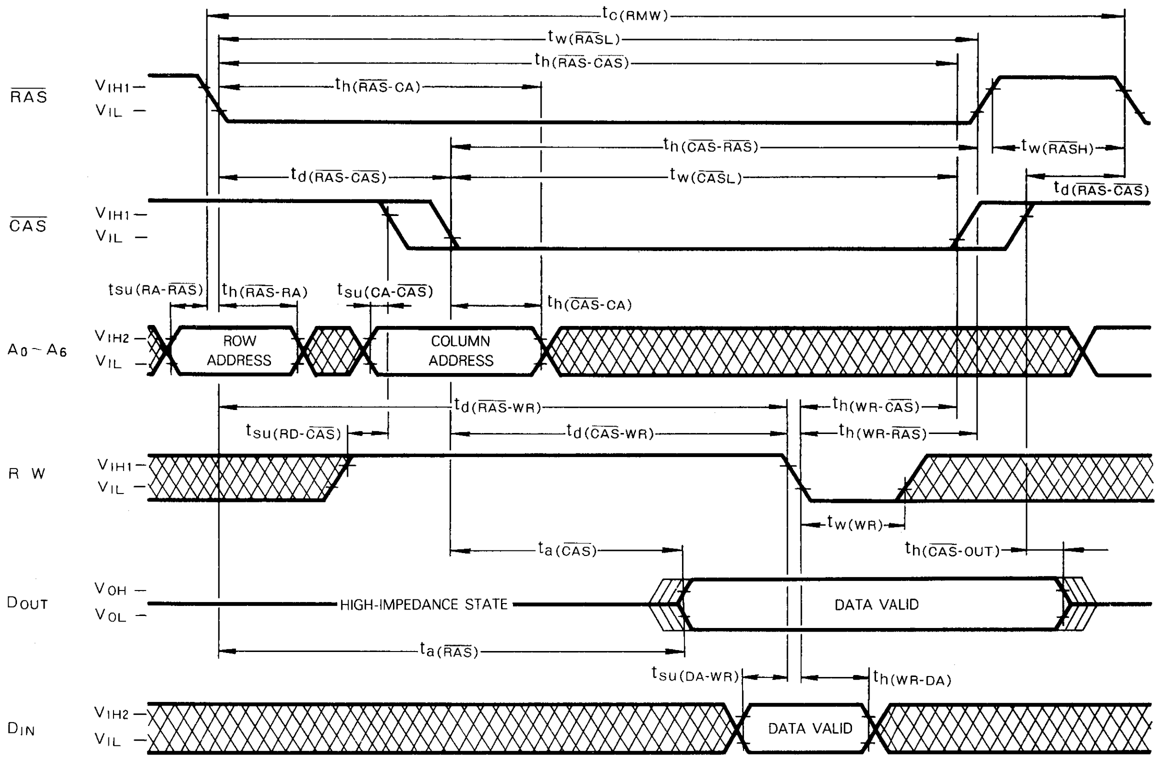
Read Cycle



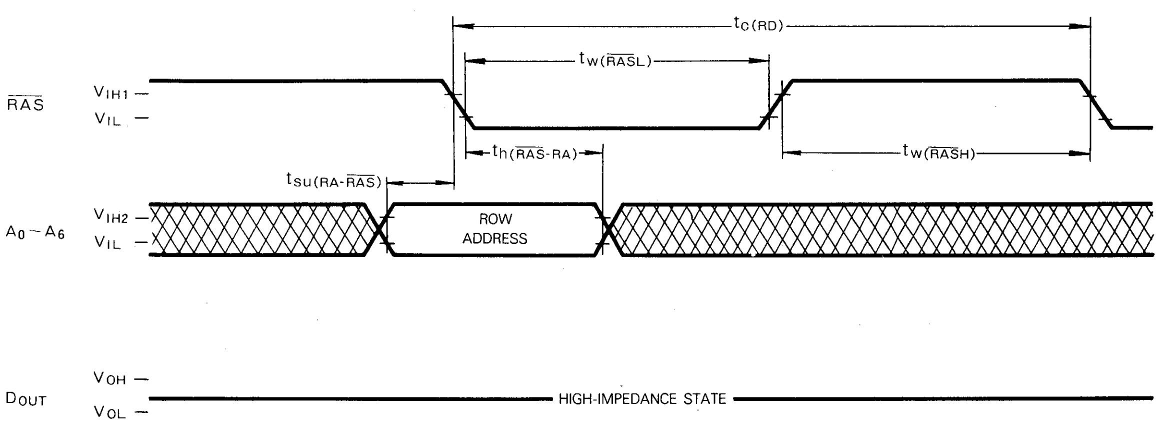
Write and Early Write Cycles



Read-Write and Read-Modify-Write Cycles

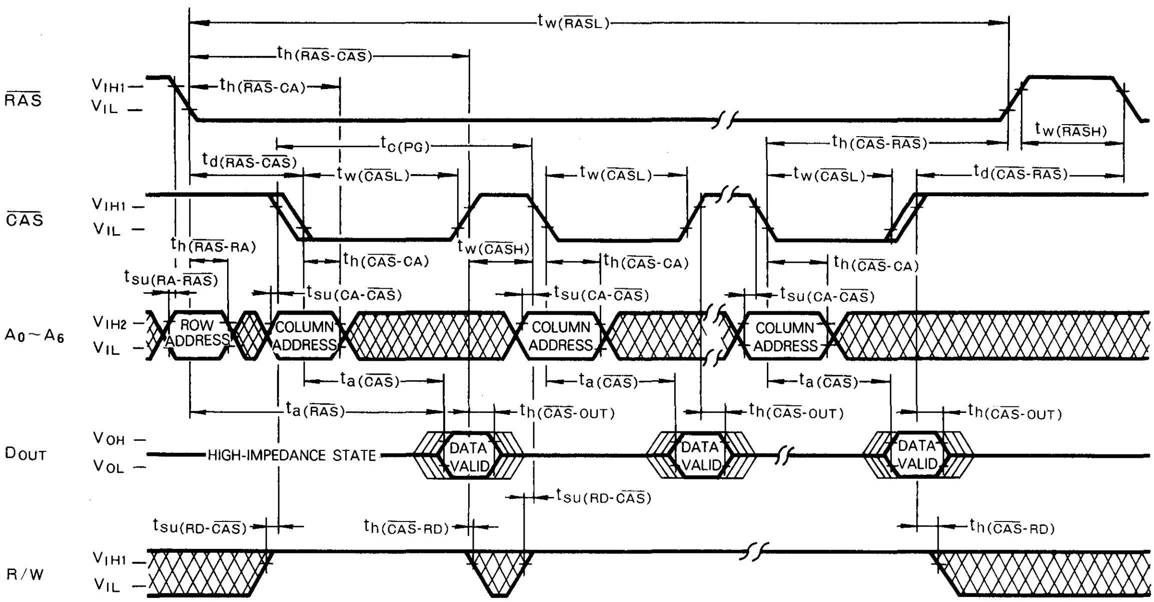


$\overline{RAS}$-Only Refresh Cycle

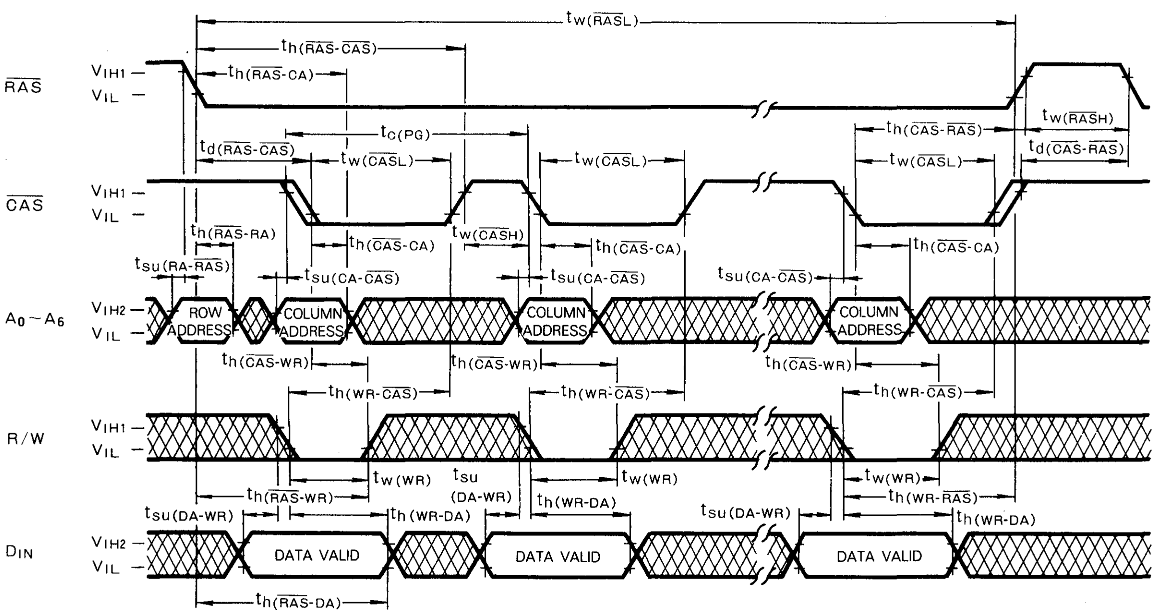


Note 13 : $\overline{CAS} = V_{IH1}$, R / W = don't care.

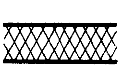
Page-Mode Read Cycle

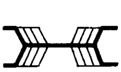


Page-Mode Write Cycle



Note 14 ;

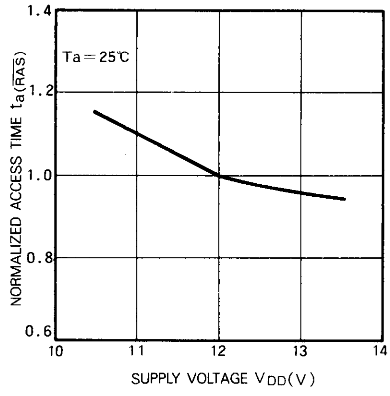

Indicates the don't care input.


The center-line indicates the high-impedance state.

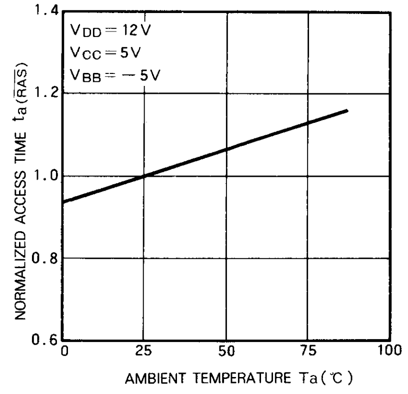
2

TYPICAL CHARACTERISTICS

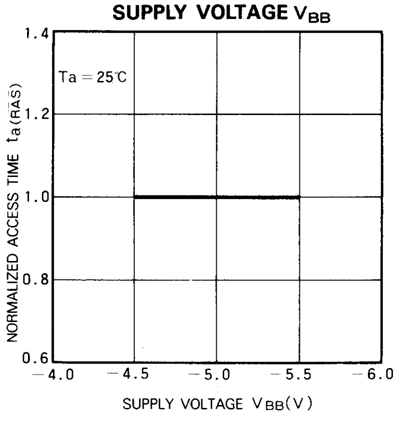
NORMALIZED ACCESS TIME VS. SUPPLY VOLTAGE V_{DD}



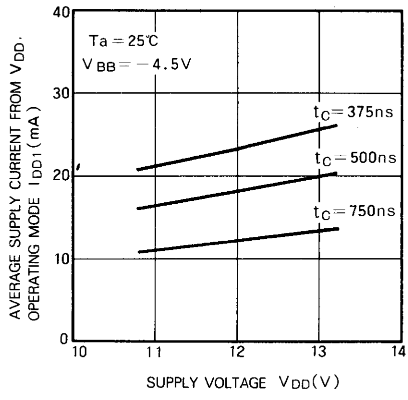
NORMALIZED ACCESS TIME VS. AMBIENT TEMPERATURE



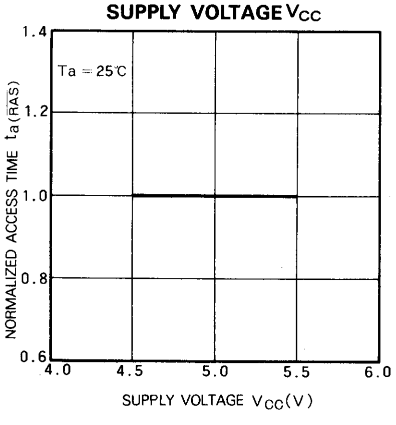
NORMALIZED ACCESS TIME VS. SUPPLY VOLTAGE V_{BB}



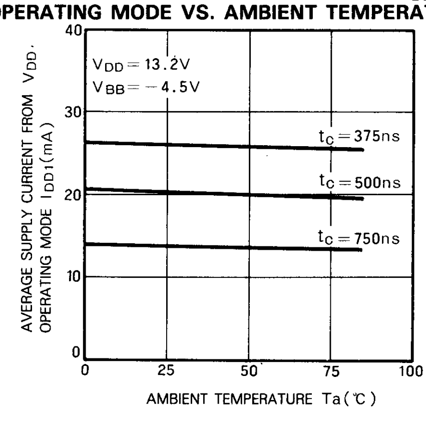
AVERAGE SUPPLY CURRENT FROM V_{DD}, OPERATING MODE VS. SUPPLY VOLTAGE



NORMALIZED ACCESS TIME VS. SUPPLY VOLTAGE V_{CC}

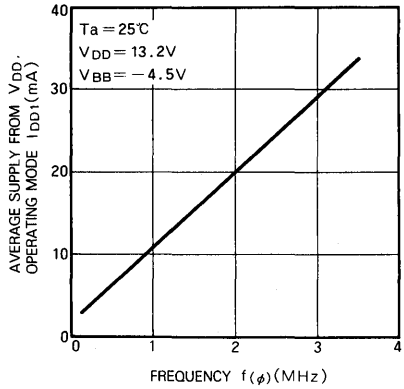


AVERAGE SUPPLY CURRENT FROM V_{DD}, OPERATING MODE VS. AMBIENT TEMPERATURE

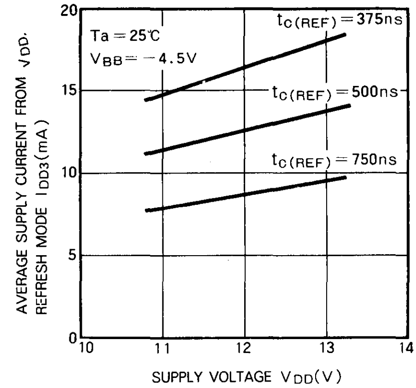


16 384-BIT (16 384-WORD BY 1-BIT) DYNAMIC RAM

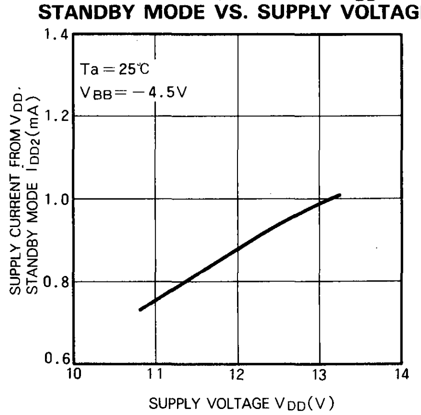
AVERAGE SUPPLY CURRENT FROM V_{DD}, OPERATING MODE VS. FREQUENCY



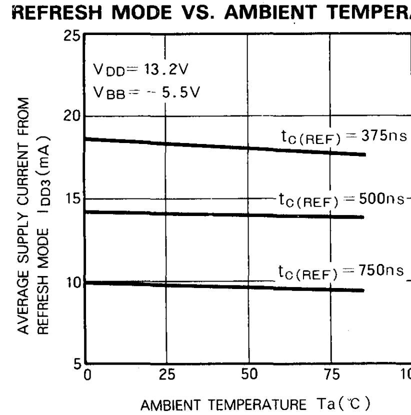
AVERAGE SUPPLY CURRENT FROM V_{DD}, REFRESH MODE VS. SUPPLY VOLTAGE



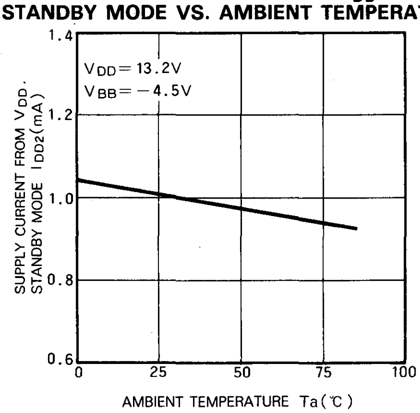
SUPPLY CURRENT FROM V_{DD}, STANDBY MODE VS. SUPPLY VOLTAGE



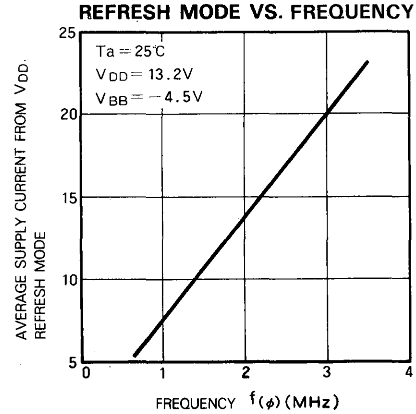
AVERAGE SUPPLY CURRENT FROM V_{DD}, REFRESH MODE VS. AMBIENT TEMPERATURE



SUPPLY CURRENT FROM V_{DD}, STANDBY MODE VS. AMBIENT TEMPERATURE

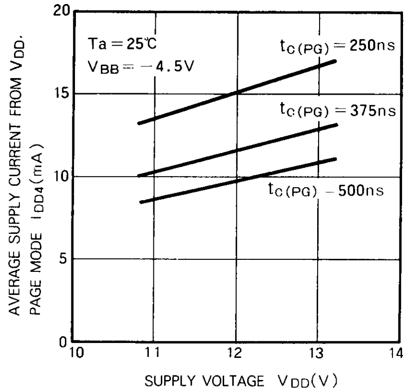


AVERAGE SUPPLY CURRENT FROM V_{DD}, REFRESH MODE VS. FREQUENCY

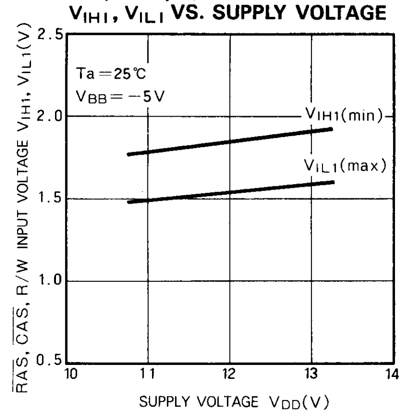


16 384-BIT (16 384-WORD BY 1-BIT) DYNAMIC RAM

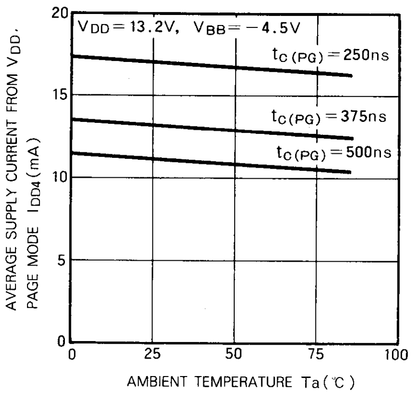
AVERAGE SUPPLY CURRENT FROM V_{DD}, PAGE MODE VS. SUPPLY VOLTAGE



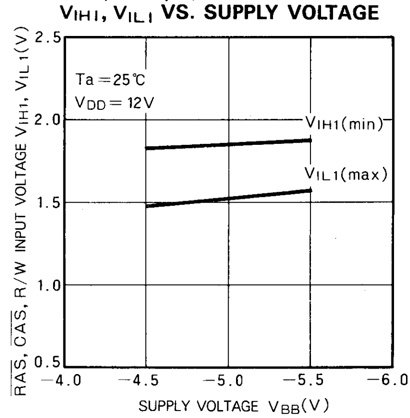
$\overline{RAS}$, $\overline{CAS}$, R/W INPUT VOLTAGE V_{IH1}, V_{IL1} VS. SUPPLY VOLTAGE



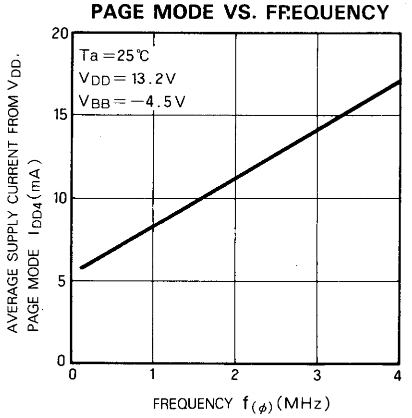
AVERAGE SUPPLY CURRENT FROM V_{DD}, PAGE MODE VS. AMBIENT TEMPERATURE



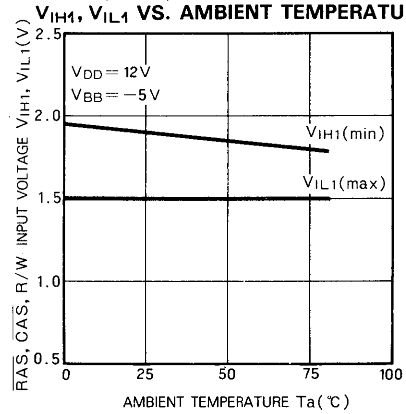
$\overline{RAS}$, $\overline{CAS}$, R/W INPUT VOLTAGE V_{IH1}, V_{IL1} VS. SUPPLY VOLTAGE



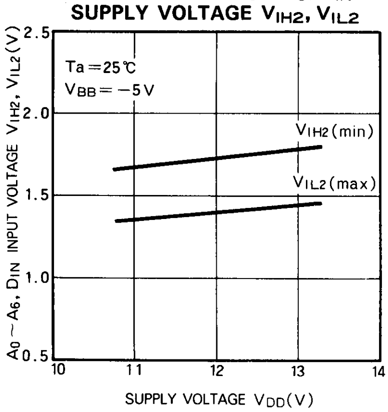
AVERAGE SUPPLY CURRENT FROM V_{DD}, PAGE MODE VS. FREQUENCY



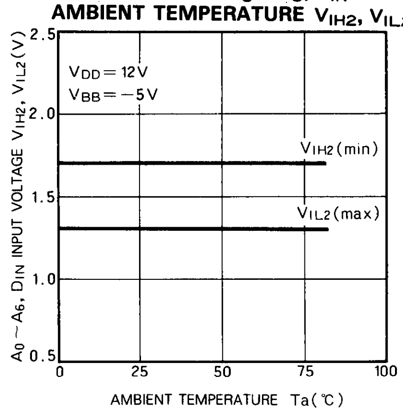
$\overline{RAS}$, $\overline{CAS}$, R/W INPUT VOLTAGE V_{IH1}, V_{IL1} VS. AMBIENT TEMPERATURE



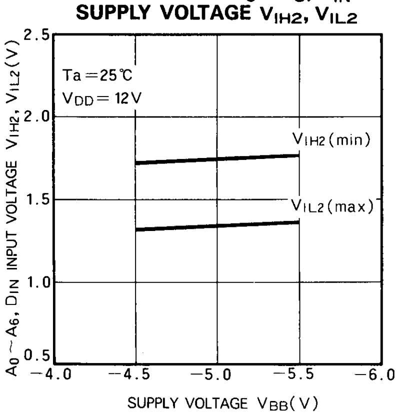
INPUT VOLTAGE $A_0 \sim A_6$, D_{IN} VS. SUPPLY VOLTAGE V_{IH2}, V_{IL2}



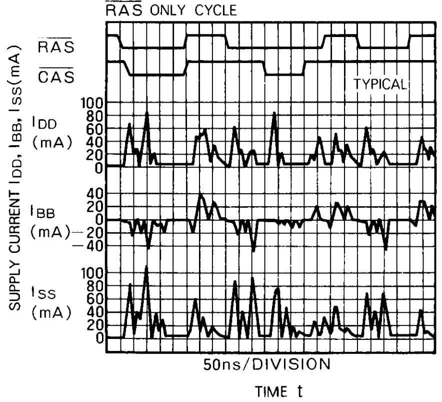
INPUT VOLTAGE $A_0 \sim A_6$, D_{IN} VS. AMBIENT TEMPERATURE V_{IH2}, V_{IL2}



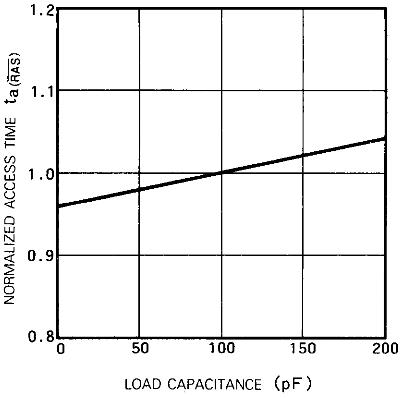
INPUT VOLTAGE $A_0 \sim A_6$, D_{IN} VS. SUPPLY VOLTAGE V_{IH2}, V_{IL2}



SUPPLY CURRENT VS. TIME



NORMALIZED ACCESS TIME VS. LOAD CAPACITANCE

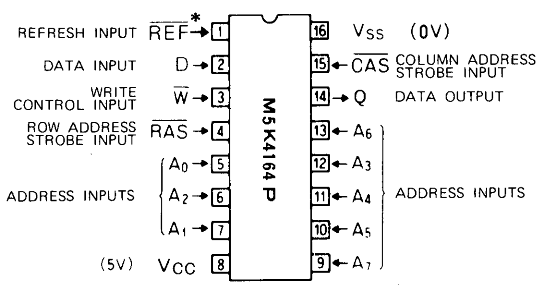


65 536-BIT (65 536-WORD BY 1-BIT) DYNAMIC RAM

DESCRIPTION

This is a family of 65 536-word by 1-bit dynamic RAMs, fabricated with the high performance N-channel silicon-gate MOS process, and is ideal for large-capacity memory systems where high speed, low power dissipation, and low costs are essential. The use of double-layer polysilicon process technology and a single-transistor dynamic storage cell provide high circuit density at reduced costs, and the use of dynamic circuitry including sense amplifiers assures low power dissipation. Multiplexed address inputs permit both a reduction in pins to the standard 16-pin package configuration and an increase in system densities. The M5K4164P operates on a 5V power supply using the on-chip substrate bias generator.

PIN CONFIGURATION (TOP VIEW)



* If the pin 1 (REF) function is not used, pin 1 may be left open (not connect).

Outline 16P4

FEATURES

- Performance ranges

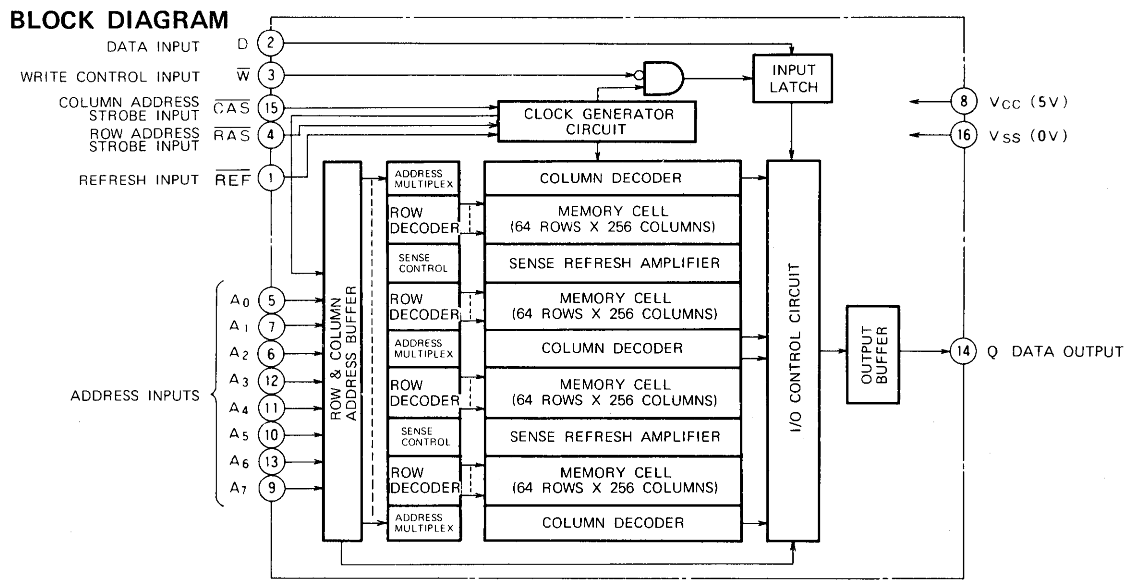
Type name	Access time (max) (ns)	Cycle time (min) (ns)	Power dissipation (typ) (mW)
M5K4164P-15	150	260	200
M5K4164P-20	200	330	170

- Standard 16-pin package
- Single 5V ±10% supply
- Low standby power dissipation: 22mW (max)
- Low operating power dissipation:
  - M5K4164P-15 275mW (max)
  - M5K4164P-20 250mW (max)
- Unlatched output enables two-dimensional chip selection and extended page boundary
- Early-write operation gives common I/O capability
- Read-modify-write, RAS-only refresh, and page-mode capabilities
- All input terminals have low input capacitance and are directly TTL-compatible
- Output is three-state and directly TTL-compatible
- 128 refresh cycles every 2ms (16K dynamic RAMs M5K4116P, S compatible)
- Pin 1 controls automatic- and self-refresh mode
- CAS controlled output allows hidden refresh, hidden automatic refresh and hidden self-refresh
- Output data can be held infinitely by CAS
- Interchangeable with Mostek's MK4164 and Motorola's MCM 6664 in pin configuration

APPLICATION

- Main memory unit for computers

BLOCK DIAGRAM



FUNCTION

The M5K4164P provides, in addition to normal read, write, and read-modify-write operations, a number of other functions, e.g., page mode, $\overline{RAS}$-only refresh, and delayed-write. The input conditions for each are shown in Table 1.

Table 1 Input conditions for each mode

Operation	inputs							Output	Refresh	Remarks
	$\overline{RAS}$	$\overline{CAS}$	$\overline{W}$	D	Row address	Column address	$\overline{REF}$	Q		
Read	ACT	ACT	NAC	DNC	APD	APD	NAC	VLD	YES	Page mode identical except refresh is NO.
Write	ACT	ACT	ACT	VLD	APD	APD	NAC	OPN	YES	
Read-modify-write	ACT	ACT	ACT	VLD	APD	APD	NAC	VLD	YES	
$\overline{RAS}$-only refresh	ACT	NAC	DNC	DNC	APD	DNC	NAC	OPN	YES	
Hidden refresh	ACT	ACT	DNC	DNC	APD	DNC	NAC	VLD	YES	
Automatic refresh	NAC	DNC	DNC	DNC	DNC	DNC	ACT	OPN	YES	
Self refresh	NAC	DNC	DNC	DNC	DNC	DNC	ACT	OPN	YES	
Hidden automatic refresh	NAC	ACT	DNC	DNC	DNC	DNC	ACT	VLD	YES	
Hidden self refresh	NAC	ACT	DNC	DNC	DNC	DNC	ACT	VLD	YES	
Standby	NAC	DNC	DNC	DNC	DNC	DNC	NAC	OPN	NO	

Note: ACT : active, NAC : nonactive, DNC : don't care, VLD : valid, APD : applied, OPN : open.

SUMMARY OF OPERATIONS

Addressing

To select one of the 65 536 memory cells in the M5K4164P the 16-bit address signal must be multiplexed into 8 address signals, which are then latched into the on-chip latch by two externally-applied clock pulses. First, the negative-going edge of the row-address-strobe pulse ($\overline{RAS}$) latches the 8 row-address bits; next, the negative-going edge of the column-address-strobe pulse ($\overline{CAS}$) latches the 8 column-address bits. Timing of the $\overline{RAS}$ and $\overline{CAS}$ clocks can be selected by either of the following two methods:

1. The delay time from $\overline{RAS}$ to $\overline{CAS}$ $t_{d(RAS-CAS)}$ is set between the minimum and maximum values of the limits. In this case, the internal $\overline{CAS}$ control signals are inhibited almost until $t_{d(RAS-CAS)max}$ ('gated $\overline{CAS}$' operation). The external $\overline{CAS}$ signal can be applied with a margin not affecting the on-chip circuit operations, e.g. access time, and the address inputs can be easily changed from row address to column address.
2. The delay time $t_{d(RAS-CAS)}$ is set larger than the maximum value of the limits. In this case the internal inhibition of $\overline{CAS}$ has already been released, so that the internal $\overline{CAS}$ control signals are controlled by the externally applied $\overline{CAS}$, which also controls the access time.

Data Input

Date to be written into a selected cell is strobed by the later of the two negative transitions of $\overline{W}$ input and $\overline{CAS}$ input. Thus when the $\overline{W}$ input makes its negative transition prior to $\overline{CAS}$ input (early write), the data input is strobed by $\overline{CAS}$, and the negative transition of $\overline{CAS}$ is set as the reference point for set-up and hold times. In the read-write or read-modify-write cycles, however, when the $\overline{W}$ input makes its negative transition after $\overline{CAS}$, the $\overline{W}$ negative transition is set as the reference point for setup and hold times.

Data Output Control

The output of the M5K4164P is in the high-impedance state when $\overline{CAS}$ is high. When the memory cycle in progress is a read, read-modify-write, or a delayed-write cycle, the data output will go from the high-impedance state to the active condition, and the data in the selected cell will be read. This data output will have the same polarity as the input data. Once the output has entered the active condition, this condition will be maintained until $\overline{CAS}$ goes high, irrespective of the condition of $\overline{RAS}$.

The output will remain in the high-impedance state throughout the entire cycle in an early-write cycle.

These output conditions, of the M5K4164P, which can readily be changed by controlling the timing of the write pulse in a write cycle, and the width of the $\overline{CAS}$ pulse in a read cycle, offer capabilities for a number of applications, as follows.

1. Common I/O Operation

If all write operations are performed in the early-write mode, input and output can be connected directly to give a common I/O data bus.

2. Data Output Hold

The data output can be held between read cycles, without lengthening the cycle time. This enables extremely flexible clock-timing settings for $\overline{RAS}$ and $\overline{CAS}$.

3. Two Methods of Chip Selection

Since the output is not latched, $\overline{\text{CAS}}$ is not required to keep the outputs of selected chips in the matrix in a high-impedance state. This means that $\overline{\text{CAS}}$ and/or $\overline{\text{RAS}}$ can both be decoded for chip selection.

4. Extended-Page Boundary

By decoding $\overline{\text{CAS}}$, the page boundary can be extended beyond the 256 column locations in a single chip. In this case, $\overline{\text{RAS}}$ must be applied to all devices.

Page-Mode Operation

This operation allows for multiple-column addressing at the same row address, and eliminates the power dissipation associated with the negative-going edge of $\overline{\text{RAS}}$, because once the row address has been strobed, $\overline{\text{RAS}}$ is maintained. Also, the time required to strobe in the row address for the second and subsequent cycles is eliminated, thereby decreasing the access and cycle times.

Refresh

Each of the 128 rows ($A_0 \sim A_6$) of the M5K4164P must be refreshed every 2 ms to maintain data. The methods of refreshing for the M5K4164P are as follows.

1. Normal Refresh

Read cycle and Write cycle (early write, delayed write or read-modify-write) refresh the selected row as defined by the low order ($\overline{\text{RAS}}$) addresses. Any write cycle, of course, may change the state of the selected cell. Using a read, write, or read-modify-write cycle for refresh is not recommended for systems which utilize "wire-OR" outputs since output bus contention will occur.

2. $\overline{\text{RAS}}$ Only Refresh

A $\overline{\text{RAS}}$-only refresh cycle is the recommended technique for most applications to provide for data retention. A $\overline{\text{RAS}}$-only refresh cycle maintains the output in the high-impedance state with a typical power reduction of 20% over a read or write cycle.

3. Automatic Refresh

Pin 1 ($\overline{\text{REF}}$) has two special functions. The M5K4164P has a refresh address counter, refresh address multiplexer and refresh timer for these operations. Automatic refresh is initiated by bringing $\overline{\text{REF}}$ low after $\overline{\text{RAS}}$ has precharged and is used during standard operation just like $\overline{\text{RAS}}$-only refresh, except that sequential row addresses from an external counter are no longer necessary.

At the end of automatic refresh cycle, the internal refresh address counter will be automatically incremented. The output state of the refresh address counter is initiated by some eight $\overline{\text{REF}}$, $\overline{\text{RAS}}$ or $\overline{\text{RAS}}/\overline{\text{CAS}}$ cycle after power is applied. Therefore, a special operation is not necessary to initiate it.

$\overline{\text{RAS}}$ must remain inactive during $\overline{\text{REF}}$ activated cycles. Likewise, $\overline{\text{REF}}$ must remain inactive during $\overline{\text{RAS}}$ generated cycle.

4. Self-Refresh

The other function of pin 1 ($\overline{\text{REF}}$) is self-refresh. Timing for self-refresh is quite similar to that for automatic refresh. As long as $\overline{\text{RAS}}$ remains high and $\overline{\text{REF}}$ remains low, the M5K4164P will refresh itself. This internal sequence repeats asynchronously every 12 to 16 μs. After 2 ms, the on-chip refresh address counter has advanced through all the row addresses and refreshed the entire memory. Self-refresh is primarily intended for trouble free power-down operation.

For example, when battery backup is used to maintained data integrity in the memory. $\overline{\text{REF}}$ may be used to place the device in the self-refresh mode with no external timing signals necessary to keep the information alive.

In summary, the pin 1 ($\overline{\text{REF}}$) refresh function gives the user a feature that is free, save him hardware on the board, and in fact, will simplify his battery backup procedures, increase his battery life, and save him overall cost while giving him improved system performance.

There is an internal pullup resister ($\approx 3\text{M}\Omega$) on pin 1, so if the pin 1 ($\overline{\text{REF}}$) function is not used, pin 1 may be left open (not connect) without affecting the normal operations.

5. Hidden Refresh

A features of the M5K4164P is that refresh cycle may be performed while maintaining valid data at the output pin by extending the $\overline{\text{CAS}}$ active time from a previous memory read cycle. This feature is refered to as hidden refresh.

Hidden refresh is performed by holding $\overline{\text{CAS}}$ at V_{IL} and taking $\overline{\text{RAS}}$ high and after a specified precharge period, executing a $\overline{\text{RAS}}$-only cycling, automatic refresh and self-refresh, but with $\overline{\text{CAS}}$ held low.

The advantage of this refresh mode is that data can be held valid at the output data port indefinitely by leaving the $\overline{\text{CAS}}$ asserted. In many applications this eliminates the need for off-chip latches.

Power Dissipation

Most of the circuirty in the M5K4164P is dynamic, and most of the power is dissipated when addresses are strobed. Both $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ are decoded and applied to the M5K4164P as chip-select in the memory system, but if $\overline{\text{RAS}}$ is decoded, all unselected devices go into stand-by independent of the $\overline{\text{CAS}}$ condition, minimizing system power dissipation.

Power Supplies

The M5K4164P operates on a single 5V power supply.

A wait of some 500μs and eight or more dummy cycle is necessary after power is applied to the device before memory operation is achieved.

65 536-BIT (65 536-WORD BY 1-BIT) DYNAMIC RAM

ABSOLUTE MAXIMUM RATINGS

Symbol	Paramater	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−1～7	V
V_I	Input voltage		−1～7	V
V_O	Output voltage		−1～7	V
I_O	Output current		50	mA
Pd	Power dissipation	Ta = 25°C	700	mW
Topr	Operating free-air temperature range		0～70	°C
Tstg	Storage temperature range		−65～150	°C

RECOMMENDED OPERATING CONDITIONS (Ta = 0～70°C, unless otherwise noted) (Note 1)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage	0	0	0	V
V_{IH}	High-level input voltage, all inputs	2.4		6.5	V
V_{IL}	Low-level input voltage, all inputs	−2		0.8	V

Note 1: All voltage values are with respect to V_{SS}

ELECTRICAL CHARACTERISTICS (Ta = 0～70°C, V_{CC} = 5V ± 10%, V_{SS} = 0V, unless otherwise noted) (Note 2)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
V_{OH}	High-level output voltage		$I_{OH} = -5mA$	2.4		V_{CC}	V
V_{OL}	Low-level output voltage		$I_{OL} = 4.2mA$	0		0.4	V
I_{OZ}	Off-state output current		Q floating $0V \leqq V_{OUT} \leqq 5.5V$	−10		10	μA
I_I	Input current		$0V \leqq V_{IN} \leqq 6.5V$, All other pins = 0V	−10		10	μA
$I_{CC1(AV)}$	Average supply current from V_{CC}, operating (Note 3, 4)	M5K4164P-15	$\overline{RAS}$, $\overline{CAS}$ cycling			50	mA
		M5K4164P-20	$t_{CR} = t_{CW}$ = min output open			45	mA
I_{CC2}	Supply current from V_{CC}, standby		$\overline{RAS} = V_{IH}$ output open			4	mA
$I_{CC3(AV)}$	Average supply current from V_{CC}, refreshing (Note 3)	M5K4164P-15	$\overline{RAS}$ cycling $\overline{CAS} = V_{IH}$			40	mA
		M5K4164P-20	$t_{C(REF)}$ = min, output open			35	mA
$I_{CC4(AV)}$	Average supply current from V_{CC}, page mode (Note 3, 4)	M5K4164P-15	$\overline{RAS} = V_{IL}$, $\overline{CAS}$ cycling			40	mA
		M5K4164P-20	t_{CPG} = min, output open			35	mA
$I_{CC5(AV)}$	Average supply current from V_{CC}, automatic refreshing (Note 3)	M5K4164P-15	$\overline{RAS} = V_{IH}$, $\overline{REF}$ cycling			40	mA
		M5K4164P-20	$t_{C(REF)}$ = min, output open			35	mA
$I_{CC6(AV)}$	Average supply current from V_{CC}, self refreshing		$\overline{RAS} = V_{IH}$, $\overline{REF} = V_{IL}$ output open			8	mA
$C_{I(A)}$	Input capacitance, address inputs		$V_I = V_{SS}$ f = 1MHz V_I = 25mVrms			5	pF
$C_{I(D)}$	Input capacitance, data input					5	pF
$C_{I(W)}$	Input capacitance, write control input					7	pF
$C_{I(RAS)}$	Input capacitance, $\overline{RAS}$ input					10	pF
$C_{I(CAS)}$	Input capacitance, $\overline{CAS}$ input					10	pF
$C_{I(REF)}$	Input capacitance, $\overline{REF}$ input					10	pF
C_O	Output capacitance		$V_O = V_{SS}$, f = 1MHz, V_I = 25mVrms			7	pF

Note 2: Current flowing into an IC is positive ; out is negative.
3: $I_{CC1(AV)}$, $I_{CC3(AV)}$, $I_{CC4(AV)}$ and $I_{CC5(AV)}$ are dependent on cycle rate. Maximum current is measured at the fastest cycle rate.
4: $I_{CC1(AV)}$ and $I_{CC4(AV)}$ are dependent on output loading. Specified values are obtained with the output open.

2

TIMING REQUIREMENTS (For Read, Write, Read-Modify-Write, Refresh, and Page-Mode Cycle)

($T_a = 0 \sim 70°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted, See notes 5, 6 and 7)

Symbol	Parameter	Alternative Symbol	M5K4164P-15		M5K4164P-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{CRF}	Refresh cycle time	t_{REF}		2		2	ms
$t_{w(RASH)}$	$\overline{RAS}$ high pulse width	t_{RP}	100		120		ns
$t_{w(RASL)}$	$\overline{RAS}$ low pulse width	t_{RAS}	150	10000	200	10000	ns
$t_{w(CASL)}$	$\overline{CAS}$ low pulse width	t_{CAS}	75	∞	100	∞	ns
$t_{w(CASH)}$	$\overline{CAS}$ high pulse width (Note 8)	t_{CPN}	35		40		ns
$t_{h(RAS-CAS)}$	$\overline{CAS}$ hold time after $\overline{RAS}$	t_{CSH}	150		200		ns
$t_{h(CAS-RAS)}$	$\overline{RAS}$ hold time after $\overline{CAS}$	t_{RSH}	75		100		ns
$t_{d(CAS-RAS)}$	Delay time, $\overline{CAS}$ to $\overline{RAS}$ (Note 9)	t_{CRP}	−20		−20		ns
$t_{d(RAS-CAS)}$	Delay time, $\overline{RAS}$ to $\overline{CAS}$ (Note 10)	t_{RCD}	25	75	30	100	ns
$t_{su(RA-RAS)}$	Row address setup time before $\overline{RAS}$	t_{ASR}	0		0		ns
$t_{su(CA-CAS)}$	Column address setup time before $\overline{CAS}$	t_{ASC}	−5		−5		ns
$t_{h(RAS-RA)}$	Row address hold time after $\overline{RAS}$	t_{RAH}	20		25		ns
$t_{h(CAS-CA)}$	Column address hold time after $\overline{CAS}$	t_{CAH}	25		35		ns
$t_{h(RAS-CA)}$	Column address hold time after $\overline{RAS}$	t_{AR}	95		120		ns
t_{THL} t_{TLH}	Transition time	t_T	3	35	3	50	ns

Note 5: An initial pause of 500μs is required after power-up followed by any eight $\overline{REF}$, $\overline{RAS}$ or $\overline{RAS}/\overline{CAS}$ cycles before proper device operation is achieved.
6: The switching characteristics are defined as $t_{THL} = t_{TLH} = 5ns$.
7: Reference levels of input signals are $V_{IH\ min.}$ and $V_{IL\ max.}$ Reference levels for transition time are also between V_{IH} and V_{IL}.
8: Except for page-mode.
9: $t_{d(CAS-RAS)}$ requirement is only applicable for $\overline{RAS}/\overline{CAS}$ cycles preceeded by a $\overline{CAS}$ only cycle (i.e., For systems where $\overline{CAS}$ has not been decoded with $\overline{RAS}$)
10: Operation within the $t_{d(RAS-CAS)}$ max limit insures that $t_{a(RAS)}$max can be met. $t_{d(RAS-CAS)}$max is specified reference point only;if $t_{d(RAS-CAS)}$ is greater than the specified $t_{d(RAS-CAS)}$ max limit, then access time is controlled exclusively by $t_{a(CAS)}$.
$t_{d(RAS-CAS)min} = t_{h(RAS-RA)min} + 2t_{THL}(t_{TLH}) + t_{su(CA-CAS)min}$.

SWITCHING CHARACTERISTICS ($T_a = 0 \sim 70°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted)

Read Cycle

Symbol	Parameter	Alternative Symbol	M5K4164P-15		M5K4164P-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{CR}	Read cycle time	t_{RC}	260		330		ns
$t_{su(R-CAS)}$	Read setup time before $\overline{CAS}$	t_{RCS}	0		0		ns
$t_{h(CAS-R)}$	Read hold time after $\overline{CAS}$ (Note 11)	t_{RCH}	0		0		ns
$t_{h(RAS-R)}$	Read hold time after $\overline{RAS}$ (Note 11)	t_{RRH}	20		25		ns
$t_{dis(CAS)}$	Output disable time (Note 12)	t_{OFF}	0	40	0	50	ns
$t_{a(CAS)}$	$\overline{CAS}$ access time (Note 13)	t_{CAC}		75		100	ns
$t_{a(RAS)}$	$\overline{RAS}$ access time (Note 14)	t_{RAC}		150		200	ns

Note 11: Either $t_{h(RAS-R)}$ or $t_{h(CAS-R)}$ must be satisfied for a read cycle.
Note 12: $t_{dis(CAS)}$max defines the time at which the output achieves the open circuit condition and is not reference to V_{OH} or V_{OL}.
Note 13: This is the value when $t_{d(RAS-CAS)} \geqq t_{d(RAS-CAS)}$max. Test conditions ; Load = 2T TL, C_L = 100pF
Note 14: This is the value when $t_{d(RAS-CAS)} < t_{d(RAS-CAS)}$max. When $t_{d(RAS-CAS)} \geqq t_{d(RAS-CAS)}$max, $t_{a(RAS)}$ will increase by the amount that $t_{d(RAS-CAS)}$ exceeds the value shown. Test conditions ; Load = 2T TL, C_L = 100pF

Write Cycle

Symbol	Parameter	Alternative Symbol	M5K4164P-15		M5K4164P-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{CW}	Write cycle time	t_{RC}	260		330		ns
$t_{su(W-CAS)}$	Write setup time before $\overline{CAS}$ (Note 17)	t_{WCS}	−10		−10		ns
$t_{h(CAS-W)}$	Write hold time after $\overline{CAS}$	t_{WCH}	45		55		ns
$t_{h(RAS-W)}$	Write hold time after $\overline{RAS}$	t_{WCR}	95		120		ns
$t_{h(W-RAS)}$	$\overline{RAS}$ hold time after write	t_{RWL}	45		55		ns
$t_{h(W-CAS)}$	$\overline{CAS}$ hold time after write	t_{CWL}	45		55		ns
$t_{w(W)}$	Write pulse width	t_{WP}	45		55		ns
$t_{su(D-CAS)}$	Data-in setup time before $\overline{CAS}$	t_{DS}	0		0		ns
$t_{h(CAS-D)}$	Data-in hold time after $\overline{CAS}$	t_{DH}	45		55		ns
$t_{h(RAS-D)}$	Data-in hold time after $\overline{RAS}$	t_{DHR}	95		120		ns

Read-Write and Read-Modify-Write Cycles

Symbol	Parameter	Alternative Symbol	M5K4164P-15		M5K4164P-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{CRW}	Read-write cycle time (Note 15)	t_{RWC}	280		340		ns
t_{CRMW}	Read-modify-write cycle time (Note 16)	t_{RMWC}	310		390		ns
$t_{h(W-RAS)}$	$\overline{RAS}$ hold time after write	t_{RWL}	45		55		ns
$t_{h(W-CAS)}$	$\overline{CAS}$ hold time after write	t_{CWL}	45		55		ns
$t_{w(W)}$	Write pulse width	t_{WP}	45		55		ns
$t_{su(R-CAS)}$	Read setup time before $\overline{CAS}$	t_{RCS}	0		0		ns
$t_{d(RAS-W)}$	Delay time, $\overline{RAS}$ to write (Note 17)	t_{RWD}	120		150		ns
$t_{d(CAS-W)}$	Delay time, $\overline{CAS}$ to write (Note 17)	t_{CWD}	60		80		ns
$t_{su(D-W)}$	Data-in setup time before write	t_{DS}	0		0		ns
$t_{h(W-D)}$	Data-in hold time after write	t_{DH}	45		55		ns
$t_{dis(CAS)}$	Output disable time	t_{OFF}	0	40	0	50	ns
$t_{a(CAS)}$	$\overline{CAS}$ access time (Note 13)	t_{CAC}		75		100	ns
$t_{a(RAS)}$	$\overline{RAS}$ access time (Note 14)	t_{RAC}		150		200	ns

Note 15: t_{CRW} min is defined as $t_{CRW}\ min = t_{d(RAS-W)} + t_{h(W-RAS)} + t_{w(RASH)} + 3t_{TLH(tTHL)}$

16: t_{CRMW} min is defined as $t_{CRMW}\ min = t_{a(RAS)}max + t_{h(W-RAS)} + t_{w(RASH)} + 3t_{TLH(tTHL)}$

17: $t_{su(W-CAS)}$, $t_{d(RAS-W)}$, and $t_{d(CAS-W)}$ do not define the limits of operation, but are included as electrical characteristics only.
When $t_{su(W-CAS)} \geqq t_{su(W-CAS)}min$, an early-write cycle is performed, and the data output keeps the high-impedance state.
When $t_{d(RAS-W)} \geqq t_{d(RAS-W)}min$, and $t_{d(CAS-W)} \geqq t_{su(W-CAS)}min$ a read-write cycle is performed, and the data of the selected address will be read out on the data output.
For all conditions other than those described above, the condition of data output (at access time and until $\overline{CAS}$ goes back to V_{IH}) is not defined.

Page-Mode Cycle

Symbol	Parameter	Alternative Symbol	M5K4164P-15		M5K4164P-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
$t_{C\ PGR}$	Page-mode read cycle time	t_{PC}	145		190		ns
$t_{C\ PGW}$	Page-Mode write cycle time	t_{PC}	145		190		ns
$t_{C\ PGRW}$	Page-Mode read-write cycle time	—	180		230		ns
$t_{C\ PGRMW}$	Page-Mode read-modify-write cycle time	—	190		245		ns
$t_{w(CASH)}$	$\overline{CAS}$ high pulse width	t_{CP}	60		80		ns

Automatic Refresh Cycle

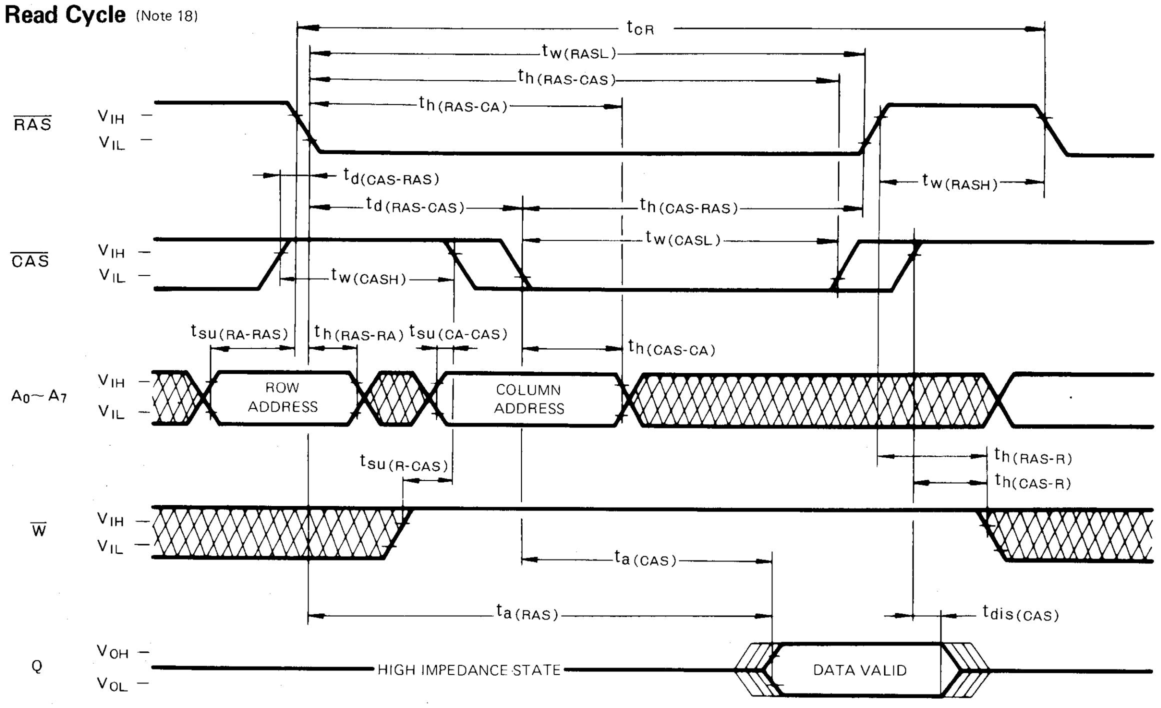
Symbol	Parameter	Alternative Symbol	M5K4164P-15		M5K4164P-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
$t_{c(REF)}$	Automatic Refresh cycle time	t_{FC}	260		330		ns
$t_{d(RAS-REF)}$	Delay time, $\overline{RAS}$ to $\overline{REF}$	t_{RFD}	100		120		ns
$t_{w(REFL)}$	$\overline{REF}$ low pulse width	t_{FP}	60	8000	60	8000	ns
$t_{w(REFH)}$	$\overline{REF}$ high pulse width	t_{FI}	30		30		ns
$t_{d(REF-RAS)}$	Delay time, $\overline{REF}$ to $\overline{RAS}$	t_{FSR}	30		30		ns
$t_{su(REF-RAS)}$	$\overline{REF}$ pulse setup time before $\overline{RAS}$	t_{FRD}	295		360		ns

Self-Refresh Cycle

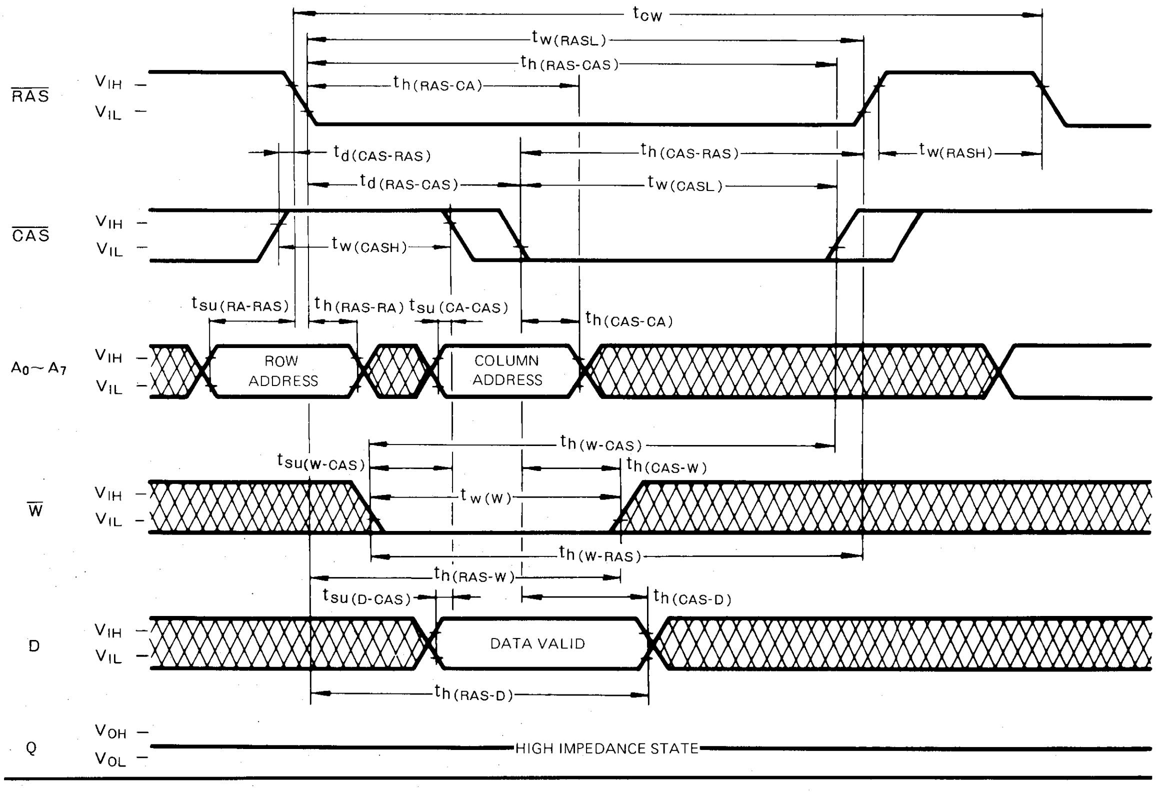
Symbol	Parameter	Alternative Symbol	M5K4164P-15		M5K4164P-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
$t_{d(RAS-REF)}$	Delay time, $\overline{RAS}$ to $\overline{REF}$	t_{RFD}	100		120		ns
$t_{w(REFL)}$	$\overline{REF}$ low pulse width	t_{FBP}	8000	∞	8000	∞	ns
$t_{d(REF-RAS)}$	Delay time, $\overline{REF}$ to $\overline{RAS}$	t_{FBR}	295		360		ns

TIMING DIAGRAMS (Note 17)

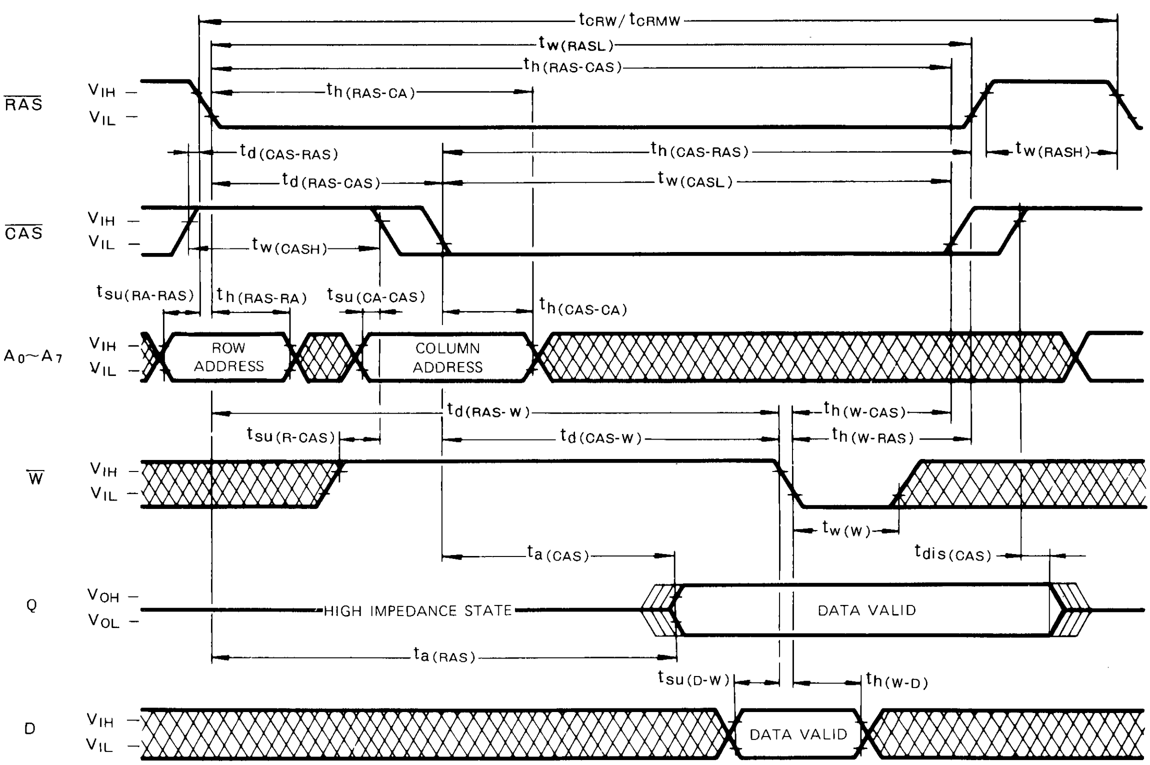
Read Cycle (Note 18)



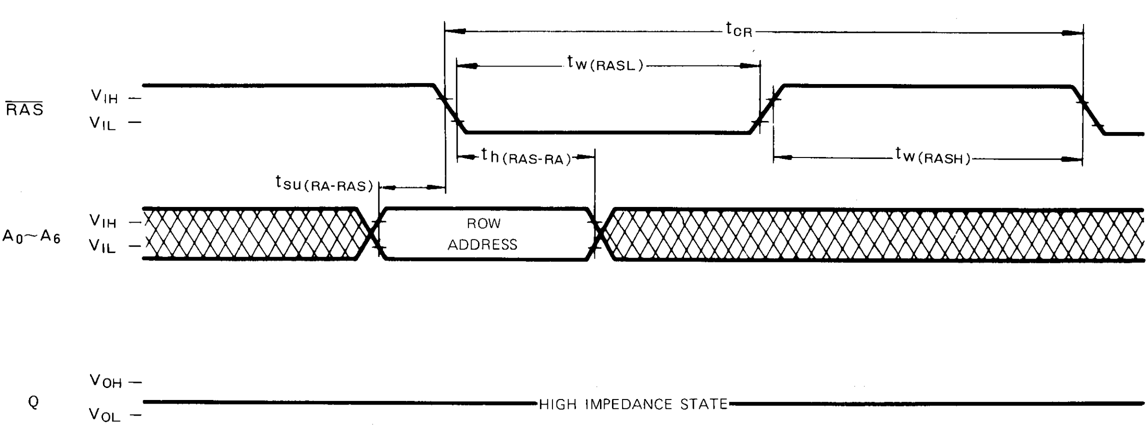
Write Cycle (Early Write) (Note 18)



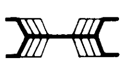
Read-Write and Read-Modify-Write Cycles (Note 18)



$\overline{\text{RAS}}$-Only Refresh Cycle (Note 19)



Note 17  Indicates the don't care input

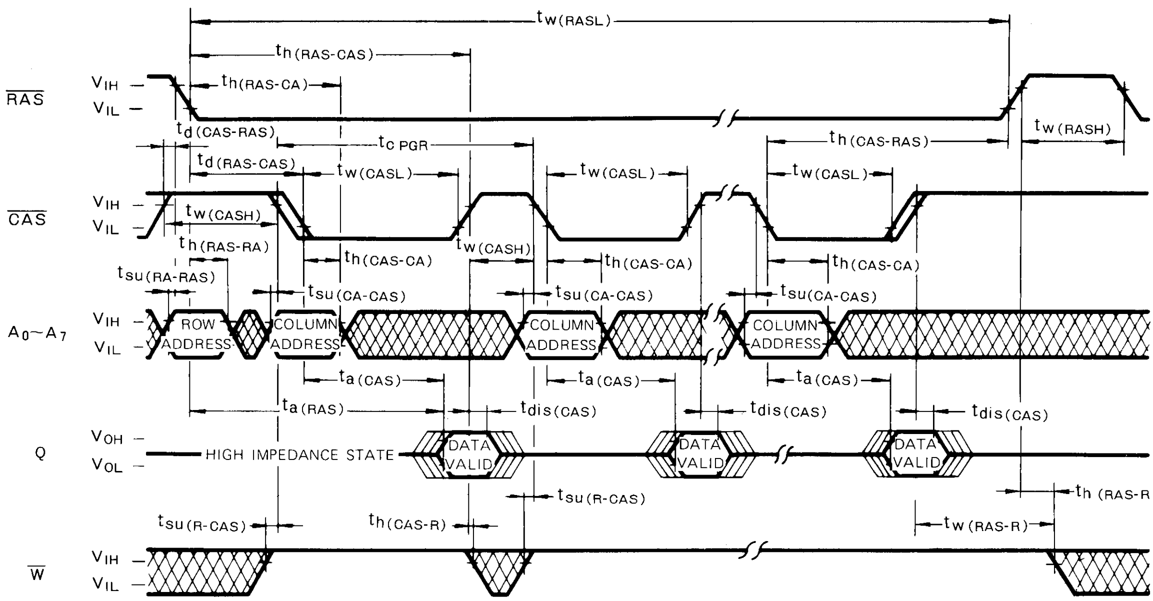
 The center-line indicates the high-impedance state

Note 18. $\overline{\text{REF}} = V_{IH}$

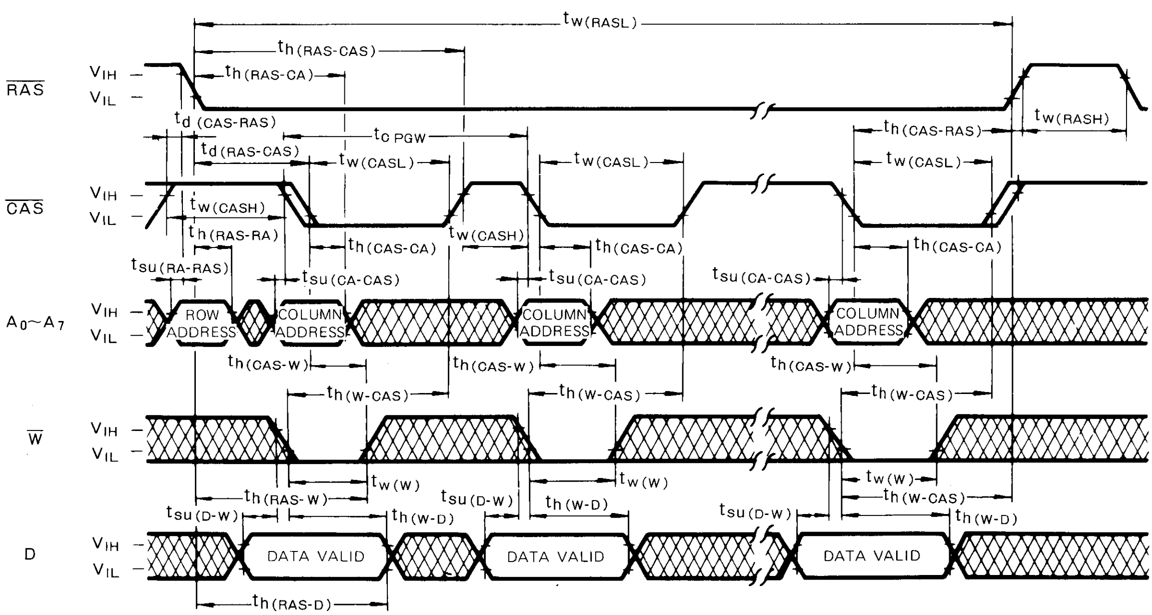
19. $\overline{\text{CAS}} = \overline{\text{REF}} = V_{IH}$, $\overline{\text{W}}$, A_7, D = don't care.

2

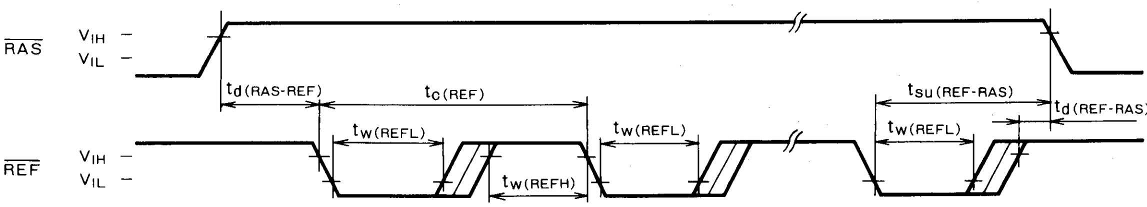
Page-Mode Read Cycle (Note 18)



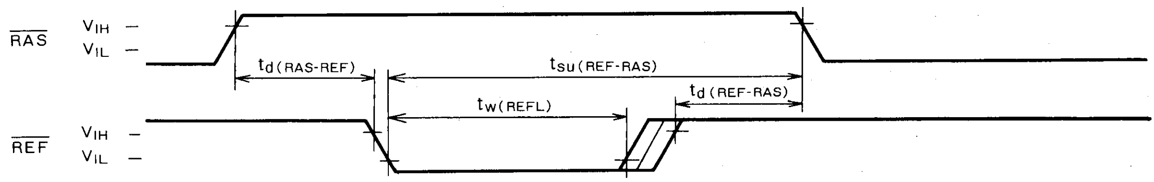
Page-Mode Write Cycle (Note 18)



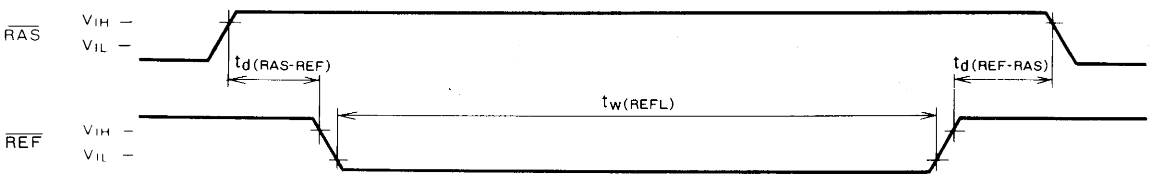
Automatic Pulse Refresh Cycle (Multiple Pulse) (Note 20)



Automatic Pulse Refresh Cycle (Single Pulse) (Note 20)

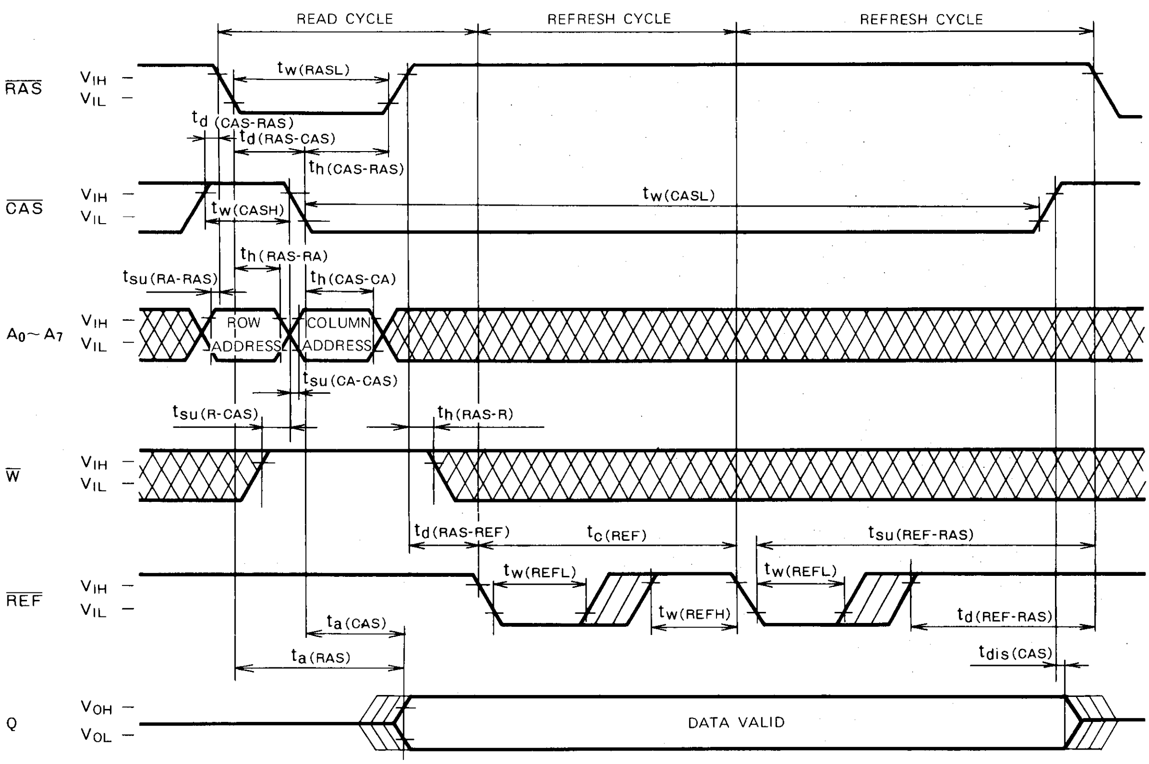


Self-Refresh Cycle (Note 20)

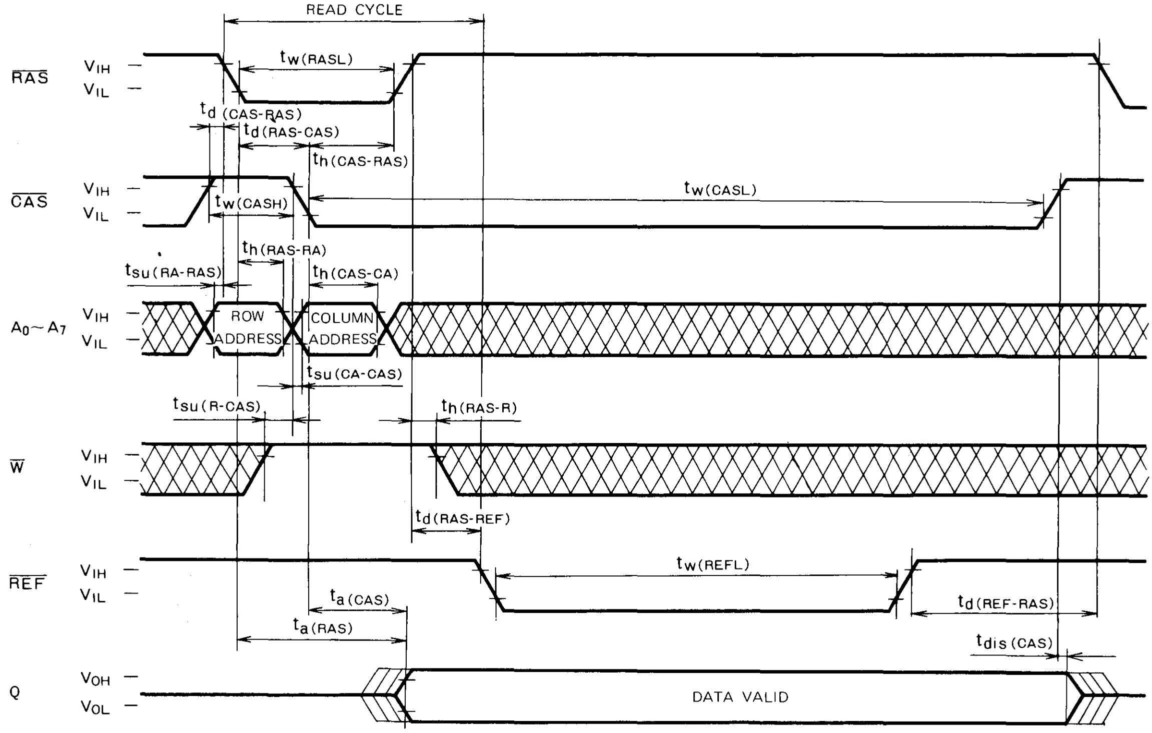


Note 20: $\overline{CAS}$, Addresses, D and $\overline{W}$ are don't care.

Hidden Automatic Pulse Refresh Cycle

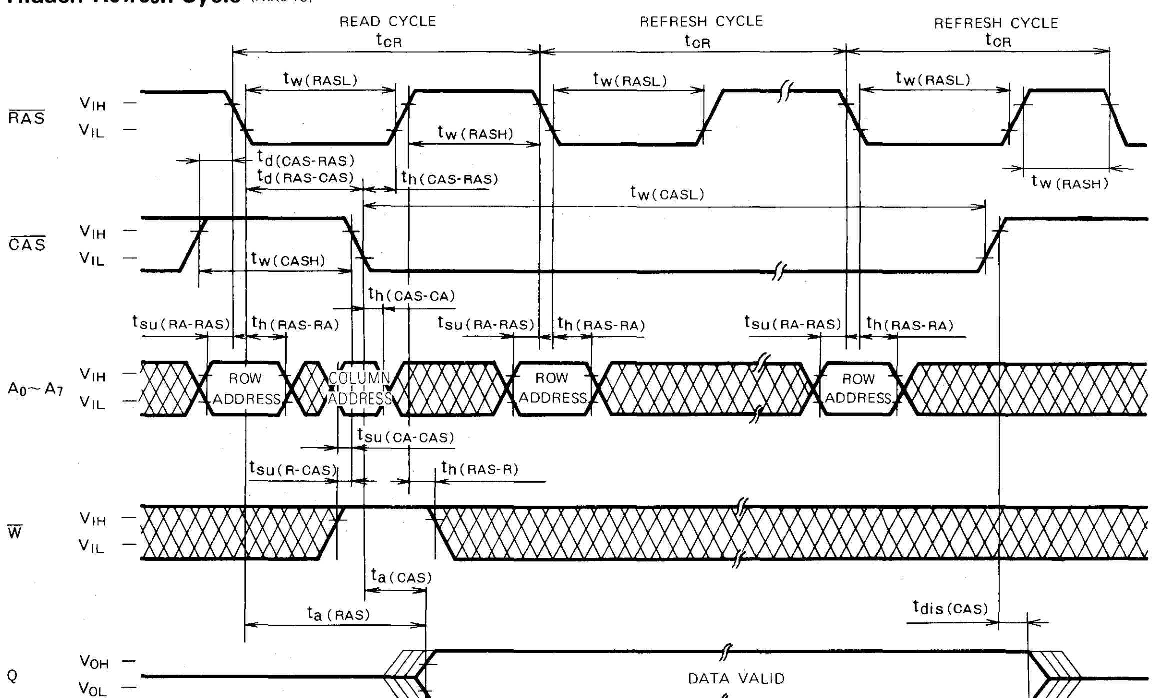


Hidden Self-Refresh Cycle



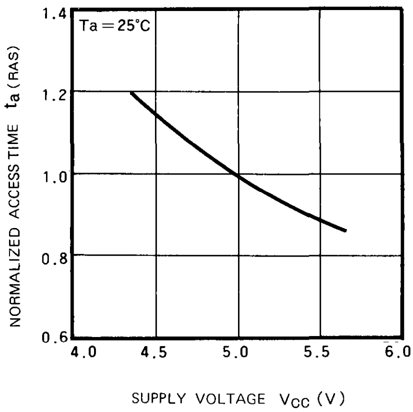
Note 21: If the pin 1 ($\overline{REF}$) function is not used, pin 1 may be left open (not connect).

Hidden Refresh Cycle (Note 18)

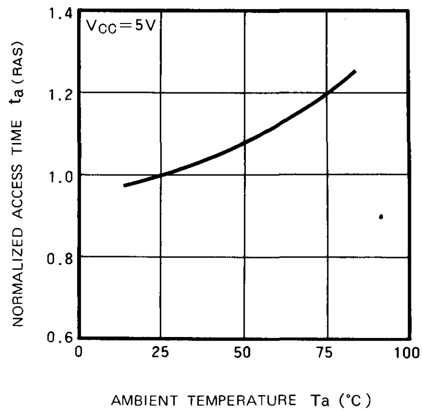


TYPICAL CHARACTERISTICS

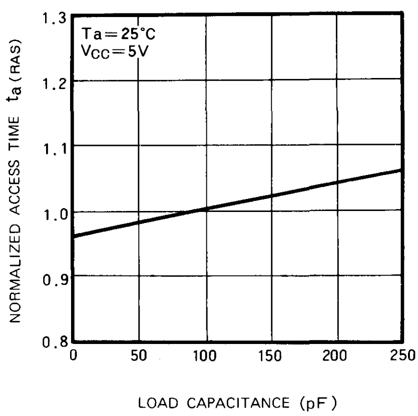
NORMALIZED ACCESS TIME VS. V_{CC} SUPPLY VOLTAGE



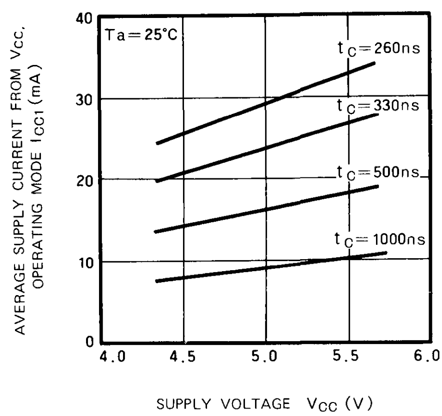
NORMALIZED ACCESS TIME VS. AMBIENT TEMPERATURE



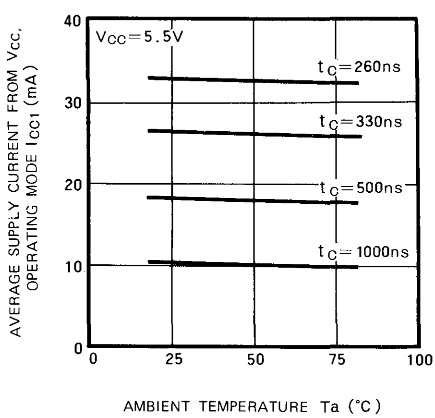
NORMALIZED ACCESS TIME VS. LOAD CAPACITANCE



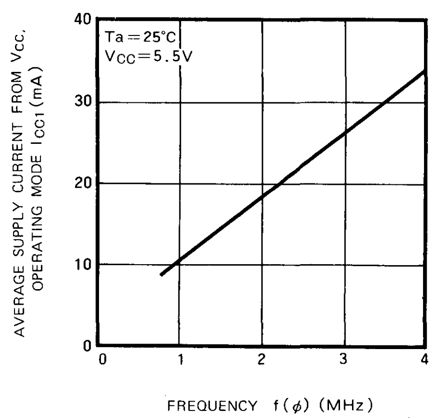
AVERAGE SUPPLY CURRENT FROM V_{CC}, OPERATING MODE VS. SUPPLY VOLTAGE



AVERAGE SUPPLY CURRENT FROM V_{CC}, OPERATING MODE VS. AMBIENT TEMPERATURE

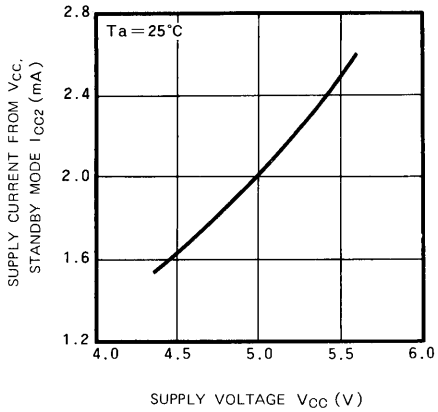


AVERAGE SUPPLY CURRENT FROM V_{CC}, OPERATING MODE VS. FREQUENCY

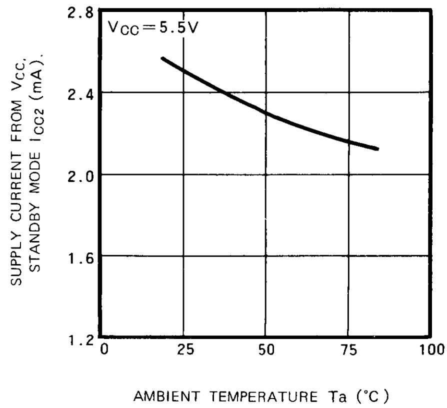


2

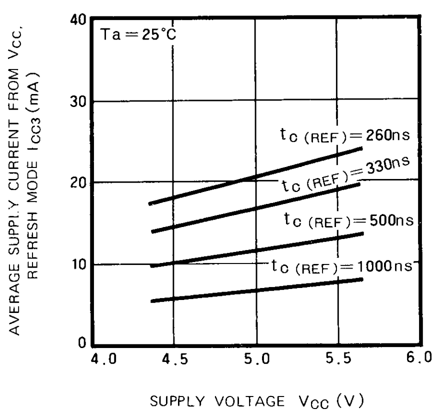
SUPPLY CURRENT FROM V_{CC}, STANDBY MODE VS. SUPPLY VOLTAGE



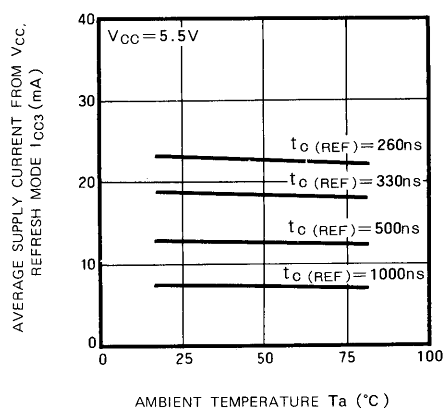
SUPPLY CURRENT FROM V_{CC}, STANDBY MODE VS. AMBIENT TEMPERATURE



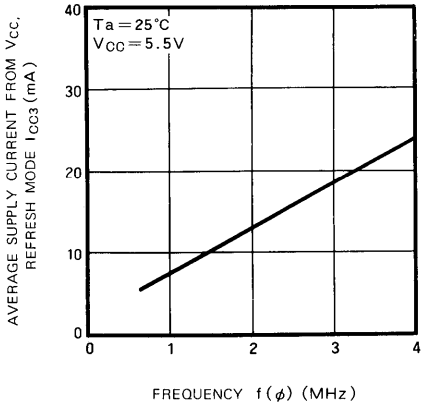
AVERAGE SUPPLY CURRENT FROM V_{CC}, REFRESH MODE VS. SUPPLY VOLTAGE



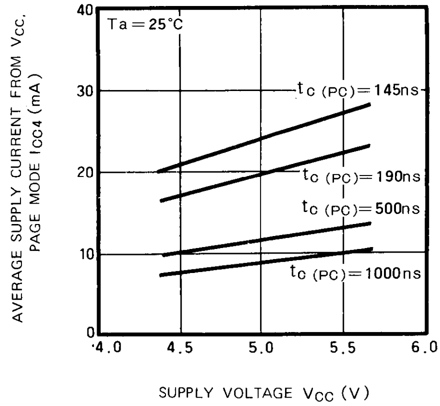
AVERAGE SUPPLY CURRENT FROM V_{CC}, REFRESH MODE VS. AMBIENT TEMPERATURE



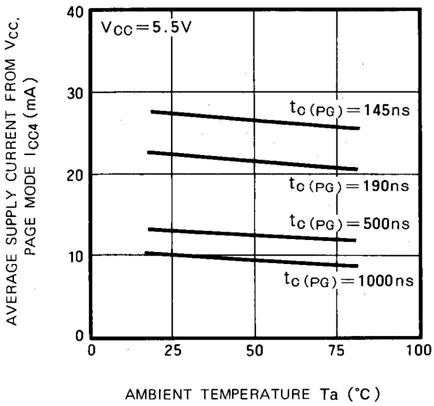
AVERAGE SUPPLY CURRENT FROM V_{CC}, REFRESH MODE VS. FREQUENCY



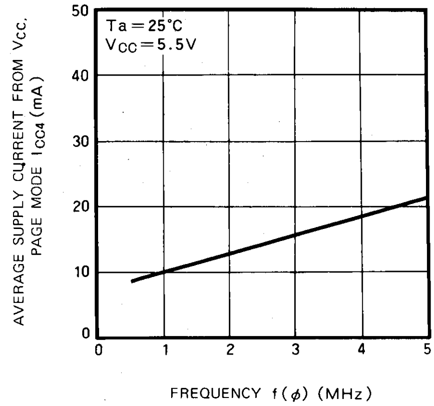
AVERAGE SUPPLY CURRENT FROM V_{CC}, PAGE MODE VS. SUPPLY VOLTAGE



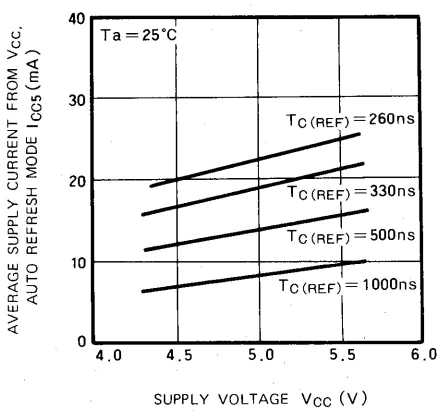
AVERAGE SUPPLY CURRENT FROM V_{CC}, PAGE MODE VS. AMBIENT TEMPERATURE



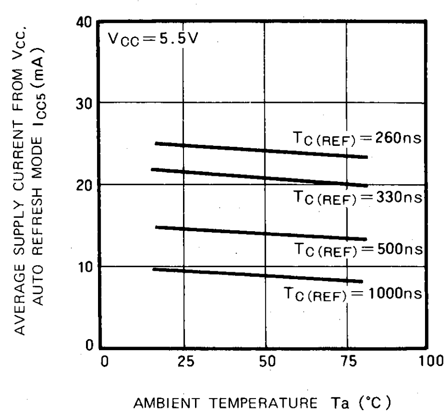
AVERAGE SUPPLY CURRENT FROM V_{CC}, PAGE MODE VS. FREQUENCY



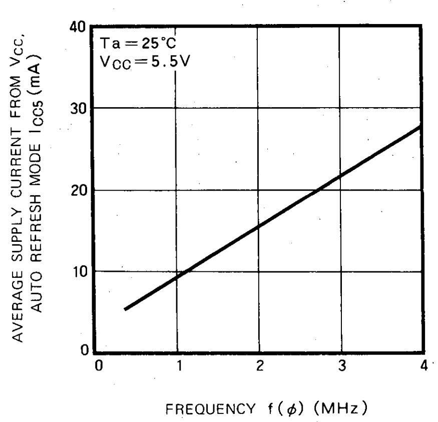
AVERAGE SUPPLY CURRENT FROM V_{CC}, AUTO REFRESH MODE VS. SUPPLY VOLTAGE



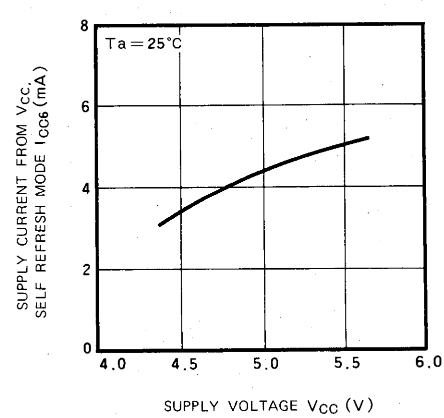
AVERAGE SUPPLY CURRENT FROM V_{CC}, AUTO REFRESH MODE VS. AMBIENT TEMPERATURE



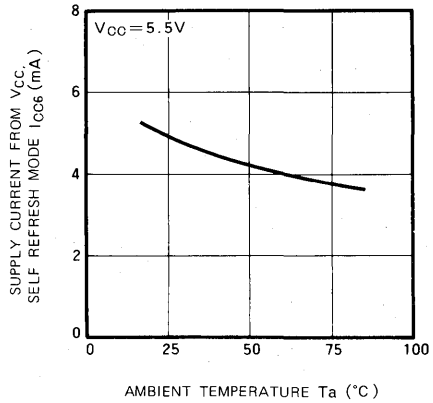
AVERAGE SUPPLY CURRENT FROM V_{CC}, AUTO REFRESH MODE VS. FREQUENCY



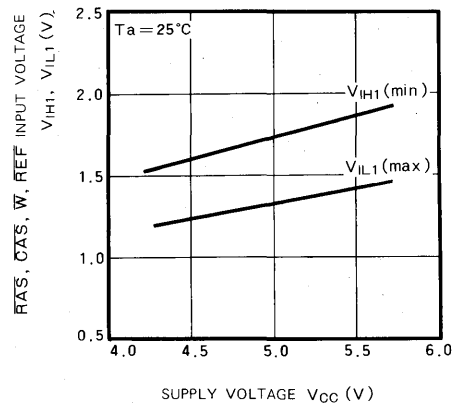
SUPPLY CURRENT FROM V_{CC}, SELF REFRESH MODE VS. SUPPLY VOLTAGE



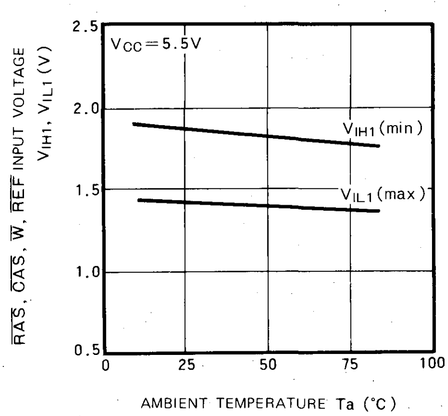
SUPPLY CURRENT FROM V_{CC}, SELF REFRESH MODE VS. AMBIENT TEMPERATURE



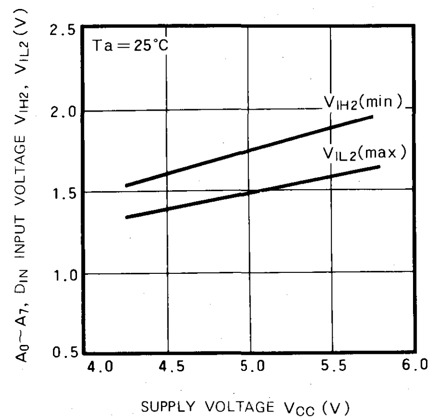
$\overline{RAS}$, $\overline{CAS}$, $\overline{W}$, $\overline{REF}$ INPUT VOLTAGE V_{IH1}, V_{IL1} VS. SUPPLY VOLTAGE



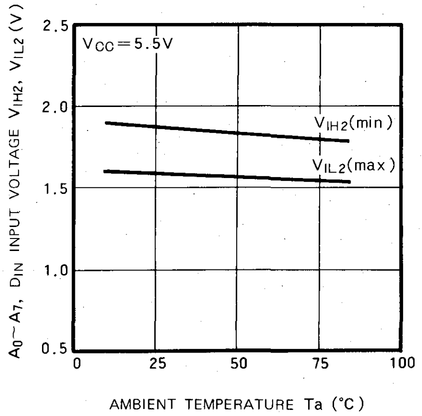
$\overline{RAS}$, $\overline{CAS}$, $\overline{W}$, $\overline{REF}$ INPUT VOLTAGE V_{IH1}, V_{IL1} VS. AMBIENT TEMPERATURE



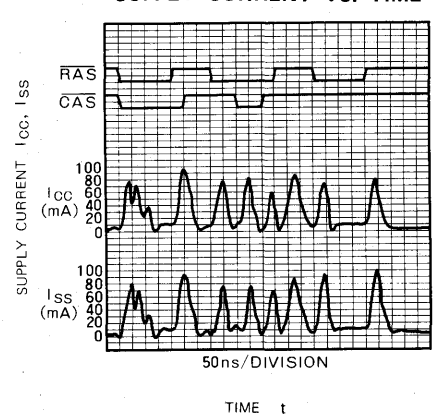
A_0~A_7, D_{IN} INPUT VOLTAGE V_{IH2}, V_{IL2} VS. SUPPLY VOLTAGE



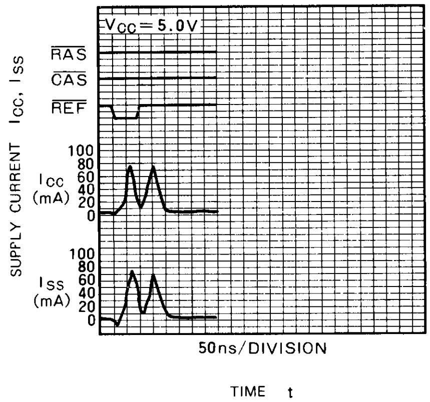
A_0~A_7, D_{IN} INPUT VOLTAGE V_{IH2}, V_{IL2} VS. AMBIENT TEMPERATURE



SUPPLY CURRENT VS. TIME



SUPPLY CURRENT VS. TIME

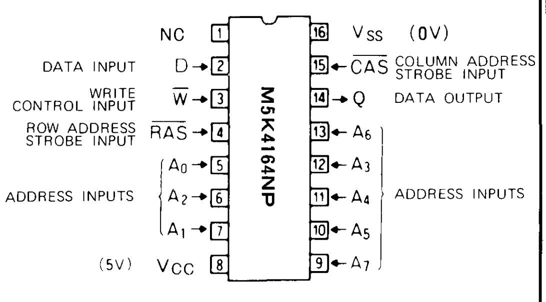


65 536-BIT (65 536-WORD BY 1-BIT) DYNAMIC RAM

DESCRIPTION

This is a family of 65 536-word by 1-bit dynamic RAMs, fabricated with the high performance N-channel silicongate MOS process, and is ideal for large-capacity memory systems where high speed, low power dissipation, and low costs are essential. The use of double-layer polysilicon process technology and a single-transistor dynamic storage cell privide high circuit density at reduced costs, and the use of dynamic circuitry including sense amplifiers assures low power dissipation. Multiplexed address inputs permit both a reduction in pins to the standard 16-pin package configuration and an increase in system densities. The M5K4164NP operates on a 5V power supply using the on-chip substrate bias generator.

PIN CONFIGURATION (TOP VIEW)



Outline 16P4

FEATURES

- Performance ranges

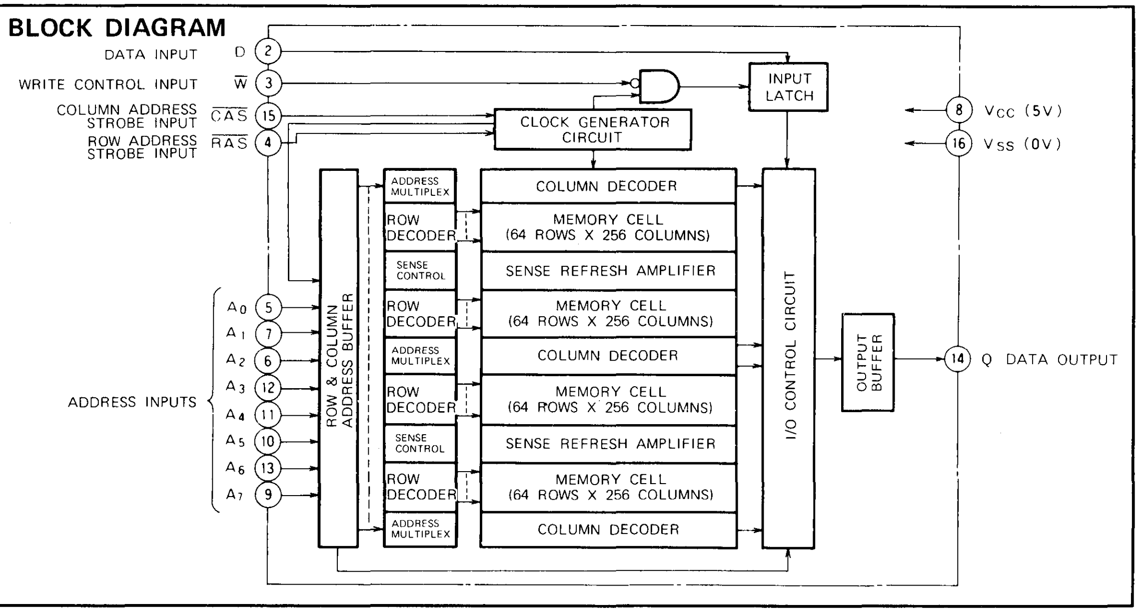
Type name	Access time (max) (ns)	Cycle time (min) (ns)	Power dissipation (typ) (mW)
M5K4164NP-15	150	260	200
M5K4164NP-20	200	330	170

- Standard 16-pin package
- Single 5V±10% supply
- Low standby power dissipation: 22mW (max)
- Low operating power dissipation:
  - M5K4164NP-15 275mW (max)
  - M5K4164NP-20 250mW (max)
- Unlatched output enables two-dimensional chip selection and extended page boundary
- Early-write operation gives common I/O capability
- Read-modify-write, $\overline{RAS}$-only refresh, and page-mode capabilities
- All input terminals have low input capaciatance and are directly TTL-compatible
- Output is three-state and directly TTL-compatible
- 128 refresh cycles every 2ms (16K dynamic RAMs M5K4116P, S compatible)
- $\overline{CAS}$ controlled output allows hidden refresh
- Output data can be held infinitely by $\overline{CAS}$
- Interchangeable with Mostek's MK4564 and Motorola's MCM6665 in pin configuration

APPLICATION

- Main memory unit for computers

BLOCK DIAGRAM



FUNCTION

The M5K4164NP provides, in addition to normal read, write, and read-modify-write operations, a number of other functions, e.g., page mode, RAS-only refresh, and delayed-write. The input conditions for each are shown in Table 1.

Table 1 Input conditions for each mode

Operation	Inputs						Output	Refresh	Remarks
	RAS	CAS	W	D	Row address	Column address	Q		
Read	ACT	ACT	NAC	DNC	APD	APD	VLD	YES	Page mode identical except refresh is NO.
Write	ACT	ACT	ACT	VLD	APD	APD	OPN	YES	
Read-modify-write	ACT	ACT	ACT	VLD	APD	APD	VLD	YES	
RAS-only refresh	ACT	NAC	DNC	DNC	APD	DNC	OPN	YES	
Hidden refresh	ACT	ACT	DNC	DNC	APD	DNC	VLD	YES	
Standby	NAC	DNC	DNC	DNC	DNC	DNC	OPN	NO	

Note: ACT : active, NAC : nonactive, DNC : don't care, VLD : valid, APD : applied, OPN : open.

SUMMARY OF OPERATIONS

Addressing

To select one of the 65536 memory cells in the M5K4164NP the 16-bit address signal must be multiplexed into 8 address signals, which are then latched into the on-chip latch by two externally-applied clock pulses. First, the negative-going edge of the row-address-strobe pulse (RAS) latches the 8 row-address bits; next, the negative-going edge of the column-address-strobe pulse (CAS) latches the 8 column-address bits. Timing of the RAS and CAS clocks can be selected by either of the following two methods:

1. The delay time from RAS to CAS $t_{d\ (RAS\text{-}CAS)}$ is set between the minimum and maximum values of the limits. In this case, the internal CAS control signals are inhibited almost until $t_{d(RAS\text{-}CAS)\,max}$ ('gated CAS' operation). The external CAS signal can be applied with a margin not affecting the on-chip circuit operations, e.g. access time, and the address inputs can be easily changed from row address to column address.
2. The delay time $t_{d(RAS\text{-}CAS)}$ is set larger than the maximum value of the limits. In this case the internal inhibition of CAS has already been released, so that the internal CAS control signals are controlled by the externally applied CAS, which also controls the access time.

Data Input

Data to be written into a selected cell is strobed by the later of the two negative transistons of W input and CAS input. Thus when the W input makes its negative transition prior to CAS input (early write), the data input is strobed by CAS, and the negative transition of CAS is set as the reference point for set-up and hold times. In the read-write or read-modify-write cycles, however, when the W input makes its negative transition after CAS, the W negative transition is set as the reference point for setup and hold times.

Data Output Control

The output of the M5K4164NP is in the high-impedance state when CAS is high. When the memory cycle in progress is a read, read-modify-write, or a delayed-write cycle, the data output will go from the high-impedance state to the active condition, and the data in the selected cell will be read. This data output will have the same polarity as the input data. Once the output has entered the active condition, this condition will be maintained until CAS goes high, irrespective of the condition of RAS.

The output will remain in the high-impedance state throughout the entire cycle in an early-write cycle.

These output conditions, of the M5K4164NP, which can readily be changed by controlling the timing of the write pulse in a write cycle, and the width of the CAS pulse in a read cycle, offer capabilities for a number of applications, as follows.

1. Common I/O Operation

If all write operations are performed in the early-write mode, input and output can be connected directly to give a common I/O data bus.

2 Data Output Hold

The data output can be held between read cycles, without lengthening the cycle time. This enables extremely flexible clock-timing settings for RAS and CAS.

3. Two Methods of Chip Selection

Since the output is not latched, $\overline{CAS}$ is not required to keep the outputs of selected chips in the matrix in a high-impedance state. This means that $\overline{CAS}$ and/or $\overline{RAS}$ can both be decoded for chip selection.

4. Extended-Page Boundary

By decoding $\overline{CAS}$, the page boundary can be extended beyond the 256 column locations in a single chip. In this case, $\overline{RAS}$ must be applied to all devices.

Page-Mode Operation

This operation allows for multiple-column addressing at the same row address, and eliminates the power dissipation associated with the negative-going edge of $\overline{RAS}$, because once the row address has been strobed, $\overline{RAS}$ is maintained. Also, the time required to strobe in the row address for the second and subsequent cycles is eliminated, thereby decreasing the access and cycle times.

Refresh

Each of the 128 rows ($A_0 \sim A_6$) of the M5K4164NP must be refreshed every 2 ms to maintain data. The methods of refreshing for the M5K4164NP are as follows.

1. Normal Refresh

Read cycle and Write cycle (early write, delayed write or read-modify-write) refresh the selected row as defined by the low order ($\overline{RAS}$) addresses. Any write cycle, of course, may change the state of the selected cell. Using a read, write, or read-modify-write cycle for refresh is not recommended for systems which utilize "write-OR" outputs since output bus contention will occur.

2. $\overline{RAS}$ Only Refresh

A $\overline{RAS}$-only refresh cycle is the recommended technique for most applications to provide for data retention. A $\overline{RAS}$-only refresh cycle maintains the output in the high-impedance state with a typical power reduction of 20% over a read or write cycle.

3. Hidden Refresh

A features of the M5K4164NP is that refresh cycles may be performed while maintaining valid data at the output pin by extending the $\overline{CAS}$ active time from a previous memory read cycle. This feature is refered to as hidden refresh.

Hidden refresh is performed by holding $\overline{CAS}$ at V_{IL} and taking $\overline{RAS}$ high and after a specified precharge period, executing a $\overline{RAS}$-only cycling, but with $\overline{CAS}$ held low.

The advantage of this refresh mode is that data can be held valid at the output data port indefinitely by leaving the $\overline{CAS}$ asserted. In many applications this eliminates the need for off-chip latches.

Power Dissipation

Most of the circuitry in the M5K4164NP is dynamic, and most of the power is dissipated when addresses are strobed. Both $\overline{RAS}$ and $\overline{CAS}$ are decoded and applied to the M5K4164NP as chip-select in the memory system, but if $\overline{RAS}$ is decoded, all unselected devices go into stand-by independent of the $\overline{CAS}$ condition, minimizing system power dissipation.

Power Supplies

The M5K4164NP operates on a single 5V power supply.

A wait of some 500μs and eight or more dummy cycles is necessary after power is applied to the device before memory operation is achieved.

ABSOLUTE MAXIMUM RATINGS

Symbol	Paramater	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	$-1 \sim 7$	V
V_I	Input voltage		$-1 \sim 7$	V
V_O	Output voltage		$-1 \sim 7$	V
I_O	Output current		50	mA
Pd	Power dissipation	Ta = 25°C	700	mW
Topr	Operating free-air temperature range		$0 \sim 70$	°C
Tstg	Storage temperature range		$-65 \sim 150$	°C

RECOMMENDED OPERATING CONDITIONS (Ta = 0 ~ 70°C, unless otherwise noted) (Note 1)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage	0	0	0	V
V_{IH}	High-level input voltage, all inputs	2.4		6.5	V
V_{IL}	Low-level input voltage, all inputs	−2		0.8	V

Note 1: All voltage values are with respect to V_{SS}

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 70°C, V_{CC} = 5V ± 10%, V_{SS} = 0V, unless otherwise noted) (Note 2)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
V_{OH}	High-level output voltage		$I_{OH} = -5$mA	2.4		V_{CC}	V
V_{OL}	Low-level output voltage		$I_{OL} = 4.2$mA	0		0.4	V
I_{OZ}	Off-state output current		Q floating 0V ≦ V_{OUT} ≦ 5.5V	−10		10	μA
I_I	Input current		0V ≦ V_{IN} ≦ 6.5V, All other pins = 0V	−10		10	μA
$I_{CC1(AV)}$	Average supply current from V_{CC}, operating (Note 3, 4)	M5K4164NP-15	$\overline{RAS}$, $\overline{CAS}$ cycling			50	mA
		M5K4164NP-20	$t_{CR} = t_{CW}$ = min output open			45	mA
I_{CC2}	Supply current from V_{CC}, standby		$\overline{RAS} = V_{IH}$ output open			4	mA
$I_{CC3(AV)}$	Average supply current from V_{CC}, refreshing (Note 3)	M5K4164NP-15	$\overline{RAS}$ cycling $\overline{CAS} = V_{IH}$			40	mA
		M5K4164NP-20	$t_{C(\overline{REF})}$ = min, output open			35	mA
$I_{CC4(AV)}$	Average supply current from V_{CC}, page mode (Note 3, 4)	M5K4164NP-15	$\overline{RAS} = V_{IL}$, $\overline{CAS}$ cycling			40	mA
		M5K4164NP-20	t_{CPG} = min, output open			35	mA
$C_{I(A)}$	Input capacitance, address inputs		$V_I = V_{SS}$ f = 1MHz V_I = 25mVrms			5	pF
$C_{I(D)}$	Input capacitance, data input					5	pF
$C_{I(W)}$	Input capacitance, write control input					7	pF
$C_{I(RAS)}$	Input capacitance, $\overline{RAS}$ input					10	pF
$C_{I(CAS)}$	Input capacitance, $\overline{CAS}$ input					10	pF
C_O	Output capacitance		$V_O = V_{SS}$, f = 1MHz, V_I = 25mVrms			7	pF

Note 2: Current flowing into an IC is positive ; out is negative.
3: $I_{CC1(AV)}$, $I_{CC3(AV)}$, and $I_{CC4(AV)}$ are dependent on cycle rate. Maximum current is measured at the fastest cycle rate.
4: $I_{CC1(AV)}$ and $I_{CC4(AV)}$ are dependent on output loading. Specified values are obtained with the output open.

TIMING REQUIREMENTS (For Read, Write, Read-Modify-Write, Refresh, and Page-Mode Cycle)

($T_a = 0 \sim 70°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted, See notes 5, 6 and 7)

Symbol	Parameter	Alternative Symbol	M5K4164NP-15		M5K4164NP-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{CRF}	Refresh cycle time	t_{REF}		2		2	ms
$t_{w(RASH)}$	RAS high pulse width	t_{RP}	100		120		ns
$t_{w(RASL)}$	RAS low pulse width	t_{RAS}	150	10000	200	10000	ns
$t_{w(CASL)}$	CAS low pulse width	t_{CAS}	75	∞	100	∞	ns
$t_{w(CASH)}$	CAS high pulse width (Note 8)	t_{CPN}	35		40		ns
$t_{h(RAS-CAS)}$	CAS hold time after RAS	t_{CSH}	150		200		ns
$t_{h(CAS-RAS)}$	RAS hold time after CAS	t_{RSH}	75		100		ns
$t_{d(CAS-RAS)}$	Delay time, CAS to RAS (Note 9)	t_{CRP}	−20		−20		ns
$t_{d(RAS-CAS)}$	Delay time, RAS to CAS (Note 10)	t_{RCD}	25	75	30	100	ns
$t_{su(RA-RAS)}$	Row address setup time before RAS	t_{ASR}	0		0		ns
$t_{su(CA-CAS)}$	Column address setup time before CAS	t_{ASC}	−5		−5		ns
$t_{h(RAS-RA)}$	Row address hold time after RAS	t_{RAH}	20		25		ns
$t_{h(CAS-CA)}$	Column address hold time after CAS	t_{CAH}	25		35		ns
$t_{h(RAS-CA)}$	Column address hold time after RAS	t_{AR}	95		120		ns
t_{THL} t_{TLH}	Transition time	t_T	3	35	3	50	ns

Note 5: An initial pause of 500μs is required after power-up followed by any eight RAS or RAS/CAS cycles before proper device operation is achieved.
6: The switching characteristics are defined as $t_{THL} = t_{TLH} = 5ns$.
7: Reference levels of input signals are $V_{IH\ min}$ and $V_{IL\ max}$. Reference levels for transition time are also between V_{IH} and V_{IL}.
8: Except for page-mode.
9: $t_{d(CAS-RAS)}$ requirement is only applicable for RAS/CAS cycles preceeded by a CAS only cycle (i,e., For systems where CAS has not been decoded with RAS.)
10: Operation within the $t_{d(RAS-CAS)}$ max limit insures that $t_{a(RAS)}$max can be met. $t_{d(RAS-CAS)}$max is specified reference point only;if $t_{d(RAS-CAS)}$ is greater than the specified $t_{d(RAS-CAS)}$ max limit, then access time is controlled exclusively by $t_{a(CAS)}$.
$t_{d(RAS-CAS)}min = t_{h(RAS-RA)}min + 2t_{THL}(t_{TLH}) + t_{su(CA-CAS)}min$.

SWITCHING CHARACTERISTICS ($T_a = 0 \sim 70°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted)

Read Cycle

Symbol	Parameter	Alternative Symbol	M5K4164NP-15		M5K4164NP-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{CR}	Read cycle time	t_{RC}	260		330		ns
$t_{su(R-CAS)}$	Read setup time before CAS	t_{RCS}	0		0		ns
$t_{h(CAS-R)}$	Read hold time after CAS (Note 11)	t_{RCH}	0		0		ns
$t_{h(RAS-R)}$	Read hold time after RAS (Note 11)	t_{RRH}	20		25		ns
$t_{dis(CAS)}$	Output disable time (Note 12)	t_{OFF}	0	40	0	50	ns
$t_{a(CAS)}$	CAS access time (Note 13)	t_{CAC}		75		100	ns
$t_{a(RAS)}$	RAS access time (Note 14)	t_{RAC}		150		200	ns

Note 11: Either $t_{h(RAS-R)}$ or $t_{h(CAS-R)}$ must be satisfied for a read cycle.
Note 12: $t_{dis(CAS)}$max defines the time at which the output achieves the open circuit condition and is not reference to V_{OH} or V_{OL}.
Note 13: This is the value when $t_{d(RAS-CAS)} \geqq t_{d(RAS-CAS)}max$. Test conditions ; Load = 2T TL, C_L = 100pF
Note 14: This is the value when $t_{d(RAS-CAS)} < t_{d(RAS-CAS)}max$. When $t_{d(RAS-CAS)} \geqq t_{d(RAS-CAS)}max$, $t_{a(RAS)}$ will increase by the amount that $t_{d(RAS-CAS)}$ exceeds the value shown. Test conditions ; Load = 2T TL, C_L = 100pF

Write Cycle

Symbol	Parameter	Alternative Symbol	M5K4164NP-15		M5K4164NP-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{CW}	Write cycle time	t_{RC}	260		330		ns
$t_{su(W-CAS)}$	Write setup time before CAS (Note 17)	t_{WCS}	−10		−10		ns
$t_{h(CAS-W)}$	Write hold time after CAS	t_{WCH}	45		55		ns
$t_{h(RAS-W)}$	Write hold time after RAS	t_{WCR}	95		120		ns
$t_{h(W-RAS)}$	RAS hold time after write	t_{RWL}	45		55		ns
$t_{h(W-CAS)}$	CAS hold time after write	t_{CWL}	45		55		ns
$t_{w(W)}$	Write pulse width	t_{WP}	45		55		ns
$t_{su(D-CAS)}$	Data-in setup time before CAS	t_{DS}	0		0		ns
$t_{h(CAS-D)}$	Data-in hold time after CAS	t_{DH}	45		55		ns
$t_{h(RAS-D)}$	Data-in hold time after RAS	t_{DHR}	95		120		ns

Read-Write and Read-Modify-Write Cycles

Symbol	Parameter	Alternative Symbol	M5K4164NP-15 Limits Min	M5K4164NP-15 Limits Max	M5K4164NP-20 Limits Min	M5K4164NP-20 Limits Max	Unit
t_{CRW}	Read-write cycle time (Note 15)	t_{RWC}	280		340		ns
t_{CRMW}	Read-modify-write cycle time (Note 16)	t_{RMWC}	310		390		ns
$th_{(W-RAS)}$	$\overline{RAS}$ hold time after write	t_{RWL}	45		55		ns
$th_{(W-CAS)}$	$\overline{CAS}$ hold time after write	t_{CWL}	45		55		ns
$tw_{(W)}$	Write pulse width	t_{WP}	45		55		ns
$tsu_{(R-CAS)}$	Read setup time before $\overline{CAS}$	t_{RCS}	0		0		ns
$td_{(RAS-W)}$	Delay time, $\overline{RAS}$ to write (Note 17)	t_{RWD}	120		150		ns
$td_{(CAS-W)}$	Delay time, $\overline{CAS}$ to write (Note 17)	t_{CWD}	60		80		ns
$tsu_{(D-W)}$	Data-in setup time before write	t_{DS}	0		0		ns
$th_{(W-D)}$	Data-in hold time after write	t_{DH}	45		55		ns
$tdis_{(CAS)}$	Output disable time	t_{OFF}	0	40	0	50	ns
$ta_{(CAS)}$	$\overline{CAS}$ access time (Note 13)	t_{CAC}		75		100	ns
$ta_{(RAS)}$	$\overline{RAS}$ access time (Note 14)	t_{RAC}		150		200	ns

Note 15: t_{CRW}min is defined as $t_{CRW}\min = td_{(RAS-W)} + th_{(W-RAS)} + tw_{(RASH)} + 3t_{TLH(tTHL)}$

16: t_{CRMW} min is defined as $t_{CRMW}\min = ta_{(RAS)}\max + th_{(W-RAS)} + tw_{(RAS\,H)} + 3t_{TLH(tTHL)}$

17: $tsu_{(W-CAS)}$, $td_{(RAS-W)}$, and $td_{(CAS-W)}$ do not define the limits of operation, but are included as electrical characteristics only.

When $tsu_{(W-CAS)} \geqq tsu_{(W-CAS)}\min$, an early-write cycle is performed, and the data output keeps the high-impedance state.

When $td_{(RAS-W)} \geqq td_{(RAS-W)}\min$, and $td_{(CAS-W)} \geqq tsu_{(W-CAS)}\min$ a read-write cycle is performed, and the data of the selected address will be read out on the data output.

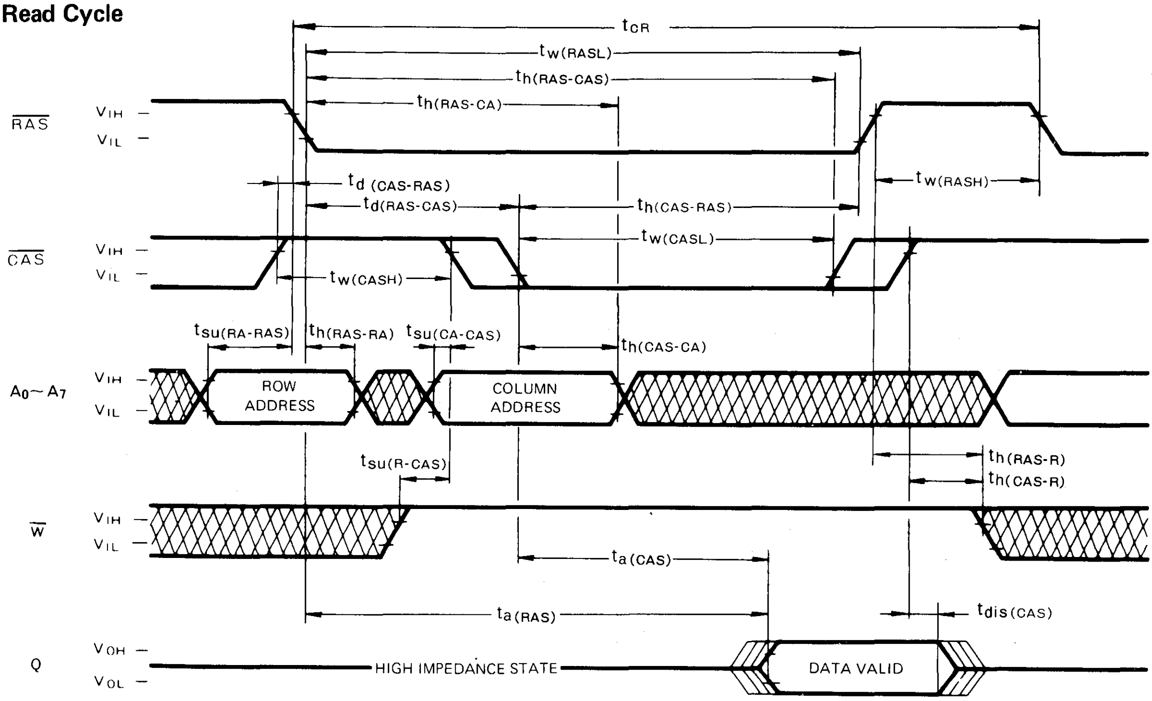
For all conditions other than those described above, the condition of data output (at access time and until $\overline{CAS}$ goes back to V_{IH}) is not defined.

Page-Mode Cycle

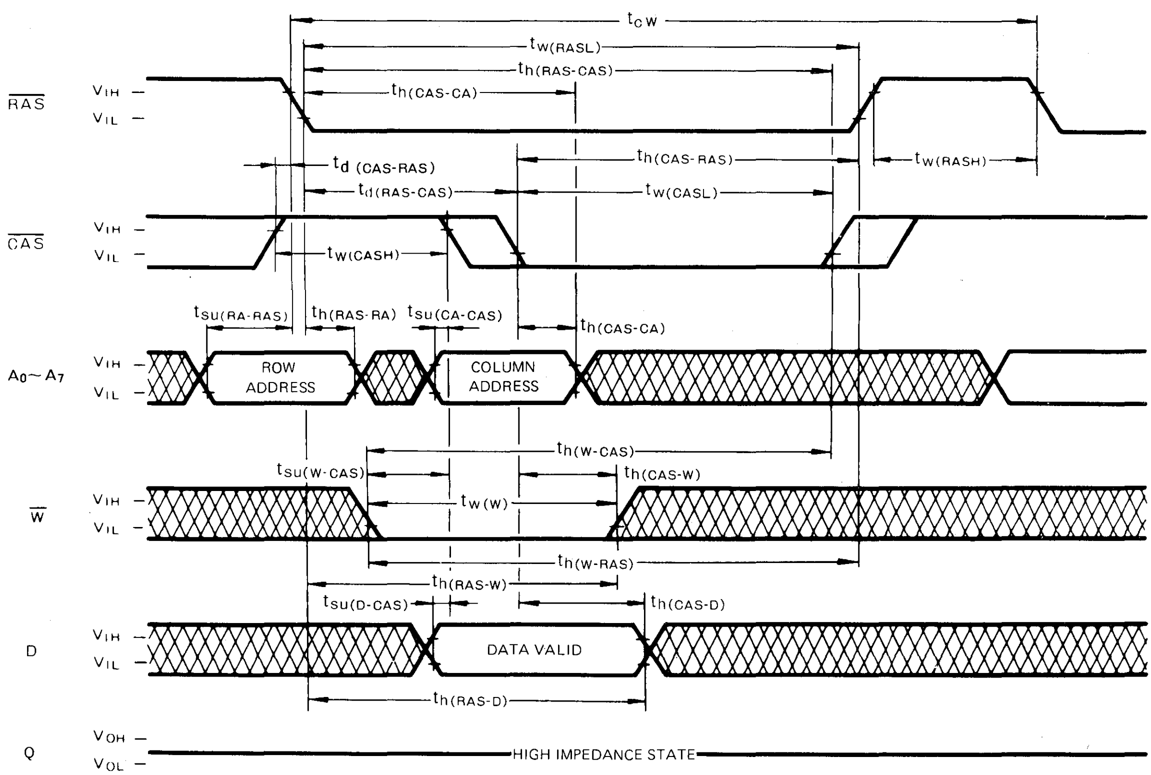
Symbol	Parameter	Alternative Symbol	M5K4164NP-15 Limits Min	M5K4164NP-15 Limits Max	M5K4164NP-20 Limits Min	M5K4164NP-20 Limits Max	Unit
$t_{C\,PGR}$	Page-mode read cycle time	t_{PC}	145		190		ns
$t_{C\,PGW}$	Page-Mode write cycle time	t_{PC}	145		190		ns
$t_{C\,PGRW}$	Page-Mode read-write cycle time	—	180		230		ns
$t_{C\,PGRMW}$	Page-Mode read-modify-write cycle time	—	190		245		ns
$tw_{(CASH)}$	$\overline{CAS}$ high pulse width	t_{CP}	60		80		ns

TIMING DIAGRAMS (Note 17)

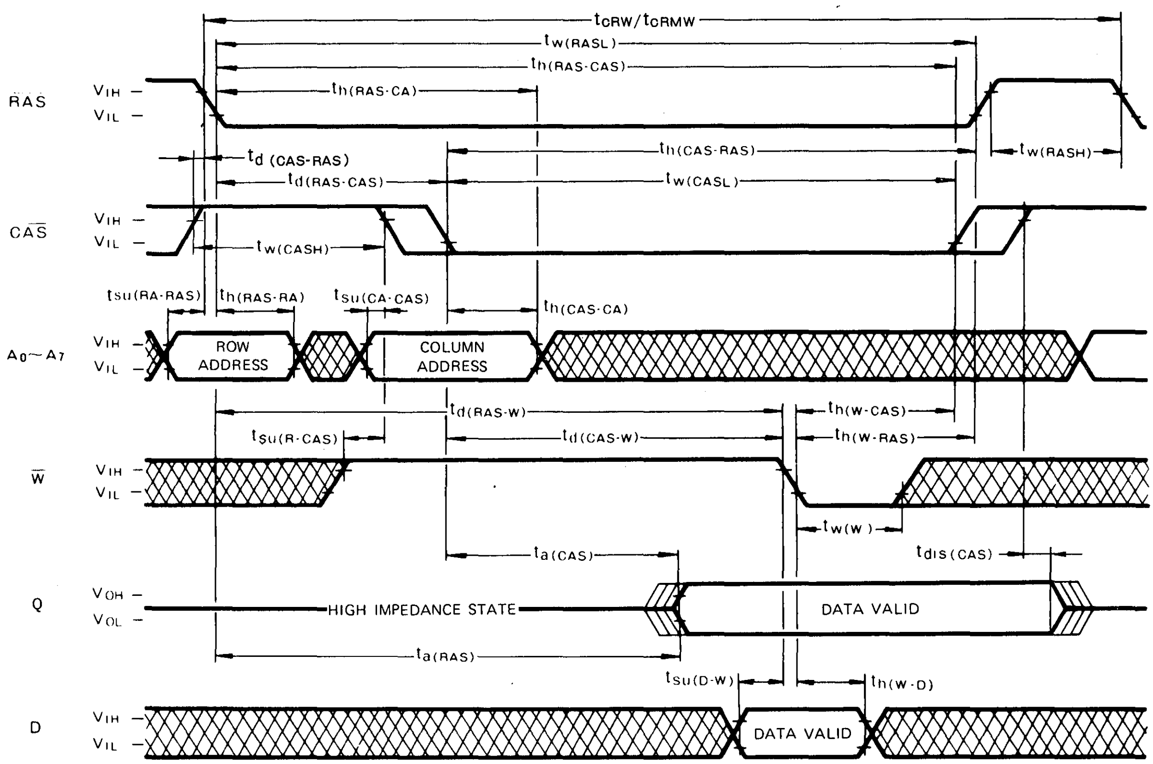
Read Cycle



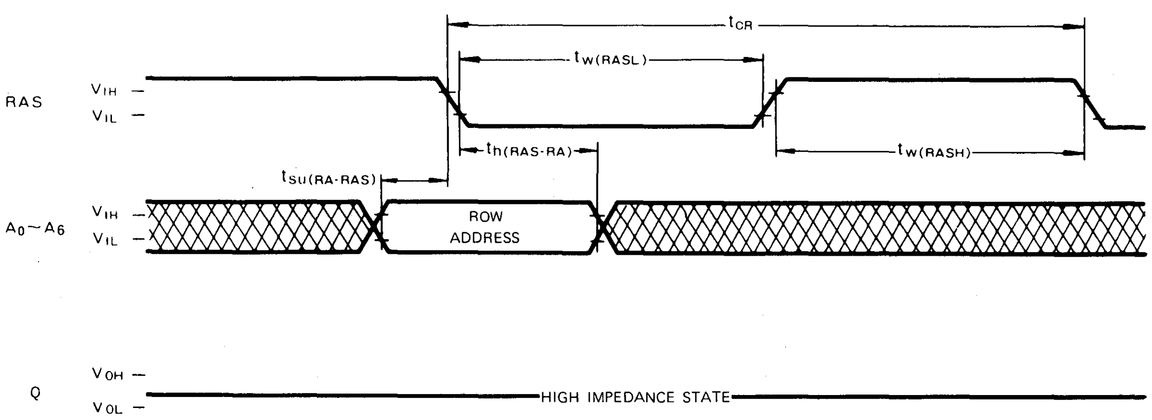
Write Cycle (Early Write)



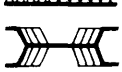
Read-Write and Read-Modify-Write Cycles



RAS-Only Refresh Cycle (Note 18)



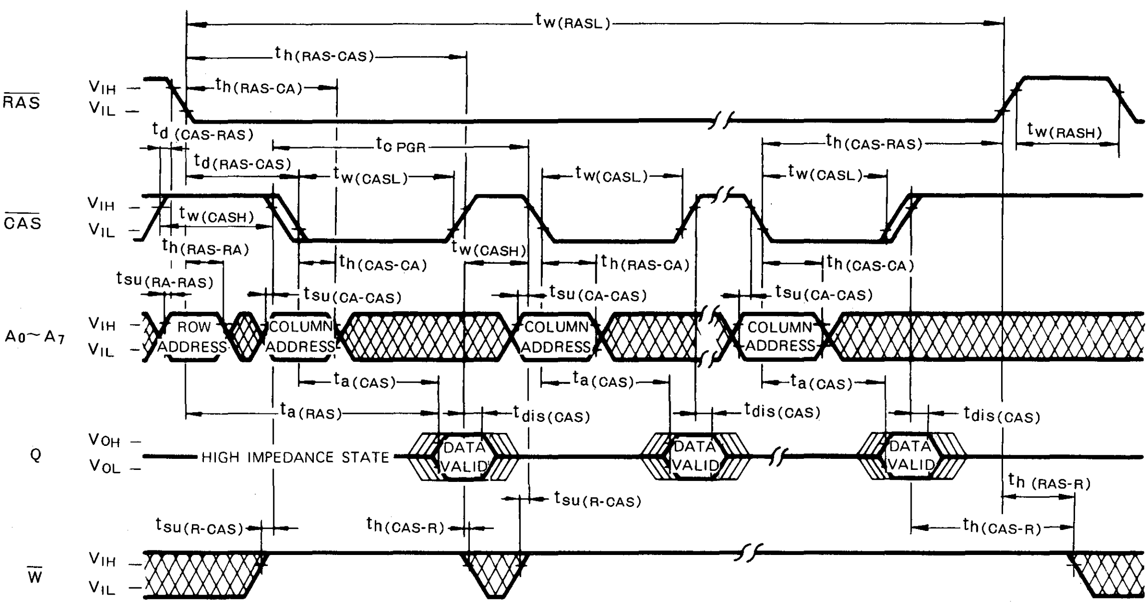
Note 17  Indicates the don't care input

 The center-line indicates the high-impedance state

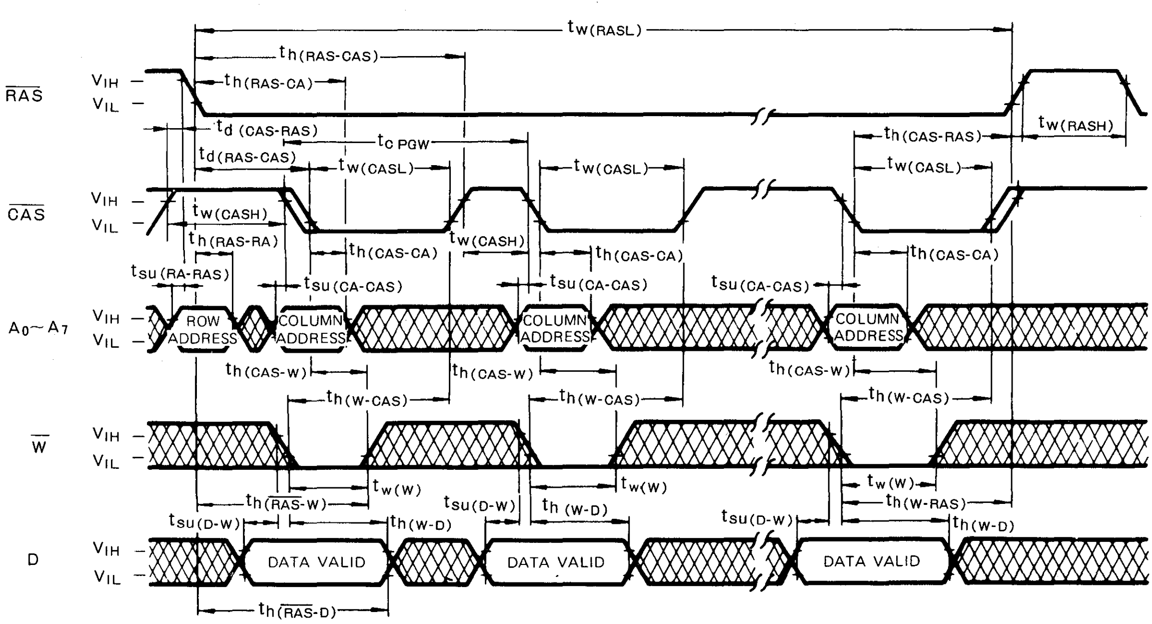
Note 18. $\overline{CAS} = V_{IH}$, $\overline{W}$, A_7, D = don't care.

2

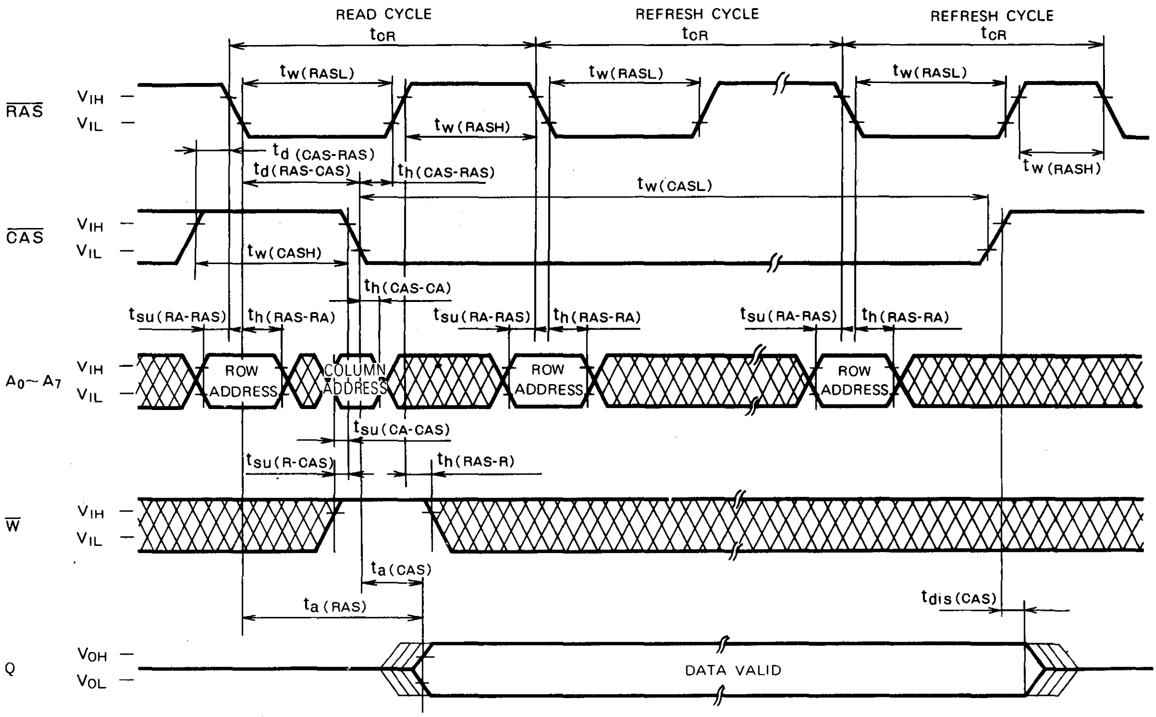
Page-Mode Read Cycle



Page-Mode Write Cycle



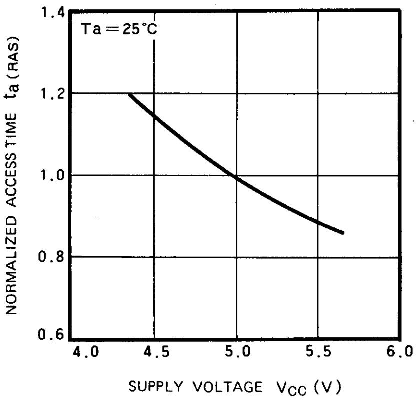
Hidden Refresh Cycle



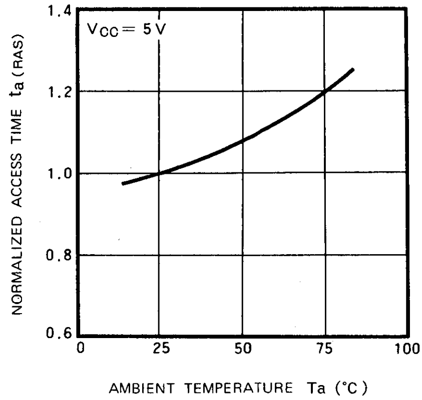
MITSUBISHI ELECTRIC

TYPICAL CHARACTERISTICS

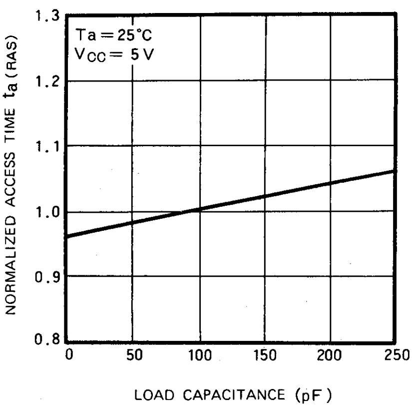
NORMALIZED ACCESS TIME VS. V_{CC} SUPPLY VOLTAGE



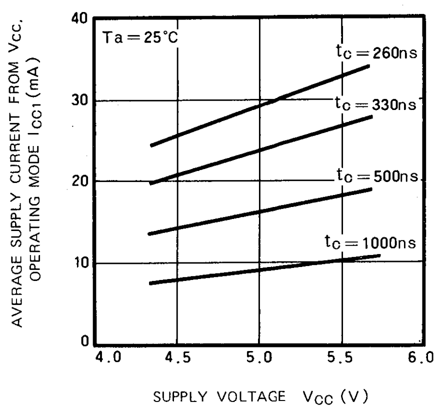
NORMALIZED ACCESS TIME VS. AMBIENT TEMPERATURE



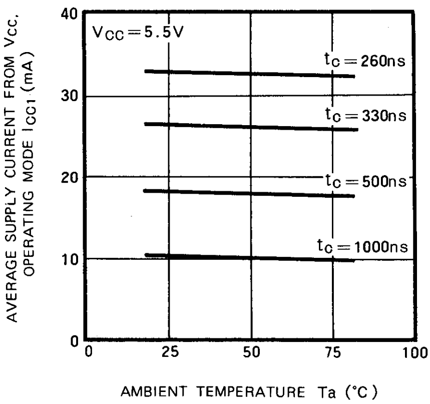
NORMALIZED ACCESS TIME VS. LOAD CAPACITANCE



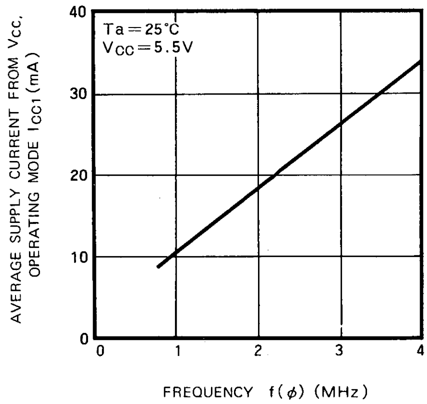
AVERAGE SUPPLY CURRENT FROM V_{CC}, OPERATING MODE VS. SUPPLY VOLTAGE



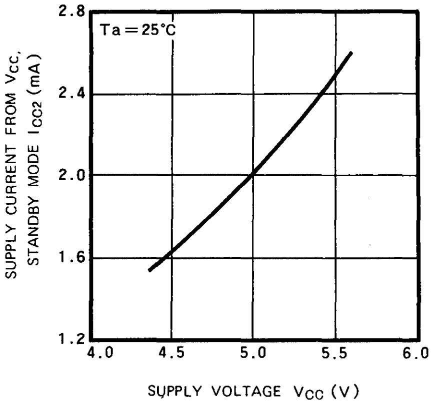
AVERAGE SUPPLY CURRENT FROM V_{CC}, OPERATING MODE VS. AMBIENT TEMPERATURE



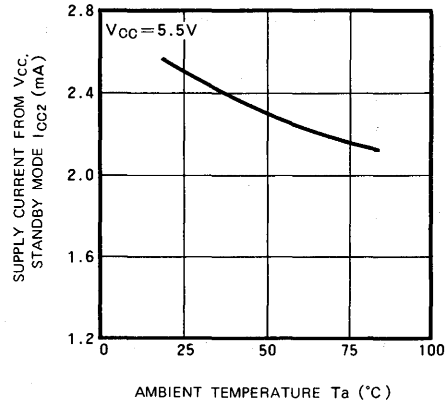
AVERAGE SUPPLY CURRENT FROM V_{CC}, OPERATING MODE VS. FREQUENCY



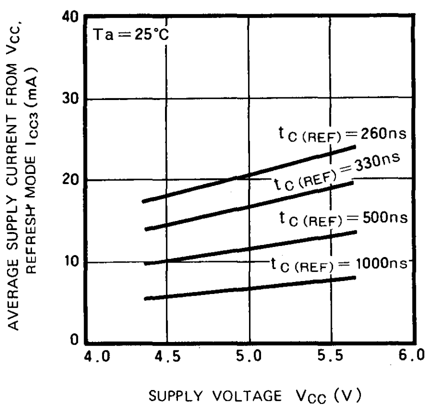
SUPPLY CURRENT FROM V_{CC}, STANDBY MODE VS. SUPPLY VOLTAGE



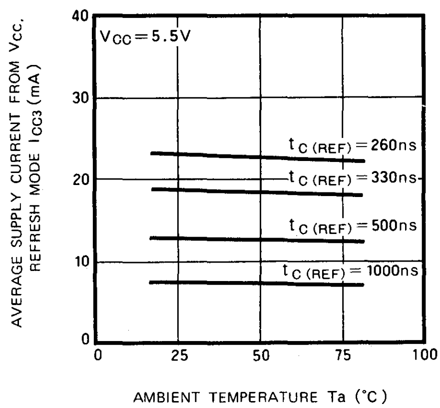
SUPPLY CURRENT FROM V_{CC}, STANDBY MODE VS. AMBIENT TEMPERATURE



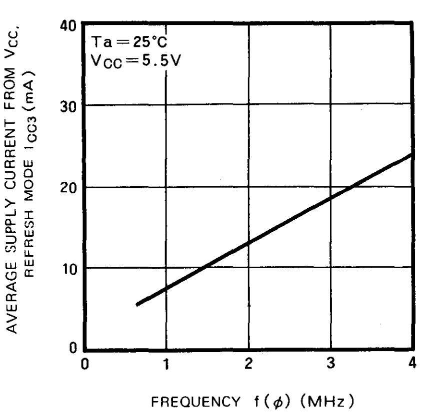
AVERAGE SUPPLY CURRENT FROM V_{CC}, REFRESH MODE VS. SUPPLY VOLTAGE



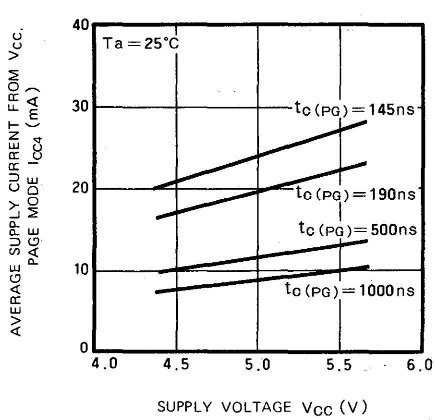
AVERAGE SUPPLY CURRENT FROM V_{CC}, REFRESH MODE VS. AMBIENT TEMPERATURE



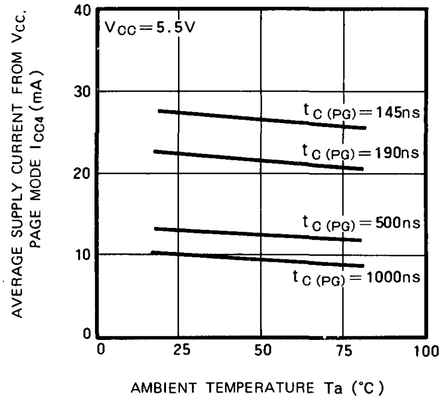
AVERAGE SUPPLY CURRENT FROM V_{CC}, REFRESH MODE VS. FREQUENCY



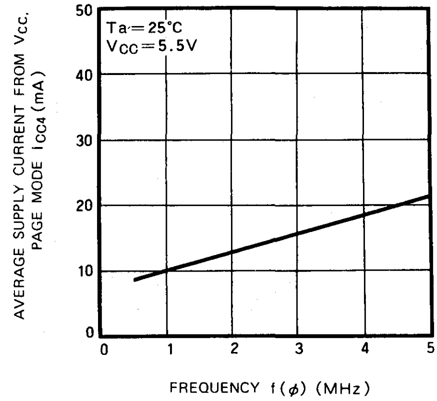
AVERAGE SUPPLY CURRENT FROM V_{CC}, PAGE MODE VS. SUPPLY VOLTAGE



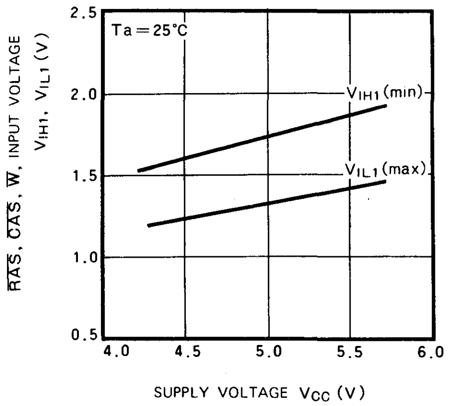
AVERAGE SUPPLY CURRENT FROM V_{CC}, PAGE MODE VS. AMBIENT TEMPERATURE



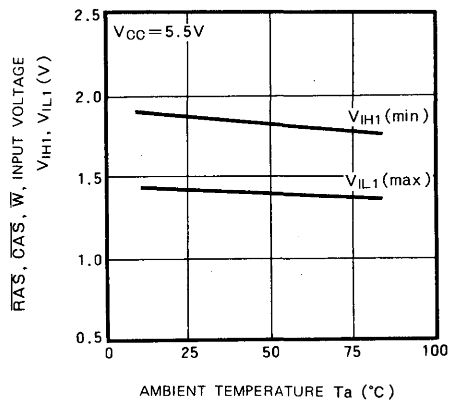
AVERAGE SUPPLY CURRENT FROM V_{CC}, PAGE MODE VS. FREQUENCY



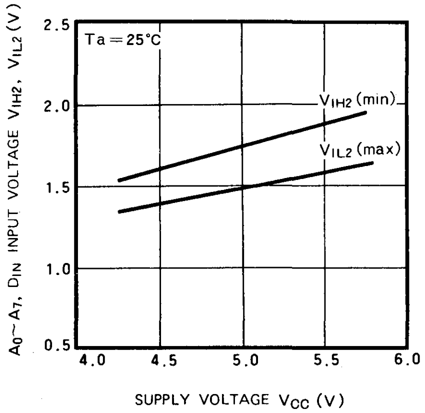
$\overline{RAS}$, $\overline{CAS}$, $\overline{W}$, INPUT VOLTAGE V_{IH1}, V_{IL1} VS. SUPPLY VOLTAGE



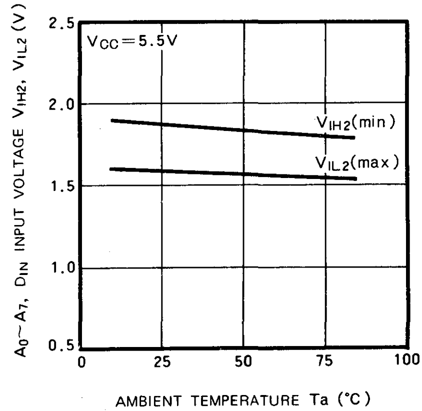
$\overline{RAS}$, $\overline{CAS}$, $\overline{W}$, INPUT VOLTAGE V_{IH1}, V_{IL1} VS. AMBIENT TEMPERATURE



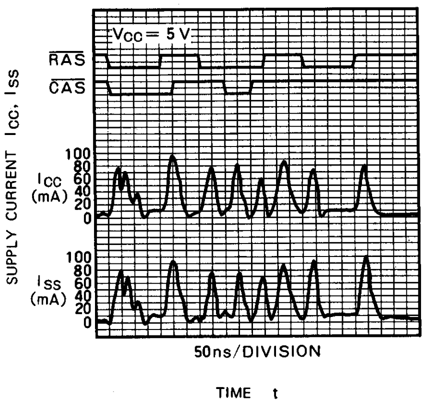
$A_0 \sim A_7$, D_{IN} INPUT VOLTAGE V_{IH2}, V_{IL2} VS. SUPPLY VOLTAGE



$A_0 \sim A_7$, D_{IN} INPUT VOLTAGE V_{IH2}, V_{IL2} VS. AMBIENT TEMPERATURE



SUPPLY CURRENT VS. TIME

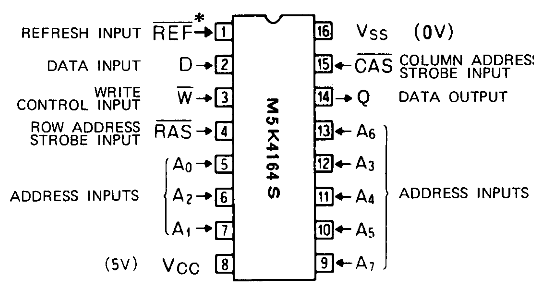


65 536-BIT (65 536-WORD BY 1-BIT) DYNAMIC RAM

DESCRIPTION

This is a family of 65 536-word by 1-bit dynamic RAMs, fabricated with the high performance N-channel silicon-gate MOS process, and is ideal for large-capacity memory systems where high speed, low power dissipation, and low costs are essential. The use of double-layer polysilicon process technology and a single-transistor dynamic storage cell provide high circuit density at reduced costs, and the use of dynamic circuitry including sense amplifiers assures low power dissipation. Multiplexed address inputs permit both a reduction in pins to the standard 16-pin package configuration and an increase in system densities. The M5K4164S operates on a 5V power supply using the on-chip substrate bias generator.

PIN CONFIGURATION (TOP VIEW)



Outline 16S1

* If the pin 1 (REF) function is not used, pin 1 may be left open (not connect).

FEATURES

- Performance ranges

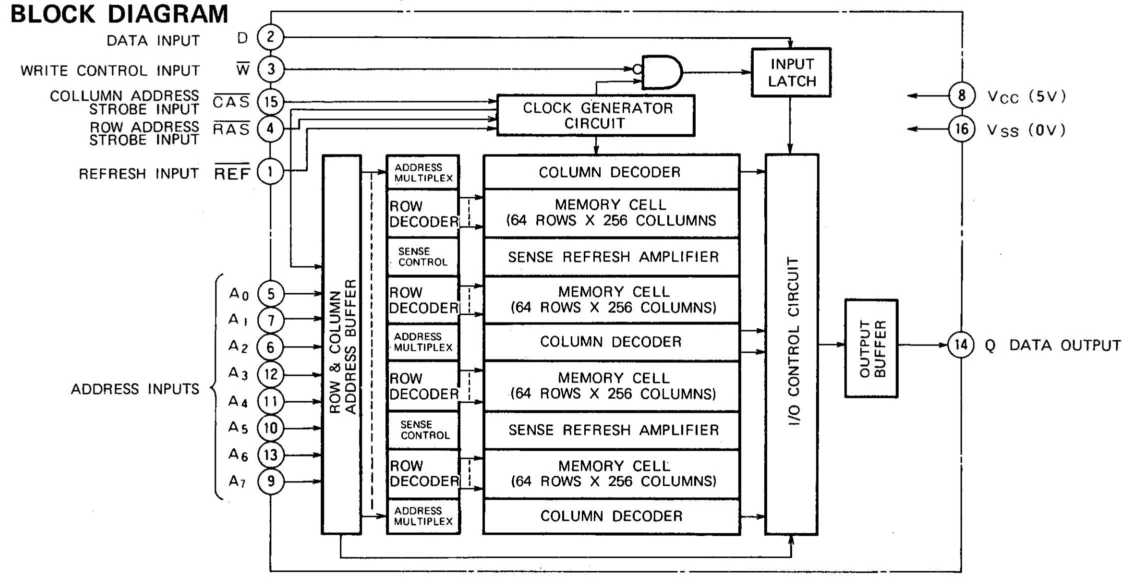
Type name	Access time (max) (ns)	Cycle time (min) (ns)	Power dissipation (typ) (mW)
M5K4164S-15	150	260	200
M5K4164S-20	200	330	170

- Standard 16-pin package
- Single 5V ±10% supply
- Low standby power dissipation: 28mW (max)
- Low operating power dissipation:
  M5K4164S-15 275mW (max)
  M5K4164S-20 250mW (max)
- Unlatched output enables two-dimensional chip selection and extended page boundary
- Early-write operation gives common I/O capability
- Read-modify-write, RAS-only refresh, and page-mode capabilities
- All input terminals have low input capacitance and are directly TTL-compatible
- Output is three-state and directly TTL-compatible
- 128 refresh cycles every 2ms (16K dynamic RAMs M5K4116P, S compatible)
- Pin 1 controls automatic- and self-refresh mode.
- CAS controlled output allows hidden refresh, hidden automatic refresh and hidden self-refresh.
- Output data can be held infinitely by CAS.
- Interchangeable with Mostek's MK4164 and Motorola's MCM 6664 in pin configuration.

APPLICATION

- Main memory unit for computers.

BLOCK DIAGRAM



FUNCTION

The M5K4164S provides, in addition to normal read, write, and read-modify-write operations, a number of other functions, e.g., page mode, $\overline{\text{RAS}}$-only refresh, and delayed-write. The input conditions for each are shown in Table 1.

Table 1 Input conditions for each mode

Operation	Inputs							Output	Refresh	Remarks
	$\overline{\text{RAS}}$	$\overline{\text{CAS}}$	$\overline{\text{W}}$	D	Row address	Column address	$\overline{\text{REF}}$	Q		
Read	ACT	ACT	NAC	DNC	APD	APD	NAC	VLD	YES	Page mode identical except refresh is NO.
Write	ACT	ACT	ACT	VLD	APD	APD	NAC	OPN	YES	
Read-modify-write	ACT	ACT	ACT	VLD	APD	APD	NAC	VLD	YES	
$\overline{\text{RAS}}$-only refresh	ACT	NAC	DNC	DNC	APD	DNC	NAC	OPN	YES	
Hidden refresh	ACT	ACT	DNC	DNC	APD	DNC	NAC	VLD	YES	
Automatic refresh	NAC	DNC	DNC	DNC	DNC	DNC	ACT	OPN	YES	
Self refresh	NAC	DNC	DNC	DNC	DNC	DNC	ACT	OPN	YES	
Hidden automatic refresh	NAC	ACT	DNC	DNC	DNC	DNC	ACT	VLD	YES	
Hidden self refresh	NAC	ACT	DNC	DNC	DNC	DNC	ACT	VLD	YES	
Standby	NAC	DNC	DNC	DNC	DNC	DNC	NAC	OPN	NO	

Note: ACT : active, NAC : nonactive, DNC : don't care, VLD : valid, APD : applied, OPN : open.

SUMMARY OF OPERATIONS

Addressing

To select one of the 65 536 memory cells in the M5K4164S the 16-bit address signal must be multiplexed into 8 address signals, which are then latched into the on-chip latch by two externally-applied clock pulses. First, the negative-going edge of the row-address-strobe pulse ($\overline{\text{RAS}}$) latches the 8 row-address bits; next, the negative-going edge of the column-address-strobe pulse ($\overline{\text{CAS}}$) latches the 8 column-address bits. Timing of the $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ clocks can be selected by either of the following two methods:

1. The delay time from $\overline{\text{RAS}}$ to $\overline{\text{CAS}}$ $t_{d(RAS-CAS)}$ is set between the minimum and maximum values of the limits. In this case, the internal $\overline{\text{CAS}}$ control signals are inhibited almost until $t_{d(RAS-CAS)max}$ ('gated $\overline{\text{CAS}}$' operation). The external $\overline{\text{CAS}}$ signal can be applied with a margin not affecting the on-chip circuit operations, e.g. access time, and the address inputs can be easily changed from row address to column address.
2. The delay time $t_{d(RAS-CAS)}$ is set larger than the maximum value of the limits. In this case the internal inhibition of $\overline{\text{CAS}}$ has already been released, so that the internal $\overline{\text{CAS}}$ control signals are controlled by the externally applied $\overline{\text{CAS}}$, which also controls the access time.

Data Input

Date to be written into a selected cell is strobed by the later of the two negative transitions of $\overline{\text{W}}$ input and $\overline{\text{CAS}}$ input. Thus when the $\overline{\text{W}}$ input makes its negative transition prior to $\overline{\text{CAS}}$ input (early write), the data input is strobed by $\overline{\text{CAS}}$, and the negative transition of $\overline{\text{CAS}}$ is set as the reference point for set-up and hold times. In the read-write or read-modify-write cycles, however, when the $\overline{\text{W}}$ input makes its negative transition after $\overline{\text{CAS}}$, the $\overline{\text{W}}$ negative transition is set as the reference point for setup and hold times.

Data Output Control

The output of the M5K4164S is in the high-impedance state when $\overline{\text{CAS}}$ is high. When the memory cycle in progress is a read, read-modify-write, or a delayed-write cycle, the data output will go from the high-impedance state to the active condition, and the data in the selected cell will be read. This data output will have the same polarity as the input data. Once the output has entered the active condition, this condition will be maintained until $\overline{\text{CAS}}$ goes high, irrespective of the condition of $\overline{\text{RAS}}$.

The output will remain in the high-impedance state throughout the entire cycle in an early-write cycle.

These output conditions, of the M5K4164S, which can readily be changed by controlling the timing of the write pulse in a write cycle, and the width of the $\overline{\text{CAS}}$ pulse in a read cycle, offer capabilities for a number of applications, as follows.

1. Common I/O Operation

If all write operations are performed in the early-write mode, input and output can be connected directly to give a common I/O data bus.

2. Data Output Hold

The data output can be held between read cycles, without lengthening the cycle time. This enables extremely flexible clock-timing settings for $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$.

3. Two Methods of Chip Selection

Since the output is not latched, CAS is not required to keep the outputs of selected chips in the matrix in a high-impedance state. This means that CAS and/or RAS can both be decoded for chip selection.

4. Extended-Page Boundary

By decoding CAS, the page boundary can be extended beyond the 256 column locations in a single chip. In this case, RAS must be applied to all devices.

Page-Mode Operation

This operation allows for multiple-column addressing at the same row address, and eliminates the power dissipation associated with the negative-going edge of RAS, because once the row address has been strobed, RAS is maintained. Also, the time required to strobe in the row address for the second and subsequent cycles is eliminated, thereby decreasing the access and cycle times.

Refresh

Each of the 128 rows ($A_0 \sim A_6$) of the M5K4164S must be refreshed every 2 ms to maintain data. The methods of refreshing for the M5K4164S are as follows.

1. Normal Refresh

Read cycle and Write cycle (early write, delayed write or read-modify-write) refresh the selected row as defined by the low order (RAS) addresses. Any write cycle, of course, may change the state of the elected cell. Using a read, write, or read-modify-write cycle for refresh is not recommended for systems which utilize "wire-OR" outputs since output bus contention will occur.

2. RAS Only Refresh

A RAS-only refresh cycle is the recommended technique for most applications to provide for data retention. A RAS-only refresh cycle maintains the output in the high-impedance state with a typical power reduction of 20% over a read or write cycle.

3. Automatic Refresh

Pin 1 (REF) has two special functions. The M5K4164S has a refresh address counter, refresh address multiplexer and refresh timer for these operations. Automatic refresh is initiated by bringing REF low after RAS has precharged and is used during standard operation just like RAS-only refresh, except that sequential row addresses from an external counter are no longer necessary.

At the end of automatic refresh cycle, the internal refresh address counter will be automatically incremented. The output state of the refresh address counter is initiated by some eight REF, RAS or RAS/CAS cycles after power is applied. Therefore, a special operation is not necessary to initiate it.

RAS must remain inactive during REF activated cycles. Likewise, REF must remain inactive during RAS generated cycle.

4. Self-Refresh

The other function of pin 1 (REF) is self-refresh. Timing for self-refresh is quite similar to that for automatic refresh. As long as RAS remains high and REF remains low, the M5K4164S will refresh itself. This internal sequence repeats asynchronously every 12 to 16 μs. After 2 ms, the on-chip refresh address counter has advanced through all the row addresses and refreshed the entire memory. Self-refresh is primarily intended for trouble free power-down operation.

For example, when battery backup is used to maintained data integrity in the memory. REF may be used to place the device in the self-refresh mode with no external timing signals necessary to keep the information alive.

In summary, the pin 1 (REF) refresh function gives the user a feature that is free, save him hardware on the board, and in fact, will simplify his battery backup procedures, increase his battery life, and save him overall cost while giving him improved system performance.

There is an internal pullup resister (≈ 3MΩ) on pin 1, so if the pin 1 (REF) function is not used, pin 1 may be left open (not connect) without affecting the normal operations.

5. Hidden Refresh

A features of the M5K4164S is that refresh cycle may be performed while maintaining valid data at the output pin by extending the CAS active time from a previous memory read cycle. This feature is refered to as hidden refresh.

Hidden refresh is performed by holding CAS at V_{IL} and taking RAS high and after a specified precharge period, executing a RAS-only cycling, automatic refresh and self-refresh, but with CAS held low.

The advantage of this refresh mode is that data can be held valid at the output data port indefinitely by leaving the CAS asserted. In many applications this eliminates the need for off-chip latches.

Power Dissipation

Most of the circuirty in the M5K4164S is dynamic, and most of the power is dissipated when addresses are strobed. Both RAS and CAS are decoded and applied to the M5K4164S as chip-select in the memory system, but if RAS is decoded, all unselected devices go into stand-by independent of the CAS condition, minimizing system power dissipation.

Power Supplies

The M5K4164S operates on a single 5V power supply.

A wait of some 500μs and eight or more dummy cycle is necessary after power is applied to the device before memory operation is achieved.

ABSOLUTE MAXIMUM RATINGS

Symbol	Paramater	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	$-1 \sim 7$	V
V_I	Input voltage		$-1 \sim 7$	V
V_O	Output voltage		$-1 \sim 7$	V
I_O	Output current		50	mA
Pd	Power dissipation	Ta = 25°C	1000	mW
Topr	Operating free-air temperature range		$0 \sim 70$	°C
Tstg	Storage temperature range		$-65 \sim 150$	°C

RECOMMENDED OPERATING CONDITIONS (Ta = 0 ~ 70°C, unless otherwise noted) (Note 1)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage	0	0	0	V
V_{IH}	High-level input voltage, all inputs	2.4		6.5	V
V_{IL}	Low-level input voltage, all inputs	−2		0.8	V

Note 1: All voltage values are with respect to V_{SS}

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 70°C, V_{CC} = 5V ± 10%, V_{SS} = 0V, unless otherwise noted) (Note 2)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
V_{OH}	High-level output voltage		$I_{OH} = -5mA$	2.4		V_{CC}	V
V_{OL}	Low-level output voltage		$I_{OL} = 4.2mA$	0		0.4	V
I_{OZ}	Off-state output current		Q floating $0V \leq V_{OUT} \leq 5.5V$	−10		10	μA
I_I	Input current		$0V \leq V_{IN} \leq 6.5V$, All other pins = 0V	−10		10	μA
$I_{CC1(AV)}$	Average supply current from V_{CC}, operating (Note 3, 4)	M5K4164S-15	$\overline{RAS}$, $\overline{CAS}$ cycling			50	mA
		M5K4164S-20	$t_{CR} = t_{CW}$ = min output open			45	mA
I_{CC2}	Supply current from V_{CC}, standby		$\overline{RAS} = V_{IH}$ output open			5	mA
$I_{CC3(AV)}$	Average supply current from V_{CC}, refreshing (Note 3)	M5K4164S-15	$\overline{RAS}$ cycling $\overline{CAS} = V_{IH}$			40	mA
		M5K4164S-20	$t_{C(REF)}$ = min, output open			35	mA
$I_{CC4(AV)}$	Average supply current from V_{CC}, page mode (Note 3, 4)	M5K4164S-15	$\overline{RAS} = V_{IL}$, $\overline{CAS}$ cycling			40	mA
		M5K4164S-20	t_{CPG} = min, output open			35	mA
$I_{CC5(AV)}$	Average supply current from V_{CC}, automatic refreshing (Note 3)	M5K4164S-15	$\overline{RAS} = V_{IH}$, $\overline{REF}$ cycling			40	mA
		M5K4164S-20	$t_{C(REF)}$ = min. output open			35	mA
$I_{CC6(AV)}$	Average supply current from V_{CC}, self refreshing		$\overline{RAS} = V_{IH}$, $\overline{REF} = V_{IL}$ output open			8	mA
$C_{I(A)}$	Input capacitance, address inputs		$V_I = V_{SS}$ f = 1MHz V_I = 25mVrms			5	pF
$C_{I(D)}$	Input capacitance, data input					5	pF
$C_{I(W)}$	Input capacitance, write control input					7	pF
$C_{I(RAS)}$	Input capacitance, $\overline{RAS}$ input					10	pF
$C_{I(CAS)}$	Input capacitance, $\overline{CAS}$ input					10	pF
$C_{I(REF)}$	Input capacitance, $\overline{REF}$ input					10	pF
C_O	Output capacitance		$V_O = V_{SS}$, f = 1MHz, V_I = 25mVrms			7	pF

Note 2: Current flowing into an IC is positive ; out is negative.
3: $I_{CC1(AV)}$, $I_{CC3(AV)}$, $I_{CC4(AV)}$ and $I_{CC5(AV)}$ are dependent on cycle rate. Maximum current is measured at the fastest cycle rate.
4: $I_{CC1(AV)}$ and $I_{CC4(AV)}$ are dependent on output loading. Specified values are obtained with the output open.

TIMING REQUIREMENTS (For Read, Write, Read-Modify-Write, Refresh, and Page-Mode Cycle)

($T_a = 0 \sim 70°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted, See notes 5, 6 and 7)

Symbol	Parameter	Alternative Symbol	M5K4164S-15		M5K4164S-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{CRF}	Refresh cycle time	t_{REF}		2		2	ms
$t_{w(RASH)}$	$\overline{RAS}$ high pulse width	t_{RP}	100		120		ns
$t_{w(RASL)}$	$\overline{RAS}$ low pulse width	t_{RAS}	150	10000	200	10000	ns
$t_{w(CASL)}$	$\overline{CAS}$ low pulse width	t_{CAS}	75	∞	100	∞	ns
$t_{w(CASH)}$	$\overline{CAS}$ high pulse width (Note 8)	t_{CPN}	35		40		ns
$t_{h(RAS-CAS)}$	$\overline{CAS}$ hold time after $\overline{RAS}$	t_{CSH}	150		200		ns
$t_{h(CAS-RAS)}$	$\overline{RAS}$ hold time after $\overline{CAS}$	t_{RSH}	75		100		ns
$t_{d(CAS-RAS)}$	Delay time, $\overline{CAS}$ to $\overline{RAS}$ (Note 9)	t_{CRP}	−20		−20		ns
$t_{d(RAS-CAS)}$	Delay time, $\overline{RAS}$ to $\overline{CAS}$ (Note 10)	t_{RCD}	25	75	30	100	ns
$t_{su(RA-RAS)}$	Row address setup time before $\overline{RAS}$	t_{ASR}	0		0		ns
$t_{su(CA-CAS)}$	Column address setup time before $\overline{CAS}$	t_{ASC}	−5		−5		ns
$t_{h(RAS-RA)}$	Row address hold time after $\overline{RAS}$	t_{RAH}	20		25		ns
$t_{h(CAS-CA)}$	Column address hold time after $\overline{CAS}$	t_{CAH}	25		35		ns
$t_{h(RAS-CA)}$	Column address hold time after $\overline{RAS}$	t_{AR}	95		120		ns
t_{THL}, t_{TLH}	Transition time	t_T	3	35	3	50	ns

Note 5: An initial pause of 500μs is required after power-up followed by any eight $\overline{REF}$, $\overline{RAS}$ or $\overline{RAS}/\overline{CAS}$ cycles before proper device operation is achieved.
6: The switching characteristics are defined as $t_{THL} = t_{TLH} = 5ns$.
7: Reference levels of input signals are $V_{IH\ min.}$ and $V_{IL\ max.}$ Reference levels for transition time are also between V_{IH} and V_{IL}.
8: Except for page-mode.
9: $t_{d(CAS-RAS)}$ requirement is only applicable for RAS/CAS cycles preceded by aCAS only cycle (i. e. for systems where CAS has not been decoded with RAS).
10. Operation within the $t_{d(RAS-CAS)}$ max limit insures that $t_{a(RAS)}$max can be met. $t_{d(RAS-CAS)}$max is specified reference point only;if $t_{d(RAS-CAS)}$ is greater than the specified $t_{d(RAS-CAS)}$ max limit, then access time is controlled exclusively by $t_{a(CAS)}$.
$t_{d(RAS-CAS)}\text{min} = t_{h(RAS-RA)}\text{min} + 2t_{THL}(t_{TLH}) + t_{su(CA-CAS)}\text{min}$.

SWITCHING CHARACTERISTICS ($T_a = 0 \sim 70°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted)

Read Cycle

Symbol	Parameter	Alternative Symbol	M5K4164S-15		M5K4164S-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{CR}	Read cycle time	t_{RC}	260		330		ns
$t_{su(R-CAS)}$	Read setup time before $\overline{CAS}$	t_{RCS}	0		0		ns
$t_{h(CAS-R)}$	Read hold time after $\overline{CAS}$ (Note 11)	t_{RCH}	0		0		ns
$t_{h(RAS-R)}$	Read hold time after $\overline{RAS}$ (Note 11)	t_{RRH}	20		25		ns
$t_{dis(CAS)}$	Output disable time (Note 12)	t_{OFF}	0	40	0	50	ns
$t_{a(CAS)}$	$\overline{CAS}$ access time (Note 13)	t_{CAC}		75		100	ns
$t_{a(RAS)}$	$\overline{RAS}$ access time (Note 14)	t_{RAC}		150		200	ns

Note 11: Either $t_{h(RAS-R)}$ or $t_{h(CAS-R)}$ must be satisfied for a read cycle.
Note 12: $t_{dis(CAS)}$max defines the time at which the output achieves the open circuit condition and is not reference to V_{OH} or V_{OL}.
Note 13: This is the value when $t_{d(RAS-CAS)} \geqq t_{d(RAS-CAS)}\text{max}$. Test conditions ; Load = 2TTL, C_L = 100pF
Note 14: This is the value when $t_{d(RAS-CAS)} < t_{d(RAS-CAS)}\text{max}$. When $t_{d(RAS-CAS)} \geqq t_{d(RAS-CAS)}\text{max}$, $t_{a(RAS)}$ will increase by the amount that $t_{d(RAS-CAS)}$ exceeds the value shown. Test conditions ; Load = 2TTL, C_L = 100pF

Write Cycle

Symbol	Parameter	Alternative Symbol	M5K4164S-15		M5K4164S-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{CW}	Write cycle time	t_{RC}	260		330		ns
$t_{su(W-CAS)}$	Write setup time before $\overline{CAS}$ (Note 17)	t_{WCS}	−10		−10		ns
$t_{h(CAS-W)}$	Write hold time after $\overline{CAS}$	t_{WCH}	45		55		ns
$t_{h(RAS-W)}$	Write hold time after $\overline{RAS}$	t_{WCR}	95		120		ns
$t_{h(W-RAS)}$	$\overline{RAS}$ hold time after write	t_{RWL}	45		55		ns
$t_{h(W-CAS)}$	$\overline{CAS}$ hold time after write	t_{CWL}	45		55		ns
$t_{w(W)}$	Write pulse width	t_{WP}	45		55		ns
$t_{su(D-CAS)}$	Data-in setup time before $\overline{CAS}$	t_{DS}	0		0		ns
$t_{h(CAS-D)}$	Data-in hold time after $\overline{CAS}$	t_{DH}	45		55		ns
$t_{h(RAS-D)}$	Data-in hold time after $\overline{RAS}$	t_{DHR}	95		120		ns

Read-Write and Read-Modify-Write Cycles

Symbol	Parameter		Alternative Symbol	M5K4164S-15		M5K4164S-20		Unit
				Limits		Limits		
				Min	Max	Min	Max	
t_{CRW}	Read-write cycle time	(Note 15)	t_{RWC}	280		340		ns
t_{CRMW}	Read-modify-write cycle time	(Note 16)	t_{RMWC}	310		390		ns
$th_{(W-RAS)}$	$\overline{RAS}$ hold time after write		t_{RWL}	45		55		ns
$th_{(W-CAS)}$	$\overline{CAS}$ hold time after write		t_{CWL}	45		55		ns
$tw_{(W)}$	Write pulse width		t_{WP}	45		55		ns
$tsu_{(R-CAS)}$	Read setup time before $\overline{CAS}$		t_{RCS}	0		0		ns
$td_{(RAS-W)}$	Delay time, $\overline{RAS}$ to write	(Note 17)	t_{RWD}	120		150		ns
$td_{(CAS-W)}$	Delay time, $\overline{CAS}$ to write	(Note 18)	t_{CWD}	60		80		ns
$tsu_{(D-W)}$	Data-in set-up time before write		t_{DS}	0		0		ns
$th_{(W-D)}$	Data-in hold time after write		t_{DH}	45		55		ns
$tdis_{(CAS)}$	Output disable time		t_{OFF}	0	40	0	50	ns
$ta_{(CAS)}$	$\overline{CAS}$ access time	(Note 13)	t_{CAC}		75		100	ns
$ta_{(RAS)}$	$\overline{RAS}$ access time	(Note 14)	t_{RAC}		150		200	ns

Note 15: t_{CRW}min is defined as t_{CRW} min $= td_{(RAS-W)} + th_{(W-RAS)} + tw_{(RASH)} + 3t_{TLH(tTHL)}$

16: t_{CRMW} min is defined as t_{CRMW} min $= ta_{(RAS)}max + th_{(W-RAS)} + tw_{(RASH)} + 3t_{TLH(tTHL)}$

17: $tsu_{(W-CAS)}$, $td_{(RAS-W)}$, and $td_{(CAS-W)}$ do not define the limits of operation, but are included as electrical characteristics only.
When $tsu_{(W-CAS)} \geqq tsu_{(W-CAS)}$min, an early-write cycle is performed, and the data output keeps the high-impedance state.
When $td_{(RAS-W)} \geqq td_{(RAS-W)}$min, and $td_{(CAS-W)} \geqq tsu_{(W-CAS)}$min a read-write cycle is performed, and the data of the selected address will be read out on the data output.
For all conditions other than those described above, the condition of data output (at access time and until $\overline{CAS}$ goes back to V_{IH}) is not defined.

Page-Mode Cycle

Symbol	Parameter	Alternative Symbol	M5K4164S-15		M5K4164S-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{cPGR}	Page-mode read cycle time	t_{PC}	145		190		ns
t_{cPGW}	Page-mode write cycle time	t_{PC}	145		190		ns
t_{cPGRW}	Page-mode read-write cycle time	—	180		230		ns
t_{cPGRMW}	Page-mode read-modify-write cycle time	—	190		245		ns
$tw_{(CASH)}$	$\overline{CAS}$ high pulse width	t_{CP}	60		80		ns

Automatic Refresh Cycle

Symbol	Parameter	Alternative Symbol	M5K4164S-15		M5K4164S-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
$tc_{(REF)}$	Automatic Refresh Cycle Time	t_{FC}	260		330		ns
$tw_{(RASH)}$	$\overline{RAS}$ high pulse width	t_{RP}	395		480		ns
$td_{(RAS-REF)}$	Delay time, $\overline{RAS}$ to $\overline{REF}$	t_{RFD}	100		120		ns
$tw_{(REFL)}$	$\overline{REF}$ low pulse width	t_{FP}	80	8000	100	8000	ns
$tw_{(REFH)}$	$\overline{REF}$ high pulse width	t_{FI}	135		150		ns
$td_{(REF-RAS)}$	Delay time, $\overline{REF}$ to $\overline{RAS}$	t_{FSR}	30		30		ns
$tsu_{(REF-RAS)}$	$\overline{REF}$ pulse setup time before $\overline{RAS}$	t_{FRD}	295		360		ns

Self-Refresh Cycle

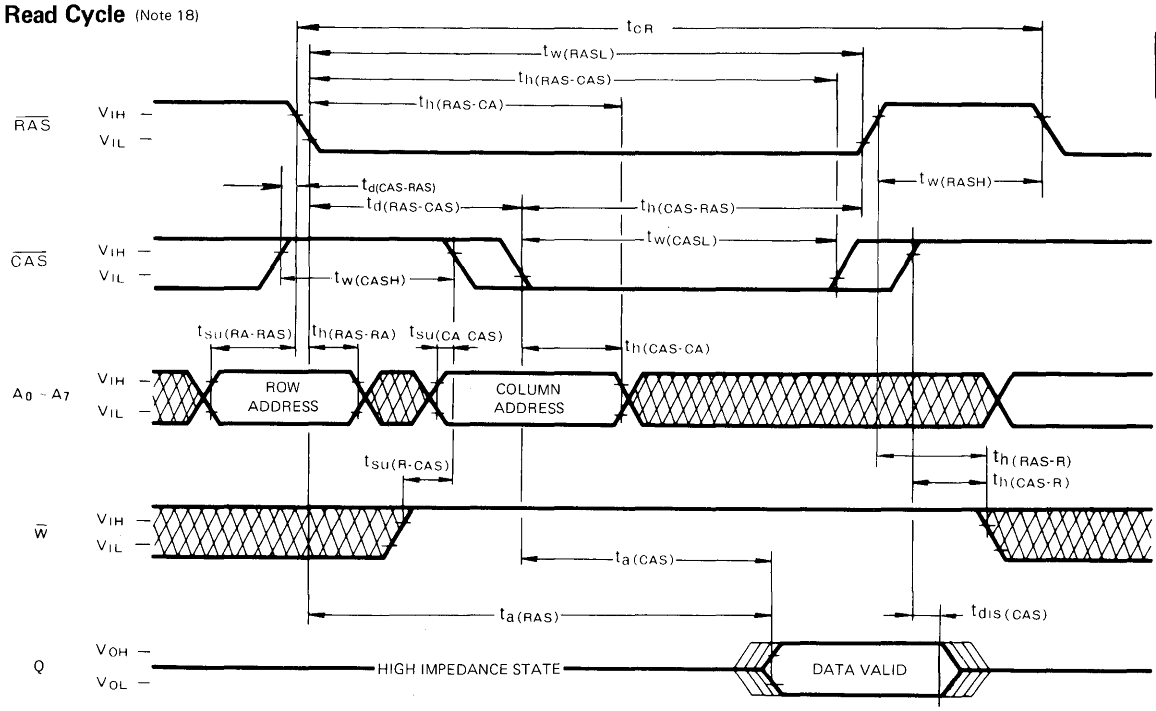
Symbol	Parameter	Alternative Symbol	M5K4164S-15		M5K4164S-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
$td_{(RAS-REF)}$	Delay time, $\overline{RAS}$ to $\overline{REF}$	t_{RFD}	100		120		ns
$tw_{(REFL)}$	$\overline{REF}$ low pulse width	t_{FBP}	8000	∞	8000	∞	ns
$td_{(REF-RAS)}$	Delay time, $\overline{REF}$ to $\overline{RAS}$	t_{FBR}	295		360		ns

MITSUBISHI ELECTRIC

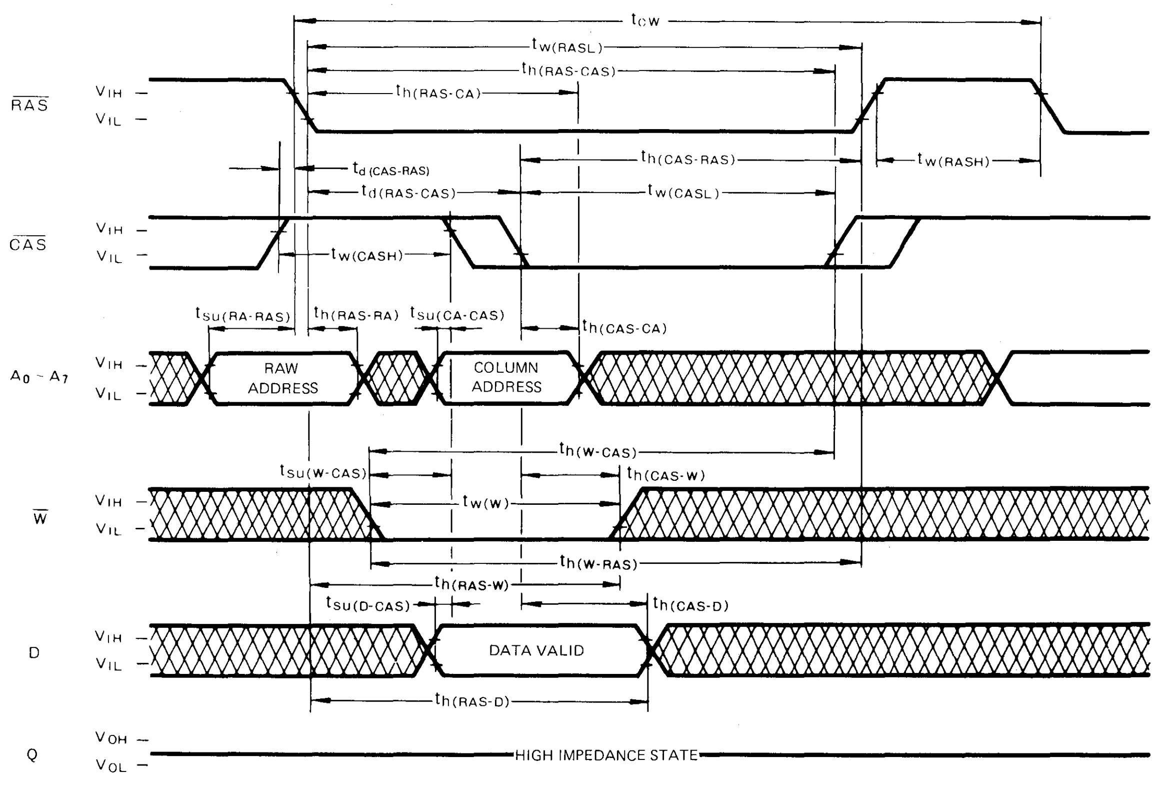
2

TIMING DIAGRAMS (Note 17)

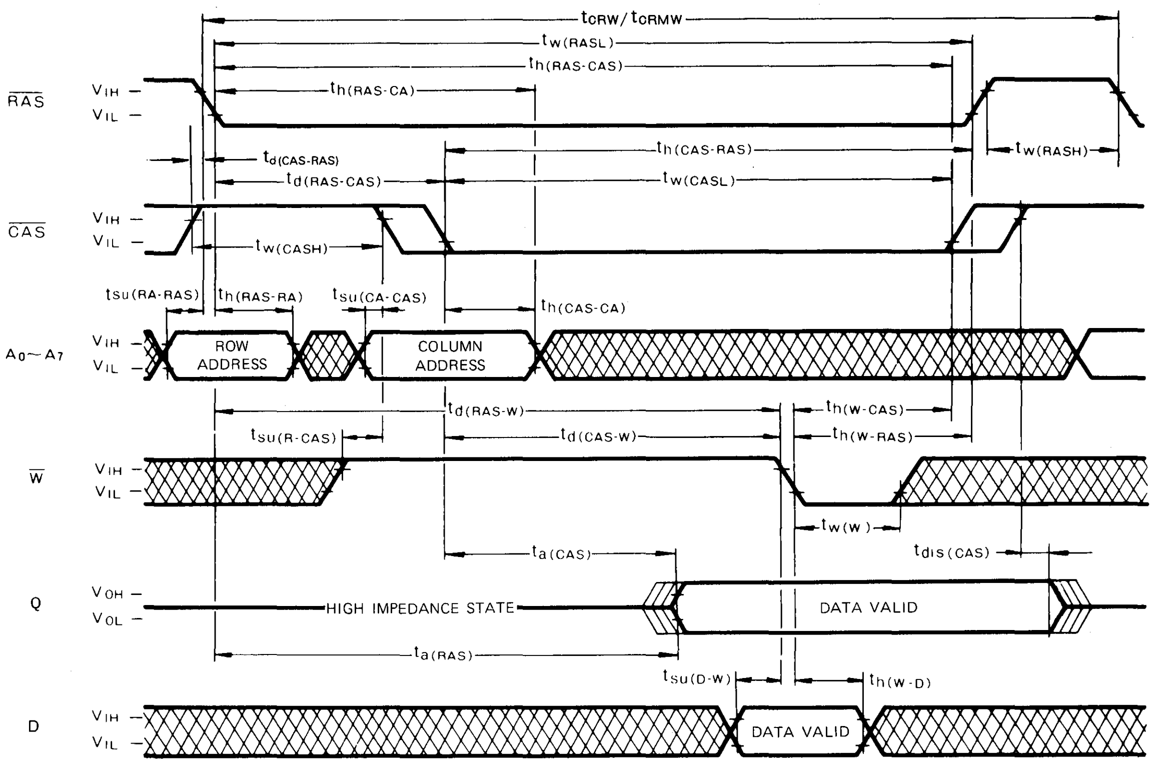
Read Cycle (Note 18)



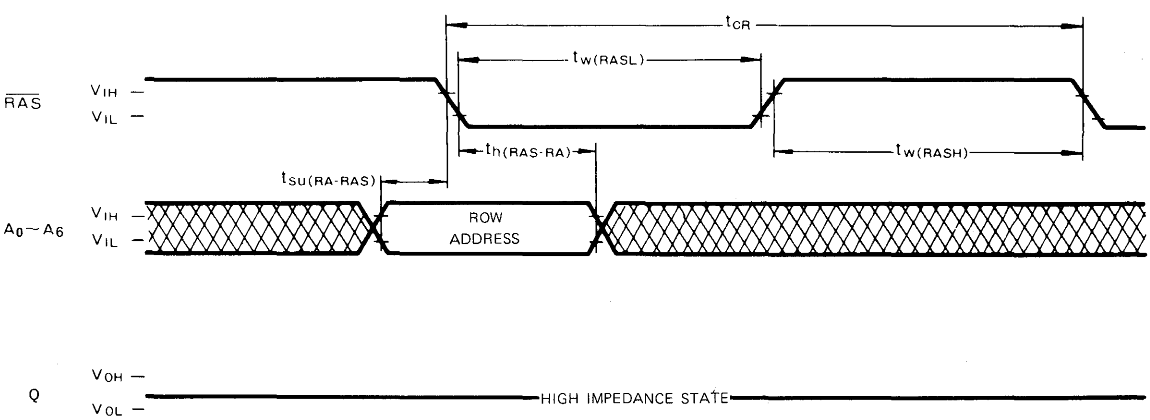
Write Cycle (Early Write) (Note 18)

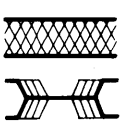


Read-Write and Read-Modify-Write Cycles (Note 18)



RAS-Only Refresh Cycle (Note 19.)



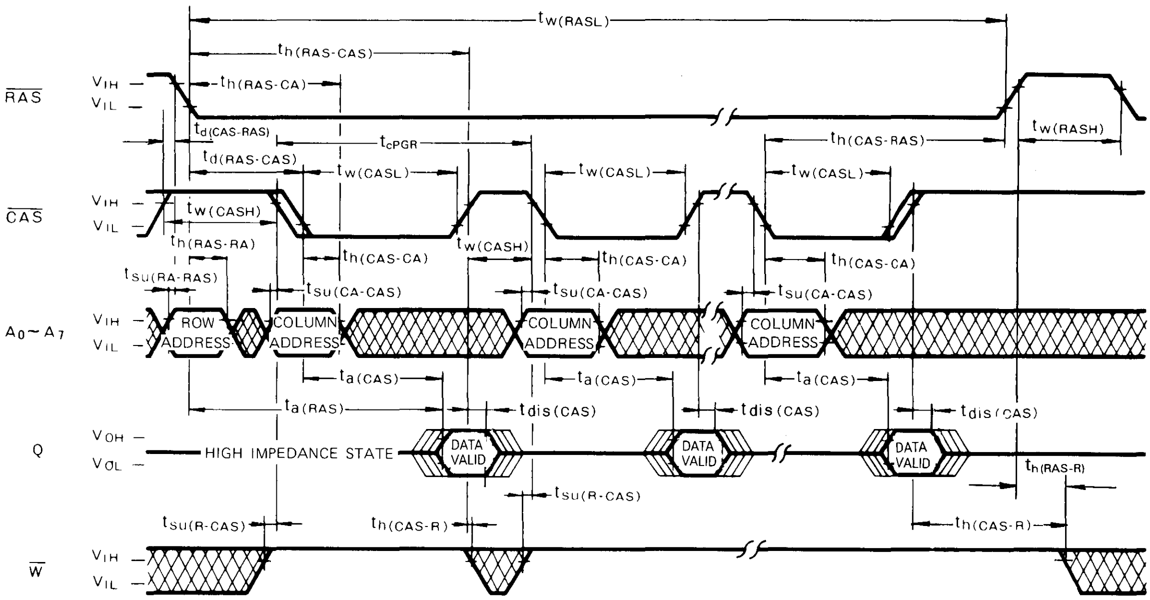
Note 17 
Indicates the don't care input

The center-line indicates the high-impedance state

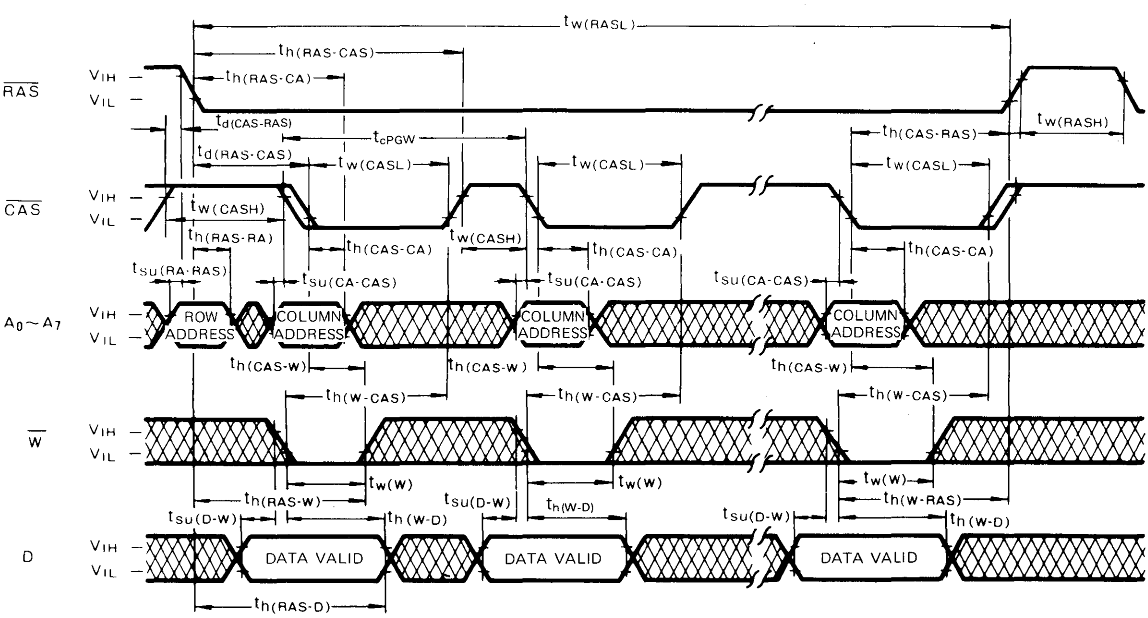
Note 18. $\overline{REF} = V_{IH}$

19. $\overline{CAS} = \overline{REF} = V_{IH}$, $\overline{W}$, A_7, D = don't care.

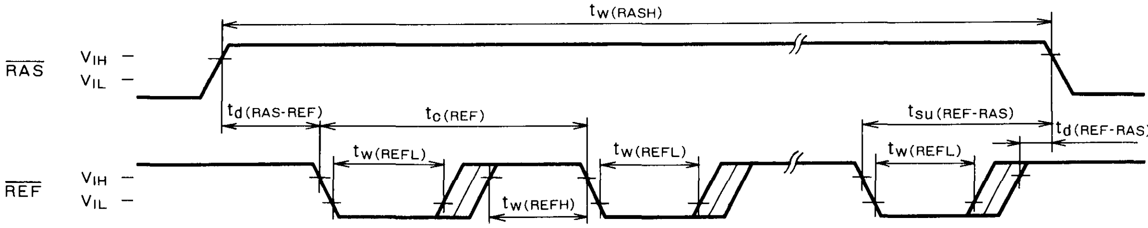
Page-Mode Read Cycle (Note 18)



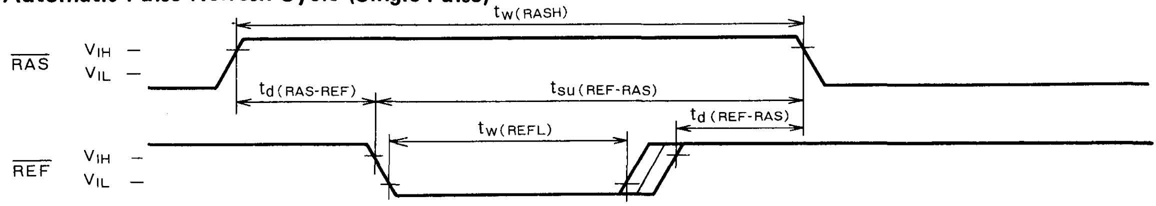
Page-Mode Write Cycle (Note 18)



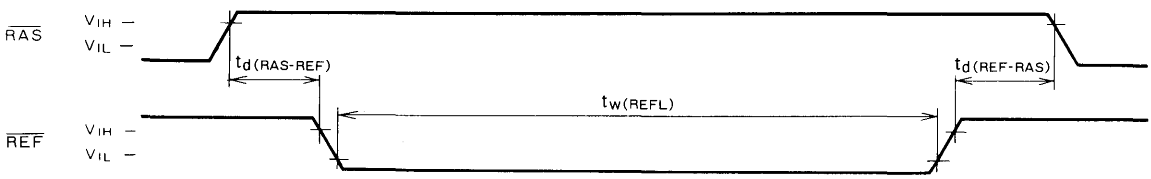
Automatic Pulse Refresh Cycle (Multiple Pulse) (Note 20)



Automatic Pulse Refresh Cycle (Single Pulse) (Note 20)

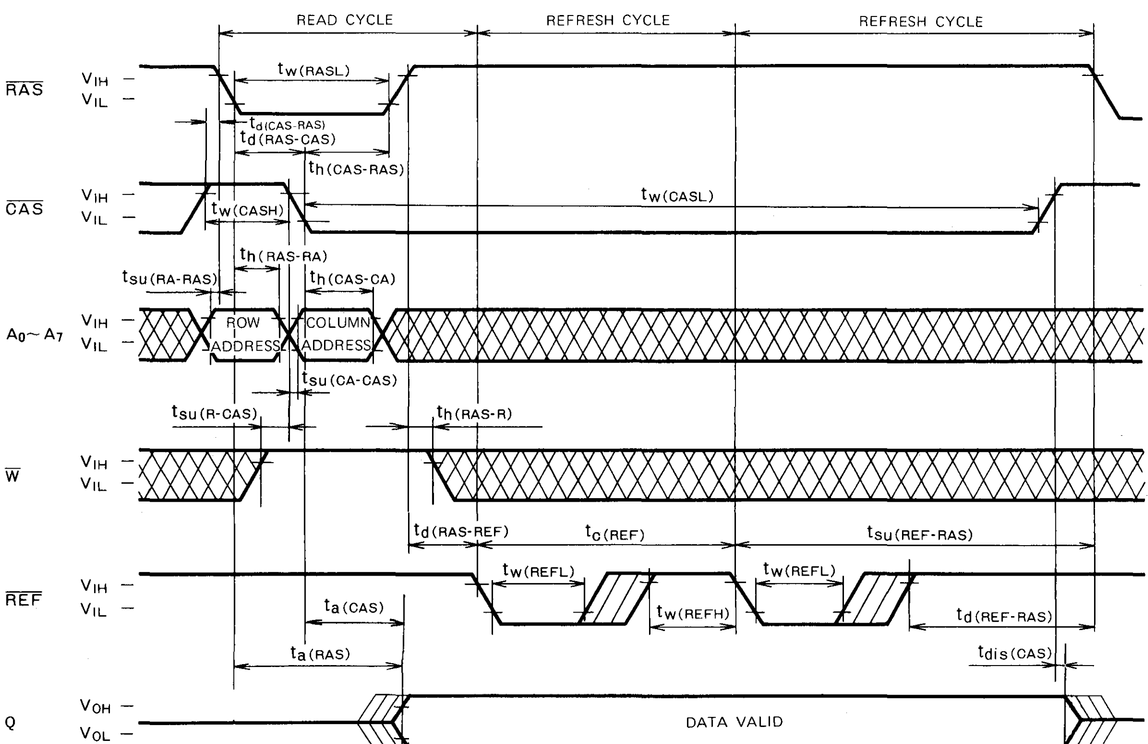


Self-Refresh Cycle (Note 20)

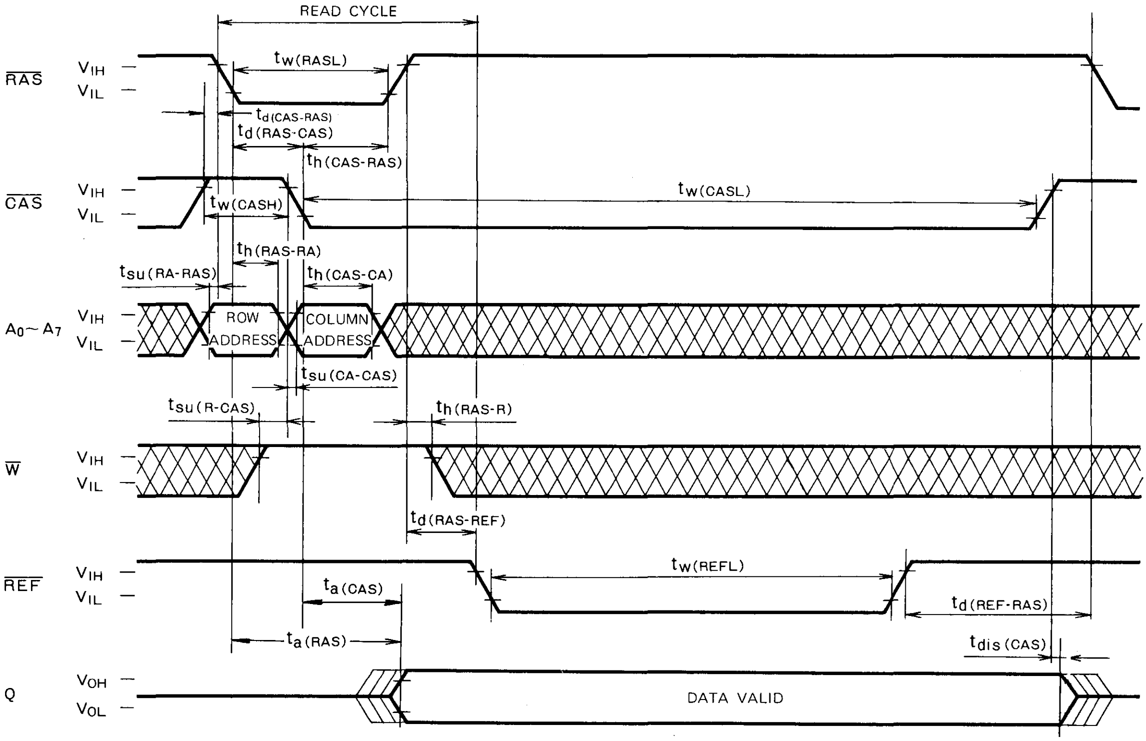


Note 20: $\overline{CAS}$, Addresses, D and $\overline{W}$ are don't care.

Hidden Automatic Pulse Refresh Cycle

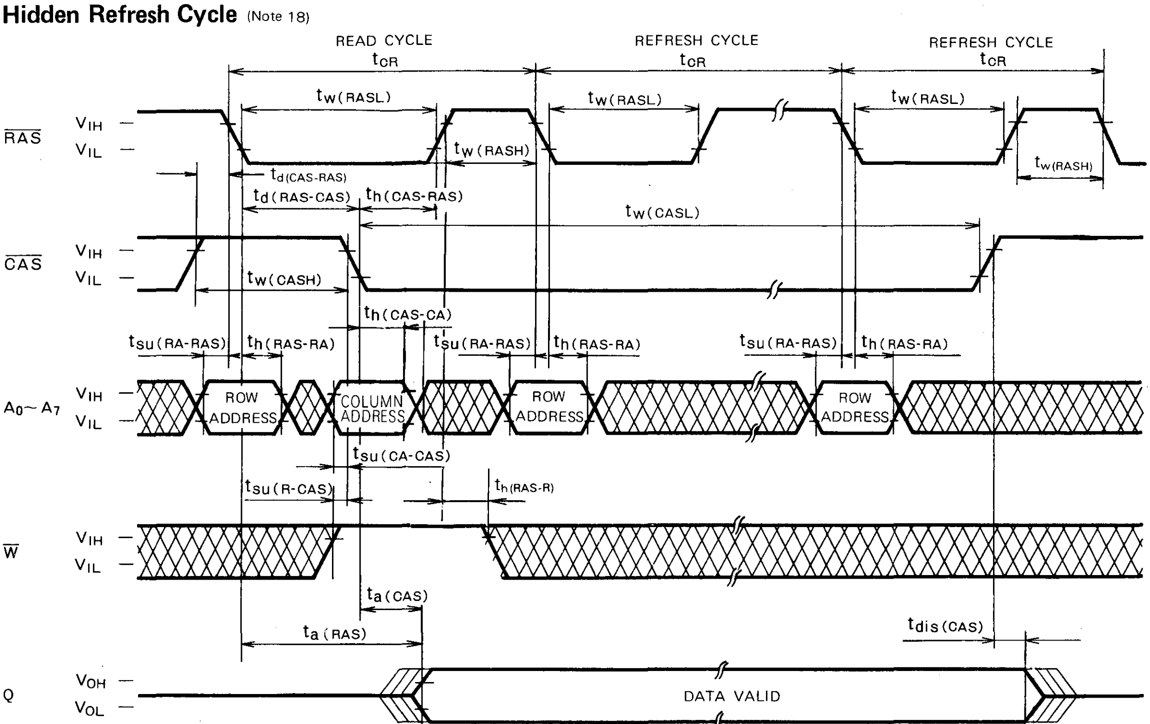


Hidden Self-Refresh Cycle



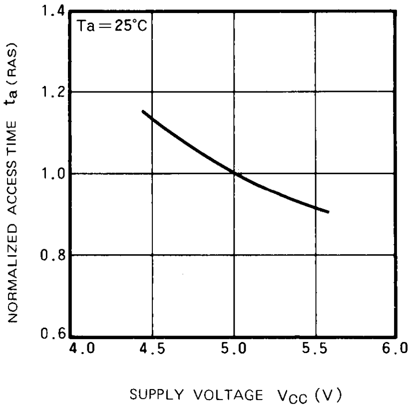
Note 21: If the pin 1 ($\overline{REF}$) function is not used, pin 1 may be left open (not connect).

Hidden Refresh Cycle (Note 18)

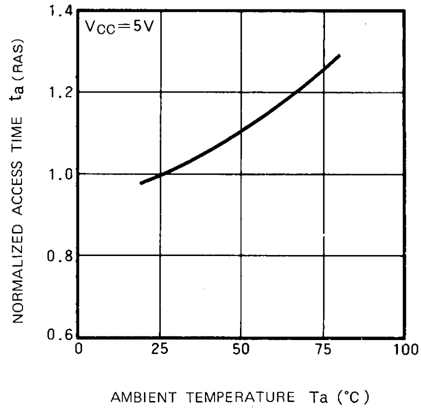


TYPICAL CHARACTERISTICS

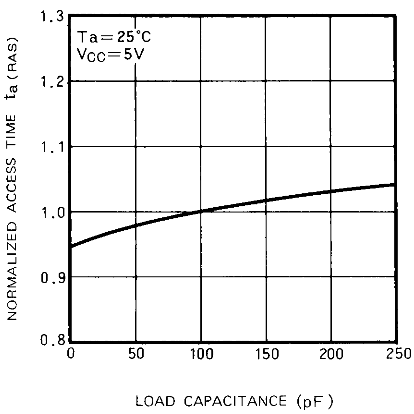
NORMALIZED ACCESS TIME VS. V_{CC} SUPPLY VOLTAGE



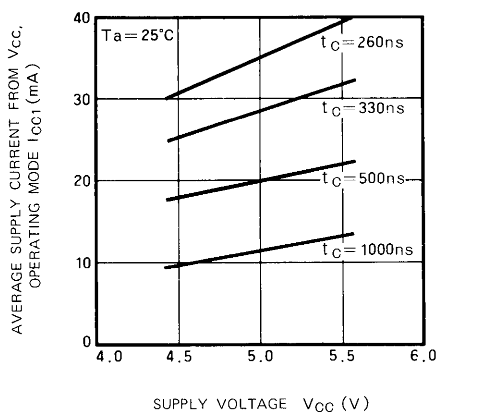
NORMALIZED ACCESS TIME VS. AMBIENT TEMPERATURE



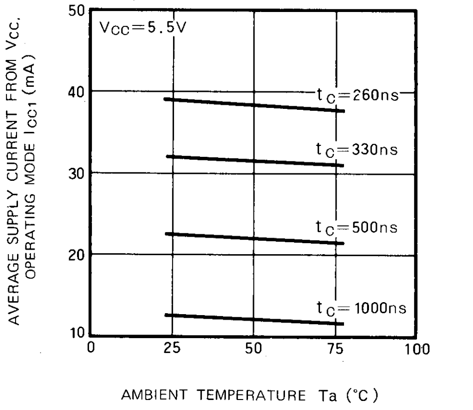
NORMALIZED ACCESS TIME VS. LOAD CAPACITANCE



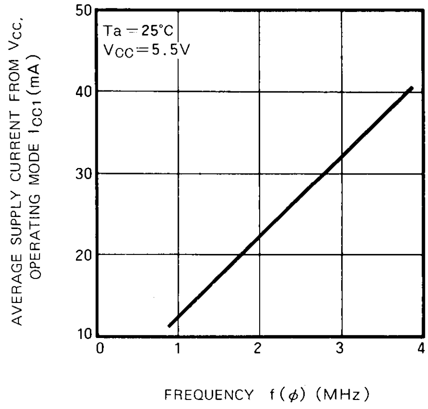
AVERAGE SUPPLY CURRENT FROM V_{CC}, OPERATING MODE VS. SUPPLY VOLTAGE



AVERAGE SUPPLY CURRENT FROM V_{CC}, OPERATING MODE VS. AMBIENT TEMPERATURE

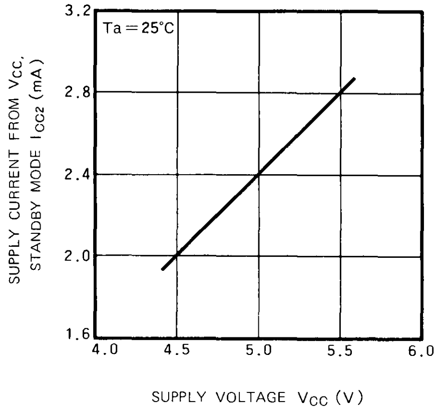


AVERAGE SUPPLY CURRENT FROM V_{CC}, OPERATING MODE VS. FREQUENCY

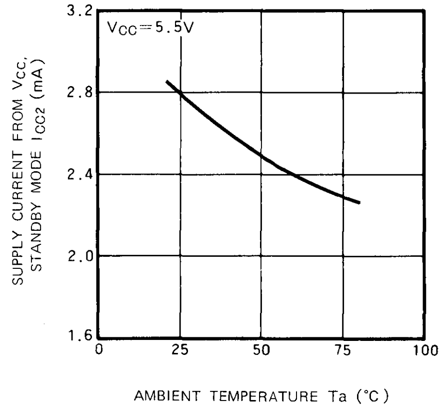


65 536-BIT (65 536-WORD BY 1-BIT) DYNAMIC RAM

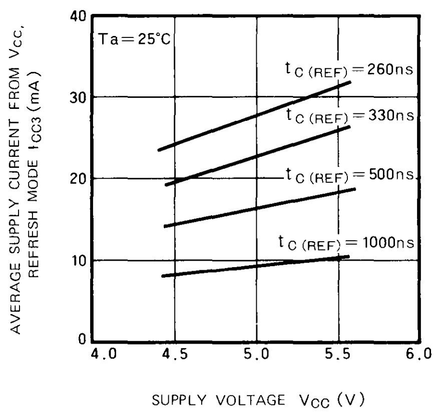
SUPPLY CURRENT FROM V_{CC}, STANDBY MODE VS. SUPPLY VOLTAGE



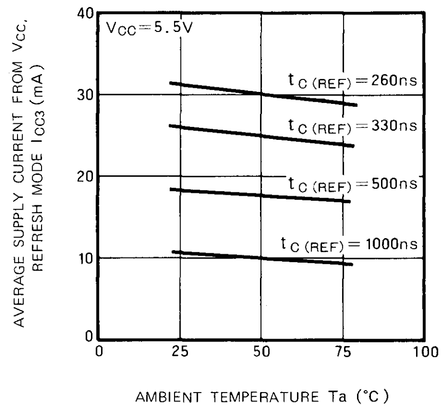
SUPPLY CURRENT FROM V_{CC}, STANDBY MODE VS. AMBIENT TEMPERATURE



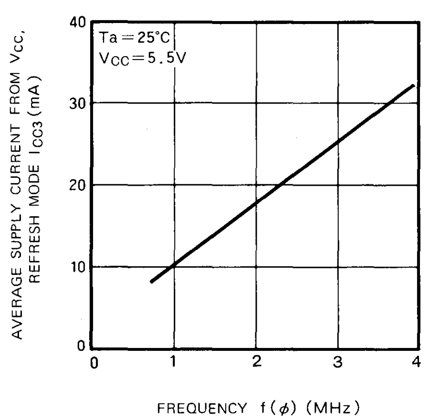
AVERAGE SUPPLY CURRENT FROM V_{CC}, REFRESH MODE VS. SUPPLY VOLTAGE



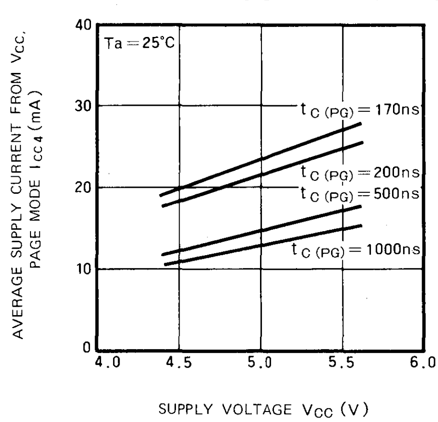
AVERAGE SUPPLY CURRENT FROM V_{CC}, REFRESH MODE VS. AMBIENT TEMPERATURE



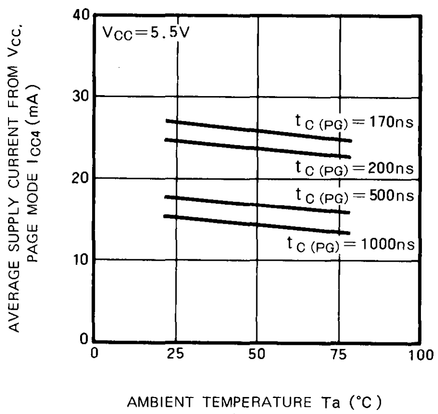
AVERAGE SUPPLY CURRENT FROM V_{CC}, REFRESH MODE VS. FREQUENCY



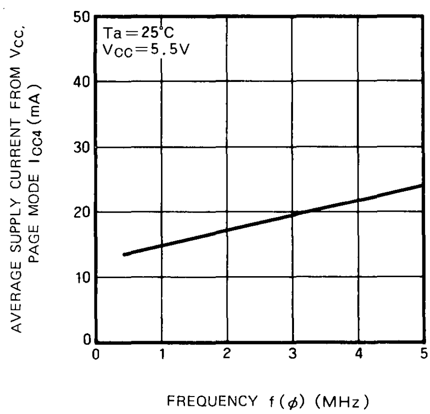
AVERAGE SUPPLY CURRENT FROM V_{CC}, PAGE MODE VS. SUPPLY VOLTAGE



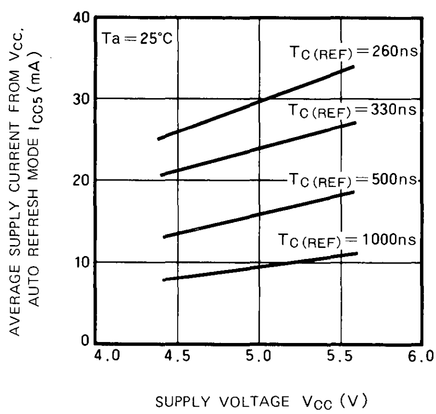
AVERAGE SUPPLY CURRENT FROM V_{CC}, PAGE MODE VS. AMBIENT TEMPERATURE



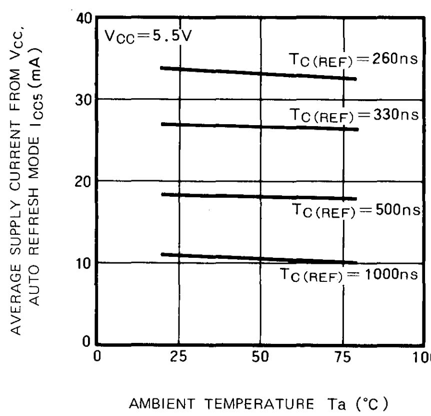
AVERAGE SUPPLY CURRENT FROM V_{CC}, PAGE MODE VS. FREQUENCY



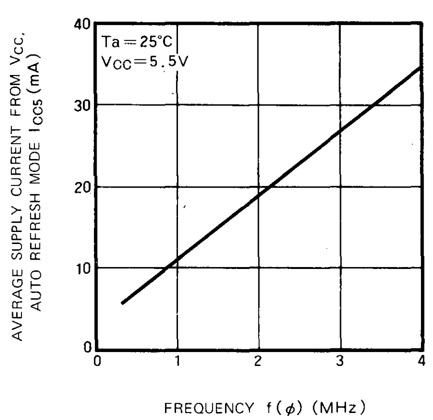
AVERAGE SUPPLY CURRENT FROM V_{CC}, AUTO REFRESH MODE VS. SUPPLY VOLTAGE



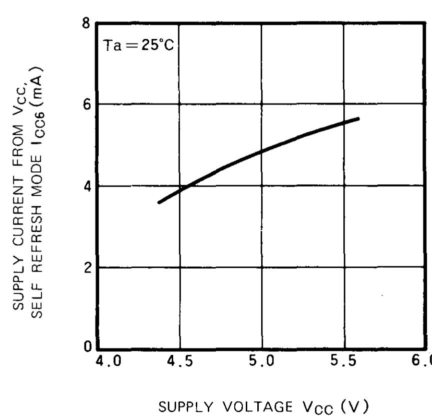
AVERAGE SUPPLY CURRENT FROM V_{CC}, AUTO REFRESH MODE VS. AMBIENT TEMPERATURE



AVERAGE SUPPLY CURRENT FROM V_{CC}, AUTO REFRESH MODE VS. FREQUENCY

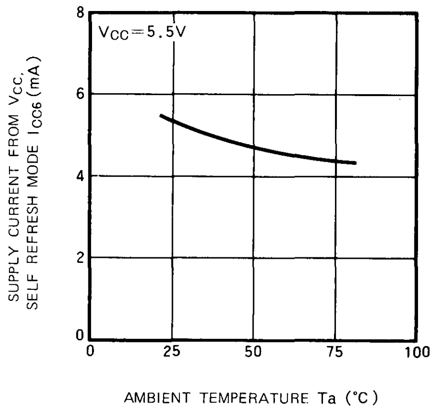


SUPPLY CURRENT FROM V_{CC}, SELF REFRESH MODE VS. SUPPLY VOLTAGE

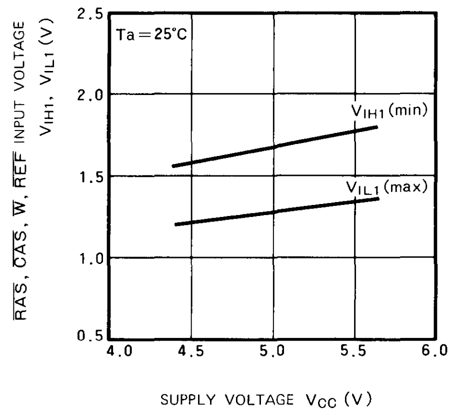


2

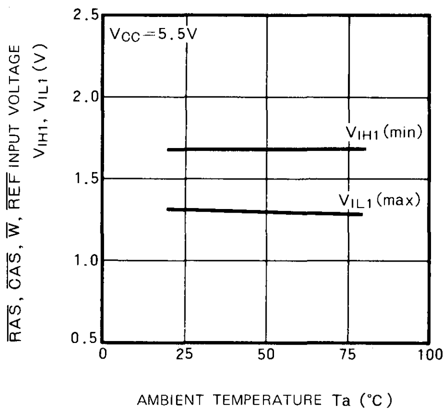
SUPPLY CURRENT FROM V_{CC}, SELF REFRESH MODE VS. AMBIENT TEMPERATURE



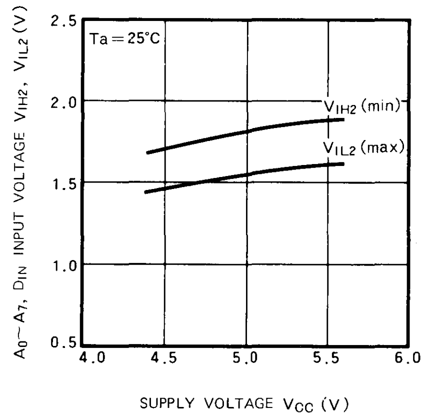
$\overline{RAS}$, $\overline{CAS}$, $\overline{W}$, $\overline{REF}$ INPUT VOLTAGE V_{IH1}, V_{IL1} VS. SUPPLY VOLTAGE



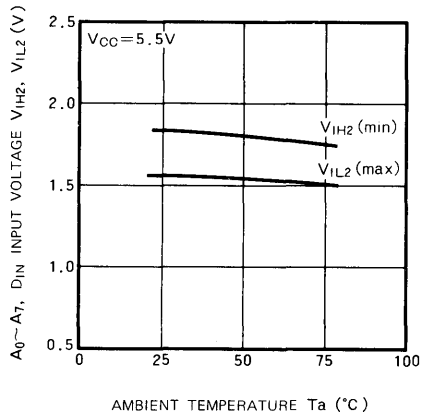
$\overline{RAS}$, $\overline{CAS}$, $\overline{W}$, $\overline{REF}$ INPUT VOLTAGE V_{IH1}, V_{IL1} VS. AMBIENT TEMPERATURE



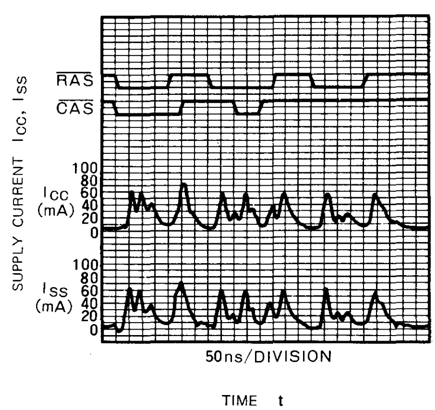
$A_0 \sim A_7$, D_{IN} INPUT VOLTAGE V_{IH2}, V_{IL2} VS. SUPPLY VOLTAGE



$A_0 \sim A_7$, D_{IN} INPUT VOLTAGE V_{IH2}, V_{IL2} VS. AMBIENT TEMPERATURE

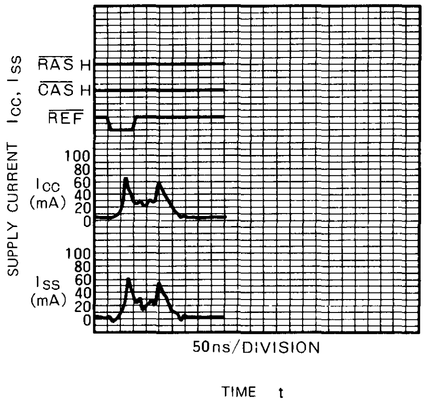


SUPPLY CURRENT VS. TIME



65 536-BIT (65 536-WORD BY 1-BIT) DYNAMIC RAM

SUPPLY CURRENT VS. TIME



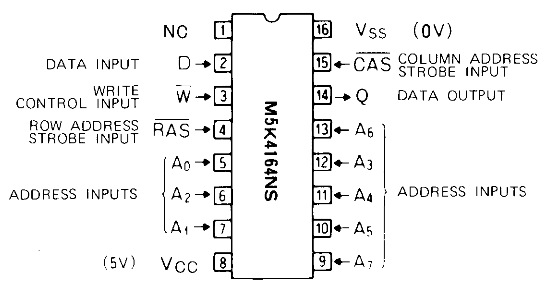
M5K4164NS-15, NS-20

65 536-BIT (65 536-WORD BY 1-BIT) DYNAMIC RAM

DESCRIPTION

This is a family of 65 536-word by 1-bit dynamic RAMs, fabricated with the high performance N-channel silicongate MOS process, and is ideal for large-capacity memory systems where high speed, low power dissipation, and low costs are essential. The use of double-layer polysilicon process technology and a single-transistor dynamic storage cell privide high circuit density at reduced costs, and the use of dynamic circuitry including sense amplifiers assures low power dissipation. Multiplexed address inputs permit both a reduction in pins to the standard 16-pin package configuration and an increase in system densities. The M5K4164NS operates on a 5V power supply using the on-chip substrate bias generator.

PIN CONFIGURATION (TOP VIEW)



Outline 16S1

FEATURES

- Performance ranges

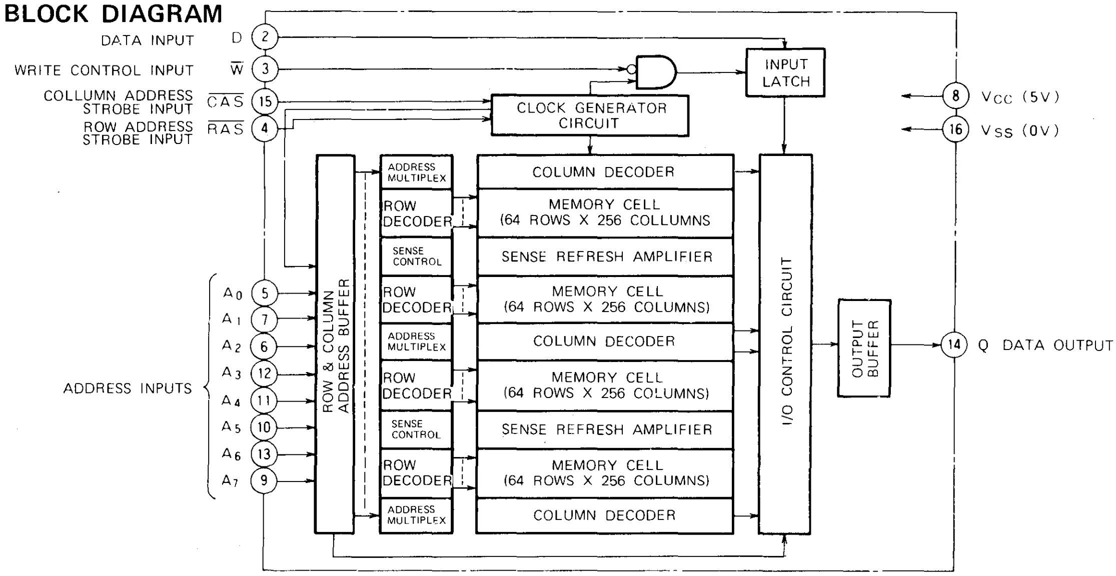
Type name	Access time (max) (ns)	Cycle time (min) (ns)	Power dissipation (typ) (mW)
M5K4164NS-15	150	260	200
M5K4164NS-20	200	330	170

- Standard 16-pin package
- Single 5V±10% supply
- Low standby power dissipation: 28.0mW (max)
- Low operating power dissipation:
  - M5K4164NS-15 275mW (max)
  - M5K4164NS-20 250mW (max)
- Unlatched output enables two-dimensional chip selection and extended page boundary
- Early-write operation gives common I/O capability
- Read-modify-write, $\overline{RAS}$-only refresh, and page-mode capabilities
- All input terminals have low input capaciatance and are directly TTL-compatible
- Output is three-state and directly TTL-compatible
- 128 refresh cycles every 2ms (16K dynamic RAMs M5K4116P, S compatible)
- $\overline{CAS}$ controlled output allows hidden refresh.
- Output data can be held infinitely by $\overline{CAS}$.
- Interchangeable with Mostek's MK4564 and Motorola's MCM 6665 in pin configuration.

APPLICATION

- Main memory unit for computers.

BLOCK DIAGRAM



FUNCTION

The M5K4164NS provides, in addition to normal read, write, and read-modify-write operations, a number of other functions, e.g., page mode, $\overline{RAS}$-only refresh, and delayed-write. The input conditions for each are shown in Table 1.

Table 1 Input conditions for each mode

Operation	Inputs						Output	Refresh	Remarks
	$\overline{RAS}$	$\overline{CAS}$	$\overline{W}$	D	Row address	Column address	Q		
Read	ACT	ACT	NAC	DNC	APD	APD	VLD	YES	Page mode identical except refresh is NO.
Write	ACT	ACT	ACT	VLD	APD	APD	OPN	YES	
Read-modify-write	ACT	ACT	ACT	VLD	APD	APD	VLD	YES	
RAS-only refresh	ACT	NAC	DNC	DNC	APD	DNC	OPN	YES	
Hidden refresh	ACT	ACT	DNC	DNC	APD	DNC	VLD	YES	
Standby	NAC	DNC	DNC	DNC	DNC	DNC	OPN	NO	

Note: ACT : active, NAC : nonactive, DNC : don't care, VLD : valid, APD : applied, OPN : open.

SUMMARY OF OPERATIONS

Addressing

To select one of the 65536 memory cells in the M5K4164NS the 16-bit address signal must be multiplexed into 8 address signals, which are then latched into the on-chip latch by two externally-applied clock pulses. First, the negative-going edge of the row-address-strobe pulse ($\overline{RAS}$) latches the 8 row-address bits; next, the negative-going edge of the column-address-strobe pulse ($\overline{CAS}$) latches the 8 column-address bits. Timing of the $\overline{RAS}$ and $\overline{CAS}$ clocks can be selected by either of the following two methods:

1. The delay time from $\overline{RAS}$ to $\overline{CAS}$ $t_{d\ (RAS\text{-}CAS)}$ is set between the minimum and maximum values of the limits. In this case, the internal $\overline{CAS}$ control signals are inhibited almost until $t_{d(RAS\text{-}CAS)\,max}$ ('gated $\overline{CAS}$' operation). The external $\overline{CAS}$ signal can be applied with a margin not affecting the on-chip circuit operations, e.g. access time, and the address inputs can be easily changed from row address to column address.
2. The delay time $t_{d(RAS\text{-}CAS)}$ is set larger than the maximum value of the limits. In this case the internal inhibition of $\overline{CAS}$ has already been released, so that the internal $\overline{CAS}$ control signals are controlled by the externally applied $\overline{CAS}$, which also controls the access time.

Data Input

Data to be written into a selected cell is strobed by the later of the two negative transistons of $\overline{W}$ input and $\overline{CAS}$ input. Thus when the $\overline{W}$ input makes its negative transition prior to $\overline{CAS}$ input (early write), the data input is strobed by $\overline{CAS}$, and the negative transition of $\overline{CAS}$ is set as the reference point for set-up and hold times. In the read-write or read-modify-write cycles, however, when the $\overline{W}$ input makes its negative transition after $\overline{CAS}$, the $\overline{W}$ negative transition is set as the reference point for setup and hold times.

Data Output Control

The output of the M5K4164NS is in the high-impedance state when $\overline{CAS}$ is high. When the memory cycle in progress is a read, read-modify-write, or a delayed-write cycle, the data output will go from the high-impedance state to the active condition, and the data in the selected cell will be read. This data output will have the same polarity as the input data. Once the output has entered the active condition, this condition will be maintained until $\overline{CAS}$ goes high, irrespective of the condition of $\overline{RAS}$.

The output will remain in the high-impedance state throughout the entire cycle in an early-write cycle.

These output conditions, of the M5K4164NS, which can readily be changed by controlling the timing of the write pulse in a write cycle, and the width of the $\overline{CAS}$ pulse in a read cycle, offer capabilities for a number of applications, as follows.

1. Common I/O Operation

If all write operations are performed in the early-write mode, input and output can be connected directly to give a common I/O data bus.

2. Data Output Hold

The data output can be held between read cycles, without lengthening the cycle time. This enables extremely flexible clock-timing settings for $\overline{RAS}$ and $\overline{CAS}$.

3. Two Methods of Chip Selection

Since the output is not latched, $\overline{CAS}$ is not required to keep the outputs of selected chips in the matrix in a high-impedance state. This means that $\overline{CAS}$ and/or $\overline{RAS}$ can both be decoded for chip selection.

4. Extended-Page Boundary

By decoding $\overline{CAS}$, the page boundary can be extended beyond the 256 column locations in a single chip. In this case, $\overline{RAS}$ must be applied to all devices.

Page-Mode Operation

This operation allows for multiple-column addressing at the same row address, and eliminates the power dissipation associated with the negative-going edge of $\overline{RAS}$, because once the row address has been strobed, $\overline{RAS}$ is maintained. Also, the time required to strobe in the row address for the second and subsequent cycles is eliminated, thereby decreasing the access and cycle times.

Refresh

Each of the 128 rows ($A_0 \sim A_6$) of the M5K4164NS must be refreshed every 2 ms to maintain data. The methods of refreshing for the M5K4164NS are as follows.

1. Normal Refresh

Read cycle and Write cycle (early write, delayed write or read-modify-write) refresh the selected row as defined by the low order ($\overline{RAS}$) addresses. Any write cycle, of course, may change the state of the selected cell. Using a read, write, or read-modify-write cycle for refresh is not recommended for systems which utilize "write-OR" outputs since output bus contention will occur.

2. $\overline{RAS}$ Only Refresh

A $\overline{RAS}$-only refresh cycle is the recommended technique for most applications to provide for data retention. A $\overline{RAS}$-only refresh cycle maintains the output in the high-impedance state with a typical power reduction of 20% over a read or write cycle.

3. Hidden Refresh

A features of the M5K4164NS is that refresh cycles may be performed while maintaining valid data at the output pin by extending the $\overline{CAS}$ active time from a previous memory read cycle. This feature is refered to as hidden refresh.

Hidden refresh is performed by holding $\overline{CAS}$ at V_{IL} and taking $\overline{RAS}$ high and after a specified precharge period, executing a $\overline{RAS}$-only cycling, but with $\overline{CAS}$ held low.

The advantage of this refresh mode is that data can be held valid at the output data port indefinitely by leaving the $\overline{CAS}$ asserted. In many applications this eliminates the need for off-chip latches.

Power Dissipation

Most of the circuitry in the M5K4164NS is dynamic, and most of the power is dissipated when addresses are strobed. Both $\overline{RAS}$ and $\overline{CAS}$ are decoded and applied to the M5K4164NS as chip-select in the memory system, but if $\overline{RAS}$ is decoded, all unselected devices go into stand-by independent of the $\overline{CAS}$ condition, minimizing system power dissipation.

Power Supplies

The M5K4164NS operates on a single 5V power supply.

A wait of some 500μs and eight or more dummy cycles is necessary after power is applied to the device before memory operation is achieved.

ABSOLUTE MAXIMUM RATINGS

Symbol	Paramater	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−1∼7	V
V_I	Input voltage		−1∼7	V
V_O	Output voltage		−1∼7	V
I_O	Output current		50	mA
Pd	Power dissipation	Ta=25°C	1000	mW
Topr	Operating free-air temperature range		0∼70	°C
Tstg	Storage temperature range		−65∼150	°C

RECOMMENDED OPERATING CONDITIONS (Ta=0∼70°C, unless otherwise noted) (Note 1)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage	0	0	0	V
V_{IH}	High-level input voltage, all inputs	2.4		6.5	V
V_{IL}	Low-level input voltage, all inputs	−2		0.8	V

Note 1: All voltage values are with respect to V_{SS}

ELECTRICAL CHARACTERISTICS (Ta=0∼70°C, V_{CC}=5V±10%, V_{SS}=0V, unless otherwise noted) (Note 2)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
V_{OH}	High-level output voltage		I_{OH}=−5mA	2.4		V_{CC}	V
V_{OL}	Low-level output voltage		I_{OL}=4.2mA	0		0.4	V
I_{OZ}	Off-state output current		Q floating 0V≦V_{OUT}≦5.5V	−10		10	μA
I_I	Input current		0V≦V_{IN}≦6.5V, All other pins=0V	−10		10	μA
$I_{CC1(AV)}$	Average supply current from V_{CC}, operating (Note 3, 4)	M5K4164NS-15	$\overline{RAS}$, $\overline{CAS}$ cycling			50	mA
		M5K4164NS-20	t_{CR}=t_{CW}= min output open			45	mA
I_{CC2}	Supply current from V_{CC}, standby		$\overline{RAS}$=V_{IH} output open			5	mA
$I_{CC3(AV)}$	Average supply current from V_{CC}, refreshing (Note 3)	M5K4164NS-15	$\overline{RAS}$ cycling $\overline{CAS}$=V_{IH}			40	mA
		M5K4164NS-20	$t_{C(\overline{REF})}$= min, output open			35	mA
$I_{CC4(AV)}$	Average supply current from V_{CC}, page mode (Note 3, 4)	M5K4164NS-15	$\overline{RAS}$=V_{IL}, $\overline{CAS}$ cycling			40	mA
		M5K4164NS-20	t_{CPG}= min, output open			35	mA
$C_{I(A)}$	Input capacitance, address inputs		V_I=V_{SS} f=1MHz V_I=25mVrms			5	pF
$C_{I(D)}$	Input capacitance, data input					5	pF
$C_{I(W)}$	Input capacitance, write control input					7	pF
$C_{I(RAS)}$	Input capacitance, $\overline{RAS}$ input					10	pF
$C_{I(CAS)}$	Input capacitance, $\overline{CAS}$ input					10	pF
C_O	Output capacitance		V_O=V_{SS}, f=1MHz, V_I=25mVrms			7	pF

Note 2: Current flowing into an IC is positive ; out is negative.
3: $I_{CC1(AV)}$, $I_{CC3(AV)}$, and $I_{CC4(AV)}$ are dependent on cycle rate. Maximum current is measured at the fastest cycle rate.
4: $I_{CC1(AV)}$ and $I_{CC4(AV)}$ are dependent on output loading. Specified values are obtained with the output open.

2

TIMING REQUIREMENTS (For Read, Write, Read-Modify-Write, Refresh, and Page-Mode Cycle)

($T_a = 0 \sim 70°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted, See notes 5, 6 and 7)

Symbol	Parameter	Alternative Symbol	M5K4164NS-15		M5K4164NS-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{CRF}	Refresh cycle time	t_{REF}		2		2	ms
$t_{w(RASH)}$	$\overline{RAS}$ high pulse width	t_{RP}	100		120		ns
$t_{w(RASL)}$	$\overline{RAS}$ low pulse width	t_{RAS}	150	10000	200	10000	ns
$t_{w(CASL)}$	$\overline{CAS}$ low pulse width	t_{CAS}	75	∞	100	∞	ns
$t_{w(CASH)}$	$\overline{CAS}$ high pulse width (Note 8)	t_{CPN}	35		40		ns
$t_{h(RAS-CAS)}$	$\overline{CAS}$ hold time after $\overline{RAS}$	t_{CSH}	150		200		ns
$t_{h(CAS-RAS)}$	$\overline{RAS}$ hold time after $\overline{CAS}$	t_{RSH}	75		100		ns
$t_{d(CAS-RAS)}$	Delay time, $\overline{CAS}$ to $\overline{RAS}$ (Note 9)	t_{CRP}	−20		−20		ns
$t_{d(RAS-CAS)}$	Delay time, $\overline{RAS}$ to $\overline{CAS}$ (Note 10)	t_{RCD}	25	75	30	100	ns
$t_{su(RA-RAS)}$	Row address setup time before $\overline{RAS}$	t_{ASR}	0		0		ns
$t_{su(CA-CAS)}$	Column address setup time before $\overline{CAS}$	t_{ASC}	−5		−5		ns
$t_{h(RAS-RA)}$	Row address hold time after $\overline{RAS}$	t_{RAH}	20		25		ns
$t_{h(CAS-CA)}$	Column address hold time after $\overline{CAS}$	t_{CAH}	25		35		ns
$t_{h(RAS-CA)}$	Column address hold time after $\overline{RAS}$	t_{AR}	95		120		ns
t_{THL} t_{TLH}	Transition time	t_T	3	35	3	50	ns

Note 5: An initial pause of 500μs is required after power-up followed by any eight $\overline{RAS}$ or $\overline{RAS}/\overline{CAS}$ cycles before proper device operation is achieved.
6: The switching characteristics are defined as $t_{THL} = t_{TLH} = 5ns$.
7: Reference levels of input signals are $V_{IH\ min.}$ and $V_{IL\ max.}$ Reference levels for transition time are also between V_{IH} and V_{IL}.
8: Except for page-mode.
9: $t_{d(CAS-RAS)}$ requirement is only applicable for $\overline{RAS}/\overline{CAS}$ cycles preceded by a $\overline{CAS}$ only cycle (i. e. for systems where $\overline{CAS}$ has not been decoded with $\overline{RAS}$).
10: Operation within the $t_{d(RAS-CAS)}$ max limit insures that $t_{a(RAS)}$ max can be met. $t_{d(RAS-CAS)}$max is specified reference point only;if $t_{d(RAS-CAS)}$ is greater than the specified $t_{d(RAS-CAS)}$ max limit, then access time is controlled exclusively by $t_{a(CAS)}$.
$t_{d(RAS-CAS)min} = t_{h(RAS-RA)min} + 2t_{THL(t_{TLH})} + t_{su(CA-CAS)min}$.

SWITCHING CHARACTERISTICS ($T_a = 0 \sim 70°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted)

Read Cycle

Symbol	Parameter	Alternative Symbol	M5K4164NS-15		M5K4164NS-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{CR}	Read cycle time	t_{RC}	260		330		ns
$t_{su(R-CAS)}$	Read setup time before $\overline{CAS}$	t_{RCS}	0		0		ns
$t_{h(CAS-R)}$	Read hold time after $\overline{CAS}$ (Note 11)	t_{RCH}	0		0		ns
$t_{h(RAS-R)}$	Read hold time after $\overline{RAS}$ (Note 11)	t_{RRH}	20		25		ns
$t_{dis(CAS)}$	Output disable time (Note 12)	t_{OFF}	0	40	0	50	ns
$t_{a(CAS)}$	$\overline{CAS}$ access time (Note 13)	t_{CAC}		75		100	ns
$t_{a(RAS)}$	$\overline{RAS}$ access time (Note 14)	t_{RAC}		150		200	ns

Note 11: Either $t_{h(RAS-R)}$ or $t_{h(CAS-R)}$ must be satisfied for a read cycle.
Note 12: $t_{dis(CAS)}$max defines the time at which the output achieves the open circuit condition and is not reference to V_{OH} or V_{OL}.
Note 13: This is the value when $t_{d(RAS-CAS)} \geqq t_{d(RAS-CAS)}$max. Test conditions ; Load = 2TTL, C_L = 100pF
Note 14: This is the value when $t_{d(RAS-CAS)} < t_{d(RAS-CAS)}$max. When $t_{d(RAS-CAS)} \geqq t_{d(RAS-CAS)}$max, $t_{a(RAS)}$ will increase by the amount that $t_{d(RAS-CAS)}$ exceeds the value shown. Test conditions ; Load = 2TTL C_L = 100pF

Write Cycle

Symbol	Parameter	Alternative Symbol	M5K4164NS-15		M5K4164NS-20		Unit
			Limits		Limits		
			Min	Max	Min	Max	
t_{CW}	Write cycle time	t_{RC}	260		330		ns
$t_{su(W-CAS)}$	Write setup time before $\overline{CAS}$ (Note 17)	t_{WCS}	−10		−10		ns
$t_{h(CAS-W)}$	Write hold time after $\overline{CAS}$	t_{WCH}	45		55		ns
$t_{h(RAS-W)}$	Write hold time after $\overline{RAS}$	t_{WCR}	95		120		ns
$t_{h(W-RAS)}$	$\overline{RAS}$ hold time after write	t_{RWL}	45		55		ns
$t_{h(W-CAS)}$	$\overline{CAS}$ hold time after write	t_{CWL}	45		55		ns
$t_{w(W)}$	Write pulse width	t_{WP}	45		55		ns
$t_{su(D-CAS)}$	Data-in setup time before $\overline{CAS}$	t_{DS}	0		0		ns
$t_{h(CAS-D)}$	Data-in hold time after $\overline{CAS}$	t_{DH}	45		55		ns
$t_{h(RAS-D)}$	Data-in hold time after $\overline{RAS}$	t_{DHR}	95		120		ns

Read-Write and Read-Modify-Write Cycles

Symbol	Parameter	Alternative Symbol	M5K4164NS-15 Limits Min	M5K4164NS-15 Limits Max	M5K4164NS-20 Limits Min	M5K4164NS-20 Limits Max	Unit
t_{CRW}	Read-write cycle time (Note 15)	t_{RWC}	280		340		ns
t_{CRMW}	Read-modify-write cycle time (Note 16)	t_{RMWC}	310		390		ns
$th_{(W-RAS)}$	$\overline{RAS}$ hold time after write	t_{RWL}	45		55		ns
$th_{(W-CAS)}$	$\overline{CAS}$ hold time after write	t_{CWL}	45		55		ns
$tw_{(W)}$	Write pulse width	t_{WP}	45		55		ns
$tsu_{(R-CAS)}$	Read setup time before $\overline{CAS}$	t_{RCS}	0		0		ns
$td_{(RAS-W)}$	Delay time, $\overline{RAS}$ to write (Note 17)	t_{RWD}	120		150		ns
$td_{(CAS-W)}$	Delay time, $\overline{CAS}$ to write (Note 17)	t_{CWD}	60		80		ns
$tsu_{(D-W)}$	Data-in set-up time before write	t_{DS}	0		0		ns
$th_{(W-D)}$	Data-in hold time after write	t_{DH}	45		55		ns
$tdis_{(CAS)}$	Output disable time	t_{OFF}	0	40	0	50	ns
$ta_{(CAS)}$	$\overline{CAS}$ access time (Note 13)	t_{CAC}		75		100	ns
$ta_{(RAS)}$	$\overline{RAS}$ access time (Note 14)	t_{RAC}		150		200	ns

Note 15: $t_{CRW}min$ is defined as $t_{CRW}\,min = t_{d(RAS-W)} + th_{(W-RAS)} + tw_{(RASH)} + 3t_{TLH(tTHL)}$

16: t_{CRMW} min is defined as $t_{CRMW}\,min = ta_{(RAS)}max + th_{(W-RAS)} + tw_{(RAS\,H)} + 3t_{TLH(tTHL)}$

17: $tsu_{(W-CAS)}$, $td_{(RAS-W)}$, and $td_{(CAS-W)}$ do not define the limits of operation, but are included as electrical characteristics only.
When $tsu_{(W-CAS)} \geqq tsu_{(W-CAS)}min$, an early-write cycle is performed, and the data output keeps the high-impedance state.
When $td_{(RAS-W)} \geqq td_{(RAS-W)}min$, and $td_{(CAS-W)} \geqq tsu_{(W-CAS)}min$ a read-write cycle is performed, and the data of the selected address will be read out on the data output.
For all conditions other than those described above, the condition of data output (at access time and until $\overline{CAS}$ goes back to V_{IH}) is not defined.

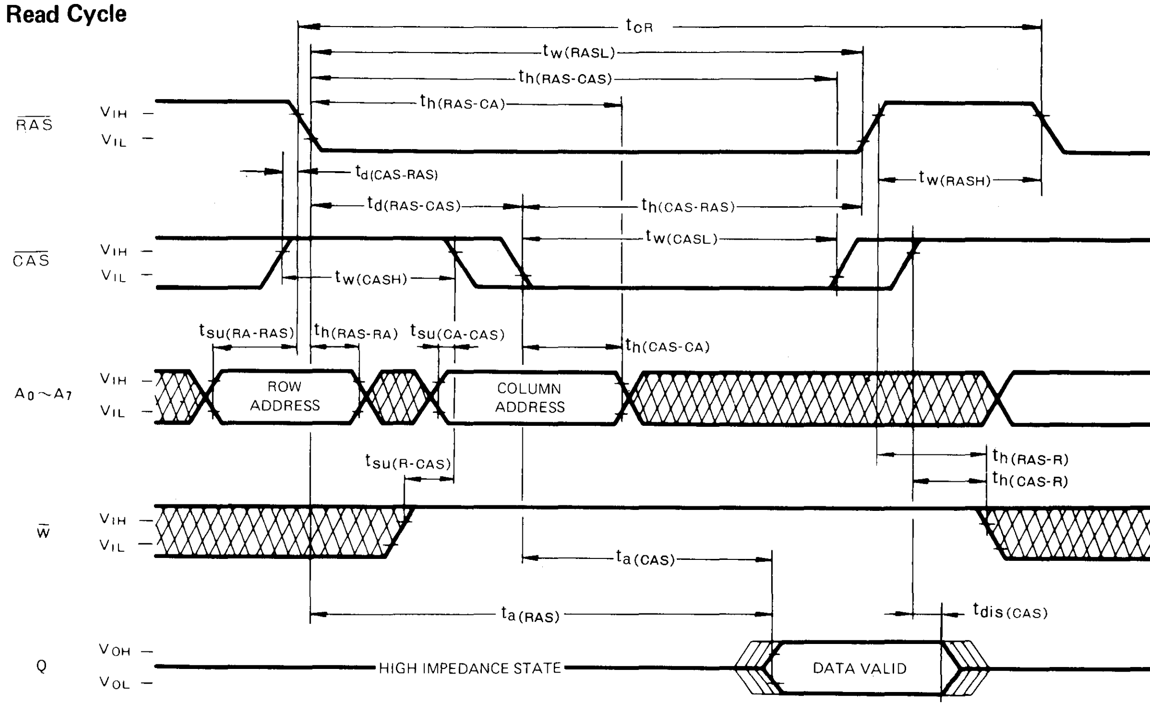
Page-Mode Cycle

Symbol	Parameter	Alternative Symbol	M5K4164NS-15 Limits Min	M5K4164NS-15 Limits Max	M5K4164NS-20 Limits Min	M5K4164NS-20 Limits Max	Unit
t_{cPGR}	Page-mode read cycle time	t_{PC}	145		190		ns
t_{cPGW}	Page-mode write cycle time	t_{PC}	145		190		ns
t_{cPGRW}	Page-mode read-write cycle time	—	180		230		ns
t_{cPGRMW}	Page-mode read-modify-write cycle time	—	190		245		ns
$tw_{(CASH)}$	$\overline{CAS}$ high pulse width	t_{CP}	60		80		ns

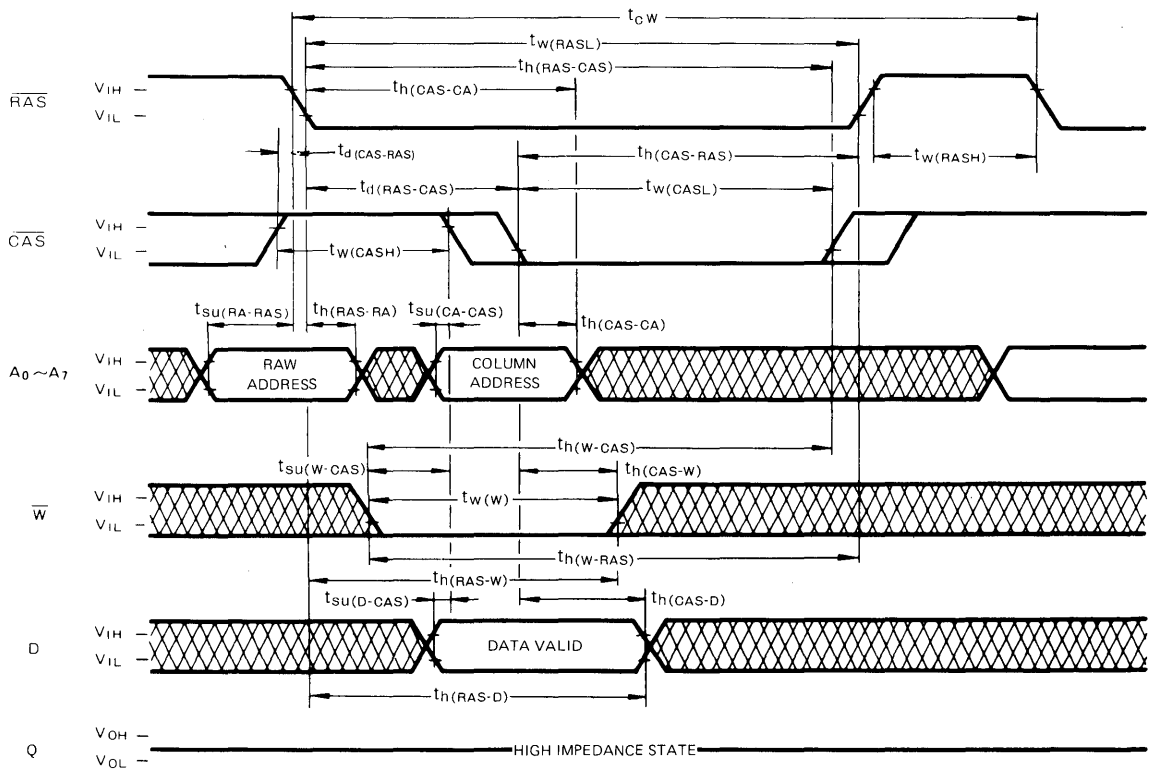
2

TIMING DIAGRAMS (Note 17)

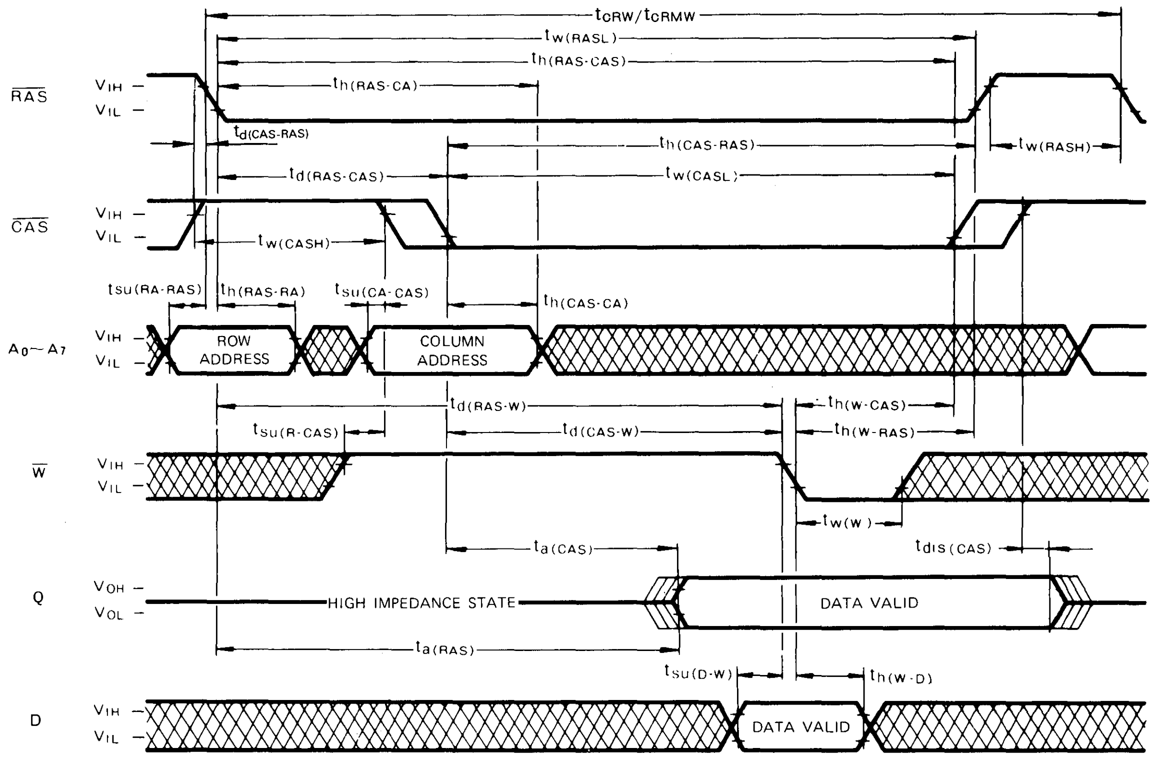
Read Cycle



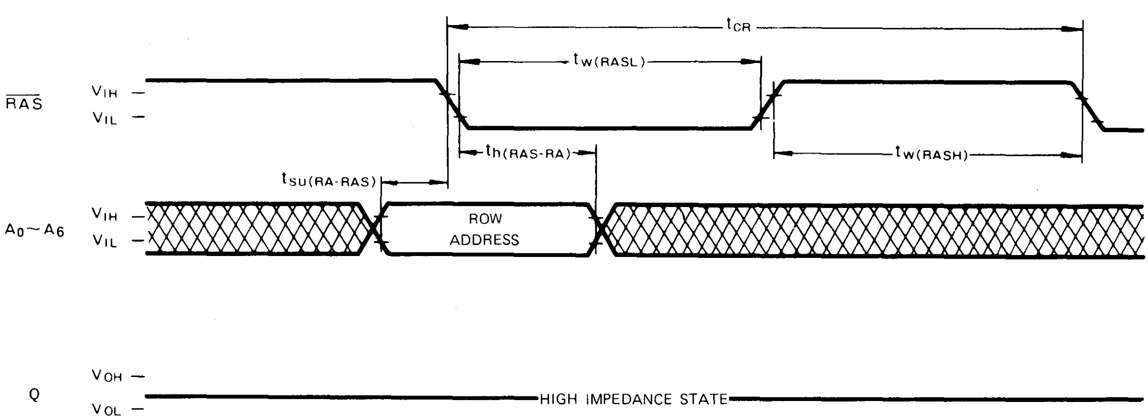
Write Cycle (Early Write)



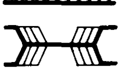
Read-Write and Read-Modify-Write Cycles



RAS-Only Refresh Cycle (Note 18)

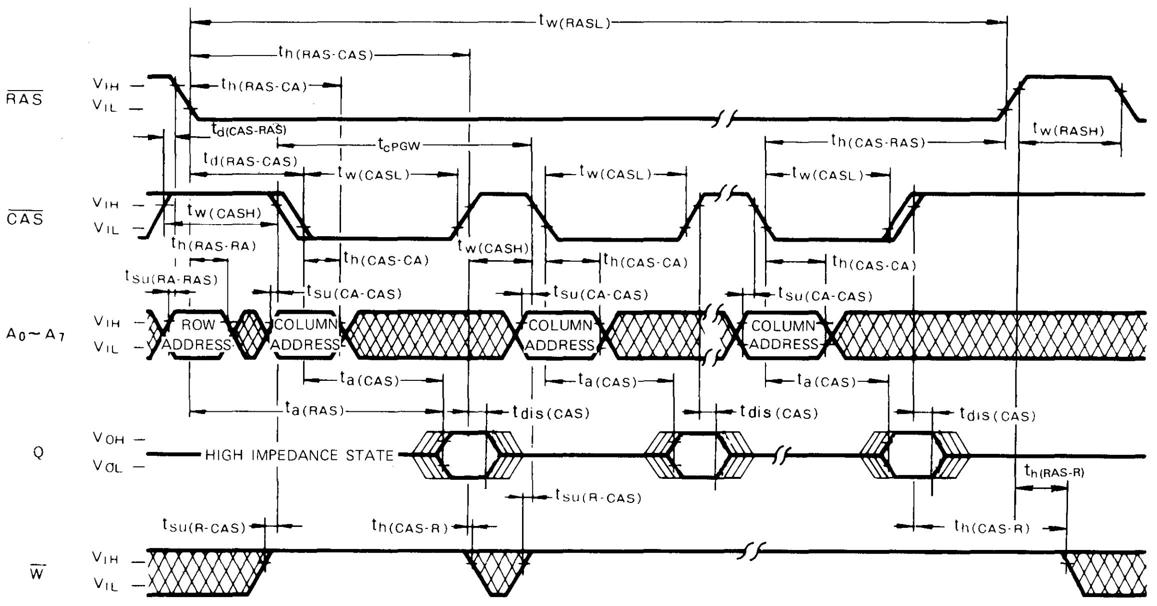


Note 17.  Indicates the don't care input

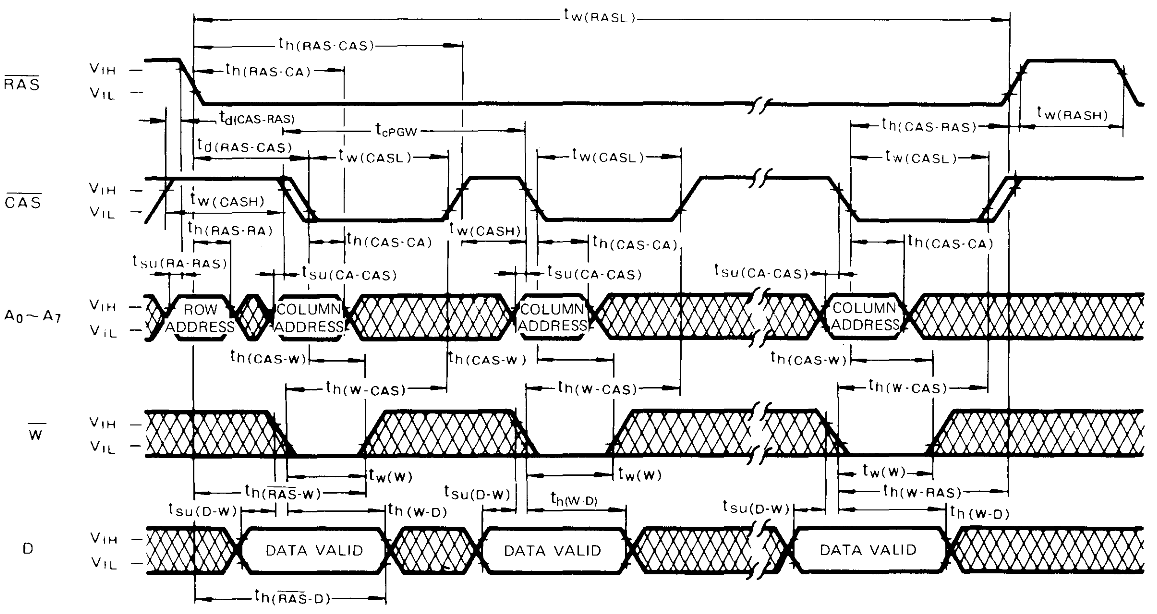
 The center-line indicates the high-impedance state

Note 18. $\overline{CAS} = V_{IH}$, $\overline{W}$, A_7, D = don't care.

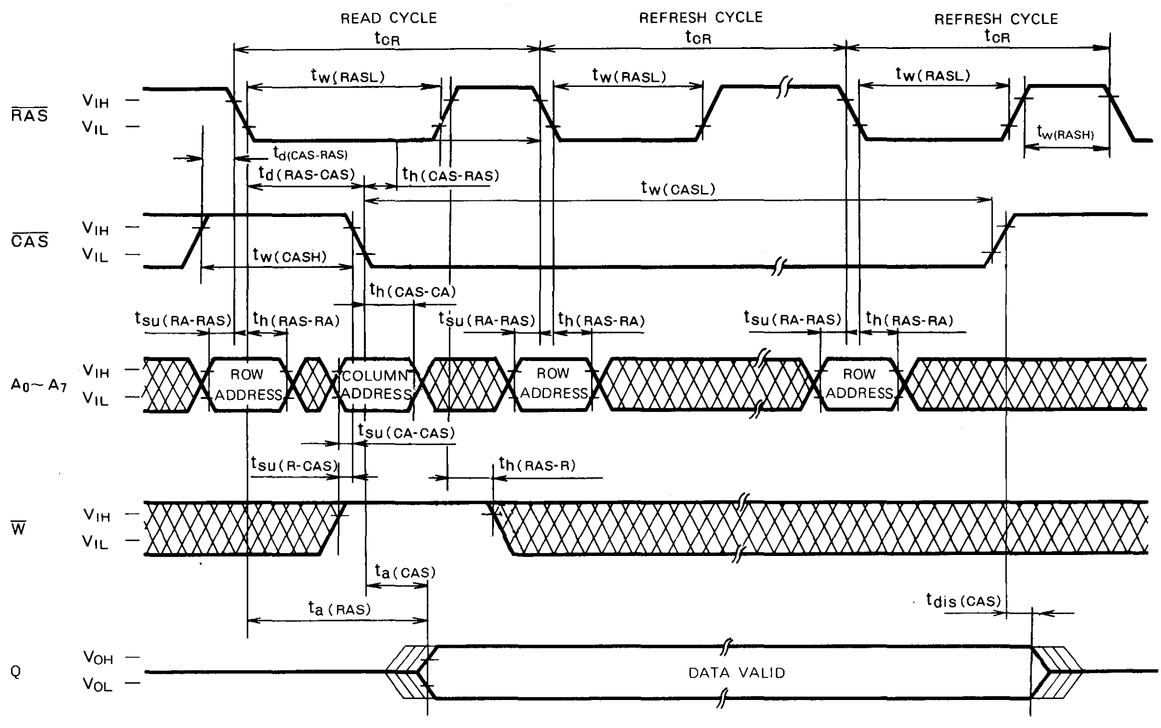
Page-Mode Read Cycle



Page-Mode Write Cycle



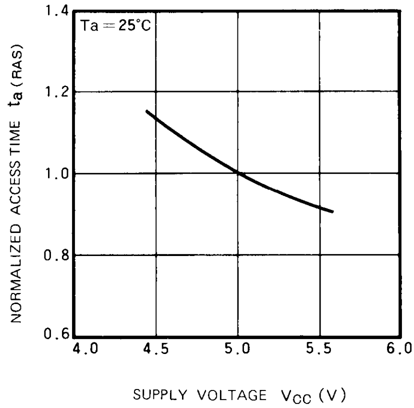
Hidden Refresh Cycle



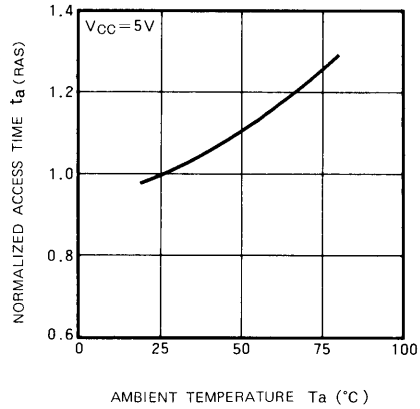
2

TYPICAL CHARACTERISTICS

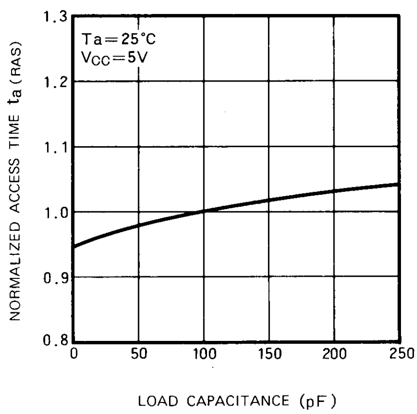
NORMALIZED ACCESS TIME VS. V_{CC} SUPPLY VOLTAGE



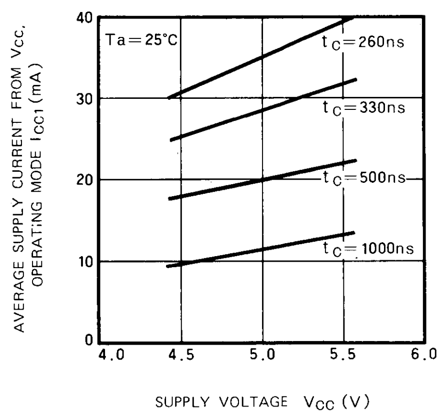
NORMALIZED ACCESS TIME VS. AMBIENT TEMPERATURE



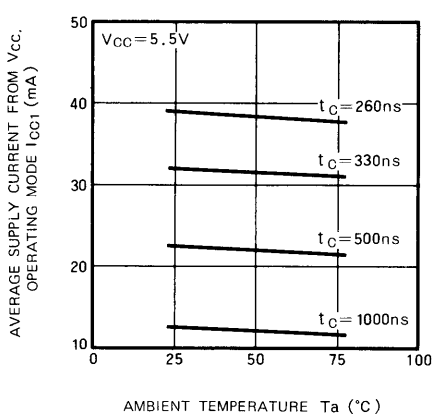
NORMALIZED ACCESS TIME VS. LOAD CAPACITANCE



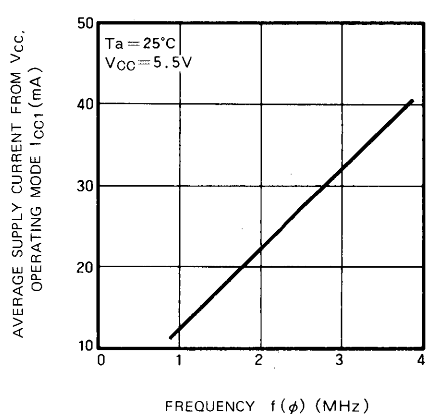
AVERAGE SUPPLY CURRENT FROM V_{CC}, OPERATING MODE VS. SUPPLY VOLTAGE



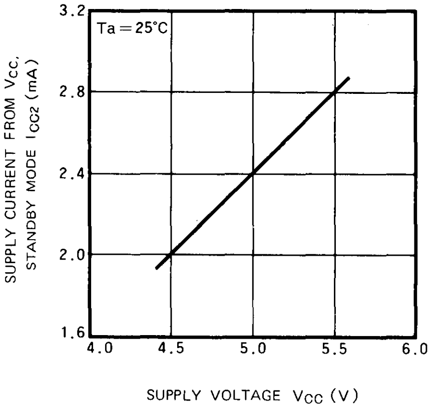
AVERAGE SUPPLY CURRENT FROM V_{CC}, OPERATING MODE VS. AMBIENT TEMPERATURE



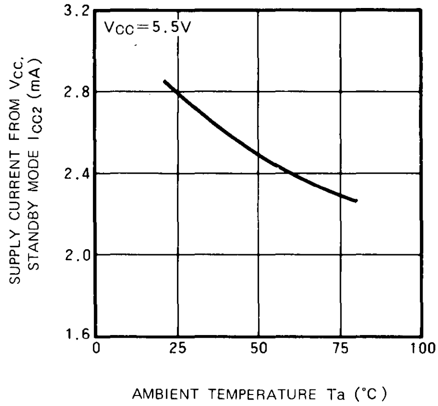
AVERAGE SUPPLY CURRENT FROM V_{CC}, OPERATING MODE VS. FREQUENCY



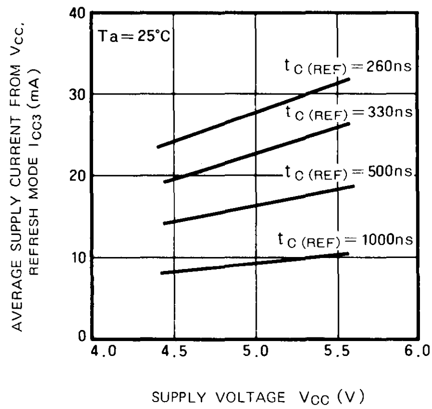
SUPPLY CURRENT FROM V_{CC}, STANDBY MODE VS. SUPPLY VOLTAGE



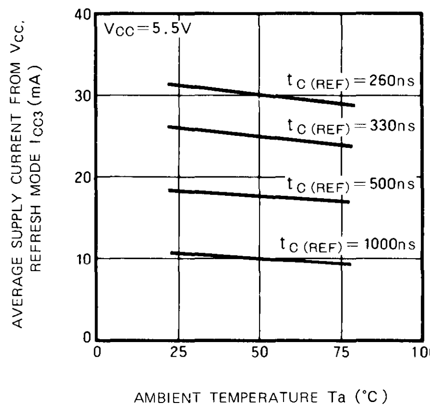
SUPPLY CURRENT FROM V_{CC}, STANDBY MODE VS. AMBIENT TEMPERATURE



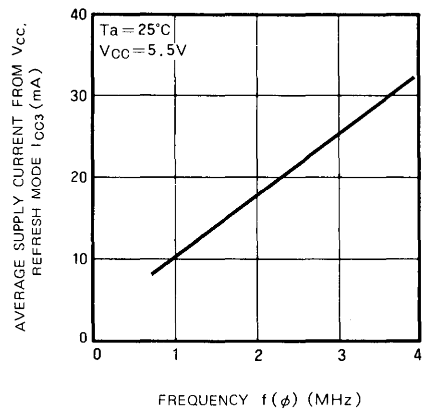
AVERAGE SUPPLY CURRENT FROM V_{CC}, REFRESH MODE VS. SUPPLY VOLTAGE



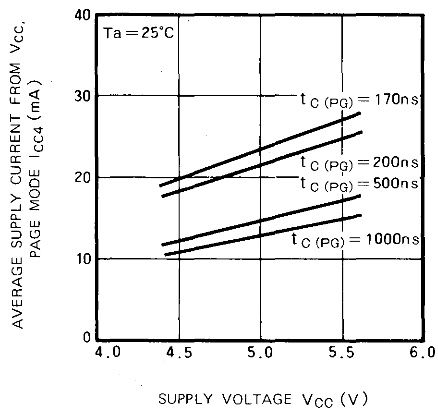
AVERAGE SUPPLY CURRENT FROM V_{CC}, REFRESH MODE VS. AMBIENT TEMPERATURE



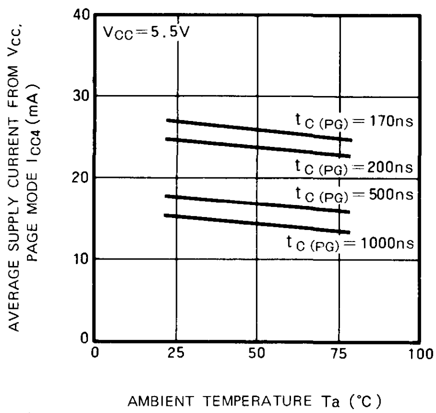
AVERAGE SUPPLY CURRENT FROM V_{CC}, REFRESH MODE VS. FREQUENCY



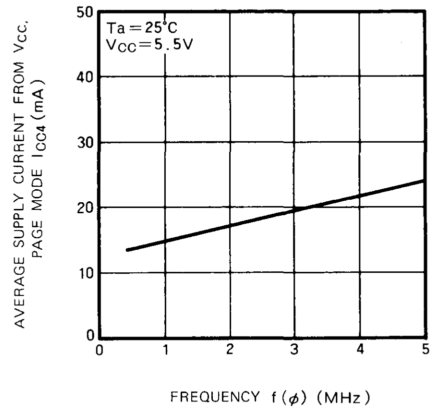
AVERAGE SUPPLY CURRENT FROM V_{CC}, PAGE MODE VS. SUPPLY VOLTAGE



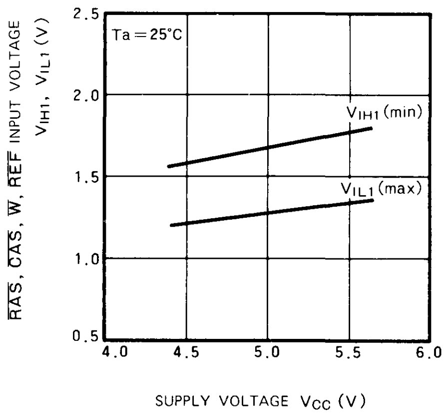
AVERAGE SUPPLY CURRENT FROM V_{CC}, PAGE MODE VS. AMBIENT TEMPERATURE



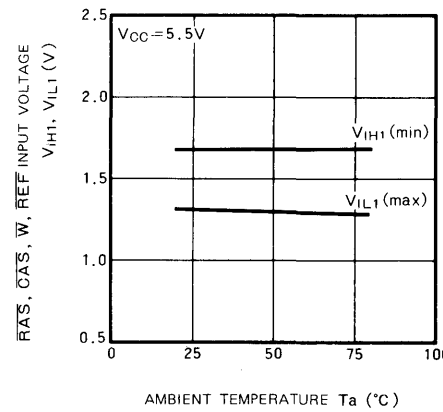
AVERAGE SUPPLY CURRENT FROM V_{CC}, PAGE MODE VS. FREQUENCY



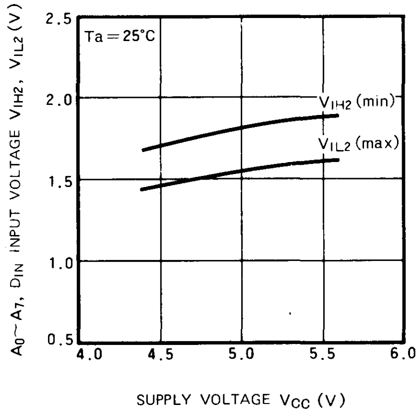
$\overline{RAS}$, $\overline{CAS}$, $\overline{W}$, $\overline{REF}$ INPUT VOLTAGE V_{IH1}, V_{IL1} VS. SUPPLY VOLTAGE



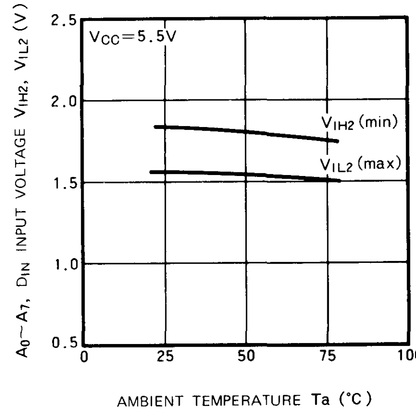
$\overline{RAS}$, $\overline{CAS}$, $\overline{W}$, $\overline{REF}$ INPUT VOLTAGE V_{IH1}, V_{IL1} VS. AMBIENT TEMPERATURE



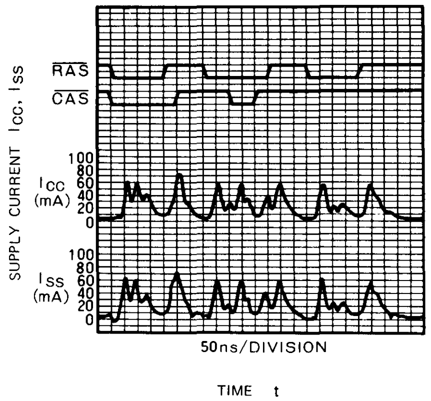
$A_0 \sim A_7$, D_{IN} INPUT VOLTAGE V_{IH2}, V_{IL2} VS. SUPPLY VOLTAGE



$A_0 \sim A_7$, D_{IN} INPUT VOLTAGE V_{IH2}, V_{IL2} VS. AMBIENT TEMPERATURE



SUPPLY CURRENT VS. TIME



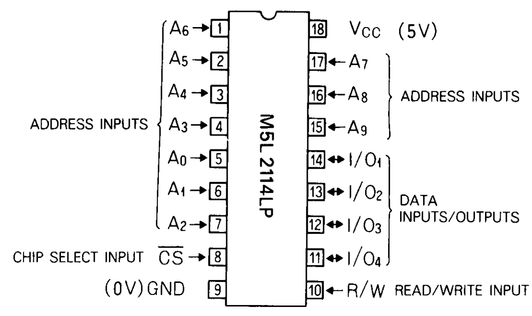
MITSUBISHI ELECTRIC

4096-BIT (1024-WORD BY 4-BIT) STATIC RAM

DESCRIPTION

This is a family of 4096-bit static RAMs organized as 1024 words of 4 bits and designed for simple interfacing. They are fabricated using N-channel silicon-gate MOS technology. They operate with a single 5V supply, as does TTL, and the inputs and outputs are directly TTL compatible. I/O terminals are common.

PIN CONFIGURATION (TOP VIEW)



Outline 18P4

FEATURES

Parameter	M5L 2114LP-2	M5L 2114LP-3	M5L 2114LP
Access time (max)	200ns	300ns	450ns
Cycle time (min)	200ns	300ns	450ns

- Low power dissipation: 50μw/bit (typ)
- Single 5V supply voltage (±10% tolerance)
- Requires neither external clock nor refreshing
- All inputs and outputs are directly TTL compatible
- All outputs are three-state, with OR-tie capability
- Easy memory expansion by chip-select ($\overline{CS}$) input
- Common data I/O terminals
- Interchangeable with Intel's 2114L and TI's TMS4045 in pin configuration and electrical characteristics

APPLICATION

- Small-capacity memory units

FUNCTION

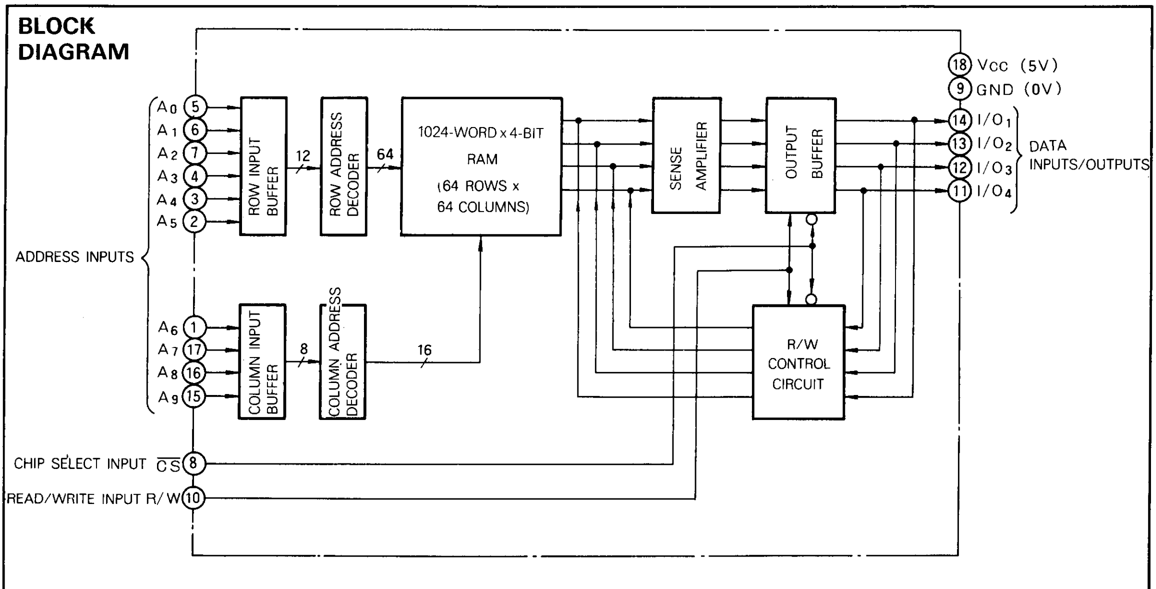
These devices operate with a single 5V power supply, and the inputs and outputs are directly compatible with TTL. All circuits are completely static, rendering external clock and refresh operations unnecessary, and making the members of the series extremely easy to use. Common data input and output terminals are provided.

During a write cycle, when a location is designated by address signals $A_0 \sim A_9$ and the R/W signal goes low, the data at the I/O terminals is written.

During a read cycle, when the R/W signal goes high and a location is designated by address signals $A_0 \sim A_9$, the data of the designated address is available at the I/O terminals.

When signal $\overline{CS}$ is high, the chip is in the non-selectable state, disabling both reading and writing. In this case the data outputs are in the floating (high-impedance) state, useful for OR-ties with the output terminals of other chips.

BLOCK DIAGRAM



4096-BIT (1024-WORD BY 4-BIT) STATIC RAM

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	−0.5~7	V
V_I	Input voltage		−0.5~7	V
V_O	Output voltage		−0.5~7	V
P_d	Maximum power dissipation	Ta=25℃	700	mW
T_{opr}	Operating free-air ambient temperature range		0~70	℃
T_{stg}	Storage temperature range		−65~150	℃

RECOMMENDED OPERATING CONDITIONS (Ta=0~70℃, unless otherwise noted)

Symbol	Parameter	Limits Min	Limits Nom	Limits Max	Units
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{IL}	Low-level input voltage	−0.5		0.8	V
V_{IH}	High-level input voltage	2		V_{CC}	V

ELECTRICAL CHARACTERISTICS (Ta=0~70℃, V_{CC}=5V±10%, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
V_{IH}	High-level input voltage		2		V_{CC}	V
V_{IL}	Low-level input voltage		−0.5		0.8	V
V_{OH}	High-level output voltage	I_{OH}=−200μA, V_{CC}=4.5V	2.4			V
V_{OH}	High-level output voltage	I_{OH}=−1mA, V_{CC}=4.75V	2.4			V
V_{OL}	Low-level output voltage	I_{OL}=2.1mA			0.4	V
I_I	Input current	V_I=0~5.5V			10	μA
I_{OZH}	Off-state high-level output current	$V_{I(\overline{CS})}$=2V, V_O=2.4V~V_{CC}			10	μA
I_{OZL}	Off-state low-level output current	$V_{I(\overline{CS})}$=2V, V_O=0.4V			−10	μA
I_{CC}	Supply current from V_{CC}	V_I=5.5V, (all inputs), output open, Ta=25℃		40	65	mA
C_i	Input capacitance, all inputs	V_I=GND, V_i=25mVrms, f=1MHz		3	5	pF
C_O	Output capacitance	V_O=GND, V_o=25mVrms, f=1MHz		5	8	pF

Note 1 : Current flowing into an IC is positive; out is negative.

TIMING REQUIREMENTS (For Write Cycle) (Ta=0~70℃, V_{CC}=5V±10%, unless otherwise noted) (Note 2)

Symbol	Parameter	Alt. symbol	M5L 2114L P-2 Min	P-2 Typ	P-2 Max	M5L 2114L P-3 Min	P-3 Typ	P-3 Max	M5L 2114L P Min	P Typ	P Max	Unit
$t_{C(WR)}$	Write cycle time	t_{WC}	200			300			450			ns
$t_{SU(AD)}$	Address setup time with respect to write pulse		0			0			0			ns
$t_{W(WR)}$	Write pulse width	t_W	120			150			200			ns
t_{wr}	Write recovery time	t_{WR}	0			0			0			ns
$t_{SU(DA)}$	Data setup time	t_{DW}	120			150			200			ns
$t_{h(DA)}$	Data hold time	t_{DH}	0			0			0			ns
$t_{SU(\overline{CS})}$	Chip select setup time		120			150			200			ns
$t_{PXZ(WR)}$	Output disable time with respect to write pulse	t_{OTW}			40			80			100	ns

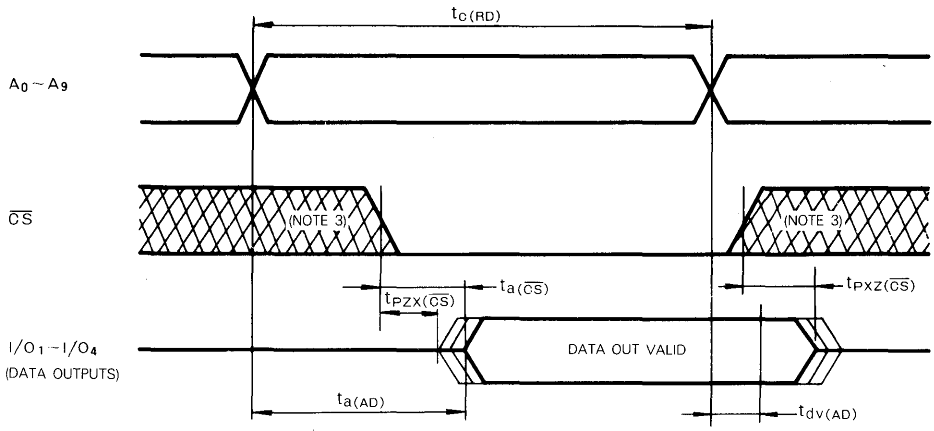
SWITCHING CHARACTERISTICS (For Read Cycle) (Ta=0~70℃, V_{CC}=5V±10%, unless otherwise noted) (Note 2)

Symbol	Parameter	Alt. symbol	M5L 2114L P-2 Min	P-2 Typ	P-2 Max	M5L 2114L P-3 Min	P-3 Typ	P-3 Max	M5L 2114L P Min	P Typ	P Max	Unit
$t_{C(RD)}$	Read cycle time	t_{RC}	200			300			450			ns
$t_{a(AD)}$	Address access time	t_A			200			300			450	ns
$t_{a(\overline{CS})}$	Chip select access time	t_{CO}			80			100			120	ns
$t_{PXZ(\overline{CS})}$	Output disable time with respect to chip select	t_{OTD}			40			80			100	ns
$t_{dv(AD)}$	Data valid time with respect to address	t_{OHA}	50			50			50			ns
$t_{PZX(\overline{CS})}$	Chip select to output active	t_{CX}	20			20			20			ns

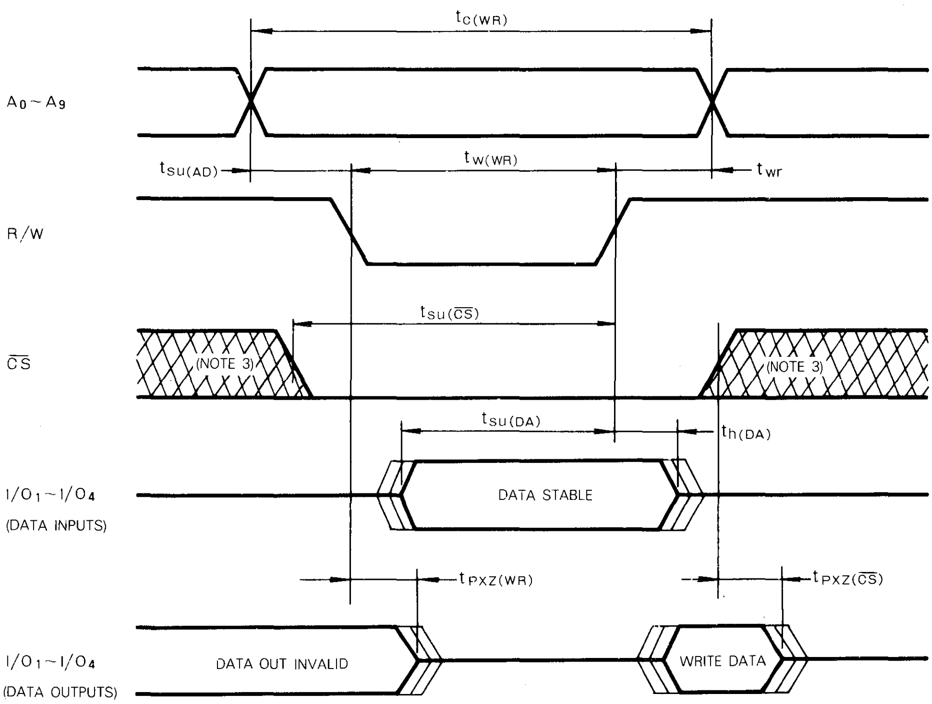
2

TIMING DIAGRAMS

Read Cycle



Write Cycle

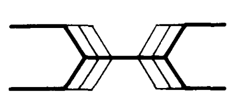


Note : 2 Test conditions

Input pulse level	0.8 ~ 2V
Input pulse rise time	20ns
Input pulse fall time	20ns
Reference level	
Input	1.5V
Output	1.5V

Load = 1TTL, $C_L = 100pF$

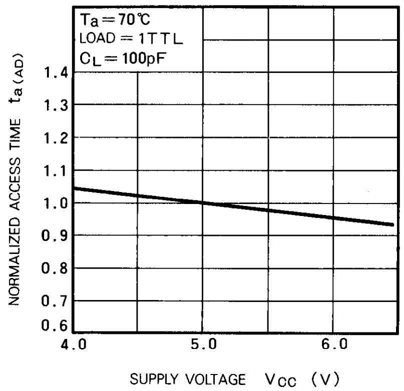
Note 3 : Hatching indicates the state is don't care.



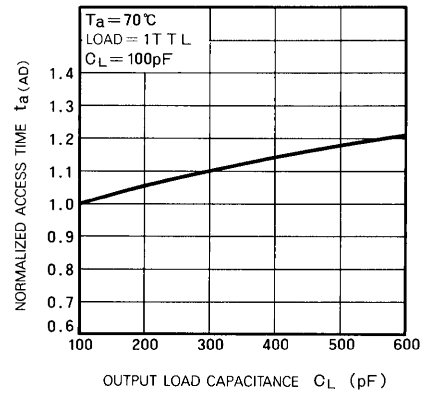
The center line indicates a floating (high-impedance) state.

TYPICAL CHARACTERISTICS

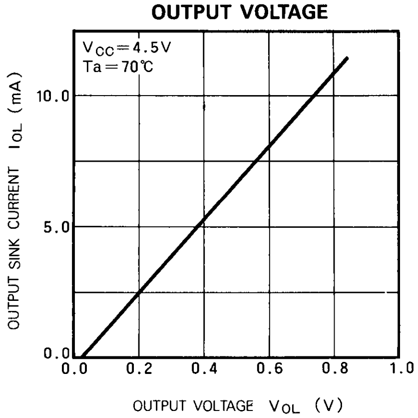
NORMALIZED ACCESS TIME VS. SUPPLY VOLTAGE



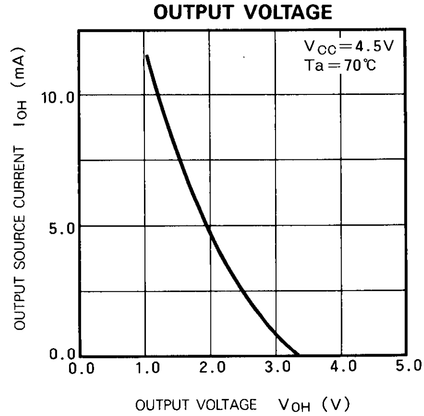
NORMALIZED ACCESS TIME VS. OUTPUT LOAD CAPACITANCE



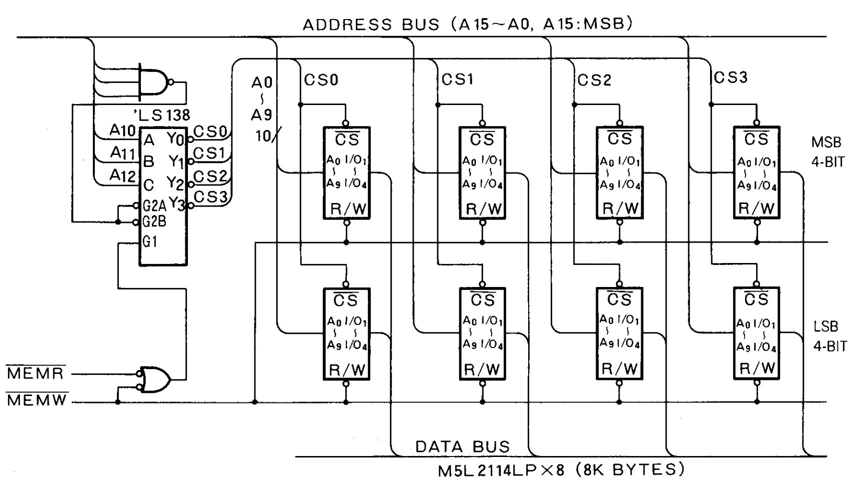
OUTPUT SINK CURRENT VS. OUTPUT VOLTAGE



OUTPUT SOURCE CURRENT VS. OUTPUT VOLTAGE



APPLICATION EXAMPLE (for an M5L8080A P CPU)



1024-BIT (256-WORD BY 4-BIT) CMOS STATIC RAM

DESCRIPTION

This is a 256-word by 4-bit static RAM fabricated with the silicon-gate CMOS process and designed for low power dissipation and easy application of battery back-up.

The device has two chip-select inputs $\overline{CS}_1$ and CS_2. While maintained in the chip non-select state, the device consumes power at the low value of only 10μA (max) standby current and accordingly is especially suitable as a memory system for battery-operated applications and for battery back-up.

The device operates on a single 5V supply, as does TTL, and inputs and outputs are directly TTL-compatible and are provided with common I/O terminals.

FEATURES

- Access time: 450ns (max)
- Low power dissipation in the standby mode: 10μA (max)
- Single 5V power supply
- Data holding at 2V supply voltage
- No external clock or refreshing operation required
- Both inputs and outputs are directly TTL-compatible
- Outputs are three-state, with OR-tie capability
- Simple memory expansion by chip-select signals
- Input and output data terminals are separate
- Interchangeable with Intel's 5101L-1 in pin configuration and electrical characteristics

APPLICATION

- Battery-driven or battery back-up small-capacity memory units

FUNCTION

The device provides separate data input and output terminals.

During a write cycle, when a location is designated by address signals $A_0 \sim A_7$ and signal R/W goes low, the data of the DI inputs at that time is written.

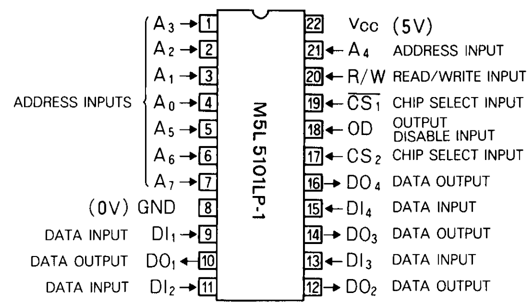
During a read cycle, when a location is designated by address signals $A_0 \sim A_7$, and signal R/W goes high, the data of the designated address is available at the DO terminals.

When signal $\overline{CS}_1$ is high or CS_2 is low, the chip is in the non-selectable state, disabling both reading and writing. In this case, the output is in the floating (high-impedance state) useful for OR-ties with the output terminals of other chips.

When the signal OD is high, the output is in the floating state, so that OD is used as an input/output select control signal for common input/output operation.

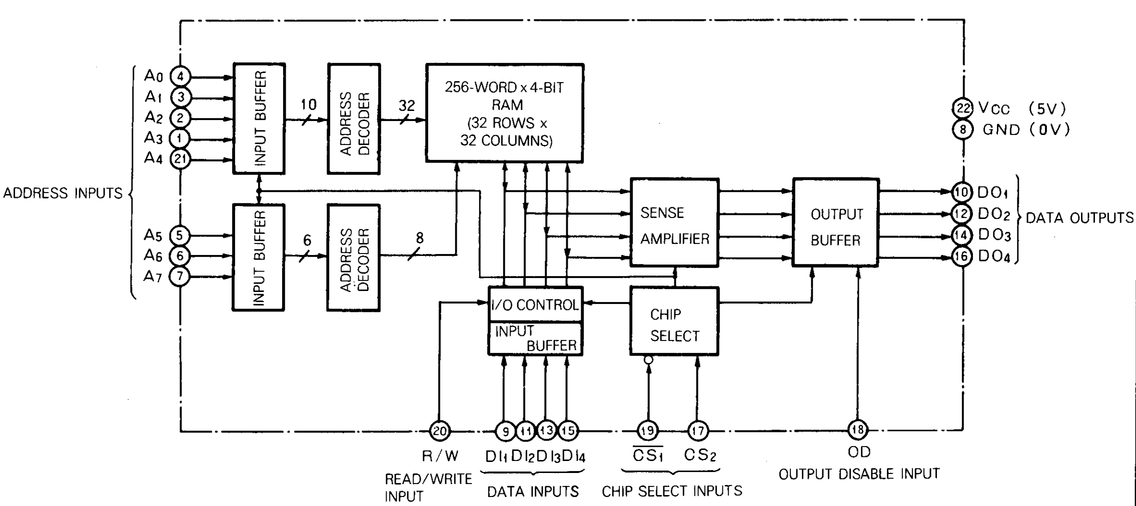
The memory data can be held at a supply voltage of 2V, enabling battery back-up operation during power failure and power-down operation in the standby mode.

PIN CONFIGURATION (TOP VIEW)



Outline 22P1

BLOCK DIAGRAM



1024-BIT (256-WORD BY 4-BIT) CMOS STATIC RAM

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	−0.3~7	V
V_I	Input voltage		$-0.3 \sim V_{CC}+0.3$	V
V_O	Output voltage		$0 \sim V_{CC}$	V
P_d	Maximum power dissipation	$T_a=25°C$	700	mW
T_{opr}	Operating free-air ambient temperature range		0~70	°C
T_{stg}	Storage temperature range		−65~150	°C

RECOMMENDED OPERATING CONDITIONS (Ta = 0 ~ 70℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage	0	0	0	V
V_{IL}	Low-level input voltage	−0.3		0.65	V
V_{IH}	High-level input voltage	2.2		V_{CC}	V

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 70℃, V_{CC} = 5V ± 10%, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{IH}	High-level input voltage		2.2		V_{CC}	V
V_{IL}	Low-level input voltage		−0.3		0.65	V
V_{OH}	High-level output voltage	$I_{OH}=-1mA$			0.4	V
V_{OL}	Low-level output voltage	$I_{OL}=2mA$	2.4			V
I_I	Input current	$V_I=0\sim5.5V$			±1	μA
I_{OZH}	Off-state high-level output current	$V_{I(\overline{CS1})}=2.2V,\ V_O=2.4V\sim V_{CC}$			1	μA
I_{OZL}	Off-state low-level output current	$V_{I(\overline{CS1})}=2.2V,\ V_O=0.4V$			−1	μA
I_{CC1}	Supply current from V_{CC}	$\overline{CS}_1 \leqq 0.65V$, other inputs $=V_{CC}$, Output open		9	22	mA
I_{CC2}	Supply current from V_{CC}	$\overline{CS}_1 \leqq 0.65V$, other inputs $=2.2V$, Output open		13	27	mA
I_{CC3}	Supply current from V_{CC}	$CS_2 \leqq 0.2V$			10	μA
C_I	Input capacitance, all inputs	$V_I=GND$, $V_i=25mVrms$, $f=1MHz$		4	8	pF
C_O	Output capacitance	$V_O=GND$, $V_o=25mVrms$, $f=1MHz$		8	12	pF

Note 1 : Current flowing into an IC is positive; out is negative.

TIMING REQUIREMENTS (For Write Cycle) (Ta = 0 ~ 70℃, V_{CC} = 5V ± 10%, unless otherwise noted)

Symbol	Parameter	Alt. symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{C(WR)}$	Write cycle time	t_{WC}	Input pulse $V_{IH}=2.2V$ $V_{IL}=0.65V$ $t_r=t_f=20ns$ Reference level $=1.5V$ Load $=1TTL$, $C_L=100pF$	450			ns
$t_{W(WR)}$	Write pulse width	t_{WP}		250			ns
$t_{SU(AD)}$	Address setup time with respect to write pulse	t_{AW}		130			ns
t_{WR}	Write recovery time	t_{WR}		50			ns
$t_{SU(OD)}$	OD setup time with respect to data-in	t_{DS}		130			ns
$t_{SU(DA)}$	Data setup time	t_{DW}		250			ns
$t_{h(DA)}$	Data hold time	t_{DH}		50			ns
$t_{SU(CS1)}$	Chip select setup time	t_{CW1}		350			ns
$t_{SU(CS2)}$	Chip select setup time	t_{CW2}		350			ns

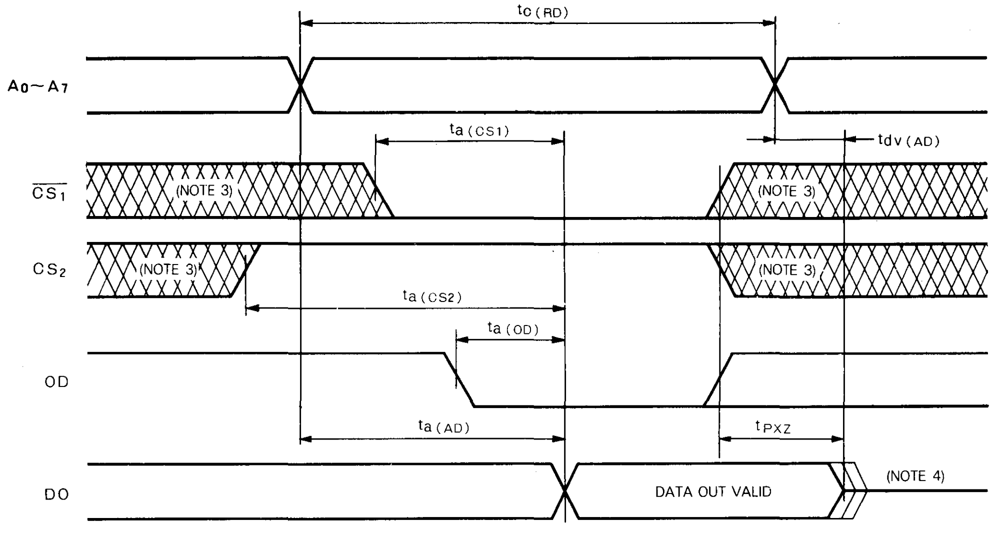
SWITCHING CHARACTERISTICS (For Read Cycle) (Ta = 0 ~ 70°C, V_{CC} = 5V ± 10%, unless without noted)

Symbol	Parameter	Alt. symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{C(RD)}$	Read cycle time	t_{RC}	Input pulse $V_{IH}=2.2V$ $V_{IL}=0.65V$ $t_r=t_f=20ns$ Reference level $=1.5V$ Load $=1TTL$, $C_L=100pF$	450			ns
$t_{a(AD)}$	Address access time	t_A				450	ns
$t_{a(CS1)}$	Chip select access time	t_{CO1}				400	ns
$t_{a(CS2)}$	Chip select access time	t_{CO2}				500	ns
$t_{a(OD)}$	OD access time	t_{OD}				250	ns
t_{PXZ}	Output disable time (note 2)	t_{DF}		0		130	ns
$t_{dv(CS)}$	Date valid time with respect to Chip select			0			
$t_{dv(AD)}$	Data valid time with respect to address	t_{OH1}		0			ns

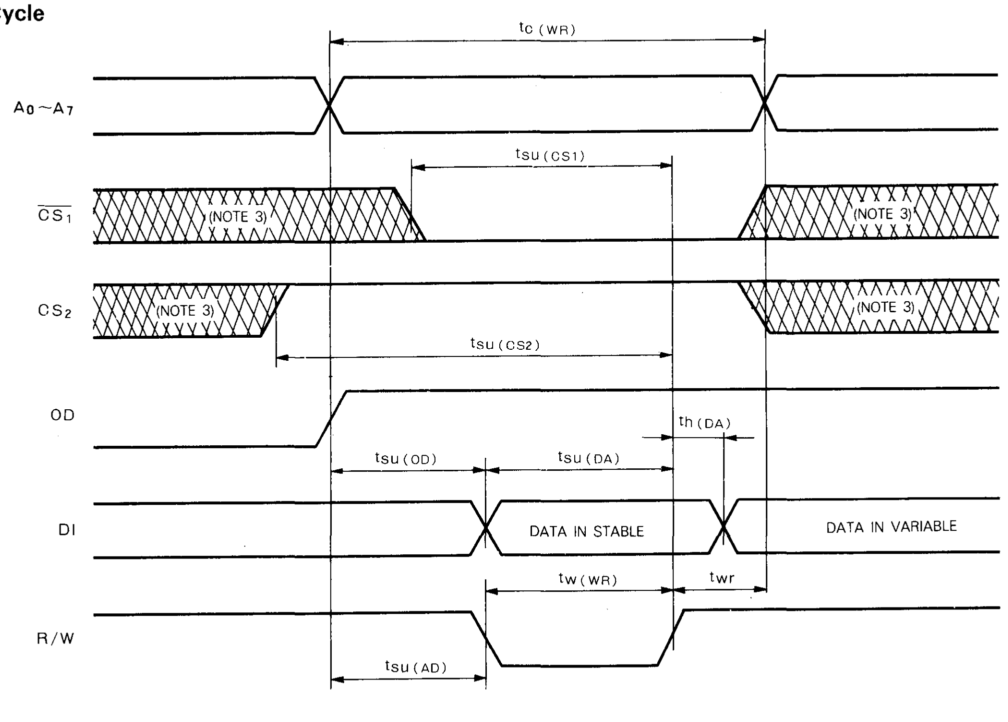
Note 2 : t_{PXZ} is from $\overline{CS_1}$, CS_2, or OD, whichever occurs first.

TIMING DIAGRAMS

Read Cycle

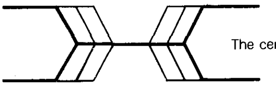


Write Cycle



Note 3 : Hatching indicates the state is unknown.

4 : Indicates that during this period the data-out is invalid for this definition of $t_{dv\,(AD)}$ and is in the floating state for this definition of t_{PXZ}.



The center line indicates a floating (high-impedance) state.

1024-BIT (256-WORD BY 4-BIT) CMOS STATIC RAM

POWER-DOWN OPERATION

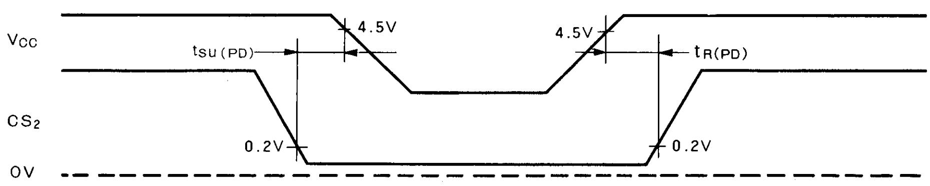
Electrical Characteristics (Ta = 0 ~ 70°C unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$V_{CC(PD)}$	Power-down supply voltage	$CS_2 \leq 0.2V$	2			V
$I_{CC(PD)}$	Power-down supply current from V_{CC}	$V_{CC}=2V$, all inputs = 2 V			10	μA

Timing Requirements (Ta = 0 ~ 70°C, V_{CC} = 5 V ± 10%, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Typ	Max	
$t_{su(PD)}$	Power-down setup time	0			ns
$t_{R(PD)}$	Power-down recovery time	$t_{C(RD)}$			ns

Timing Diagram



MITSUBISHI ELECTRIC

4096-BIT (4096-WORD BY 1-BIT) STATIC RAM

DESCRIPTION

This is a family of 4096-word by 1-bit static RAMs, fabricated with the N-channel silicon-gate MOS process and designed for simple interfacing. They operate with a single 5V supply, as does TTL, and are directly TTL-compatible.

FEATURES

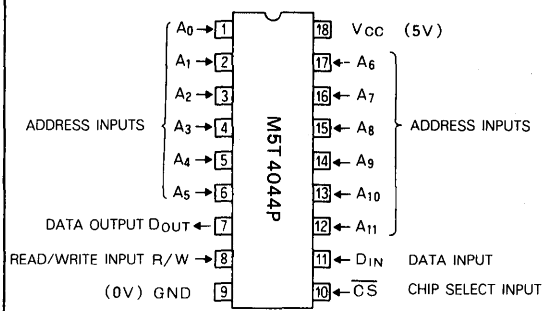
Parameter	M5T 4044P-20	M5T 4044P-30	M5T 4044P-45
Access time (max)	200ns	300ns	450ns
Cycle time (min)	200ns	300ns	450ns

- Low power dissipation: 50μw/bit (typ)
- Single 5V supply (±10% tolerance)
- Requires no clocks or refreshing
- All inputs and outputs are directly TTL-compatible
- All outputs are three-state and have OR-tie capability
- Simple memory expansion by chip-select ($\overline{CS}$) input
- Interchangeable with TI's TMS4044 in pin configuration and electrical characteristics

APPLICATION

- Small-capacity memory units

PIN CONFIGURATION (TOP VIEW)



Outline 18P4

FUNCTION

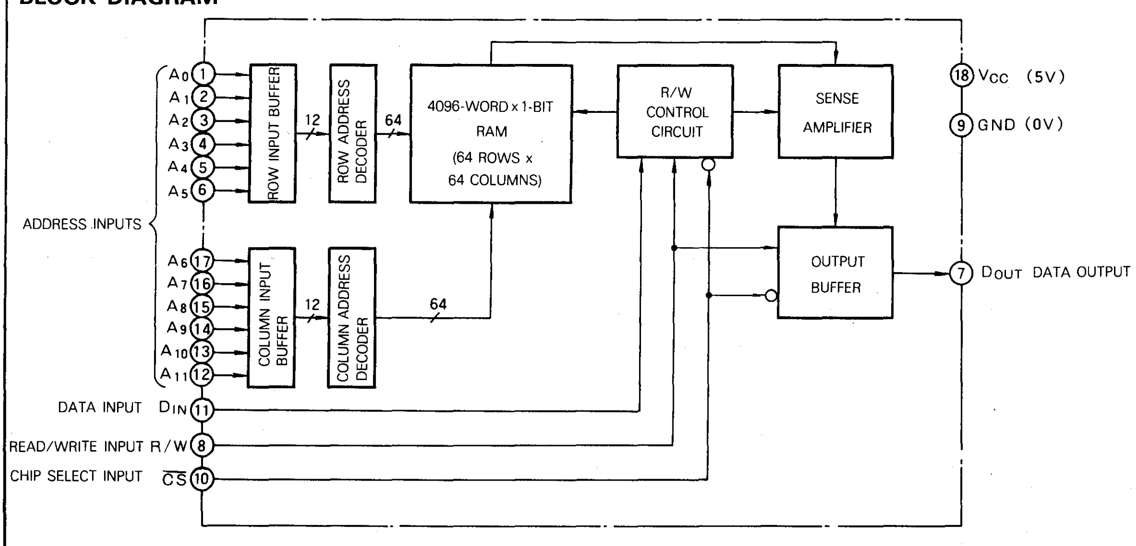
These devices are very convenient to use, as they feature static circuits which require neither external clocks nor refreshing, and all inputs and outputs are directly compatible with TTL.

During a write cycle, when a location is designated by address signals $A_0 \sim A_{11}$ and the R/W signal goes low, the D_{IN} signal data at that time is written.

During a read cycle, when the R/W signal goes high and a location is designated by address signals $A_0 \sim A_{11}$, the data of the designated address is available at the D_{OUT} terminals.

When signal $\overline{CS}$ is high, the chip is in the non-selectable state, disabling both reading and writing. In this case the output is in the floating (high-impedance) state, useful for OR-ties with other output terminals.

BLOCK DIAGRAM



4096-BIT (4096-WORD BY 1-BIT) STATIC RAM

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	−0.5~7	V
V_I	Input voltage		−0.5~7	V
V_O	Output voltage		−0.5~7	V
P_d	Maximum power dissipation	Ta=25℃	700	mW
T_{opr}	Operating free-air ambient temperature range		−65~150	℃
T_{stg}	Storage temperature range		−40~125	℃

RECOMMENDED OPERATING CONDITIONS (Ta = 0 ~ 70℃, unless otherwise noted)

Symbol	Parameter	Limits			Units
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{IL}	Low-level input voltage	−0.5		0.8	V
V_{IH}	High-level input voltage	2		V_{CC}	V

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 70℃, V_{CC} = 5V ± 10%, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{IH}	High-level input voltage		2		V_{CC}	V
V_{IL}	Low-level input voltage		−0.5		0.8	V
V_{OH}	High-level output voltage	I_{OH} = −200μA, V_{CC}=4.5V	2.4			V
V_{OH}	High-level output voltage	I_{OH} = −1.0mA, V_{CC}=4.75V	2.4			V
V_{OL}	Low-level output voltage	I_{OL}=2.1mA			0.4	V
I_I	Input current	V_I=0~5.5V			10	μA
I_{OZH}	Off-state high-level output current	$V_{I(\overline{CS})}$=2V, V_O=2.4V~V_{CC}			10	μA
I_{OZL}	Off-state low-level output current	$V_{I(\overline{CS})}$=2V, V_O=0.4V			−10	μA
I_{CC}	Supply current from V_{CC}	V_I=5.5V, (all inputs), output open, Ta=25°C		40	65	mA
C_i	Input capacitance, all inputs	V_I=GND, V_i=25mVrms, f=1MHz		3	5	pF
C_o	Output capacitance	V_O=GND, V_o=25mVrms, f=1MHz		5	8	pF

Note 1 : Current flowing into an IC is positive; out is negative.

TIMING REQUIREMENTS (For Write Cycle) (Ta = 0 ~ 70℃, V_{CC} = 5V ± 10%, unless otherwise noted) (Note 2)

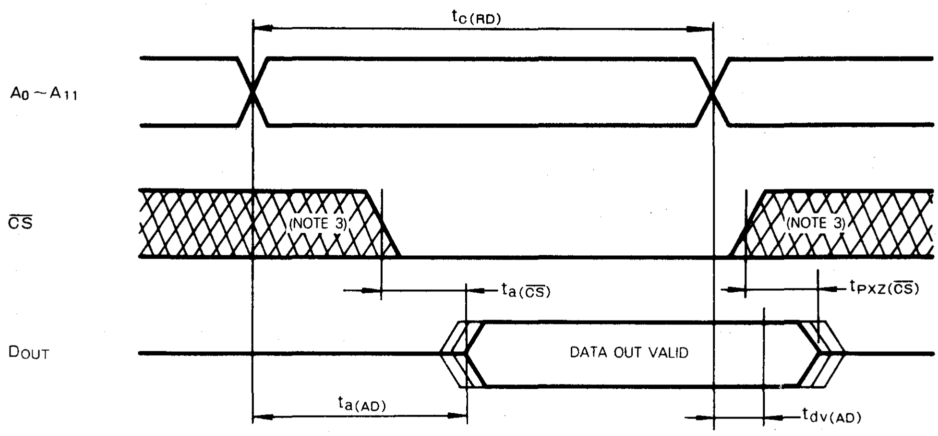
Symbol	Parameter	M5T 4044P-20			M5T 4044P-30			M5T 4044P-45			Unit
		Limits			Limits			Limits			
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
$t_{C(WR)}$	Write cycle time	200			300			450			ns
$t_{SU(AD)}$	Address setup time with respect to write pulse	0			0			0			ns
$t_{W(WR)}$	Write pulse width	120			150			200			ns
t_{WR}	Write recovery time	0			0			0			ns
$t_{SU(DA)}$	Data setup time	120			150			200			ns
$t_{h(DA)}$	Data hold time	0			0			0			ns
$t_{SU(\overline{CS})}$	Chip select setup time	120			150			200			ns
$t_{PXZ(WR)}$	Output disable time with respect to write pulse			40			80			100	ns

SWITCHING CHARACTERISTICS (For Read Cycle) (Ta = 0 ~ 70℃, V_{CC} = 5V ± 10%, unless otherwise noted) (Note 2)

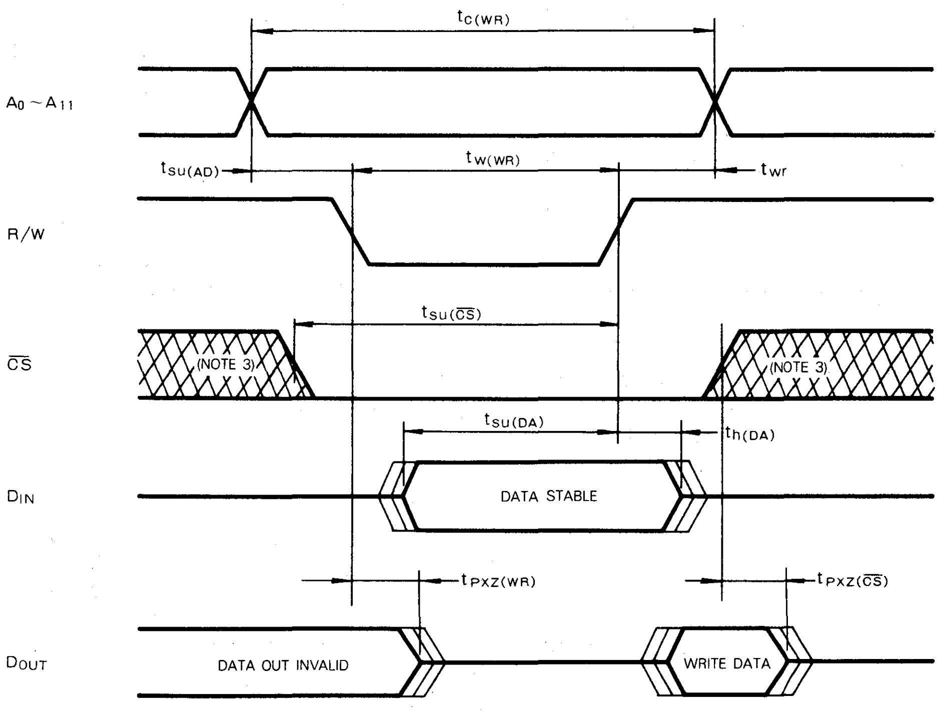
Symbol	Parameter	M5T 4044P-20			M5T 4044P-30			M5T 4044P-45			Unit
		Limits			Limits			Limits			
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
$t_{C(RD)}$	Read cycle time	200			300			450			ns
$t_{a(AD)}$	Address access time			200			300			450	ns
$t_{a(\overline{CS})}$	Chip select access time			70			100			100	ns
$t_{PXZ(\overline{CS})}$	Output disable time with respect to chip select			40			80			100	ns
$t_{dv(AD)}$	Data valid time with respect to address	50			50			50			ns

2

TIMING DIAGRAMS
Read Cycle



Write Cycle

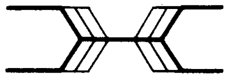


Note : 2 Test conditions

Input pulse level	0.8 ~ 2V
Input pulse rise time	20ns
Input pulse fall time	20ns
Reference level	
Input	1.5V
Output	1.5V

Load = 1TTL, C_L = 100pF

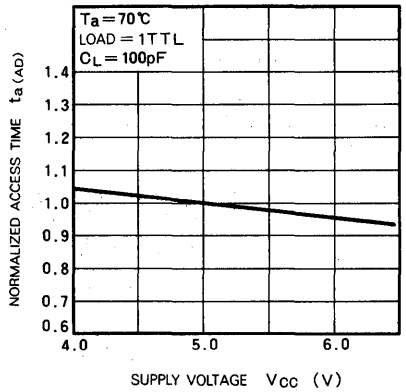
Note 3 : Hatching indicates the state is don't care.



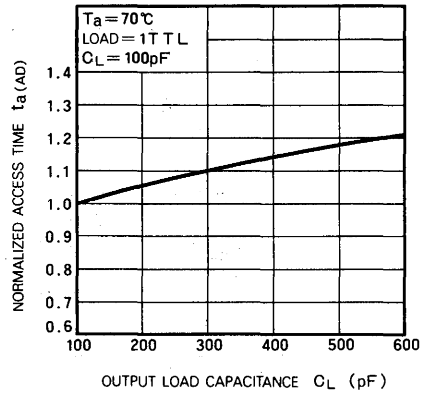
The center line indicates a floating (high-impedance) state.

TYPICAL CHARACTERISTICS

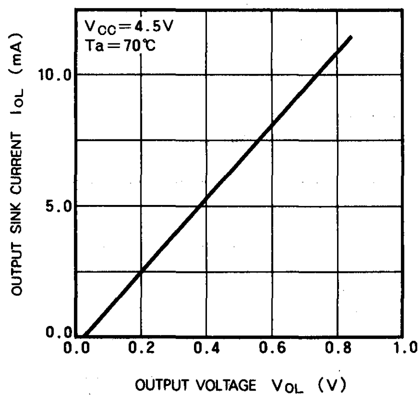
NORMALIZED ACCESS TIME VS. SUPPLY VOLTAGE



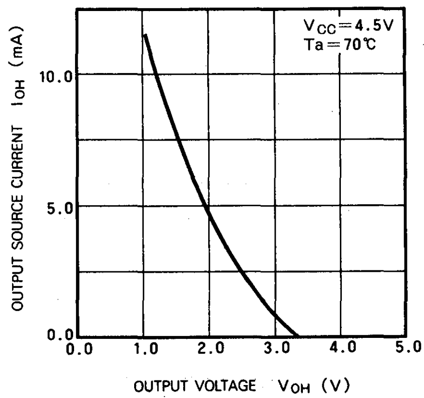
NORMALIZED ACCESS TIME VS. OUTPUT LOAD CAPACITANCE



OUTPUT SINK CURRENT VS. OUTPUT VOLTAGE

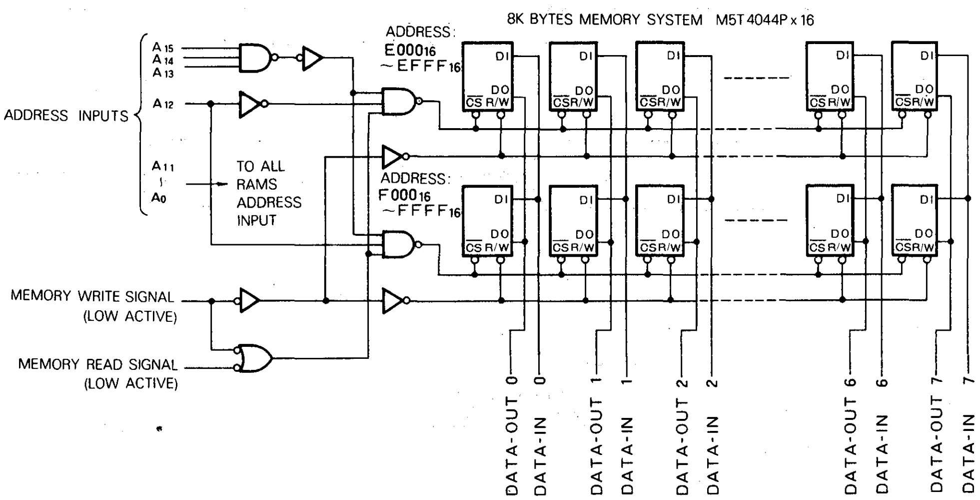


OUTPUT SOURCE CURRENT VS. OUTPUT VOLTAGE



APPLICATION EXAMPLE (for 8K-Byte Memory System)

This circuit is designed for a separate data bus application; if a common data bus application is required, the output and input can be tied.



READ-ONLY MEMORIES

3

DEVELOPMENT OF MASK-PROGRAMMABLE ROMs

GENERAL INFORMATION

This information explains how to specify the object program for the automatic design system for mask ROMs. This system for mask ROM production has been developed to accept a customer's specifications in a number of forms and media.

The main segments of the automatic design system are:

1. The plotter instructions for mask production.
2. A check list for verifing that the customer's specifications have been met.
3. A test program to assure that the production ROMs meet specifications.

When loading an object program into masked ROM, the object program is provided at the time of ordering in the form of EPROM memory. The EPROM device or devices are sent with the required verification sheets, three devices of each type.

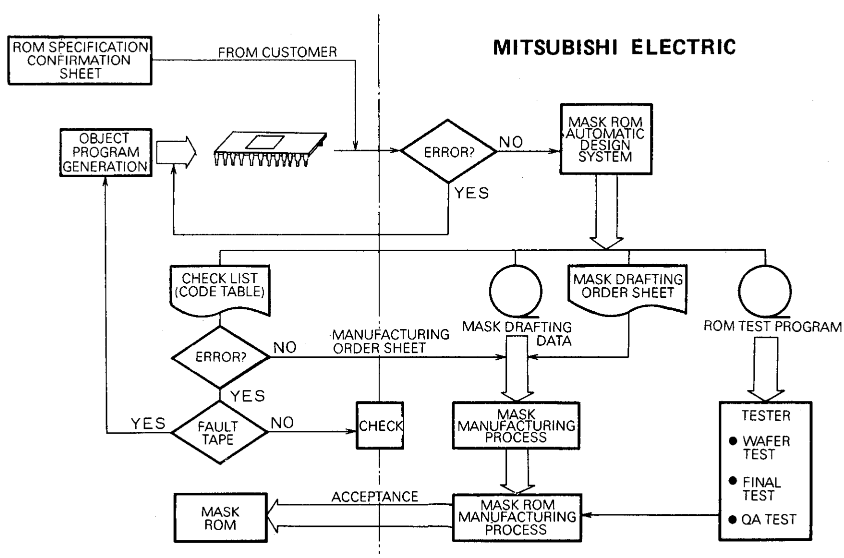
EPROM SPECIFICATIONS

1. Mitsubishi M5L2708K, M5L2716K, M5L5732K and Intel 2708, 2716, or 2764 or equivalent EPROMs may be used. The standard for ordering is, however, Mitsubishi M5L2708K, M5L2716K and M5L2732K EPROM.
2. EPROM data and addresses should be programmed with the high logic level as a logic 1.
3. The EPROM data should include all valid data from the starting address to the last address to be programmed.

ITEMS TO BE VERIFIED

1. Specify the type number as M58333-XXXP, M58735-XXXP, or M58334-XXXP (the three digits after the hyphen of the type number is a code to indicate the customer contents and is assigned by Mitsubishi).
2. Mark the EPROM device tops with the EPROM type and the address specification symbols A,B,C, or D.
   The addresses indicated by these symbols are listed on the ROM verification sheets.

MASK ROM DEVELOPMENT PROCESS



Mitsubishi M58735-XXXP Masked ROM Verification Sheet

MITSUBISHI ELECTRIC

Customer		Signature
Company name ______________________		Prepared
Company address ______________________	Tel __________	
Company contact ______________________	Date __________	Approved

* 1. Specify the EPROM to be supplied.
Three copies of each type of EPROM are required.
(Place a check in the appropriate boxes)

Type \ EPROM type	2708	2716	2732	2764
□M58735	□ A (0000~03FF) B (0400~07FF) C (0800~0BFF) D (0C00~0FFF)	□ A (0000~07FF) B (0800~0FFF)	□A (0000~0FFF)	□A (0000~0FFF)

Chip Select Input (combinations for output)
Circle the desired programmed level as H (high) or L (low).

	CS1	CS2
M58735	H, L	H, L

* 2. Part No.

1. Mark not required　　　　2. Mark required

MITSUBISHI IC TYPE NO.

Note 1. The marking is located flush right
2. The length is limited to 12 character combinations of letters, numbers and hyphens. The letters J, I, and O should not be used, however.

* 3. Special Notes

* 4. Description of final product (describe in as much detail as possible)

1024-BIT (256-WORD BY 4-BIT) FIELD-PROGRAMMABLE ROM WITH OPEN COLLECTOR OUTPUTS

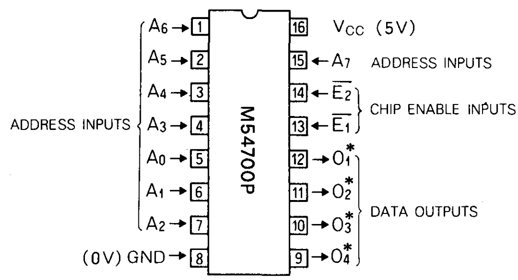
DESCRIPTION

The memory cells of the M54700P,S are a 256-word by 4-bit matrix of diodes and Ni-Cr fuse links. Data can be electrically programmed by open-circuiting fuses in the field with simple programming equipment. These 1024-bit field-programmable ROMs (PROMs) are composed of an address decoder, memory, output and chip enable TTL circuits.

FEATURES

- Field-programmable ROM
- Low power dissipation: 0.40mW/bit
- Fast access time: 50ns (typ)
- 5V±5% single supply voltage
- Inputs and outputs TTL-compatible
- Open collector outputs
- Two chip enable inputs ($\overline{E_1}$, $\overline{E_2}$) for easy memory expansion
- Organized as 256 words of 4 bits
- 16-pin ceramic or plastic package
- Interchangeable with MMI's 6300 in pin configuration and electrical characteristics

PIN CONFIGURATION (TOP VIEW)



Outline 16P1 (M54700P)
16S1 (M54700S)

*: OPEN-COLLECTOR OUTPUTS

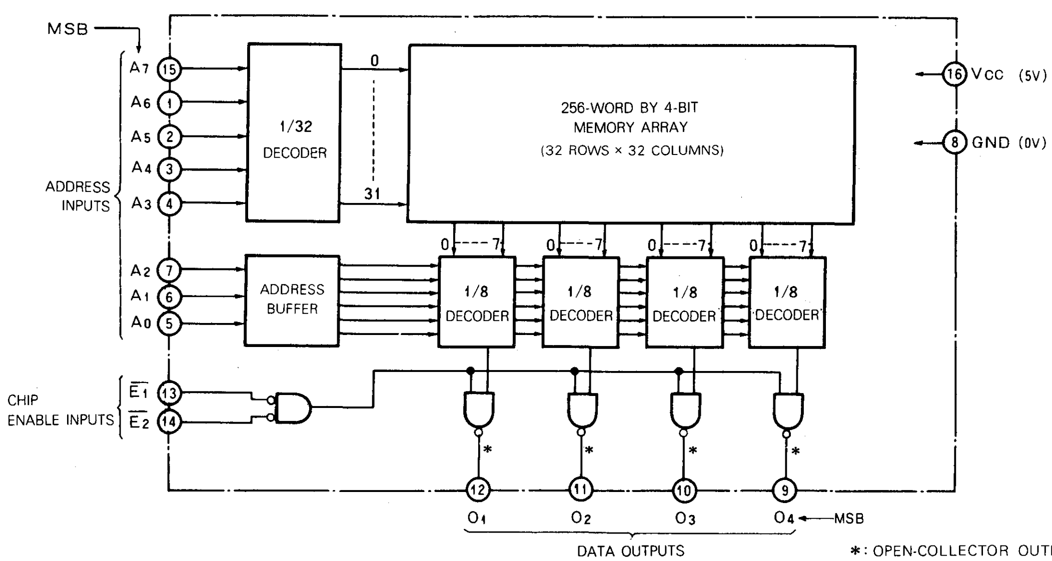
APPLICATION

- Programmable memory for the M5L 8080A 8-bit parallel CPU. Used for prototype design, microprogramming and control storage.

FUNCTION

This device is accessed by address inputs $A_0 \sim A_7$, selecting one of 256 words. The 4-bits are read out in parallel on data outputs $0_1 \sim 0_4$. All inputs are TTL-compatible. An external decoder is not necessary. All outputs are open-collector outputs, so it is possible to AND-tie them to other ROMs and TTL devices. The AND-tie fanout of each output can accommodate up to 10 standard TTL loads. The chip enables $\overline{E_1}$ and $\overline{E_2}$ are used to inhibit data outputs $O_1 \sim O_4$.

BLOCK DIAGRAM



*: OPEN-COLLECTOR OUTPUTS

ABSOLUTE MAXIMUM RATINGS (Ta = 25°C, unless otherwise noted)

	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage		7	V
V_I	Input voltage		5.5	V
V_O	Output voltage		V_{CC}	V
Topr	Operating free-air temperature range		0~75	°C
Tstg	Storage temperature range		−65~150	°C
V_O	Output apply voltage	In case of programming	27	V
V_E	Chip enable apply voltage	In case of programming	35	V
$t_{w(P)}/t_{c(P)}$	Duty cycle	In case of programming	25	%

READ OPERATION

Recommended Operating Conditions (Ta = 0~75°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V

Electrical Characteristics (Ta = 0~75°C, unless otherwise noted)

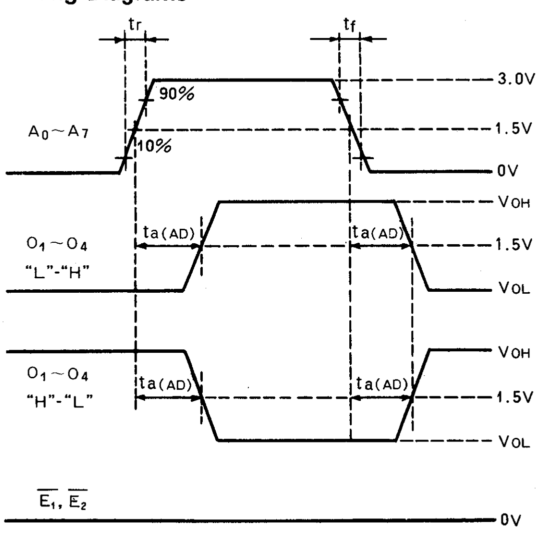
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ(Note 1)	Max	
V_{IH}	High-level input voltage		2			
V_{IL}	Low-level input voltage				0.8	
V_{OL}	Low-level output voltage	I_{OL} = 16mA		0.3	0.45	V
I_{OH}	High-level output current	V_{OH} = 5.25V			100	μA
I_{IL}	Low-level input current	V_I = 0.4V			−1.6	mA
I_{IH}	High-level input current	V_I = 2.4V			40	μA
		V_I = 4.5V			60	
I_{CC}	Supply current from V_{CC}			85	125	mA
V_{IC}	Input clamped voltage	I_I = −10mA			−1.5	V

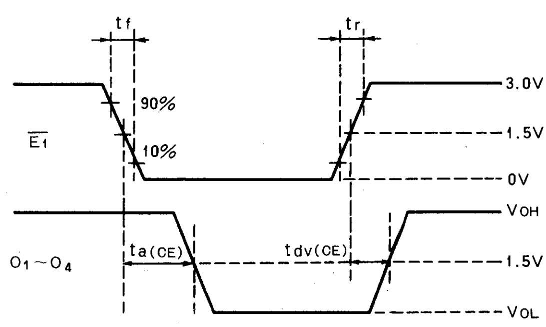
Note 1 : Typical values are at V_{CC} = 5V, Ta = 25°C

Switching Characteristics (V_{CC} = 5V, Ta = 25°C, unless otherwise noted)

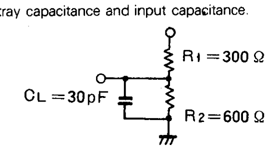
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{a(AD)}$	Address access time (Note 3)	See Timing Diagrams and Note 4			90	ns
$t_{a(CE)}$	Chip enable access time				50	ns
$t_{dv(CE)}$	Data valid time with respect to chip enable				50	ns

Timing Diagrams





Note 2 : Rise time t_r ≦5ns; fall time t_f ≦5ns
3 : The chip enable inputs $\overline{E_1}$ and $\overline{E_2}$ should be low-level at measurement time during address access time.
4 : Load circuit: capacitance (C_L) includes stray capacitance and input capacitance.



PROGRAMMING OPERATION

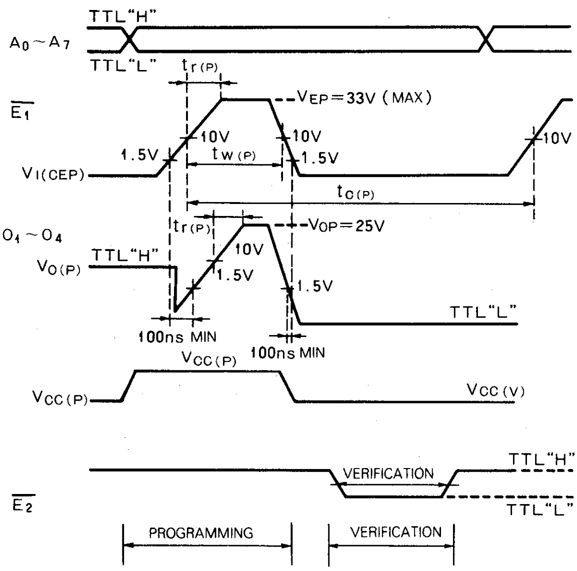
Recommended Operating Conditions

Symbol	Test conditions	Limits Min	Limits Nom	Limits Max	Unit
$V_{I(CEP)}$	Chip enable program input voltage	29		33	V
$V_{O(P)}$	Output apply voltage			25	V
$V_{CC(P)}$	Program input voltage	5.40	5.50	5.60	V
$V_{CC(V)}$	Program verify input voltage	4.10	4.20	4.30	V

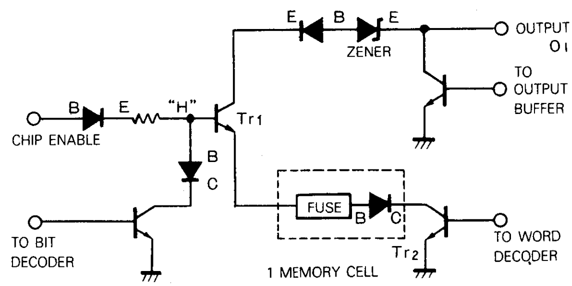
Timing Requirements

Symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{r(P)}$	Pulse rise time	10	25	100	μs
$t_{w(P)}$	Pulse width	0.04		100	ms
$t_{w(P)}/t_{c(P)}$	Duty cycle			25	%

Timing Diagram



Programming Circuit



Programming (Writing) Procedure

All 1024 Ni-Cr fuse-link memory elements are manufactured in a high-logic-level (fuse closed) output conditon. To program:

1. Apply 5.5V to the supply voltage V_{CC} and select a fuse link to be programmed with address inputs A_0~A_7.
2. Apply a high-logic-level to the chip enable input $\overline{E_2}$.
3. After applying a program pulse $V_{I(CEP)}$ to the chip enable input $\overline{E_1}$ (see Timing Diagrams), apply an output pulse $V_{O(P)}$ to the fuse link of the output to be programmed. The output pulses should be separately applied to each output.
4. After programming, the fuse link is open and the output level is changed to a low-logic-level.
5. After programming is completed, apply an additional three programming pulses.
6. Test the programmed memory to verify that the outputs are low-level or high-level as desired. Both chip enable inputs $\overline{E_1}$ and $\overline{E_2}$ must be low-level for testing.

The word decoder circuit selects any one of 32 columns, and sets the transistor Tr_2 to the on state. The bit decoder circuit selects any four of 32 rows, and supplies the base current to transistor Tr_1 from chip enable input $\overline{E_1}$.

The fuse link is opened not by the base current, but by the collector current which is supplied to transistor Tr_1 from the selected output O_1~O_4, plus the base current. At this time, the other three fuse links of the selected word line are in a half-selected stage and the remaining 1020 fuse links are in a non-selected state.

Typical Programming Conditions

Condition sequence	Pulse sequence	Pulse width $t_{w(P)}$ (ms)	Chip enable program voltage $V_{I(CEP)}$ (V)	Output voltage (V)
1	1 ~ 4	0.5	29	25
2	5 ~ 8	1	29	25
3	9 ~ 12	5	30	25
4	13~19	20	33	25

APPLICATIONS

Chip Enable Circuit

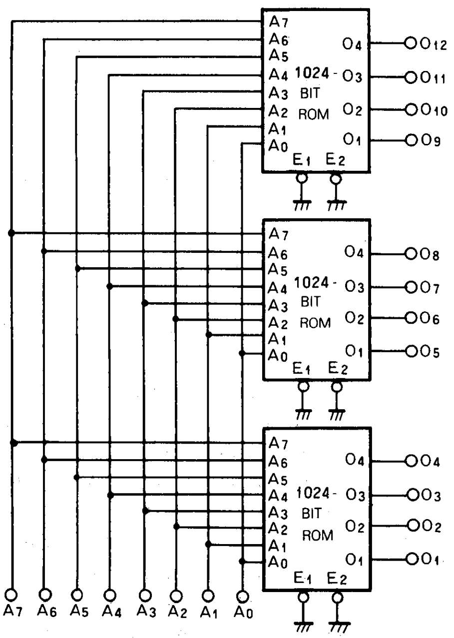
The chip enable inputs $\overline{E_1}$ and $\overline{E_2}$ are used for activating or inhibiting output $O_1 \sim O_4$. $\overline{E_1}$ and $\overline{E_2}$ are NORed. Output is inhibited when any of the inputs are high-logic-level. Chip enable inputs $\overline{E_1}$ and $\overline{E_2}$ allow easy memory expansion by one of the following procedures:

1. Expanding the Number of Bits in a Word

For example, using three 1024-bit ROMs, each organized as 256 words of 4 bits, the number of bits in a word can be expanded as described below:

1. Apply a low-logic-level to both chip enable inputs $\overline{E_1}$ and $\overline{E_2}$ of each ROM.
2. Connect address inputs $A_0 \sim A_7$ of each ROM in parallel. Memory is thus expanded and reorganized as 256 words of 12 bits.

Fig. 1 Expansion of number of bits

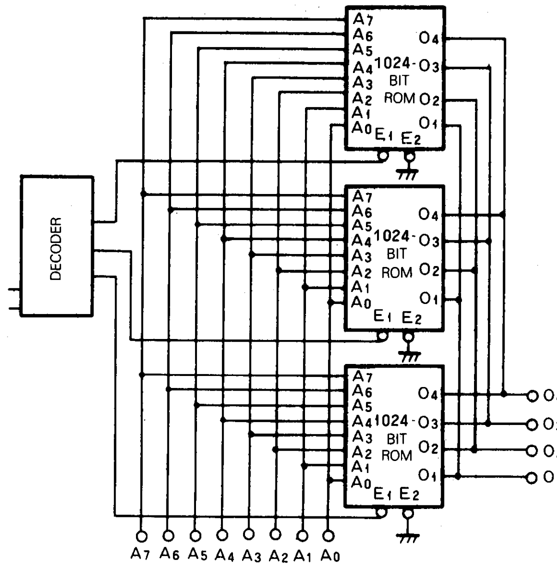


2. Expanding the Number of Words in Memory

For example, using three 1024-bit ROMs, each organized as 256 words of 4 bits, the number of words in memory can be expanded as described below:

1. Connect one of the chip enable inputs $\overline{E_1}$ or $\overline{E_2}$ of each ROM to the decoder while keeping the remaining input at low-logic-level.
2. Connect the outputs from each ROM with AND-tie connections so that each output is an open-collector output circuit or a three-state output. Memory is thus expanded and organized as 768 words of 4 bits.

Fig. 2 Expansion of number of words



3. Expanding the Number of Words in Memory and the Number of Bits in a Word

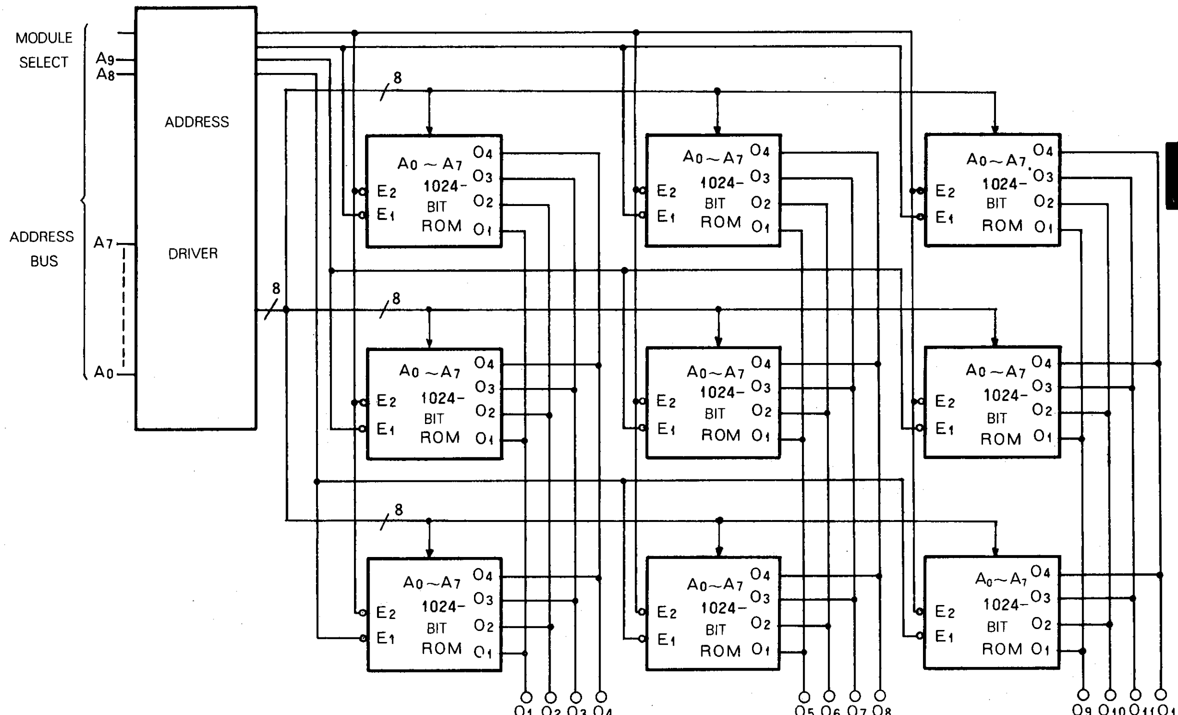
For example, using nine 1024-bit ROMs, each organized as 256 words of 4 bits, and by combining procedures 1 and 2 above, the number of words in memory along with the number of bits in a word, can be expanded as described below:

1. The chip enable input $\overline{E_2}$ of all ROMs is connected in parallel for module selection.
2. The chip enable input $\overline{E_1}$ activates selected ROMs the same as 2 above.

   Memory is thus expanded and reorganized as 768 words of 12 bits.

1024-BIT (256-WORD BY 4-BIT) FIELD-PROGRAMMABLE ROM WITH OPEN COLLECTOR OUTPUTS

Fig. 3 ROM module



Pull-up Resistors

The outputs are open collectors; therefore, AND-tie connections are also possible, and normal loads can be connected. The resistance of a pull-up resistor RL that may be connected between the voltage supply and the collectors of the output transistors should be determined by equations (1) and (2) as shown below:

$$R_L(max) = \frac{\overline{V_{CC}} - \underline{V_{OH}}}{M \cdot \overline{I_{OH}^{*}} + N \cdot \overline{I_{IH}}} \quad \cdots\cdots\cdots (1)$$

where, M : number of AND-ties

N : number of fanouts (number of loads)

$\overline{V_{CC}}$: maximum value of supply voltage

$\underline{V_{OH}}$: minimum value of high-level output voltage

$\overline{I_{OH}^{*}}$: maximum value of high-level output current at the open collector output

$\overline{I_{IH}}$: maximum value of high-level input current

$$R_L(min) = \frac{\underline{V_{CC}} - \overline{V_{OL}}}{\overline{I_{OL}} - N \cdot \overline{I_{IL}}} \quad \cdots\cdots\cdots (2)$$

where, $\underline{V_{CC}}$: minimum value of supply voltage

$\overline{V_{OL}}$: maximum value of low-level output voltage

$\overline{I_{OL}}$: maximum value of low-level output current

$\overline{I_{IL}}$: maximum value of low-level input current

then,

$$R_L(min) < R_L < R_L(max) \quad \cdots\cdots\cdots (3)$$

The resistance of a pull-up resistor R_L should be within the range as shown in equation (3). R_L (min) and R_L (max) should be calculated using the appropriate number of AND-ties and fanouts. Calculation examples of TTL load are shown below:

(1) When

$M=4$, $N=3$, $\overline{V_{CC}}=5.25V$, $\underline{V_{OH}}=2.4V$, $\overline{I_{OH}^{*}}=100\mu A$, $\overline{I_{IH}}=40\mu A$

$$R_L(max) = \frac{\overline{V_{CC}} - \underline{V_{OH}}}{M \cdot \overline{I_{OH}^{*}} + N \cdot \overline{I_{IH}}} = \frac{5.25V - 2.4V}{4 \times (100\mu A) + 3 \times (40\mu A)} = 5090\Omega$$

(2) When

$N=3$, $\underline{V_{CC}}=4.75V$, $\overline{V_{OL}}=0.45V$, $\overline{I_{OL}}=16mA$, $\overline{I_{IL}}=1.6mA$

$$R_L(min) = \frac{\underline{V_{CC}} - \overline{V_{OL}}}{\overline{I_{OL}} - N \cdot \overline{I_{IL}}} = \frac{4.75V - 0.45V}{16mA - 3 \times (1.6mA)} = 384\Omega$$

256-BIT (32-WORD BY 8-BIT) FIELD-PROGRAMMABLE ROM WITH OPEN COLLECTOR OUTPUTS

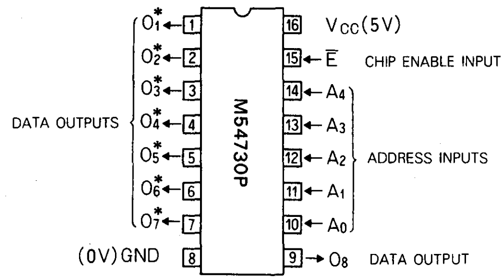
DESCRIPTION

The memory cells of the M54730P, S are a 32-word by 8-bit matrix of diodes and Ni-Cr fuse links. Data can be electrically programmed by open-circuiting fuses in the field with simple programming equipment. These 256-bit field programmable ROMs (PROMs) are composed of an address decoder, memory, output and chip enable TTL circuits.

FEATURES

- Field programmable ROM
- Low power dissipation: 1.5mW/bit
- Fast access time: 45ns (typ)
- 5V±5% single supply voltage
- Inputs and outputs TTL-compatible
- Open-collector outputs
- Chip enable inputs ($\overline{E}$) for easy memory expansion
- Organized as 32 words of 8 bits
- 16-pin ceramic or plastic package
- Interchangeable with MMI's 6300 in pin configuration and electrical characteristics

PIN CONFIGURATION (TOP VIEW)



Outline 16P1 (M54730P)
16S1 (M54730S)

*: OPEN-COLLECTOR OUTPUTS

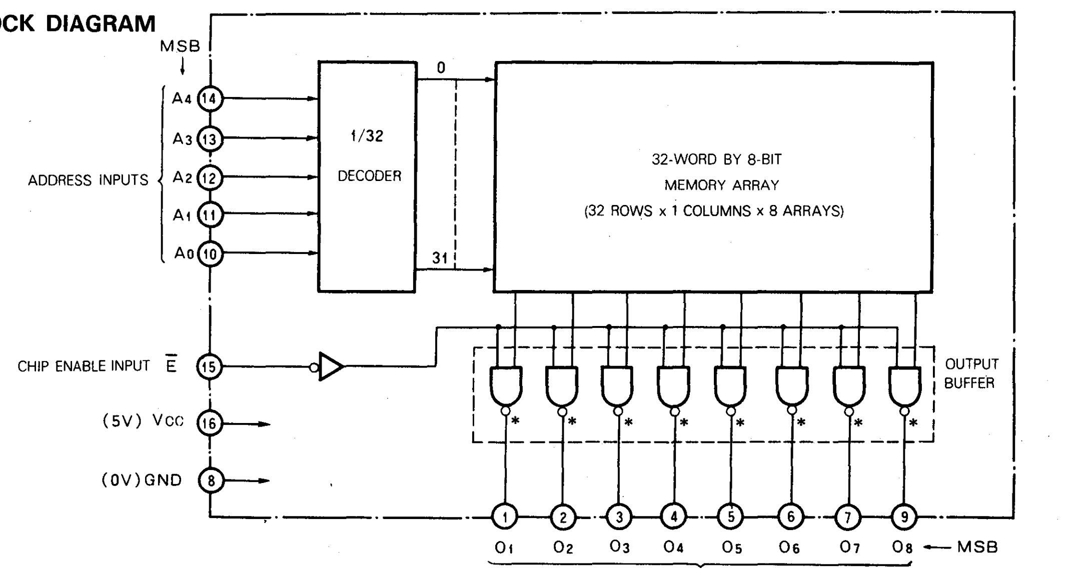
APPLICATION

- Programmable memory for the M5L 8080A 8-bit parallel CPU. Used for prototype design, microprogramming and control strage.

FUNCTION

This device is accessed by address inputs $A_0 \sim A_4$, selecting one of 32 words. The 8 bits are read out in parallel on data outputs $O_1 \sim O_8$. All inputs are TTL-compatible. An external decoder is not necessary. All outputs are open-collector outputs, so it is possible to AND-tie them to other ROMs and TTL devices. The AND-tie fanout of each output can accommodate up to 10 standard TTL loads. The chip enable $\overline{E}$ is used to inhibit data outputs $O_1 \sim O_8$.

BLOCK DIAGRAM



*: OPEN-COLLECTOR OUTPUTS

MITSUBISHI ELECTRIC

256-BIT (32-WORD BY 8-BIT) FIELD-PROGRAMMABLE ROM WITH OPEN COLLECTOR OUTPUTS

ABSOLUTE MAXIMUM RATINGS (Ta = 25°C, unless otherwise noted)

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage		7	V
V_I	Input voltage		5.5	V
V_O	Output voltage		V_{CC}	V
T_{opr}	Operating free-air temperature range		0 ~ 75	°C
T_{stg}	Storage temperature range		−65 ~ 150	°C
V_O	Output apply voltage	In case of programming	27	V
$t_{w(P)}/t_{c(P)}$	Duty cycle	In case of programming	25	%

READ OPERATION

Recommended Operating Conditions (Ta = 0 ~ 75°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V

Electrical Characteristics (Ta = 0 ~ 75°C, unless otherwise noted)

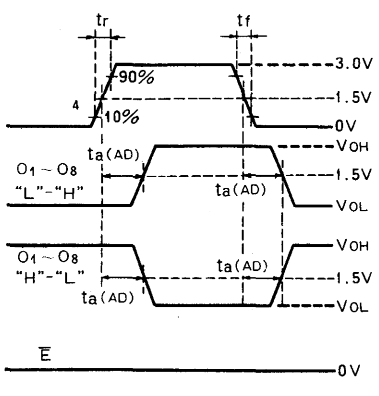
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ(Note 1)	Max	
V_{IH}	High-level input voltage		2			
V_{IL}	Low-level input voltage				0.8	
V_{OL}	Low-level output voltage	I_{OL} = 16mA		0.3	0.45	V
I_{OH}	High-level output current	V_{OH} = 5.25V			100	μA
I_{IL}	Low-level input current	V_I = 0.4V			−1.6	mA
I_{IH}	High-level input current	V_I = 2.4V			40	μA
		V_I = 4.5V			60	
I_{CC}	Supply current from V_{CC}			85	125	mA
V_{IC}	Input clamped voltage	I_I = −10mA			−1.5	V

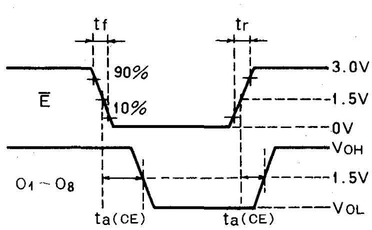
Note 1 : Typical values are at V_{CC} = 5V, Ta = 25°C.

Switching Characteristics (V_{CC} = 5V, Ta = 25°C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{a(AD)}$	Address access time	See Timing Diagrams			80	ns
$t_{a(CE)}$	Chip enable access time				50	ns
$t_{dv(CE)}$	Data valid time with respect to chip enable				50	ns

Timing Diagrams

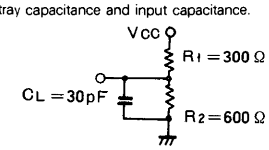




Note 2 : Rise time t_r ≦5ns; fall time t_f ≦5ns

3 : The chip enable input E should be low-level at measurement time during address access time.

4 : Load circuit: capacitance (C_L) includes stray capacitance and input capacitance.



1024-BIT (256-WORD BY 4-BIT) FIELD-PROGRAMMABLE ROM WITH OPEN COLLECTOR OUTPUTS

PROGRAMMING OPERATION

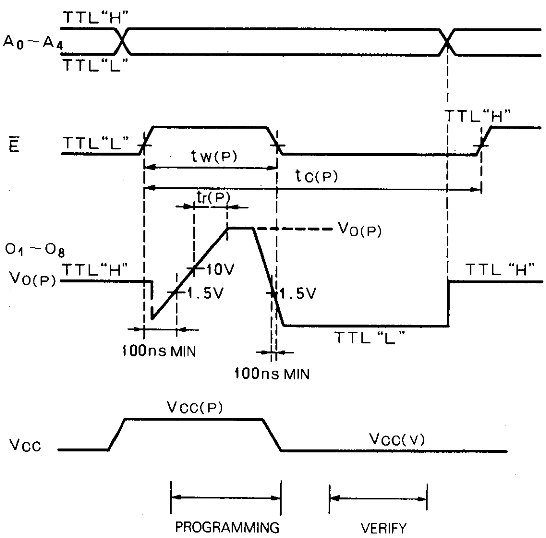
Recommended Operating Conditions

Symbol	Test conditions	Limits			Unit
		Min	Nom	Max	
$V_{O(P)}$	Output apply voltage	20		25	V
$V_{CC(P)}$	Program input voltage	5.40	5.50	5.60	V
$V_{CC(V)}$	Program verify input voltage	4.10	4.20	4.30	V

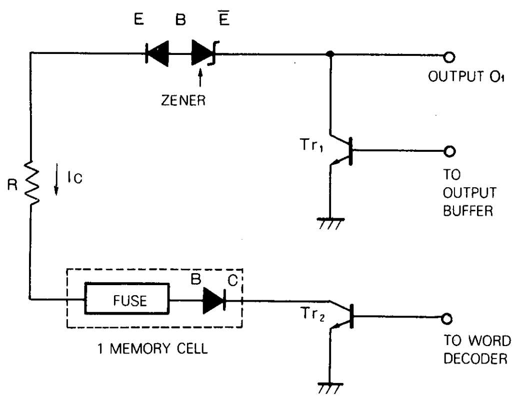
Timing Requirements

Symbol	Test conditions	Limits			Unit
		Min	Typ	Max	
$t_{r(P)}$	Pulse rise time	10	25	100	μs
$t_{w(P)}$	Pulse width	0.04		100	ms
$t_{w(P)}/t_{c(P)}$	Duty cycle			25	%

Timing Diagram



Programming Circuit



Programming (Writing) Procedure

All 256 Ni-Cr fuse-link memory elements are manufactured in a high-logic-level (fuse closed) output conditon. To program:

1. Apply 5.5V to the supply voltage V_{CC} and select a fuse link to be programmed with address inputs $A_0 \sim A_4$.
2. Apply a high-logic-level to the chip enable input $\overline{E}$.
3. After applying a program pulse $V_{I(CEP)}$ to the chip enable input $\overline{E}$ (see Timing Diagram), apply an output pulse $V_{O(P)}$ to the fuse link of the output to be programmed. The output pulses should be separately applied to each output.
4. After programming, the fuse link is open and the output level is changed to a low-logic-level.
5. After programming is completed, apply an additional three programming pulses.
6. Test the programmed memory to verify that the outputs are low-level or high-level as desired. Chip enable input $\overline{E}$ must be low-level for testing.

As the chip enable input $\overline{E}$ is kept high-level during programming, transistor Tr_1 maintains the off state. The word decoder circuit selects any one of 32 words, and sets the transistor Tr_1 to the on state. The collector current of the transistor Tr_2, which is supplied from the selected output O_1, opens the fuse links. At this time, the other seven fuse links of the selected word line are in a half-selected state and the other 248 fuse links are in a nonselected state.

Typical Programming Conditions

Condition sequence	Pulse sequence	Pulse width $t_{w(P)}$ (ms)	Chip enable program voltage $V_{I(CEP)}$ (V)	Output voltage (V)
1	1 ~ 4	0.5	29	25
2	5 ~ 8	1	29	25
3	9 ~ 12	5	30	25
4	13 ~ 19	20	33	25

256-BIT (32-WORD BY 8-BIT) FIELD-PROGRAMMABLE ROM WITH OPEN COLLECTOR OUTPUTS

APPLICATIONS

Chip Enable Circuit

The chip enable input $\overline{E}$ is used for activating or inhibiting output $O_1 \sim O_8$. Chip enable $\overline{E}$ allows easy memory expansion by one of the following procedures:

1. Expanding the Number of Bits in a Word

For example, using three 256-bit ROMs, each organized as 32 words of 8 bits, the number of bits in a word can be expanded as described below:

1. Apply a low-logic-level to chip enable input $\overline{E}$ of each ROM.
2. Connect address inputs $A_0 \sim A_4$ of each ROM in parallel. Memory is thus expanded and reorganized as 32 words of 24 bits.

2. Expanding the Number of Words in Memory

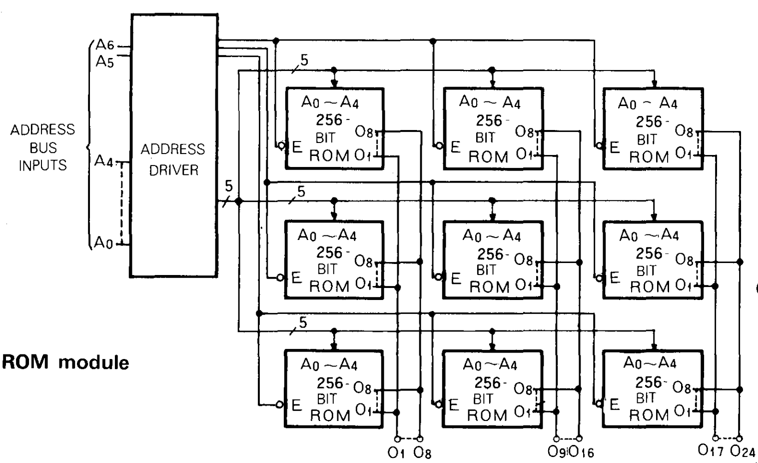
For example, using three 256-bit ROMs, each organized as 32 words of 8 bits, the number of words in memory can be expanded as described below:

1. Connect the chip enable input $\overline{E}$ of each ROM to the decoder.
2. Connect the outputs from each ROM with AND-tie connections.
3. Connect each address input $A_0 \sim A_4$ commonly. Memory is thus expanded and organized as 96 words of 4 bits.

3. Expanding the Number of Words in Memory and the Number of Bits in a Word

For example, using nine 256-bit ROMs, each organized as 32 words of 8 bits, and by combining procedures 1 and 2 above, the number of words in memory along with the number of bits in a word, can be expanded as shown in the diagram below.

Memory is thus expanded and reorganized as 96 words of 24 bits.



ROM module

Pull-up Resistors

The outputs are open collectors; therefore, AND-tie connections are also possible, and normal loads can be connected. The resistance of a pull-up resistor R_L that may be connected between the voltage supply and the collectors of the output transistors should be determined by equations (1) and (2) as shown below:

$$R_{L\,(max)} = \frac{\overline{V_{CC}} - \underline{V_{OH}}}{M \cdot \overline{I_{OH}^{*}} + N \cdot \overline{I_{IH}}} \quad \cdots\cdots\cdots (1)$$

where
- M : number of AND-ties
- N : number of fanouts (number of loads)
- $\overline{V_{CC}}$: maximum value of supply voltage
- $\underline{V_{OH}}$: minimum value of high-level output voltage
- $\overline{I_{OH}^{*}}$: maximum value of high-level output current at the open collector output
- $\overline{I_{IH}}$: maximum value of high-level input current

$$R_{L\,(min)} = \frac{\underline{V_{CC}} - \overline{V_{OL}}}{\overline{I_{OL}} - N \cdot |\overline{I_{IL}}|} \quad \cdots\cdots\cdots (2)$$

where
- $\underline{V_{CC}}$: minimum value of supply voltage
- $\overline{V_{OL}}$: maximum value of low-level output voltage
- $\overline{I_{OL}}$: maximum value of low-level output current
- $\overline{I_{IL}}$: maximum value of low-level input current

then

$$R_{L\,(min)} < R_L < R_{L\,(max)} \quad \cdots\cdots\cdots (3)$$

The resistance of a pull-up resistor R_L should be within the range as shown in equation (3). R_L (min) and R_L (max) should be calculated using the appropriate number of AND-ties and fanouts. Calculation examples of TTL load are shown below:

(1) When

$M = 4$, $N = 3$, $\overline{V_{CC}} = 5.25V$,
$\underline{V_{OH}} = 2.4V$, $\overline{I_{OH}^{*}} = 100\mu A$,
$\overline{I_{IH}} = 40\mu A$

$$R_{L\,(max)} = \frac{\overline{V_{CC}} - \underline{V_{OH}}}{M \cdot \overline{I_{OH}^{*}} + N \cdot \overline{I_{IH}}} = \frac{5.25V - 2.4V}{4 \times (100\mu A) + 3 \times (40\mu A)} = 5090\Omega$$

(2) When

$N = 3$, $\underline{V_{CC}} = 4.75V$,
$\overline{V_{OL}} = 0.45V$, $\overline{I_{OL}} = 16mA$,
$\overline{I_{IL}} = 1.6mA$

$$R_{L\,(min)} = \frac{\underline{V_{CC}} - \overline{V_{OL}}}{\overline{I_{OL}} - N \cdot |\overline{I_{IL}}|} = \frac{4.75V - 0.45V}{16mA - 3 \times (1.6mA)} = 384\Omega$$

4096-BIT(1024-WORD BY 4-BIT)FIELD PROGRAMMABLE ROM

DESCRIPTION

The M54740AP,S'(provided with open collector outputs) and the M54741AP, S (provided with 3-state outputs) are field programmable ROMs with a fusecoupling system and a memory capacity of 4096 bits (1024 words x 4-bit configuration).

FEATURES

- Fast·address access time .... 25ns (typ)
- Unique built-in test circuit guarantees a high programming yield as well as various performance characteristics after programming
- Fuse technology is used
- Memory capacity of 4096 bits (1024 words x 4-bit organization)
- Open collector outputs for M54740AP,S;
  3-state outputs for M54741AP,S
- High output level before programming
- Chip enable pins $\overline{E_1}$ and $\overline{E_2}$ provided for easy expansion of memory capacity
- Inputs and outputs compatible with TTL system
- 18-pin DIL ceramic or plastic package

APPLICATION

General purpose, for use in industrial and consumer equipment.

FUNCTIONAL DESCRIPTION

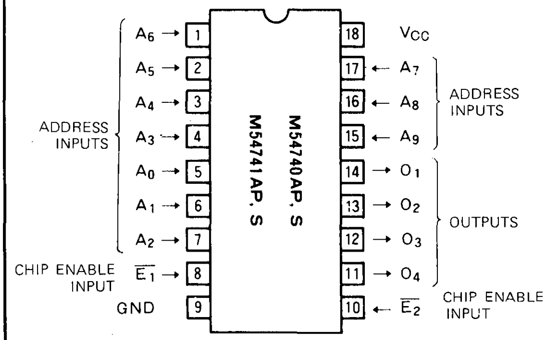
The PROM consists of an address circuit, decoder circuit, memory circuit, output circuit and chip enable circuit, and the memory cells consist of fuses and diodes. Data can be programmabled into the PROM by the user using a writer by cutting the fuses of the memory cells. The output level is high before programming and low when written into.

The 4096 memory cells have a capacity of 1024 words and one word is composed of 4 bits. A word is selected from the 1024 words by address inputs $A_0 \sim A_9$, and 4-bit parallel outputs $O_1 \sim O_4$ are produced.

The input and output threshold voltage is the same as that for a TTL system and thus direct coupling can be made with TTL logic. The open-collector outputs (in the M54740AP,S) or 3-state outputs (in the M54741AP,S) enable AND-tie connection.

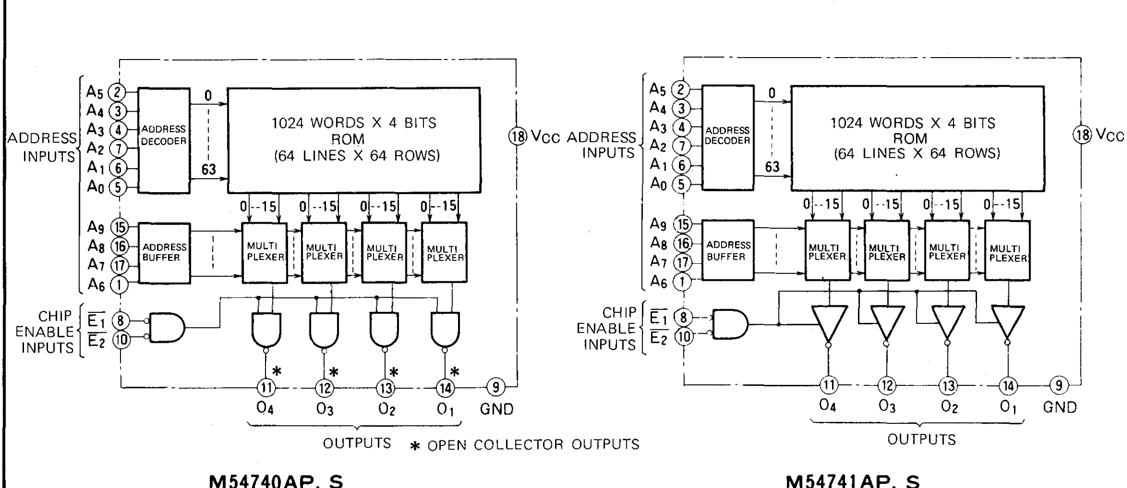
When both chips enable inputs $\overline{E_1}$ and $\overline{E_2}$ are low, the output is enabled and the contents of the memory selected

PIN CONFIGURATION (TOP VIEW)



Outline 18S1 (M54740AS, M54741AS)
18P4 (M54740AP, M54741AP)

BLOCK DIAGRAM



M54740AP, S　　　M54741AP, S

MITSUBISHI ELECTRIC

by the address input appear in the outputs. When either chip enable input $\overline{E_1}$ or $\overline{E_2}$ is high, the output is disabled, and regardless of the address input, the output is set high (open-collector) or put in the high-impedance mode (3-state).

READ FUNCTION TABLE (Note 1)

Read function table for M54740AP,S

$\overline{E_1}$	$\overline{E_2}$	$O_1 \sim O_4$
L	L	Wn
H	L	H
L	H	H
H	H	H

Read function table for M54741AP,S

$\overline{E_1}$	$\overline{E_2}$	$O_1 \sim O_4$
L	L	Wn
H	L	Z
L	H	Z
H	H	Z

Note 1. Wn: Memory contents written in Wn word appear in output.
Z: High-impedance state

3

ABSOLUTE MAXIMUM RATINGS (T_a=25°C, unless otherwise noted)

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage		−0.5 ~ +7	V
V_I	Input voltage		−0.5 ~ +5.5	V
V_O	Output voltage	High-level state	−0.5 ~ +5.5	V
V_{OP}	Applied output voltage	When writing	21	V
$t_{W(P)}/t_{C(P)}$	Duty cycle	When writing	25	%
Topr	Operating free-air ambient temperature range		0 ~ +75	°C
Tstg	Storage temperature range		−65 ~ +150	°C

RECOMMENDED OPERATING CONDITIONS (T_a=0~+75°C, unless otherwise noted)

Symbol	Parameter	Limits Min	Nom	Max	Unit
V_{CC}	Supply voltage	4.75	5	5.25	V
I_{OH}	High-level output current (M54741AP/S only) $V_{OH} \geqq 2.4$	0		−2	mA
I_{OH}	High-level output current (M54740AP/S only) V_O=5V	0		50	μA
I_{OL}	Low-level output current $V_{OL} \leqq 0.45$V	0		16	mA

ELECTRICAL CHARACTERISTICS (T_a=−20~+75°C, unless otherwise noted)

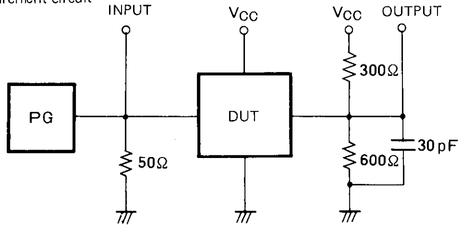
Symbol	Parameter	Test conditions	Limits Min	Typ *	Max	Unit
V_{IH}	High-level input voltage		2			V
V_{IL}	Low-level input voltage				0.8	V
V_{IC}	Input clamp voltage	V_{CC}=4.75V, I_{IC}=−18mA			−1.2	V
V_{OH}	High-level output voltage (M54741AP,S)	V_{CC}=4.75V, V_I=2V, V_I=0.8V, I_{OH}=−2mA	2.4	3.1		V
I_{OH}	High-level output current (M54740AP,S)	V_{CC}=5.25V, V_I=2V, V_I=0.8V, V_O=5V			50	μA
V_{OL}	Low-level output voltage	V_{CC}=4.75V, V_I=2V, V_I=0.8V, I_{OL}=16mA		0.3	0.45	V
I_{OZH}	Off-state high-level output current (M54741AP,S)	V_{CC}=5.25V, V_I=0.8V, V_I=2V, V_O=2.4V			50	μA
I_{OZL}	Off-state high-level output current (M54740AP,S)	V_{CC}=5.25V, V_I=0.8V, V_I=2V, V_O=0.4V			−50	μA
I_{IH}	High-level input current	V_{CC}=5.25V, V_I=2.4V			40	μA
I_{IL}	Low-level input current	V_{CC}=5.25V, V_I=0.4V		−160	−250	μA
I_{OS}	Short-circuit output current (M54741AP,S) (Note 2)	V_{CC}=5.25V, V_O=0V	−15		−100	mA
I_{CC}	Supply current (Note 3)	V_{CC}=5.25V, V_I=0V		120	170	mA
C_{IN}	Input capacitance	V_{CC}=5V, V_I=2V, f=1MHz		4		pF
C_{OUT}	Output capacitance	V_{CC}=5V, V_O=2V, f=1MHz		7		pF

* All typical values are at V_{CC}=5V, T_a=25°C.
Note 2. All measurements should be done quickly and not more than one output should be shorted at a time.
Note 3. I_{CC} is measured with all inputs at GND .

SWITCHING CHARACTERISTICS (Vcc=5V±5%, Ta=0~75°C, unless otherwise noted)

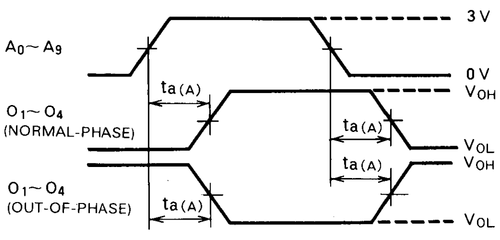
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
ta (A)	Address access time	(Note 4)		25	55	ns
ta (Ē)	Chip enable access time			15	25	ns
tCD (Ē)	Chip disable time			15	25	ns

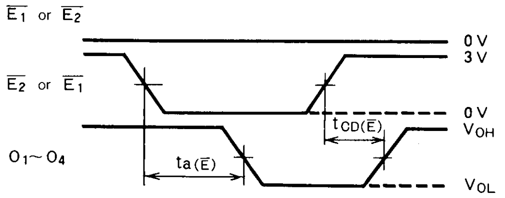
Note 4. Measurement circuit



(1) The pulse generator (PG) has the following characteristics: PRR=1MHz, t_r=6ns, t_f=6ns, t_{pw}=500ns, V_P=3V_{P-P}, Z_o=50Ω.
(2) C_L includes probe and jig capacitance.

TIMING DIAGRAM (Reference level=1.5V)





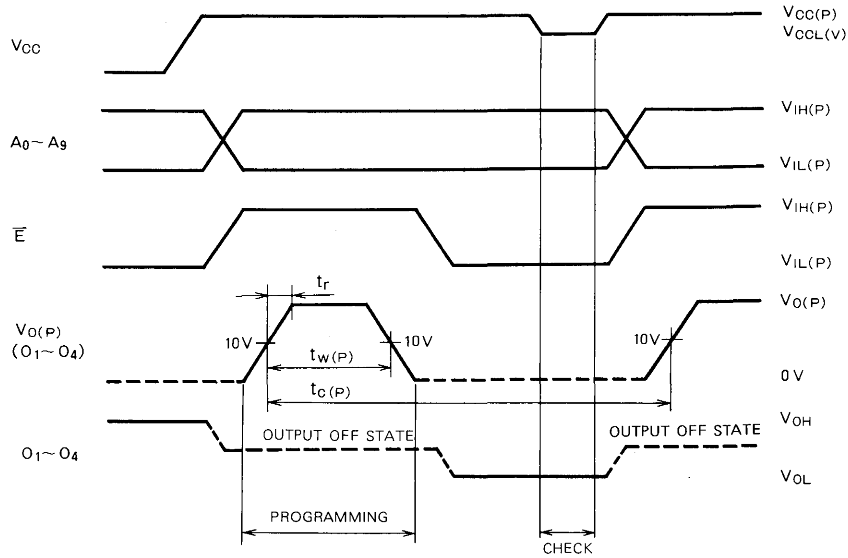
RECOMMENDED WRITE CONDITIONS (Ta=25°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Typ	Max	
VIH (P)	High-level input voltage	2.4	5	5	V
VIL (P)	Low-level input voltage	0	0	0.4	V
VO (P)	Applied output voltage	20	21	21	V
tw (P)	Applied pulse width	0.05	0.18	50	ms
tw (P)/tC (P)	Duty cycle		20	25	%
tr	Pulse risetime	5	10	30	μs
N (P)	Number of pulses applied		4		—
VCC (P)	Write supply voltage		5		V
IOP	Applied output current			100	mA
VCCL (V)	Low-level supply voltage with check after writing		4.4		V

MITSUBISHI ELECTRIC

4096-BIT(1024-WORD BY 4-BIT)FIELD PROGRAMMABLE ROM

PROGRAMMING TIMING DIAGRAM



Note 5. The $V_{O(P)}$ waveform is the voltage waveform applied to the output during programming; the O_1~O_4 waveforms indicate the output level of the device itself.
Note 6. Waveform $\overline{E}$ indicates either the $\overline{E_1}$ or $\overline{E_2}$ waveform; the other is $V_{IL(P)}$.

PROGRAMMING PROCEDURE

The area into which the data are programmed is the fuses which are composed of 4096 memory cells. When no data are programmed into a memory cell, the output is set to the logic high level (fuse: closed). Proceed as instructed below to set to the logic low level (fuse: open).

(1) Apply the supply voltage $V_{CC(P)}$ (5V typ).
(2) Select the programming word with the address inputs A_0~A_9 (input voltage: $V_{IH(P)}$ 5V typ, $V_{IL(P)}$ 0V typ).
(3) Set either chip enable input pin $\overline{E_1}$ or $\overline{E_2}$ high ($V_{IH(P)}$ 5V typ) and set the outputs off.
(4) Apply the output pulse $V_{O(P)}$ (21V typ) to the output which corresponds to the bit into which the data are to be programmed. Apply this to one output at a time and not to two or more outputs simultaneously.
(5) Set both $\overline{E_1}$ and $\overline{E_2}$ low ($V_{IL(P)}$ 0V typ).
(6) Reduce the supply voltage to $V_{CCL(V)}$ (4.4V typ) and check whether the data has been programmed.
(7) If the check is affirmative in step (6), repeat steps (1) through (6) to program the next word or bit.

If the check is negative in step (6), repeat steps (1) through(6) but if the check is still negative even after four repetitions, the device may be considered defective.

Refer to the programming timing diagram for the timing of the programming operation.

700-BIT (50-WORD BY 14-BIT) ELECTRICALLY ALTERABLE ROM

DESCRIPTION

The M58653P is a serial input/output 700 bit electrically erasable and reprogrammable ROM organized as 50 words of 14 bits, and fabricated using MNOS technology. Data and addresses are transferred serially via a one-bit bidirectional bus.

FEATURES

- Word-by-word electrically alterable
- Non-volatile data storage: 10 years (min)
- Write/erase time: 20ms/word
- Typical power supply voltages: -30V, +5V
- Number of erase-write cycles: 10^5 times (min)
- Number of read access unrefreshed: 10^9 times (min)
- 5V I/O interface

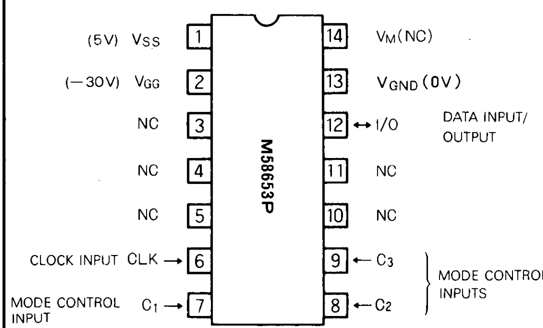
APPLICATION

- Non-volatile channel memories for electronic tuning systems and field-reprogrammable read-only memory systems

FUNCTION

The address is designated by two consecutive one-of-ten-coded digits. Seven modes—accept address, accept data, shift data output, erase, write, read, and standby—are all selected by a 3-bit code applied to C_1, C_2, and C_3. Data is stored by internal negative writing pulses that selectively tunnel charges into the SiO_2—Si_3N_4 interface of the gate insulators of the MNOS memory transistors.

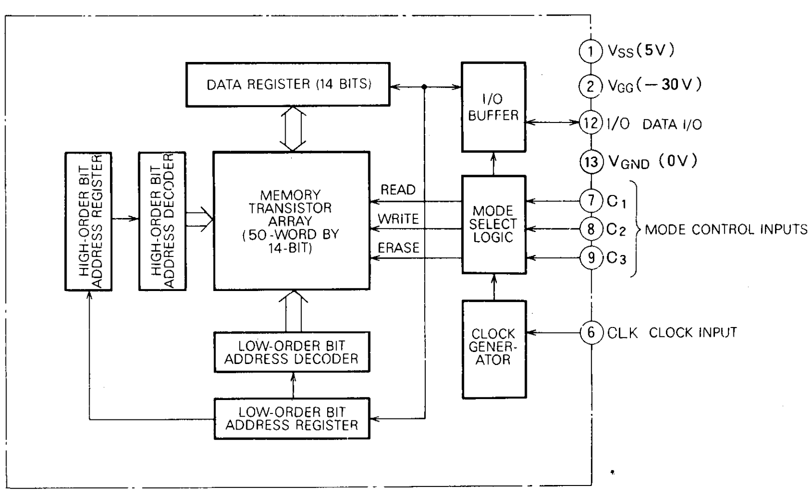
PIN CONFIGURATION (TOP VIEW)



Outline 14P4

NC: NO CONNECTION

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

700-BIT (50-WORD BY 14-BIT) ELECTRICALLY ALTERABLE ROM

PIN DESCRIPTION

Pin	Name	Functions
I/O	I/O	In the accept address and accept data modes, used for input. In the shift data output mode, used for output. In the standby, read, erase and write modes, this pin is in a floating state.
V_M	Test	Used for testing purposes only. It should be left unconnected during normal operation.
V_{SS}	Chip substrate voltage	Normally connected to +5V.
V_{GG}	Power supply voltage	Normally connected to −30V.
CLK	Clock input	14kHz timing reference. Required for all operating modes. High-level input is possible during standby mode.
$C_1 \sim C_3$	Mode control input	Used to select the operation mode.
V_{GND}	Ground voltage	Connected to ground (OV)

OPERATION MODES

C_1	C_2	C_3	Functions
H	H	H	Standby mode: The contents of the address registers and the data register remain unchanged. The output buffer is held in the floating state.
H	H	L	Not used.
H	L	H	Erase mode: The word stored at the addressed location is erased. The data bits after erasing are all low-level.
H	L	L	Accept address mode: Data presented at the I/O pin is shifted into the address registers one bit with each clock pulse. The address is designated by two one-of-ten-coded digits.
L	H	H	Read mode: The addressed word is read from the memory into the data register.
L	H	L	Shift data output mode: The output driver is enabled and the contents of the data register are shifted to the I/O pin one bit with each clock pulse.
L	L	H	Write mode: The data contained in the data register is written into the location designated by the address registers.
L	L	L	Accept data mode: The data register accepts serial data from the I/O pin one bit with each clock pulse. The address registers remain unchanged.

700-BIT (50-WORD BY 14-BIT) ELECTRICALLY ALTERABLE ROM

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{GG}	Supply voltage	With respect to V_{SS}	0.3 ~ −40	V
V_I	Input voltage		0.3 ~ −20	V
V_O	Output voltage		0.3 ~ −20	V
Tstg	Storage temperature range		−40 ~ 125	℃
Topr	Operating free-air temperature range		−10 ~ 70	℃

RECOMMENDED OPERATING CONDITIONS (Ta = −10 ~ 70℃, unless otherwise noted.)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{GG}-V_{SS}	Supply voltage	−32.2	−35	−37.8	V
V_{SS}-V_{GND}	Supply voltage	4.75	5	6	V
V_{IH}	High-level input voltage	$V_{SS}-1$		$V_{SS}+0.3$	V
V_{IL}	Low-level input voltage	V_{SS}-6.5		V_{SS}-4.25	V

ELECTRICAL CHARACTERISTICS (Ta = −10 ~ 70℃, V_{GG}-V_{SS} = −35V ± 8%, V_{SS}-V_{GND} = 5V $^{+20\%}_{-5\%}$, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{IH}	High-level input voltage		$V_{SS}-1$		$V_{SS}+0.3$	V
V_{IL}	Low-level input voltage		V_{SS}-6.5		V_{SS}-4.25	V
I_{IL}	Low-level input current	V_I-V_{SS} = −6.5V			±10	μA
I_{OZL}	Off-state output current, low-level voltage applied	V_O-V_{SS} = −6.5V			±10	μA
V_{OH}	High-level output voltage	I_{OH} = −200μA	$V_{SS}-1$			V
V_{OL}	Low-level output voltage	I_{OL} = 10μA			$V_{GND}+0.5$	V
I_{GG}	Supply current from V_{GG}	I_O = 0μA		5.5	8.8	mA

Note 1 : Typical values are at Ta=25°C and nominal supply voltage.

TIMING REQUIREMENTS (Ta = −10 ~ 70℃, V_{GG}-V_{SS} = −35V ± 8%, V_{SS}-V_{GND} = 5V $^{+20\%}_{-5\%}$, unless otherwise noted.)

Symbol	Parameter	Alternative symbols	Test conditions	Limits			Unit
				Min	Typ	Max	
f(φ)	Clock frequency	fφ		11.2	14	16.8	kHz
D(φ)	Clock duty cycle	Dφ		30	50	55	%
tw(w)	Write time	tw		16	20	24	ms
tw(E)	Erase time	te		16	20	24	ms
tr, tf	Risetime, falltime	tr, tf				1	μs
tsu(c−φ)	Control setup time before the fall of the clock pulse	t_{CS}		0			ns
th(φ−c)	Control hold time after the rise of the clock pulse	t_{CH}		0			ns

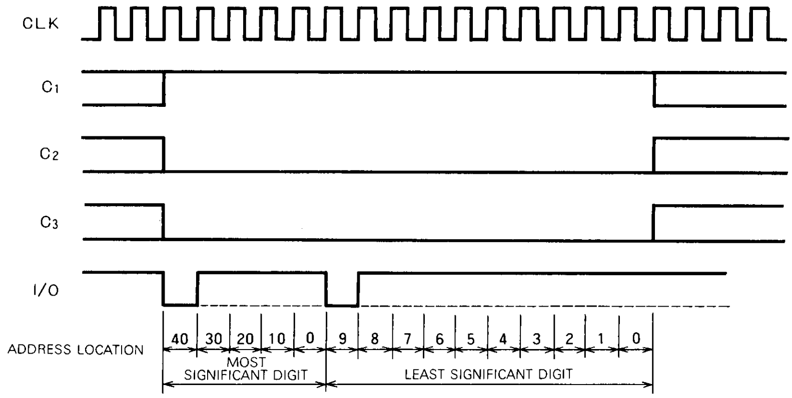
SWITCHING CHARACTERISTICS (Ta = −10 ~ 70℃, V_{GG} = −35V ± 8%, unless otherwise noted.)

Symbol	Parameter	Alternative symbols	Test conditions	Limits			Unit
				Min	Typ	Max	
ta(c)	Read access time	t_{PW}	C_L = 100PF, $V_{OH} = V_{SS} - 2V$, $V_{OL} = V_{GND} + 1.5V$			20	μs
ts	Unpowered nonvolatile data retention time	T_S	$N_{EW} = 10^4$, tw(w) = 20ms, tw(E) = 20ms	10			Year
		T_S	$N_{EW} = 10^5$, tw(w) = 20ms, tw(E) = 20ms	1			Year
N_{EW}	Number of erase/write cycles	N_W		10^5			Times
N_{RA}	Number of read access unrefreshed	N_{RA}		10^9			Times
tdv	Data valid time	t_{PW}				20	μs

700-BIT (50-WORD BY 14-BIT) ELECTRICALLY ALTERABLE ROM

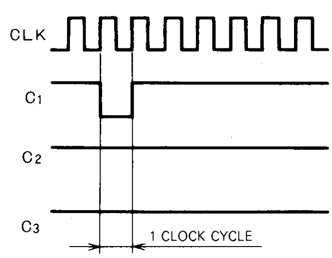
TIMING DIAGRAM

Accept Address Mode

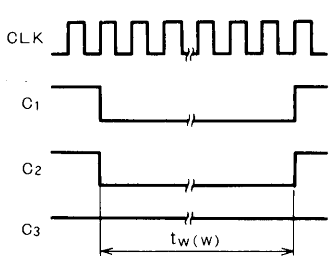


Note 2 : The address is designated by two one-of-ten-coded digits. The figure shows designation of the address 49.

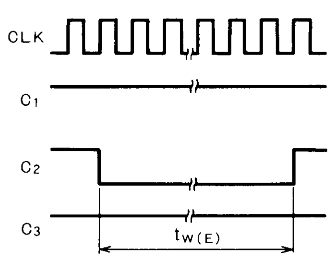
Read Mode



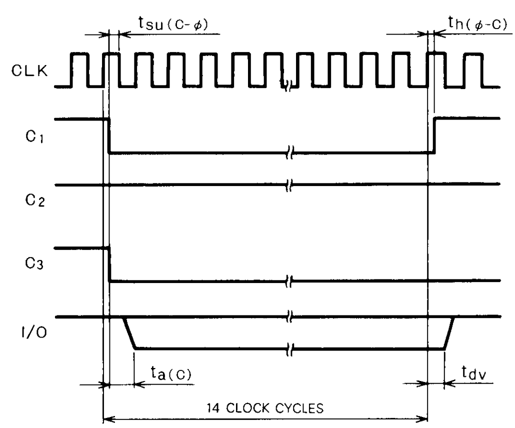
Write Mode



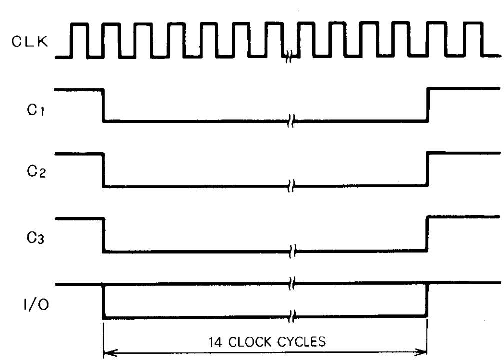
Erase Mode



Shift Data Output Mode



Accept Data Mode



32768-BIT (4096-WORD BY 8-BIT) MASK-PROGRAMMABLE ROM

DESCRIPTION

The M58735-XXXP is a 32768-bit static mask-programmable read-only memory organized as 4096 words of eight bits. It is housed in a 24-pin DIL package using N-channel silicon gate MOS technology. The inputs and outputs are TTL compatible.

The XXX in the type code is a three-digit decimal number assigned by Mitsubishi to identify the customer's specification to which the ROM has been programmed.

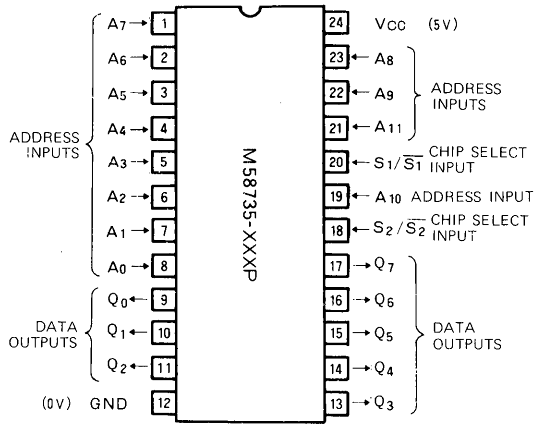
FEATURES

- Maximum access time: 350ns (max)
- 8-bit parallel output
- By floating the output (high-impedance) using the chip select inputs (S_1, S_2), OR-tie connection is possible, facilitating memory expansion.
- The active logic level of the chip select inputs (S_1 and S_2) can be programmed at the time of ROM masking
- All inputs and outputs are TTL and DTL compatible
- All inputs are provided with built-in protective circuits
- Pin-compatibility with the M5L2732K

APPLICATION

- Microcomputer memories

PIN CONFIGURATION (TOP VIEW)

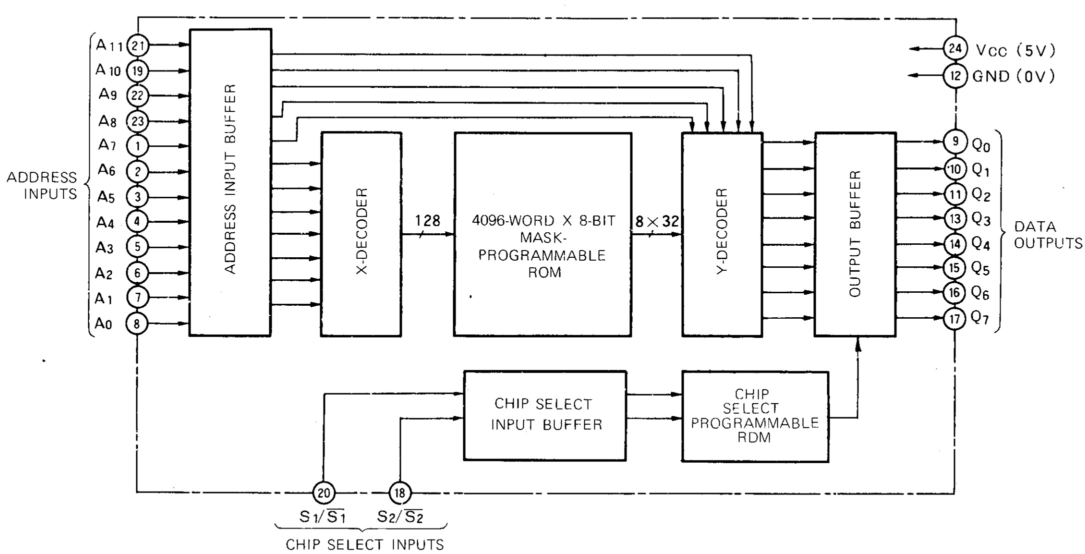


Outline 24P4

FUNCTION

The M58735-XXXP is a 4096 x 8-bit parallel output ROM. Address inputs ($A_0 \sim A_{11}$) are decoded to select one of the 4096 words, and the contents of that address are made available at data outputs ($Q_0 \sim Q_7$). Chip selects (S_1 and S_2) are used to expand memory using two or more M58735-XXXP ROMs. The contents of the ROM can be read only when S_1 and S_2 are at the programmed input levels. Otherwise, data outputs ($Q_0 \sim Q_7$) are held in the floating (high-impedance) state. The active logic level of S_1 and S_2 can be programmed at the time of fabricating the ROM mask.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

32768-BIT(4096-WORD BY 8-BIT)MASK-PROGRAMMABLE ROM

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With reference to the GND (with V_I and V_O at V_{CC}=5V)	−0.5~7	V
V_I	Input voltage		−0.5~7	V
V_O	Output voltage		−0.5~7	V
Pd	Power dissipation	Ta=25°C	1000	mW
Topr	Operating temperature		0~70	°C
Tstg	Storage temperature		−65~150	°C

RECOMMENDED OPERATING CONDITIONS (Ta=0~70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
GND			0		V
V_{IH}	High-level input voltage	2		V_{CC}+1	V
V_{IL}	Low-level input voltage	−0.5		0.8	V

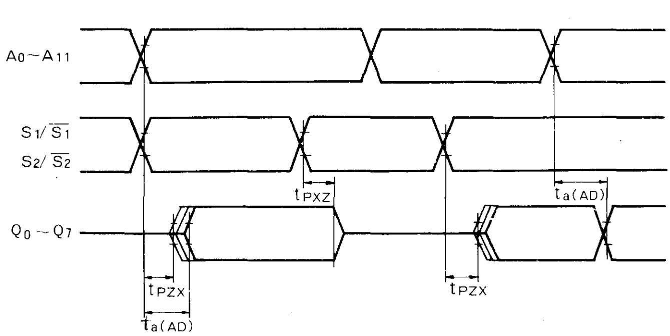
ELECTRICAL CHARACTERISTICS (Ta=0~70°C, V_{CC}=5V±10%, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	I_{OH}=−400 μA	2.4			V
V_{OL}	Low-level output voltage	I_{OL}=2.2mA			0.45	V
I_I	Input current	V_I=0~V_{CC}	−10		10	μA
I_{OZ}	Off-state output current	V_O=0.45~V_{CC}	−10		10	μA
I_{CC}	Supply current from V_{CC}	Output open, Ta=25°C		80	120	mA
Ci	Input capacitance	V_{CC}=5V, V_I=V_O=0V, f=1MHz, 25mVrms, Ta=25°C			10	pF
Co	Output capacitance				15	pF

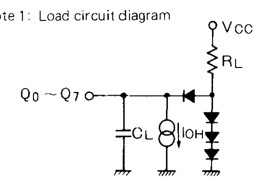
SWITCHING CHARACTERISTICS (Ta=0~70°C, V_{CC}=5V±10%, unless otherwise noted)

Symbol	Parameter	Test conditions (Note 2)	Limits			Unit
			Min	Typ	Max	
$t_{a(AD)}$	Access time from Address	C_L=100pF, R_L=2.1kΩ (Note 1)			350	ns
t_{PZX}	Chip select propagation time				120	ns
t_{PXZ}	Chip non-select propagation time		0		150	ns

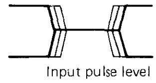
TIMING DIAGRAM



Note 1: Load circuit diagram



Note 2:



The center line indicates a floating (high-impedance) state

Input pulse level .... 0.45~2.4V
Input pulse risetime t_r 20ns
Input pulse falltime t_f 20ns
Reference voltage for switching characteristic measurements
Input V_{IH} 2V
V_{IL} 0.8V
Output V_{OH} 2V
V_{OL} 0.8V

1400-BIT (100-WORD BY 14-BIT) ELECTRICALLY ALTERABLE ROM

DESCRIPTION

The M5G1400P is a serial input/output 1400-bit electrically erasable and reprogrammable ROM organized as 100 words of 14 bits, and fabricated using MNOS technology. Data and addresses are transferred serially via a one-bit bidirectional bus.

FEATURES

- Word-by-word electrically alterable
- Non-volatile data storage: ·················· 10 years (min)
- Write/erase time: ································ 20ms/word
- Single 35V power supply
- Number of erase-write cycles: ············ 10^5 times (min)
- Number of read access unrefreshed:···· 10^6 times (min)
- Interchangeable with GI's ER1400 in pin configuration and electrical characteristics

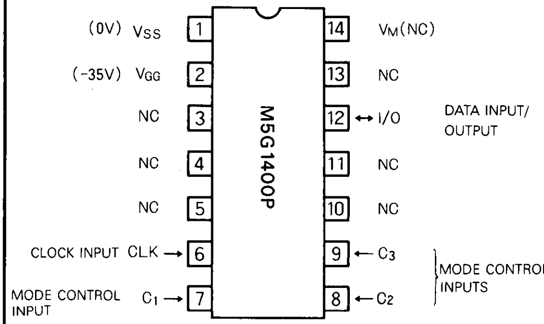
APPLICATION

- Non-volatile channel memories for electronic tuning systems and field-reprogrammable read-only memory systems

FUNCTION

The address is designated by two consecutive one-of-ten-coded digits. Seven modes—accept address, accept data, shift data output, erase, write, read, and standby—are all selected by a 3-bit code applied to C_1, C_2, and C_3. Data is stored by internal negative writing pulses that selectively tunnel charges into the SiO_2—Si_3N_4 interface of the gate insulators of the MNOS memory transistors.

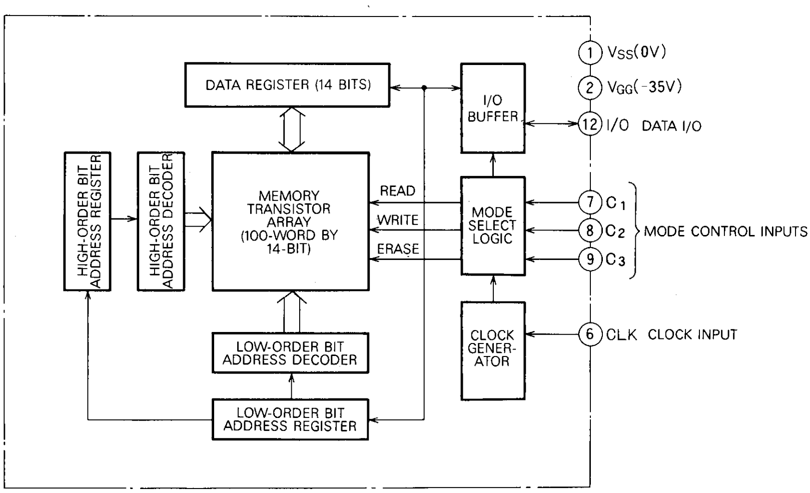
PIN CONFIGURATION (TOP VIEW)



Outline 14P4

NC: NO CONNECTION

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

1400-BIT (100-WORD BY 14-BIT) ELECTRICALLY ALTERABLE ROM

PIN DESCRIPTION

Pin	Name	Functions
I/O	I/O	In the accept address and accept data modes, used for input. In the shift data output mode, used for output. In the standby, read, erase and write modes, this pin is in a floating state.
V_M	Test	Used for testing purposes only. It should be left unconnected during normal operation.
V_{SS}	Chip substrate voltage	Normally connected to ground.
V_{GG}	Power supply voltage	Normally connected to −35V.
CLK	Clock input	14kHz timing reference. Required for all operating modes. High-level input is possible during standby mode.
$C_1 \sim C_3$	Mode control input	Used to select the operation mode.

OPERATION MODES

C_1	C_2	C_3	Functions
H	H	H	Standby mode: The contents of the address registers and the data register remain unchanged. The output buffer is held in the floating state.
H	H	L	Not used.
H	L	H	Erase mode: The word stored at the addressed location is erased. The data bits after erasing are all low-level.
H	L	L	Accept address mode: Data presented at the I/O pin is shifted into the address registers one bit with each clock pulse. The address is designated by two one-of-ten-coded digits.
L	H	H	Read mode: The addressed word is read from the memory into the data register.
L	H	L	Shift data output mode: The output driver is enabled and the contents of the data register are shifted to the I/O pin one bit with each clock pulse.
L	L	H	Write mode: The data contained in the data register is written into the location designated by the address registers.
L	L	L	Accept data mode: The data register accepts serial data from the I/O pin one bit with each clock pulse. The address registers remain unchanged.

1400-BIT (100-WORD BY 14-BIT) ELECTRICALLY ALTERABLE ROM

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{GG}	Supply voltage	With respect to V_{SS}	0.3~−40	V
V_I	Input voltage		0.3~−20	V
V_O	Output voltage		0.3~−20	V
Tstg	Storage temperature range		−65~150	℃
Topr	Operating free-air temperature range		−10~70	℃

RECOMMENDED OPERATING CONDITIONS (Ta=−10~70℃, unless otherwise noted.)

Symbol	Parameter	Limits Min	Limits Nom	Limits Max	Unit
V_{GG}	Supply voltage	−32.2	−35	−37.8	V
V_{SS}	Supply voltage (GND)		0		V
V_{IH}	High-level input voltage	$V_{SS}-1$		$V_{SS}+0.3$	V
V_{IL}	Low-level input voltage	$V_{SS}-15$		$V_{SS}-8$	V

Note 1:
The order of V_{SS} V_{GG} with on or off.
With on, V_{GG} is turned on after V_{SS} is done.
With off, V_{SS} is turned off after V_{GG} is done.

ELECTRICAL CHARACTERISTICS (Ta=−10~70℃, V_{GG}=−35V±8%, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
V_{IH}	High-level input voltage		$V_{SS}-1$		$V_{SS}+0.3$	V
V_{IL}	Low-level input voltage		$V_{SS}-15$		$V_{SS}-8$	V
I_{IL}	Low-level input current	$V_I=-15V$			±10	μA
I_{OZL}	Off-state output current, low-level voltage applied	$V_O=-15V$			±10	μA
V_{OH}	High-level output voltage	$I_{OH}=-200\mu A$	$V_{SS}-1$			V
V_{OL}	Low-level output voltage	$I_{OL}=10\mu A$			$V_{SS}-12$	V
I_{GG}	Supply current from V_{GG}	$I_O=0\mu A$		5.5	8.8	mA

Note 2: Typical values are at Ta=25°C and nominal supply voltage.

TIMING REQUIREMENTS (Ta=−10~70℃, V_{GG}=−35V±8%, unless otherwise noted.)

Symbol	Parameter	Alternative symbols	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
f(φ)	Clock frequency	fφ		11.2	14	16.8	kHz
D(φ)	Clock duty cycle	Dφ		30	50	55	%
tw(w)	Write time	tw		16	20	24	ms
tw(E)	Erase time	te		16	20	24	ms
tr, tf	Risetime, falltime	tr, tf				1	μs
tsu(c−φ)	Control setup time before the fall of the clock pulse	t_{CS}		0			ns
th(φ−c)	Control hold time after the rise of the clock pulse	t_{CH}		0			ns

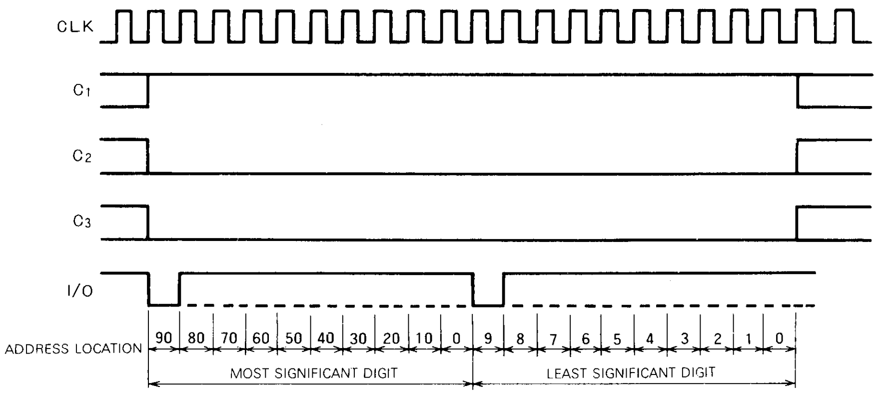
SWITCHING CHARACTERISTICS (Ta=−10~70℃, V_{GG}=−35V±8%, unless otherwise noted.)

Symbol	Parameter	Alternative symbols	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
ta(c)	Read access time	t_{PW}	C_L= $V_{OH}=V_{SS}-2V$, $V_{OL}=V_{SS}-8V$			20	μs
ts	Unpowered nonvolatile data retention time	Ts	$N_{EW}=10^4$, tw(w)=20ms, tw(E)=20ms	10			Year
		Ts	$N_{EW}=10^5$, tw(w)=20ms, tw(E)=20ms	1			Year
N_{EW}	Number of erase/write cycles	Nw		10^5			Times
N_{RA}	Number of read access unrefreshed	N_{RA}		10^6	10^9		Times
tdv	Data valid time	t_{PW}				20	μs

1400-BIT (100-WORD BY 14-BIT) ELECTRICALLY ALTERABLE ROM

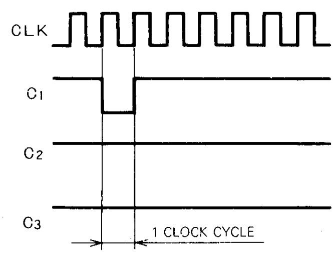
TIMING DIAGRAM

Accept Data Mode

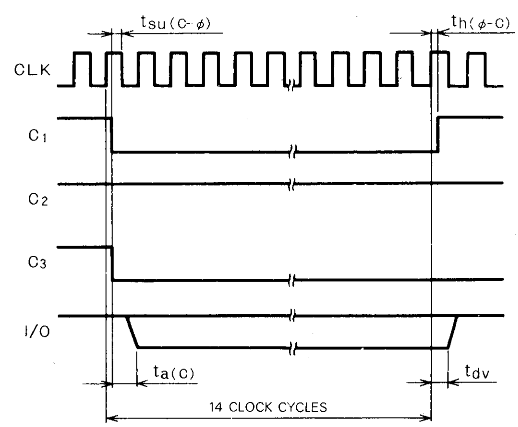


Note 3: The address is designated by two one-of-ten-coded digits. The figure shows designation of the address 99.

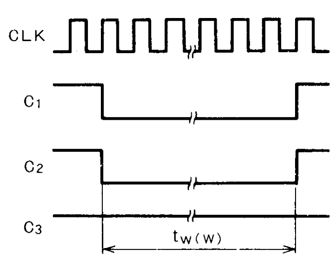
Read Mode



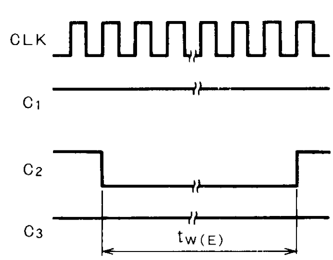
Shift Data Output Mode



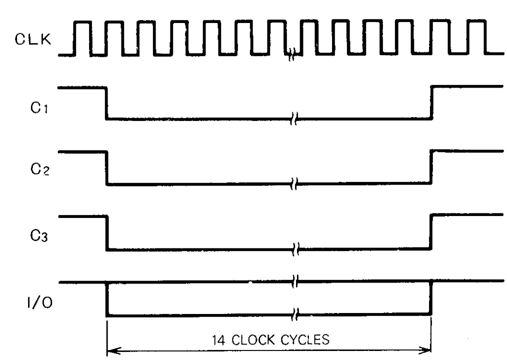
Write Mode



Erase Mode



Accept Data Mode



M5L 2716 K, K-65

16 384-BIT (2048-WORD BY 8-BIT) ERASABLE AND ELECTRICALLY REPROGRAMMABLE ROM

DESCRIPTION

These are ultraviolet-light erasable and electrically reprogrammable 16 384-bit (2048-word by 8-bit) EPROMs. They incorporate N-channel silicon-gate MOS technology, and are designed for microprocessor programming applications.

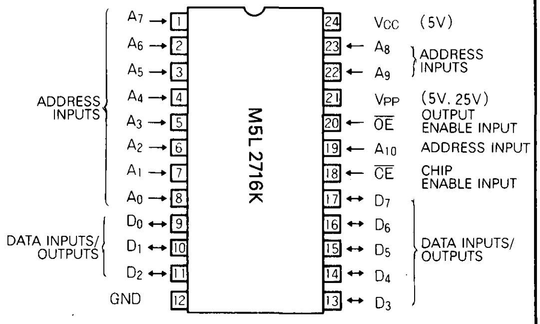
FEATURES

- Fast programming: 100s/16 384 bits (typ)
- Access time M5L2716K: 450ns (max)
  M5L2716K-65: 650ns (max)
- Static circuits are used throughout
- Inputs and outputs TTL-compatible in read and program modes
- Single 5V power supply for read mode
  (25V power supply required for program)
- Low power dissipation: Operating: 525mW (max)
  Standby: 132mW (max)
- Single-location programming
  (requires one 50ms pulse/address)
- Interchangeable with Intel's 2716 in pin configuration and electrical characteristics

APPLICATION

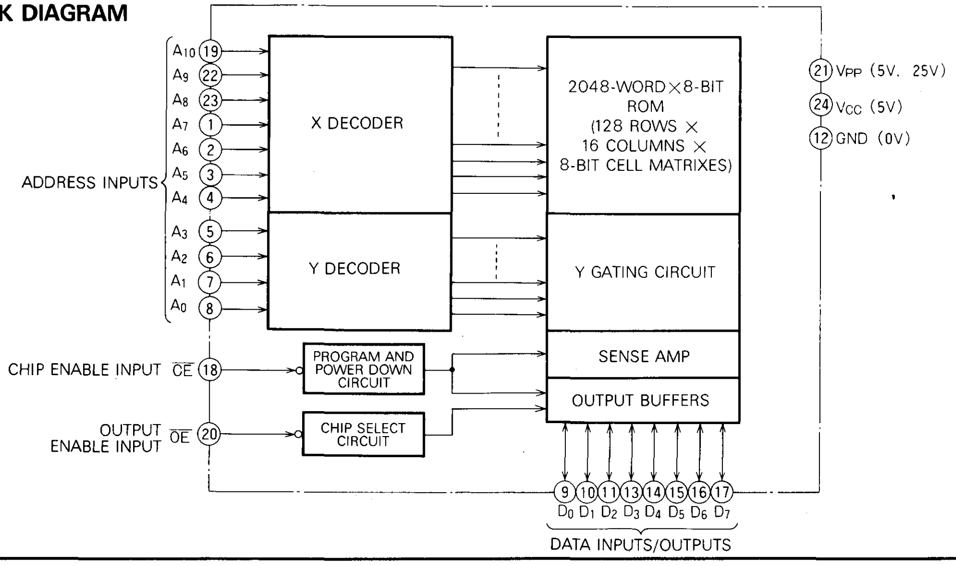
- Computers and peripheral equipment

PIN CONFIGURATION (TOP VIEW)



Outline 24K10

BLOCK DIAGRAM



**16 384-BIT (2048-WORD BY 8-BIT)
ERASABLE AND ELECTRICALLY REPROGRAMMABLE ROM**

FUNCTION

Read

Set the $\overline{CE}$ and $\overline{OE}$ terminals to the read mode (low-level). Low-level input to $\overline{CE}$ and $\overline{OE}$ and address signals to the address inputs ($A_0 \sim A_{10}$) make the data contents of the designated address location available at the data inputs/outputs ($D_0 \sim D_7$). When the $\overline{CE}$ or $\overline{OE}$ signal is high, data inputs/outputs ($D_0 \sim D_7$) are in a floating state.

When the $\overline{CE}$ signal is high, the device is in the standby mode or power-down mode.

Programming

The chip enters the programming mode when 25V is supplied to the V_{PP} power supply input and $\overline{OE}$ is at high-level. A location is designated by address signals $A_0 \sim A_{10}$, and the data to be programmed must be applied at 8 bits in parallel to the data inputs $D_0 \sim D_7$. A program pulse to the $\overline{CE}$ at this state will effect programming. Only one programming pulse is required, but its width must satisfy the condition $45ms \leqq t_{w(CE)} \leqq 55ms$.

Mode selection (Unit: V)

Mode \ Pin	$\overline{CE}$	$\overline{OE}$	V_{PP}	V_{CC}	Outputs
Read	V_{IL}	V_{IL}	5	5	Output
Deselect	$V_{IL} \sim V_{IH}$	V_{IH}	5	5	Floating
Power down	V_{IH}	$V_{IL} \sim V_{IH}$	5	5	Floating
Program	Pulsed V_{IL} to V_{IH}	V_{IH}	25	5	Input
Program verify	V_{IL}	V_{IL}	5 or 25	5	Output
Program inhibit	V_{IL}	V_{IH}	25	5	Floating

Erase

Erase is effected by exposure to ultraviolet light with a wavelength of 2537Å at an intensity of approximately $15Ws/cm^2$.

PRECAUTIONS FOR READ OPERATION

1. V_{CC} should be turned on with or before V_{PP} and turned off with or after V_{PP}.
2. V_{PP} should be connected directly to V_{CC} except during programming. For supply current design, therefore, V_{PP} and V_{CC} should be added.

HANDLING PRECAUTIONS

1. Sunlight and fluorescent light may contain ultraviolet light sufficient to erase the programmed information. For any operation in the read mode, the transparent window should be covered with opaque tape.
2. High voltages are used when programming, and the conditions under which it is performed must be carefully controlled to prevent the application of excessively high voltages. Specifically, the voltage applied to V_{PP} should be kept below 26V including overshoot. Special precautions should be taken at the time of power-on.
3. Before erasing, clean the surface of the transparent lid to remove completely oily impurities or paste, which may impede irradiation and affect the erasing characteristics.

ABSOLUTE MAXIMUM RATING

Symbol	Parameter	Conditions	Limits	Unit
V_{I1}	Input voltage, V_{PP}	With respect to GND	$-0.3 \sim 26.5$	V
V_{I2}	Input voltage, V_{CC}, address, $\overline{OE}$, $\overline{CE}$, data		$-0.3 \sim 6$	V
Topr	Operating free-air temperature range		$0 \sim 70$	℃
Tstg	Storage temperature range		$-65 \sim 125$	℃

READ OPERATION

Recommended Operating Conditions ($T_a = 0 \sim 70$℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{PP}	Supply voltage		$(V_{PP} = V_{CC})$		V
GND	Supply voltage		0		V
V_{IL}	Low-level input voltage	-0.1		0.8	V
V_{IH}	High-level input voltage	2.2		$V_{CC}+1$	V

Electrical Characteristics (Ta = 0 ~ 70℃, V_{CC} = 5 V ± 5 %, V_{PP} = V_{CC}, unless otherwise noted)

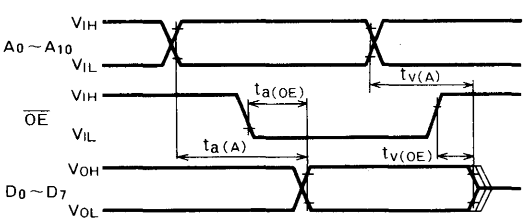
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ (Note 1)	Max	
I_{IL}	High-level input current, address, $\overline{OE}$, $\overline{CE}$	V_I = 5.25V			10	μA
I_{OZ}	Off-state output current	V_O = 5.25V, $\overline{OE}$ = 5 V			10	μA
I_{PP1}	Supply current from V_{PP}	V_{PP} = 5.85V			6	mA
I_{CC1}	Supply current from V_{CC} (standby)	$\overline{CE}$ = V_{IH}, $\overline{OE}$ = V_{IL}		10	25	mA
I_{CC2}	Supply current from V_{CC} (operating)	$\overline{OE}$ = $\overline{CE}$ = V_{IL}		57	100	mA
V_{OL}	Low-level output voltage	I_{OL} = 2.1mA			0.45	V
V_{OH}	High-level output voltage	I_{OH} = −400μA	2.4			V

Switching Characteristics (Ta = 0 ~ 70℃, V_{CC} = 5 V ± 5 %, V_{PP} = V_{CC}, unless otherwise noted)

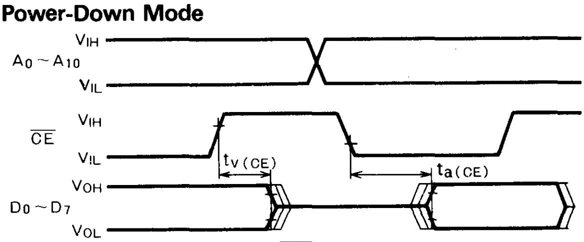
Symbol	Parameter		Test conditions		Limits			Unit
					Min	Typ (Note 1)	Max	
$t_{a(A)}$	Address access time	M5L 2716K	$\overline{OE}$ = $\overline{CE}$ = V_{IL}	t_r ≦ 20ns, t_f ≦ 20ns, V_{IL} = 0.8V, V_{IH} = 2.2V			450	ns
		M5L 2716K−65					650	ns
$t_{a(CE)}$	Chip enable access time	M5L 2716K	$\overline{OE}$ = V_{IL}				450	ns
		M5L 2716K−65					650	ns
$t_{a(OE)}$	Output enable access time	M5L 2716K	$\overline{CE}$ = V_{IL}			80	150	ns
		M5L 2716K−65					300	ns
$t_{V(OE)}$	Data valid time after output enable		$\overline{CE}$ = V_{IL}		0		100	ns
$t_{V(CE)}$	Data valid time after chip select		$\overline{OE}$ = V_{IL}		0		100	ns
$t_{V(A)}$	Data valid time after address		$\overline{OE}$ = $\overline{CE}$ = V_{IL}		0			ns

Note 1: at Ta = 25℃ and normal supply voltage.

Timing Diagrams (Read Operation) When Power-Down Mode Not Used



Power-Down Mode



Note 2: The center line indicates a floating (high-impedance) state.

3

PROGRAM MODE

Recommended Operating Conditions (Ta = 25 ± 5 ℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{PP}	Supply voltage	24	25	26	V
GNG	Supply voltage		0		V
V_{IL}	Low-level input voltage	−0.1		0.8	V
V_{IH}	High-level input voltage	2.2		$V_{CC}+1$	V

Electrical Characteristics (Ta = 25 ± 5 ℃, V_{CC} = 5 V ± 5 %, V_{PP} = 25 ± 1 V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
I_{IL}	High-level input current, address, $\overline{OE}$, $\overline{CE}$	V_{IN} = 5.25V			10	μA
I_{PP1}	Supply current from V_{PP}	$\overline{CE} = V_{IL}$			6	mA
I_{PP2}	Supply current from V_{PP}	$\overline{CE} = V_{IH}$			30	mA
I_{CC}	Supply current from V_{CC}				100	mA

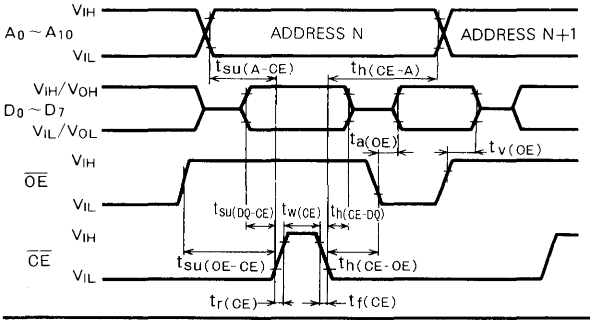
Timing Requirements (Ta = 25 ± 5 ℃, V_{CC} = 5 V ± 5 %, V_{PP} = 25 ± 1 V unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{su(A-CE)}$	Address setup time before chip enable		2			μs
$t_{su(OE-CE)}$	Output enable setup time before chip enable		2			μs
$t_{su(DQ-CE)}$	Data input setup time before chip enable		2			μs
$t_{h(CE-A)}$	Address hold time after chip enable		2			μs
$t_{h(CE-OE)}$	Output enable hold time after chip enable		2			μs
$t_{h(CE-DQ)}$	Data input hold time after chip enable		2			μs
$t_{w(CE)}$	Chip enable pulse width		45	50	55	ms
$t_{r(CE)}$	Chip enable pulse rise time		5			ns
$t_{f(CE)}$	Chip enable pulse fall time		5			ns

Switching Characteristics (Ta = 25 ± 5 ℃, V_{CC} = 5 V ± 5 %, V_{PP} = 25 ± 1 V, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{v(OE)}$	Data valid time after output enable			0		120	ns
$t_{a(OE)}$	Output enable access time	M5L 2716K				150	ns
		M5L 2716K−65				300	ns

Timing Diagram (for Program and Verify)



32 768-BIT(4096-WORD BY 8-BIT) ERASABLE AND ELECTRICALLY REPROGRAMMABLE ROM

DESCRIPTION

These are ultraviolet-light erasable and electrically reprogrammable 32 768-bit (4096-word by 8-bit) EPROMS. They incorporate N-channel silicon-gate MOS technology, and are designed for microprocessor programming applications.

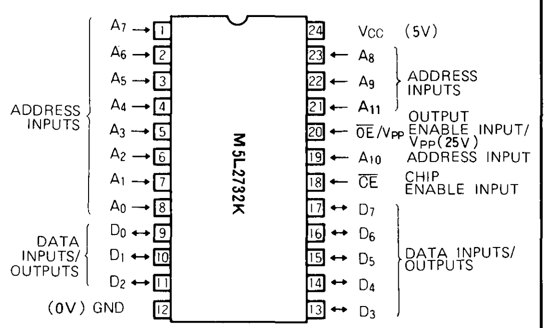
FEATURES

- Fast programming: 200s/32 768 bits (typ)
- Access time M5L 2732K: 450ns (max)
  M5L 2732K-6: 550ns (max)
- Static circuits are used throughout
- Inputs and outputs TTL-compatible in read and program modes
- Single 5V power supply for read mode
  (25V power supply required for program)
- Low power dissipation: Operating: 787mW (max)
  Standby: 157mW (max)
- Single-location programming
  (requires one 50ms pulse/address)
- Interchangeable with Intel's 2732 in pin configuration

APPLICATION

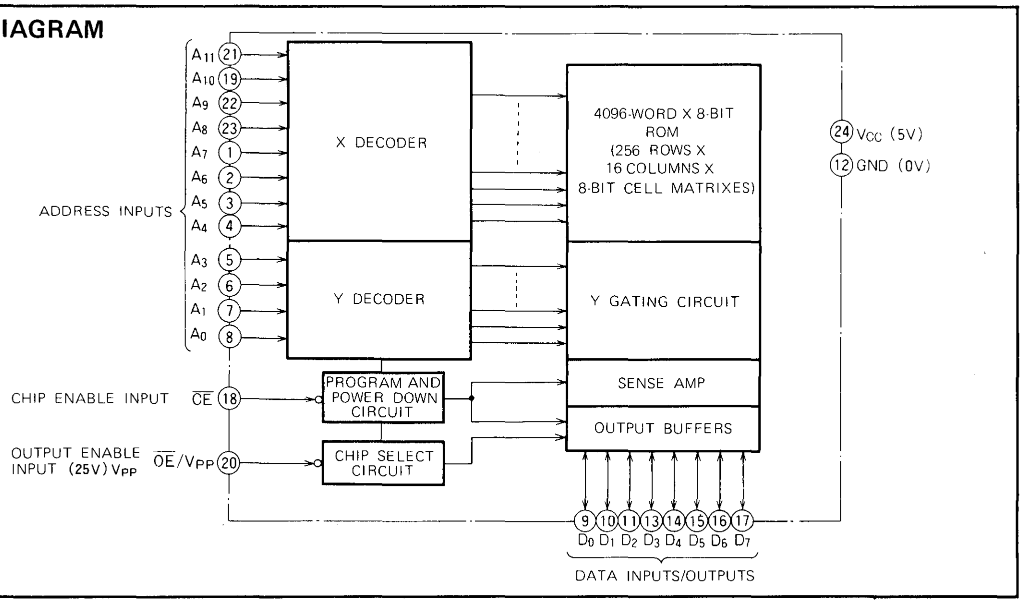
- Computers and peripheral equipment

PIN CONFIGURATION (TOP VIEW)



Outline 24K10

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

32 768-BIT(4096-WORD BY 8-BIT) ERASABLE AND ELECTRICALLY REPROGRAMMABLE ROM

FUNCTION

Read

Set the $\overline{CE}$ and $\overline{OE}$ terminals to the read mode (low-level). Low-level input to $\overline{CE}$ and $\overline{OE}$ and address signals to the address inputs ($A_0 \sim A_{11}$) make the data contents of the designated address location available at the data inputs/outputs ($D_0 \sim D_7$). When the $\overline{CE}$ or $\overline{OE}$ signal is high, data inputs/outputs ($D_0 \sim D_7$) are in a floating state.

When the $\overline{CE}$ signal is high, the device is in the standby mode or power-down mode.

Programming

The chip enters the programming mode when 25V is supplied to the $\overline{OE}/V_{PP}$ input. A location is designated by address signals $A_0 \sim A_{11}$, and the data to be programmed must be applied at 8 bits in parallel to the data inputs $D_0 \sim D_7$. A program pulse, an active low pulse, to the $\overline{CE}$ at this state will effect the programming operation. Only one programming is required, but its width must satisfy the condition $45ms \leqq t_{W(CE)} \leqq 55ms$.

Erase

Erase is effected by exposure to ultraviolet light with a wavelength of 2537Å at an intensity of approximately $15Ws/cm^2$.

Mode selection

(Unit: V)

Mode \ Pin	$\overline{CE}$	$\overline{OE}/V_{PP}$	V_{CC}	Outputs
Read	V_{IL}	V_{IL}	5	Output
Deselect	$V_{IL} \sim V_{IH}$	V_{IH}	5	Floating
Power down	V_{IH}	$V_{IL} \sim V_{IH}$	5	Floating
Program	Pulsed V_{IH} to V_{IL}	25	5	Input
Program verify	V_{IL}	V_{IL}	5	Output
Program inhibit	V_{IH}	25	5	Floating

HANDLING PRECAUTIONS

1. Sunlight and fluorescent light may contain ultraviolet light sufficient to erase the programmed information. For any operation in the read mode, the transparent window should be covered with opaque tape.
2. High voltages are used when programming, and the conditions under which is it performed must be carefully controlled to prevent the application of excessively high voltages. Specifically, the voltage applied to V_{PP} should be kept below 26V including overshoot. Special precautions should be taken at the time of power-on.
3. Before erasing, clean the surface of the transparent lid to remove completely oily impurities or paste, which may impede irradiation and affect the erasing characteristics.

32 768-BIT(4096-WORD BY 8-BIT) ERASABLE AND ELECTRICALLY REPROGRAMMABLE ROM

ABSOLUTE MAXIMUM RATING

Symbol	Parameter	Conditions	Limits	Unit
V_{I1}	Input voltage, $\overline{OE}$/V_{PP} input	With respect to GND	−0.3∼26.5	V
V_{I2}	Input voltage, V_{CC}, address, $\overline{CE}$, data inputs		−0.3∼6	V
Topr	Operating free air temperature range		0∼70	℃
Tstg	Storage temperature range		−65∼125	℃

READ OPERATION

Recommended Operating Conditions (Ta = 0 ∼ 70℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
GND	Supply voltage		0		V
V_{IL}	Low-level input voltage	−0.1		0.8	V
V_{IH}	High-level input voltage	2.2		V_{CC}+1	V

Electrical Characteristics (Ta = 0 ∼ 70℃, V_{CC} = 5V ± 5%, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ (Note 1)	Max	
I_{IL1}	High-level input current, address, $\overline{CE}$ input	V_I=5.25V			10	μA
I_{IL2}	High-level input current, $\overline{OE}$/V_{PP} input	V_I=4.75V			10	μA
I_{OZ}	Off-state output current	V_O=5.25V, $\overline{OE}$ = 5 V			10	μA
I_{CC1}	Supply current from V_{CC} (standby)	$\overline{CE}$ = V_{IH}, $\overline{OE}$ = V_{IL}		15	30	mA
I_{CC2}	Supply current from V_{CC} (operating)	$\overline{OE}$ = $\overline{CE}$ = V_{IL}		85	150	mA
V_{OL}	Low-level output voltage	I_{OL} = 2.1mA			0.45	V
V_{OH}	High-level output voltage	I_{OH} = −400μA	2.4			V

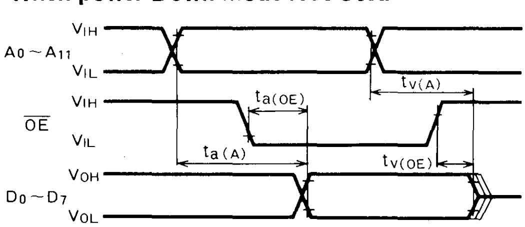
Switching Characteristics (Ta = 0 ∼ 70℃, V_{CC} = 5 V + 5 %, unless otherwise noted.)

Symbol	Parameter		Test conditions		Limits			Unit
					Min	Typ (Note 1)	Max	
ta(A)	Address access time	M5L 2732K	$\overline{OE}$ = $\overline{CE}$ = V_{IL}	t_r ≦ 20ns t_f ≦ 20ns V_{IL} = 0.8V V_{IH} = 2.2V Load: 100pF+1TTL			450	ns
		M5L 2732K−6					550	ns
ta(CE)	Chip enable access time	M5L 2732K	$\overline{OE}$ = V_{IL}				450	ns
		M5L 2732K−6					550	ns
ta(OE)	Output enable access time	M5L 2732K	$\overline{CE}$ = V_{IL}			100	150	ns
		M5L 2732K−6					200	ns
tv(OE)	Data valid time after output enable		$\overline{CE}$ = V_{IL}		0		100	ns
tv(CE)	Data valid time after chip select		$\overline{OE}$ = V_{IL}		0		100	ns
tv(A)	Data valid time after address		$\overline{OE}$ = $\overline{CE}$ = V_{IL}		0			ns

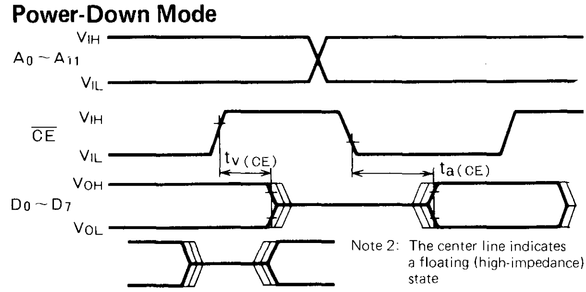
Note 1: at Ta = 25°C and normal supply voltage.

TIMING DIAGRAMS (Read Operation)

When power-Down Mode Not Used



Power-Down Mode



Note 2: The center line indicates a floating (high-impedance) state

3

PROGRAM MODE

Recommended Operating Conditions (Ta = 25 + 5 ℃, unless otherwise noted.)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{PP}	Supply voltage	24	25	26	V
GNG	Supply voltage		0		V
V_{IL}	Low-level input voltage	−0.1		0.8	V
V_{IH}	High-level input voltage	2.2		$V_{CC}+1$	V

Electrical Characteristics (Ta = 25 ± 5 ℃, V_{CC} = 5 V ± 5 %, V_{PP} = 25 ± 1 V, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
I_{IL}	High-level input current, address, $\overline{CE}$ inputs	V_{IN} = 5.25V			10	μA
I_{PP}	Supply current from V_{PP}	$\overline{CE} = V_{IL}$			30	mA
I_{CC}	Supply current from V_{CC}				150	mA

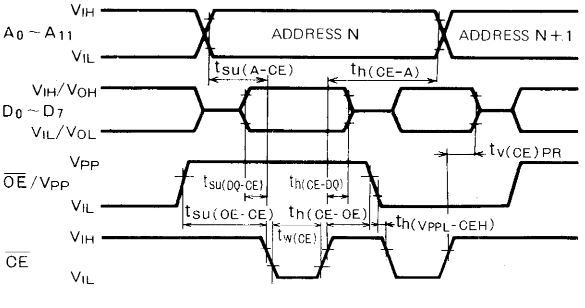
Timing Requirements (Ta = 25 ± 5 ℃, V_{CC} = 5 V ± 5 %, V_{PP} = 25 ± 1 V, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{su(A-CE)}$	Address setup time before chip enable		2			μs
$t_{su(OE-CE)}$	Output enable setup time before chip enable		2			μs
$t_{su(DQ-CE)}$	Data input setup time before chip enable		2			μs
$t_{h(CE-A)}$	Address hold time after chip enable		2			μs
$t_{h(CE-OE)}$	Output enable hold time after chip enable		2			μs
$t_{h(CE-DQ)}$	Data input hold time after chip enable		2			μs
$t_{h(V_{PP}L-CEH)}$	Chip enable high hold time after V_{PP} low		2			μs
$t_{w(CE)}$	Chip enable pulse width		45	50	55	ms

Switching Characteristics (Ta = 25 ± 5 ℃, V_{CC} = 5 V ± 5 %, V_{PP} = 25 ± 1 V, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{v(CE)PR}$	Data valid time after chip enable in program mode		0		120	ns

Timing Diagram (for Program and Verify)



M5L2764K, K-2, K-3

65536-BIT (8192-WORD BY 8-BIT) ERASABLE AND ELECTRICALLY REPROGRAMMABLE ROM

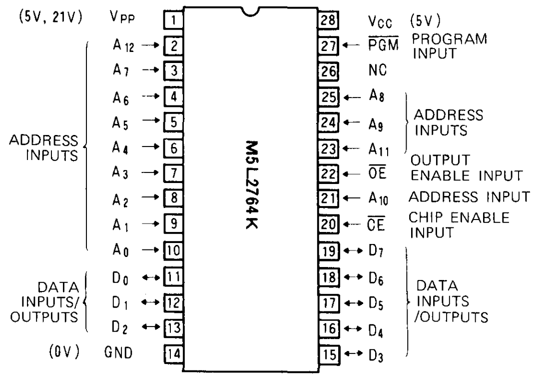
DESCRIPTION

The Mitsubishi M5L2764K is a high-speed 65536-bit ultraviolet erasable and electrically reprogrammable read only memory. It is suitable for microprocessor programming applications where rapid turn-around is required. The M5L2764K is fabricated by N-channel double polysilicon gate technology and is available in a 28-pin DIL package with a transparent lid.

FEATURES

- 8192 Word x 8-bit Organization
- Access Time M5L2764K-2 200 ns (Max)
  M5L2764K 250 ns (Max)
  M5L2764K-3 300 ns (Max)
- Two Line Control $\overline{OE}$, $\overline{CE}$
- Low Power Current (I_{CC}) Active ..... 150 mA (Max)
  Standby ... 35 mA (Max)
- Single 5V Power Supply
- 3-State Output Buffer
- Input and Output TTL-Compatible in Read and Program Mode
- Standard 28-pin DIL Package
- Single Location Programming with One 50 ms Pulse
- Interchangeable with INTEL 2764

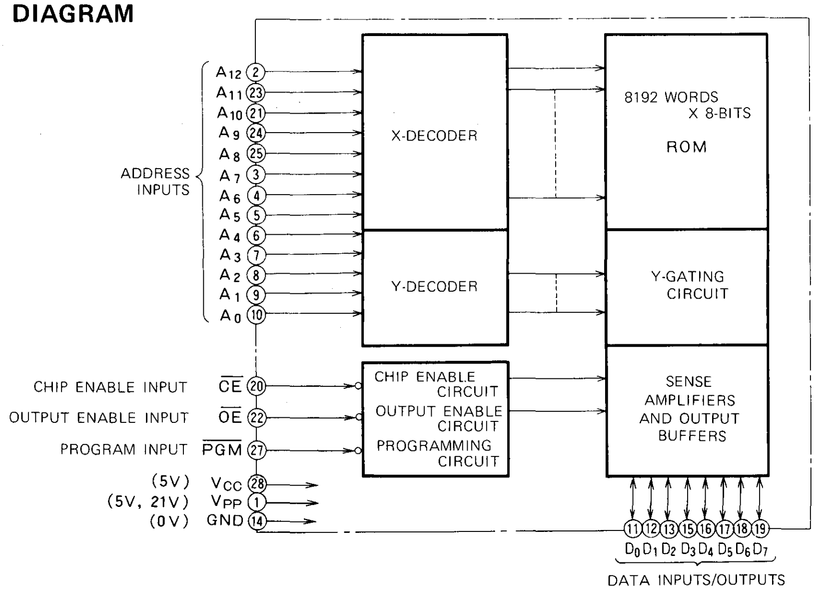
PIN CONFIGURATION (TOP VIEW)



NC : NO CONNECTION

Outline 28 K 10

BLOCK DIAGRAM



3

FUNCTION

Read
Set the $\overline{CE}$ and $\overline{OE}$ terminals to the read mode (low level).

Low level input to $\overline{CE}$ and $\overline{OE}$ and address signals to the address inputs ($A_0 \sim A_{12}$) make the data contents of the designated address location available at the data input/output ($D_0 \sim D_7$). When the $\overline{CE}$ or $\overline{OE}$ signal is high, data input/output are in a floating state.

When the $\overline{CE}$ signal is high, the device is in the standby mode or power-down mode.

Programming
The device enters the programming mode when 21V is supplied to the V_{PP} power supply input and $\overline{CE}$ is at low level. A location is designated by address signals ($A_0 \sim A_{12}$), and the data to be programmed must be applied at 8-bits in parallel to the data inputs ($D_0 \sim D_7$). A program pulse to the $\overline{PGM}$ at this state will effect programming. Only one programming pulse is required, but its width must satisfy the condition 45 mS $\leqq t_{PW} \leqq$ 55 mS.

Erase
Erase is effected by exposure to ultraviolet light with a wavelength of 2537Å at an intensity of approximately 15 WS/cm^2. Sunlight and fluorescent light may contain ultraviolet light sufficient to erase the programmed information. For any operation in the read mode, the transparent lid should be covered with opaque tape.

MODE SELECTION

MODE \ PINS	$\overline{CE}$ (20)	$\overline{OE}$ (22)	$\overline{PGM}$ (27)	V_{PP} (1)	V_{CC} (28)	Outputs (11-13, 15-19)
Read	V_{IL}	V_{IL}	V_{IH}	V_{CC}	V_{CC}	Data Out
Standby	V_{IH}	X	X	V_{CC}	V_{CC}	Floating
Program	V_{IL}	X	V_{IL}	V_{PP}	V_{CC}	Data In
Program Verify	V_{IL}	V_{IL}	V_{IH}	V_{PP}	V_{CC}	Data Out
Program Inhibit	V_{IH}	X	X	V_{PP}	V_{CC}	Floating

X can be either V_{IL} or V_{IH}

ABSOLUTE MAXIMUM RATINGS*

Symbol	Parameter	Conditions	Limits	Unit
V_{I1}	Input voltage , all input	With respect to GND	−0.6∼7	V
V_{I2}	Input voltage, V_{PP} input	With respect to GND	−0.6∼26.5	V
T_{opr}	Operating free air temperature range		0∼70	°C
T_{stg}	Storage temperature range		−65∼125	°C

READ OPERATION

$T_a = 0°$ to 70°C, $V_{CC} = 5V \pm 5\%$, $V_{PP} = V_{CC}$

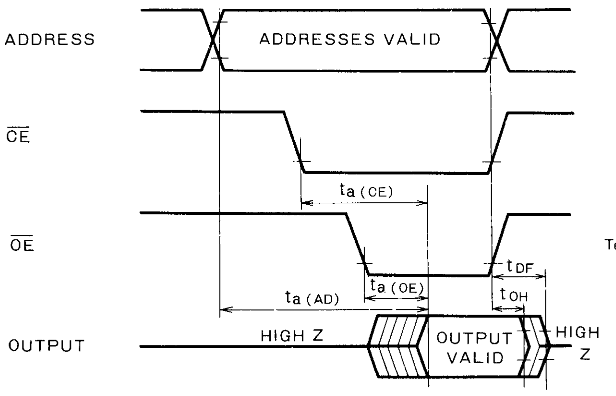
D. C. CHARACTERISTICS

Symbol	Parameter	Conditions	Limits Min	Limits Typ	Limits Max	Unit
I_{LI}	Input load current	V_{IN}=5.25V			10	μA
I_{LO}	Output leakage current	V_{OUT}=5.25V			10	μA
I_{PP1}	V_{PP} current read	V_{PP}=5.25V			15	mA
I_{CC1}	V_{CC} current standby	$\overline{CE}=V_{IH}$			35	mA
I_{CC2}	V_{CC} current active	$\overline{CE}=\overline{OE}=V_{IL}$			150	mA
V_{IL}	Low-level input voltage		−0.1		0.8	V
V_{IH}	High-level input voltage		2.0		V_{CC}+1	V
V_{OL}	Low-level output voltage	I_{OL}=2.1mA			0.45	V
V_{OH}	High-level output voltage	I_{OH}=−400μA	2.4			V

A. C. CHARACTERISTICS

Symbol	Parameter	Conditions	2764-2 Min	2764-2 Max	2764 Min	2764 Max	2764-3 Min	2764-3 Max	Unit
$t_{a(AD)}$	Address to output delay	$\overline{CE}=\overline{OE}=V_{IL}$		200		250		300	ns
$t_{a(CE)}$	$\overline{CE}$ to output delay	$\overline{OE}=V_{IL}$		200		250		300	ns
$t_{a(OE)}$	Output enable to output delay	$\overline{CE}=V_{IL}$	10	70	10	100	10	150	ns
t_{DF}	Output enable high to output float	$\overline{CE}=V_{IL}$	0	60	0	90	0	130	ns
t_{OH}	Output hold from $\overline{CE}$ or $\overline{OE}$	$\overline{CE}=\overline{OE}=V_{IL}$	0		0		0		ns

TIMING DIAGRAM



Test Conditions for A.C. Characteristics
Input Voltage: V_{IL} = 0.8V, V_{IH} = 2.2V
Input Rise and Fall Times: ≦ 20ns
Reference Voltage at Timing Measurement: Inputs 1V and 2V, Outputs 0.8V and 2V
Output Load: 1 TTL gate, C_L = 100pF

CAPACITANCE (T_a = 25°C, f = 1 MHz)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
C_{IN}	Input capacitance	V_{IN}= 0 V		4	6	pF
C_{OUT}	Output capacitance	V_{OUT}= 0 V		8	12	pF

3

PROGRAMMING OPERATION

$Ta = 25 \pm 5°C$, $V_{CC} = 5V \pm 5\%$, $V_{PP} = 21 \pm 0.5V$

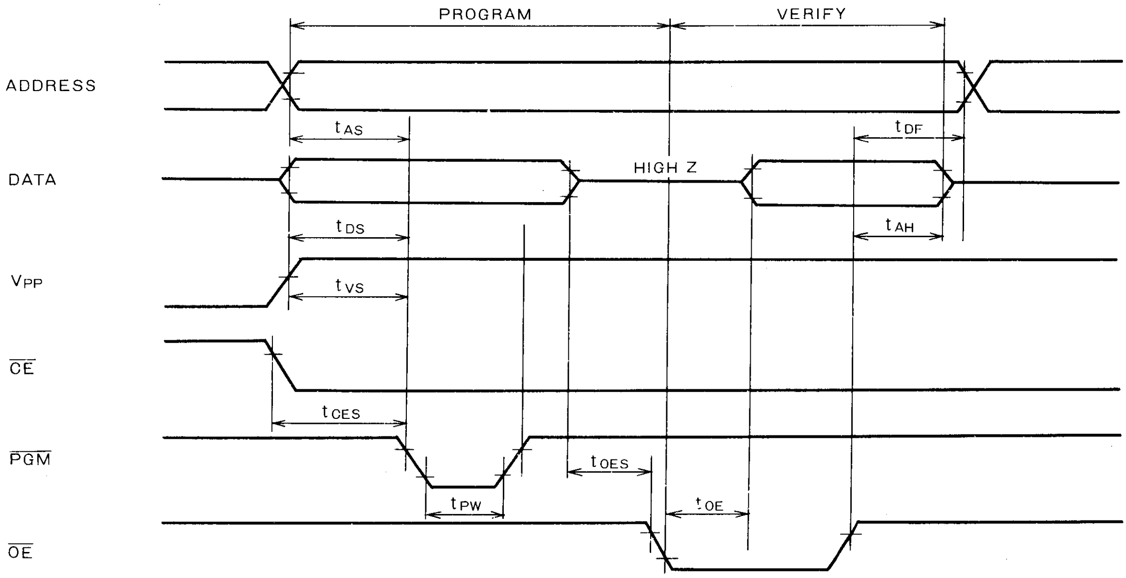
D. C. CHARACTERISTICS

Symbol	Parameter	Conditions	Limits			Unit
			Min	Typ	Max	
I_{LI}	Input current	$V_{IN} = V_{IL}$ or V_{IH}			10	μA
V_{OL}	Low-level output voltage (verify)	$I_{OL} = 2.1mA$			0.45	V
V_{OH}	High-level output voltage (verify)	$I_{OH} = -400\mu A$	2.4			V
I_{CC2}	V_{CC} supply current (active)				150	mA
V_{IL}	Low-level input voltage		−0.1		0.8	V
V_{IH}	High-level input voltage		2.0		$V_{CC}+1$	V
I_{PP}	V_{PP} supply current	$\overline{CE} = V_{IL} = \overline{PGM}$			30	mA

A. C. CHARACTERISTICS

Symbol	Parameter	Conditions	Limits			Unit
			Min	Typ	Max	
t_{AS}	Address setup time		2			μs
t_{OES}	$\overline{OE}$ setup time		2			μs
t_{DS}	Data setup time		2			μs
t_{AH}	Address hold time		0			μs
t_{DH}	Data hold time		2			μs
t_{DF}	Chip enable to output delay		0		130	ns
t_{VS}	V_{PP} setup time		2			μs
t_{PW}	$\overline{PGM}$ pulse width (programming)		45	50	55	ms
t_{CES}	$\overline{CE}$ setup time		2			μs
t_{OE}	Data valid from $\overline{OE}$				150	ns

TIMING DIAGRAM



Test Conditions for AC Characteristics
Input Voltage: $V_{IL} = 0.8V$, $V_{IH} = 2.2V$
Input Rise and Fall Times: ≦ 20ns
Reference Voltage at Timing Measurement: Inputs 1V and 2V
Outputs 0.8V and 2V

MELPS4MICROCOMPUTERS

4

M58840-XXXP,M58841-XXXSP

SINGLE-CHIP 4-BIT MICROCOMPUTER WITH 8-BIT A/D CONVERTER

DESCRIPTION

The M58840-XXXP and M58841-XXXSP are single chip 4-bit microcomputers developed using p-channel aluminum gate ED-MOS technology and are housed in 42-pin plastic DIL packages. These single-chip microcomputers feature a built-in 8-bit A-D converter.

Differences between the M58840-XXXP and M58841-XXXSP.

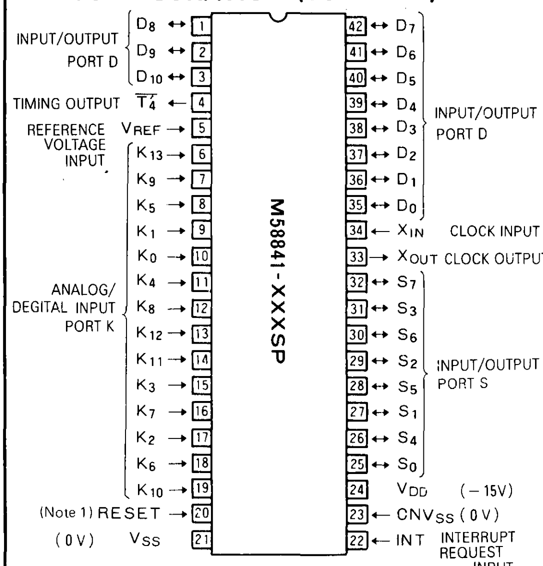
M58840-XXXP	Pin 5 is used as both the V_{REF} input and RESET input
M58841-XXXSP	The V_{REF} input pin and RESET input pin are separate, with pin 20 being used as the RESET input. Therefore, port K is a 14-bit port.

Except for the above differences, unless otherwise noted, the M58840-XXXP is the same as the M58841-XXXSP.

FEATURES

- Basic machine instructions 68
- Basic instruction execution time (1-word instruction at a clock frequency of 600kHz) 10μs
- Memory capacity: ROM .... 2048 words x 9 bits
  RAM .... 128 words x 4 bits
- Single −15V power supply
- Built-in A/D converter (14 or 15 analog inputs)
- 2 built-in data pointers
- Analog/digital input (port K):
  M58840-XXXP .... 15 inputs
  M58841-XXXSP .... 14 inputs
- Input/output port (ports D and S) .... 19 lines
- Direct drive for large fluorescent display tubes is possible
- Interrupt function .... 1 factor 1 level
- Built-in decoder PLA for port S output (mask option)
- On-chip clock generator

PIN CONFIGURATION (TOP VIEW)



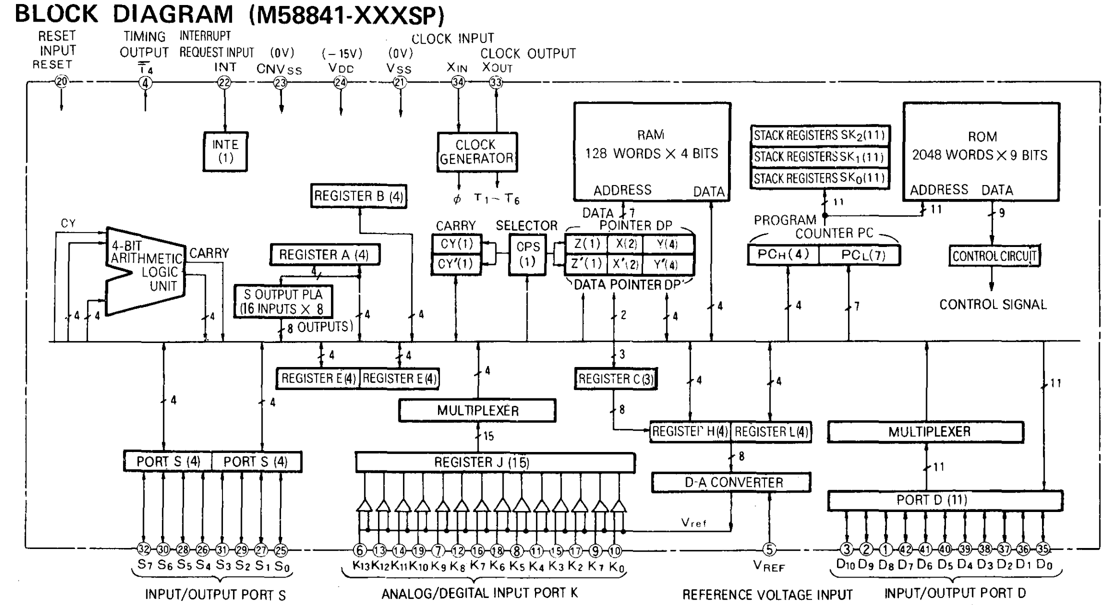
Outline 42P4B (M58841-XXXSP)
42P1 (M58840-XXXP)

Note 1. Pin 20 is the port K K_{14} pin for the M58840-XXXP.

APPLICATIONS

- Microwave ovens, air conditioners, heaters, washing machines, home sewing machines
- Office equipment, copying machines, medical equipment

BLOCK DIAGRAM (M58841-XXXSP)



4

PERFORMANCE SPECIFICATIONS

Parameter			Performance M58840-XXXP	Performance M58841-XXXSP
Basic machine instructions			68	68
Instruction execution time (1-word instruction)			10μs	10μs
Clock frequency			300～600kHz	300～600kHz
Memory capacity	ROM		2048 words x 9 bits	2048 words x 9 bits
	RAM		128 words x 4 bits	128 words x 4 bits
I/O port	K	Input	1 bit x 15	1 bit x 14
	S	Output	8 bits x 1	8 bits x 1
		Input	4 bits x 2	4 bits x 2
	D	Output	1 bit x 11	1 bit x 11
		Sense input	1 bit x 11	1 bit x 11
A/D conversion circuit			Built-in (accuracy ± 2LSB)	Built-in (accuracy±2LSB)
RESET input			Common with V_{REF} pin	Independent RESET pin
Subroutine nesting			3 levels (including one level of interrupt)	3 levels (including one level of interrupt)
Clock generator			Built-in (externally connected RC circuit or ceramic resonator)	Built-in (externally connected RC circuit or ceramic resonator)
I/O characteristics of ports	I/O withstanding voltage		−33 V	−33V
	Port S output current		−8mA	− 8 mA
	Port D output current		−15mA	−15mA
Supply voltage	V_{DD}		−15V (typ)	−15V (typ)
	V_{SS}		0 V	0 V
Device structure			p-channel aluminum gate ED-MOS	P-channel aluminum gate ED-MOS
Package			42-pin plastic molded DIL	42-pin shrink plastic molded DIL
Power dissipation (exluding ports)			400mW (typ)	400mW (typ)

PIN DESCRIPTION

Pin	Name	Input or output	Function
V_{DD} V_{SS}	Power supplies	In	V_{DD} and V_{SS} are applied as −15V±10% and 0V respectively
K_0 ~ K_{13}	Input port K	In	The input port K consists of 14 (15 for the M58840-XXXP) independent analog input pins. They can be programmed to receive digital quantities as well.
S_0 ~ S_7	Input/output port S	In/out	The I/O port S can be used as either an 8-bit output port or a pair of 4-bit input ports. Since it has open drain circuits, it is suitable for directly driving segments of a large fluorescent display tube. When the output port S is programmed to a low level, it remains in the floating state (high-impedance) so that it can be used as an input port.
D_0 ~ D_{10}	Input/output port D	In/out	The I/O port D is composed of 11 bits that can be used as independent I/O bits. When the port D outputs are programmed to a low level, the output remains in the floating state (high-impedance) and the input signal level is sensed.
X_{IN}	Clock input	In	A clock generator is built into the device so that the clock frequency is determined by connecting an external RC circuit or ceramic resonator between pins X_{IN} and X_{OUT}. When an external clock source is used, it should be connected to the X_{IN} pin, leaving the X_{OUT} pin open.
X_{OUT}	Clock output	Out	This pin is the output of the built-in clock generator circuit. The oscillation frequency is controlled by connecting an RC circuit or ceramic resonator element between this pin and the X_{IN} pin.
INT	Interrupt request input	In	This pin is used to input the interrupt request signal. The level of the interrupt signal can be programmed as either high or low.
$\overline{T_4}$	Timing output	Out	This is the basic timing output. It is used for testing and should be connected to V_{SS} (0V).
V_{REF}	Reference voltage input	In	This is the input for the reference voltage applied to the D-A converter. For the M58840-XXXP it serves as the RESET input pin as well.
CNV_{SS}	CNV_{SS} input	In	This input is connected to V_{SS} and must have a high-level input applied to it (0V).
RESET	Reset input	In	This is the reset input pin for the M58841-XXXSP. The reset state is enabled when it is kept high for at least two machine cycles.

BASIC FUNCTION BLOCKS

Program Memory (ROM)

This 2048-word x 9-bit mask programmable ROM can be programmed with machine instruction codes in accordance with the customer's specifications. It consists of 16 pages, each containing an address range of 0 ~ 127. Fig. 1 shows the address map of this ROM.

Program Counter (PC)

This counter is used to specify ROM addresses and the sequence of read-out of instructions stored in ROM. The program counter (PC) is an 11-bit counter, the upper order 4 bits of each (PC_H) indicate the ROM page, and the lower order 7 bits (PC_L) of which are a pure binary address designation. Each time an instruction is executed, PC_L is incremented by one step. For branching, subroutine call instructions and return instructions, its value is set to the designated address.

When the 127th address is reached for every page, the address value returns to the first address of that page. Therefore, for moving from one page to another page, the page byte itself must be modified. This is done using the BL and BLA instructions.

Page 14 and page 15 are special pages used for subroutine calls. The page 14 subroutine can be called with a one word instruction from any arbitrary page. This instruction is either BM or BMA. When either BM or BMA is executed, subsequent BM or BMA instructions are equivalent to B and BA on page 14. Also, B or BA is equivalent to B or BA on page 15. This condition is cancelled when the RT, RTS, BL, BML, BLA or BMLA instruction is executed. Table 3 shows the instruction codes and corresponding states.

Stack Registers (SK_0, SK_1, SK_2)

These registers are used to temporarily store the contents of the PC while executing subroutines or interrupt programs until the program returns to its main routine. The SK registers are organized in three words of 11 bits each, enabling up to three levels of subroutine nesting. If one level is used for an interrupt routine, the remaining two levels can be used for subroutine calls.

Data Memory (RAM)

This 512-bit (128 words x 4 bits) RAM is used to store both processing and control data. One RAM word consists of 4 bits with bit manipulation possible over the entire storage area. The 128 words are arranged as 2 file groups x 4 files x 16 digits x 4 bits. Fig. 2 shows the RAM address map.

The RAM address specification is made by the combination of data pointer DP (register Z, register X and register Y.) Thus, the selector CPS and data pointer DP must be set. However, as long as the address is not changed this is not necessary.

Data Pointers (DP, DP')

These registers are used to designate RAM address, and bit positions for the I/O port D and register J. Each data pointer is composed of a 7-bit register. Register Z (the most significant bit of DP) designates the RAM file group; register X (the central 2 bits) designates a RAM file; and register Y (the least significant 4 bits) designates the digit position of the RAM file. At the same time, register Y designates bit positions of the I/O port D and register J.

4-bit Arithmetic Logic Unit (ALU)

This unit executes 4-bit arithmetic and logical operations by means of a 4-bit adder and related logic circuitry. The arithmetic logic unit performs subtraction, addition, logical comparisons, arithmetic comparisons, and bit manipulation.

Register A and Carry flag (CY)

Register A is a 4-bit accumulator that constitutes the basis for arithmetic operations. Data processing operations such as arithmetic and logical operations, data transfer, exchange, conversion and data input/output are executed by means of this register. Overflow of register A is stored in the carry flag's CY or CY' after execution of arithmetic or logical operations. The carry flags can also be used as 1-bit flags. Carry flags and data pointer DP selection is done by means of the selector CPS.

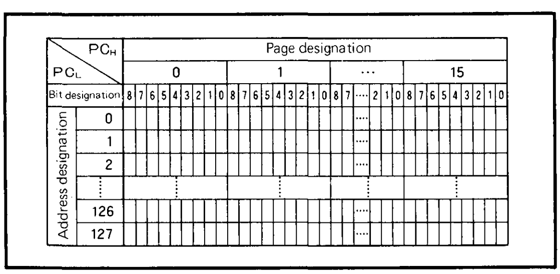


Fig. 1 ROM Address map

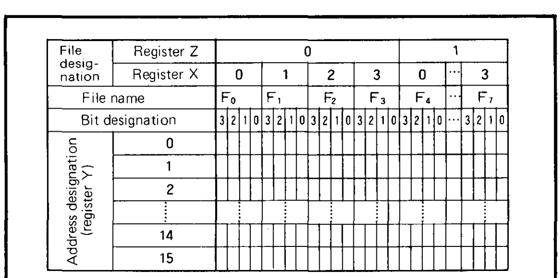


Fig. 2 RAM Address map

Registers B and E

Register B is composed of 4 bits and can be used as a 4-bit temporary storage register or for 8-bit data transfer in conjunction with register A. Register E is composed of 8 bits and is used not only as an 8-bit temporary storage register, but also as a temporary storage register for I/O port S.

A/D Conversion Circuit

The following A/D conversion functions are controlled by software as described below. Fig. 3 shows the block diagram.

(1) Comparators
These comparators are implemented entirely with PMOS devices and use a chopper-type amplification method. They are capable of determining the difference of the D/A converter output Vref and the port K input signals $V_{K(Y)}$ (where Y=0~13).

(2) Register J
Register J is composed of 14 1-bit registers, each representing the comparison result from the comparators. All register bits are set simultaneously. The value of the register J with respect to the comparison results is as follows.

$$1 \text{ when } |V_{ref}| > |V_{K(Y)}|$$

$$0 \text{ when } |V_{ref}| < |V_{K(Y)}|$$

In this relationship(Y) represents the bit position in register J which is designated by register Y. The comparison results can be checked for each bit using the SZJ instruction.

(3) Registers H and L
These two 4-bit registers are capable of transferring and exchanging data to and from register A. The 8-bit digital data for the D-A converter is transferred from these registers, the higher order 4 bits from H and lower order 4 bits from L.

(4) Register C
This 3-bit register is used as a counter to designate bit positions in the H and L registers.

(5) D/A Converter
The D/A converter converts the digital values stored in the registers H and L, referencing with the external reference voltage V_{REF} applied at the pin V_{REF}, to the analog value of the internal reference voltage Vref. The theoretical value of the internal reference voltage V_{ref} is defined as follows.

$$V_{ref} = \frac{n-0.5}{256} \times V_{REF}$$

where n = 1, 2, ... 255

$V_{ref}=0$ where n=0

In the above relationships n is the value weighted accorded to the contents of registers H and L.

A/D Conversion Algorithms

A/D conversion is controlled by the programming of the previously described functional blocks. Thus, by modifying the program either the successive approximation method or the sequential comparison method may be selected. In addition, a digital input of high or low level may be used to select the method, eliminating software selection of the A/D conversion technique.

(1) Successive Approximation Method
In this method, a constant conversion speed is maintained regardless of the amplitude of the analog signal. The A/D conversion process requires 0.6ms (at 600kHz clock frequency). 12 program words are required.

(2) Sequential Comparison Method
In this method the conversion speed varies in accordance with the rate of change of the analog quantity. When the rate of change is slow, the conversion rate increases. 30 program words are required.

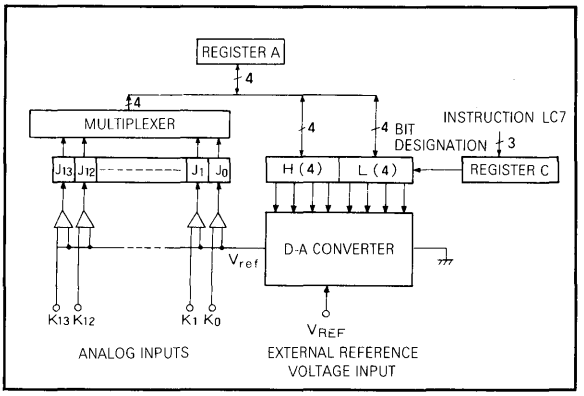


Fig. 3 A/D Conversion circuit block diagram

Interrupt Function

The flag INTE is a 1-bit flip-flop used to control interrupt operation. When an interrupt request signal is applied to the pin INT while the interrupt is enabled, the INTE flag is reset to disable further interrupts, after which the program jumps from the main program to address 0 of page 12. When an interrupt program is used, one level of the three-level stack register is required, the remaining two levels being used for subroutines. After the interrupt program is started, the data pointer DP, register A, carry flag CY, and registers used by the interrupt program must be saved and these must be restored before returning to the main program. The returning may be done by the execution of RTI instruction.

When an interrupt occurs, the microcomputer internal states are as follows.

(1) Program Counter
The current address in the main program is stored in a stack register and the program counter is set to page 12, address 0.

(2) Interrupt Flag INTE

The flag INTE is reset to disable further interrupts. This disabled state will continue even after the program has returned from the interrupt routine to the main program by the execution of the RTI instruction. EI is executed and when the input level of the INT input changes, this state is disabled. Thus, when the INTH instruction is executed the interrupt state is enabled when the INT input goes high. As long as it remains in the high state, further interrupts are prohibited. If the INT input should change to a low level and return to high, the next interrupt will be accepted.

(3) Skip Flags

Skip flags are provided for skip instructions and consecutively described instructions and these skip flags discriminate the skip and non skip conditions. Each flag has its own stack within which the skip state is saved.

As a mask option, the interrupt pin may be provided with Schmitt input circuits.

Input/Output Pins

(1) Input port K

The input port K consists of 14 pins (15 pins for the M58840-XXXP). The voltage level input at these pins is compared with the D-A converter output voltage Vref by a comparator and the results stored in register J. As a mask option, it is possible to build into the input port K load resistors. These are implemented using an enhancement-type (M58840-XXXP) or depletion-type (M58841-XXXSP) MOS transistors. In addition, to enable the use of capacitive touch-type keys, it is possible to provide these inputs with the required discharge transistors.

(2) Input/Output Port S

The input/output port S consists of 8 bits, each bit with an output latch. These latches are used to store data transferred by means of a PLA from register A, or data transferred from register A or register B directly. 4 bits at a time of the 8 input bits of port S may be transferred to register A.

Because port S outputs are provided with a built-in PLA, it is possible to output any arbitrarily settable 8-bit code from 4 input bits specified by register A. These PLA output codes can be specified arbitrarily as a masked option.

(3) Input/Output Port D

The input/output port D consists of 11 bits. Each bit can be individually designated as either input or output and is provided with its own latch. The contents of the data pointer register Y can be used to designate a single bit of port D for output or sensing.

When port S or port D is used as an input port, the output should first be cleared to the low state.

Reset Function

For the M58840-XXXP, when a power source satisfying the conditions shown in Fig. 4 is applied, an internal power-on reset function operates to reset the microcomputer. Cancelling of the reset state also is performed automatically, the program being started at page 0, address 0.

If the power-on reset function does not operate properly because of the trailing edge characteristics of the power supply, reset can be enabled by inputting a high level at the V_{REF} pin. Setting this V_{REF} pin to low starts the program at page 0, address 0.

For the M58841-XXXSP, if the RESET input is kept high for at least two machine cycles, the reset state is enabled. Because the M58841-XXXSP is provided with an internal charging transistor it requires only an external diode and capacitor as shown in Fig. 5.

For this configuration, when the supply voltage falls below −13.5V, the circuit design should ensure that the RESET input is above −4V.

When the reset state is enabled, the following operations are performed.

(1) The program counter is set to page 0, address 0 (PC) ← 0

(2) The interrupt mode is in the interrupt disabled state (INTE) ← 0

This is the same state as when the instruction DI is executed.

(3) By setting the interrupt request signal INT to high, the interrupt enabled state is entered. This is the same state as when the instruction INTH is executed.

(4) All outputs of port S are cleared to low (S) ← 0

(5) All outputs of port D are cleared to low (D) ← 0

(6) The carry and data pointer selector CPS is cleared to low to designate DP and CY (CPS) ← 0

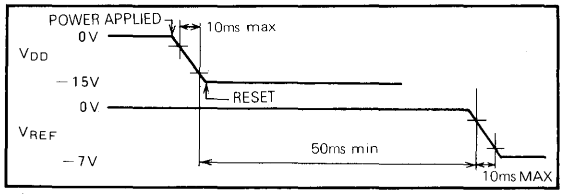


Fig.4 M58840-XXXP Power on reset

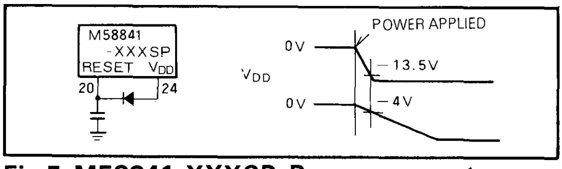
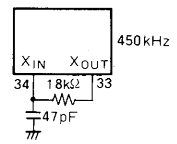


Fig.5 M58841-XXXSP Power on reset

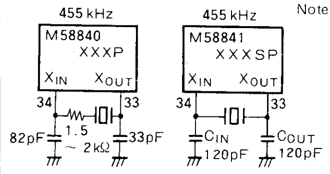
Clock Generator Circuit

A clock generator circuit has been built-in to the M58840-XXXP and M58841-XXXSP, allowing control of the frequency by means of an externally connected RC circuit or ceramic resonator. In addition, an external clock signal may be applied at the X_{IN} pin, leaving the X_{OUT} pin open. Circuit examples are shown in Fig. 6 to Fig. 8.



Note 1. Since when using an RC circuit to determine frequency the frequency will vary in accordance with the device characteristics, the constants should be chosen so that the frequency falls within a specified limit even including the effect of these variations.

Fig. 6 External RC circuit



Note 2. Constants should be chosen to provide a clock duty cycle in the range 50~60%. The duty cycle can be measured at the T_4 pin. Note that the required constants will also vary with the frequency and type of ceramic resonator used.

Fig. 7 Externally connected ceramic resonator

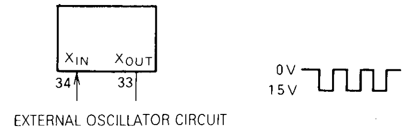


Fig. 8 External clock input circuit

4

MACHINE INSTRUCTIONS

Type of instruction	Mnemonic	Instruction code D_8 $D_7D_6D_5D_4$ $D_3D_2D_1D_0$	16 mal notation	Skip conditions	No. of cycles	Functions	Skip conditions	Flag CY	Description of operation
RAM addresses	LXY x,y	0 11xx yyyy	0 C y + x	1	1	(X)←x, where, x=0~3 (Y)←y, where, y=0~15	Written successively	X	Loads value of "x" into register X, and of "y" into Y. When LXY is written successively, the first is executed and successive ones are skipped
	LZ z	0 0100 101z	0 4A + z	1	1	(Z)←z, where, z=0, 1		X	Loads value of "z" into register Z.
	INY	0 0000 0010	0 02	1	1	(Y)←(Y)+1	(Y)=0	X	Increments contents of register Y by 1. Skips next instruction when new contents of register Y are "0".
	DEY	0 0000 0011	0 03	1	1	(Y)←(Y)−1	(Y)=15	X	Decrements contents of register Y by 1 . Skips next instruction when new contents of register Y are "15".
	LCPS i	0 0100 000i	0 4i	1	1	(CPS)←i, where, i=0, 1	—	X	DP and CY are active when i=0, DP' and CY', when i=1.
Register-to-register transfers	TAB	0 0001 1110	0 1E	1	1	(A)←(B)	—	X	Transfers contents of register B to register A.
	TBA	0 0001 1100	0 1C	1	1	(B)←(A)	—	X	Transfers contents of register A to register B.
	TAY	0 0001 1101	0 1D	1	1	(A)←(Y)	—	X	Transfers contents of register Y to register A.
	TYA	0 0000 1100	0 0C	1	1	(Y)←(A)	—	X	Transfers contents of register A to register Y.
	TLA	0 0001 1001	0 19	1	1	(L)←(A)	—	X	Transfers contents of register A to register L.
	THA	0 0101 1001	0 59	1	1	(H)←(A)	—	X	Transfers contents of register A to register H.
	TEAB	0 0001 1010	0 1A	1	1	(E_7~E_4)←(B), (E_3~E_0)←(A)	—	X	Transfers contents of registers A and B to register E.
	TEPA	0 0001 0110	0 16	1	1	(E_7~E_0)←through PLA←(A)	—	X	Decodes contents of register A in the PLA and transfers result to register E.
	TAJ	0 0000 1101	0 0D	1	1	(A)←($J_3J_2J_1J_0$) when, (Y_1Y_0)=0 (A)←($J_7J_6J_5J_4$) when, (Y_1Y_0)=1 (A)←($J_{11}J_{10}J_9J_8$) when, (Y_1Y_0)=2 (A)←(0 $J_{14}J_{13}J_{12}$) when, (Y_1Y_0)=3	—	X	Transfers designated contents of register J to register A.
	XAL	0 0001 1000	0 18	1	1	(A)↔(L)	—	X	Exchanges contents of register A with contents of register L.
	XAH	0 0101 1000	0 58	1	1	(A)↔(H)	—	X	Exchanges contents of register A with contents of register H.
RAM-accumulator transfers	TAM j	0 0110 01jj	0 64 + j	1	1	(A)←(M(DP)) (X)←(X)∀j, where, j=0~3	—	X	Transfers the RAM contents addressed by the active DP to register A. Register X is then "exclusive OR-ed" with the value j in the instruction, and the result stored in register X.
	XAM j	0 0110 00jj	0 6j	1	1	(A)↔(M(DP)) (X)←(X)∀j, where, j=0~3	—	X	Exchanges the contents of the RAM and register A. Contents of X are then "exclusive OR-ed" with the value j, and the result stored in register X.
	XAMD j	0 0110 10jj	0 68 + j	1	1	(A)↔(M(DP)), (Y)←(Y)−1 (X)←(X)∀j, where, j=0~3	(Y)=15	X	Exchanges the contents of the RAM and register A. Contents of X are then "exclusive OR-ed" with the value j in the instruction, and the result stored in register X. The contents of register Y are decremented by 1, and when the result is 15, the next instruction is skipped.
	XAMI j	0 0110 11jj	0 6C + j	1	1	(A)↔(M(DP)), (Y)←(Y)+1 (X)←(X)∀j where, j=0~3	(Y)=masked skip condition	X	Exchanges the contents of the RAM and register A.Contents of X are then "exclusive OR-ed" with the value j in the instruction and result stored in register X. The contents of register Y are incremented by 1, and when the result meets with the marked skip condition the next instruction is skipped.
Arithmetic operations	LA n	0 1011 nnnn	0 Bn	1	1	(A)←n, where, n=0~15	Written successively	X	Loads the value n into register A. When LA is written conseutively the first is executed, and successive ones are skipped.
	AM	0 0000 1010	0 0A	1	1	(A)←(A)+(M(DP))	—	X	Adds the contents of the RAM to register A. The result is retained in register A, and the contents of flag CY are unaffected.
	AMC	0 0000 1110	0 0E	1	1	(A)←(A)+(M(DP))+(CY) (CY)←carry	—	0/1	Adds the RAM contents addressed by the active DP and contents of flag CY to register A.The result is stored in register A and the carry in the active flag CY.
	AMCS	0 0000 1111	0 0F	1	1	(A)←(A)+(M(DP))+(CY) (CY)←carry	(CY)=1	0/1	Adds the contents of the RAM and flag CY to register A. The result is stored in register A and the carry in the CY, but the next instruction is skipped when a carry is produced.
	A n	0 1010 nnnn	0 An	1	1	(A)←(A)+n, where, n=0~15	A carry =0 but n≠6	X	Adds value n in the instruction to register A. The contents of flag CY are unaffected and their next instruction is skipped if a carry is not produced, except when n=6.
	SC	0 0100 1001	0 49	1	1	(CY)←1	—	1	Sets active flag CY
	RC	0 0100 1000	0 48	1	1	(CY)←0	—	0	Resets active flag CY
	SZC	0 0010 1111	0 2F	1	1		(CY)=0	X	Skips next instruction when contents of the active flag CY are 0.
	CMA	0 1000 1111	0 8F	1	1	(A)←$\overline{(A)}$	—	X	Stores complement of register A in register A.
Bit operations	SB j	0 0100 11jj	0 4C + j	1	1	(M_j(DP))←1, where, j=0~3	—	X	Set the jth bit of the RAM addressed by the active DP (the bit designated by the value j in the instruction).
	RB j	0 0101 11jj	0 5C + j	1	1	(M_j(DP))←0, where, j=0~3		X	Resets the jth bit of the RAM addressed by the active DP (the bit designated by the value j in the instruction)
	SZB j	0 0010 00jj	0 2j	1	1		(M_j(DP))=0 where, j=0~3	X	Skips next instruction when the contents of the jth bit of the RAM addressed by the active DP (the bit which is designated by the value y in the instruction) are 0.
Compares	SEAM	0 0010 0110	0 26	1	1		(M(DP))=(A)	X	Skips next instruction when contents of register A are equal to the RAM contents addressed by the active DP.
	SEY y	0 0011 yyyy	0 3y	1	1		(Y)=y where, y=0~15		Skips next instruction when the contents of register Y are equal to the value j in the instruction.
A/D converter operations	LC7	0 0101 0111	0 57	1	1	(C)←7		X	Loads 7 to register C.
	DEC	0 0000 1001	0 09	1	1	(C)←(C)−1	(C)=7	X	Decrements contents of register C by 1, when result is 7, skips next instruction.
	SHL	0 0100 0010	0 42	1	1	(H(C_1 C_0))←1 when, (C_2)=1 (L(C_1 C_0))←1 when, (C_2)=0	—	X	Sets the bit in register L or H designated by register C. The box instruction shows the relationship between register C and bit position. (C): 7 6 5 4 3 2 1 0 / Bit: H_3 H_2 H_1 H_0 L_3 L_2 L_1 L_0
	RHL	0 0101 0010	0 52	1	1	(H(C_1 C_0))←0 when, (C_2)=1 (L(C_1 C_0))←0 when, (C_2)=0	—	X	Resets the bit in register L or H that is designated by register C.
	CPA	0 0000 1000	0 08	1	1 (2)	($J_{(i)}$)←1 when, $\|V_{ref}\|>\|V_{K(i)}\|$ ($J_{(i)}$)←0 when, $\|V_{ref}\|<\|V_{K(i)}\|$ i=0~14		X	Reads all analog values from input port K for comparison with D-A converter output V_{ref}, and either sets the respective bit of register J to the next instruction cycle, whenever $V_{ref} < V_{K(i)}$ is true, or resets it, whenever $V_{ref} < V_{K(i)}$ is true.
	CPAS	0 0101 0001	0 51	1	1	($J_{(i)}$)←1 when, $\|V_{ref}\|>\|V_{K(i)}\|$ ($J_{(i)}$)←0 when, $\|V_{ref}\|<\|V_{K(i)}\|$ i=0~14	—	X	Reads and stores temporarily all analog values from the input port K, which are the unaffected by changes in port K inputs. These values are compared with the D-A converter output V_{ref} calculated from contents of registers H and L and respective bits of register J are set/reset.
	CPAE	0 0101 0000	0 50	1	1	Execution of the instruction CPAS is over, and no more changes will made in ($J_{(Y)}$).		X	Terminates execution of instruction CPAS. Contents of register J remain unaffected, maintaining the value immediately before termination, and input port K is again ready to receive inputs.
	SZJ	0 0010 1001	0 29	1	1		($J_{(Y)}$)=0 (Y)=15	X	Skips next instruction when the bit in register J, designated by register Y, is 0. The next instruction is unconditionally skipped when the contents of register Y are 15.

Type of instruction	Mnemonic	Instruction code D_8 $D_7D_6D_5D_4$ $D_3D_2D_1D_0$	16mal notation	Skip conditions	No. of cycles	Functions	Skip conditions	Flag CY	Description of operation
Jumps	B xy	1 1xxx yyyy	1 8y + x	1	1	$(PC_L) \leftarrow 16x+y$	–	X	Jumps to address xy of the current page.
						$(PC_H) \leftarrow 15$, $(PC_L) \leftarrow 16x+y$			Jumps to address xy on page 15 when executed, provided that none of instructions RT, RTS, BL, BML or BMLA was executed after execution of instruction BM or BMA.
	BL pxy	0 0111 pppp 1 1xxx yyyy	0 7p 1 8y + x	2	2	$(PC_H) \leftarrow p$ $(PC_L) \leftarrow 16x+y$	–	X	Jumps to address xy of page p.
	BA xX	0 0000 0001	0 01	2	2	$(PC_L) \leftarrow 16x+(A)$	–	X	Jumps to address x(A) of the current page.
		1 1xxx XXXX	1 8X + x			$(PC_H) \leftarrow 15$, $(PC_L) \leftarrow 16x+(A)$			Jumps to the address x(A) of page 15 provided that none of instructions. RT, RTS, BL, BML, BLA or BMLA was executed after execution of instruction BM or BMA.
	BLA pxX	0 0000 0001 0 0111 pppp 1 1xxx XXXX	0 01 0 7p 1 8x + x	3	3	$(PC_H) \leftarrow p$ $(PC_L) \leftarrow 16x+(A)$	–	X	Jumps to the address x(A) of page p.
Subroutine calls	BM xy	1 0xxx yyyy	1 xy	1	1	$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$ $(PC_H) \leftarrow 14$, $(PC_L) \leftarrow 16x+y$	–	X	Calls for the subroutine starting at address xX on page 14.
						$(PC_H) \leftarrow 14$, $(PC_L) \leftarrow 16x+y$			Jumps to address xy of page 14 provided that none of instructions RT, RTS, BL, BML, BLA or BMLA was executed after the execution of instructions BM or BMA.
	BML pxy	0 0111 pppp 1 0xxx yyyy	0 7p 1 xy	2	2	$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$ $(PC_H) \leftarrow p$, $(PC_L) \leftarrow 16x+y$	–	X	Calls for the subroutine starting at address xx of page P.
	BMA xX	0 0000 0001 1 0xxx XXXX	0 01 1 xX	2	2	$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$ $(PC_H) \leftarrow 14$, $(PC_L) \leftarrow 16x+(A)$	–	X	Calls for the subroutine starting at address x(A) of page 14.
						$(PC_H) \leftarrow 14$, $(PC_L) \leftarrow 16x+(A)$			Jumps to address x(A) of page 14 provided that none of instructions RT, RTS, BL, BML, BLA or BMLA was executed after the execution of instructions BM or BMA.
	BMLA pxX	0 0000 0001 0 0111 pppp 1 0xxx XXXX	0 01 0 7p 1 XX	3	3	$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$ $(PC_H) \leftarrow p$, $(PC_L) \leftarrow 16x+(A)$	–	X	Calls for the subroutine starting at address x(A) of page p.
Program returns	RTI	0 0100 0110	0 46	1	1	$(PC) \leftarrow (SK0) \leftarrow (SK1) \leftarrow (SK2)$ Resets interrupt flip-flop	–	X	Returns from interrupt routine to main routin. The internal flip-flops is restored
	RT	0 0100 0100	0 44	1	1	$(PC) \leftarrow (SK0) \leftarrow (SK1) \leftarrow (SK2)$	–	X	Returns to the main routine from the subroutine.
	RTS	0 0100 0101	0 45	1	2	$(PC) \leftarrow (SK0) \leftarrow (SK1) \leftarrow (SK2)$	Unconditional skip	X	Returns to the main routine from the subroutine, and unconditionally skips the next instruction.
Input/output	SD	0 0001 0101	0 15	1	1	$(D(Y)) \leftarrow 1$, where, $(Z)=1, 0 \leqq (Y) \leqq 10$	–	X	Sets the bit of port D that is designated by register Y, when the contents of register Z are 1.
	RD	0 0001 0100	0 14	1	1	$(D(Y)) \leftarrow 0$, where, $(Z)=1, 0 \leqq (Y) \leqq 10$	–	X	Resets the bit of port D that is designated by register Y, when the contents of register Z are 1.
	SZD	0 0010 1011	0 2B	1	1	… where, $(Z)=1, 0 \leqq (Y) \leqq 10$	$(D(Y))=0$	X	Skips the next instruction if the contents of the bit of port D that is designated by register Y are 0 and the contents of register Z are 1.
	OSAB	0 0001 1011	0 1B	1	1	$(S_7 \sim S_4) \leftarrow (B)$, $(S_3 \sim S_0) \leftarrow (A)$		X	Outputs contents of register A and B to port S.
	OSPA	0 0001 0111	0 17	1	1	$(S_7 \sim S_0) \leftarrow$ through $PLA \leftarrow (A)$		X	Decodes contents of register A by PLA and the result is output to port.
	OSE	0 0000 1011	0 0B	1	1	$(S) \leftarrow (E)$		X	Outputs contents of register E to port S.
	IAS i	0 0101 010i	0 54 + i	1	1	$i=0$: $(A) \leftarrow (S_7 \sim S_4)$ $i=1$: $(A) \leftarrow (S_3 \sim S_0)$		X	Transfers from port S to register A. The high-order four bits of port S are transferred when the value of i in the instruction is 0 or the low-order four bits are transferred when the value of i is 1.
	CLD	0 0001 0011	0 13	1	1	$(D) \leftarrow 0$		X	Clears port D.
	CLS	0 0001 0000	0 10	1	1	$(S) \leftarrow 0$		X	Clears port S.
	CLDS	0 0001 0001	0 11	1	1	$(D) \leftarrow 0$, $(S) \leftarrow 0$		X	Clears ports S and D.
Interrupts	EI	0 0000 0101	0 05	1	1	$(INTE) \leftarrow 1$		X	Sets interrupt flag INTE to enable interrupts.
	DI	0 0000 0100	0 04	1	1	$(INTE) \leftarrow 0$		X	Resets interrupt flag INTE to disable interrupts.
	INTH	0 0000 0110	0 06	1	1	$(INTP) \leftarrow 1$		X	Sets interrupt polarity flag INTP to enable interrupts when the interrupt request signal is turned high.
	INTL	0 0000 0111	0 07	1	1	$(INTP) \leftarrow 0$		X	Resets interrupt polarity flag INTP to enable interrupts when the interrupt request signal is turned low.
Misc	NOP	0 0000 0000	0 00	1	1	$(PC) \leftarrow (PC)+1$		X	No operation

Symbol	Contents	Symbol	Contents	Symbol	Contents
A	4-bit register (aacumulator)	SK0	11-bit stack register	INTE	Interrupt enable flag
B	4-bit register	SK1	11-bit stack register	INTP	Interrupt polarity flag
C	3-bit register	SK2	11-bit stack register	INT	Interrupt request signal.
E	8-bit register	CY	1-bit carry flag	←	Shows direction of data flow.
H	4-bit register	xx	2-bit binary variable	( )	Indicates contents of the register, memory, etc.
J	15-bit register	yyyy	4-bit binary variable	⊻	Exclusive OR.
L	4-bit register	z	1-bit binary variable	—	Negation.
X	2-bit register	nnnn	4-bit binary constant	X	Indicates flag is unaffected by instruction execution.
Y	4-bit register	i	1-bit binary constant	xy	Label used to indicate the address xxyyyy.
Z	1-bit register	ij	2-bit binary constant	pxy	Label used to indicate the address xxyyyy of page pppp.
DP	7-bit data pointer, combination of registers, Z, X and Y	XXXX	4-bit unknown binary number	CPS	Indicates which data pointer and carry are active.
PC_H	The high-order four bits of the program counter.	D	11-bit port	C + x	Hexadecimal number C + binary number x.
PC_L	The low-order seven bits of the program counter.	K	15-bit port		
PC	11-bit program counter, combination of PC_H and PC_L.	S	8-bit port		

LIST OF INSTRUCTION CODES

D3–D0 \ D8–D4	Hexadecimal notation	0 0000	0 0001	0 0010	0 0011	0 0100	0 0101	0 0110	0 0111	0 1000	0 1001	0 1010	0 1011	0 1100	0 1101	0 1110	0 1111	1 0000 ~ 1 0111	1 1000 ~ 1 1111
		0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	0 A	0 B	0 C	0 D	0 E	0 F	10~17	18~1F
0000	0	NOP	CLS	SZB 0	SEY 0	LCPS 0	CPAE	XAM 0	BL BML	–	–	A 0	LA 0	LXY 0,0	LXY 1,0	LXY 2,0	LXY 3,0	BM	B
0001	1	BA BMA BLA BMLA	CLDS	SZB 1	SEY 1	LCPS 1	CPAS	XAM 1	BL BML	–	–	A 1	LA 1	LXY 0,1	LXY 1,1	LXY 2,1	LXY 3,1	BM	B
0010	2	INY	–	SZB 2	SEY 2	SHL	RHL	XAM 2	BL BML	–	–	A 2	LA 2	LXY 0,2	LXY 1,2	LXY 2,2	LXY 3,2	BM	B
0011	3	DEY	CLD	SZB 3	SEY 3	–	–	XAM 3	BL BML	–	–	A 3	LA 3	LXY 0,3	LXY 1,3	LXY 2,3	LXY 3,3	BM	B
0100	4	DI	RD	–	SEY 4	RT	IAS 0	TAM 0	BL BML	–	–	A 4	LA 4	LXY 0,4	LXY 1,4	LXY 2,4	LXY 3,4	BM	B
0101	5	EI	SD	–	SEY 5	RTS	IAS 1	TAM 1	BL BML	–	–	A 5	LA 5	LXY 0,5	LXY 1,5	LXY 2,5	LXY 3,5	BM	B
0110	6	INTH	TEPA	SEAM	SEY 6	RTI	–	TAM 2	BL BML	–	–	A 6	LA 6	LXY 0,6	LXY 1,6	LXY 2,6	LXY 3,6	BM	B
0111	7	INTL	OSPA	–	SEY 7	–	LC7	TAM 3	BL BML	–	–	A 7	LA 7	LXY 0,7	LXY 1,7	LXY 2,7	LXY 3,7	BM	B
1000	8	CPA	XAL	–	SEY 8	RC	XAH	XAMD 0	BL BML	–	–	A 8	LA 8	LXY 0,8	LXY 1,8	LXY 2,8	LXY 3,8	BM	B
1001	9	DEC	TLA	SZJ	SEY 9	SC	THA	XAMD 1	BL BML	–	–	A 9	LA 9	LXY 0,9	LXY 1,9	LXY 2,9	LXY 3,9	BM	B
1010	A	AM	TEAB	–	SEY 10	LZ 0	–	XAMD 2	BL BML	–	–	A 10	LA 10	LXY 0,10	LXY 1,10	LXY 2,10	LXY 3,10	BM	B
1011	B	OSE	OSAB	SZD	SEY 11	LZ 1	–	XAMD 3	BL BML	–	–	A 11	LA 11	LXY 0,11	LXY 1,11	LXY 2,11	LXY 3,11	BM	B
1100	C	TYA	TBA	–	SEY 12	SB 0	RB 0	XAMI 0	BL BML	–	–	A 12	LA 12	LXY 0,12	LXY 1,12	LXY 2,12	LXY 3,12	BM	B
1101	D	TAJ	TAY	–	SEY 13	SB 1	RB 1	XAMI 1	BL BML	–	–	A 13	LA 13	LXY 0,13	LXY 1,13	LXY 2,13	LXY 3,13	BM	B
1110	E	AMC	TAB	–	SEY 14	SB 2	RB 2	XAMI 2	BL BML	–	–	A 14	LA 14	LXY 0,14	LXY 1,14	LXY 2,14	LXY 3,14	BM	B
1111	F	AMCS	–	SZC	SEY 15	SB 3	RB 3	XAMI 3	BL BML	CMA	–	A 15	LA 15	LXY 0,15	LXY 1,15	LXY 2,15	LXY 3,15	BM	B

Note 1. The list shows the machine codes and corresponding machine instructions. D3~D0 indicate the low-order 4 bits of the machine code and D8~D4 indicate the high-order 5 bits. The hexadecimal values are also shown that represent these codes. An instruction may consist of 1, 2, or 3 words, but only the first word is listed. Codes indicated with bar (–) must not be used.

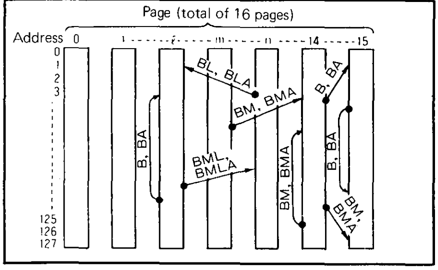
Note 2. Two-Word Instructions

	Second word
BL	1 1xxx yyy
BML	1 0xxx yyyy
BA	1 1xxx XXXX
BMA	1 0xxx XXXX

Three-Word Instructions

	Second word	Third word
BLA	0 0111 pppp	1 1xxx XXXX
BMLA	0 0111 pppp	1 0xxx XXXX

Note 3. Relationships of Branching and Paging for Branching, and Sub-Routine Call Instructions



MITSUBISHI ELECTRIC

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	0.3 ~ −20	V
V_I	Input voltage, port S and D inputs		0.3 ~ −35	V
V_I	Input voltage, inputs other than port S and D		0.3 ~ −20	V
V_O	Output voltage, port S and D outputs		0.3 ~ −35	V
V_O	Output voltage, outputs other than port S and D		0.3 ~ −20	V
P_d	Power dissipation	Ta = 25°C	1100	mW
Topr	Operating temperature		−10 ~ 70	°C
Tstg	Storage temperature		−40 ~ 125	°C

RECOMMENDED OPERATING CONDITIONS (T_a=−10~70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Typ	Max	
V_{DD}	Supply voltage	−13.5	−15	−16.5	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage	−1.5		0	V
$V_{IH(\phi)}$	High-level clock input voltage	−1.5		0	V
V_{IL}	Low-level input voltage, inputs other than port D and S	V_{DD}		−4.2	V
V_{IL}	Low-level input voltage, port D and S inputs	−33		−4.2	V
$V_{IL(\phi)}$	Low-level clock input voltage	V_{DD}		$V_{DD}+2$	V
$V_{I(K)}$	Analog input voltage, port K input	V_{REF}		0	V
V_{REF}	Reference voltage	−5		−7	V
V_{OL}	Low-level output voltage, port D and S outputs	−33		0	V
$f(\phi)$	Internal clock oscillation frequency	300		600	kHz

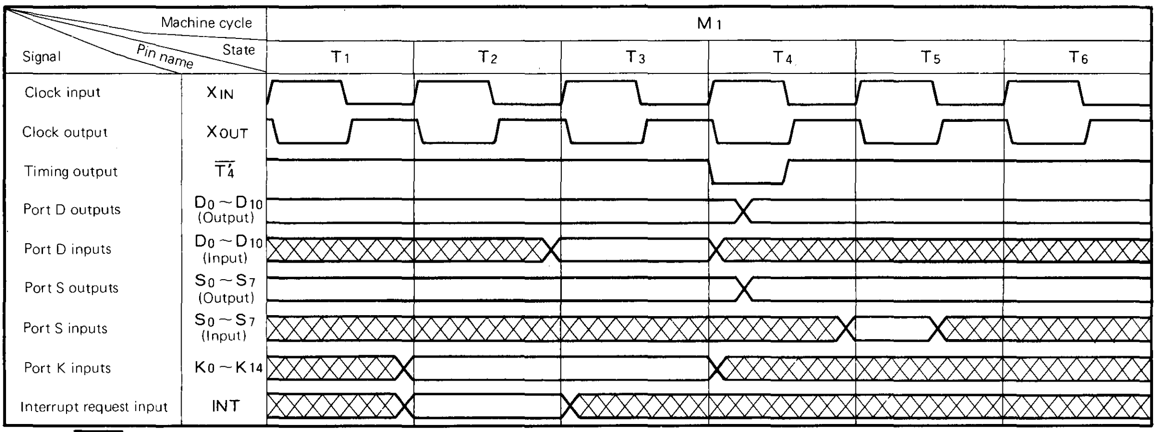
Note 1. $V_{IL(\phi)}$ is specified with respect to the maximum value of V_{DD}. The maximum allowable value is −33V when using a ceramic resonator with the M58841-XXXSP. This maximum allowable value is 1.1V for $V_{IH(\phi)}$ when using the M58841-XXXSP.

ELECTRICAL CHARACTERISTICS (T_a=−10~70°C, V_{DD}=−15V±10%, V_{SS}=0V, f(ϕ)=300~600kHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{T-}	Negative threshold voltage, RESET input	V_{DD} = −15V, Ta = 25°C	−4		−7	V
$V_{T+} - V_{T-}$	RESET input hysteresis	V_{DD} = −15V, Ta = 25°C	2		3.5	V
V_{OH}	High-level output voltage, port D outputs	V_{DD} = −15V, I_{OH} = −15mA, Ta = 25°C	−2.5			V
V_{OH}	High-level output voltage, port S outputs	V_{DD} = −15V, I_{OH} = −8mA, Ta = 25°C	−2.5			V
I_{IH}	High-level input current, port K (depletion load)	V_{DD} = −15V, V_{IH} = 0V, Ta = 25°C	100		370	μA
I_{IH}	High-level input current, port K (enhancement load)	V_{DD} = −15V, V_{IH} = 0V, Ta = 25°C	40		200	μA
I_{IH}	High-level input current, RESET	V_{DD} = −15V, V_{IH} = 0V, Ta = 25°C	30		100	μA
I_I	Input current, port K inputs	To be measured when the instruction CPAS or CPA is not being executed. V_I=−7V			−7	μA
$I_{I(\phi)}$	Clock input current	$V_{I(\phi)}$ = −15V, Ta = 25°C		−20	−40	μA
I_{OH}	High-level output current, port D outputs	V_{DD} = −15V, V_{OH} = −2.5V, Ta = 25°C			−15	mA
I_{OH}	High-level output current, port S outputs	V_{DD} = −15V, V_{OH} = −2.5V, Ta = 25°C			−8	mA
I_{OL}	Low-level output current, port D and port S outputs	V_{OL} = −33V, Ta = 25°C			−33	μA
I_{DD}	Supply current from V_{DD}	V_{DD} = −15V, Ta = 25°C			−41	mA
I_{REF}	Current from V_{REF}	V_{REF} = −7V, Ta = 25°C			−0.7	mA
C_I	Input capacitance, port K inputs	V_{DD} = V_I = V_O = V_{SS}, f = 1MHz 25mVrms		7	10	pF
$C_{I(\phi)}$	Clock input capacitance	V_{DD} = X_{OUT} = V_{SS}, f = 1MHz 25mVrms		7	10	pF
	A-D conversion linearity error	V_{REF} = −7V	Overall	±2	±3	LSB
	A-D conversion zero error	V_{REF} = −7V				
	A-D conversion fullscale error	V_{REF} = −7V				

Note 2. Currents are taken as positive when flowing into the IC (No sign), with the minimum and maximum values as absolute values.
3. The overall sum of the port D high-level output currents should be kept below 75mA.
4. The negative threshold voltage, hysteresis, high-level input current (depletion load), and high-level input current. For reset refer to the M58841-XXXSP.
5. The high-level input current (enhancement load), refer to the M58840-XXXP.

TIMING DIAGRAM



Note 1. The crosshatched area indicates invalid input.

DOCUMENTATION REQUIRED UPON ORDERING

The following information should be provided when ordering a custom mask.

(1) M58840-XXXP, M58841-XXXSP mask confirmation sheet.
(2) ROM data — 3 EPROM sets
(3) S output PLA coding — On confirmation sheets
(4) Interrupt input Schmitt circuits — On confirmation sheets
(5) M58840-XXXP reset circuits — On confirmation sheets
(6) Port K pulldown transistors — On confirmation sheets
(7) Pork K input discharge transistors — On confirmation sheets

MASK OPTIONS

The following type of mask options are available, specifiable at the time of ordering

(1) S output PLA data
(2) Interrupt input Schmitt circuit
(3) M58840-XXXP reset circuit
(4) Port K input pulldown resistors
(5) Port K input discharge transistors

DESCRIPTION

The M58842S MELPS 4 system evaluation device is designed to emulate the M58840-XXXP, M58841-XXXSP, M58843-XXXP and M58844-XXXP single-chip 4-bit microcomputer. It has been developed using P-channel aluminum-gate ED-MOS technology, and has a 64-pin ceramic DIL package. The M58842S facilitates fast development of new systems by using a program memory ROM external to the M58842S.

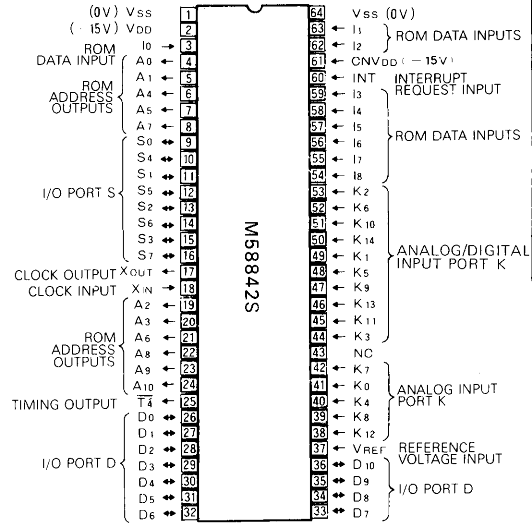
FEATURES

- Except for the mask ROM, all functions are equivalent to the M58840-XXXP.
- RAM capacity .... 128-word x 4-bit
- Single − 15V power supply
- Built-in A/D converter/(15 analog inputs)
- Two data pointers
- Subroutine nesting .... 3 levels
- Analog/digital input ports (K ports) 15
- I/O ports (ports D, S) 19
- Capable of direct drive of large fluorescent display tubes
- Interrupt .... 1 level
- Internal PLA (mask option) for port S
- Internal clock generator

APPLICATION

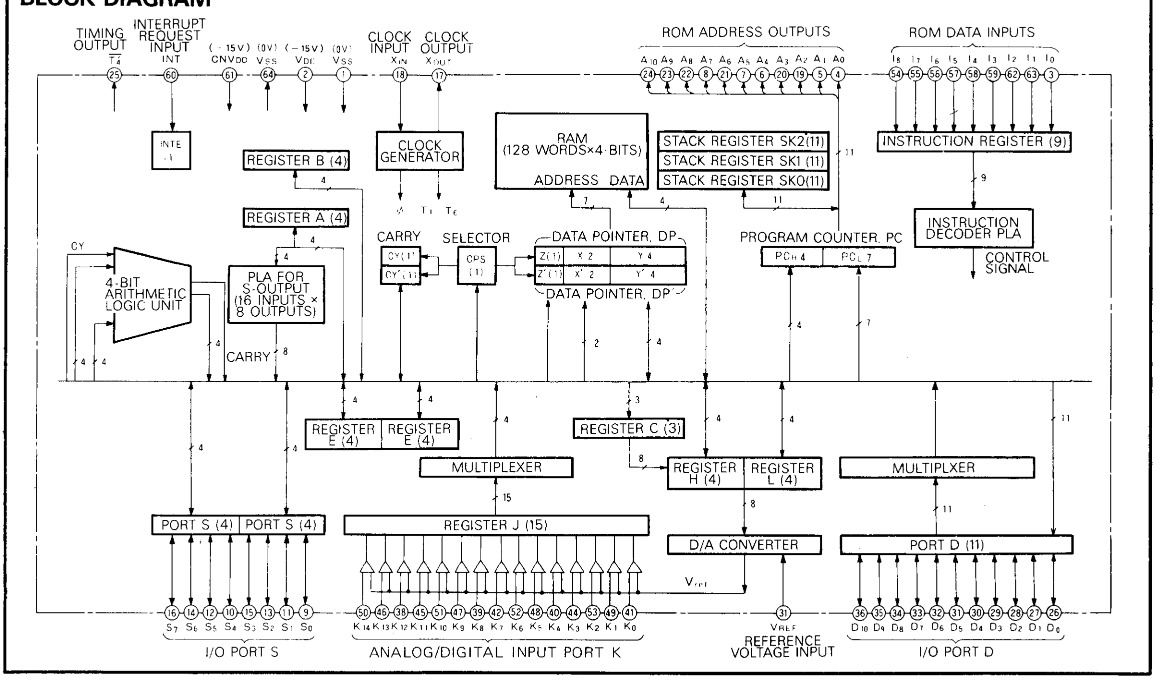
- System development and prototyping of equipment using the M58840-XXXP, M58841-XXXSP, M58843-XXXP, and M58844-XXXP single-chip 4-bit microcomputers.

PIN CONFIGURATION (TOP VIEW)



Outline 64S1

BLOCK DIAGRAM



FUNCTION

The M58842S MELPS 4 system evaluation device has the same functions as the M58840-XXXP single-chip 4-bit microcomputer except for the program memory ROM, which must be provided for from an external source connected through the address output pins ($A_0 \sim A_{10}$) and instruction input pins ($I_0 \sim I_8$).

In using the single-chip 4-bit microcomputer to control the operations of equipment, the operational procedures have to be put in a program and stored in the program memory (ROM). It may, however, consume a lot of time and effort, not to mention the cost, when a program correction is needed. This would naturally call for simulation of the application program before masking it into a ROM. In order to satisfy such a requirement, the M58842S has been prepared for evaluating a trial program before programming it into a mask-programmable ROM.

When using the M58842S for evaluating the M58841-XXXSP, M58843-XXXP and M58844-XXXP which are partially different from the M58840-XXXP (e.g. in the number of I/O ports), use the appropriate pins only. For example, since the M58843-XXXP is provided with a IK-word ROM, use the last IK words of the M58842S. Also since only the K_0 to K_3 ports are available, use K_0 through K_3 of the M58842S.

DESCRIPTION OF OPERATION

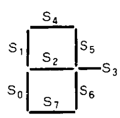
Programmable Logic Array (PLA) for the S-Output

The standard code listed below is stored in the PLA for the S-output. This code is used for numerical indication on 7-segment display units.

Input of ROM Data

Machine instructions can be executed by the M58842S if input from an external source. During the state T_2, the ROM address signal appears on the ROM address output pins $A_0 \sim A_{10}$. Then ROM data corresponding to this address should be applied to the ROM data input pins $I_0 \sim I_8$ during state T_6. For further details, refer to the instruction fetch timing diagram. During this application the input pin CNV_{DD} should be connected to V_{DD}.

LIST OF S-OUTPUT PLA CODES

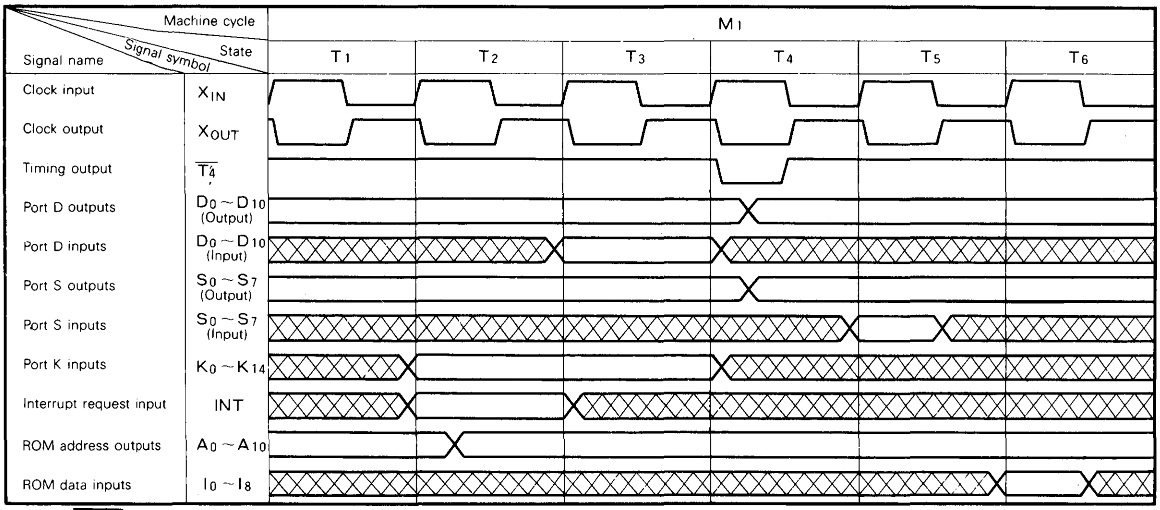


	Register A				Port S output								Display
Hexadecimal notation	A_3	A_2	A_1	A_0	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7	
0	0	0	0	0	H	H	L	L	H	H	H	H	0
1	0	0	0	1	L	L	L	L	L	H	H	L	1
2	0	0	1	0	H	L	H	L	H	H	L	H	2
3	0	0	1	1	L	L	H	L	H	H	H	H	3
4	0	1	0	0	L	H	H	L	L	H	H	L	4
5	0	1	0	1	L	H	H	L	H	L	H	H	5
6	0	1	1	0	H	H	H	L	H	L	H	H	6
7	0	1	1	1	L	H	L	L	H	H	H	L	7
8	1	0	0	0	H	H	H	L	H	H	H	H	8
9	1	0	0	1	L	H	H	L	H	H	H	H	9
A	1	0	1	0	H	L	H	L	L	L	H	H	o
B	1	0	1	1	L	L	L	H	L	L	L	L	—
C	1	1	0	0	H	H	H	L	H	L	L	H	E
D	1	1	0	1	H	H	L	L	H	L	L	H	C
E	1	1	1	0	L	L	H	L	L	L	L	L	—
F	1	1	1	1	L	L	L	L	L	L	L	L	Blank

PIN CONFIGURATION

Pin	Name	Input or output	Function
K_0 ~ K_{14}	Analog input port K	In	Analog port K has 15 independent analog input terminals. All signals applied to the 15 input lines of port K are simultaneously compared with the V_{ref} generated by the D-A converter. Corresponding bits of register J are set when the condition, $\|V_{ref}\|>\|V_{K(Y)}\|$ is met. This port is utilized for receiving input signals from the touch panel or receiving analog inputs from temperature and other sensing devices. It can also be used as a value threshold digital signal input port when the V_{ref} is properly selected.
S_0 ~ S_7	I/O port S	In/out	The I/O port S can be used as either an 8-bit output port or a pair of 4-bit input ports. Since it has open-drain circuits, it is suitable for directly driving segments of a large fluorescent display tube. It has an 8-bit output latch and can perform to drive 8 bits simultaneously. When the output of port S is programmed to low-level, it remains in the floating (high-impedance) state so that it can be used as an input port.
D_0 ~ D_{10}	I/O port D	In/out	The I/O port D is composed of 11 bits that can be used as discrete I/O units. Latches are provided on the output side to maintain individual output signals. When port D output is programmed to low-level, to keep it in floating (high-impedance) state, it can be used as a sense input port. The level of the input signal is sensed at the input terminal and is tested to determine if it is high or low by executing a skip instruction.
A_0 ~ A_{10}	ROM address output	Out	The address output is composed of 11 bits that output the contents of the program counter PC to the external program memory (ROM).
I_0 ~ I_8	ROM data input	In	The data input is composed of 9 bits that are used to fetch the instruction code for the CPU from the external program memory (ROM).
X_{IN}	Clock input	In	As the clock generator is contained internally, clock frequency is determined by connecting an external CR circuit or an IF ceramic resonator between the pins X_{IN} and X_{OUT}. In case an external clock source is to be used, it should be connected to the pin X_{IN}, leaving the pin X_{OUT} open.
X_{OUT}	Clock output	Out	This pin generates the clock frequency from the internal clock oscillation circuit. The oscillation frequency is controlled by connecting the CR circuit or IF ceramic resonator between this pin and the pin X_{IN}.
INT	Interrupt request input	In	This signal is used for requesting interrupts. Whether high or low-level interrupt signals are in used for requests is selected by means of the program. When the instruction INTH is executed, interrupt is accepted with a high-level signal, and accepted with a low-level signal when the instruction INTL is executed. When an interrupt is requested and accepted, program execution is jumped to address 0 of page 12. The instruction RTI is used for the return instruction.
V_{REF}	External reference voltage input	In	A reference voltage input is applied to the D-A converter from the external terminal. Its nominal value is V_{REF} = −7V. The value (n-0.5) $V_{REF}/256$ is generated by the D-A converter, and is compared with the analog signals from the input port K; where n represents the contents of the register H-L, but when n = 0, the output voltage is treated as 0V. It can also be used as an automatic reset signal input. When a high-level is applied to the V_{REF} input, it actuates the automatic reset circuit, and then the V_{REF} input is changed to low-level ready to start the program from address 0 of page 0.
$\overline{T_4}$	Timing output	Out	This pin generates a part of the basic timing pulse. This signal is used for testing other devices incorporated in the system.
CNV_{DD}	CNV_{DD} input	In	This input terminal should be conected with the V_{DD} and have a low-level input (−15V) applied.

BASIC TIMING CHART



Note 1: XXX indicates invalid signal input.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	0.3 ~ −20	V
V_I	Input voltage, port S and D inputs		0.3 ~ −35	V
V_I	Input voltage, other than port S and D inputs		0.3 ~ −20	V
V_O	Output voltage, port S and D outputs		0.3 ~ −35	V
V_O	Output voltage, other than port S and D outputs		0.3 ~ −20	V
P_d	Power dissipation	Ta = 25°C	1100	mW
Topr	Operating temperature		0 ~ 70	°C
Tstg	Storage temperature		−40 ~ 125	°C

RECOMMENDED OPERATING CONDITIONS (Ta = 0 ~ 70°C, unless otherwise noted)

Symbol	Parameter	Limits Min	Limits Nom	Limits Max	Unit
V_{DD}	Supply voltage	−13.5	−15	−16.5	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage	−1.5		0	V
$V_{IH(\phi)}$	High-level clock input voltage	−1.5		0	V
V_{IL}	Low-level input voltage, other than port D, port S and INT inputs	V_{DD}		−4.2	V
V_{IL}	Low-level input voltage, INT input	V_{DD}		−7	V
V_{IL}	Low-level input voltage; port D and S inputs	−33		−4.2	V
$V_{IL(\phi)}$	Low-level clock input voltage	V_{DD}		$V_{DD}+2$	V
$V_{I(K)}$	Analog input voltage; port K input	V_{REF}		0	V
V_{REF}	Reference voltage	−5		−7	V
V_{OL}	Low-level output voltage; port D and S outputs	−33		0	V
V_{OL}	Low-level output voltage, ROM address output	V_{DD}		0	V
$f(\phi)$	Internal clock oscillation frequency	300		600	kHz

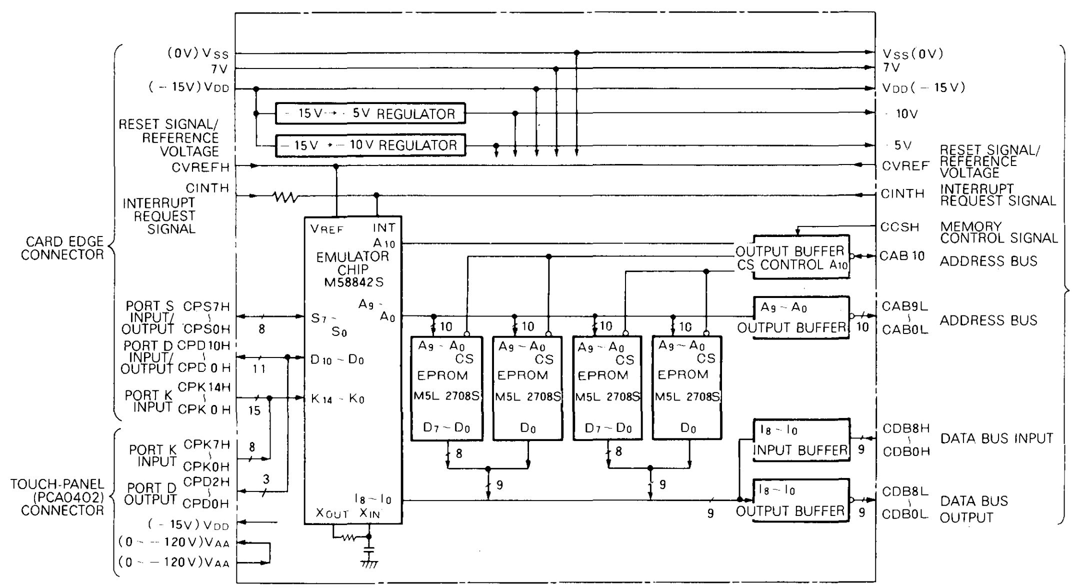
Note 1: The standard $V_{IL}(\phi)$ is with respect to the maximum V_{DD}.

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 70°C, V_{DD} = −15V ± 10%, V_{SS} = 0V, f(ϕ) = 300 ~ 600kHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
V_{IH}	High-level input voltage, port D and S inputs		−1.5		0	V
V_{IH}	High-level input voltage, ROM data inputs		−1.5		0	V
V_{IL}	Low-level input voltage, port D and S inputs		−33		−4.2	V
V_{IL}	Low-level input voltage, ROM data inputs		V_{DD}		−4.2	V
V_{OH}	High-level output voltage, port D outputs	V_{DD} = −15V, I_{OH} = −15mA, Ta = 25°C	−2.5			V
V_{OH}	High-level output voltage, port S outputs	V_{DD} = −15V, I_{OH} = −8mA, Ta = 25°C	−2.5			V
V_{OH}	High-level output voltage, ROM address outputs	V_{DD} = −15V, I_{OH} = −2mA, Ta = 25°C	−2			V
I_I	Input current, port K inputs	To be measured when the instruction CPAS or CPA is not being executed. V_I = −7V			−7	μA
$I_{I(\phi)}$	Clock input current	$V_{I(\phi)}$ = −15V, Ta = 25°C		−20	−40	μA
I_{OH}	High-level output current, port D outputs	V_{DD} = −15V, V_{OH} = −2.5V, Ta = 25°C			−15	mA
I_{OH}	High-level output current, port S outputs	V_{DD} = −15V, V_{OH} = −2.5V, Ta = 25°C			−8	mA
I_{OL}	Low-level output current, ports D and S outputs	V_{OL} = −33V, Ta = 25°C			−33	μA
I_{OL}	Low-level output current, ROM address outputs	V_{OL} = −17V, Ta = 25°C			−17	μA
C_i	Input capacitance, port K inputs	V_{DD} = V_I = V_O = V_{SS}, f = 1MHz 25mVrms		7	10	pF
$C_{i(\phi)}$	Clock input capacitance	V_{DD} = X_{OUT} = V_{SS}, f = 1MHz 25mVrms		7	10	pF
	A-D conversion linearity error	V_{REF} = −7V	Total	±2	±3	LSB
	A-D conversion zero error	V_{REF} = −7V				
	A-D conversion full-scale error	V_{REF} = −7V				

Note 2 : Current flowing into an IC is positive; out is negative.
3 : The sum of high-level output current from port D must be 75mA (max).

APPLICATION EXAMPLE



M58843-XXXP,M58844-XXXSP

SINGLE-CHIP 4-BIT MICROCOMPUTER WITH 8-BIT A/D CONVERTER

DESCRIPTION

The M58843-XXXP and M58844-XXXSP are single-chip 4-bit microcomputers fabricated using p-channel aluminum gate ED-MOS technology. They include an on-chip 8-bit A-D converter. The M58843-XXXP is housed in a 28-pin plastic moulded DIL package while the M58844-XXXSP is housed in a 40-pin shrink plastic molded DIL package.

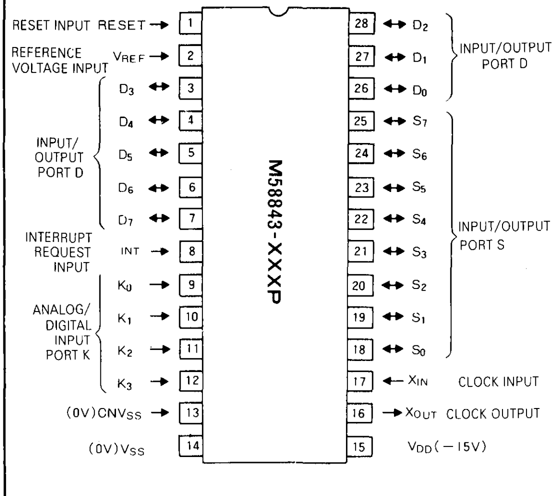
FEATURES

- Basic machine instructions 67
- Basic instruction execution time (for single-word instructions using a 600kHz clock frequency) 10μs
- Memory capacity ROM .... 1024 words x 9 bits
  RAM .... 64 words x 4 bits
- Single −15V power supply
- Built-in 8-bit A-D converter
- Two built-in data pointers
- Subroutine nesting .... 3 levels
- Analog/digital inputs (port K)
  M58843-XXXP .... 4 lines
  M58844-XXXSP .... 11 lines
- Input/output (ports D and S)
  M58843-XXXP .... 16 lines
  M58844-XXXSP .... 19 lines
- Capable of driving large fluorescent tube displays
- Interrupt function .... 1 factor, 1 level
- Built-in port S output decoder PLA (mask option)
- Built-in pull-down transistors (ports D, K, and S, mask option)
- Built-in clock generator circuit

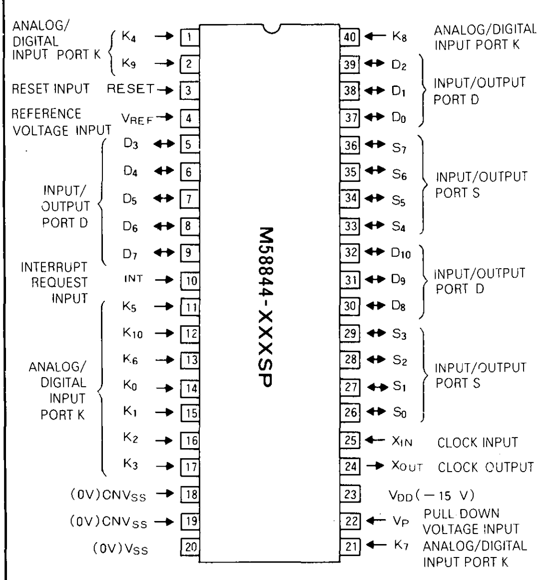
APPLICATIONS

- Electronic ranges, air conditioners, heaters, washing machines, rice cookers
- Office equipment, copying machines

PIN CONFIGURATIONS (TOP VIEW)



Outline 28P4

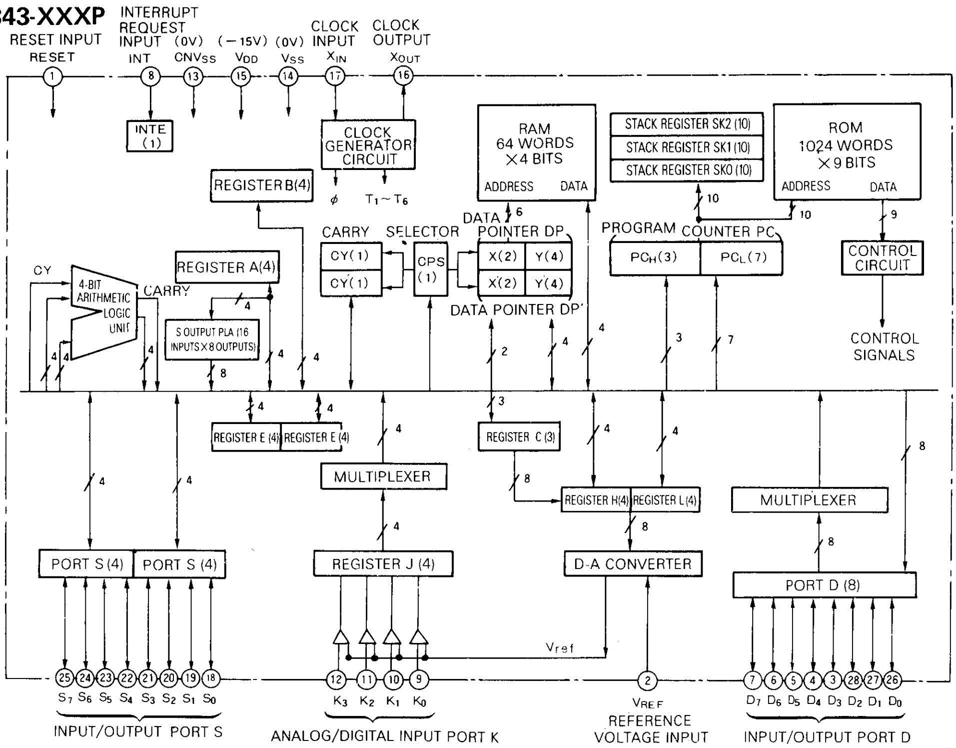


Outline 40P4B

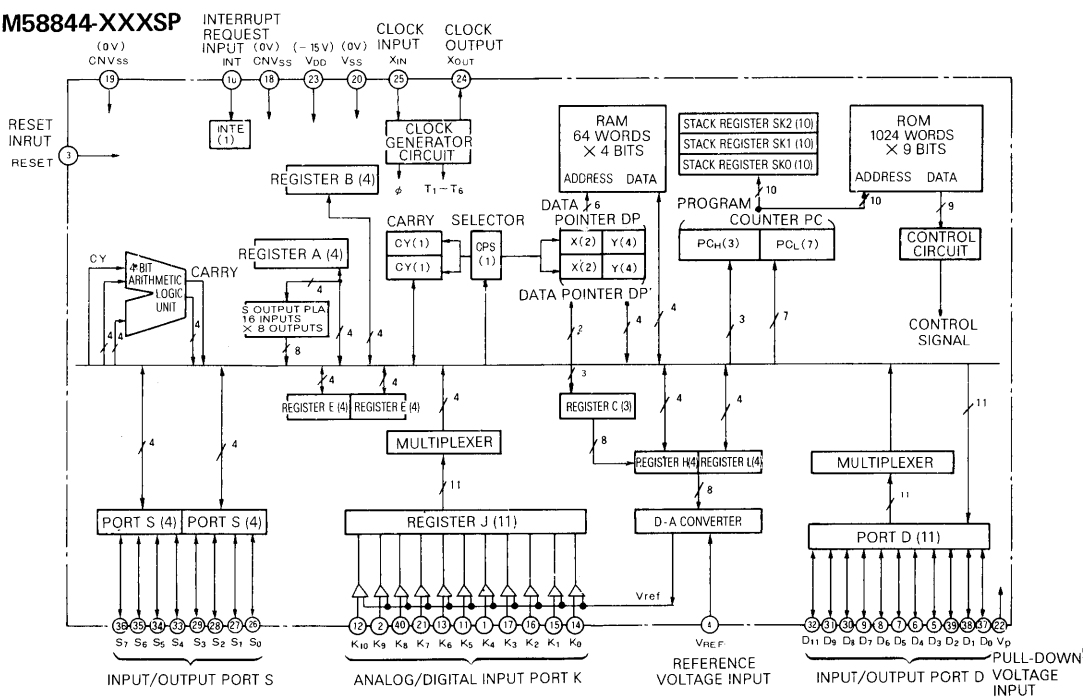
SINGLE-CHIP 4-BIT MICROCOMPUTER WITH 8-BIT A/D CONVERTER

BLOCK DIAGRAMS

M58843-XXXP



M58844-XXXSP



SINGLE-CHIP 4-BIT MICROCOMPUTER WITH 8-BIT A/D CONVERTER

PERFORMANCE SPECIFICATIONS

Parameter			Performance	
			M58843-XXXP	**M58844-XXXSP**
Basic machine instructions			67	67
Instruction execution time(1-word instruction)			10μs (with a clock frequency of 600kHz)	10μs (with a clock frequency of 600kHz)
Clock frequency			300～600kHz	300～600kHz
Memory capacity	ROM		1024 words x 9 bits	1024 words x 9 bits
	RAM		64 words x 4 bits	64 words x 4 bits
I/O port	K	Input	1 bit x 4	1 bit x 11
	S	Output	8 bits x 1	8 bits x 1
		Input	4 bits x 2	4 bits x 2
	D	Output	1 bit x 8	1 bit x 11
		Sense input	1 bit x 8	1 bit x 11
A-D conversion circuit			Built-in (accuracy ± 2LSB, typ)	Built-in (accuracy ± 2LSB, typ)
RESET input			1 pin	1 pin
Subroutine nesting			3 levels (including one level of interrupt)	3 levels (including one level of interrupt
Clock generator			Built-in(externally connected RC circuit or ceramic resonator)	Built-in (externally connected RC circuit or ceramic resonator)
I/O characteristics of ports	I/O withstanding voltage		-33V (max)	-33V (max)
	Port S output current		-8mA (max)	-8mA (max)
	Port D output current		-15mA (max)	-15mA (max)
Supply voltage	V_{DD}		-15V (typ)	-15V (typ)
	V_{SS}		0 V	0 V
Device structure			p-channel aluminum gate ED-MOS	P-channel aluminum gate ED-MOS
Package			28-pin plastic molded DIL package	40-pin shrink plastic molded DIL package
Power dissipation (excluding ports)			400mW (typ)	400mW (typ)

PIN DESCRIPTIONS

Pin	Name	Input or output	Function
V_{DD} V_{SS}	Power supplies	In	V_{DD} and V_{SS} are applied as -15V ±10% and 0V respectively
$K_0 \sim K_3$ (M58843-XXXP) $K_0 \sim K_{10}$ (M58844-XXXSP)	Analog/digital input port K	In	The input port K consists of 4 (11 for the M58844-XXXP) independent analog input pins. They can be programmed to receive digital quantities as well.
$S_0 - S_7$	Input/output port S	In/out	The I/O port S can be used as either an 8-bit output port or a pair of 4-bit input ports. Since it has open drain circuits, it is suitable for directly driving segments of a large fluorescent display tube. When the output port S is programmed to a low level, it remains in the floating state (high-impedance) so that it can be used as an input port.
$D_0 \sim D_7$ (M58843-XXXP) $D_0 \sim D_{10}$ (M58844-XXXSP)	Input/output port D	In/out	Port D consists of 8 bits for the M58843-XXXP and 11 bits for the M58844-XXXSP, all bits operating individually for input and output functions. When a port D output is programmed to low, the output floats (goes to high, impedance state) and the input signal can be sensed.
X_{IN}	Clock input	In	A clock generator is built into the device so that the clock frequency is determined by connecting an external RC circuit or ceramic resonator between pins X_{IN} and X_{OUT}. When an external clock source is used, it should be connected to the X_{IN} pin, leaving the X_{OUT} pin open.
X_{OUT}	Clock output	Out	This pin is the output of the built-in clock generator circuit. The oscillation frequency is controlled by connecting an RC circuit or ceramic resonator element between this pin and the X_{IN} pin.
INT	Interrupt request input	In	This pin is used to input the interrupt request signal. The level of the interrupt signal can be programmed as either high or low.
V_{REF}	Reference voltage input	In	This is the input for the reference voltage applied to the D-A converter.
CNV_{SS}	CNV_{SS} input	In	This input is connected to V_{SS} and must have a high-level input applied to it (0V).
RESET	Reset input	In	When this input is kept high for at least two machine cycles, the reset state is enabled.
V_P (M58844 XXXSP only)	Pull-down voltage input	In	This pin is used to supply the pull-down voltage for port D outputs and port S outputs.

BASIC FUNCTION BLOCKS

Program Memory (ROM)

This 1024-word x 9-bit mask programmable ROM can be programmed with machine instruction codes in accordance with the customer's specifications. It consists of 8 pages, each containing an address range of 0 ~ 127. Fig. 1 shows the address map of this ROM.

Program Counter (PC)

This counter is used to specify ROM addresses and the sequence of read-out of instructions stored in ROM. The program counter (PC) is an 10-bit binary counter, the upper order 3 bits of which (PC_H) indicate the ROM page, and the lower order 7 bits of which (PC_L) are a pure binary address designation. Each time an instruction is executed, PC_L is incremented by one step. For branching, and subroutine call instructions, its value is set to the designated address.

When the 127th address is reached for every page, the address value returns to the first address of that page. Therefore, for moving from one page to another page, the page byte itself must be modified. This is done using the BL and BLA instructions.

Page 14 and page 15 are special pages used for subroutine calls. The page 14 subroutine can be called with a one word instruction from any arbitrary page. This instruction is either BM or BMA. When either BM or BMA is executed, subsequent BM or BMA instructions are equivalent to B and BA on page 14. Also, B or BA is equivalent to B or BA on page 15. This condition is cancelled when the RT, RTS, BL, BML, BLA or BMLA instruction is executed. Note 3 under the instruction codes shows corresponding states.

Stack Registers (SK_0, SK_1, SK_2)

These registers are used to temporarily store the contents of the PC while executing subroutines or interrupt programs until the program returns to its original routine. The SK registers are organized in three words of 10 bits each, enabling up to three levels of subroutine nesting. If one word is used for an interrupt routine, the remaining two levels can be used for subroutine calls.

Data Memory (RAM)

This 256-bit (64 words x 4 bits) RAM is used to store both processing and control data. One RAM word consists of 4 bits with bit manipulation possible over the entire storage area. The 64 words are arranged as 4 files x 16 digits x 4 bits. Fig. 2 shows the RAM address map.

The RAM address specification is made by the combination of data pointer DP register X, and register Y. Thus, the selector CPS and data pointer DP must be set. However, as long as the address is not changed this is not necessary.

Data Pointers (DP, DP')

These registers are used to designate RAM address, and bit position for the I/O port D and register J. Each data pointer is composed of a 6-bit register group. Register X (the upper order 2 bits of DP) designates a RAM file; and register Y (the lower order 4 bits of DP) designates the digit position of the RAM file. At the same time, register Y designates bit positions of the I/O port D and register J.

4

4-bit Arithmetic Logic Unit (ALU)

This unit executes 4-bit arithmetic and logical operations by means of a 4-bit adder and related logic circuitry. The arithmetic logic unit performs addition, logical comparisons, arithmetic comparisons, and bit manipulation.

Register A and Carry flag (CY)

Register A is a 4-bit accumulator that constitutes the basis for arithmetic operations. Data processing operations such as arithmetic and logical operations, data transfer, exchange, conversion and data input/output are executed by means of this register. Carry or borrow from register is stored in the carry flag's CY and CY' after execution of arithmetic or logical operations. The carry flags CY and CY' can also be used as 1-bit flags. Carry flags and data pointer DP selection is done by means of the selector CPS.

Registers B and E

Register B is composed of 4 bits and can be used as a 4-bit temporary storage register or for 8-bit data transfer in conjunction with register A. Register E is composed of 8 bits and is used not only as an 8-bit temporary storage register, but also as a temporary storage register for I/O port S.

Page designation	8	9	------	15
PC_H \ PC_L	0	1	------	7
Bit designation	8 7 6 5 4 3 2 1 0	8 7 6 5 4 3 2 1 0	8 7 2 1 0	8 7 6 5 4 3 2 1 0
Address designation 0				
1				
2				
⋮	⋮	⋮	⋮	⋮
126				
127				

Note : The M58843-XXXP and M58844-XXXSP programs are developed using a support system having a 2048 word x 9-bit ROM memory. When using such a system pages 8 through 15 of the 2048 words (pages 0 through 15) are used so that the program counter $PC_H = 0 \sim 7$ is defined as pages 8 through 15.

Fig. 1 ROM Address map

File designation	Register X	0				1				2				3			
File name		F_0				F_1				F_2				F_3			
Bit designation		3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0
Digit designation (register Y)	0																
	1																
	2																
	⋮	⋮				⋮				⋮				⋮			
	14																
	15																

Fig. 2 RAM Address map

A/D Conversion Circuit

The following A/D conversion functions are controlled by software as described below. Fig. 3 shows the block diagram.

(1) Comparators

These comparators are implemented entirely with PMOS devices and use a chopper-type amplification method. They are capable of determining the larger of the D-A converter output Vref and the port K input signals $V_{K(Y)}$ (where(Y) = 0 ~ 10).

(2) Register J

Register J is composed of 11 1-bit registers, each representing the comparison result from the comparators. All register bits are set simultaneously. The value of the register J with respect to the comparison results is as follows.

1 when $|Vref| > |V_{K(Y)}|$

0 when $|Vref| < |V_{K(Y)}|$

In this relationship (Y) represents the bit position in register J which is designated by register Y. The comparison results can be checked for each bit using the SZJ instruction.

(3) Registers H – L

These two 4-bit registers are capable of transferring and exchanging data to and from register A.

The 8-bit digital data for the D-A converter is transferred from these registers, the higher order 4 bits from H and lower order 4 bits from L.

(4) Register C

This 3-bit register is used as a counter to designate bit positions in the H and L registers.

(5) D-A Converter

The D-A converter converts the digital values stored in the registers H and L, referencing with the external reference voltage V_{REF} applied at the pin V_{REF}, to the analog value of the internal reference voltage Vref.

The theoretical value of the internal reference voltage Vref is defined as follows.

$$Vref = \frac{n-0.5}{256} \times V_{REF} \text{ , where, n = 1, 2, ... 255}$$

$$Vref = 0 \text{ , where, n = 0}$$

In the above relationships n is the value weighted according to the contents of registers H and L.

A-D Conversion Algorithms

A-D conversion is controlled by the programming of the previously described functional blocks. Thus, by modifying the program, either the successive approximation method or the sequential comparison method may be selected. In addition, a digital input of high or low level may be used to select the method, eliminating software selection of the A-D conversion technique.

(1) Successive Approximation

In this method, the conversion speed is maintained constant regardless of the amplitude of the analog signal. The A-D conversion process requires 0.6ms (at 600KHz clock frequency). 12 programs words are required.

(2) Sequential Comparison

In this method the conversion speed varies in accordance with the rate of change of the analog quantity. When the rate of change is slow, the conversion rate increases. 30 program words are required.

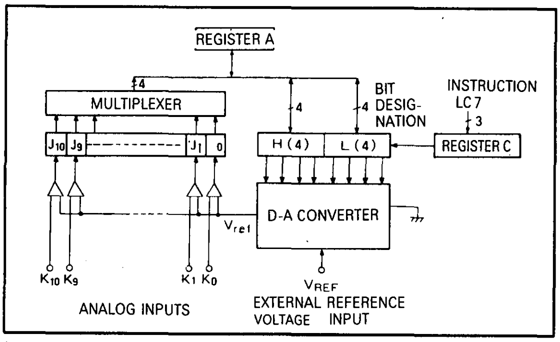


Fig. 3 A-D Conversion circuit block diagram

MITSUBISHI ELECTRIC

Interrupt

The flag INTE is a 1-bit flip-flop used to control interrupt operation. When an interrupt request signal is applied to the pin INT while the interrupt is enabled, the INTE flag is reset to disable further interrupts, after which the program jumps from the main program to address 0 of page 12. When an interrupt program is used, one level of the three-level stack register is required, the remaining two levels being used for subroutines. After the interrupt program is started, the data pointer DP, register A, carry flag CY, and registers used by the interrupt program are saved. It is necessary to restore these before returning to the main program by using the instruction RTI.

When an interrupt occurs, the microcomputer internal states are as follows.

(1) Program Counter
The current address in the main program is stored in a stack register and the program counter is set to page 12, address 0.

(2) Interrupt Flag INTE
The flag INTE is reset to disable further interrupts. This disable state will continue even after the program has returned from the interrupt routine to the main program by the execution of the RTI instruction. EI is executed and when the input level of the INT input changes, this state is disabled. Thus, when the INTH instruction is executed the interrupt state is enabled when the INT input goes high. As long as it remains in the high state, further interrupts are prohibited. If the INT input should change to a low level and return to high, the next interrupt will be accepted.

(3) Skip Flags
Skip flags are provided to discriminate skip instructions and consecutively described skip instructions. Each flag has its own stack within which the skip state is saved.
As a mask option, the interrupt pins may be provided with Schmitt input circuits.

Input/Output Pins

(1) Input port K
The input port K consists of 4 bits for the M58843-XXXP and 11 bits for the M58844-XXXSP. The voltage level input at these pins is compared with the D-A converter voltage output Vref by a comparator and the results stored in register J. As a mask option, it is possible to build load resistors into the input port K. These are implemented using depletion-type MOS transistors. In addition, to enable the use of capacitive touch-type keys, it is possible to provide these inputs with the required discharge transistors.

(2) Input/Output Port S
The input/output port S consists of 8 bits, each bit with an output latch. These latches are used to store data transferred by means of a PLA from register A, or data transferred from register A and register B directly, or data transferred from register E directly. 4 bits at a time of the 8 input bits of port S may be transferred to register A.
Because port S outputs are provided with a built-in PLA, it is possible to output any arbitrarily settable 8-bit code from an input specified by register A. These PLA output codes can be specified arbitrarily as a mask option.
In addition, as a mask option, it is possible to build-in load resistors at the input/output port S. The load resistors are implemented with depletion-type MOS transistors.

(3) Input/Output Port D
The input/output port D consists of 8 bits for the M58843-XXXP and 11 bits for the M58844-XXXSP. Each bit can be individually designated as either input or output and is provided with its own latch. The contents of the data pointer register Y can be used to designate a single bit of port D for output or sensing.
In addition, as a mask option load resistors may be built-in at the input/output port D. These resistors are implemented by means of depletion-type MOS transistors.
When port S or port D is used as an input port, the output should first be cleared to the low state.

Reset Function

When the RESET input is kept high for at least two machine cycles, the reset state is enabled. As shown in Fig. 4, it is possible to implement a power-on reset circuit using an externally connected capacitor, resistor and diode. For this configuration, when the supply voltage falls below −13.5V, the circuit design should insure that the RESET input is above −4V.

When the reset state is enabled, the following operations are performed.

(1) The program counter is set to page 8, address 0
(PC) ← 0

Note 1: The M58843-XXXP and M58844-XXXSP programs are developed using a support system having a 2048 word x 9-bit ROM memory. When using such a system pages 8 through 15 of the 2048 words (pages 0 through 15) are used so that the program counter $PC_H = 0 \sim 7$ is defined as pages 8 through 15.

(2) The interrupt mode is in the interrupt disabled state
(INTE) ← 0
This is the same state as when the instruction DI is executed.

(3) By setting the interrupt request signal INT to high, the interrupt enabled state is entered. This is the same state

as when the instruction INTH is executed.

(4) All outputs of port S are cleared to low (S) ← 0

(5) All outputs of port D are cleared to low (D) ← 0

(6) The carry and data pointer selector CPS is cleared to low to designate DP and CY (CPS) ← 0

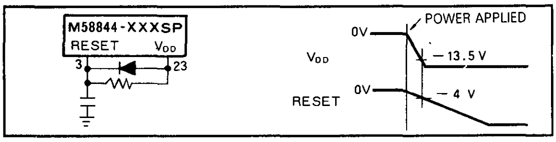
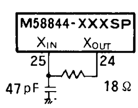


Fig. 4 Power-on reset

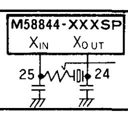
Clock Generator Circuit

A clock generator circuit has been built-in, to allow control of the frequency by means of an externally connected RC circuit or ceramic resonator. In addition, an external clock signal may be applied at the X_{IN} pin, leaving the X_{OUT} pin open. Circuit examples are shown in Fig. 5 to Fig. 7.



Note 2: Since when using an RC circuit the frequency will vary in accordance with the device characteristics, the constants should be chosen so that the frequency falls within a specified limit even including the effect of these variations.

Fig. 5 External RC circuit



Note 3: The circuit constants will depend on the frequency and type of ceramic resonator used.

Fig. 6 Externally connected ceramic resonator

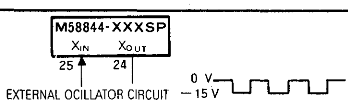


Fig. 7 External clock input circuit

MASK OPTIONS

The following mask options are available, specifiable at the time of initial ordering.

(1) S output PLA data
(2) Interrupt input Schmitt circuit
(3) Port Kinput pull-down resistors
(4) Port K input discharge transistors
(5) Port S input/output pull-down resistors
(6) Port D input/output pull-down resistors

DOCUMENTATION REQUIRED UPON ORDERING

The following information should be provided when ordering a custom mask.

(1) M58843-XXXP, M58844-XXXSP mask confirmation sheet
(2) ROM data — 3 EPROM sets
(3) S output PLA coding — On confirmation sheets
(4) Interrupt input Schmitt circuits — On confirmation sheets
(5) Port K input pull-down resistors — On confirmation sheets
(6) Port K input discharge transistors — On confirmation sheets
(7) Port S input/output pull-down resistors — On confirmation sheets
(8) Port D input/output pull-down resistors — On confirmation sheets

LIST OF INSTRUCTION CODES

$D_3 \sim D_0$ \ $D_8 \sim D_4$	Hexadecimal notation	0 0000 0 0	0 0001 0 1	0 0010 0 2	0 0011 0 3	0 0100 0 4	0 0101 0 5	0 0110 0 6	0 0111 0 7	0 1000 0 8	0 1001 0 9	0 1010 0 A	0 1011 0 B	0 1100 0 C	0 1101 0 D	0 1110 0 E	0 1111 0 F	1 0000 ~ 1 0111 10~17	1 1000 ~ 1 1111 18~1F
0000	0	NOP	CLS	SZB 0	SEY 0	LCPS 0	CPAE	XAM 0	BL BML	—	—	A 0	LA 0	LXY 0, 0	LXY 1, 0	LXY 2, 0	LXY 3, 0	BM	B
0001	1	BA BMA BLA BMLA	CLDS	SZB 1	SEY 1	LCPS 1	CPAS	XAM 1	BL BML	—	—	A 1	LA 1	LXY 0, 1	LXY 1, 1	LXY 2, 1	LXY 3, 1	BM	B
0010	2	INY	—	SZB 2	SEY 2	SHL	RHL	XAM 2	BL BML	—	—	A 2	LA 2	LXY 0, 2	LXY 1, 2	LXY 2, 2	LXY 3, 2	BM	B
0011	3	DEY	CLD	SZB 3	SEY 3	—	—	XAM 3	BL BML	—	—	A 3	LA 3	LXY 0, 3	LXY 1, 3	LXY 2, 3	LXY 3, 3	BM	B
0100	4	DI	RD	—	SEY 4	RT	IAS 0	TAM 0	BL BML	—	—	A 4	LA 4	LXY 0, 4	LXY 1, 4	LXY 2, 4	LXY 3, 4	BM	B
0101	5	EI	SD	—	SEY 5	RTS	IAS 1	TAM 1	BL BML	—	—	A 5	LA 5	LXY 0, 5	LXY 1, 5	LXY 2, 5	LXY 3, 5	BM	B
0110	6	INTH	TEPA	SEAM	SEY 6	RTI	—	TAM 2	BL BML	—	—	A 6	LA 6	LXY 0, 6	LXY 1, 6	LXY 2, 6	LXY 3, 6	BM	B
0111	7	INTL	OSPA	—	SEY 7	—	LC7	TAM 3	BL BML	—	—	A 7	LA 7	LXY 0, 7	LXY 1, 7	LXY 2, 7	LXY 3, 7	BM	B
1000	8	CPA	XAL	—	SEY 8	RC	XAH	XAMD 0	BL BML	—	—	A 8	LA 8	LXY 0, 8	LXY 1, 8	LXY 2, 8	LXY 3, 8	BM	B
1001	9	DEC	TLA	SZJ	SEY 9	SC	THA	XAMD 1	BL BML	—	—	A 9	LA 9	LXY 0, 9	LXY 1, 9	LXY 2, 9	LXY 3, 9	BM	B
1010	A	AM	TEAB	—	SEY 10	—	—	XAMD 2	BL BML	—	—	A 10	LA 10	LXY 0, 10	LXY 1, 10	LXY 2, 10	LXY 3, 10	BM	B
1011	B	OSE	OSAB	SZD	SEY 11	—	—	XAMD 3	BL BML	—	—	A 11	LA 11	LXY 0, 11	LXY 1, 11	LXY 2, 11	LXY 3, 11	BM	B
1100	C	TYA	TBA	—	SEY 12	SB 0	RB 0	XAMI 0	BL BML	—	—	A 12	LA 12	LXY 0, 12	LXY 1, 12	LXY 2, 12	LXY 3, 12	BM	B
1101	D	TAJ	TAY	—	SEY 13	SB 1	RB 1	XAMI 1	BL BML	—	—	A 13	LA 13	LXY 0, 13	LXY 1, 13	LXY 2, 13	LXY 3, 13	BM	B
1110	E	AMC	TAB	—	SEY 14	SB 2	RB 2	XAMI 2	BL BML	—	—	A 14	LA 14	LXY 0, 14	LXY 1, 14	LXY 2, 14	LXY 3, 14	BM	B
1111	F	AMCS	—	SZC	SEY 15	SB 3	RB 3	XAMI 3	BL BML	CMA	—	A 15	LA 15	LXY 0, 15	LXY 1, 15	LXY 2, 15	LXY 3, 15	BM	B

Note 1: This list shows the machine codes and corresponding machine instructions. $D_3 \sim D_0$ indicate the low-order 4 bits of the machine code and $D_8 \sim D_4$ indicate the high-order 5 bits. Hexadecimal numbers are also shown that represent the codes. An instruction may consist of one, two, or three words, but only the first word is listed. Code combination indicated with a bar (-) must not be used.

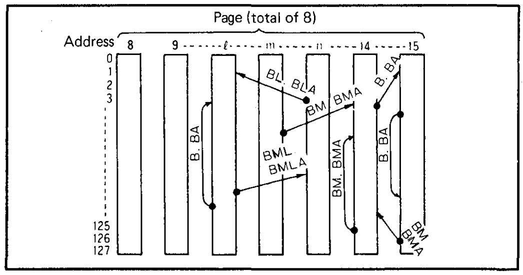
Note 2: Two-word instruction

	Second word	
BL	1	1xxx yyyy
BML	1	0xxx yyyy
BA	1	1xxx XXXX
BMA	1	0xxx XXXX

Three-word instruction

	Second word		Third word	
BLA	0	0111 pppp	1	1xxx XXXX
BMLA	0	0111 pppp	1	0xxx XXXX

Note 3: Page relationships for branching by means of branching instructions and subroutine calling instructions.



MACHINE INSTRUCTIONS

Type of instruction	Mnemonic	Instruction code D_8 $D_7D_6D_5D_4$ $D_3D_2D_1D_0$	16 mal notation	No. of words	No. of cycles	Functions	Skip conditions	Flag CY	Description of operation
Register-to-register transfers	TAB	0 0001 1110	0 1E	1	1	(A)←(B)	–	X	Transfers contents of register B to register A.
	TBA	0 0001 1100	0 1C	1	1	(B)←(A)	–	X	Transfers contents of register A to register B.
	TAY	0 0001 1101	0 1D	1	1	(A)←(Y)	–	X	Transfers contents of register Y to register A.
	TYA	0 0000 1100	0 0C	1	1	(Y)←(A)	–	X	Transfers contents of register A to register Y.
	TEAB	0 0001 1010	0 1A	1	1	$(E_7{\sim}E_4)\leftarrow(B)$, $(E_3{\sim}E_0)\leftarrow(A)$	–	X	Transfers contents of registers A and B to register E.
	TEPA	0 0001 0110	0 16	1	1	$(E_7{\sim}E_0)\leftarrow$ through PLA←(A)	–	X	Docodes contents of register A in the PLA and trasfers result to register E.
RAM addresses	LXY x,y	0 11xx yyyy	0 Cy + x	1	1	(X)←x, where x=0∼3 (Y)←y, where, y=0∼15	Written successively	X	Loads value of "x" into register X and of "y" into Y. When LXY is written successively, the first is executed and successive ones are skipped
	INY	0 0000 0010	0 02	1	1	(Y)←(Y)+1	(Y)=0	X	Increments contents of register Y by 1. Skips next instruction when new contents of register Y are "0"
	DEY	0 0000 0011	0 03	1	1	(Y)←(Y)−1	(Y)=15	X	Decrements contents of register Y by 1. Skips next instruction when new contents of register Y are "15".
	LCPS i	0 0100 000i	0 4i	1	1	(CPS)←i, where, i=0, 1	–	X	DP and CY are active when i=O, DP' and CY' when i=1.
RAM-accumulator transfers	TAM j	0 0110 01jj	0 64 + j	1	1	(A)←(M(DP)) (X)←(X)∀j, where, j=0∼3	–	X	Transfer the RAM contents addressed by the active DP to register A. Register X is then "exclusive OR-ed" with the value j in the instruction, and the result stored in register X.
	XAM j	0 0110 00jj	0 6j	1	1	(A)↔(M(DP)) (X)←(X)∀j, where, j=0∼3	–	X	Exchanges the contents of the RAM DP and register A. Contents of X are then "exclusive OR-ed" with the valuej and the result stored in register X.
	XAMD j	0 0110 10jj	0 68 + j	1	1	(A)↔(M(DP)), (Y)←(Y)−1 (X)←(X)∀j, where, j=0∼3	(Y)=15	X	Exchanges the contents of the RAM and register A. Contents of X are then "exclusive OR-ed" with the value j in the instruction, and the result stored in register X. The contents of register Y are decremented by 1, and when the result is 15, the next instruction is skipped.
	XAMI j	0 0110 11jj	0 6C + j	1	1	(A)↔(M(DP)), (Y)←(Y)+1 (X)←(X)∀j, where, j=0∼3	(Y)=0	X	Exchanges the contents of the RAM and register A. Contents of X are then "exclusive OR-ed" with the value j in the instruction and result stored in register X.The contents of register Y are when the incremented by 1, and when the result is O, the next instruction is skipped.
Arithmetic operations	LA n	0 1011 nnnn	0 Bn	1	1	(A)←n, where, n=0∼15	Written successively	X	Loads the value n into register A. When LA is written consecutively the first is executed, and successive ones are skipped.
	AM	0 0000 1010	0 0A	1	1	(A)←(A)+(M(DP))	–	X	Adds the contents of the RAM to register A. The result is retained in register A, and the contents of flag CY are unaffected.
	AMC	0 0000 1110	0 0E	1	1	(A)←(A)+(M(DP))+(CY) (CY)← carry	–	0/1	Adds the RAM contents addressed by the active DP and contents of flag CY to register A. The result is stored in register A, and the carry in the active flag CY.
	AMCS	0 0000 1111	0 0F	1	1	(A)←(A)+(M(DP))+(CY) (CY)← carry	(CY)=1 A carry is not produced and	0/1	Adds the contents of the RAM and flag CY to register A. The result is stored in register A and the carry in the CY, but the next instruction is skipped when a carry is produced.
	A n	0 1010 nnnn	0 An	1	1	(A)←(A)+n, where, n=0∼15	=0 n≠6	X	Adds value n in the instruction to register A. The contents of flag CY are unaffected and their next instruction is skipped if a carry is not produced, except when n=6.
	SC	0 0100 1001	0 49	1	1	(CY)←1	–	1	Sets active flag CY.
	RC	0 0100 1000	0 48	1	1	(CY)←0	–	0	Resets active flag CY.
	SZC	0 0010 1111	0 2F	1	1	…	(CY)=0	X	Skips next instruction when contents of the active flag CY are O.
	CMA	0 1000 1111	0 8F	1	1	$(A)\leftarrow(\overline{A})$	–	X	Stores complement of register A in register A.
Bit operations	SB j	0 0100 11jj	0 4C + j	1	1	(Mj(DP))←1, where, j=0∼3	–	X	Sets the jth bit of the RAM addressed by the active DP (the bit designated by the value j in the instruction)
	RB j	0 0101 11jj	0 5C + j	1	1	(Mj(DP))←0, where, j=0∼3	–	X	Resets the jth bit of the RAM addressed by the active DP (the bit designated by the value j in the instruction).
	SZB j	0 0010 00jj	0 2j	1	1		(Mj(DP)) =0 where, j=0∼3	X	Skips next instruction when the contents of the jth bit of the RAM addressed by the active DP (the bit which is designated by the value j in the instruction) are O.
Compares	SEAM	0 0010 0110	0 26	1	1		(M(DP))= (A)	X	Skips next instruction when contents of register A are equal to the RAM contents addressed by the active DP.
	SEY y	0 0011 yyyy	0 3y	1	1		(Y)=y where, y=0∼15	X	Skips next instruvtion. when the contents of register Y are equal to the value y in the instruction.
A/D converter operations	TLA	0 0001 1001	0 19	1	1	(L)←(A)	–	X	Transfers contents of register A to register L.
	THA	0 0101 1001	0 59	1	1	(H)←(A)	–	X	Transfers contents of register A to register H.
	TAJ	0 0000 1101	0 0D	1	1	$(Y_1Y_0)=0$ when, : $(A)\leftarrow(J_3J_2J_1J_0)$ $(Y_1Y_0)=1$ when, : $(A)\leftarrow(J_7J_6J_5J_4)$ $(Y_1Y_0)=2$ when, : $(A)\leftarrow(0\,J_{10}J_9J_8)$ $(Y_1Y_0)=3$ when, : $(A)\leftarrow(0\,0\,0\,0)$	–	X	Transfers designated contents of register J to register A.
	XAL	0 0001 1000	0 18	1	1	(A)↔(L)	–	X	Exchanges contents of register A with contents of register L.
	XAH	0 0101 1000	0 58	1	1	(A)↔(H)	–	X	Exchanges contents of register A with contents of register H.
	LC7	0 0101 0111	0 57	1	1	(C)←7		X	Loads 7 to register C.
	DEC	0 0000 1001	0 09	1	1	(C)←(C)−1	(C)=7	X	Decrements contents of register C by 1, when result is 7, skips next.
	SHL	0 0100 0010	0 42	1	1	$(C_2)=1$ when, : $(H_{(C_1-C_0)})\leftarrow1$ $(C_2)=0$ when, : $(L_{(C_1-C_0)})\leftarrow1$	–	X	Sets the bit in register L or H designated by register C. The box instruction shows the relationship between register C and bit position. (C): 7 6 5 4 3 2 1 0 / Bit: H_3 H_2 H_1 H_0 L_3 L_2 L_1 L_0
	RHL	0 0101 0010	0 52	1	1	$(C_2)=1$ when, : $(H_{(C_1-C_0)})\leftarrow0$ $(C_2)=0$ when, : $(L_{(C_1-C_0)})\leftarrow0$	–	X	Resets the bit in register L or H that is designated by register C.
	CPA	0 0000 1000	0 08	1	1 (2)	$\lvert V_{ref}\rvert>\lvert V_{K(i)}\rvert$ when, : $(J_{(i)})\leftarrow1$ $\lvert V_{ref}\rvert<\lvert V_{K(i)}\rvert$ when, : $(J_{(i)})\leftarrow0$ i=0∼10		X	Reads all analog values from input port K for comparison with D-A converter output V_{ref} and either sets the respective bit of register J to the next instruction cycle wherever $V_{ref} < V_{K(1)}$ is true, or resets it wherever $V_{ref} < V_{K(1)}$ is true.
	CPAS	0 0101 0001	0 51	1	1	$\lvert V_{ref}\rvert>\lvert V_{K(i)}\rvert$ when, : $(J_{(i)})\leftarrow1$ $\lvert V_{ref}\rvert<\lvert V_{K(i)}\rvert$ when, : $(J_{(i)})\leftarrow0$ i=0∼10	–	X	Reads and stores temporarily all analog values from input port K, which are then unaffected by changes in port K inputs. These values are compared with the D-A converter output V_{ref}, calculated from contents of registers H and L and respective bits of register J are set/reset. Repeated when contents of registers H-L are changed.
	CPAE	0 0101 0000	0 50	1	1	Execution of the instruction CPAS is over, and no more changes will made in $(J_{(Y)})$.		X	Terminates execution of instruction CPAS. Contents of register J remain unaffected, maintaining the value immediately before termination, and input port K is again ready to receive inputs.
	SZJ	0 0010 1001	0 29	1	1		$(J_{(Y)})=0$	X	Skips next instruction when the bit in register J, designated by register Y, is 0.

MITSUBISHI ELECTRIC

Type of instruction	Mnemonic	Instruction code D_8 $D_7D_6D_5D_4$ $D_3D_2D_1D_0$	16mal notation	No. of words	No. of cycles	Functions	Skip conditions	Flag CY	Description of operation
Jumps	B xy	1 1xxx yyyy	1 8y + x	1	1	$(PC_L)\leftarrow 16x+y$ $(PC_H)\leftarrow 7$, $(PC_L)\leftarrow 16x+y$	—	X	Jumps to address xy of the current page. Jumps to address xy on page 15 when executed, provided that none of instruction RT, RTS, BL, BML, BLA or BMLA was executed after execution of instruction BM or BMA.
	BL pxy	0 0111 pppp 1 1xxx yyyy	0 7p 1 8y + x	2	2	$(PC_H)\leftarrow p$ $(PC_L)\leftarrow 16x+y$	—	X	Jumps to address xy of page p.
	BA xX	0 0000 0001 1 1xxx XXXX	0 01 1 8X + x	2	2	$(PC_L)\leftarrow 16x+(A)$ $(PC_H)\leftarrow 7$, $(PC_L)\leftarrow 16x+(A)$	—	X	Jumps to address x(A) of the current page. Jumps to the address x(A) of page 15 provided that none of instructions. RT, RTS, BL, BML, BLA or BMLA was executed after execution of instruction BM or BMA.
	BLA pxX	0 0000 0001 0 0111 pppp 1 1xxx XXXX	0 01 0 7p 1 8X + x	3	3	$(PC_H)\leftarrow p$ $(PC_L)\leftarrow 16x+(A)$	—	X	Jumps to the address x(A) of page p.
Subroutine calls	BM xy	1 0xxx yyyy	1 xy	1	1	$(SK2)\leftarrow(SK1)\leftarrow(SK0)\leftarrow(PC)$ $(PC_H)\leftarrow 6$, $(PC_L)\leftarrow 16x+y$ $(PC_H)\leftarrow 6$, $(PC_L)\leftarrow 16x+y$	—	X	Calls for the subroutine starting at address xy on page 14. Jumps to address xy of page 14 provided that none of instructions. RT, RTS, BL, BML, BLA or BMLA was executed after the execution of instructions BM or BMA.
	BML pxy	0 0111 pppp 1 0xxx yyyy	0 7p 1 xy	2	2	$(SK2)\leftarrow(SK1)\leftarrow(SK0)\leftarrow(PC)$ $(PC_H)\leftarrow p$, $(PC_L)\leftarrow 16x+y$	—	X	Calls for the subroutine starting at address xy of page p.
	BMA xX	0 0000 0001 1 0xxx XXXX	0 01 1 xX	2	2	$(SK2)\leftarrow(SK1)\leftarrow(SK0)\leftarrow(PC)$ $(PC_H)\leftarrow 6$, $(PC_L)\leftarrow 16x+(A)$ $(PC_H)\leftarrow 6$, $(PC_L)\leftarrow 16x+(A)$	—	X	Calls for the subroutine starting at address x(A) of page 14. Jumps to address xy of page 14 provided that none of instructions. RT, RTS, BL, BML, BLA or BMLA was executed after the execution of instructions BM or BMA.
	BMLA pxX	0 0000 0001 0 0111 pppp 1 0xxx XXXX	0 01 0 7p 1 xX	3	3	$(SK2)\leftarrow(SK1)\leftarrow(SK0)\leftarrow(PC)$ $(PC_H)\leftarrow p$, $(PC_L)\leftarrow 16x+(A)$	—	X	Calls for the subroutine starting at address x(A) of page p.
Program returns	RTI	0 0100 0110	0 46	1	1	$(PC)\leftarrow(SK0)\leftarrow(SK1)\leftarrow(SK2)$ Restore internal flip-flop	—	X	Returns from interrupt routine to main routine. The internal flip-flop is restored to the value held immediately before the inter rupt.
	RT	0 0100 0100	0 44	1	1	$(PC)\leftarrow(SK0)\leftarrow(SK1)\leftarrow(SK2)$	—	X	Returns to the main routine from the subroutine.
	RTS	0 0100 0101	0 45	1	2	$(PC)\leftarrow(SK0)\leftarrow(SK1)\leftarrow(SK2)$	Unconditional skip	X	Returns to the main routine from the subroutine, and uncidi-tionally skips the next instruction.
Input/output	SD	0 0001 0101	0 15	1	1	$(D(Y))\leftarrow 1$, where, $0\leqq(Y)\leqq 10$	—	X	Sets the bit of port D that is designated by register Y.
	RD	0 0001 0100	0 14	1	1	$(D(Y))\leftarrow 0$, where, $0\leqq(Y)\leqq 10$	—	X	Resets the bit of port D that is designated by register Y.
	SZD	0 0010 1011	0 2B	1	1	where, $0\leqq(Y)\leqq 10$	$(D(Y))=0$	X	Skip the next instruction if the contents of the bit of port D that is designated by register Y are o.
	OSAB	0 0001 1011	0 1B	1	1	$(S_7\sim S_4)\leftarrow(B)$, $(S_3\sim S_0)\leftarrow(A)$		X	Outputs contents of registers A and B to port S.
	OSPA	0 0001 0111	0 17	1	1	$(S_7\sim S_0)\leftarrow$ through PLA$\leftarrow(A)$		X	Decodes contents of register A by PLA and the result is output to port.
	OSE	0 0000 1011	0 0B	1	1	$(S)\leftarrow(E)$		X	Outputs contents of register E to port S.
	IAS i	0 0101 010i	0 54 + i	1	1	$i=0$: $(A)\leftarrow(S_7\sim S_4)$ $i=1$: $(A)\leftarrow(S_3\sim S_0)$		X	Transfers from port S to register A. The high-order four bit of port S are transferred when the value of i in the instruction is O or the low-order four bits are transferred when the value of i is 1.
	CLD	0 0001 0011	0 13	1	1	$(D)\leftarrow 0$		X	Clears port D.
	CLS	0 0001 0000	0 10	1	1	$(S)\leftarrow 0$		X	Clears port S.
	CLDS	0 0001 0001	0 11	1	1	$(D)\leftarrow 0$, $(S)\leftarrow 0$		X	Clears ports S and D.
Interrupts	EI	0 0000 0101	0 05	1	1	$(INTE)\leftarrow 1$		X	Sets interrupt flag INTE to enable interrupts.
	DI	0 0000 0100	0 04	1	1	$(INTE)\leftarrow 0$		X	Resets interrupt flag INTE to disable interrupts.
	INTH	0 0000 0110	0 06	1	1	$(INTP)\leftarrow 1$		X	Sets interrupt polarity flag INTP to enable interrupts when the interrupt request signal is turned high.
	INTL	0 0000 0111	0 07	1	1	$(INTP)\leftarrow 0$		X	Resets interrupt polarity flag INTP to enable interrupts when the interrupt request signal is turned low.
Misc	NOP	0 0000 0000	0 00	1	1	$(PC)\leftarrow(PC)+1$		X	No operation
Pin	RESET					$(PC_H)\leftarrow 0$, $(PC_L)\leftarrow 0$			Start from address "o" of page 8.
	INT					$(SK_2)\leftarrow(SK_1)\leftarrow(SK_0)\leftarrow(PC)$, $(PC_H)\leftarrow 4$ $(PC_L)\leftarrow 0$			Calls for the subroutine starting at address "0" of page 12.

Symbol	Contents	Symbol	Contents	Symbol	Contents
A	4-bit register (accumulator)	SK1	10-bit stack register	INT	Interrupt request signal.
B	4-bit register	SK2	10-bit stack register	←	Shows direction of data flow.
C	3-bit register	CY	1-bit carry flag	( )	Indicates contents of the register, memory, etc.
E	8-bit register	xx	2-bit binary variable	⊻	Exclusive OR
H	4-bit register	yyyy	4-bit binary variable	–	Negation.
J	1-bit register	nnnn	4-bit binary constant	X	Indicates flag is unaffected by instruction execution.
L	4-bit register	i	1-bit binary constant	xy	Label used to indicate the address xxxyyyy
X	2-bit register	i j	2-bit binary constant	pxy	Label used to indicate the address xxxyyyy of page pppp.
Y	4-bit register	XXXX	4-bit unknown binary number	CPS	Indicates which data pointer and carry are active.
DP	6-bit data pointer, combination of registers X and Y.	D	11-bit port	C	Hexadecimal number C + binary number x.
PC_H	The high-order three bits of the program counter.	K	11-bit port	+	
PC_L	The low-order seven bits of the program counter.	S	8-bit port	×	
PC	10-bit program counter, combination of PC_H and PC_L.	INTE	Interrupt enable flag		
SK0	10-bit stack register	INTP	Interrupt polarity flag		

Note 1. When a skip is used with either the M58843-XXXP or M58843-XXXP, the next instruction becomes invalid and the program counter is not incremented by 2. Therefore the number of cycles does not change in accordance with the existence or non-existence of a a skip.In addition, since the M58843-XXXP is housed in a 28-pin package, some pins of the port K and D are not usable.

2. The M58843-XXXP and M58844-XXXSP programs are developed using a support system having a 2048 word × 9-bit ROM memory. When using such a system, page 8 through 15 of the 2048 words (page 0 through 15) are used so that the program counter $PC_H=0\sim 7$ is defined as page 8 through 15.

SINGLE-CHIP 4-BIT MICROCOMPUTER WITH 8-BIT A/D CONVERTER

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	0.3~−20	V
V_I	Input voltage, port S and D, X_{IN} and V_P inputs		0.3~−35	V
V_I	Input voltage, other than port S and D, X_{IN} and V_P inputs		0.3~−20	V
V_O	Output voltage, ports S and D		0.3~−35	V
V_O	Output voltage, other than ports S and D		0.3~−20	V
P_d	Power dissipation	Ta=25℃	1100	mW
Topr	Operating temperature		−10~70	℃
Tstg	Storage temperature		−40~125	℃

RECOMMENDED OPERATING CONDITIONS (Ta = -10~70°C, unless otherwise noted)

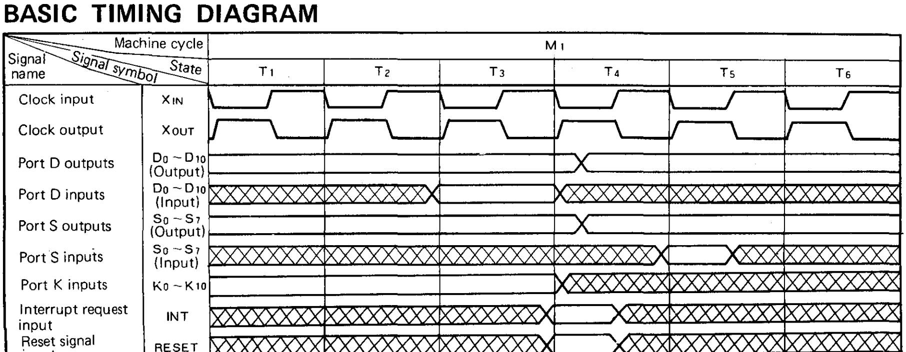
Symbol	Parameter	Limits Min	Limits Nom	Limits Max	Unit
V_{DD}	Supply voltage	−13.5	−15	−16.5	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage, port D	−1		0	V
V_{IH}	High-level input voltage other than port D	−1.5		0	V
$V_{IH(\phi)}$	High-level clock input voltage	−1.5		0	V
V_{IL}	Low-level input voltage, RESET and INT (Schmitt)	V_{DD}		$V_{DD}+2$	V
V_{IL}	Low-level input voltage, INT (TTL compatible)	V_{DD}		−4.2	V
V_{IL}	Low-level input voltage, ports D and S	−33		−4.2	V
$V_{IL(\phi)}$	Low-level clock input voltage	−33		$V_{DD}+2$	V
$V_{I(K)}$	Digital input voltage, port K	V_{DD}		0	V
$V_{I(K)}$	Analog input voltage, port K	V_{REF}		0	V
V_{REF}	Reference voltage	−5		−7	V
V_{OL}	Low-level output voltage, ports D and S	−33		0	V
f(φ)	Internal clock oscillation frequency	300		600	kHz

ELECTRICAL CHARACTERISTICS (Ta = -10~70°C, VDD= -15V ±10%, VSS=0V, f(φ)=300~600kHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
V_{T-}	Negative threshold voltage, RESET input	$V_{DD}=-15V$, $T_a=25℃$	$V_{DD}+2$		−4	V
$V_{T+}-V_{T-}$	Hysteresis, RESET input	$V_{DD}=-15V$, $T_a=25℃$		1		V
V_{OH}	High-level output voltage, port D	$V_{DD}=-15V$, $I_{OH}=-15mA$,	−2.5			V
V_{OH}	High-level output voltage, port S	$V_{DD}=-15V$, $I_{OH}=-8mA$,	−2.5			V
I_{IH}	High-level input current, port K (with pull-down resistors)	$V_{DD}=-15V$, $V_{IH}=0V$, $T_a=25℃$	50		250	μA
I_{IH}	High-level input current, ports D and S (with pull-down resistors)	$V_P=-33V$, $V_{IH}=0V$, $T_a=25℃$	80		280	μA
I_I	Input current, port K	To be measured when the instruction CPAS or CPA is not being executed; V_I = -7V		−1	−7	μA
$I_{I(\phi)}$	Clock input current	$V_{I(\phi)}=-33V$, $T_a=25℃$		−20	−40	μA
I_{OH}	High-level output current, port D	$V_{DD}=-15V$, $V_{OH}=-2.5V$,			−15	mA
I_{OH}	High-level output current, port S	$V_{DD}=-15V$, $V_{OH}=-2.5V$,			−8	mA
I_{OL}	Low-level output current, ports D and S	$V_{OL}=-33V$, $T_a=25℃$			−33	μA
I_{DD}	Supply current	$V_{DD}=-15V$, $T_a=25℃$		−27	−41	mA
I_{REF}	Reference current	$V_{REF}=-7V$, $T_a=25℃$			−1	mA
I_P	Pull-down supply current	$V_P=-33$, $T_a=25℃$			−5.5	mA
C_i	Input capacitance, port K	$V_{DD}=V_I=V_0=V_{SS}$, f=1MHz 25mVrms		7	10	pF
$C_{i(\phi)}$	Clock input capacitance	$V_{DD}=X_{OUT}=V_{SS}$, f=1MHz 25mVrms		7	10	pF
	A-D conversion linearity error	$V_{REF}=-7V$	Overall	±2	±3	LSB
	A-D conversion zero error					
	A-D conversion fullscale error					

Note 1. Currents are taken as positive when flowing into the IC (zero-signal conditions) with the minimum and maximum values as absolute values.
2. The overall sum of the port D high-level output currents should be kept below 75mA.

BASIC TIMING DIAGRAM



Note 3. The crosshatched area indicates invalid input.

MITSUBISHI ELECTRIC

SINGLE-CHIP 4-BIT MICROCOMPUTER WITH 8-BIT A/D CONVERTER AND TWO TIMER/EVENT COUNTER

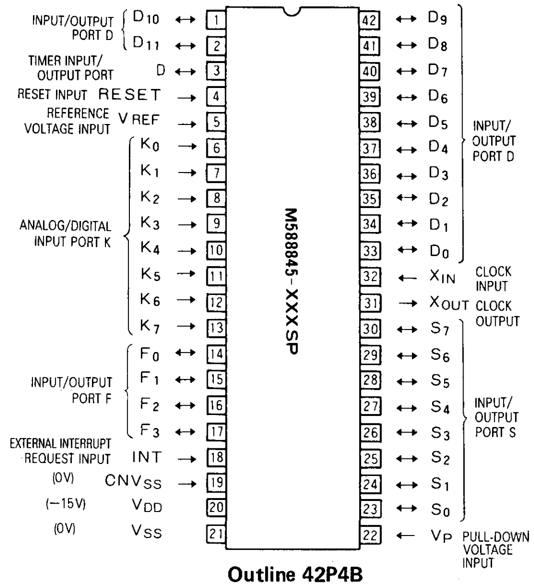
DESCRIPTION

The M58845-XXXSP is a single-chip 4-bit microcomputer developed using p-channel aluminum gate ED-MOS technology. The device includes an 8-bit A-D converter and two timers (one 8-bit timer/counter and one 8-bit timer/event counter). It is housed in a 42-pin shrink plastic molded DIL package.

FEATURES

- Basic machine instructions 77
- Basic instruction execution time (1-word instruction at a clock frequency of 600kHz) 10μs
- Memory capacity ROM: ...... 2048 words x 9 bits
  RAM: ...... 128 words x 4 bits
- Single −15V power supply
- Built-in 8-bit A-D converter (12 analog inputs)
- Two built-in timers (timer 1: 8-bit timer/counter, timer 2: 8-bit timer/event counter, 7-bit prescaler, timer input/output port T) .... 2 lines
- Interrupt function
  .... 3 factors (external, timer 1, timer 2), 1 level
- Two built-in data pointers
- Subroutine nesting .... 3 levels
- Analog/digital inputs (port K) .... 8 ports
- Input/output (ports D, F, and S) .... 24 ports
- Timer input/outputs (port T) .... 1 port
- Direct drive for large fluorescent display tubes is possible
- Built-in decoder PLA for port S outputs (mask option)
- Built-in pull-down transistors (ports D, K, and S mask option)
- Built-in clock generator circuit

PIN CONFIGURATION (TOP VIEW)

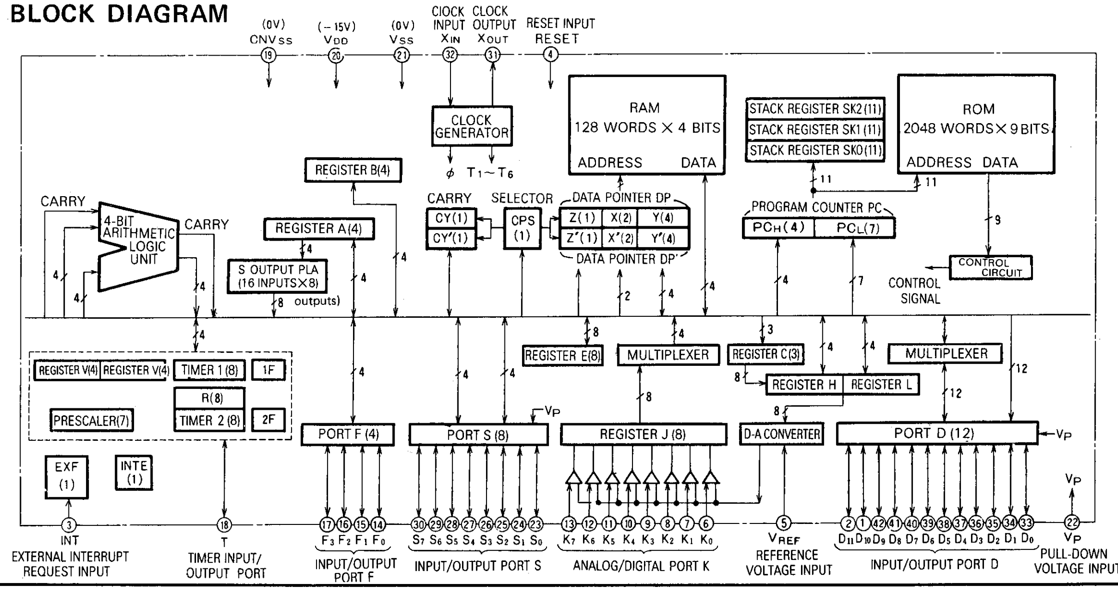


Outline 42P4B

APPLICATIONS

- Microwave ovens, air conditioners, heaters, home sewing machines
- Office equipment, copying machines, medical equipment
- VTR, TVs, cassette decks
- Educational equipment, electronic games

BLOCK DIAGRAM



PERFORMANCE SPECIFICATIONS

Parameter			Performance
Basic machine instructions			77
Instruction execution time (1-word instructions)			10μs (with a clock frequency of 600kHz)
Clock frequency			300 ~ 600kHz
Memory capacity	ROM		2048 words x 9 bits
	RAM		128 words x 4 bits
Input/output ports, and interrupt request inputs (34 lines)	K(Note 1)	Input	1 bit x 8 or 4 bits x 2 (analog/digital)
	D(Note 2)	Input	1 bit x 12
		Output	1 bit x 12
	F	Input	4 bits x 1
		Output	4 bits x 1
	S(Note 2)	Input	4 bits x 2
		Output	8 bits x 1
	T (Note 3)	Input	1 bit x 1
		Output	1 bit x 1
	INT (external interrupt request)(Note 3)		1 bit x 1
A-D conversion circuit			Built-in (accuracy±2LSB)
Timers (2)			Timer 1: 8-bit timer/counter Timer 2: 8-bit timer/event counter 7-bit prescaler, timer input/output port
Pull-down voltage input pin			Used for driving devices such as large fluorescent display tubes (ports D and S)
Subroutine nesting			3 levels
Interrupts			3 factors (external, timer 1, timer 2), 1 level
Clock generator			Built-in (for use with externally connected RC circuit or ceramic resonator)
I/O characteristics of ports	Port D		−33V input/output withstanding voltage, output current −15mA
	Port S		−33V input/output withstanding voltage, output current −8mA
	Ports other than D and S		−20V input/output withstanding voltage, output current −6mA
Supply voltage			−15V (typ)
Device structure			p-channel aluminum gate ED-MOS
Package			42-pin silicon plastic molded DIL package
Power dissipation (excluding ports)			350mW (typ)

Note 1. Built-in pull-down transistors and discharge transistors (mask options)
2. Built-in pull-down transistors (mask option)
3. Input characteristics mask option (TTL compatible, with a Schmitt circuit)

PIN DESCRIPTION

Pin	Name	Input or output	Function
V_{SS}	Ground		Connected to 0V potential
V_{DD}	Supply voltage		Connected to a −15V supply
V_P	Pull-down supply	In	Input for the supply voltage connected to the load resistors (mask option) for ports D and S
$K_7 \sim K_0$	I/O port K	In	This port can be used for analog and digital input, acting as 8 individual bit inputs or 2 4-bit input groups. Pull-down transistors and input discharge transistors are available as mask options.
$D_{11} \sim D_0$	I/O port D	In/out	Port D consists of a 12-bit input/output port, all bits operating individually. When a port D output is programmed low, the output floats and the input signal can be sensed. The outputs are open drain circuits which can be provided with pull-down transistors as a mask option.
$F_3 \sim F_0$	I/O port F	In/out	Port F is a 4-bit input/output port. When the output is programmed to low, the output floats and the input signal can be sensed. The output circuits are open drain circuits.
$S_7 \sim S_0$	I/O port S	In/out	The I/O port S can be used as either an 8-bit output port or a pair of 4-bit input ports. When the output port S is programmed to the low level, it remains in the floating state so that it can be used as an input port.
T	Timer I/O port T	In/out	This port is used as the timer to event counter input, and the timer to overflow output, the function being software selectable.
INT	Interrupt request input	In	This is the input for interrupt requests.
RESET	Reset	In	When this input is kept high for at least 3 machine cycles, the reset state is enabled.
V_{REF}	Reference voltage input	In	This is the input for the reference voltage required by the D-A converter.
X_{IN}	Clock input	In	These are the input and output pins for the built-in clock generator. A ceramic resonator (300 kHz ~ 600 kHz) or a resistor/capacitor combination are connected to these pins to provide the required oscillation stability.
X_{OUT}	Clock output	Out	
CNV_{SS}	CNV_{SS}	In	This input is connected to V_{SS} and must have a high-level input applied to it (0V).

4

BASIC FUNCTION BLOCKS

Program Memory (ROM)

This 2048-word x 9-bit ROM can be programmed with machine instruction codes in accordance with the customer's specifications. It consists of 16 pages, each containing an address range of 0~127. Fig. 1 shows the address map for this ROM.

Program Counter (PC)

This counter is used to specify ROM addresses and the sequence of read-out of instructions stored in ROM. The program counter is an 11-bit counter, the upper order 4 bits of which (PC_H) indicate the ROM page, and the lower 7 bits of which are a pure binary address designation. Each time an instruction is executed, PC_L is incremented by 1 step. For branching and subroutine call instructions, its value is set to the designated address.

When the 127 address is reached for every page, the address value returns to the first address of that page. Therefore, for moving from one page to another page, the page byte itself must be modified. This is done using the **BL** and **BLA** instructions.

Page 2 and page 3 are special pages used for subroutine calls. Page 2 can be called with a 1-word instruction from any arbitrary page. This instruction is either **BM** or **BMA**. When either **BM** or **BMA** is executed, subsequent **BM** or **BMA** instructions are equivalent to **B** and **BA** on page 2. Also, **B** or **BA** is equivalent to **B** or **BA** on page 3. This condition is cancelled when the **RT, RTS, BL, BML, BLA,** or **BMLA** instruction is executed. Table 3 shows the instruction codes and corresponding states.

Stack Registers (SK_0, SK_1, SK_2)

These registers are used to temporarily store the contents of the PC while executing subroutines or interrupt programs until the program returns to its original routine. The SK registers are organized in 3 words of 11 bits each, enabling up to 3 levels of subroutine nesting. If 1 level is used for an interrupt routine, the remaining 2 levels can be used for subroutine calls.

Data Memory (RAM)

This 512-bit (128 words x 4 bits) RAM is used to store both processing and control data. One RAM word consists of 4 bits with bit manipulation possible over the entire storage area. The 128 words are arranged as 2 file groups x 4 files x 16 digits x 4 bits. Fig. 2 shows the RAM address map. The RAM address specification is made by the combination of data pointer DP register Z, register X, and register Y. Thus, the selector CPS and data pointer DP must be set. However, as long as the address is not changed this is not necessary.

Data Pointers (DP, DP')

These registers are used to designate the RAM address, and bit position for the I/O port D and register J. Each data pointer is composed of a 7-bit register. Register Z (the most significant bit of DP) designates the RAM file group; register X (the central 2 bits) designates the RAM file; and register Y (the least significant 4 bits) designates the digit position of the RAM file. At the same time, register Y designates the bit positions of the I/O port D and register J.

4-Bit Arithmetic Logic Unit (ALU)

This unit executes 4-bit arithmetic and logical operations by means of a 4-bit adder and related logic circuitry.

PC_H \ PC_L	Page designation 0	1	···	15
Bit designation	8 7 6 5 4 3 2 1 0	8 7 6 5 4 3 2 1 0	8 7 6 2 1 0	8 7 6 5 4 3 2 1 0
Address designation 0				
1				
2				
⋮	⋮	⋮	⋮	⋮
126				
127				

Fig. 1 ROM Address map

File designation: Register Z	0				1		
Register X	0	1	2	3	0	···	3
File name	F_0	F_1	F_2	F_4	F_4	···	F_7
Bit designation	3 2 1 0	3 2 1 0	3 2 1 0	3 2 1 0	3 2 1 0	···	3 2 1 0
Digit designation (register Y) 0							
1							
2							
⋮	⋮	⋮	⋮	⋮	⋮		⋮
14							
15							

Fig. 2 RAM Address map

Register A and Carry Flag (CY)

Register A is a 4-bit accumulator that constitutes the basis for arithmetic operations. Data processing operations such as arithmetic and logical operations, data transfer, exchange, conversion, and data input/output are executed by means of this register. The carry flag CY is used to store carry or overflow after execution of arithmetic and logical operations by the arithmetic logic unit. The carry flag may also be used as a 1-bit flag. Two carry flags, CY and CY', are available and selected by selector CPS, as is the data pointer DP.

Registers B and E

Register B is composed of 4 bits and can be used as a 4-bit temporary storage register or for 8-bit data transfer in conjunction with register A. Register E is composed of 8 bits and is used not only as an 8-bit temporary storage register, but also as a temporary for the I/O port S.

A/D Conversion Circuit

The following A-D conversion functions are controlled by software as described below.

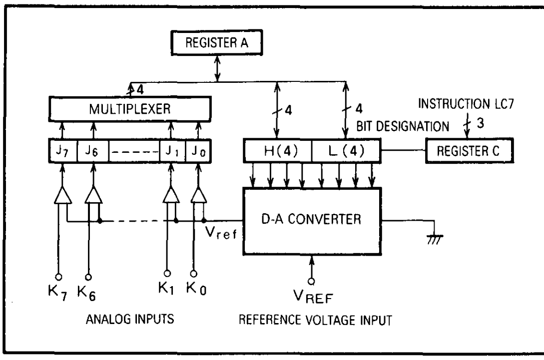


Fig. 3 A-D conversion circuit block diagram

(1) Comparators
The comparators are implemented entirely with PMOS devices and use a chopper-type amplification method. They are capable of determining the larger of the D-A converter output V_{ref} and the port K input signals $V_K(Y)$ (where (Y)=0~7).

(2) Register J
Register J is composed of 8 1-bit registers, each representing the comparison result from the comparators. All register bits are set simultaneously. The value of the register J with respect to the comparison results is as follows.

1 when $|V_{ref}| > |V_K(Y)|$
0 when $|V_{ref}| < |V_K(Y)|$

In this relationship Y represents the bit position in register J which is designated by register Y. The comparison results can be checked for each bit using the SZJ instruction.

(3) Registers H and L
These two 4-bit registers are capable of transferring and exchanging data to and from register A. The 8-bit digital data for the D-A converter is transferred from these registers, the higher order 4 bits from H and the lower order 4 bits from L.

(4) Register C
This 3-bit register is used as a counter to designate bit positions in the H and L registers.

(5) D-A Converter
The D-A converter converts the digital values stored in the registers H and L, referencing with the external reference voltage V_{REF} applied at the pin V_{REF}, to the analog value of the internal reference voltage V_{ref}. The theoretical value of the internal reference voltage V_{ref} if defined as follows.

$$Vref = \frac{n-0.5}{256} \times V_{REF}, \text{ where, } n = 1, 2, \ldots\ldots\ldots 255$$

$$Vref = 0 \qquad , \text{ where, } n = 0$$

In the above relationships n is the value weighted according to the contents of registers H and L.

A/D Conversion Algorithms

A/D conversion is controlled by the programming of the previously described functional blocks. Thus, by modifying the program, either the successive approximation method or the sequential comparision method may be selected. In addition, a digital input of high or low level may be used to select the method, eliminating the software selection of the A/D conversion technique.

(1) Successive Approximation Method
In this method, the conversion speed is maintained at a constant 600kHz regardless of the amplitude of the analog signal. The A/D conversion process requires 0.6ms. 12 program words are required.

(2) Sequential Comparison Method
In this method the conversion speed varies in accordance with the rate of change of the analog quantity. When the rate of change is slow, the conversion rate increases. 30 program words are required.

Interrupt Functions

The M58845-XXXSP provides 3-factor, 1-level vector interrupt capability, enabling unique branching addresses for each interrupt factor.

The interrupt vector addresses are shown in Table 1.

Table 1 Vector Interrupt Addresses

Interrupt factor		Interrupt address
Interrupt type	Causal condition	
External interrupt	Rising edge at the INT input pin	Page 1, address 0
Timer 1 interrupt	Timer 1 overflow	Page 1, address 2
Timer 2 interrupt	Timer 2 overflow	Page 1, address 4

An interrupt is generated whenever any of the causal conditions listed in Table 1 are satisfied at a time when the INTE flag is set to 1 (when the EI instruction is executed the INTE flag is set to 1, enabling interrupt; the DI instruction clears this flag to 0, prohibiting interrupts). If any of the interrupt causing conditions continues when the INTE flag is 0, an interrupt is generated when the INTE flag is set to 1.

The interrupts generated as a result of timer 1 and timer 2 overflow conditions can be software controlled, allowing confirmation of the overflow condition using a skip instruction.

When an interrupt program is used, one level of the three-level stack register is required, the remaining two levels being used for subroutines. After the interrupt program is started, the data pointer DP, register A, carry flag CY, and registers used by the interrupt program are saved. The **RTI** instruction is required to restore these before returning to the main program.

When an interrupt occurs, the microcomputer internal states are as follows.

(1) Program counter
The current address in the main program is stored in a stack register and the vector interrupt address as shown in Table 4 is loaded into the program counter.

(2) Interrupt flag INTE
The flag INTE is reset to disable further interrupts. This disabled state will continue even after return to the main program by the **RTI** instruction until the execution of an **EI** instruction.

(3) Skip flags
Skip flags are provided to discriminate skip instructions and consecutively described skip instructions. Each flag has its own stack within which the skip state is saved. As a mask option, the interrupt pins may be provided with Schmitt input circuits.

Timer/Event Counter (2 Lines)

The timer/event counter section consists of two lines (timers). As shown in Fig. 8, this section includes timer 1 and its overflow flag (1F) and timer 2 and its overflow flag (2F), as well as the timer input/output port T and the timer control registers V and W.

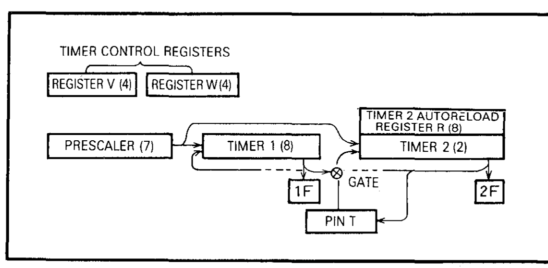


Fig. 4 Timer/event counter block diagram

The two timers (timer 1 and 2) are controlled by means of the timer control registers.

(1) Timer 1
Timer 1 is implemented using an 8-bit binary counter capable of being set and read by means of the **T1AB** and **TAB1** instructions respectively. Starting and stopping of the counter as well as the selection of the source (prescaler or timer 2) is accomplished by means of the timer control register. When an overflow condition occurs, setting the 1F to 1 stops the

counting operation.

(2) Timer 2

Time 2 is implemented using an 8-bit binary counter and is provided with an auto-reload register (register R). Timer 2 data can be read using the **TAB2** instruction and register R may be set as well as ready by means of the **TRAB** and **TABR** instructions respectively. Starting and stopping the counter as well as the selection of the source (prescaler or external input from port T) is controlled by the timer control registers. In addition, when port T has been chosen as the source, if only timer 1 is counting, gating is possible by means of using counter enabling controlled by the timer control registers. The overflow condition results in the setting of the flag 2F, after which timer 2 can be set with data once more by register R (auto-reload register) and continue counting.

(3) Prescaler

The overflow time can be selected as either 160μs or 1270μs (when using a 600kHz clock frequency) by means of the counter control registers.

(4) Timer I/O port T

This port can be selected by the counter control register as the source for timer 2. In addition, when another source has been selected, a pulse is available at this port every time timer 2 reaches the overflow condition.

(5) Timer 1 and 2 overflow flags 1F and 2F

These flags are set when the corresponding timer has reached the overflow condition. To test these flags, generation of an interrupt and skip instructions (**SNZ1, SNZ2**) can be used. The selection of which will be used is made by the timer control registers. By using either, these flags will be reset.

(6) Timer control registers V and W

The timer control registers are used to perform the above described control functions. Instructions **TVA** and **TWA** are used to transfer control data to these register.

Input/Output Ports

(1) Port K (K_7~K_0)

This analog/digital input port is capable of 8-bit input using the **SZJ** instruction and two groups of 4-bit inputs using the **IAS i** instruction. The analog signal may be A/D converted using either successive approximation or sequential comparison, as determined by the program. Also, an arbitrary threshold level in the range 0~–7V with respect to the digital signal may be input, enabling the use of the port as a high-noise immunity input.

Pull-down transistors and discharge transistors (for use with capacity touch-type keys) may be selected as mask options.

(2) Port D (D_{11}~D_0)

This port consists of 12 bits which can be used for both input and output functions by means of the **SZD, SD,** and **RD** instructions. The output section provides individual bit latching and the contents of register Y can be used to designate a single bit of port D for output or sensing. When using the port for input, the output must be cleared to 0 first. The instructions **CLD** and **CLDS** can be used to clear all bits of the port to 0. The outputs are open-drain circuits which can be provided with pull-down transistors as a mask option.

(3) Port F (F_3~F_0)

This 4-bit port is controlled for output and input by the **OFA** and **IAF** instructions respectively. When using a bit for input, that bit output must first be set to 0. The outputs are open drain circuits.

(4) Port S (S_7~S_0)

This port can perform 8-bit output using the **OSAB, OSPA,** and **OSE** instructions and 4-bit input using the **IAS** i instruction.

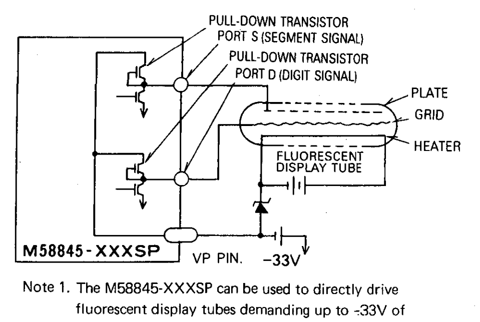
A built-in S output PLA has been provided which can code 4 bits of register A data arbitrarily and provide output using the **OSPA** instruction. The PLA output coding is a mask option.

When the port is used for input, the outputs must first be set to 0. All the port S bits may be set to 0 by means of the **CLS** or **CLDS** instructions.

The outputs are open-drain circuits which can be provided with pull-down transistors as a mask option.

V_P Pin

This pin is used to supply the required voltage for the port D and port S pull-down transistors. Built-in pull-down transistors can be provided as a mask option for driving



Note 1. The M58845-XXXSP can be used to directly drive fluorescent display tubes demanding up to -33V of output level.

Note 2. To provide complete blanking, the heater should be provided with a voltage of approximately 4 V from the VP pin.

Fig. 5 Fluorescent display tube drive circuit

fluorescent display tubes, as shown in Fig. 5, eliminating the need for the usual externally connected pull-down resistors and resulting in a reduction in the number of system components.

Reset

When the RESET pin is kept high for at least 3 machine cycles, the reset state is enabled. After reset has been performed, when the RESET input is driven low, program execution will begin at page 0, address 0.

When the reset state is enabled, the following operations are performed.

(1) The program counter is set to 0, address 0, (PC) ← 0
(2) The interrupt mode is in the disabled state. INTE ← 0 (the same as for the execution of the **DI** instruction)
(3) The carry and data pointer selector is set to 0, specifying DP and CY.
(4) Registers V and W are set to 0. V=W ← 0_{16}
(5) The 3 interrupt flags, external interrupt flag (EXF), timer 1 overflow flag (1F), and timer 2 overflow flag (2F) are reset. EXF=1F=2F ← 0
(6) All outputs of port D are cleared to low (D) ← 0
(7) All outputs of port F are cleared to low (F) ← 0
(8) All outputs of port S are cleared to low (S) ← 0
(9) All outputs of port T are cleared to low (T) ← 0

Clock Generator Circuits

A clock generator circuit has been built in, to allow control of the frequency by means of an externally connected RC circuit or ceramic resonator. In addition, an external clock signal may be applied at the X_{IN} pin, leaving the X_{OUT} pin open. Circuit examples are shown in Fig. 6~8.

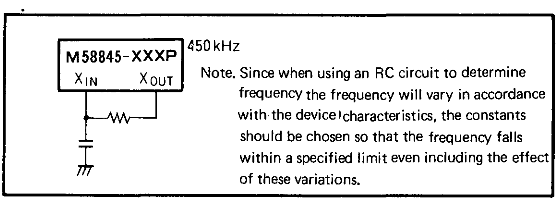


Fig. 6 External RC circuit

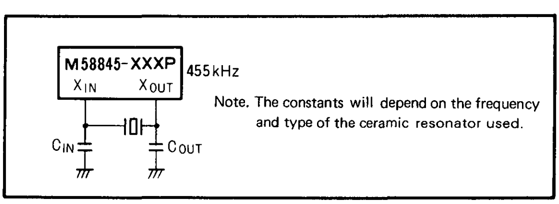


Fig. 7 Externally connected ceramic resonator

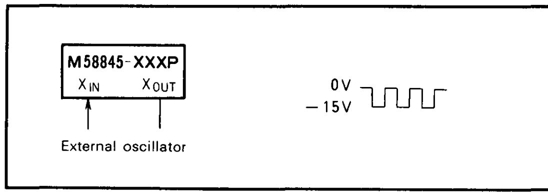


Fig. 8 External clock input circuit

Mask Options

The following mask options are available, specifiable at the time of initial ordering.

(1) S output PLA data
(2) Port K (K_7~K_0) discharge transistors
(3) Port K (K_7~K_0) pull-down transistors
(4) Port D (D_{11}~D_0) pull-down transistors
(5) Port S (S_7~S_0) pull-down transistors
(6) Selection of interrupt input TTL-compatible Schmitt circuits
(7) Selection of RESET input TTL-compatible Schmitt circuits
(8) Selection of port T TTL-compatible Schmitt circuits

Documentation Required upon Ordering

The following information should be provided when ordering a custom mask.

(1) M58845-XXXSP mask confirmation sheet
(2) ROM data — 3 EPROM sets
(3) S output PLA coding — On confirmation sheets
(4) Port K input discharge transistors — On confirmation sheets
(5) Port K pull-down transistors
(6) Port D pull-down transistors
(7) Port S pull-down transistors
(8) Selection of interrupt input TTL-compatible Schmitt circuits
(9) Selection RESET input TTL-compatible Schmitt circuits
(10) Selection of Port T input TTL-compatible Schmitt circuits

MACHINE INSTRUCTIONS

Type of instruction	Mnemonic	Instruction code D8 D7D6D5D4 D3D2D1D0	16mal notation	No. of words	No. of cycles	Functions	Skip conditions	Flag CY	Description of operation
Register-to-register transfers	TAB	0 0001 1110	0 1E	1	1	(A)←(B)	—	x	Transfers contents of register B to register A.
	TBA	0 0001 1100	0 1C	1	1	(B)←(A)	—	x	Transfers contents of register A to register B.
	TAY	0 0001 1101	0 1D	1	1	(A)←(Y)	—	x	Transfers contents of register Y to register A.
	TYA	0 0000 1100	0 0C	1	1	(Y)←(A)	—	x	Transfers contents of register A to register Y.
	TEAB	0 0001 1010	0 1A	1	1	$(E_7\sim E_4)$←(B) $(E_3\sim E_0)$←(A)	— —	x	Transfer contents of registers A and B to register E.
	TEPA	0 0001 0110	0 16	1	1	$(E_7\sim E_0)$← through PLA←(A)	—	x	Decodes contents of register A in the PLA and transfers result to register E.
RAM addresses	LXY x, y	0 11xx yyyy	0 CY + x	1	1	(X)←x where, x = 0 ~ 3 (Y)←y where, y = 0 ~ 15	Written successively	x	Loads value of "x" into register X, and of "y" into Y. When LXY is written successively the first is executed and successive ones are skipped.
	LZ z	0 0100 101z	0 4A + z	1	1	(Y)←z where, z = 0, 1	—	x	Loads value of "z" into register Z.
	INY	0 0000 0010	0 02	1	1	(Y)←(Y)+1	(Y)=0	x	Increments contents of register Y by 1. Skips next instruction when new contents of register Y are "0".
	DEY	0 0000 0011	0 03	1	1	(Y)←(Y)−1	(Y)=15	x	Decrements contents of register Y by 1. Skips next instruction when new contents of register Y are "15".
	LCPS i	0 0100 000i	0 4i	1	1	(CPS)←i where, i = 0, 1	—	x	Transfers designated contents of register J to register A.
RAM-accumulator transfers	TAM j	0 0110 01jj	0 64 + j	1	1	(A)←(M(DP)) (X)←(X)∀j where, j = 0 ~ 3	—	x	Transfers the RAM contents addressed by the active DP to register A. Register X is then "exclusive OR-ed" with the value j in the instruction, and the result stored in register X.
	XAM j	0 0110 00jj	0 6j	1	1	(A)←→(M(DP)) (X)←(X)∀j where, j = 0 ~ 3	—	x	Exchanges the contents of the RAM DP and register A. Contents of X are then "exclusive OR-ed" with the value j, and the result stored in register X.
	XAMD j	0 0110 10jj	0 68 + j	1	1	(A)←→(M(DP)) (Y)←(Y)−1 (X)←(X)∀j where, j = 0 ~ 3	(Y)=15	x	Exchanges the contents of the RAM and register A. Contents of X are then "exclusive OR-ed" with the value j in the instruction, and the result stored in register X. The contents of register Y are decremented by 1, and when the result is 15, the next instruction is skipped.
	XAMI j	0 0110 11jj	0 6C + j	1	1	(A)←→(M(DP)) (Y)←(Y)+1 (X)←(X)∀j where, j = 0 ~ 3	(Y)=0 (Y)=masked skip condition	x	Exchanges the contents of the RAM and register A. Contents of X are then "exclusive OR-ed" with the value j in the instruction and result stored in register X. The contents of register Y are incremented by 1, and when the result meets the next instruction is skipped with the marked skip condition.
Arithmetic operations	LA n	0 1011 nnnn	0B n	1	1	(A)←n where, n = 0 ~ 15	Written successively	x	Loads the value n into register A. When LA is written consecutively the first is executed, and successive ones are skipped.
	AM A	0 0000 1010	0 0A	1	1	(A)←(A)+(M(DP))	—	x	Adds the contents of the RAM to register A. The result is retained in register A, and the contents of flag CY are unaffected.
	AMC	0 0100 0011	0 43	1	1	(A)←(A)+(M(DP))+(CY) CY←Carry	—	0 1	Adds the RAM contents addressed by the active DP and contents of flag CY to register A. The result is stored in register A and the carry in the active flag CY.
	AMCS	0 0101 0011	0 53	1	1	(A)←(A)+(M(DP))+(CY) CY←Carry	— A carry is not produced and	0 1	Adds the contents of the RAM and flag CY to register A. The result is stored in register A and the carry in the CY, but the next instruction is skipped when a carry is produced.
	A n	0 1010 nnnn	0 An	1	1	(A)←(A)+n where, n = 0 ~ 15	= 0 n ≠ 6	x	Adds value n in the instruction to register A. The contents of flag CY are unaffected and their next instruction is skipped if a carry is not produced, except when n=6.
	SC	0 0100 1001	0 49	1	1	(CY)←1	—	1	Sets active flag CY.
	RC	0 0100 1000	0 48	1	1	(CY)←0	—	0	Resets active flag CY.
	SZC	0 0010 1111	0 2F	1	1		(CY=0)	x	Skips next instruction when contents of the active flag CY are 0.
	CMA	0 0000 1111	0 0F	1	1	$(A)\leftarrow(\overline{A})$	—	x	Stores complement of register A in register A.
Bit operations	SB j	0 0100 11jj	0 4C + j	1	1	$(M_j(DP))$←1 where, j = 0 ~ 3	—	x	Sets the jth bit of the RAM addressed by the active DP (the bit designated by the value j in the instruction).
	RB j	0 0101 11jj	0 5C + j	1	1	$(M_j(DP))$←0 where, j = 0 ~ 3	—	x	Resets the jth bit of the RAM addressed by the active DP (the bit designated by the value j in the instruction).
	SZB j	0 0010 00jj	0 2j	1	1		$(M_j(DP))$ = 0 where, j = 0 ~ 3	x	Skips next instruction when the contents of the jth bit of the RAM addressed by the active DP (the bit which is designated by the value j in the instruction) are 0.
Compares	SEAM	0 0010 0110	0 26	1	1		(M(DP)) = (A)	x	Skips next instruction when contents of register A are equal to the RAM contents addressed by the active DP.
	SEY y	0 0011 yyyy	0 3y	1	1		(Y) = y where, y = 0 ~ 15	x	Skips next instruction when the contents of register Y are equal to the value y in the instruction.
A/D converter operations	TLA	0 0001 1001	0 19	1	1	(L)←(A)	—	x	Transfers contents of register A to register L.
	THA	0 0101 1001	0 59	1	1	(H)←(A)	—	x	Transfers contents of register A to register H.
	XAL	0 0001 1000	0 18	1	1	(A)←→(L)	—	x	Exchanges contents of register A with contents of register L.
	XAH	0 0101 1000	0 58	1	1	(A)←→(H)	—	x	Exchanges contents of register A with contents of register H.
	LC7	0 0101 0111	0 57	1	1	(C)←7	—	x	Loads 7 to register C.
	DEC	0 0000 1001	0 09	1	1	(C)←(C)−1	(C)=7	x	Decrements contents of register C by 1, when result is 7, skips
	SHL	0 0100 0010	0 42	1	1	(C_2) = 1 when : $(H(C_1\sim C_0))$←1 (C_2) = 0 when : $(L(C_1\sim C_0))$←1	—	x	Sets the bit in register L or H designated by register C. The box instruction shows the relationship between register C and bit position. (C): 7, 6, 5, 4, 3, 2, 1, 0 Bit: H_3, H_2, H_1, H_0, L_3, L_2, L_1, L_0
	RHL	0 0101 0010	0 52	1	1	(C_2) = 1 when : $(H(C_1\sim C_0))$←0 (C_2) = 0 when : $(L(C_1\sim C_0))$←0	—	x	Resets the bit in register L or H that is designated by register C.
	CPA	0 0000 1000	0 08	1	2	$\lvert Vref\rvert > \lvert V_{K(i)}\rvert$ when : (J(i))←1 $\lvert Vref\rvert < \lvert V_{K(i)}\rvert$ when : (J(i))←0 i = 0 ~ 7	—	x	Reads all analog values from input port K for comparison with D-A converter output V_{ref}, and either sets the respective bit of register J to the next instruction cycle, wherever $V_{ref} > V_{K(1)}$ is true, or resets it, wherever $V_{ref} < V_{K(1)}$ is true.
	CPAS	0 0101 0001	0 51	1	1	$\lvert Vref\rvert > \lvert V_{K(i)}\rvert$ when : (J(i))←1 $\lvert Vref\rvert < \lvert V_{K(i)}\rvert$ when : (J(i))←0	—	x	Reads and stores temporarily all analog values from input port K, which are then unaffected by changes in port K inputs. These values are compared with the D-A converter output V_{ref}, calculated from contents of registers H and L, and respective bits of register J are set/reset. Repeated when contents of registers H-L are changed.

Type of instruction	Mnemonic	Instruction code D_8 $D_7D_6D_5D_4$ $D_3D_2D_1D_0$	16mal notation	No. of words	No. of cycles	Functions	Skip conditions	Flag CY	Description of operation
A/D converter operations	CPAE	0 0101 0000	0 5 0	1	1	Execution of the instruction CPAS is over, and no more changes will made in $(J_{(Y)})$	—	X	Terminates execution of instruction CPAS Contents of register J remain unaffected, maintaining the value immediately before termination, and input port K is again ready to receive input.
	TAJ	0 0000 1101	0 0 D	1	1	$(Y_0)=0$ when : $(A) \leftarrow (J_3\ J_2\ J_1\ J_0)$ $(Y_0)=1$ when : $(A) \leftarrow (J_7\ J_6\ J_5\ J_4)$	—	X	
	SZJ	0 0010 1001	0 2 9	1	1		$(J(y))=0$	X	Skips next instruction when the bit in register J, designated by register Y, is 0.
Timer operation	T1AB	0 1000 0100	0 8 4	1	1	$(1_7 \sim 1_4) \leftarrow (B)$ $(1_3 \sim 1_0) \leftarrow (A)$	—	X	Transfers contents of register A and register B to timer 1.
	TRAB	0 1000 0101	0 8 5	1	1	$(R_7 \sim R_4) \leftarrow (B)$ $(R_3 \sim R_0) \leftarrow (A)$	—	X	Transfers contents of register A and register B to timer 2 auto reload register R.
	TAB1	0 1000 1000	0 8 8	1	1	$(B) \leftarrow (1_7 \sim 1_4)$ $(A) \leftarrow (1_3 \sim 1_0)$	—	X	Transfers contents of timer 1 to register A and register B.
	TABR	0 1000 1001	0 8 9	1	1	$(B) \leftarrow (R_7 \sim R_4)$ $(A) \leftarrow (R_3 \sim R_0)$	—	X	Transfers contents of timer 1 auto reload register R to register A and register B.
	TAB2	0 1000 1010	0 8 A	1	1	$(B) \leftarrow (2_7 \sim 2_4)$ $(A) \leftarrow (2_3 \sim 2_0)$	—	X	Transfers contents of timer 2 to register A and register B.
	TVA	0 1000 0110	0 8 6	1	1	$(V) \leftarrow (A)$	—	X	Transfers contents of register A to timer control register V.
	TWA	0 1000 0111	0 8 7	1	1	$(W) \leftarrow (A)$	—	X	Transfers contents of register A to timer control register W.
	SNZ1	0 1000 0010	0 8 2	1	1		$(1F)=1$	X	Skips the next instruction if flag 1F is 1.
	SNZ2	0 1000 0011	0 8 3	1	1		$(2F)=1$	X	Skips the next instruction if flag 2F is 1.
Branch	B xy	1 1xxx yyyy	1 8y + x	1	1	$(PC_L) \leftarrow 16x+y$ $(PC_H) \leftarrow 3,\ (PC_L) \leftarrow 16x+y$	—	X	Jumps to address xy of the current page. Jumps to address xy on page 3 when executed, provided that none of instruction RT, RTS, BL, BML, BLA or BMLA was executed after execution of instruction BM or BMA.
	BL pxy	0 0111 PPPP 1 1xxx yyyy	0 7P 1 8y + x	2	2	$(PC_H) \leftarrow p$ $(PC_L) \leftarrow 16x+y$	—	X	Jumps to address xy of page p.
	BA xy	0 0000 0001 1 1xxx XXXX	0 01 1 8X × x	2	2	$(PC_L) \leftarrow 16x+(A)$ $(PC_H) \leftarrow 3,\ (PC_L) \leftarrow 16x+(A)$	—	X	Subroutine on the current page. Exchange the lower 4 bits of the contents of address xX with the contents of register A and branch to address 16x+A Page 3 subroutine: After execution of a BM or BMA instruction without execution of a RT, RTS, BL, BML, BLA, or BMLA instruction, when a BA instruction is executed branching is done to address 16x+(A) on page 3
	BLA pxy	0 0000 0001 0 0111 PPPP 1 1xxx XXXX	0 01 0 7P 1 8X + x	3	3	$(PC_H) \leftarrow p$ $(PC_L) \leftarrow 16x+(A)$	—	X	Subroutine on a different page: Exchange the lower 4 bits of the contents of address xX with the contents of register A and branch to the address 16x+(A)
Subroutine calls	BM xy	1 0xxx yyyy	1 xy	1	1	$(SK_2) \leftarrow (SK_1) \leftarrow (SK_0) \leftarrow (PC)$ $(PC_H) \leftarrow 2,\ (PC_L) \leftarrow 16x+y$ $(PC_H) \leftarrow 2,\ (PC_L) \leftarrow 16x+y$	—	X	Calls for the subroutine starting at address x(A) of page 2. Jumps to address xy of page 2 provided that none of instructions. RT, RTS, BL, BML, BLA or BMLA was executed after the execution of instructions BM or BMA.
	BML pxy	0 0111 PPPP 1 0xxx yyyy	0 7P 1 xy	2	2	$(SK_2) \leftarrow (SK_1) \leftarrow (SK_0) \leftarrow (P_C)$ $(PC_H) \leftarrow p,\ (PC_L) \leftarrow 16x+y$	—	X	Calls for the subroutine starting at address xy of page p.
	BMA xX	0 0000 0001 1 0xxx XXXX	0 01 1 xX	2	2	$(SK_2) \leftarrow (SK_1) \leftarrow (SK_0) \leftarrow (PC)$ $(PC_H) \leftarrow 2,\ (PC_L) \leftarrow 16x+(A)$ $(PC_H) \leftarrow 2,\ (PC_L) \leftarrow 16x+(A)$	—	X	Calls for the subroutine starting at address x(A) of page 2. Jumps to address x(A) of page 2 provided that none of instructions. RT, RTS, BL, BML, BLA or EMLA was executed after the execution of instructions BM or BMA.
	BMLA pxX	0 0000 0001 0 0111 PPPP 1 0xxx XXXX	0 01 0 7P 1 xX	3	3	$(SK_2) \leftarrow (SK_1) \leftarrow (SK_0) \leftarrow (PC)$ $(PC_H) \leftarrow p,\ (PC_L) \leftarrow 16x+(A)$	—	X	Calls for the subroutine starting at address x(A) of page p.
Program returns	RTI	0 0100 0110	0 4 6	1	1	$(PC) \leftarrow (SK_0) \leftarrow (SK_1) \leftarrow (SK_2)$	—	X	Returns from interrupt routine to main routine. The internal flip-flop is restored to the value held immediately before the interrupt.
	RT	0 0100 0100	0 4 4	1	1	$(PC) \leftarrow (SK_0) \leftarrow (SK_1) \leftarrow (SK_2)$	—	X	Returns to the main routine from the subroutine.
	RTS	0 0100 0101	0 4 5	1	1	$(PC) \leftarrow (SK_0) \leftarrow (SK_1) \leftarrow (SK_2)$		X	Returns to the main routine from the subroutine, and unconditionally skips the next instruction.
Input/output	CLD	0 0001 0011	0 1 3	1	1	$(D) \leftarrow 0$	—	X	Clears port D. (low level output)
	CLS	0 0001 0000	0 1 0	1	1	$(S) \leftarrow 0$	—	X	Clears port S.
	CLDS	0 0001 0001	0 1 1	1	1	$(D) \leftarrow 0$ $(S) \leftarrow 0$	—	X	Clears ports S and D.
	SD	0 0001 0101	0 1 5	1	1	$(D(Y)) \leftarrow 1$ where, $Y=0 \sim 11$	—	X	Sets the bit of port D that is designated by register Y.
	RD	0 0001 0100	0 1 4	1	1	$(D(Y)) \leftarrow 0$ where, $Y=0 \sim 11$	—	X	Resets the bit of port D that is designated by register Y.
	SZD	0 0010 1011	0 2 B	1	1		$(D(Y))=0$ where, $Y=0 \sim 11$	X	Skips the next instruction if the contents of the bit of port D that is designated by register Y are 0.
	OSAB	0 0001 1011	0 1 B	1	1	$(S_7 \sim S_4) \leftarrow (B)$ $(S_3 \sim S_0) \leftarrow (A)$	—	X	Output contents of registers A and B to port S.
	OSPA	0 0001 0111	0 1 7	1	1	$(S_7 \sim S_4) \leftarrow$ through PLA $\leftarrow (A)$	—	X	Decodes contents of register A by PLA and the result is output to ports.
	OSE	0 0000 1011	0 0 B	1	1	$(S) \leftarrow (E)$	—	X	Outputs contents of register E to port S.
	IAS i	0 0101 010i	0 54 + j	1	1	$i=0\ (A) \leftarrow (S_7 \sim S_4)$ $i=1\ (A) \leftarrow (S_3 \sim S_0)$	—	X	Transfers from port S to register A. The high-order four bits of port S are transferred when the value of i in the instruction is 0 or the low-order four bits are transferred when the value of i is 1.
	OFA	0 1000 0001	0 8 1	1	1	$(F) \leftarrow (A)$	—	X	Sets interrupt flag INTE to enable interrupts.
	IAF	0 1000 1100	0 8 C	1	1	$(A) \leftarrow (F)$	—	X	Resets interrupt flag INTE to disable interrupts.
Interrupts	EI	0 0000 0101	0 0 5	1	1	$(INTE) \leftarrow 1$		X	Outputs contents of register A to port F.
	DI	0 0000 0100	0 0 4	1	1	$(INTE) \leftarrow 0$		X	Transfers input from port F to register A.
Misc	NOP	0 0000 0000	0 0 0	1	1	$(PC_L) \leftarrow (PC_L)+1$		X	No operation.

Symbol	Contents	Symbol	Contents	Symbol	Contents
A	4-bit register (accumulator)	SK_0	11-bit stack register	←	Shows direction of data flow
B	4-bit register	SK_1	11-bit stack register	( )	Indicates contents of register, memory, etc.
C	3-bit register	SK_2	11-bit stack register	xx	2-bit binary variable
E	8-bit register	1	Timer 1	yyyy	4-bit binary variable
H	4-bit register	2	Timer 2	z	1-bit binary variable
J	8-bit register	CY	1-bit carry flag	nnnn	4-bit binary variable
L	4-bit register	INTE	Interrupt enable flag	i	1-bit binary constant
R	8-bit timer 2 auto reload register	CPS	Indicates which data pointer and carry are active	ij	2-bit binary constant
V	4-bit register	1F	1-bit timer 1 overflow flag	XXXX	4-bit unknown bibary number
W	4-bit register	2F	1-bit timer 2 overflow flag	⊻	Exclusive-OR
X	2-bit register	EXF	1-bit external interrupt flag	–	Negation
Y	4-bit register	D	12-bit port	X	Indicates flag is uneffected by instruction execution
Z	1-bit register	F	4-bit port	xy	Label used to indicate the address XXXYYYY
DP	7-bit data pointer, combination of registers X, Y, and Z	K	8-bit port	pxy	Label used to indicate the address XXXYYYY of page PPPP
PC_H	The high-roder 4-bits of the program counter	S	8-bit port	C + x	Hexadecimal number C + binary number x
PC_L	The low-order 7-bits of the program counter	T	1-bit port		
PC	11-bit program counter, combination of PC_H and PC_L	INT	i Interrupt request signal		

Note 1. When a skip has occurred, the next instruction only is ignored and the program counter is not incremented by 2, therefore, the number of cycles does not change in accordance with the existence or non-existence of skip

INSTRUCTION CODE TABLE

D_3~D_0 \ D_8~D_4	Hexadecimal number C	0 0000 00	0 0001 01	0 0010 02	0 0011 03	0 0100 04	0 0101 05	0 0110 06	0 0111 07	0 1000 08	0 1001 09	0 1010 0A	0 1011 0B	0 1100 0C	0 1101 0D	0 1110 0E	0 1111 0F	1 0000 ~ 1 0111 10~17	1 1000 ~ 1 1111 18~1F
0000	0	NOP	CLS	SZB 0	SEY 0	LCPS 0	CPAE	XAM 0	BL BML	–	–	A 0	LA 0	LXY 0,0	LXY 1,0	LXY 2,0	LXY 3,0	BM	B
0001	1	BA BMA BLA BMLA	CLDS	SZB 1	SEY 1	LCPS 1	CPAS	XAM 1	BL BML	OFA	–	A 1	LA 1	LXY 0,1	LXY 1,1	LXY 2,1	LXY 3,1	BM	B
0010	2	INY	–	SZB 2	SEY 2	SHL	RHL	XAM 2	BL BML	SNZ1	–	A 2	LA 2	LXY 0,2	LXY 1,2	LXY 2,2	LXY 3,2	BM	B
0011	3	DEY	CLD	SZB 3	SEY 3	AMC	AMCS	XAM 3	BL BML	SNZ2	–	A 3	LA 3	LXY 0,3	LXY 1,3	LXY 2,3	LXY 3,3	BM	B
0100	4	DI	RD	–	SEY 4	RT	IAS 0	TAM 0	BL BML	T1AB	–	A 4	XA 4	LXY 0,4	LXY 1,4	LXY 2,4	LXY 3,4	BM	B
0101	5	EI	SD	–	SEY 5	RTS	IAS 1	TAM 1	BL BML	TRAB	–	A 5	LA 5	LXY 0,5	LXY 1,5	LXY 2,5	LXY 3,5	BM	B
0110	6	–	TEPA	SEAM	SEY 6	RTI	–	TAM 2	BL BML	TVA	–	A 6	LA 6	LXY 0,6	LXY 1,6	LXY 2,6	LXY 3,6	BM	B
0111	7	–	OSPA	–	SEY 7	–	LC7	TAM 3	BL BML	TWA	–	A 7	LA 7	LXY 0,7	LXY 1,7	LXY 2,7	LXY 3,7	BM	B
1000	8	CPA	XAL	–	SEY 8	RC	XAH	XAMD 0	BL BML	TAB1	–	A 8	LA 8	LXY 0,8	LXY 1,8	LXY 2,8	LXY 3,8	BM	B
1001	9	DEC	TLA	SZJ	SEY 9	SC	THA	XAMD 1	BL BML	TABR	–	A 9	LA 9	LXY 0,9	LXY 1,9	LXY 2,9	LXY 3,9	BM	B
1010	A	AM	TEAB	–	SEY 10	LZ 0	–	XAMD 2	BL BML	TAB2	–	A 10	LA 10	LXY 0,10	LXY 1,10	LXY 2,10	LXY 3,10	BM	B
1011	B	OSE	OSAB	SZD	SEY 11	LZ 1	–	XAMD 3	BL BML	–	–	A 11	LA 11	LXY 0,11	LXY 1,11	LXY 2,11	LXY 3,11	BM	B
1100	C	TYA	TBA	–	SEY 12	SB 0	RB 0	XAMI 0	BL BML	IAF	–	A 12	LA 2	LXY 0,12	LXY 1,12	LXY 2,12	LXY 3,12	BM	B
1101	D	TAJ	TAY	–	SEY 13	SB 1	RB 1	XAMI 1	BL BML	–	–	A 13	LA 13	LXY 0,13	LXY 1,13	LXY 2,13	LXY 3,13	BM	B
1110	E	–	TAB	–	SEY 14	SB 2	RB 2	XAMI 2	BL BML	–	–	A 14	LA 14	LXY 0,14	LXY 1,14	LXY 2,14	LXY 3,14	BM	B
1111	F	CMA	–	SZC	SEY 15	SB 3	RB 3	XAMI 3	BL BML	–	–	A 15	LA 15	LXY 0,15	LXY 1,15	LXY 2,15	LXY 3,15	BM	B

Note 1: This list shows the machine codes and corresponding machine instructions. $D_3 \sim D_0$ indicate the low-order 4 bits of the machine code and $D_8 \sim D_4$ indicate the high-order 5 bits. Hexadecimal numbers are also shown that represent the codes An instruction may consist of one, two, or three words, but only the first word is listed. Code combination indicated with a bar (-) must not be used.

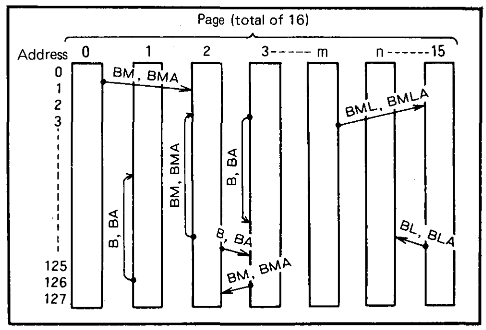
Note 2: Two-word instruction

	Second word
BL	1 1xxx yyy
BML	1 0xxx yyyy
BA	1 1xxx XXXX
BMA	1 0xxx XXXX

Three-word instruction

	Second word	Third word
BLA	0 0111 pppp	1 1xxx XXXX
BM	0 0111 pppp	1 0xxx XXXX

Note 3. Relationships between branching and page by means of branching instructions and subroutine calling instructions.



4

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	0.3~−20	V
V_I	Input votage (ports D and S, and input VP)		0.3~−35	V
V_I	Input voltage, inputs other than ports D and S, and input VP		0.3~−20	V
V_O	Output voltage, ports D and S		0.3~−35	V
V_O	Output voltage, other outputs than ports D and S		0.3~−20	V
Pd	Power dissipation	Ta=25°C	1100	mW
Topr	Operating temperature		−10~70	°C
Tstg	Storage temperature		−40~125	°C

RECOMMENDED OPERATING CONDITIONS (Ta = -10 ~ 70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	−13.5	−15	−16.5	V
V_{SS}	Supply voltage		0		V
V_P	Pull-down transistor supply voltage	0		−33	V
V_{IH}	High-level input voltage	−1.5		0	V
$V_{IH(\phi)}$	High-level clock input voltage	−1.5		0	V
V_{IL}	Low-level input voltage, inputs other than ports D and S	V_{DD}		−4.2	V
V_{IL}	Low-level input voltage ports D and S	−33		−4.2	V
$V_{IL(\phi)}$	Low-level clock input votlage	V_{DD}		$V_{DD}+2$	V
$V_{I(K)}$	Analog input voltage, port K	V_{REF}		0	V
$V_{I(K)}$	Digital input voltage, port K	V_{DD}		0	V
V_{REF}	Reference voltage	−5		−7	V
V_{OL}	Low-level output voltage, ports D and S	−33		0	V
$f(\phi)$	Internal clock oscillation frequency	300		600	kHz

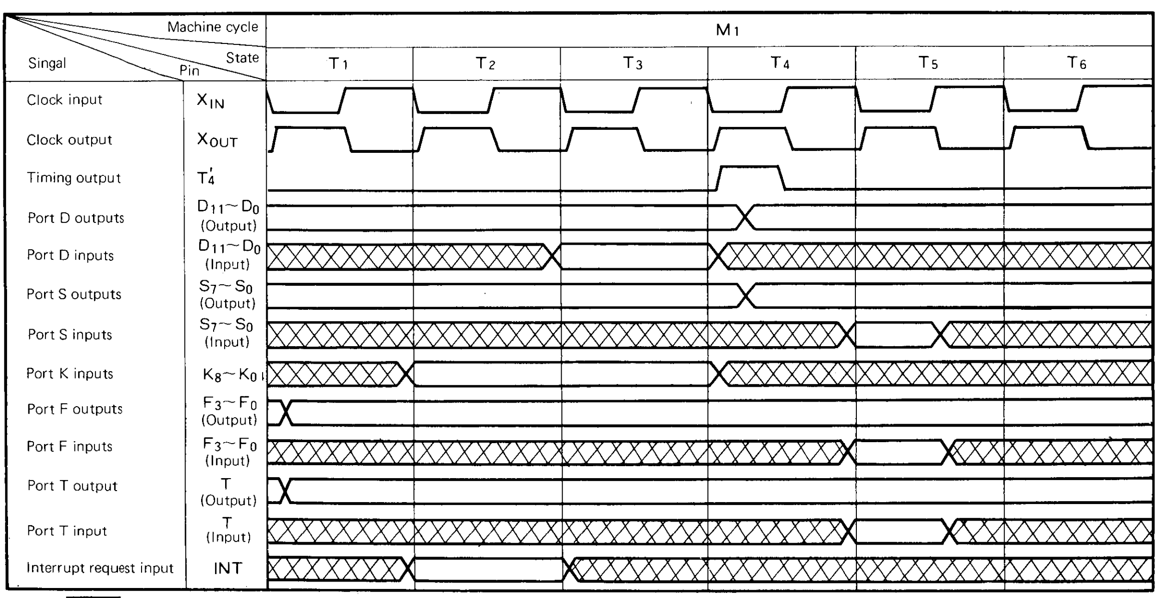
Note 4. $V_{IL}(\phi)$ is specified for the maximum V_{DD} value.

ELECTRICAL CHARACTERISTICS (Ta = -10 ~ 70°C, V_{DD} = -15±10%, V_{SS} = 0V, f(φ) = 300 ~ 600 kHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min	Typ	Max	Unit
V_{OH}	High-level output voltage, port D	$V_{DD}=-15V$, $I_{OH}=-15mA$, $Ta=25°C$	−2.5			V
V_{OH}	High-level output voltage, ports S and F	$V_{DD}=-15V$, $I_{OH}=-8mA$ (port S) $I_{OH}=-6mA$ (Port F), $Ta=25°C$	−2.5			V
V_{T-}	Negative threshold voltage (Schmitt input mask option)	$V_{DD}=-15V$, $Ta=25°C$	−7		−4	V
$V_{T+}-V_{T-}$	Hysteresis (Schmitt input mask option)	$V_{DD}=-15V$, $Ta=25°C$	1.5		3.5	V
I_I	Input current, port K	Measured when not executing CPA or CPAS $V_I=-7V$		−1	−7	μA
I_{IH}	High-level input current, port K (with pull-down resistors)	$V_{DD}=-15V$, $V_{IH}=0V$, $Ta=25°C$	50		250	μA
I_{IH}	High-level input current, ports D and S (with pull-down resistors)	$V_P=-33V$, $V_{IH}=0V$, $Ta=25°C$	80		280	μA
$I_{I(\phi)}$	Clock input current	$V_{I(\phi)}=-15V$, $Ta=25°C$		−20	−40	μA
I_{OH}	High-level output current, port D (Note 2)	$V_{DD}=-15V$, $V_{OH}=-25V$, $Ta=25°C$			−15	mA
I_{OH}	High-level output current, ports S and F	$V_{DD}=-15V$, $V_{OH}=-25V$, $Ta=25°C$		−6 (port F) −8 (port S)		mA
I_{OL}	Low-level output current, ports D and S	$V_{OL}=-33V$, $Ta=25°C$			−33	μA
I_{OL}	Low-level output current, port F	$V_{DD}=-15V$, $Ta=25°C$			−33	μA
I_{DD}	Supply current	$V_{DD}=-15V$, $Ta=25°C$		21		mA
I_{REF}	Reference supply current	$V_{REF}=-7V$, $Ta=25°C$			−1	mA
C_i	Input capacitance, port K	$V_{DD}=V_I=V_O=V_{SS}$, f=1MHz, 25mVrms		7	10	pF
$C_{i(\phi)}$	Clock input capacitance	$V_{DD}=X_{OUT}=V_{SS}$, f=1MHz, 25mVrms		7	10	pF
	A-D conversion linearity error	$V_{REF}=-7V$	Overall	±2	±3	LSB
	A-D conversion zero error					
	A-D conversion fullscale error					

Note 1. Currents are taken as positive when flowing into the IC (zero signal condition), with the minimum and maximum values as absolute values.
2. It is possible to connect up to 5 lines of the port D at maximum ratings (-15mA) or all lines of port S and F at maximum ratings of -8mA and -6mA respectively.

BASIC TIMING DIAGRAM



Note 3. The crosshatch area indicates invalid input.

SINGLE-CHIP 4-BIT MICROCOMPUTER WITH TWO TIMER/EVENT COUNTER

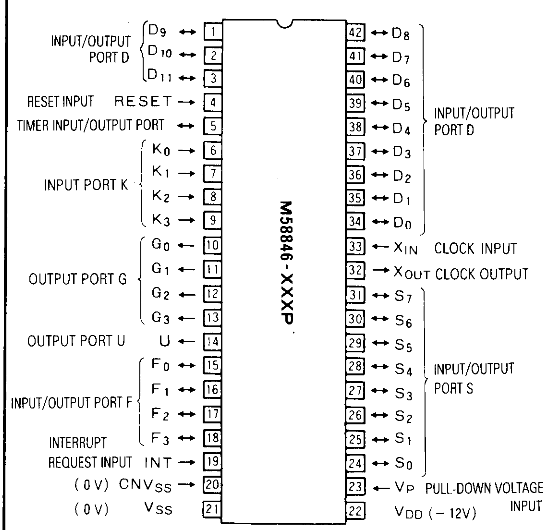
DESCRIPTION

The M58846-XXXSP is a single-chip 4-bit microcomputer developed using p-channel aluminum gate ED-MOS technology. The device includes two timers (one 7-bit timer and one 8-bit timer/event counter). It is housed in a 42-pin shrink plastic molded DIL package.

FEATURES

- Basic machine instructions 65
- Basic instruction execution time (1-word instruction at a clock frequency of 600kHz) 10μs
- Memory capacity ROM: .... 2048 words x 9 bits
  RAM: .... 128 words x 4 bits
- Single −12V power supply
- Two built-in timers (timer 1: 7-bit timer/counter, timer 2: 8-bit timer/event counter) .... 2 lines
- Interrupt function .... 3 factors (external, timer 1, timer 2), 1 level
- Two built-in data pointers
- Subroutine nesting .... 3 levels
- Input (port K) .... 4 lines
- Input/output (ports D, F, and S) .... 24 lines
- Output (ports G and U) .... 5 lines
- Timer input/output (port T) .... 1 line
- Direct drive for large fluorescent display tubes is possible
- Built-in decoder PLA for port S outputs (mask option)
- Built-in pull-down transistors (ports D, K, and S mask option)
- Built-in clock generator circuit

PIN CONFIGURATION (TOP VIEW)

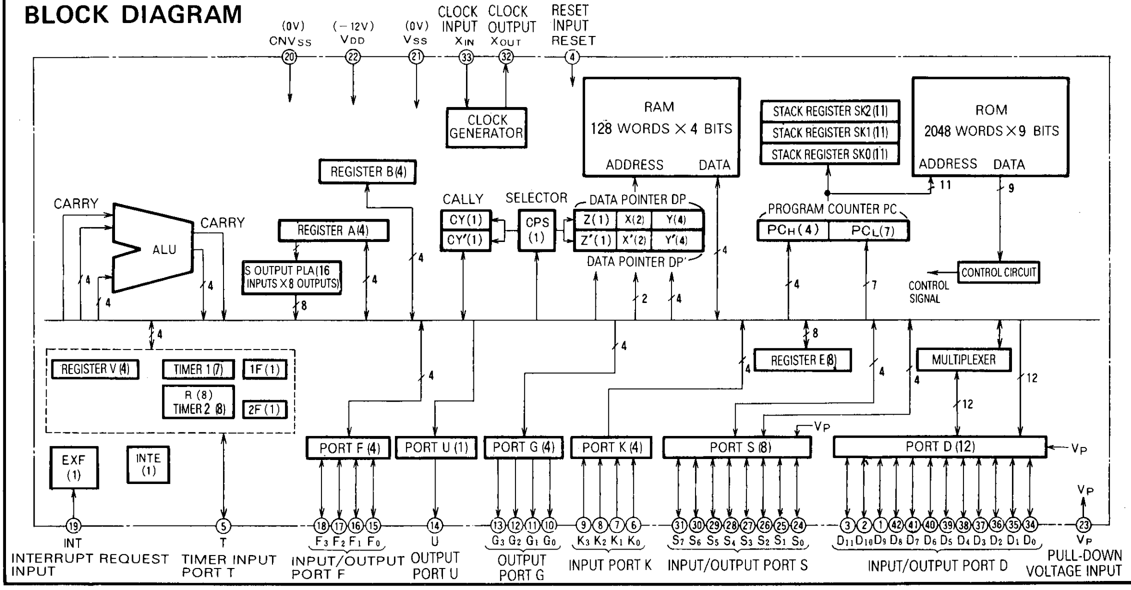


Outline 42P4B

APPLICATIONS

- VTRs, TVs, cassette decks
- Office equipment, copying machines, medical equipment
- Educational equipment, games

BLOCK DIAGRAM



PERFORMANCE SPECIFICATIONS

Parameter			Performance
Basic machine instructions			65
Instruction execution time (1-word instructions)			10μs (with a clock frequency of 600kHz)
Clock frequency			300～600kHz
Memory capacity	ROM		2048 words x 9 bits
	RAM		128 words x 4 bits
Input/output ports	K (Note 1)	Input	4 bits x 1
	D (Note 2)	Input	1 bit x 12
		Output	1 bit x 12
	F	Input	4 bits x 1
		Output	4 bits x 1
	S (Note 2)	Input	4 bits x 2
		Output	8 bits x 1
	G	Output	4 bits x 1
	U	Output	1 bit x 1
	T (Note 1)	Input	1 bit x 1
		Output	1 bit x 1
	INT (external interrupt request) (Note 1)		1 bit x 1
Timers			Timer 1: 7-bit timer Timer 2: 8-bit timer/event counter, timer input output port T
Pull-down voltage input pin			Used for driving devices such as large fluorescent display tubes (ports D and S)
Subroutine nesting			3 levels
Interrupts			3 factors (external, timer 1, timer 2), 1 level
Clock generator			Built-in
I/O characteristics of ports	Port D		−33V input/output withstanding voltage, output current −15mA
	Port S		−33V input/output withstanding voltage, output current −8mA
	Ports other than D and S		−20V input/output withstanding voltage, output current −6mA
Supply voltage			−12V (Typ)
Device structure			p-channel aluminum gate ED-MOS
Package			42-pin silicone plastic molded DIL package
Power dissipation (excluding ports)			280mW (typ)

Note 1. Input characteristics mask option (TTL-compatible Schmitt circuit)
2. Built-in pull-down transistors (mask options)

PIN DESCRIPTION

Pin	Name	Input or output	Function
V_{SS}	Ground		Connected to 0V potential
V_{DD}	Supply voltage		Connected to a −12V supply
V_P	Pull-down supply	In	Input for the supply voltage connected to the load resistors (mask option) for ports D and S
$K_3 \sim K_0$	Input port K	In	This port can be used to perform 4-bit TTL-compatible or Schmitt input. Pull-down transistors and input discharge transistors are available as mask options.
$D_{11} \sim D_0$	I/O port D	In/out	Port D consists of a 12-bit input/output port, all bits operating individually. When a port D output is programmed low, the output floats and the input signal can be sensed. The outputs are open drain circuits which can be provided with pull-down transistors as a mask option.
$F_3 \sim F_0$	I/O port F	In/out	Port F is a 4-bit input/output port. When the output is programmed to low, the output floats and the input signal can be sensed. The output circuits are open drain circuits.
$S_7 \sim S_0$	I/O port S	In/out	The I/O port S can be used as either an 8-bit output port or a pair of 4-bit input ports. When the output port S is programmed to the low level, it remains in the floating state so that it can be used as an input port.
$G_3 \sim G_0$	Output port G	Out	This is a 4-bit output port.
U	Output port U	Out	This is a 1-bit output port.
T	Timer I/O port T	In/out	This port is used as the timer 2 event counter input and the timer 2 overflow output, the functions being software selectable.
INT	Interrupt request input	In	This is the input for interrupt requests.
RESET	Reset	In	When this input is kept high for at least 3 machine cycles, the reset state is enabled.
X_{IN}	Clock input	In	These are the input and output pins for the built-in clock generator. A ceramic filter element (300kHz ~ 600kHz) or a resistor/capacitor combination are connected to these pins to provide the required oscillation stability.
X_{OUT}	Clock output	Out	
CNV_{SS}	CNV_{SS}	In	This input is connected to V_{SS} and must have a high-level input applied to it (0V).

BASIC FUNCTION BLOCKS

Program Memory (ROM)

This 2048-word x 9-bit Mask ROM can be programmed with machine instruction codes in accordance with the customer's specifications. It consists of 16 pages, each containing an address range of 0~127. Fig. 1 shows the address map for this ROM.

Program Counter (PC)

This counter is used to specify ROM addresses and the sequence of read-out of instructions stored in ROM. The program counter is an 11-bit counter, the upper-order 4 bits of which (PC_H) indicate the ROM page, and the lower 7 bits of which (PC_L) are a pure binary address designation. Each time an instruction is executed, PC_L is incremented by 1 step. For branching and subroutine call instructions, its value is set to the designated address.

When the 127 address is reached for every page, the address value returns to the first address of that page. Therefore, for moving from one page to another page, the page byte itself must be modified. This is done using the **BL** and **BLA** instructions.

Page 2 and page 3 are special pages used for subroutine calls. Page 2 can be called with a 1-word instruction from any arbitrary page. This instruction is either **BM** or **BMA**. When either **BM** or **BMA** is executed, subsequent **BM** or **BMA** instructions are equivalent to **B** and **BA** on page 2. Also, **B** or **BA** is equivalent to **B** or **BA** on page 3. This condition is cancelled when the **RT**, **RTS**, **BL**, **BML**, **BLA**, or **BMLA** instruction is executed. Note 3 shows the instruction codes and corresponding states.

Stack Registers (SK_0, SK_1, SK_2)

These registers are used to temporarily store the contents of the PC while executing subroutines or interrupt programs until the program returns to its original routine. The SK registers are organized in 3 words of 11 bits each, enabling up to 3 levels of subroutine nesting. If 1 level is used for an interrupt routine, the remaining 2 levels can be used for subroutine calls.

Data Memory (RAM)

This 512-bit (128 words x 4 bits) RAM is used to store both processing and control data. One RAM word consists of 4 bits with bit manipulation possible over the entire storage area. The 128 words are arranged as 2 file groups x 4 files x 16 digits x 4 bits. Fig. 2 shows the RAM address map. The RAM address specification is made by the combination of data pointer DP register Z, register X, and register Y. Thus, the selector CPS and data pointer DP must be set. However, as long as the address is not changed this is not necessary.

Data Pointers (DP, DP')

These registers are used to designate the RAM address, and bit position for the I/O port D and register J. Each data pointer is composed of a 7-bit register. Register Z (the most significant bit of DP) designates the RAM file group; register X (the central 2 bits) designates the RAM file; and register Y (the least significant 4 bits) designates the digit position of the RAM file. At the same time, register Y designates the bit positions of the I/O port D J.

4-Bit Arithmetic Logic Unit (ALU)

This unit executes 4-bit arithmetic and logical operations by means of a 4-bit adder and related logic circuitry.

Register A and Carry Flag (CY)

Register A is a 4-bit accumulator that constitutes the basis for arithmetic operations. Data processing operations such as arithmetic and logical operations, data transfer, exchange, conversion, and data input/output are executed by means of this register. The carry flag CY is used to store carry or overflow after execution of arithmetic and logical operations by the arithmetic logic unit. The carry flag may also be used as a 1-bit flag. Two carry flags, CY and CY', are available and selected by selector CPS, as is the data pointer DP.

Registers B and E

Register B is composed of 4 bits and can be used as a 4-bit temporary storage register or for 8-bit data transfer in conjunction with register A. Register E is composed of 8 bits and is used not only as an 8-bit temporary storage register, but also as a tempoı ary for the I/O port S.

PC_H / PC_L	Page designation 0	1	⋯	15
Bit designation	8 7 6 5 4 3 2 1 0	8 7 6 5 4 3 2 1 0	8 7 6 2 1 0	8 7 6 5 4 3 2 1 0
Address designation 0				
1				
2				
⋮	⋮	⋮	⋮	⋮
126				
127				

Fig. 1 ROM Address map

File designation: Register Z	0				1		
File designation: Register X	0	1	2	3	0	⋯	3
File name	F_0	F_1	F_2	F_4	F_4	⋯	F_7
Bit designation	3 2 1 0	3 2 1 0	3 2 1 0	3 2 1 0	3 2 1 0	⋯	3 2 1 0
Digit designation (register Y) 0							
1							
2							
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
14							
15							

Fig. 2 RAM Address map

Interrupt Functions

The M58846-XXXSP provides 3-factor, 1-level vector interrupt capability, enabling unique branching addresses for each interrupt factor.

Interrupt factor		Interrupt address
Interrupt type	Causal condition	
External interrupt	Rising edge at the INT input pin	Page 1, address 0
Timer 1 interrupt	Timer 1 overflow	Page 1, address 2
Timer 2 interrupt	Timer 2 overflow	Page 1, address 4

Fig. 3 Vector Interrupt Addresses

The interrupt vector addresses are shown in Fig. 1.

An interrupt is generated whenever any of the casual conditions listed in Fig. 3 are satisfied at a time when the INTE flag is set to 1 (when the EI instruction is executed the INTE flag is set to 1, enabling interrupt; the DI instruction clears this flag to 0, prohibiting interrupts). If any of the interrupt causing conditions continues when the INTE flag is 0, an interrupt is generated when the INTE flag is set to 1.

The interrupts generated as a result of timer 1 and timer 2 overflow conditions can be software controlled, allowing confirmation of the overflow condition using a skip instruction.

When an interrupt program is used, one level of the three-level stack register is required, the remaining two levels being used for subroutines. After the interrupt program is started, the data pointer DP, register A, carry flag CY, and registers used by the interrupt program are saved. The instruction **RTI** is required to restore these before returning to the main program.

When an interrupt occurs, the microcomputer internal states are as follows.

(1) Program counter
The current address in the main program is stored in a stack register and the vector interrupt address as shown in Fig. 1 is loaded into the program counter.

(2) Interrupt flag INTE
The flag INTE is reset to disable further interrupts. This diabled state will continue even after return to the main program by the **RTI** instruction until the execution of an **EI** instruction.

(3) Skip flags
Skip flags are provided to discriminate skip instructions and consecutively described skip instructions. Each flag has its own stack within which the skip state is saved.

Timer/Event Counter (2 Lines)

The timer/event counter section consists of two lines (timers). As shown in Fig. 4, this section includes timer 1 and its overflow flag (1F) and timer 2 and its overflow register (register R), as well as the timer input/output port T and the timer control registers V and W.

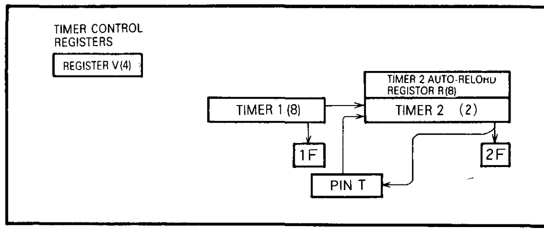


Fig. 4 Timer/event counter block diagram

The two timers (timer 1 and 2) are controlled by means of the timer control registers.

(1) Timer 1
Timer 1 is implemented using a 7-bit counter which divides the machine cycle (100kHz for a 600kHz clock frequency) by 127, setting the flag 1F every time an overflow condition occurs.
The timer is ready to count after a system reset has occured.

(2) Timer 2
Time 2 is implemented using an 8-bit binary counter and is provided with an auto-reload register (register R). Timer 2 data can be read using the **TAB2** instruction and register R may be set as well as read by means of the **T2AB** instruction. Starting and stopping the counter as well as the selection of the source (timer 1 or external input from port T) is controlled by the timer control registers. The overflow condition results in the setting of the flag 2F, after which timer 2 can be set with data once more by register R (auto-reload register) and continue counting.

(3) Timer I/O port T
This port can be selected by the counter control register as the source for timer 2. In addition, when another source has been selected, a pulse is available at this port every time timer 2 reaches the overflow condition.

(4) Timer 1 and 2 overflow flags 1F and 2F
These flags are set when the corresponding timer has reached the overflow condition. To test these flags, generation of an interrupt and skip instructions (**SNZ1**, **SNZ2**) can be used. The selection of which will be used is made by the timer control registers. By using either, these flags will be reset.

(5) Timer control registers V

The timer control register is used to perform the above described control functions. Instruction TVA is used to transfer control data to this register.

INPUT/OUTPUT PORTS

(1) Port K ($K_3 \sim K_0$)

This port is capable of performing 4-bit input by means of the **IAK** instruction or single-bit input by means of the **SZK** instruction.

The port K input circuits are TTL-compatible and may be provided with Schmitt circuits as a mask option.

In addition, pull-down transistors may be provided as a mask option.

(2) Port D ($D_{11} \sim D_0$)

This port consists of 12 bits which can be used for both input and output functions by means of the **SZD**, **SD**, and **RD** instructions. The output section provides individual bit latching and the contents of register Y can be used to designate a single bit of port D for output. When using the port for input, the output must be cleared to 0 first. The instructions **CLD** and **CLDS** can be used to clear all bits of the port to 0.

The outputs are open-drain circuits which can be provided with pull-down transistors as a mask option.

(3) Port F ($F_3 \sim F_0$)

This 4-bit port is controlled for output and input by the **OFA** and **IAF** instructions respectively. When using a bit for input, that bit output must first be set to 0. The outputs are open-drain circuits.

(4) Port S ($S_7 \sim S_0$)

This port can perform 8-bit output using the **OSAB**, **OSPA**, and **OSE** instructions and 4-bit input using the **IAS** instruction.

A built-in S output PLA has been provided which can code 4 bits of register A data arbitrarily and provide output using the **OSPA** instruction. The PLA output coding is a mask option.

When the port is used for input, the outputs must first be set to 0. All the port S bits may be set to 0 by means of the **CLS** and **CLDS** instructions.

The outputs are open-drain circuits which can be provided with pull-down transistors as a mask option.

(5) Port G ($G_3 \sim G_0$)

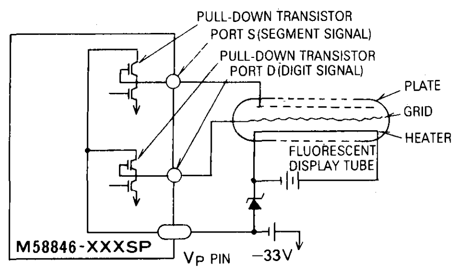
This port can be used to perform 4-bit output by means of the **OGA** instruction. The outputs are open-drain circuits.

(6) Port U

This port can be used to perform 1-bit output by means of the **SU** and **RU** instructions. The outputs are open-drain circuits.

VP PIN

This pin is used to supply the required voltage for the port D and port S pull-down transistors. Built-in pull-down transistors can be provided as a mask option for driving fluorescent display tubes, as shown in Fig. 5, eliminating the need for the usual externally connected pull-down resistors and resulting in a reduction in the number of system components.



Note 1. M58846-XXXSP can be used to directly drive fluorescent display tudes demanding up to −33 V of output level.

2. To provide complete blanking, the heater should be provided with a voltage of approximately 4V from the VP pin.

Fig.5 Fluorescent display tube drive circuit

RESET

When the RESET pin is kept high for at least 3 machine cycles, the reset state is enabled. After reset has been performed, when the RESET input is driven low, program execution will begin at page 0, address 0.

When the reset state is enabled, the following operations are performed.

(1) The program counter is set to 0, address 0. (PC) ← 0

(2) The interrupt mode is in the disabled state. INTE ← 0 (the same as for the execution of the **DI** instruction)

(3) The carry and data pointer selector CPS is set to 0, specifying DP and CY.

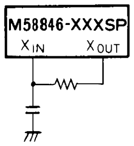
(4) Register V is set to 0. V ← 0_{16}

(5) The 3 interrupt flags, external interrupt flag (EXF), timer 1 overflow flag (1F), and timer 2 overflow flag (2F) are reset. EXF=1F=2F ← 0

(6) All outputs of ports D, F, S, G, U, and T are cleared to low (D)=(F)=(S)=(G)=(T) ← 0

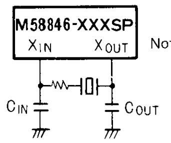
CLOCK GENERATOR CIRCUIT

A clock generator circuit has been built in, to allow control of the frequency by means of an externally connected RC circuit or ceramic resonator. In addition, an external clock signal may be applied at the X_{IN} pin, leaving the X_{OUT} pin open. Circuit examples are shown in Fig. 6~8.



Note 1. Since when using an RC circuit to determine frequency the frequency will vary in accordance with the device characteristics, the constants should be chosen so that the frequency falls within a specified limit even including the effect of these variations.

Fig. 6 External RC circuit



Note 2. The constants will depend on the frequency and type of the ceramic resonator used.

Fig. 7 Externally connected ceramic resonator

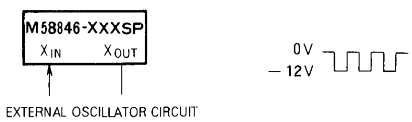


Fig. 8 External clock input circuit

MASK OPTIONS

The following mask options are available, specifiable at the time of initial ordering.

(1) Port S output PLA data
(2) Port K (K_3~K_0) pull-down transistors
(3) Port D (D_{11}~D_0) pull-down transistors
(4) Selection of port K input TTL-compatible Schmitt circuits
(5) Selection of interrupt input TTL-compatible Schmitt circuits
(6) Selection of RESET input TTL-compatible Schmitt circuits
(7) Selection of port T TTL-compatible Schmitt circuits

DOCUMENTATION REQUIRED UPON ORDERING

The following information should be provided when ordering a custom mask

(1) M58846-XXXSP mask confirmation sheet
(2) ROM data — 3 EPROM sets
(3) Port S output PLA coding — On confirmation sheets
(4) Port K pull-down transistors
(5) Port D pull-down transistors
(6) Port S pull-down transistors
(7) Selection of interrupt input TTL-compatible Schmitt circuits
(8) Selection RESET input TTL-compatible Schmitt circuits
(9) Selection of port T input TTL-compatible Schmitt circuits
(10) Selection of port K input TTL-compatible Schmitt circuits

MACHINE INSTRUCTIONS

Type of instruction	Mnemonic	Instruction code D_8 $D_7D_6D_5D_4$ $D_3D_2D_1D_0$	16mal notation	No. of words	No. of cycles	Functions	Skip condition	Flag CY	Description of operation
Register-to-register transfers	TAB	0 0001 1110	0 1 E	1	1	$(A) \leftarrow (B)$	–	x	Transfers conents of register B to register A.
	TBA	0 0001 1100	0 1 C	1	1	$(B) \leftarrow (A)$	–	x	Transfers contents of register A to register B.
	TAY	0 0001 1101	0 1 D	1	1	$(A) \leftarrow (Y)$	–	x	Transfers contents of register Y to register A.
	TYA	0 0000 1100	0 0 C	1	1	$(Y) \leftarrow (A)$	–	x	Transfers conents of register A to register Y.
	TEAB	0 0001 1010	0 1 A	1	1	$(E_7 \sim E_4) \leftarrow (B)$ $(E_3 \sim E_0) \leftarrow (A)$	–	x	Transfers contents of register A and B to register E.
	TEPA	0 0001 0110	0 1 6	1	1	$(E_7 \sim E_0) \leftarrow$ throughPLA $\leftarrow (A)$	–	x	Decodes contents of register A in the PLA and transfers result to register E.
RAM address	LXY x,y	0 11xx yyyy	0 Cy + x	1	1	$(X) \leftarrow x$ where $x = 0 \sim 3$ $(Y) \leftarrow y$ where $y = 0 \sim 15$	Written successively	x	Loads value of "x" into register X, and of "y" into Y. When LXY is written successively, the first is executed and successive ones are skipped.
	LZ z	0 0100 101z	0 4 A + z	1	1	$(Y) \leftarrow z$ where $z = 0, 1$	–	x	Loads value of "z" into register Z.
	INY	0 0000 0010	0 0 2	1	1	$(Y) \leftarrow (Y) + 1$	$(Y) = 0$	x	Increments contents of register Y by 1. Skips next instruction when new contents of register Y are "0".
	DEY	0 0000 0011	0 0 3	1	1	$(Y) \leftarrow (Y) - 1$	$(Y) = 15$	x	Decrements contents of register Y by 1. Skips nect instruction when new contents of register Y are "15".
	LCPS i	0 0100 000i	0 4 i	1	1	$(CPS) \leftarrow i$ where $i = 0, 1$	–	x	DP and CY are active when i=0, DP' and CY', when i=1.
RAM-accumulator	TAM j	0 0110 01jj	0 6 4 + j	1	1	$(A) \leftarrow (M(DP))$ $(X) \leftarrow (X) \forall$, where $j = 0 \sim 3$	–	x	Transfers the RAM contents addressed by the active DP to register A. Register X is then "exclusive OR-ed" with the value j in the instruction, and the result stored in register X.
	XAM j	0 0110 00jj	0 6 j	1	1	$(A) \leftrightarrow (M(DP))$ $(X) \leftarrow (X) \forall j$ where $j = 0 \sim 3$	–	x	Exchanges the contents of the RAM DP and register A. Contents of X are then "exclusive OR-ed" with the value j, and the result stored in register X.
	XAMD j	0 0110 10jj	0 6 8 + j	1	1	$(A) \leftrightarrow (M(DP))$ $(Y) \leftarrow (Y) - 1$ $(X) \leftarrow (X) \forall J$ where $j = 0 \sim 3$	$(Y) = 15$	x	Exchanges the contents of the RAM and register A. Contents of X are then "exclusive OR-ed" with the value j in the instruction, and the result stored in register X. The contents of register Y are decremented by 1, and when the result is 15, the next instruction is skipped.
	XAMI j	0 0110 11jj	0 6C + j	1	1	$(A) \leftrightarrow (M(DP))$ $(Y) \leftarrow (Y) + 1$ $(X) \leftarrow (X) \forall j$ where $j = 0 \sim 3$	$(Y) = 0$	x	Exchanges the contents of the RAM and register A. Contents of X are then "exclusive OR-ed" with the value j in the instruction and result stored in register X. The contents of register Y are incremented by 1, and when the result meets the next instruction is skipped with the marked skip condition.
Arithmetic operation	LA n	0 1011 nnnn	0 B n			$(A) \leftarrow n$ where $n = 0 \sim 15$	Written successively	x	Loads the value n in to register A. When LA is written consecutively the first is executed, and successive ones are skipped.
	AM	0 0000 1010	0 0 A	1	1	$(A) \leftarrow (A) + (M(DP))$	–	x	Adds the contents of the RAM to register A. The result is retained in register A, and the conents of flag CY are unaffected.
	AMC	0 0100 0011	0 4 3	1	1	$(A) \leftarrow (A) + (M(DP)) + (CY)$	–	0/1	Adds the RAM contents addressed by the active DP and contents of flag CY to register A. The result is stored in register A, and the carry in the active flag CY.
	AMCS	0 0101 0011	0 5 3	1	1	$(A) \leftarrow (A) + (M(DP)) + (CY)$ $CY \leftarrow$ Carry	$(CY) = 1$	0/1	Adds the contents of the RAM and flag CY to register A. The result is stored in register A and the carry in the CY, but the next instruction is skipped when a carry is produced.
	A n	0 1010 nnnn	0 A n	1	1	$(A) \leftarrow (A) + n$ where $n = 0 \sim 15$	Carry=0 where $n \neq 6$	x	Adds value n in the instruction to register A. The contents of flag CY are unaffected and their next instruction is skipped a carry is not produced, except when n=6.
	SC	0 0100 1001	0 4 9	1	1	$(CY) \leftarrow 1$	–	1	Sets active flag CY.
	RC	0 0100 1000	0 4 8	1	1	$(CY) \leftarrow 0$	–	0	Resets active flag CY.
	SZC	0 0010 1111	0 2 F	1	1		$(CY = 0)$	x	Skips next instruction when contents of the active flag CY are O.
	CMA	0 0000 1111	0 0 F	1	1	$(A) \leftarrow (\bar{A})$	–	x	Stores complement of register A in register A.
Bit operations	SB j	0 0100 11jj	0 4 C + j	1	1	$(M_j(DP)) \leftarrow 1$ where $j = 0 \sim 3$	–	x	Sets the jth bit of the RAM(immediate field value)addressed by the active DP (the bit designated by the value j in theinstruction)
	RB j	0 0101 11jj	0 5 C + j	1	1	$(M_j(DP)) \leftarrow 0$ where $j = 0 \sim 3$	–	x	Resets the jth bit of the RAM(immediate field value)addressed by the active DP (the bit designated by the value j in the instruction)
	SZB j	0 0010 00jj	0 2 j	1	1		$(M_j(DP)) = 0$ where $j = 0 \sim 3$		Skips next instruction when the contents of the jth bit of the RAM(immediate field value)addressed by the active DP (the bit which is designated by the value j in the instruction) are O.
Compares	SEAM	0 0010 0110	0 2 6	1	1		$(M(DP)) = (A)$	x	Skips next instruction when contents of register A are equal to the RAM contents addressed by the active DP.
	SEY y	0 0011 yyyy	0 3 y	1	1		$(Y) = y$ where $y = 0 \sim 15$	x	Skips next instruction when the content of register Y are equal to the value y in the instruction
Timer instruction	T2AB	0 1000 0101	0 8 5	1	1	$(R_7 \sim R_4) \leftarrow (B)$, $(2_7 \sim 2_4) \leftarrow (B)$ $(R_3 \sim R_0) \leftarrow (A)$, $(2_3 \sim 2_0) \leftarrow (A)$	–	x	Transfers the contents of registers A and B to timer 2 and the reload register.
	TAB2	0 1000 1010	0 8 A	1	1	$(B) \leftarrow (2_7 \sim 2_4)$ $(A) \leftarrow (2_3 \sim 2_0)$	–	x	Transfers the contents of timer 2 to registers A and B
	TVA	0 1000 0110	0 8 6	1	1	$(V) \leftarrow (A)$	–	x	Transfers the contents of register A to register V
	SNZ1	0 1000 0010	0 8 2	1	1		$(1F) = 1$	x	Skips the next instruction when the flag 1F is 1
	SNZ2	0 1000 0011	0 8 3	1	1		$(2F) = 1$	x	Skips the next instruction when the flag 2F is 1
A/D convetor operations	B xy	1 1xxx yyyy	1 8y + x	1	1	$(PC_L) \leftarrow 16x + y$	–	x	Jumps to address xy of the current page.
						$(PC_H) \leftarrow 3$, $(PC_L) \leftarrow 16x + y$			Jumps to address xy on page 3 when executed, provided that none of instructions, RT, RTS, BL, BML, BLA or BMLA was executed after execution of instruction BM or BMA.
	BL pxy	0 0111 PPPP 1 1xxx yyyy	0 7P 1 8y + x	2	2	$(PC_H) \leftarrow p$ $(PC_L) \leftarrow 16x + y$	–	x	Jumps to address xy of page p.
	BA xX	0 0000 0001 1 1xxx XXXX	0 0 1 1 8X + x	2	2	$(PC_L) \leftarrow 16x + (A)$	–	x	Jumps to address x(A) of the current page.
						$(PC_H) \leftarrow 3$, $(PC_L) \leftarrow 16x + (A)$			Jumps to the address x(A) of page 3 provided that none of instructions, RT, RTS, BL, BML, BLA or BMLA was executed alter execution of instruction BM or BMA.

Type of instruction	Mnemonic	Instruction code D_8 $D_7D_6D_5D_4$ $D_3D_2D_1D_0$	16mal notation	No. of words	No. of cycles	Functions	Skip conditions	Flag CY	Description of operation
Jumps	BLA pxy	0 0000 0001 0 0111 PPPP 1 1XXX XXXX	0 01 0 7P 1 8X + X	3	3	$(PC_H) \leftarrow p$ $(PC_L) \leftarrow 16x + (A)$	—	X	Jumps to the address x(A) of page p.
Subroutine calls	BM xy	1 0xxx yyyy	1 xy	1	1	$(SK_2) \leftarrow (SK_1) \leftarrow (SK_0) \leftarrow (PC)$ $(PC_H) \leftarrow 2,\ (PC_L) \leftarrow 16x + y$	—	X	Calls for the subroutine starting at address xX on page 2
						$(PC_H) \leftarrow 2,\ (PC_L) \leftarrow 16x + y$			Jumps to address xy of page 2 provided that none of instructions. RT, RTS, BL, BML, BLA or BMLA was executed after the execution of instructions BM or BMA.
	BML pxy	0 0111 PPPP 1 0xxx yyyy	0 7P 1 xy	2	2	$(SK_2) \leftarrow (SK_1) \leftarrow (SK_0) \leftarrow (P_C)$ $(PC_H) \leftarrow p,\ (PC_L) \leftarrow 16x + y$	—	X	Calls for the subroutine starting at address xy of page p.
	BMA xX	0 0000 0001 1 0xxx XXXX	0 01 1 XX	2	2	$(SK_2) \leftarrow (SK_1) \leftarrow (SK_0) \leftarrow (PC)$ $(PC_H) \leftarrow 2,\ (PC_L) \leftarrow 16x + (A)$	—	X	Calls for the subroutine starting at address x(A) of page 2.
						$(PC_H) \leftarrow 2,\ (PC_L) \leftarrow 16x + (A)$			Jumps to address x(A) of page 2 provided that none of instructions. RT, RTS, BL, BML, BLA or BMLA was executed after the execution of instructions BM or BMA.
	BMLA pxX	0 0000 0001 0 0111 PPPP 1 0xxx XXXX	0 01 0 7P 1 XX	3	3	$(SK_2) \leftarrow (SK_1) \leftarrow (SK_0) \leftarrow (PC)$ $(PC_H) \leftarrow p,\ (PC_L) \leftarrow 16x + (A)$	—	X	Calls for the subroutine starting at address x(A) of page p.
Program returns	RTI	0 0100 0110	0 46	1	1	$(PC) \leftarrow (SK0) \leftarrow (SK1) \leftarrow (SK2)$ Resets interrupt flip-flop	—	X	Returns from interrupt routine to main routine. The internal flip-flop is restored to the value held immediately before the interrupt.
	RT	0 0100 0100	0 44	1	1	$(PC) \leftarrow (SK0) \leftarrow (SK1) \leftarrow (SK2)$	—	X	Returns to the main routine from the subroutine.
	RTS	0 0100 0101	0 45	1	1	$(PC) \leftarrow (SK0) \leftarrow (SK1) \leftarrow (SK2)$	Unconditional skip	X	Returns to the main routine from the subroutine, and unconditionally skips the next instruction.
Input/output	CLD	0 0001 0011	0 13	1	1	$(D) \leftarrow 0$	—	X	Clears port D.
	CLS	0 0001 0000	0 10	1	1	$(S) \leftarrow 0$	—	X	Clears port S.
	CLDS	0 0001 0001	0 11	1	1	$(D) \leftarrow 0$ $(S) \leftarrow 0$	—	X	Clears ports S and D.
	SD	0 0001 0101	0 15	1	1	$(D(Y)) \leftarrow 1$ where $Y = 0 \sim 11$	—	X	Sets the bit of port D that is designated by register Y.
	RD	0 0001 0100	0 14	1	1	$(D(Y)) \leftarrow 0$ where $Y = 0 \sim 11$	—	X	Resets the bit of port D that is designated by register Y
	SZD	0 0010 1011	0 2B	1	1		$(D(Y)) = 0$ where $Y = 0 \sim 11$	X	Skips the next instruction if the contents of the bit of port D that is designated by register Y and 0.
	OSAB	0 0001 1011	0 1B	1	1	$(S_7 - S_4) \leftarrow (B)$ $(S_3 - S_0) \leftarrow (A)$	—	X	Output contents of registers A and B to port S.
	OSPA	0 0001 0111	0 17	1	1	$(S_7 - S_4) \leftarrow$ through PLA $\leftarrow (A)$	—	X	Decords conents of register A by PLA and the result is output to port S.
	OSE	0 0000 1011	0 0B	1	1	$(S) \leftarrow (E)$	—	X	Outputs contents of register E to port S.
	IAS i	0 0101 010i	0 54 + i	1	1	$i = 0: (A) \leftarrow (S_7 - S_4)$ $i = 1: (A) \leftarrow (S_3 - S_0)$	—	X	Transfers from port S to register A. The high-order four bits of port S are transferred when the value of i in the instruction is 0 or the low-order four bits are transferred when the value of i is1.
	OFA	0 1000 0001	0 81	1	1	$(F) \leftarrow (A)$	—	X	
	IAF	0 1000 1100	0 8C	1	1	$(A) \leftarrow (F)$	—	X	Transfers the port F input to register A.
	OGA	0 1000 0100	0 84	1	1	$(G) \leftarrow (A)$		X	Outputs contents of register A to port G.
	IAK	0 0101 0111	0 57	1	1	$(A) \leftarrow (K)$	—	X	Transfer the port K input to register A.
	SZK j	0 0101 10jj	0 58 + j	1	1		$(K(j)) = 0$	X	Skips the next instruction if the jth bit of port K input is 0.
	SU	0 0000 0111	0 07	1	1	$(U) \leftarrow 1$	—	X	Sets port U to 1.
	RU	0 0000 0110	0 06	1	1	$(U) \leftarrow 0$	—	X	Resets port U to 0.
Interrupts	EI	0 0000 0101	0 05	1	1	$(INTE) \leftarrow 1$		X	Sets interrupt flag INTE to enable interrupts.
	DI	0 0000 0100	0 04	1	1	$(INTE) \leftarrow 0$		X	Resets interrupt flag INTE to disabel interrupts.
Misc	NOP	0 0000 0000	0 00	1	1	$(PC_L) \leftarrow (PC_L) + 1$		X	No operation.

Symbol	Contents	Symbol	Contents	Symbol	Contents
A	4-bit register (accumulator)	CY	1-bit carry flag	K	4-bit port
B	4-bit register	1F	1-bit timer 1 overflow flag	S	8-bit port
E	8-bit register	2F	1-bit timer 2 overflow flag	INTE	Interrupt enable flag
R	8-bit timer overflow register	xx	2-bit binary variable	INT	Interrupt request signal
V	4-bit register	yyyy	4-bit binary variable	EXF	1-bit external interrupt flag
X	2-bit register	z	1-bit binary variable	←	Shows the direction of data flow.
Y	4-bit register	nnnn	4-bit binary constant	( )	Indicates the contents of register, memory,etc.
Z	1-bit register	i	1-bit binary constant	∀	Exclusive OR
DP	7-bit data pointer, combination of registers, X, Y and Z	ii	2-bit binary constant	—	Negation.
PC_H	The high-order four bits of the program counter.	XXXX	4-bit unknown binary number	X	Indicates flag is unaffected by instruction execution
PC_L	The low-order seven bits of the program counter.	1	Timer 1	xy	Label used to indicate the address xxx yyyy
PC	11-bit program counter combination of PC_H and PC_L	2	Timer 2	CPS	Indicate which data pointer and carry flag are active
SK0	11-bit stack register	D	12-bit port	pxy	Label used to indicate the address xxx yyyyon page ppp.
SK1	11-bit stack register	F	4-bit port	C + x	Hexadecimal number C + binary number X
SK2	11-bit stack register	G	4-bit port		
		U	1-bit port		

4

SINGLE-CHIP 4-BIT MICROCOMPUTER WITH TWO TIMER/EVENT COUNTER

INSTRUCTION CODE LIST

D_3~D_0 \ D_8~D_4		0 0000	0 0001	0 0010	0 0011	0 0100	0 0101	0 0110	0 0111	0 1000	0 1001	0 1010	0 1011	0 1100	0 1101	0 1110	0 1111	1 0000 ~ 1 0111	1 1000 ~ 1 1111
	Hexadecimal notation	0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	0 A	0 B	0 C	0 D	0 E	0 F	10 ~ 17	18 ~ 1F B
0000	0	NOP	CLS	SZB 0	SEY 0	LCPS 0	—	XAM 0	BL BML	—	—	A 0	LA 0	LXY 0,0	LXY 1,0	LXY 2,0	LXY 3,0	BM	B
0001	1	BA BMA BLA BMLA	CLDS	SZB 1	SEY 1	LCPS 1	—	XAM 1	BL BML	OFA	—	A 1	LA 1	LXY 0,1	LXY 1,1	LXY 2,1	LXY 1,1	BM	B
0010	2	INY	—	SZB 2	SEY 2	—	—	XAM 2	BL BML	SNZ1	—	A 2	LA 2	LXY 0,2	LXY 1,2	LXY 2,2	LXY 3,2	BM	B
0011	3	DEY	CLD	SZB 3	SEY 3	AMC	AMCS	XAM 3	BL BML	SNZ2	—	A 3	LA 3	LXY 0,3	LXY 1,3	LXY 2,3	LXY 3,3	BM	B
0100	4	DI	RD	—	SEY 4	RT	IAS 0	TAM 0	BL BML	OGA	—	A 4	XA 4	LXY 0,4	LXY 1,4	LXY 2,4	LXY 3,4	BM	B
0101	5	EI	SD	—	SEY 5	RTS	IAS 1	TAM 1	BL BML	T2AB	—	A 5	LA 5	LXY 0,5	LXY 1,5	LXY 2,5	LXY 3,5	BM	B
0110	6	RU	TEPA	SEAM	SEY 6	RTI	—	TAM 2	BL BML	TVA	—	A 6	LA 6	LXY 0,6	LXY 1,6	LXY 2,6	LXY 3,6	BM	B
0111	7	SU	OSPA	—	SEY 7	—	IAK	TAM 3	BL BML	—	—	A 7	LA 7	LXY 0,7	LXY 1,7	LXY 2,7	LXY 3,7	BM	B
1000	8	—	—	—	SEY 8	RC	SZK 0	XAMD 0	BL BML	—	—	A 8	LA 8	LXY 0,8	LXY 1,8	LXY 2,8	LXY 3,8	BM	B
1001	9	—	—	—	SEY 9	SC	SZK 1	XAMD 1	BL BML	—	—	A 9	LA 9	LXY 0,9	LXY 1,9	LXY 2,9	LXY 3,9	BM	B
1010	A	AM	TEAB	—	SEY 10	LZ 0	SZK 2	XAMD 2	BL BML	TAB2	—	A 10	LA 10	LXY 0, 10	LXY 1, 10	LXY 2, 10	LXY 3, 10	BM	B
1011	B	OSE	OSAB	SZD	SEY 11	LZ 1	SZK 3	XAMD 3	BL BML	—	—	A 11	LA 11	LXY 0, 11	LXY 1, 11	LXY 2, 11	LXY 3, 11	BM	B
1100	C	TYA	TBA	—	SEY 12	SB 0	RB 0	XAMI 0	BL BML	IAF	—	A 12	LA 13	LXY 0, 12	LXY 1, 12	LXY 2, 12	LXY 3, 12	BM	B
1101	D	—	TAY	—	SEY 13	SB 1	RB 1	XAMI 1	BL BML	—	—	A 13	LA 13	LXY 0, 13	LXY 1, 13	LXY 2, 13	LXY 3, 13	BM	B
1110	E	—	TAB	—	SEY 14	SB 2	RB 2	XAMI 2	BL BML	—	—	A 14	LA 14	LXY 0, 14	LXY 1, 14	LXY 2, 14	LXY 3, 14	BM	B
1111	F	CMA	—	SZC	SEY 15	SB 3	RB 3	XAMI 3	BL BML	—	—	A 15	LA 15	LXY 0, 75	LXY 1, 15	LXY 2, 15	LXY 3, 15	BM	B

Note 1. This list shows the machine codes and corresponding machine instructions. D_3~D_0 indicate the low order 4 bits of the machine code and D_8~D_4 indicate the high-order 5 bits. Hexadecimal numbers are also shown that represent the codes. An instruction may consist of one, two ,or three words, but only the first word is listed Code combination indicated with a bar (—) must not be used.

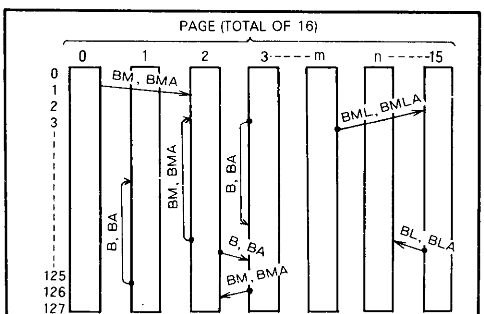
Note 2. Two-Word Instructions

	Second word
BL	1 1xxx yyy
BML	1 0xxx yyyy
BA	1 1xxx XXXX
BMA	1 0xxx XXXX

Three-Word Instructions

	Second word	Third word
BLA	0 0111 pppp	1 1xxx XXXX
BMLA	0 0111 pppp	1 0xxx XXXX

Note 3. Relationships for branching by means of branching instructions and subroutine calling instructions.



MITSUBISHI ELECTRIC

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	0.3～−20	V
V_I	Input voltage (ports D and S, and input V_P)		0.3～−33	V
V_I	Input voltage, inputs other than ports D and S, and input V_P		0.3～−20	V
V_O	Output voltage, ports D and S		0.3～−33	V
V_O	Output voltage, other outputs than ports D and S		0.3～−20	V
Pd	Power dissipation	Ta=25°C	1100	mW
Topr	Operating temperature		−10～70	°C
Tstg	Storage temperature		−40～125	°C

RECOMMENDED OPERATING CONDITIONS (Ta=−10~70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	−11	−12	−13	V
V_{SS}	Supply voltage		0		V
V_P	Pull-down transistor supply voltage	0		−33	V
V_{IH}	High-level input voltage, ports S and F	−1.5		0	V
V_{IH}	High-level input voltage, port D	−1.0		0	V
$V_{IH(\phi)}$	High-level clock input voltage	−1.5		0	V
V_{IL}	Low-level input voltage, inputs other than ports D and S	V_{DD}		−4.2	V
V_{IL}	Low-level input voltage, ports D and S	−33		−4.2	V
$V_{IL(\phi)}$	Low-level clock input voltage	V_{DD}		$V_{DD}+2$	V
V_{OL}	Low-level output voltage, ports D and S	−33		0	V
f (φ)	Internal clock oscillation frequency	300		600	kHz

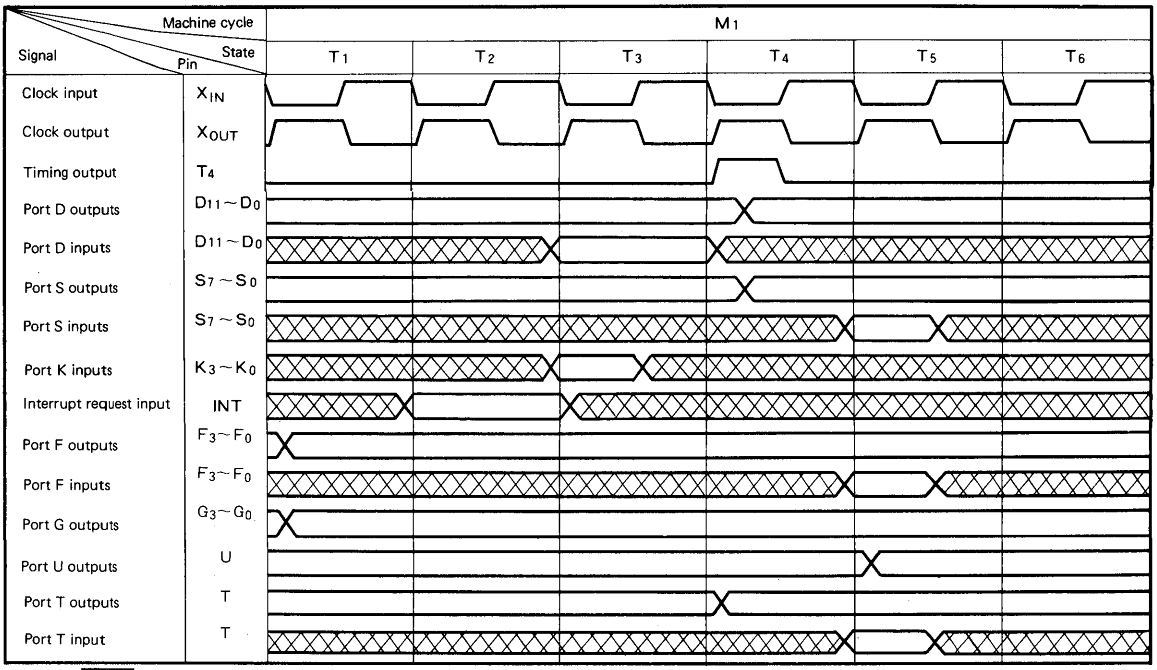
Note 1. $V_{IL(\phi)}$ is specified for the maximum V_{DD} value

ELECTRICAL CHARACTERISTICS (T_a=−10~70°C, V_{DD}=−12V 10%, V_{SS}=0V, f(φ)=300~600kHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{IL}	Low-level output voltage, port F		V_{DD}		−4.2	V
V_{OH}	High-level output voltage, port D	V_{DD}=−12V, I_{OH}=−15mA, Ta=25℃	−2.5			V
V_{OH}	High-level output voltage, ports S and F	V_{DD}=−12V, I_{OH}=−8mA (port S) I_{OH}=−5mA (port F), Ta=25°C	−2.5			V
V_{T-}	Negative threshold voltage (Schmitt input mask option)	V_{DD}=−12V, Ta=25°C	−7		−4	V
$V_{T+}-V_{T-}$	Hysteresis (Schmitt input mask option).	V_{DD}=−12V, Ta=25°C	2		3.5	V
$I_{I(\phi)}$	Clock input current	$V_{I(\phi)}$=−12V, Ta=25°C		−20	−40	μA
I_{IH}	High-level input current, port K (with pull-down resistors)	V_{DD}=−12V, V_{IH}=0V, Ta=25°C	50		250	μA
I_{IH}	High-level input current, ports D and S (with pull-down resistors)	V_P=−33V, V_{IH}=0V, Ta=25°C	80		280	μA
I_{OH}	High-level output current, port D (Note 2)	V_{DD}=−12V, V_{OH}=−2.5V, Ta=25°C			−15	mA
I_{OH}	High-level output current, ports S and F	V_{DD}=−12V, V_{OH}=−2.5V, Ta=25℃		−5(port F) −8(port S)		mA
I_{OL}	Low-level output current, ports D and S	V_{OL}=−33V, Ta=25℃			−33	μA
I_{OL}	Low-level output current, port outputs	V_{DD}=−12V, Ta=25℃			−33	μA
I_{DD}	Supply current	V_{DD}=−12V, Ta=25℃		21		mA
Ci (φ)	Clock input capacitance	$V_{DD}=X_{OUT}=V_{SS}$, f=1MHz, 25mVrms			10	pF

Note 2. Currents are taken as positive when flowing into the IC (zero signal condition), with the minimum and maximum values as absolute values.
3. It is possible to connect up to 5 lines of the port D at maximum ratings (−15mA) or all lines of port S and F at maximum ratings of (−8mA) and (−5mA) respectively.

BASIC TIMING



Note 1. The crosshatch area indicates invalid input

SINGLE-CHIP 4-BIT MICROCOMPUTER

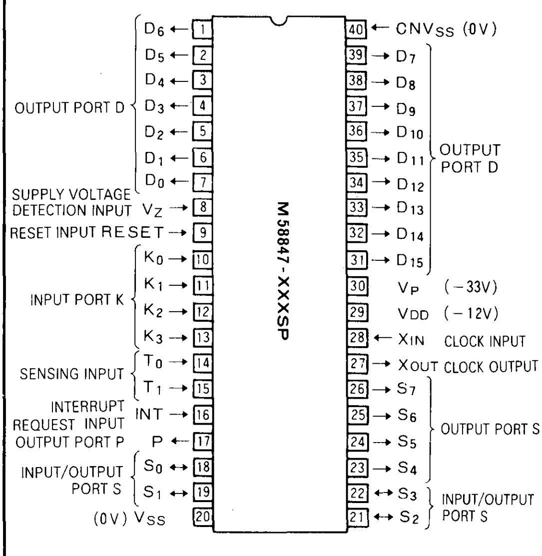
DESCRIPTION

The M58847-XXXSP is a single-chip 4-bit microcomputer fabricated using p-channel aluminum gate ED-MOS technology. It is housed in a 40-pin shrink plastic molded DIL package and provides 25 output lines, 4 input lines, 2 sensing lines and 1 interrupt input. Because of its low power consumption, it is ideal for consumer electronics applications requiring many control signals.

FEATURES

- Basic machine instructions 52
- Instruction execution time (for 1 word instructions using a 400kHz clock frequency) 15μs
- Memory capacity: ROM .... 2048 words x 9 bits
  RAM .... 128 words x 4 bits
- Single −12V power supply
- Subroutine nesting .... 2 levels
- Interrupt function .... 1 factor, 1 level
- Input (port K) .... 4 ports
- Output (ports D and P) .... 17 ports
- Input/output (port S) .... 8 ports
- Sensing input (port T) .... 2 ports
- High withstanding voltage and large current output
- Built-in pull-down transistors (ports T, K, D, P, and S, mask option)
- Built-in clock generator circuit

PIN CONFIGURATION (TOP VIEW)

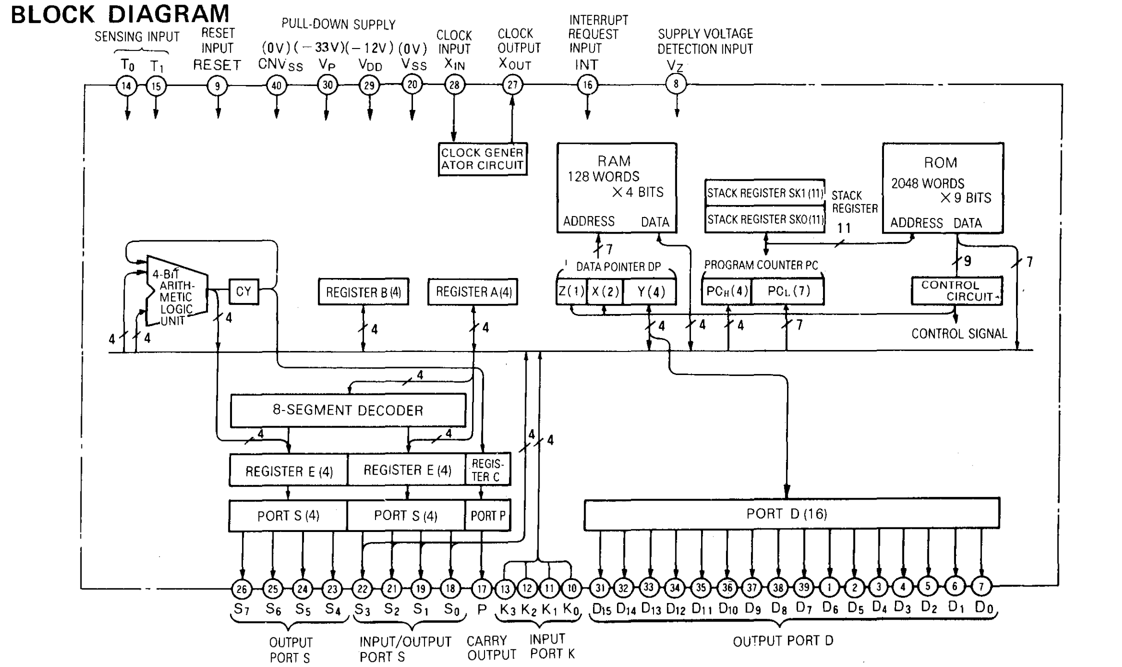


Outline 40P4B

APPLICATIONS

- VTRs, TVs, cassette decks
- Microwave ovens, air conditioners, heaters, washing machines, home sewing machines
- Office equipment, copying machines, medical equipment
- Educational equipment, electronic games

BLOCK DIAGRAM



SINGLE-CHIP 4-BIT MICROCOMPUTER

PERFORMANCE SPECIFICATIONS

Parameter			Performance
Basic machine instructions			52
Instruction execution time			15 μs (1-word instructions using a clock frequency of 400 kHz)
Clock frequency			240kHz ~ 400kHz
Memory capacity	ROM		2048 words x 9 bits
	RAM		128 words x 4 bits
Input/output ports	K	Input	4 bits x 1
	S	Output	8 bits x 1
		Input	4 bits x 1
	P	Output	1 bit x 1
	D	Output	1 bit x 16
	T	Sensing input	1 bit x 2
Subroutine nesting			2 levels (including one level of interrupt)
Clock generator			Built-in (externally connected RC circuit or ceramic resonator)
I/O chracteristics of ports	I/O withstanding voltage		−33V
	Ports P and S output current		− 8 mA
	Port D output current		−15mA
Supply voltage	V_{DD}		−12V
	V_{SS}		0 V
Device structure			p-channel aluminum gate ED-MOS
Package			40-pin shrink plastic molded DIL package
Power dissipation			10mW (typ)

PIN DESCRIPTIONS

Pin	Name	Input or output	Function
V_{DD}	Supply voltage		V_{DD} and V_{SS} are the power supply pins. V_{DD} should be connected to −12V±10% and V_{SS} should be grounded. V_P is the pull-down supply voltage input for the pull-down transistors (mask options) for ports P, S, and D.
V_{SS}	Supply voltage		
V_P	Pull-down voltage input		
V_Z	Supply voltage detection input	In	This input pin is provided for use in detecting a drop in the supply voltage.
RESET	Reset input	In	This pin is used to intialize the microcomputer. If it is held high for at least two machine cycles after V_{DD} reaches to within 10% of −12V, the reset condition is enabled.
CNV_{SS}	CNV_{SS} input	In	This pin is not reserved for customer use but should be connected to V_{SS}.
X_{IN}	Clock input	In	These are the input and output pins for the built-in clock generator. A ceramic resonator element (240~400 kHz) or RC circuit may be connected to these pins to provide the required oscillation stability.
X_{OUT}	Clock output	Out	
T_1, T_0	Sensing input	In	Sensing input pin
$K_3 \sim K_0$	Input port K	In	4-bit input port
$S_3 \sim S_0$	I/O port S	In/out	$S_7 \sim S_4$ and P comprise an output port $S_3 \sim S_0$ comprise an input/output port
$S_7 \sim S_4$	Output port S	Out	
P	Output port P	Out	
$D_{15} \sim D_0$	Output port D	Out	The individual bits of this 16-bit output port may be set and reset separately.
INT	Interrupt request input	In	This interrupt signal input pin triggers on the input signal edge.

BASIC FUNCTION BLOCKS

Program Memory (ROM)

This 2048 word x 9-bit ROM can be programmed with machine instruction codes in accordance with the customer's specifications. It consists of 16 pages, each containing an address range of 0~127.

The page is specified by the upper order 4 bits (PC_H) of the program counter.

The address within a particular page is specified by the lower order 7 bits which form a polynomial counter (PC_L). When the last address is reached (127), the address wraps around to the 0th address.

The BL instruction is used to branch to a different page than the current page. While the program counter is in reality a polynomial counter, a cross-assembly technique is used to allow the programmer to think of this counter as a normal pure binary counter, for ease in programming.

Page 0 and page 1 are special pages used for subroutine calls. The single-word instruction BM can be used to call a subroutine on page 0 from any arbitrary page. When the BM instruction is executed, the SM flag is set and until any of the BL, BML, RT or RTS instructions are executed, the B and BM instructions are used for functions differing from their normal functions.

Until any of the above listed instructions is executed after an BM instruction execution, the B instruction has the effect of branching to the 1st page and the BM instruction has the effect of branching to the 0th page. The flag SM is reset when the BL, BML, RT, or RTS instruction is executed.

Fig. 1 shows the ROM address map.

Program Counter (PC)

This counter is used to specify ROM addresses and the sequence of read-out of instructions stored in ROM. The upper 4 bits (PC_H) are used to specify the page in ROM and the lower 7 bits (PC_L) of which are a polynomial counter used to specify the address on the specified page.

Stack Registers (SK_0, SK_1)

These registers are used to temporarily store the contents of the PC while executing subroutines or interrupt programs until the program returns to the main routine.

The stack registers are organized in 2 words of 11 bits, allowing 2 levels of subroutine nesting.

Data Memory (RAM)

This 512-bit (128 words x 4 bits) RAM is used to store both processing and control data. One RAM word consists of 4 bits with bit manipulation possible over the entire storage area. The 128 words are arranged in 2 file groups x 4 files x 16 digits x 4 bits.

The word addresses for the data RAM are specified by means of the data pointer which consists of 1 bit of the register Z, 2 bits of the register X and 4 bits of the register Y. Fig. 2 shows the RAM address map. There are 8 files (F_0~F_7) consisting of 16 words of 4 bits, which are convenient as a 16-digit register.

The specification for these file grimps is made by registers Z and X.

Data Pointer (DP)

This register is used to designate RAM addresses as well as bit position for the output port D. The data pointer is a 7-bit register, the uppermost bit of which is register Z which is used to specify the RAM file group, the central 2 bits of which form register X which is used to specify the RAM file, and the lower 4 bits of which form register Y which is used to specify the digit within the file. In addition, when the register Z's bit is 1, register Y is used to specify the bit position for the output port D.

4-bit Arithmetic Logic Unit (ALU)

This unit executes 4-bit arithmetic and logical operations by means of a 4-bit adder and related logic circuitry.

Register A and Carry Flag (CY)

Register A is a 4-bit accumulator that constitutes the basis for arithmetic operations. Data processing operations such as arithmetic and logical operations, data transfer, exchange, conversion, and data input/output are executed by means of this register. The carry flag CY is used to store carry or overflow after execution of arithmetic and logical operations by the arithmetic unit. The carry flag may also be used as a 1-bit flag.

PC_H		Page designation																																
PC_L		0									1									…						15								
Bit designation		8	7	6	5	4	3	2	1	0	8	7	6	5	4	3	2	1	0	8	7		2	1	0	8	7	6	5	4	3	2	1	0
Address designation	0																																	
	1																																	
	2																																	
	⋮	⋮									⋮									⋮						⋮								
	126																																	
	127																																	

Fig. 1 ROM address map

File designation	Register Z	0																1								
	Register X	0				1				2				3				0				…	3			
File name		F_0				F_1				F_2				F_3				F_4				…	F_7			
Bit designation		3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0	…	3	2	1	0
Digit designation (Register Y)	0																									
	1																									
	2																									
	⋮	⋮				⋮				⋮				⋮				⋮					⋮			
	14																									
	15																									

Fig. 2 RAM address map

Registers B, E, and C

Register B is composed of 4 bits and can be used as a 4-bit temporary storage register or for 8-bit data transfer in conjunction with register A. Register E is an 8-bit register which can be used for temporary data storage or as an auxiliary register for input/output port S, and it also has left shift capability. Register C is a 1-bit register, to which the contents of the carry flag can be transferred. It can also be used to perform left shift when linked to register E.

Interrupt Functions

An interrupt input has been provided to allow the M58847-XXXSP to accept external interrupts. When the input signal changes from low to high, an edge-sensing flag is set, causing the interruption of the normally executed program if the interrupt enable flag is set. When an interrupt is received, the following things occur.

(1) The program counter and SM flag are saved on each stack.
(2) The program counter is set to the 0th address on page 2.
(3) The SM flag and edge-sensing flag are reset.
(4) The interrupt enable flag is reset.

In the above state the program begins at page 2, address 0, the first address of the interrupt program. The instruction RTI is used to end the interrupt program and return the processor to the main program flow.

Since the SM flag has a single-level stack, one level of interrupt is possible. The program counter, however, has a two level stack, enabling subroutine nesting of one level after an interrupt uses one of these levels.

The microcomputer can accept an interrupt request in the following conditions.

When not executing a B, BL, BM, BML, LA, LXY, RT, RTS, RTI, DI, or EI instruction or not executing a skip operation and the interrupt enable flag is set.

Input/Output Ports

Ports T, K, S, D, and P may be provided with pull-down transistors as a mask option. Fig. 3 shows the circuits for the input/output ports.

In addition, the contents of the register A are decoded to 8 bits by built-in 8 segment decoder and transferred to register E or port S. The decoder function is fixed and not available in special forms as a mask option. Table 1 shows the decoder function.

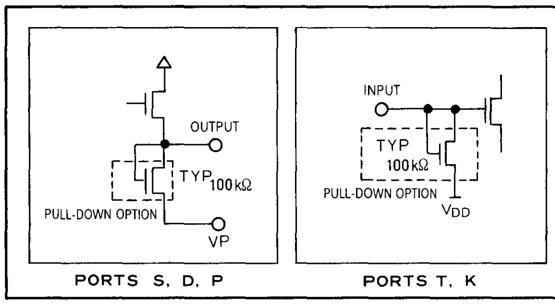


Fig. 3 Input/output circuits

Table 1 Decoder function table

Hexadecimal value	Register A				Port S output								Display
	A_3	A_2	A_1	A_0	S_7	S_6	S_5	S_4	S_3	S_2	S_1	S_0	
0	0	0	0	0	L	L	H	H	H	H	H	H	0
1	0	0	0	1	L	L	L	L	L	H	H	L	1
2	0	0	1	0	L	H	L	H	H	L	H	H	2
3	0	0	1	1	L	H	L	L	H	H	H	H	3
4	0	1	0	0	L	H	H	L	L	H	H	L	4
5	0	1	0	1	L	H	H	L	H	H	L	H	5
6	0	1	1	0	L	H	H	H	H	H	L	H	6
7	0	1	1	1	L	L	H	L	L	H	H	H	7
8	1	0	0	0	L	H	H	H	H	H	H	H	8
9	1	0	0	1	L	H	H	L	H	H	H	H	9
A	1	0	1	0	L	H	L	H	H	H	L	L	o
B	1	0	1	1	H	L	L	L	L	L	L	L	-
C	1	1	0	0	L	H	H	H	H	L	L	H	E
D	1	1	0	1	L	L	H	H	H	L	L	H	C
E	1	1	1	0	L	H	L	L	L	L	L	L	—
F	1	1	1	1	L	L	L	L	L	L	L	L	Blank

Reset

The RESET input has been provided to enable initialization of the microcomputer. If the input is kept high for at least two machine cycles after the supply voltage V_{DD} reaches to within 10% of −12V, the microcomputer will be reset, enabling the following states.

(1) The program counter is set to 0 (PC) ← 0
(2) Ports S, P and D are turned off (S) ← 0 (P) ← 0 (D) ← 0
(3) Flag SM is reset (SM) ← 0
(4) The edge-sensing flag is reset
(5) The interrupt enable flag is reset (INTE) ← 0

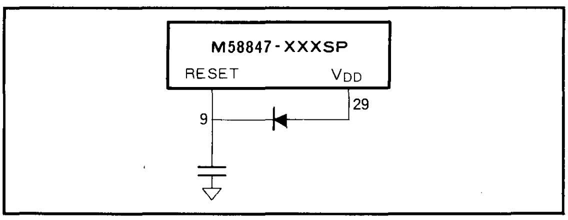


Fig. 4 Power-on reset circuit

In addition, when the supply voltage V_{DD} is in the range −7V ~ −13.2V and the V_Z input is driven high, an internal transistor is turned on and the RESET pin is set to the level of V_{SS}. Even if the V_Z pin returns to low, the internal transistor will remain turned on until the RESET pin is driven high. By using this function it is possible to sense temporary drops in the supply voltage to allow reset at these times to return to normal operation.

Clock Generator Circuits

A clock generator circuit has been built in to allow control of the frequency by means of an externally connected RC circuit or ceramic resonator. The choice of frequency determining element is made at the time of purchase as a mask option. Circuit examples are shown in Fig. 5~7.

Mask Options

- Port T pull-down transistors
- Port K pull-down transistors
- Port D pull-down transistors
- Port S pull-down transistors
- Port P pull-down transistors
- Oscillation conditions

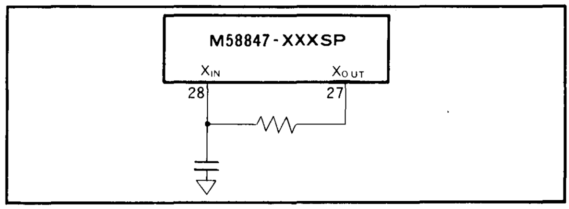


Fig. 5 External RC circuit

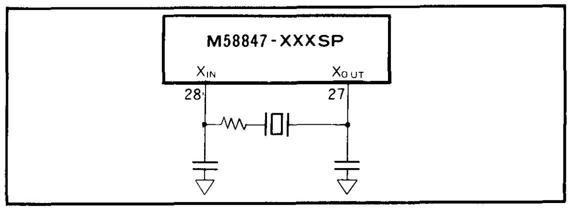


Fig. 6 External ceramic resonator

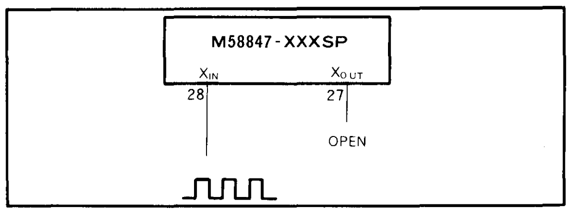


Fig. 7 External clock circuit

Documentation Required upon Ordering

The following information should be provided when ordering a custom mask.

(1)	M58847-XXXSP mask confirmation sheet	
(2)	ROM data	3 EPROM sets
(3)	Port D pull-down transistors	On confirmation sheets
(4)	Port S pull-down transistors	On confirmation sheets
(5)	Port P pull-down transistors	On confirmation sheets
(6)	Port T pull-down transistors	On confirmation sheets
(7)	Port K pull-down transistors	On confirmation sheets
(8)	Oscillation conditions	On confirmation sheets

MACHINE INSTRUCTIONS

Type of instruction	Mnemonic	Instruction code D_8 $D_7D_6D_5D_4$ $D_3D_2D_1D_0$	16mal notation	No. of words	No. of cycles	Functions	Skip conditions	Flag CY	Description of operation
Register-to-register transfers	TAB	0 1011 1001	0B9	1	1	(A)←(B)	—	×	Transfers contents of register B to register A.
	TBA	0 1011 0101	0B5	1	1	(B)←(A)	—	×	Transfers contents of register A to register B.
	TAY	0 1011 1010	0BA	1	1	(A)←(Y)	—	×	Transfers contents of register Y to register A.
	TYA	0 1011 0110	0B6	1	1	(Y)←(A)	—	×	Transfers contents of register A to register Y.
	TEAB	0 1011 0001	0B1	1	1	$(E_7 \sim E_4)$←(B), $(E_3 \sim E_0)$←(A), (C)←(CY) (C)←(CY)	—	×	Transfers contents of registers A and B to register E and the contents of the carry flag to register C.
	TEPA	0 1011 0000	0B0	1	1	$(E_7 \sim E_0)$← 8-segment decoder ←(A), (C)←(CY)	—	×	Decodes contents of register A in the 8-segment decoder and transfers result to register E. The contents of the carry flag are transferred to register.
	TXA	0 1111 0101	0F5	1	1	(X)←(A_1A_0) (Z)←$(\overline{A_2})$ (CY)←$(\overline{A_3})$	—	0/1	Transfers the first and second bits of the register A to register X, complement of the third bit to register Z, and complement of the fourth bit to the carry flag CY.
	TAX	0 1111 1001	0F9	1	1	(A_1A_0)←(X) (A_2)←$(\overline{Z})$ (A_3)←$(\overline{CY})$	—	×	Transfers the contents of register X to the first and second bits of register A, complement of the contents of register Z to the third bit of register A, and complement of the contents of the carry flag CY to the fourth bit of register A.
RAM addresses	LXY x,y	0 01yy yyxx	048 + y_1 2 + y_3 4 + y_0 y_2 + x	1	1	(X)←x, where, x = 0 ~ 3 (Y)←y, where, y = 0 ~ 15	Written successively	×	Loads value of "x" into register X, and of "y" into Y. When LXY is written successively, the first is executed and successive ones are skipped.
	LZ z	0 0000 110z	00C + z	1	1	(Z)←z, where z = 0, 1	—	×	Loads value of "z" into register Z.
	INY	0 1110 0100	0E4	1	1	(Y)←(Y)+1	(Y)=0	×	Increments contents of register Y by 1. Skips next instruction when new contents of register Y are "0".
	DEY	0 1110 1000	0E8	1	1	(Y)←(Y)−1	(Y)=15	×	Decrements contents of register Y by 1. Skips next instruction when new contents of register Y are "15".
	SADR j	0 1100 00jj	0Cj	1	1	j = 0 : specifies the 0th digit of F4 j = 1 : specifies the 0th digit of F5 j = 2 : specifies the 0th digit of F6 j = 3 : specifies the 0th digit of F7	—	×	During the following instruction cycle only, the 0th digit of the file soecified by the immediate field (in the range F4 to F7). The contents of the data pointer remain unchanged.
RAM-accumulator transfers	TAM j	0 1010 00jj	0Aj	1	1	(A)←(M(DP)), (X)←(X)∀j, where, j = 0 ~ 3	—	×	Transfers the RAM contents addressed by the active DP to register A. Register X is then "exclusive OR-ed" with the value j in the instruction, and the result stored in register X.
	XAM j	0 1010 11jj	0AC + j	1	1	(A)↔(M(DP)), (X)←(X)∀j, where, j = 0 ~ 3	—	×	Exchanges the contents of the RAM and register A. Contents of X are then "exclusive OR-ed" with the value j and the result stored in register X.
	XAMD j	0 1010 10jj	0A8 + j	1	1	(A)↔(M(DP)), (X)←(X)∀j, (Y)←(Y)−1 where, j = 0 ~ 3	(Y)=15	×	Exchanges the contents of the RAM and register A. Contents of X are then "exclusive OR-ed" with the value j in the instruction, and the result stored in register X. The contents of register Y are decremented by 1, and when the result is 15, the next instruction is skipped.
	XAMI j	0 1010 01jj	0A4 + j	1	1	(A)↔(M(DP)), (X)←(X)∀j, (Y)←(Y)+1 where, j = 0 ~ 3	(Y)=0	×	Exchanges the contents of the RAM and register A. Contents of X are then "exclusive OR-ed" with the value j in the instruction and the result stored in register x. The contents of register Y are incremented by 1, and when the result meets with the marked skip condition, the next instruction is skipped
Arithmetic operation	LA n	0 1000 nnnn	08n	1	1	(A)←n, where, n = 0 ~ 15	Written successively	×	Loads the value n into register A. When LA is written consecutively the first is executed, and successive ones are skipped.
	AM	0 1011 1110	0BE	1	1	(A)←(A)+(M(DP))	—	×	Adds the contents of the RAM to register A. The result is retained in register A, and the contents of flag CY are unaffected.
	AMC	0 1011 1100	0BC	1	1	(A)←(A)+(M(DP))+(CY), (CY)← Carry	—	0/1	Adds the RAM contents addressed by the active DP and contents of flag CY to register A. The result is stored in register A, and the carry in the active flag CY.
	AMCS	0 1011 1101	0BD	1	1	(A)←(A)+(M(DP))+(CY), (CY)← Carry	Carry = 1	0/1	Adds the contents of the RAM and flag CY to register A. The result is stored in register A and the carry in the CY, but the next instruction is skipped when a carry is produced.
	A n	0 1001 nnnn	09n	1	1	(A)←(A)+n, where, n = 0 ~ 15	A carry is not produced and = 0 n ≠ 6	×	Adds value n in the instruction to register A. The contents of flag CY are unaffected and the next instruction is skipped if a carry is not produced, except when n=6.
	SC	0 0000 0110	006	1	1	(CY)← 1	—	1	Sets active flag CY.
	RC	0 0000 0101	005	1	1	(CY)← 0	—	0	Resets active flag CY.
	SZC	0 0000 0010	002	1	1	(CY)= 0 ?	(CY)= 0	×	Skips next instruction when contents of the active flag CY are 0 .
	CMA	0 1011 0111	0B7	1	1	(A)←$(\overline{A})$	—	×	Stores complement of register A in register A.
Bit operation	SB j	0 0001 00jj	01j	1	1	$(M_j(DP))$← 1, where, j = 0 ~ 3	—	×	Sets the jth bit of the RAM addressed by the active DP (the bit designated by the value j in the instruction).
	RB j	0 0001 01jj	014 + j	1	1	$(M_j(DP))$← 0, where, j = 0 ~ 3	—	×	Resets the jth bit of the RAM addressed by the active DP (the bit designated by the value j in the instruction).
	SZB j	0 0001 10jj	018 + j	1	1	$(M_j(DP))$= 0 ?, where, j = 0 ~ 3	$(M_j(DP))$=0	×	Skips next instruction when the contents of the jth bit of the RAM addressed by the active DP (the bit which is designated by the value j in the instruction) are 0.
Compares	SEAM	0 1011 1111	0BF	1	1	(M(DP))=(A) ?	(M(DP))=(A)	×	Skips next instruction when contents of register A are equal to the RAM contents addressed by the active DP.
	SEY y	0 0010 yyyy	02y	1	1	(Y)=y ?, where, y = 0 ~ 15	(Y)=y	×	Skips next instruction when the contents of register Y are equal to the value y in the instruction.
Jumps	B xy	1 1xxx yyyy	18y + x	1	2	(PCL)←16x+y, where, (SM)= 0	—	×	Jumps to address xy of the current page.
						(PCH)← 1 (PCL)←16x+y, where, (SM)= 1			Jumps to address xy on page 1 when executed, provided that none of instructions RT, RTS, BL or BML, was executed after execution of instruction BM.
	BL pxy	0 0011 pppp 1 1xxx yyyy	03p 18y + x	2	3	(PCH)←p, (SM)← 0 (PCL)←16x+y	—	×	Jumps to address xy of page p.

Type of instruction	Mnemonic	Instruction code D_8 $D_7D_6D_5D_4$ $D_3D_2D_1D_0$	Instruction code 16mal notation	No. of words	No. of cycles	Functions	Skip conditions	Flag CY	Description of operation
Subroutine calls	BM xy	1 0xxx yyyy	1xy	1	2	$(SK_1)\leftarrow(SK_0)\leftarrow(PC)$, where, $(SM)=0$ $(PC_H)\leftarrow 0$, $(PC_L)\leftarrow 16x+y$, $(SM)\leftarrow 1$	—	×	Calls for the subroutine starting at address xy on page 0.
						$(PC_H)\leftarrow 0$ $(PC_L)\leftarrow 16x+y$, where, $(SM)=1$			Jumps to address xy of page 0 provided that none of instructions RT. RTS, BL or BML, was executed after the execution of instruction. BM.
	BML pxy	0 0011 pppp 1 0xxx yyyy	03p 1xy	2	3	$(SK_1)\leftarrow(SK_0)\leftarrow(PC)$ $(PC_H)\leftarrow p$, $(PC_L)\leftarrow 16x+y$, $(SM)\leftarrow 0$	—	×	Calls for the subroutine starting at address xy of page p.
Program returns	RT	0 0001 1111	01F	1	2	$(PC)\leftarrow(SK_0)\leftarrow(SK_1)$, $(SM)\leftarrow 0$	—	×	Returns to the main routine from the subroutine.
	RTS	0 0001 1110	01E	1	2	$(PC)\leftarrow(SK_0)\leftarrow(SK_1)$, $(SM)\leftarrow 0$	Unconditional skip	×	Returns to the main routine from the subroutine, and unconditionally skips the next instruction.
Interrupts	EI	0 0000 1011	00B	1	1	$(INTE)\leftarrow 1$	—	×	Sets interrupt flag INTE to enable interrupts.
	DI	0 0000 1010	00A	1	1	$(INTE)\leftarrow 0$	—	×	Resets interrupt flag INTE to disable interrupts.
	RTI	0 0001 1101	01D	1	2	$(PC)\leftarrow(SK_0)\leftarrow(SK_1)$, $(SM)\leftarrow(SM_0)$	—	×	Returns from interrupt routine to main routine. The internal subroutine mode flag is restored to the value held immediately before the interrupt.
Input/output	SD	0 0000 1111	00F	1	1	$(D(Y))\leftarrow 1$, where, $(Z)=1$	—	×	Sets the bit of port D, that is designated by register Y, when the contents of register Z are 1.
	RD	0 0000 1110	00E	1	1	$(D(Y))\leftarrow 0$, where, $(Z)=1$	—	×	Resets the bit of port D, that is designated by register Y, when the contents of register Z are 1.
	OSAB	0 1011 0011	0B3	1	1	$(S_7\sim S_4)\leftarrow(E_7\sim E_4)\leftarrow(B)$, $(P)\leftarrow(C)\leftarrow(CY)$, $(S_3\sim S_0)\leftarrow(E_3\sim E_0)\leftarrow(A)$	—	×	Output contents of registers A and B to port S., and the contents of the carry flag to port P.
	OSPA	0 1011 0010	0B2	1	1	$(S_7\sim S_0)\leftarrow(E_7\sim E_0)\leftarrow$8-segment decoder $\leftarrow(A)$, $(P)\leftarrow(C)\leftarrow(CY)$	—	×	Decodes contents of register A by 8 segment decoder and the result is output to port S. and output the contents of the carry flag to port P
	OSE	0 1111 0010	0F2	1	1	$(S_7\sim S_0)\leftarrow(E_7\sim E_0)$, $(P)\leftarrow(C)$	—	×	Outputs contents of registers E and C to ports S and P.
	SHFT	0 0000 1000	008	1	1	$(En)\leftarrow(En-1)$, $(E_0)\leftarrow(C)\leftarrow(CY)$	—	×	Links register E and register C and shifts left. The contents of register C are shifted into the least significant bit of register E and the contents of the flag CY are shifted into register C. The most significant bit of register E is lost.
	IAS	0 1111 1000	0F8	1	1	$(A)\leftarrow(S_3\sim S_0)$	—	×	Transfers the 4 lower order bits of port S to register A.
	IAK	0 1011 1000	0B8	1	1	$(A)\leftarrow(K_3\sim K_0)$	—	×	Transfers the 4 bits of port K to register A.
	SZTO	0 0000 0001	001	1	1	$T_0=0$?	$T_0=0$	×	Skips the next instruction if the sensing input T_0 is low.
	SZT1	0 0000 0100	004	1	1	$T_1=0$?	$T_1=0$	×	Skips the next instruction if the sensing input T_1 is low.
	CLDS	0 0000 0111	007	1	1	$(D)\leftarrow 0$, $(S)\leftarrow 0$, $(P)\leftarrow 0$	—	×	Clears ports D, S and P
Misc	NOP	0 0000 0000	000	1	1	$(PC)\leftarrow(PC)+1$	—	×	No operation.

Symbol	Contents	Symbol	Contents	Symbol	Contents
A	4-bit register (accumulator)	SK_1	11-bit stack register	( )	Indicates contents of the register, memory, etc.
B	4-bit register	CY	1-bit carry flag	∀	Exclusive OR
C	1-bit register	xx	2-bit binary variable	—	Negation
E	8-bit register	yyyy	4-bit binary variable	×	Indicates flag is unaffected by instruction execution
X	2-bit register	z	1-bit binary variable	xy	Label used to indicate the address xxxyyyy
Y	4-bit register	nnnn	4-bit binary constant	pxy	Label used to indicate the address xxxyyyy of page pppp.
Z	1-bit register	ii	2-bit binary constant	C	Hexadecimal number C + binary number x.
DP	7-bit data pointer, combination of registers, X Y and Z	D	16-bit port	+	
PC_H	The high-order four bits of the program counter.	K	4-bit port	x	
PC_L	The low-order seven bits of the program counter.	S	8-bit port	SM	1-bit subroutine mode flag
PC	11-bit program counter, combination of PC_H and PC_L	P	1-bit port	SM_0	1-bit subroutine mode flag save register
SK_0	11-bit stack register	←	Shows direction of data flow	INTE	Interrupt enable flag

LIST OF INSTRUCTION CODES

D_3 ~ D_0 \ D_8 ~ D_4		0 0000	0 0001	0 0010	0 0011	0 0100	0 0101	0 0110	0 0111	0 1000	0 1001	0 1010	0 1011	0 1100	0 1101	0 1110	0 1111	1 0000 ~ 1 0111	1 1000 ~ 1 1111
Hexadecimal notation		0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	0 A	0 B	0 C	0 D	0 E	0 F	10~17	18~1F
0000	0	NOP	SB 0	SEY 0	BL BML	LXY 0, 0	LXY 0, 4	LXY 0, 8	LXY 0, 12	LA 0	A 0	TAM 0	TEPA	SADR 0	—	—	—	BM	B
0001	1	SZTO	SB 1	SEY 1	BL BML	LXY 1, 0	LXY 1, 4	LXY 1, 8	LXY 1, 12	LA 1	A 1	TAM 1	TEAB	SADR 1	—	—	—	BM	B
0010	2	SZC	SB 2	SEY 2	BL BML	LXY 2, 0	LXY 2, 4	LXY 2, 8	LXY 2, 12	LA 2	A 2	TAM 2	OSPA	SADR 2	—	—	OSE	BM	B
0011	3	—	SB 3	SEY 3	BL BML	LXY 3, 0	LXY 3, 4	LXY 3, 8	LXY 3, 12	LA 3	A 3	TAM 3	OSAB	SADR 3	—	—	—	BM	B
0100	4	SZT1	RB 0	SEY 4	BL BML	LXY 0, 1	LXY 0, 5	LXY 0, 9	LXY 0, 13	LA 4	A 4	XAMI 0	—	—	—	INY	—	BM	B
0101	5	RC	RB 1	SEY 5	BL BML	LXY 1, 1	LXY 1, 5	LXY 1, 9	LXY 1, 13	LA 5	A 5	XAMI 1	TBA	—	—	—	TXA	BM	B
0110	6	SC	RB 2	SEY 6	BL BML	LXY 2, 1	LXY 2, 5	LXY 2, 9	LXY 2, 13	LA 6	A 6	XAMI 2	TYA	—	—	—	—	BM	B
0111	7	CLDS	RB 3	SEY 7	BL BML	LXY 3, 1	LXY 3, 5	LXY 3, 9	LXY 3, 13	LA 7	A 7	XAMI 3	CMA	—	—	—	—	BM	B
1000	8	SHFT	SZB 0	SEY 8	BL BML	LXY 0, 2	LXY 0, 6	LXY 0, 10	LXY 0, 14	LA 8	A 8	XAMD 0	IAK	—	—	DEY	IAS	BM	B
1001	9	—	SZB 1	SEY 9	BL BML	LXY 1, 2	LXY 1, 6	LXY 1, 10	LXY 1, 14	LA 9	A 9	XAMD 1	TAB	—	—	—	TAX	BM	B
1010	A	DI	SZB 2	SEY 10	BL BML	LXY 2, 2	LXY 2, 6	LXY 2, 10	LXY 2, 14	LA 10	A 10	XAMD 2	TAY	—	—	—	—	BM	B
1011	B	EI	SZB 3	SEY 11	BL BML	LXY 3, 2	LXY 3, 6	LXY 3, 10	LXY 3, 14	LA 11	A 11	XAMD 3	—	—	—	—	—	BM	B
1100	C	LZ 0	—	SEY 12	BL BML	LXY 0, 3	LXY 0, 7	LXY 0, 11	LXY 0, 15	LA 12	A 12	XAM 0	AMC	—	—	—	—	BM	B
1101	D	LZ 1	RTI	SEY 13	BL BML	LXY 1, 3	LXY 1, 7	LXY 1, 11	LXY 1, 15	LA 13	A 13	XAM 1	AMCS	—	—	—	—	BM	B
1110	E	RD	RTS	SEY 14	BL BML	LXY 2, 3	LXY 2, 7	LXY 2, 11	LXY 2, 15	LA 14	A 14	XAM 2	AM	—	—	—	—	BM	B
1111	F	SD	RT	SEY 15	BL BML	LXY 3, 3	LXY 3, 7	LXY 3, 11	LXY 3, 15	LA 15	A 15	XAM 3	SEAM	—	—	—	—	BM	B

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS} (output transistors cutoff)	0.3 ~ −18	V
V_I	Input voltage, Port S and V_P		0.3 ~ −35	V
V_I	Input voltage, other than port S (Note 1)		0.3 ~ −18	V
V_O	Output voltage, ports S, P, and D		0.3 ~ −35	V
V_O	Output voltage, other than ports S, P and D		0.3 ~ −18	V
Pd	Power dissipation	Ta = 25°C	1000	mW
Topr	Operating temperature		−10 ~ 70	°C
Tstg	Storage temperature		−40 ~ 125	°C

Note 1. $V_{I(\phi)}$ = 1.1 ~ −35V for use of ceramic resonater

RECOMMENDED OPERATING CONDITIONS (Ta = −10 ~ 70°C, unless otherwise noted)

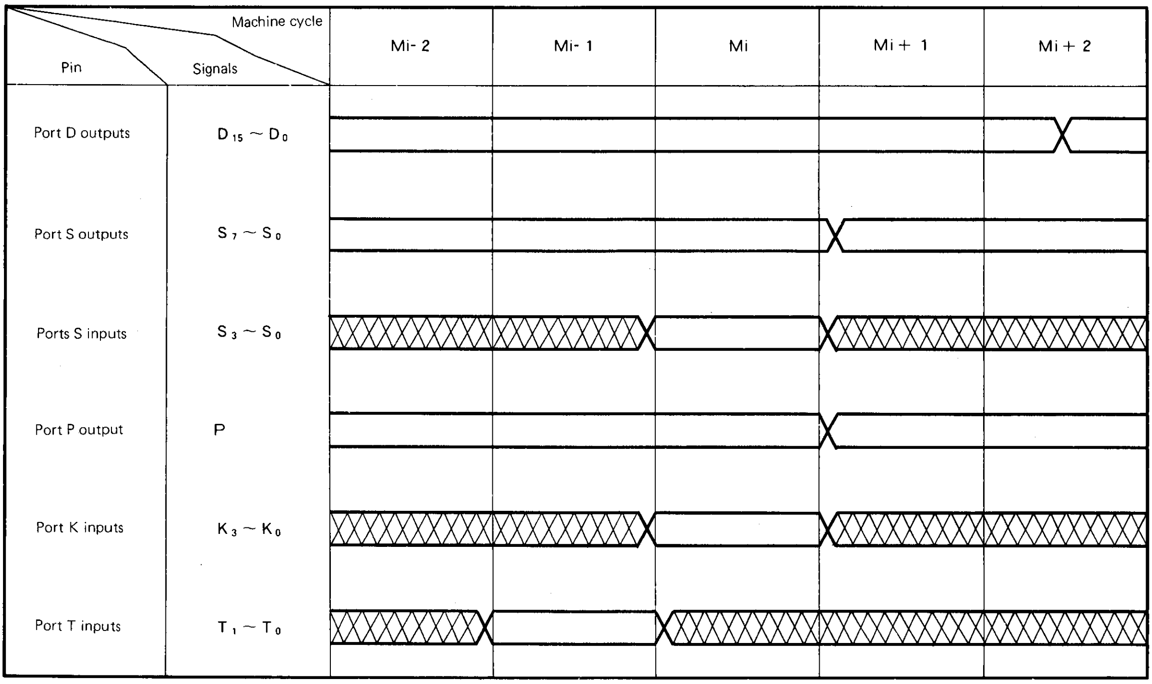
Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	−10.8	−12	−13.2	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage, ports T and K and RESET inputs	−1.5	−1.0	0	V
V_{IH}	High-level input voltage, port S, INT and V_Z inputs	−0.4		0	V
$V_{IH(\phi)}$	High-level clock input voltage	−0.9		0	V
V_{IL}	Low-level input voltage, ports T and K and RESET inputs	V_{DD}		−6	V
V_{IL}	Low-level input voltage, INT and V_Z inputs	V_{DD}		−4	V
V_{IL}	Low-level input voltage, port S input	−33		−4	V
$V_{IL(\phi)}$	Low-level clock input voltage for external clock	V_{DD}		$V_{DD}+2$	V
V_{OL}	Low-level output voltage, Ports S, P and D	−33		0	V
$f(\phi)$	Internal clock oscillation frequency	240	300	400	kHz

ELECTRICAL CHARACTERISTICS (V_{DD} = −12V±10%, V_{SS} = 0V, $f(\phi)$ = 300 kHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Nom	Max	
V_{OH}	High-level output voltage, port D	V_{DD} = −12V, I_{OH} = −15mA Ta = 25°C	−2.5			V
V_{OH}	High-level output voltage, ports S and P	V_{DD} = −12V, I_{OH} = −8mA Ta = 25°C	−2.5			V
I_{IH}	High-level input current, ports T and K inputs	V_{DD} = −12V, V_{IH} = 0V Ta = 25°C	80		490	μA
I_{IH}	High-level input current, RESET input	V_{DD} = −12V, V_{IH} = 0V, Ta = 25°C	20		120	μA
I_{IL}	Low-level input current, ports T and K, RESET and INT inputs	V_{DD} = −12V, V_{IL} = −12V, Ta = 25°C			−12	μA
I_{OH}	High-level output current, ports S, P and D	V_{DD} = −12V, V_P = −12V, V_{OH} = 0V, Ta = 25°C with output transistors cutoff	80		560	μA
I_{OL}	Low-level output current, ports S, P and D	V_{DD} = −12V, V_P = −12V, V_{OL} = −12V, Ta = 25°C with output transistors cutoff			−12	μA
$I_{I(\phi)}$	Clock input current	V_{DD} = −12V, $V_{I(\phi)}$ = −12V Ta = 25°C			−10	μA
I_{DD}	Supply current	V_{DD} = −12V, Ta = 25°C, with input and output pins open		−0.9	−3.4	mA

Note 1. Currents are taken as positive when flowing into the IC (zero-signal condition), with the minimum and maximum values as absolute values.
2. Total sum of high-level output current of port D must be under 75mA.

INPUT/OUTPUT INSTRUCTION TIMING



Note 1. The above timing relationships apply for the case of an instruction executed at the M_ith machine cycle.
2. ⊠⊠⊠ The crosshatched area indicates invalid input.

MELPS 41/42 MICROCOMPUTERS

5

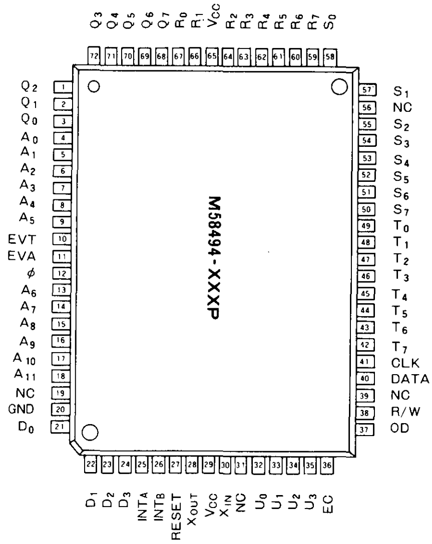
SINGLE-CHIP 4-BIT CMOS MICROCOMPUTER

DESCRIPTION

The M58494-XXXP is a single-chip 4-bit microcomputer fabricated using CMOS technology in a 72-pin plastic flat package. It has a 4096-word by 10-bit mask-programmable ROM and a 32-word by 4-bit RAM. RAM capacity can be expanded to as much as 4096 words by 4 bits by directly connecting generally available CMOS RAMs.

The device is designed for application where the low power dissipation of CMOS is essential.

PIN CONFIGURATION (TOP VIEW)



Package Outline 72P2

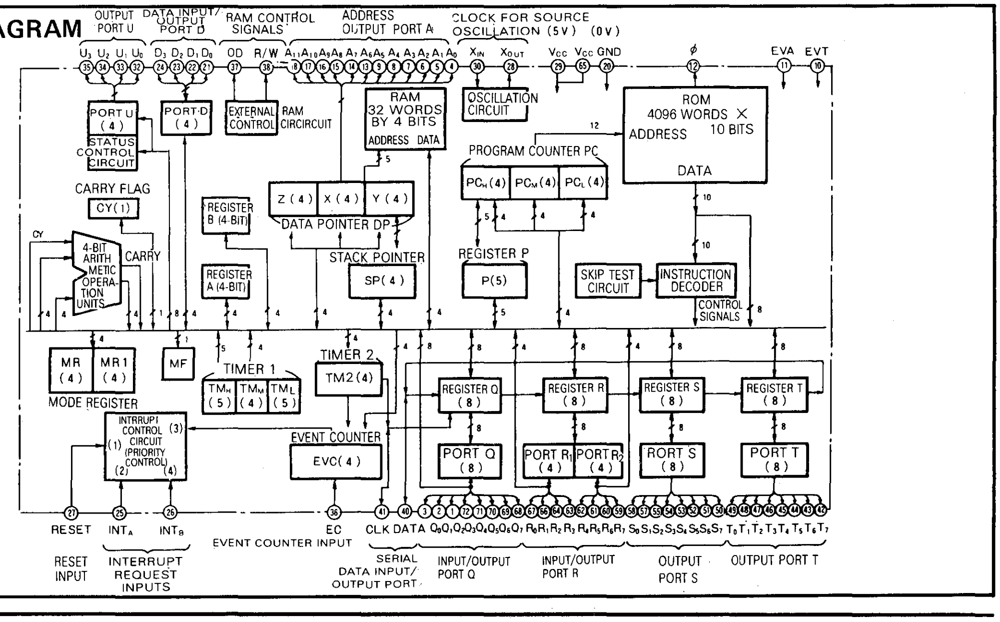
FEATURES

- Basic machine instructions 92
- Basic instruction execution time (at 455kHz clock frequency) 8.8μs
- Large memory capacity:
  - ROM .... 4096-word x 10-bit
  - Internal RAM .... 32-word x 4-bit
  - External RAM .... 4906-word x 4-bit (max)
- Single 5V power supply
- Saving of last data pointer 4-level
- Subroutine nesting 12-level
- Internal timer: Timer 1 .... 14-bit
  - Timer 2 .... 4-bit
- Internal event-counter 4-bit
- I/O port for external RAMs (all three-state)
  - Address (port A) 12-bit
  - Control signals (R/W, OD) 2-bit
  - Data I/O (port D) 4-bit
- General-purpose registers 32-bit
- I/O port (port Q) 8-bit
- I/O port (port R) .... 4-bit x 2
- I/O port (serial data port) 2-bit
- Output ports (port S, port T) .... 8-bit x 2
- Output port (port U, three-state output) 4-bit
- Event-counter input (port EC) 1-bit
- Interrupt function (priority interrupt type) 4-factor, 1-level

APPLICATIONS

- Electronic cash registers, electronic calculators (with printer and/or programmable)
- Office machines, intelligent terminals, data terminals
- Sewing machines, knitting machines, etc.

BLOCK DIAGRAM



Outline Specifications of M58494-XXXP

Item			Performance
Number of basic instructions			92
Execution time of basic instructions			8.8μs (at V_{CC} = 5V, f = 455kHz)
Clock frequency			100～455kHz
Memory capacity	ROM		4096 words × 10 bits
	RAM (built-in)		32 words × 4 bits
	RAM (external)		4095 words × 4 bits (max.)
Input/output port for external RAM	Address (port A)		12 bits x 1 (3 states)
	Control signal (port OD and R/W)		2 bits (3 states)
	Data bus (port D)		4 bits x 1 (3 states)
Input/output port	Q	Input	8 bits x 1
		Output	8 bits x 1
	R	Input	4 bits x 2
		Output	8 bits x 1
	S	Output	8 bits x 1
	T	Output	8 bits x 1
	DATA	Serial data	1 bit (input/output port)
	CLK	Synchronizing pulse	1 bit (input/output port)
	U	Output	4 bits x 1 (3-state)
	EC	Input	1 bit
Subroutine nesting			12 levels
Interrupt request			4 factors 1 level
Saving of data pointer			4 levels
Clock generation circuit			Built-in (oscillation reference element is outside)
Ports input/output characteristics	Absolute maximum rating voltage		V_{CC}
	Input/output characterstics		Interchangeable with CMOS logic series
Power supply voltage	V_{CC}		5V (nominal)
	V_{SS}		0 V
Element structure			CMOS
Package			72-pin plastic molded flat package
Power dissipation			5 mW (at V_{CC} = 5V, f = 455kHz)

PIN DESCRIPTIONS

Pin	Name	Input or output	At reset	Function
X_{IN}	Source oscillation clock input	Input	—	Incorporates the clock oscillation circuit, for setting of the oscillation frequency. The oscillation reference device such as a ceramic filter for IF is connected between X_{IN} and X_{OUT}. When an external clock is used, connect the clock oscillation source to the X_{IN} pin and leave the X_{OUT} pin open.
X_{OUT}	Source oscillation clock output	Output	—	
RESET	Reset signal	Input	—	Resets the program counter PC and mode registers, and performs the reset initiation of the related input ports and output ports. For input/output ports, refer to the column for "At reset" of this table.
INT_A	Interrupt request signal A	Input	Disable	Input signals for interrupt request. Request is accepted on the rising edge of the signal. Besides these external input signals, the interrupt requests. T from timer 2/event counter are also received in the relative order RESET > INT_A > INT_T > INT_B. Since the interrupt requests are held at each latch, there will be none undetected.
INT_B	Interrupt request signal B	Input	Disable	
EC	Event counter input	Input	—	The input signal for the event counter, which program $2^0 \sim 2^4$ events of the event mode. This value is set as an initial value and countdown starts from this value to reach F_{16}, which then generates interrupt request signal INT_T.
A_0~A_{11}	Address output port A	Output	Floating	The address signal for main memory (RAM) externally connected, in the form of a 3-state output. At MM mode where external memory is used the data of the data pointer DP is read out directly. In SM mode where internal memory (RAM) is used, the data of the data pointer Y immediately before switching to MM mode is transferred to the auxiliary latch (4 bits) prior to read-out. However, the lower 8 bits of the address signal (A_0~A_7) are not affected by this mode, since data pointers X and Z are not related to latch operation.
D_0~D_3	Data input/output port D	Input/ output	Floating	A 3-state input/output port to execute data transfer in 4-bit units to/from an externally connected main memory (RAM). Switching of input-output is made automatically by instruction.
OD	External RAM read signal	Output	Floating	The output port is 3-state and the read signal generated at the data input cycle is in the externally connected main memory(RAM). During a read cycle, it becomes automatically set to low-level.
R/W	External RAM write signal	Output	Floating	The output port is 3-state and the write signal generated at the data write cycle is in the externally connected main memory (RAM). During a write cycle, it is automatically set to low-level.
U_0~U_3	Output port U	Output	Floating	The output port enables 3-state setting per 1-bit unit. The 3-state condition is modified by the data content of register B, and the data of register A is output. The output setting of port U, however, is made either by instruction SU unconditionally or by the instruction TPRA or TPRN, which transfers the data of the general-purpose register to ports Q, R, S and T.
Q_0~Q_7	Input/output port Q	Input/ output	Input	The input/output port for 8-bit data transfer to/from register Q. Register Q enables data transfer between register A and register B. By instruction OPI, this port also functions to load the value (8-bit) of the immediate field of the ROM to register Q. Port Q data can be transferred to registers A and B as an input signal of 8 bits.
R_0~R_7	Input/output port R	Input/ output	Input	The input/output port for 8-bit data transfer to/from register R. Register R enables data transfer between register A and register B. By instruction OPI, this port also functions to load the value (8-bit) of the data field of the ROM to register R. When port R is used as the input signal of a 4-bit unit, the data, 4 bits each can be transferred to register B.
S_0~S_7	Output port S	Output	Low-level	The output port that enables 8-bit data transfer to/from register S. Register S enables data transfer between register A and register B. By instruction OPI, this port also functions to load the value (8-bit) of the data field of the ROM to register S.
T_0~T_7	Output port T	Output	Low-level	The output port for 8-bit data transfer to register S. Register T enables data transfer between register A and register B. By instruction OPI, this also functions to load the value (8-bit) of the immediate field of the ROMto register T.
DATA	Serial data port	Input/ output	Floating	The input/output port normally is floating to handle the serial data of the 32-bit general-purpose register. At output mode data of the least significant bit of the general purpose register (the least significant bit of register T) is read out, and at the input mode, the input is to the most significant bit of the general-purpose register (the most significant bit of register O).
CLK	Serial data shift clock signal	Input/ output	Floating	The input/output port is normally floating to generate a shift clock pulse synchronized with the above serial data port. At output mode a shift clock pulse synchronized with the data transmission is generated and at the input mode, a shift pulse synchronized with the rate of data receiving is applied.

BASIC FUNCTIONAL BLOCKS AND THEIR OPERATIONS

Program Memory ROM

The ROM stores 32-pages by 128 words of program and its addressing is performed by a program counter. The program counter consists of a 7-bit binary sequential counter and a 5-bit page register.

Program Counter PC

The ROM is composed of 32 pages of 128 words, and when program execution completes instruction at address 127, the binary counter is set to 0 and the next page is automatically incremented in the page-designation register.

The 12-bit contents of the program counter PC can be saved for up to 12 levels in the fixed stack area of the external main memory (RAM). In the execution of instructions BM and BMA, control can be returned to a former routine by storing the contents of the program counter before branching, in the execution of instructions RT, RTS, and RTI.

Register P

In the page register, the contents of register P are loaded by instructions BL, BA, BM and BMA. Instruction BMAB branches unconditionally to the address derived by using the contents of register A for the low-order 4-bits of the 12-bit PC, those of register B for the middle 4-bits, and those of the upper 4-bits of the 5-bit register P for the upper 4-bits, and then executes the instruction OPI of the branch, and simultaneously returns automatically.

Stack Pointer SP

A stack of 12 levels is provided for saving of the program counter PC in the fixed address area within the external main memory (RAM), and the contents of the stack pointer are used during addressing. The contents of the stack pointer are incremented by an interruption or in the execution of instructions BM and BMA, and are decremented in the execution of instructions RT, RTS and RTI.

Data Memory RAM

The internal RAM is used to store data in the form of two files each consisting of 16 words by 4 bits. The external RAM can be expanded up to 4096 words by 4 bits. These addresses are designated by a 12-bit data pointer. The contents of the data pointer can be saved for up to 4 levels in the stack region (fixed region in the external RAMs) by execution of a special instruction. The external RAM can be easily expanded without any extra interface circuits by connecting a 12-bit address signal, the 2-bit RAM control signal and the 4-bit data input/output signal. These signals can address external RAMs for up to 4096 x 4-bit words, thus incrementing the basic external minimum RAM organization of 256 x 4-bit words.

Data Pointer DP

This is a register of 12 bits addressing memory, being composed of registers X, Y, and Z, having 4 bits each. Register X address 16 files, each of which comprises 16 words. Register Y address data of 16 files (a file comprises 16 words). Register Z permits address specification such that data memory may be extended up to maximum of 16 sets of 4096 words by 4 bits, where one unit comprises 16 files (256 words by 4 bits).

Since the address of the external main memory (4096 words by 4 bits maximum) and the internal scratch-pad memory (32 words by 4 bits) are designated identically, the external main memory is selected by instruction MM, and the internal scratch-pad memory by instruction SM.

The contents of DP can be saved for up to 4 levels in the fixed stack region of the external main memory. This pointer is saved during the execution of instruction SDP, and is restored by instruction LDP.

When the data pointer stack is not used, the entire stack may be used as a program counter stack.

Register X → / Register Y ↓ (Address designation)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0														PC_L	PC_M	PC_H	PC stack (12 levels)
1																	
2																	
3																	
4																	
5																	
6																	
7																	
8																	
9																	
A																	
B																	
C														Y	X	Z	DP stack (4 levels)
D																	
E																	
F																	

(File designation: columns 0–F)

Fig. 1 External basic main memory (Z = 0) and RAM map

Table 1 Address designation of data pointer stack

Value of data field during execution of instructions SDP and LDP		Stack DP (file designated by register Y)
I_1	I_0	
0	0	C
0	1	D
1	0	E
1	1	F

Accumulator (Register A), Carry Flag CY

Register A is an accumulator forming the central unit of a 4-bit-wide microcomputer. Data processing operations such as arithmetic, data transfer, data exchange, data conversion, input/output, etc. are executed principally with this register.

The carry flag CY stores the carry or borrow from the most significant bit of the arithmetic unit in the execution of specific arithmetic instructions, and is available for multipurpose uses as a one-bit flag.

Auxiliary Register (Register B)

Register B is composed of four bits. It is employed for bit operating functions, temporary memory of four-bit data and transfer of eight-bit data when coupled with register A, etc.

Four-Bit Arithmetic Logic Unit (ALU)

This unit carries out four-bit arithmetic and logical functions, and is composed of a four-bit adder and a logic circuit associated with it. It carries out addition, complement conversion, logic arithmetic comparison, arithmetic comparison, bit processing, etc.

General-Purpose Registers Q, R, S, and T

These general-purpose registers comprise a set of four 8-bit shift registers. When using combinations of functions such as serial input, serial output, parallel input and parallel output, by properly selected instructions, they are employed for data transfer between register A and register B, data transfer between output ports or input/output ports, data storage of the data field of the ROM value (8 bits), transmission of internal serial data, receiving of external serial data, etc.

Table 2 Relationship between input/output address N and general-purpose registers

Input/output address	Immediate data N in execution of the instructions BMAB TNAB TPRN and TRPN		General-purpose register to be selected
N	I_1	I_0	
0	0	0	Register Q
1	0	1	Register R
2	1	0	Register S
3	1	1	Register T

Instruction OPI loads one of the four general-purpose registers selected by the input/output address N with the value (8 bits) of the data field. The input/output address N is latched with the contents of the lower 2 bits of the data field in the execution of the instructions BMAB, TNAB, TABN, TPRN and TRPN and determines the register which loads data in the execution of the instruction OPI.

When the general-purpose registers are used as a single 32-bit shift register, four kinds of modes as shown in Table 3 can be set by instruction SMR1.

Mode Register

The mode register is composed of 8 bits, and can select operation modes and functions, etc. of the associated input port or output port by setting or resetting the mode flag corresponding to a bit in register A.

The mode setting by the instruction SMR is shown in Table 4.

The mode setting by instruction SMR1 is shown in Table 5.

Interrupt Function

This microcomputer has a hardware interrupt function for four conditions by one-level. The interrupt requests comprise: the RESET signal; the interrupt request signals INT_A and INT_B as external signals; and the interrupt request signal INT_T by the internal event counter.

The fixed addresses to be jumped to and the priority order of four factors in the interrupt request are defined as follows:

(1) In case of by reset signal RESET page 0, address 0
(2) In case of interrupt signal INT_A page 0, address 2
(3) In case of interrupt signal INT_T page 0, address 8
(4) In case of interrupt signal INT_B page 0, address 4

A RESET signal restores the hardware to the initial state, independent of any current instruction.

In an interrupt enable state, the interrupt is accepted at the rising edge of interrupt request signals INT_A and INT_B. When an interruption is requested in an interrupt disable state, the interrupt is not executed. If the interrupt disable state is removed thereafter and a corresponding interrupt enable instruction is executed, the interrupt routine will be

Table 3 Mode setting by instruction SMR 1; when the general-purpose registers are employed as a 32-bit shift register

Mode flag	SDM	0	0	1	1
	RVM	0	1	0	1
DATA pin		Input	Output	Output	Output
CLK pin		Floating	Input (rising edge trigger)	Output (gernated by timer 2)	Output (generated by shift instruction)
Shift data input	SST, RST	Immediate field data	O input independent of executable instruction	Immediate field data	Immediate field data
	IST	DATA pin output		DATA pin output	DATA pin output
Shift clock pulse		Instructions SST, RST IST	CLK input	Instructions SST, RST, IST	Instructions SST, RST, IST
Transmission receiving		Receiving (only in instruction IST)	Transmission	Transmission	Transmission

executed immediately because the interrupt request has been held in the latch. The current interrupt request, held in a latch during the interrupt disable state, is reset by the interrupt disable instruction.

When two and more interrupt requests of four factors occur simultaneously, the interrupt processing is by order of the highest priority routine. The interrupt request of lower priority order is held in the corresponding latch in an interrupt disable state. When the interrupt disable state is removed by the interrupt enable instruction (after completion of the interrupt process of upper priority order), the interrupt request of next lower priority is initiated.

Table 4 SMR mode setting

Bits of register A	Mode flag (contents of register A are stored)	Status	Function	Mode flag at reset
A_0	IMQ	0	Port Q is used as an 8-bit input port	0
		1	Port Q is used as an 8-bit output port	
A_1	LCD	0	For output port U, only instruction cat set port U.	0
		1	For output port U, instructions TPRN and TPRA for port Q, R, S and T can also set port U.	
A_2	IMR1	0	Port R_1 is used as a 4-bit input port	0
		1	Port R_1 is used as a 4-bit output port	
A_3	IMR2	0	Port R_2 is used as a 4-bit input port	0
		1	Port R_2 is used as a 4-bit output port	

Table 5 SMR 1 mode setting

Bits of register A	Mode flag (contents of register A are stored)	Status	Function	Mode flag at reset
A_0	TMM	0	Event mode, event counter is used with EC input.	0
		1	Timer mode, event counter is used in combination with timer 2.	
A_1	BF	0	All signals (A_{11}~A_0, D_3~D_0, OD and R/W) for external main memory. (RAM) are put in floating.	0
		1	All signals (A_{11}~A_0, D_3~D_0, OD and R/W) for external main memory (RAM) are activated.	
A_2	RVM	0	When the general-purpose registers are used as a 32-bit shift register, functions of transmission/receiving, terminals DATA and CLK are employed properly by RVM, SDM flags. For further details, refer to explanation of the general-purpose register.	0
		1		
A_3	SDM	0		0
		1		

Timers and Event Counter

This block is composed of a 14-bit timer 1, a 4-bit timer 2 and a 4-bit event counter.

Timer 1 is a standard timer that continuously counts the frequency X_{IN}, divided by fourteen. The timer performs accurate counting and the period is given by the following formula:

$$(\text{Fundamental output frequency } X_{IN}) \times 2^5 (TM_L) \times 2^4 (TM_M) \times 2^5 (TM_H) = \text{cycle time of timer 1}$$

By the continuous use of instructions TATM and TBTM, the contents of TM_M are stored in register B, the contents of the lower 4 bits of TM_H in register A, and the high-order bit of TM_H in carry flag CY, respectively. The contents of timer 1 can be accessed. Instruction

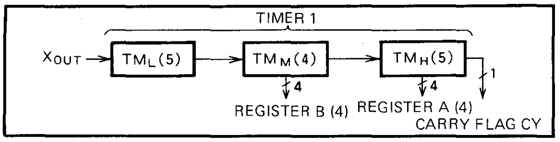
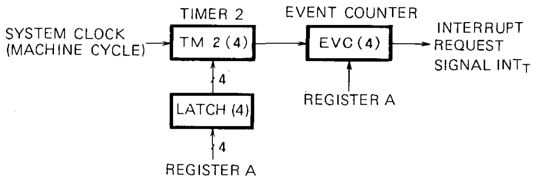


Fig. 2 Outline of timer 1 configuration

RTM clears the contents of timer 1 and resets it to 0.

Timer 2 is composed of a 4-bit counter and a 4-bit latch. The contents of register A are stored as the starting value in the latch and the counter by an STM instruction, whereupon counting down starts in synchronization with each machine cycle. When the contents of the counter become F during countdown, the pre-programmed starting value is restored in the counter from the latch.

(1) Timer mode: When TMM = 1 is set by the instruction SMR1



(2) Event mode: When TMM = 0 is set by the instruction SMR1

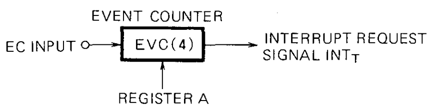


Fig. 3 Outline configuration of timer 2 and event counter

The cycle period of timer 2 is given by the following formula:

Machine cycle x $[1 + (2^0 \sim 2^4)]$

Where the timer mode is set by SMR1 instruction, timer 2 is connected to the event counter. Every time the contents of timer 2 become F the event counter counts down once. For the event counter, the contents of register A can be stored in the counter and used as a starting value by using instruction SEC.

When the event mode is set using instruction SMR1, the event counter is counted down by sensing the rising edge of external event counter input EC.

In both timer mode and event mode, the event counter is counted down from a starting value, and an interrupt request signal is generated when the contents become F.

The time necessary for INT_T generation from the starting value is given by the following formulas:

Timer mode

Machine cycle x $[1 + (2^0 \sim 2^4)] \times (2^0 \sim 2^4)$

Event mode

EC input period x $(2^0 \sim 2^4)$

Reset Function

Applying a low-level input to the RESET input pin for 3 machine cycles or more causes the reset state. Power-on reset is provided by such circuit as shown in Fig. 4.

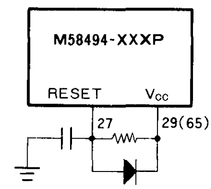


Fig. 4 Power-on reset circuit

Clock Generation Circuit

Clock pulses are easily generated by connecting an external IF ceramic filter between the pins X_{IN} and X_{OUT}. An example of such as circuit is shown in Fig. 5. If the clock signal is to be supplied from an external source, the clock source should be connected to pin X_{IN}, leaving the X_{OUT} pin open. An example of such circuit is shown in Fig. 6.

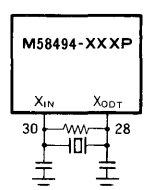


Fig. 5 External oscillation element connections

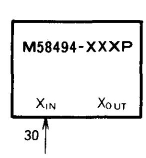


Fig. 6 Exteranl clock input circuit (Note 1)

Note 1. Low and high input levels should be set such that input level = 0~0.8V
Output level = $V_{CC} \sim (V_{CC} - 0.8)_V$
and such that the duty cycle is 40 to 60% with respect to the X_{IN} input.

5

MACHINE INSTRUCTIONS

Item / Classification	Symbol	Code		No. of words	No. of cycle	Function	Skip conditions	Flag CY
		I_9I_8 $I_7I_6I_5I_4$ $I_3I_2I_1I_0$	16 mal notation					
RAM address	MM	00 1000 0010	082	1	1	(MF)←1, Selects external main memory	—	—
	SM	00 1000 0000	080	1	1	(MF)←0, Selects internal scratch-pad memory	—	—
	LY y	01 1000 yyyy	18y	1	1	(Y)←y, where y = 0 ~ 15	Consecutively described	—
	LX x	01 1011 xxxx	1Bx	1	1	(X)←x, where x = 0 ~ 15	Consecutively described	—
	LZ z	01 1010 zzzz	1Az	1	1	(Z)←z, where z = 0 ~ 15	Consecutively described	—
	INY	00 0111 1100	07C	1	1	(Y)←(Y)+1	(Y)=0	—
	DEY	00 0111 1000	078	1	1	(Y)←(Y)−1	(Y)=15	—
	TAY	00 0010 0000	020	1	1	(A)←(Y)	—	—
	TAX	00 0010 0010	022	1	1	(A)←(X)	—	—
	TAZ	00 0010 0011	023	1	1	(A)←(Z)	—	—
	TYA	00 0100 0000	040	1	1	(Y)←(A)	—	—
	TXA	00 0100 0010	042	1	1	(X)←(A)	—	—
	TZA	00 0100 0011	043	1	1	(Z)←(A)	—	—
	SDP j	00 0111 01jj	074 + j	1	3	(Mj)←(DP), where j = 0 ~ 3	—	—
	LDP j	00 1111 01jj	0F4 + j	1	3	(DP)←(Mj), where j = 0 ~ 3	—	—
Register-to-register transfer	TSM	00 1011 1100	0BC	1	1	(SM(DP)) ← (MM(DP))	—	—
	TSMI	00 1111 1100	0FC	1	1	(SM(DP)) ← (MM(DP)), (Y)←(Y)+1	(Y)=0	—
	TMS	00 1011 1110	0BE	1	1	(MM(DP)) ← (SM(DP))	—	—
	TMSI	00 1111 1110	0FE	1	1	(MM(DP)) ← (SM(DP)), (Y)←(Y)+1	(Y)=0	—
	TAB	00 1010 0000	0A0	1	1	(A)←(B)	—	—
	TBA	00 1100 0000	0C0	1	1	(B)←(A)	—	—
	TASP	00 1010 0010	0A2	1	1	(A)←(SP)	—	—
	TSPA	00 1100 0010	0C2	1	1	(SP)←(A)	—	—
	TACM	00 1000 0100	084	1	1	(A)←(N, MF, CY), where $A_{3\sim2}$=N, A_1=MF, A_0=CY	—	—
	TCMA	00 1100 1100	0CC	1	1	(N, MF, CY)←(A), where $A_{3\sim2}$=N, A_1=MF, A_0=CY	—	—
Transfer between RAM and accumulator	TAM j	00 0010 01jj	024 + j	1	1	(A)←(M(DP)), (X)←(X)∀j, where, j = 0 ~ 3	—	—
	XAM j	00 0110 01jj	064 + j	1	1	(A)⟷(M(DP)) (X)←(X)∀j, where, j = 0 ~ 3	—	—
	XAMD j	00 0110 10jj	068 + j	1	1	(A)⟷(M(DP)), (Y)←(Y)−1 (X)←(X)∀j, where j = 0 ~ 3	(Y)=15	—
	XAMI j	00 0110 11jj	06C + j	1	1	(A)⟷(M(DP)), (Y)←(Y)+1 (X)←(X)∀j, where j = 0 ~ 3	(Y)=0	—
	XAMD1 j	00 1110 10jj	0E8 + j	1	1	(A)⟷(M(DP)), (Y)←(Y)−1 (X)←(X)∀j, where j = 0 ~ 3	(Y)=3, 7, 11, 15	—
	XAMI1 j	00 1110 11jj	0EC + j	1	1	(A)⟷(M(DP)), (Y)←(Y)+1 (X)←(X)∀j, where j = 0 ~ 3	(Y)=4, 8, 12, 0	—
	TMA	00 0100 0100	044	1	1	(M(DP))←(A)		
Arithmetic	LA n	01 1001 nnnn	19n	1	1	(A)←n, where, n = 0 ~ 15	Consecutively described	—
	AM	00 0110 0000	060	1	1	(A)←(A)+(M(DP))	—	—
	AMC	00 0110 0010	062	1	1	(A)←(A)+(M(DP))+(CY), (CY)← Carry	—	0/1
	AMCS	00 0110 0011	063	1	1	(A)←(A)+(M(DP))+(CY), (CY)← Carry	Carry = 1	0/1
	A n	00 0101 nnnn	05n	1	1	(A)←(A)+n, where, n = 0 ~ 15	Carry = 0	—
	SC	00 1000 1010	08A	1	1	(CY)←1	—	1
	RC	00 1000 1000	088	1	1	(CY)←0	—	0
	SZC	00 1011 1000	0B8	1	1		(CY)=0	—
	CMA	00 1011 1010	0BA	1	1	(A)←($\overline{A}$)	—	—

Classification \ Item	Symbol	Code I_9I_8 $I_7I_6I_5I_4$ $I_3I_2I_1I_0$	Code 16 mal notation	No. of words	No. of cycle	Function	Skip condition	Flag CY
Bit manipulation	SB j	00 1000 11jj	08C + j	1	1	(B(j))←1, where, j = 0 ~ 3	—	—
	RB j	00 1010 11jj	0AC + j	1	1	(B(j))←0, where, j = 0 ~ 3	—	—
	SZB j	00 0011 10jj	038 + j	1	1		(B(j)) = 0 where, j = 0 ~ 3	—
	SZM j	00 0000 01jj	004 + j	1	1		(Mj(DP)) = 0 where, j = 0 ~ 3	—
Compare	SEAM	00 1110 0000	0E0	1	1		(A) = (M(DP))	—
	SEY n	00 0001 nnnn	01n	1	1		(Y) = n where, n = 0 ~ 15	—
	SEI n	00 1001 nnnn	09n	1	1		(A) = n where, n = 0 ~ 15	—
Branch	B xy	01 0xxx yyyy	1xy	1	1	(PC_L)←y, (PC_M)←x where 16x + y = 0 ~ 127	—	—
	BL xy	11 0xxx yyyy	3xy	1	1	(PC_L)←y, (PC_M)←(P_0, x) (PC_H)←(P_4, P_3, P_2, P_1) where 16x + y = 0 ~ 127	—	—
	BA i	00 1101 0iii	0Di	1	1	(PC_L)←(A_0, i) where, i = 0 ~ 7 (PC_M)←(P_0, A_3, A_2, A_1) (PC_H)←(P_4, P_3, P_2, P_1)	—	—
	BMAB r	00 1100 10rr	0C8 + r	1	1	(PC_L)←(A) (PC_M)←(B) (PC_H)←(P_4, P_3, P_2, P_1) but returns unconditionally after one machine cycle. Input/output address r = 0 ~ 3 designates general-purpose register	—	—
	LP p	01 110p pppp	1CP + p	1	1	(P)←p where, p = 0 ~ 31	Consecutively described	—
	TPAC	00 1100 0100	0C4	1	1	(P)←(CY, A)	—	—
	TACP	00 1010 0100	0A4	1	1	(CY, A)←(P)	—	—
Subroutine call	BM xy	11 1xxx yyyy	38y + x	1	3	(PC_L)←y (PC_M)←(P_0, x), where 16x + y = 0 ~ 127 (PC_H)←(P_4, P_3, P_2, P_1) (M(SP))←(PC) (SP)←(SP) + 1	—	—
	BMA i	00 1101 1iii	0D8 + j	1	3	(PC_L)←(A_0, i), where, j = 0 ~ 7 (PC_M)←(P_0, A_3, A_2, A_1) (PC_H)←(P_4, P_3, P_2, P_1) (M(SP))←(PC) (SP)←(SP) + 1	—	—
Return	RT	00 1111 1000	0F8	1	3	(PC)←(M(SP)) (SP)←(SP) − 1	—	—
	RTS	00 1111 1010	0FA	1	4	(PC)←(M(SP)) (SP)←(SP) − 1 (PC)←(PC) + 1	Unconditionally	—
	RTI	00 1111 1001	0F9	1	3	(PC)←(M(SP)) (SP)←(SP) − 1	—	—

Classification	Symbol	Code $I_9I_8\ I_7I_6I_5I_4\ I_3I_2I_1I_0$	Code 16 mal notation	No. of works	No. of cycle	Function	Skip conditions	Flag CY
Interrupt flip-flop control	EIA	00 0000 1001	009	1	1	Enables interruption of INT_A signal.	—	—
	EIB	00 0000 1010	00A	1	1	Enables interruption of INT_B signal.	—	—
	EIAB	00 0000 1011	00B	1	1	Enables interruption of INT_A and INT_B signals.	—	—
	EIT	00 0000 1000	008	1	1	Enables interruption of INT_T signal.	—	—
	DIA	00 0000 1101	00D	1	1	Disables interruption of INT_A signal.	—	—
	DIB	00 0000 1110	00E	1	1	Disables interruption of INT_B signal.	—	—
	DIAB	00 0000 1111	00F	1	1	Disables interruption of INT_A and INT_B signals.	—	—
	DIT	00 0000 1100	00C	1	1	Disables interruption of INT_T signal.	—	—
Timer	TBTM	00 0010 1111	02F	1	1	$(B) \leftarrow (TM_M)$	—	—
	TATM	00 1010 0111	0A7	1	1	$(A) \leftarrow (TM_{H3}, TM_{H2}, TM_{H1}, TM_{H0})$ $(CY) \leftarrow (TM_{H4})$		
	RTM	00 1011 0100	0B4	1	1	$(TM_L) \leftarrow 0,\ (TM_M) \leftarrow 0,\ (TM_H) \leftarrow 0$	—	—
	STM	00 1100 0111	0C7	1	1	$(TM2) \leftarrow (A)$	—	—
	SEC	00 1100 0110	0C6	1	1	$(EVC) \leftarrow (A)$	—	—
Input/output	ID	00 0010 1110	02E	1	1	$(B) \leftarrow (D),\ (OD) \leftarrow$ "L"	—	—
	OD	00 0100 1100	04C	1	1	$(D) \leftarrow (B),\ (R/W) \leftarrow$ "L"	—	—
	OPI s	10 ssss ssss	2ss	1	1	$(R(r)) \leftarrow s$	—	—
	TNAB r	00 0100 10rr	048 + r	1	1	$(R(r)) \leftarrow (A, B)$ where the general-purpose register is designated with $r = 0 \sim 3$	—	—
	TABN r	00 0010 10rr	028 + r	1	1	$(A, B) \leftarrow (R(r))$ where the general-purpose register is designated with $r = 0 \sim 3$	—	—
	IQ	00 1010 1000	0A8	1	1	$(A, B) \leftarrow (P(Q))$	—	—
	IR1	00 0010 1100	02C	1	1	$(B) \leftarrow (P(R_1))$	—	—
	IR2	00 0010 1101	02D	1	1	$(B) \leftarrow (P(R_2))$	—	—
Input/output control	SMR	00 0011 0100	034	1	1	$(MR) \leftarrow (A)$	—	—
	SMR1	00 0011 0110	036	1	1	$(MR1) \leftarrow (A)$	—	—
	SST	00 0011 1100	03C	1	1	$(R(Q_0)) \leftarrow 1,\ R(All) \leftarrow$ 1-bit shift R (All)	—	—
	RST	00 0011 1101	03D	1	1	$(R(Q_0)) \leftarrow 0,\ R(All) \leftarrow$ 1-bit shift R (All)	—	—
	IST	00 0011 1110	03E	1	1	$(R(Q_0)) \leftarrow (DATA),\ R(All) \leftarrow$ 1-bit shift R (All)	—	—
	SU	00 0100 1110	04E	1	1	$(U) \leftarrow (A, B)$	—	—
	CLP	00 0000 0001	001	1	1	$(P(All)) \leftarrow 0$	—	—
	TPRA	00 1011 0000	0B0	1	1	$(P(All)) \leftarrow (R(All))$	—	—
	TPRN r	00 1111 00rr	0Fr	1	1	$(P(r)) \leftarrow (R(r))$	—	—
	TRPN r	00 0111 00rr	07r	1	1	$(R(r)) \leftarrow (P(r))$	—	—
Others	NOP	00 0000 0000	000	1	1	No operation	—	—

Symbol	Details	Symbol	Details
A	4-bit register (accumulator)	$P(R_2)$	4-bit port R_2
A_i	Indicates the bits of register A. Where i = 0 ~ 3	P(Q)	8-bit port Q
B	4-bit auxiliary register	R(All)	Indicates all the 8-bit registers. Q, R, S, T (32-bit)
B(j)	The bit of register B addressed when j = 0 ~ 3		
CY	1-bit carry flag	R(r)	The register selected by r (r corresponds with registers Q, R, S, and T where r = 0 ~ 3)
D	4-bit input/output port (3-state)		
DATA	1-bit input/output port for serial data	$R(Q_0)$	1st bit of register Q
DP	12-bit data pointer composed of registers X, Y and Z	R/W	1-bit output port which is used for the write signal of the external main memory
EVC	4-bit event counter		
M(DP)	4-bit data memory addressed by the data pointer DP	SM(DP)	The 4-bit internal scratch-pad memory addressed by the data pointer DP
Mj	12-bit data from the scratch-pad memory addressed by i = 0 ~ 3 (data pointer number in the fixed area)	SP	4-bit stack pointer
		TMI	14-bit counter composed of TM_L, TM_M and TM_H counters
Mj (DP)	4-bit data from external memory addressed by the contents data pointer DP, where j = 0 ~ 3	TML	5-bit counter
		TM_M	4-bit counter
MF	1-bit flat for selection of internal scratch-pad memory (MF ← Q at instruction SM) or external main memory (MF ← 1 at instruction MM)	TM_H	5-bit counter
		TM_{H_i}	Indicates the bit of TM_H counter, where i = 0 ~ 4
		TM2	4-bit counter
MM(DP)	4-bit external main memory data addressed by the data pcinter DP	U	4-bit output port (3-state)
		X	4-bit register where X = 0 ~ 15, addressing the field of 16 words by 4 bits per file.
M(SP)	12-bit data from external memory addressed by the stack pointer SP (return address stored in the fixed area)		
MR	4-bit mode flag (IMQ, LCD, IMR1, IMR2)	Y	4-bit register where Y = 0 ~ 15, which addresses the word unit of 16 words by 4 bits.
MR1	4-bit mode flag (TMM, BF, RVM, SDM)		
r	Input/output address to select one of the general-purpose registers Q, R, S and T (r = 0 ~ 3)	Z	4-bit register where Z = 0 ~ 15, which addresses 16 files x 16 words x 4 bits
OD	1-bit output port used for the read signal for external main memory		
		iii	3-bit binary variable
P	5-bit page register	jj	2-bit binary constant
P_i	Indicates the bits of register P, where i = 0 ~ 4	nnnn	4-bit binary constant
PC	12-bit program counter composed of counters PC_L, PC_M and PC_H	ppppp	5-bit binary constant
		rr	2-bit binary constant
PC_L	4-bit counter	ssss ssss	8-bit binary constant
PC_M	4-bit counter	xxxx	4-bit binary variable
PC_H	4-bit counter	yyyy	4-bit binary variable
P(All)	Indicates all the 8-bit ports. Q, R, S, T (32-bit)	zzzz	4-bit binary variable
P(r)	The port selected by r (corresponds with ports Q, R, S, and T at r = 0 ~ 3)		
$P(R_1)$	4-bit port R_1		

5

INSTRUCTION CODE LIST

$I_9 \sim I_4$ / 16 mal notation / $I_3 \sim I_0$		00 0000	00 0001	00 0010	00 0011	00 0100	00 0101	00 0110	00 0111	00 1000	00 1001	00 1010	00 1011	00 1100	00 1101	00 1110	00 1111	01 0000 ~ 01 0111	01 1000	01 1001	01 1010	01 1011	01 1100 ~ 01 1101	01 1110 ~ 01 1111	10 0000 ~ 10 1111	11 0000 ~ 11 0111	11 1000 ~ 11 1111
		00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10~17	18	19	1A	1B	1C~1D	1E~1F	20~2F	30~37	38~3F
0000	0	NOP	SEY 0	TAY	—	TYA	A 0	AM	TRPN 0	SM	SEI 0	TAB	TPRA	TBA	BA 0	SEAM	TPRN 0	B	LY 0	LA 0	LZ 0	LX 0	LP	—	OPI	BL	BM
0001	1	CLP	SEY 1	*	—	*	A 1	*	TRPN 1	*	SEI 1	*	*	*	BA 1	*	TPRN 1	B	LY 1	LA 1	LZ 1	LX 1	LP	—	OPI	BL	BM
0010	2	—	SEY 2	TAX	—	TXA	A 2	AMC	TRPN 2	MM	SEI 2	TASP	*	TSPA	BA 2	*	TPRN 2	B	LY 2	LA 2	LZ 2	LX 2	LP	—	OPI	BL	BM
0011	3	—	SEY 3	TAZ	—	TZA	A 3	AMCS	TRPN 3	*	SEI 3	*	*	*	BA 3	*	TPRN 3	B	LY 3	LA 3	LZ 3	LX 3	LP	-	OPI	BL	BM
0100	4	SZM 0	SEY 4	TAM 0	SMR	TMA	A 4	XAM 0	SDP 0	TACM	SEI 4	TACP	RTM	TPAC	BA 4	—	LDP 0	B	LY 4	LA 4	LZ 4	LX 4	LP	—	OPI	BL	BM
0101	5	SZM 1	SEY 5	TAM 1	*	*	A 5	XAM 1	SDP 1	*	SEI 5	*	*	*	BA 5	—	LDP 1	B	LY 5	LA 5	LZ 5	LX 5	LP	—	OPI	BL	BM
0110	6	SZM 2	SEY 6	TAM 2	SMR1	*	A 6	XAM 2	SDP 2	*	SEI 6	—	—	SEC	BA 6	—	LDP 2	B	LY 6	LA 6	LZ 6	LX 6	LP	—	OPI	BL	BM
0111	7	SZM 3	SEY 7	TAM 3	*	*	A 7	XAM 3	SDP 3	*	SEI 7	TATM	—	STM	BA 7	—	LDP 3	B	LY 7	LA 7	LZ 7	LX 7	LP	—	OPI	BL	BM
1000	8	EIT	SEY 8	TABN 0	SZB 0	TNAB 0	A 8	XAMD 0	DEY	RC	SEI 8	IQ	SZC	BMAB 0	BMA 0	XAMD1 0	RT	B	LY 8	LA 8	LZ 8	LX 8	LP	—	OPI	BL	BM
1001	9	EIA	SEY 9	TABN 1	SZB 1	TNAB 1	A 9	XAMD 1	*	*	SEI 9	*	*	BMAB 1	BMA 1	XAMD1 1	RTI	B	LY 9	LA 9	LZ 9	LX 9	LP	—	OPI	BL	BM
1010	A	EIB	SEY 10	TABN 2	SZB 2	TNAB 2	A 10	XAMD 2	*	SC	SEI 10	*	CMA	BMAB 2	BMA 2	XAMD1 2	RTS	B	LY 10	LA 10	LZ 10	LX 10	LP	—	OPI	BL	BM
1011	B	EIAB	SEY 11	TABN 3	SZB 3	TNAB 3	A 11	XAMD 3	*	*	SEI 11	*	*	BMAB 3	BMA 3	XAMD1 3	*	B	LY 11	LA 11	LZ 11	LX 11	LP	—	OPI	BL	BM
1100	C	DIT	SEY 12	IR1	SST	OD	A 12	XAMI 0	INY	SB 0	SEI 12	RB 0	TSM	TCMA	BMA 4	XAMI1 0	TSMI	B	LY 12	LA 12	LZ 12	LX 12	LP	—	OPI	BL	BM
1101	D	DIA	SEY 13	IR2	RST	*	A 13	XAMI 1	*	SB 1	SEI 13	RB 1	*	*	BMA 5	XAMI1 1	*	B	LY 13	LA 13	LZ 13	LX 13	LP	—	OPI	BL	BM
1110	E	DIB	SEY 14	ID	IST	SU	A 14	XAMI 2	*	SB 2	SEI 14	RB 2	TMS	*	BMA 6	XAMI1 2	TMSI	B	LY 14	LA 14	LZ 14	LX 14	LP	—	OPI	BL	BM
1111	F	DIAB	SEY 15	TBTM	*	*	A 15	XAMI 3	*	SB 3	SEI 15	RB 3	*	*	BMA 7	XAMI1 3	*	B	LY 15	LA 15	LZ 15	LX 15	LP	—	OPI	BL	BM

Note: $I_3 \sim I_0$ indicate the low-order 4 bits of the machine code and $I_9 \sim I_4$ show the high-order 6 bits Hexadecimal expressions of the codes are also given. All instructions are one word.

* — : Do not use these codes.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	$-0.3 \sim 6.0$	V
V_I	Input voltage		$-0.3 \sim V_{CC}+0.3$	V
V_O	Output voltage		$0 \sim V_{CC}$	V
Pd	Power dissipation	Ta=25℃	300	mW
Topr	Operating free-air temperature range		$0 \sim 50$	℃
Tstg	Storage temperature range		$-40 \sim 125$	℃

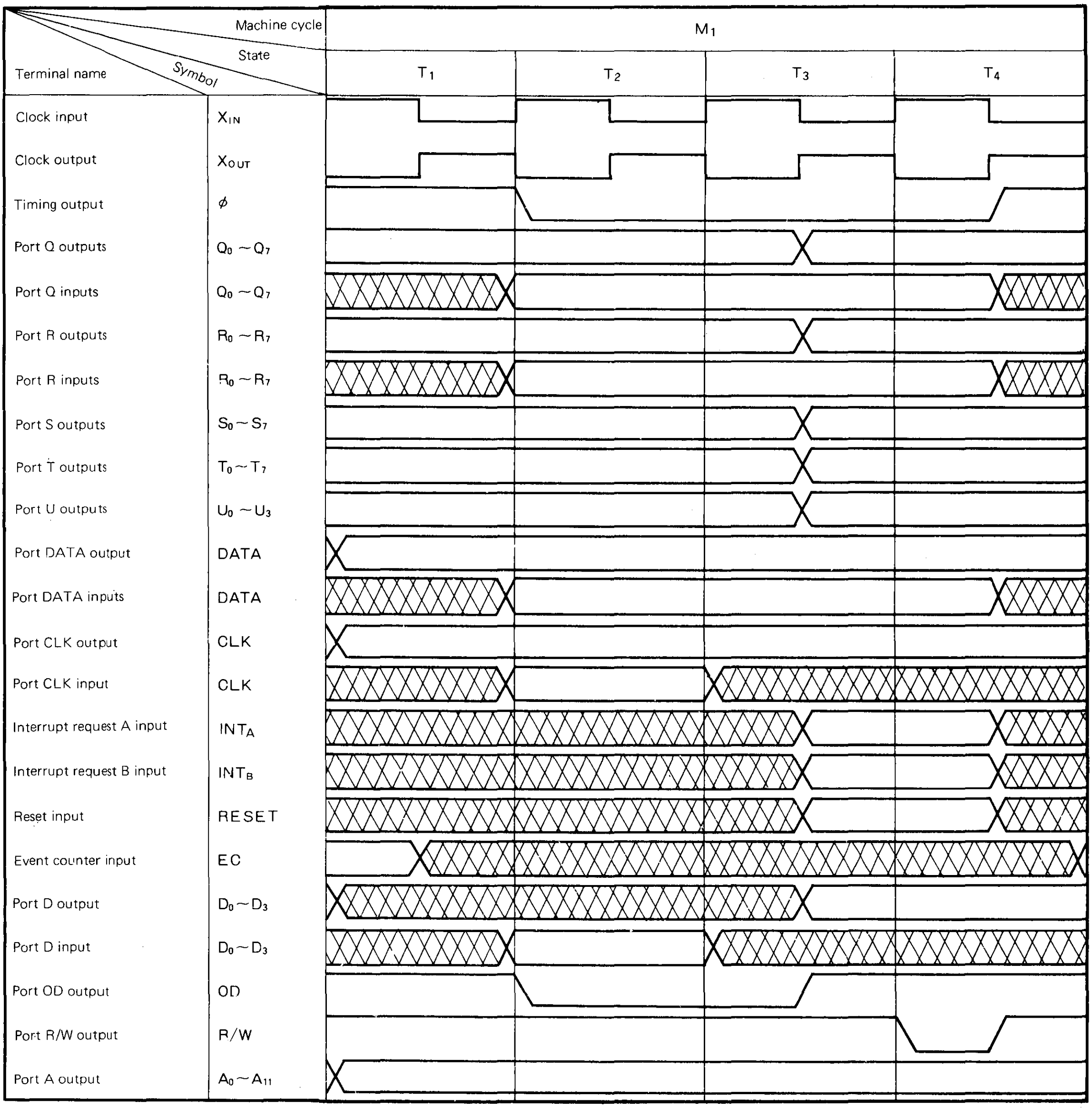
RECOMMENDED OPERATING CONDITIONS ($T_a = 0 \sim 50°C$, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage	$V_{CC}-0.8$		V_{CC}	V
V_{IL}	Low-level input voltage	0		0.8	V
$f(\phi)_1$	Internal clock oscillation frequency (Delay time is not taken into account by external RAM)	100		455	kHz
$f(\phi)_2$	Internal clock oscillation frequency (Standard external RAM is connected)	100		350	kHz
$D(\phi)$	Clock duty cycle	40	50	60	%

ELECTRICAL CHARACTERISTICS ($T_a = 0 \sim 50°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$, $f(\phi) = 100 \sim 455kHz$)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
I_{OH}	High-level output current	$V_{OH}=(V_{CC}-0.8)V$	-0.36			mA
I_{OL}	Low-level output current	$V_{OL}=0.8V$			0.36	mA
I_{CC}	Supply current from V_{CC}	f=455kHz, V_{CC}=5V Ta=25℃ Clock input applied from the external		0.4	1	mA

BASIC TIMING DIAGRAM



Note: The crosshatched area indicates invalid input.

SINGLE-CHIP 4-BIT CMOS MICROCOMPUTER

DESCRIPTION

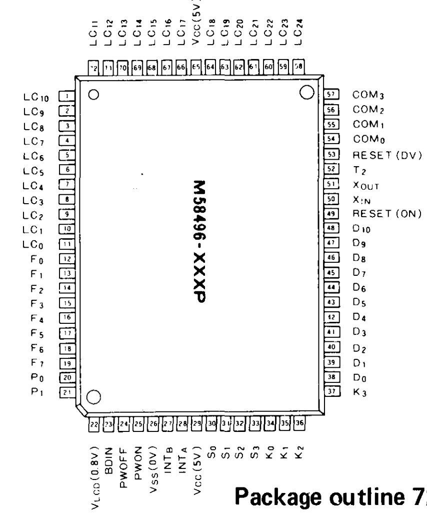
The M58496-XXXP is a single-chip 4-bit microcomputer fabricated using CMOS technology. Its features are liquid crystal display direct drive circuit, current saving circuit for back-up of a 22-stage frequency divider and RAM.

This device is designed for applications in which clock and liquid crystal display functions are included and where the low-power dissipation achieved by CMOS is especially important.

FEATURES

- Single 5V power supply
- Basic machine instructions 77
- Basic instruction execution time
  (at 4.2MHz liquid crystal frequency) 7.7μs
- Memory capacity: ROM . . . . . . 2048 words x 10 bits
  Internal RAM . . 128 words x 4 bits
  External RAM . . 256 words x 4 bits
- Internal crystal oscillation circuit
- Internal 22-stage frequency divider
- Low voltage detector circuit
- Internal current saving circuit while idling
- Subroutine nesting . . . . 3 levels
  Internal timer: Prescaler . . . 7 bits Timer . . . 4 bits
- Output ports for liquid crystal display
  segment signal (port LC) . . . . 25 bits
  common signal (port COM) . . . . 4 bits
- I/O Ports (ports K and S) . . . . 4 bits x 2
- Output port (port D) . . . . 1 bit x 11

PIN CONFIGURATION (TOP VIEW)



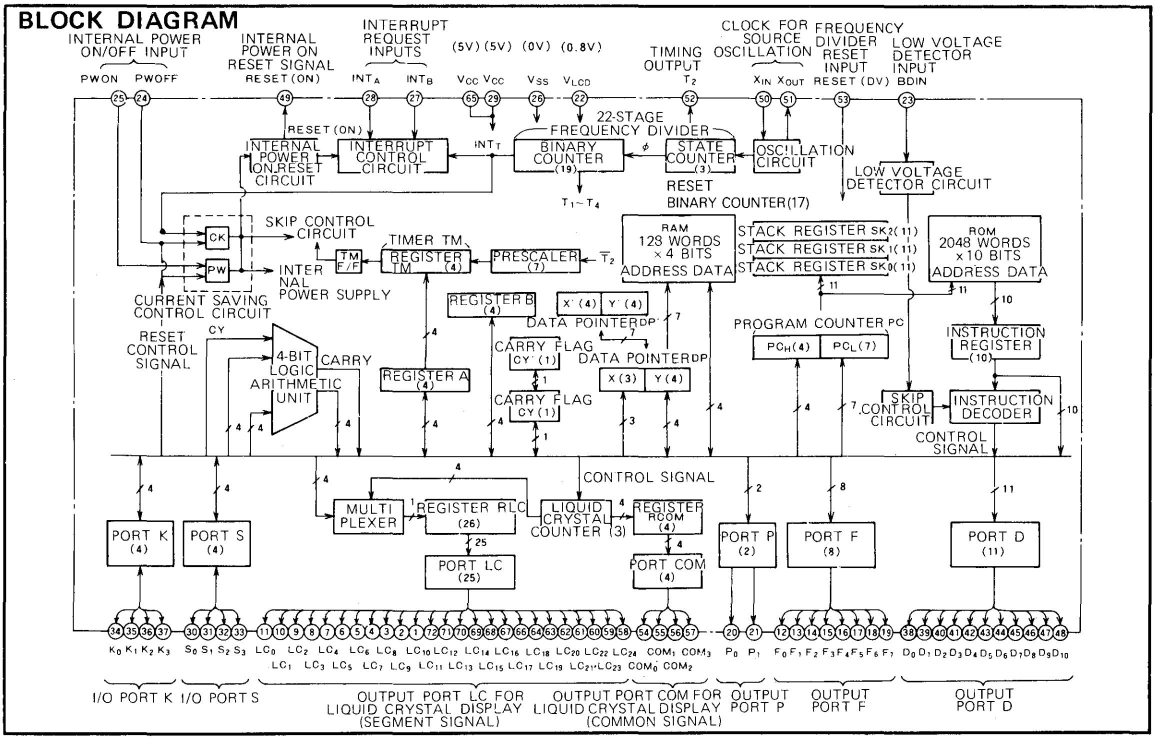
Package outline 72P2

- Output port (port F) . . . . 1 bit x 8
- Output port (port P) . . . . 1 bit x 2
- Interrupt function . . . . 4 factors, 1 level

APPLICATIONS

- Electronic cash registers and calculators with printer
- Office machines, intelligent terminals and data terminals
- Electronic Games
- Electronic coin and changer machines
- Sewing machines

BLOCK DIAGRAM



SINGLE-CHIP 4-BIT CMOS MICROCOMPUTER

FUNCTION

The M58496-XXXP consists of mask ROM and RAM, a 4-bit arithmetic logic unit, crystal oscillation circuit, 22-stage frequency divider, power saving circuit, low voltage detector circuit, 4-bit timer, interrupt circuit and a liquid crystal display direct drive circuit. The RAM capacity can easily be expanded by the external connection of 256-word by 4-bit CMOS RAM.

The ROM storage is organized as 16 pages of 128 words which is used mainly for programs. Addressing the ROM is done through the program counter. The address register is structured as a 7-bit address register and a 4-bit page register. The address register is counted up as nonbranching instructions are executed. When a nonbranching instruction at address 127 on a page is executed an overflow of the address register is produced. This carry (overflow) is disregarded so the page register is not counted up and the next instruction to be executed will come from address 0 on the same page.

When an interrupt request is accepted control is transferred to fixed addresses as follows: in case of an internal power on reset signal (RESET(ON)) the program is set to page 0 address 0, for the INT_A signal it is set to page 0 address 2, for the INT_B signal it is set to page 0 address 4 and for the output signal INT_T(second signal) of the 22-stage frequency divider it is set to page 0 address 8.

The internal RAM which is configured as 8 files of 16 words is used for data storage and each word can be addressed. The internal RAM is addressed by a 7-bit data pointer. The internal RAM can be augmented by external RAM consisting of up to 16 files of 16 words. The external RAM is addressed by the 8-bit combined register Y (4 bits) and register B (4 bits).

RAM addressing, register-to-register transfers, RAM-to-accumulator transfers, arithmetic operations, input/output operations and timer operation are performed mainly through register A (accumulator).

The current saving circuit used in conjunction with the 22-stage frequency divider and RAM can be controlled by the PWOFF input and instruction.

The low voltage detector circuit is also active while the power source is a battery. Low voltage is sensed by the program and an indication can be output.

The output ports for direct drive of the liquid crystal display are port LC (25 terminals) and port COM (4 terminals). The liquid crystal display can be driven by 1/4 duty, 1/3 bias or 1/3 duty, 1/3 bias.

Output port D consists of 11 individually latched bits that can be used to output not only 1-bit data but can also output data such as the contents of register Y of the data pointer and 8-bit addresses for external RAM.

Output port F consists of 8 individually latched bits that can be used to output data. It can be set or reset by instructions.

Output port P consists of 2 terminals through which a synchronous signal of 1 machine cycle width can be output by instruction.

The combined 7-bit output of ports F and P can be used to directly fetch the contents of ROM addressed by the data field of an instruction.

The I/O ports K and S consist of 4 terminals through which data can be transferred to and from register A.

PERFORMANCE SPECIFICATIONS

Item			Performance
Number of basic instructions			77
Execution time of basic instructions			7.7μs (V_{CC}=5V, f=4.1943MHz)
Clock frequency			250～525kHz
Memory Capacity	ROM		2048 words x 10 bits
	Internal RAM		128 words x 4 bits
	External RAM		256 words x 4 bits
I/O Port	LC	Liquid crystal display output	25 x 1 bit
	COM		4 bits
	K	Input	4 bits
		Output	4 bits (Note 1)
	S	Input	4 bits
		Output	4 bits (Note 1)
	D	Output	11x 1 bit (open drain)
	F	Output	8 x 1 bit (Note 1)
	P	Output	2 x 1 bit (Note 1)
Frequency divider			22-stage built in
Current saving circuit			Built in
Low voltage detector			Built in
Subroutine nesting			3 levels (including 1 level of interrupt)
Interrupt request			4 factors, 1 level
Clock generation circuit			Built in (4.1943 MHz crystal oscillator external) (Note 2)
Input/output port	Output voltage		6V (max)
	Output current		-0.4 mA (min.)
Power supply voltage	V_{CC}		5V (nom)
	V_{SS}		0V
Liquid crystal display driving supply voltage			0.8V (nom)
Element structure			CMOS
Package			72-pin plastic molded flat package
Power dissipation (open output terminals)	In operation		5mW (V_{CC}=5V, 525 kHz)
	In idle		1.5mW (V_{CC}=5V, 525 kHz)

Note 1: Ports K, S, F, and P are connected to high-impedance pull-down resistors. When high-driving current is required, external resistors are required.

2: External oscillator can be selected by mask option.
(1) 4.1943 MHz crystal oscillator
(2) 455 kHz ceramic oscillator

PIN DESCRIPTION

Pin	Name	Input or output	At reset (internal power-on)	Function
X_{IN}	Source oscillation clock input	Input	—	Incorporates the clock oscillation circuit, for setting the frequency. An oscillation reference device such as a crystal oscillator is connected between X_{IN} and X_{OUT}. When an external clock is used, connect the clock oscillation source to the X_{IN} pin and leave the X_{OUT} pin open.
X_{OUT}	Source oscillation clock output	Output	—	
PWON	Internal power on input	Input	—	Incorporates the power saving circuit. Its control inputs are PWON and PWOFF. The 22-stage frequency divider and RAM are put in the idle state by a PWOFF input.
PWOFF	Internal power off input	Input	Low level	
RESET (DV)	Frequency divider reset input	Input	—	Incorporates the 22-stage frequency divider as the crystal oscillation reference device. This is a reset input for up to lower 17 steps of the divider.
BDIN	Low voltage detector input	Input	—	The low voltage detector circuit is built in. A resistor should be connected to the BDIN pin for voltage sensing.
INT_A	Interrupt request A signal	Input	Interrupt disable	This input signal is for an interrupt request. The request is accepted on the rising edge of the signal. Besides these external input signals, an interrupt request INT_T from the 22-stage frequency divider output signal is sensed as an interrupt.
INT_B	Interrupt request B signal	Input	Interrupt disable	
LC_0 ~ LC_{24}	Liquid crystal display segment output	Output	—	Incorporates the liquid crystal display direct drive circuit. It is suitable for liquid crystal display at 1/4 duty and 1/3 bias. The output ports for direct drive of the liquid crystal display are port LC (LC_0~LC_{24}) and port COM (COM_0~COM_3).
COM_0 ~ COM_3	Liquid crystal display common output	Output	—	
V_{LCD}	Power supply for liquid crystal display	—	—	This is the power supply terminal for a liquid crystal display. It includes the bias resistor for the segment and common signals.
D_0 ~ D_{10}	Output port D	Output	Floating	This output port consists of 11 bits. Each output is individually latched and can be selected to be set or reset by the contents of register Y. Also 8 bits of the port can be used to fetch 8-bit addresses for external RAM.
F_0 ~ F_7	Output port F	Output	Low level	The output port consists of 8 bits. Each output is individually latched and can be set or reset by instructions.
P_0, P_1	Output port P	Output	Low level	This output port consists of 2 bits from which 1 synchronous signal of 1 machine cycle width can be output per instruction. The immediate 7-bit field of an instruction can be output through this port in combination with 5 bits of port F.
K_0 ~ K_3	Input/output port K	Input/output	Low level	Ports K and S are 4-bit latched input/output ports through which data can be transferred to and from register A. When output is low-level the output will be high-impedance so it can be used as an input port.
S_0 ~ S_3	Input/output S	Input/output	Low level	
T_2	Timing output	Output	—	The timing output is used for testing the device.
RESET(ON)	Internal power-on reset signal	Output	Low level	When the internal power supply is switched on, a built in automatic reset circuit generates a high-level reset signal that resets the I/O ports.

DESCRIPTION OF OPERATION

Program Counter PC

The program counter is an 11-bit address register. The high-order 4 bits designate the page number and as a group are called PC_H. The low-order 7 bits designate the address on the page and as a group are called PC_L. The PC designates the address of the 2048 words by 10-bit mask-programmable ROM. The ROM is organized into 16 pages of 128 words. As instructions are fetched from ROM, PC_L is incremented so that unless there is a branch executed instructions are fetched and executed in sequence. Care must be taken when the last instruction on a page (address 127) is executed because when PC_L is incremented it becomes zero with a carry, but the carry is disregarded so the next instruction to be fetched will be the start of the same page. Therefore to move to the next page PC_H must modified by using branch instructions such as BL, BML, BLA and BMLA.

Pages 14 and 15 are special pages designed to accommodate subroutines. Subroutines starting on page 14 can be called by 1-word instructions BM or BMA. These instructions automatically load PC_H to designate page 14 and in addition the return address and control status are saved so they can be restored when the subroutine transfers control back to the main program. If the instructions BM or BMA are executed on page 14, they execute a branch within page 14 without saving any information. If the instructions B or BA are executed on page 14, they execute a branch to page 15.

Stack Registers SK_0, SK_1, SK_2

The 3-level stack register consists of 11-bit registers for storing the contents of the program counter when control is transferred from the main program to a subroutine or interrupt. When control is transferred back to the main program, the PC can be restored. There are 3 levels, but when 1 level is saved for interrupts it leaves 2 levels for subroutine nesting.

Data Pointers DP, DP'

The data pointer is a 7-bit register used to designate the address of RAM or the bit position of output port D. The data pointer is composed of the 3-bit register X and the 4-bit register Y. Internal RAM is organized as 8 files of 16 words. Register X designates the file and register Y designates the word position of a file or the bit position of output port D.

The data pointer DP' is selected by software during interrupt processing to leave the contents of DP unchanged (saves the DP).

External RAM is organized as 16 files of 16 words that can be added to the system to expand memory. Register Y designates the word position of a file while register B designates the file.

Register A (accumulator) and Carry Flags CY, CY'

Register A is the 4-bit accumulator forming the heart of the 4-bit microcomputer. Data processing operations such as arithmetic, transfer, exchange, conversion, and input/output are executed principally through this register.

The carry flags CY are to store the carry or borrow from the most significant bit of the arithmetic unit resulting from executing the various instructions. It can be tested and used for various purposes. In principle it acts as a 1-bit flag.

The carry flag CY' is selected by software to leave the contents of CY unchanged (saves the CY).

Register B (Auxiliary Register)

Register B is a 4-bit register used for temporary storage of 4-bit data. It also is used to designate the file number of external RAM.

Arithmetic Logic Unit (ALU)

The arithmetic logic unit performs 4-bit arithmetic and logical operations. The heart of the ALU is a 4-bit adder and the logic circuit associated with it. It performs operations such as additions, complement conversions, logic arithmetic comparisons and bit processing.

Frequency Divider and Timer

The frequency divider divides the basic oscillation frequency into 22 stages. It is connected to the basic oscillation device through X_{IN} and X_{OUT}. The frequency divider generates the interrupt request signal INT_T to the interrupt control circuit. The frequency divider sets flag CK for controlling the power saving circuit.

Basic oscillation for the timer is the timing signal T_2. The timer is composed of a 7-bit prescaler and a 4-bit counter. Timer flag TMF/F is set when a timer overflows, and is sensed by the TTM instruction. The 4-bit timer counter is set by the STM instruction. Prescaler and timer flag are reset at the same time.

Power Saving Circuit

The power saving circuit is controlled by the CK flag and PW. Its output is input to the internal power supply reset circuit and generates an interrupt request signal RESET (ON). Control is transferred unconditionally to address 0 on page 0 and resets the I/O ports. The interrupt request signal RESET (ON) generates on the rising edge of internal power supply on reset output. Internal power supply is switched off by the external terminal and stop instruction, but power is maintained to the following circuits:

1. Internal data memory (RAM)
2. Clock oscillation circuit
3. 22-stage frequency divider
4. Low voltage detector circuit
5. Power saving circuit

Low Voltage Detector Circuit

The low voltage detector circuit connects the resistor for sensing voltage to the BDIN terminal. A falling voltage level is sensed by the program and can be displayed by using apt output port.

Interrupt Functions

The M58496-XXXP has internal circuits to process interrupt requests from 4 single level sources. The 4 interrupt request sources are external interrupt signals INT_A and INT_B, internal power supply reset output RESET (ON), output INT_T from the 22-stage frequency divider. Interrupt requests INT_A, INT_B and INT_T are enabled by the instructions EIA, EIB and EIT respectively and disabled by the instruction DIA, DIB and DIT respectively. Interrupt requests from the internal power supply through reset output RESET (ON) cannot be disabled and will cause an interrupt whenever received.

During the interrupt enable state an interrupt request by INT_A or INT_B is accepted on the rising edge of the signal. When an interrupt request is received during the interrupt disable state it is latched, but is not executed. When the disable is removed thereafter by executing the corresponding interrupt enable instruction, the interrupt request will be accepted immediately and control transferred to the interrupt routine because the request was latched. A current interrupt request, held by latching during interrupt disable state is reset when the corresponding interrupt disable instruction is executed.

One level of the 3-level stack register is required when interrupt programs are used. This leaves 2 levels available for subroutine processing. After an interrupt is processed control is returned to the main program by executing a return instruction such as RTI. Care must be taken after starting an interrupt program to save the contents the data pointer DP, register A, carry flag and any other registers used, so the contents can be restored before returning to the main program. The contents must be saved and restored by the interrupt program.

When an interrupt request is accepted the program counter, interrupt enable flag and skip flag are affected as follows:

(1) Program counter
The contents (the current program address) are stored in the stack register. Control is transferred to address 0 on page 0 by a RESET (ON) interrupt, to address 2 on page 0 by an INT_A interrupt, to address 4 on page by an INT_B interrupt or to address 8 on page 0 by an INT_T interrupt by setting the control counter to 00, 02, 04 or 08 respectively. When control is transferred to address 0 page 0, the instruction is invalid and is not executed, so the first instruction is executed from address 1 on page 0.

(2) Interrupt enable flags
When an interrupt request is accepted additional interrupts are disabled until the accepted interrupt is processed. Except that a RESET (ON) interrupt may be accepted at any time.

(3) Skip flags
The skip flags are used to indicate an instruction skip and the NOP state for instructions LXY and LA are saved. A special stack is provided for saving these flags.

General-Purpose I/O ports K, S, F, P and D

These 4-bit or 1-bit general-purpose registers are used for such things as data transfer between register A, instruction transfers, 1-bit transfers as selected by register Y, storing 7-bit immediate field data of instructions fetched from ROM, and data transfers between external RAM. Each output has a latch and its output circuit contains an open drain resistor or a pulldown resistor (high-impedance).

I/O ports K, S

Ports K and S are 4-bit latched I/O ports, that can transfer data to and from register A. Output latches are reset by the DIKS instruction when the port is being used as an input port.

Output port F

Port F is an 8-bit latched output port, that has independent latches for each bit. The individual bits can be set by the SF instruction and reset by the RF instruction.

Output port P

Port P is a 2-bit latched output port, that is usually in low-level, but can output the machine cycle high-level synchronous signal by SP_0 or SP_1 instructions. The 7 bits (F_4~F_0, P_1, P_0) can be used for direct fetching of the immediate field of the OTRO instruction.

Output port D

Port D is an 11-bit latched output port, that has independent latches for each bit. The contents for register Y indicate the individual bit to be set by the SD instruction or to be reset by the RD instruction. The 8-bit address of external memory (RAM) is output through this port.

Liquid Crystal Display Drive Circuit

The liquid crystal display direct drive circuit is composed of the following units. A block diagram of the units is shown in Fig. 1.

1 Control counter for the liquid crystal display
This is an octal counter composed of 3 bits and is counted down by the ELC instruction. The contents of the counter select 1 bit of register A and transfer data in order to the segment register RLC by the TLC instruction and determines the frame frequency for the liquid crystal display by transferring the contents of the counter to common register RCOM.

2 Register A
This 4-bit register is the accumulator. Its function is to control data processing, arithmetic operations control functions and input/output of the microcomputer.

3 Segment register RLC
The 26-bit segment register stores selected 1-bit data from register A by execution of the TLC instruction. It shifts 1 bit in order and stores the segment signals for the liquid crystal display device.

4 Common register RCOM
The 4-bit common register stores the common signal for the liquid crystal display. The input for the common register is the converted contents of the control counter for the liquid crystal display.

5 Port LC
The 26-bit latched port LC stores data in parallel by the ELC or DLC instruction from the segment register RLC. A bias resistor provides for the output at 2 levels and the 25 low-order bits are output as standard type. The high-order bit is not output to an external terminal.

6 Port COM
Port COM has 4 bits of latched storage. The data is transferred in parallel by the ELC or DLC instruction through the common register (RCOM). The outputs of this port have 3 biased levels by means of bias resistors.

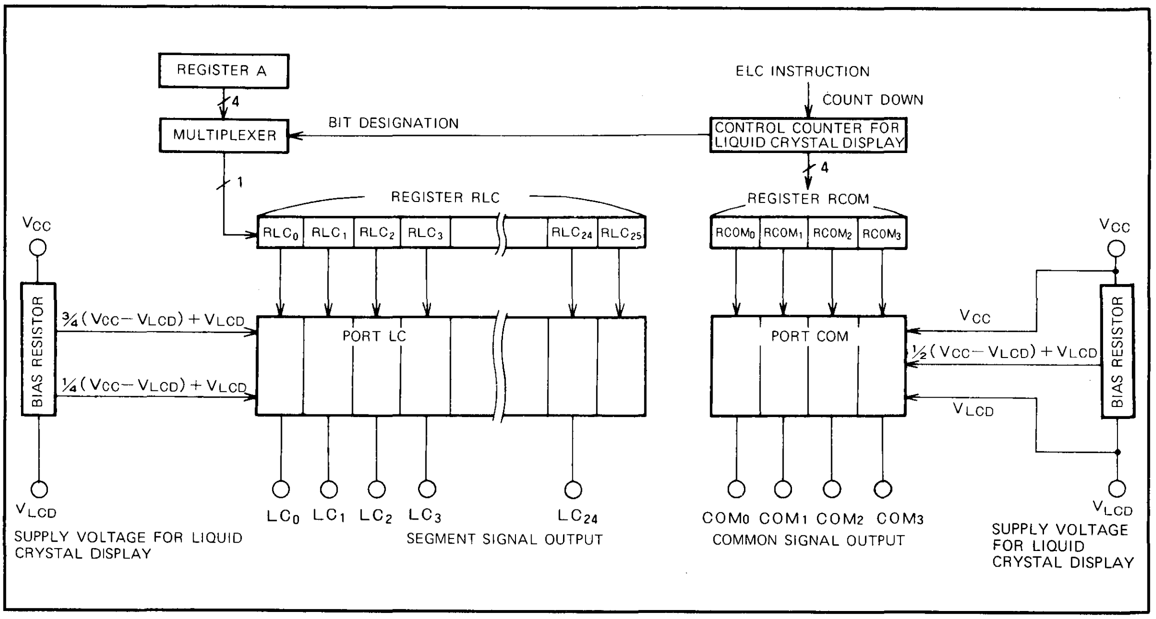


Fig. 1 Liquid crystal display drive circuit block diagram

RESET FUNCTION

As shown in Fig. 2, when the PWOFF input of the M58496-XXXP is driven low for at least 10ms, the input/output ports are reset and the interrupt disabled state is entered. (Refer to the descriptions of the power-on reset states in the Pin Description.) Next, if the PWON input is driven high, or an interrupt is generated by the internal power-on reset RESET (ON) caused by a frequency divider output INT_T, the program counter is set to address 0 page 0 as a starting location.

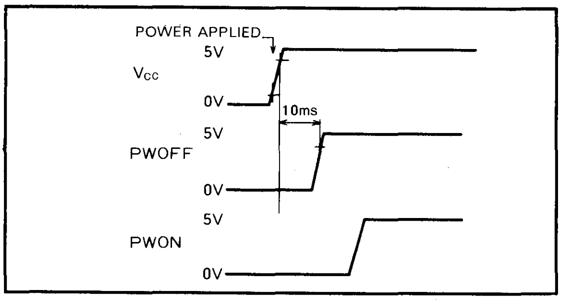


Fig. 2 Power-on reset circuit

CLOCK GENERATOR CIRCUIT

A built-in clock generator circuit has been provided and a quartz crystal or ceramic element (mask option) can be externally connected. In addition, an external clock source may be connected to pin X_{IN}, leaving pin X_{OUT} open. Circuit examples are shown in Fig. 3 and Fig. 4.

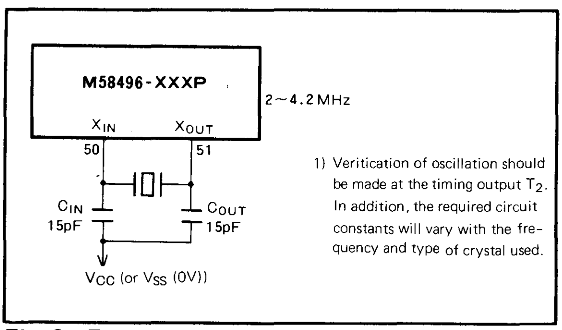


Fig. 3 External circuit connected by crystal oscillator

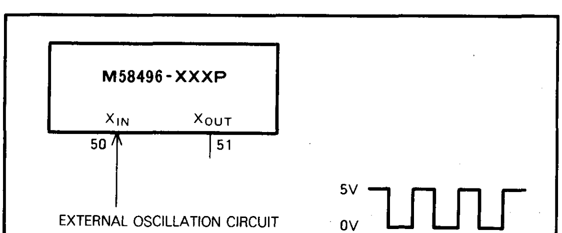


Fig. 4 External clock input circuit

Documentation Required Upon Ordering

The following information should be provided when ordering a custom mask.

(1) M58496-XXXP mask confirmation sheet
(2) ROM data .... 3 EPROM sets
(3) Oscillation frequency selection
.... On confirmation sheets
(4) Frequency divider output selection (1Hz/2Hz)
.... On confirmation sheets

INSTRUCTION CODE LIST (Note 1)

D9~D4	Hexadecimal notation	00 0000	00 0001	00 0010	00 0011	00 0100	00 0101	00 0110	00 0111	00 1000	00 1001	00 1010	00 1011	00 1100	00 1101	00 1110 ~ 00 1111	01 0000 ~ 01 0111	01 1000 ~ 01 1111	10 0000 ~ 10 0111	10 1000 ~ 10 1111	11 0000 ~ 11 0111	11 1000 ~ 11 1111
D3~D0		0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	0 A	0 B	0 C	0 D	0E~0F	10~17	18~1F	20~27	28~2F	30~37	38~3F
0000	0	NOP	TLC	INY	SZB 0	SEY 0	SEI 0	SF 0	BL BLA BML BMLA	–	RAR	TAM 0	XAMD 0	A 0	LA 0	–	OTRO	LXY	BM	BMA	B	BA
0001	1	SCOM	DIKS	DEY	SZB 1	SEY 1	SEI 1	SF 1	BL BLA BML BMLA	–	–	TAM 1	XAMD 1	A 1	LA 1	–	OTRO	LXY	BM	BMA	B	BA
0010	2	EIA	SFK	XDP	SZB 2	SEY 2	SEI 2	SF 2	BL BLA BML BMLA	*	IK	TAM 2	XAMD 2	A 2	LA 2	–	OTRO	LXY	BM	BMA	B	BA
0011	3	DIA	SFS	TYA	SZB 3	SEY 3	SEI 3	SF 3	BL BLA BML BMLA	SEAM	IS	TAM 3	XAMD 3	A 3	LA 3	–	OTRO	LXY	BM	BMA	B	BA
0100	4	EIB	*	SC	RT	SEY 4	SEI 4	SF 4	BL BLA BML BMLA	*	TBA	TAM 4	XAMD 4	A 4	LA 4	–	OTRO	LXY	BM	BMA	B	BA
0101	5	DIB	DLC	RC	RTS	SEY 5	SEI 5	SF 5	BL BLA BML BMLA	TAY	–	TAM 5	XAMD 5	A 5	LA 5	–	OTRO	LXY	BM	BMA	B	BA
0110	6	DETS	*	XC	RTI	SEY 6	SEI 6	SF 6	BL BLA BML BMLA	AND	XAB	TAM 6	XAMD 6	A 6	LA 6	–	OTRO	LXY	BM	BMA	B	BA
0111	7	DETR	ELC	*	*	SEY 7	SEI 7	SF 7	BL BLA BML BMLA	EXL	TAB	TAM 7	XAMD 7	A 7	LA 7	–	OTRO	LXY	BM	BMA	B	BA
1000	8	EIT	SP0	*	*	SEY 8	SEI 8	RF 0	BL BLA BML BMLA	*	SB 0	XAM 0	XAMI 0	A 8	LA 8	–	OTRO	LXY	BM	BMA	B	BA
1001	9	DIT	*	SD	*	SEY 9	SEI 9	RF 1	BL BLA BML BMLA	CMA	SB 1	XAM 1	XAMI 1	A 9	LA 9	–	OTRO	LXY	BM	BMA	B	BA
1010	A	STM	SP1	*	*	SEY 10	SEI 10	RF 2	BL BLA BML BMLA	AM	SB 2	XAM 2	XAMI 2	A 10	LA 10	–	OTRO	LXY	BM	BMA	B	BA
1011	B	POF2	*	*	*	SEY 11	SEI 11	RF 3	BL BLA BML BMLA	*	SB 3	XAM 3	XAMI 3	A 11	LA 11	–	OTRO	LXY	BM	BMA	B	BA
1100	C	POF1	OTAD	*	*	SEY 12	SEI 12	RF 4	BL BLA BML BMLA	*	RB 0	XAM 4	XAMI 4	A 12	LA 12	–	OTRO	LXY	BM	BMA	B	BA
1101	D	SDET	*	RD	*	SEY 13	SEI 13	RF 5	BL BLA BML BMLA	*	RB 1	XAM 5	XAMI 5	A 13	LA 13	–	OTRO	LXY	BM	BMA	B	BA
1110	E	TTM	ADRT	*	*	SEY 14	SEI 14	RF 6	BL BLA BML BMLA	AMC	RB 2	XAM 6	XAMI 6	A 14	LA 14	–	OTRO	LXY	BM	BMA	B	BA
1111	F	TCK	TPW	*	SZC	SEY 15	SEI 15	RF 7	BL BLA BML BMLA	AMCS	RB 3	XAM 7	XAMI 7	A 15	LA 15	–	OTRO	LXY	BM	BMA	B	BA

Note 1: This list shows the machine codes and corresponding machine instructions D_3~D_0 indicate the low-order 4 bits of the machine code and D_9~D_4 indicate the high-order 6 bits. Hexadecimal numbers are also shown that represent the codes. An instruction may consist of one or two words, but only the first word is listed. Code combination indicated with asterisk (*) and bar (–) must not be used.

Two-word instructions

	Second word
BL	11 0xxx yyyy
BLA	11 1xxx XXXX
BML	10 0xxx yyyy
BMLA	10 1xxx XXXX

MACHINE INSTRUCTIONS (Note 1)

Classification	Mnemonic	Instruction code D_9D_8	$D_7D_6D_5D_4$	$D_3D_2D_1D_0$	Hexadecimal	No. of words	No. of cycles	Functions	Skip conditions	Flag CY
RAM address	LXY x,y	0 1	1 x x x	y y y y	18y + x	1	1	(X)←x, where, x=0~7 (Y)←y, where, y=0~15	Consecutively described	×
	INY	0 0	0 0 1 0	0 0 0 0	020	1	1	(Y)←(Y)+1	—	×
	DEY	0 0	0 0 1 0	0 0 0 1	021	1	1	(Y)←(Y)−1	—	×
Register to register transfer	TAB	0 0	1 0 0 1	0 1 1 1	097	1	1	(A)←(B)	—	×
	TBA	0 0	1 0 0 1	0 1 0 0	094	1	1	(B)←(A)	—	×
	XAB	0 0	1 0 0 1	0 1 1 0	096	1	1	(A)↔(B)	—	×
	TAY	0 0	1 0 0 0	0 1 0 1	085	1	1	(A)←(Y)	—	×
	TYA	0 0	0 0 1 0	0 0 1 1	023	1	1	(Y)←(A)	—	×
	XDP	0 0	0 0 1 0	0 0 1 0	022	1	1	(DP)↔(DP')	—	×
RAM to accumulator transfer	TAM j	0 0	1 0 1 0	0 j j j	0Aj	1	1	(A)←(M(DP)) (X)←(X)∀j, where, j=0~7	—	×
	XAM j	0 0	1 0 1 0	1 j j j	0A8 + j	1	1	(A)↔(M(DP)) (X)←(X)∀j, where, j=0~7	—	×
	XAMD j	0 0	1 0 1 1	0 j j j	0Bj	1	1	(A)↔(M(DP)), (Y)←(Y)−1 (X)←(X)∀j, where, j=0~7	(Y)=15	×
	XAMI j	0 0	1 0 1 1	1 j j j	0B8 + j	1	1	(A)↔(M(DP)), (Y)←(Y)+1 (X)←(X)∀j, where, j=0~7	(Y)=0	×
Arithmetic	LA n	0 0	1 1 0 1	n n n n	0Dn	1	1	(A)←n, where, n=0~15	Consecutively described	×
	AM	0 0	1 0 0 0	1 0 1 0	08A	1	1	(A)←(A)+(M(DP))	—	×
	AMC	0 0	1 0 0 0	1 1 1 0	08E	1	1	(A)←(A)+(M(DP))+(CY) (CY)← Carry	—	0/1
	AMCS	0 0	1 0 0 0	1 1 1 1	08F	1	1	(A)←(A)+(M(DP))+(CY) (CY)← Carry	(CY)=0	0/1
	A n	0 0	1 1 0 0	n n n n	0Cn	1	1	(A)←(A)+n, where, n=0~15	Carry = 0	×
	SC	0 0	0 0 1 0	0 1 0 0	024	1	1	(CY)←1	—	1
	RC	0 0	0 0 1 0	0 1 0 1	025	1	1	(CY)←0	—	0
	XC	0 0	0 0 1 0	0 1 1 0	026	1	1	(CY)↔(CY')	—	(CY')
	SZC	0 0	0 0 1 1	1 1 1 1	03F	1	1	Skip if (CY)=0	(CY)=0	×
	AND	0 0	1 0 0 0	0 1 1 0	086	1	1	(A)←(A)∧(M(DP))	—	×
	EXL	0 0	1 0 0 0	0 1 1 1	087	1	1	(A)←(A)∀(M(DP))	—	×
	CMA	0 0	1 0 0 0	1 0 0 1	089	1	1	(A)←($\overline{A}$)	—	×
	RAR	0 0	1 0 0 1	0 0 0 0	090	1	1	(A_{n-1})←(A_n) (CY)←(A_0), (A_3)←(CY)	—	(A_0)
Bit manipulation	SB i	0 0	1 0 0 1	1 0 i i	098 + i	1	1	(Mi(DP))←1, where, i=0~3	—	×
	RB i	0 0	1 0 0 1	1 1 i i	09C + i	1	1	(Mi(DP))←0, where, i=0~3	—	×
	SZB i	0 0	0 0 1 1	0 0 i i	03i	1	1	Skip if (Mi(DP))=0, where, i=0~3	(Mi(DP))=0 where, i=0~3	×
Compare	SEAM	0 0	1 0 0 0	0 0 1 1	083	1	1	Skip if (M(DP))=(A)	(M(DP))=(A)	×
	SEY y	0 0	0 1 0 0	y y y y	04y	1	1	Skip if (Y)=y, where, y=0~15	(Y)=y, where, y=0~15	×
	SEI n	0 0	0 1 0 1	n n n n	05n	1	1	Skip if (A)=n, where, n=0~15	(A)=n, where, n=0~15	×
	SCOM	0 0	0 0 0 0	0 0 0 1	001	1	1	Skip if (SCA=0) and (SCB=0)	SCA=0 and SCB=0	×

Classification	Mnemonic	D_9D_8	$D_7D_6D_5D_4$	$D_3D_2D_1D_0$	Hexadecimal	No. of words	No. of cycles	Functions	Skip conditions	Flag CY
Branch	**B xy** (Note 2)	1 1	0 x x x	y y y y	3xy	1	1	$(PC_L) \leftarrow 16x+y$ $(PC_H) \leftarrow 15$, $(PC_L) \leftarrow 16x+y$	—	×
	BL pxy	0 0 1 1	0 1 1 1 0 x x x	p p p p y y y y	07p 3xy	2	2	$(PC_H) \leftarrow p$ $(PC_L) \leftarrow 16x+y$	—	×
	BA xX (Note 2)	1 1	1 x x x	X X X X	38X + x	1	1	$(PC_L) \leftarrow 16x+(A)$ $(PC_H) \leftarrow 15$, $(PC_L) \leftarrow 16x+(A)$	—	×
	BLA pxX	0 0 1 1	0 1 1 1 1 x x x	p p p p X X X X	07p 38X + x	2	2	$(PC_H) \leftarrow p$ $(PC_L) \leftarrow 16x+(A)$	—	×
Subroutine call	**BM xy** (Note 2)	1 0	0 x x x	y y y y	2xy	1	1	$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$ $(PC_H) \leftarrow 14$, $(PC_L) \leftarrow 16x+y$ $(PC_H) \leftarrow 14$, $(PC_L) \leftarrow 16x+y$	—	×
	BML pxy	0 0 1 0	0 1 1 1 0 x x x	p p p p y y y y	07p 2xy	2	2	$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$ $(PC_H) \leftarrow p$, $(PC_L) \leftarrow 16x+y$	—	×
	BMA xX (Note 2)	1 0	1 x x x	X X X X	28X + x	1	1	$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$ $(PC_H) \leftarrow 14$, $(PC_L) \leftarrow 16x+(A)$ $(PC_H) \leftarrow 14$, $(PC_L) \leftarrow 16x+(A)$	—	×
	BMLA pxX	0 0 1 0	0 1 1 1 1 x x x	p p p p X X X X	07p 28X + x	2	2	$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$ $(PC_H) \leftarrow p$, $(PC_L) \leftarrow 16x+(A)$	—	×
Return	**RTI**	0 0	0 0 1 1	0 1 1 0	036	1	1	$(PC) \leftarrow (SK0) \leftarrow (SK1) \leftarrow (SK2)$ Restore interrupt skip flags	—	×
	RT	0 0	0 0 1 1	0 1 0 0	034	1	1	$(PC) \leftarrow (SK0) \leftarrow (SK1) \leftarrow (SK2)$	—	×
	RTS	0 0	0 0 1 1	0 1 0 1	035	1	1	$(PC) \leftarrow (SK0) \leftarrow (SK1) \leftarrow (SK2)$	Unconditional	×
Input/output	**DIKS**	0 0	0 0 0 1	0 0 0 1	011	1	1	$(K) \leftarrow 0$, $(S) \leftarrow 0$	—	×
	IK	0 0	1 0 0 1	0 0 1 0	092	1	1	$(A) \leftarrow (K)$	—	×
	IS	0 0	1 0 0 1	0 0 1 1	093	1	1	$(A) \leftarrow (S)$	—	×
	SFK	0 0	0 0 0 1	0 0 1 0	012	1	1	$(K) \leftarrow (A)$	—	×
	SFS	0 0	0 0 0 1	0 0 1 1	013	1	1	$(S) \leftarrow (A)$	—	×
	SD	0 0	0 0 1 0	1 0 0 1	029	1	1	$(D(Y)) \leftarrow 1$, where, $0 \leqq (Y) \leqq 10$	—	×
	RD	0 0	0 0 1 0	1 1 0 1	02D	1	1	$(D(Y)) \leftarrow 0$, where, $0 \leqq (Y) \leqq 10$	—	×
	ADRT	0 0	0 0 0 1	1 1 1 0	01E	1	1	$(D) \leftarrow 0$	—	×
	OTAD	0 0	0 0 0 1	1 1 0 0	01C	1	1	$(D_7 \sim D_4) \leftarrow (B)$ $(D_3 \sim D_0) \leftarrow (Y)$	—	×
	SF m	0 0	0 1 1 0	0 m m m	06m	1	1	$(F\,m) \leftarrow 1$, where, $m = 0 \sim 7$	—	×
	RF m	0 0	0 1 1 0	1 m m m	068 + m	1	1	$(F\,m) \leftarrow 0$, where, $m = 0 \sim 7$	—	×
	OTRO mn	0 1	0 m m m	n n n n	1mn	1	1	$(F_0 \sim F_3) \leftarrow n$, where, $n = 0 \sim 15$ $(F_4, P_0, P_1) \leftarrow m$, where, $m = 0 \sim 7$	—	×
	SP0	0 0	0 0 0 1	1 0 0 0	018	1	1	$(P_0) \leftarrow 1$, where output 1 machine cycle	—	×
	SP1	0 0	0 0 0 1	1 0 1 0	01A	1	1	$(P_1) \leftarrow 1$, where output 1 machine cycle	—	×
	TLC	0 0	0 0 0 1	0 0 0 0	010	1	1	$(R(LC_0)) \leftarrow (Ai)$, where, $i = 0 \sim 3$ $(R(LC_{n+1})) \leftarrow (R(LC\,n))$	—	×
	ELC	0 0	0 0 0 1	0 1 1 1	017	1	1	$(P(LCn)) \leftarrow (R(LCn))$ $(P(COMn)) \leftarrow (R(COMn))$	—	×
	DLC	0 0	0 0 0 1	0 1 0 1	015	1	1	$(P(LCn)) \leftarrow (R(LCn))$ $(P(COM)) \leftarrow \frac{1}{2}(V_{CC} - V_{LCD}) + V_{LCD}$	—	×

Classification \ Item	Mnemonic	Instruction code D_9D_8	$D_7D_6D_5D_4$	$D_3D_2D_1D_0$	Hexadecimal	No. of words	No. of cycles	Functions	Skip conditions	Flag CY
Interrupt	EIA	00	0000	0010	002	1	1	Enables interruption of INT_A signal.	—	×
	DIA	00	0000	0011	003	1	1	Disables interruption of INT_A signal.	—	×
	EIB	00	0000	0100	004	1	1	Enables interruption of INT_B signal.	—	×
	DIB	00	0000	0101	005	1	1	Disables interruption of INT_B signal.	—	×
	EIT	00	0000	1000	008	1	1	Enables interruption of INT_T signal.	—	×
	DIT	00	0000	1001	009	1	1	Disables interruption of INT_T signal.	—	×
Timer	STM	00	0000	1010	00A	1	1	(TM)←(A), (TM F/F)←0 7-bit prescaler presetting	—	×
	TTM	00	0000	1110	00E	1	1	Skip if (TM F/F)=1	(TM F/F)=1	×
Power supply control	TCK	00	0000	1111	00F	1	1	Skip if (CK F/F)=1	(CK F/F)=1	×
	POF1	00	0000	1100	00C	1	1	(CK F/F)←0	—	×
	POF2	00	0000	1011	00B	1	1	(PW F/F)←0	—	×
	TPW	00	0001	1111	01F	1	1	Skip if (PW F/F)=1	(PW F/F)=1	×
	DETS	00	0000	0110	006	1	1	(DET F/F)←1	—	×
	DETR	00	0000	0111	007	1	1	(DET F/F)←0	—	×
	SDET	00	0000	1101	00D	1	1	Skip if (BD_{OUT})=1	(BD_{OUT})=1	×
Misc.	NOP	00	0000	0000	000	1	1	No operation	—	×

Note 1 : When the M58496-XXXP generates a skip it is not necessary to increment the program counter so no additional cycles are required for execution.
2 : Instructions B, BA, BM or BMA execute the second function of the functions column when executed, provided that none of instructions RT, RTS, BL, BML, BLA or BMLA was executed after execution of instruction BM or BMA.

Symbol	Meaning	Symbol	Meaning
A	4-bit register (accumulator)	P(COMn)	Common output port for liquid crystal display
Ai	Indicates the bits of register A. Where i=1~3	P(LCn)	Segment output port for liquid crystal display
B	4-bit auxiliary register	PW F/F	1-bit power supply control flag display
BD_{OUT}	Battery detector signal	R(COMn)	Common register for liquid crystal display (4 bits)
CK F/F	1-bit 1-second flag	R(LCn)	Segment register for liquid crystal display (25 bits)
CY	1-bit carry flag	S	4-bit I/O port
CY'	1-bit carry flag	SCA	Output of bit A of control counter for liquid crystal display
D	11-bit output port	SCB	Output of bit B of control counter for liquid crystal display
Di	Indicates the bits of port D. Where i=0~3	SK0	11-bit stack register
D(Y)	The bit of port D addressed by Y	SK1	11-bit stack register
DP	7-bit data pointer composed of register Y, X	SK2	11-bit stack register
		TM	4-bit timer/counter
Y, Y'	4-bit register	TM F/F	1-bit timer/counter flag
X, X'	3-bit register	xx	2-bit binary variable
DP'	7-bit data pointer	yyyy	4-bit binary variable
DET F/F	1-bit battery detector flag	mmm	3-bit binary variable
F	8-bit output port	nnnn	4-bit binary variable
Fi	Indicates the bits of port F. Where i=0~7	ii	2-bit binary variable
K	4-bit I/O port	jjj	3-bit binary variable
M(DP)	4-bit data of memory addressed by data pointer DP	XXXX	4-bit unknown binary variable (the value does not affect execution)
		←	Indicates direction of data flow
Mi(DP)	A bit of data of memory addressed by data pointer DP where i=0~3	( )	Indicates contents of register memory, etc.
		⩡	Exclusive OR
PC	11-bit program acounter composed of PC_L, PC_H	∧	AND
		—	Negation
PC_L	Low-order 7 bits of the program counter	X	Indicates flag is unaffected by instruction execution
PC_H	High-order 4 bits of the program counter	xy	Label used to indicate the address xxx yyyy
P_0	1-bit output port	C + x	Hexadecimal number C + binary number-X
P_1	1-bit output port		

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.3~6.0	V
V_I	Input voltage		−0.3~V_{CC}+0.3	V
V_O	Output voltage		0 ~V_{CC}	V
P_d	Power dissipation	T_a=25°C	300	mW
T_{opr}	Operating free-air temperature range		0 ~50	°C
T_{stg}	Storage temperature range		−40~125	°C

RECOMMENDED OPERATING CONDITIONS (Ta=0~50°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage		0		V
V_{LCD}	Liquid crystal supply voltage		0.8		V
V_{IH}	High-level input voltage	V_{CC}−0.8		V_{CC}	V
V_{IL}	Low-level input voltage	0		0.8	V
f_{XIN}	Oscillator frequency	2		4.2	MHz
f_ϕ	Internal clock oscillator frequency	250		525	kHz

ELECTRICAL CHARACTERISTICS (Ta=0~50°C, V_{CC}=5V±10%, V_{SS}=0V, f_{XIN}=2~4. 2MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
I_{OH}	High-level output current, port D	V_{OH}=(V_{CC}−0.5)V	−0.4	−0.8		mA
I_{OH}	High-level output current, ports F, P, K, and S	V_{OH}=(V_{CC}−0.5)V	−0.4	−0.6		mA
I_{OL}	Low-level output current, ports F, P, K, and S	V_{OL}=0.5V		2	20	μA
V_{OH}	High-level output voltage, port LC	V_{CC}=5V, V_{LCD}=0.8V, Ta=25°C	3.75	3.95		V
V_{OH}	High-level output voltage, port COM	V_{CC}=5V, V_{LCD}=0.8V, Ta=25°C	4.8	5		V
V_{OX}	Medium-level output voltage, port COM (Note 1)	V_{CC}=5V, V_{LCD}=0.8V, Ta=25°C	2.7	2.9	3.1	V
V_{OL}	Low-level output voltage, port LC	V_{CC}=5V, V_{LCD}=0.8V, Ta=25°C		1.85	2.05	V
V_{OL}	Low-level output voltage, port COM	V_{CC}=5V, V_{LCD}=0.8V, Ta=25°C		0.8	1	V
I_{CC}	Supply current, full operating condition	V_{CC}=5V, Ta=25°C, Output pins open		0.7	1	mA
I_{CC}	Supply current, partial operating condition	V_{CC}=5V, Ta=25°C, Output pins open		200	300	μA
I_{LCD}	Liquid crystal supply current, full operating condition	V_{CC}−V_{LCD}=4.2V, Ta=25°C, Output pins open		60	120	μA
C_i	Input capacitance	V_{CC}=V_I=V_O=V_{SS}, f=1MHz, 25mVrms		7	10	pF
$C_{i(XIN)}$	Oscillator input capacitance	V_{CC}=X_{OUT}=V_{SS}, f=1MHz, 25mVrms		7	10	pF
V_{BD}	Battery voltage detection voltage range (Note 2)	10kΩ≤R_{BD}≤200kΩ, Ta=25°C	4.5		5.5	V

Note 1. V_{OX} is the medium level of the 3-level output of port COM.
2. The detection resistance R_{BD} is connected between the V_{SS} and pin BDIN
3. Currents are taken to be positive when flowing into the IC with minimum and maximum values taken as absolute values.

TIMING REQUIREMENTS (Ta=0~50°C, V_{CC}=5V±10%, V_{SS}=0V, unless otherwise noted)

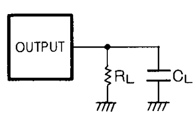
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{su(K-X_{IN})}$	Data setup time before clock input, port K inputs	f_{ϕ}=525kHz (Note 1)	0			μs
$t_{su(S-X_{IN})}$	Data setup time before clock input, port S inputs		0			μs
$t_{su(INT_A-X_{IN})}$	Data setup time before clock input, INT_A input		0			μs
$t_{su(INT_B-X_{IN})}$	Data setup time before clock input, INT_B input		0			μs
$t_{h(K-X_{IN})}$	Data hold time after clock input, port K inputs		0.4			μs
$t_{h(S-X_{IN})}$	Data hold time after clock input, port S inputs		0.4			μs
$t_{h(INT_A-X_{IN})}$	Data hold time after clock input, INT_A input		0.4			μs
$t_{h(INT_B-X_{IN})}$	Data hold time after clock input, INT_B input		0.4			μs

Note 1. $f\phi$=1/8 f_{XIN} which corresponds to the internal clock frequency.

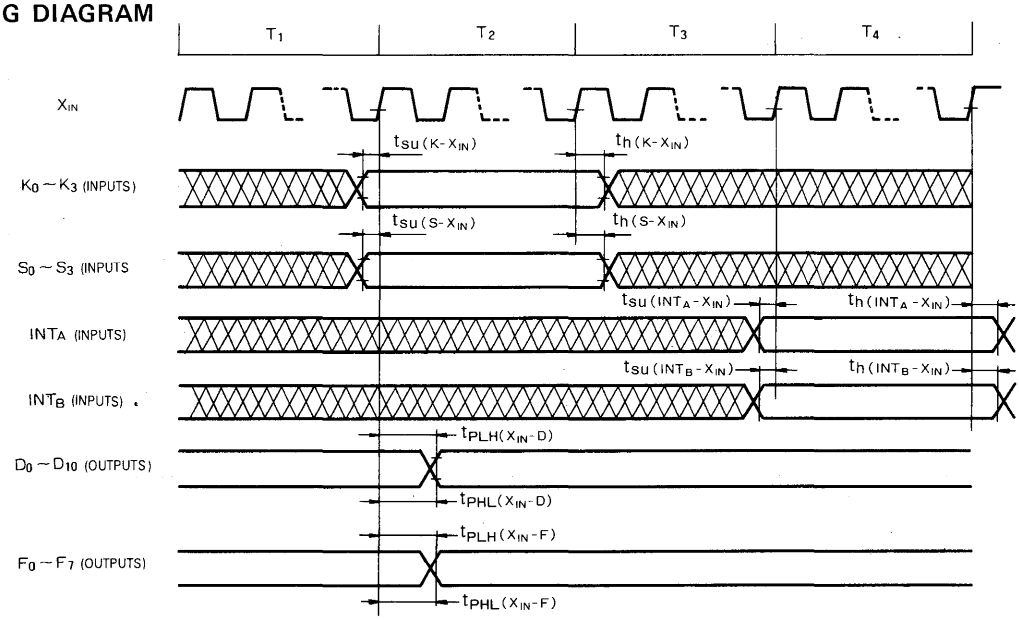
SWITCHING CHARACTERISTICS (Ta=0~50°C, V_{CC}=5V±10%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{PLH(X_{IN}-D)}$	Low-to-high-level propagation time from clock input to port data output, port D	f_{ϕ}=525kHz R_L=20kΩ C_L=100pF (Note 2)		0.7	1.5	μs
$t_{PLH(X_{IN}-F)}$	Low-to-high-level propagation time from clock input to port data output, port F, P, K, and S			0.7	1.5	μs
$t_{PHL(X_{IN}-D)}$	High-to-low-level propagation time from clock input to port data output, port D			2.2	3.0	μs
$t_{PHL(X_{IN}-F)}$	High-to-low-level propagation time from clock input to port data output, port F, P, K, and S			2.2	3.0	μs

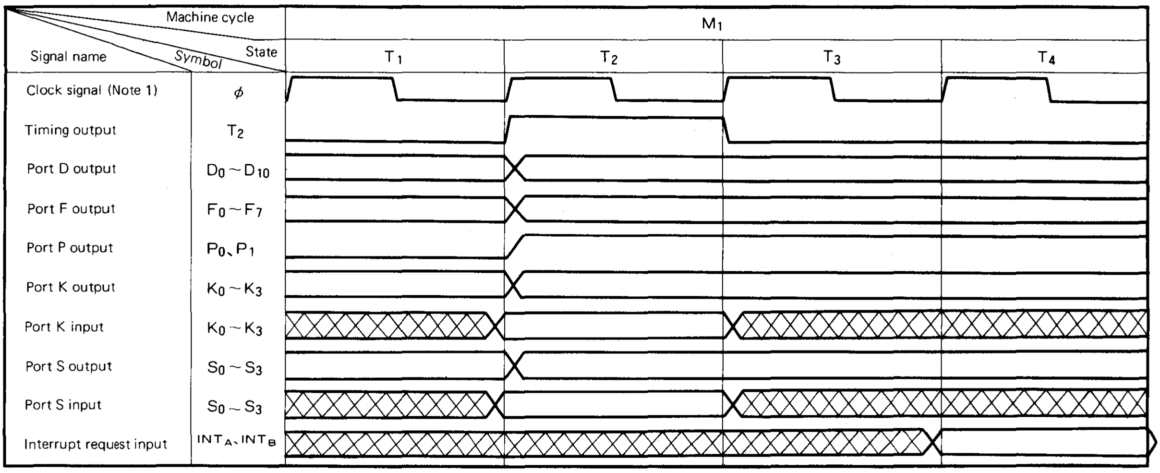
Note 2. Measurement circuit



TIMING DIAGRAM

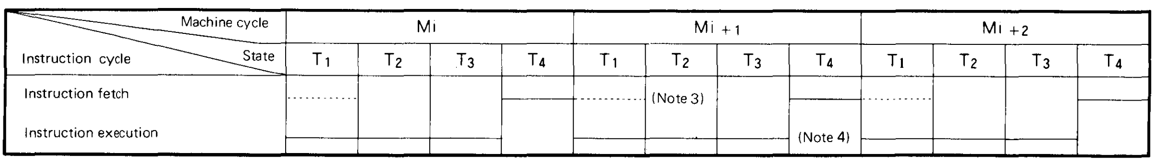


BASIC TIMING CHART (Note 2)



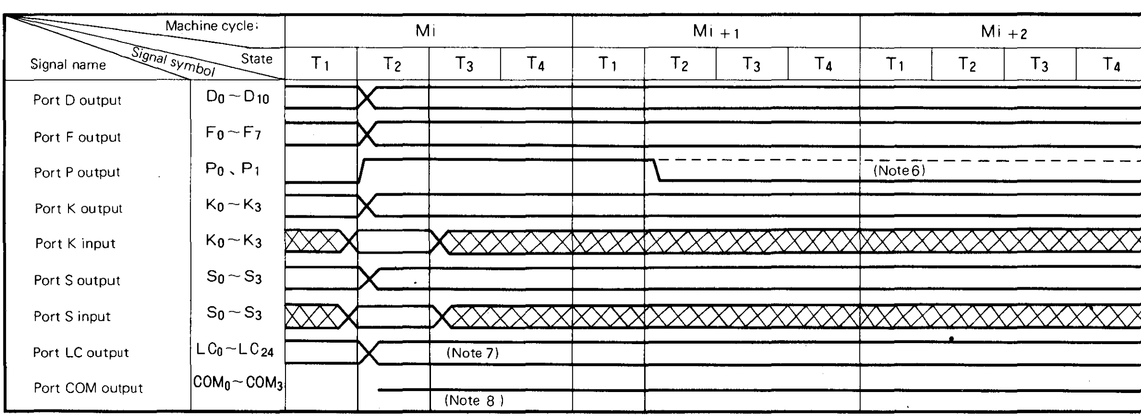
Note 1: Internal clock signal which is 1/8 of basic oscillation frequency.
2: ╳╳╳ indicates an invalid signal input.

INSTRUCTION FETCH TIMING



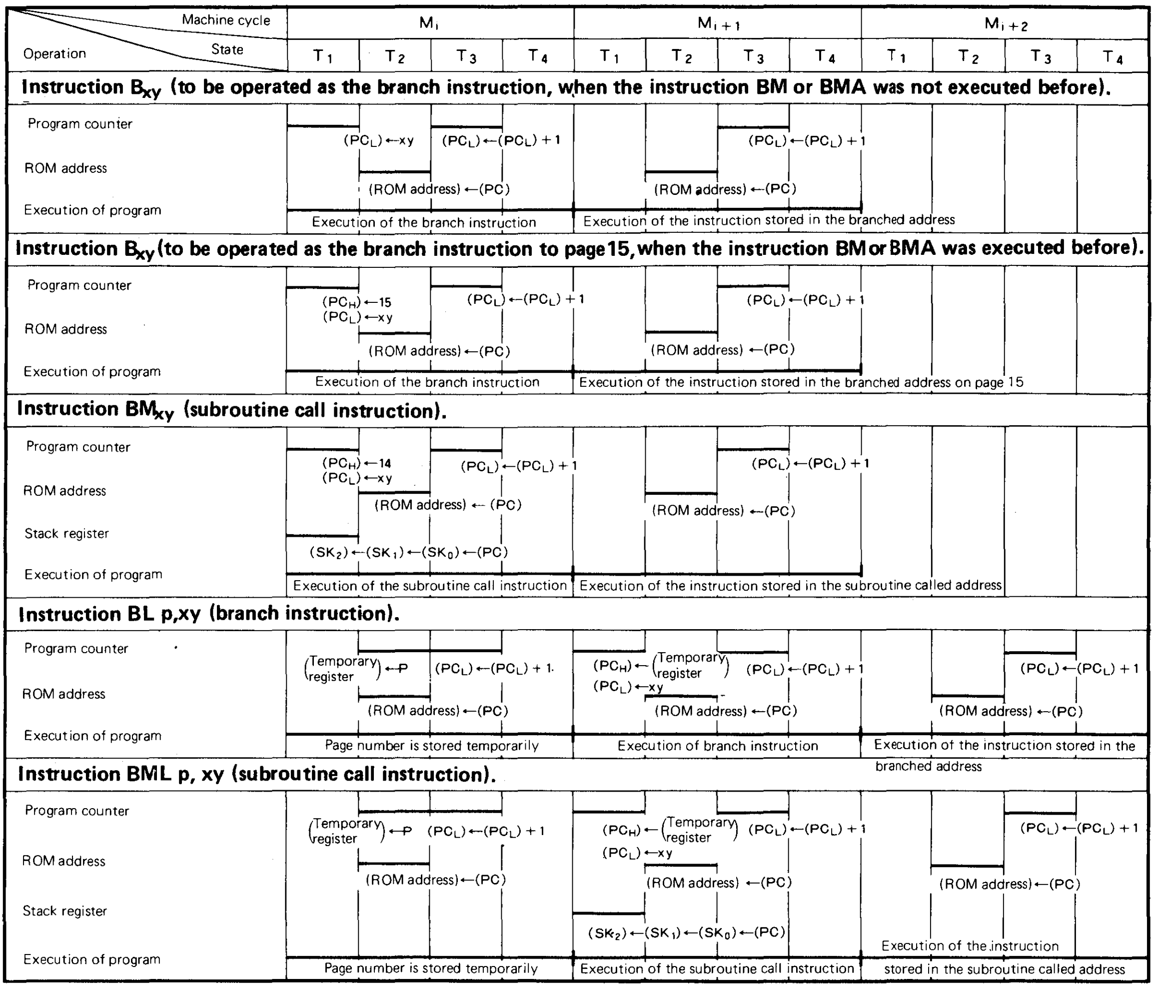
Note 3: Instruction fetch time can differ depending on the types of the instructions.
4: The instruction which was fetched in the preceding cycle is executed.
5: The execution of the instruction and addressing of ROM and RAM are performed simultaneously.

I/O INSTRUCTION EXECUTION TIMING



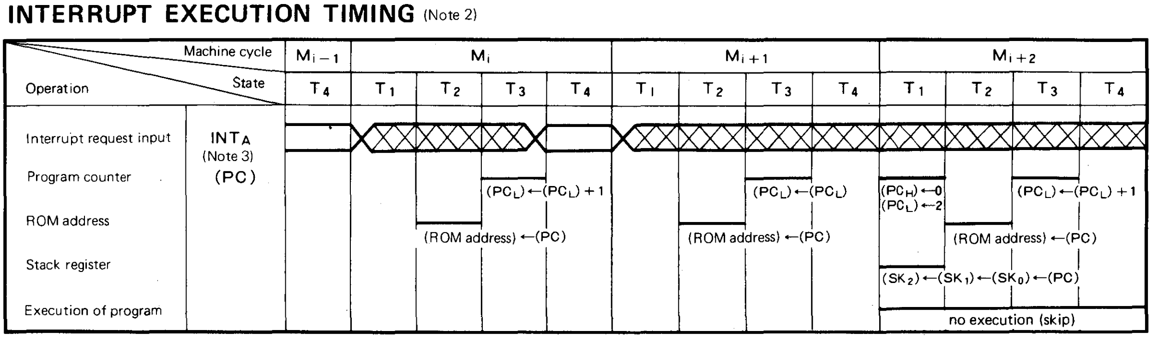
Note 6: When an OTRO instruction is executed, the output is latched.
7: Output voltage of port LC depends upon power supply V_{LCD} for the liquid crystal display.
8: Output voltage of port COM has 3 levels depending on the power supply V_{LCD} for the liquid crystal display.

BRANCH AND SUBROUTINE CALL INSTRUCTION EXECUTION TIMING (Note 1)



Note 1: The instructions BA, BMA, BLA and BMLA have the same execution timing as B, BM, BL and BML respectively as shown. The only difference is that $(PC_L) \leftarrow xy$ is replaced by $(PC_L) \leftarrow x(A)$.

INTERRUPT EXECUTION TIMING (Note 2)



Note 2: When the instruction executed in the machine cycle M_{i+1} is a BL, BML, BLA or BMLA, the value of address 2 of page 0 is stored in the program counter during M_{i+3}.

3: The interrupt request input INT_B has the same execution timing as INT_A. If the input is low level in the machine cycle M_{i-1} and high level in the machine cycle M_i, the interrupt is executed during the interrupt enable state.

SINGLE-CHIP 4-BIT CMOS MICROCOMPUTER

DESCRIPTION

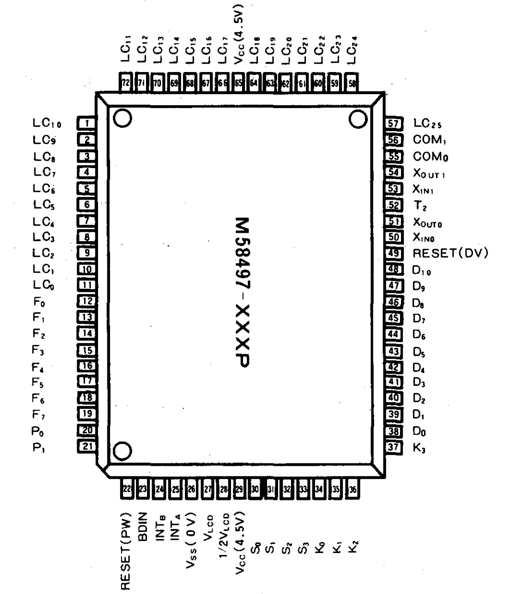
The M58497-XXXP is a single-chip 4-bit microcomputer fabricated using CMOS technology. Its features are liquid crystal display direct drive circuit, current saving circuit for back-up of a 15-stage frequency divider and RAM.

The device is designed for applications in which clock and liquid crystal display functions are included and where the low-power dissipation achieved by CMOS is especially important.

FEATURES

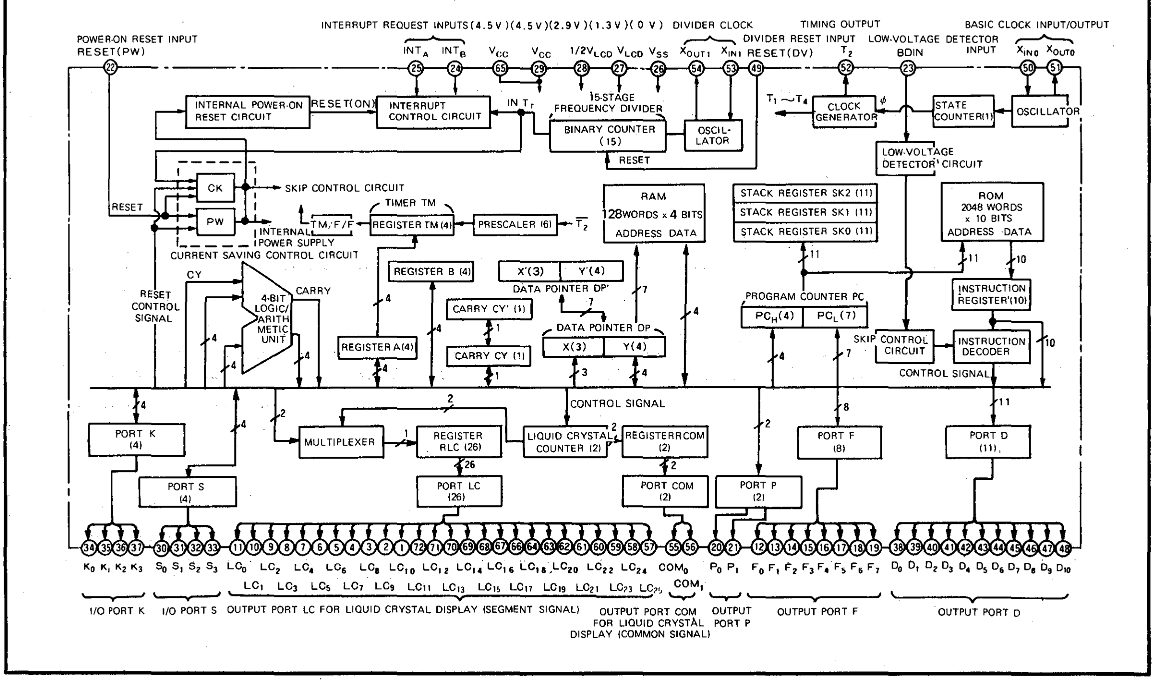
- Basic machine instructions 77
- Instruction execution time (at an oscillation frequency of 455kHz) 17.6μs
- Memory capacity: ROM: ...... 2048 words x 10 bits
  Internal RAM: .. 128 words x 4 bits
  External RAM: .. 256 words x 4 bits
- Single 4.5V power supply
- Internal oscillator circuit
- Internal 15-stage frequency divider
- Internal current saving cirucit
- Internal low-voltage detector circuit
- Subroutine nesting .... 3 levels
- Internal timer: Prescaler: .... 6 bits
  Timer: .... 4 bits
- Output ports for liquid crystal display
  Segment signals (port LC) 26 bits
  Common signals (port COM) 2 bits
- I/O ports (ports K and S) .... 2 x 4 bits
- Output port (port D) .... 11 x 1 bit
- Output port (port F) .... 8 x 1 bit
- Output port (port P) .... 2 x 1 bit
- Interrupt function .... 4 factors, 1 level

PIN CONFIGURATION (TOP VIEW)



Package Outline 72P2

BLOCK DIAGRAM



APPLICATIONS

- Electronic tuners for radios and TVs
- Medical equipment
- Measurement instruments
- Vending machines

FUNCTION

The M58497-XXXP consists of a 2,048 word x 10-bit mask ROM, 128 word x 4-bit RAM, 4-bit arithmetic logic unit, oscillator circuit, 15-stage frequency divider, power saving backup circuit for the RAM memory, low-voltage detector circuit, 4-bit timer, interrupt circuit, and liquid crystal display direct drive circuit. By connecting external 256-word x 4-bit CMOS RAMs to this 4-bit microcomputer, RAM capacity can be easily expanded.

The ROM is capable of storing 16 pages of 128 words of program, addresses being specified by the program counter. The program counter consists of a 7-bit address designating counter and a 4-bit page designating counter. Wrap-around to address zero is automatic after exceeding the address 127. The return address from subroutines and interrupts is stored in a stack register of 11 bits x 3 levels.

When an interrupt request has occurred, control is transferred to a fixed address as follows. If the case of internal power-on reset (RESET(ON)), the program is set to page 0, address 0, for the INT_A signal it is set to page 0, address 2, for the INT_B signal it is set to page 0, address 4, and for the output signal INT_T (1 second signal) of the 15-stage frequency divider, it is set to page 0, address 8.

The internal RAM is configured as 8 files of 16 words, addressed by the 7-bit data pointer. A 16 file x 16 digit external expansion memory can be addressed using 8 bits of address composed of the 4-bit register Y and 4-bit register B.

RAM addressing, register-to-register transfers, RAM-accumulator transfers, arithmetic operations, input/output operations, and timer operations are performed chiefly through the 4-bit register A (accumulator).

The current saving circuit used in conjunction with the 15-stage frequency divider and RAM can be controlled by the RESET (PW) input and program instructions.

The low-voltage detector circuit is operative when using a battery power source, and can be program controlled to provide an appropriate output upon sensing a low voltage level.

5

Direct drive of a liquid crystal display is possible using the 26 LC pins and 2 COM pins. 1/2 duty cycle and 1/2 bias or static drive is possible.

The output port D consists of 11 individually latched bits, and in addition to the ability to output a single bit, the position of which is determined by the contents of the data point register Y, 8 bits of the port D can be used as the external RAM address signal output.

Output port F consists of 8 individually latched bits that can be used to output data. It can be set or reset using program instructions.

PERFORMANCE SPECIFICATIONS

Parameter			Performance
Number of basic instructions			77
Execution time of basic instructions (one-word instruction)			17.6μs (V_{CC}=4.5V, f=455kHz)
Clock frequency			120~260kHz
Memory capacity	ROM		2,048 words x 10 bits
	Internal RAM		128 words x 4 bits
	External RAM		256 words x 4 bits
I/O ports	LC	Liquid crystal display output	26 bits
	COM	Liquid crystal display output	2 bits
	K	Input	4 bits
		Output	4 bits
	S	Input	4 bits
		Output	4 bits
	D	Output	11 x 1 bit
	F	Output	8 x 1 bit
	P	Output	2 x 1 bit
Frequency divider			15-stage built-in divider
Current saving circuit			Built-in
Low-voltage detector circuit			Built-in
Subroutine nesting			3 levels (including 1 level of interrupt)
Interrupt requests			4 factors, 1 level
Clock generator circuit			Built-in (for use with 455kHz ceramic filter externally connected)
Input/output ports	Output voltage		6V (max)
	Output current (I_{OL})		1.8mA (min)
Supply voltages	V_{CC}		4.5V (nom)
	V_{SS}		0V
Liquid crystal display driving voltage V_{LCD}			1.3V (nom)
Element structure			CMOS
Package			72-pin plastic molded flat package
Power dissipation (open output terminals)	In full operation		2mW (nom) (V_{CC}=4.5V, f=455kHz)
	Divider and RAM in the idle state		90μW (nom) (V_{CC}=4.5V, f=455kHz)

Output port P consists of 2 pins through which a synchronous signal of one machine cycle width can be output by program instructions.

Seven bits of output port F and P can be used to directly fetch the immediate field of the ROM.

The I/O ports K and S consist of 4 bits through which data can be transferred to and from register A.

PIN DESCRIPTIONS

Pin	Name	Input or output	State at reset (internal power-on)	Function
X_{IN0}	Clock oscillator input	Input	—	These are the clock input and output pins. A ceramic element or other frequency-determing element is connected between pins X_{IN0} and X_{OUT0}. When an external clock is used, connect the clock source to the X_{IN0} pin and leave the X_{OUT0} pin open.
X_{OUT0}	Clock oscillator output	Output	—	
X_{IN1}	Divider clock input	Input	—	These are the divider clock input and output pins. The quartz crystal that determines the reference oscillation frequency is connected between pins X_{IN1} and X_{OUT1}.
X_{OUT1}	Divider clock output	Output	—	
$RESET_{(DV)}$	Divider reset input	Input	—	This is the divider reset input for the 15-stage divider circuit used to divide the 32k Hz crystal reference signal.
BDIN	Low-voltage detector input	Input	—	A built-in low-voltage detector circuit has been provided. A resistor should be connected to the BDIN pin for voltage sensing.
INT_A	Interrupt request A signal	Input	Interrupt disable	These input signals are for an interrupt request. The request is accepted on the rising edge of the signal. In addition to these external input signals, an interrupt request INT_T from the 15-stage frequency divider output signal is also treated as an interrupt.
INT_B	Interrupt request B signal	Input	Interrupt disable	
$LC_0 \sim LC_{25}$	Liquid crystal display segment outputs	Output	—	These liquid crystal display outputs are suitable for driving a liquid crystal display at 1/2 duty cycle and 1/2 bias. The output ports for direct drive of such liquid crystal displays are port LC ($LC_0 \sim LC_{25}$) for the segments and port COM ($COM_0 \sim COM_1$) for the common outputs.
$COM_0 \sim COM_1$	Liquid crystal display common outputs	Output	—	
V_{LCD}, $\frac{1}{2}V_{LCD}$	Power supply for liquid crystal display	—	—	These are the liquid crystal display power supply pins for segment signals and common signals.
$D_0 \sim D_{10}$	Output port D	Output	High level	This output port consists of 11 bits, each of which is individually latched and can be selected to be set or reset according to the contents of register Y. In addition, 8 bits of this port can be used to fetch an 8-bit address for external RAM.
$F_0 \sim F_7$	Output port F	Output	High level	This output port consists of 8 bits, each of which is individually latched and can be set or reset using machine instructions.
P_0, P_1	Output port P	Output	High level	This output port consists of 2 bits from which one synchronous signal of one machine cycle width can be output per instruction. The 7-bit immediate field of an instruction can be output through this port in combination with 5 bits of port F.
$K_0 \sim K_3$	Input/output port K	Input/output	High level	Ports K and S are 4-bit latched input/output ports through which data can be transferred to and from register A (accumulator). When the output is programmed high, the high-impedance state is enabled allowing use of the pins as input pins.
$S_0 \sim S_3$	Input/output port S	Input/output	High level	
T_2	Timing output	Output	—	This timing output is used for testing the device.
$RESET_{(PW)}$	Power-on reset input	Input	Low level	When the internal power supply is switched on, a built-in automatic reset circuit generates a high level reset signal that resets the I/O ports and starts the system.

DESCRIPTION OF BASIC FUNCTIONAL BLOCKS

Program Counter PC

The program counter is an 11-bit address register. The 4 high-order bits are used to designate the page number and are as a group called PC_H. The 7 low-order bits are used to designate the address on the page and as a group are called PC_L. The PC designates the address of the 2048 words by 10-bit mask-programmable ROM. The ROM is organized into 16 pages of 128 words. As instructions are fetched from ROM, PC_L is incremented, so that, unless there is a branch, executed instructions are fetched and executed in sequence. Care must be taken when the last instruction on a page (address 127) is executed, because when PC_L is incremented it becomes zero with a carry, but the carry is disregarded so that the next instruction to be fetched will be the instruction at the first address of the same page. Therefore, to move to the next page, PC_H must be modified by using branch instructions such as BL, BML, BLA, and BMLA.

Pages 14 and 15 are special pages set aside to accommodate subroutines. Page 14 can be used to store subroutines, which are callable from pages other than page 14 by using the instructions BM and BMA which can be used as single-instructions to call page 14 subroutines.

When BM or BMA instructions are executed within page 14, they are equivalent to the branch instructions B and BA. When B or BA instructions are executed within page 14, a branch to the specified address on page 15 is executed.

Stack Registers SK_0, SK_1, SK_2

The 3-level stack register consists of 11-bit registers for storing the contents of the program counter when control is transferred from the main program to a subroutine or interrupt. Subroutines can use 3 levels, so that when 1 level is used for an interrupt routine 2 levels are reserved for subroutine nesting.

Data Pointers DP, DP'

The data pointer is used to designate the address of RAM or the bit position of output port D and consists of the 3-bit register X and the 4-bit register Y. The internal RAM is organized as 8 files of 16 words. Register X designates the file and register Y designates the word position of a file or the bit position of the output port D.

The data pointer DP' is selected by software during interrupt processing, leaving the contents of DP saved (unchanged).

The external RAM memory is organized as 16 files of 16 words that can be added to the system to expand memory capacity. Register Y is used to designate the word position of a file while register B designates the file itself.

Register A (Accumulator) and Carry Flags, CY, CY'

Register A is the 4-bit accumulator which forms the heart of the 4-bit microcomputer. Data processing operations such as arithmetic, transfer, exchange, conversion, and input/output operations are executed basically through this register.

The carry flag CY are used to store the carry or borrow from the most significant bit of the arithmetic unit resulting from the execution of various instructions. It can be tested and used for a variety of purposes. In principle, it acts as a 1-bit flag.

The carry flag CY' is used during interrupt processing to save the contents of the carry flag CY.

5

Register B (Auxiliary Register)

This register consists of a 4-bit register used for temporary storage as well as designate the file number of external RAM.

4-Bit Arithmetic Logic Unit

This unit is used to perform 4-bit arithmetic and logical operations and consists of a 4-bit adder and the associated logic circuitry. It is used to perform operations such as additions, complementing, logical and arithmetic comparisons, and bit manipulation.

Frequency Divider and Timer

A 15-stage frequency divider is used to divide the basic oscillator frequency. It is connected to the oscillator source device through pins X_{IN1} and X_{OUT1}. The frequency divider generates the interrupt request signal INT_T which is input to the interrupt control circuit. It also sets the CK flag for controlling the power saving circuit.

The basic oscillator circuit for the timer is the timing signal T_2. The timer consists of a 6-bit prescaler and a 4-bit counter. The timer flags TMF/F are set when a timer overflow occurs and sensed by the TTM instruction. The 4-bit timer counter is set by the STM instruction. Prescaler and timer flags are reset at the same time.

Power Saving Circuit

The power saving circuit is controlled by the CK flag and PW. Its output is sent to the built-in power supply reset circuit and causes the generation of an interrupt request signal RESET(ON). Control is unconditionally transferred to address 0 on page 0 and results in the resetting of I/O ports. The interrupt request signal RESET(ON) is generated on the rising edge of the built-in power supply reset output. The built-in power supply may be switched off by means of either an external signal or stop instruction, but power is maintained to the following circuits:

1. Internal data memory (RAM)

2. Divider Clock oscillator circuit
3. 15-stage frequency divider
4. Low-voltage detector circuit
5. Power saving circuit

Low-Voltage Detector Circuit

The low-voltage detector circuit is effective when using the M58497-XXXP with a battery power supply. The resistor which is used to determine the low sensing voltage is connected to the BDIN pin. A voltage falling below this level is sensed by the program and can be displayed by using an output port.

Interrupt Functions

Four factors in one level of hardware interrupt functions have been provided. The four interrupt request sources consist of the external interrupt requests INT_A and INT_B, the internal power-on reset output RESET(ON), and the 15-stage frequency divider output INT_T. Interrupt is enabled by the instructions EIA, EIB, and EIT, and disabled by the DIA, DIB, and DIT instructions respectively. Interrupt requests generated by the internal power supply by means of the reset output RESET(ON) cannot be disabled and will cause an unconditional hardware initialization whenever received.

In the interrupt enable state, interrupt requests INT_A and INT_B are accepted on the rising edge of these signals. When an interrupt request is received when interrupt is disabled, interrupt processing does not occur but the interrupt request is stored in a latch so that when the interrupt disable condition is cancelled the appropriate interrupt enabling instruction can be used to execute the interrupt routine immediately.

One level of the 3-level stack register is required when using an interrupt program. This leaves the remaining two levels available for subroutine processing. After an interrupt is processed, control is returned to the main program by means of an instruction such as RTI. Care must be taken, however, after starting an interrupt program to save the content of data pointer DP, register A, carry flag CY, and any other registers used by the interrupt program so that the contents may be restored before returning to the main program.

When an interrupt has been accepted, the microcomputer internal states are as follows.

(1) Program counter
The main program current address is stored in the stack register. Control is transferred to address 0 page 0 by a RESET(ON) interrupt, to address 2 page 0 by an INT_A interrupt to address 4 page 0 by an INT_B interrupt, and to address 8 page 0 by an INT_T interrupt. Note, however, that for the RESET(ON) signal the instruction on address 0 page 0 is invalid.

(2) Interrupt Enable Flags
If any of the four available interrupt factors are executed, all the interrupt enable flags are reset and the interrupt disable state is entered.

(3) Skip flags
Skip flags have been provided to indicate skip conditions for skip or continuous skip instructions. These are provided for all stacks, the stack flags being saved and the skip condition for an interrupt being held in memory.

General-Purpose I/O Ports K, S, F, P and D

These 4-bit and 1-bit general-purpose registers are used for such operations as data transfers to and from register A, instruction transfers, 1-bit transfers as selected by register Y, storage of the 7-bit immediate filed of instructions fetched from ROM, and data transfers between external RAM. Each output circuit is a latched CMOS circuit.

Input/output ports K and S are 4-bit ports, capable of data transfer with register A. When used as input ports, the DIKS instruction is used to reset the output latches.

The output port F consists of an 8-bit port with each bit independently latched. Each bit is settable and resettable by means of the SF and RF instructions respectively.

Output port P is a 2-bit port which is normally at the high level. The instructions SP_0 and SP_1 can be used to generate a low-level synchronous signal for one machine cycle.

Seven bits of the output ports F and P can be used to directly fetch the ROM immediate field value (7 bits) by means of the OTRO instruction.

The output port D consists of 11 bits independently latched. The contents of register Y indicate the individual bit to be set by the SD instruction or to be reset by the RD instruction. The 8-bit address of external memory (RAM) is output by means of this port.

Liquid Crystal Display Drive Circuit

The liquid Crystal display direct drive circuit consists of the following units. A block diagram of these units is shown in Fig. 1.

(1) Liquid crystal display control counter
This 2-bit quaternary counter counts down under control of the ELC instruction. The contents of this counter select 1 bit of register A and transfer data sequentially to the segment register RLC by a TLC instruction while determining the frame frequency by means of transferring the contents of the counter to the common register RCOM.

(2) Register A
This 4-bit register serves as an accumulator. Its func-

tions include microcomputer processing, control, and central processing for input and output.

(3) Segment Register RLC
This 26-bit serial register is used to store selected single data bit from register A by means of the TLC instruction. It shifts single bit in order and temporarily stores the segment signals for the liquid crystal display.

(4) Common Register RCOM
This 2-bit register is used to convert the contents of the liquid crystal display control counter to the common signals required for the display.

(5) Port LC
This 26-bit latched port is used to store data in parallel by means of the ELC or DLC instructions from the segment register RLC. It provides two levels of bias, the liquid crystal drive voltage V_{LCD} and the supply voltage V_{CC}.

(6) Port COM
Port COM consists of 2 latched bits used for parallel storage of data transferred from the common register RCOM by the ELC and DLC instructions. It provides 3 levels of bias including liquid crystal drive voltage (V_{LCD}, 1/2 V_{LCD}) and the supply voltage V_{CC}.

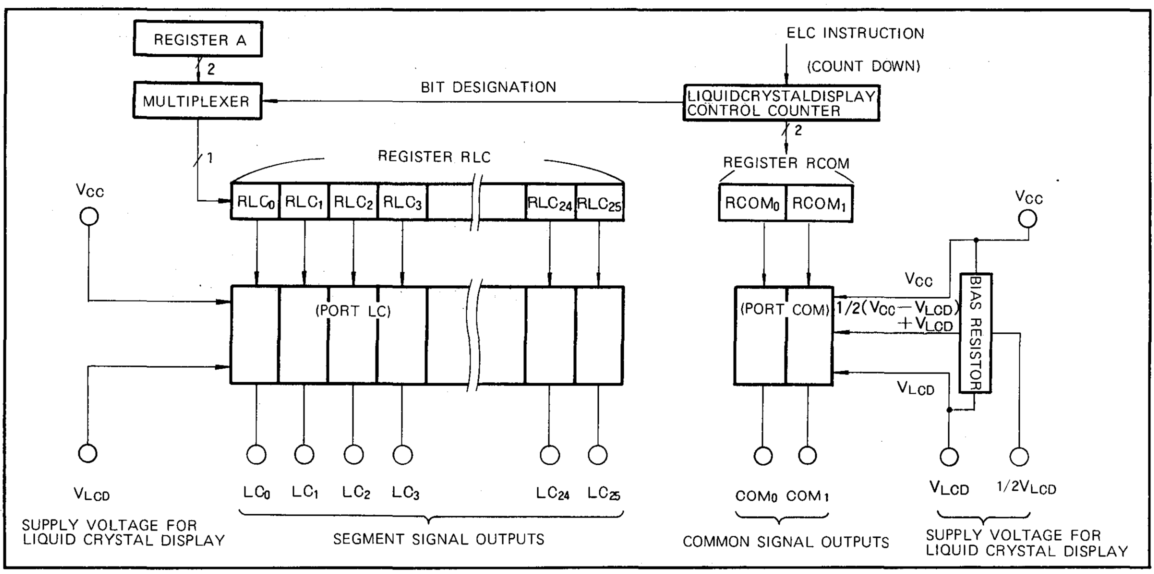


Fig. 1 Liquid crystal display drive circuit block diagram

Reset Function

As shown in Fig. 2, when a low level of at least 10ms is applied to the M58497-XXXP RESET(PW) input pin, all input/output ports are reset and interrupt is disabled. (Refer to the section on Power-on Reset States in the pin descriptions.) Next, when the RESET(PW) input is set to high, the internal power-on reset output RESET(ON) causes the generation of an interrupt and the program coutner is set to address 0 on page 0 as the starting address.

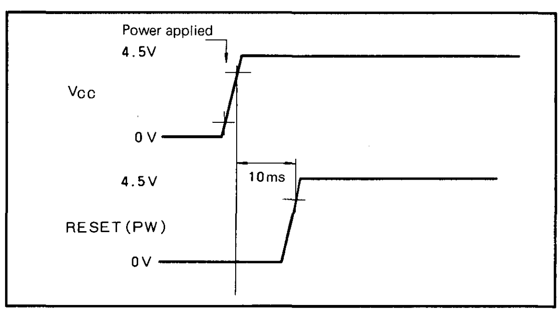


Fig. 2 Power-on reset circuit

Clock Generator Circuit

A built-in clock generator circuit has been provided for use with a ceramic element connected between the clock input and output pins. In addition, an external clock source may be input at pin X_{IN0}, leaving X_{OUT0} open. Circuit examples are shown in Fig. 3 and Fig. 4.

Documentation Required Upon Ordering

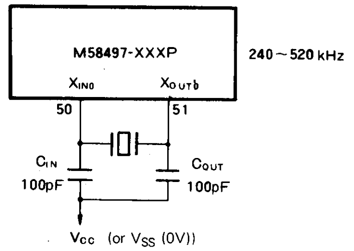
The following information should be provided when ordering a custom mask.

(1) M58497-XXXP mask confirmation sheet

(2) ROM data .... 3 EPROM sets

(3) Oscillation frequency selection
On confirmation sheets

(4) Frequency divider output selection (1Hz/2Hz)
On confirmation sheets



1) Verification of oscillation should be made at the timing output T_2. In addition, the required circuit constants will vary with the frequency and type of ceramic element used.

Fig. 3 Externally connected ceramic filter

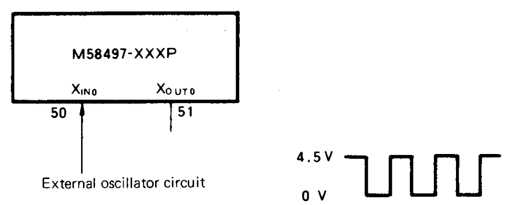


Fig. 4 External clock input circuit

INSTRUCTION CODE LIST (Note 1)

$D_9 \sim D_4$ / $D_3 \sim D_0$	Hexadecimal notation	00 0000 00	00 0001 01	00 0010 02	00 0011 03	00 0100 04	00 0101 05	00 0110 06	00 0111 07	00 1000 08	00 1001 09	00 1010 0A	00 1011 0B	00 1100 0C	00 1101 0D	00 1110 ~ 00 1111 0E~0F	01 0000 ~ 01 0111 10~17	01 1000 ~ 01 1111 18~1F	10 0000 ~ 10 0111 20~27	10 1000 ~ 10 1111 28~2F	11 0000 ~ 11 0111 30~37	11 1000 ~ 11 1111 38~3F
0000	0	NOP	TLC	INY	SZB 0	SEY 0	SEI 0	SF 0	BL BLA BML BMLA	–	RAR	TAM 0	XAMD 0	A 0	LA 0	–	OTRO	LXY	BM	BMA	B	BA
0001	1	SCOM	DIKS	DEY	SZB 1	SEY 1	SEI 1	SF 1	BL BLA BML BMLA	–	–	TAM 1	XAMD 1	A 1	LA 1	–	OTRO	LXY	BM	BMA	B	BA
0010	2	EIA	SFK	XDP	SZB 2	SEY 2	SEI 2	SF 2	BL BLA BML BMLA	*	IK	TAM 2	XAMD 2	A 2	LA 2	–	OTRO	LXY	BM	BMA	B	BA
0011	3	DIA	SFS	TYA	SZB 3	SEY 3	SEI 3	SF 3	BL BLA BML BMLA	SEAM	IS	TAM 3	XAMD 3	A 3	LA 3	–	OTRO	LXY	BM	BMA	B	BA
0100	4	EIB	*	SC	RT	SEY 4	SEI 4	SF 4	BL BLA BML BMLA	*	TBA	TAM 4	XAMD 4	A 4	LA 4	–	OTRO	LXY	BM	BMA	B	BA
0101	5	DIB	DLC	RC	RTS	SEY 5	SEI 5	SF 5	BL BLA BML BMLA	TAY	–	TAM 5	XAMD 5	A 5	LA 5	–	OTRO	LXY	BM	BMA	B	BA
0110	6	DETS	*	XC	RTI	SEY 6	SEI 6	SF 6	BL BLA BML BMLA	AND	XAB	TAM 6	XAMD 6	A 6	LA 6	–	OTRO	LXY	BM	BMA	B	BA
0111	7	DETR	ELC	*	*	SEY 7	SEI 7	SF 7	BL BLA BML BMLA	EXL	TAB	TAM 7	XAMD 7	A 7	LA 7	–	OTRO	LXY	BM	BMA	B	BA
1000	8	EIT	SP0	*	*	SEY 8	SEI 8	RF 0	BL BLA BML BMLA	*	SB 0	XAM 0	XAMI 0	A 8	LA 8	–	OTRO	LXY	BM	BMA	B	BA
1001	9	DIT	*	SD	*	SEY 9	SEI 9	RF 1	BL BLA BML BMLA	CMA	SB 1	XAM 1	XAMI 1	A 9	LA 9	–	OTRO	LXY	BM	BMA	B	BA
1010	A	STM	SP1	*	*	SEY 10	SEI 10	RF 2	BL BLA BML BMLA	AM	SB 2	XAM 2	XAMI 2	A 10	LA 10	–	OTRO	LXY	BM	BMA	B	BA
1011	B	POF2	*	*	*	SEY 11	SEI 11	RF 3	BL BLA BML BMLA	*	SB 3	XAM 3	XAMI 3	A 11	LA 11	–	OTRO	LXY	BM	BMA	B	BA
1100	C	POF1	OTAD	*	*	SEY 12	SEI 12	RF 4	BL BLA BML BMLA	*	RB 0	XAM 4	XAMI 4	A 12	LA 12	–	OTRO	LXY	BM	BMA	B	BA
1101	D	SDET	*	RD	*	SEY 13	SEI 13	RF 5	BL BLA BML BMLA	*	RB 1	XAM 5	XAMI 5	A 13	LA 13	–	OTRO	LXY	BM	BMA	B	BA
1110	E	TTM	ADRT	*	*	SEY 14	SEI 14	RF 6	BL BLA BML BMLA	AMC	RB 2	XAM 6	XAMI 6	A 14	LA 14	–	OTRO	LXY	BM	BMA	B	BA
1111	F	TCK	TPW	*	SZC	SEY 15	SEI 15	RF 7	BL BLA BML BMLA	AMCS	RB 3	XAM 7	XAMI 7	A 15	LA 15	–	OTRO	LXY	BM	BMA	B	BA

Note 1: This list shows the machine codes and corresponding machine instructions. $D_3 \sim D_0$ indicate the low-order 4 bits of the machine code and $D_9 \sim D_4$ indicate the high-order 6 bits. Hexadecimal numbers are also shown that represent the codes. An instruction may consist of one or two words, but only the first word is listed. Code combinations indicated with asterisk (*) and bar (–) must not be used.

Two-word instructions

	Second word
BL	11 0xxxx yyyy
BLA	11 1xxxx XXXX
BML	10 0xxxx yyyy
BMLA	10 1xxxx XXXX

5

MACHINE INSTRUCTIONS (Note 1)

Classification	Mnemonic	D9D8	D7D6D5D4	D3D2D1D0	Hexadecimal	No. of words	No. of cycles	Functions	Skip conditions	Flag CY
RAM address	LXY x,y	0 1	1 x x x	y y y y	18y + x	1	1	(X)←x, where, x=0~7 (Y)←y, where, y=0~15	Consecutively described	×
	INY	0 0	0 0 1 0	0 0 0 0	020	1	1	(Y)←(Y)+1	—	×
	DEY	0 0	0 0 1 0	0 0 0 1	021	1	1	(Y)←(Y)−1	—	×
Register to register transfer	TAB	0 0	1 0 0 1	0 1 1 1	097	1	1	(A)←(B)	—	×
	TBA	0 0	1 0 0 1	0 1 0 0	094	1	1	(B)←(A)	—	×
	XAB	0 0	1 0 0 1	0 1 1 0	096	1	1	(A)↔(B)	—	×
	TAY	0 0	1 0 0 0	0 1 0 1	085	1	1	(A)←(Y)	—	×
	TYA	0 0	0 0 1 0	0 0 1 1	023	1	1	(Y)←(A)	—	×
	XDP	0 0	0 0 1 0	0 0 1 0	022	1	1	(DP)↔(DP')	—	×
RAM to accumulator transfer	TAM j	0 0	1 0 1 0	0 j j j	0Aj	1	1	(A)←(M(DP)) (X)←(X)∀j, where, j=0~7	—	×
	XAM j	0 0	1 0 1 0	1 j j j	0A8 + j	1	1	(A)↔(M(DP)) (X)←(X)∀j, where, j=0~7	—	×
	XAMD j	0 0	1 0 1 1	0 j j j	0Bj	1	1	(A)↔(M(DP)), (Y)←(Y)−1 (X)←(X)∀j, where, j=0~7	(Y)=15	×
	XAMI j	0 0	1 0 1 1	1 j j j	0B8 + j	1	1	(A)↔(M(DP)), (Y)←(Y)+1 (X)←(X)∀j, where, j=0~7	(Y)=0	×
Arithmetic	LA n	0 0	1 1 0 1	n n n n	0Dn	1	1	(A)←n, where, n=0~15	Consecutively described	×
	AM	0 0	1 0 0 0	1 0 1 0	08A	1	1	(A)←(A)+(M(DP))	—	×
	AMC	0 0	1 0 0 0	1 1 1 0	08E	1	1	(A)←(A)+(M(DP))+(CY) (CY)← Carry	—	0/1
	AMCS	0 0	1 0 0 0	1 1 1 1	08F	1	1	(A)←(A)+(M(DP))+(CY) (CY)← Carry	(CY)=0	0/1
	A n	0 0	1 1 0 0	n n n n	0Cn	1	1	(A)←(A)+n, where, n=0~15	Carry = 0	×
	SC	0 0	0 0 1 0	0 1 0 0	024	1	1	(CY)←1	—	1
	RC	0 0	0 0 1 0	0 1 0 1	025	1	1	(CY)←0	—	0
	XC	0 0	0 0 1 0	0 1 1 0	026	1	1	(CY)↔(CY')	—	(CY')
	SZC	0 0	0 0 1 1	1 1 1 1	03F	1	1	Skip if (CY)=0	(CY)=0	×
	AND	0 0	1 0 0 0	0 1 1 0	086	1	1	(A)←(A)∧(M(DP))	—	×
	EXL	0 0	1 0 0 0	0 1 1 1	087	1	1	(A)←(A)∀(M(DP))	—	×
	CMA	0 0	1 0 0 0	1 0 0 1	089	1	1	$(A)\leftarrow(\overline{A})$	—	×
	RAR	0 0	1 0 0 1	0 0 0 0	090	1	1	$(A_{n-1})\leftarrow(A_n)$ $(CY)\leftarrow(A_0)$, $(A_3)\leftarrow(CY)$	—	(A_0)
Bit manipulation	SB i	0 0	1 0 0 1	1 0 i i	098 + i	1	1	(Mi(DP))←1, where, i=0~3	—	×
	RB i	0 0	1 0 0 1	1 1 i i	09C + i	1	1	(Mi(DP))←0, where, i=0~3	—	×
	SZB i	0 0	0 0 1 1	0 0 i i	03i	1	1	Skip if (Mi(DP))=0, where, i=0~3	(Mi(DP))=0 where, i=0~3	×
Compare	SEAM	0 0	1 0 0 0	0 0 1 1	083	1	1	Skip if (M(DP))=(A)	(M(DP))=(A)	×
	SEY y	0 0	0 1 0 0	y y y y	04y	1	1	Skip if (Y)=y, where, y=0~15	(Y)=y, where, y=0~15	×
	SEI n	0 0	0 1 0 1	n n n n	05n	1	1	Skip if (A)=n, where, n=0~15	(A)=n, where, n=0~15	×
	SCOM	0 0	0 0 0 0	0 0 0 1	001	1	1	Skip if (SCA=0)	SCA=0	×

Classification	Mnemonic	D_9 D_8	D_7 D_6 D_5 D_4	D_3 D_2 D_1 D_0	Hexadecimal	No. of words	No. of cycles	Functions	Skip conditions	Flag CY
Branch	B xy	1 1	0 x x x	y y y y	3xy	1	1	$(PC_L) \leftarrow 16x+y$	—	×
	(Note 2)							$(PC_H) \leftarrow 15$, $(PC_L) \leftarrow 16x+y$		
	BL pxy	0 0 1 1	0 1 1 1 0 x x x	p p p p y y y y	07p 3xy	2	2	$(PC_H) \leftarrow p$ $(PC_L) \leftarrow 16x+y$	—	×
	BA xX	1 1	1 x x x	X X X X	38X + x	1	1	$(PC_L) \leftarrow 16x+(A)$	—	×
	(Note 2)							$(PC_H) \leftarrow 15$, $(PC_L) \leftarrow 16x+(A)$		
	BLA pxX	0 0 1 1	0 1 1 1 1 x x x	P P P P X X X X	07p 38X + x	2	2	$(PC_H) \leftarrow p$ $(PC_L) \leftarrow 16x+(A)$	—	×
Subroutine call	BM xy	1 0	0 x x x	y y y y	2xy	1	1	$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$ $(PC_H) \leftarrow 14$, $(PC_L) \leftarrow 16x+y$	—	×
	(Note 2)							$(PC_H) \leftarrow 14$, $(PC_L) \leftarrow 16x+y$		
	BML pxy	0 0 1 0	0 1 1 1 0 x x x	p p p p y y y y	07p 2xy	2	2	$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$ $(PC_H) \leftarrow p$, $(PC_L) \leftarrow 16x+y$	—	×
	BMA xX	1 0	1 x x x	X X X X	28X + x	1	1	$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$ $(PC_H) \leftarrow 14$, $(PC_L) \leftarrow 16x+(A)$	—	×
	(Note 2)							$(PC_H) \leftarrow 14$, $(PC_L) \leftarrow 16x+(A)$		
	BMLA pxX	0 0 1 0	0 1 1 1 1 x x x	p p p p X X X X	07p 28X + x	2	2	$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$ $(PC_H) \leftarrow p$, $(PC_L) \leftarrow 16x+(A)$	—	×
Return	RTI	0 0	0 0 1 1	0 1 1 0	036	1	1	$(PC) \leftarrow (SK0) \leftarrow (SK1) \leftarrow (SK2)$ Restore interrupt skip flags	—	×
	RT	0 0	0 0 1 1	0 1 0 0	034	1	1	$(PC) \leftarrow (SK0) \leftarrow (SK1) \leftarrow (SK2)$	—	×
	RTS	0 0	0 0 1 1	0 1 0 1	035	1	1	$(PC) \leftarrow (SK0) \leftarrow (SK1) \leftarrow (SK2)$	Unconditional	×
Input/output	DIKS	0 0	0 0 0 1	0 0 0 1	011	1	1	$(K) \leftarrow 1$, $(S) \leftarrow 1$,	—	×
	IK	0 0	1 0 0 1	0 0 1 0	092	1	1	$(A) \leftarrow (K)$	—	×
	IS	0 0	1 0 0 1	0 0 1 1	093	1	1	$(A) \leftarrow (S)$	—	×
	SFK	0 0	0 0 0 1	0 0 1 0	012	1	1	$(K) \leftarrow (A)$	—	×
	SFS	0 0	0 0 0 1	0 0 1 1	013	1	1	$(S) \leftarrow (A)$	—	×
	SD	0 0	0 0 1 0	1 0 0 1	029	1	1	$(D(Y)) \leftarrow 1$, where, $0 \leqq (Y) \leqq 10$	—	×
	RD	0 0	0 0 1 0	1 1 0 1	02D	1	1	$(D(Y)) \leftarrow 0$, where, $0 \leqq (Y) \leqq 10$	—	×
	ADRT	0 0	0 0 0 1	1 1 1 0	01E	1	1	$(D) \leftarrow 1$	—	×
	OTAD	0 0	0 0 0 1	1 1 0 0	01C	1	1	$(D_7 \sim D_4) \leftarrow (B)$ $(D_3 \sim D_0) \leftarrow (Y)$	—	×
	SF m	0 0	0 1 1 0	0 m m m	06m	1	1	$(Fm) \leftarrow 1$, where, $m=0 \sim 7$	—	×
	RF m	0 0	0 1 1 0	1 m m m	068 + m	1	1	$(Fm) \leftarrow 0$, where, $m=0 \sim 7$	—	×
	OTRO mn	0 1	0 m m m	n n n n	1mn	1	1	$(F_0 \sim F_3) \leftarrow n$, where, $n=0 \sim 15$ $(F_4, P_0, P_1) \leftarrow m$, where, $m=0 \sim 7$	—	×
	SP0	0 0	0 0 0 1	1 0 0 0	018	1	1	$(P_0) \leftarrow 0$, output for 1 machine cycle only	—	×
	SP1	0 0	0 0 0 1	1 0 1 0	01A	1	1	$(P_1) \leftarrow 0$, output for 1 machine cycle only	—	×
	TLC	0 0	0 0 0 1	0 0 0 0	010	1	1	$(R(LC_0)) \leftarrow (Ai)$, where, $i=0 \sim 1$ $(R(LC_{n+1})) \leftarrow (R(LCn))$	—	× ×
	ELC	0 0	0 0 0 1	0 1 1 1	017	1	1	$(P(LCn)) \leftarrow (R(LCn))$ $(P(COMn)) \leftarrow (R(COMn))$	—	×
	DLC	0 0	0 0 0 1	0 1 0 1	015	1	1	$(P(LCn)) \leftarrow (R(LCn))$ $(P(COM)) \leftarrow \frac{1}{2}(V_{CC}-V_{LCD})+V_{LCD}$	—	×

Classification \ Item	Mnemonic	Instruction code D_9D_8	$D_7D_6D_5D_4$	$D_3D_2D_1D_0$	Hexadecimal	No. of words	No. of cycles	Functions	Skip conditions	Flag CY
Interrupt	EIA	00	0000	0010	002	1	1	Enables interruption of INT_A signal.	—	×
	DIA	00	0000	0011	003	1	1	Disables interruption of INT_A signal.	—	×
	EIB	00	0000	0100	004	1	1	Enables interruption of INT_B signal.	—	×
	DIB	00	0000	0101	005	1	1	Disables interruption of INT_B signal.	—	×
	EIT	00	0000	1000	008	1	1	Enables interruption of INT_T signal.	—	×
	DIT	00	0000	1001	009	1	1	Disables interruption of INT_T signal.	—	×
Timer	STM	00	0000	1010	00A	1	1	(TM)←(A), (TM F/F)←0 6-bit prescaler presetting	—	×
	TTM	00	0000	1110	00E	1	1	Skip if (TM F/F)=1	(TM F/F)=1	×
Power supply control	TCK	00	0000	1111	00F	1	1	Skip if (CK F/F)=1	(CK F/F)=1	×
	POF1	00	0000	1100	00C	1	1	(CK F/F)←0, with no CK flag input	—	×
	POF2	00	0000	1011	00B	1	1	(PW F/F)←0, with no PW flag input	—	×
	TPW	00	0001	1111	01F	1	1	Skip if (PW F/F)=1	(PW F/F)=1	×
	DETS	00	0000	0110	006	1	1	(DET F/F)←1	—	×
	DETR	00	0000	0111	007	1	1	(DET F/F)←0	—	×
	SDET	00	0000	1101	00D	1	1	Skip if $(BD_{OUT})=1$, (Skip if normal supply voltage apply)	$(BD_{OUT})=1$	×
Misc.	NOP	00	0000	0000	000	1	1	No operation	—	×

Note 1. When a skip has been generated, the next instruction only is invalid and the program counter is not incremented by 2. Therefore, the number of cycles does not change even if a skip is not generated.
2. Instructions Bxy, BAxX, BMxy and BMAxX execute the second function of the functions column when executed, provided that none of the instructions RT, RTS, BL, BML, BLA, or BMLA was executed after the execution of a BM or BMA instruction.

Symbol	Meaning	Symbol	Meaning
A	4-bit register (accumulator)	P(COMn)	Common output port for liquid crystal display
Ai	Indicates the bits of register A. Where i=0~1	P(LCn)	Segment output port for liquid crystal display
B	4-bit auxiliary register	PW F/F	1-bit power supply control flag display
BD_{OUT}	Battery detector signal	R(COMn)	Common register for liquid crystal display (4 bits)
CK F/F	1-bit 1-second flag	R(LCn)	Segment register for liquid crystal display (25 bits)
CY	1-bit carry flag	S	4-bit I/O port
CY'	1-bit carry stack flag	SCA	Output of bit A of control counter for liquid crystal display
D	11-bit output port	SK0	11-bit stack register
Di	Indicates the bits of port D. Where i=0~3	SK1	11-bit stack register
D(Y)	The bit of port D addressed by Y	SK2	11-bit stack register
DP	7-bit data pointer composed of register Y, X	TM	4-bit timer/counter
		TM F/F	1-bit timer/counter flag
Y	4-bit register	xx	2-bit binary variable
X	3-bit register	yyyy	4-bit binary variable
DP'	7-bit data pointer	mmm	3-bit binary variable
DET F/F	1-bit battery detector flag	nnnn	4-bit binary variable
F	8-bit output port	ii	2-bit binary variable
Fi	Indicates the bits of port F. Where i=0~7	jjj	3-bit binary variable
K	4-bit I/O port	XXXX	4-bit unknown binary variable (the value doesn't affect execution)
M(DP)	4-bit data of memory addressed by data pointer DP	←	Indicates direction of data flow
		( )	Indicates contents of register memory, etc.
Mi(DP)	A bit of data of memory addressed by data pointer DP where i=0~3	∀	Exclusive OR
		∧	AND
PC	11-bit program acounter composed of PC_L, PC_H	—	Negation
PC_L	Low-order 7 bits of the program counter	X	Indicates flag is unaffected by instruction execution
PC_H	High-order 4 bits of the program counter	xy	Label used to indicate the address
P_0	4-bit output port	C + x	Hexadecimal number C + binary number-X
P_1	4-bit output port		

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.3~6.0	V
V_I	Input voltage		−0.3~V_{CC}+0.3	V
V_O	Output voltage		0~V_{CC}	V
P_d	Power dissipation	Ta=25℃	300	mW
Topr	Operating free-air temperature range		−20~70	℃
Tstg	Storage temperature range		−40~125	℃

RECOMMENDED OPERATING CONDITIONS (Ta=−20~70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	3	4.5	5.5	V
V_{SS}	Supply voltage		0		V
V_{LCD}	Liquid crystal supply voltage		1.3		V
V_{IH}	High-level input voltage	0.7V_{CC}		V_{CC}	V
V_{IL}	Low-level input voltage	0		0.3V_{CC}	V
f_{XIN}	Oscillator frequency	240	455	520	kHz
$f\phi$	Internal clock oscillator frequency	120		260	kHz

ELECTRICAL CHARACTERISTICS (Ta=−20~70°C, V_{CC}=4.5V, V_{SS}=0V, f_{XIN}=455kHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage, ports D, K, and S	I_{OH} = −10μA	4			V
V_{OH}	High-level output voltage, ports F, and P	I_{OH} = −200μA	2.4			V
V_{OL}	Low-level output voltage, ports D, K, and S	I_{OL} = 1.8mA			0.5	V
V_{OL}	Low-level output voltage, ports F, and P	I_{OL} = 1.8mA			0.5	V
V_{OH}	High-level output voltage, port LC	V_{LCD}=1.3V, Ta=25℃	4.3	4.5		V
V_{OH}	High-level output voltage, port COM	V_{LCD}=1.3V, Ta=25℃	4.3	4.5		V
V_{OX}	Medium output voltage, port COM (Note 1)	V_{LCD}=1.3V, Ta=25℃	2.7	2.9	3.1	V
V_{OL}	Low-level output voltage, port LC	V_{LCD}=1.3V, Ta=25℃		1.3	1.5	V
V_{OL}	Low-level output voltage, port COM	V_{LCD}=1.3V, Ta=25℃		1.3	1.5	V
I_{CC}	Supply current for full operation	Ta=25℃, Output pins open		0.4		mA
I_{CC}	Supply current for partial operation	Ta=25℃, Output pins open		20		μA
C_i	Input capacitance	V_{CC}=V_I=V_O=V_{SS}, f=1MHz, 25mVrms		7	10	pF
$C_{i(XIN)}$	Oscillator input capacitance	V_{CC}=X_{OUT}=V_{SS}, f=1MHz, 25mVrms		7	10	pF
V_{BD}	Battery voltage detection voltage range (Note 2)	10kΩ≤R_{BD}≤200kΩ, Ta=25℃	4		5.5	V

Note 1. V_{OX} is the medium level of the 3-level output of port COM.
2. The detection resistance R_{BD} is connected between the V_{SS} and BDIN pin.
3. Currents are taken to be positive when flowing into the IC with minimum and maximum values taken as absolute values.

TIMING REQUIREMENTS (Ta=−20~70°C, Vcc=3~5.5V, Vss=0V, unless otherwise noted)

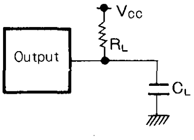
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{su(K-X_{IN})}$	Data setup time before clock input, port K inputs	f_ϕ=230kHz (Note 1)	0			μs
$t_{su(S-X_{IN})}$	Data setup time before clock input, port S inputs		0			μs
$t_{su(INT_A-X_{IN})}$	Data setup time before clock input, INT_A input		0			μs
$t_{su(INT_B-X_{IN})}$	Data setup time before clock input, INT_B input		0			μs
$t_{h(K-X_{IN})}$	Data hold time after clock input, port K inputs		0.4			μs
$t_{h(S-X_{IN})}$	Data hold time after clock input, port S inputs		0.4			μs
$t_{h(INT_A-X_{IN})}$	Data hold time after clock input, INT_A input		0.4			μs
$t_{h(INT_B-X_{IN})}$	Data hold time after clock input, INT_B input		0.4			μs

Note 1. $f\phi = 1/2.f_{XIN}$ which corresponds to the internal clock frequency.

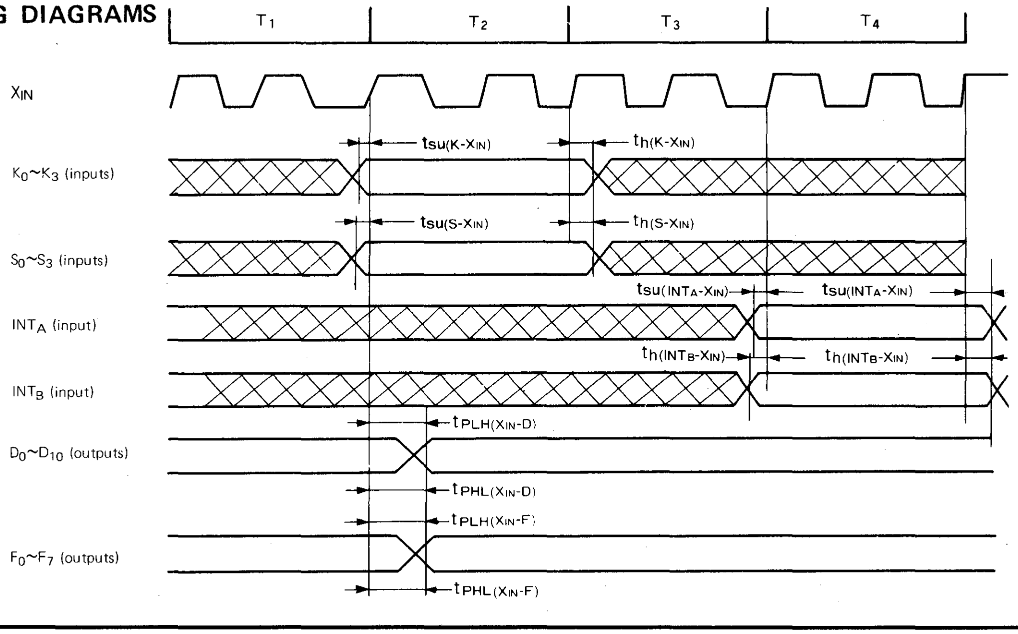
SWITCHING CHARACTERISTICS (Ta=−20~70°C, Vcc=3~5.5V, Vss =0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{PLH(X_{IN}-D)}$	Low-to-high-level propagation time from clock input to port data output, port D	f_ϕ=230kHz, R_L=20kΩ, C_L=100pF (Note 2)		2.2	3	μs
$t_{PLH(X_{IN}-F)}$	Low-to-high level propagation time from clock input to port data output, ports F, P, K, and S			2.2	3	μs
$t_{PHL(X_{IN}-D)}$	High-to-low-level propagation time from clock input to port data output, port D			0.7	1.5	μs
$t_{PHL(X_{IN}-F)}$	High-to-low-level propagation time from clock input to port data output, ports F, P, K, and S			0.7	1.5	μs

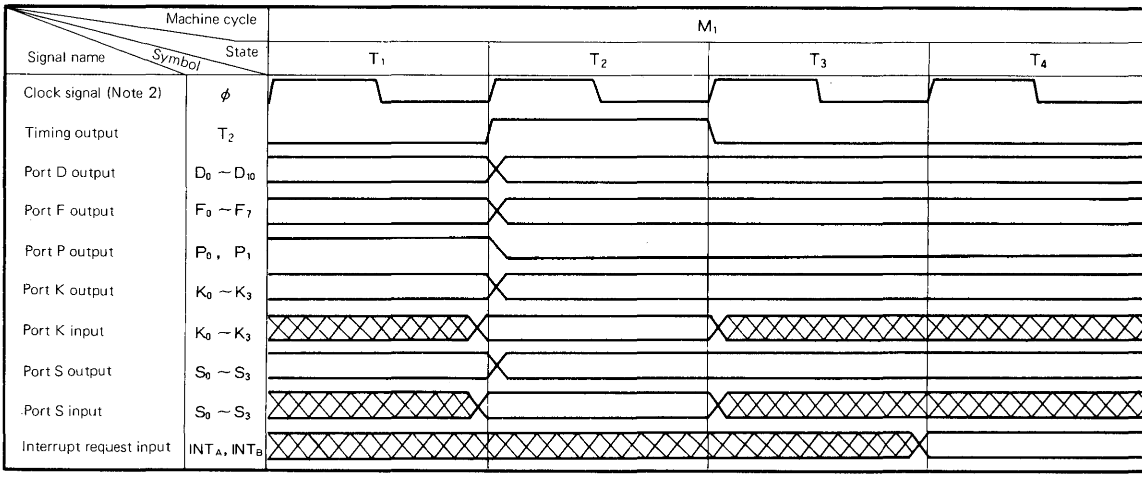
Note 2. Measurement circuit



TIMING DIAGRAMS

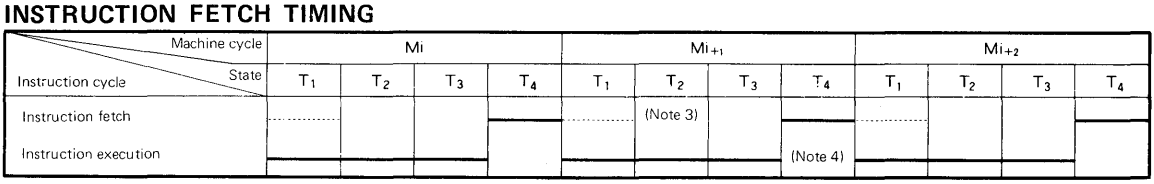


BASIC TIMING CHART (Note 1)



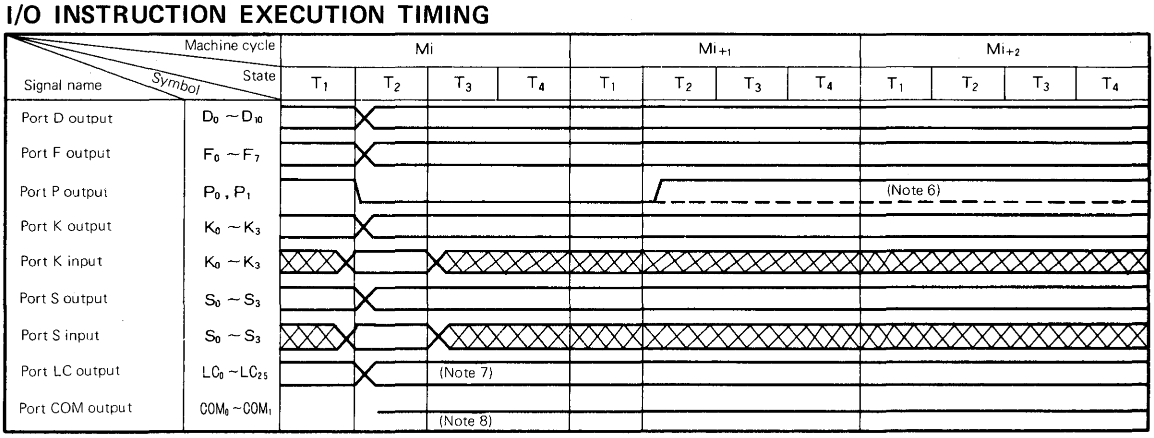
Note 1. [XXX] Indicates an invalid signal input.
2. Internal clock signal which is 1/2 of basic oscillation frequency.

INSTRUCTION FETCH TIMING



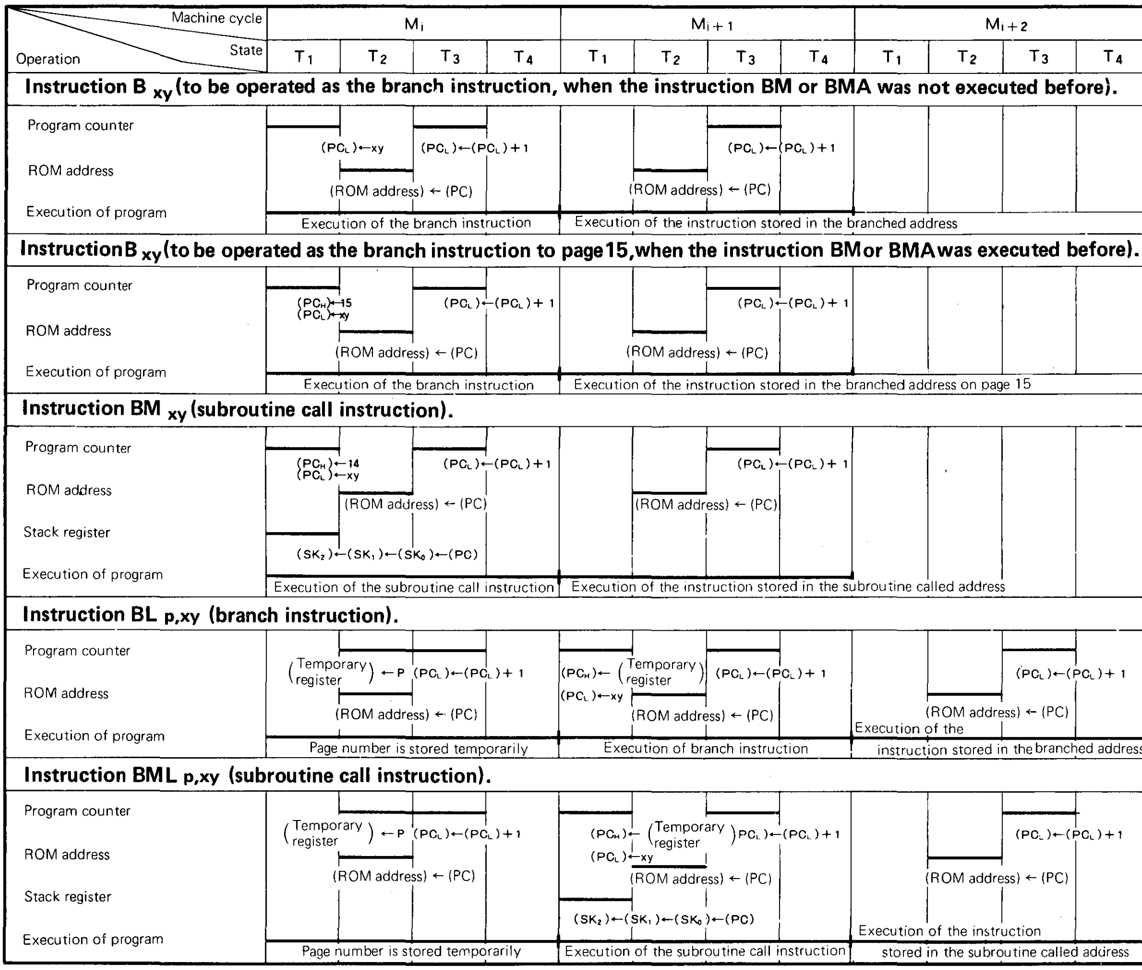
Note 3. Instruction fetch time can differ depending on the types of the instructions.
4. The instruction which was fetched in the preceding cycle is executed.
5. The execution of the instruction and addressing of ROM and RAM are performed simultaneously.

I/O INSTRUCTION EXECUTION TIMING



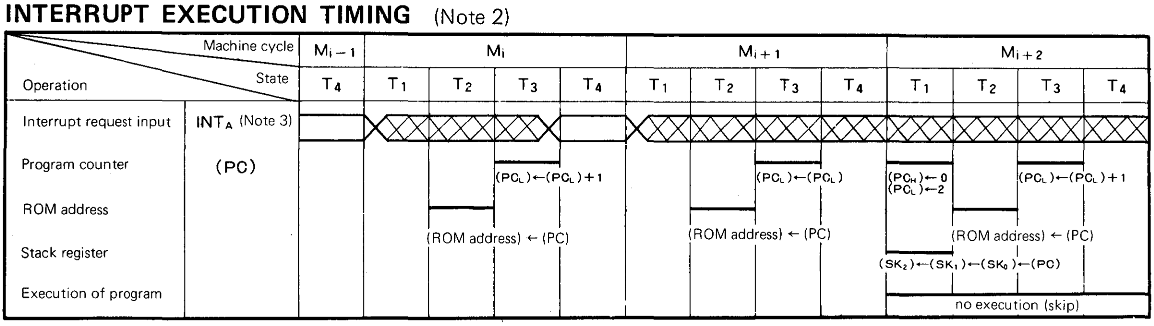
Note 6. When an OTRO instruction is executed, the output is latched.
7. Output voltage of port LC depends upon power supply V_{LCD} for the liquid crystal display.
8. Output voltage of port COM has 3 levels depending on the power supply V_{LCD} for the liquid crystal display.

BRANCH AND SUBROUTINE CALL INSTRUCTION EXECUTION TIMING (Note 1)



Note 1. The instructions BA, BMA, BLA and BMAL have the same execution timing as B, BM, BL and BML respectively as shown. The only difference is that $(PC_L) \leftarrow xy$ is replaced by $(PC_L) \leftarrow x(A)$.

INTERRUPT EXECUTION TIMING (Note 2)



Note 2 When the instruction executed in the machine cycle M_{i+1} is a BL, BML, BLA or BMLA, the value of address 2 of page 0 is stored in the program counter during M_{i+3}.

3 The interrupt request input INT_B has the same execution timing as INT_A. If the input is low level in the machine cycle M_{i-1} and high level in the machine cycle M_i, the interrupt is executed during the interrupt enable state.

MELP S8-48 MICROCOMPUTERS

6

FUNCTION OF MELPS 8-48 MICROCOMPUTERS

DESCRIPTION

The MELPS 8-48 LSI family is a low-cost high-performance single-chip microcomputer series. The functions have been integrated. For example the CPU, ROM, RAM, I/O ports, timer and other circuits are all on one chip. The MELPS 8-48 family has the following three configurations to meet the requirements of different applications for various ROM and RAM capacities.

M5L8048-XXXP

ROM	1024 bytes
RAM	64 bytes
I/O	27 pins

M5L8049-XXXP

ROM	2048 bytes
RAM	128 bytes
I/O	27 pins

M5L8748S

EPROM	1024 bytes
RAM	64 bytes
I/O	27 pins

Timer and interrupt inputs are also built into these chips. The program memory capacity can easily be expanded to 4K bytes. The M5L8243P input/output expander chip can be used to extend the I/O capability. The family of microcomputers allows designers to fabricate systems for applications simply and quickly.

The M5L8048-XXXP contains 1K bytes of read only memory and the M5L8049-XXXP contains 2K bytes. The contents of the memory is set by a mask during manufacture. This makes it practical to mass produce ROMs containing customer developed programs.

The M5L8748S contains 1K bytes of EPROM and is pin-compatible with the M5L8048-XXXP. Its memory can be electrically written and changed by the user. This chip can be used while a system is being developed and subject to modifications. Once the system has been checked out and the program debugged, the program can be masked in the M5L8048-XXXP.

A cross assembler, the MELPS 8-48, is available for use with this family of microcomputers. Designers will find the assembler convinient and easy to use.

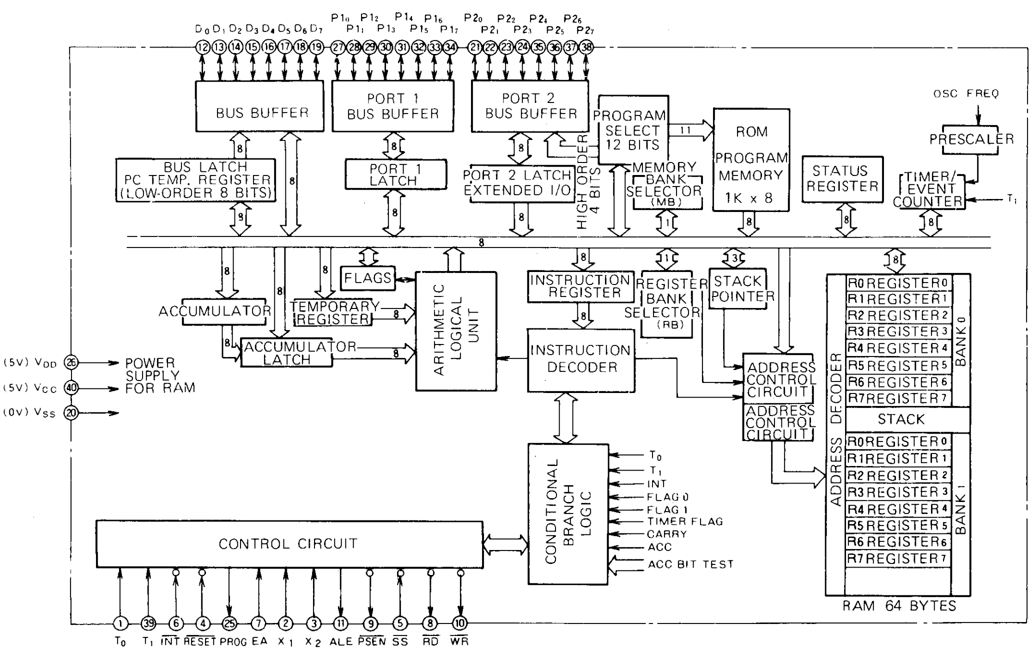


Fig. 1 Block diagram of M5L8048-XXXP

FUNCTION OF MELPS 8-48 MICROCOMPUTERS

BLOCK DIAGRAM

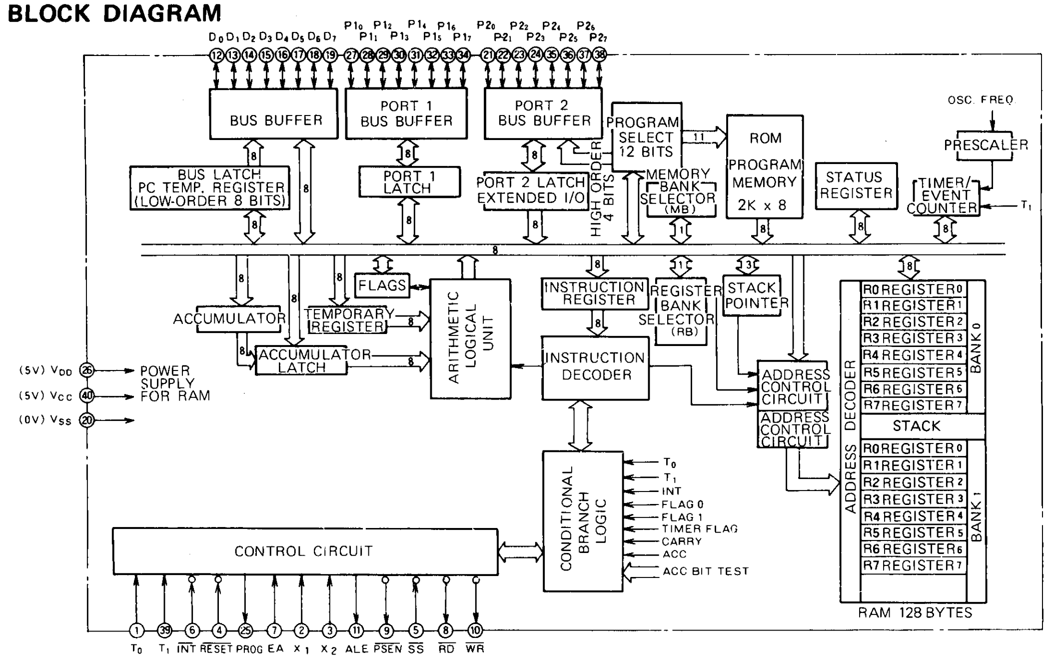


Fig. 2 Block diagram of M5L8049-XXXP

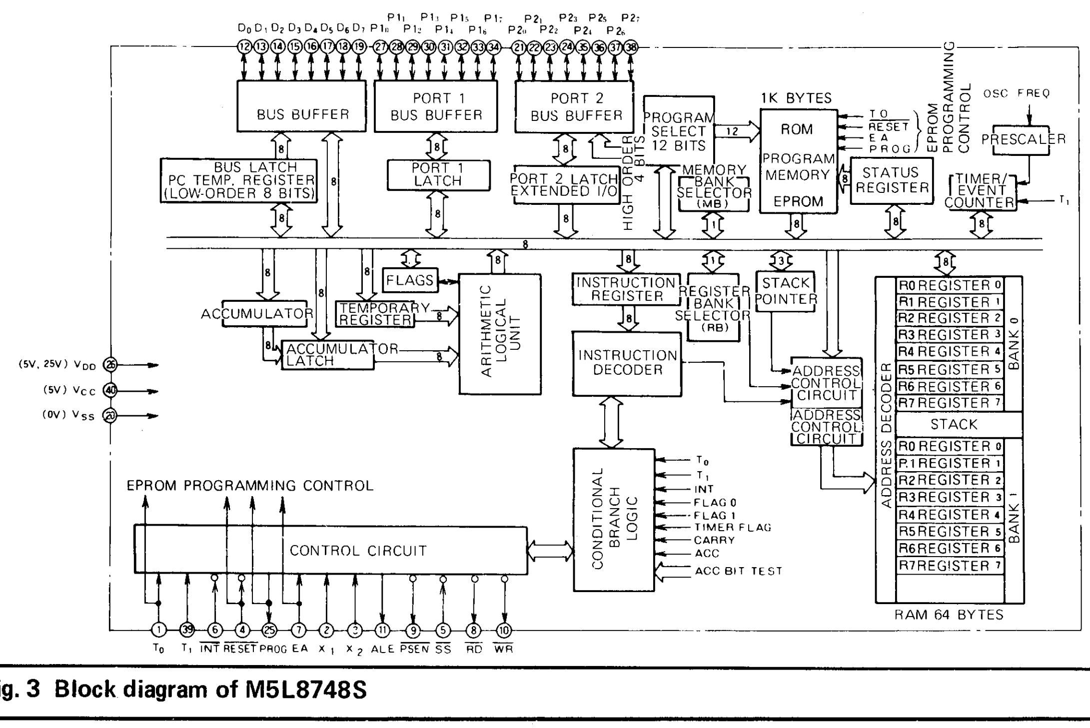


Fig. 3 Block diagram of M5L8748S

BASIC FUNCTION BLOCKS

Program Memory (ROM)

The M5L8048-XXXP and M5L8748S contain 1024 bytes of ROM, in the case of the M5L8748S, it is EPROM and its contents can easily be changed by the user. The M5L8049-XXXP contains 2048 bytes of ROM. The program for the users application is stored in this ROM. Addresses 0, 3, 7 of the ROM are reserved for special functions. Table 1 shows the meaning and function of these three special addresses.

Table 1 Reserved, defined addresses and their meanings and functions

Address	Meaning and function
0	The first instruction executed after a system reset.
3	The first instruction executed after an external interrupt is accepted.
7	The first instruction executed after a timer interrupt is accepted.

The ROM can be used to store constants and other 8-bit fixed data in addition to the program. Instructions such as MOVP A, @A and MOVP3 A, @A can be used to access the constants and data. The data could be in the form of tables, and can be easily looked up.

Data Memory (RAM)

The M5L8048-XXXP and M5L8748S contain 64 bytes of RAM. The M5L8049-XXXP contains 128 bytes of RAM. The RAM is used for data storage and manipulation and is divided into sections for more efficient processing. Addresses 0~7 and 24~31 form two banks of general purpose registers that can be directly addressed. Addresses 0~7 compose bank 0 and are numbered R0~R7. Addresses 24~31 compose bank 1 and are also numbered R0~R7. Only one bank is active at a time. The instructions SEL RB0 and SEL RB1 are used to select the working bank. Fig. 1 shows the division of the RAM and its mapping.

Addresses 8~23 compose an 8-level program counter stack. The details for using the stack will be found in the "Program Counter and Stack" section. Please refer to that section for details.

The remaining section, addresses 32 and above, must be accessed indirectly using the general-purpose registers R0 or R1. Of course all addresses can be indirectly addressed using the general-purpose registers R0 and R1.

A good practice to simplify programming is to reserve general-purpose register bank 0 for use of the main program and register bank 1 for interrupt programs. For example if register bank 0 (addressed 0~7) is reserved for processing data by the main program, when an interrupt is accepted the first instruction would be to switch the working registers from bank 0 to bank 1. This would save the data of the main program (addresses 0~7). The interrupt program can then freely use register bank1 (addresses 24~31) without destroying or altering data of the main program. When the interrupt processing is complete and control is returned to the main program by the RETR instruction, register bank 0 (in this example) is automatically restored as the working register bank at the same time the main program counter is restored.

Addresses 0~31 have special functions, but when not all of the registers are required, the ones not needed can be used for general storage. This includes both banks of general-purpose registers and the stack.

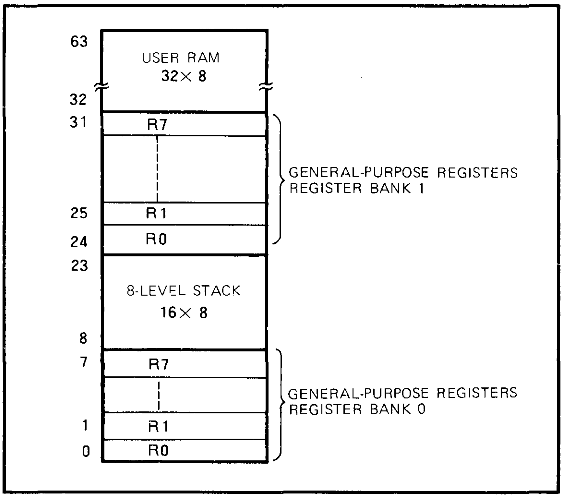


Fig. 4 Data memory (RAM)

PROGRAM COUNTER (PC) AND STACK (SK)

The MELPS 8-48 program counter is composed of a 12-bit binary counter as shown in Fig. 5. The low-order 10 bits can address 1024 bytes of memory. When the high-order 2 bits are zero, the internal, on chip memory is accessed. The high-order 2 bits can have the values 1~3, which allows the user to add up to three banks of 1024 bytes. The program counter can address up to 4096 bytes of memory.

Addresses 8~23 of RAM are used for the stack (program counter stack). The stack provides an easy and automatic means of saving the program counter and other control information when an interrupt is accepted or a subroutine is called. For example, if control is with the main program and an interrupt is accepted, the contents of the 12-bit PC (program counter) is saved in the top of the stack, so it can be restored when control is returned to the main program. In addition to the PC, the high-order 4 bits of the PSW (program status word) are saved in the stack and restored along with the PC. A total of 16 bits are saved, the 12-bit

PC and 4 bits of the PSW. A 3-bit stack pointer is associated with the stack. This pointer is a part of the PSW and indicates the top of the stack. The stack pointer indicates the next empty location (top of the stack), in case of an empty stack the top of the stack is the bottom of the stack. The data memory addresses associated with the stack pointer along with the data storage sequence are shown in Fig. 6.

PC_{11}	PC_{10}	PC_9	PC_8	PC_7	PC_6	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0

Fig. 5 Program counter

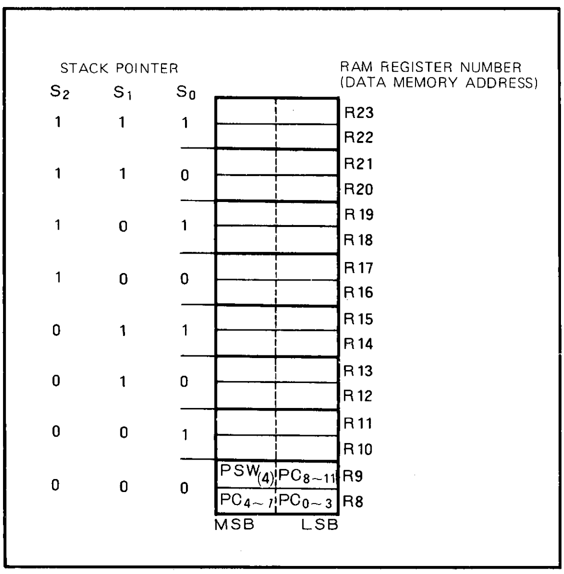


Fig. 6 Relation between the program counter stack and the stack pointer

PROGRAM STATUS WORD (PSW)

The PSW (program status word) is stored in 8 bits of register storage. The configuration of the PSW is shown in Fig. 7. The high-order 4 bits of the PSW are stored in the stack, along with the PC, when an interrupt is accepted or a subroutine call executed. When control is returned to the main program by RETR both the PC and the high-order 4 bits of PSW are restored. When control is returned by RET only the PC is restored, so care must be taken to assure that the contents of the PSW was not unintentionally changed.

The order and meaning of the 8 bits of the PSW are shown below.

Bit 0~2: Stack pointer (S_0, S_1, S_2)
Bit 3: Unused (always 1)
Bit 4: Working register bank indicator
0 = Bank 0
1 = Bank 1
Bit 5: Flag 0 (value is set by the user and can be tested)
Bit 6: Auxiliary carry (AC) (it is set/reset by instructions ADD and ADC and used by instruction DA A).
Bit 7: Carry bit (C) (indicates an overflow after execution)

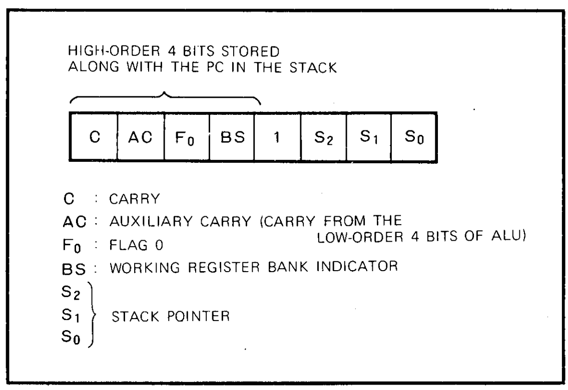


Fig. 7 Program status word

I/O PORTS

The MELPS 8-48 has three 8-bit ports, which are called data bus, port 1 and port 2.

Port 1 and Port 2

Ports 1 and 2 and both 8-bit ports with identical properties. The output data of these ports are retained and do not change until another output is loaded into them. When used as inputs the input data is not retained so the input signals must be maintained until an input instruction is executed and completed.

Ports 1 and 2 so-called quasi-bidirectional ports have a special circuit configuration to accomplish this. The special circuit is shown in Fig. 8. All terminals of ports 1 and 2 can be used for input or output.

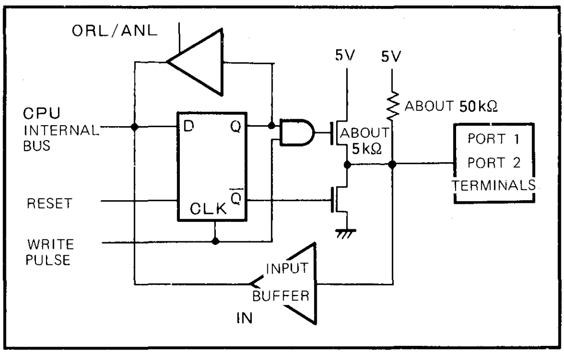


Fig. 8 I/O ports 1 and 2 circuit

Internal on chip pull-up resistors are provided for all the ports. Through the use of pull-up resistors, TTL standard high-level or low-level signals can be supplied. Therefore each terminal can be used for both input and output. To shorten switching time from low-level to high-level, when 1s are output, a device of about 5kΩ or lower is inserted for a short time (about 500ns when using a 6MHz crystal oscillator).

A port used for input must output all 1s before it reads the data from the input terminal. After resetting, a port is set to an input port and remains in this state, therefore it is not necessary to output all 1s if it is to be used for input. In short a port being used for output must output 1s before it can be used for input.

The individual terminals of quasi-bidirectional ports can be used for input or output. Therefore some terminals can be in the input mode while the remaining terminals of a port are in the output mode. This capability of ports 1 and 2 is convenient for inputting or outputting 1-bit or data with few bits. The logical instructions ANL and ORL can easily be used to manipulate the input or output of these ports.

Data Bus (Port 0)

The data bus is an 8-bit bidirectional port, which is used with I/O strobed signals. When the data bus is used for output the output data is latched, but if it is used for input the data is not latched. Unlike ports 1 and 2, which can have individual terminals in the input or output mode, all terminals of the data bus are in the input or output mode.

When the data bus is used as a static port the OUTL instruction can be used to output data and the INS instruction to input data. Strobe pulse $\overline{RD}$ is generated while the INS instruction is being executed or $\overline{WR}$ while OUTL is being executed.

The data bus read/write using MOVX instructions, but then the data bus is a bidirectional port. To write into the data bus a $\overline{WR}$ signal is generated and the data is valid when $\overline{WR}$ goes high. When reading from the data bus, an $\overline{RD}$ signal is generated. The input levels must be maintained until $\overline{RD}$ goes high. When the data bus is not reading/writing, it is in the high-impedance state.

6

CONDITIONAL JUMPS USING TERMINALS T_0, T_1 and $\overline{INT}$

Conditional jump instructions are used to alter program depending on internal and external conditions (states). Details of the jump instructions for the MELPS 8-48 can be found in the section on machine instructions.

The input signal status of T_0, T_1 and $\overline{INT}$ can be checked by the conditional jump instructions. These input terminals, through conditional jump instructions such as JTO and JNTO, can be used to control a program. Programs and processing time can be reduced by being able to test data in input terminal rather than reading the data into a register and then testing it in the register.

Terminal T_0, T_1 and $\overline{INT}$ have other functions and uses that are not related to conditional jump instructions. The details of these other functions and uses can be found in the section on terminal functions.

INTERRUPT

The CPU recognizes an external interrupt by a low-level state at the $\overline{INT}$ terminal. A "Wired-OR" connection can be used for checking multiple interrupts.

The $\overline{INT}$ terminal is tested for an interrupt request at the ALE signal output of every machine cycle. When an interrupt is recognized and accepted, control is transferred to the interrupt handling program. This is accomplished by an unconditional jump to address 3 of program memory, which is the start of the interrupt handling program, at the same time the program counter and 4 high-order bits of PSW are automatically moved to the top of the stack.

The interrupt level is one, so the next interrupt cannot be accepted until the current interrupt processing has been completed. The RETR instruction terminates the interrupt processing. That is to say, the next interrupt can not be accepted until the RETR instruction is executed. The next interrupt can be accepted at the start of the second cycle of the RETR instruction (2-cycle instruction). Time/event counter overflow which causes an interupt request also will not be accepted.

After the processing for an interrupt is completed control is returned to the main program. This is accomplished by executing RETR which restores the program counter and PSW automatical and checks $\overline{INT}$ and the time/event counter overflow for an interrupt request. If there is an interrupt request, the control will not be returned to the main program but will be transferred to the interrupt handling program.

An external interrupt has a higher priority than a timer interrupt. This means that, if an external and timer interrupt request are generated at the same time, the external interrupt has the priority and will be accepted first.

When a second level of external interrupt is required, the timer interrupt, if not being used, can provide this. The procedure for this is to first disable the timer interrupt, set the timer/event counter to FF_{16} and put the CPU in the event counter mode. After this has been done, if T_1 input is changed to low-level from high-level, an interrupt is generated in address 7.

Terminal $\overline{INT}$ can also be tested using a conditional jump instruction. For more details on this procedure, check the "Conditional Jumps Using Terminals T_0, T_1 and $\overline{INT}$" section.

TIMER/EVENT COUNTER

The timer/event counter for the MELPS 8-48 is an 8-bit counter, that is used to measure time delays or count external events. The same counter is used to measure time delays or count external events by simply changing the input to the counter.

The counter can be initialized by executing an MOV T, A instruction. The value of the counter can be read for checking by executing an MOV A, T instruction. Reset will stop the counting but the counter is not cleared, so counting can be resumed.

The largest number the counter can contain is FF_{16}. If it is incremented by 1 when it contains FF_{16}, the counter will be reset to 0, the overflow flag is set and a timer interrupt request is generated.

The conditional jump instruction JTF can be used to test the overflow flag. Care must be used in executing the JTF instruction because the overflow flag is cleared (reset) when executed. When a timer interrupt is accepted, the control is transferred to address 7 of program memory.

When both a timer and external interrupt request are generated at the same time, the external interrupt is given priority and will be accepted first by automatically jumping to address 3 of program memory. The timer interrupt request is kept and will be processed when the external interrupt has been completed and a PETR is executed. A latched timer interrupt request is cancelled when a timer interrupt request is generated. A timer interrupt request can be disabled by executing a DIS TCNTI instruction.

The STRT CNT instruction is used to change the counter to an event counter. Then terminal T_1 signal becomes the input to the event counter and an event is counted each full cycle (low-high-low one event). The maximum rate that can be counted is one time in 3 machine cycles (7.5μs when using 6MHz crystal). The high-level at T_1 must be maintained at least 1/5 of the cycle time (500ns when using 6MHz crystal).

The STRT T instruction is used to change the counter to a timer. The internal clock signal becomes the input to the timer. The internal clock is 1/32 of 400kHz (when using 6MHz crystal) or 12.5kHz. The timer is therefore counted up every 80μs. Fig. 9 shows the timer/event counter.

The counter can be initialized by executing an MOV T, A instruction. The timer can be used to measure 80μs~20ms in multiples of 80μs. When it is necessary to measure over 20ms (maximum count 256x80μs) of delay time the number of overflows,one every 20ms, can be counted by the program. To measure times of less than 80μs; external clock pulses can be input through T_1 while the counter is in the event counter mode. Every third (or more) ALE signal can be used instead of an external clock.

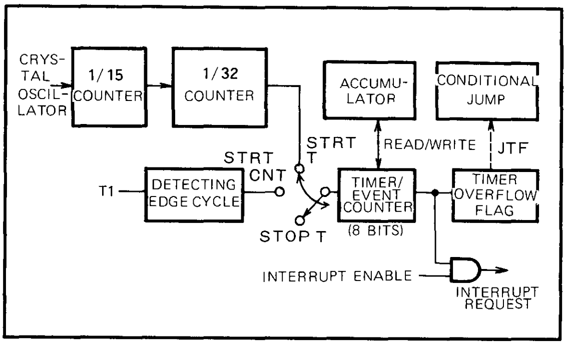


Fig. 9 Timer/event counter

MELPS 8-48 CYCLE TIMING

The output of the state counter is 1/3 the input frequency from the oscillator. When a 6MHz crystal is used for input, the output would be 2MHz (500ns). A CLK signal is generated every 500ns (one state cycle) which is used for the demarcation of each machine state. The instruction ENTO CLK will output the CLK signal through terminal T_0. The input of the cycle counter is CLK (state cycle) and the output is an ALE signal which is generated every 5 state cycles.

Fig. 11 Shows the relationship between clock and generated cycles.

One machine cycle contains 5 states with a CLK signal for demarcation of each state. The MELPS 8-48 instructions are executed in one machine cycle or two machine cycles. An instruction cycle can be one or two machine cycles as shown in Fig. 12.

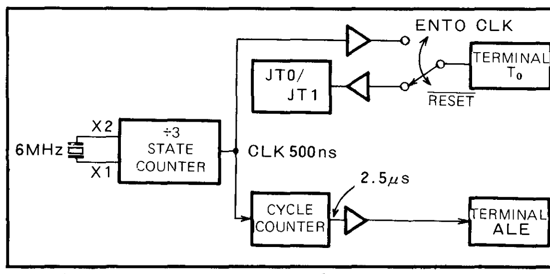


Fig.10 Clocking cycle generation

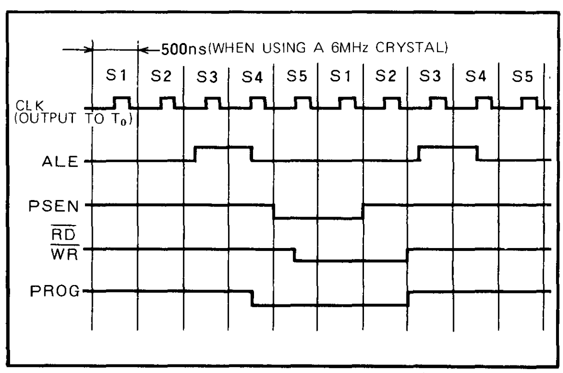


Fig.11 Clock and generated cycle signals

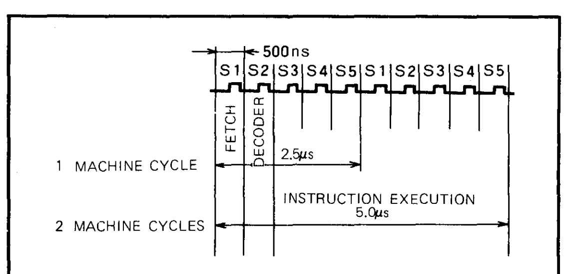


Fig.12 Instruction execution timing

RESET

The reset terminal is for resetting the CPU. A Schmitt trigger circuit along with a pull-up register are connected to it on the chip. A reset can easily be generated by attaching a 1μF as capacitor as shown in Fig. 13. An external reset pulse applied at RESET must remain at low-level for at least 50ms after power has been turned on and reached its normal level.

The reset function causes the following initialization within the CPU.

1. Program counter is reset to 0.
2. Stack pointer is reset to 0.
3. Register bank is reset to 0.
4. Memory bank is reset to 0.
5. Data bus is cleared to high-impedance state.
6. Ports 1 and 2 are reset to input mode.
7. External and timer interrupts are reset to disable state.
8. Timer is stopped.
9. Timer overflow flag is cleared.
10. Flags F_0 and F_1 are cleared.
11. Clock output for terminal T_0 is disabled.

Note 1: On the M5L8748S the RESET terminal, in addition to being used for the reset function, is also used when reading and writing data in the EPROM on the chip. Details on this will be found in the section on reading and writing data in the M5L8748S.

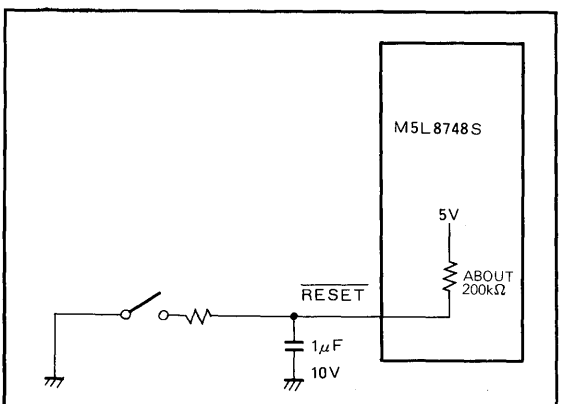


Fig. 13 Example of a reset circuit

SINGLE-STEP OPERATION

The terminal SS on the MELPS 8-48 is provided to facilitate single-step operation. In single-step operation, the CPU stops after the execution of each instruction is completed and the memory address (12 bits) of the next instruction to be fetched is output through the data bus (8 bits) plus the low-order 4 bits of port 2 (P_{20}~P_{23}). The user can use this to trace the flow of this program instruction by instruction and will find this an aid in program debugging. Single-step operation is controlled through SS and ALE as shown in Fig. 14.

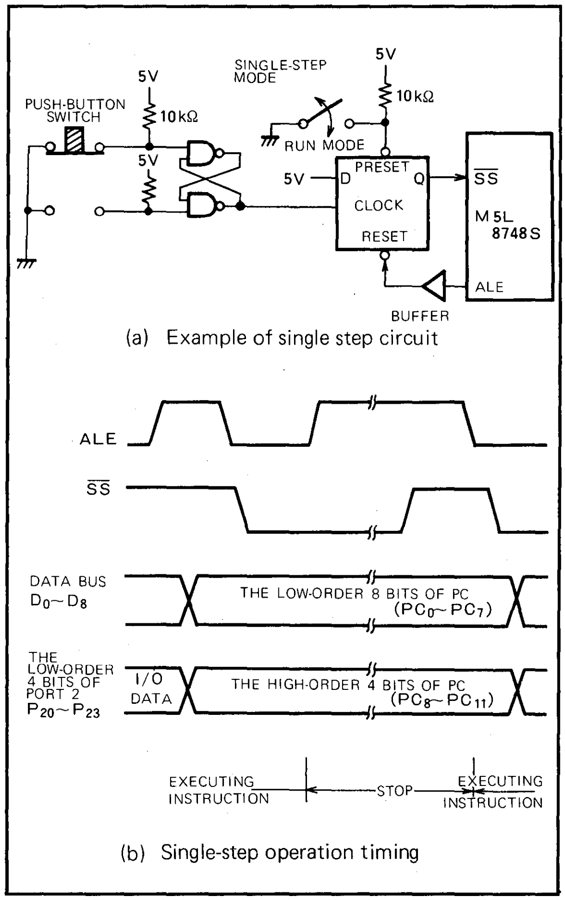


Fig. 14 Single-step operation circuit and timing

A type D flip-flop with preset and reset terminals, as shown in Fig. 11, is used to generate the signal for SS. When the preset terminal goes to low-level, SS goes to high-level, which puts the CPU in RUN mode. When the preset terminal is grounded it goes to high-level. Then SS goes to low-level. When SS goes to low-level, the CPU stops. Then when the push-button switch is pushed, a pulse is sent to the clock terminal of the type D flip-flop which turns SS to high-level. When SS goes to high-level the CPU fetches the next instruction and begins to execute it, but then an ALE signal is sent to the reset terminal of the type D flip-flop which turns $\overline{SS}$ to low-level. The CPU again stops as soon as execution of the current instruction is completed. When the push-button switch is again pushed, the cycle is repeated and the CPU is in single-step operation as shown in Fig. 12. While the CPU is stopped in single-step operation, the data bus and the low-order 4 bits of port 2 are used to output the memory address of the next instruction to be fetched. This interferes with input and output, but essential input/output can be latched by using the rising edge of ALE as clock.

Central Processing Unit (CPU)

Central Processing Unit (CPU) is composed of an 8-bit parallel arithmetic unit, accumulator, flag flip-flop and instruction decoder. The 8-bit parallel arithmetic unit has circuitry to perform the four basic arithmetic operations (plus, minus, multiply and divide) as well as logical operations such as AND and OR. The flag flip-flop is used to indicate status such as carry and zero. The accumulator contains one of the operations and the result is usually retained in the accumulator.

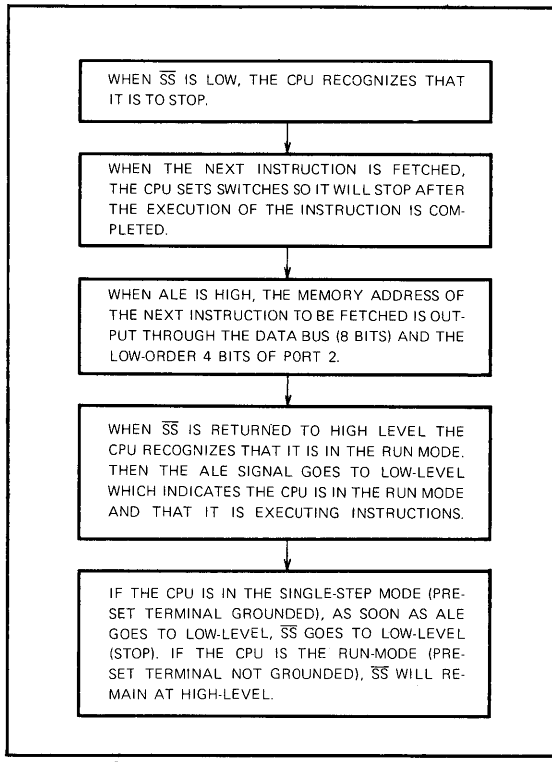


Fig. 15 CPU operation in single-step mode

FUNCTION OF MELPS 8-48 MICROCOMPUTERS

MACHINE INSTRUCTIONS

Type \ Item	Mnemonic	Instruction code $D_7 D_6 D_5 D_4$ $D_3 D_2 D_1 D_0$	Hexa-decimal	Bytes	Cycles	Function	Effected carry C	Effected carry AC	Note	Description
Transfer	**MOV A, #n**	0 0 1 0 0 0 1 1 $n_7n_6n_5n_4$ $n_3n_2n_1n_0$	23 n	2	2	$(A) \leftarrow n$				Transfers data n to register A.
	MOV A, PSW	1 1 0 0 0 1 1 1	C7	1	1	$(A) \leftarrow (PSW)$				Transfers the contents of the program status word to register A.
	MOV A, Rr	1 1 1 1 1 $r_2r_1r_0$	F8 + r	1	1	$(A) \leftarrow (Rr)$ $r = 0 \sim 7$				Transfers the contents of register R_r to register A.
	MOV A, @Rr	1 1 1 1 0 0 0 r_0	F0 + r	1	1	$(A) \leftarrow (M(Rr))$ $r = 0 \sim 1$				Transfers the contents of memory location, of the current page, whose address is in register R_r to register A.
	MOV PSW, A	1 1 0 1 0 1 1 1	D7	1	1	$(PSW) \leftarrow (A)$ $(C) \leftarrow (A_7)$, $(AC) \leftarrow (A_6)$	○	○		Transfers the contents of register A to the program status word.
	MOV Rr, A	1 0 1 0 1 $r_2r_1r_0$	A8 + r	1	1	$(Rr) \leftarrow (A)$ $r = 0 \sim 7$				Transfers the contents of register A to register R_r.
	MOV Rr, #n	1 0 1 1 1 $r_2r_1r_0$ $n_7n_6n_5n_4$ $n_3n_2n_1n_0$	B8 + r n	2	2	$(Rr) \leftarrow n$ $r = 0 \sim 7$				Transfers data n to register R_r.
	MOV @Rr, A	1 0 1 0 0 0 0 r_0	A0 + r	1	1	$(M(Rr)) \leftarrow (A)$ $r = 0 \sim 1$				Transfers the contents of register A to memory location, of the current page, whose address is in register R_r.
	MOV @Rr, #n	1 0 1 1 0 0 0 r_0 $n_7n_6n_5n_4$ $n_3n_2n_1n_0$	B0 + r n	2	2	$(M(Rr)) \leftarrow n$ $r = 0 \sim 1$				Transfers data n to memory location, of the current page, whose address is in register R_r.
	MOVP A, @A	1 0 1 0 0 0 1 1	A3	1	2	$(A) \leftarrow (M(A))$				Transfers the data of memory location, of the current page, whose address is in register A to register A.
	MOVP3 A, @A	1 1 1 0 0 0 1 1	E3	1	2	$(A) \leftarrow (M(\text{page } 3, A))$				Transfers the data of memory location, of page 3, whose address is in register A to register A.
	MOVX @Rr, A	1 0 0 1 0 0 0 r_0	90 + r	1	2	$(Mx(Rr)) \leftarrow (A)$ $r = 0 \sim 1$				Transfers the contents of register A to memory location, of the current page, whose address is in register R_r.
	MOVX A, @Rr	1 0 0 0 0 0 0 r_0	80 + r	1	2	$(A) \leftarrow (Mx(Rr))$ $r = 0 \sim 1$				Transfers the contents of memory location, of the current page, whose address is in register R_r to register A.
	XCH A, Rr	0 0 1 0 1 $r_2r_1r_0$	28 + r	1	1	$(A) \longleftrightarrow (Rr)$ $r = 0 \sim 7$				Exchanges the contents of register R_r with the contents of register A.
	XCH A, @Rr	0 0 1 0 0 0 0 r_0	20 + r	1	1	$(A) \longleftrightarrow (M(Rr))$ $r = 0 \sim 1$				Exchanges the contents of memory location, of the current page, whose address is in register R_r with the contents of register A.
	XCHD A, @Rr	0 0 1 1 0 0 0 r_0	30 + r	1	1	$(A_0 \sim A_3) \longleftrightarrow (M(Rr_0 \sim Rr_3))$ $r = 0 \sim 1$				Exchanges the contents of the low-order four bits of register A with the low-order four bits of memory location, of the current page, whose address is in register R_r.
Arithmetic	**ADD A, #n**	0 0 0 0 0 0 1 1 $n_7n_6n_5n_4$ $n_3n_2n_1n_0$	03 n	2	2	$(A) \leftarrow (A) + n$	○	○	1	Adds data n to the contents of register A and sets the carry flags to 1 if there is an overflow otherwise resets the carry flags to 0. The result is stored in register A.
	ADD A, Rr	0 1 1 0 1 $r_2r_1r_0$	68 + r	1	1	$(A) \leftarrow (A) + (Rr)$ $r = 0 \sim 7$	○	○	1	Adds the contents of register R_r to the contents of register A and sets the carry flags to 1 if there is an overflow otherwise resets the carry flags to 0. The result is stored in register A.
	ADD A, @Rr	0 1 1 0 0 0 0 r_0	60 + r	1	1	$(A) \leftarrow (A) + (M(Rr))$ $r = 0 \sim 1$	○	○	1	Adds the contents of register A and the contents of memory location, of the current page, whose address is in register A and sets the carry flags to 1 if there is an overflow otherwise resets the carry flags to 0. The result is stored in register A.
	ADDC A, #n	0 0 0 1 0 0 1 1 $n_7n_6n_5n_4$ $n_3n_2n_1n_0$	13 n	2	2	$(A) \leftarrow (A) + n + (C)$	○	○	1	Adds the carry and data n to the contents of register A and sets the carry flags to 1 if there is an overflow otherwise resets the carry flags to 0. The result is stored in register A.
	ADDC A, Rr	0 1 1 1 1 $r_2r_1r_0$	78 + r	1	1	$(A) \leftarrow (A) + (Rr) + (C)$ $r = 0 \sim 7$	○	○	1	Adds the carry and the contents of register R_r to the contents of register A and sets the carry flags to 1 if there is an overflow otherwise resets the carry flags to 0. The result is stored in register A.
	ADDC A, @Rr	0 1 1 1 0 0 0 r_0	70 + r	1	1	$(A) \leftarrow (A) + (M(Rr)) + (C)$ $r = 0 \sim 1$	○	○	1	Adds the carry and the contents of memory location, of the current page, whose address is in register R_r to the contents of register A and sets the carry flags to 1 if there is an overflow otherwise resets the carry flags to 0. The result is stored in register A.

MITSUBISHI ELECTRIC

FUNCTION OF MELPS 8-48 MICROCOMPUTERS

Type \ Item	Mnemonic	Instruction code $D_7 D_6 D_5 D_4$	$D_3 D_2 D_1 D_0$	Hexa-decimal	Bytes	Cycles	Function	Effected carry C	AC	Note	Description
Arithmetic	ANL A, #n	0 1 0 1 $n_7 n_6 n_5 n_4$	0 0 1 1 $n_3 n_2 n_1 n_0$	5 3 n	2	2	(A) ← (A) ∧ n				The logical product of the contents of register A and data n, is stored in register A.
	ANL A, Rr	0 1 0 1	1 $r_2 r_1 r_0$	5 8 + r	1	1	(A) ← (A) ∧ (Rr) r = 0 ~ 7				The logical product of the contents of register A and the contents of register R_r, is stored in register A.
	ANL A, @Rr	0 1 0 1	0 0 0 r_0	5 0 + r	1	1	(A) ← (A) ∧ (M(Rr)) r = 0 ~ 1				The logical product of the contents of register A and the contents of memory location, of the current page, whose address is in register R_r, is stored in register A.
	ORL A, #n	0 1 0 0 $n_7 n_6 n_5 n_4$	0 0 1 1 $n_3 n_2 n_1 n_0$	4 3 n	2	2	(A) ← (A) V n				The logical sum of the contents of register A and data n, is stored in register A.
	ORL A, Rr	0 1 0 0	1 $r_2 r_1 r_0$	4 8 + r	1	1	(A) ← (A) V (Rr) r = 0 ~ 7				The logical sum of the contents of register A and the contents of register R_r is stored in register A.
	ORL A, @Rr	0 1 0 0	0 0 0 r_0	4 0 + r	1	1	(A) ← (A) V (M(Rr)) r = 0 ~ 1				The logical sum of the contents of register A and the contents of memory location, of the current page, whose address is in register R_r, is stored in register A.
	XRL A, #n	1 1 0 1 $n_7 n_6 n_5 n_4$	0 0 1 1 $n_3 n_2 n_1 n_0$	D 3 n	2	2	(A) ← (A) ∀ n				The exclusive OR of the contents of register A and data n, is stored in register A.
	XRL A, Rr	1 1 0 1	1 $r_2 r_1 r_0$	D 8 + r	1	1	(A) ← (A) ∀ (Rr) r = 1 ~ 7				The exclusive OR of the contents of register A and the contents of register R_r is stored in register A.
	XRL A, @Rr	1 1 0 1	0 0 0 r_0	D 0 + r	1	1	(A) ← (A) ∀ (M(Rr)) r = 0 ~ 1				The exclusive OR of the contents of register A and the contents of memory location, of the current page, whose address is in register R_r, is stored in register A.
	INC A	0 0 0 1	0 1 1 1	1 7	1	1	(A) ← (A) + 1				Increments the contents of register A by 1. The result is stored in register A, and the carries are unchanged.
	DEC A	0 0 0 0	0 1 1 1	0 7	1	1	(A) ← (A) − 1				Decrements the contents of register A by 1. The result is stored in register A, and the carries are unchanged.
	CLR A	0 0 1 0	0 1 1 1	2 7	1	1	(A) ← 0				Clears the contents of register A, resets to 0.
	CPL A	0 0 1 1	0 1 1 1	3 7	1	1	(A) ← $(\bar{A})$				Forms 1's complement of register A, and stores it in register A.
	DA A	0 1 0 1	0 1 1 1	5 7	1	1	(A) ← (A) 10 Hexadecimal	○	○	1	The contents of register A is converted to binary coded decimal notion, and it is stored in register A. If the contents of register A are more than 99 the carry flags are set to 1 otherwise they are reset to 0.
	SWAP A	0 1 0 0	0 1 1 1	4 7	1	1	$(A_4 \sim A_7) \longleftrightarrow (A_0 \sim A_3)$				Exchanges the contents of bits 0~3 of register A with the contents of bits 4~7 of register A.
	RL A	1 1 1 0	0 1 1 1	E 7	1	1	$(A_{n+1}) \leftarrow (A_n)$ $(A_0) \leftarrow (A_7)$ n = 0 ~ 6				Shifts the contents of register A left one bit. A_7 the MSB is rotated to A_0 the LSB.
	RLC A	1 1 1 1	0 1 1 1	F 7	1	1	$(A_{n+1}) \leftarrow (A_n)$ $(A_0) \leftarrow (C)$ $(C) \leftarrow (A_7)$ n = 0 ~ 6	○			Shifts the contents of register A left one bit. A_7 the MSB is shifted to the carry flag and the carry flag is shifted to A_0 the LSB.
	RR A	0 1 1 1	0 1 1 1	7 7	1	1	$(A_n) \leftarrow (A_{n+1})$ $(A_7) \leftarrow (A_0)$ n = 0 ~ 6				Shifts the contents of register A right one bit. A_0 the LSB is rotated to A_7 the MSB.
	RRC A	0 1 1 0	0 1 1 1	6 7	1	1	$(A_n) \leftarrow (A_{n+1})$ $(A_7) \leftarrow (C)$ $(C) \leftarrow (A_0)$ n = 0 ~ 6	○			Shifts the contents of register A right one bit. A_0 the LSB is shifted to the carry flag and the carry flag is shifted to A_7 the MSB.
Register arithmetic	INC Rr	0 0 0 1	1 $r_2 r_1 r_0$	1 8 + r	1	1	(Rr) ← (Rr) + 1 r = 0 ~ 7				Increments the contents of register R_r by 1. The result is stored in register R_r and the carries are unchanged.
	INC @Rr	0 0 0 1	0 0 0 r_0	1 0 + r	1	1	(M(Rr)) ← (M(Rr)) + 1 r = 0 ~ 1				Increments the contents of the memory location, of the current page, whose address is in register R_r by 1. Register R_r uses bit 0~5.
	DEC Rr	1 1 0 0	1 $r_2 r_1 r_0$	C 8 + r	1	1	(Rr) ← (Rr) − 1 r = 0 ~ 7				Decrements the contents of register R_r by 1. The result is stored in register R_r and the carries are unchanged.

6

FUNCTION OF MELPS 8-48 MICROCOMPUTERS

Type	Mnemonic	Instruction code $D_7 D_6 D_5 D_4$	Instruction code $D_3 D_2 D_1 D_0$	Hexa-decimal	Bytes	Cycles	Function	Effected carry C	Effected carry AC	Note	Description
Jump	JBb m	$b_7 b_6 b_5$ 1 $m_7 m_6 m_5 m_4$	0 0 1 0 $m_3 m_2 m_1 m_0$	1 2 + b×2 m	2	2	$(A_b)=1$ then $(PC_0 \sim PC_7) \leftarrow m$ $(A_b)=0$ then $(PC) \leftarrow (PC)+2$ $b_7 b_6 b_5 = 0 \sim 7$				Jumps to address m of the current page when bit b of register A is 1. Executes the next instruction when bit b of register A is 0.
Jump	JTF m	0 0 0 1 $m_7 m_6 m_5 m_4$	0 1 1 0 $m_3 m_2 m_1 m_0$	1 6 m	2	2	$(TF)=1$ then $(PC_0 \sim PC_7) \leftarrow m$ $(TF)=0$ then $(PC) \leftarrow (PC)+2$				Jumps to address m of the current page when the overflow flag of the timer is 1 otherwise the next instruction is executed. Flag is cleared after executing.
Jump	JMP m	$m_{10} m_9 m_8$ 0 $m_7 m_6 m_5 m_4$	0 1 0 0 $m_3 m_2 m_1 m_0$	0 4 + $(m_8 \sim m_{10})$ ×2 m	2	2	$(PC_8 \sim PC_{10}) \leftarrow m_8 \sim m_{10}$ $(PC_0 \sim PC_7) \leftarrow m_0 \sim m_7$ $(PC_{11}) \leftarrow (MBF)$				Jumps to address m on page m_{10} m_9 m_8 in the memory bank indicated by MBF.
Jump	JMPP @A	1 0 1 1	0 0 1 1	B 3	1	2	$(PC_0 \sim PC_7) \leftarrow (M(A))$				Jumps to the memory location, of the current page, whose address is in register A. But when the instruction executed was in address 255, jumps to next page.
Jump	DJNZ Rr, m	1 1 1 0 $m_7 m_6 m_5 m_4$	1 $r_2 r_1 r_0$ $m_3 m_2 m_1 m_0$	E 8 + r m	2	2	$(Rr) \leftarrow (Rr)-1$ $r=0 \sim 7$ $(Rr) \neq 0$ then $(PC_0 \sim PC_7) \leftarrow m$ $(Rr)=0$ then $(PC) \leftarrow (PC)+2$				Decrements the contents of register R_r by 1. Jumps to address m of the current page when the result is not 0, otherwise the next instruction is executed.
Jump	JC m	1 1 1 1 $m_7 m_6 m_5 m_4$	0 1 1 0 $m_3 m_2 m_1 m_0$	F 6 m	2	2	$(C)=1$ then $(PC_0 \sim PC_7) \leftarrow m$ $(C)=0$ then $(PC) \leftarrow (PC)+2$				Jumps to address m of the current page if the carry flag C is 1, otherwise the next instruction is executed.
Jump	JNC m	1 1 1 0 $m_7 m_6 m_5 m_4$	0 1 1 0 $m_3 m_2 m_1 m_0$	E 6 m	2	2	$(C)=0$ then $(PC_0 \sim PC_7) \leftarrow m$ $(C)=1$ then $(PC) \leftarrow (PC)+2$				Jumps to address m of the current page if the carry flag C is 0, otherwise the next instruction is executed.
Jump	JZ m	1 1 0 0 $m_7 m_6 m_5 m_4$	0 1 1 0 $m_3 m_2 m_1 m_0$	C 6 m	2	2	$(A)=0$ then $(PC_0 \sim PC_7) \leftarrow m$ $(A) \neq 0$ then $(PC) \leftarrow (PC)+2$				Jumps to address m of the current page when the contents of register A are 0, otherwise the next instruction is executed.
Jump	JNZ m	1 0 0 1 $m_7 m_6 m_5 m_4$	0 1 1 0 $m_3 m_2 m_1 m_0$	9 6 m	2	2	$(A) \neq 0$ then $(PC_0 \sim PC_7) \leftarrow m$ $(A)=0$ then $(PC) \leftarrow (PC)+2$				Jumps to address m of the current page when the contents of register A are not 0, otherwise the next instruction is executed.
Jump	JTO m	0 0 1 1 $m_7 m_6 m_5 m_4$	0 1 1 0 $m_3 m_2 m_1 m_0$	3 6 m	2	2	$(T_0)=1$ then $(PC_0 \sim PC_7) \leftarrow m$ $(T_0)=0$ then $(PC) \leftarrow (PC)+2$				Jumps to address m of the current page when flag T_0 is 1 otherwise the next instruction is executed.
Jump	JNTO m	0 0 1 0 $m_7 m_6 m_5 m_4$	0 1 1 0 $m_3 m_2 m_1 m_0$	2 6 m	2	2	$(T_0)=0$ then $(PC_0 \sim PC_7) \leftarrow m$ $(T_0)=1$ then $(PC) \leftarrow (PC)+2$				Jumps to address m of the current page when flag T_0 is 0, otherwise the next instruction is executed.
Jump	JT1 m	0 1 0 1 $m_7 m_6 m_5 m_4$	0 1 1 0 $m_3 m_2 m_1 m_0$	5 6 m	2	2	$(T_1)=1$ then $(PC_0 \sim PC_7) \leftarrow m$ $(T_1)=0$ then $(PC) \leftarrow (PC)+2$				Jumps to address m of the current page when flag T_1 is 1. otherwise the next instruction is executed.
Jump	JNT1 m	0 1 0 0 $m_7 m_6 m_5 m_4$	0 1 1 0 $m_3 m_2 m_1 m_0$	4 6 m	2	2	$(T_1)=0$ then $(PC_0 \sim PC_7) \leftarrow m$ $(T_1)=1$ then $(PC) \leftarrow (PC)+2$				Jumps to address m of the current page when flag T_1 is 0. otherwise the next instruction is executed.
Jump	JFO m	1 0 1 1 $m_7 m_6 m_5 m_4$	0 1 1 0 $m_3 m_2 m_1 m_0$	B 6 m	2	2	$(F_0)=1$ then $(PC_0 \sim PC_7) \leftarrow m$ $(F_0)=0$ then $(PC) \leftarrow (PC)+2$				Jumps to address m of the current page when flag F_0 is 1.
Jump	JF1 m	0 1 1 1 $m_7 m_6 m_5 m_4$	0 1 1 0 $m_3 m_2 m_1 m_0$	7 6 m	2	2	$(F_1)=1$ then $(PC_0 \sim PC_7) \leftarrow m$ $(F_1)=0$ then $(PC) \leftarrow (PC)+2$				Jumps to address m of the current page when flag F_1 is 1.
Flag control	CLR C	1 0 0 1	0 1 1 1	9 7	1	1	$(C) \leftarrow 0$	○			Clears the carry flag C, resets it to 0. AC is not affected.
Flag control	CPL C	1 0 1 0	0 1 1 1	A 7	1	1	$(C) \leftarrow (\bar{C})$	○			Complements the carry flag C. AC is not affected.
Flag control	CLR F_0	1 0 0 0	0 1 0 1	8 5	1	1	$(F_0) \leftarrow 0$				Clears the flag F_0, resets it to 0.
Flag control	CPL F_0	1 0 0 1	0 1 0 1	9 5	1	1	$(F_0) \leftarrow (\bar{F}_0)$				Complements the flag F_0.
Flag control	CLR F_1	1 0 1 0	0 1 0 1	A 5	1	1	$(F_1) \leftarrow 0$				Clears flag F_1 resets it to 0.
Flag control	CPL F_1	1 0 1 1	0 1 0 1	B 5	1	1	$(F_1) \leftarrow (\bar{F}_1)$				Complements the flag F_1.

MITSUBISHI ELECTRIC

FUNCTION OF MELPS 8-48 MICROCOMPUTERS

Type	Mnemonic	Instruction code $D_7D_6D_5D_4$ $D_3D_2D_1D_0$	Hexa-decimal	Bytes	Cycles	Function	Effected carry C	AC	Note	Description
Subroutine call	**CALL m**	$m_{10}m_9m_8$ 1 0 1 0 0 $m_7m_6m_5m_4$ $m_3m_2m_1m_0$	1 4 + $(m_8 \sim m_{10})$ ×2 m	2	2	$((SP)) \leftarrow (PC)\ (PSW_4 \sim PSW_7)$ $(SP) \leftarrow (SP)+1$ $(PC_{0\sim10}) \leftarrow m$ $(PC_{11}) \leftarrow MBF$				Calls subroutine from address m. The program counter and the 4 high-order bits of the PSW are stored in the address indicated by the stack pointer (SP). The SP is incremented by 1 and m is transferred to $PC_0 \sim PC_{10}$ and the MBF is transferred to PC_{11}.
	RET	1 0 0 0 0 0 1 1	8 3	1	2	$(SP) \leftarrow (SP)-1$ $(PC) \leftarrow ((SP))$				The SP is decremented by 1. The program counter is restored to the saved setting in the stack indicated by the stack pointer. The PSW is not changed and interrupt disabled is maintained.
	RETR	1 0 0 1 0 0 1 1	9 3	1	2	$(SP) \leftarrow (SP)-1$ $(PC)\ (PSW_4 \sim PSW_7) \leftarrow ((SP))$				The SP is decremented by 1. The program counter and the 4 high-order bits of the PSW are restored with the saved data in the stack indicated by the stack pointer. The interrupt becomes enabled after the execution is completed.
Input/Output	**IN A, Pp**	0 0 0 0 1 0 p_1p_0	0 8 + p	1	2	$(A) \leftarrow (Pp)$ $p=1\sim2$				Loads the contents of P_p to register A.
	OUTL Pp, A	0 0 1 1 1 0 p_1p_0	3 8 + p	1	2	$(Pp) \leftarrow (A)$ $p=1\sim2$				Output latches the contents of register A to P_p
	ANL Pp, #n	1 0 0 1 1 0 p_1p_0 $n_7n_6n_5n_4$ $n_3n_2n_1n_0$	9 8 + p n	2	2	$(Pp) \leftarrow (Pp) \wedge n$ $p=1\sim2$				Logical ANDs the contents of P_p and data n. Outputs the result to P_p
	ORL Pp, #n	1 0 0 0 1 0 p_1p_0 $n_7n_6n_5n_4$ $n_3n_2n_1n_0$	8 8 + p n	2	2	$(Pp) \leftarrow (Pp) \vee n$ $p=1\sim2$				Logical ORs the contents of P_p and data n. Outputs the result to P_p
	INS A, BUS	0 0 0 0 1 0 0 0	0 8	1	2	$(A) \leftarrow (BUS)$				Enters the contents of data bus (port 0) to register A
	OUTL BUS, A	0 0 0 0 0 0 1 0	0 2	1	2	$(BUS) \leftarrow (A)$				Output latches the contents of register A data to data bus (port 0)
	ANL BUS, #n	1 0 0 1 1 0 0 0 $n_7n_6n_5n_4$ $n_3n_2n_1n_0$	9 8 n	2	2	$(BUS) \leftarrow (BUS) \wedge n$				Logical ANDs the contents of data bus (port 0) and data n. Outputs the result to data bus (port 0)
	ORL BUS, #n	1 0 0 0 1 0 0 0 $n_7n_6n_5n_4$ $n_3n_2n_1n_0$	8 8 n	2	2	$(BUS) \leftarrow (BUS) \vee n$				Logical ORs the contents of data bus (port 0) and data n. Outputs the result to data bus (port 0)
	MOVD A, Pp	0 0 0 0 1 1 p_1p_0	0 C + p_1p_0	1	2	$(A_0 \sim A_3) \leftarrow (Pp_0 \sim Pp_3)$ $(A_4 \sim A_7) \leftarrow 0$ $p=4\sim7$				Inputs the contents of P_p to the low-order 4 bits of register A and inputs 0 to the high-order 4 bits of register A. P_p's used for multiplying 8243 ports are $P_4 \sim P_7$. Correspondence to p_2, p_1 is shown below. P4 ··· p_1p_2=00 P5 ··· p_1p_2=01 P6 ··· p_1p_2=10 P7 ··· p_1p_2=11
	MOVD Pp, A	0 0 1 1 1 1 p_1p_0	3 C + p_1p_0	1	2	$(Pp_0 \sim Pp_3) \leftarrow (A_0 \sim A_3)$ $p=4\sim7$				Outputs the low-order 4 bits of register A to P_p.
	ANLD Pp, A	1 0 0 1 1 1 p_1p_0	9 C + p_1p_0	1	2	$(Pp_0 \sim Pp_3) \leftarrow (Pp_0 \sim Pp_3) \wedge (A_0 \sim A_3)$ $p=4\sim7$				Logical ANDs the 4 low-order bits of register A and the contents of P_p. P_p contains the result.
	ORLD Pp, A	1 0 0 0 1 1 p_1p_0	8 C + p_1p_0	1	2	$(Pp_0 \sim Pp_3) \leftarrow (Pp_0 \sim Pp_3) \vee (A_0 \sim A_3)$ $p=4\sim7$				Logical ORs the 4 low-order bits of register A and the contents of P_p. P_p contains the result.

FUNCTION OF MELPS 8-48 MICROCOMPUTERS

Type \ Item	Mnemonic	Instruction code $D_7\ D_6\ D_5\ D_4\ \ D_3\ D_2\ D_1\ D_0$	Hexa-decimal	Bytes	Cycles	Function	Effected carry C	AC	Note	Description
Control	EN I	0 0 0 0 0 1 0 1	0 5	1	1	(INTF) ← 1				Enables outside interrupt.
	DIS I	0 0 0 1 0 1 0 1	1 5	1	1	(INTF) ← 0				Disables outside interrupt.
	SEL RB_0	1 1 0 0 0 1 0 1	C 5	1	1	(BS) ← 0				Selects working register bank 0.
	SEL RB_1	1 1 0 1 0 1 0 1	D 5	1	1	(BS) ← 1				Selects working register bank 1.
	SEL MB_0	1 1 1 0 0 1 0 1	E 5	1	1	(MBF) ← 0				Selects memory bank 0.
	SEL MB_1	1 1 1 1 0 1 0 1	F 5	1	1	(MBF) ← 1				Selects memory bank 1.
	ENTO CLK	0 1 1 1 0 1 0 1	7 5	1	1					Enables output of clock signal from terminal T_0
Timer/event counter control	MOV A, T	0 1 0 0 0 0 1 0	4 2	1	1	(A) ← (T)				Transfers the contents of timer/event counter to register A.
	MOV T, A	0 1 1 0 0 0 1 0	6 2	1	1	(T) ← (A)				Transfers the contents of register A to timer/event counter.
	STRT T	0 1 0 1 0 1 0 1	5 5	1	1					Starts timer operation of timer/event counterm. Minimum count cycle is 80μs.
	STRT CNT	0 1 0 0 0 1 0 1	4 5	1	1					Starts operation as event counter of time/event counter. Counts up when terminated T_1 changes to input high-level for input low-level. Minimum count cycle is 7.5μs.
	STOP TCNT	0 1 1 0 0 1 0 1	6 5	1	1					Stops operation of timer or event counter.
	EN TCNTI	0 0 1 0 0 1 0 1	2 5	1	1	(TCNTF) ← 1				Enables interrupt of timer/event counter.
	DIS TCNTI	0 0 1 1 0 1 0 1	3 5	1	1	(TCNTF) ← 0				Disables interrupt of timer/event counter. Resets interrupt flip-flop of CPU which is set during the CPU stands-by. Timer overflow flag isn't affected.
Misc.	NOP	0 0 0 0 0 0 0 0	0 0	1	1					No operation. Execution time is 1 cycle.

Note 1: Executing an instruction may produce a carry (overflow or underflow). The carry may be disregarded (lost) or it may be transferred to C/AC (saved). The saving of a carry is not shown in the function equations, but is instead shown in the carry columns C and AC. The detail affection of carries for instructions ADD ADDC and DA is as follows:

(C) ← 1 at overflow of the accumulator is produced.
(C) ← 0 at no overflow of the accumulator is produced.
(AC) ← 1 at overflow of the bit 3 of the accumulator.
(AC) ← 0 at no overflow.

2: The contents of $ST_4 \sim ST_7$ is read when the host computer reads the status of M5L8041A-XXXP.

FUNCTION OF MELPS 8-48 MICROCOMPUTERS

Symbol	Meaning	Symbol	Meaning
A	8-bit register (accumulator)	PC	Program counter
$A_0 \sim A_3$	The low-order 4 bits of the register A	$PC_0 \sim PC_7$	The low-order 8 bits of the program counter
$A_4 \sim A_7$	The high-order 4 bits of the register A	$PC_8 \sim PC_{10}$	The high-order 3 bits of the program counter
$A_0 \sim A_n, A_{n+1}$	The bits of the register A	PSW	Program status word
b	The value of the bits 5~7 of the first byte machine code		
$b_7b_6b_5$	The bits 5~7 of the first byte machine code	Rr	Rcgister designator
BS	Register bank select	r	Register number
BUS	Corresponds to the port 0 (bus I/O port)	r_0	The value of bit 0 of the machine code
		$r_2r_1r_0$	The value of bits 0~2 of the machine code
AC	Auxiliary carry flag	$s_2s_1s_0$	The value of bits 0~2 of the stack pointer
C	Carry flag	SP	Stack pointer
DBB	Data bus buffer	$ST_4 \sim ST_7$	Bits 4~7 of the status register
		STS	System status
F_0	Flag 0	T	Timer/event counter
F_1	Flag 1	T_0	Test pin 0
INTF	Interrupt flag	T_1	Test pin 1
IBF	Input buffer full flag	TCNTF	Timer/event counter overflow interrupt flag
m	The value of the 11-bit address	TF	Timer flag
$m_7m_6m_5m_4m_3m_2m_1m_0$	The second byte (low-order 8 bits) machine code of the 11-bit address		
$m_{10}m_9\ m_8$	The bits 5~7 of the first byte (high-order 3 bits) machine code ofothe 11-bit address	#	Symbol to indicate the immediate data
(M (A))	The content of the memory location addressed by the register A	@	Symbol to indicate the cuntent of the memory location address by the register
(M (Rr))	The content of rhe memory location addressed by the register Rr		
(Mx(Rr))	The content of the external memory location addressed by the register Rr	←	Shows direction of data flow
MBF	Memory bank flag	←→	Exchanges the contents of data
n	The value of the immediate data	( )	Contents of register, memory location or flag
$n_7n_6n_5n_4n_3n_2n_1n_0$	The immediate data of the second byte machine code	$\wedge$	Logical AND
OBF	Output buffer full flag	$\vee$	Inclusive OR
		$\veebar$	Exclusive OR
p	Port number	—	Negation
Pp	Port designator	○	Content of flag is set or reset after execution
p_1p_0	The bits of the machine code corresponding to the port number		

6

FUNCTION OF MELPS 8-48 MICROCOMPUTERS

Instruction Code List

$D_3 \sim D_0$ \ $D_7 \sim D_4$	Hexadecimal	0000 0	0001 1	0010 2	0011 3	0100 4	0101 5	0110 6	0111 7	1000 8	1001 9	1010 A	1011 B	1100 C	1101 D	1110 E	1111 F
0000	0	NOP	INC @R0	XCH A, @R0	XCHD A, @R0	ORL A, @R0	ANL A, @R0	ADD A, @R0	ADDC A, @R0	MOVX A, @R0	MOVX @R0, A	MOV @R0, A	MOV @R0, #n		XRL A, @R0		MOV A, @R0
0001	1		INC @R1	XCH A, @R1	XCHD A, @R1	ORL A, @R1	ANL A, @R1	ADD A, @R1	ADDC A, @R1	MOVX A, @R1	MOVX @R1, A	MOV @R1, A	MOV @R1, #n		XRL A, @R1		MOV A, @R1
0010	2	OUTL BUS,A	JB0 m		JB1 m	MOV A, T	JB2 m	MOV T, A	JB3 m		JB4 m		JB5 m		JB6 m		JB7 m
0011	3	ADD A, #n	ADDC A, #n	MOV A, #n		ORL A, #n	ANL A, #n			RET	RETR	MOVP A, @A	JMPP @A		XRL A, #n	MOVP3 A, @A	
0100	4	JMP 0XX	CALL 0XX	JMP 1XX	CALL 1XX	JMP 2XX	CALL 2XX	JMP 3XX	CALL 3XX	JMP 4XX	CALL 4XX	JMP 5XX	CALL 5XX	JMP 6XX	CALL 6XX	JMP 7XX	CALL 7XX
0101	5	EN I	DIS I	EN TCNTI	DIS TCNTI	STRT CNT	STRT T	STOP TCNT	ENTO CLK	CLR F0	CPL F0	CLR F1	CPL F1	SEL RB0	SEL RB1	SEL MB0	SEL MB1
0110	6		JTF m	JNT0 m	JT0 m	JNT1 m	JT1 m		JF1 m	JNI m	JNZ m		JF0 m	JZ m		JNC m	JC m
0111	7	DEC A	INC A	CLR A	CPL A	SWAP A	DA A	RRC A	RR A		CLR C	CPL C		MOV A,PSW	MOV PSW, A	RL A	RLC A
1000	8	INS A,BUS	INC R0	XCH A, R0		ORL A, R0	ANL A, R0	ADD A, R0	ADDC A, R0	ORL BUS, #n	ANL BUS, #n	MOV R0, A	MOV R0, #n	DEC R0	XRL A, R0	DJNZ R0, m	MOV A, R0
1001	9	IN A, P1	INC R1	XCH A, R1	OUTL P1, A	ORL A, R1	ANL A, R1	ADD A, R1	ADDC A, R1	ORL P1, #n	ANL P1, #n	MOV R1, A	MOV R1, #n	DEC R1	XRL A, R1	DJNZ R1, m	MOV A, R1
1010	A	IN A, P2	INC R2	XCH A, R2	OUTL P2, A	ORL A, R₂	ANL A, R2	ADD A, R2	ADDC A, R2	ORL P2, #n	ANL P2, #n	MOV R2, A	MOV R2, #n	DEC R2	XRL A, R2	DJNZ R2, m	MOV A, R2
1011	B		INC R3	XCH A, R3		ORL A, R3	ANL A, R3	ADD A, R3	ADDC A, R3			MOV R3, A	MOV R3, #n	DEC R3	XRL A, R3	DJNZ R3, m	MOV A, R3
1100	C	MOVD A, P4	INC R4	XCH A, R4	MOVD P4, A	ORL A, R4	ANL A, R4	ADD A, R4	ADDC A, R4	ORLD P4, A	ANLD P4, A	MOV R4, A	MOV R4, #n	DEC R4	XRL A, R4	DJNZ R4, m	MOV A, R4
1101	D	MOVD A, P5	INC R5	XCH A, R5	MOVD P5, A	ORL A, R5	ANL A, R5	ADD A, R5	ADDC A, R5	ORLD P5, A	ANLD P5, A	MOV R5, A	MOV R5, #n	DEC R5	XRL A, R5	DJNZ R5, m	MOV A, R5
1110	E	MOVD A, P6	INC R6	XCH A, R6	MOVD P6, A	ORL A, R6	ANL A, R6	ADD A, R6	ADDC A, R6	ORLD P6, A	ANLD P6, A	MOV P6, A	MOV R6, #n	DEC R6	XRL A, R6	DJNZ R6, m	MOV A, R6
1111	F	MOVD A, P7	INC R7	XCH A, R7	MOVD P7, A	ORL A, R7	ANL A, R7	ADD A, R7	ADDC A, R7	ORLD P7, A	ANLD P7, A	MOV R7, A	MOV R7, #n	DEC R7	XRL A, R7	DJNZ R7, m	MOV A, R7

Legend: dark shading = 2-byte, 2-cycle instruction; light shading = 1-byte, 2-cycle instruction

DEVELOPMENT OF MASK-PROGRAMMABLE ROMs

GENERAL INFORMATION

This information explains how to specify the object program for the automatic design system for mask ROMs. This system for mask ROM production has been developed to accept a customer's object program specifications for the automatic design system for a mask ROM.

The main segments of the automatic design system are:

1. The plotter instructions for mask production.
2. A check list for verifing that the customer's specifications have been met.
3. A test program to assure that the production ROMs meet specifications.

An EPROM in which a program is stored is used for a customer's specifications. A separate (set of) EPROM(s) should be produced for each object program.

Three sets of EPROM(s) should be supplied with the confirmation material.

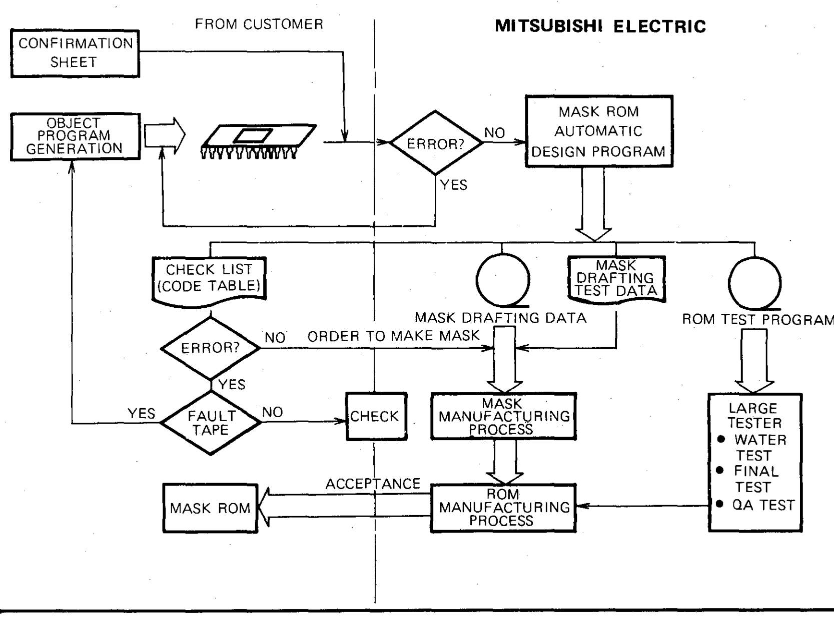
EPROM SPECIFICATIONS

1. The Mitsubishi M5L2708K, M5L2716K, M5L2732K or M5L8748S are standard, but Intel 2708, 2716, 2732, 8748 or equivalent devices may be used.
2. The high-level data of both data outputs and address inputs of the supplied EPROM will be programmed as '1', and low-level as '0'.
3. All the data stored in the EPROM are considered as valid and processed to make masks.

ITEMS TO CONFIRM FOR ORDERING

1. Specify the type number M5L8048-XXXP or M5L8049-XXXP. The 3-digit number XXX will be assigned by Mitsubishi.
2. Cleary indicate the type number of EPROM and address designation letter symbols A and B on the supplied EPROMs.

MASK ROM DEVELOPMENT FLOW CHART



MELPS 8-48 MASK-PROGRAMMABLE ROM CONFIRMATION MATERIAL

SINGLE-CHIP 8-BIT MICROCOMPUTERS M5L8048-XXXP, M5L8049-XXXP

MITSUBISHI ELECTRIC

Customer Company name ________________ Company address ________________ Tel ________ Company contact ________________ Date ________	Signature Prepared Approved

The single-chip microcomputer type number to order and the type of EPROMs to be supplied should be specified by checking √ in the boxes. Three sets of EPROMs should be supplied.

Single-chip microcomputer type number \ EPROM type number	2708	2716	2732	8748
□ M5L8048 − XXXP	□ A(000_{16} ~ $3FF_{16}$)	□ A(000_{16} ~ $3FF_{16}$)	□ A(000_{16} ~ $3FF_{16}$)	□ A(000_{16} ~ $3FF_{16}$)
□ M5L8049 − XXXP	□ A(000_{16} ~ $3FF_{16}$) B(400_{16} ~ $7FF_{16}$)	□ A(000_{16} ~ $7FF_{16}$)	□ A(000_{16} ~ $7FF_{16}$)	−

Note 1 The high-level data of both data outputs and address inputs of the supplied EPROM will be programmed as '1', and low-level as '0'.
2 Cleary indicate the type number of EPROMs and address designation letter symbols A and B on the supplied EPROMs.
3 The data of the addresses in parentheses on the EPROM are programmed onto the ROM.
4 The data from each PROM in the set is compared and if 2 of the 3 are equal, the equal value will be programmed into the ROM. When the 3 values are different programming is halted and the customer is notified of the error. The error report will show the address and data.

CUSTOMER'S IDENTIFICATION MARK

If you require a special identification mark, please specify in the following format.

Mitsubishi IC type number

Note 5 A mark field should start with the box at the extreme right
6 The identification mark should be no more than 12 characters consisting of alphanumeric characters (except J.I. and O) or dashes

COMMENTS

SINGLE-CHIP 8-BIT MICROCOMPUTER

DESCRIPTION

The M5L8048-XXXP and M5L8035LP are 8-bit parallel microcomputer fabricated on a single chip using high-speed N-channel silicon-gate ED-MOS technology.

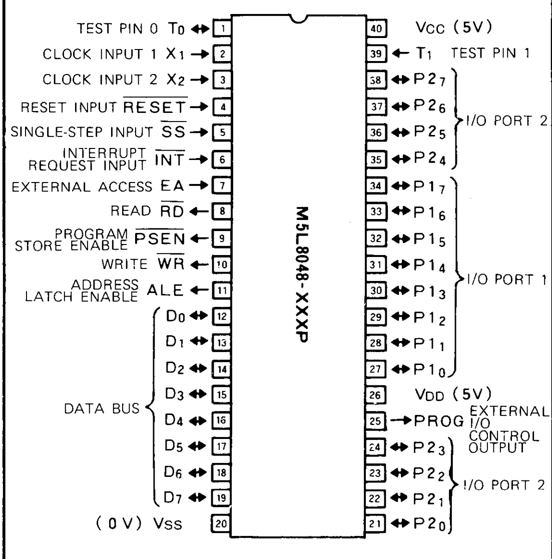
FEATURES

- Single 5V power supply
- Instruction cycle .... 2.5μs (min)
- Basic machine instructions: 96
  - 1-byte instructions: 68
  - 2-byte instructions: 28
- Direct addressing .... up to 4096 bytes
- Internal ROM .... 1024 bytes (for M5L8048-XXXP only)
- Internal RAM .... 64 bytes
- Built-in timer/event counter .... 8 bits
- I/O Ports .... 27 lines
- Easily expandable Memory and I/O
- Subroutine nesting .... 8 levels
- External and timer/event counter interrupt . 1 level each
- Low power standby mode
- External RAM .... 256 bytes
- Interchangeable with Intel's P8048 and P8035L in pin configuration and electrical characteristics

APPLICATION

- Control processor or CPU for a wide variety of applications

PIN CONFIGURATION (TOP VIEW)

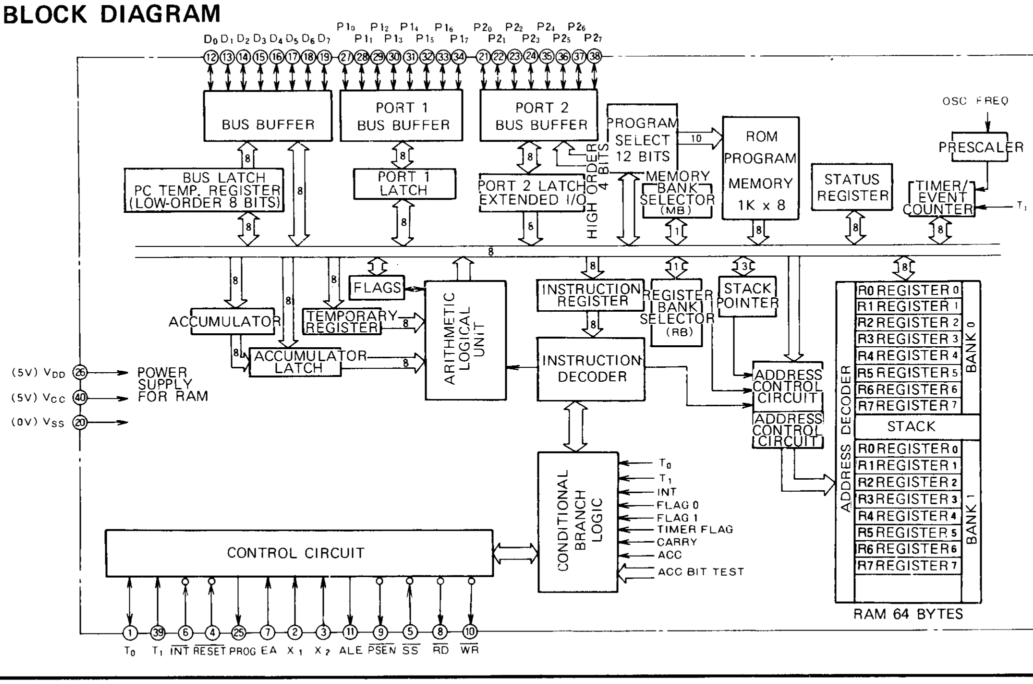


Outline 40P1

FUNCTION

The M5L8048-XXXP and M5L8035LP are integrated 8-bit CPU, with memory (ROM, RAM) and timer/event counter interrupt all contained on a single chip.

BLOCK DIAGRAM



PIN DESCRIPTION

Pin	Name	Input or Output	Function
V_{SS}	Ground		Normally connected to ground (0V).
V_{CC}	Main power supply		Connected to 5V power supply.
V_{DD}	Power supply		① Connected to 5V power supply. ② Used for memory hold when V_{CC} is cut.
PROG	Program	Output	Strobe signal for M5L8243P I/O Expander.
$P1_0 \sim P1_7$	Port 1	Input/output	Quasi-bidirectional port. When used as an input port, FF_{16} must first be output to this port. After reset, when not used as an output port nothing can be output.
$P2_0 \sim P2_7$	Port 2	Input/output	① The same as port 1.
		Output	② $P2_0 \sim P2_3$ output the high-order 4 bits of the program counter when using external program memory.
		Input/output	③ $P2_0 \sim P2_3$ serve as a 4-bit I/O expander bus for the M5L8243P.
$D_0 \sim D_7$	Data bus	Input/output	① Provides true bidirectional bus transfer of instructions and data between the CPU and external memory. Synchronizing is done with signals $\overline{RD}/\overline{WR}$. The output data is latched.
			② When using external program memory the output of the low-order 8 bits of the program counter are synchronized with ALE. After that the transfer of the instruction code or data from external program memory is synchronized with $\overline{PSEN}$.
			③ The output of addresses for data using external data memory is synchronized with ALE. After that the transfer of data with the external data memory is synchronized with $\overline{RD}/\overline{WR}$. (MOVX A, @Rr and MOVX @Rr, A)
T_0	Test pin 0	Input	① Control signal from an external source for conditional jumping in a program. Jumping is dependent on external conditions. (JT0 m and JNT0 m)
		Output	② Used for outputting the internal clock signal. (ENT0 CLK)
T_1	Test pin 1	Input	① Control signal from an external source for conditional jumping in a program. Jumping is dependent on external conditions. (JT1 m and JNT1 m) ② When enabled event signals are transferred to the timer/event counter. (STRT CNT)
$\overline{INT}$	Interrupt	Input	① Control signal from an external source for conditional jumping in a program. Jumping is dependent on external conditions. (JN1 m) ② Used for external interrupt to CPU.
$\overline{RD}$	Read control	Output	Read control signal used when the CPU requests data from external data memory or external devices to be transferred to the data bus. (MOVX A, @Rr and INS A, BUS)
$\overline{WR}$	Write control	Output	Write control signal used when the CPU sends data through the data bus to external data memory or external device. (MOVX @R, A and OUTL BUS, A)
$\overline{RESET}$	Reset	Input	Control used to initialize the CPU.
ALE	Address latch enable	Output	A signal used for latching the address on the data bus. An ALE signal occurs once during each cycle.
$\overline{PSEN}$	Program store enable	Output	Strobe signal to fetch external program memory.
$\overline{SS}$	Single step	Input	Control signal used in conjunction with ALE to stop the CPU through each instruction, in the single step mode.
EA	External access	Input	① Normally maintained at 0V. ② When the level is raised to 5V, external memory will be accessed even when the address is less than 400_{16}(1024). The M5L8035LP is raised to 5V.
X_1, X_2	Crystal inputs	Input	External crystal oscillator or RC circuit input for generating internal clock signals. An external clock signal can be input through X_1 or X_2.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.5∼7	V
V_{DD}	Supply voltage		−0.5∼7	V
V_I	Input voltage		−0.5∼7	V
V_O	Output voltage		−0.5∼7	V
Pd	Power dissipation	Ta=25℃	1.5	W
Topr	Operating free-air temperature range		0∼70	℃
Tstg	Storage temperature range		−65∼150	℃

RECOMMENDED OPERATING CONDITIONS (Ta=0∼70℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{DD}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage		0		V
V_{IH1}	High-level input voltage, except X1, X2 and $\overline{RESET}$	2		V_{CC}	V
V_{IH2}	High-level input voltage, except X1, X2 and $\overline{RESET}$	3.8		V_{CC}	V
V_{IL}	Low-level input voltage	−0.5		0.8	V

ELECTRICAL CHARACTERISTICS (Ta=0∼70℃, V_{CC}=V_{DD}=5V±10%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OL}	Low-level output voltage, BUS, $\overline{RD}$, $\overline{WR}$, $\overline{PSEN}$, ALE	I_{OL}=2mA			0.45	V
V_{OL1}	Low-level output voltage, except the above and PROG	I_{OL}=1.6mA			0.45	V
V_{OL2}	Low-level output voltage, PROG	I_{OL}=1mA			0.45	V
V_{OH}	High-level output voltage, BUS, $\overline{RD}$, $\overline{WR}$, $\overline{PSEN}$, ALE	I_{OH}=−100μA	2.4			V
V_{OH1}	High-level output voltage, except the above	I_{OH}=−50μA	2.4			V
I_{IL}	Input leak current, T1, $\overline{INT}$	V_{SS}≦V_{IN}≦V_{CC}	−10		10	μA
I_{OL}	Output leak current, BUS, T0 high-impedance state	V_{SS}+0.45≦V_{IN}≦V_{CC}	−10		10	μA
I_{LI1}	Input current during low-level input, port	V_{IL}=0.8V		−0.2		mA
I_{LI2}	Input current during low-level input, $\overline{RESET}$, $\overline{SS}$	V_{IL}=0.8V		−0.05		mA
I_{DD}	Supply current from V_{DD}			10	20	mA
I_{DD}+I_{CC}	Supply current from V_{DD} and V_{CC}			65	135	mA

TIMING REQUIREMENTS (Ta=0∼70℃, V_{CC}=V_{DD}=5V±10%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Limits			Unit
			Min	Typ	Max	
t_C	Cycle time	t_{CY}	2.5		15.0	μs
t_h (PSEN-D)	Data hold time after $\overline{PSEN}$	t_{DR}	0		200	ns
t_h (R-D)	Data hold time after $\overline{RD}$	t_{DR}	0		200	ns
t_{SU} (PSEN-D)	Data setup time after $\overline{PSEN}$	t_{RD}			500	ns
t_{SU} (R-D)	Data setup time after $\overline{RD}$	t_{RD}			500	ns
t_{SU} (A-D)	Data setup time after address	t_{AD}			950	ns
t_{SU} (PROG-D)	Data setup time after PROG	t_{PR}			810	ns
t_h (PROG-D)	Data hold time before PROG	t_{PF}	0		150	ns

Note 1: The input voltage level of the input voltage is V_{IL}=0.45V and V_{IH}=2.4V.

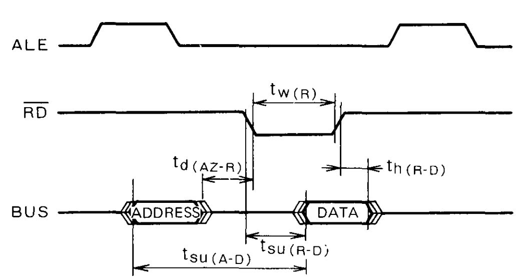
SWITCHING CHARACTERISTICS (Ta = 0 ~ 70℃, $V_{CC} = V_{DD} = 5V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Limits			Unit
			Min	Typ	Max	
$t_{w(ALE)}$	ALE pulse width	t_{LL}	400			ns
$t_{d(A-ALE)}$	Delay time, address to ALE signal	t_{AL}	120			ns
$t_{v(ALE-A)}$	Address valid time after ALE	t_{LA}	80			ns
$t_{w(PSEN)}$	$\overline{PSEN}$ pulse width	t_{CC}	700			ns
$t_{w(R)}$	$\overline{RD}$ pulse width	t_{CC}	700			ns
$t_{w(W)}$	$\overline{WR}$ pulse width	t_{CC}	700			ns
$t_{d(Q-W)}$	Delay time, data to $\overline{WR}$ signal	t_{DW}	500			ns
$t_{v(W-Q)}$	Data valid time after $\overline{WR}$	t_{WD}	120			ns
$t_{d(A-W)}$	Delay time, address to $\overline{WR}$ signal	t_{AW}	230			ns
$t_{d(AZ-R)}$	Delay time, address disable to $\overline{RD}$ signal	t_{AFC}	0			ns
$t_{d(AZ-PSEN)}$	Delay time, address disable to $\overline{PSEN}$ signal	t_{AFC}	0			ns
$t_{d(PC-PROG)}$	Delay time, port control to PROG signal	t_{CP}	110			ns
$t_{v(PROG-PC)}$	Port control valid time after PROG	t_{PC}	100			ns
$t_{p(Q-PROG)}$	Delay time, data to PROG signal	t_{DP}	250			ns
$t_{v(PROG-Q)}$	Data valid time after PROG	t_{PD}	65			ns
$t_{w(PROGL)}$	PROG low pulse width	t_{PP}	1200			ns
$t_{d(Q-ALE)}$	Delay time, data to ALE signal	t_{PL}	350			ns
$t_{v(ALE-Q)}$	Data valid time after ALE	t_{LP}	150			ns

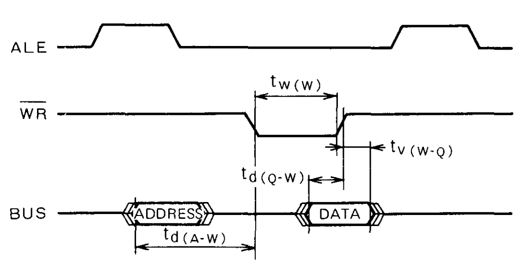
Note 2: Conditions of measurement: control output C_L=80pF
data bus output, port output C_L=150pF, t_C=2.5μS
3: Reference levels for the input/output voltages are low level=0.8V and high level=2V

TIMING DIAGRAM

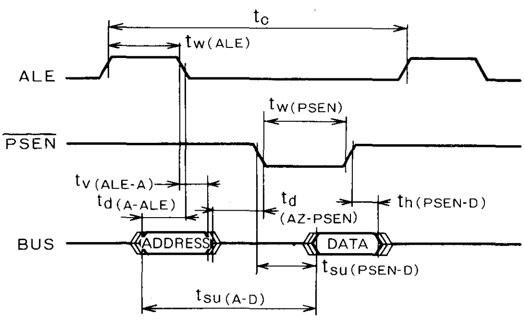
Read from External Data Memory



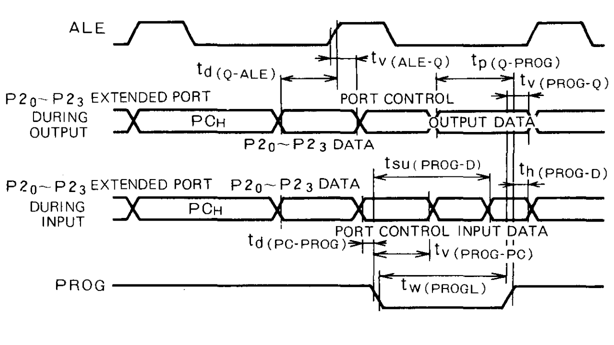
Write to External Data Memory



Instruction Fetch from External Program Memory



Port 2



M5L8049-XXXP,P-8,P-6
M5L8039P-11,P-8,P-6

SINGLE-CHIP 8-BIT MICROCOMPUTER

DESCRIPTION

The M5L8049-XXXP, P-8, P-6 and M5L8039P-11, P-8, P-6 are 8-bit parallel microcomputers fabricated on a single chip using high-speed N-channel silicon gate ED-MOS technology.

Speed \ ROM Type	Internal ROM Type	External ROM Type
11 MHz Type	M5L8049-XXXP	M5L8039P-11
8 MHz Type	M5L8049-XXXP-8	M5L8039P-8
6 MHz Type	M5L8049-XXXP-6	M5L8039P-6

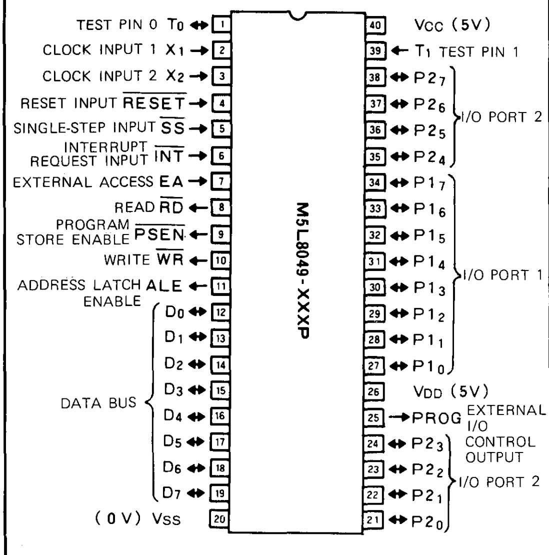
FEATURES

- Single 5V power supply
- Basic machine instructions 96
  - 1-byte instructions: 68
  - 2-byte instructions: 28
- Direct addressing .... up to 4096 bytes
- Internal RAM .... 128 bytes
- Built-in timer/event counter .... 8 bits
- I/O Ports .... 27 lines
- Easily expandable Memory and I/O:
- Subroutine nesting .... 8 levels
- External and timer/event counter interrupt . 1 level each
- External RAM .... 256 bytes
- M5L8049-XXXP/M5L8039P-11, P-6 are interchangeable with Intel's P8049/P8039, P8039-6 in pin configuration and electrical characteristics.

APPLICATION

- Control processor or CPU for a wide variety of applications

PIN CONFIGURATION (TOP VIEW)

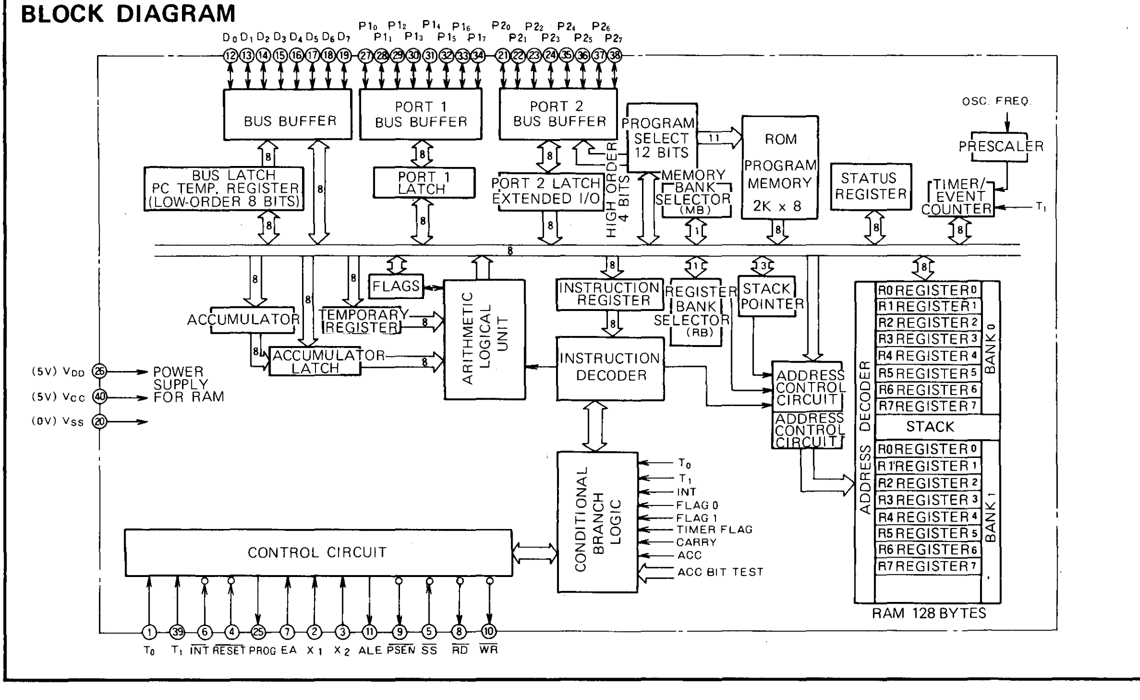


Outline 40P1

FUNCTION

The M5L8049-XXXP and M5L8039P are integrated 8-bit CPUs, with memory (ROM, RAM) and timer/event counter interrupt all contained on a single chip.

BLOCK DIAGRAM



PIN DESCRIPTION

Pin	Name	Input or Output	Function
V_{SS}	Ground		Normally connected to ground (0V)
V_{CC}	Main power supply		Connected to 5V power supply
V_{DD}	Power supply		① Connected to 5V power supply ② Used for memory hold when V_{CC} is cut
PROG	Program	Output	Strobe signal for M5L8243P I/O Expander
$P1_0 \sim P1_7$	Port 1	Input/output	Quasi-bidirectional port. When used as an input port, FF_{16} must first be output to this port. After reset, when not used as an output port nothing can be output.
$P2_0 \sim P2_7$	Port 2	Input/output	① The same as port 1
		Output	② $P2_0 \sim P2_3$ output the high-order 4 bits of the program counter when using external program memory
		Input/output	③ $P2_0 \sim P2_3$ serve as a 4-bit I/O expander bus for the M5L8243P
$D_0 \sim D_7$	Data bus	Input/output	① Provides true bidirectional bus transfer of instructions and data between the CPU and external memory. Synchronizing is done with signals $\overline{RD}/\overline{WR}$. The output data is latched.
			② When using external program memory the output of the low-order 8 bits of the program counter are synchronized with ALE. After that the transfer of the instruction code or data from external program memory is synchronized with $\overline{PSEN}$.
			③ The output of addresses for data using external data memory is synchronized with ALE. After that the transfer of data with the external data memory is synchronized with $\overline{RD}/\overline{WR}$. (MOVX A, @Rr and MOVX @Rr, A)
T_0	Test pin 0	Input	① Control signal from an external source for conditional jumping in a program. Jumping is dependent on external conditions. (JT0 m and JNT0 m)
		Output	② Used for outputting the internal clock signal. (ENT0 CLK)
T_1	Test pin 1	Input	① Control signal from an external source for conditional jumping in a program. Jumping is dependent on external conditions. (JT1 m and JNT1 m) ② When enabled event signals are transferred to the timer/event counter. (STRT CNT)
$\overline{INT}$	Interrupt	Input	① Control signal from an external source for conditional jumping in a program. Jumping is dependent on external conditions. (JN1 m) ② Used for external interrupt to CPU
$\overline{RD}$	Read control	Output	Read control signal used when the CPU requests data from external data memory or external devices to be transferred to the data bus. (MOVX A, @Rr and INS A, BUS)
$\overline{WR}$	Write control	Output	Write control signal used when the CPU sends data through the data bus to external data memory or external device. (MOVX @R, A and OUTL BUS, A)
$\overline{RESET}$	Reset	Input	Control used to initialize the CPU.
ALE	Address latch enable	Output	A signal used for latching the address on the data bus. An ALE signal occurs once during each cycle.
$\overline{PSEN}$	Program store enable	Output	Strobe signal to fetch external program memory.
$\overline{SS}$	Single step	Input	Control signal used in conjunction with ALE to stop the CPU through each instruction, in the single step mode.
EA	External access	Input	① Normally maintained at 0V ② When the level is raised to 5V, external memory will be accessed even when the address is less than 400_{16} (2048).
X_1, X_2	Crystal inputs	Input	External crystal oscillator or RC circuit input for generating internal clock signals. An external clock signal can be input through X_1 or X_2.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.5～7	V
V_{DD}	Supply voltage		−0.5～7	V
V_I	Input voltage		−0.5～7	V
V_O	Output voltage		−0.5～7	V
Pd	Power dissipation	Ta＝25℃	1.5	W
Topr	Operating free-air temperature range		0～70	℃
Tstg	Storage temperature range		−65～150	℃

RECOMMENDED OPERATING CONDITIONS (Ta＝0～70℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{DD}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage		0		V
V_{IH1}	High-level input voltage, except for X_1, X_2, $\overline{RESET}$	2		V_{CC}	V
V_{IH2}	High-level input voltage, X_1, X_2, $\overline{RESET}$	3.8		V_{CC}	V
V_{IL}	Low-level input voltage	−0.5		0.8	V

ELECTRICAL CHARACTERISTICS (Ta＝0～70℃, V_{CC}＝V_{DD}＝5V±10%, V_{SS}＝0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OL}	Low-level output voltage, BUS, $\overline{RD}$, $\overline{WR}$, $\overline{PSEN}$, ALE	I_{OL}＝2mA			0.45	V
V_{OL1}	Low-level output voltage, except for the above and PROG	I_{OL}＝1.6mA			0.45	V
V_{OL2}	Low-level output voltage PROG	I_{OL}＝1mA			0.45	V
V_{OH}	High-level output voltage, BUS, $\overline{RD}$, $\overline{WR}$, $\overline{PSEN}$, ALE	I_{OH}＝−100μA	2.4			V
V_{OH1}	High-level output voltage, except for the above	I_{OH}＝−50μA	2.4			V
I_{IL}	Input leak current, T1, $\overline{INT}$	$V_{SS} \leqq V_{IN} \leqq V_{CC}$	−10		10	μA
I_{OL}	Output leak current, BUS, TO, high-impedance state	$V_{SS}+0.45 \leqq V_{IN} \leqq V_{CC}$	−10		10	μA
I_{LI1}	Input current during low-level input, port	V_{IL}＝0.8V		−0.2		mA
I_{LI2}	Input current during low-level input, $\overline{RESET}$, $\overline{SS}$	V_{IL}＝0.8V		−0.05		mA
I_{DD}	Supply current from V_{DD}	Ta＝25℃		25	50	mA
$I_{DD}+I_{CC}$	Supply current from V_{DD} and V_{CC}	Ta＝25℃		100	170	mA

TIMING REQUIREMENTS (Ta＝0～70℃, V_{CC}＝V_{DD}＝5V±10%, V_{SS}＝0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Limits									Unit
			M5L8049-XXXP M5L8039P-11			M5L8049-XXXP-8 M5L8039P-8			M5L8049-XXXP-6 M5L8039P-6			
			Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
tc	Cycle time	t_{CY}	1.36		15.0	1.875		15.0	2.5		15.0	μs
th (PSEN-D)	Data hold time after $\overline{PSEN}$	t_{DR}	0		100	0		150	0		200	ns
th (R-D)	Data hold time after $\overline{RD}$	t_{DR}	0		100	0		150	0		200	ns
tsu (PSEN-D)	Data setup time after $\overline{PSEN}$	t_{RD}			250			350			500	ns
tsu (R-D)	Data setup time after $\overline{RD}$	t_{RD}			250			350			500	ns
tsu (A-D)	Data setup time after address	t_{AD}			400			650			950	ns
tsu (PROG-D)	Data setup time after PROG	t_{PR}			650			700			810	ns
th (PROG-D)	Data hold time before PROG	t_{PF}	0		150	0		150	0		150	ns

Note 1: The input voltages are V_{IL}＝0.45V and V_{IH}＝2.4V.

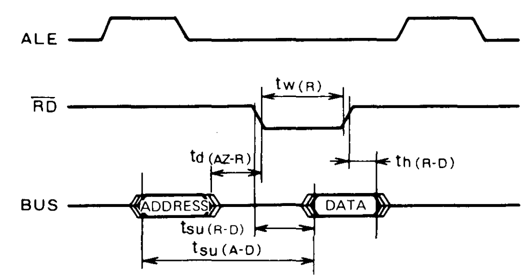
SWITCHING CHARACTERISTICS (T_a = 0~70°C, V_{CC} = V_{DD} = 5V±10%, V_{SS} = 0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Limits									Unit
			M5L8049-XXXP M5L8039-11			M5L8049-XXXP-8 M5L8039P-8			M5L8049-XXXP-6 M5L8039P-6			
			Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
$t_{w(ALE)}$	ALE pulse width	t_{LL}	150			300			400			ns
$t_{d(A-ALE)}$	Delay time, address to ALE signal	t_{AL}	70			120			150			ns
$t_{v(ALE-A)}$	Address valid time after ALE	t_{LA}	50			70			80			ns
$t_{w(PSEN)}$	$\overline{PSEN}$ pulse width	t_{CC}	300			500			700			ns
$t_{w(R)}$	$\overline{RD}$ pulse width	t_{CC}	300			500			700			ns
$t_{d(W)}$	$\overline{WR}$ pulse width	t_{CC}	300			500			700			ns
$t_{v(Q-W)}$	Delay time, data to $\overline{WR}$ signal	t_{DW}	250			380			500			ns
$t_{d(W-Q)}$	Data valid time after $\overline{WR}$	t_{WD}	40			80			120			ns
$t_{d(A-W)}$	Delay time, address to $\overline{WR}$ signal	t_{AW}	200			220			230			ns
$t_{d(AZ-R)}$	Delay time, address disable to $\overline{RD}$ signal	t_{AFC}	−10			−5			0			ns
$t_{d(AZ-PSEN)}$	Delay time, address disable to $\overline{PSEN}$ signal	t_{AFC}	−10			−5			0			ns
$t_{d(PC-PROG)}$	Delay time, port control to PROG signal	t_{CP}	100			105			110			ns
$t_{v(PROG-PC)}$	Port control valid time after PROG	t_{PC}	60			100			130			ns
$t_{p(Q-PROG)}$	Delay time, data to PROG signal	t_{DP}	200			210			220			ns
$t_{v(PROG-Q)}$	Data valid time after PROG	t_{PD}	20			45			65			ns
$t_{w(PROGL)}$	PROG low pulse width	t_{PP}	700			1150			1510			ns
$t_{d(Q-ALE)}$	Delay time, data to ALE signal	t_{PL}	150			300			400			ns
$t_{v(ALE-Q)}$	Data valid time after ALE	t_{LP}	20			100			150			ns

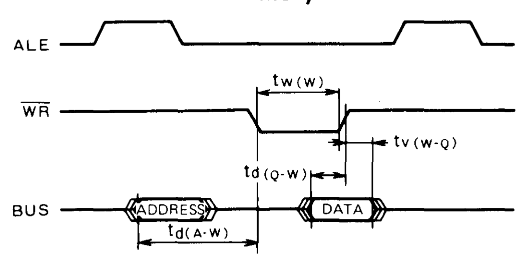
Note 2: Conditions of measurement: control output C_L=80pF
data bus output, port output C_L=150pF t_C=t_C (Min)
3: Reference levels for the input/output voltages are low level=0.8V and high level=2V.

TIMING DIAGRAM

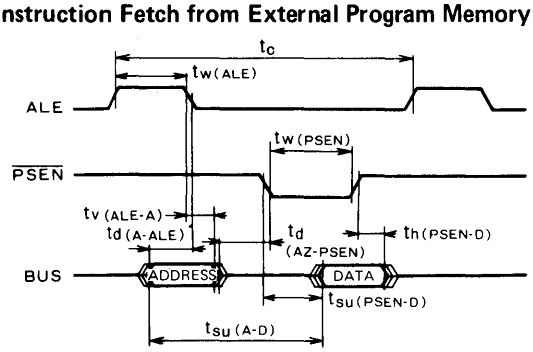
Read from External Data Memory



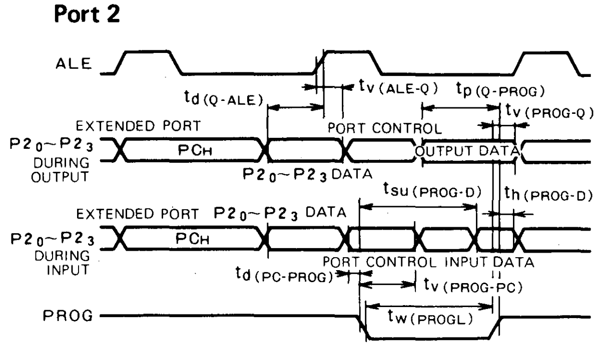
Write to External Data Memory



Instruction Fetch from External Program Memory



Port 2



SINGLE-CHIP 8-BIT MICROCOMPUTER WITH EPROM

DESCRIPTION

The M5L8748S is an 8-bit parallel microcomputer frabricated on a single-chip using high-speed N-channel silicon-gate ED-MOS technology. This contains ultraviolet-light erasable and electrically reprogrammable ROM (EPROM) on a chip, so it is easy to change the program stored in the EPROM.

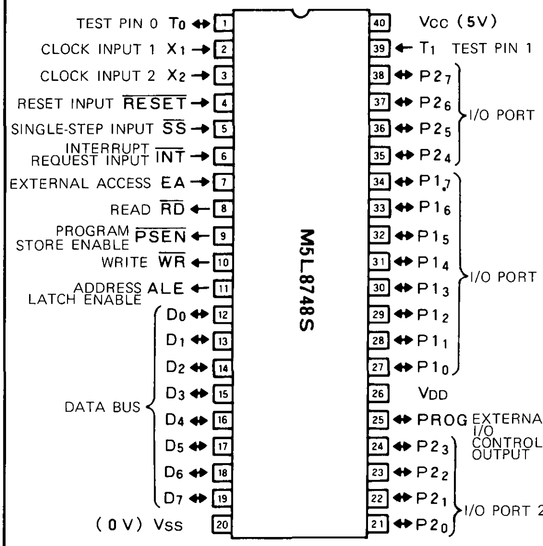
FEATURES

- Single 5V power supply
- Instruction cycle .... 2.5μs (min)
- Basic machine instructions 96
  - 1-byte instructions: 68
  - 2-byte instructions: 28
- Direct addressing .... up to 4096 bytes
- Internal EPROM .... 1024 bytes
- Internal RAM .... 64 bytes
- Built-in timer/event counter .... 8 bits
- I/O Ports .... 27 lines
- Easily expandable memory and I/O
- Subroutine nesting .... 8 levels
- External and timer/event counter interrupt . 1 level each
- External RAM .... 256 bytes
- Interchangeable with the Intel's D8748 in pin configuration and electrical characteristics

APPLICATIONS

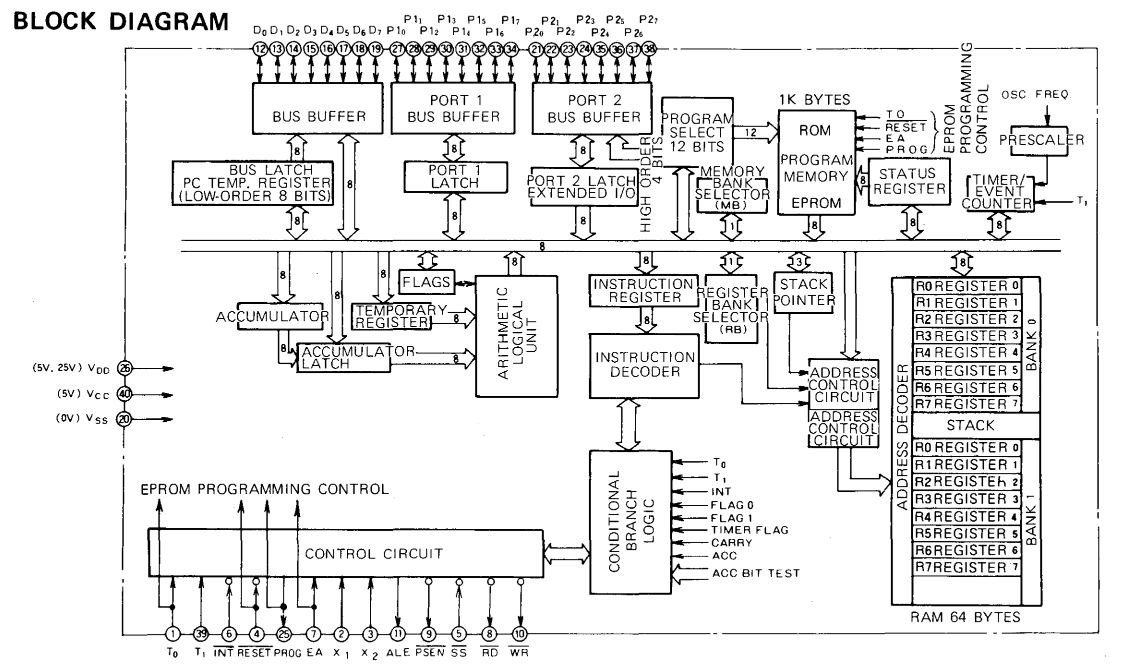
- A CPU for special repetitive processing or control for which a small number of units are to be produced.
- A debugging CPU for program, application and system design development
- A CPU for prototype and preproduction systems prior to factory-programmed mask ROM production

PIN CONFIGURATION (TOP VIEW)



Outline 40S10

BLOCK DIAGRAM



PIN DESCRIPTION

Pin	Name	Input or Output	Function
V_{SS}	Ground		Normally connected to ground (0V).
V_{CC}	Main power supply		Connected to 5V power supply.
V_{DD}	Program power supply		① Normally connected to 5V power supply. ② When programming to EPROM, 25V is required.
PROG	Program	Input	① Used to supply 25V program pulses (50 ms width) from an outside source when programming to EPROM.
		Output	② Strobe signal for M5L8243P I/O Expander.
$P1_0 \sim P1_7$	Port 1	Input/output	Quasi-bidirectional port. When used as an input port, FF_{16} must first be output to this port. After reset, when not used as an output port nothing can be output.
$P2_0 \sim P2_7$	Port 2	Input/output	① The same as port 1.
		Output	② $P2_0 \sim P2_3$ output the high-order 4 bits of the program counter when using external program memory.
		Input/output	③ $P2_0 \sim P2_3$ serve as a 4-bit I/O expander bus for the M5L8243P.
$D_0 \sim D_7$	Data bus	Input/output	① Provides true bidirectional bus transfer of instructions and data between the CPU and external memory. Synchronizing is done with signals $\overline{RD}/\overline{WR}$. The output data is latched.
			② When using external program memory the output of the low-order 8 bits of the program counter are synchronized with ALE. After that the transfer of the instruction code or data from external program memory is synchronized with PSEN.
			③ The output of addresses for data using external data memory is synchronized with ALE. After that the transfer of data with the external data memory is synchronized with $\overline{RD}/\overline{WR}$. (MOVX A, @Rr and MOVX @Rr, A)
T_0	Test pin 0	Input	① Control signal from an external source for conditonal jumping in a program. Jumping is dependent on external conditions. (JT0 m and JNT0 m)
		Output	② Used for outputting the internal clock signal. (ENT0 CLK)
T_1	Test pin 1	Input	① Control signal from an external source for conditional jumping in a program. Jumping is dependent on external conditions. (JT1 m and JNT1 m)
			② When enabled event signals are transferred to the timer/event counter. (STRT CNT)
$\overline{INT}$	Interrupt	Input	① Control signal from an external source for conditional jumping in a program. Jumping is dependent on external conditions. (JN1 m)
			② Used for external interrupt to CPU.
$\overline{RD}$	Read control	Output	Read control signal used when the CPU requests data from external data memory or external devices to be transferred to the data bus. (MOVX A, @Rr and INS A, BUS)
$\overline{WR}$	Write control	Output	Write control signal used when the CPU sends data through the data bus to external data memory or external devices. (MOVX @R, A and OUTL BUS, A)
$\overline{RESET}$	Reset	Input/output	① Control used to initialize the CPU.
			② Latch signal for the EPROM address when programming to EPROM and for reading from EPROM (verify mode).
ALE	Address latch enable	Output	A signal used for latching the address on the data bus. An ALE signal occurs once during each cycle.
$\overline{PSEN}$	Program store enable	Output	Strobe signal used to fetch from external program memory.
$\overline{SS}$	Single step	Input	Control signal used in conjunction with ALE to stop program execution at the finish of each instruction, in the single step mode.
EA	External access	Input	① Normally maintained at 0V. ② When the level is raised to 5V, external memory will be accessed even when the address is less than 400_{16}(1024). ③ When in the programming mode for the EPROM a 25V power supply must be available at this terminal.
X_1, X_2	Crystal inputs	Input	External crystal oscillator or RC circuit input for generating internal clock signals. An external clock signal can be input through X_1 or X_2.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.5~7	V
V_{DD}	Supply voltage		−0.5~26.5	V
V_I	Input voltage		−0.5~7	V
V_O	Output voltage, all outputs except ϕ_1 and ϕ_2		−0.5~7	V
Pd	Power dissipation	Ta=25℃	1.5	W
Topr	Operating free-air temperature range		0~70	℃
Tstg	Storage temperature range		−65~150	℃

RECOMMENDED OPERATING CONDITIONS

CPU Operation (Ta=0~70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{DD}	Supply voltage, except programming EPROM	4.75	5	5.25	V
V_{DD}	Supply voltage, programming EPROM	24	25	26	V
V_{SS}	Supply voltage		0		V
V_{IH1}	High-level input voltage, except X_1, X_2, $\overline{RESET}$	2		V_{CC}	V
V_{IH2}	High-level input voltage, X_1, X_2, $\overline{RESET}$	3.8		V_{CC}	V
V_{IL}	Low-level input voltage	−0.5		0.8	V

EPROM PROGRAMMING (Ta=25±5°C, V_{CC}=5V±5%, V_{DD}=25±1V, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
$V_{DD(H)}$	High-level program supply voltage	24.		26	V
$V_{DD(L)}$	Low-level program supply voltage	4.75		5.25	V
$V_{IH(PROG)}$	High-level program pulse input voltage	21.5		24.5	V
$V_{IL(PROG)}$	Low-level program pulse input voltage			0.2	V
$V_{EA(H)}$	High-level EA input voltage	21.5		24.5	V
$V_{EA(L)}$	Low-level EA input voltage			5.25	V

ELECTRICAL CHARACTERISTICS

CPU Operation (Ta=0~70℃, V_{CC}=V_{DD}=5V ± 5%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OL}	Low-level output voltage, BUS, $\overline{RD}$, $\overline{WR}$, $\overline{PSEN}$, ALE	I_{OL}=2mA			0.45	V
V_{OL1}	Low-level output voltage, except the above and PROG	I_{OL}=1.6mA			0.45	V
V_{OL2}	Low-level output voltage, PROG	I_{OL}=1mA			0.45	V
V_{OH}	High-level output voltage, BUS, $\overline{RD}$, $\overline{WR}$, $\overline{PSEN}$, ALE	I_{OH}=−100μA	2.4			V
V_{OH1}	High-level output voltage, except the above	I_{OH}=−50μA	2.4			V
I_{IL}	Input leak current, T1, $\overline{INT}$	V_{SS}≦V_{IN}≦V_{CC}	−10		10	μA
I_{LI1}	Low-level input current, ports	V_{IL}=0.8V		−0.2		mA
I_{LI2}	Low-level input current, $\overline{RESET}$, $\overline{SS}$	V_{IL}=0.8V		−0.05		mA
I_{DD}	Supply current from V_{DD}	Ta=25°C		10	20	mA
$I_{DD}+I_{CC}$	Supply current from V_{DD} and V_{CC}	Ta=25°C		65	135	mA

EPROM PROGRAMMING (Ta=25±5°C, V_{CC}=5V±5%, V_{DD}=25±1V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
I_{DD}	Supply current from V_{DD}				30	mA
$I_{IH(PROG)}$	High-level input current, PROG				16	mA
$I_{IH(EA)}$	High-level input current, EA				1	mA

TIMING REQUIREMENTS

Read/Write of External Memory (Ta=0~70°C, VCC=VDD=5V±5%, VSS=0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Limits			Unit
			Min	Typ	Max	
t_c	Cycle time	t_{CY}	2.5		15.0	μs
$t_{h(PSEN-D)}$	Data hold time after $\overline{PSEN}$	t_{DR}	0		200	ns
$t_{h(R-D)}$	Data hold time after $\overline{RD}$	t_{DR}	0		200	ns
$t_{su(PSEN-D)}$	Data setup time after $\overline{PSEN}$	t_{RD}			500	ns
$t_{su(R-D)}$	Data setup time after $\overline{RD}$	t_{RD}			500	ns
$t_{su(A-D)}$	Data setup time after address	t_{AD}			950	ns

Note 1: The input voltage level is V_{IL}=0.45 and V_{IH}=2.4V.

Port 2 (Ta=0~70°C, VCC=VDD=5V±5%, VSS=0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Limits			Unit
			Min	Typ	Max	
$t_{su(PROG-D)}$	Data setup time after PROG	t_{PR}			810	ns
$t_{h(PROG-D)}$	Data hold time after PROG	t_{PF}	0		150	ns

Note 2: The input voltage level of the input voltage is V_{IL}= 0.45V and V_{IH}=2.4V.

EPROM PROGRAMMING (Ta=25±5°C, V_{CC}=5V±5%, V_{DD}=25V±1V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Limits			Unit
			Min	Typ	Max	
$t_{su(A-RES)}$	Address setup time before $\overline{RESET}$	t_{AW}	$4t_C$			
$t_{h(RES-A)}$	Address hold time after $\overline{RESET}$	t_{WA}	$4t_C$			
$t_{su(D-PROG)}$	Data setup time before PROG	t_{DW}	$4t_C$			
$t_{h(PROG-D)}$	Data hold time after PROG	t_{WD}	$4t_C$			
$t_{h(T_0-RESH)}$	$\overline{RESET}$ high hold time after T_0 (verify mode)	t_{PH}	$4t_C$			
$t_{su(V_{DD}-PROG)}$	V_{DD} setup time before PROG	t_{VDDW}	$4t_C$			
$t_{h(PROG-V_{DD})}$	V_{DD} hold time after PROG	t_{VDDH}	0			ns
$t_{w(PROG)}$	PROG pulse width	t_{PW}	50		60	ms
$t_{su(T_0-RES)}$	Setup time before $\overline{RES}$	t_{TW}	$4t_C$			
$t_{h(VDD-T_0)}$	Hold time after V_{DD}	t_{WT}	$4t_C$			
$t_{w(RES)}$	$\overline{RESET}$ pulse width	t_{WW}	$4t_C$			

Note 3: CPU cycle time t_c requires 5μs min.
4: Rise time (t_r) and fall time (t_f) of V_{DD} and PROG should be within the range of 0.5~2μs.
5: RESET setup time for the positive-going EA requires 4 t_c min.

SWITCHING CHARACTERISTICS

Read/Write of External Memory (Ta = 0 ~ 70℃, VCC = VDD = 5V ± 5%, VSS = 0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Limits			Unit
			Min	Typ	Max	
$t_{w(ALE)}$	ALE pulse width	t_{LL}	400			ns
$t_{d(A-ALE)}$	Delay time, address to ALE signal	t_{AL}	120			ns
$t_{v(ALE-A)}$	Address valid time after ALE	t_{LA}	80			ns
$t_{w(PSEN)}$	$\overline{PSEN}$ pulse width	t_{CC}	700			ns
$t_{w(R)}$	$\overline{RD}$ pulse width	t_{CC}	700			ns
$t_{w(W)}$	$\overline{WR}$ pulse width	t_{CC}	700			ns
$t_{d(Q-W)}$	Delay time, data to $\overline{WR}$ signal	t_{DW}	500			ns
$t_{v(W-Q)}$	Data valid time after $\overline{WR}$	t_{WD}	120			ns
$t_{d(A-W)}$	Delay time, address to $\overline{WR}$ signal	t_{AW}	230			ns
$t_{d(AZ-R)}$	Delay time, address floating to $\overline{RD}$ signal	t_{AFC}	0			ns
$t_{d(AZ-PSEN)}$	Delay time, address floating to $\overline{PSEN}$ signal	t_{AFC}	0			ns

Note 6: Conditions of measurement: control output C_L=80pF
data bus output C_L=150pF, t_c=2.5μs
7: Reference level for the input/output voltage is low level=0.8V and high level=2V.

Port 2 (Ta = 0 ~ 70℃, VCC = VDD = 5V ± 5%, VSS = 0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Limits			Unit
			Min	Typ	Max	
$t_{d(PC-PROG)}$	Delay time, port control to $\overline{PROG}$ signal	t_{CP}	110			ns
$t_{v(PROG-PC)}$	Port control valid time after $\overline{PROG}$	t_{PC}	100			ns
$t_{p(Q-PROG)}$	Delay time, data to $\overline{PROG}$ signal	t_{DP}	250			ns
$t_{v(PROG-Q)}$	Data valid time after $\overline{PROG}$	t_{PD}	65			ns
$t_{w(PROGL)}$	$\overline{PROG}$ low-level pulse width	t_{PP}	1200			ns
$t_{d(Q-ALE)}$	Delay time, data to ALE signal	t_{PL}	350			ns
$t_{v(ALE-Q)}$	Data valid time after ALE	t_{LP}	150			ns

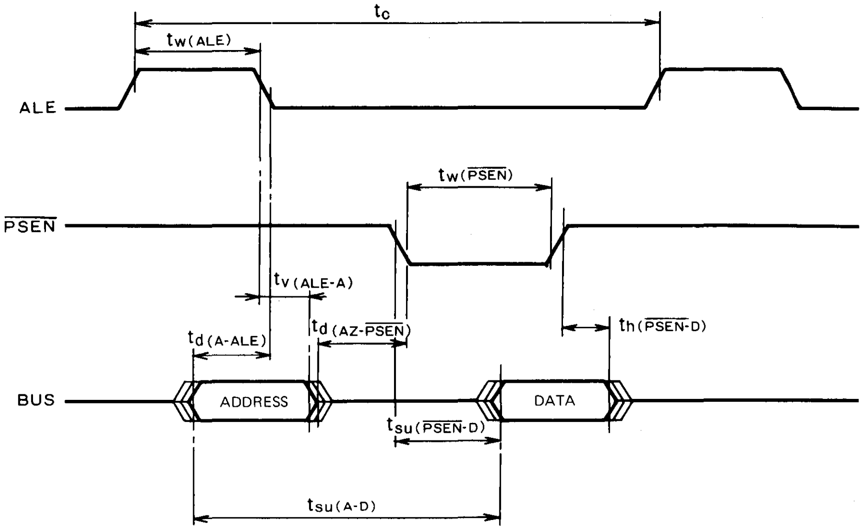
Note 8: Condition of measurement is C_L=150pF, t_c=2.5μs
9: Reference level for the input/output voltage is low level=0.8V and high level=2V.

EPROM PROGRAMMING (Ta = 25 ± 5℃, VCC = 5V ± 5%, VDD = 25V ± 1V, unless otherwise noted)

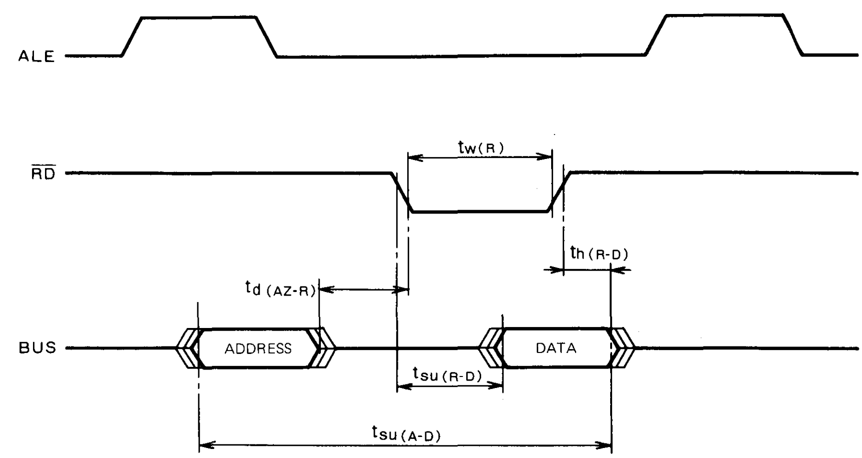
Symbol	Parameter	Alternative symbol	Limits			Unit
			Min	Typ	Max	
$t_{p(T_0-Q)}$	Propagation time between T_0 and data.	t_{DO}			$4t_C$	

TIMING DIAGRAM

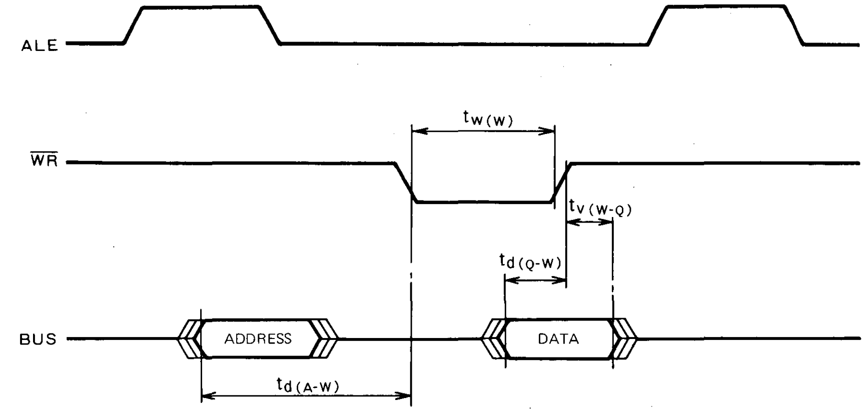
Instruction fetch from external program memory



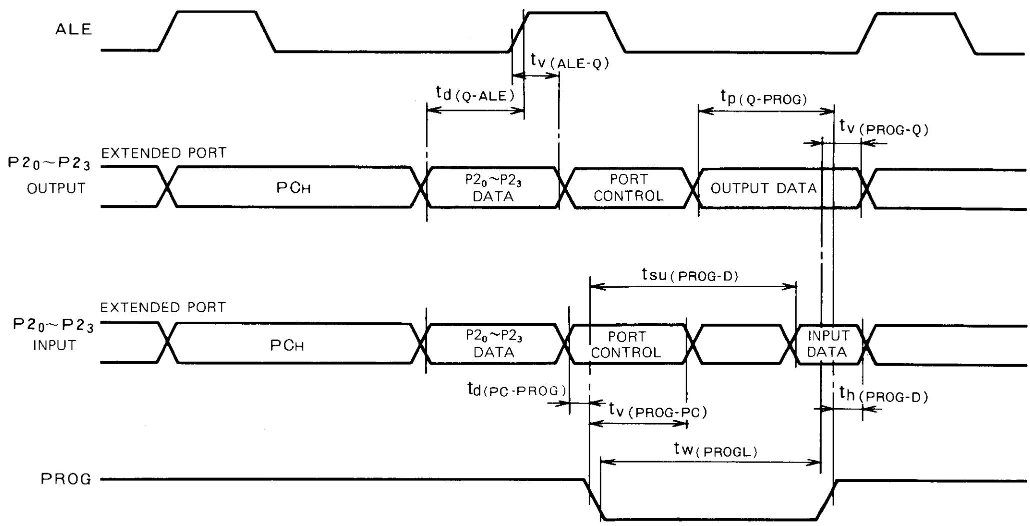
Reading from external data memory



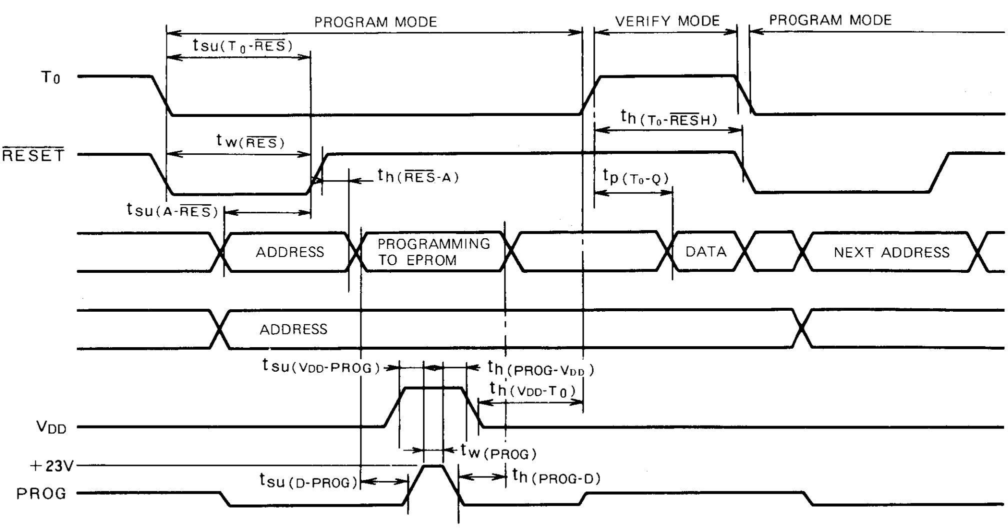
Writing to external data memory



Port 2



PROGRAMMING to EPROM



PROGRAMMING TO EPROM AND ERASING

Details of how to program data onto the EPROM for the M5L8748S is shown in Fig. 1. The EPROM can be erased by approximately 15 Ws/cm^2 of exposure to high-intensity 2537Å short-wave ultraviolet rays.

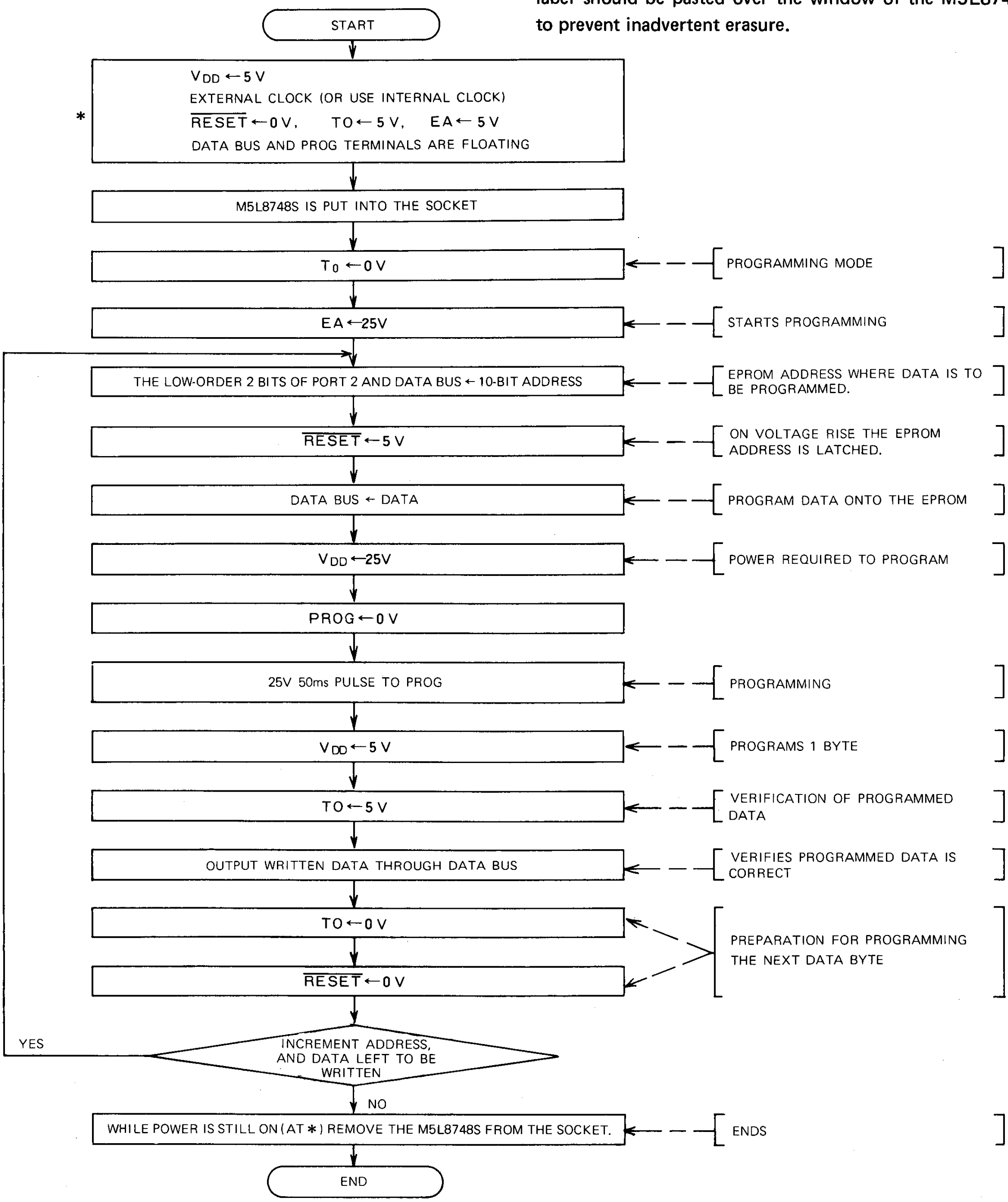
For example: the S-52 ultraviolet lamp has an intensity of 17,000μW/cm^2 at a distance of 2.4 cm from the lamp.

The necessary exposure would be:

$$\frac{15\text{Ws/cm}^2}{17{,}000\mu\text{W/cm}^2} \fallingdotseq 900 \text{ seconds} = 15 \text{ minutes}$$

Once data has been entered on the EPROM an opaque label should be pasted over the window of the M5L8748S to prevent inadvertent erasure.

Fig. 1 Flowchart of programming onto the EPROM



INPUT/OUTPUT EXPANDER

DESCRIPTION

The M5L8243P is an input/output expander fabricated using N-channel silicon-gate ED-MOS technology. This device is designed specifically to provide a low-cost means of I/O expansion for the MELPS 8-48 single-chip microcomputer and M5L8041A-XXXP.

FEATURES

- 16 Input/output pins ($I_{OL}=5.0mA(max)$)
- Simple interface to MELPS 8-48 microcomputers
- Single 5V power supply
- Low power dissipation: 50mW (typ)
- Interchangeable with Intel's 8243 in pin configuration and electrical characteristics

APPLICATION

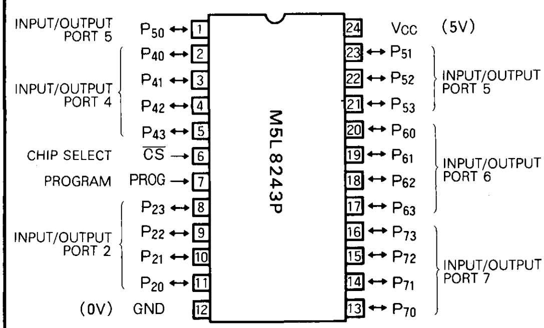
- I/O expansion for the MELPS 8-48 single-chip microcomputers.

FUNCTION

The M5L8243P is designed to provide a low-cost means of I/O expansion for the M5L8041A-XXXP universal peripheral interface and the M5L8048 and M5L8049 single-chip microcomputers. The M5L8243P consists of four 4-bit bidirectional static I/O ports and one 4-bit port which serves as an interface to the M5L8041A-XXXP and M5L8048/9. Thus multiple M5L8243Ps can be added to a single master.

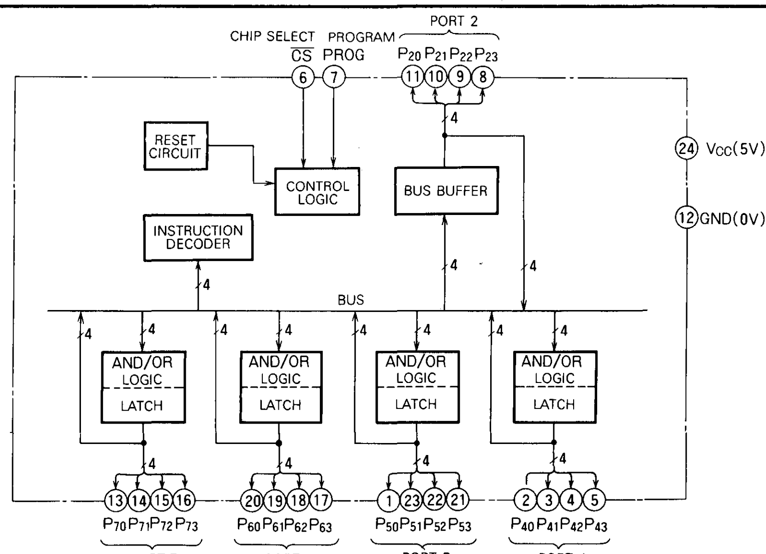
Using the original instruction set of the master, the M5L8243P serves as the in resident I/O facility. Its I/O ports are accessed by instructions MOV, ANL and ORL.

PIN CONFIGURATION (TOP VIEW)



Outline 24P1

BLOCK DIAGRAM



PIN DESCRIPTION

Symbol	Name	Input or output	Function
PROG	Program	In	A high-to-low transition on PROG signifies that address (PORT 4-7) and control are available on PORT 2, and a low-to-high transition signifies that the designated data is available on the disignated port through PORT 2. The designation is shown in Table 1.
$\overline{CS}$	Chip select	In	Chip select input. A high on $\overline{CS}$ causes PROG input to be regarded high inside the M5L8243P, then this inhibits any change of output or internal status.
P_{20}~P_{23}	Input/output port 2	In/out	The 4-bit bidirectional port contains the address and control bits shown in Table 1 on a high-to-low transition of PROG. During a low-to-high transition it contains the input (output) data on this port.
P_{40}~P_{43} P_{50}~P_{53} P_{60}~P_{63} P_{70}~P_{73}	Input/output port 4 Input/output port 5 Input/output port 6 Input/output port 7	In/out	The 4-bit bidirectional I/O port. May be programmed to be input, low-impedance latched output or a three-state. This port is automatically set output mode when it is written, ANLed or ORLed, then continues its mode until next read operation. After reset on a read operation, this port is in high-impedance and input mode.

OPERATION

The M5L8243P is an input/output expander designed specifically for the M5L8014A-XXXP and MELPS 8-48 single-chip 8-bit microcomputer. The M5L8041A-XXXP and MELPS 8-48 already have instructions and PROG pin to communicate with the M5L8243P.

An example of the M5L8243P and the M5L8041A-XXXP is shown in Fig. 1. The following description of the M5L8243P basic operation is made according to Fig. 1.

Upon initial application of power supply to the device, and then about 50ms after, resident bias circuits become stable and each device is ready to operate. And each port of the M5L8243P is set input mode (high-impedance) by means of a resident power-on initialization circuit.

When the microcomputer begins to execute a transfer instruction

MOVD A, Pi i = 4, 5, 6, 7

which means the value on the port Pi is transferred to the accumulator, then the signals are sent out on the pins PROG and P_{20}~P_{23} as shown in Timing Diagram.

On the high-to-low transition of the pin PROG, the M5L8243P latches the instructions (ex. 0000) into itself from pins P_{20}~P_{23} and transfers them to the instruction register (① in Timing Diagram). During the low-level of PROG, the M5L8243P continuously outputs the contents of the specified input (output) port (in this case port P_4) to pins P_{20}~P_{23} (② in Timing Diagram). The microcomputer, at an appropriate time, latches the level of pins P_{20}~P_{23} and resumes high-level of PROG.

The next example is the case in which the microcomputer executes

MOVD Pi, A i = 4, 5, 6, 7

the transfer (output) instruction.

In this case, as in the previous case, on the high-to-low transition of the pin PROG, the M5L8243P latches the instructions (ex. 0110) into itself from pins P_{20}~P_{23} and transfers them to the instruction register (① in Timing Diagram).

After this, the microcomputer sends out high to the pin PROG, transferring the data to pins P_{20}~P_{23} which is an output data to input/output port. Then the M5L8243P transfers the data of pins P_{20}~P_{23} to the port latch of the designated input/output port (in this case P_6). In a few seconds after a low-to-high transition on the PROG, the designated port (P_6) becomes in an output mode and the data of the port latch are transferred to the port pins (③ in Timing Diagram).

When instructions

ANLD Pi, A
ORLD Pi, A i = 4, 5, 6, 7

are executed, the microcomputer generally operates as same function as MOVD Pi, A.

It only differs in that the data of port latch after ④ in the Timing Diagram is ANDed or ORed with the data of port latch before ④ and the data of pins P_{20}~P_{23}.

When instructions

MOVD Pi, A
ANLD Pi, A
ORLD Pi, A i = 4, 5, 6, 7

are executed toward the port in an output mode, the outputs are generated on the port as soon as low-to-high transition on the PROG occurs.

When the mode of the output port is going to be changed during the execution and the instruction

MOVD A, Pi i = 4, 5, 6, 7

is executed, it is preferable to execute one dummy instruction. Because it takes a little time to turn the designated port into a high-impedance state after high-to-low transition on the PROG, the result may be that the first instruction is not read correctly.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.5~7	V
V_I	Input voltage		−0.5~7	V
V_O	Output voltage		−0.5~7	V
Pd	Maximum power dissipation	Ta=25℃	600	mW
Topr	Operating free-air temperature range		0~70	℃
Tstg	Storage temperature range		−65~ 150	℃

RECOMMENDED OPERATING CONDITIONS (Ta=−20~70℃, V_{CC}=5V±10%, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage	2			V
V_{IL}	Low-level input voltage	−0.5		0.8	V

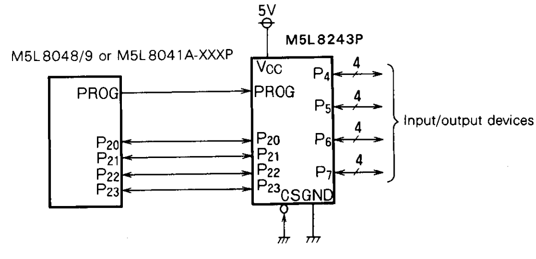
ELECTRICAL CHARACTERISTICS (T_a=−20~70℃, V_{CC}=5V±10%, unless otherwise noted)

Symbol	Parameter	Test condition	Limits			Unit
			Min	Typ	Max	
V_{IL}	Low-level input voltage		0.5		0.8	V
V_{IH}	High-level input voltage				V_{CC}+0.5	V
V_{OL1}	Low-level output voltage, ports 4~7	I_{OL} = 5 mA			0.45	V
V_{OL2}	Low-level output voltage, port 7	I_{OL}=20mA			1	V
V_{OL3}	Low-level output voltage, port 2	I_{OL} = 0.6mA			0.45	V
V_{OH1}	High-level output voltage, ports 4~7	I_{OH}=240μA	2.4			V
V_{OH2}	High-level output voltage, port 2	I_{OH} = 100μA	2.4			V
I_{I1}	Input leakage current, ports 4~7	0V≦Vin≦V_{CC}	−10		20	μA
I_{I2}	Input leakage current, port 2, CS, PROG	0V≦Vin≦V_{CC}	−10		10	μA
I_{CC}	Supply current from V_{CC}			10	20	mA
I_{OL}	Sum of all I_{OL} from 16 outputs	I_{OL}=5mA (V_{OL}=0.45V) Each pin			80	mA

Table 1 Instruction and address codes

Instruction code	P_{23}	P_{22}	Address code	P_{21}	P_{20}
Read	0	0	port 4	0	0
Write	0	1	port 5	0	1
ORLD	1	0	port 6	1	0
ANLD	1	1	port 7	1	1

Fig. 1 Basic connection



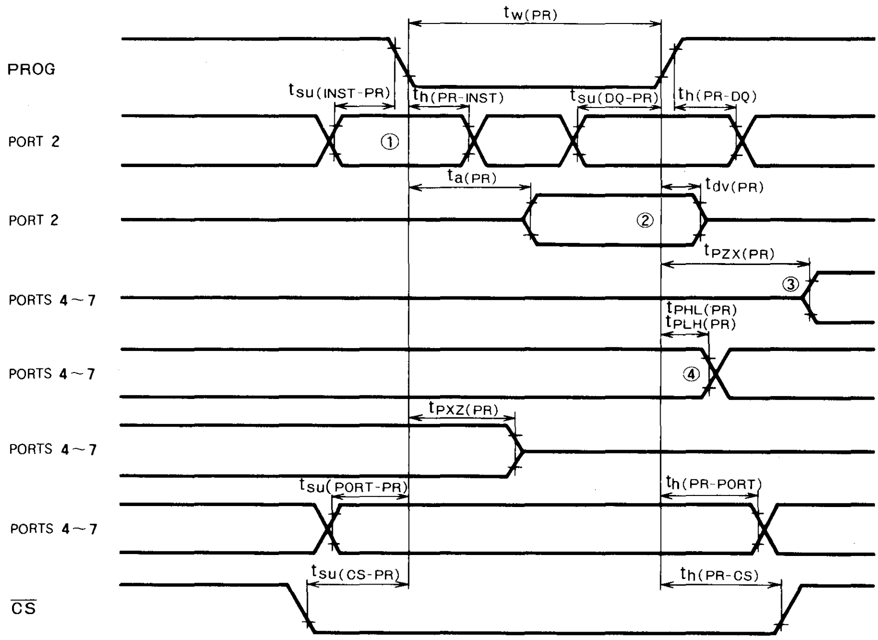
TIMING REQUIREMENTS ($T_a = -20 \sim 70°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{su(INST-PR)}$	Instruction code setup time before PROG	t_A	80pF Load	100			ns
$t_{h(PR-INST)}$	Instruction code hold time after PROG	t_B	20pF Load	60			ns
$t_{su(DQ-PR)}$	Data setup time before PROG	t_C	80pF Load	200			ns
$t_{h(PR-DQ)}$	Data hold time after PROG	t_D	20pF Load	20			ns
$t_{w(PR)}$	PROG pulse width	t_K		700			ns
$t_{su(CS-PR)}$	Chip-select setup time before PROG	t_{CS}		50			ns
$t_{h(PR-CS)}$	Chip-select hold time after PROG	t_{CS}		50			ns
$t_{su(PORT-PR)}$	Port setup time before PROG	t_{IP}		100			ns
$t_{h(PR-PORT)}$	Port hold time after PROG	t_{IP}		100			ns

SWITCHING CHARACTERISTICS

Symbol	Parameter	Alternative symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{a(PR)}$	Data access time after PROG	t_{ACC}	80pF Load	0		650	ns
$t_{dv(PR)}$	Data valid time after PROG	t_H	20pF Load	0		150	ns
$t_{PHL(PR)}$ $t_{PLH(PR)}$	Output valid time after PROG	t_{PO}	100pF Load			700	ns
$t_{PZX(PR)}$ $t_{PXZ(PR)}$	Input/output switching time	—				800	ns

TIMING DIAGRAM

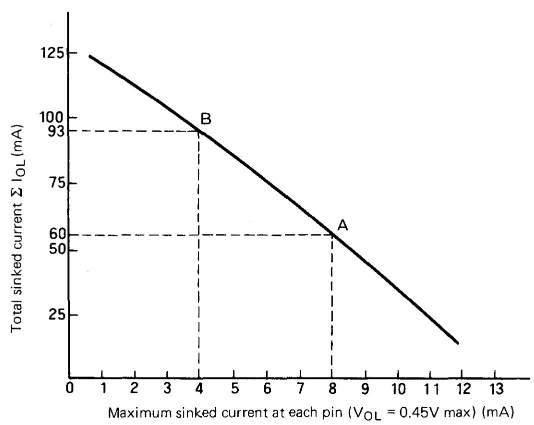


Note 1 : AC test conditions

Input pulse level: 0.45～2.4V
Input pulse rise time t_r (10%～90%): 20ns
Input pulse fall time t_f (10%～90%): 20ns
Reference voltage for switching characteristic measurement.

Input	V_{IH}:	2V	V_{IL}:	0.8V
Output	V_{OH}:	2V	V_{OL}:	0.8V

Current Sinking Capability



Each of the 16 I/O lines of the M5L8243P is capable of sinking 5mA simultaneously (V_{OL} = 0.45V max). However, the drive capacity of each line depends upon whether all lines are sinking current simultaneously and on the degree of loading. This is illustrated in the curve shown.

Example

Assuming that the remaining pins are not loaded, how many pins would be able to accommodate 5TTL loads (1.6mA)?

I_{OL} = 1.6mA x 5 = 8mA (sink current for each pin)

ΣI_{OL} = 60mA from curve (curve A)
(total sinking current)

Number of pins = 60mA ÷ 8mA/pin = 7.5 = 7 lines

For this case, each of th 7 lines could sink 8mA for a total of 56mA. Since 4mA reserve sinking capability exists, 9 of the I/O lines of the M5L8243P can be divided arbitrarily.

Example

To use 20mA sinking capability at port 7, find the effects on the skinking capabilities of the other I/O lines.

Assume the M5L8243P is driving loads as shown below.

2 lines: -20mA (V_{OL} = 1.0V max, port 7 only)

8 lines: -4mA (V_{OL} = 0.45V max)

6 lines: -3.2mA (V_{OL} = 0.5V max)

Is this within the allowable limit?

ΣI_{OL} = (20mA x 2) + (4mA x 8) + (3.2mA x 6) = 91.2mA

From the curve we see that with respect to I_{OL} = 4mA, I_{OL} is 93mA (curve B) and that the above load of 91.2mA is within the limit of 93mA.

Note: The sinking current of ports 4 ~ 7 must not exceed 30mA regardless of the value of V_{OL}.

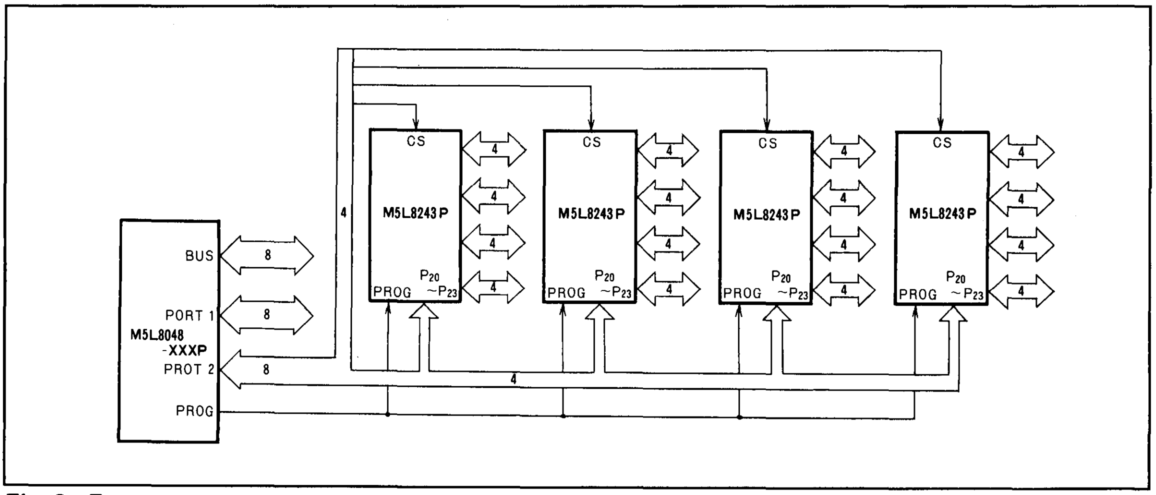


Fig. 2 Expansion interface example

MELPS 8/85 MICROPROCESSORS

7

8-BIT PARALLEL MICROPROCESSOR

DESCRIPTION

This is a family of single-chip 8-bit parallel central processing units (CPUs) developed using the N-channel silicon-gate ED-MOS process. It requires a single 5V power supply and has a basic clock rate of 3MHz. With an instruction set that is completely compatible with that of the M5L8080AP,S, this device is designed to improve on the M5L8080A with higher system speed.

FEATURES

- Single 5V power supply
- Software compatibility with the M5L8080AP,S (with two additional instructions)
- Instruction cycle .... 1.3 μs (min.)
- Clock generator (with an external crystal or RC circuit)
- Built-in system controller
- Four vectored interrupts (one of which is non-maskable)
- Serial I/O port: 1 each
- Decimal, binary, and double precision arithmetic operations
- Direct addressing up to 64K bytes of memory
- Interchangeable with Intel's 8085A in pin connection and electrical characteristics

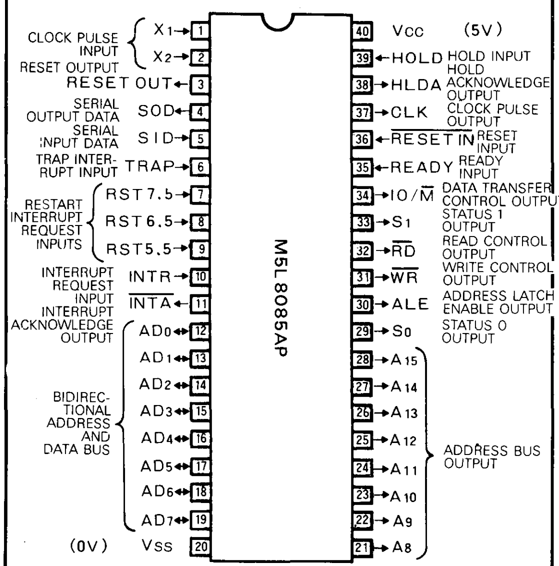
APPLICATION

- Central processing unit for a microcomputer

FUNCTION

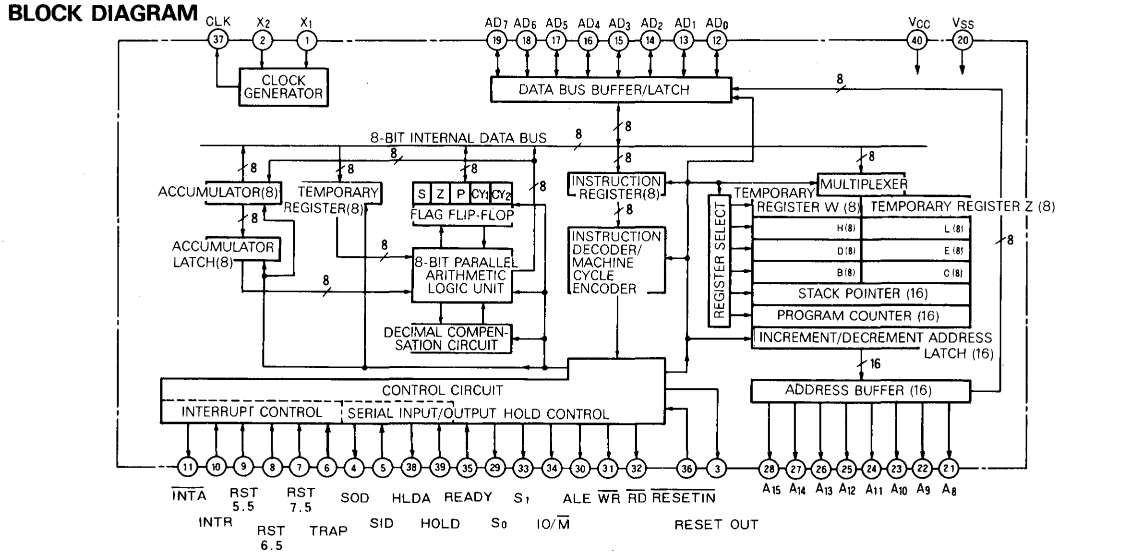
Under the multiplexed data bus concept adopted, the high-order 8 bits of the address are used only as an address bus and the low-order 8 bits are used as an address/data bus. During the first clock cycle of an instruction cycle, the address is transferred. The low-order 8 bits of the address are stored in the external latch by the address latch enable (ALE) signal. During the second and third clock cycles, the address/data bus functions as the data bus, transferring the data to memory or to the I/O. For bus control, the device provides $\overline{RD}$, $\overline{WR}$, and IO/$\overline{M}$ signals and an interrupt acknowledge signal $\overline{INTA}$. The HOLD, READY and all interrupt signals are synchronized with the clock pulse. For simple serial data transfer it provides both a serial input data (SID) line and a serial output data (SOD) line. It also has three maskable restart interrupts and one non-maskable trap interrupt.

PIN CONFIGURATION (TOP VIEW)



Outline 40P1 (M5L8085AP), 40S1 (M5L8085AS)

BLOCK DIAGRAM



PIN DESCRIPTIONS

Pin	Name	Input or output	Functions
$A_8 \sim A_{15}$	Address bus	Out	Outputs the high-order 8 bits of the memory address or the 8 bits of the I/O address. It remains in the high-impedance state during the HOLD and HALT modes.
$AD_0 \sim AD_7$	Bidirectional address and data bus	In/out	The low-order (I/O address) appears during the first clock cycle. During the second and third clock cycles, it becomes the data bus. It remains in the high-impedance state during the HOLD and HALT modes
ALE	Address latch enable	Out	This signal is generated during the first clock cycle, to enable the address to be latched into the latches of peripherals. The falling edge of ALE is guaranteed to latch the address information. The ALE can also be used to strobe the status information, but it is kept in the low-level state during bus idle machine cycles.
S_0, S_1	Status	Out	Indicates the status of the bus. S_1 S_0 HALT 0 0 WRITE 0 1 READ 1 0 FETCH 1 1 The S_1 signal can be used as an advanced $R/\overline{W}$ status.
$\overline{RD}$	Read control	Out	Indicates that the selected memory or I/O address is to be read and that the data bus is active for data transfer. It remains in the high-impedance state during the HOLD and HALT modes
$\overline{WR}$	Write control	Out	Indicates that the data on the data bus is to be written into the selected memory at the trailing edge of the signal $\overline{WR}$. It remains the high-impedance state during the HOLD and HALT modes.
RST5.5 RST6.5 RST7.5	Restart interrupt request	In	Input timing is the same as for INTR for these three signals. They all cause an automatic insertion of an internal RESTART. RST 7.5 has the highest priority while RST 5.5 has the lowest. All three signals have a higher priority than INTR.
TRAP	Trap interrupt	In	A non-maskable restart interrupt which is recognized at the same time as an INTR. It is not affected by any mask or another interrupt. It has the highest interrupt priority.
$\overline{RESET\ IN}$	Reset input	In	This signal (at least three clock cycles are necessary) sets the program counter to zero and resets the interrupt enable and HLDA flip-flops. None of the other flags or registers (except the instruction register) are affected. The CPU is held in the reset mode as long as the signal is applied.
RESET OUT	Reset output	Out	This signal indicates that the CPU is in the reset mode. It can be used as a system RESET. The signal is synchronized to the processor clock.
X_1, X_2	Clock input	In	These pins are used to connect an external crystal or CR circuit to the internal clock generator. An external clock pulse can also be input through X_1.
CLK	Clock output	Out	Clock pulses are available from this pin when a crystal or CR circuit is used as an input to the CPU.
$IO/\overline{M}$	Data transfer control output	Out	This signal indicates whether the read/write is to memory or to I/Os. It remains in the high-ipedance state during the HOLD and HALT modes.
READY	Ready input	In	When it is at high-level during a read or write cycle the READY indicates that the memory or peripheral is ready to send or receive data. When the signal is at low-level, the CPU will wait for the signal to turn high-level before completing the read or write cycle.
HOLD	Hold request signal	In	When the CPU receives a HOLD request, it relinquishes the use of the buses as soon as the current machine cycle is completed. The CPU can regain the use of buses only after the HOLD state is removed. Upon acknowledging the HOLD signal, the address bus, the data bus, $\overline{RE}$, $\overline{WR}$ and $IO/\overline{M}$ lines are put in the high-impedance state.
HLDA	Hold acknowledge signal	Out	By this signal the processor acknowledges the HOLD request signal and indicates that it will relinquish the buses in the next clock cycle. The signal is returned to the low-level state after the HOLD request is completed. The processor resumes the use of the buses one half clock cycle after the signal HLDA goes low.
INTR	Interrupt request signal	In	This signal is for a general purpose interrupt and is sampled only during the last clock cycle of the instruction. When an interrupt is acknowledged, the program counter (PC) is held and an INTA signal is generated. During this cycle, a RESTART or CALL can be inserted to jump to an interrupt service routine. Immediately after an interrupt is accepted it may be enabled and disabled by means of software. The interrupt request is disabled by the RESET.
$\overline{INTA}$	Interrupt acknowledge control signal	Out	This signal is used instead of $\overline{RD}$ during the instruction cycle after an INTR is accepted.
SID	Serial input data	In	This is an input data line for serial data, and the data on this line is moved to the 7th. bit of the accumulator whenever a RIM instruction is executed.
SOD	Serial output data	Out	This is an output data line for serial data. The output SOD may be set or reset by means of the SIM instruction.

Note: HOLD, READY and all interrupt signals are synchronized with clock signal.

MITSUBISHI ELECTRIC

STATUS INFORMATION

Status information can be obtained directly from the M5L8085A. ALE is used as a status strobe. As the status is partially encoded, it informs the user in advance what type of bus transfer is being performed. The IO/$\overline{M}$ cycle status signal is also obtained directly. Decoded S_0 and S_1 signals carry:

	S_1	S_0
HALT	0	0
WRITE	0	1
READ	1	0
FETCH	1	1

S_1 can be used in determining the $\overline{R/W}$ status of all bus transfers.

In the M5L8085A the low-order 8 bits of the address are multiplexed with data. When entering the low-order of the address into memory or peripheral latch circuits, the ALE is used as a strobe.

INTERRUPT AND SERIAL I/O

The M5L8085A has five interrupt inputs—INTR, RST 5.5, RST 6.5, RST 7.5, and TRAP. INTR has the same function as INT of the M5L8080A. The three RST inputs, 5.5, 6.5, 7.5, are provided with programmable masks. TRAP has the same function as the restart interrupt, except that it is non-maskable.

When an interrupt is enabled and the corresponding interrupt mask is not set, the three RST interrupts will cause the internal execution of the RST. When nonmaskable TRAP is applied, it causes the internal execution of an RST regardless of the state of the interrupt enable or masks. The restart addresses (hexadecimal) of the interrupts are:

Interrupt	Address
TRAP	24_{16}
RST 5.5	$2C_{16}$
RST 6.5	34_{16}
RST 7.5	$3C_{16}$

Two different types of signal are used for restart interrupts. Both RST 5.5 and RST 6.5 are sensitive to high-level as in INTR and INT of the M5L8080A, and are acknowledged in the same timing as INTR. RST 7.5 is sensitive to rising-edge, and existence of a pulse sets the RST 7.5 interrupt request. This condition will be maintained until the request is fulfilled or reset by a SIM or RESET instruction.

Each of the restart interrupts may be masked independently to avoid interrupting the CPU. An interrupt requested by an RST 7.5 will be stored even when its mask is set and the interrupt is disabled. Masks can only be changed in the RESET mode. When two enabled interrupts are requested at the same time the interrupt with the highest priority will be accepted. The TRAP has the highest priority followed in order by RST 7.5, RST 6.5, RST 5.5 and INTR. This priority system does not take into consideration the priority of an interrupt routine that is already started. In other words, when an RST 5.5 interrupt is reenabled before the termination of the RST 7.5 interrupt routine, it will interrupt the RST 7.5.

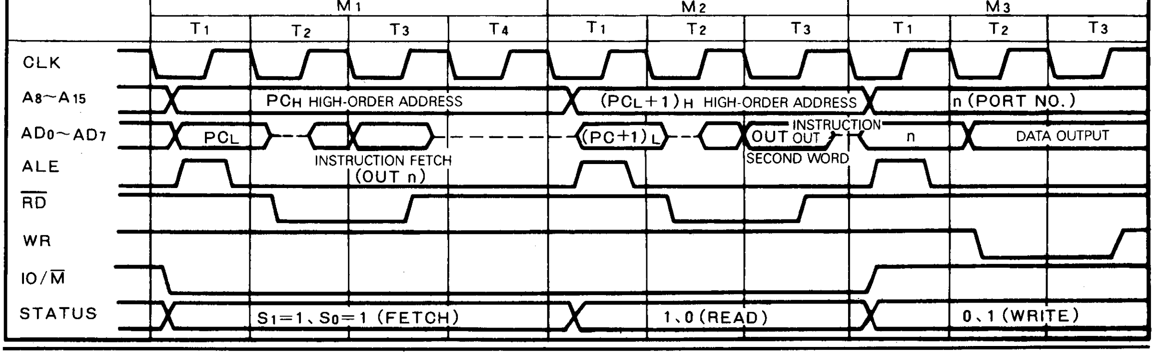
The TRAP interrupt is very useful in preventing disastrous errors and bus errors resulting from power failures. The TRAP input is recognized in the same manner as any other interrupt, but it has the highest priority, and is not affected by any flags or masks. The TRAP input can be sensed by either edge or level. TRAP should be maintained high-level until it is acknowledged. But, it will not be acknowledged again unless it turns low and high again. In this manner, faulty operation due to noise or logic glitches is prevented.

The serial I/O system is also considered to be an interrupt as it is controlled by instructions RIM and SIM. The SID is read by instruction RIM and the SOD data is set by instruction SIM.

BASIC TIMING

The M5L8085A is provided with a multiplexed data bus. The ALE is utilized as a strobe with which the low-order 8 bits of the address on the data bus are sampled. Fig.1 shows the basic cycle in which an out instruction is fetched, and memory is read and written to the I/O port. The I/O port address is stored in both the address bus and the address/data bus during the I/O write and read cycle. To enable the M5L8085A to be used with a slow memory, the READY line is used for extending the read and write pulse width in the same manner as in the M5L8080A.

Fig. 1 Basic cycle



MACHINE INSTRUCTIONS

Instr class \ Item	Mnemonic	Instruction code D7 D6	D5 D4 D3	D2 D1 D0	16mal notatn	No. of states	No. of bytes	No. of cycles	Functions	Flags S Z P CY2 CY1	Address bus Contents	Address bus Mach cycle*	Data bus Contents	Data bus I/O	Data bus Mach cycle**
Data transfer	MOV r1, r2	0 1	D D D	S S S		4	1	1	(r1) ← (r2)	X X X X X					
	MOV M, r	0 1	1 1 0	S S S		7	1	2	(M) ← (r) Where. M=(H) (L)	X X X X X	M	M4	(r)	O	M4
	MOV r, M	0 1	D D D	1 1 0		7	1	2	(r) ← (M) Where. M=(H) (L)	X X X X X	M	M4	(M)	I	M4
	MVI r, n	0 0	D D D <B2>	1 1 0		7	2	2	(r) ← n	X X X X X			<B2>	I	M4
	MVI M, n	0 0	1 1 0 <B2>	1 1 0	3 6	10	2	3	(M) ← n Where. M=(H) (L)	X X X X X	M	M5	<B2>	I	M5
	LXI B, m	0 0	0 0 0 <B2> <B3>	0 0 1	0 1	10	3	3	(C) ← <B2> (B) ← <B3> Where. m=<B3> <B2>	X X X X X			<B2> <B3>	I I	M2 M3
	LXI D, m	0 0	0 1 0 <B2> <B3>	0 0 1	1 1	10	3	3	(E) ← <B2> (D) ← <B3> Where. m=<B3> <B2>	X X X X X			<B2> <B3>	I I	M2 M3
	LXI H, m	0 0	1 0 0 <B2> <B3>	0 0 1	2 1	10	3	3	(L) ← <B2> (H) ← <B3> Where. m=<B3> <B2>	X X X X X			<B2> <B3>	I I	M2 M3
	LXI SP, m	0 0	1 1 0 <B2> <B3>	0 0 1	3 1	10	3	3	(SP) ← m	X X X X X			<B2> <B3>	I I	M2 M3
	SPHL	1 1	1 1 1	0 0 1	F 9	6	1	1	(SP) ← (H) (L)	X X X X X					
	STAX B	0 0	0 0 0	0 1 0	0 2	7	1	2	((B) (C)) ← (A)	X X X X X	(B) (C)	M4	(A)	O	M4
	STAX D	0 0	0 1 0	0 1 0	1 2	7	1	2	((D) (E)) ← (A)	X X X X X	(D) (E)	M4	(A)	O	M4
	LDAX B	0 0	0 0 1	0 1 0	0 A	7	1	2	(A) ← ((B) (C))	X X X X X	(B) (C)	M4	((B) (C))	I	M4
	LDAX D	0 0	0 1 1	0 1 0	1 A	7	1	2	(A) ← ((D) (E))	X X X X X	(D) (E)	M4	((D) (E))	I	M4
	STA m	0 0	1 1 0 <B2> <B3>	0 1 0	3 2	13	3	4	(m) ← (A)	X X X X X	m	M4	(A)	O	M4
	LDA m	0 0	1 1 1 <B2> <B3>	0 1 0	3 A	13	3	4	(A) ← (m)	X X X X X	m	M4	(m)	I	M4
	SHLD m	0 0	1 0 0 <B2> <B3>	0 1 0	2 2	16	3	5	(m) ← (L) (m+1) ← (H)	X X X X X	m m+1	M4 M5	(L) (H)	O O	M4 M5
	LHLD m	0 0	1 0 1 <B2> <B3>	0 1 0	2 A	16	3	5	(L) ← (m) (H) ← (m+1)	X X X X X	m m+1	M4 M5	(m) (m+1)	I I	M4 M5
	XCHG	1 1	1 0 1	0 1 1	E B	4	1	1	(H) (L) ↔ (D) (E)	X X X X X					
	XTHL	1 1	1 0 0	0 1 1	E 3	16	1	5	(H) (L) ↔ ((SP)+1) ((SP))	X X X X X	(SP) (SP)+1	M2 M3	((SP)) ((SP)+1)	I I	M2 M3
Arithmetic, logical, compare	ADD r	1 0	0 0 0	S S S		4	1	1	(A) ← (A) + (r)	O O O O O					
	ADD M	1 0	0 0 0	1 1 0	8 6	7	1	2	(A) ← (A) + (M) Where. M=(H) (L)	O O O O O	M	M4	(M)	I	M4
	ADI n	1 1	0 0 0 <B2>	1 1 0	C 6	7	2	2	(A) ← (A) + n	O O O O O			<B2>	I	M4
	ADC r	1 0	0 0 1	S S S		4	1	1	(A) ← (A) + (r) + (CY2)	O O O O O					
	ADC M	1 0	0 0 1	1 1 0	8 E	7	1	2	(A) ← (A) + (M) + (CY2) Where. M=(H) (L)	O O O O O	M	M4	(M)	I	M4
	ACI n	1 1	0 0 1 <B2>	1 1 0	C E	7	2	2	(A) ← (A) + n + (CY2)	O O O O O			<B2>	I	M4
	DAD B	0 0	0 0 1	0 0 1	0 9	10	1	3	(H) (L) ← (H) (L) + (B) (C)	X X X O X					
	DAD D	0 0	0 1 1	0 0 1	1 9	10	1	3	(H) (L) ← (H) (L) + (D) (E)	X X X O X					
	DAD H	0 0	1 0 1	0 0 1	2 9	10	1	3	(H) (L) ← (H) (L) + (H) (L)	X X X O X					
	DAD SP	0 0	1 1 1	0 0 1	3 9	10	1	3	(H) (L) ← (H) (L) + (SP)	X X X O X					
	SUB r	1 0	0 1 0	S S S		4	1	1	(A) ← (A) − (r)	O O O O O					
	SUB M	1 0	0 1 0	1 1 0	9 6	7	1	2	(A) ← (A) − (M) Where. M=(H) (L)	O O O O O	M	M4	(M)	I	M4
	SUI n	1 1	0 1 0 <B2>	1 1 0	D 6	7	2	2	(A) ← (A) − n	O O O O O			<B2>	I	M4
	SBB r	1 0	0 1 1	S S S		4	1	1	(A) ← (A) − (r) − (CY2)	O O O O O					
	SBB M	1 0	0 1 1	1 1 0	9 E	7	1	2	(A) ← (A) − (M) − (CY2) Where. M=(H) (L)	O O O O O	M	M4	(M)	I	M4
	SBI n	1 1	0 1 1 <B2>	1 1 0	D E	7	2	2	(A) ← (A) − n − (CY2)	O O O O O			<B2>	I	M4
	ANA r	1 0	1 0 0	S S S		4	1	1	(A) ← (A) ∧ (r)	O O O 0 1					
	ANA M	1 0	1 0 0	1 1 0	A 6	7	1	2	(A) ← (A) ∧ (M) Where. M=(H) (L)	O O O 0 1	M	M4	(M)	I	M4
	ANI n	1 1	1 0 0 <B2>	1 1 0	E 6	7	2	2	(A) ← (A) ∧ n	O O O 0 1			<B2>	I	M4
	XRA r	1 0	1 0 1	S S S		4	1	1	(A) ← (A) ∀ (r)	O O O 0 0					
	XRA M	1 0	1 0 1	1 1 0	A E	7	1	2	(A) ← (A) ∀ (M) Where. M=(H) (L)	O O O 0 0	M	M4	(M)	I	M4
	XRI n	1 1	1 0 1 <B2>	1 1 0	E E	7	2	2	(A) ← (A) ∀ n	O O O 0 0			<B2>	I	M4
	ORA r	1 0	1 1 0	S S S		4	1	1	(A) ← (A) ∨ (r)	O O O 0 0					
	ORA M	1 0	1 1 0	1 1 0	B 6	7	1	2	(A) ← (A) ∨ (M) Where. M=(H) (L)	O O O 0 0	M	M4	(M)	I	M4
	ORI n	1 1	1 1 0 <B2>	1 1 0	F 6	7	2	2	(A) ← (A) ∨ n	O O O 0 0			<B2>	I	M4
	CMP r	1 0	1 1 1	S S S		4	1	1	(A) − (r)	O O O O O					
	CMP M	1 0	1 1 1	1 1 0	B E	7	1	2	(A) − (M) } Compare; Where. M=(H) (L)	O O O O O	M	M4	(M)	I	M4
	CPI n	1 1	1 1 1 <B2>	1 1 0	F E	7	2	2	(A) − n	O O O O O			<B2>	I	M4
Register increment/decrement	INR r	0 0	D D D	1 0 0		4	1	1	(r) ← (r) + 1	O O O X O					
	INR M	0 0	1 1 0	1 0 0	3 4	10	1	3	(M) ← (M) + 1 Where. M=(H) (L)	O O O X O	M	M4	(M)	I	M4
	DCR r	0 0	D D D	1 0 1		4	1	1	(r) ← (r) − 1	O O O X O					
	DCR M	0 0	1 1 0	1 0 1	3 5	10	1	3	(M) ← (M) − 1 Where. M=(H) (L)	O O O X O	M	M4	(M)	I	M4
	INX B	0 0	0 0 0	0 1 1	0 3	6	1	1	(B) (C) ← (B) (C) + 1	X X X X X					
	INX D	0 0	0 1 0	0 1 1	1 3	6	1	1	(D) (E) ← (D) (E) + 1	X X X X X					
	INX H	0 0	1 0 0	0 1 1	2 3	6	1	1	(H) (L) ← (H) (L) + 1	X X X X X					
	INX SP	0 0	1 1 0	0 1 1	3 3	6	1	1	(SP) ← (SP) + 1	X X X X X					
	DCX B	0 0	0 0 1	0 1 1	0 B	6	1	1	(B) (C) ← (B) (C) − 1	X X X X X					
	DCX D	0 0	0 1 1	0 1 1	1 B	6	1	1	(D) (E) ← (D) (E) − 1	X X X X X					
	DCX H	0 0	1 0 1	0 1 1	2 B	6	1	1	(H) (L) ← (H) (L) − 1	X X X X X					
	DCX SP	0 0	1 1 1	0 1 1	3 B	6	1	1	(SP) ← (SP) − 1	X X X X X					
Rotate & shift contents of accumulator	RLC	0 0	0 0 0	1 1 1	0 7	4	1	1	Left shift CY2 [ ] ← [A7 A6 ······ A1 A0] ←	X X X O X					
	RRC	0 0	0 0 1	1 1 1	0 F	4	1	1	Right shift CY2 [ ] [A7 A6 ······ A1 A0] →	X X X O X					
	RAL	0 0	0 1 0	1 1 1	1 7	4	1	1	Left shift CY2 [ ] ← [A7 A6 ······ A1 A0] ←	X X X O X					
	RAR	0 0	0 1 1	1 1 1	1 F	4	1	1	Right shift CY2 [ ] → [A7 A6 ······ A1 A0] →	X X X O X					
Accumu. compen.	CMA	0 0	1 0 1	1 1 1	2 F	4	1	1	(A) ← $\overline{(A)}$	X X X X X					
	DAA	0 0	1 0 0	1 1 1	2 7	4	1	1	Results of binary addition are adjusted to BCD	O O O O O					
Carry set	STC	0 0	1 1 0	1 1 1	3 7	4	1	1	(CY2) ← 1	X X X 1 X					
	CMC	0 0	1 1 1	1 1 1	3 F	4	1	1	(CY2) ← $\overline{(CY2)}$	X X X O X					

* : State is T_1 ** : State is T_2

8-BIT PARALLEL MICROPROCESSOR

Instr. class \ Item	Mnemonic	Instruction code D7 D6	D5 D4 D3	D2 D1 D0	16mal notatn	No. of states	No. of bytes	No. of cycles	Functions	Flags S Z P CY2 CY1	Address bus Contents	Address bus Mach. cycle*	Data bus Contents	Data bus I/O	Data bus Mach. cycle**
Jump	JMP m	1 1	0 0 0 <B2> <B3>	0 1 1	C 3	10	3	3	(PC)←m	X X X X X			<B2> <B3>	I I	M2 M3
	PCHL	1 1	1 0 1	0 0 1	E 9	6	1	1	(PC)←(H)(L)	X X X X X					
	JC m	1 1	0 1 1 <B2> <B3>	0 1 0	D A	10/7	3	3/2	(CY2) = 1	X X X X X					
	JNC m	1 1	0 1 0 <B2> <B3>	0 1 0	D 2	10/7	3	3/2	(CY2) = 0 If condition is true. (PC)←m	X X X X X	If condition is true.				
	JZ m	1 1	0 0 1 <B2> <B3>	0 1 0	C A	10/7	3	3/2	(Z) = 1	X X X X X			<B2> <B3>	I I	M2 M3
	JNZ m	1 1	0 0 0 <B2> <B3>	0 1 0	C 2	10/7	3	3/2	(Z) = 0	X X X X X					
	JP m	1 1	1 1 0 <B2> <B3>	0 1 0	F 2	10/7	3	3/2	(S) = 0 If condition is false. (PC)←(PC) + 3	X X X X X					
	JM m	1 1	1 1 1 <B2> <B3>	0 1 0	F A	10/7	3	3/2	(S) = 1	X X X X X					
	JPE m	1 1	1 0 1 <B2> <B3>	0 1 0	E A	10/7	3	3/2	(P) = 1	X X X X X					
	JPO m	1 1	1 0 0 <B2> <B3>	0 1 0	E 2	10/7	3	3/2	(P) = 0	X X X X X					
Subroutine call	CALL m	1 1	0 0 1 <B2> <B3>	1 0 1	C D	18	3	5	((SP)−1)((SP)−2)←(PC)+3, (PC)←m (SP) ← (SP)−2	X X X X X	 (SP)−1 (SP)−2	 M4 M5	<B2> <B3> (PC)+3 (PC)+3	I I O O	M2 M3 M4 M5
	RST n	1 1	A A A	1 1 1		12	1	3	((SP)−1)((SP)−2)←(PC)+1, (PC)←n×8, (SP) ← (SP)−2 Where 0≦n≦7	X X X X X	(SP)−1 (SP)−2	M4 M5	(PC)+1 (PC)+1	O O	M4 M5
	CC m	1 1	0 1 1 <B2> <B3>	1 0 0	D C	18/9	3	5/2	(CY2) = 1	X X X X X					
	CNC m	1 1	0 1 0 <B2> <B3>	1 0 0	D 4	18/9	3	5/2	(CY2) = 0 If condition is true.	X X X X X	If condition is true.				
	CZ m	1 1	0 0 1 <B2> <B3>	1 0 0	C C	18/9	3	5/2	(Z) = 1 ((SP)−1) ((SP)−2)←(PC) + 3 (PC)←m	X X X X X	 (SP)−1	 M4	<B2> <B3> (PC) + 3	I I O	M2 M3 M4
	CNZ m	1 1	0 0 0 <B2> <B3>	1 0 0	C 4	18/9	3	5/2	(Z) = 0 (SP)←(SP)−2	X X X X X	(SP)−2	M5	(PC) + 3	O	M5
	CP m	1 1	1 1 0 <B2> <B3>	1 0 0	F 4	18/9	3	5/2	(S) = 0	X X X X X					
	CM m	1 1	1 1 1 <B2> <B3>	1 0 0	F C	18/9	3	5/2	(S) = 1 If condition is false. (PC)←(PC)+ 3	X X X X X					
	CPE m	1 1	1 0 1 <B2> <B3>	1 0 0	E C	18/9	3	5/2	(P) = 1	X X X X X					
	CPO m	1 1	1 0 0 <B2> <B3>	1 0 0	E 4	18/9	3	5/2	(P) = 0	X X X X X					
Return	RET	1 1	0 0 1	0 0 1	C 9	10	1	3	(PC) ←((SP)+1)((SP)), (SP)←(SP)+2	X X X X X	(SP) (SP)+1	M4 M5	((SP)) ((SP)+1)	I I	M4 M5
	RC	1 1	0 1 1	0 0 0	D 8	12/6	1	3/1	(CY2) = 1 If condition is true	X X X X X	If condition is true.				
	RNC	1 1	0 1 0	0 0 0	D 0	12/6	1	3/1	(CY2) = 0	X X X X X					
	RZ	1 1	0 0 1	0 0 0	C 8	12/6	1	3/1	(Z) = 1 (PC)←((SP)+1) ((SP))	X X X X X	(SP)	M4	((SP))	I	M4
	RNZ	1 1	0 0 0	0 0 0	C 0	12/6	1	3/1	(Z) = 0 (SP)←(SP)+2	X X X X X	(SP)+1	M5	((SP)+1)	I	M5
	RP	1 1	1 1 0	0 0 0	F 0	12/6	1	3/1	(S) = 0	X X X X X					
	RM	1 1	1 1 1	0 0 0	F 8	12/6	1	3/1	(S) = 1 If condition is false.	X X X X X					
	RPE	1 1	1 0 1	0 0 0	E 8	12/6	1	3/1	(P) = 1 (PC)←(PC)+1	X X X X X					
	RPO	1 1	1 0 0	0 0 0	E 0	12/6	1	3/1	(P) = 0	X X X X X					
Input/ output control	IN n	1 1	0 1 1 <B2>	0 1 1	D B	10	2	3	(A) ← (Input buffer) ← (Input device of number n) (Input data)	X X X X X	 <B2> B2>	 M5	<B2> (Input data)	O I	M4 M5
	OUT n	1 1	0 1 0 <B2>	0 1 1	D 3	10	2	3	(Output device of number n) ← (A)	X X X X X	 <B2> <B2>	 M5	<B2> (A)	O O	M4 M5
Interrupt control	E I	1 1	1 1 1	0 1 1	F B	4	1	1	(INTE)←1	X X X X X					
	D I	1 1	1 1 0	0 1 1	F 3	4	1	1	(INTE)←0	X X X X X					
Stack control	PUSH PSW	1 1	1 1 0	1 0 1	F 5	12	1	3	((SP)−1)←(A), ((SP)−2)←(F) (SP) ←(SP)−2	X X X X X	(SP) 1 (SP)−2	M4 M5	(A) (F)	O O	M4 M5
	PUSH B	1 1	0 0 0	1 0 1	C 5	12	1	3	((SP)−1)←(B), ((SP)−2)←(C) (SP) ←(SP)−2	X X X X X	(SP)−1 (SP)−2	M4 M5	(B) (C)	O O	M4 M5
	PUSH D	1 1	0 1 0	1 0 1	D 5	12	1	3	((SP)−1)←(D), ((SP)−2)←(E) (SP) ←(SP)−2	X X X X X	(SP)−1 (SP)−2	M4 M5	(D) (E)	O O	M4 M5
	PUSH H	1 1	1 0 0	1 0 1	E 5	12	1	3	((SP)−1)←(H), ((SP)−2)←(L) (SP) ←(SP)−2	X X X X X	(SP)−1 (SP)−2	M4 M5	(H) (L)	O O	M4 M5
	POP PSW	1 1	1 1 0	0 0 1	F 1	10	1	3	(F) ←((SP)), (A)←((SP)+1) (SP) ←(SP)+2	O O O O O	(SP) (SP)+1	M4 M5	(SP)) ((SP)+1)	I I	M4 M5
	POP B	1 1	0 0 0	0 0 1	C 1	10	1	3	(C) ←((SP)), (B)←((SP)+1) (SP) ←(SP)+2	X X X X X	(SP) (SP)+1	M4 M5	((SP)) ((SP)+1)	I I	M4 M5
	POP D	1 1	0 1 0	0 0 1	D 1	10	1	3	(E) ←((SP)), (D)←((SP)+1) (SP) ←(SP)+2	X X X X X	(SP) (SP)+1	M4 M5	((SP)) ((SP)+1)	I I	M4 M5
	POP H	1 1	1 0 0	0 0 1	E 1	10	1	3	(L) ←((SP)), (H)←((SP)+1) (SP) ←(SP)+2	X X X X X	(SP) (SP)+1	M4 M5	((SP)) ((SP)+1)	I I	M4 M5
Others	HLT	0 1	1 1 0	1 1 0	7 6	5	1	1	(PC) ←(PC)+1	X X X X X					
	NOP	0 0	0 0 0	0 0 0	0 0	4	1	1	(PC) ←(PC)+1	X X X X X					
Mask set instructions	RIM	0 0	1 0 0	0 0 0	2 0	4	1	1	All RST interrupt masks, any pending RST interrupt requests, and the serial input data from the SID pin are read into the accumlator.	X X X X X					
	SIM	0 0	1 1 0	0 0 0	3	4	1	1	Mask is enabled (or disabled) to the RST interrupt corresponding to the contents (bit pattern) of the accumulator. The serial output is enabled and the serial output bit is loaded into the SOD latch.	X X X X X					

* : State is T1. ** : State is T2.

Symbol	Meaning	Symbol	Meaning	Symbol	Meaning
r	Register	S S S or D D D	Bit pattern designating register or memory. Where M (H)(L)	←	Data is transferred in direction shown
m	Two-byte data			( )	Contents of register or memoy location
n	One-byte data			V	Inclusive OR
<B2>	Second byte of instruction			∀	Exclusive OR
<B3>	Third byte of instruction			Λ	Logical AND
AAA	Binary representation for RST instruction n			—	1's complement
F	8-bit data from the most to the least significant bit S, Z, X, CY1, X, P, X, CY2			X	Content of flag is not changed after execution
PC	Program counter			O	Content offlag is set or reset after execution
SP	Stack pointer			I	Input mode
				O	Output mode

Register or memory	S S S or D D D
B	0 0 0
C	0 0 1
D	0 1 0
E	0 1 1
H	1 0 0
L	1 0 1
M	1 1 0
A	1 1 1

INSTRUCTION CODE LIST

$D_3 \sim D_0$ \ $D_7 \sim D_4$	Hexadecimal notation	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NOP	(−)	RIM	SIM	MOV B, B	MOV D, B	MOV H, B	MOV M, B	ADD B	SUB B	ANA B	ORA B	RNZ	RNC	RPO	RP
0001	1	LXI B	LXI D	LXI H	LXI SP	MOV B, C	MOV D, C	MOV H, C	MOV M, C	ADD C	SUB C	ANA C	ORA C	POP B	POP D	POP H	POP PSW
0010	2	STAX B	STAX D	SHLD	STA	MOV B, D	MOV D, D	MOV H, D	MOV M, D	ADD D	SUB D	ANA D	ORA D	JNZ	JNC	JPO	JP
0011	3	INX B	INX D	INX H	INX SP	MOV B, E	MOV D, E	MOV H, E	MOV M, E	ADD E	SUB E	ANA E	ORA E	JMP	OUT	XTHL	DI
0100	4	INR B	INR D	INR H	INR M	MOV B, H	MOV D, H	MOV H, H	MOV M, H	ADD H	SUB H	ANA H	ORA H	CNZ	CNC	CPO	CP
0101	5	DCR B	DCR D	DCR H	DCR M	MOV B, L	MOV D, L	MOV H, L	MOV M, L	ADD L	SUB L	ANA L	ORA L	PUSH B	PUSH D	PUSH H	PUSH PSW
0110	6	MVI B	MVI D	MVI H	MVI M	MOV B, M	MOV D, M	MOV H, M	HLT	ADD M	SUB M	ANA M	ORA M	ADI	SUI	ANI	ORI
0111	7	RLC	RAL	DAA	STC	MOV B, A	MOV D, A	MOV H, A	MOV M, A	ADD A	SUB A	ANA A	ORA A	RST 0	RST 2	RST 4	RST 6
1000	8	(−)	(−)	(−)	(−)	MOV C, B	MOV E, B	MOV L, B	MOV A, B	ADC B	SBB B	XRA B	CMP B	RZ	RC	RPE	RM
1001	9	DAD B	DAD D	DAD H	DAD SP	MOV C, C	MOV E, C	MOV L, C	MOV A, C	ADC C	SBB C	XRA C	CMP C	RET	(−)	PCHL	SPHL
1010	A	LDAX B	LDAX D	LHLD	LDA	MOV C, D	MOV E, D	MOV L, D	MOV A, D	ADC D	SBB D	XRA D	CMP D	JZ	JC	JPE	JM
1011	B	DCX B	DCX D	DCX H	DCX SP	MOV C, E	MOV E, E	MOV L, E	MOV A, E	ADC E	SBB E	XRA E	CMP E	(−)	IN	XCHG	EI
1100	C	INR C	INR E	INR L	INR A	MOV C, H	MOV E, H	MOV L, H	MOV A, H	ADC H	SBB H	XRA H	CMP H	CZ	CC	CPE	CM
1101	D	DCR C	DCR E	DCR L	DCR A	MOV C, L	MOV E, L	MOV L, L	MOV A, L	ADC L	SBB L	XRA L	CMP L	CALL	(−)	(−)	(−)
1110	E	MVI C	MVI E	MVI L	MVI A	MOV C, M	MOV E, M	MOV L, M	MOV A, M	ADC M	SBB M	XRA M	CMP M	ACI	SBI	XRI	CPI
1111	F	RRC	RAR	CMA	CMC	MOV C, A	MOV E, A	MOV L, A	MOV A, A	ADC A	SBB A	XRA A	CMP A	RST 1	RST 3	RST 5	RST 7

This list shows the machine codes and corresponding machine instruction. $D_3 \sim D_0$ indicate the low-order 4 bits of the machine code and $D_7 \sim D_4$ indicate the high-order 4 bits. Hexadecimal numbers are also used to indicate this code. The instruction may consists of one, two, or three bytes, but only the first byte is listed.

- (light shading) indicates a three-byte instruction.
- (dark shading) indicates a two-byte instruction.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.3～7	V
V_I	Input voltage		−0.3～7	V
Pd	Power dissipation	Ta=25°C	1.5	W
Topr	Operating free-air temperature range		0～70	°C
Tstg	Storage temperature range		−65～150	°C

RECOMMENDED OPERATING CONDITIONS (Ta= 0～70°C unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{IH}	High-level input voltage	2.0		V_{CC}+0.5	V
V_{IL}	Low-level input voltage	−0.5		0.8	V
$V_{IH(\overline{RESIN})}$	High-level reset input voltage	2.4		V_{CC}+0.5	V
$V_{IL(\overline{RESIN})}$	Low-level reset input voltage	−0.5		0.8	V

ELECTRICAL CHARACTERISTICS (Ta=0～70°C, V_{CC}=5V±5%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OL}	Low-level output voltage	I_{OL}=2mA			0.45	V
V_{OH}	High-level output voltage	I_{OH}= −400μA	2.4			V
I_{CC}	Supply current from V_{CC}				170	mA
I_I	Input leak current, except $\overline{RESIN}$ (Note 1)	V_I=V_{CC}	−10		10	μA
I_{OZL}	Output floating leak current	0.45V≦V_O≦V_{CC}	−10		10	μA
$V_{IH}-V_{IL}$	Hysterisis, $\overline{RESIN}$ input		0.25			V

Note 1: The input $\overline{RESET\ IN}$ is pulled up to Vcc with the resistor 3kΩ (typ) when V_I≧$V_{IH(\overline{RESIN})}$

TIMING REQUIREMENTS (Ta= 0～70°C, V_{CC}= 5 V± 5%, V_{SS}= 0 V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	M5L 8085AP, S			Unit
			Limits			
			Min	Typ	Max	
$t_{C(CLK)}$	Clock cycle time	T_{CYC}	320		2000	ns
$t_{SU(DA-AD)}$	DA input setup time	$-t_{AD}$	−575			ns
$t_{SU(DA-\overline{RD})}$	DA input setup time	$-t_{RD}$	−300			ns
$t_{h(DA-\overline{RD})}$	DA input hold time	t_{RDH}	0			ns
$t_{SU(RDY-AD)}$	READY input setup time	$-t_{ARY}$	−220			ns
$t_{SU(RDY-CLK)}$	READY input setup time	$-t_{RYS}$			−110	ns
$t_{h(RDY-CLK)}$	READY input hold time	t_{RYH}	0			ns
$t_{SU(DA-ALE)}$	DA input setup time	$-t_{LDR}$	−460			ns
$t_{SU(HLD-CLK)}$	HOLD input setup time	t_{HDS}	170			ns
$t_{h(HLD-CLK)}$	HOLD input hold time	t_{HDH}	0			ns
$t_{SU(INT-CLK)}$	Interrupt setup time	t_{INS}	160			ns
$t_{h(INT-CLK)}$	Interrupt hold time	t_{INH}	0			ns
$t_{SU(RDY-ALE)}$	READY input setup time	$-t_{LRY}$	−110			ns

Note 2: The input voltage level of the input voltage level is V_{IL}=0.45V and V_{IH}=2.4V

SWITCHING CHARACTERISTICS (T_a = 0 ~ 70℃, V_{CC} = 5 V ± 5 %, V_{SS} = 0V, unless otherwise noted)

Symbol	Parameter		Alternative symbol	Limits			Unit
				Min	Typ	Max	
$t_{W(\overline{CLK})}$	CLK output low-level pulse width		t_1	80			ns
$t_{W(CLK)}$	CLK output high-level pulse width		t_2	120			ns
$t_{r(CLK)}$	CLK output rise time		t_r			30	ns
$t_{f(CLK)}$	CLK output fall time		t_f			30	ns
$t_{d(X_1-CLK)}$	Delay time, X_1 to CLK		t_{XKR}	30		120	ns
$t_{d(X_1-\overline{CLK})}$	Delay time, X_1 to $\overline{CLK}$		t_{XKF}	30		150	ns
$t_{d(AD-ALE)}$	Delay time, address output to ALE signal	AD_0~AD_7	t_{AL}	90			ns
		A_8 ~ A_{15}		115			
$t_{d(ALE-AD)}$	Delay time, ALE signal to address output		t_{LA}	100			ns
$t_{W(ALE)}$	ALE pulse width		t_{LL}	140			ns
$t_{d(ALE-CLK)}$	Delay time, ALE to CLK		t_{LCK}	100			ns
$t_{d(ALE-CONT)}$	Delay time, ALE to control signal		t_{LC}	130			ns
$t_{DXZ(\overline{RD}-AD)}$	Address disable time from read		t_{AFR}			0	ns
$t_{DZX(\overline{RD}-AD)}$	Address enable time from read		t_{RAE}	150			ns
$t_{d(\overline{CONT}-AD)}$	ADdress valid time after control signal		t_{CA}	120			ns
$t_{d(DA-\overline{WR})}$	Delay time, data output to $\overline{WR}$ signal		t_{DW}	420			ns
$t_{d(\overline{WR}-DA)}$	Delay time, $\overline{WR}$ signal to data output		t_{WD}	100			ns
$t_{W(\overline{CONT})}$	Control signal pulse width		t_{CC}	400			ns
$t_{d(CONT-ALE)}$	Delay time, CLK to ALE signal		t_{CL}	50			ns
$t_{d(CLK-HLDA)}$	Delay time, CLK to HLDA signal		t_{HACK}	110			ns
$t_{DXZ(HLDA-BUS)}$	Bus disable time from HLDA		t_{HABF}			210	ns
$t_{DZX(HLDA-BUS)}$	Control signal disable time		t_{HABE}			210	ns
$t_{d(\overline{CONT}-CONT)}$	Control signal disable time		t_{RV}	400			ns
$t_{d(AD-CONT)}$	Delay time, address output to control signal	AD_0~AD_7	t_{AC}	240			ns
		A_8 ~ A_{15}		270			
$t_{d(ALE-DA)}$	Delay time, ALE to data output		t_{LDW}			200	ns
$t_{d(WRHL-DA)}$	Delay time, $\overline{WR}$ signal to data output		t_{WDL}			40	ns

Note 3 at A_8 ~ A_{15}, and IO/$\overline{M}$ $t_{d(AD-CONT)}$ after the release of the high-impedance state is 100ns.

4 Conditions of measurement : M5L 8085AP, S $t_{C(CLK)} \geqq 320$ns, C_L = 150pF
M5L 8085AP-20, S-20 $t_{C(CLK)} \geqq 500$ns, C_L = 150pF

5 Reference evel for the input/output voltage is V_{OL} = 0.8V, V_{OH} = 2V.

6 $t_{W(\overline{CLK})}$, $t_{W(CLK)}$ are 100ns(Min), 150ns(Min) respectively when 50pF+1TTL loaded

Parameters described in the timing requirements and switching characteristics take relevant values in accordance with the relational expression shown in Table 1 when the frequency is varied.

Table 1 Relational expression with the frequency T ($t_{C(CLK)}$) in the M5L8085A

TIMMING REQUIREMENTS (T_a = 0~70°C, V_{CC} = 5V±5%, V_{SS} = 0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Test conditions	Relational expression (Note 6)	Limit
$t_{su(DA-AD)}$	DA input setup time	$-t_{AD}$		225−(5/2+N)T	Min
$t_{su(DA-\overline{RD})}$	DA input setup time	$-t_{RD}$		180−(3/2+N)T	Min
$t_{su(RDY-AD)}$	READY input setup time	$-t_{ARY}$		260−(3/2)T	Min
$t_{su(DA-ALE)}$	DA input setup time	$-t_{LDR}$		180−2T	Min

Note7. N indicates the total number of wait cycles.
$T = t_{C(CLK)}$

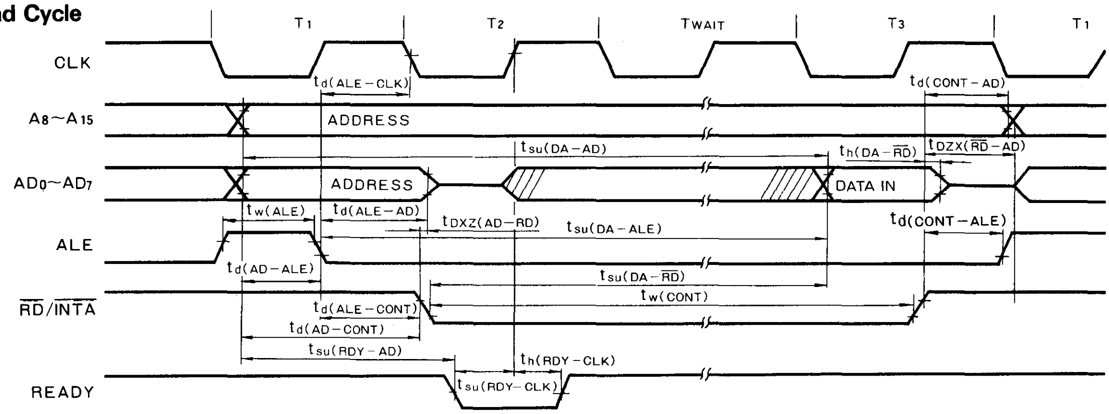
SWITCHING CHARACTERISTICS (Ta = 0~70°C, Vcc = 5V±5%, Vss = 0V, unless otherwise noted)

Symbol	Parameter		Alternative symbol	Test conditions	Relational expression (Note 6)	Limit
$t_{w(\overline{CLK})}$	CLK output low-level pulse width		t_1	$C_L = 150pF$	$(1/2)T - 80$	Min
$t_{w(CLK)}$	CLK output high-level pulse width		t_2		$(1/2)T - 40$	Min
$t_{d(AD\text{-}ALE)}$	Delay time, address output to ALE signal	$AD_0 \sim AD_7$	t_{AL}		$(1/2)T - 70$	Min
		$A_8 \sim A_{15}$			$(1/2)T - 45$	
$t_{d(ALE\text{-}AD)}$	Delay time, ALE signal to address output		t_{LA}		$(1/2)T - 60$	Min
$t_{w(ALE)}$	ALE pulse width		t_{LL}		$(1/2)T - 20$	Min
$t_{d(ALE\text{-}CLK)}$	Delay time, ALE to CLK		t_{LCK}		$(1/2)T - 60$	Min
$t_{d(ALE\text{-}\overline{CONT})}$	Delay time, ALE to control signal		t_{LC}		$(1/2)T - 30$	Min
$t_{DZX(\overline{RD}\text{-}AD)}$	Address enable time from read		t_{RAE}		$(1/2)T - 10$	Min
$t_{d(\overline{CONT}\text{-}AD)}$	Address valid time after control signal		t_{CA}		$(1/2)T - 40$	Min
$t_{d(DA\text{-}\overline{WR})}$	Delay time, data output to $\overline{WR}$ signal		t_{DW}		$(3/2+N)T - 60$	Min
$t_{d(\overline{WR}\text{-}DA)}$	Delay time, $\overline{WR}$ signal to data output		t_{WD}		$(1/2)T - 60$	Min
$t_{w(\overline{CONT})}$	Control signal pulse width		t_{CC}		$(3/2+N)T - 80$	Min
$t_{d(CONT-ALE)}$	Delay time, CONT to ALE signal		t_{CL}		$(1/2)T - 110$	Min
$t_{d(CLK\text{-}HLDA)}$	Delay time, CLK to HLDA signal		t_{HACK}		$(1/2)T - 50$	Min
$t_{DXZ(HLDA\text{-}BUS)}$	Bus disable time from HLDA		t_{HABF}		$(1/2)T + 50$	Max
$t_{DZX(HLDA\text{-}BUS)}$	Bus enable time from HLDA		t_{HABE}		$(1/2)T + 50$	Max
$t_{d(\overline{CONT}\text{-}CONT)}$	Control signal disable time		t_{RV}		$(3/2)T - 80$	Min
$t_{d(AD\text{-}\overline{CONT})}$	Delay time, address output to control signal	$AD_0 \sim AD_7$	t_{AC}		$T - 80$	Min
		$A_8 \sim A_{15}$			$T - 50$	

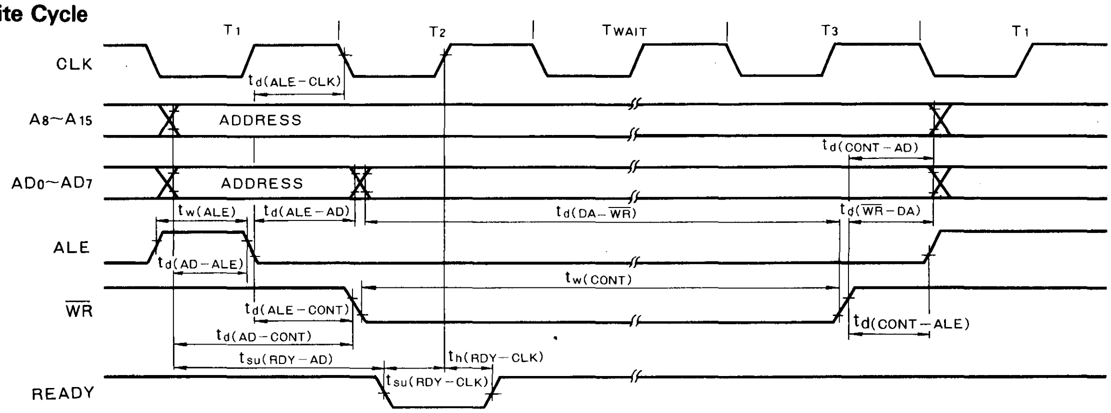
7

TIMING DIAGRAM

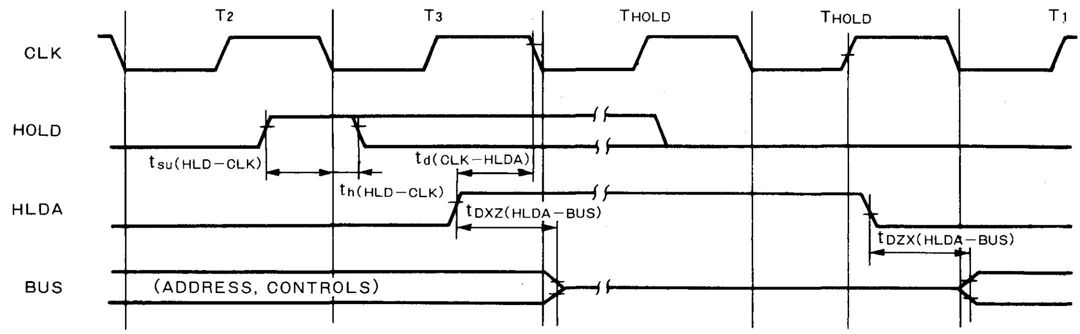
Read Cycle



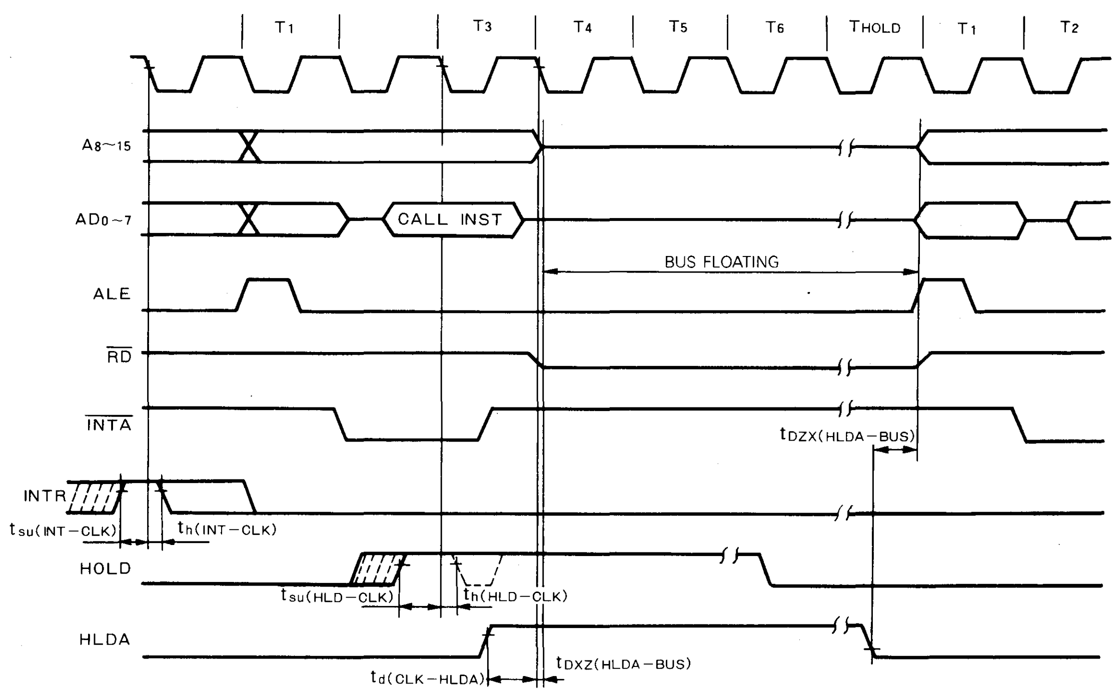
Write Cycle



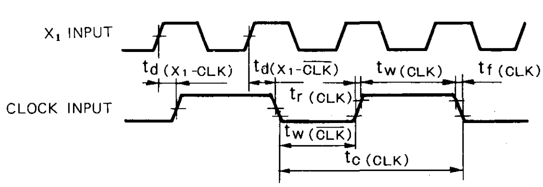
Hold Cycle



Interrupt and Hold Cycle



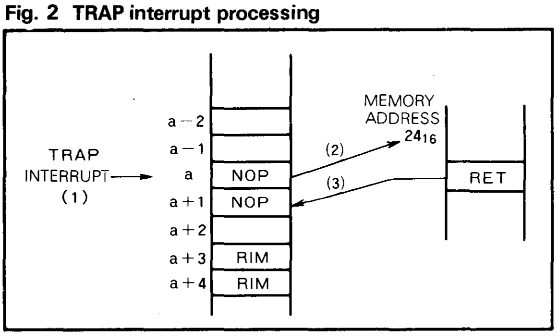
Clock Output Timing Waveform



TRAP INTERRUPT AND RIM INSTRUCTIONS

TRAP generates interrupts regardless of the interrupt enable filp-flop (INTE FF). The current state of the INTE FF is stored in flip flop A (AFF) of the CPU and then the INTE FF is reset. The first RIM instruction after the generation of a TRAP interrupt differs in function from the ordinary RIM instruction. That is, the bit 3 (INTE FF information) in the accumulator ($(A)_3$) after the execution of the RIM instruction contains the contents of the A FF, regardless of the state of the INTE FF at the time the RIM instruction is executed. These details are shown in Fig.2, Tables 1 and 2.

Fig. 2 TRAP interrupt processing



Below are the explanations of Fig. 2.

1. The TRAP interrupt request is issued while the instruction in address a is being executed.
2. The TRAP interrupt causes the same action as an RST instruction and then jumps to address 24_{16}.
3. It returns to address a+1 after executing the RET instruction.

Table 1 shows the information in the INTE FF when the instructions EI and/or DI are executed at addresses a−1 and a+2.

Fig. 3 is a flow chart of the TRAP interrupt processing routine.

Fig. 3 TRAP interrupt processing routine

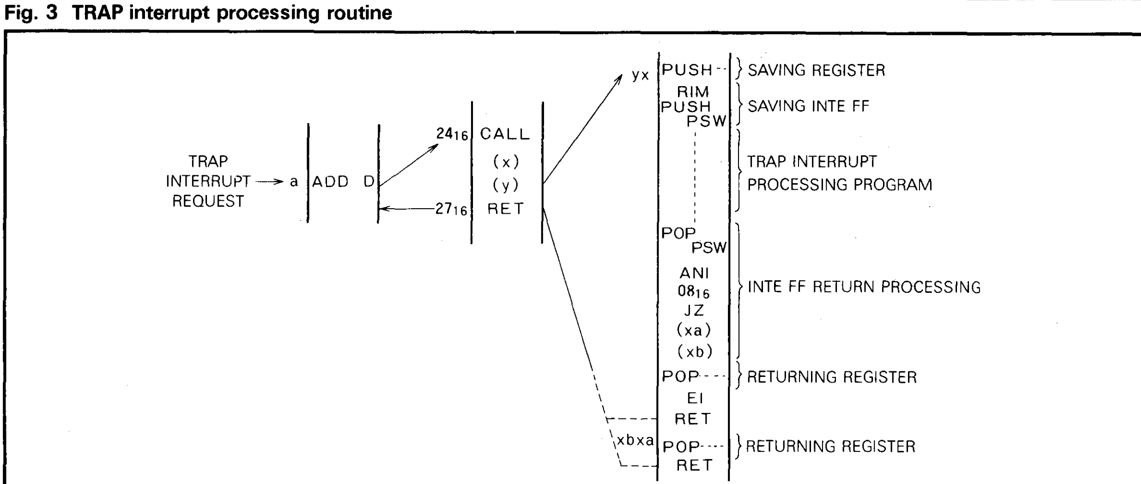
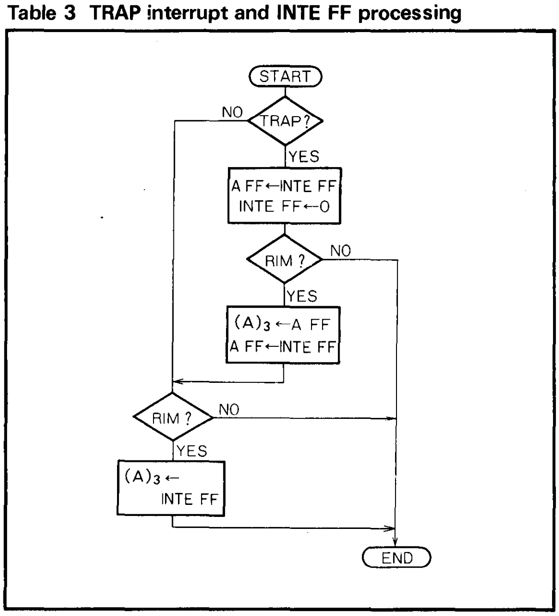


Table 2 TRAP interrupt and RIM instructions

Condition \ Number	1	2	3	4	5	6
Instruction in address a−1	EI	EI	EI	DI	DI	DI
Instruction in address a+2	EI	NOP	DI	EI	NOP	DI
Contents of $(A)_3$ after the execution of the RIM instruction in address a+3	1	1	1	0	0	0
State of INTE FF after the execution of the RIM instruction in address a+3	1	0	0	1	0	0
Contents of $(A)_3$ after the execution of the RIM instruction in address a+4	1	0	0	1	0	0
State of INTE FF after the execution of the RIM instruction in address a+4	1	0	0	1	0	0

Note 3: The contents of $(A)_3$ after the excution of the RIM instruction is an information of the INTE FF The INTE FF assumes state "1" when it is in the EI state, and "0" when it is in the DI state.

Table 3 TRAP interrupt and INTE FF processing



MITSUBISHI
ELECTRIC

PULL-UP OF THE $\overline{\text{RESET IN}}$ INPUT

In order to increase the noise margin, the $\overline{\text{RESET IN}}$ input terminal is pulled up by about 3kΩ (typ) when the condition $V_I \geqq V_{IH(\overline{RESIN})}$ is satisfied. Fig. 4 is a connection diagram of the $\overline{\text{RESET IN}}$ input, and Fig. 5 shows the relation between input voltage and input current.

Fig. 4 Connections of $\overline{\text{RESET IN}}$ input

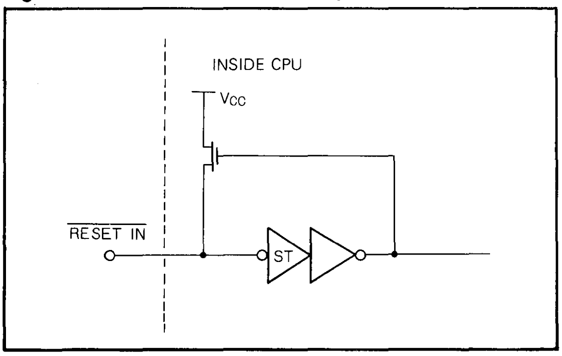
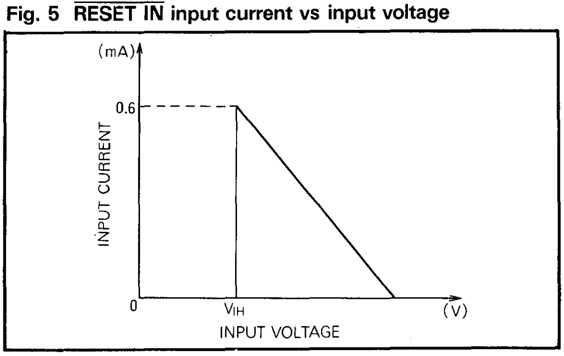


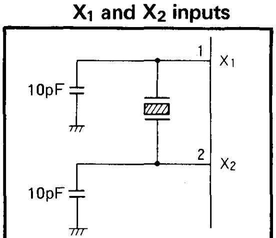
Fig. 5 $\overline{\text{RESET IN}}$ input current vs input voltage



DRIVING CIRCUIT OF X_1 AND X_2 INPUTS

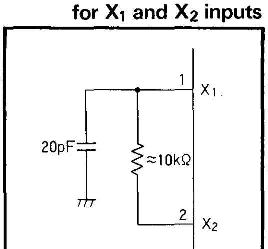
Input terminals, X_1 and X_2 of the M5L8085A can be driven by either a crystal, RC network, or external clock. Since the drive clock frequency is divided to 1/2 internally, the input frequency required is twice the actual execution frequency (6MHz for the M5L8085A, which is operated at 3MHz). Figs. 6 and 7 are typical connection diagrams for a crystal and CR circuit respectively.

Fig. 6 Connections when crystal is used for X_1 and X_2 inputs



OSCILLATION FREQUENCY:
A 1~6 MHz PARALLEL RESONANT CRYSTAL OSCILLATOR IS USED.

Fig. 7 Connections when RC network is used for X_1 and X_2 inputs



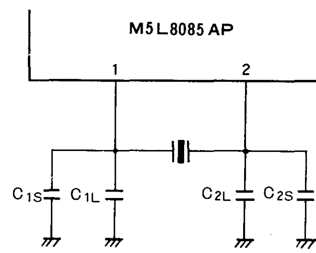
OSCILLATION FREQUENCY:
ABOUT 3MHz

Conditions for Using a Quartz Crystal Element

1. Quartz Crystal Specifications

- Parallel resonance
- The frequency is 2 times the operation frequency (2~6.25MHz)
- Internal load capacitance: Approx. 16pF
- Parallel capacitance: Below 7pF
- Equivalent resistance: Below 75Ω (for operation above 4MHz)
- For operation in the range 2~4MHz, the resistance should be made as small as possible.
- Drive capability: Above 5mW (the power at which the crystal will be destoryed)

2. External Circuitry



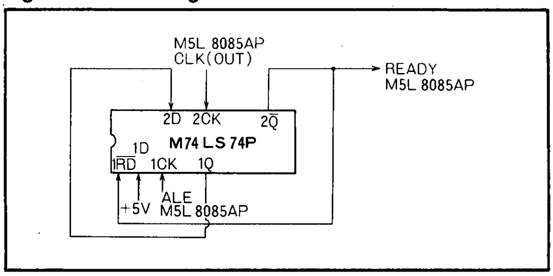
C_{1S}: Wiring capacitance of pin X_1
C_{2S}: Wiring capacitance of pin X_2
C_{1S}: External capacitance at pin X_1
C_{2L}: External capacitance at pin X_2
$C_1 = C_{1L} + C_{1S}$
$C_2 = C_{2L} + C_{2S}$

- For operation above 4MHz:
  $C_1 = C_2 = 10pF$
- For operation below 4MHz:
  $C_1 = C_2 = 15pF$

WAIT STATE GENERATOR

Fig. 8 shows a typical 1-wait state generator for low speed RAM and ROM applications.

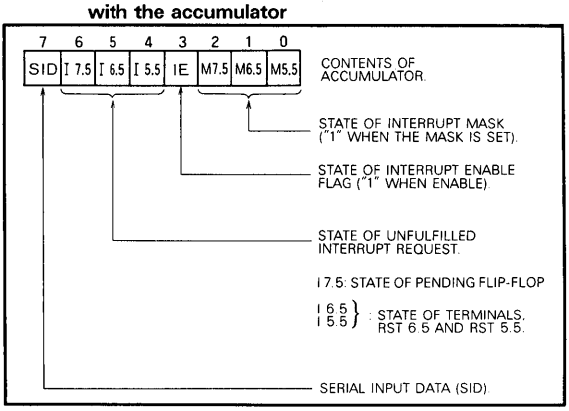
Fig. 8 1-wait state generator



RELATION OF RIM AND SIM INSTRUCTIONS WITH THE ACCUMULATOR (SUPPLEMENTARY DESCRIPTION).

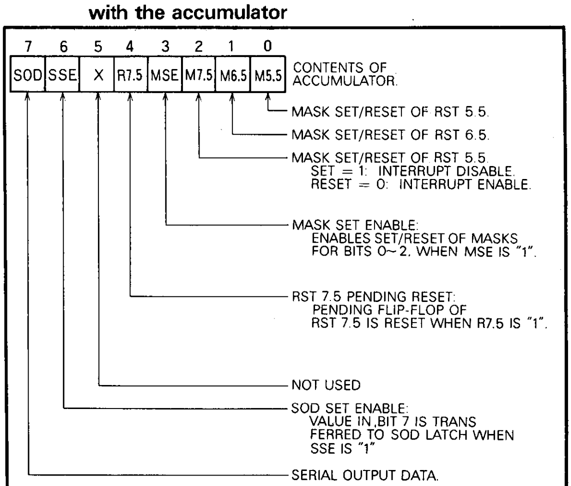
The contents of the accumulator after the execution of a RIM instruction is shown in Table 4.

Table 4 Relation of the instruction RIM with the accumulator



The contents of the accumulator after the execution of a SIM instruction is shown in Table 5.

Table 5 Relation of the SIM instruction with the accumulator



M5L 8212P

8-BIT INPUT/OUTPUT PORT WITH 3-STATE OUTPUT

DESCRIPTION

The M5L8212P is an input/output port consisting of an 8-bit latch with 3-state output buffers along with control and device selection logic. Also included is a service request flip-flop for the generation and control of interrupts to a microprocessor. It is fabricated using bipolar Schottky TTL technology.

FEATURES

- Parallel 8-bit data register and buffer
- Service request flip-flop for interrupt generation
- Three-state outputs
- Low input load current: I_{IL} = absolute -250μA (max)
- High output sink current: I_{OL} = 16mA (max)
- High-level output voltage for direct interface to a M5L8080AP, S CPU: V_{OH} = 3.65V (min)
- Interchangeable with Intel's 8212 in terms of electrical characteristics and pin configuration

APPLICATIONS

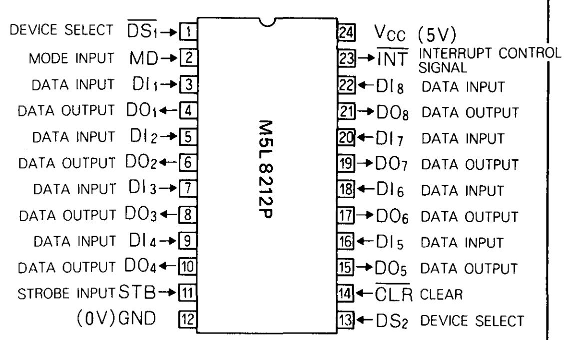
- Input/output port for a M5L8080AP, S
- Latches, gate buffers or multiplexers
- Peripheral and input/output functions for microcomputer systems

FUNCTION

Device select 1 ($\overline{DS_1}$) and device select 2 (DS_2) are used for chip selection when the mode input MD is low. When $\overline{DS_1}$ is low and DS_2 is high, the data in the latches is transferred to the data outputs $DO_1 \sim DO_8$; and the service request flip-flop SR is set. Also, the strobed input STB is active, the data inputs $DI_1 \sim DI_8$ are latched in the data latches, and the service request flip-flop SR is reset.

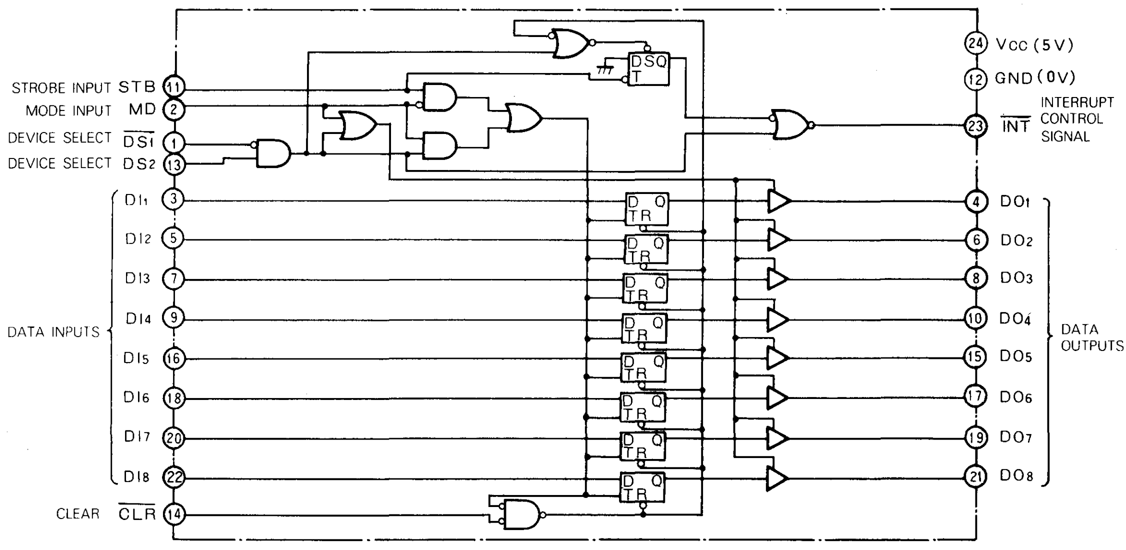
When MD is high, the data in the data latches is transferred to the data outputs. When $\overline{DS_1}$ is low and DS_2 is high, the data inputs are latched in the data latches. The low-level clear input $\overline{CLR}$ resets the data latches and sets the service request flip-flop SR, but the state of the output buffers is not changed.

PIN CONFIGURATION (TOP VIEW)



Outline 24P1

BLOCK DIAGRAM



8-BIT INPUT/OUTPUT PORT WITH 3-STATE OUTPUT

ABSOLUTE MAXIMUM RATINGS (Ta = 0 ~ 75°C unless otherwise noted)

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage		7.0	V
V_I	Input voltage, $\overline{DSI}$, MD inputs		V_{CC}	V
V_I	Input voltage, all other inputs except $\overline{DSI}$, MD		5.5	V
V_O	Output voltage		V_{CC}	V
P_d	Power dissipation		800	mW
T_{opr}	Operating free-air temperature range		0 ~ 75	°C
T_{stg}	Storage temperature range		−55 ~ 125	°C

RECOMMENDED OPERATING CONDITIONS (Ta = 0 ~ 75°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5.0	5.25	V
I_{OH}	High-level output current			−1	mA
I_{OL}	Low-level output current			16	mA

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 75°C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{IH}	High-level input voltage		2			V
V_{IL}	Low-level input voltage				0.85	V
V_{IC}	Input clamp voltage	$V_{CC}=4.75V$, $I_{IC}=-5mA$			−1	V
V_{OH}	High-level output voltage	$V_{CC}=4.75V$, $V_{IH}=2V$, $V_{IL}=0.85V$, $I_{OH}=-1mA$	3.65			V
V_{OL}	Low-level output voltage	$V_{CC}=4.75V$, $V_{IH}=2V$, $V_{IL}=0.85V$, $I_{OL}=16mA$			0.5	V
I_{OZ}	Three-state output current	$V_{CC}=5.25V$, $V_{IH}=2V$, $V_{IL}=0.85V$, $V_C=5.25V$			20	μA
I_{OZ}	Three-state output current	$V_{CC}=5.25V$, $V_{IH}=2V$, $V_{IL}=0.85V$, $V_O=0.5V$			−20	μA
I_{IH}	High-level input current, STB, DS2, $\overline{CLR}$, $DI_1 \sim DI_8$ inputs	$V_{CC}=5.25V$, $V_I=5.25V$			10	μA
I_{IH}	High-level input current, MD input	$V_{CC}=5.25V$, $V_I=5.25V$			30	μA
I_{IH}	High-level input current, $\overline{DS1}$ input	$V_{CC}=5.25V$, $V_I=5.25V$			40	μA
I_{IL}	Low-level input current, STB, DS2, $\overline{CLR}$, $DI_1 \sim DI_8$ inputs	$V_{CC}=5.25V$, $V_I=0.5V$			−0.25	mA
I_{IL}	Low-level input current, MD input	$V_{CC}=5.25V$, $V_I=0.5V$			−0.75	mA
I_{IL}	Low-level input current, $\overline{DS1}$ input	$V_{CC}=5.25V$, $V_I=0.5V$			−1	mA
I_{OS}	Short-circuit output current (Note 3)	$V_{CC}=5.25V$	−20		−65	mA
I_{CC}	Supply current from V_{CC}	$V_{CC}=5.25V$			130	mA

Note 1 : All voltages are with respect to GND terminal. Reference voltage (pin 12) is considered as 0V, and all maximum and minimum values are defined in absolute values.
2 : Current flowing into an IC is positive; out is negative. The maximum and minimum values are defined in absolute values.
3 : All measurements should be done quickly, and two outputs should not be measured at the same time.

TIMING REQUIREMENTS (Ta = 25°C, V_{CC} = 5 V, unless otherwise noted)

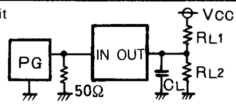
Symbol	Parameter	Test conditions	Limits			Unit
				Typ	Max	
$t_{w(DS2)}$	Input pulse width, $\overline{DS1}$, DS2 and STB		30			ns
$t_{w(CLR)}$	Input pulse width $\overline{CLR}$		45			ns
$t_{su(DA)}$	Data setup time with respect to $\overline{DS1}$, DS2 and STB		15			ns
$t_{h(DA)}$	Data hold time with respect to $\overline{DS1}$, DS2 and STB		20			ns

8-BIT INPUT/OUTPUT PORT WITH 3-STATE OUTPUT

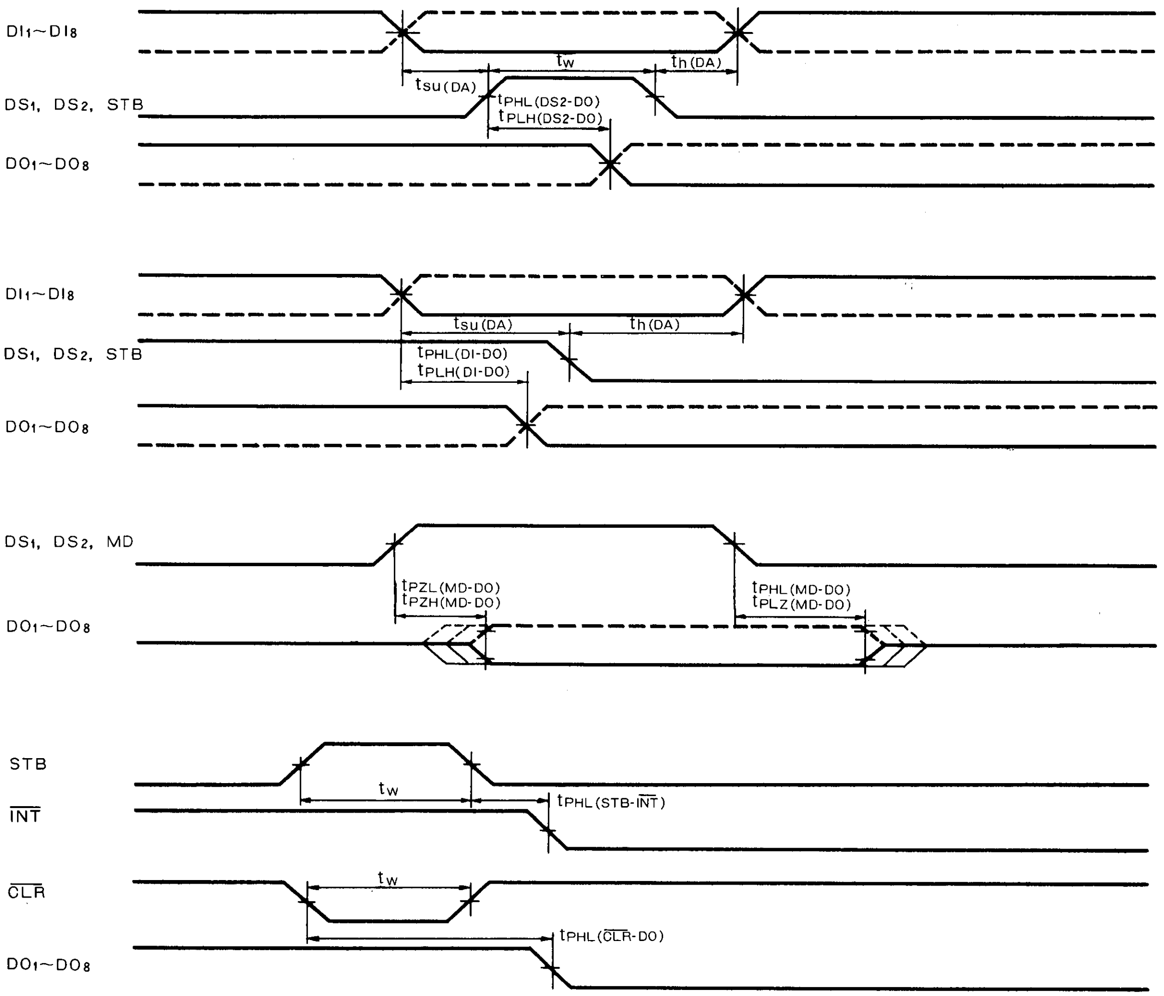
SWITCHING CHARACTERISTICS (Ta = 25°C, V_{CC} = 5V, unless otherwise noted)

Symbol	Parameter	Test conditions (Note 4)	Limits			Unit
			Min	Typ	Max	
$t_{PHL(DI-DO)}$ $t_{PLH(DI-DO)}$	High-to-low-level and low-to-high-level output propagation time, from input DI to output DO	C_L = 30pF, R_{L1} = 300Ω, R_{L2} = 600Ω			35	ns
$t_{PHL(DS2-DO)}$ $t_{PLH(DS2-DO)}$	High-to-low-level and low-to-high-level output propagation time, from input $\overline{DS1}$, DS2 and STB to output DO				50	ns
$t_{PHL(STB-\overline{INT})}$	High-to-low-level output propagation time, from input STB to output $\overline{INT}$				40	ns
$t_{PZL(MD-DO)}$ $t_{PZH(MD-DO)}$	Z-to-low-level and Z-to-high-level output propagation time, from inputs MD, $\overline{DS1}$ and DS2 to output DO	C_L = 30pF, R_{L1} = 1kΩ, R_{L2} = 1kΩ			70	ns
$t_{PHZ(MD-DO)}$ $t_{PLZ(MD-DO)}$	High-to-Z-level and low-to-Z-level output propagation time, from inputs MD, $\overline{DS1}$ and DS2 to output DO	C_L = 5pF, R_{L1} = 1kΩ, R_{L2} = 1kΩ			45	ns
$t_{PHL(\overline{CLR}-DO)}$	High-to-low-level output propagation time, from input $\overline{CLR}$ to output DO	C_L = 30pF, R_{L1} = 300Ω, R_{L2} = 600Ω			55	ns

Note 4 : Measurement circuit



TIMING DIAGRAMS REFERENCE LEVEL = 1.5V



MITSUBISHI ELECTRIC

4-BIT PARALLEL BIDIRECTIONAL BUS DRIVERS

DESCRIPTION

The M5L8216P and M5L8226P are 4-bit bidirectional bus drivers and suitable for the 8-bit parallel CPU M5L8080AP, S (8080A). They are fabricated by using bipolar Schottky TTL technology, and have high fan-out.

FEATURES

- Parallel 8-bit data bus buffer driver
- Low input current $\overline{DIEN}$, $\overline{CS}$: $I_{IL}=-500\mu A$(max)
  DI, DB: $I_{IL}=-250\mu A$(max)
- High output current M5L8216P
  DB: $I_{OL}=55mA$(max)
  $I_{OH}=-10mA$(max)
  DO: $I_{OH}=-1mA$(max)
  M5L8226P
  DB: $I_{OL}=50mA$(max)
  $I_{OH}=-10mA$(max)
  DO: $I_{OH}=-1mA$(max)
- Outputs can be connected with the CPU M5L8080AP, S: $V_{OH}=3.65V$(min)
- Three-state output
- The M5L8216P has interchangeability with Intel's 8216 in pin configuration and electrical characteristics, and the M5L8226P with Intel's 8226.

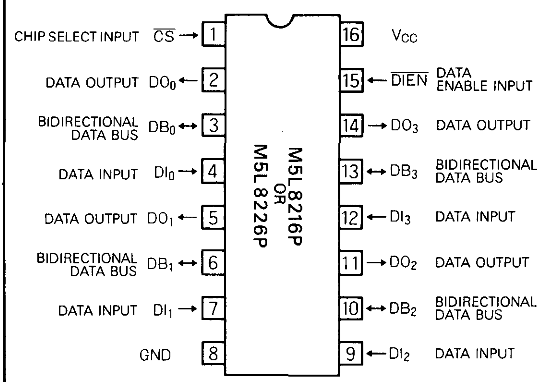
APPLICATION

Bidirectional bus driver/receiver for various types of microcomputer systems.

FUNCTION

The M5L8216P is a noninverting and the M5L8226P is an inverting 4-bit bidirectional bus driver.

PIN CONFIGURATION (TOP VIEW)



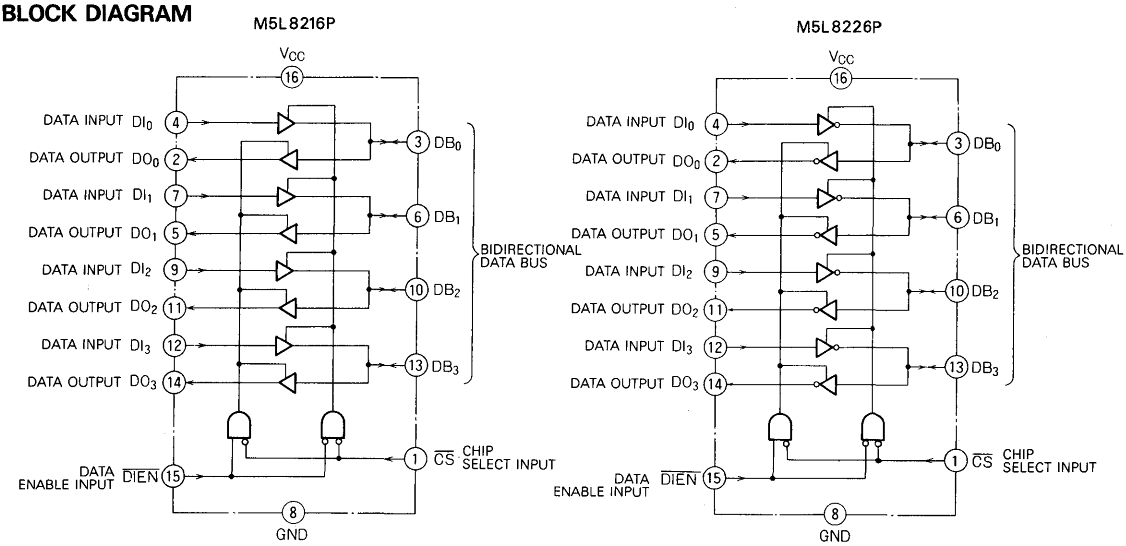
Outline 16P4

When the terminal $\overline{CS}$ is high-level, all outputs are in high-impedance state, and when low-level, the direction of the bidirectional bus can be controlled by the terminal $\overline{DIEN}$.

The terminal $\overline{DIEN}$ controls the data flow. The data flow control is performed by placing one of a pair of buffers in high-impedance state and allowing the other to transfer the data.

7

BLOCK DIAGRAM



ABSOLUTE MAXIMUM RATINGS (T_a=0~75℃, unless otherwise noted)

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	7	V
V_I	Input voltage, $\overline{CS}$, $\overline{DIEN}$, DI inputs		5.5	V
V_I	Input voltage, DB input		V_{CC}	V
V_O	High-level output voltage		V_{CC}	V
Pd	Power dissipation	Ta=25℃	700	mW
Topr	Operating free-air temperature range		0 ~75	℃
Tstg	Storage temperaturerange		−65~+150	℃

RECOMMENDED OPERATING CONDITIONS (T_a=0~75℃, unless otherwise noted)

Symbol	Parameter	Limits Min	Limits Nom	Limits Max	Unit
V_{CC}	Supply voltage	4.75	5	5.25	V
I_{OH}	High-level output current, DO output			−1	mA
I_{OH}	High-level output current, DB output			−10	mA
I_{OL}	Low-level output current, DO output			15	mA
I_{OL}	Low-level output current, DB output			25	mA

ELECTRICAL CHARACTERISTICS (T_a=0~75℃, unless otherwise noted)

Symbol	Parameter		Conditions		Limits Min	Limits Typ	Limits Max	Unit
V_{IH}	High-level input voltage				2			V
V_{IL}	Low-level input voltage						0.95	V
V_{IC}	Input clamp voltage		V_{CC}=4.75V, I_{IC}=−5mA				−1	V
V_{OH}	High-level output voltage, DO output		V_{CC}=4.75V V_{IH}=2V V_{IL}=0.95V	I_{OH}=−1mA	3.65			V
V_{OH}	High-level output voltage, DB output			I_{OH}=−10mA	2.4			V
V_{OL1}	Low-level output voltage, DO output			I_{OL}=15mA			0.5	V
V_{OL1}	Low-level output voltage, DB output			I_{OL}=25mA			0.5	V
V_{OL2}	Low-level output voltage, DB output	M5L 8216P		I_{OL}=55mA			0.7	V
		M5L 8226P		I_{OL}=50mA			0.7	V
I_{OZH}	Off-state output current, DO output		V_{CC}=5.25V	V_O=5.25V			20	μA
I_{OZH}	Off-state output current, DB output						100	μA
I_{OZL}	Off-state output current, DO output			V_O=0.5V			−20	μA
I_{OZL}	Off-state output current, DB output						−100	μA
I_{IH}	High-level input current, $\overline{DIEN}$, $\overline{CS}$ inputs		V_{CC}=5.25V, V_{IH}=4.5V				20	μA
I_{IH}	High-level input current, DI, DB inputs		V_{IL}=0V, V_I=5.25V				10	μA
I_{IL}	Low-level input current, $\overline{DIEN}$ $\overline{CS}$ intputs		V_{CC}=5.25V, V_{IH}=4.5V				−500	μA
I_{IL}	Low-level input current, DI, DB input		V_{IL}=0V, V_I=0.5V				−250	μA
I_{OS}	Short-circuit output DO output (Note 2)		V_{CC}=5.25V, V_O=0V		−15		−65	mA
I_{OS}	Short-circuit output, DB output (Note 2)				−30		−120	mA
I_{CC}	Supply current	M5L8216P	V_{CC}=5.25V				100	mA
		M5L8226P					100	mA
I_{CCZ}	Supply current Z	M5L8216P					120	mA
		M5L8226P					100	mA

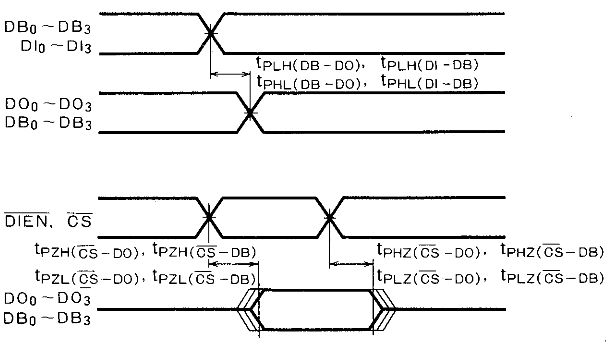
Note 1 : Current flowing into an IC is positive, out is negative.
2 : All measurements should be done quickly, and not more than one output should be shorted at a time.

4-BIT PARALLEL BIDIRECTIONAL BUS DRIVERS

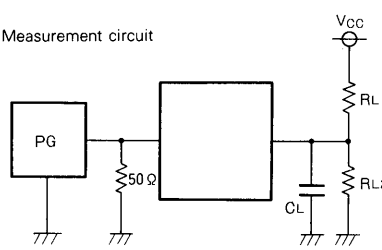
SWITCHING CHARACTERISTICS (V_{CC}=5V, Ta=25℃, unless otherwise noted)

Symbol	Parameter		Conditions (Note 3)	Limits Min	Limits Typ	Limits Max	Unit
$t_{PHL(DB-DO)}$, $t_{PLH(DB-DO)}$	High-to-low-and low-to-high output propagation time, from input DB to output DO		C_L=30pF, R_{L1}=300Ω, R_{L2}=600Ω			25	ns
$t_{PHL(DI-DB)}$, $t_{PLH(DI-DB)}$	High-to-low and low-to-high output propagation time, from input DI to output DB	M5L8216P	C_L=300pF, R_{L1}=90Ω, R_{L2}=180Ω			30	ns
		M5L8226P				25	ns
$t_{PHZ(\overline{CS}-DO)}$	High-to-Z and low-to-Z output propagation time, from inputs $\overline{DIEN}$, $\overline{CS}$, to output DO		C_L=5pF, R_{L1}=10kΩ, R_{L2}=1kΩ			35	ns
$t_{PLZ(\overline{CS}-DO)}$			C_L=5pF, R_{L1}=300Ω, R_{L2}=600Ω				
$t_{PZH(\overline{CS}-DO)}$	Output enable time, from inputs $\overline{DIEN}$, $\overline{CS}$ to output DO	M5L 8216P	C_L=30pF, R_{L1}=10kΩ, R_{L2}=1kΩ			65	ns
		M5L 8226P				54	ns
$t_{PZL(\overline{CS}-DO)}$		M5L 8216P	C_L=30pF, R_{L1}=300Ω, R_{L2}=600Ω			65	ns
		M5L 8226P				54	ns
$t_{PHZ(\overline{CS}-DB)}$	Output disable time, from inputs $\overline{DIEN}$, $\overline{CS}$, to output DB		C_L=5pF, R_{L1}=10kΩ, R_{L2}=1kΩ			35	ns
$t_{PLZ(\overline{CS}-DB)}$			C_L=5pF, R_{L1}=90Ω, R_{L2}=180Ω				
$t_{PZH(\overline{CS}-DB)}$	Output enable time, from inputs $\overline{DIEN}$, $\overline{CS}$ to output DB	M5L 8216P	C_L=300pF, R_{L1}=10kΩ, R_{L2}=1kΩ			65	ns
		M5L 8226P				54	ns
$t_{PZL(\overline{CS}-DB)}$		M5L 8216P	C_L=300pF, R_{L1}=90Ω, R_{L2}=180Ω			65	ns
		M5L 8226P				54	ns

TIMING DIAGRAM (Reference level = 1.5V)



Note 3 : Measurement circuit



APPLICATION EXAMPLES

Fig. 1 shows a pair of M5L8216PS or M5L8226PS which are directly connected with the M5L8080A CPU data bus, and their control signal. Fig. 2 shows an example circuit in which the M5L8216P or M5L8226P is used as an interface for memory and I/O to a bidirectional bus.

Fig. 1 Data bus buffer

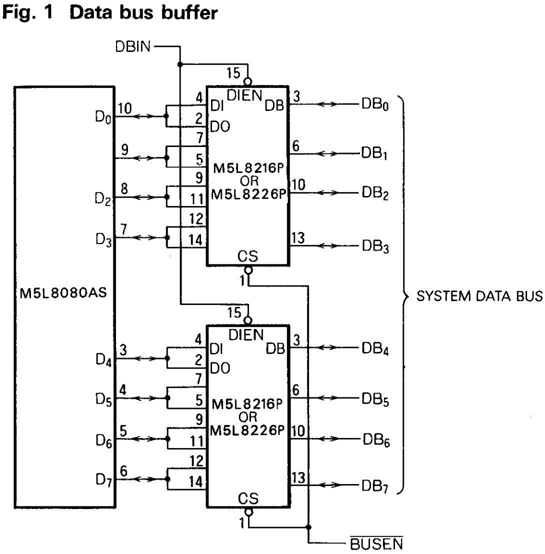
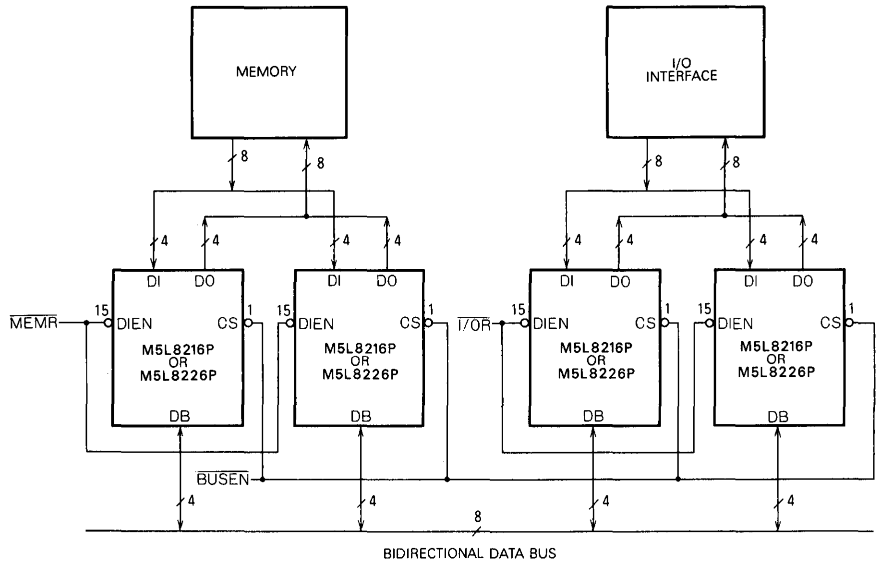


Fig. 2 Memory and I/O interface to bidirectional data bus



Precautions for Use

When the M5L8216P data input or two-way data bus is set to high to disable-output from the two-way bus or data output, care is required as a low glitch of approximate width 10nS will be generated.

2048-BIT STATIC RAM WITH I/O PORTS AND TIMER

DESCRIPTION

The M5L8155P is a 2K-bit RAM (256-word by 8-bit) fabricated with the N-channel silicon-gate ED-MOS technology. This IC has 3 I/O ports and a 14-bit counter/timer which make it a good choice to extend the functions of an 8-bit microcomputer. It is incased in a 40-pin plastic DIL package and operates with a single 5V power supply.

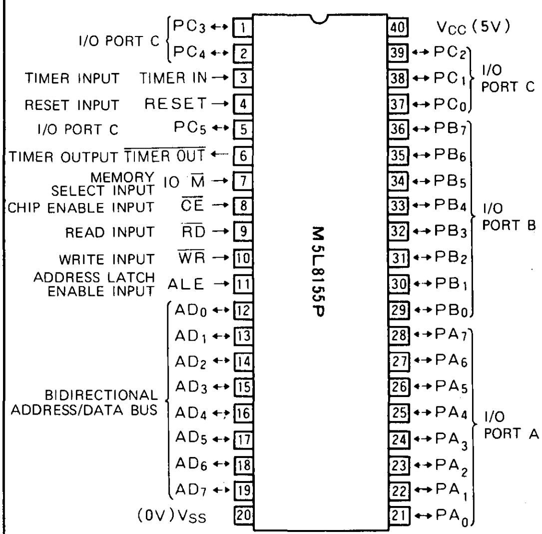
FEATURES

- Compatible with MELPS 85 devices
- Static RAM: 256 words by 8 bits
- Programmable 8-bit I/O port: 2
- Programmable 6-bit I/O port: 1
- Programmable counter/timer: 14 bits
- Multiplexed address/data bus
- Single 5V power supply
- Interchangeable with Intel's P8155 in pin
- Configuration and electrical characteristics

APPLICATION

- Extension of I/O ports and timer function for MELPS 8/85 and MELPS 8-48 devices

PIN CONFIGURATION (TOP VIEW)

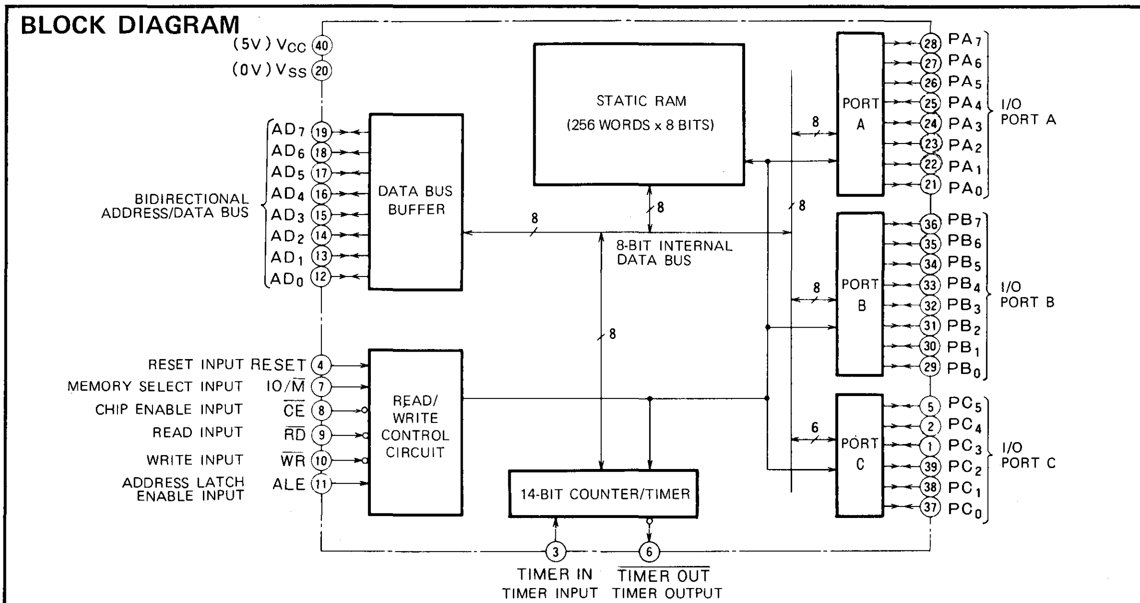


Outline 40P1

FUNCTION

The M5L8155P is composed of RAM, I/O ports and counter/timer. The RAM is a 2K-bit static RAM organized as 256 words by 8 bits. The I/O ports consist of 2 programmable 8-bit ports and 1 programmable 6-bit port. The terminals of the 6-bit port can be programmed to function as control terminals for the 8-bit ports, so that the 8-bit ports can be operated in a handshake mode. The counter/timer is composed of 14 bits that can be used to count down (events or time) and it can generate square wave pulses that can be used for counting and timing.

BLOCK DIAGRAM



OPERATION

Data Bus Buffer

This 3-state bidirectional 8-bit buffer is used to transfer the data while input or output instructions are being executed by the CPU. Command and address information is also transferred through the data bus buffer.

Read/Write Control Logic

The read/write control logic controls the transfer of data by interpreting I/O control bus output signals ($\overline{RD}$, $\overline{WR}$, IO/$\overline{M}$ and ALE) along with CPU signal ($\overline{CE}$). RESET signal is also used to control the transfer of data and commands.

Bidirectional Address/Data Bus ($AD_0 \sim AD_7$)

The bidirectional address/data bus is a 3-state 8-bit bus. The 8-bit address is latched in the internal latch by the falling edge of ALE. Then if IO/$\overline{M}$ input signal is at high-level, the address of I/O port, counter/timer, or command register is selected. If it is at low-level, memory address is selected.

The 8-bit address data is transferred by read input ($\overline{RD}$) or write input ($\overline{WR}$).

Chip Enable Input ($\overline{CE}$)

When $\overline{CE}$ is at low-level, the address information on address/data bus is stored in the M5L8155P

Read Input ($\overline{RD}$)

When $\overline{RD}$ is at low-level the data bus buffer is active. If IO/$\overline{M}$ input signal is at low-level, the contents of RAM are read through the address/data bus. If IO/$\overline{M}$ input is at high-level, the selected contents of I/O port or counter/timer are read through the address/data bus.

Write Input ($\overline{WR}$)

When $\overline{WR}$ is at low-level, the data on the address/data bus are written into RAM if IO/$\overline{M}$ is at low-level, or if IO/$\overline{M}$ is at high-level they are written into I/O port, counter/timer or command register.

Address Latch Enable Input (ALE)

An address on the address/data bus along with the levels of CE and IO/$\overline{M}$ are latched in the M5L8155P on the falling edge of ALE.

IO/Memory Input (IO/$\overline{M}$)

When IO/$\overline{M}$ is at low-level, the RAM is selected, while at high-level the I/O port, counter/timer or command register are selected.

I/O Port A ($PA_0 \sim PA_1$)

Port A is an 8-bit general-purpose I/O port. Input/output setting is controlled by the system software.

I/O Port B ($PB_0 \sim PB_7$)

Port B is an 8-bit general-purpose I/O port. Input/output setting is controlled by the system software.

I/O Port C ($PC_0 \sim PC_5$)

Port C is a 6-bit I/O port that can also be used to output control signals of port A (PA) or port B (PB). The functions of port C are controlled by the system software. When port C is used to output control signals of ports A or B the assignment of the signals to the pins is as shown in Table 1.

Table 1 Pin assignment of control signals of port C

Pin	Function
PC_5	$\overline{B\ STB}$ (port B strobe)
PC_4	B BF (port B buffer full)
PC_3	B INTR (port B interrupt)
PC_2	$\overline{A\ STB}$ (port A strobe)
PC_1	A BF (port A buffer full)
PC_0	A INTR (port A interrupt)

Timer Input (TIMER IN)

The signal at this input terminal is used by the counter/timer for counting events or time. (3MHz max.)

Timer Output ($\overline{TIMER\ OUT}$)

A square wave signal or pulse from the counter/timer is output through this pin when in the operation mode.

Command Register (8 bits)

The command register is an 8-bit latched register. The low-order 4 bits (bits 0 ~ 3) are used for controlling and determination of the mode of the ports. Bits 4 and 5 are used as interrupt enable flags for ports A and B when port C is used as a control port. Bits 6 and 7 are used for controlling the counter/timer. The contents of the command register are rewritten by output instructions (address I/O XXXXX000).

Details of the functions of the individual bits of the command register are shown in Fig. 1.

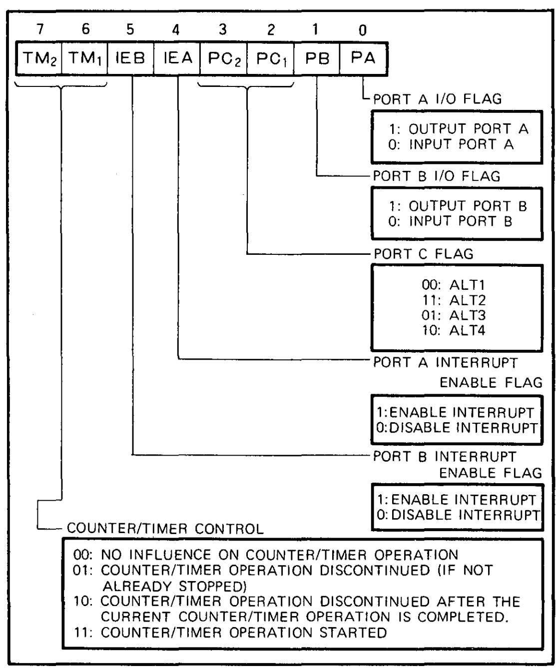


Fig. 1 Bit functions of the command register

Status Register (7 bits)

The status register is a 7-bit latched register. The low-order 5 bits (bits 0 ~ 4) are used as status flags for the I/O ports. Bit 6 is as a status flag for the counter/timer. The contents of the status register are transferred into the CPU by reading (INPUT instruction, address I/O XXXXX000). Details of the functions of the individual bits of the status register are shown in Fig. 2.

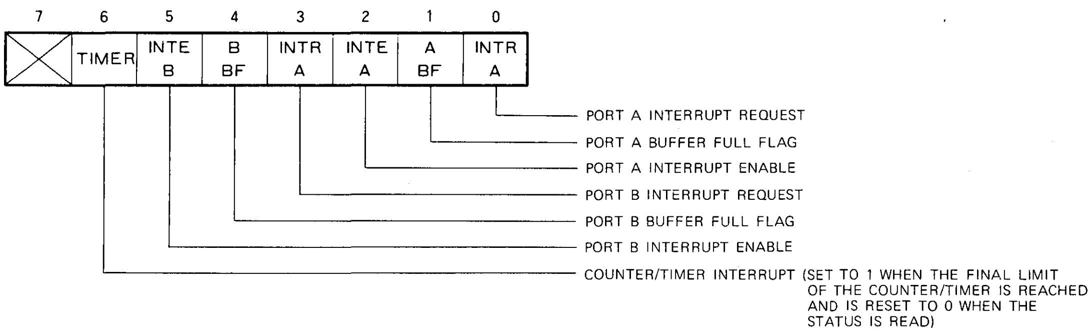


Fig. 2 Bit functions of the status register

I/O Ports

Command/status registers (8 bits/7 bits)

These registers are assigned address XXXXX000. When executing an OUTPUT instruction, the contents of the command register are rewritten. When executing an INPUT instruction the contents of the status register are read.

Port A Register (8 bits)

Port A register is assigned address XXXXX001. This register can be programmed as an input or output by setting the appropriate bits of the command register as shown in Fig. 1.

Port A can be operated in basic or strobe mode and is assigned I/O terminal $PA_0 \sim PA_7$.

Port B Register (8 bits)

Port B register is assigned address XXXXX010. As with Port A register, this register can be programmed as an input or output by setting the appropriate bits of the command register as shown in Fig. 1. Port B can be operated in basic or strobe mode and is assigned I/O terminals $PB_0 \sim PB_7$.

Port C Register (6 bits)

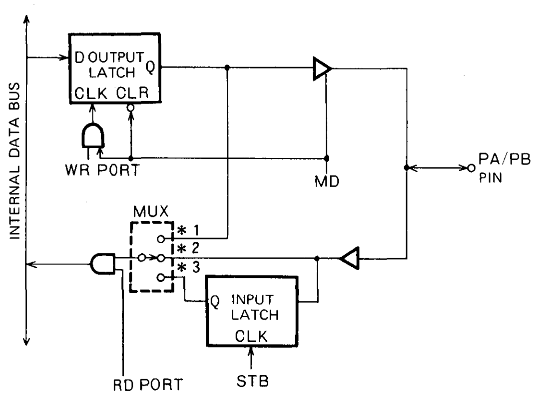
Port C register is assigned address XXXXX011. This port is used for controlling input/output operations of ports A and B by selectively setting bits 2 and 3 of the command register as shown in Fig. 1. Details of the functions of the various setting of bits 2 and 3 are shown in Table 2. Port C is assigned I/O terminals $PC_0 \sim PC_5$ and when used as port control signals, the 3 low-order bits are assigned for port A while the 3 high-order bits are assigned for port B.

Table 2 Functions of port C

State / Terminal	ALT1	ALT2	ALT 3	ALT 4
PC_5	Input	Output	Output	$\overline{\text{B STB}}$ (port B strobe)
PC_4	Input	Output	Output	B BF (port B buffer full)
PC_3	Input	Output	Output	B INTR (port B interrupt)
PC_2	Input	Output	$\overline{\text{A STB}}$ (port A strobe)	$\overline{\text{A STB}}$ (port A strobe)
PC_1	Input	Output	A BF (port A buffer full)	A BF (port A buffer full)
PC_0	Input	Output	A INTR (port A interrupt)	A INTR (port A interrupt)

Configuration of Ports

A block diagram of 1 bit of ports A and B is shown in Fig. 3. While port A or B is programmed as an output port, if the port is addressed by an input instruction, the contents of the selected port can be read. When a port is put in input mode, the output latch is cleared and writing into the output latch is disabled. Therefore when a port is changed to output mode from input mode, low-level signals are output through the port. When a reset signal is applied, all 3 ports (PA, PB, and PC) will be input ports and their output latches are cleared. Port C has the same configuration as ports A and B in modes ALT1 and ALT2.



1. *1 OUTPUT MODE } MULTIPLEX CONTROL
   *2 INPUT MODE
   *3 STROBE INPUT MODE
2. MD= 1 : OUTPUT MODE
   0 : INPUT MODE
3. RD PORT=IO/$\overline{M}$·RD·CE·
   (PORT ADDRESS SELECTED)
4. WR PORT=IO/$\overline{M}$·WR·CE·
   (PORT ADDRESS SELECTED)

Fig. 3 Configuration for 1 bit of port A or B

The basic functions of the I/O ports are shown in Table 3. The control signal levels to ports A and B, when port C is programmed as a control port, are shown in Table 4.

Table 3 Basic functions of I/O ports

Address	$\overline{RD}$	$\overline{WR}$	Function
XXXXX000	0	1	AD bus ← status register
	1	0	Command register ← AD bus
XXXXX001	0	1	AD bus ← port A
	1	0	Port A ← AD bus
XXXXX010	0	1	AD bus ← port B
	1	0	Port B ← AD bus
XXXXX011	0	1	AD bus ← port C
	1	0	Port C ← AD bus

Table 4 Port control signal levels at ALT3 and ALT4

Control signal	Input mode	Output mode
BF	"L"	"L"
INTR	"L"	"H"
$\overline{STB}$	Input	Input

Counter/Timer

The counter/timer is a 14-bit counting register plus 2 mode flags. The register has two sections: address I/O XXXXX100 is assigned to the low-order 8 bits and address I/O XXXXX101 is assigned to the high-order 8 bits. The low-order bits 0 ~ 13 are used for counting or timing. The counter is initialized by the program and then counted down to zero. The initial setting can range from 2_{16} to $3FFF_{16}$. Bits 14 and 15 are used as mode flags.

The mode flags select 1 of 4 modes with functions as follow:

Mode 0: Outputs high-level signal during the former half of the counter operation
Outputs low-level signal during the latter half of the counter operation

Mode 1: Outputs square wave signals as in mode 0

Mode 2: Outputs a low-level pulse during the final count down

Mode 3: Outputs a low-level pulse during each final count down

Starting and stopping the counter/timer is controlled by bits 6 and 7 of the command register (see Fig. 1 for details). The format and timer modes of the counter/timer register are shown in Fig. 4 and Table 5.

M_2	M_1	T_{13}	T_{12}	T_{11}	T_{10}	T_9	T_8
TIMER MODE		THE HIGH-ORDER 6 BITS OF THE COUNTER REGISTER					

T_7	T_6	T_5	T_4	T_3	T_2	T_1	T_0
THE LOW-ORDER 8 BITS OF THE COUNTER REGISTER							

Fig. 4 Format of counter/timer

Table 5 Timer mode

M_2	M_1	Timer operation
0	0	Outputs high-level signal during the former half of the counter operation Outputs low-level signal during the latter half of the counter operation (mode 0)
0	1	Outputs square wave signals as in mode 0 (mode 1)
1	0	Outputs a low-level pulse during the final count down (mode 2)
1	1	Outputs a low-level pulse during each final count down (mode 3)

The counter/timer is not influenced by a reset, but counting is discontinued. To resume counting, a start command must be written into the command register as shown in Fig. 1. While operating 2n+1 count down in mode 0, a high-level signal is output during the n+1 counting and a low-level signal is output during the n counting.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.5~7	V
V_I	Input voltage		−0.5~7	V
V_O	Output voltage		−0.5~7	V
Pd	Maximum power dissipation	Ta=25°C	1.5	W
Topr	Operating free-air temperature range		0~70	°C
Tstg	Storage temperature range		−65~150	°C

RECOMMENDED OPERATING CONDITIONS (Ta=0~70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{SS}	Power-supply voltage		0		V
V_{IL}	Low-level input voltage	−0.5		0.8	V
V_{IH}	High-level input voltage	2		V_{CC}+0.5	V

ELECTRICAL CHARACTERISTICS (Ta=0~70°C, V_{CC}=5V+5%, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	V_{SS}=0V, I_{OH}=−400μA	2.4			V
V_{OL}	Low-level output voltage	V_{SS}=0V, I_{OL}=2mA			0.45	V
I_I	Input leak current	V_{SS}=0V, V_I=0~V_{CC}	−10		10	μA
$I_{I(CE)}$	Input leak current, $\overline{CE}$ pin	V_{SS}=0V, V_I=0~V_{CC}	−100		100	μA
I_{OZ}	Output floating leak current	V_{SS}=0V, V_I=0.45~V_{CC}	−10		10	μA
Ci	Input capacitance	V_{IL}=0V, f=1MHz, 25mVrms, Ta=25°C			10	pF
Ci/o	Input/output terminal capacitance	$V_{I/OL}$=0V, f=1MHz, 25mVrms, Ta=25°C			20	pF
I_{CC}	Supply current from V_{CC}	V_{SS}=0V			180	mA

Note 5 Current flowing into an IC is positive, out is negative.

TIMING REQUIREMENTS (Ta=0~70°C, V_{CC}=5V±5%, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{su (A-L)}$	Address setup time before latch	t_{AL}		50			ns
$t_{h (L-A)}$	Address hold time after latch	t_{LA}		80			ns
$t_{h (L-RWH)}$	Read/write hold time after latch	t_{LC}		100			ns
$t_{w(L)}$	Latch pulse width	t_{LL}		100			ns
$t_{h (RW-L)}$	Latch hold time after read/write	t_{CL}		20			ns
$t_{w (RWL)}$	Read/write low-level pulse width	t_{CC}		250			ns
$t_{su (D-W)}$	Data setup time before write	t_{DW}		150			ns
$t_{h (W-D)}$	Data hold time after write	t_{WD}		0			ns
$t_{w (RWH)}$	Read/write high-level pulse width	t_{RV}		300			ns
$t_{su (P-R)}$	Port setup time before read	t_{PR}		70			ns
$t_{h (R-P)}$	Port hold time after read	t_{RP}		50			ns
$t_{w (STB)}$	Strobe pulse width	t_{SS}		200			ns
$t_{su (P-STB)}$	Port setup time before strobe	t_{PSS}		50			ns
$t_{h (STB-P)}$	Port hold time after strobe	t_{PHS}		120			ns
$t_{w(\phi H)}$	Timer input high-level pulse width	t_2		120			ns
$t_{w(\phi L)}$	Timer input low-level pulse width	t_1		80			ns
$t_{c (\phi)}$	Timer input cycle time	t_{CYC}		320			ns
$t_{r (\phi)}$	Timer input rise time	t_r				30	ns
$t_{f (\phi)}$	Timer input fall time	t_f				30	ns

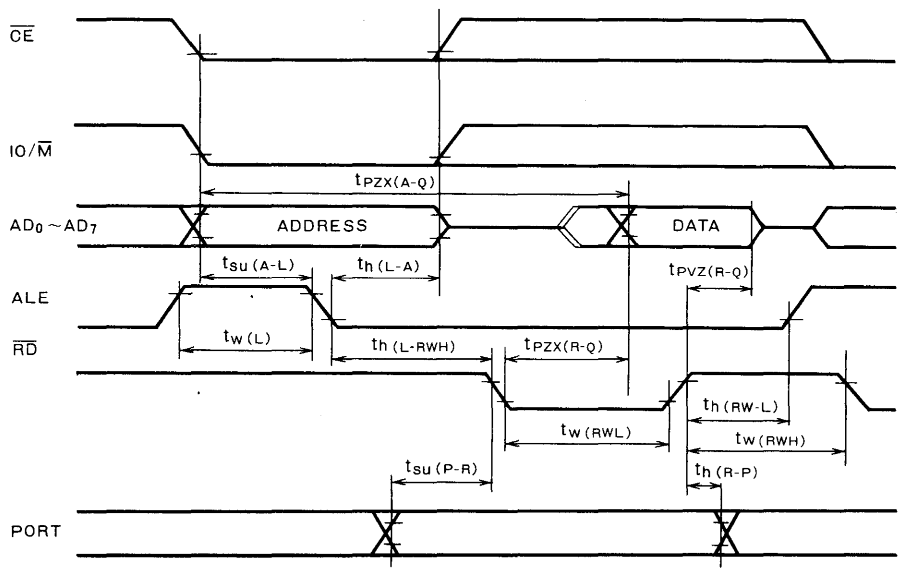
SWITCHING CHARACTERISTICS (Ta=0~70°C, V_{CC}=5V±5%, unless otherwise noted.)

Symbol	Parameter	Alternative symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{PZX(R-Q)}$	Propagation time from read to data output	t_{RD}				170	ns
$t_{PZX(A-Q)}$	Propagation time from address to data output	t_{AD}				400	ns
$t_{PVZ(R-Q)}$	Propagation time from read to data floating (Note 7)	t_{RDF}				100	ns
$t_{PHL(W-P)}$ $t_{PLH(W-P)}$	Propagation time from write to data output	t_{WP} t_{WP}				400	ns
$t_{PLH(STB-BF)}$	Propagation time from strobe to BF flag	t_{SBF}				400	ns
$t_{PHL(R-BF)}$	Propagation time from read to BF flag	t_{RBE}				400	ns
$t_{PLH(STB-INTR)}$	Propagation time from strobe to interrupt	t_{SI}				400	ns
$t_{PHL(R-INTR)}$	Propagation time from read to interrupt	t_{RDI}				400	ns
$t_{PHL(STB-BF)}$	Propagation time from strobe to BF flag	t_{SBE}				400	ns
$t_{PLH(W-BF)}$	Propagation time from write to BF flag	t_{WBF}				400	ns
$t_{PHL(W-INTR)}$	Propagation time from write to interrupt	t_{WI}				400	ns
$t_{PHL(\phi-OUT)}$ $t_{PLH(\phi-OUT)}$	Propagation time from timer input to timer output	t_{TL} t_{TH}				400	ns

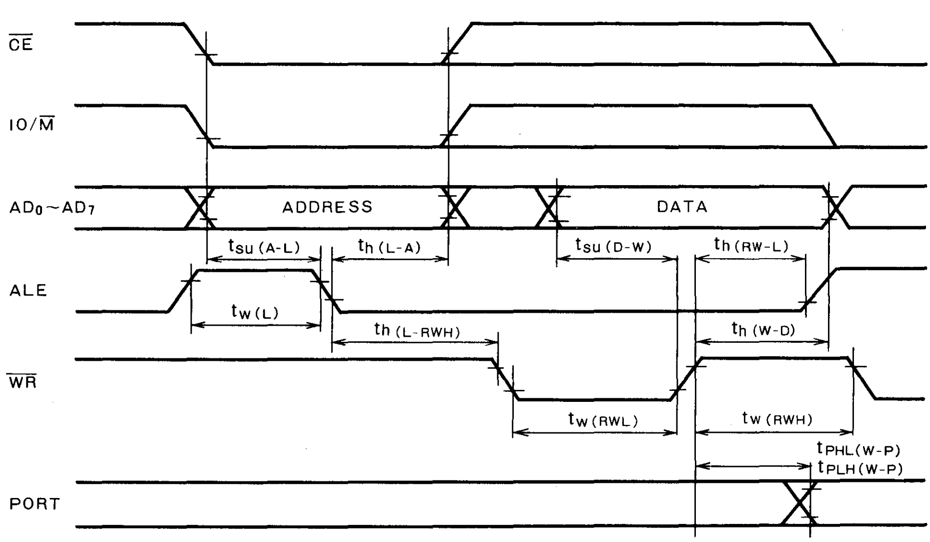
Note 6: Measurement conditions C = 150pF
7: Measurement conditions of note 6 are not applied.

TIMING DIAGRAM (reference level, high-level=2V, low-level=0.8V)

Basic Input

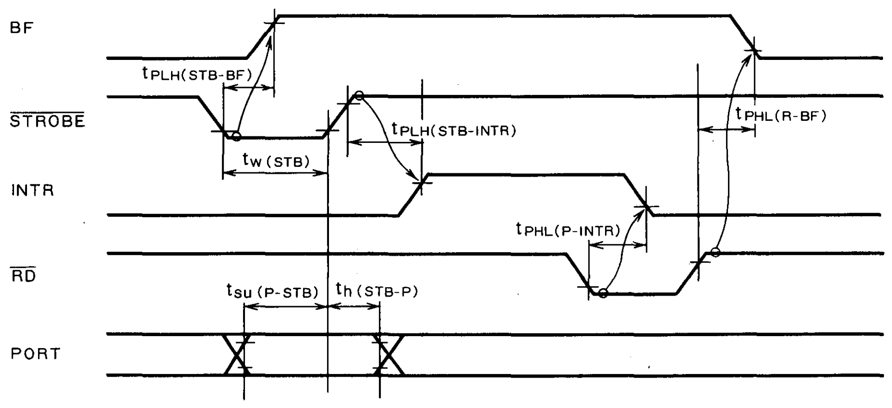


Basic Output

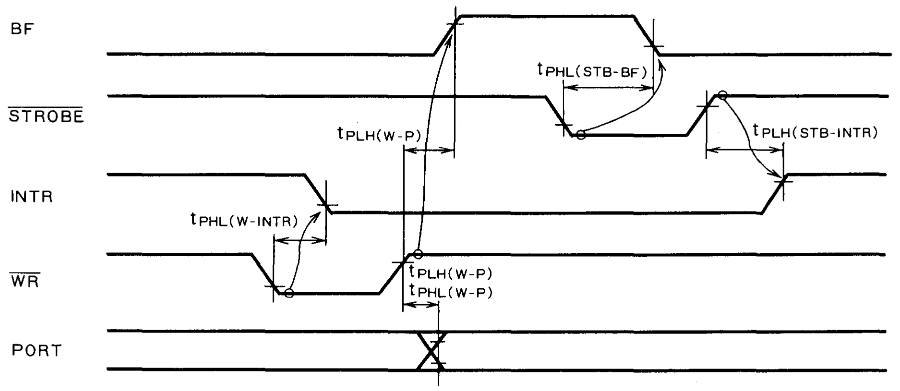


7

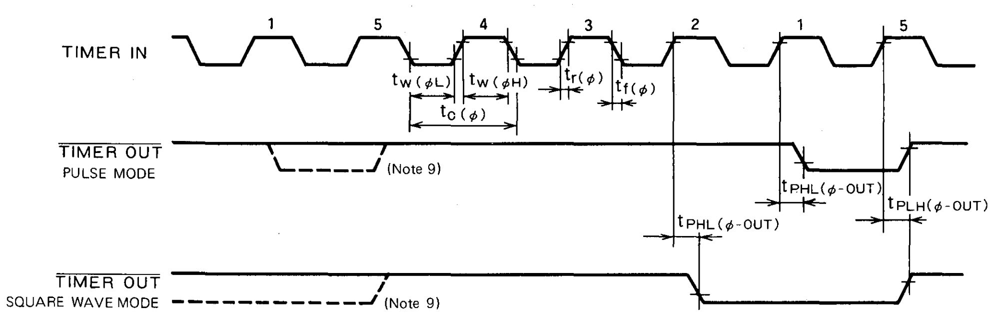
Strobed Input



Strobed Output



Timer (Note 8)



Note 8: The wave form is shown counting down from 5 to 1.
9: As long as the M1 mode flag of the timer register is at high-level, pulses are continuously output.

2048-BIT STATIC RAM WITH I/O PORTS AND TIMER

DESCRIPTION

The M5L8156P is a 2K-bit RAM (256-word by 8-bit) fabricated with the N-channel silicon-gate ED-MOS technology. This IC has 3 I/O ports and a 14-bit counter/timer which make it a good choice to extend the functions of an 8-bit microcomputer. It is incased in a 40-pin plastic DIL package and operates with a single 5V power supply.

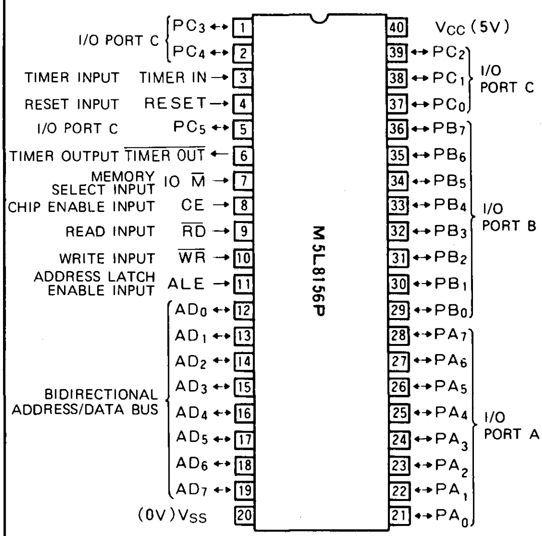
FEATURES

- Compatible with MELPS 85 devices
- Static RAM: 256 words by 8 bits
- Programmable 8-bit I/O port: 2
- Programmable 6-bit I/O port: 1
- Programmable counter/timer: 14 bits
- Multiplexed address/data bus
- Single 5V power supply
- Interchangeable with Intel's P8156 in pin
- Configuration and electrical characteristics

APPLICATION

- Extension of I/O ports and timer function for MELPS 8/85 and MELPS 8-48 devices

PIN CONFIGURATION (TOP VIEW)

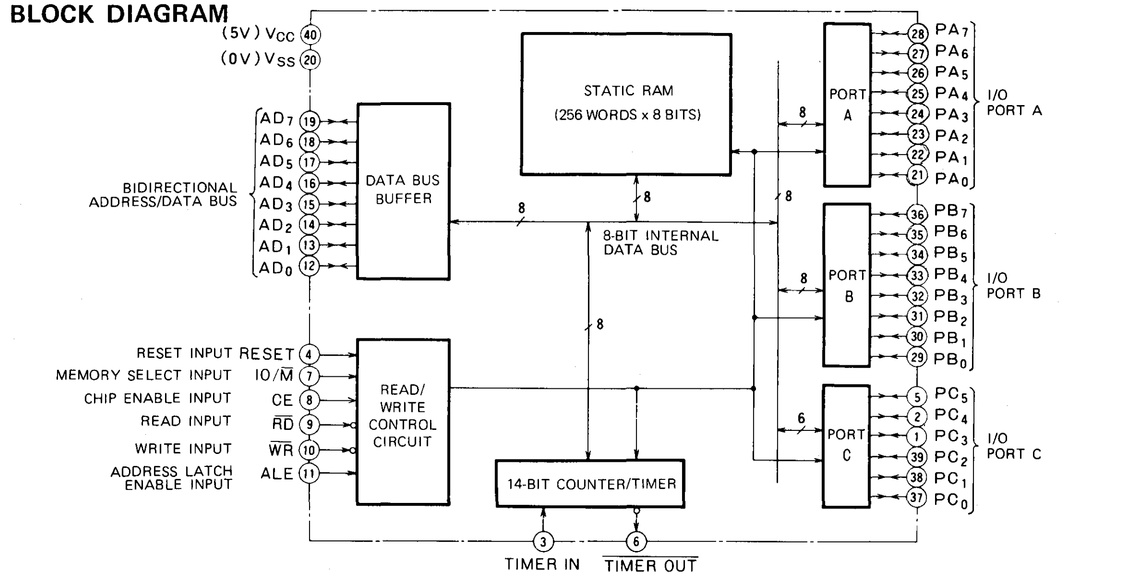


Outline 40P1

FUNCTION

The M5L8156P is composed of RAM, I/O ports and counter/timer. The RAM is a 2K-bit static RAM organized as 256 words by 8 bits. The I/O ports consist of 2 programmable 8-bit ports and 1 programmable 6-bit port. The terminals of the 6-bit port can be programmed to function as control terminals for the 8-bit ports, so that the 8-bit ports can be operated in a handshake mode. The counter/timer is composed of 14 bits that can be used to count down (events or time) and it can generate square wave pulses that can be used for counting and timing.

BLOCK DIAGRAM



OPERATION

Data Bus Buffer

This 3-state bidirectional 8-bit buffer is used to transfer the data while input or output instructions are being executed by the CPU. Command and address information is also transferred through the data bus buffer.

Read/Write Control Logic

The read/write control logic controls the transfer of data by interpreting I/O control bus output signals ($\overline{RD}$, $\overline{WR}$, IO/$\overline{M}$ and ALE) along with CPU signal (CE). RESET signal is also used to control the transfer of data and commands.

Bidirectional Address/Data Bus ($AD_0 \sim AD_7$)

The bidirectional address/data bus is a 3-state 8-bit bus. The 8-bit address is latched in the internal latch by the falling edge of ALE. Then if IO/$\overline{M}$ input signal is at high-level, the address of I/O port, counter/timer, or command register is selected. If it is at low-level, memory address is selected.

The 8-bit address data is transferred by read input ($\overline{RD}$) or write input ($\overline{WR}$).

Chip Enable Input (CE)

When CE is at high-level, the address information on address/data bus is stored in the M5L8156P.

Read Input ($\overline{RD}$)

When $\overline{RD}$ is at low-level the data bus buffer is active. If IO/$\overline{M}$ input signal is at low-level, the contents of RAM are read through the address/data bus. If IO/$\overline{M}$ input is at high-level, the selected contents of I/O port or counter/timer are read through the address/data bus.

Write Input ($\overline{WR}$)

When $\overline{WR}$ is at low-level, the data on the address/data bus are written into RAM if IO/$\overline{M}$ is at low-level, or if IO/$\overline{M}$ is at high-level they are written into I/O port, counter/timer or command register.

Address Latch Enable Input (ALE)

An address on the address/data bus along with the levels of CE and IO/$\overline{M}$ are latched in the M5L8156P on the falling edge of ALE.

IO/Memory Input (IO/$\overline{M}$)

When IO/$\overline{M}$ is at low-level, the RAM is selected, while at high-level the I/O port, counter/timer or command register are selected.

I/O Port A ($PA_0 \sim PA_1$)

Port A is an 8-bit general-purpose I/O port. Input/output setting is controlled by the system software.

I/O Port B ($PB_0 \sim PB_7$)

Port B is an 8-bit general-purpose I/O port. Input/output setting is controlled by the system software.

I/O Port C ($PC_0 \sim PC_5$)

Port C is a 6-bit I/O port that can also be used to output control signals of port A (PA) or port B (PB). The functions of port C are controlled by the system software. When port C is used to output control signals of ports A or B the assignment of the signals to the pins is as shown in Table 1.

Table 1 Pin assignment of control signals of port C

Pin	Function	
PC_5	$\overline{B\ STB}$	(port B strobe)
PC_4	B BF	(port B buffer full)
PC_3	B INTR	(port B interrupt)
PC_2	$\overline{A\ STB}$	(port A strobe)
PC_1	A BF	(port A buffer full)
PC_0	A INTR	(port A interrupt)

Timer Input (TIMER IN)

The signal at this input terminal is used by the counter/timer for counting events or time. (3MHz max.)

Timer Output ($\overline{TIMER\ OUT}$)

A square wave signal or pulse from the counter/timer is output through this pin when in the operation mode.

Command Register (8 bits)

The command register is an 8-bit latched register. The low-order 4 bits (bits $0 \sim 3$) are used for controlling and determination of the mode of the ports. Bits 4 and 5 are used as interrupt enable flags for ports A and B when port C is used as a control port. Bits 6 and 7 are used for controlling the counter/timer. The contents of the command register are rewritten by output instructions (address I/O XXXXX000).

Details of the functions of the individual bits of the command register are shown in Fig. 1.

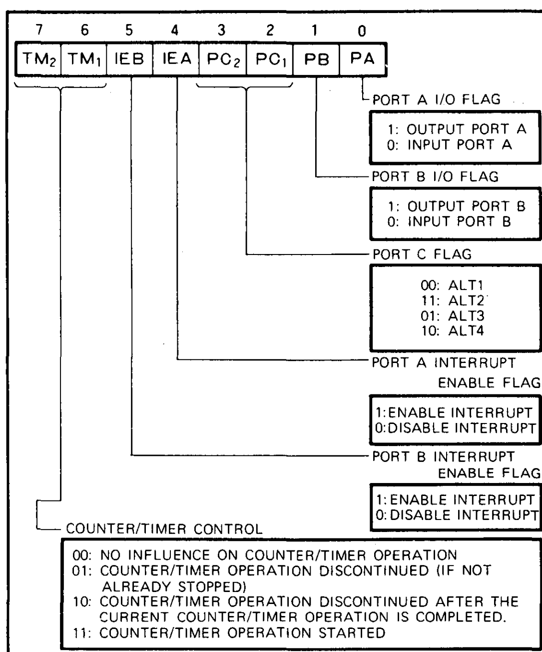


Fig. 1 Bit functions of the command register

Status Register (7 bits)
The status register is a 7-bit latched register. The low-order 5 bits (bits 0 ~ 4) are used as status flags for the I/O ports. Bit 6 is as a status flag for the counter/timer. The contents of the status register are transferred into the CPU by reading (INPUT instruction, address I/O XXXXX000). Details of the functions of the individual bits of the status register are shown in Fig. 2.

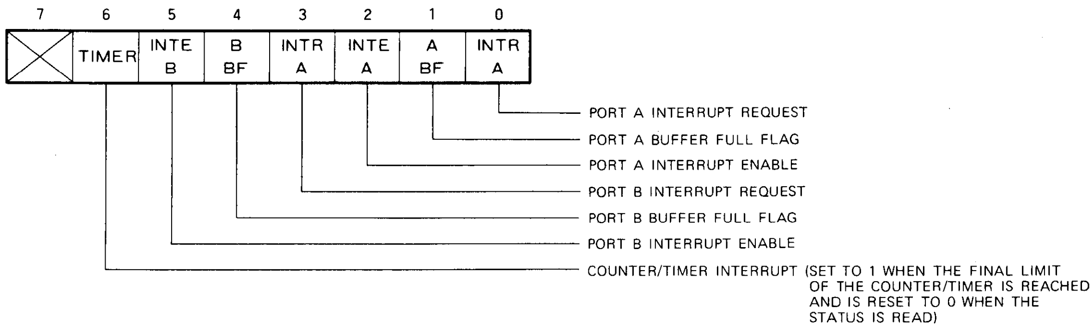


Fig. 2 Bit functions of the status register

I/O Ports

Command/status registers (8 bits/7 bits)
These registers are assigned address XXXXX000. When executing an OUTPUT instruction, the contents of the command register are rewritten. When executing an INPUT instruction the contents of the status register are read.

Port A Register (8 bits)
Port A register is assigned address XXXXX001. This register can be programmed as an input or output by setting the appropriate bits of the command register as shown in Fig. 1.

Port A can be operated in basic or strobe made and is assigned I/O terminal $PA_0 \sim PA_7$.

Port B Register (8 bits)
Port B register is assigned address XXXXX010. As with Port A register, this register can be programmed as an input or output by setting the appropriate bits of the command register as shown in Fig. 1. Port B can be operated in basic or strobe mode and is assigned I/O terminals $PB_0 \sim PB_7$.

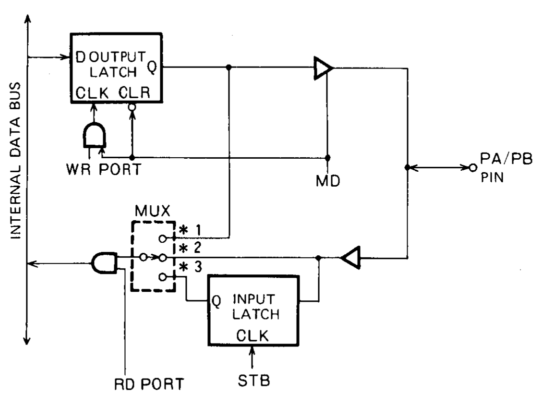
Port C Register (6 bits)
Port C register is assigned address XXXXX011. This port is used for controlling input/output operations of ports A and B by selectively setting bits 2 and 3 of the command register as shown in Fig. 1. Details of the functions of the various setting of bits 2 and 3 are shown in Table 2. Port C is assigned I/O terminals $PC_0 \sim PC_5$ and when used as port control signals, the 3 low-order bits are assigned for port A while the 3 high-order bits are assigned for port B.

Table 2 Functions of port C

State / Terminal	ALT 1	ALT 2	ALT 3	ALT 4
PC_5	Input	Output	Output	$\overline{\text{B STB}}$ (port B strobe)
PC_4	Input	Output	Output	B BF (port B buffer full)
PC_3	Input	Output	Output	B INTR (port B interrupt)
PC_2	Input	Output	$\overline{\text{A STB}}$ (port A strobe)	$\overline{\text{A STB}}$ (port A strobe)
PC_1	Input	Output	A BF (port A buffer full)	A BF (port A buffer full)
PC_0	Input	Output	A INTR (port A interrupt)	A INTR (port A interrupt)

Configuration of Ports

A block diagram of 1 bit of ports A and B is shown in Fig. 3. While port A or B is programmed as an output port, if the port is addressed by an input instruction, the contents of the selected port can be read. When a port is put in input mode, the output latch is cleared and writing into the output latch is disabled. Therefore when a port is changed to output mode from input mode, low-level signals are output through the port. When a reset signal is applied, all 3 ports (PA, PB, and PC) will be input ports and their output latches are cleared. Port C has the same configuration as ports A and B in modes ALT1 and ALT2.



1. *1 OUTPUT MODE, *2 INPUT MODE, *3 STROBE INPUT MODE } MULTIPLEX CONTROL
2. MD = 1 : OUTPUT MODE; 0 : INPUT MODE
3. RD PORT = IO/$\overline{M}$·RD·CE·(PORT ADDRESS SELECTED)
4. WR PORT = IO/$\overline{M}$·WR·CE·(PORT ADDRESS SELECTED)

Fig. 3 Configuration for 1 bit of port A or B

The basic functions of the I/O ports are shown in Table 3. The control signal levels to ports A and B, when port C is programmed as a control port, are shown in Table 4.

Table 3 Basic functions of I/O ports

Address	$\overline{RD}$	$\overline{WR}$	Function
XXXXX000	0	1	AD bus ← status register
	1	0	Command register ← AD bus
XXXXX001	0	1	AD bus ← port A
	1	0	Port A ← AD bus
XXXXX010	0	1	AD bus ← port B
	1	0	Port B ← AD bus
XXXXX011	0	1	AD bus ← port C
	1	0	Port C ← AD bus

Table 4 Port control signal levels at ALT3 and ALT4

Control signal	Input mode	Output mode
BF	"L"	"L"
INTR	"L"	"H"
$\overline{STB}$	Input	Input

Counter/Timer

The counter/timer is a 14-bit counting register plus 2 mode flags. The register has two sections: address I/O XXXXX100 is assigned to the low-order 8 bits and address I/O XXXXX101 is assigned to the high-order 8 bits. The low-order bits 0 ~ 13 are used for counting or timing. The counter is initialized by the program and then counted down to zero. The initial setting can range from 2_{16} to $3FF_{16}$. Bits 14 and 15 are used as mode flags.

The mode flags select 1 of 4 modes with functions as follow:

Mode 0: Outputs high-level signal during the former half of the counter operation
Outputs low-level signal during the latter half of the counter operation

Mode 1: Outputs square wave signals as in mode 0

Mode 2: Outputs a low-level pulse during the final count down

Mode 3: Outputs a low-level pulse during each final count down

Starting and stopping the counter/timer is controlled by bits 6 and 7 of the command register (see Fig. 1 for details). The format and timer modes of the counter/timer register are shown in Fig. 4 and Table 5.

M_2	M_1	T_{13}	T_{12}	T_{11}	T_{10}	T_9	T_8
TIMER MODE		THE HIGH-ORDER 6 BITS OF THE COUNTER REGISTER					

T_7	T_6	T_5	T_4	T_3	T_2	T_1	T_0
THE LOW-ORDER 8 BITS OF THE COUNTER REGISTER							

Fig. 4 Format of counter/timer

The counter/timer is not influenced by a reset, but counting is discontinued. To resume counting, a start command must be written into the command register as shown in Fig. 1. While operating 2n+1 count down in mode 0, a high-level signal is output during the n+1 counting and a low-level signal is output during the n counting.

Table 5 Timer mode

M_2	M_1	Timer operation
0	0	Outputs high-level signal during the former half of the counter operation Outputs low-level signal during the latter half of the counter operation (mode 0)
0	1	Outputs square wave signals as in mode 0 (mode 1)
1	0	Outputs a low-level pulse during the final count down (mode 2)
1	1	Outputs a low-level pulse during each final count down (mode 3)

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.5 ~ 7	V
V_I	Input voltage		−0.5 ~ 7	V
V_O	Output voltage		−0.5 ~ 7	V
Pd	Maximum power dissipation	Ta = 25 °C	1.5	W
Topr	Operating free-air temperature range		0 ~ 70	°C
Tstg	Storage temperature range		−65 ~ 150	°C

RECOMMENDED OPERATING CONDITIONS (Ta = 0 ~ 70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{SS}	Power-supply voltage		0		V
V_{IL}	Low-level input voltage	−0.5		0.8	V
V_{IH}	High-level input voltage	2		V_{CC}+0.5	V

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 70°C, V_{CC} = 5V ± 5%, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	V_{SS} = 0V, I_{OH} = −400μA	2.4			V
V_{OL}	Low-level output voltage	V_{SS} = 0V, I_{OL} = 2mA			0.45	V
I_I	Input leak current	V_{SS} = 0V, V_I = 0 ~ V_{CC}	−10		10	μA
$I_{I(CE)}$	Input leak current, CE pin	V_{SS} = 0V, V_I = 0 ~ V_{CC}	−100		100	μA
I_{OZ}	Output floating leak current	V_{SS} = 0V, V_I = 0.45 ~ V_{CC}	−10		10	μA
Ci	Input capacitance	V_{IL} = 0V, f = 1MHz, 25mVrms, Ta = 25 °C			10	pF
Ci/o	Input/output terminal capacitance	$V_{I/OL}$ = 0V, f = 1MHz, 25mVrms, Ta = 25°C			20	pF
I_{CC}	Supply current from V_{CC}	V_{SS} = 0V			180	mA

Note 5 Current flowing into an IC is positive, out is negative.

TIMING REQUIREMENTS (Ta = 0 ~ 70°C, V_{CC} = 5V + 5%, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{su(A-L)}$	Address setup time before latch	t_{AL}		50			ns
$t_{h(L-A)}$	Address hold time after latch	t_{LA}		80			ns
$t_{h(L-RWH)}$	Read/write hold time after latch	t_{LC}		100			ns
$t_{w(L)}$	Latch pulse width	t_{LL}		100			ns
$t_{h(RW-L)}$	Latch hold time after read/write	t_{CL}		20			ns
$t_{w(RWL)}$	Read/write low-level pulse width	t_{CC}		250			ns
$t_{su(D-W)}$	Data setup time before write	t_{DW}		150			ns
$t_{h(W-D)}$	Data hold time after write	t_{WD}		0			ns
$t_{w(RWH)}$	Read/write high-level pulse width	t_{RV}		300			ns
$t_{su(P-R)}$	Port setup time before read	t_{PR}		70			ns
$t_{h(R-P)}$	Port hold time after read	t_{RP}		50			ns
$t_{w(STB)}$	Strobe pulse width	t_{SS}		200			ns
$t_{su(P-STB)}$	Port setup time before strobe	t_{PSS}		50			ns
$t_{h(STB-P)}$	Port hold time after strobe	t_{PHS}		120			ns
$t_{w(\phi H)}$	Timer input high-level pulse width	t_2		120			ns
$t_{w(\phi L)}$	Timer input low-level pulse width	t_1		80			ns
$t_{c(\phi)}$	Timer input cycle time	t_{CYC}		320			ns
$t_{r(\phi)}$	Timer input rise time	t_r				30	ns
$t_{f(\phi)}$	Timer input fall time	t_f				30	ns

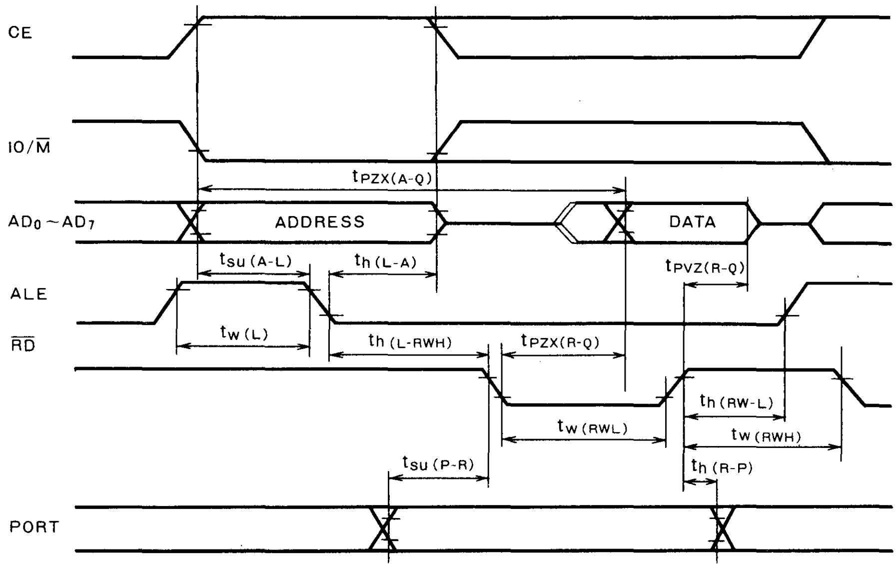
SWITCHING CHARACTERISTICS (Ta = 0 ~ 70°C, V_{CC} = 5V ± 5%, unless otherwise noted.)

Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{PZX(R-Q)}$	Propagation time from read to data output	t_{RD}				170	ns
$t_{PZX(A-Q)}$	Propagation time from address to data output	t_{AD}				400	ns
$t_{PVZ(R-Q)}$	Propagation time from read to data floating (Note 7)	t_{RDF}				100	ns
$t_{PHL(W-P)}$ / $t_{PLH(W-P)}$	Propagation time from write to data output	t_{WP} / t_{WP}				400	ns
$t_{PLH(STB-BF)}$	Propagation time from strobe to BF flag	t_{SBF}				400	ns
$t_{PHL(R-BF)}$	Propagation time from read to BF flag	t_{RBE}				400	ns
$t_{PLH(STB-INTR)}$	Propagation time from strobe to interrupt	t_{SI}				400	ns
$t_{PHL(R-INTR)}$	Propagation time from read to interrupt	t_{RDI}				400	ns
$t_{PHL(STB-BF)}$	Propagation time from strobe to BF flag	t_{SBE}				400	ns
$t_{PLH(W-BF)}$	Propagation time from write to BF flag	t_{WBF}				400	ns
$t_{PHL(W-INTR)}$	Propagation time from write to interrupt	t_{WI}				400	ns
$t_{PHL(\phi-OUT)}$ / $t_{PLH(\phi-OUT)}$	Propagation time from timer input to timer output	t_{TL} / t_{TH}				400	ns

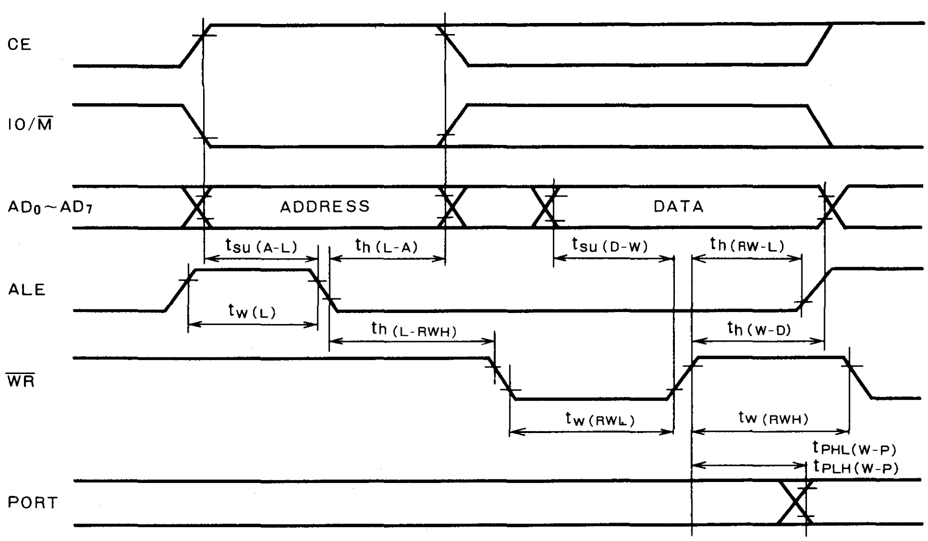
Note 6: Measurement conditions C = 150pF
7: Measurement conditions of note 6 are not applied.

TIMING DIAGRAM (reference level, high-level=2V, low-level=0.8V)

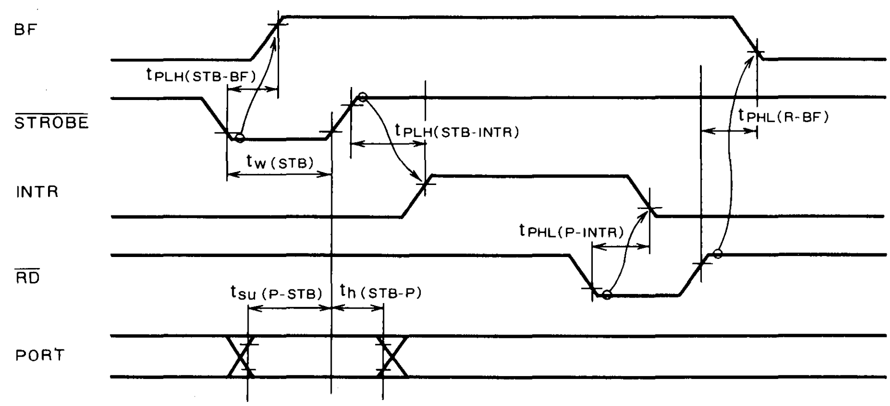
Basic Input



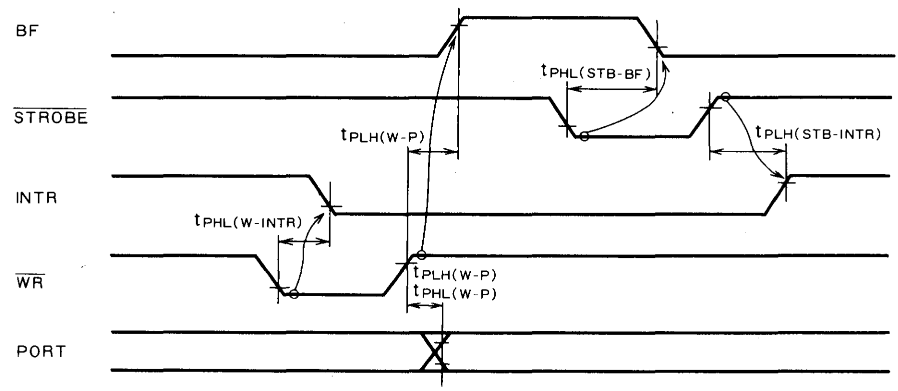
Basic Output



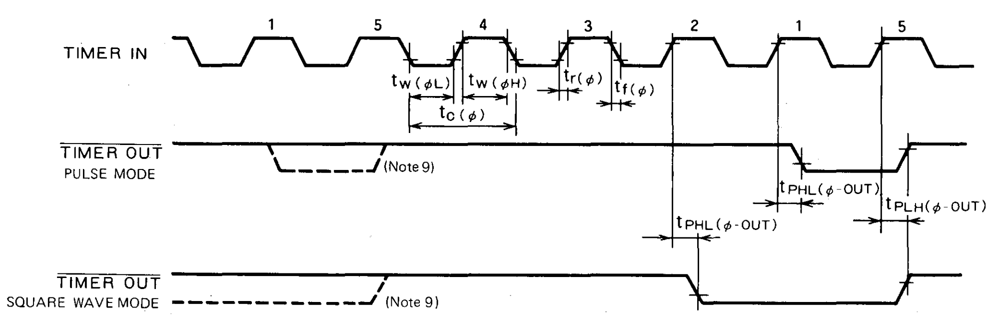
Strobed Input



Strobed Output



Timer (Note 8)



Note 8: The wave form is shown counting down from 5 to 1.
9: As long as the M1 mode flag of the timer register is at high-level, pulses are continuously output.

LSIs FOR PERIPHERAL CIRCUITS

8

8-BIT 8-CHANNEL A-D CONVERTER

DESCRIPTION

The M58990P A-D converter is used to convert analog signals to 8-bit digital values. The A-D converter is fabricated using silicon-gates and CMOS technology. The M58990P can selectively multiplex 8 channels of analog input.

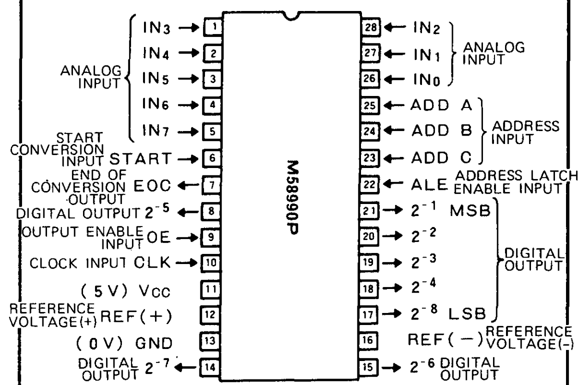
FEATURES

- Single 5V power supply
- Conversion resolution of 8 bits
- Multiplex 8 channels of analog input
- Broad range of analog input voltages: 0V ~ V_{CC}
- Conversion time: 100μs
- Conversion by successive approximation
- Can be used online through the data bus of a microprocessor
- The I/O pins can be connected directly to TTL circuits
- Interchangeable with NS's ADC0808 (in pin configuration)

APPLICATION

- Used with microcomputers to control analog systems

PIN CONFIGURATION (TOP VIEW)



Outline 28P4

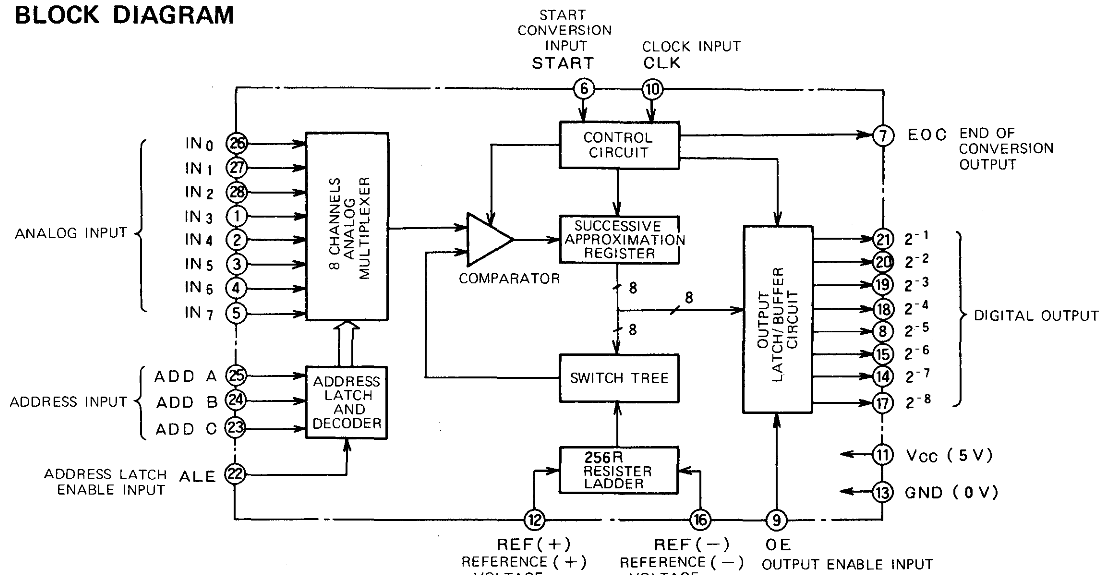
FUNCTION

The M58990P has eight analog input terminals that are selected by the input signals to the 3 address terminals (ADD A ~ ADD C). The address signals of these terminals are read and latched in the internal address latches by the ALE signal. When the OE terminal is at low-level, the output terminals $2^{-1} \sim 2^{-8}$ are in a floating state so they can be connected directly to the data bus of a microcomputer.

The input terminal START is used to call for the start of an analog to digital conversion and a signal is output through terminal EOC when the conversion is completed.

8

BLOCK DIAGRAM



PIN DESCRIPTIONS

Pin	Name	Input or Output	Functions
IN_0 ～ IN_7	Analog signal	Input	These are analog signal input pins. Which of the 8 inputs is selected, is determined by ADD A ～ ADD C. An analog voltage applied at the selected pin is converted to a digital value in the range of 2^{-1} ～ 2^{-8} and output.
ADD A ～ ADD C	Address signal	Input	The input is used for selecting which of the 8 terminals IN_0 ～ IN_7 is to be converted from analog to digital. The address input through ADD A ～ ADD C is read to the address latch by the riseing edge of ALE.
ALE	Address latch enable signal	Input	This is the strobe signal which causes the address signal input through ADD A ～ ADD C to be read and latched for use as an internal address.
REF(+)	Reference voltage (+)	Input	This is one of the input terminals for the reference voltage that is applied to the 256R resistor ladder circuit. The other terminal is REF (−) and the voltage levels of these two inputs must meet the condition: REF (+) > REF (−).
REF(−)	Reference voltage (−)	Input	This is one of the input terminals for the reference voltage that is applied to the 256R resistor ladder circuit. The other terminal is REF (+) and the voltage levels of these two inputs must meet the condition: REF (+) > REF (−).
OE	Output enable signal	Input	The signal at this pin controls the digital output. When the signal is low-level, pins 2^{-1} ～ 2^{-8} are in a floating state. When it is high-level, the data is output.
2^{-1} ～ 2^{-8}	Digital signal	Output	The analog signal, which was input through IN_0 ～ IN_7, is converted to digital data and is output from these terminals. When OE is low-level, these terminals are floating. When OE is high-level, the converted digital data is output. The MSB is 2^{-1} and the LSB is 2^{-8}.
EOC	End of conversion signal	Output	This terminals is used to indicate the completion of an analog to digital conversion. It is reset by a START signal (high-level to low-level) and is set on completion of the conversion (low-level to high-level). This output is normally used to generate an interrupt request for the CPU.
START	Start conversion signal	Input	The input signal at this terminal is used to start a conversion cycle by setting the successive approximation register. The successive approximation register is reset by rising from low-level to high-level and conversion is started after being set by falling from high-level to low-level.
CLK	Clock input	Input	The signal at this terminal is the basic clocking signal used to determine internal timing.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	−0.3∼7	V
V_I	Input voltage		$-0.3\sim V_{CC}+0.3$	V
V_O	Output voltage		$0\sim V_{CC}$	V
Pd	Maximum power dissipation	Ta=25°C	500	mW
T_{opr}	Operating free-air temperature range		0∼70	°C
T_{stg}	Storage temperature range		−65∼150	°C

RECOMMENDED OPERATING CONDITIONS (Ta=0∼70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
GND	Supply voltage		0		V
V_{IH}	High-level input voltage	2		V_{CC}	V
V_{IL}	Low-level input voltage	−0.3		0.8	V
$V_{REF(+)}$	Max of reference voltage (+)		V_{CC}	$V_{CC}+0.1$	V
$V_{REF(-)}$	Min of reference voltage (-)	−0.1	0		V
$\triangle V_{REF}$	Defferential of reference voltage		5.12	5.25	V
$V_{I(IN)}$	Analog input voltage	0		$V_{REF(+)}$	V

ELECTRICAL CHARACTERISTICS (Ta=0∼70°C, V_{CC}=5V ±5%, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
V_{IH}	High-level input voltage	$V_{IN(1)}$	V_{CC}= 5 V	2			V
V_{IL}	Low-level input voltage	$V_{IN(0)}$				0.8	V
V_{OH}	High-level output voltage	$V_{OUT(1)}$	$I_{OH}=-360\mu A$, Ta=70°C	$V_{CC}-0.4$			V
V_{OL}	Low-level output voltage, $2^{-1}\sim2^{-8}$ output	$V_{OUT(0)}$	I_{OL}= 1.6mA			0.45	V
$V_{OL(EOC)}$	Low-level output voltage, EOC output	$V_{OUT(0)}$	I_{OL}= 1.2mA			0.45	V
I_{IH}	High-level input current	$I_{IN(1)}$	V_{IH}=5.25V			1.0	μA
I_{IL}	Low-level input current	$I_{IN(0)}$	V_{IL}= 0 V			−1.0	μA
I_{OZH}	Off-state (high-impedance state) output current, $2^{-1}\sim2^{-8}$ output	I_{OUT}	V_O= 5 V			3	μA
I_{OZL}	Off-state (Low-impedance state) current current, $2^{-1}\sim2^{-8}$ output	I_{OUT}	V_O= 0 V			−3	μA
I_{CC}	Supply current from V_{CC} input		$f(\phi)$=500kHz, Ta=70°C			1000	μA
I_{IZ}	Off-state input current, ($IN_0\sim IN_7$ input)	$I_{OFF(+)}$	V_{CC}= 5 V, V_I = 5 V			200	nA
I_{IZ}	Off-state input current, ($IN_0\sim IN_7$ input	$I_{OFF(-)}$	V_{CC}= 5 V, V_I= 0 V			−200	nA
	Conversion resolution			8			Bits
	Linearity error				±1/4	±1/2	LSB
	Zero error				±1/4	±1/2	LSB
	Full-scale error				±1/4	±1/2	LSB
	Absolute precision					±1	LSB
R_{LADDER}	Ladder resistances		V_{CC}= 5 V	1			kΩ
Ci	Input capacitance	C_{IN}	V_I=GND, V_O=25mVrms,f=1MHz			8	pF
Co	Output capacitance	C_{OUT}	V_O=GND,V_O=25mVrms,f=1MHz			12	pF

Note 1. Current flowing into an IC is positive, and Min and Max show the absolute limit.

TIMING REQUIREMENTS (Ta = 25°C, V_{CC} = $V_{REP(+)}$ = 5V, $V_{REF(-)}$ = GND unless otherwise noted)

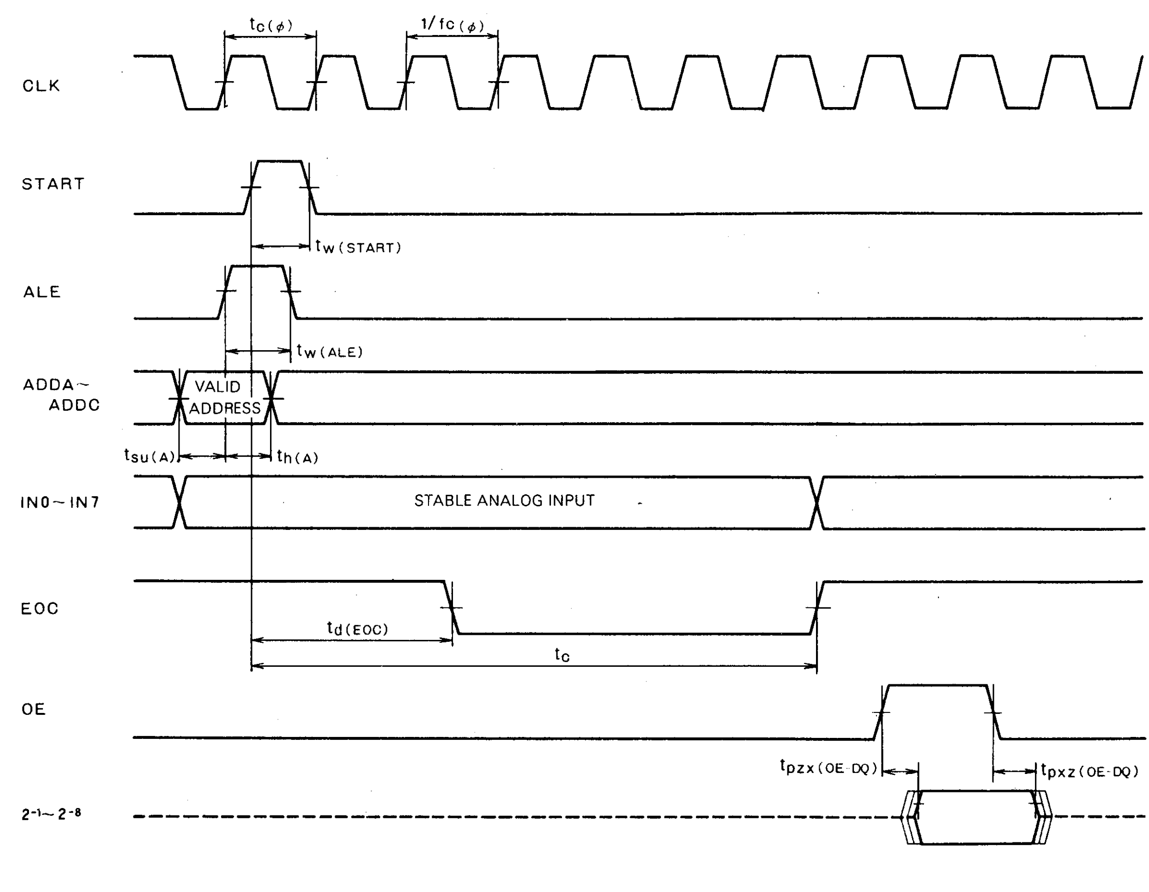
Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{W(START)}$	Start pulse width	t_{WS}		200			ns
$t_{W(ALE)}$	ALE pulse width	t_{WALE}		200			ns
$t_{SU(A)}$	Address setup time	t_S		50			ns
$t_{h(A)}$	Address hold time	t_H		50			ns
$f_{C(\phi)}$	Clock frequency	f_C		10	640	1200	kHz
$t_{C(\phi)}$	Clock cycle	—		100	1.56	0.83	μs

Note 2. Input voltage level is V_{IL} = 0.8V, V_{IH} = 2V

SWITCHING CHARACTERISTICS (Ta = 25°C, V_{CC} = $V_{REF(+)}$ = 5V, $V_{REF(-)}$ = GND, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{PZX(OE-DQ)}$	Propagation time from OE to output	t_{HI}, t_{HO}	C_L = 50pF			250	ns
$t_{PXZ(OE-DQ)}$	Propagation time from OE to output floating	t_{IH}, t_{OH}				250	ns
t_C	Cycle time	t_C	$f_{C(\phi)}$ = 640kHz			114	ns
$t_{d(EOC)}$	EOC dilay time	t_{EOC}		1		8	Clock cycle time

TIMING DIAGRAM



DESCRIPTION

The M5C6847P-1 is a color or monochrome television interface device, fabricated using N-channel silicon gate ED-MOS technology. The M5C6847P-1 has a 64-character (6-bit ASCII code) generator and memory interface.

FEATURES

- Can be easily connected to the MELPS 85 series 8-bit CPUs.
- Alphanumeric display: 4 modes
- Graphic display: 8 modes
- Can connect directly with the M51342P RF modulator
- Alphanumeric display: 32 characters per line by 16 lines
- Character generator for 64 ASCII characters
- Can be used with an external character generator
- Generates composite video signals
- Generates intensity signal Y, color signal R-Y (ϕA) and B-Y (ϕB)
- Display RAM capacity (depends on mode): 512~6K bytes
- Single 5V power supply
- Interchangeable with the Motorola's MC6847P in pin configuration

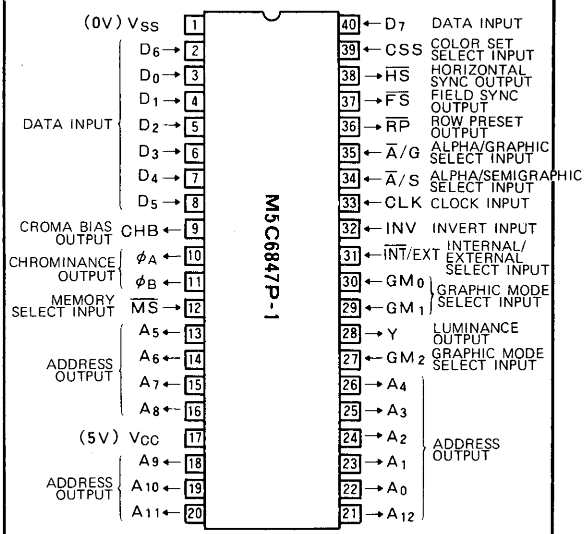
APPLICATION

- Microcomputer system or terminals using a color or monochrome CRT.

FUNCTION

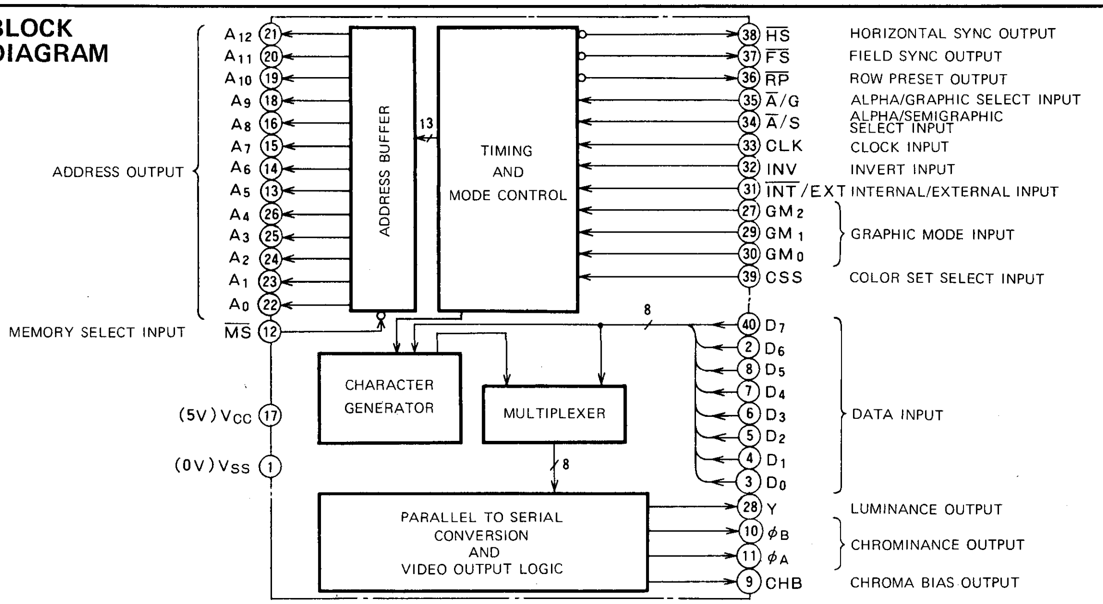
The picture on the television set is composed of the synchronization signals such as horizontal synchronization signal, vertical synchronization signal and color burst signal, and syncronizing serial data. M5C6847P-1 can generate these signals. The information or data to be shown on the screen is written in the display memory by the CPU. (When the picture is to be composed on a CRT) the data for one screen in the display memory is read in the order of th scan cycles and synchronization signals are added. This

PIN CONFIGURATION (TOP VIEW)



Outline 40P1

BLOCK DIAGRAM



serial is sent to the RF modulator. The M5C6847P-1 performs these functions by reading the display memory in the order of the CRT scan, adding the required synchronization signals such as luminance signal, color signal and then transferring the data stream serially to the RF modulator.

OPERATION

Address Outputs (A_{12}~A_0)

Thirteen address lines are used by the M5C6847P-1 to access the display memory (refresh memory). The starting address of the display memory is located at the upper-left corner of the display screen. As the television sweeps from the left to right and top to bottom, the VDG increments the RAM display address. The address lines are TTL-compatible and may be forced in a high-impedance state when input $\overline{MS}$ goes low.

Data Input (D_7~D_0)

Eight TTL-compatible data lines are used to input data from the display memory to be processed by the M5C-6847P-1. The data is interpreted and transformed into video analog level signals.

Video Output (Y, ϕ_A, ϕ_B, CHB)

These video outputs are used to transfer luminance and color information of pictures displayed on television with standard NTSC systems. These outputs can be directly connected to the RF modulator M51342P.

Luminance Output (Y)

The luminance output is a 6-level analog output. The six level analog outputs contain composite, blank, and four levels of video intensity.

Chrominance Output (ϕ_A)

The chrominance output ϕ_A is a 3-level analog output. The signal is used in combination with ϕ_B and Y to specify one of eight colors.

Chrominance Output (ϕ_B)

The chrominance output ϕ_B is a 4-level analog output. These levels of the signal are used in combination with ϕ_A and Y to specify one of eight colors. The other level is used to specify the time of the color burst reference signal.

Chroma Bias Output (CHB)

The chroma bias output is a single level analog output that provides the DC reference for chrominance outputs.

Synchronization Input ($\overline{MS}$, CLK)

Memory Select Input ($\overline{MS}$)

This is a TTL compatible input. When it goes low-level, address outputs (A_{12}~A_0) are forced in high-impedance state. When other devices such as the CPU access the display memory, it must be kept at low-level to prevent interference.

Clock input (CLK)

The clock input requires a 3.579545 MHz clock with a duty cycle of 50±5%. The M51342P RF modulator may be used to supply the 3.579545 MHz clock.

Synchronization output ($\overline{FS}$, $\overline{HS}$, $\overline{RP}$)

The synchronization outputs $\overline{FS}$, $\overline{HS}$ and $\overline{RP}$ are TTL-compatible and provide circuits, exterior to the M5C6847P-1 states.

Table 1 Operation modes

$\overline{A}$/G	$\overline{A}$/S	$\overline{INT}$/EXT	INV	GM_2	GM_1	GM_0	Mode
0	0	0	0	X	X	X	Internal alphanumerics
0	0	0	1	X	X	X	Internal alphanumerics inverted
0	0	1	0	X	X	X	External alphanumerics
0	0	1	1	X	X	X	External alphanumerics inverted
0	1	0	X	X	X	X	Semigraphics 4
0	1	1	X	X	X	X	Semigraphics 6
1	X	X	X	0	0	0	64× 64 Color graphics
1	X	X	X	0	0	1	128× 64 Graphics
1	X	X	X	0	1	0	128× 64 Color graphics
1	X	X	X	0	1	1	128× 96 Graphics
1	X	X	X	1	0	0	128× 96 Color graphics
1	X	X	X	1	0	1	128×192 Graphics
1	X	X	X	1	1	0	128×192 Color graphics
1	X	X	X	1	1	1	256×192 Graphics

Note 1: X is "don't care" bit

Table 2 Alphanumeric mode display memory, color and display element

Mode	Memory capacity (bits)	Color	Display elements
Internal alphanumerics	512×8	2	8 dots, 12 dots; Character is 5 x 7 dots
External alphanumerics	512×8	2	8 dots, 12 dots
Semigraphics 4	512×8	8	Elements 64×32
Semigraphics 6	512×8	4	Elements 64×48

Table 3 Graphic mode display memory, color and display element

Mode	Memory capacity (bits)	Color	Display elements
64× 64 Color graphics	1K×8	4	64×64
128× 64 Graphics 128× 64 Color graphics	1K×8 2K×8	2 4	128×64
128× 96 Graphics 128× 96 Color graphics	2K×8 3K×8	2 4	128×96
128×192 Graphics 128×192 Color graphics	3K×8 6K×8	2 4	128×192
256×192 Graphics	6K×8	2	256×192

Field synchronization output ($\overline{FS}$)
The high to low transition of the $\overline{FS}$ output coincides with the end of active display area. The low to high transition of $\overline{FS}$ coincides with the trailing edge of the vertical synchronization pulse. The CPU should not access display memory while $\overline{FS}$ is at low-level to avoid undesired flicker on the screen.

Horizontal synchronization output ($\overline{HS}$)
This signal is used for horizontal synchronization on the CRT. A fall from high-level to low-level indicates the leading edge of the horizontal synchronization signal.

Row preset output ($\overline{RP}$)
This signal can be used when an external character generator ROM that is used with the VDG. An external 4-bit binary counter must also be added to supply row selection.

The counter is clocked by the $\overline{HS}$ signal and cleared by the $\overline{RP}$ signal. See Table 4 ② for details.

Mode Control Inputs ($\overline{A}$/G, $\overline{A}$/S, $\overline{INT}$/EXT, GM_2, GM_1, GM_0, CSS and INV)

These eight TTL-compatible input signals are used to determine and control the operational modes of the M5C6847P-1. Outline and details of the operational modes are shown in Table 1~3.

Alphanumeric mode

A screen in the alphanumeric mode is composed of 32 characters x 16 lines. Each character occupies space equivalent to an 8 x 12 dot matrix. The internal character generator can generate 64 characters (6-bit ASCII). Each character is formed by a 5 x 7 dot matrix. The low-order 6 bits of the 8-bit data input are used to select 1 of 64 characters and the remaining 2 bits can be used to implement the CSS and INV signal inputs. Operation in this mode requires a display memory of a least 512 bytes.

Semigraphic 4 mode
A screen in the semigraphics 4 mode is composed of 64 x 32 display elements. A display element is a 4 x 6 dot matrix; that is to say, each 8 x 12 character dot matrix is split into 4 display elements, each display element being a 4 x 6 dot matrix. The low-order 4 bits of the 8-bit data input correspond to the 4 display elements of a character. Three data bits of the remaining 4 bits may be used to select one of eight colors for the entire character box. The extra bit is available to switch the operation mode. Operation in this mode requires a display memory of at least 512 bytes.

Semigraphics 6 mode
A screen in the semigraphics 6 mode is composed of 64 x 48 display elements. A display element is a 4 x 4 dot matrix; that is to say, each 8 x 12 character dot matrix is split into 6 display elements, each display element being a 4 x 4 dot matrix. The low-order 6 bits of the 8-bit data input to the 6 display elements of a character and the remaining 2 bits are used to determine color. Operation in this mode requires a display memory of at least 512 bytes.

Full Graphic Modes

There are 8 full graphic modes. The border color (green or white) is selected by the level of the CSS signal. The CSS pin selects one of two sets of four colors in the four color graphic modes.

Color Graphic Mode 64 x 64
A screen in the 64 x 64 color graphic mode is composed of 64 x 64 display elements. Each display element can be 1 of 4 colors. Operation in this mode requires a display memory of at least 1024 bytes.

Graphic mode 128 x 64
A screen in the 128 x 64 graphic mode is composed of 128 x 64 display elements. Each display element can be green or white depending on the level of the CSS signal. Operation in this mode requires a display memory of at least 1024 bytes.

Color graphic mode 128 x 64
A screen in the 128 x 64 color graphic mode is composed of 128 x 64 display elements. Each display element can be 1 of 4 colors. Operation in this mode requires a display memory of at least 2048 bytes.

Graphic mode 128 x 96
A screen in the 128 x 96 graphic mode is composed of 128 x 96 picture elements. Each display element can be geeen or white depending on the level of the CSS signal. Operation in this mode requires a display memory of at least 2048 bytes.

Color graphic mode 128 x 96
A screen in the 128 x 96 color graphic mode is composed of 128 x 96 display elements. Each display element can be 1 of 4 colors. Operation in this mode requires a display memory of at least 3072 bytes.

Graphic mode 128 x 192
A screen in the 128 x 192 graphic mode is composed of 128 x 192 display elements. Each display element can be green or white depending on the level of the CSS signal. Operation in this mode requires a display memory of at least 3072 bytes.

Color graphic mode 128 x 192
A screen in the 128 x 192 color graphic mode is composed of 128 x 192 display elements. Each picture element can be 1 of 4 colors. Operation in this mode requires a display memory of at least 6144 bytes.

Graphic mode 256 x 192
A screen in the 256 x 192 graphic mode is composed of 256 x 192 display elements. Each display element can be green or white depending on the level of the CSS signal. Operation in this mode requires a display memory of at least 6144 bytes.

Details of the 8 graphic modes are shown in Table 4 which gives more information in an easy to understand form.

Table 4 Operational characteristics in the various graphic modes

	Input Pin									Color			TV Screen (1 screen is composed of 256x192 dots)		Data Bus	Display Mode
	$\overline{MS}$	$\overline{A}$/G	$\overline{A}$/S	$\overline{INT}$/EXT	GM_2	GM_1	GM_0	CSS	INV	Character Color	Background	Border	Mode	Display Elements		
①	1	0	0	0	X	X	X	0	0	green	black	black	16 lines of 32 characters	5x7 Dots 1 Character	D_7 6 5 4 3 2 1 0 ASCII Code	Alphanumeric mode
									1	black	green					
								1	0	orange	black	black				
									1	black	orange					
②	1	0	0	1	X	X	X	0	0	green	black	black	16 lines of 32 characters	8x12 Dots 1 Character	External ROM Code	Alphanumeric mode
									1	black	green					
								1	0	orange	black	black				
									1	black	orange					
③	1	0	1	0	X	X	X	X	X	$D_{0\sim3}$ D_6 D_5 D_4 0 X X X black 1 0 0 0 green 1 0 0 1 yellow 1 0 1 0 blue 1 0 1 1 red 1 1 0 0 white 1 1 0 1 cyan 1 1 1 0 magenta 1 1 1 1 orange		black	64×32 display elements	4 Dots; 6 Dots D_3 D_2 / D_1 D_0 All 4 picture elements of the character group are the same color. The color intensity is 0 (black) or 1 (full color).	Color, Luminance	Semigraphics 4 mode
④	1	0	1	1	X	X	X	0	X	$D_{0\sim5}$ D_7 D_6 0 X X black 1 0 0 green 1 0 1 yellow 1 1 0 blue 1 1 1 red		black	64×48 display elements	4 Dots; 4 Dots D_5 D_4 / D_3 D_2 / D_1 D_0 All 6 picture elements of the character group are the same color. The color intensity is 0 (black) or 1 (full color).	Color, Luminance	Semigraphics 6 mode
								1		0 X X black 1 0 0 white 1 0 1 cyan 1 1 0 magenta 1 1 1 orange						
⑤	1	1	X	X	0	0	0	0	X	D_7 D_6 (D_5, D_4, D_3, D_2, D_1, D_0) 0 0 green 0 1 yellow 1 0 blue 1 1 red		green	64×64 display elements	2 Dots E_3 E_2 E_1 E_0 3 Dots	Color E_3 E_2 E_1 E_0	Color graphic mode 64x64
								1		0 0 white 0 1 cyan 1 0 magenta 1 1 orange		white				
⑥	1	1	X	X	0	0	1	0	X	D_7 (D_6, D_5, D_4, D_3, D_2, D_1, D_0) 0 black 1 green		green	128×64 display elements	2 Dots 7 6 5 4 3 2 1 0 3Dots	Luminance	Graphic mode 128x64
								1		0 black 1 white		white				
⑦	1	1	X	X	0	1	0	0	X	The same as ⑤		green	128×64 display elements	2 Dots E_3 E_2 E_1 E_0 3Dots	Color E_3 E_2 E_1 E_0	Graphic mode 128x64
								1				white				
⑧	1	1	X	X	0	1	1	0	X	The same as ⑥		green	128×96 display elements	2 Dots 7 6 5 4 3 2 1 0 2Dots	Luminance	Graphic mode 128x96
								1				white				
⑨	1	1	X	X	1	0	0	0	X	The same as ⑤		green	128×96 display elements	2 Dots E_3 E_2 E_1 E_0 2 Dots	Color E_3 E_2 E_1 E_0	Color graphic mode 128x96
								1				white				
⑩	1	1	X	X	1	0	1	0	X	The same as ⑥		green	128×192 display elements	2 Dots 7 6 5 4 3 2 1 0 1 Dot	Luminance	Graphic mode 128x192
								1				white				
⑪	1	1	X	X	1	1	0	0	X	The same as ⑤		green	128×192 display elements	2 Dots E_3 E_2 E_1 E_0 1 Dot	Color E_3 E_2 E_1 E_0	Color graphic mode 128x192
								1				white				
⑫	1	1	X	X	1	1	1	0	X	The same as ⑥		green	256×192 display elements	1 Dot 7 6 5 4 3 2 1 0 1 Dot	Luminance	Graphic mode 256x192
								1				white				

Internal Character Generator

The M5C6847P-1 generates the 64 standard ASCII characters in a 5 x 7 dot matrix form. It generates the 64 standard ASCII characters according to a 6-bit code. The code for each character is showed in Table 5.

Table 5 M5C6847P-1 character set

Code						Character	Code						Character
D_5	D_4	D_3	D_2	D_1	D_0		D_5	D_4	D_3	D_2	D_1	D_0	
0	0	0	0	0	0	@	1	0	0	0	0	0	SP
0	0	0	0	0	1	A	1	0	0	0	0	1	!
0	0	0	0	1	0	B	1	0	0	0	1	0	"
0	0	0	0	1	1	C	1	0	0	0	1	1	#
0	0	0	1	0	0	D	1	0	0	1	0	0	$
0	0	0	1	0	1	E	1	0	0	1	0	1	%
0	0	0	1	1	0	F	1	0	0	1	1	0	&
0	0	0	1	1	1	G	1	0	0	1	1	1	'
0	0	1	0	0	0	H	1	0	1	0	0	0	(
0	0	1	0	0	1	I	1	0	1	0	0	1	)
0	0	1	0	1	0	J	1	0	1	0	1	0	*
0	0	1	0	1	1	K	1	0	1	0	1	1	+
0	0	1	1	0	0	L	1	0	1	1	0	0	,
0	0	1	1	0	1	M	1	0	1	1	0	1	—
0	0	1	1	1	0	N	1	0	1	1	1	0	.
0	0	1	1	1	1	O	1	0	1	1	1	1	/
0	1	0	0	0	0	P	1	1	0	0	0	0	0
0	1	0	0	0	1	Q	1	1	0	0	0	1	1
0	1	0	0	1	0	R	1	1	0	0	1	0	2
0	1	0	0	1	1	S	1	1	0	0	1	1	3
0	1	0	1	0	0	T	1	1	0	1	0	0	4
0	1	0	1	0	1	U	1	1	0	1	0	1	5
0	1	0	1	1	0	V	1	1	0	1	1	0	6
0	1	0	1	1	1	W	1	1	0	1	1	1	7
0	1	1	0	0	0	X	1	1	1	0	0	0	8
0	1	1	0	0	1	Y	1	1	1	0	0	1	9
0	1	1	0	1	0	Z	1	1	1	0	1	0	:
0	1	1	0	1	1	[	1	1	1	0	1	1	;
0	1	1	1	0	0	\	1	1	1	1	0	0	<
0	1	1	1	0	1	]	1	1	1	1	0	1	=
0	1	1	1	1	0	↑	1	1	1	1	1	0	>
0	1	1	1	1	1	←	1	1	1	1	1	1	?

EXAMPLE OF DISPLAY ON CRT

The M5C6847P-1 can be used to generate characters for display on a video screen. An example of a display is shown in Fig. 1.

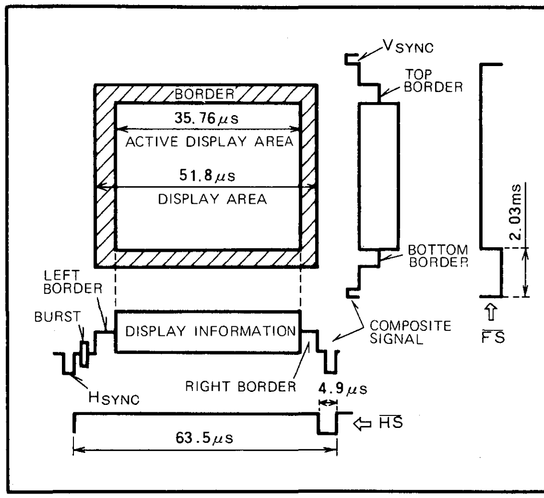


Fig. 1 Example of a display by a M5C6847P-1

APPLICATION EXAMPLE

One example of interfacing a M5C6847P-1 with a television set for home use is shown in Fig. 2. A M5L8085AP is used as the CPU in the example shown. The CPU executes the programs to control display and write the information for one screen into display memory. The M5C6847P-1 performs the main functions of interfacing with the CRT such as synchronizing scan, reading the display information from the display memory while adding necessary synchronization signals and sending to the RF modulator.

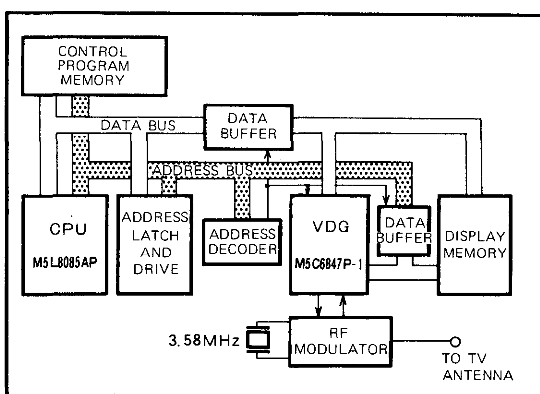


Fig. 2 Application example using the M5C6847P-1

8

A schematic for using the M5C6847P-1 with the M51342P RF modulator is shown in Fig. 3. M51342 requires ±5V power supplies. The video signal and chroma signal from the M5C6847P-1 can be modulated with the sound signal to form a RF signal that appears the same as the television antenna input signal. The video amp circuit to enable direct connection to a M5C6847P-1 is shown in Fig. 4. This can be connected to the monochrome video monitor. In this case, the inpedance is 75Ω.

Four levels of brightness (black, low, medium and high) can display a clear picture.

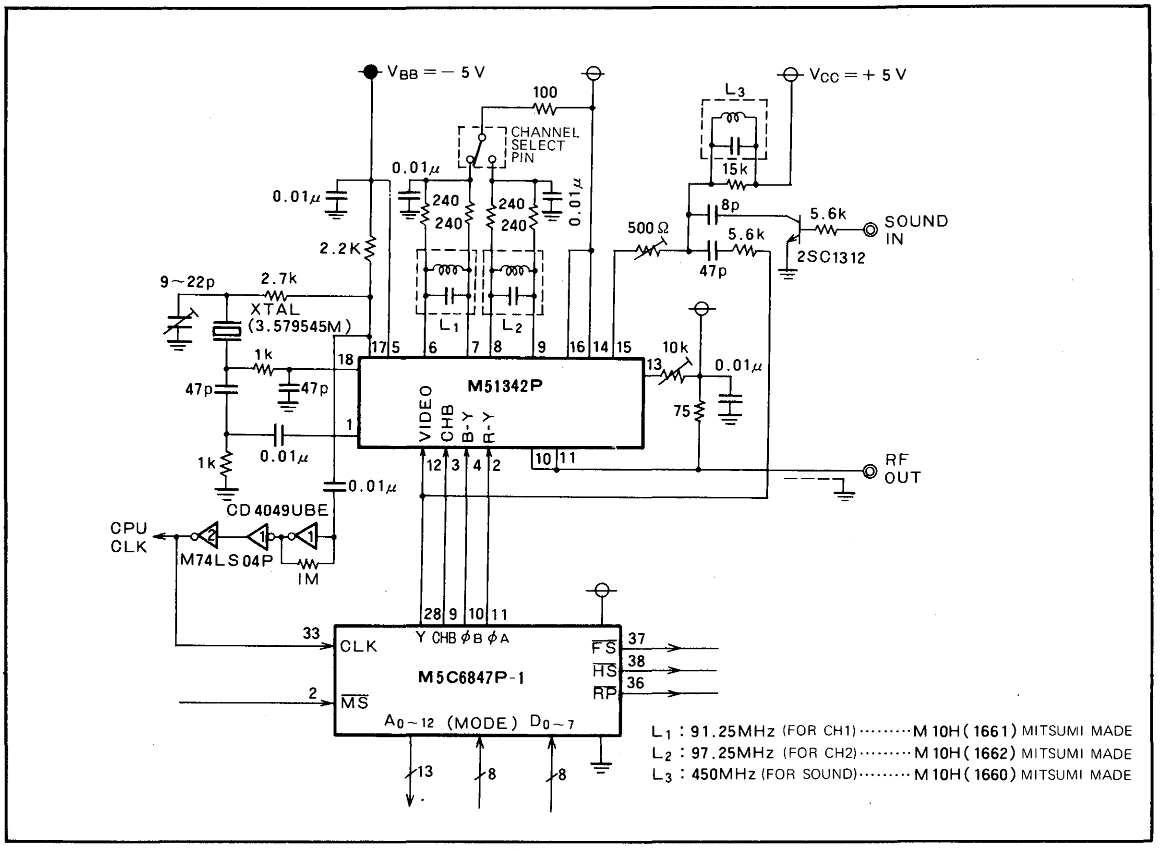


Fig. 3 Schematic for using the M51342P (RF modulator) with the M5C6847P-1

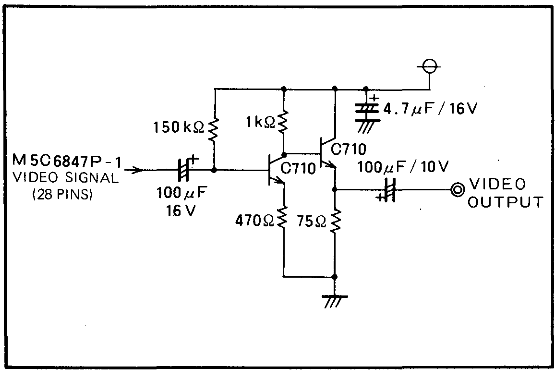


Fig. 4 Video amp circuit

Data and Display Relation

The relation between data and 5 display modes is shown in Table 6.

Table 6 Data and display relation

Mode	Data	Display
Character	D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0 6-bit ASCII (D_5–D_0)	5 dots; Height 12 dots; 7 dots; Width 8 dots
Semigraphic 4	D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0 Color designation (D_6–D_4) Display element ON/OFF indication (D_3–D_0)	D_1 D_0 / D_2 D_3; Display element
Semigraphic 6	D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0 Color designation (D_7–D_6) Display element ON/OFF indication (D_5–D_0)	D_1 D_0 / D_3 D_2 / D_5 D_4; Display element
Color-graphic (4 colors)	D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0 Color designation, Color designation, Color designation, Color designation	Display element
Graphic (2 colors)	D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0 Display element ON/OFF indication	Display element

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.3～7	V
V_I	Input voltage	With respect to V_{SS}	−0.3～7	V
V_O	Output voltage	With respect to V_{SS}	−0.3～7	V
Pd	Power dissipation	Ta=25°C	1000	mW
Topr	Operating free-air temperature range		0～70	°C
Tstg	Storage temperature range		−65～150	°C

RECOMMENDED OPERATING CONDITIONS (Ta=0～70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{SS}	Supply voltage		0		V
$V_{IH(\phi)}$	High-level input voltage, clock	2.4		V_{CC}	V
V_{IH}	High-level input voltage	2		V_{CC}	V
$V_{IL(\phi)}$	Low-level input voltage, clock	−0.3		0.4	V
V_{IL}	Low-level input voltage	−0.3		0.8	V

ELECTRICAL CHARACTERISTICS (Ta=0～70°C, V_{CC}=5V±5%, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage, except for ϕ_A, ϕ_B, Y, and CHB output	V_{SS}=0V, I_{OH}=−100μA, C_L=30pF	2.4			V
V_{OL}	Low-level output voltage, except for ϕ_A, ϕ_B, Y and CHB output	V_{SS}=0V, I_{OL}=1.6mA, C_L=30pF			0.4	V
I_{IH}	High-level input current	V_{SS}=0V, V_I=5.25V	−10		10	μA
I_{IL}	Low-level output current	V_{SS}=0V, V_I=0V	−10		10	μA
I_{OZ}	Output floating leak current	V_{SS}=0V, V_I=0.4V, MS=0.4V	−10		10	μA
I_{CC}	Supply current from V_{CC}	V_{SS}=0V			150	mA
Ci	Input capacitance	V_I=0V, f=1MHz, Ta=25°C			10	pF
Co	Output capacitance	V_I=0V, f=1MHz, Ta=25°C			20	pF
V_{CHB}	Chroma bias voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		0.6V_{CC}		V
$V_{\phi A,H}$	ϕ_A chrominance high-level output voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		V_{CHB}+0.16V_{CC}		V
$V_{\phi A,M}$	ϕ_A chrominance medium-level output voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		V_{CHB}		V
$V_{\phi A,L}$	ϕ_A chrominance low-level output voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		V_{CHB}−0.16V_{CC}		V
$V_{\phi B,H}$	ϕ_B chrominance high-level output voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		V_{CHB}+0.16V_{CC}		V
$V_{\phi B,M}$	ϕ_B chrominance medium-level output voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		V_{CHB}		V
$V_{\phi B,B}$	ϕ_B chrominance burst-level output voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		V_{CHB}−0.08V_{CC}		V
$V_{\phi B,L}$	ϕ_B chrominance low-level output voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		V_{CHB}−0.16V_{CC}		V
V_{YSYNC}	Luminance sync output voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		0.74V_{CC}		V
V_{YBLANK}	Luminance blank output voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		0.85 V_{YSYNC}		V
V_{YBLACK}	Luminance black output voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		0.81 V_{YSYNC}		V
$V_{YW(H)}$	White luminance high-level output voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		0.62 V_{YSYNC}		V
$V_{YW(M)}$	White luminance medium-level output voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		0.69 V_{YSYNC}		V
$V_{YW(L)}$	White luminance low-level output voltage	V_{SS}=0V, C_L=20pF, R_L=200kΩ		0.77 V_{YSYNC}		V

TIMING REQUIREMENTS (Ta=0~70°C, V_{CC}=5V±5%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$f_{C(\phi)}$	Clock frequency		3.579535	3.579545	3.579555	MHz
f_{DUTY}	Clock duty ratio		45	50	55	%
$t_{r(\phi)}$	Clock rise time				10	ns
$t_{f(\phi)}$	Clock fall time				10	ns
$t_{a(A-D)I}$	Address access time of display memory	Internal character mode			900	ns
$t_{a(A-D)E}$	Address access time of display memory + Address access time of external character ROM	External character mode			900	ns

SWITCHING CHARACTERISTICS

Composite video and chroma (Ta=0~70°C, V_{CC}=5V±5%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{w(YSYNC)}$	Luminance output synchronization signal pulse width			4.89		μs
$t_{w(YFP)}$	Luminance output front pot signal pulse width			1.96		μs
$t_{w(YHBLANK)}$	Luminance output horizontal blank signal pulse width			11.73		μs
$t_{r(YHSYNC)}$	Luminance output horizontal synchronization signal rise time				250	ns
$t_{f(YHSYNC)}$	Luminance output horizontal synchronization signal fall time				250	ns
$t_{r(YHBLANK)}$	Luminance output horizontal blank signal rise time				340	ns
$t_{f(YHBLANK)}$	Luminance output horizontal blank signal fall time				340	ns

CHROMA

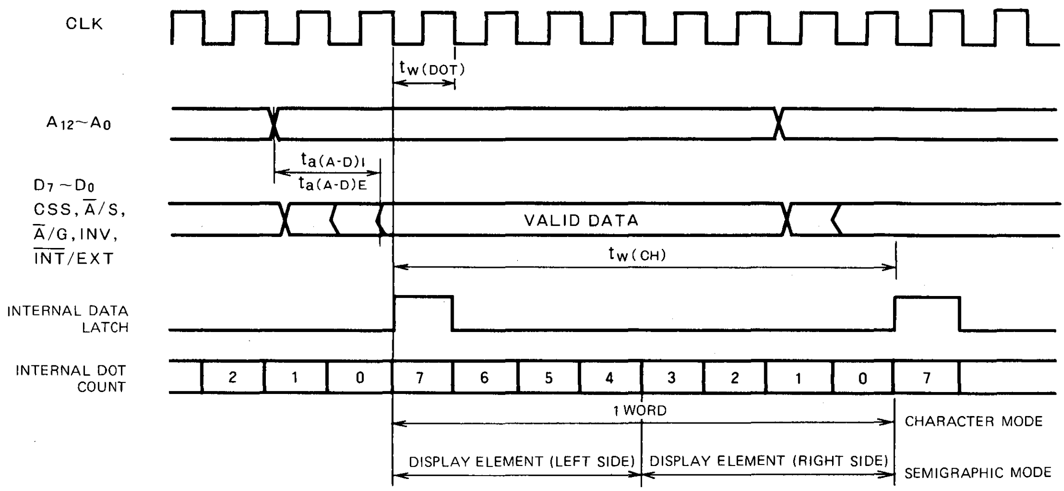
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{r(\phi A)}$	ϕ_A chrominance output rise time			60		ns
$t_{f(\phi A)}$	ϕ_A chrominance output fall time			60		ns
$t_{r(\phi B)}$	ϕ_B chrominance output rise time			60		ns
$t_{f(\phi B)}$	ϕ_B chrominance output fall time			60		ns
$t_{PHL(SYNC-BURST)}$	ϕ_B chrominance output propagation time after luminance syncronization signal output			980		ns
$t_{w(BURST)}$	ϕ_B chrominance output burst signal pulse width			2.93		μs
$t_{r(BURST)}$	ϕ_B chrominance output burst signal rise time			60		ns
$t_{f(BURST)}$	ϕ_B chrominance output burst signal fall time			60		ns
$t_{PHL(Y-CH)}$ $t_{PLH(Y-CH)}$	Chrominance propagation time after luminance output			0		ns

MISCELLANEOUS

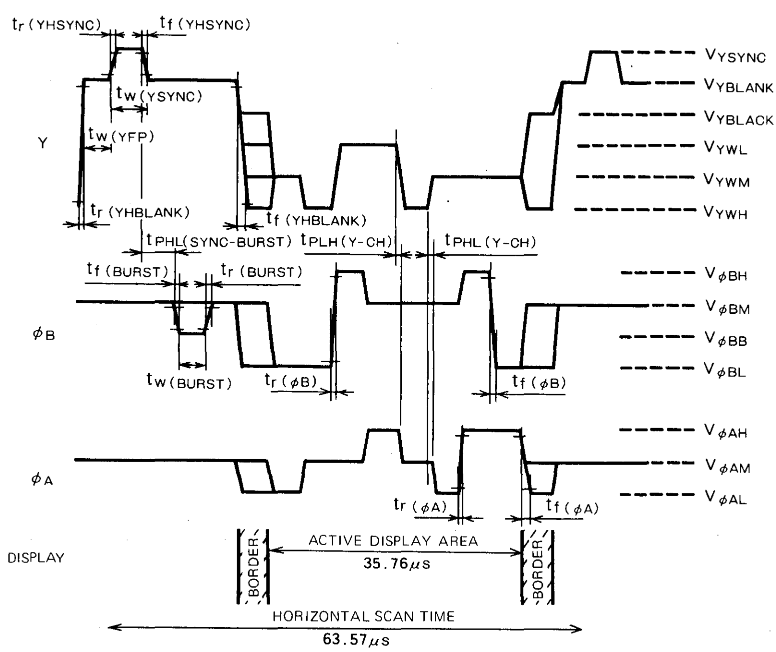
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{w(FS)}$	Field syncronization pulse width			2.03		ms
$t_{w(RP)}$	Row preset pulse width			980		ns
$t_{PHL(HS-RP)}$	$\overline{RP}$ propagation time after $\overline{HS}$			980		ns
$t_{w(HS)}$	Horizontal syncronization pulse width			4.9		μs
$t_{w(CH)}$	Character width			1.12		μs
$t_{w(DOT)}$	Dot width			140		ns

TIMING DIAGRAM

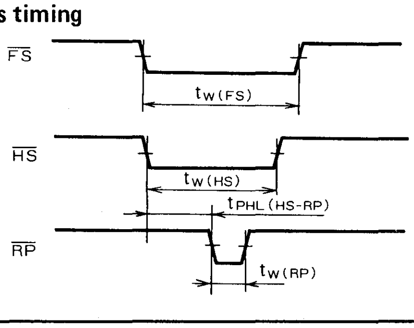
Display memory access



Composite video and chroma



Miscellaneous timing



UNIVERSAL PERIPHERAL INTERFACE

DESCRIPTION

The M5L8041A-XXXP is a general-purpose, programmable interface device deisgned for use with a variety of 8-bit microcomputer systems. This device is fabricated using N-channel silicon-gate ED-MOS technology.

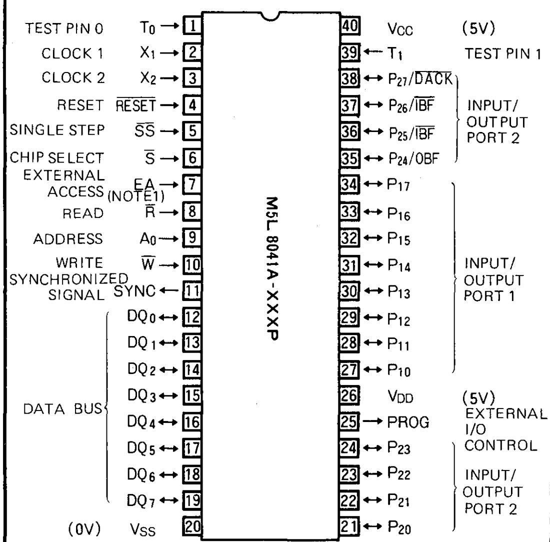
FEATURES

- Mask ROM: .... 1024-word by 8-bit
- Static RAM: .... 64-word by 8-bit
- 18 programmable I/O pins
- Asynchronous data register for interface to master processor
- 8-bit CPU, ROM, RAM, I/O, timer, clock and low power standby mode
- Single 5V supply
- Alternative to custom LSI
- Interchangeable with Intel's 8041A in function, electrical characteristics and pin configuration

APPLICATION

- Alternative to custom LSI for peripheral interface

PIN CONFIGURATION (TOP VIEW)



Package Outline 40P1

Note 1: Connect to V_{SS} in the operating condition.

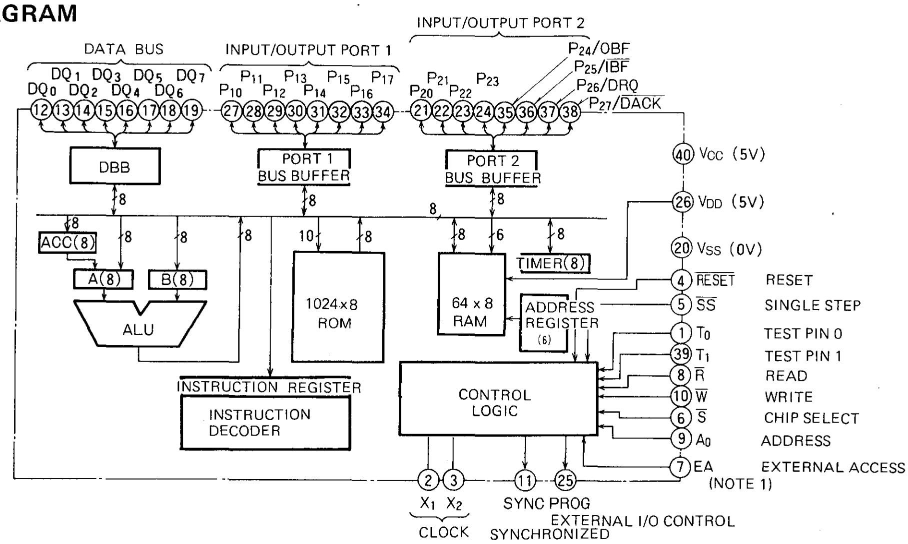
FUNCTION

The M5L8041A-XXXP contains a small stand-alone microcomputer.

When it is used as a peripheral controller, it is called the slave computer in contrast to the master processor. These two devices can transfer the data alternatively through the buffer register between them. The M5L8041A-XXXP contains the buffer register to use this LSI as a slave computer, and can be accessed the same as other standard peripheral devices. Because M5L8041A-XXXP is a complete microcomputer, it is easy to develop a user-oriented mask-programmed peripheral LSI only by changing control software.

8

BLOCK DIAGRAM



PIN DESCRIPTION

Pin	Name	Input or output	Function
V_{SS}	Ground	—	Connected to a 0V supply (ground)
V_{CC}	Main power supply	—	Connected to a 5V supply
V_{DD}	Power supply	—	1 Connected to a 5V supply 2 Used as a memory hold supply when V_{CC} is cut off
PROG	Program	Out	Serves as the strobe signal when an M5L8243P I/O expander is used
P_{10}~P_{17}	Port 1	In/out	Quaisi-bidirectional port. When used as an input port, FF_{16} must first be output to this prot. After resetting, however, when not used afterwards as an output port, this is not necessary.
P_{20}~P_{27}	Port 2	In/out	1 The same as port 1 2 P_{20}~P_{23} are used when an M5L8243P I/O port expander is used
DQ_0~DQ_7	Data bus	In/out	Serves as a sync signal for read and write operations to and from the bidirectional bus. Data remains latched.
T_0	Test pin 0	In	Provides external control of conditional program jumps (JT0/JNT0 instructions).
T_1	Test pin 1	In	1 Provides external control of conditional program jumps (JT1/JNT1 instructions). 2 Can serve as the input pin for the event counter (STRT CNT instructions).
$\overline{S}$	Chip select input	In	Chip select input for data bus control
$\overline{R}$	Read enable signal	In	Serves as the read signal when the master CPU is accepting data on the data bus from the M5L8041A-XXXP.
$\overline{W}$	Write enable signal	In	Serves as the write signal when the master CPU is outputting data from the bus to the M5L8041A-XXXP.
$\overline{RESET}$	Reset	In	CPU initialization input
SYNC	Sync signal output	Out	Output 1 time for each machine cycle
A_0	Address input	In	An address input used to indicate whether the signal on the data bus is data or a command
$\overline{SS}$	Single step	In	Used to halt the execution of a command by the CPU. When used in combination with the SYNC signal, the command execution of the CPU can be halted every instruction to enable single step operation.
EA	External access	In	Normally maintained at 0V
X_1, X_2	Crystal inputs	In	An internal clock circuit is provided so that by connecting an RC circuit or crystal to these input pins the clock frequency can be determined. Pins X_1 and X_2 can also be used to input an external clock signal.

BASIC FUNCTION BLOCKS

Program Memory (ROM)

The M5L8041A-XXXP contains 1024 bytes of ROM. The program for the users application is stored in this ROM. Addresses 0, 3, 7 of the ROM are reserved for special functions. Table 1 shows the meaning and function of these three special addresses.

Table 1 Reserved, defined addresses and their meanings and functions

Address	Meaning and function
0	The first instruction executed after a system reset.
3	The first instruction executed after an external interrupt is accepted.
7	The first instruction executed after a timer interrupt is accepted.

The ROM can be used to store constants and other 8-bit fixed data in addition to the program. Instructions such as MOVPA, @A and MOVP3A, @A can be used to access the constants and data. The data could be in the form of tables, and can be easily looked up.

Data Memory (RAM)

The M5L8041A-XXXP contains 64 bytes of RAM. The RAM is used for data storage and manipulation and is divided into sections for more efficient processing. Addresses 0~7 and 24~31 form two banks of general purpose registers that can be directly addressed. Addresses 0~7 compose bank 0 and are numbered R_0~R_7. Addresses 24~31 compose bank 1 and are also numbered R_0~R_7. Only one bank is active at a time. The instructions SEL RB0 and SEL RB1 are used to select the working bank. Fig. 1 shows the division of the RAM and its mapping.

The remaining section, addresses 32 and above, must be accessed indirectly using the general-purpose registers R_0 or R_1. Of course all addresses can be indirectly addressed using the general-purpose registers R_0 and R_1.

A good practice to simplify programming is to reserve general-purpose register bank 0 for use of the main program and register bank 1 for interrupt programs. For example if register bank 0 (addressed 0~7) is reserved for processing data by the main program, when an interrupt is accepted the first instruction would be to switch the working registers from bank 0 to bank 1. This would save the data of the main program (addresses 0~7). The interrupt program can then freely use register bank 1 (addresses 24~31) without destroying or altering data of the main program. When the interrupt processing is complete and control is returned to the main program by the RETR instruction, register bank 0 (in this example) is automatically restored as the working register bank at the same time the main program counter is restored.

Addresses 8~23 compose an 8-level program counter stack. The details for using the stack will be found in the "Program Counter and Stack" section. Please refer to that section for details.

Address 0~31 have special functions, but when not all of the registers are required, the ones not needed can be used for general storage. This includes both banks of general-purpose registers and the stack.

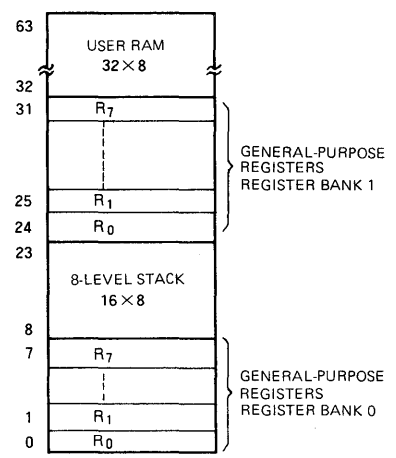


Fig. 1 Data memory (RAM)

8

Program Counter (PC) and Stack (SK)

The M5L8041A-XXXP program counter is a 10-bit counter configured as shown in Fig. 2.

When an interrupt or a subroutine call has occurred, the program currently being executed is interrupted and the execution flow must transfer to the interrupt program or subroutine. When such a condition has been encountered, the value currently stored in the program counter must be saved for use when restarting execution of the original program flow. The storage provided for this saving operation is the program counter stack. The program stack counter is used to store not only the program counter value but simultaneously store 4 bits of the PSW (Program Stack Word). The program counter stack uses addressed 8~23 of RAM. Ten bits are required to store the program counter value while 4 bits are required for the PSW. Therefore, each save operation uses 2 bytes (16 bits) of RAM. Thus, using RAM addresses 8~23, 8 levels of program counter value and PSW can be stored on the program stack. This storage scheme is shown in Fig. 3. To store which program counter stack location has last been entered, a 3-bit stack pointer is used. This stack pointer is also part of he PSW. However, the stack pointer itself is not stored in the program counter stack. The stack pointer is automatically incremented by 1 every time the program counter value and PSW are stored on the program stack. In the reverse operation in which these values are read from the program stack, the program stack pointer is decremented by 1. Note that the program counter stack always indicates the next storage location for the program counter value. Therefore, when return is made from a subroutine (using RET or RETR), the stack pointer is first decremented by 1, after which the contents of the program stack at the location indicated by the stack pointer are transferred to the program counter.

PC_9	PC_8	PC_7	PC_6	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0

Fig. 2 Program counter

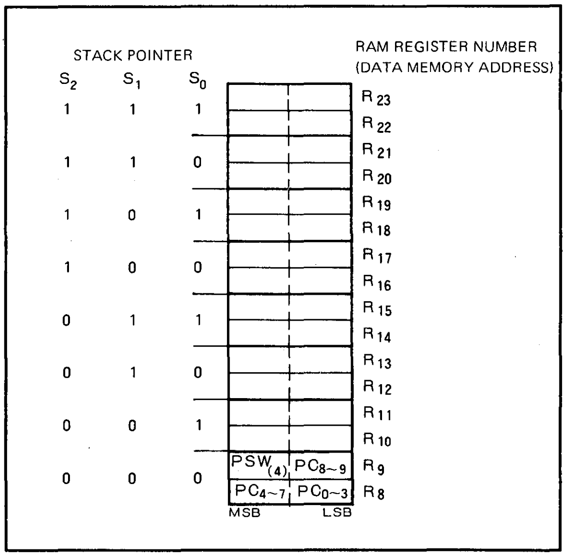


Fig. 3 Relation between the program counter stack and the stack pointer

PROGRAM STATUS WORD (PSW)

The PSW (program status word) is stored in 8 bits of register storage. The configuration of the PSW is shown in Fig. 4. The high-order 4 bits of the PSW are stored in the stack, along with the PC, when an interrupt is accepted or a subroutine call executed. When control is returned to the main program by RETR both the PC and the high-order 4 bits of PSW are restored. When control is returned by RET only the PC is restored, so care must be taken to assure that the contents of the PSW was not unintentionally changed.

The order and meaning of the 8 bits of the PSW are shown below.

Bit 0~2: Stack pointer (S_0, S_1, S_2)
Bit 3: Unused (always 1)
Bit 4: Working register bank indicator
0 = Bank 0 1 = Bank 1
Bit 5: Flag 0 (F_0) (value is set by the user and can be tested)
This value can be checked by JFO.
Bit 6: Auxiliary carry (AC) (it is set/reset by instructions ADD and ADC and used by instruction DA A).
Bit 7: Carry bit (C) (indicates an overflow after execution)

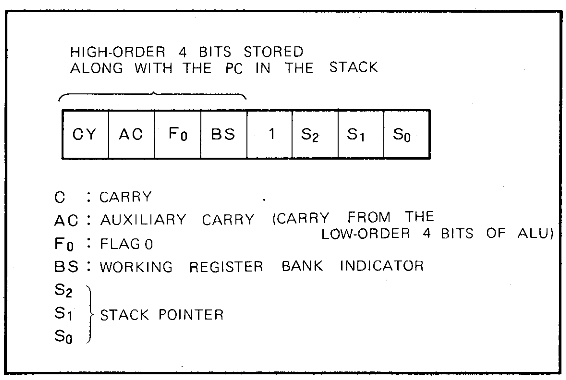


Fig. 4 Program status word

I/O PORTS

The M5L8041-XXXP has two 8-bit ports, which are called data bus, port 1 and port 2.

Port 1 and Port 2

Ports 1 and 2 are both 8-bit ports with identical properties. The output data of these ports are retained and do not change until another output is loaded into them. When used as inputs the input data is not retained so the input signals must be maintained until an input instruction is executed and completed.

Ports 1 and 2 so-called quasi-bidirectional ports have a special circuit configuration to accomplish this. The special circuit is shown in Fig. 5. All pin of ports 1 and 2 can be used for input or output.

Internal on chip pull-up resistors are provided for all the ports. Through the use of approximately 50kΩ pull-up

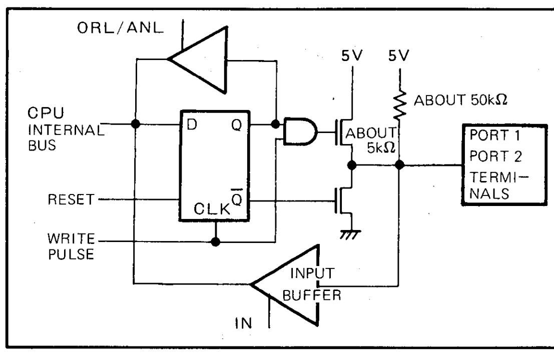


Fig. 5 I/O ports 1 and 2 circuit

resistors, TTL standard high-level or low-level signals can be supplied. Therefore each pin cán be used for both input and output. To shorten switching time from low-level to high-level, when 1s are output, a device of about 5kΩ or lower is inserted for a short time (about 500ns when using a 6MHz crystal oscillator).

A port used for input must output all 1s before it reads the data from the input pin. After resetting, a port is set to an input port and remains in this state, therefore it is not necessary to output all 1s if it is to be used for input. In short a port being used for output must output 1s before it can be used for input.

The individual pins of quasi-bidirectional ports can be used for input or output. Therefore some pins can be in the input mode while the remaining pins of a port are in the output mode. This capability of ports 1 and 2 is convenient for inputting or outputting 1-bit or data with few bits. The logical instructions ANL and ORL can easily be used to manipulate the input or output of these ports.

(2) Data Bus

The data bus (DQ_0~DQ_7) is used for accepting data, commands, and statuses, between the master CPU and the M5L8041A-XXXP. The data bus is controlled by the following 4 control signals, the relationship between the control signals and the data bus being shown in table 2.

A_0: Address input indicating data/command, bus buffer register and status, and register

$\overline{R}$: Read input

$\overline{W}$: Write input

$\overline{S}$: Chip select input

Table 2 Control Signals and Data Bus Relationships

$\overline{S}$	$\overline{R}$	$\overline{W}$	A_0	Data bus status	Data bus data
0	0	1	0	Read	Data
0	0	1	1	Read	Status
0	1	0	0	Write	Data
0	1	0	1	Write	Command (F1← 1)
1	X	X	X	High impedance	—

Fig. 6 shows the internal structure of the data bus. The 3 registers' (status register, output data bus buffer register, and input data/command bus buffer register) functions are described below.

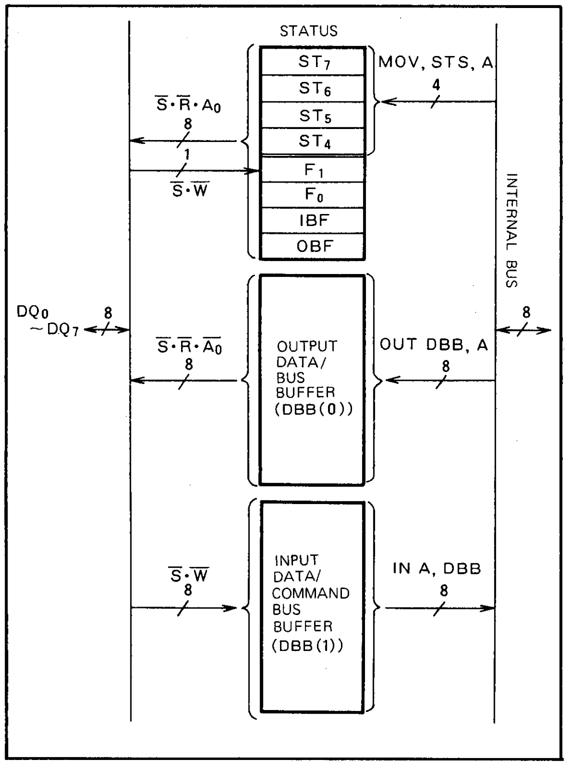


Fig. 6 Data bus internal structure

- Status Registers

The status registers consist of 8-bit registers, the upper 4 bits of which (ST_4~ST_7) being arbitrarily settable by software (MOVSTS, A instruction). The lower 4 bits (OBF, IBF, F_0, F_1) being set as follows.

OBF (Output Buffer Full)

The OBF flag is automatically set to 1 when the M5L8041A-XXXP internally executes an output instruction (OUT DDB, A), upon which the master CPU reads the contents of the output data bus buffer and clears this flag.

IBF (Input Buffer Full)

The IBF flag is set to 1 when the master CPU causes the writing of data or a command into the input data/command bus buffer, whereupon the input instruction (IN A DDB) is executed by the M5L8041A-XXXP and this flag is subsequently cleared.

F_0 (Flag 0)

The F_0 flag is set by the flag setting instructions (CPL F_0, CLR F_0) to inform the master CPU of the M5L8041A-XXXP internal status.

F_1 (Flag 1)

The F_1 flag is set when data or commands are input to the input data/command bus buffer by the master CPU to indicate the A_0 status.

The F_1 flag may also be set by the flag setting instructions (CPL F_1, CLR F_1).

- Output Data Bus Buffer (DBB (0)) Register

The output data bus buffer (DBB (0)) register is loaded with the contents of accumulator (A) by means of the OUT DBB, A instruction. Since the OBF flag is set at this time, the master CPU checks the status of this OBF flag to ascertain whether data has been transferred to the DBB (0) register.

- Input Data/Command Bus Buffer (DBB (I)) Register

When the master CPU has generated a write request ($\overline{W}$=0), the data on the data bus is transferred to the DBB (I) register. At this time, because the IBF flag is set, the status of the IBF flag is checked internally by the M5L8041A-XXXP to ascertain whether or not data or commands have been transferred.

CONDITIONAL JUMPS USING PINS T_0, T_1 and FLAGS IBF, OBF

Conditional jump instructions are used to alter program depending on internal and external conditions (states). Details of the jump instructions can be found in the section on machine instructions.

The input signal status of T_0, T_1, and flags IBF and OBF can be checked by the conditional jump instructions. These input pins, through conditional jump instructions such as JT0 and JNT0, can be used to control a program. Programs and processing time can be reduced by being able to test data in input pin rather than reading the data register and then testing it in the register.

Pin T_1 has other functions and uses that are not related to conditional jump instructions. The details of these other functions and uses can be found in the section on pin functions.

INTERRUPT

When an external interrupt is encountered by the CPU, the $\overline{S}$ and $\overline{W}$ pins are made low. The IBF flag is used to provide recognition of this interrupt condition.

Sampling of interrupt requests is done each machine cycle during the output of the SYNC signal. When an interrupt request is recognized, upon the completion of execution of the present instruction, a subroutine jump is made to address 3 of program memory. Just as would be the case in a normal subroutine jump, the program counter and program status word are saved on the program stack.

The program memory address 3 is normally used to store an unconditional jump to the address at which is stored the interrupt processing program.

The interrupt level is one, so the next interrupt cannot be accepted until the current interrupt processing has been completed. The RETR instruction terminates the interrupt processing. That is to say, the next interrupt can not be accepted until the RETR instruction is executed. The next interrupt can be accepted at the start of the second cycle of the RETR instruction (2-cycle instruction). Time/event counter overflow which causes an interrupt request also will not be accepted.

After the processing for an interrupt is completed control is returned to the main program. This is accomplished by execuiting RETR which restores the program counter and PSW automatical and checks $\overline{INT}$ and the time/event counter overflow for an interrupt request. If there is an interrupt request, the control will not be returned to the main program but will be transferred to the interrupt handling program.

An external interrupt has a higher priority than a timer interrupt. This means that, if an external and timer interrupt request are generated at the same time, the external interrupt has the priority and will be accepted first.

When a second level of external interrupt is required, the timer interrupt, if not being used, can provide this. The procedure for this is to first enable the timer interrupt, set the timer/event counter to FF_{16} and put the CPU in the event counter mode. After this has been done, if T_1 input is changed to low-level from high-level, an interrupt is generated in address 7.

Flag IBF can also be tested using a conditional jump instruction. For more details on this procedure, check the "Conditional Jumps Using pins T_0,T_1 and Flags IBF, OBF" section.

TIMER/EVENT COUNTER

The timer/event counter for the M5L8041A-XXXP is an 8-bit counter, that is used to measure time delays or count external events but not both The same counter is used to measure time delays or count external events by simply changing the input to the counter.

The counter can be preset by executing an MOV T, A instruction. The value of the counter can be read for checking by executing an MOV A, T instruction. Reset will stop the counting but the counter is not cleared, so counting can be resumed.

The largest number the counter can contain is FF_{16}. If it is incremented by 1 when it contains FF_{16}, the counter will be reset to 0, the overflow flag is set and a timer interrupt request is generated.

The conditional jump instruction JTF can be used to test the overflow flag. Care must be used in executing the JTF instruction because the overflow flag is cleared when executed (reset). When a timer interrupt is accepted, the control is transferred to address 7 of program memory.

When both a timer and external interrupt request are generated at the same time, the external interrupt is given priority and will be accepted first by automatically subroutine jumping to address 3 of program memory. The timer interrupt request is kept and will be processed when the external interrupt has been completed and a RETR is executed. A latched timer interrupt request is cancelled when a timer interrupt request is generated.

The START CNT instruction is used to change the counter to an event counter. Then pin T_1 signal becomes the input to the event counter and an event is counted each full cycle (low-high-low one event). The maximum rate that can be counted is one time in 3 machine cycles (7.5μs when using 6MHz crystal). The high-level at T_1 must be maintained at least 1/5 of the cycle time (500ns when using 6MHz crystal).

The START T instruction is used to change the counter to a timer. The internal clock signal becomes the input to the timer. The internal clock is 1/32 of 400kHz (when using 6MHz crystal) or 12.5kHz. The timer is therefore counted up every 80μs. Fig. 7 shows the timer/event counter.

The counter can be preset by executing an MOV T, A instruction. The timer can be used to measure 80μs~20ms in multiples of 80μs. When it is necessary to measure over 20ms (maximum count 256x80μs) of delay time the number of overflows, one every 20ms, can be counted by the program. To measure times of less than 80μs; external clock pulses can be input through T_1 while the counter is in the event counter mode. Every third (or more) ALE signal can be used instead of an external clock pulses.

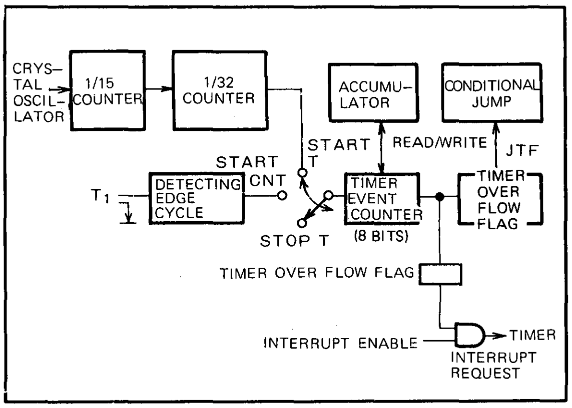


Fig. 7 Timer/event counter

CYCLE TIMING

The output of the state counter is 1/3 the input frequency from the oscillator. A CLK signal is generated every 500ns (one state cycle) which is used for the demarcation of each machine state.

Fig 9 shows the relationship between clock and generated cycles.

One machine cycle contains 5 states with a CLK signal for demarcation of each state. The M5L8014A-XXXP instructions are executed in one machine cycle or two machine cycles. An instruction cycle can be one or two machine cycles as shown in Fig. 10.

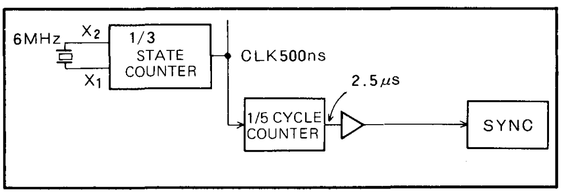


Fig. 8 Clocking cycle generation

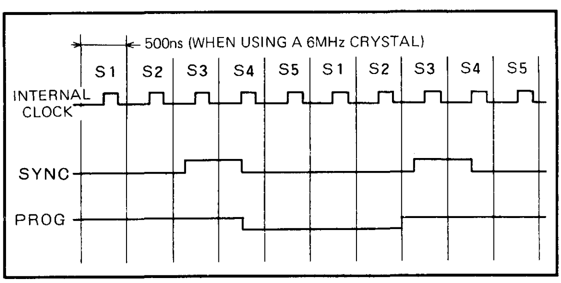


Fig. 9 Clock and generated cycle signals

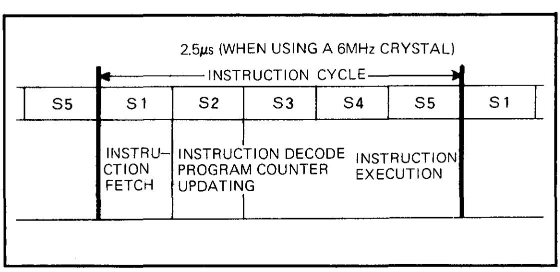


Fig. 10 Instruction execution timing

$\overline{\text{RESET}}$

The reset pin is for resetting the CPU. A Schmitt trigger circuit along with a pull-up register are connected to it on the chip. A reset can easily be generated by attaching a 1μF as capacitor as shown in Fig. 11. An external reset pulse applied at $\overline{\text{RESET}}$ must remain at low-level for at least 10ms after power has been turned on and reached its normal level.

The reset function causes the following initialization within the CPU.

1. Program counter is reset to 0.
2. Stack pointer is reset to 0.
3. Memory bank is reset to 0.
4. Ports 1 and 2 are reset to input mode.
5. External and timer interrupts are reset to disable state.
6. Timer is stopped.
7. Timer overflow flag is cleared.
8. Flags F_0 and F_1 are cleared.

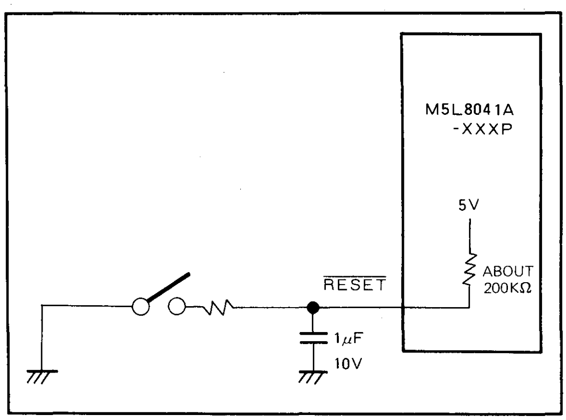


Fig. 11 Example of a reset circuit

SINGLE-STEP OPERATION

The pin $\overline{SS}$ on the M5L8041A-XXXP is provided to facilitate single-step operation. In single-step operation, the CPU stops after the execution of each instruction is completed and the memory address (12 bits) of the next instruction to be fetched is output through the data bus (8 bits) plus the low-order 4 bits of port 2. The user can use this to trace the flow of this program instruction by instruction and will find this an aid in program debugging. Single-step operation is controlled through $\overline{SS}$ and SYNC as shown in Fig. 12.

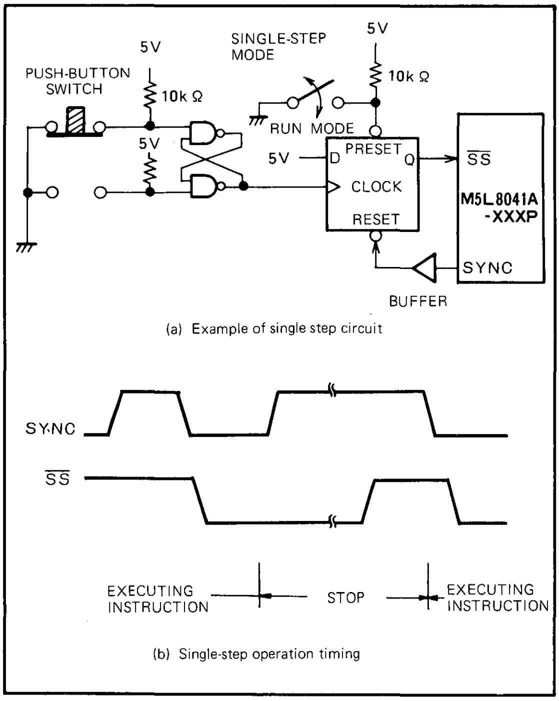


Fig. 12. Single-step operation circuit and timing

A type D flip-flop with preset and reset pins, as shown in Fig. 12, is used to generate the signal for $\overline{SS}$. When the preset pin goes to low-level, $\overline{SS}$ goes to high-level, which puts the CPU in RUN mode. When the preset pin is grounded it goes to high-level. Then $\overline{SS}$ goes to low-level. When $\overline{SS}$ goes to low-level, the CPU stops. Then when the push-button switch is pushed, a pulse is sent to the clock pin of the type D flip-flop which turns $\overline{SS}$ to high-level. When $\overline{SS}$ goes to high-level the CPU fetches the next instruction and begins to execute it, but then an ALE signal is sent to the reset pin of the type D flip-flop which turns $\overline{SS}$ to low-level. The CPU again stops as soon as execution of the current instruction is completed. The CPU is in single-step operation as shown in Fig. 13.

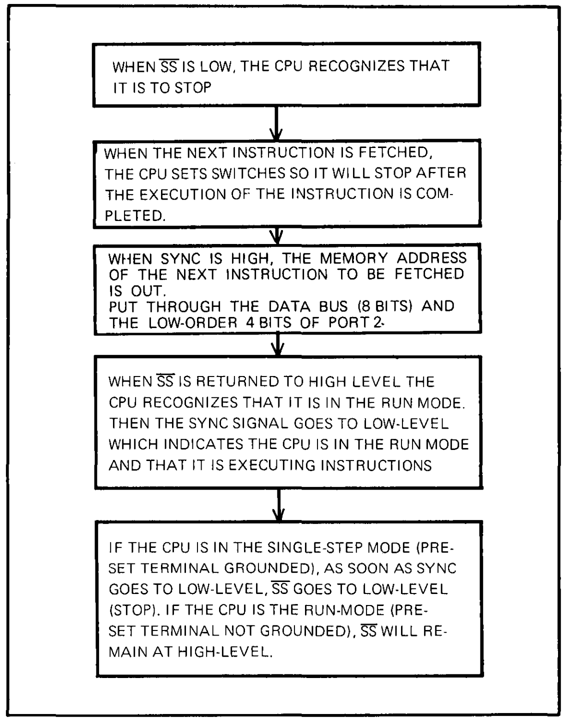


Fig. 13 CPU operation in single-step mode

Central Processing Unit (CPU)

Central Processing Unit (CPU) is composed of an 8-bit parallel arithmetic unit, accumulator, flag flip-flop and instruction decoder. The 8-bit parallel arithmetic unit has circuitry to perform the four basic arithmetic operations (plus, minus, multiply and divide) as well as logical operations such as AND and OR. The flag flip-flop is used to indicate status such as carry and zero. The accumulator contains one of the operations and the result is usually retained in the accumulator. (The flag flip- frops hold the carry and borrow states for execution of all processing instructions.)

DMA Control

In addition to use as a normal input/output port, the M5L8041A-XXXP port P_{26} and P_{27} can be used to provide control signals for handshake control of DMA operations. Immediately after resetting, P_{26} and P_{27} can be used as a normal port (Fig. 14).

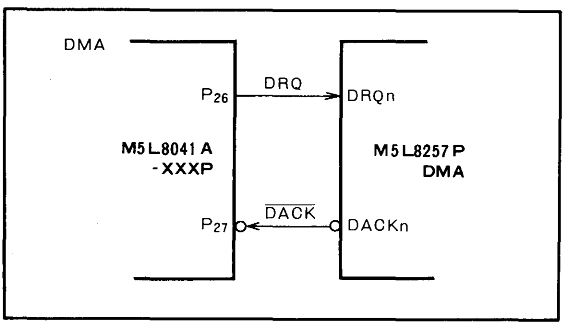


Fig. 14 DMA Control

When the EN DMA instruction is executed, P_{26} becomes the DRQ (DMA request) output. Subsequently, when P_{26} is set to 1, DRQ becomes 1 and DMA data transfer is requested.

DRQ is returned to 0 when $\overline{DACK}\cdot\overline{R}$, $\overline{DACK}\cdot\overline{W}$, or EN DMA is executed.

In addition, when EN DMA is executed, P_{27} becomes the $\overline{DACK}$ (DMA acknowledgment) input. This $\overline{DACK}$ input serves as the chip select input during DMA transfer operations. Therefore, the normal chip select $\overline{S}$ status has no meaning during DMA transfers.

Interrupts of the Master CPU

In addition to use as a normal input/output port, the M5L8041A-XXXP P_{24} and P_{25} pins may be used as the IBF (Input Buffer Full) flag and OBF (Output Buffer Full) flag outputs. Immediately after resetting, P_{24} and P_{25} are usable as normal input ports.

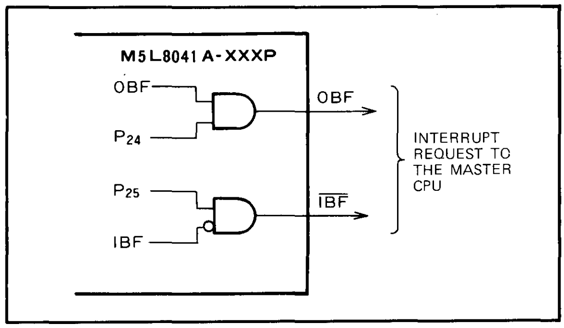


Fig. 15 Interrupt requests to the master CPU

When the EN FLAGS instruction is executed, P_{24} becomes the OBF pin and P_{25} the $\overline{IBF}$ pin. At this time, in order to enable OBF flag and IBF flag status outputs for pin 24 and pin 25, it is necessary to set pin 24 and pin 25 to 1. With P_{24} and P_{25} set to 0, these flag statuses are not output. The OBF flag output indicates that data is being output to the M5L8041A-XXXP output data bus buffer register while the IBF flag output indicates that data can be accepted by the input data/command bus buffer register.

8

MACHINE INSTRUCTIONS

Type \ Item	Mnemonic	Instruction code $D_7\ D_6\ D_5\ D_4$	$D_3\ D_2\ D_1\ D_0$	Hexa-decimal	Bytes	Cycles	Function	Effected carry C	AC	Note	Description
Transfer	MOV A, #n	0 0 1 0 $n_7n_6n_5n_4$	0 0 1 1 $n_3n_2n_1n_0$	23 n	2	2	(A)←n				Transfers data n to register A.
Transfer	MOV A, PSW	1 1 0 0	0 1 1 1	C7	1	1	(A)←(PSW)				Transfers the contents of the program status word to register A.
Transfer	MOV A, Rr	1 1 1 1	1 $r_2 r_1 r_0$	F8 + r	1	1	(A)←(Rr) r=0～7				Transfers the contents of register R_r to register A.
Transfer	MOV A, @Rr	1 1 1 1	0 0 0 r_0	F0 + r	1	1	(A)←(M(Rr)) r=0～1				Transfers the contents of memory location, of the current page, whose address is in register R_r to register A.
Transfer	MOV PSW, A	1 1 0 1	0 1 1 1	D7	1	1	(PSW)←(A) (C)←(A_7), (AC)←(A_6)	○	○		Transfers the contents of register A to the program status word.
Transfer	MOV STS, A	1 0 0 1	0 0 0 0	90	1	1	(STS)←(A) (ST_4～ST_7)←(A_4～A_7)			4	Transfers the contents of register A to the system status register.
Transfer	MOV Rr, A	1 0 1 0	1 $r_2 r_1 r_0$	A8 + r	1	1	(Rr)←(A) r=0～7				Transfers the contents of register A to register R_r.
Transfer	MOV Rr, #n	1 0 1 1 $n_7n_6n_5n_4$	1 $r_2 r_1 r_0$ $n_3n_2n_1n_0$	B8 + r n	2	2	(Rr)←n r=0～7				Transfers data n to register R_r.
Transfer	MOV @Rr, A	1 0 1 0	0 0 0 r_0	A0 + r	1	1	(M(Rr))←(A) r=0～1				Transfers the contents of register A to memory location, of the current page, whose address is in register R_r.
Transfer	MOV @Rr, #n	1 0 1 1 $n_7n_6n_5n_4$	0 0 0 r_0 $n_3n_2n_1n_0$	B0 + r n	2	2	(M(Rr))←n r=0～1				Transfers data n to memory location, of the current page, whose address is in register R_r.
Transfer	MOVP A, @A	1 0 1 0	0 0 1 1	A3	1	2	(A)←(M(A))				Transfers the data of memory location, of the current page, whose address is in register A to register A.
Transfer	MOVP3 A, @A	1 1 1 0	0 0 1 1	E3	1	2	(A)←(M(page 3, A))				Transfers the data of memory location, of page 3, whose address is in register A to register A.
Transfer	XCH A, Rr	0 0 1 0	1 $r_2 r_1 r_0$	28 + r	1	1	(A)←→(Rr) r=0～7				Exchanges the contents of register R_r with the contents of register A.
Transfer	XCH A, @Rr	0 0 1 0	0 0 0 r_0	20 + r	1	1	(A)←→(M(Rr)) r=0～1				Exchanges the contents of memory location, of the current page, whose address is in register R_r with the contents of register A.
Transfer	XCHD A, @Rr	0 0 1 1	0 0 0 r_0	30 + r	1	1	(A_0～A_3)←→(M(Rr_0～Rr_3)) r=0～1				Exchanges the contents of the low-order 4-bits of register A with the low-order 4-bits of memory location, of the current page, whose address is in register R_r.
Arithmetic	ADD A, #n	0 0 0 0 $n_7n_6n_5n_4$	0 0 1 1 $n_3n_2n_1n_0$	03 n	2	2	(A)←(A)+n	○	○	1	Adds data n to the contents of register A and sets the carry flags to 1 if there is an overflow otherwise resets the carry flags to 0 The result is stored in register A.
Arithmetic	ADD A, Rr	0 1 1 0	1 $r_2 r_1 r_0$	68 + r	1	1	(A)←(A)+(Rr) r=0～7	○	○	1	Adds the contents of register R_r to the contents of register A and sets the carry flags to 1 if there is an overflow otherwise resets the carry flags to 0. The result is stored in register A.
Arithmetic	ADD A, @Rr	0 1 1 0	0 0 0 r_0	60 + r	1	1	(A)←(A)+(M(Rr)) r=0～1	○	○	1	Adds the contents of register A and the contents of memory location, of the current page, whose address is in register A and sets the carry flags to 1 if there is an overflow otherwise resets the carry flags to 0. The result is stored in register A.
Arithmetic	ADDC A, #n	0 0 0 1 $n_7n_6n_5n_4$	0 0 1 1 $n_3n_2n_1n_0$	13 n	2	2	(A)←(A)+n+(C)	○	○	1	Adds the carry and data n to the contents of register A and sets the carry flags to 1 if there is an overflow otherwise resets the carry flags to 0. The result is stored in register A.
Arithmetic	ADDC A, Rr	0 1 1 1	1 $r_2 r_1 r_0$	78 + r	1	1	(A)←(A)+(Rr)+(C) r=0～7	○	○	1	Adds the carry and the contents of register R_r to the contents of register A and set the carry flags to 1 if there is an overflow otherwise resets the carry flags to. 0. The result is stored in register A.
Arithmetic	ADDC A, @Rr	0 1 1 1	0 0 0 r_0	70 + r	1	1	(A)←(A)+(M(Rr))+(C) r=0～1	○	○	1	Adds the carry and the contents of memory location, of the current page, whose address is in register R_r to the contents of register A and sets the carry flags to 1if there is an overflow otherwise resets the carry flags to 0. The result is stored in register A.

Type \ Item	Mnemonic	Instruction code $D_7D_6D_5D_4$	$D_3D_2D_1D_0$	Hexa-decimal	Bytes	Cycles	Function	Effected carry C	AC	Note	Description	
Subroutine call	**CALL m**	$m_{10}m_9m_8$ 1 $m_7m_6m_5m_4$	0 1 0 0 $m_3m_2m_1m_0$	1 4 + m_8 m_{10} m	2	2	$((SP)) \leftarrow (PC)\ (PSW_4 \sim PSW_7)$ $(SP) \leftarrow (SP)+1$ $(PC_{0\sim10}) \leftarrow m$ $(PC_{11}) \leftarrow MBF$				Calls subroutine from address m. The program counter and the 4 high-order bits of the PSW are stored in the address indicated by the stack pointer (SP). The SP is incremented by 1 and m is transferred to $PC_0 \sim PC_{10}$ and the MBF is transferred to PC_{11}.	
	RET	1 0 0 0	0 0 1 1	8 3	1	2	$(SP) \leftarrow (SP)-1$ $(PC) \leftarrow ((SP))$				The SP is decremented by 1 The program counter is restored to the saved setting in the stack indicated by the stack pointer. The PSW is not changed and interrupt disabled is maintained.	
	RETR	1 0 0 1	0 0 1 1	9 3	1	2	$(SP) \leftarrow (SP)-1$ $(PC)\ (PSW_4 \sim PSW_7) \leftarrow ((SP))$				The SP is decremented by 1. The program counter and the 4 high-order bits of the PSW are restored with the saved data in the stack indicated by the stack pointer. The interrupt becomes enabled after the execution is completed.	
Input/Output	**IN A, Pp**	0 0 0 0	1 0 p_1p_0	0 8 + p	1	2	$(A) \leftarrow (Pp)$ $p=1 \sim 2$				Loads the contents of P_p to register A.	
	OUTL Pp, A	0 0 1 1	1 0 p_1p_0	3 8 + p	1	2	$(Pp) \leftarrow (A)$ $p=1 \sim 2$				Output latches the contents of register A to P_p	
	ANL Pp, #n	1 0 0 1 $n_7n_6n_5n_4$	1 0 p_1p_0 $n_3n_2n_1n_0$	9 8 + p n	2	2	$(Pp) \leftarrow (Pp) \wedge n$ $p=1 \sim 2$				Logical AND_s the contents of P_p and data n. Outputs the result to P_p	
	ORL Pp, #n	1 0 0 0 $n_7n_6n_5n_4$	1 0 p_1p_0 $n_3n_2n_1n_0$	8 8 + p n	2	2	$(Pp) \leftarrow (Pp) \vee n$ $p=1 \sim 2$				Logical ORs the contents of P_p and data n Outputs the result to P_p.	
	IN A, DBB	0 0 1 0	0 0 1 0	2 2	1	1	$(A) \leftarrow (DBB)$				Enters the contents of data bus buffer (DBB) in register A.	
	OUT DBB, A	0 0 0 0	0 0 1 0	0 2	1	1	$(DBB) \leftarrow (A)$				Outputs the contents of register A to data bus buffer (DBB) OBF is set.	
	MOVD A, Pp	0 0 0 0	1 1 p_1p_0	0 C + p_1p_0	1	2	$(A_0 \sim A_3) \leftarrow (Pp_0 \sim Pp_3)$ $(A_4 \sim A_7) \leftarrow 0\quad p=4 \sim 7$				Inputs the contents of P_P of 8243 to the low-order 4 bits of register A and inputs 0 to the high-order 4 bits of register A.	P_p's used for multi-plying 8243 ports are $P_4 \sim P_7$. Correspon-dence to p_2, p_1 is shown below. $P4 \cdots p_1p_2=00$ $P5 \cdots p_1p_2=01$ $P6 \cdots p_1p_2=10$ $P7 \cdots p_1p_2=11$
	MOVD Pp, A	0 0 1 1	1 1 p_1p_0	3 C + p_1p_0	1	2	$(Pp_0 \sim Pp_3) \leftarrow (A_0 \sim A_3)$ $p=4 \sim 7$				Outputs the low-order 4 bits of register A to P_P.	
	ANLD Pp, A	1 0 0 1	1 1 p_1p_0	9 C + p_1p_0	1	2	$(Pp_0 \sim Pp_3) \leftarrow (Pp_0 \sim Pp_3) \wedge (A_0 \sim A_3)$ $p=4 \sim 7$				Logical AND_S the 4 low-order bits of register A and the contents of P_P. P_P contains the result.	
	ORLD Pp, A	1 0 0 0	1 1 p_1p_0	8 C + p_1p_0	1	2	$(Pp_0 \sim Pp_3) \leftarrow (Pp_0 \sim Pp_3) \vee (A_0 \sim A_3)$ $p=4 \sim 7$				Logical OR_S the 4 low-order bits of register A and the contents of P_P. P_P contains the result.	

Type \ Item	Mnemonic	Instruction code $D_7D_6D_5D_4$ $D_3D_2D_1D_0$	Hexa-decimal	Bytes	Cycles	Function	Effected carry C	AC	Note	Description
Arithmetic	**ANL A, #n**	0 1 0 1 0 0 1 1 $n_7n_6n_5n_4$ $n_3n_2n_1n_0$	5 3 n	2	2	$(A) \leftarrow (A) \wedge n$				The logical product of the contents of register A and data n, is stored in register A.
	ANL A, Rr	0 1 0 1 1 r_2 r_1 r_0	5 8 + r	1	1	$(A) \leftarrow (A) \wedge (Rr)$ $r=0\sim7$				The logical product of the contents of register A and the contents of register R_r, is stored in register A.
	ANL A, @Rr	0 1 0 1 0 0 0 r_0	5 0 + r	1	1	$(A) \leftarrow (A) \wedge (M(Rr))$ $r=0\sim1$				The logical product of the contents of register A and the contents of memory location, of the current page, whose address is in register R_r, is stored in register A.
	ORL A, #n	0 1 0 0 0 0 1 1 $n_7n_6n_5n_4$ $n_3n_2n_1n_0$	4 3 n	2	2	$(A) \leftarrow (A) \vee n$				The logical sum of the contents of register A and data n, is stored in register A.
	ORL A, Rr	0 1 0 0 1 r_2 r_1 r_0	4 8 + r	1	1	$(A) \leftarrow (A) \vee (Rr)$ $r=0\sim7$				The logical sum of the contents of register A and the contents of register R_r is stored in register A.
	ORL A, @Rr	0 1 0 0 0 0 0 r_0	4 0 + r	1	1	$(A) \leftarrow (A) \vee (M(Rr))$ $r=0\sim1$				The logical sum of the contents of register A and the contents of memory location, of the current page, whose address is in register R_r, is stored in register A.
	XRL A, #n	1 1 0 1 0 0 1 1 $n_7n_6n_5n_4$ $n_3n_2n_1n_0$	D 3 n	2	2	$(A) \leftarrow (A) \forall n$				The exclusive OR of the contents of register A and data n, is stored in register A.
	XRL A, Rr	1 1 0 1 1 r_2 r_1 r_0	D 8 + r	1	1	$(A) \leftarrow (A) \forall (Rr)$ $r=1\sim7$				The exclusive OR of the contents of register A and the contents of register R_r is stored in register A.
	XRL A, @Rr	1 1 0 1 0 0 0 r_0	D 0 + r	1	1	$(A) \leftarrow (A) \forall (M(Rr))$ $r=0\sim1$				The exclusive OR of the contents of register A and the contents of memory location, of the current page, whose address is in register R_r, is stored in register A.
	INC A	0 0 0 1 0 1 1 1	1 7	1	1	$(A) \leftarrow (A)+1$				Increments the contents of register A by 1. The result is stored in register A, and the carries are unchanged.
	DEC A	0 0 0 0 0 1 1 1	0 7	1	1	$(A) \leftarrow (A)-1$				Decrements the contents of register A by 1. The result is stored in register A, and the carries are unchanged.
	CLR A	0 0 1 0 0 1 1 1	2 7	1	1	$(A) \leftarrow 0$				Clears the contents of register A, resets to 0.
	CPL A	0 0 1 1 0 1 1 1	3 7	1	1	$(A) \leftarrow (\bar{A})$				Forms 1's complement of register A, and stores it in register A.
	DA A	0 1 0 1 0 1 1 1	5 7	1	1	$(A) \leftarrow (A)$	○	○	1	The contents of register A is converted to binary coded decimal notion, and it is stored in register A. If the contents of register A are more than 99 the carry flags are set to 1 other wise they are reset to 0.
	SWAP A	0 1 0 0 0 1 1 1	4 7	1	1	$(A_4\sim A_7) \longleftrightarrow (A_0\sim A_3)$				Exchanges the contents of bits 0~3 of register A with the contents of bits 4~7 of register A.
	RL A	1 1 1 0 0 1 1 1	E 7	1	1	$(A_{n+1}) \leftarrow (A_n)$ $(A_0) \leftarrow (A_7)$ $n=0\sim6$				Shifts the contents of register A left one bit. A_7 the MSB is rotated to A_0 the LSB.
	RLC A	1 1 1 1 0 1 1 1	F 7	1	1	$(A_{n+1}) \leftarrow (A_n)$ $(A_0) \leftarrow (C)$ $(C) \leftarrow (A_7)$ $n=0\sim6$	○			Shifts the contents of register A left one bit. A_7 the MSB is shifted to the carry flag and the carry flag is shifted to A_0 the LSB.
	RR A	0 1 1 1 0 1 1 1	7 7	1	1	$(A_n) \leftarrow (A_{n+1})$ $(A_7) \leftarrow (A_0)$ $n=0\sim6$				Shifts the contents of register A right one bit. A_0 the LSB is rotated to A_7 the MSB.
	RRC A	0 1 1 0 0 1 1 1	6 7	1	1	$(A_n) \leftarrow (A_{n+1})$ $(A_7) \leftarrow (C)$ $(C) \leftarrow (A_0)$ $n=0\sim6$	○			Shifts the contents of register A right one bit. A_0 the LSB is shifted to the carry flag and the carry flag is shifted to A_7 the MSB.
Register arithmetic	**INC Rr**	0 0 0 1 1 r_2 r_1 r_0	1 8 + r	1	1	$(Rr) \leftarrow (Rr)+1$ $r=0\sim7$				Increments the contents of register R_r by 1. The result is stored in register R_r and the carries are unchanged.
	INC @Rr	0 0 0 1 0 0 0 r_0	1 0 + r	1	1	$(M(Rr)) \leftarrow (M(Rr))+1$ $r=0\sim1$				Increments the contents of the memory location, of the current page, whose address is in register R_r by 1. Register R_r uses bit 0~5.
	DEC Rr	1 1 0 0 1 r_2 r_1 r_0	C 8 + r	1	1	$(Rr) \leftarrow (Rr)-1$ $r=0\sim7$				Decrements the contents of register R_r by 1. The result is stored in register R_r and the carries are unchanged.

Type	Mnemonic	Instruction code $D_7D_6D_5D_4$	$D_3D_2D_1D_0$	Hexa-decimal	Bytes	Cycles	Function	Effected carry C	AC	Note	Description
Jump	JBb m	$b_7b_6b_5$ 1 $m_7m_6m_5m_4$	0 0 1 0 $m_3m_2m_1m_0$	1 2 + b m	2	2	$(A_b)=1$ then $(PC_0\sim PC_7)\leftarrow m$ $(A_b)=0$ then $(PC)\leftarrow(PC)+2$ $b_7b_6b_5=0\sim7$				Jumps to address m of the current page when bit b of register A is 1. Executes the next instruction when bit b of register A is 0.
	JNIBF m	1 1 0 1 $m_7m_6m_5m_4$	0 1 1 0 $m_3m_2m_1m_0$	D 6	2	2	$(IBF)=0$ then $(PC_0\sim PC_7)\leftarrow m$				Jumps to address m off the curent page when IBF is 0, otherwise the next instruction is executed.
	JOBF m	1 0 0 0 $m_7m_6m_5m_4$	0 1 1 0 $m_3m_2m_1m_0$	8 6 m	2	2	$(OBF)=1$ then $(PC_0\sim PC_7)\leftarrow m$				Jumps to address m of the current page when OBF is 0, otherwise the next instruction is exected.
	JTF m	0 0 0 1 $m_7m_6m_5m_4$	0 1 1 0 $m_3m_2m_1m_0$	1 6 m	2	2	$(TF)=1$ then $(PC_0\sim PC_7)\leftarrow m$ $(TF)=0$ then $(PC)\leftarrow(PC)+2$				Jumps to address m of the curent page when the overflow flag of the timer is 1 otherwise the next instruction is executed. Flag is cleared after executing
	JMP m	$m_{10}m_9m_8$ 0 $m_7m_6m_5m_4$	0 1 0 0 $m_3m_2m_1m_0$	0 4 + $m_{8\sim10}$ m	2	2	$(PC_8\sim PC_{10})\leftarrow m_8\sim m_{10}$ $(PC_0\sim PC_7)\leftarrow m_0\sim m_7$ $(PC_{11})\leftarrow(MBF)$				Jumps to address m on page $m_{10}m_9m_8$ in the memory bank indicated by MBF
	JMPP @A	1 0 1 1	0 0 1 1	B 3	1	2	$(PC_0\sim PC_7)\leftarrow(M(A))$				Jumps to the memory location, of the current page, whose address is in register A. But when the instruction executed was in address 255, jumps to next page.
	DJNZ Rr, m	1 1 1 0 $m_7m_6m_5m_4$	1 $r_2r_1r_0$ $m_3m_2m_1m_0$	E 8 + r m	2	2	$(Rr)\leftarrow(Rr)-1$ $r=0\sim7$ $(Rr)\neq0$ then $(PC_0\sim PC_7)\leftarrow m$ $(Rr)=0$ then $(PC)\leftarrow(PC)+2$				Decrements the contents of register R_r by 1. Jumps to address m of the current page when the result is not 0, otherwise the next instruction is executed.
	JC m	1 1 1 1 $m_7m_6m_5m_4$	0 1 1 0 $m_3m_2m_1m_0$	F 6 m	2	2	$(C)=1$ then $(PC_0\sim PC_7)\leftarrow m$ $(C)=0$ then $(PC)\leftarrow(PC)+2$				Jumps to address m of the curent page if the carry flag C is 1, otherwise the next instruction isexecuted.
	JNC m	1 1 1 0 $m_7m_6m_5m_4$	0 1 1 0 $m_3m_2m_1m_0$	E 6 m	2	2	$(C)=0$ then $(PC_0\sim PC_7)\leftarrow m$ $(C)=1$ then $(PC)\leftarrow(PC)+2$				Jumps to address m of the current page if the carry flag C is 0. otherwise the next instruction is executed.
	JZ m	1 1 0 0 $m_7m_6m_5m_4$	0 1 1 0 $m_3m_2m_1m_0$	C 6 m	2	2	$(A)=0$ then $(PC_0\sim PC_7)\leftarrow m$ $(A)\neq0$ then $(PC)\leftarrow(PC)+2$				Jumps to address m of the current page when the contents of register A are 0, otherwise the next instruction is executed.
	JNZ m	1 0 0 1 $m_7m_6m_5m_4$	0 1 1 0 $m_3m_2m_1m_0$	9 6 m	2	2	$(A)\neq0$ then $(PC_0\sim PC_7)\leftarrow m$ $(A)=0$ then $(PC)\leftarrow(PC)+2$				Jumps to address m of the current page when the contents of register A are not 0, otherwise the next instruction is executed.
	JT0 m	0 0 1 1 $m_7m_6m_5m_4$	0 1 1 0 $m_3m_2m_1m_0$	3 6 m	2	2	$(T_0)=1$ then $(PC_0\sim PC_7)\leftarrow m$ $(T_0)=0$ then $(PC)\leftarrow(PC)+2$				Jumps to address m of the current page when flag T_0 is 1 otherwise the next instruction is excecuted.
	JNT0 m	0 0 1 0 $m_7m_6m_5m_4$	0 1 1 0 $m_3m_2m_1m_0$	2 6 m	2	2	$(T_0)=0$ then $(PC_0\sim PC_7)\leftarrow m$ $(T_0)=1$ then $(PC)\leftarrow(PC)+2$				Jumps to address m of the current page when flag T_0 is 0, otherwise the next instruction is executed.
	JT1 m	0 1 0 1 $m_7m_6m_5m_4$	0 1 1 0 $m_3m_2m_1m_0$	5 6 m	2	2	$(T_1)=1$ then $(PC_0\sim PC_7)\leftarrow m$ $(T_1)=0$ then $(PC)\leftarrow(PC)+2$				Jumps to address m of the current page when flag T_1 is 1. otherwise the next instruction is executed.
	JNT1 m	0 1 0 0 $m_7m_6m_5m_4$	0 1 1 0 $m_3m_2m_1m_0$	4 6 m	2	2	$(T_1)=0$ then $(PC_0\sim PC_7)\leftarrow m$ $(T_1)=1$ then $(PC)\leftarrow(PC)+2$				Jumps to address m of the current page when flag T_1 is 0, otherwise the next instruction is executed.
	JF0 m	1 0 1 1 $m_7m_6m_5m_4$	0 1 1 0 $m_3m_2m_1m_0$	B 6 m	2	2	$(F_0)=1$ then $(PC_0\sim PC_7)\leftarrow m$ $(F_0)=0$ then $(PC)\leftarrow(PC)+2$				Jumps to address m of the current page when flag F_0 is 1.
	JF1 m	0 1 1 1 $m_7m_6m_5m_4$	0 1 1 0 $m_3m_2m_1m_0$	7 6 m	2	2	$(F_1)=1$ then $(PC_0\sim PC_7)\leftarrow m$ $(F_1)=0$ then $(PC)\leftarrow(PC)+2$				Jumps to address m of the current page when flag F_1 is 1.
Flag control	CLR C	1 0 0 1	0 1 1 1	9 7	1	1	$(C)\leftarrow0$	○			Clears the carry flag C, resets it to 0. AC is not affected.
	CPL C	1 0 1 0	0 1 1 1	A 7	1	1	$(C)\leftarrow(\bar{C})$	○			Complements the carry flag C. AC is not affected.
	CLR F_0	1 0 0 0	0 1 0 1	8 5	1	1	$(F_0)\leftarrow0$				Clears the flag F_0, resets it to 0.
	CPL F_0	1 0 0 1	0 1 0 1	9 5	1	1	$(F_0)\leftarrow(\bar{F_0})$				Complements the flag F_0.
	CLR F_1	1 0 1 0	0 1 0 1	A 5	1	1	$(F_1)\leftarrow0$				Clears flag F_1 resets it to 0.
	CPL F_1	1 0 1 1	0 1 0 1	B 5	1	1	$(F_1)\leftarrow(\bar{F_1})$				Complements the flag F_1

8

Type	Mnemonic	Instruction code D_7 D_6 D_5 D_4	D_3 D_2 D_1 D_0	Hexa-decimal	Bytes	Cycles	Function	Effected carry C	AC	Note	Description
Control	EN I	0 0 0 0	0 1 0 1	0 5	1	1	(INTF) ← 1				Enables outside interrupt.
	DIS I	0 0 0 1	0 1 0 1	1 5	1	1	(INTF) ← 0				Disables outside interrupt.
	SEL RB_0	1 1 0 0	0 1 0 1	C 5	1	1	(BS) ← 0				Selects working register bank 0.
	SEL RB_1	1 1 0 1	0 1 0 1	D 5	1	1	(BS) ← 1				Selects working register bank 1.
	EN DMA	1 1 1 0	0 1 0 1	E 5	1	1					Enables DMA hand shake lines.
	EN FLAGS	1 1 1 1	0 1 0 1	F 5	1	1	$(P2_4)$ ← (OBF) $(P2_5)$ ← (IBF)				Enables interrupts from master.
Timer/event counter control	MOV A, T	0 1 0 0	0 0 1 0	4 2	1	1	(A) ← (T)				Transfers the contents of timer/event counter to register A.
	MOV T, A	0 1 1 0	0 0 1 0	6 2	1	1	(T) ← (A)				Transfers the contents of register A to timer/event counter.
	STRT T	0 1 0 1	0 1 0 1	5 5	1	1					Starts timer operation of time/event counterm. Minimum count cycle is 80μs.
	STRT CNT	0 1 0 0	0 1 0 1	4 5	1	1					Starts operation as event counter of time/event counter. Counts up when terminated T_1 changes to input high-level for input low-level. Minimum count cycle is 7.5μs.
	STOP TCNT	0 1 1 0	0 1 0 1	6 5	1	1					Stops operation of timer or event counter.
	EN TCNTI	0 0 1 0	0 1 0 1	2 5	1	1	(TCNTF) ← 1				Enables interrupt of timer/event counter.
	DIS TCNTI	0 0 1 1	0 1 0 1	3 5	1	1	(TCNTF) ← 0				Disables interrupt of timer/event counter Resets interrupt flip-flop of CPU which is set during the CPU stands-by Timer over flow flag isn't affected.
Misc.	NOP	0 0 0 0	0 0 0 0	0 0	1	1					No operation. Execution time is 1 cycle.

Note 1: Executing an instruction may produce a carry (overflow or underflow). The carry may be disregarded (lost) or it may be transferred to C/AC (saved). The saving of a carry is not shown in the function equations, but is instead shown in the carry columns C and AC. The detail affection of carries for instructions ADD ADDC and DA is as follows:

(C) ← 1 at overflow of the accumulator is produced.
(C) ← 0 at no overflow of the accumulator is produced.
(AC) ← 1 at overflow of the bit 3 of the accumulator.
(AC) ← 0 at no overflow.

2: The contents of ST_4~ST_7 is read when the host computer reads the status of M5L8041A-XXXP.

Symbol	Details	Symbol	Details
A	8-bit register (accumulator)	PC	Program counter
$A_0 \sim A_3$	The low-order 4 bits of the register A	$PC_0 \sim PC_7$	The low-order 8 bits of the program counter
$A_4 \sim A_7$	The high-order 4 bits of the register A	$PC_8 \sim PC_{10}$	The high-order 3 bits of the program counter
$A_0 \sim A_n, A_{n+1}$	The bits of the register A	PSW	Program status word
b	The value of the bits 5~7 of the first byte machine code		
$b_7b_6b_5$	The bits 5~7 of the first byte machine code	Rr	Register designator
BS	Register bank select	r	Register number
BUS	Corresponds to the port 0 (bus I/O port)	r_0	The value of bit 0 of the machine code
AC	Auxiliary carry flag	$r_2r_1r_0$	The value of bits 0~2 of the machine code
C	Carry flag	$s_2s_1s_0$	The value of bits 0~2 of the stack pointer
DBB	Data bus buffer	SP	Stack pointer
		ST_4ST_7	Bits 4~7 of the status register
F_0	Flag 0	STS	System status
F_1	Flag 1	T	Timer/event counter
INTF	Interrupt flag	T_0	Test pin 0
IBF	Input buffer full flag	T_1	Test pin 1
m	The value of the 11-bit address	TCNTF	Timer/event counter overflow interrupt flag
$m_7m_6m_5m_4m_3m_2m_1m_0$	The second byte (low-order 8 bits) machine code of the 11-bit address	TF	Timer flag
$m_{10}m_9\ m_8$	The bits 5~7 of the first byte (high-order 3 bits) machine code of the 11-bit address		
(M (A))	The content of the memory location addressed by the register A	#	Symbol to indicate the immediate data
(M (Rr))	The content of the memory location addressed by the register R_r	@	Symbol to indicate the content of the memory location addressed by the register
(Mx(Rr))	The content of the external memory location addressed by the register R_r		
MBF	Memory bank flag	←	Shows direction of data flow
n	The value of the immediate data	↔	Exchanges the contents of data
$n_7n_6n_5n_4n_3n_2n_1n_0$	The immediate data of the second byte machine code	( )	Contents of register memory location or flag
OBF	Output buffer full flag	∧	Logical AND
		∨	Inclusive OR
p	Port number	∀	Exclusive OR
Pp	Port designator	—	Negation
p_1p_0	The bits of the machine code corresponding to the port number	○	Content of flag is set or reset after execution

8

Instruction Code List

D3~D0 \ D7~D4	Hexa-decimal	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NOP	INC @R0	XCH A, @R0	XCHD A, @R0	ORL A, @R0	ANL A, @R0	ADD A, @R0	ADDC A, @R0		MOV STS,A	MOV @R0, A	MOV @R0, #n		XRL A, @R0		MOV A, @R0
0001	1		INC @R1	XCH A, @R1	XCHD A, @R1	ORL A, @R1	ANL A, @R1	ADD A, @R1	ADDC A, @R1			MOV @R1, A	MOV @R1, #n		XRL A, @R1		MOV A, @R1
0010	2	OUT DBB,A	JB0 m	IN A,DBB	JB1 m	MOV A, T	JB2 m	MOV T, A	JB3 m		JB4 m		JB5 m		JB6 m		JB7 m
0011	3	ADD A, #n	ADDC A, #n	MOV A, #n		ORL A, #n	ANL A, #n			RET	RETR	MOVP A, @A	JMPP @A		XRL A, #n	MOVP3 A, @A	
0100	4	JMP 0XX	CALL 0XX	JMP 1XX	CALL 1XX	JMP 2XX	CALL 2XX	JMP 3XX	CALL 3XX	JMP 4XX	CALL 4XX	JMP 5XX	CALL 5XX	JMP 6XX	CALL 6XX	JMP 7XX	CALL 7XX
0101	5	EN I	DIS I	EN TCNTI	DIS TCNTI	STRT CNT	STRT T	STOP TCNT		CLR F0	CPL F0	CLR F1	CPL F1	SEL RB0	SEL RB1	EN DMA	EN FLAGS
0110	6		JTF m	JNT0 m	JT0 m	JNT1 m	JT1 m		JF1 m	JOBF m	JNZ m		JF0 m	JZ m	JNIBF	JNC m	JC m
0111	7	DEC A	INC A	CLR A	CPL A	SWAP A	DA A	RRC A	RR A		CLR C	CPL C		MOV A,PSW	MOV PSW, A	RL A	RLC A
1000	8		INC R0	XCH A, R0		ORL A, R0	ANL A, R0	ADD A, R0	ADDC A, R0			MOV R0, A	MOV R0, #n	DEC R0	XRL A, R0	DJNZ R0, m	MOV A, R0
1001	9	IN A, P1	INC R1	XCH A, R1	OUTL P1, A	ORL A, R1	ANL A, R1	ADD A, R1	ADDC A, R1	ORL P1, #n	ANL P1, #n	MOV R1, A	MOV R1, #n	DEC R1	XRL A, R1	DJNZ R1, m	MOV A, R1
1010	A	IN A, P2	INC R2	XCH A, R2	OUTL P2, A	ORL A, R2	ANL A, R2	ADD A, R2	ADDC A, R2	ORL P2, #n	ANL P2, #n	MOV R2, A	MOV R2, #n	DEC R2	XRL A, R2	DJNZ R2, m	MOV A, R2
1011	B		INC R3	XCH A, R3		ORL A, R3	ANL A, R3	ADD A, R3	ADDC A, R3			MOV R3, A	MOV R3, #n	DEC R3	XRL A, R3	DJNZ R3, m	MOV A, R3
1100	C	MOVD A, P4	INC R4	XCH A, R4	MOVD P4, A	ORL A, R4	ANL A, R4	ADD A, R4	ADDC A, R4	ORLD P4, A	ANLD P4, A	MOV R4, A	MOV R4, #n	DEC R4	XRL A, R4	DJNZ R4, m	MOV A, R4
1101	D	MOVD A, P5	INC R5	XCH A, R5	MOVD P5, A	ORL A, R5	ANL A, R5	ADD A, R5	ADDC A, R5	ORLD P5, A	ANLD P5, A	MOV R5, A	MOV R5, #n	DEC R5	XRL A, R5	DJNZ R5, m	MOV A, R5
1110	E	MOVD A, P6	INC R6	XCH A, R6	MOVD P6, A	ORL A, R6	ANL A, R6	ADD A, R6	ADDC A, R6	ORLD P6, A	ANLD P6, A	MOV R6, A	MOV R6, #n	DEC R6	XRL A, R6	DJNZ R6, m	MOV A, R6
1111	F	MOVD A, P7	INC R7	XCH A, R7	MOVD P7, A	ORL A, R7	ANL A, R7	ADD R7, A	ADDC A, R7	ORLD P7, A	ANLD P7, A	MOV R7, A	MOV R7, #n	DEC R7	XRL A, R7	DJNZ R7, m	MOV A, R7

(dark shading) 2-byte, 2-cycle instruction

(light shading) 1-byte, 2-cycle instruction

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.5~7	V
V_I	Input voltage		−0.5~7	V
V_O	Output voltage		−0.5~7	V
Pd	Power dissipation	Ta=25℃	1500	mW
Topr	Operating temperature range		0~70	℃
Tstg	Storage temperature range		−65~150	℃

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage	2			V
V_{IL}	Low-level input voltage			0.8	V
f(φ)	Operating frequency	1		6	MHz

ELECTRICAL CHARACTERISTICS (Ta = -20~70°C, V_{CC}=5V±10%, unless otherwise noted)

Symbol	Parameter	Conditions	Limits			Unit
			Min	Typ	Max	
V_{IL}	Low-level input voltage (all except X_1, X_2)		−0.5		0.8	V
V_{IH1}	High-level input voltage (all except X_1, X_2, $\overline{RESET}$)		2		V_{CC}	V
V_{IH2}	High-level input voltage (X_1, X_2, $\overline{RESET}$)		3.8		V_{CC}	V
V_{OL1}	Low-level output voltage (D_0~D_7, SYNC)	I_{OL} = 2 mA			0.45	V
V_{OL2}	Low-level output voltage (all except D_0~D_7, SYNC, PROG)	I_{OL} = 1.6mA			0.45	V
V_{OL3}	Low-level output voltage (PROG)	I_{OL} = 1 mA			0.45	V
V_{OH1}	High-level output voltage (D_0~D_7)	I_{OH} = −400μA	2.4			V
V_{OH2}	High-level output voltage (all other outputs)	I_{OH} = −50μA	2.4			V
I_I	Input leakage current (T_0, T_1, $\overline{RD}$, $\overline{WR}$, $\overline{CS}$, A_0)	$V_{SS} \leqq V_I \leqq V_{CC}$			±10	μA
I_{OZL}	Off-state output leakage current (D_0~D_7)	$V_{SS}+0.45 \leqq V_O \leqq V_{CC}$			±10	μA
I_{IL1}	Low-level input current (P_{10}~P_{17}, P_{20}~P_{27})	V_{IL} = 0.8V			0.5	mA
I_{IL2}	Low-level input current ($\overline{RESET}$, $\overline{SS}$)	V_{IL} = 0.8V			0.2	mA
I_{DD}	Supply current from V_{DD}				15	mA
$I_{CC}+I_{DD}$	Total supply current				125	mA

8

TIMING REQUIREMENTS (Ta = 0~70°C, V_{CC}=5V±10%, unless otherwise noted)

DBB Read

Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{c(\phi)}$	Cycle time	t_{CY}		2.5		15	μs
$t_{w(R)}$	Read pulse with	t_{RR}	$t_{c(\phi)}=2.5\mu s$	250			ns
$t_{su(CS-R)}$	Chip-select setup time before read	t_{AR}		0			ns
$t_{h(R-CS)}$	Chip-select hold time after read	t_{RA}		0			ns

DBB Write

Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{w(W)}$	Write pulse width	t_{WW}		250			ns
$t_{su(CS-WR)}$ $t_{su(A_0-WR)}$	$\overline{CS}$, A_0 setup time before write	t_{AW}		0			ns
$t_{h(W-CS)}$ $t_{h(W-A_0)}$	$\overline{CS}$, A_0 hold tie after write	t_{WA}		0			ns
$t_{su(DQ-W)}$	Data setup time before write	t_{DW}		150			ns
$t_{h(W-DQ)}$	Data hold time after write	t_{WD}		0			ns

Port 2

Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{w(PR)}$	PROG pulse width	t_{PP}		1200			ns
$t_{su(PC-PR)}$	Port control setup time before PROG	t_{CP}		110			ns
$t_{h(PR-PC)}$	Port control hold time after PROG	t_{PC}		100			ns
$t_{su(Q-PR)}$	Output data setup time before PROG	t_{DP}		250			ns
$t_{su(D-PR)}$	Input data hold time before PROG.	t_{PR}				810	ns
$t_{h(PR-D)}$	Input data hold time after PROG	t_{PF}		0		150	ns

DMA

Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{su(DACK-R)}$	Data acknowledge time before read	t_{ACC}		0			ns
$t_{h(R-DACK)}$	Data hold time after read	t_{CAC}		0			ns
$t_{su(DACK-W)}$	Data setup time before write	t_{ACC}		0			ns
$t_{h(W-DACK)}$	Data hold time after write	t_{CAC}		0			ns

SWITCHING CHARACTERISTICS (Ta = 0~70°C, V_{CC}=5V±10%, unless otherwise noted)

DBB Read

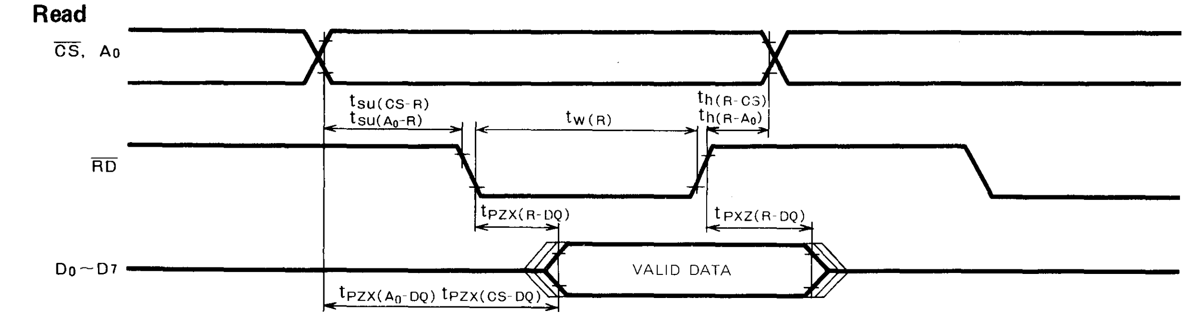
Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{PZX(CS-DQ)}$	Data enable time after $\overline{CS}$	t_{AD}	$C_L=150pF$			225	ns
$t_{PZX(A_0-DQ)}$	Data enable time after address	t_{AD}	$C_L=150pF$			225	ns
$t_{PZX(R-DQ)}$	Data enable time after read	t_{RD}	$C_L=150pF$			225	ns
$t_{PXZ(R-DQ)}$	Data disable time after read	t_{DF}				100	ns

DMA

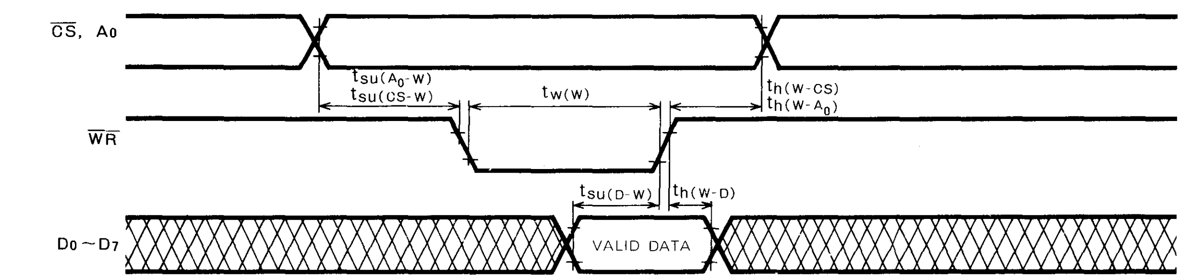
Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{PZX(DACK-DQ)}$	Data enable time after DACK	t_{ACD}	150pF Load			225	ns
$t_{PHL(R-DRQ)}$	DRQ disable time after read	t_{CRQ}	150pF Load			200	ns
$t_{PHL(W-DRQ)}$	DRQ disable time after write	t_{CRQ}	150pF Load			200	ns

TIMING DIAGRAMS

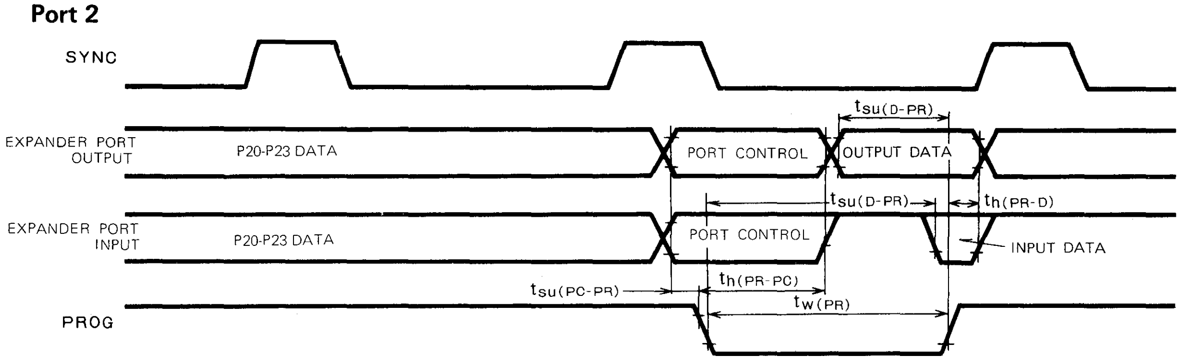
Read



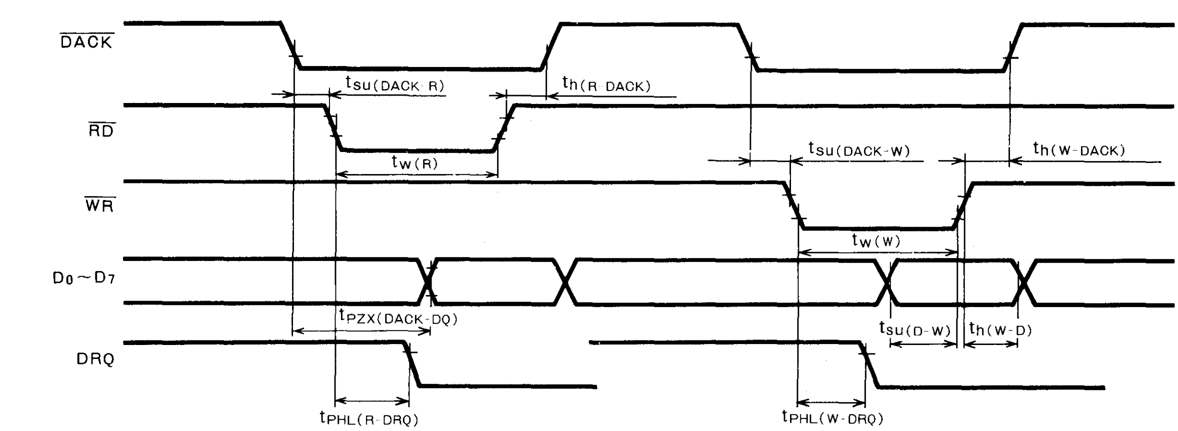
Write



Port 2



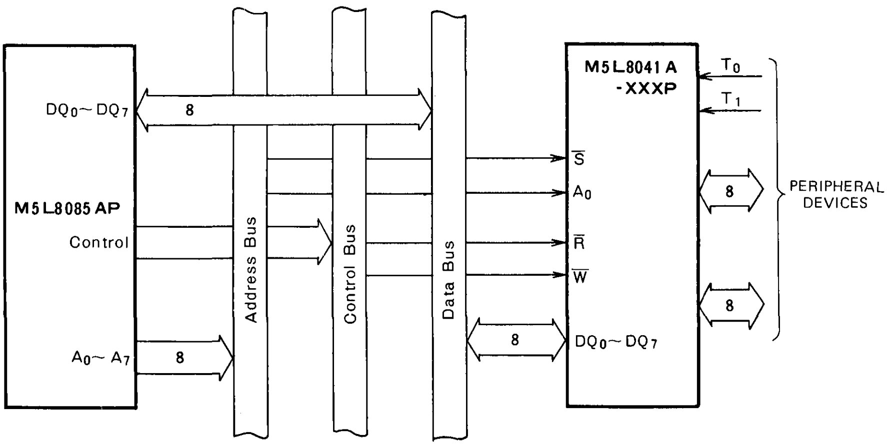
DMA



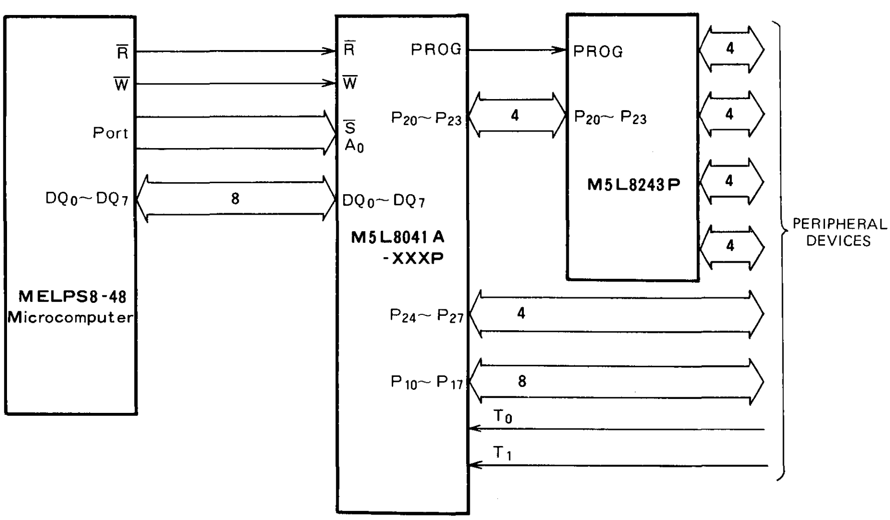
8

APPLICATION EXAMPLES

(1) Interface to an M5L8085AP



(2) Interface to an MELPS 8-48 Microcomputer and M5L8243P



GENERAL INFORMATION

This information explains how to specify the object program for the automatic design system for mask ROMs. This system for mask ROM production has been developed to accept a customer's object program specifications for the automatic design system for a mask ROM.

The main segments of he automatic design system are:

1. The plotter instructions for mask production.
2. A check list for verifyng that the customer's specifications have been met.
3. A test program to assure that the production ROMs meet specifications.

An EPROM in which a program is stored is used for a customer's specifications. A separate (set of) EPROM(s) should be produced for each object program.

Three sets of EPROM(s) should be supplied with the confirmation material.

EPROM SPECIFICATIONS

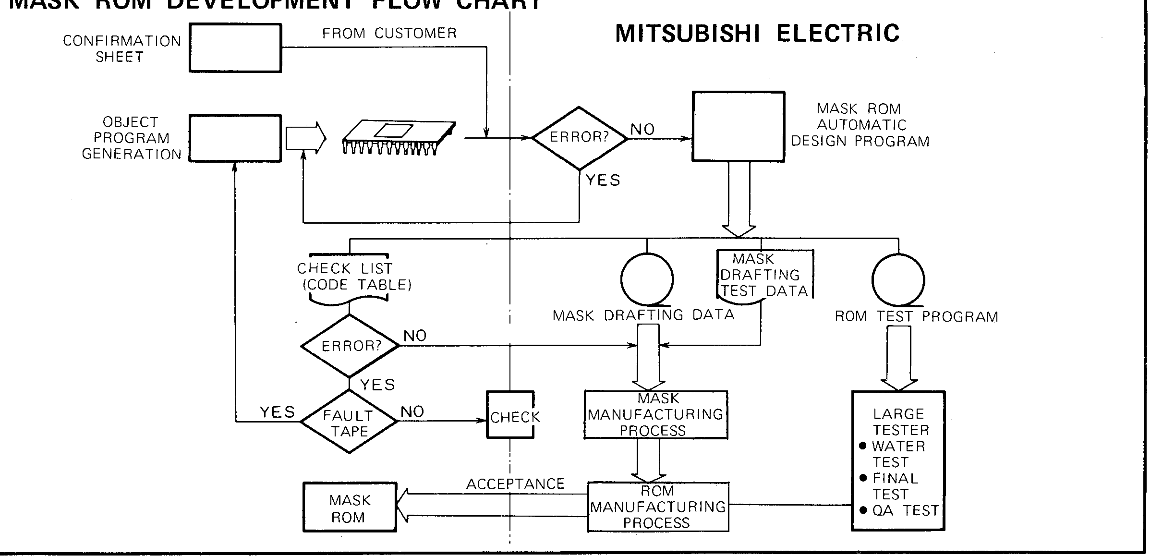
1. The Mitsubishi M5L2708K, M5L2716K, M5L2732K or M5L8748S are standard, but Intel 2708, 2716, 2732, 8741 or 8441A or equivalent devices may be used.
2. The high-level data of both data outputs and address inputs of the supplied EPROM will be programmed as '1', and low-level as '0'.
3. All the data stored in the EPROM are considered as valid and processed to make masks.

ITEMS FOR VERIFICATION

The type of EPROM and address designation symbol A should be marked on the top of the EPROM. In addition, the address indicated by the symbol A should be indicated on the ROM verification sheet.

8

MASK ROM DEVELOPMENT FLOW CHART



MASK-PROGRAMMABLE ROM CONFIRMATION MATERIAL
M5L8041A-XXXP
MASK VERIFICATION SHEET

MITSUBISHI ELECTRIC

Customer		Signature
Company name ______________________		Prepared
Company address ______________________	Tel __________	
Company contact ______________________	Date __________	Approved

* 1. Specify the EPROM to be supplied.
Supply 3 EPROMs of each pattern (□ "✓")

EPROM Type	2708	2716	2732	8741 8741 A
EPROM No.	□A(000_{16}~$3FF_{16}$)	□A(000_{16}~$3FF_{16}$)	□A(000_{16}~$3FF_{16}$)	□A(000_{16}~$3FF_{16}$)

* 2. Part No.

1. Marking required 2. Marking required

MITSUBISHI ELECTRIC IC TYPE NO.

Note 1. Justify the marking to the right.
2. Keep the length to within 12 characters, including alphamerics and hyphens. Do not use J, I, or O.

* 3. Special Remarks

* 4. Description of the final product (In as much detail as possible)

PROGRAMMABLE COMMUNICATION INTERFACE

DESCRIPTION

The M5L 8251AP is a universal synchronous/asynchronous receiver/transmitter (USART) IC chip designed for data communications use. It is produced using the N-channel silicon-gate ED-MOS process and is mainly used in combination with 8-bit microprocessors.

FEATURES

- Single 5V power supply
- Synchronous and asynchronous operation
  - Synchronous:
    - 5~8-bit characters
    - Internal or external synchronization
    - Automatic SYNC character insertion
  - Asynchronous system:
    - 5~8-bit characters
    - Clock rate—1, 16 or 64 times the baud rate
    - 1, 1½, or 2 stop bits
    - False-start-bit detection
    - Automatic break-state detection
- Baud rate: DC~64K-baud
- Full duplex, double-buffered transmitter/receiver
- Error detection: parity, overrun, and framing
- Pin connection and electrical characteristics compatible with Intel's 8251A

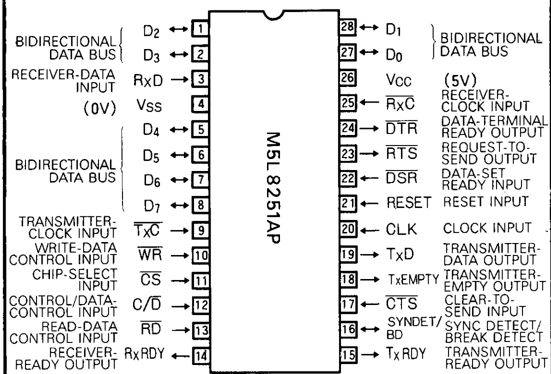
APPLICATIONS

- Modem control of data communications using microcomputers
- Control of CRT, TTY and other terminal equipment

FUNCTION

The M5L8251AP is used in the peripheral circuits of a CPU. It permits assignments, by means of software, of operations in all the currently used serial-data transfer systems including IBM's 'bi-sync.'

PIN CONFIGURATION (TOP VIEW)



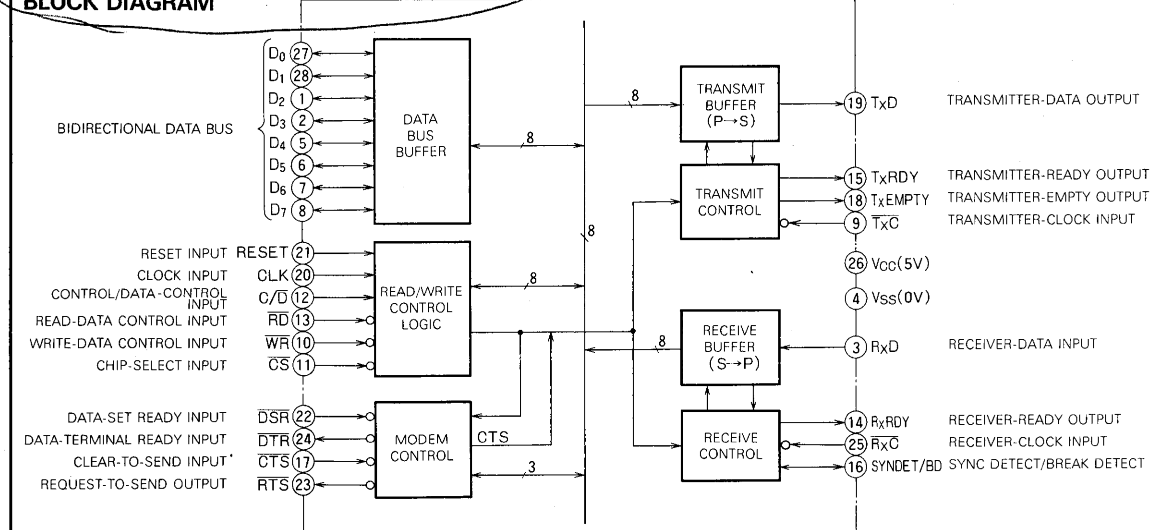
Outline 28P4

The M5L8251AP receives parallel-format data from the CPU, converts it into a serial format, and then transmits via the TxD pin. It also receives data sent in via the RxD pin from the external circuit, and converts it into a parallel format for sending to the CPU.

On receipt of parallel-format data for transmission from the CPU or serial data for the CPU from external devices, the M5L8251AP informs the CPU using the TxRDY or RxRDY pin. In addition, the CPU can read the M5L 8251AP status at any time.

The M5L8251AP can detect the data received for errors and inform the CPU of the presence of errors as status information. Errors include parity, overrun and frame errors.

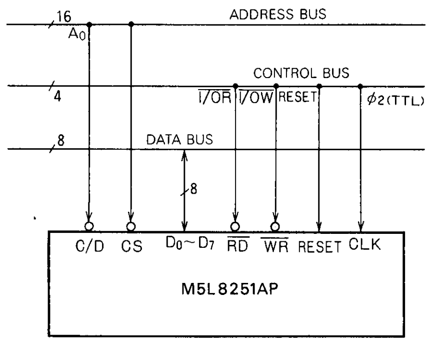
BLOCK DIAGRAM



OPERATION

The M5L 8251AP interfaces with the system bus as shown in Fig. 1, positioned between the CPU and the modem or terminal equipment, and offers all the functions required for data communication.

Fig. 1 M5L8251AP interface to 8080A standard system bus



When using the M5L 8251AP, it is necessary to program, as the initial setting, assignments for synchronous/asynchronous mode selection, baud rate, character length, parity check, and even/odd parity selection in accordance with the communication system used. Once programming is completed, functions appropriate to the communication system can be carried out continuously.

When initial setting of the USART is completed, data communication becomes possible. Though the receiver is always in the enable state, the transmitter is placed in the transmitter-enable state (T_XEN) by a command instruction, and the application of a low-level signal to the $\overline{CTS}$ pin prompts data-transfer start-up. Until this condition is satisfied, transmission is not executed. On receiving data, the receiver informs the CPU that reading of the receiver data in the USART by the CPU has become possible (the R_XRDY terminal has turned to '1'). Since data reception and the entry of the CPU into the data-readable state are output as status information, the CPU can assess USART status without accessing the R_XRDY terminal.

During receiving operation, the USART checks errors and gives out status information. There are three types of errors: parity, overrun, and frame. Even though an error occurs, the USART continues its operations, and the error state is retained until error reset (ER) is effected by a command instruction. The M5L8251AP access methods are listed in Table 1.

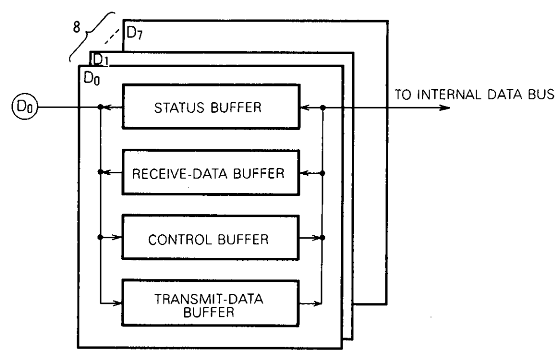
Table 1 M5L8251AP Access Methods

C/$\overline{D}$	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	Function
L	L	H	L	Data bus ← Data in USART
L	H	L	L	USART ← Data bus
H	L	H	L	Data bus ← Status
H	H	L	L	Control ← Data bus
X	H	H	L	3-State ← Data bus
X	X	X	H	3-State ← Data bus

Data-Bus Buffer

This is an 8-bit, 3-state bidirectional bus buffer through which control words, command words, status information, and transfer data are transferred. Fig. 2 shows the structure of the data-bus buffer.

Fig. 2 Data-bus-buffer structure



Read/Write Control Logic

This logic consists of a control word register and command word register. It receives signals from the CPU control bus and generates internal-control signals for the elements.

Modem Control Circuit

This is a general-purpose control-signal circuit designed to simplify the interface to the modem. Four types of control signal are available: output signals $\overline{DTR}$ and $\overline{RTS}$ are controlled by command instructions, input signal $\overline{DSR}$ is given to the CPU as status information and input signal $\overline{CTS}$ controls direct transmission.

Transmit Buffer

This buffer converts parallel-format data given to the data-bus buffer into serial data with addition of a start bit, stop bits and a parity bit, and sends out the converted data through the T_XD pin based on the control signal.

Transmit-Control Circuit

This circuit carries out all the controls required for serial-data transmission. It controls transmitter data and outputs the signals required by external devices in accordance with the instructions of the read/write control logic.

Receive Buffer

This buffer converts serial data given via the R_XD pin into a parallel format, checks the bits and characters in accordance with the communication format designated by mode setting, and transfers the assembled characters to the CPU via the data-bus buffer.

Receive Control Circuit

This circuit offers all the controls required for normal reception of the input serial data. It controls receiver data and outputs signals for the external devices in accordance with the instructions of the read/write control logic.

Clock Input (CLK)

This system-clock input is required for internal-timing generation and is usually connected to the clock-output ($\phi_{2(TTL)}$) pin of the M5L8224P. Although there is no direct relation with the data-transfer baud rate, the clock-input (CLK) frequency is more than 30 times the $\overline{T_XC}$ or $\overline{R_XC}$ input frequency in the case of the synchronous system and more than 4.5 times in the case of the asynchronous system.

Reset Input (RESET)

Once the USART is shifted to the idle mode by a high-level input, this state continues until a new control word is set. Since this is a master reset, it is always necessary to load a control word following the reset process. The reset input requires a minimum 6-clock pulse width.

Data-Set Ready Input ($\overline{DSR}$)

This is a general-purpose input signal, but is usually used as a data-set ready signal to test modem status. Its status can be known from the status reading process. The D_7 bit of the status information equals '1' when the $\overline{DSR}$ pin is in the low state, and '0' when in the high state.

$\overline{DSR}$ = L → D_7 bit of status information = 1
$\overline{DSR}$ = H → D_7 bit of status information = 0

Note: DSR indicates modem status as follows:
ON means the modem can transmit and receive;
OFF means it cannot.

Data-Terminal Ready Output ($\overline{DTR}$)

This is a general-purpose output signal, but is usually used as a data-terminal ready or rate-select signal to the modem. The $\overline{DTR}$ pin is controlled by the D_1 bit of the command instruction; if D_1 = 1, $\overline{DTR}$ = L, and if D_1 = 0, $\overline{DTR}$ = H.

D_1 of the command register = 1 → $\overline{DTR}$ = L
D_1 of the command register = 0 → $\overline{DTR}$ = H

Chip-Select Input ($\overline{CS}$)

This is a device-select signal that enables the USART by a low-level input. Usually, it is connected to the address bus directly or via the decoder. When this signal is in the high state, the M5L 8251AP is disabled.

Write-Data Control Input ($\overline{WR}$)

Data and control words output from the CPU by the low-level input are written in the M5L 8251AP. This terminal is usually used in a form connected with the control bus $\overline{I/OW}$ of the CPU.

Read-Data Control Input ($\overline{RD}$)

Receiver data and status information are output from the CPU by a low-level input for the CPU data bus.

Control/Data Control Input (C/$\overline{D}$)

This signal shows whether the information on the USART data bus is in the form of data characters or control words, or in the form of status information, in accordance with the $\overline{RD}$ and $\overline{WR}$ inputs while the CPU is accessing the M5L8251AP. The high level identifies control words or status information, and the low level, data characters.

Request-To-Send Output ($\overline{RTS}$)

This is a general-purpose output signal but is used as a request-to-send signal for the modem. The $\overline{RTS}$ terminal is controlled by the D_5 bit of the command instruction. When D_5 is equal to '1', $\overline{RTS}$ = L, and when D_3 is 0, $\overline{RTS}$ = H.

Command register D_5 = 1 → $\overline{RTS}$ = L
Command register D_5 = 0 → $\overline{RTS}$ = H

Note: RTS controls the modem transmission carrier as follows:
ON means carrier dispatch;
OFF means carrier stop.

Clear-To-Send Input ($\overline{CTS}$)

When the T_XEN bit (D_0) of the command instruction has been set to '1' and the $\overline{CTS}$ input is low, serial data is sent out from the T_XD pin. Usually this is used as a clear-to-send signal for the modem.

Note: CTS indicates the modem status as follows:
ON means data transmission is possible;
OFF means data transmission is impossible.

Transmission-Data Output (T_XD)

Parallel-format transmission characters loaded on the M5L 8251AP by the CPU are assembled into the format designated by the mode instruction and sent in serial-data form via the T_XD pin. Data is output, however, only in cases where the D_0 bit (T_XEN) of the command instruction is '1' and the $\overline{CTS}$ terminal is in the low state. Once reset, this pin is kept at the mark status (high level) until the first character is sent.

Transmitter-Ready (T_XRDY)

This signal shows that the data is ready for transmission. It is possible to confirm the status of serial-data transmission by using it as an interruption signal for the CPU or by allowing the CPU to read the D_0 bit of the status information by polling. Since the T_XRDY signal shows that the data buffer is empty, it is automatically reset when a transmission character is loaded by the CPU. The T_XRDY bit of the status information means that the transmit-data buffer shown in Fig. 2 has become empty, while the T_XRDY pin enters the high-level state only when the transmit-data buffer is empty, T_XEN equals '1', and a low-level input has been applied to the $\overline{CTS}$ pin.

Status (D_0): Transmit-data buffer (TDB) is empty and '1'.

TxRDY terminal: When (TDB is empty)·(TxEN = 1)·(CTS = 0) = 1 or resetting, it becomes active.

Transmitter-Empty Output (T_XEMPTY)

When no transmisison characters are left in the transmit buffer, this pin enters the high state. In the asynchronous mode, the following transmission character is shifted to the transmit buffer when it is loaded from the CPU. Thus, it is automatically reset. In the synchronous mode, a SYNC character is loaded automatically on the transmit buffer when no transfer-data characters are left. In this case, however, the T_XEMPTY does not enter the low state when a SYNC character has been sent out, since T_XEMPTY = H denotes the state in which there is no transfer character and one or two SYNC characters are being transferred or the state in which a SYNC character is being transferred as a filler. T_XEMPTY is unrelated to the T_XEN bit of the command instruction.

Transmitter-Clock Input ($\overline{T_XC}$)

This clock controls the baud rate for character transmission from the T_XD pin. Serial data is shifted by the rising edge of the $\overline{T_XC}$ signal. In the synchronous mode, the $\overline{T_XC}$ frequency is equal to the actual baud rate. In the asynchronous mode, the frequency is specified as 1, 16, or 64 times the baud rate by the mode setting.

Example When the baud rate is 110 bauds:

$\overline{T_XC}$ = 110Hz (1X)

$\overline{T_XC}$ = 1.76kHz (16X)

$\overline{T_XC}$ = 7.04kHz (64X)

Receiver-Data Input (R_XD)

Serial characters sent from another device are input to this pin and converted to a parallel-character format to serve as data for the CPU. Unless the '1' state is detected after a chip-master reset procedure (this resetting is carried out to prevent spurious operation such as that due to faulty connection of the R_XD to the line in a break state), the serial characters are not received. This applies to only the asynchronous mode. When the R_XD line enters the low state instantaneously because of noise, etc., the mis-start prevention function starts working. That is, the start bit is detected by its falling edge but in order to make sure that it is the correct start bit, the R_XD line is strobed at the middle of the start bit to reconfirm the low state. If it is found to be high, a faulty-start judgment is made.

Receiver-Ready Output (R_XRDY)

This signal indicates that the received characters have entered the receiver buffer, and further, the receiver-data buffer in the data-bus buffer shown in Fig. 2. It is possible to confirm the R_XRDY status by using this signal as an interruption signal for the CPU or by allowing the CPU to read the D_1 bit of the status information by polling. The R_XRDY is automatically reset when a character is read by the CPU. Even in the break state in which the R_XD line is held at low, the R_XRDY remains active. It can be masked by making the R_XE (D_2) of the command instruction '0'.

Receiver-Clock Input ($\overline{R_XC}$)

This clock signal controls the baud rate for the sending in of characters via the $\overline{R_XD}$ pin. The data is shifted in by the rising edge of the $\overline{R_XC}$ signal. In the synchronous mode, the $\overline{R_XC}$ frequency is equal to the actual baud rate. In the asynchronous mode, the frequency is specified as 1, 16, or 64 times the baud rate by mode setting. This relationship is parallel to that of $\overline{T_XC}$, and in usual communication-line systems the transmission and reception baud rates are equal. The $\overline{T_XC}$ and $\overline{R_XC}$ terminals are, therefore, used connected to the same baud-rate generator.

Sync Detect/Break Detect Output-Input (SYNDET/BD)

In the synchronous mode this pin is used for input and output operations. When it is specified for the internal synchronous mode by mode setting, this pin works as an output terminal. It enters the high state when a SYNC character is received through the R_XD pin. If the M5L 8251AP has been programmed for double SYNC characters (bi-sync), a high is entered in the middle of the last bit of the second SYNC character. This signal is automatically reset by reading the status information.

On designation of the M5L8251AP to the external synchronous mode, this pin begins to serve for input operations. Applying a high signal to this pin prompts the M5L 8251AP to begin assembling data characters at the next rising edge of the $\overline{R_XC}$. For the width of a high-level signal to be input, a minimum $\overline{R_XC}$ period is required.

Designation of the asynchronous mode causes this pin to function as a BD (output) pin. When the start, data, and parity bits and a stop bit are all in the low state, a high is entered. The BD (break detect) signal can also be read as the D_6 bit of the status information. This signal is reset by resetting the chip master or by the R_XD line's recovering the high state.

PROGRAMMING

It is necessary for the M5L8251AP to have the control word loaded by the CPU prior to data transfer. This must always be done following any resetting operation (by external RESET pin or command instruction IR). There are two types of control words: mode instructions specifying general operations required for communications and command instructions to control the M5L8251AP's actual operations.

Following the resetting operation, a mode instruction must be set first. This instruction sets the synchronous or asynchronous system to be used. In the synchronous system, a SYNC character is loaded from the CPU. In the case of the bi-sync system, however, a second SYNC character must be loaded in succession.

Loading a command instruction makes data transfer possible. This operation after resetting must be carried out for initializing the M5L8251AP. The USART command instruction contains an internal-reset IR instruction (D_6 bit) that makes it possible to return the M5L8251AP to its reset state. The initialization flowchart is shown in Fig. 3, and the mode-instruction and command-instruction formats are shown in Figs. 4 and 5.

Fig. 3 Initialization flow chart

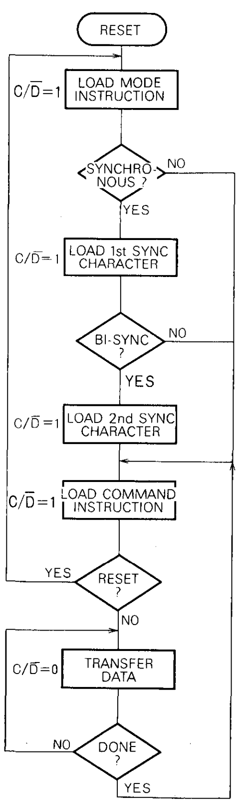


Fig. 4 Mode-instruction format

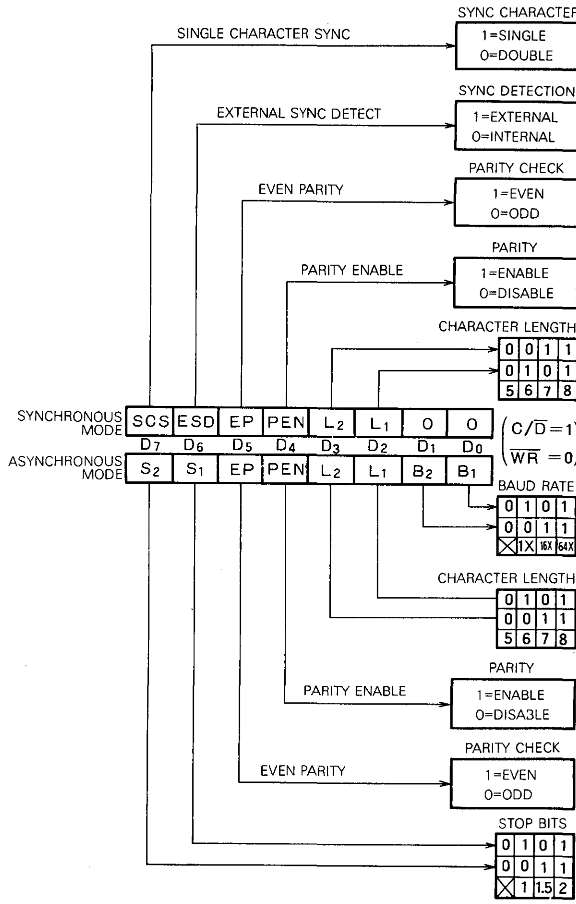
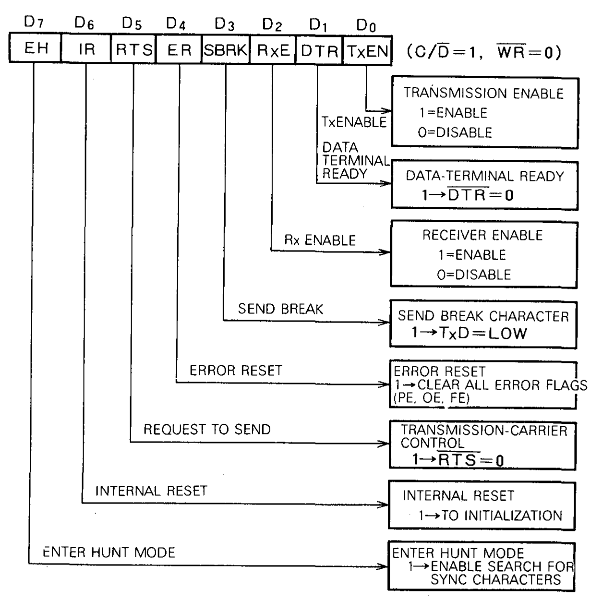


Fig. 5 Command-instruction format



Asynchronous Transmission Mode

When data characters are loaded on the M5L 8251AP after initial setting, the USART automatically adds a start bit (low), an odd or even parity bit specified by the mode instruction during initialization, and a specified number of stop bits (high). After that, the assembled data characters are transferred as serial data via the T_XD pin if transfer is enabled ($T_XEN = 1 \cdot \overline{CTS} = L$). In this case, the transfer data (baud rate) is shifted by the mode instruction at a rate of 1X, 1/16X, or 1/64X the $\overline{T_XC}$ period.

If the data characters are not loaded on the M5L 8251AP, the T_XD pin enters a mark state (high). When SBRK is programmed by the command instruction, break characters (low) are output continuously through the T_XD pin.

Asynchronous Reception Mode

The R_XD line usually starts operations in a mark state (high), triggered by the falling edge of a low-level pulse when it comes to this line. This signal is again strobed at the middle of the bit to confirm that it is a perfect start bit. The detection of a second low indicates the validity of the start bit (restrobing is carried out only in the case of 16X and 64X). After that, the bit counter inside the M5L 8251AP starts operating; each bit of the serial information on the R_XD line is shifted in by the rising edge of $\overline{R_XC}$, and the data bit, parity bit (when necessary), and stop bit are sampled at the middle position.

The occurrence of a parity error causes the setting of a parity-error flag. If the stop bit is in the low state, a frame-error flag is set. Attention should be paid to the fact that the receiver requires only one stop bit even though the program has designated 1½ or 2 stop bits.

Reception up to the stop bit means reception of a complete character. This character is then transferred to the receiver-data buffer shown in Fig. 2, and the R_XRDY becomes active. In cases where this character is not led by the CPU and where the next character is transferred to the receiver-data buffer, the preceding character is destroyed and an overrun-error flag is set.

These error flags can be read as the M5L 8251AP status information. The occurrence of an error does not stop USART operations. The error flags are cleared by the ER (D_4 bit) of the command instruction.

The asynchronous-system transfer formats are shown in Figs. 6 and 7.

Synchronous Transmission Mode

In this mode the T_XD pin remains in the high state until initial setting by the CPU is completed. After initialization, the state of $\overline{CTS} = L$ and $T_XEN = 1$ causes serial transmission of SYNC characters through the T_XD pin. Then, data characters are sent out and shifted by the falling edge of the $\overline{T_XC}$ signal. The transmission rate equals the $\overline{T_XC}$ rate.

Thus, once data-character transfer starts, it must continue through the T_XD pin at the same rate as that of T_XC. Unless data characters are provided from the CPU before the transmitter buffer becomes empty, one or two SYNC characters are automatically output from the T_XD pin. In this case, it should be noted that the T_XEMPTY pin enters the high state when there are no data characters left in the M5L8251AP to be transferred, and that the low state is not entered until the USART is provided with the next data character from the CPU. Care should also be taken over the fact that merely setting a command instruction does not effect SYNC-character insertion, because the SYNC character is sent out after loading of the data characters.

In this mode, too, break characters are sent out in succession from the T_XD pin when SBRK is designated ($D_3 = 1$) by a command instruction.

Fig. 6 Asynchronous transmission format I (transmission)

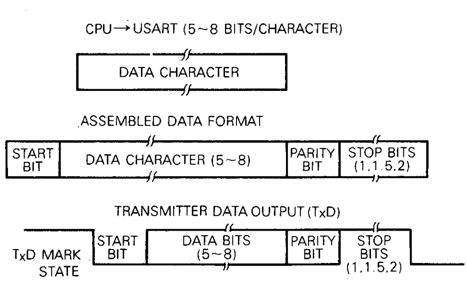
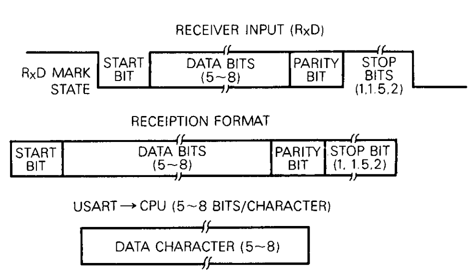


Fig. 7 Asynchronous transmission format II (reception)



Note : When the data character is 5, 6, or 7 bits/character length, the unused bits (for USART→ CPU) are set to zero.

Synchronous Reception Mode

Character synchronization in this mode is carried out internally or externally by initial-setting designation.

Programming in the internal synchronous mode requires that an EH instruction (D_7 = 1, enter hunt mode) is included in the first command instruction. Data on the RxD pin is sampled by the rising $\overline{R_xC}$ signal, and the receiver-buffer contents are compared with the SYNC character each time a bit is input. Comparison continues until an agreement is reached. When the M5L 8251AP has been programmed in the bi-sync mode, data received in further succession is compared. The detection of two SYNC characters in succession makes the USART end the hunt mode, setting the SYNDET pin to the high state. This reset operation is prompted by the reading of the status information. When the parity has been programmed, SYNDET is not set in the middle of the last data bit but in the middle of the parity bit.

In the external synchronous mode, the M5L 8251AP gets out of the hunt mode when a high synchronization signal is given to the SYNDET pin. The high signal requires a minimum duration of one $\overline{R_xC}$ cycle. In the asynchronous mode, however, the EH signal does not affect the operation at all.

Parity and overrun errors are checked in the same way as in the asynchronous system. During hunt-mode operations the parity bit is not checked, but parity checking is carried out even when the receiver is disabled.

The CPU can command the receiver to enter the hunt mode, if synchronization is lost. This prevents the SYNC character from erroneously becoming equal to the received data when all the data in the receiver buffer is set to '1' Attention should be paid to the fact that the SYNDET F/F is reset each time status information is read irrespective of the synchronous mode's being internal or external. This, however, does not return the M5L 8251AP to the hunt mode. Synchronism detection is carried out even though it is not the hunt mode. The synchronous transfer formats are shown in Figs. 8 and 9.

Command Instruction

This instruction defines actual operations in the communication mode designated by mode setting. Command instructions include transmitter/receiver enable error-reset, internal-reset, modem-control, enter-hunt and break transmission instructions.

The mode is set following the reset operation. A SYNC character is set as required, and the writing of high-level signals on the control/data pin ($C/\overline{D}$) that follows it is regarded as a command instruction. When the mode is set all over again from the beginning, the M5L 8251AP can be reset by using inputting via the reset terminal or by internal resetting based on the command instruction.

Note 1: The command error reset (ER), internal reset (IR) and enter-hunt-mode (EH) operations are only effective when the command instruction is loaded, so that these bits need not be returned to '0'.

2: When a break character is sent out by a command, the T_xD enters the low state immediately irrespective of whether or not the USART has sent out data.

3: Operations of the USART's receiver section which is always in the enable state cannot be inhibited. The command instruction R_xE = 0 does not mean that data reception via the R_xD pin is inhibited; it means that the R_xRDY is masked and error flags are inhibited.

Fig. 8 Synchronous transmission format I (transmission)

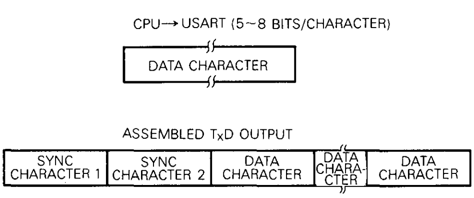
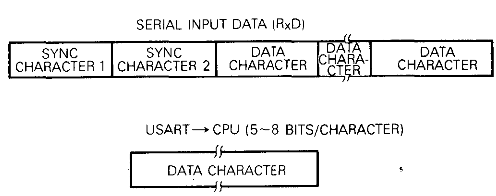


Fig. 9 Synchronous transmission format II (reception)



Note : When the data character is 5, 6, or 7 bits/character length, the unused bits (for USART→ CPU) are set to zero.

Status Information

The CPU can always read USART status by setting the $C/\overline{D}$ to '1' and $\overline{RD}$ to '0'.

The status information format is shown in Fig. 10. In this format R_XRDY, T_XEMPTY and SYNDET have the same definitions as those of the pins. This means that these three pieces of status information become '1' when each pin is in the high state. The other status information is defined as follows:

DSR: When the $\overline{DSR}$ pin is in the low state, status information DSR becomes '1'.

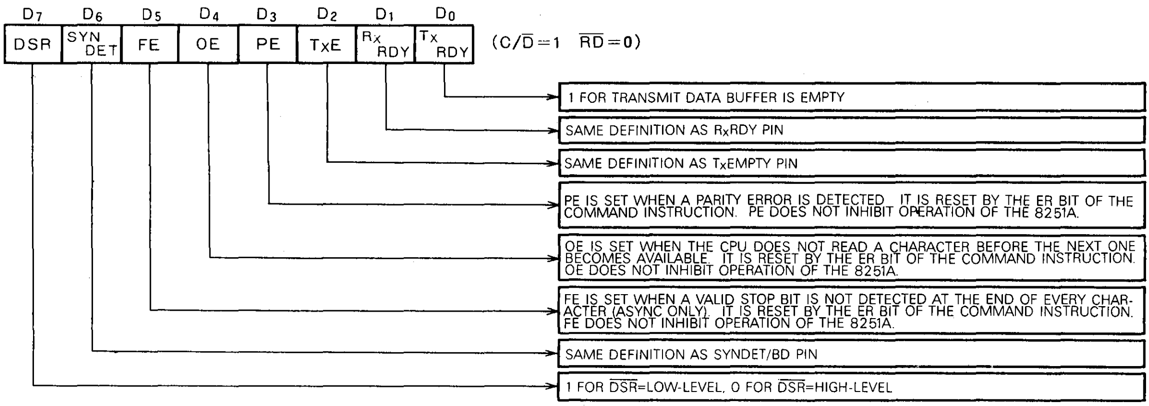
FE: The occurrence of a frame error in the receiver section makes the status information FE '1'.

OE: The occurrence of an overrun error in the receiver section makes the status information OE '1'.

PE: The occurrence of a parity error in the receiver section makes this status information PE '1'.

T_XRDY: This information becomes '1' when the transmit-data buffer is empty. Be careful because this has a different meaning from the T_XRDY pin that enters the high state only when the transmitter buffer is empty, when the $\overline{CTS}$ pin is in the low state, and when T_XEN is '1'.

Fig. 10 Status information



APPLICATION EXAMPLES

Fig. 11 shows an application example for the M5L 8251AP in the asynchronous mode. When the port addresses of the M5L 8251AP are assumed to be 00# and 01# in this figure, initial setting in the asynchronous mode is carried out in the following manner:

MVI	**A, B6#**	Mode setting
OUT	**01#**	
MVI	**A, 27#**	Command instruction
OUT	**01#**	

In this case, the following are set by mode setting:

Asynchronous mode
6 bits/character
Parity enable (even)
1½ stop bits
Baud rate: 16X

Command instructions set the following:

RTS = 1 → $\overline{RTS}$ pin = L
R_XE = 1
DTR = 1 → $\overline{DTR}$ pin = L
T_XEN = 1

When the initial setting is complete, transfer operations are allowed. The $\overline{RTS}$ pin is initially set to the low-level by setting RTS to '1', and this serves as a $\overline{CTS}$ input with T_XEN being equal to '1'. For this reason the same definition applies to the status and pin of T_XRDY, and '1' is assigned when the transmit-data buffer is empty. Actual transfer of data is carried out in the following way:

IN	**01#**	Status read

The IN instruction prompts the CPU to read the USART's status. The result is: if the T_XRDY equals '1' transmitter data is sent from the CPU and written on the M5L 8251AP. Transmitter data is written in the M5L 8251AP in the following manner:

MVI	**A, 2D#**	$2D_{16}$ is an example of transmitter data.
OUT	**00#**	USART ← (A)

Receiver data is read in the following manner:

IN	**00#**	(A) ← USART

In the above example, the status information is read and as a result, the transmitter data is written and read. Interruption processing by using the T_XRDY and R_XRDY pins is also possible.

Fig. 12 shows the status of the T_XD pin when data written in the USART is transferred from the CPU. When the data shown in Fig. 12 enters the RxD pin, data sent from the M5L 8251AP to the CPU becomes $2D_{16}$ and bits D_6 and D_7 are treated as '0'.

Fig. 11 Example of circuit using the asynchronous mode

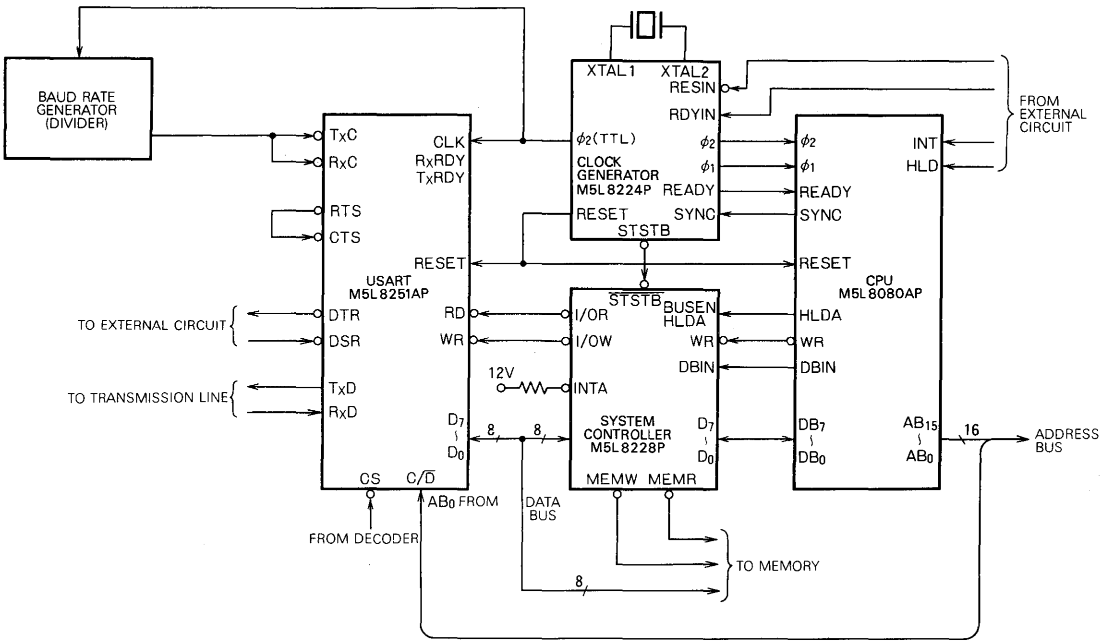
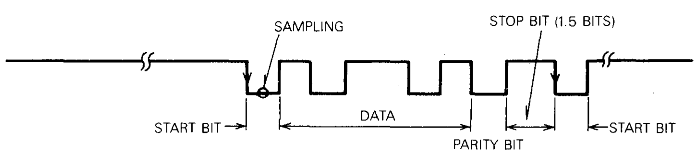


Fig. 12 Example of data transmission



8

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Power-supply voltage	With respect to V_{SS}	−0.5～7	V
V_I	Input voltage		−0.5～7	V
V_O	Output voltage		−0.5～7	V
P_d	Power dissipation		1000	mW
T_{opr}	Operating free-air temperature range		0～70	℃
T_{stg}	Storage temperature range		−65～150	℃

RECOMMENDED OPERATING CONDITIONS (Ta = 0～70℃, unless otherwise noted.)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{SS}	Power-supply voltage		0		V
V_{IH}	High-level input voltage	2.2		V_{CC}	V
V_{IL}	Low-level input voltage	−0.5		0.8	V

ELECTRICAL CHARACTERISTICS (Ta = 0～70℃, V_{CC} = 5 V ± 5%, V_{SS} = 0 V, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	$I_{OH} = -400\mu A$	2.4			V
V_{OL}	Low-level output voltage	$I_{OL} = 2.2mA$			0.45	V
I_{CC}	Supply current from V_{CC}	All outputs are high.			100	mA
I_{IH}	High-level input current	$V_I = V_{CC}$	−10		10	μA
I_{IL}	Low-level input current	$V_I = 0.45V$	−10		10	μA
I_{OZ}	Off-state input current	$V_{SS} = 0V$, $V_I = 0.45～5.25V$	−10		10	μA
C_i	Input capacitance	$V_{CC} = V_{SS}$, f = 1MHz, 25mVrms, Ta = 25℃			10	pF
$C_{i/o}$	Input/output capacitance	$V_{CC} = V_{SS}$, f = 1MHz, 25mVrms, Ta = 25℃			20	pF

TIMING REQUIREMENTS (T_a = 0 ~ 70℃, V_{CC} = 5 V ± 5 %, V_{SS} = 0 V, unless otherwise noted.)

Symbol	Parameter		Alternative symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{C(\phi)}$	Clock cycle time (Notes 1, 2)		t_{CY}		320		1350	ns
$t_{W(\phi)}$	Clock high pulse width		t_{ϕ}		140		$t_{C(\phi)}-90$	ns
$t_{W(\bar{\phi})}$	Clock low pulse width		$t_{\bar{\phi}}$		90			ns
t_r	Clock rise time		t_R		5		20	ns
t_f	Clock fall time		t_F		5		20	ns
f_{TX}	Transmitter input clock frequency	1 X baud rate	f_{TX}		DC		64	kHz
		16 X baud rate	f_{TX}		DC		310	kHz
		64 X baud rate	f_{TX}		DC		615	kHz
$t_{W(TPWL)}$	Transmitter input clock low pulse width	1X baud rate	t_{TPW}		12			$t_{C(\phi)}$
		16X, 64X baud rate	t_{TPW}		1			$t_{C(\phi)}$
$t_{W(TPWH)}$	Transmitter input clock high pulse width	1X baud rate	t_{TPD}		15			$t_{C(\phi)}$
		16X, 64X baud rate	t_{TPD}		3			$t_{C(\phi)}$
f_{RX}	Receiver input clock frequency	1X baud rate	f_{RX}		DC		64	kHz
		16X baud rate	f_{RX}		DC		310	kHz
		64X baud rate	f_{RX}		DC		615	kHz
$t_{W(RPWL)}$	Receiver input clock low pulse width	1X baud rate	t_{RPW}		12			$t_{C(\phi)}$
		16X, 64X baud rate	t_{RPW}		1			$t_{C(\phi)}$
$t_{W(RPWH)}$	Receiver input clock high pulse width	1X baud rate	t_{RPD}		15			$t_{C(\phi)}$
		16X, 64X baud rate	t_{RPD}		3			$t_{C(\phi)}$
$t_{su(A-R)}$	Address setup time before read ($\overline{CS}$, C/$\overline{D}$) (Note 3)		t_{AR}		50			ns
$t_{h(R-A)}$	Address hold time after read ($\overline{CS}$, C/$\overline{D}$) (Note 3)		t_{RA}		50			ns
$t_{W(R)}$	Read pulse width		t_{RR}		250			ns
$t_{su(A-W)}$	Address setup time before write		t_{AW}		50			ns
$t_{h(W-A)}$	Address hold time after write		t_{WA}		50			ns
$t_{W(W)}$	Write pulse width		t_{WW}		250			ns
$t_{su(DQ-W)}$	Data setup time before write		t_{DW}		150			ns
$t_{h(W-DQ)}$	Data hold time after write		t_{WD}		50			ns
$t_{su(ESD-RxC)}$	E·SYNDET setup time before RxC		t_{ES}		16			$t_{C(\phi)}$
$t_{su(C-R)}$	Control setup time before read		t_{CR}		20			$t_{C(\phi)}$
t_{RV}	Write recovery time between writes (Note 4)		t_{RV}		6			$t_{C(\phi)}$
$t_{su(RxD-IS)}$	RxD setup time before internal sampling pulse		t_{SRx}		2			μs
$t_{h(IS-RxD)}$	RxD hold time after internal sampling pulse		t_{HRx}		2			μs

Note 1 : The $\overline{TxC}$ and $\overline{RxC}$ frequencies have the following limitations with respect to CLK.
For 1X baud rate f_{TX}, $f_{RX} \leq 1/(30 t_{C(\phi)})$. For 16X, 64X baud rate f_{TX}, $f_{RX} \leq 1/(4.5 t_{C(\phi)})$

2 : Reset pulse width=6 $t_{C(\phi)}$ minimum; system clock must be running during reset.

3 : $\overline{CS}$, C/$\overline{D}$ are considered as address.

4 : This recovery time is for mode initialization only. Write data is allowed only when TxRDY=1. Recovery time between writes for asynchronous mode is 8 $t_{C(\phi)}$, and that for synchronous mode is 16 $t_{C(\phi)}$

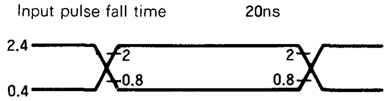
SWITCHING CHARACTERISTICS (T_a = 0 ~ 70℃, V_{CC} = 5 V ± 5 %, V_{SS} = 0 V, unless otherwise noted.)

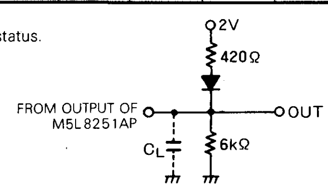
Symbol	Parameter	Alternative symbol	Test conditions (Note 7)	Limits Min	Limits Typ	Limits Max	Unit
$t_{PZV(R-DQ)}$	Output data enable time after read (Note 5)	t_{RD}	C_L = 150pF			250	ns
$t_{PVZ(R-DQ)}$	Output data disable time afer read	t_{DF}		10		100	ns
$t_{PZV(TxC-TxD)}$	TxD enable time after falling edge of $\overline{TxC}$	t_{DTx}				1	μs
$t_{PLH(CLB-TxR)}$	Propagation time from center of last bit to TxRDY clear (Note 6)	t_{TxRDY}				8	$t_{C(\phi)}$
$t_{PHL(W-TxR)}$	Propagation time from write data to TxRDY (Note 6)	$t_{TxRDY\ CLEAR}$				6	t_{CY}
$t_{PLH(CLB-RxR)}$	Propagation time from center of last bit to RxRDY (Note 6)	t_{RxRDY}				24	$t_{C(\phi)}$
$t_{PHL(R-RxR)}$	Propagation time from read data to RxRDY clear (Note 6)	$t_{RxRDY\ CLEAR}$				6	t_{CY}
$t_{PLH(RxC-SYD)}$	Propagation time from rising edge of $\overline{RxC}$ to internal SYNDET (Note 6)	t_{IS}				24	$t_{C(\phi)}$
$t_{PLH(CLB-TxE)}$	Propagation time from center of last bit to TxEMPTY (Note 6)	$t_{TxEMPTY}$		20			$t_{C(\phi)}$
$t_{PHL(W-C)}$	Propagation time from rising edge of $\overline{WR}$ to control (Note 6)	t_{WC}		8			$t_{C(\phi)}$

Note 5 : Assumes that address is valid before falling edge of $\overline{RD}$.

6 : Status-up data can have a maximum delay of 28 clock periods from the event affecting the status.

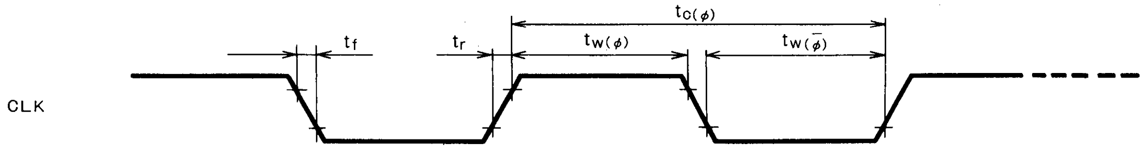
7 : Input pulse level 0.45~2.4V Reference level Input
Input pulse rise time 20ns Output V_{IH} = 2 V, V_{IL} = 0.8 V
Input pulse fall time 20ns Load V_{OH} = 2 V, V_{OL} = 0.8 V



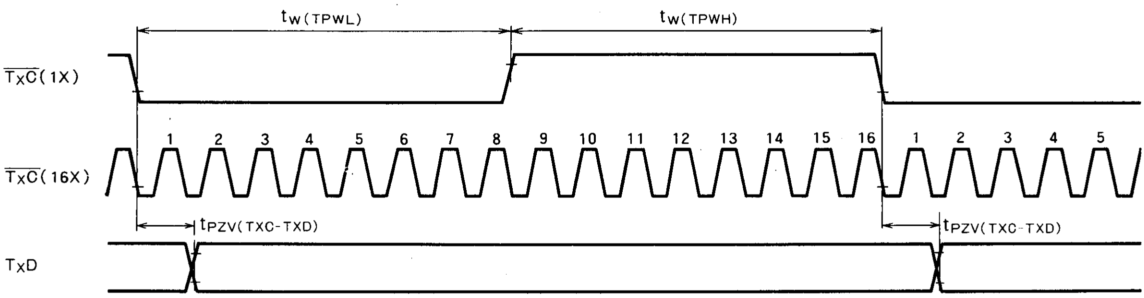


TIMING DIAGRAMS

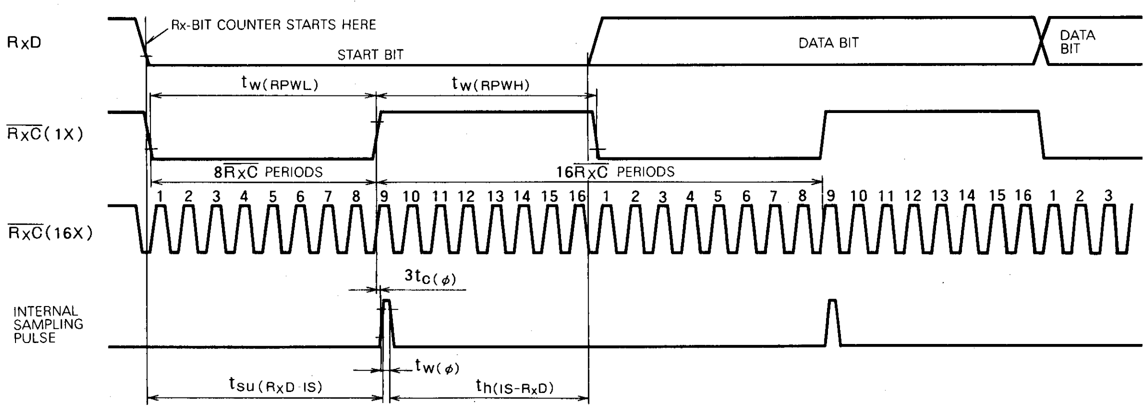
System Clock (CLK)



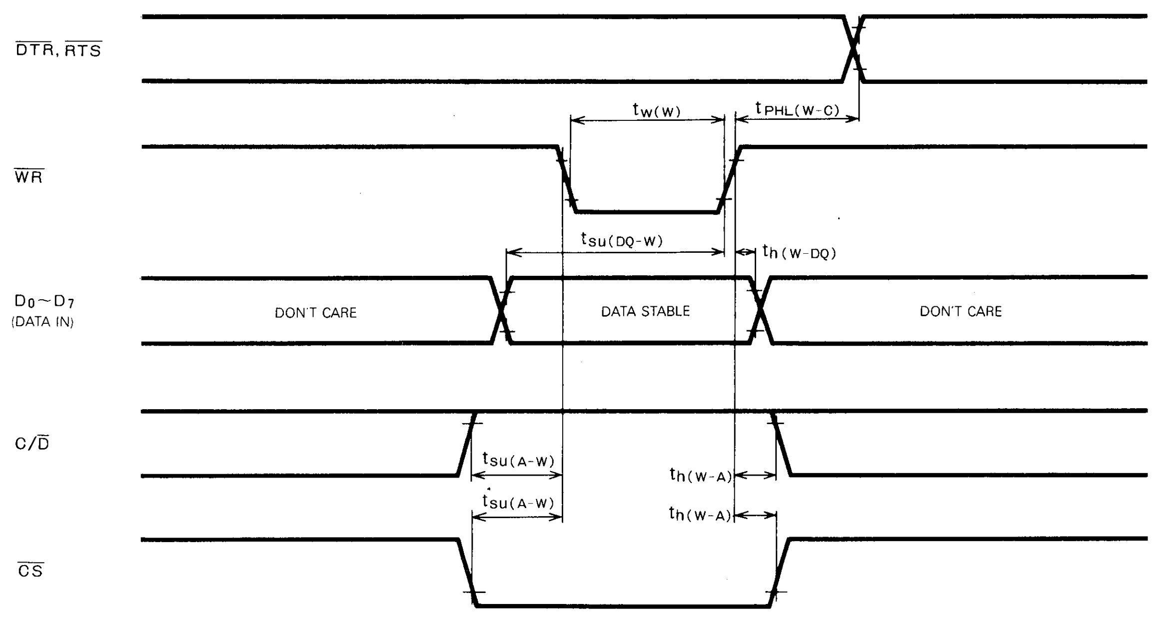
Transmitter Clock & Data



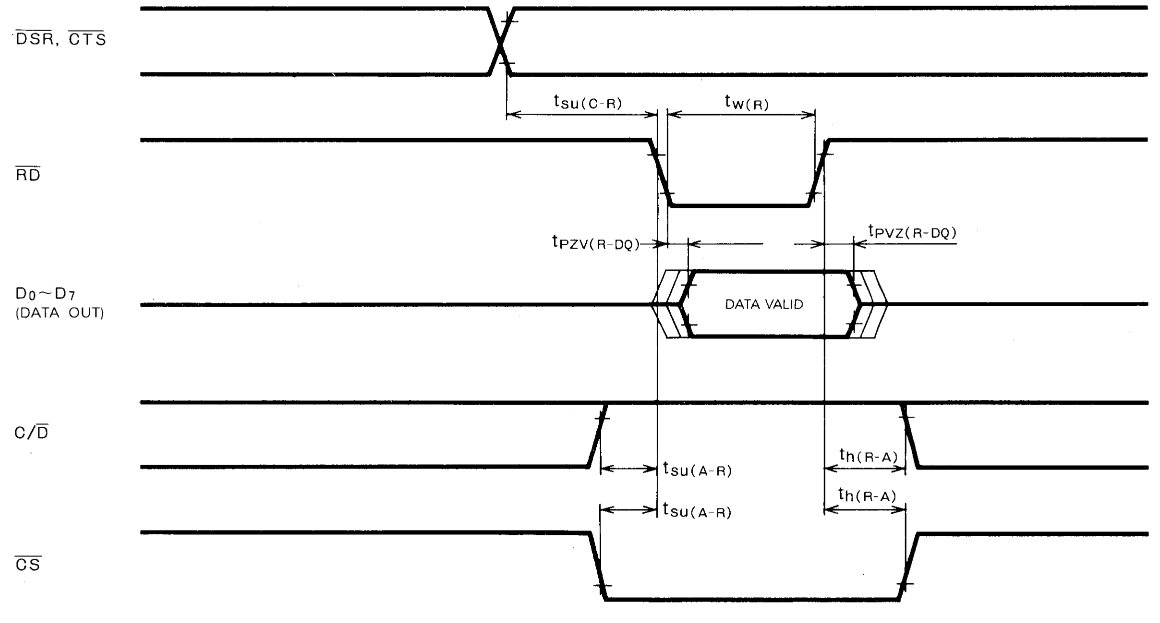
Receiver Clock & Data



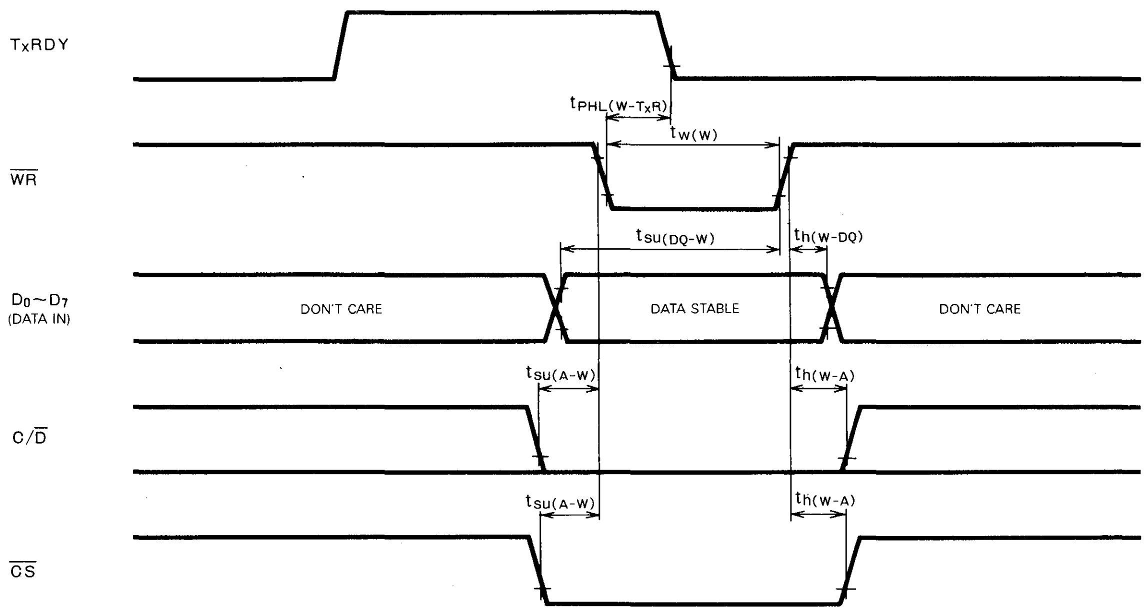
Write Control Cycle (CPU → USART)



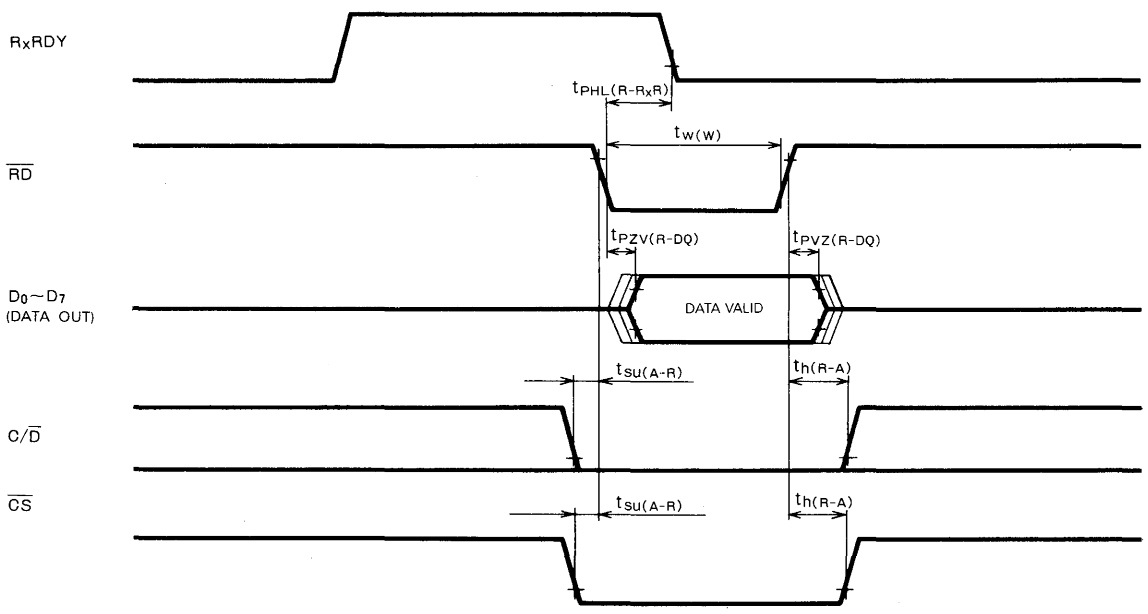
Read Control Cycle (USART→CPU)



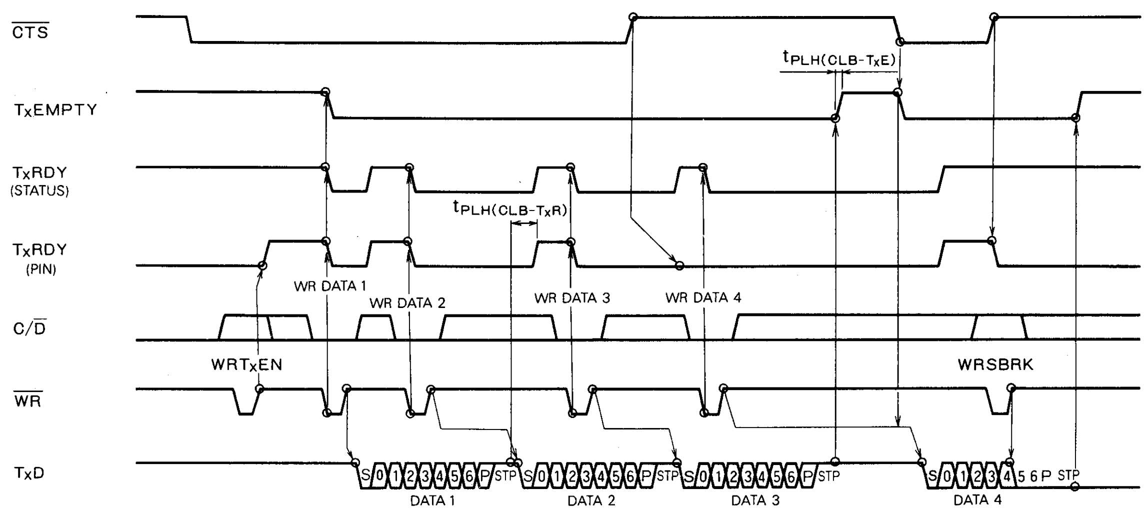
Write Data Cycle (CPU → USART)



Read Data Cycle (USART → CPU)

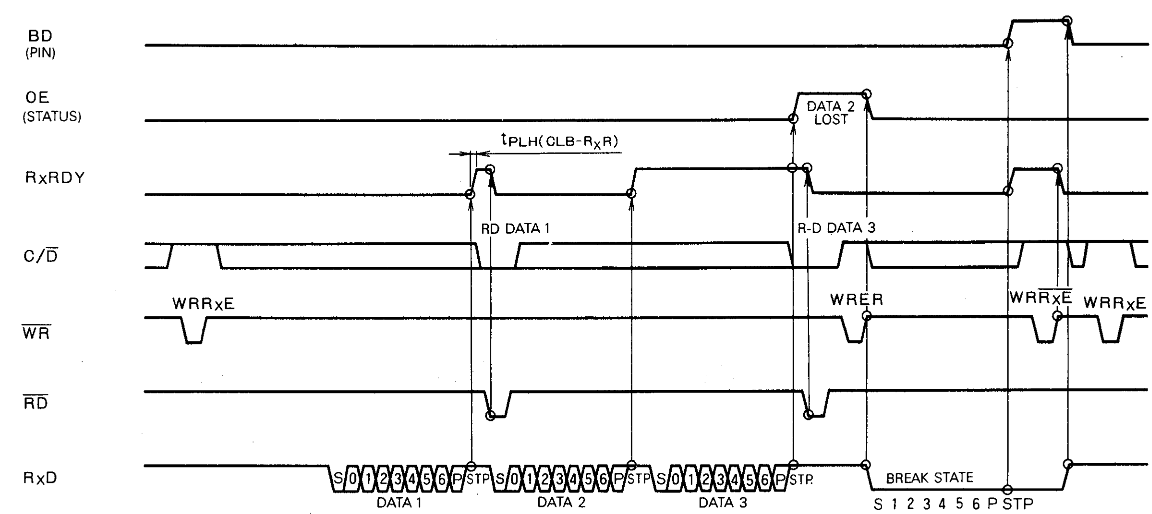


Transmitter Control & Flag Timing (Async Mode)



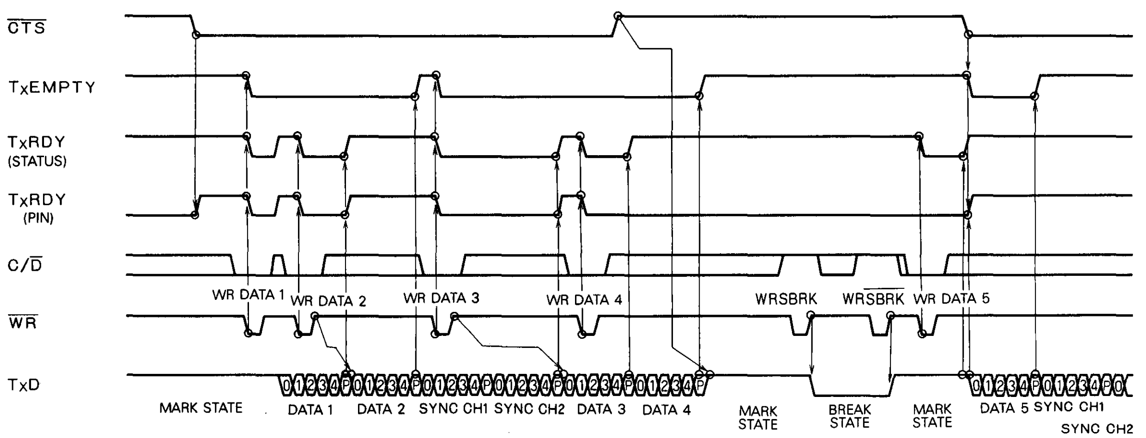
Note 8 : Example format = 7 bits/character with parity & 2 stop bits.
9 : T_xRDY (pin) = 1 ← (Transmit-data buffer is empty) · (T_xEN = 1) · (CTS =0) =1
10 : T_xRDY (status) = 1 ← (Transmit-data buffer is empty) = 1

Receiver Control & Flag Timing (Async Mode)



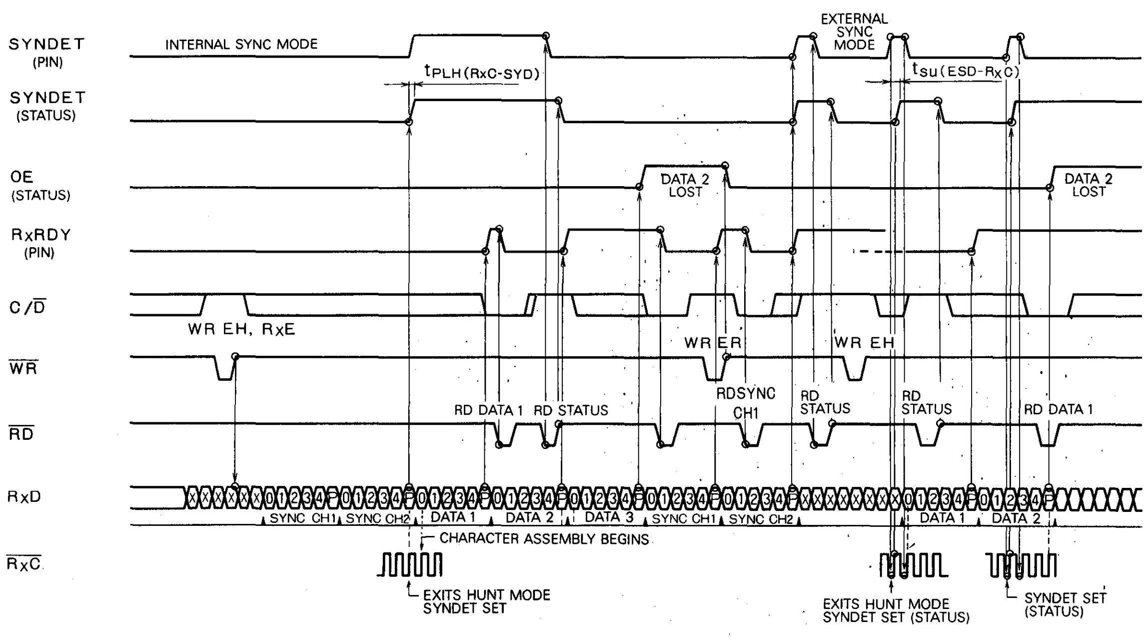
Note 11 : Example format = 7 bits/character with parity & 2 stop bits

Transmitter Control & Flat Timing (Sync Mode)



Note 12 : Example format = 5 bits/character with parity, bi-sync characters.

Receiver Control & Flag Timing (Sync Mode)



Note 13 : Example format = 5 bits/character with parity, bi-sync characters.

MITSUBISHI
ELECTRIC

PROGRAMMABLE INTERVAL TIMER

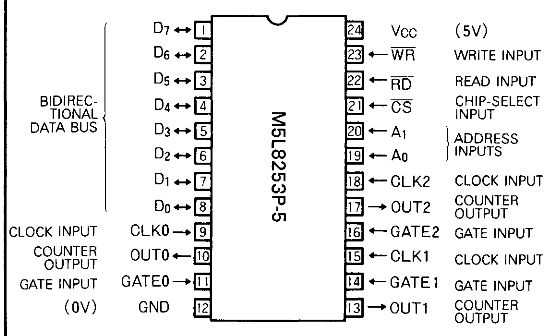
DESCRIPTION

The M5L8253P-5 is a programmable general-purpose timer device developed by using the N-channel silicon-gate ED-MOS process. It offers counter and timer functions in systems using an 8-bit parallel-processing CPU. The use of the M5L 8253P frees the CPU from the execution of looped programs, count-operation programs and other simple processing involving many repetitive operations, thus contributing to improved system throughputs. The M5L 8253P-5 works on a single power supply, and both its input and output can be connected to a TTL circuit.

FEATURES

- M5L 8253P-5 is suitable for use with MELPS 85
- 3 independent built-in 16-bit down counters
- Clock period: DC~2MHz
- 6 counter modes freely assignable for each counter
- Binary or decimal counts
- Single 5V power supply
- Pin connection and electric characteristics compatible with Intel's 8253

PIN CONFIGURATION (TOP VIEW)



Outline 24P1

APPLICATIONS

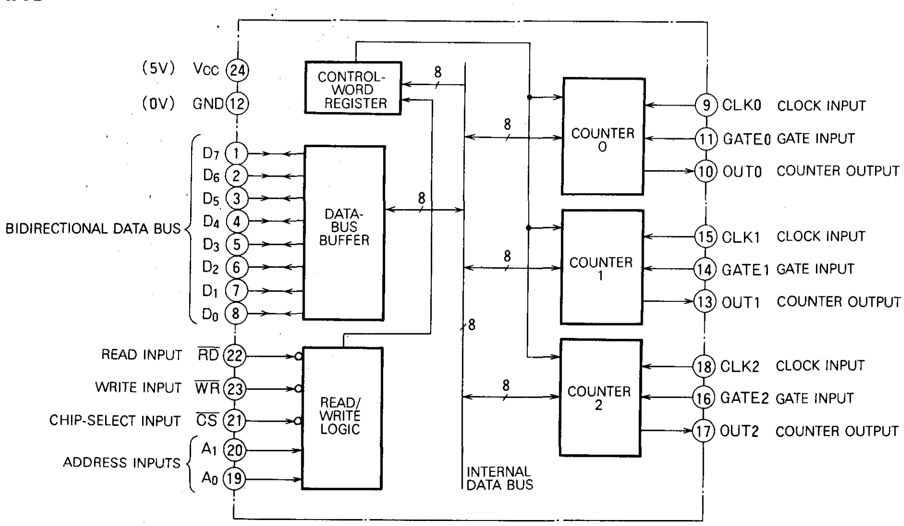
Delayed-time setting, pulse counting and rate generation in microcomputers.

FUNCTION

Three independent 16-bit counters allow free programming based on mode-control instructions from the CPU. When roughly classified, there are 6 modes (0~5). Mode 0 is mainly used as an interruption timer and event counter, mode 1 as a digital one-shot, modes 2 and 3 as rate generators, mode 4 for a software triggered strobe, and mode 5 for a hardware triggered strobe.

The count can be monitored and set at any time. The counter operates with either the binary or BCD system.

BLOCK DIAGRAM



DESCRIPTION OF FUNCTIONS

Data-Bus Buffer

This 3-state, bidirectional, 8-bit buffer is used to interface the M5L8253P-5 to the system-side data bus. Transmission and reception of all the data including control words for mode designation and values written in, and read from, the counters are carried out through this buffer.

Read/Write Logic

The read/write logic accepts control signals ($\overline{RD}$, $\overline{WR}$) from the system and generates control signals for each counter. It is enabled or disabled by the chip-select signal ($\overline{CS}$); if CS is at the high-level the data-bus buffer enters a floating (high-impedance) state.

Read Input ($\overline{RD}$)

The count of the counter designated by address inputs A_0 and A_1 on the low-level is output to the data bus.

Write Input ($\overline{WR}$)

Data on the data bus is written in the counter or control-word register designated by address inputs A_0 and A_1 on the low-level.

Address Inputs (A_0, A_1)

These are used for selecting one of the 3 internal counters and either of the control-word registers.

Chip-Select Input ($\overline{CS}$)

A low-level on this input enables the M5L8253P-5. Changes in the level of the CS input have no effect on the operation of the counters.

Control-Word Register

This register stores information required to give instructions about operational modes and to select binary or BCD counting. Unlike the counters, it allows no reading, only writing.

Counters 0, 1 and 2

These counters are identical in operation and independent of each other. Each is a 16-bit, presettable, down counter, and has clock-input, gate-input and output pins. The counter can operate in either binary or BCD using the falling edge of each clock. The mode of counter operation and the initial value from which to start counting can be designated by software. The count can be read by input instruction at any time, and there is a "read-on-the-fly" function which enables stable reading by latching each instantaneous count to the registers by a special counter-latch instruction.

Table 1 Basic Functions

$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A_1	A_0	Function
0	1	0	0	0	Data bus → Counter 0
0	1	0	0	1	Data bus → Counter 1
0	1	0	1	0	Data bus → Counter 2
0	1	0	1	1	Data bus → Control-word register
0	0	1	0	0	Data bus ← Counter 0
0	0	1	0	1	Data bus ← Counter 1
0	0	1	1	0	Data bus ← Counter 2
0	0	1	1	1	3-state
1	×	×	×	×	3-state
0	1	1	×	×	3-state

CONTROL WORD AND INITIAL-VALUE LOADING

The function of the M5L8253P-5 depends on the system software. The operational mode of the counters can be specified by writing control words (A_0, A_1 = 1, 1) into the control-word registers.

The programmer must write out to the M5L8253P-5 the programmed number of count register bytes (1 or 2) prior to actually using the selected counter.

Table 2 shows control-word format, which consists of 4 fields. Only the counter selected by the D_7 and D_6 bits of the control word is set for operation. Bits D_5 and D_4 are used for specifying operations to read values in the counter and to initialize. Bits D_3~D_1 are used for mode designation, and D_0 for specifying binary or BCD counting. When D_0 = 0, binary counting is employed, and any number from 0000_{16} to $FFFF_{16}$ can be loaded into the count register. The counter is counted down for each clock. The counting of 0000_{16} causes the transmission of a time-out signal from the count-output pin.

The maximum number of counts is obtained when 0000_{16} is set as the initial value. When D_0 = 1, BCD counting is employed, and any number from 0000_{10} to 9999_{10} can be loaded on the counter.

Neither system resetting nor connecting to the power supply sets the control word to any specific value. Thus to bring the counters into operation, the above-mentioned control words for mode designation must be given to each counter, and then 1~2 byte initial counter values must be set. The following is an example of this programming step.

To designate mode 0 for counter 1, with initial value 8253_{16} set by binary count, the following program is used:

MVI	**A , 70_{16}**	Control word 70_{16}
OUT	**n_1**	n_1 is control-word-register address
MVI	**A , 53_{16}**	Low-order 8 bits
OUT	**n_2**	n_2 is counter 1 address
MVI	**A , 82_{16}**	High-order 8 bits
OUT	**n_2**	n_2 is counter 1 address

Thus, the program generally has the following sequence:

(1) Control-word output to counter i (i = 0, 1, 2).
(2) Initialization of low-order 8 counter bits
(3) Initialization of high-order 8 counter bits

The three counters can be executed in any sequence. It is possible, for instance, to designate the mode of each counter and then load initial values in a different order. Initialization of the counters designated by RL1 and RL0 must be executed in the order of the low-order 8 bits and then the high-order 8 bits for the counter in question.

Table 2 Control-Word Format

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC1	SC0	RL1	RL0	M2	M1	M0	BCD
SC		RL		M			BCD

● SC (Select Counter)

SC1	SC0	
0	0	Select counter 0
0	1	Select counter 1
1	0	Select counter 2
1	1	Prohibited combination

● RL (Read/Load)

PL1	RL0	
0	0	Operation
0	1	Read/load low-order 8 bits only
1	0	Read/load high-order 8 bits only
1	1	Read/load low-order 8 bits and then high-order 8 bits

● M (Mode)

M2	M1	M0	
0	0	0	Mode 0
0	0	1	Mode 1
×	1	0	Mode 2
×	1	1	Mode 3
1	0	0	Mode 4
1	0	1	Mode 5

● BCD

0	Binary counter (16 bits)
1	Binary-coded decimal counter (4 decades)

8

MODE DEFINITION

Mode 0 (Interrupt on Terminal Count)

Mode set and initialization cause the counter output to go low-level (see Fig. 1). When the counter is loaded with an initial value, it will start counting the clock input. When the terminal count is reached, the output will go high and remain high until the selected count register is reloaded with the mode. This mode can be used when the CPU is to be interrupted after a certain period or at the time of counting up.

Fig. 1 shows a setting of 4 as the initial value. If gate input goes low, counting is inhibited for the duration of the low-level period.

Reloading of the initial value during count operation will stop counting by the loading of the first byte and start the new count by the loading of the second byte.

Mode 1 (Programmable One-Shot)

The gate input functions as a trigger input. A gate-input rising edge causes the generation of low-level one-shot output with a predetermined clock length starting from the next clock. Fig. 2 shows an initial setting of 4. While the counter output is at the low-level (during one-shot), loading of a new value does not change the one-shot pulse width, which has already been output. The current count can be read at any time without affecting the width of the one-shot pulse being output. This mode permits retriggering.

Mode 2 (Rate Generator)

Low-level pulses during one clock operation are generated from the counter output at a rate of one per n clock inputs (where n is the value initially set for the counter). When a new value is loaded during the counter operation, it is reflected on the output after the pulses by the current count have been output. In the example shown in Fig. 3, n is given as 4 at the outset and is then changed to 3.

In this mode, the gate input provides a reset function. While it is on the low-level, the output is maintained high; the counter restarts from the initial value, triggered by a rising gate-input edge. This gate input, therefore, makes possible external synchronization of the counter by hardware.

After the mode is set, the counter does not start counting until the rate n is loaded into the count register, with the counter output remaining at the high-level.

Mode 3 (Square Rate Generator)

This is similar to Mode 2 except that it outputs a square wave with the half count of the set rate. When the set value n is odd, the square-wave output will be high for $(n + 1)/2$ clock-input counts and low for $(n - 1)/2$ counts. When a new rate is reloaded into the count register during its operation, it is immediately reflected on the count directly following the output transition (high-to-low or low-to-high) of the current count. Gate-input operations are exactly the same as in Mode 2. Fig. 4 shows an example of Mode 3 operation.

Mode 4 (Software Triggered Strobe)

After the mode is set, the output will be high. By loading a number on the counter, however, clock-input counts can be started and on the terminal count, the output will go low for one input-clock period and then will go high again. Mode 4 differs from Mode 2 in that pulses are not output repeatedly with the same set count. The pulse output is delayed one clock period in Mode 2, as shown in Fig. 5. When a new value is loaded into the count register during its count operation, it is reflected on the next pulse output without affecting the current count. The count will be inhibited while the gate input is low-level.

Mode 5 (Hardware Triggered Strobe)

This is a variation of Mode 1. The gate input provides a trigger function, and the count is started by its rising edge. On the terminal count, the counter output goes low for on one clock period and then goes high. As in Mode 1, retriggering by the gate input is possible. An example of timing in Mode 5 is shown in Fig. 6.

As mentioned above, the gate input plays different roles according to the mode. The functions are summarized in Table 3.

Table 3 Gate Operations

Mode \ Gate	Low or going low	Rising	High
0	Disables counting		Enables counting
1		(1) Initiates counting (2) Resets output after next clock	
2	(1) Disables counting (2) Sets output high immediately	Initiates counting	Enables counting
3	(1) Disables counting (2) Sets output high immediately	Initiates counting	Enables counting
4	Disables counting		Enables counting
5		Initiates counting	

Fig. 1 Mode 0

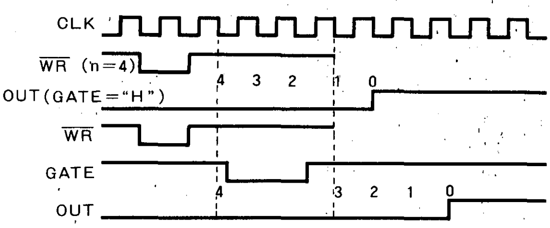


Fig. 2 Mode 1

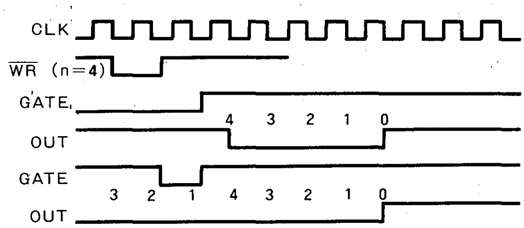


Fig. 3 Mode 2

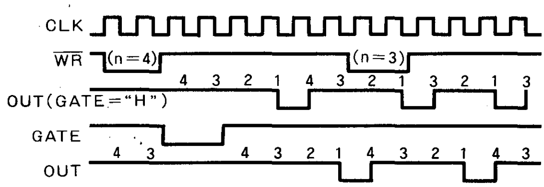


Fig. 4 Mode 3

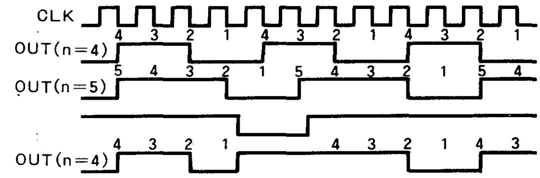


Fig. 5 Mode 4

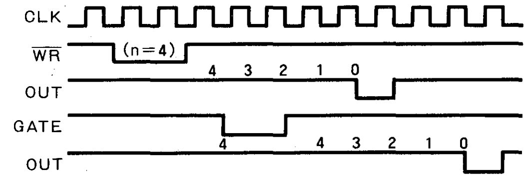
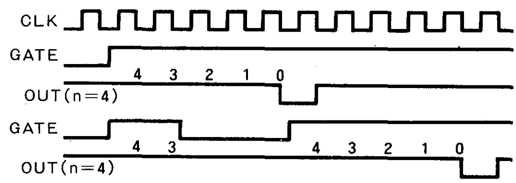


Fig. 6 Mode 5



COUNTER MONITORING

Sometimes the counter must be monitored by reading its count or using it as an event counter. The M5L8253P-5 offers the following two methods for count reading:

Read Operation

The count can be read by designating the address of the counter to be monitored and executing a simple I/O read operation. In order to ensure correct reading of the count, it is necessary to cause the clock input to pause by external logic or prevent a change in the count by gate input. An example of a program to read the counter 1 count is shown below. If RL1, RL0 = 1, 1 has been specified in the control word, the first IN instruction enables the low-order 8 bits to be read and the second IN instruction enables the high-order 8 bits.

IN **n_2** .... n_2 is the counter 1 address
MOV D, A
IN **n_2**
MOV E, A

The IN instruction should be executed once or twice by the RL1 and RL0 designations in the control-word register.

Read-on-the-Fly Operation

This method makes it possible to read the current count without affecting the count operation at all. A special counter-latch command is first written in the control-word register. This causes latching of all the instantaneous counts to the register, allowing retention of stable counts. An example of a program to execute this operation for counter 2 is given below.

MVI A, 1000XXXX.... D_5 = D_4 = 0 designates counter latching
OUT **n_1** .... n_1 is the control-word-register address
IN **n_3** .... n_3 is the counter 2 address
MOV D, A
IN **n_3**
MOV E, A

In this example, the IN instruction is executed twice. Due to the internal logic of the M5L8253P-5 it is absolutely essential to complete the entire reading procedure. If two bytes are programmed to be read, then two bytes must be read before any OUT instruction can be executed to the same counter.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Power supply voltage	With respect to GND	−0.5~7	V
V_I	Input voltage		−0.5~7	V
V_O	Output voltage		−0.5~7	V
Pd	Maximum power dissipation	Ta=25℃	1000	mW
Topr	Operating free-air temperature range		0~70	℃
Tstg	Storage temperature range		−65~150	℃

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Power supply voltage	4.75	5	5.25	V
GND	Supply voltage		0		V
V_{IH}	High-level input voltage	2.2		V_{CC}	V
V_{IL}	Low-level input voltage	−0.5		0.8	V

ELECTRICAL CHARACTERISTICS (Ta=0~70℃, V_{CC}=5V±5%, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	GND=0V (Note 1)	2.4			V
V_{OL}	Low-level output voltage	GND=0V (Note 2)			0.45	V
I_{IH}	High-level input current	GND=0V, V_I=5.25V			±10	μA
I_{IL}	Low-level input current	GND=0V, V_I=0V			±10	μA
I_{OZ}	Off-state output current	GND=0V, V_I=0~V_{CC}			±10	μA
I_{CC}	Power supply current	GND=0V			140	mA
C_i	Input capacitance	V_{IL}=GND, f=1MHz, 25mVrms, Ta=25℃			10	pF
$C_{i/o}$	Input/output capacitance	$V_{I/OL}$=GND, f=1MHz, 25mVrms, Ta=25℃			20	pF

TIMING REQUIREMENTS (Ta=0~70℃, V_{CC}=5V±5%, GND=0V, unless otherwise noted.) (Note 3)

Read Cycle

Symbol	Parameter	Alternative symbol	Test condition	Limits Min	Limits Typ	Limits Max	Unit
$t_{w(R)}$	Read pulse width	t_{RR}	C_L=150pF	300			ns
$t_{su(A-R)}$	Address setup time before read	t_{AR}		50			ns
$t_{h(R-A)}$	Address hold time after read	t_{RA}		5			ns
$t_{rec(R)}$	Read recovery time	t_{RV}		1000			ns

Write Cycle

Symbol	Parameter	Alternative symbol	Test condition	Limits Min	Limits Typ	Limits Max	Unit
$t_{w(W)}$	Write pulse width	t_{WW}	C_L=150pF	300			ns
$t_{su(A-W)}$	Address setup time before write	t_{AW}		50			ns
$t_{h(W-A)}$	Address hold time after write	t_{WA}		30			ns
$t_{su(DQ-W)}$	Data setup time before write	t_{DW}		250			ns
$t_{h(W-DQ)}$	Data hold time after write	t_{WD}		30			ns
$t_{rec(W)}$	Write recovery time	t_{RV}		1000			ns

Clock and Gate Timing

Symbol	Parameter	Alternative symbol	Test condition	Limits Min	Limits Typ	Limits Max	Unit
$t_{w(\phi H)}$	Clock high pulse width	t_{PWH}	C_L=150pF	230			ns
$t_{w(\phi L)}$	Clock low pulse width	t_{PWL}		150			ns
$t_{C(\phi)}$	Clock cycle time	t_{CLK}		380		DC	ns
$t_{w(GH)}$	Gate high pulse width	t_{GW}		150			ns
$t_{w(GL)}$	Gate low pulse width	t_{GL}		100			ns
$t_{su(G-\phi)}$	Gate setup time before clock	t_{GS}		100			ns
$t_{h(\phi-G)}$	Gate hold time after clock	t_{GH}		50			ns

Note 3 : Test conditions: M5L 8253P : C_L= 100pF, M5L 8253P-5 : C_L=150pF

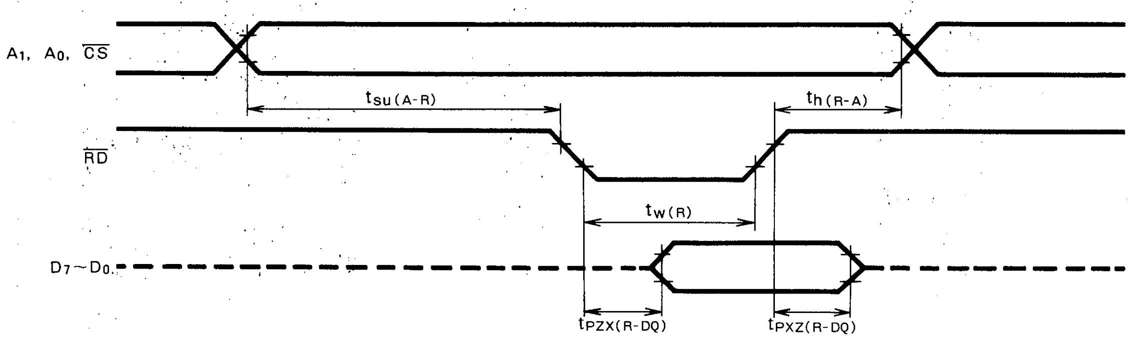
SWITCHING CHARACTERISTICS (Ta = 0 ~ 70°C, V_{CC} = 5V ± 5%, V_{SS} = 0V, unless otherwise noted.) (Note 4)

Symbol	Parameter	Alternative symbol	Test condition	Limits			Unit
				Min	Typ	Max	
$t_{PZX(R-DQ)}$	Propagation time from read to output	t_{RD}	C_L = 150pF			200	ns
$t_{PXZ(R-DQ)}$	Propagation time from read to output floating	t_{DF}		25		100	ns
$t_{PZX(G-DQ)}$	Propagation time from gate to output	t_{ODG}				300	ns
$t_{PZX(\phi-DQ)}$	Propagation time from clock to output	t_{OD}				400	ns

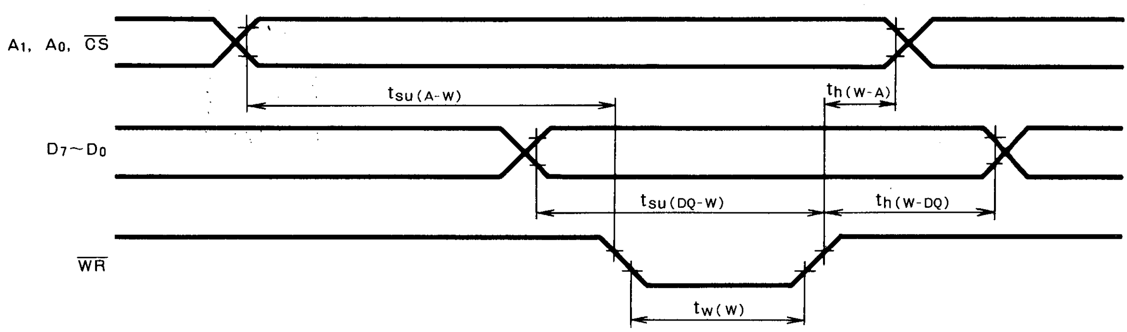
Note 4 : Test conditions: M5L 8253P : C_L = 100pF, M5L 8253P-5 : C_L = 150pF

TIMING DIAGRAMS (Reference Voltage; High = 2.2V, Low = 0.8V)

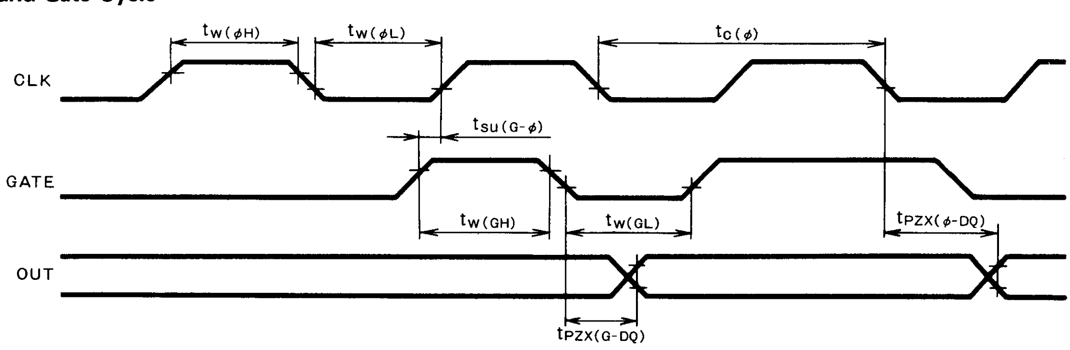
Read Cycle



Write Cycle



Clock and Gate Cycle



MITSUBISHI ELECTRIC

PROGRAMMABLE PERIPHERAL INTERFACE

DESCRIPTION

This is a family of general-purpose programmable input/output devices designed for use with the M5L 8085A 8-bit parallel CPU as input/output ports. These devices are fabricated using N-channel silicon-gate ED-MOS technology for a single supply voltage. They are simple input and output interfaces for TTL circuits, having 24 input/output pins which correspond to three 8-bit input/output ports.

FEATURES

- 24 programmable I/O pins
- Single 5V supply voltage
- TTL-compatible I_{OL} = 2.5mA (max)
- Fully compatible with MELPS 8 microprocessor series
- Direct bit set/reset capability
- Interchangeable with Intel's 8255A in terms of function, electrical characteristics and pin configuration

APPLICATION

- Input/output ports for MELPS 8/85 microprocessor

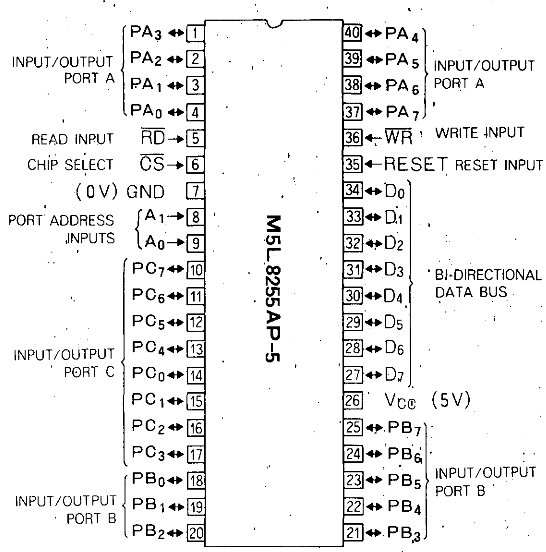
FUNCTION

These PPIs have 24 input/output pins which may be individually programmed in two 12-bit groups A and B with mode control commands from a CPU. They are used in three major modes of operation, mode 0, mode 1 and mode 2.

Operating in mode 0, each group of 12 pins may be programmed in sets of 4 to be inputs or outputs. In mode 1, the 24 I/O terminals may be programmed in two 12-bit groups, group A and group B. Each group contains one 8-bit data port, which may be programmed to serve as input or output, and one 4-bit control port used for handshaking and interrupt control signals. Mode 2 is used with group A only, as one 8-bit bidirectional bus port and one 5-bit control port.

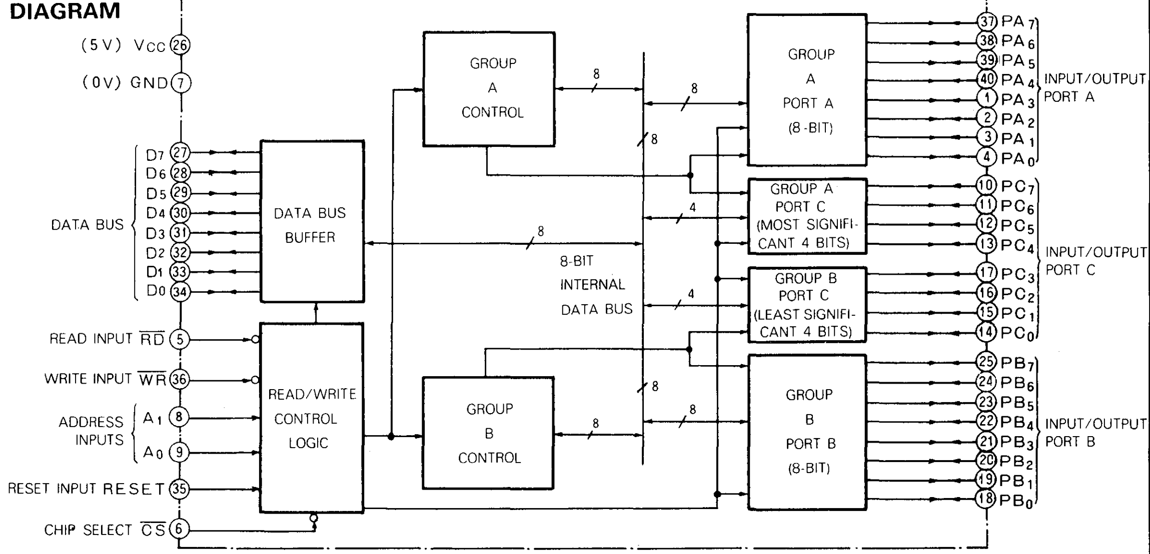
Bit set/reset is controlled by CPU. A high-level reset input (RESET) clears all internal registers, and all ports are set to the input mode (high-impedance state).

PIN CONFIGURATION (TOP VIEW)



Outline 40P1

BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

Data Bus Buffer

This three-state, bidirectional, eight-bit buffer is used to transfer the data when an input or output instruction is executed by the CPU. Control words and status information are also transferred through the data bus buffer.

Read/Write Control Logic

The function of this block is to control transfers of both data and control words. It accepts the address signals (A_0, A_1, $\overline{CS}$) from the CPU, I/O control bus outputs ($\overline{RD}$, $\overline{WR}$) from the system controller, and RESET signals, and then issues commands to both of the control groups in the PPI.

$\overline{CS}$ (Chip-Select) Input

At low-level, the communication between the PPI and the CPU is enabled. While at high-level, the data bus is kept in the high-impedance state, so that commands from the CPU are ignored. Then the previous data is kept at the output port.

$\overline{RD}$ (Read) Input

At low-level, the status or data at the port is transferred to the CPU from the PPI. In essence, it allows the CPU to read data from the PPI.

$\overline{WR}$ (Write) Input

At low-level, the data or control words are transferred from the CPU and written in the PPI.

A_0, A_1 (Port Address) Input

These input signals are used to select one of the three ports: port A, port B, and port C, or the control register. They are normally connected to the least significant two bits of the address bus.

RESET (Reset) Input

At high-level, all internal registers, including the control register, are cleared. Then all ports are set to the input mode (high-impedance state).

Group A and Group B Control

Accepting commands from the read/write control logic, the control blocks (Group A, Group B) receive 8-bit control words from the internal data bus and issue the proper commands for the associated ports. Control group A is accociated with port A and the four high-order bits of port C. Control group B is associated with port B and the four low-order bits of port C. The control register, which stores control words, can only be written into.

Port A, Port B and Port C

The PPI contains three 8-bit ports whose modes and input/output settings are programmed by the system software.

Port A has an output latch/buffer and an input latch. Port B has an I/O latch/buffer and an input buffer. Port C has an output latch/buffer and an input buffer. Port C can be divided into two 4-bit ports which can be used as ports for control signals for port A and port B.

The basic operations are shown in Table 1.

Table 1 Basic Operations

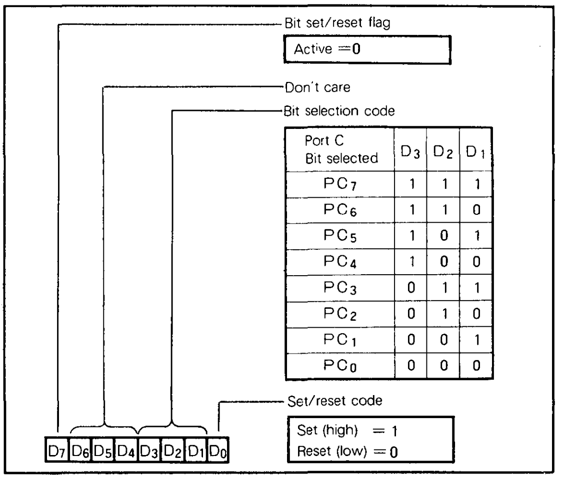
A_1	A_0	$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	Operation
0	0	0	0	1	Data bus ← Port A
0	1	0	0	1	Data bus ← Port B
1	0	0	0	1	Data bus ← Port C
0	0	0	1	0	Port A ← Data bus
0	1	0	1	0	Port B ← Data bus
1	0	0	1	0	Port C ← Data bus
1	1	0	1	0	Control register ← Data bus
X	X	1	X	X	Data bus is in high-impedance state.
1	1	0	0	1	Illegal condition

Where, "0" indicates low level
"1" indicates high level

Bit Set/Reset

When port C is used as an output port, any one bit of the eight bits can be set (high) or reset (low) by a control word from the CPU. This bit set/reset can be operated in the same way as the mode set, but the control word format is different. This operation is also used for INTE set/reset in mode 1 and mode 2.

Fig. 1 Control word format for port C set/reset



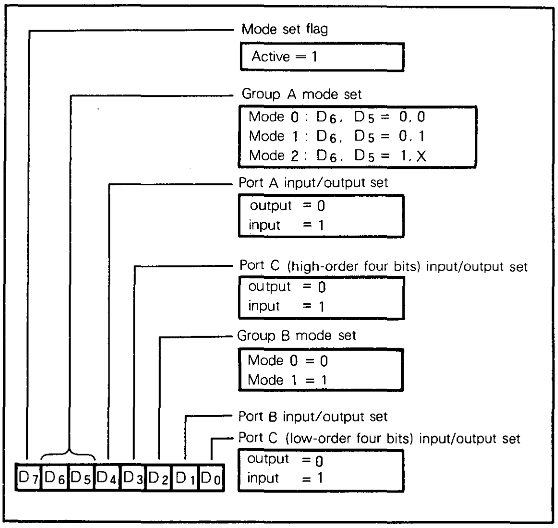
BASIC OPERATING MODES

The PPI can operate in any one of three selected basic modes.

Mode 0: Basic input/output (group A, group B)
Mode 1: Strobed input/output (group A, group B)
Mode 2: Bidirectional bus (group A only)

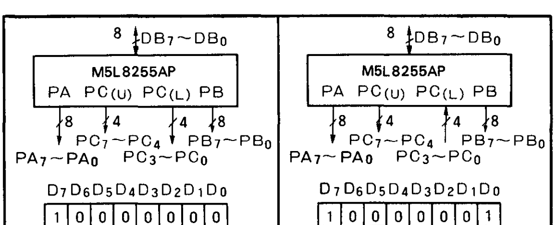
The mode of both group A and group B can be selected independently. The control word format for mode set is shown in Fig. 2.

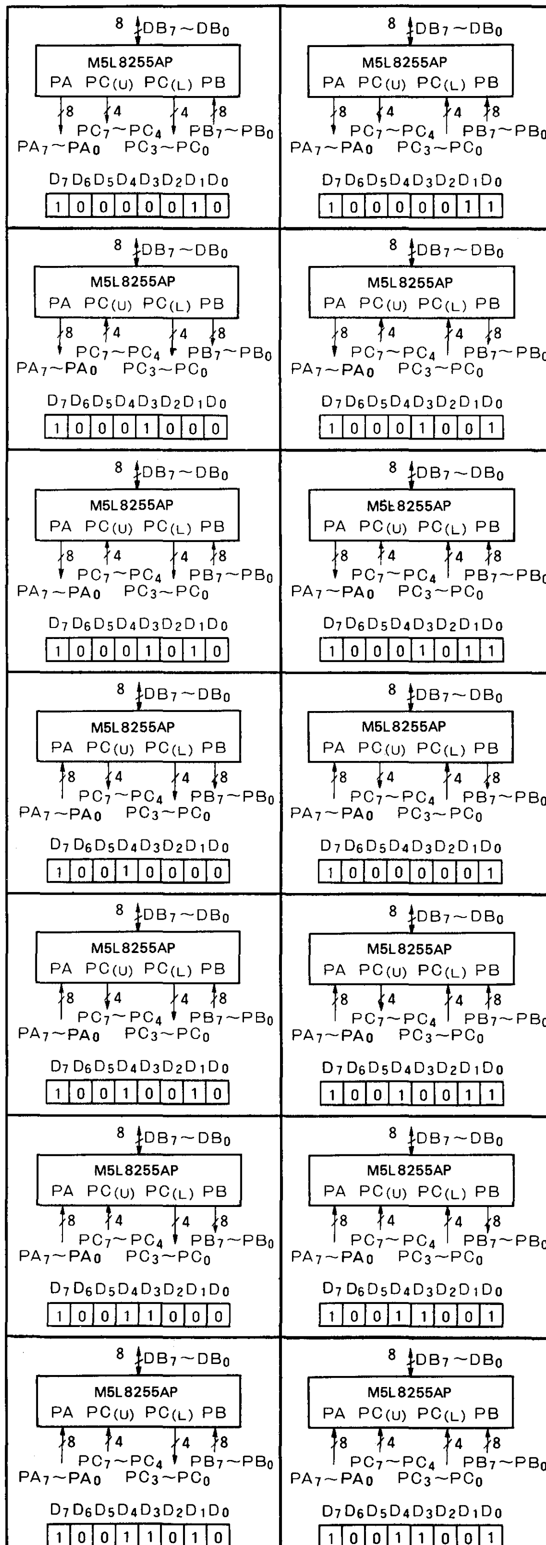
Fig. 2 Control word format for mode set.



1. Mode 0 (Basic Input/Output)

This functional configuration provides simple input and output operations for each of the three ports. No "hand-shaking" is required; data is simply written in, or read from, the specified port. Output data from the CPU to the port can be held, but input data from the port to the CPU cannot be held. Any one of the 8-bit ports and 4-bit ports can be used as an input port or an output port. The diagrams following show the basic input/output operating modes.





8

2. Mode 1 (Strobed Input/Output)

This function can be set in both group A and B. Both groups are composed of one 8-bit data port and one 4-bit control data port. The 8-bit port can be used as an input port or an output port. The 4-bit port is used for control and status signals affecting the 8-bit data port. The following shows operations in mode 1 for using input ports.

$\overline{STB}$ (Strobed Input)

A low-level on this input latches the output data from the terminal units into the input register of the port. In short, this is a lock for data latching. The data from the terminal units can be latched by the PPI independent of the control signal from the CPU. This data is not sent to the data bus until the instruction IN is executed.

IBF (Input Buffer Full Flag Output)

A high-level on this output indicates that the data from the terminal units has been latched into the input register. IBF is set to high-level by the falling edge of the $\overline{STB}$ input, and is reset to low-level by the rising edge of the $\overline{RD}$ input.

INTR (Interrupt Request Output)

This can be used to interrupt the CPU when an input device is requesting service. When INTE (interrupt enable flag) of the PPI is high-level, INTR is set to high-level by the rising edge of the $\overline{STB}$ input and is reset to low-level by the falling edge of $\overline{RD}$ input.

$INTE_A$ of group A is controlled by bit setting of PC_4. $INTE_B$ of group B is controlled by bit setting of PC_2.

Mode 1 input state is shown in Fig. 3, and the timing chart is shown in Fig. 4.

Fig. 3 An example of mode 1 input state

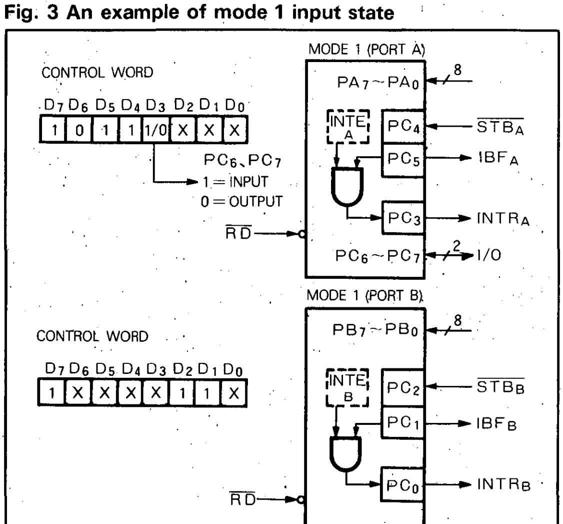
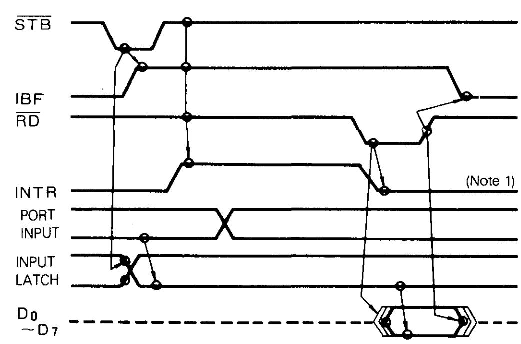


Fig. 4 Timing chart



Note 1 : When INTE is low-level, INTR is always low-level.

The following shows operations using mode 1 for output ports.

$\overline{OBF}$ (Output Buffer Full Flag Output)

This is reset to low-level by the rising edge of the $\overline{WR}$ signal and is set to high-level by the falling edge of the $\overline{ACK}$ (acknowledge input). In essence, the PPI indicates to the terminal units by the $\overline{OBF}$ signal that the CPU has sent data to the port.

$\overline{ACK}$ (Acknowledge Input)

Receiving this signal from a terminal unit can indicate to the PPI that the terminal unit has accepted data from a port.

INTR (Interrupt Request)

When a peripheral unit is accepting data from the CPU, setting INTR to high-level can be used to interrupt the CPU. When INTE (interrupt enable flag) is high and $\overline{OBF}$ is set to high-level by the rising edge of an $\overline{ACK}$ signal, then $\overline{INTR}$ will also be set to high-level by the rising edge of the $\overline{ACK}$ signal. Also, $\overline{INTR}$ is reset to low-level by the falling edge of the $\overline{WR}$ signal when the PPI has been receiving data from the CPU.

$INTE_A$ of group A is controlled by bit setting of PC_6.

$INTE_B$ of group B is controlled by bit setting of PC_2.

Mode 1 output state is shown in Fig. 5, and the timing chart is shown in Fig. 6.

Combinations for using port A and port B as input or output in mode 1 are shown in Fig. 7 and Fig. 8.

Fig. 5 Mode 1 output example

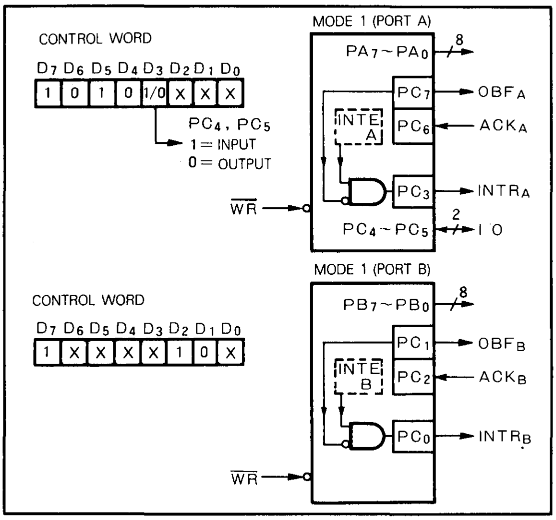


Fig. 6 Timing diagram

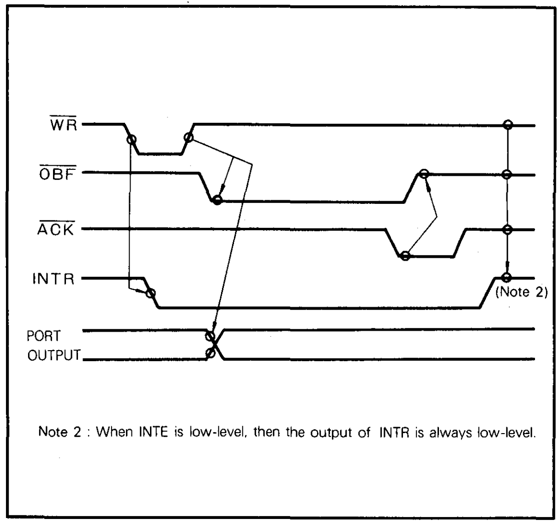


Fig. 7 Mode 1 port A and port B I/O example

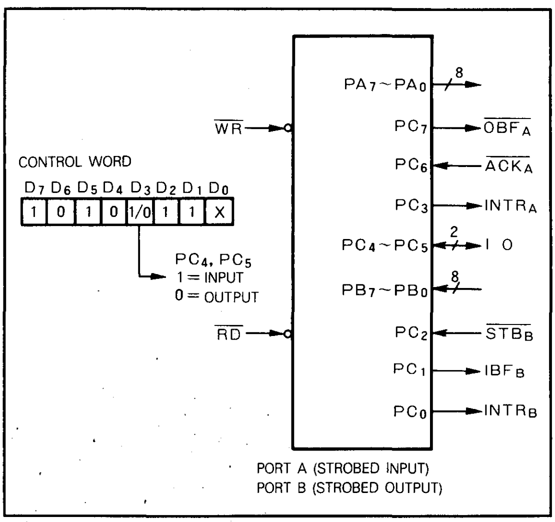
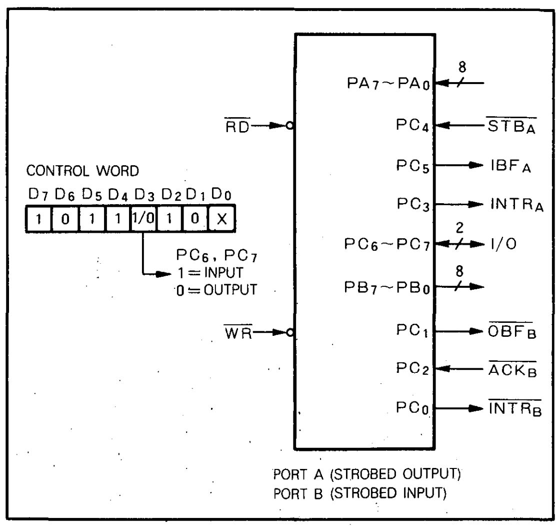


Fig. 8 Mode 1 port A and port B I/O example



3. Mode 2 (Strobed Bidirectional Bus Input/Output)

Mode 2 can provide bidirectional operations, using one 8-bit bus for communicating with terminal units. Mode 2 is only valid with group A and uses one 8-bit bidirectional bus port (port A) and a 5-bit control port (high-order five bits of port C). The bus port (port A) has two internal registers, one for input and the other for output. On the other hand, the control port (port C) is used for communicating control signals and bus-status signals. These control signals are similar to mode 1 and can also be used to control interruption of the CPU. When group A is programmed as mode 2, group B can be programmed independently as mode 0 or mode 1. When group A is in mode 2, the following five control signals can be used.

$\overline{OBF}$ (Output Buffer Full Flag Output)
The $\overline{OBF}$ output will go low-level to indicate that the CPU has sent data to the internal register of port A. This signal lets the terminal units know that the data is ready for transfer from the CPU. When this occurs, port A remains in the floating (high-impedance) state.

$\overline{ACK}$ (Acknowledge Input)
A low-level $\overline{ACK}$ input will cause the data of the internal register to be transferred to port A. For a high-level ACK input, the output buffer will be in the floating (high-impedance) state.

$\overline{STB}$ (Strobed Input)
When the $\overline{STB}$ input is low-level, the data from terminal units will be held in the internal register; and the data will be sent to the system data bus with an $\overline{RD}$ signal to the PPI.

IBF (Input Buffer Full Flag Output)
When data from terminal units is held on the internal register, IBF will be high level.

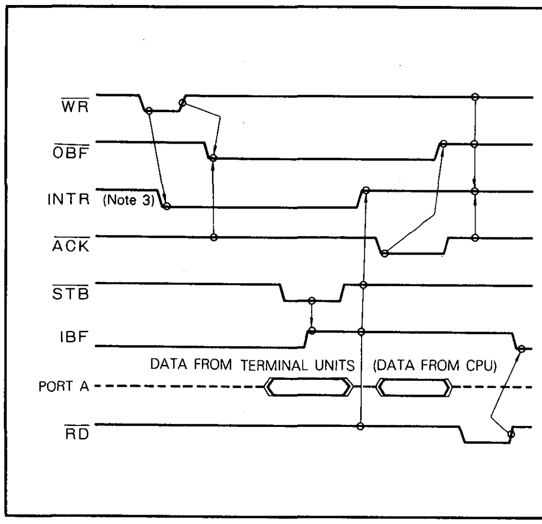
$\overline{INTR}$ (Interrupt Request Output)
This output is used to interrupt the CPU and its operations the same as in mode 1. There are two interrupt enable flags that correspond to $INTE_A$ for mode 1 output and mode 1 input.

$INTE_1$ is used in generating INTR signals in combination with $\overline{OBF}$ and $\overline{ACK}$. $INTE_1$ is controlled by bit setting of PC_6.

$INTE_2$ is used in generating INTR signals in combination with $\overline{IBF}$ and $\overline{STB}$. $INTE_2$ is controlled by bit setting of PC_4.

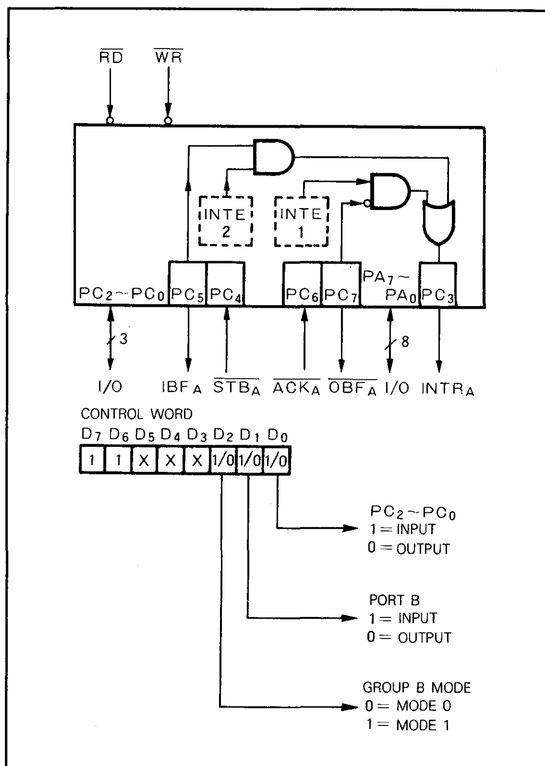
Fig. 9 shows the timing diagram of mode 2, and Fig. 10 is an example of mode 2 operation.

Fig. 9 Mode 2 timing diagram



Note 3 : $INTR = IBF \cdot \overline{MASK} \cdot \overline{STB} \cdot \overline{RD} + \overline{OBF} \cdot \overline{MASK} \cdot \overline{ACK} \cdot \overline{WR}$

Fig. 10 An example of mode 2 operation



4. Control Signal Read

In mode 1 or mode 2 when using port C as a control port, by CPU execution of an IN instruction, each control signal and bus status from port C can be read.

5. Control Word Tables

Control word formats and operation details for mode 0, mode 1, mode 2 and set/reset control of port C are given in Tables 3, 4, 5 and 6, respectively.

Table 2 Read-out control signals

Mode \ Data	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Mode 1, input	I O	I O	IBF_A	$INTE_A$	$INTR_A$	$INTE_B$	IBF_B	$INTR_B$
Mode 1, output	$\overline{OBF_A}$	$INTE_A$	I O	I O	$INTR_A$	$INTE_B$	$\overline{OBF_B}$	$INTR_B$
Mode 2	$\overline{OBF_A}$	INTE 1	IBF_A	INTE 2	$INTR_A$	By group B mode		

Table 3 Mode 0 control words

Control words									Group A		Group B	
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal	Port A	Port C (high order 4 bits)	Port C (low order 4 bits)	Port B
1	0	0	0	0	0	0	0	8 0	OUT	OUT	OUT	OUT
1	0	0	0	0	0	0	1	8 1	OUT	OUT	IN	OUT
1	0	0	0	0	0	1	0	8 2	OUT	OUT	OUT	IN
1	0	0	0	0	0	1	1	8 3	OUT	OUT	IN	IN
1	0	0	0	1	0	0	0	8 8	OUT	IN	OUT	OUT
1	0	0	0	1	0	0	1	8 9	OUT	IN	IN	OUT
1	0	0	0	1	0	1	0	8 A	OUT	IN	OUT	IN
1	0	0	0	1	0	1	1	8 B	OUT	IN	IN	IN
1	0	0	1	0	0	0	0	9 0	IN	OUT	OUT	OUT
1	0	0	1	0	0	0	1	9 1	IN	OUT	IN	OUT
1	0	0	1	0	0	1	0	9 2	IN	OUT	OUT	IN
1	0	0	1	0	0	1	1	9 3	IN	OUT	IN	IN
1	0	0	1	1	0	0	0	9 8	IN	IN	OUT	OUT
1	0	0	1	1	0	0	1	9 9	IN	IN	IN	OUT
1	0	0	1	1	0	1	0	9 A	IN	IN	OUT	IN
1	0	0	1	1	0	1	1	9 B	IN	IN	IN	IN

Note 4 : OUT indicates output port, and IN indicates input port.

Table 4 Mode 1 control words

Control words		Group A						Group B			
			Port C					Port C			
D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Hexa-decimal	Port A	PC_7	PC_6	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0	Port B
1 0 1 0 0 1 0 X	A 4 A 5	OUT	$\overline{OBF_A}$	$\overline{ACK_A}$	OUT		$INTR_A$	$\overline{ACK_B}$	$\overline{OBF_B}$	$INTR_B$	OUT
1 0 1 0 0 1 1 X	A 6 A 7	OUT	$\overline{OBF_A}$	$\overline{ACK_A}$	OUT		$INTR_A$	$\overline{STB_B}$	IBF_B	$INTR_B$	IN
1 0 1 0 1 1 0 X	AC AD	OUT	$\overline{OBF_A}$	$\overline{ACK_A}$	IN		$INTR_A$	$\overline{ACK_B}$	$\overline{OBF_B}$	$INTR_B$	OUT
1 0 1 0 1 1 1 X	AE AF	OUT	$\overline{OBF_A}$	$\overline{ACK_A}$	IN		$INTR_A$	$\overline{STB_B}$	IBF_B	$INTR_B$	IN
1 0 1 1 0 1 0 X	B 4 B 5	IN	OUT		IBF_A	$\overline{STB_A}$	$INTR_A$	$\overline{ACK_B}$	$\overline{OBF_B}$	$INTR_B$	OUT
1 0 1 1 0 1 1 X	B 6 B 7	IN	OUT		IBF_A	$\overline{STB_A}$	$INTR_A$	$\overline{STB_B}$	IBF_B	$INTR_B$	IN
1 0 1 1 1 1 0 X	BC BD	IN	IN		IBF_A	$\overline{STB_A}$	$INTR_A$	$\overline{ACK_B}$	$\overline{OBF_B}$	$INTR_B$	OUT
1 0 1 1 1 1 1 X	BE BF	IN	IN		IBF_A	$\overline{STB_A}$	$INTR_A$	$\overline{STB_B}$	IBF_B	$INTR_B$	IN

Note 5 : Mode of group A and group B can be programmed independently.

6 : It is not necessary for both group A and group B to be in mode 1.

8

Table 5 Mode 2 control words

Control words		Group A						Group B			
D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Hexa-decimal (Ex.)	Port A	Port C					Port C			Port B
			PC_7	PC_6	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0	
1 1 X X X 0 0 0	C0	Bidirectional bus	OBF_A	ACK_A	IBF_A	STB_A	$INTR_A$	OUT			OUT
1 1 X X X 0 0 1	C1	Bidirectional bus	OBF_A	ACK_A	IBF_A	STB_A	$INTR_A$	IN			OUT
1 1 X X X 0 1 0	C2	Bidirectional bus	OBF_A	ACK_A	IBF_A	STB_A	$INTR_A$	OUT			IN
1 1 X X X 0 1 1	C3	Bidirectional bus	OBF_A	ACK_A	IBF_A	STB_A	$INTR_A$	IN			IN
1 1 X X X 1 0 X	C4	Bidirectional bus	OBF_A	ACK_A	IBF_A	STB_A	$INTR_A$	$\overline{ACK_B}$	$\overline{OBF_B}$	$INTR_B$	OUT
1 1 X X X 1 1 X	C6	Bidirectional bus	OBF_A	ACK_A	IBF_A	STB_A	$INTR_A$	$\overline{STB_B}$	IBF_B	$INTR_B$	IN

Table 6 Port C set/reset control words

Control words		Port C								Remarks
D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Hexa-decimal	PC_7	PC_6	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0	
0 X X X 0 0 0 0	00								0	
0 X X X 0 0 0 1	01								1	
0 X X X 0 0 1 0	02							0		
0 X X X 0 0 1 1	03							1		
0 X X X 0 1 0 0	04						0			$INTE_B$ set/reset for mode 1 input
0 X X X 0 1 0 1	05						1			$INTE_B$ set/reset for mode 1 output
0 X X X 0 1 1 0	06					0				
0 X X X 0 1 1 1	07					1				
0 X X X 1 0 0 0	08				0					$INTE_A$ set/reset for mode 1 input
0 X X X 1 0 0 1	09				1					$INTE_2$ set/reset for mode 2
0 X X X 1 0 1 0	0A			0						
0 X X X 1 0 1 1	0B			1						
0 X X X 1 1 0 0	0C		0							$INTE_A$ set/reset for mode 1 output
0 X X X 1 1 0 1	0D		1							$INTE_1$ set/reset for mode 2
0 X X X 1 1 1 0	0E	0								
0 X X X 1 1 1 1	0F	1								

Note 7 : The terminals of port C should be programmed for the output mode, before the bit set/reset operation is executed.

8 : Also used for controlling the interrupt enable flag (INTE)

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	− 0.5~7	V
V_I	Input voltage		− 0.5~7	V
V_O	Output voltage		− 0.5~7	V
P_d	Power dissipation	T_a=25°C	1000	mW
T_{opr}	Operating free-air temperature range		0~70	°C
T_{stg}	Storage temperature range		−65~150	°C

RECOMMENDED OPERATING CONDITIONS (Ta=0~70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
GND	Supply voltage		0		V
V_{IH}	High-level input voltage	2		V_{CC}	V
V_{IL}	Low-level input voltage	−0.5		0.8	V

ELECTRICAL CHARACTERISTICS (Ta=0~70°C, V_{CC}=5V±5%, GND=0V, unless otherwise noted)

Symbol	Parameter		Test conditions		Limits			Unit
					Min	Typ	Max	
V_{OH}	High-level output voltage	Data bus	GND=0V	I_{OH}=−400μA	2.4			V
		Port		I_{OH}=−200μA				
V_{OL}	Low-level output voltage	Data bus	GND=0V	I_{OL}=2.5mA			0.45	V
		Port		I_{OL}=1.7mA				
I_{OH}	High-level output current (Note 10)		GND=0V, V_{OH}=1.5V, R_{EXT}=750Ω		−1		−4	mA
I_{CC}	Supply current from V_{CC}		GND=0V				120	mA
I_{IH}	High-level input voltage		GND=0V, V_I=5.25V				±10	μA
I_{IL}	Low-level input voltage		GND=0V, V_I=0V				±10	μA
I_{OZ}	Off-state output current		GND=0V, V_I=0~5.25V				±10	μA
C_i	Input capacitance		V_{IL}=GND, f=1MHz, 25mVrms Ta=25°C				10	pF
$C_{i/o}$	Input/output terminal capacitance		$V_{I/OL}$=GND, f=1MHz, 25mVrms Ta=25°C				20	pF

Note 9 : Current flowing into an IC is positive; out is negative.
10 : It is valid only for any 8 input/output pins of PB and PC.

TIMING REQUIREMENTS (Ta=0~70°C, V_{CC}=5V±5%, GND=0V, unless otherwise noted)

Symbol	Prameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{w(R)}$	Read pulse width	t_{RR}		300			ns
$t_{su(PER)}$	Peripheral setup time before read	t_{IR}		0			ns
$t_{h(R-PE)}$	Peripheral hold time after read	t_{HR}		0			ns
$t_{su(A-R)}$	Address setup time before read	t_{AR}		0			ns
$t_{h(R-A)}$	Address hold time after read	t_{RA}		0			ns
$t_{w(W)}$	Write pulse width	t_{WW}		300			ns
$t_{su(DQ-W)}$	Data setup time before write	t_{DW}		100			ns
$t_{h(W-DQ)}$	Data hold time after write	t_{WD}		50			ns
$t_{su(A-W)}$	Address setup time before write	t_{AW}		0			ns
$t_{h(W-A)}$	Address hold time after write	t_{WA}		40			ns
$t_{w(ACK)}$	Acknowledge pulse width	t_{AK}		300			ns
$t_{w(STB)}$	Strobe pulse width	t_{ST}		500			ns
$t_{su(PE-STB)}$	Peripheral setup time before strobe	t_{PS}		0			ns
$t_{h(STB-PE)}$	Peripheral hold time after strobe	t_{PH}		180			ns
$t_{c(RW)}$	Read/write cycle time	t_{RV}		850			ns

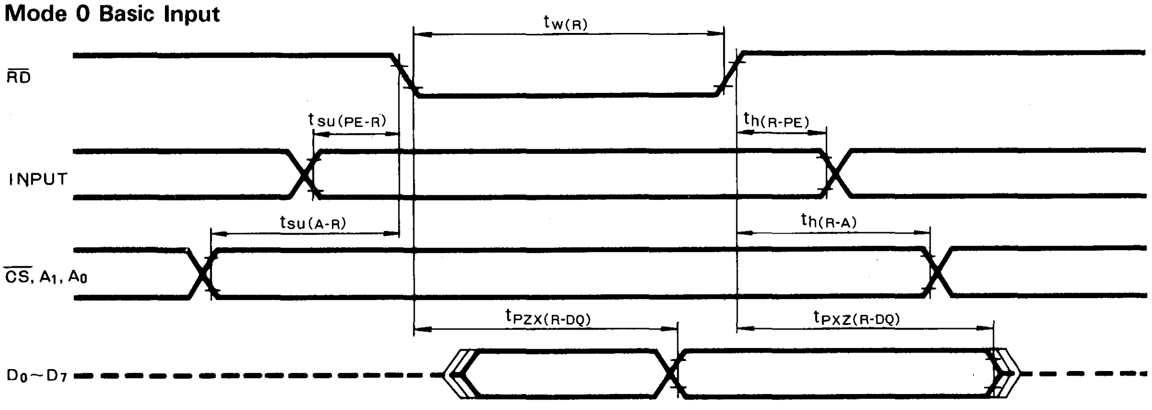
SWITCHING CHARACTERISTICS (Ta = 0 ~ 70°C, Vcc = 5V ±5%, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{PZX(R-DQ)}$	Propagation time from read to data output	t_{RD}				200	ns
$t_{PXZ(R-DQ)}$	Propagation time from read to data floating (Note 12)	t_{DF}				100	ns
$t_{PHL(W-PE)}$ $t_{PLH(W-PE)}$	Propagation time from write to output	t_{WB}				350	ns
$t_{PLH(STB-IBF)}$	Propagation time from strobe to IBF flag	t_{SIB}				300	ns
$t_{PLH(STB-INTR)}$	Propagation time from strobe to interrupt	t_{SIT}				300	ns
$t_{PHL(R-INTR)}$	Propagation time from read to interrupt	t_{RIT}				400	ns
$t_{PHL(R-IBF)}$	Propagation time from read to IBF flag	t_{RIB}				300	ns
$t_{PHL(W-INTR)}$	Propagation time from write to interrupt	t_{WIT}				850	ns
$t_{PHL(W-OBF)}$	Propagation time from write to $\overline{OBF}$ flag	t_{WOB}				650	ns
$t_{PLH(ACK-OBF)}$	Propagation time from acknowledge to $\overline{OBF}$ flag	t_{AOB}				350	ns
$t_{PLH(ACK-INTR)}$	Propagation time from acknowledge to interrupt	t_{AIT}				350	ns
$t_{PZX(ACK-PE)}$	Propagation time from acknowledge to data output	t_{AD}				300	ns
$t_{PXZ(ACK-PE)}$	Propagation time from acknowledge to data output (Note 11)	t_{KD}		20		250	ns

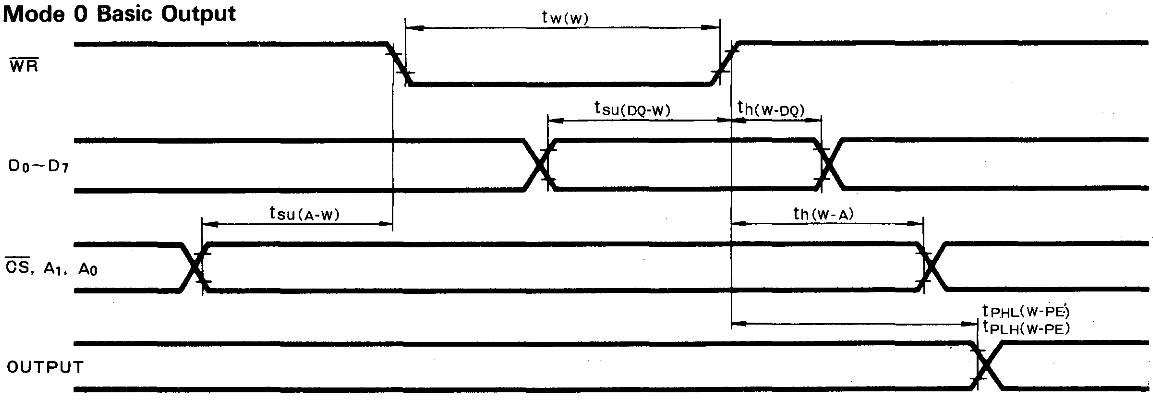
Note 11 : Measurement conditions:
C_L = 100pF for M5L8255AP, S
C_L = 150pF for M5L8255AP-5, S-5
12 : Measurement conditions of note 11 are not applied.

TIMING DIAGRAMS REFERENCE LEVEL = "H"=2V, "L"=0.8V

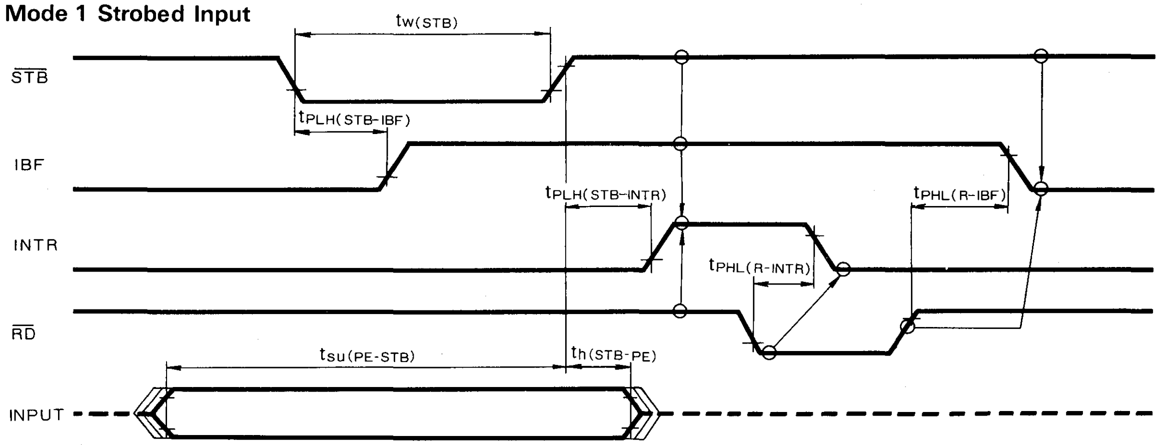
Mode 0 Basic Input



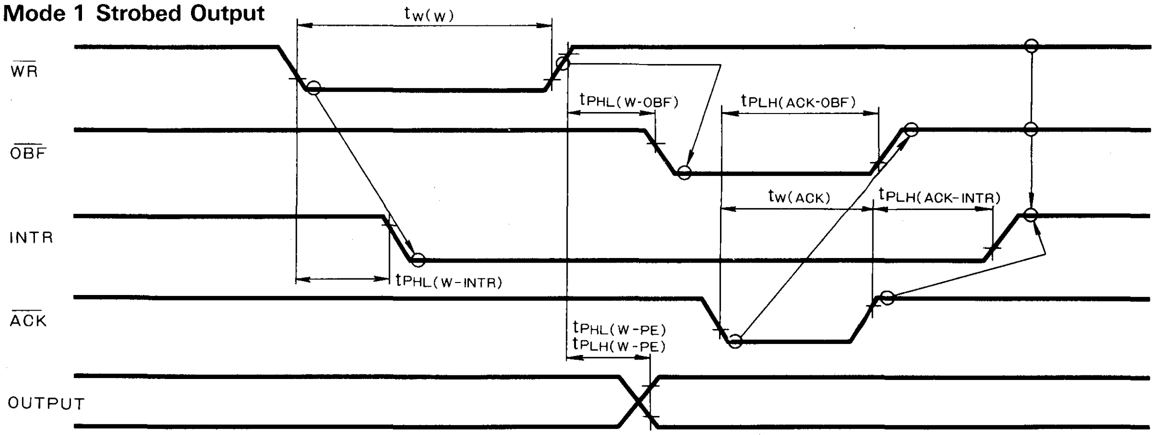
Mode 0 Basic Output



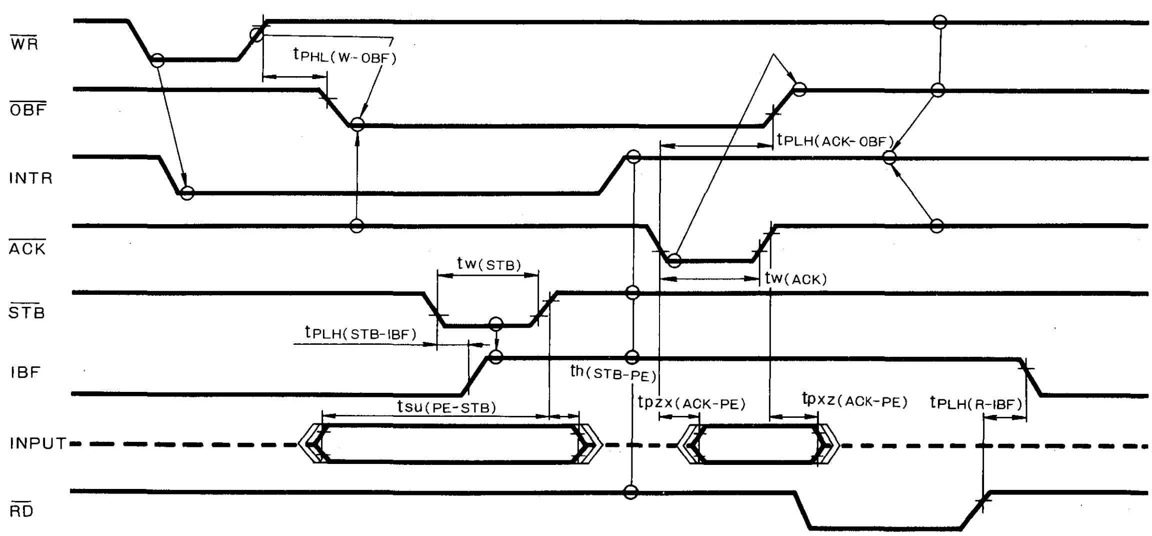
Mode 1 Strobed Input



Mode 1 Strobed Output



Mode 2 Bidirectional



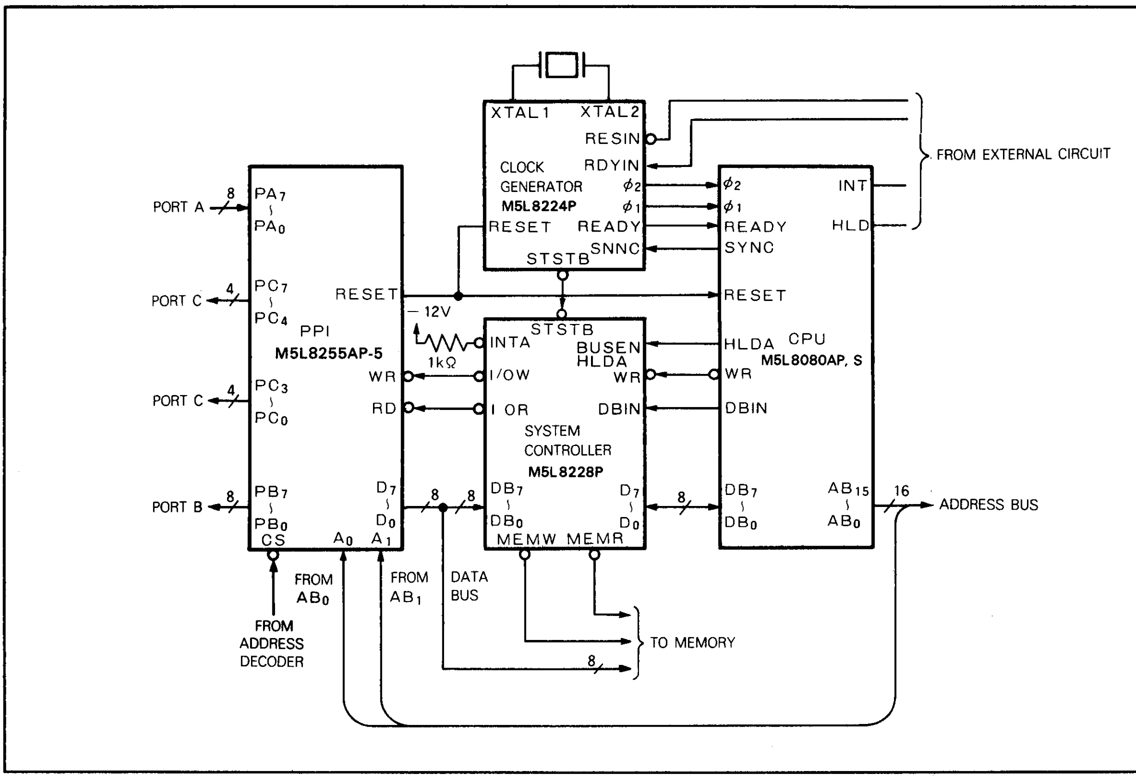
Note 13 : INTR=IBF·$\overline{\text{MASK}}$·$\overline{\text{STB}}$·$\overline{\text{RD}}$+$\overline{\text{OBF}}$·$\overline{\text{MASK}}$·$\overline{\text{ACK}}$·$\overline{\text{WR}}$

Circuit Examples for Applications

1. Mode 0

An example of a circuit for an application using mode 0 is shown in Fig. 11.

Fig. 11 Circuit example for an application using mode 0.



In this example, the PPI is in mode 0, and the control word should be 10010000 (90_{16}).

MVI	A, 90#
OUT	03#

The PPI will be initialized by executing the above two instructions.

Then, for example, to read data from port A and to output data to port B and C, the following three instructions can be used.

IN	00#	CPU A register ← Port A
OUT	01#	Port B ← A register
OUT	02#	Port C ← A register

After setting the mode each port operates as a normal port.

After setting the mode, as shown in Fig. 11, to read data from port A, to output to port B, and to set the first bit of port C "1", the following four instructions can be used.

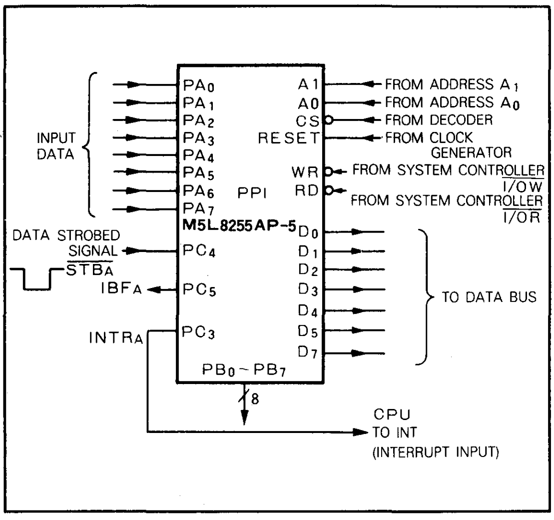
IN	00#	CPU A register ← Port A
OUT	01#	Port B ← A register
MVI	A, 01#	Bit-setting control word for PC_0
OUT	03#	Outputting to control address ($\overline{CS}$ = "0", A_1 = A_0 = "1")

The other bits of port C, in this case, are unknown.

2. Mode 1

An example of a circuit for an application using mode 1 is shown in Fig. 12.

Fig. 12 A circuit for an application using mode 1



Transferring data from a terminal unit to port A and sending a strobe signal to PC_4 will hold the data in the internal latch of the PPI, and PC_5 (IBF input buffer full flag) is set to "1". If a bit-set of PC_4 has been executed in advance, the CPU can be interrupted by the INTR signal of PC_3 when the input data is latched in the PPI. In this way, port A becomes an interrupting port; and at the same time, port B can select its mode independently.

The actual program for the circuit of Fig. 12 is as follows:

MVI	A, B0#	Control word is 10110000, port A is the mode 1 input and the others are output.
OUT	03#	Outputting to the control address
MVI	A, 09#	PC_4 bit-set 00001001
OUT	03#	Outputting to the control address
EI		Interrupt enable
HLT		Halt

If the data has been set in a terminal unit, and the strobe signal has been input; then the data will be latched in port A and the CPU INT goes high-level. In the case of Fig. 11, this is followed by outputting instruction RST 7 from the system controller as an interrupt command. Then a jump to 0038_{16} is executed to continue the program as follows:

0038_{16}	DI		
	IN	00#	CPU register A ← Port A PC_3 interrupt signal becomes low-level
	RET		

8

3. Mode 2

An example of a circuit for an application using mode 2 is shown in Fig. 13.

In Fig. 13, the data bus of the slave system is connected with the corresponding PPI port A bit of the master station. The input port consists of a three-state buffer and gate B which allow the slave CPU to read flag outputs (IBF, OBF) of the PPI as data.

When the following instruction is executed in this example, the action is as described:

I N 0 1 # (reading in from 01_{16} input port)

The data which is made up of the least significant bit (D_0), the $\overline{OBF}$ (output buffer full flag output) and the next least significant bit (D_1) of the IBF (input buffer full flag output) will be read into the slave CPU.

When the following instruction is executed, the action is as described:

I N 0 0 # (reading in from 00_{16} input port)

$\overline{ACK}$ (PC_6) of the PPI becomes low-level by gate C, and the contents of the port A output latch will be read into the slave CPU.

When the following instruction is executed, the action is as described:

O U T 0 0 # (writing out to 00_{16} output port)

$\overline{STB}$ (PC_4) of the PPI becomes low-level by gate D, then the contents of the slave CPU register A will be written into the port A input latch of the PPI.

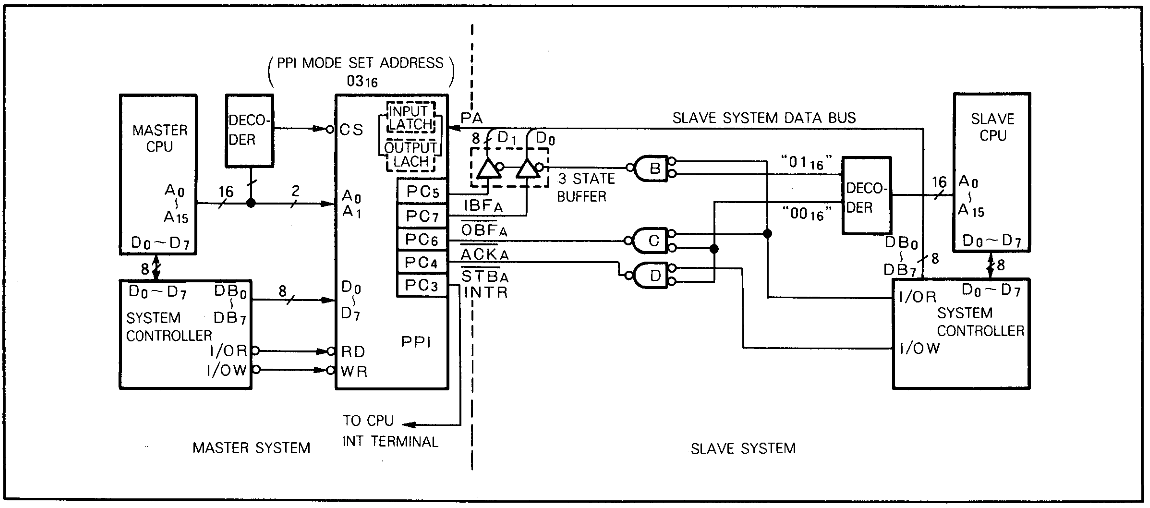
Actual operations are as follows:

1. PPI is set in mode 2 by the master CPU (03 address).
2. The master CPU writes the data, which is transferred to the slave CPU, into port A of the PPI (in turn, OBF becomes low-level).
3. The slave CPU continues to read the state of flags ($\overline{OBF}$ and $\overline{IBF}$) as data, while $\overline{OBF}$ is high-level (i.e. no data from the master CPU).
4. When the slave CPU senses that $\overline{OBF}$ has become low-level, the slave CPU starts to read the data from 00_{16} (which is the input address for the preceding data) which is in the output latch of port A (in turn, $\overline{OBF}$ returns to high-level).
5. During this period, the master CPU reads the status flags (reading in from 02 of port C) and checks the states of both the bit 7 ($\overline{OBF}$) and bit 5 (IBF). If $\overline{OBF}$ is low-level, it indicates that the slave CPU has not yet received the data; so the master does not write new data. If $\overline{OBF}$ is high-level, the master CPU writes the next data.
6. When data is to be transferred to the master CPU, the contents of the slave CPU A register will be transmitted to the port input latch of the PPI. The slave CPU transfers the data to address 00_{16} (in turn, the IBF becomes high-level).
7. The master CPU transfers data to port C and then checks the status flag. If the input latch contains data from the slave CPU, which is indicated by IBF having a high-level output, the data is read from port A (00_{16}) (in turn, the IBF returns to low-level).
8. The slave CPU reads the status flag from 02_{16} to determine if IBF has returned to low-level. If it has not, new data will not be written as long as IBF is high-level.
9. In this way, data can be exchanged. Since there are two sets of independent registers, input latch and output latch, used by port A of the PPI, it is not necessary to alternate input/output transfers.

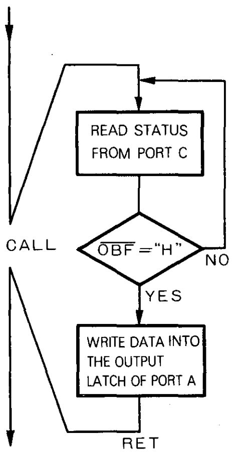
A program which has operating functions as described above, is explained as follows.

The operation, in mode 2, for group A of the PPI is considered here.

Fig. 13 A circuit for an application using mode 2



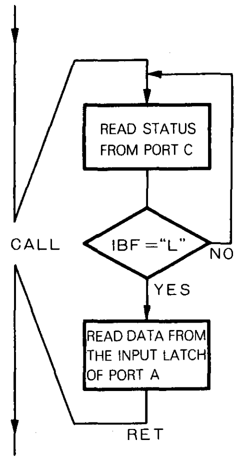
1. Master CPU subroutine for transmitting data to the slave CPU.



Program example

```
MOUT  PUSH  PSW
OBF   IN    02#
      ANI   80#
      JNZ   OBF
      POP   PSW
      OUT   00#
      RET
```

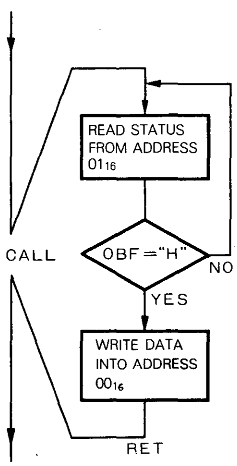
2. Subroutine for receiving data from the slave CPU.



Program example

```
MIN   IN    02#
      ANI   20#
      JZ    MIN
      IN    00#
      RET
```

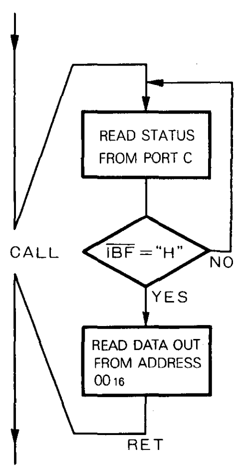
3. Slave CPU subroutine for transmitting data to the master CPU.



Program example

```
SOUT  PUSH  PSW
IBF   IN    02#
      ANI   02#
      JNZ   IBF
      POP   PSW
      OUT   00#
      RET
```

4. Subroutine for receiving data from the master CPU.



Program example

```
SIN   IN    02#
      ANI   01#
      JZ    SIN
      IN    00#
      RET
```

4. Address Decoding

Address decoding with multiple PPI units is shown in Figs. 14 and 15. These are functionally equal.

The same address data is output to both the upper and lower 8-bit address bus with the execution of IN and OUT instructions by the CPU.

Fig. 14 PPI address decoding (case 1)

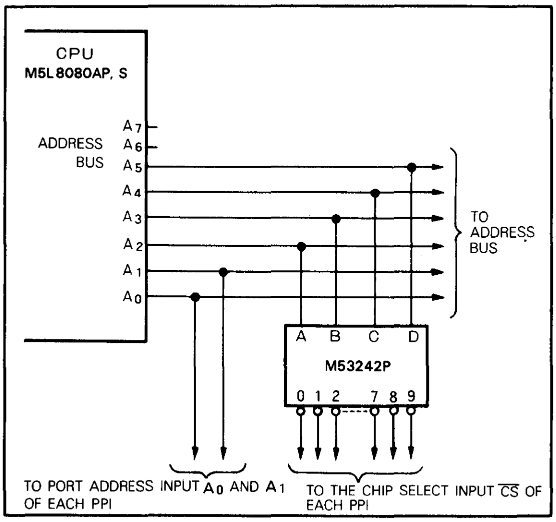
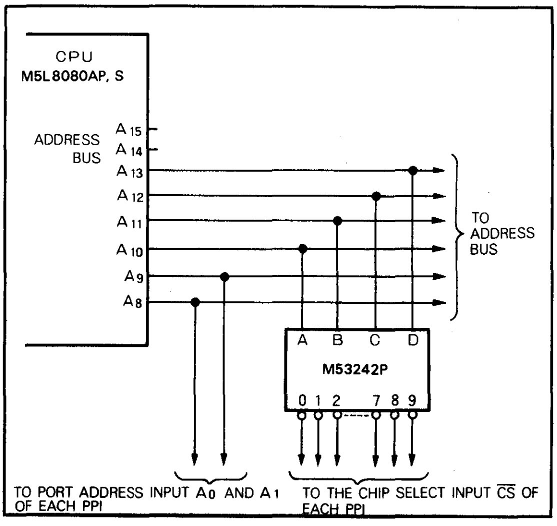


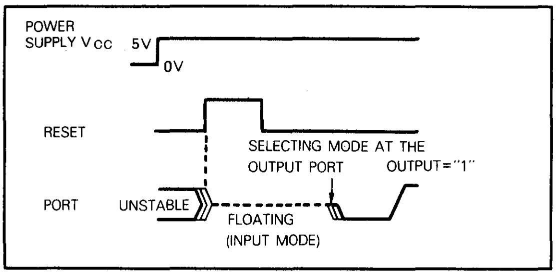
Fig. 15 PPI address decoding (case 2)



5. PPI Initialization

It is advisable to reset the PPI with a system initial reset and to select the mode at the beginning of a system program. The initial state of the PPI used as an output port is shown in Fig. 16.

Fig. 16 PPI initialization



Note 14 : Period of reset pulse must be at least 50μs during or after power on. Subsequent reset pulse can be 500ns minimum.

PROGRAMMABLE DMA CONTROLLER

DESCRIPTION

The M5L8257P-5 is a programmable, 4-channel direct memory access (DMA) controller. It is produced using the N-channel silicon-gate ED-MOS process and is specifically designed to simplify data transfer at high speeds for microcomputer systems. The LSI operates on a single 5V power supply.

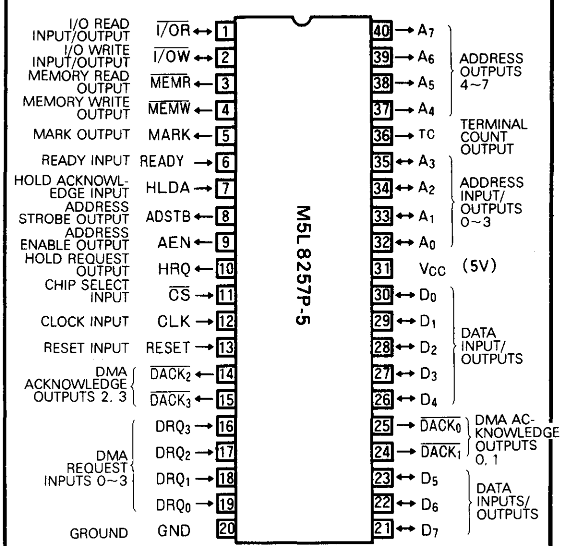
FEATURES

- 4-channel DMA controller
- Single 5V power supply
- Single TTL clock
- Priority DMA request logic
- Channel-masking function
- Terminal count and Modulo 128 outputs
- Compatible with the MELPS 8 microprocessor series
- Pin connection and electrical characteristics compatible with Intel's type 8257-5 programmable DMA controller

APPLICATIONS

- DMA control of peripheral equipment such as floppy disks and CRT terminals that require high-speed data transfer.

PIN CONFIGURATION (TOP VIEW)



Outline 40P1

FUNCTION

The M5L8257P-5 controller is used in combination with the M5L8212P 8-bit input/output port in 8-bit microcomputer systems.

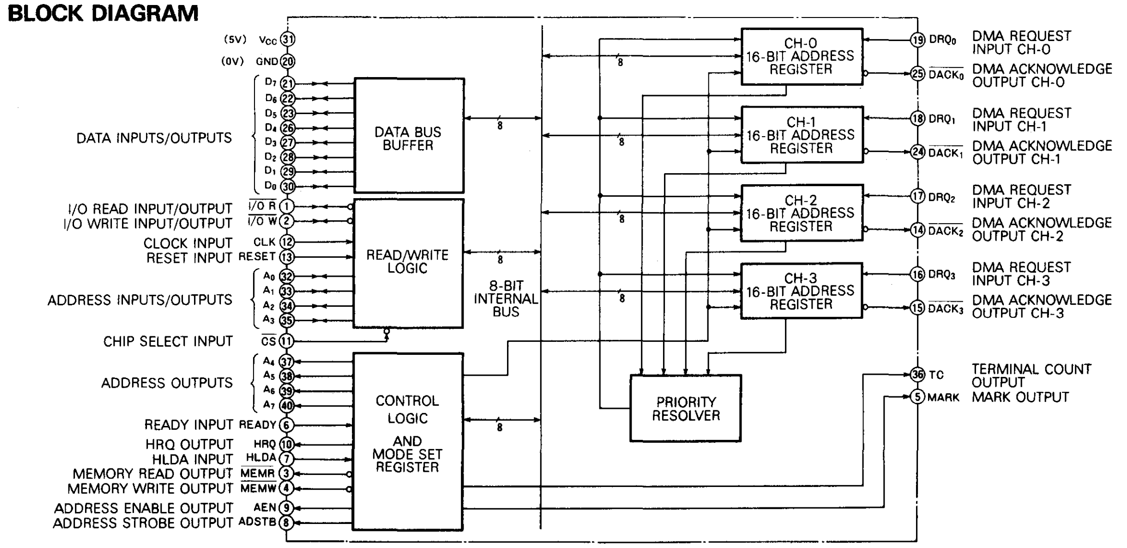
It consists of a channel section to acknowledge DMA requests, control logic to exchange commands and data with the CPU, read/write logic, and registers to hold transfer addresses and count the number of bytes to be transferred.

When a DMA request is made to an unmasked channel from the peripherals after setting of the transfer mode, transfer-start address and the number of transferred bytes for the registers, the M5L8257P-5 issues a priority request for the use of the bus to the CPU. On receiving an HLDA signal from the CPU, it sends a DMA acknowledge signal to the channel with the highest priority, starting DMA operation.

During DMA operation, the contents of the high-order 8 bits of the transfer memory address are transmitted to the M5L 8212P address-latch device through pins $D_0 \sim D_7$. The contents of the low-order 8 bits are transmitted through pins $A_0 \sim A_7$. After address transmission, DMA transfer can be started by dispatching read and write signals to the memories and peripherals.

8

BLOCK DIAGRAM



OPERATION

Data-Bus Buffer

This three-state, bidirectional, 8-bit buffer interfaces the M5L 8257P-5 to the CPU for data transfer. During a DMA cycle the upper 8 bits of the DMA address are output to the M5L 8212P latch device through this buffer.

I/O Read Input/Output ($\overline{I/OR}$)

When the M5L 8257P-5 is in slave-mode operation, this threestate, bidirectional pin serves for inputting and reads the upper/lower bytes of the 8-bit status register or 16-bit DMA address register and the high/low order bytes of the terminal counter.

In the master mode, the pin gives control output and is used to obtain data from a peripheral equipment during the DMA write cycle.

I/O Write Input/Output ($\overline{I/OW}$)

This pin is also of the three-state bidirectional type. When the M5L 8257P-5 is in slave-mode operation, it serves for inputting and loads the contents of the data bus on the upper/lower bytes of the 8-bit status register or 16-bit DMA address register and the upper/lower bytes of the terminal counter.

Clock Input (CLK)

This pin generates internal timing for the M5L 8257P-5 and is connected to the $\phi_{2(TTL)}$ output of the M5L 8224P -5 clock generator.

Reset Input (RESET)

This asynchronous input clears all registers and control lines inside the M5L 8257P-5.

Address Inputs/Outputs (A_0~A_3)

The four bits of these input/output pins are bidirectional. When the M5L 8257P-5 is in slave-mode operation, serve to input and address the internal registers. In the case of master operation, they output the low-order 4 bits of the 16-bit memory address.

Chip-Select Input (CS)

This pin is active on a low-level. It enables the IORD and IOWR. signals output from the CPU, when the M5L8257P -5 is in slave-mode operation.

In the master mode, it is disabled to prevent the chip from selecting itself while performing the DMA function.

Address Inputs/Outputs (A_4~A_7)

These four address lines are three-state outputs which constitute bits 4 through 7 of the memory address generated by the M5L 8257P-5 during all DMA cycles.

Ready Input (READY)

This asynchronous input is used to extend the memory read and write cycles in the M5L8257P-5 with wait states if the selected memory requires longer cycles.

Hold Request Output (HRQ)

This output requests control of the system bus. HRQ will normally be applied to the HOLD input on the CPU.

Hold Acknowledge Input (HLDA)

This input from the CPU indicates that the system bus is controlled by the M5L 8257P-5.

Memory Read Output ($\overline{MEMR}$)

This active-low three-state output is used to read data from the addressed memory location during DMA read cycles.

Memory Write Output ($\overline{MEMW}$)

This active-low three-state output is used to write data into the addressed memory location during DMA write cycles.

Address Strobe Output (ADSTB)

This output strobes the most significant byte of the memory address into the M5L8212P 8-bit input/output port through the data bus.

Address Enable Output (AEN)

This signal is used to disable the system data bus and system control bus by means of the bus enable pin on the M5L8228P system controller. It may also be used to inhibit non-DMA devices from responding during DMA cycles.

Terminal Count Output (TC)

This output signal notifies that the present DMA cycle is the last cycle for this data block.

Mark Output (MARK)

This signal notifies that the DMA transfer cycle for each channel is the 128th cycle since the previous MARK output.

DMA Request Inputs (DRQ0~DRQ3)

These independent, asynchronous channel-request inputs are used to secure use of the DMA cycle for the peripherals.

DMA Acknowledge Outputs ($\overline{DACK0}$~$\overline{DACK3}$)

These active-low outputs indicate that the peripheral equipment connected to the channel in question can execute the DMA cycle.

Register Initialization

Two 16-bit registers are provided for each of the 4 channels.

- **DMA Address register**

15															0
A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0

DMA TRANSFER STARTING ADDRESS

- **Terminal count register**

15	14	13													0
Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0

DMA MODE — NUMBER OF TRANSFERRED BYTES−1

The DMA transfer starting address, number of transferred bytes, and DMA mode are written for each channel in 2 steps using the 8-bit data bus. The lower-order and upper-order bytes are automatically indicated by the first-last flipflop for the writing and reading in 2 continuous steps.

The DMA mode (read, write, or verify) is indicated by the upper 2 bits of the terminal count register. The read mode refers to the operation of peripheral devices reading data out of memory. The write mode refers to data from peripheral devices being written into memory. The verify mode sends neither the read nor the write signals and performs a data check at the peripheral device.

In addition to the above-mentioned registers, there is a mode set register and a status register.

- **Mode set register (write only)**

7							0
AL	TCS	EW	RP	EN 3	EN 2	EN 1	EN 0

ADDED FUNCTION SETTING BITS — CHANNEL ENABLE BITS

- **Status Register (read only)**

7							0
0	0	0	UP	TC 3	TC 2	TC 1	TC 0

The upper-order 4-bits of the mode set register are used to select the added function, as described in Table 1. The lower-order 4-bits are mask kits for each channel. When set to 1, DMA requests are allowed. When the reset signal is input, all bits of the mode set and status registers are reset and DMA is inhibited for all channels. Therefore, to execute DMA operations, registers must first be initialized. An example of such an initialization is shown below.

MODESET:

MVI A, ADDL
OUT 00#: Channel 0 lower-order address
MVI A, ADDH
OUT 00#: Channel 0 upper-order address
MVI A, TCL
OUT 01#: Channel 0 terminal count lower-order
OUT 01#: Channel 0 terminal count upper-order
MVI A, XX
OUT 08#: Mode set register

As can be seen from the above example, until the contents of the address register and terminal count register become valid, the enable bit of the mode set register must not be set. This prevents memory contents from being destroyed by improper DRQ signals from peripheral devices.

DMA Operation Description

When a DMA request signal is received at the DRQ pin from a peripheral device after register initialization for a channel that is not masked, the M5L8257P-5 outputs a hold request signal to the CPU to begin DMA operation (S_1).

The CPU, upon receipt of the HRQ signal, outputs the HLDA signal which reserves capture of the bus after it has executed the present instruction to place this system in the hold state.

When the M5L8257P receives the HLDA signal, an internal priority determining circuit selects the channel with the highest priority for the beginning of data transfer (S_0).

Upon the next S_1 state, the address signal is sent. The lower-order 8-bits and the upper-order 8-bits are sent by means of the A_0~A_7 and D_0~D_7 pins respectively, latched into the M5L8212P and output at pins A_8~A_{15}. Simultaneous with this, the AEN signal is output to prohibit the selection of a device not capable of DMA.

In the S_2 state, the read, extended write, and DACK signals are output and data transferred from memory or a peripheral device appears on the data bus.

In the S_3 state, the write signal required to write data from the bus is output. At this time if the remaining number of bytes to be transferred from the presently selected channel has reached 0, the terminal count (TC) signal is output. Simultaneously with this, after each 128-byte data transfer a mark signal is output as required. In addition, in this state the READY pin is sampled and, if low, the wait state (S_W) is entered. This is used to perform DMA with slow access memory devices. In the verify mode, READY input is ignored.

In the S_4 state, the DRQ and HLDA pins are sampled at the end of a transferred byte as the address signal, control signals, and $\overline{DACK}$ signal are held to determine if transfer will continue.

As described above, transfer of 1 byte requires a minimum of 4 states for execution. For example, if a 2MHz clock input is used, the maximum transfer rate is 500k byte/s.

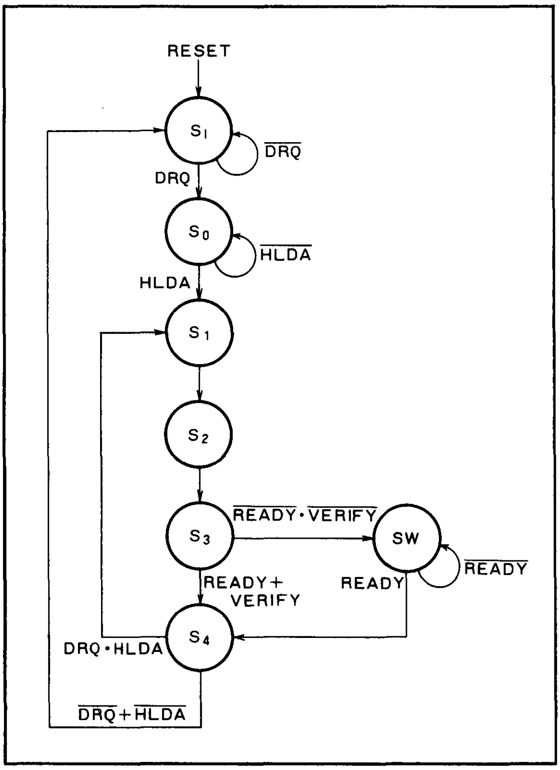


Fig. 1 DMA Operation state transistion diagram

Memory Mapped I/O

When using memory mapped I/O, it is necessary to change the connections for the control signals.

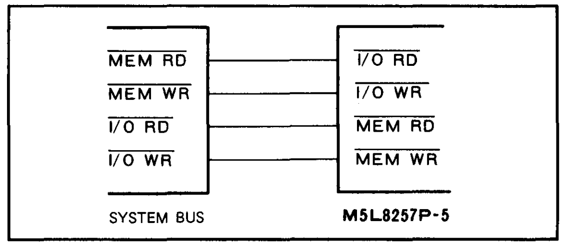


Fig. 2 Memory mapped I/O

Also, the read mode and write mode specifications for setting the mode of the terminal count are reversed.

Table 1 Internal Registers of the M5L8257P

Register	Byte	Address input				F/L	Bi-directional data bus							
		A_3	A_2	A_1	A_0		D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Channel 0 DMA address	Low-order	0	0	0	0	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
	High-order	0	0	0	0	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8
Channel 0 terminal count	Low-order	0	0	0	1	0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
	High-order	0	0	0	1	1	Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8
Channel 1 DMA address	Low-order	0	0	1	0	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
	High-order	0	0	1	0	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8
Channel 1 terminal count	Low-order	0	0	1	1	0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
	High-order	0	0	1	1	1	Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8
Channel 2 DMA address	Low-order	0	1	0	0	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
	High-order	0	1	0	0	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8
Channel 2 terminal count	Low-order	0	1	0	1	0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
	High-order	0	1	0	1	1	Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8
Channel 3 DMA address	Low-order	0	1	1	0	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
	High-order	0	1	1	0	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8
Channel 3 terminal count	Low-order	0	1	1	1	0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
	High-order	0	1	1	1	1	Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8
Mode setting (for write only)	—	1	0	0	0	0	AL	TCS	EW	RP	EN3	EN2	EN1	EN0
Status (for read only)	—	1	0	0	0	0	0	0	0	UP	TC3	TC2	TC1	TC0

$A_0 \sim A_{15}$: Addresses of the memories for which DMA will be carried out from now on. In initialization, DMA start addresses must be written.
$C_0 \sim C_{13}$: Terminal counts-in this IC (the number of remaining transfer bytes minus 1)
Rd, Wr : Used for DMA-mode setting by the following convention:

Rd	Wr	Mode to be set
0	0	DMA verify
0	1	DMA read
1	0	DMA write
1	1	Prohibition

AL : Automatic load mode. When this bit has been set, contents of the channel 3 register are written, as are, on the channel 2 register when channel 2 DMA transfer comes to an end. This mode allows quick, automatic chaining operations without intervention of the software.
EW : Extended write signal mode. When this bit has been set, write signals can be transmitted in advance to memories and peripheral equipment requiring long access time.
TCS : Terminal count stop. When a DMA transfer process is complete, with terminal-count output, the channel-enable mask of that channel is reset, prohibiting subsequent DMA cycles.
RP : Rotating priority mode. The setting of this mode allows the priority order to be rotated by each byte transfer.

Channel used for the present data transfer		CH-0	CH-1	CH-2	CH-3
Priority list for the next cycle	1	CH-1	CH-2	CH-3	CH-0
	2	CH-2	CH-3	CH-0	CH-1
	3	CH-3	CH-0	CH-1	CH-2
	4	CH-0	CH-1	CH-2	CH-3

EN0~EN3 : Channel-enable mask. This mask prohibits or allows the DMA request.
UP : Update flag. This is set when register contents are transferred in an automatic load mode from channel 3 to channel 2.
TC0~TC3 : Terminal-count status flags. At the time of terminal-count output, the flag corresponding to the channel is set.
F/L : First/last flip-flop. This is toggled when program and register-read operations for each channel are finished, and specifies whether the next program or read operation is to be for the upper bytes or the lower bytes. This means that write and read operations for each register must be carried out for a set of lower and higher bytes.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Power-supply voltage	With respect to GND	−0.5~7	V
V_I	Input voltage	With respect to GND	−0.5~7	V
V_O	Output voltage	With respect to GND	−0.5~7	V
P_d	Power dissipation (max.)	Ta=25℃	1000	mW
T_{opr}	Operating free-air temperature range		0~70	℃
T_{stg}	Storage temperature range		−65~150	℃

RECOMMENDED OPERATING CONDITIONS (Ta= 0 ~75℃, unless otherwise noted.)

Symbol	Parameter	Limits Min	Nom	Max	Unit
V_{CC}	Power-supply voltage	4.75	5	5.25	V
V_{SS}	Power-supply voltage (GND)		0		V
V_{IH}	High-level input voltage	2		V_{CC}+0.5	V
V_{IL}	Low-level input voltage	−0.5		0.8	V

ELECTRICAL CHARACTERISTICS (Ta= 0 ~70℃, V_{CC}= 5 V± 5 %, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits Min	Typ	Max	Unit
V_{OL}	Low-level output voltage	I_{OL}=1.6mA			0.45	V
V_{OH1}	High-level output voltage for AB, DB and AEN	I_{OH}=−150μA	2.4		V_{CC}	V
V_{OH2}	High-level output voltage for HRQ	I_{OH}=−80μA	3.3		V_{CC}	V
V_{OH3}	High-level output voltage for others	I_{OH}=−80μA	2.4		V_{CC}	V
I_{CC}	Power-supply current from V_{CC}				120	mA
I_I	Input current	V_I=V_{CC}~ 0 V	−10		10	μA
I_{OZ}	Off-state output current	V_I=V_{CC}~ 0 V	−10		10	μA
C_i	Input capacitance	Ta=25℃ V_{CC}=V_{SS}=0V Pins other than that under measurement are set to 0V. fc=1MHz			10	pF
$C_{i/o}$	Input/output terminal capacitance	Ta=25℃ V_{CC}=V_{SS}=0V Pins other than that under measurement are set to 0V. fc=1MHz			20	pF

TIMING REQUIREMENTS (Ta= 0 ~70℃, V_{CC}= 5 V± 5 %, V_{SS}= 0 V, V_{IH}=V_{OH}= 2 V, V_{IL}=V_{OL}=0.8V, unless otherwise noted.)

Symbol	Parameter	Alternative symbol	Test conditions	Limits Min	Typ	Max	Unit
$t_{w(R)}$	Read pulse width	T_{RR}	C_L=150pF	250			ns
$t_{su(A-R)}$ $t_{su(\overline{CS}-R)}$	Address or $\overline{CS}$ setup time before read	T_{AR}	C_L=150pF	0			ns
$t_{h(R-A)}$ $t_{h(R-\overline{CS})}$	Address or $\overline{CS}$ hold time after read	T_{RA}	C_L=150pF	0			ns
$t_{su(R-DQ)}$	Data setup time before read	T_{RD}	C_L=150pF	0		200	ns
$t_{h(R-DQ)}$	Data hold time after read	T_{DF}	C_L=150pF	20		100	ns
$t_{w(W)}$	Write pulse width	T_{WW}	C_L=150pF	200			ns
$t_{su(A-W)}$	Address setup time before write	T_{AW}	C_L=150pF	20			ns
$t_{h(W-A)}$	Address hold time after write	T_{WA}	C_L=150pF	0			ns
$t_{su(DQ-W)}$	Data setup time before write	T_{DW}	C_L=150pF	200			ns
$t_{h(W-DQ)}$	Data hold time after write	T_{WD}	C_L=150pF	0			ns
$t_{w(RST)}$	Reset pulse width	T_{RSTW}	C_L=150pF	300			ns
$t_{su(V_{CC}-RST)}$	Supply voltage setup time before reset	T_{RSTD}	C_L=150pF	500			μs
t_r	Input signal rise time	T_r	C_L=150pF			20	ns
t_f	Input signal fall time	T_f	C_L=150pF			20	ns
$t_{su(RST-W)}$	Reset setup time before write	T_{RSTS}	C_L=150pF	2			$t_{c(\phi)}$
$t_{c(\phi)}$	Clock cycle time	T_{CY}	C_L=150pF	0.32		4	μs
$t_{w(\phi)}$	Clock pulse width	T_θ	C_L=150pF	80		0.8$t_{c(\phi)}$	ns
$t_{su(DRQ-\phi)}$	DRQ setup time before clock	T_{QS}	C_L=150pF	70			ns
$t_{h(HLDA-DRQ)}$	DRQ hold time after HLDA	T_{QH}	C_L=150pF	0			ns
$t_{su(HLDA-\phi)}$	HLDA setup time before clock	T_{HS}	C_L=150pF	100			ns
$t_{su(RDY-\phi)}$	Ready setup time before clock	T_{RS}	C_L=150pF	30			ns
$t_{h(\phi-RDY)}$	Ready hold time after clock	T_{RH}	C_L=150pF	20			ns

Note 1 : Measurement conditions: M5L 8257P C_L=100pF, M5L 8257P-5 C_L=150pF

SWITCHING CHARACTERISTICS (Ta= 0 ~70℃, V_{CC}= 5 V± 5 %, V_{SS}= 0 V, V_{OH}=2 V, V_{OL}=0.8V, unless otherwise noted.)(Note 2)

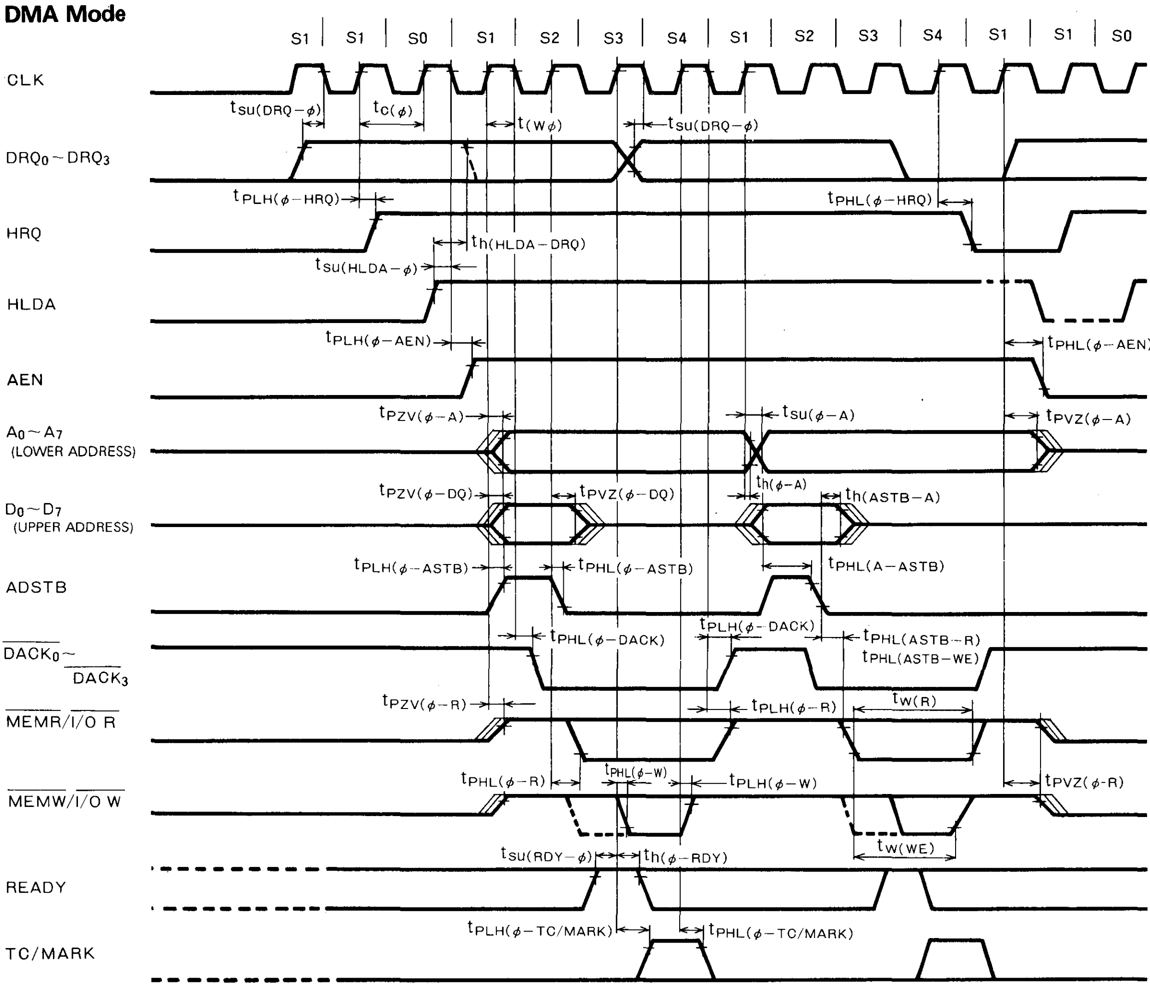
Symbol	Parameter	Alternative symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{PLH(\phi-HRQ)}$ $t_{PHL(\phi-HRQ)}$	Propagation time from clock to HRQ (Note 3)	T_{DQ}				160	ns
$t_{PLH(\phi-HRQ)}$ $t_{PHL(\phi-HRQ)}$	Propagation time from clock to HRQ (Note 5)	T_{DQI}				250	ns
$t_{PLH(\phi-AEN)}$	Propagation time from clock to AEN (Note 3)	T_{AEL}				300	ns
$t_{PHL(\phi-AEN)}$	Propagation time from clock to AEN (Note 3)	T_{AET}				200	ns
$t_{PZV(AEN-A)}$	Propagation time from AEN to address active (Note 6)	T_{AEA}		20			ns
$t_{PZV(\phi-A)}$	Propagation time from clock to address active (Note 4)	T_{FAAB}				250	ns
$t_{PVZ(\phi-A)}$	Propagation time from clock to address floating (Note 4)	T_{AFAB}				150	ns
$t_{su(\phi-A)}$	Address setup time after clock (Note 4)	T_{ASM}				250	ns
$t_{h(\phi-A)}$	Address hold time after clock (Note 4)	T_{AH}		$t_{su(\phi-A)}-50$			ns
$t_{h(R-A)}$	Address hold time after read (Note 6)	T_{AHR}		60			ns
$t_{h(W-A)}$	Address hold time after write (Note 6)	T_{AHW}		300			ns
$t_{PZV(\phi-DQ)}$	Propagation time from clock to data active	T_{FADB}				300	ns
$t_{PVZ(\phi-DQ)}$	Propagation time from clock to data floating (Note 4)	T_{AFDB}		$t_{PHL(\phi-ASTB)}+20$		170	ns
$t_{PHL(A-ASTB)}$	Propagation time from address to address strobe (Note 4)	T_{ASS}		100			ns
$t_{h(ASTB-A)}$	Propagation time from address strobe to address hold (Note 6)	T_{AHS}		50			ns
$t_{PLH(\phi-ASTB)}$	Propagation time from clock to address strobe (Note 3)	T_{STL}				200	ns
$t_{PHL(\phi-ASTB)}$	Propagation time from clock to address strobe (Note 3)	T_{STT}				140	ns
$t_{w(ASTB)}$	Address strobe pulse width (Note 6)	T_{SW}		$t_{c(\phi)}-100$			ns
$t_{PHL(AS-R)}$ $t_{PHL(AS-WE)}$	Propagation time from address strobe to read or extended write (Note 6)	T_{ASC}		70			ns
$t_{h(DQ-R)}$ $t_{h(DQ-WE)}$	Read or extended write hold time after data (Note 6)	T_{DBC}		20			ns
$t_{PLH(\phi-DACK)}$ $t_{PHL(\phi-TC/MARK)}$ $t_{PLH(\phi-TC/MARK)}$	Propagation time from clock to DACK or TC/MARK (Notes 3, 7)	T_{AK}				250	ns
$t_{PHL(\phi-R)}$ $t_{PHL(\phi-W)}$ $t_{PHL(\phi-WE)}$	Propagation time from clock to read, write or extended write (Notes 4, 8)	T_{DCL}				200	ns
$t_{PLH(\phi-R)}$ $t_{PLH(\phi-W)}$	Propagation time from clock to read or write (Notes 4, 9)	T_{DCT}				200	ns
$t_{PZV(\phi-R)}$ $t_{PZV(\phi-W)}$	Propagation time from clock to read active or write active (Note 4)	T_{FAC}				300	ns
$t_{PVZ(\phi-R)}$ $t_{PVZ(\phi-W)}$	Propagation time from clock to read floating or write floating (Note 4)	T_{AFC}				150	ns
$t_{w(R)}$	Read pulse width (Note 6)	T_{RAM}		$2t_{c(\phi)}+t_{w(\phi)}-50$			ns ns
$t_{w(W)}$	Write pulse width (Note 6)	T_{WWM}		$t_{c(\phi)}-50$			ns
$t_{w(WE)}$	Extended write pulse width	T_{WWME}		$2t_{c(\phi)}-50$			ns

Note 2 Reference level is V_{OH}=3.3V.
3 Load = 1 TTL
4 Load = 1 TTL + 50pF
5 Load = 1 TTL + (R_L=3.3kΩ), V_{OH}=3.3V.

Note 6 Tracking specification
7 $\triangle t_{PLH(\phi-DACK)} < 50ns$, $\triangle t_{PHL(\phi-TC/MARK)} < 50ns$, $\triangle t_{PLH(\phi-TC/MARK)} < 50ns$.
8 $\triangle t_{PHL(\phi-R)} < 50ns$, $\triangle t_{PHL(\phi-W)} < 50ns$, $\triangle t_{PHL(\phi-WE)} < 50ns$.
9 $\triangle t_{PLH(\phi-R)} < 50ns$, $\triangle t_{PLH(\phi-W)} < 50ns$.

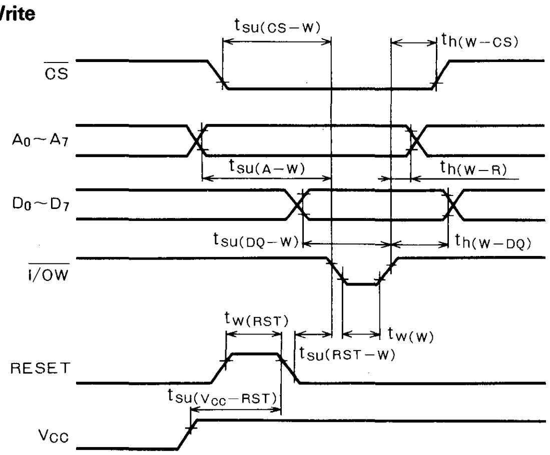
TIMING DIAGRAMS

DMA Mode

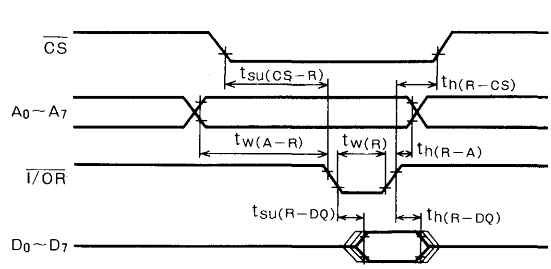


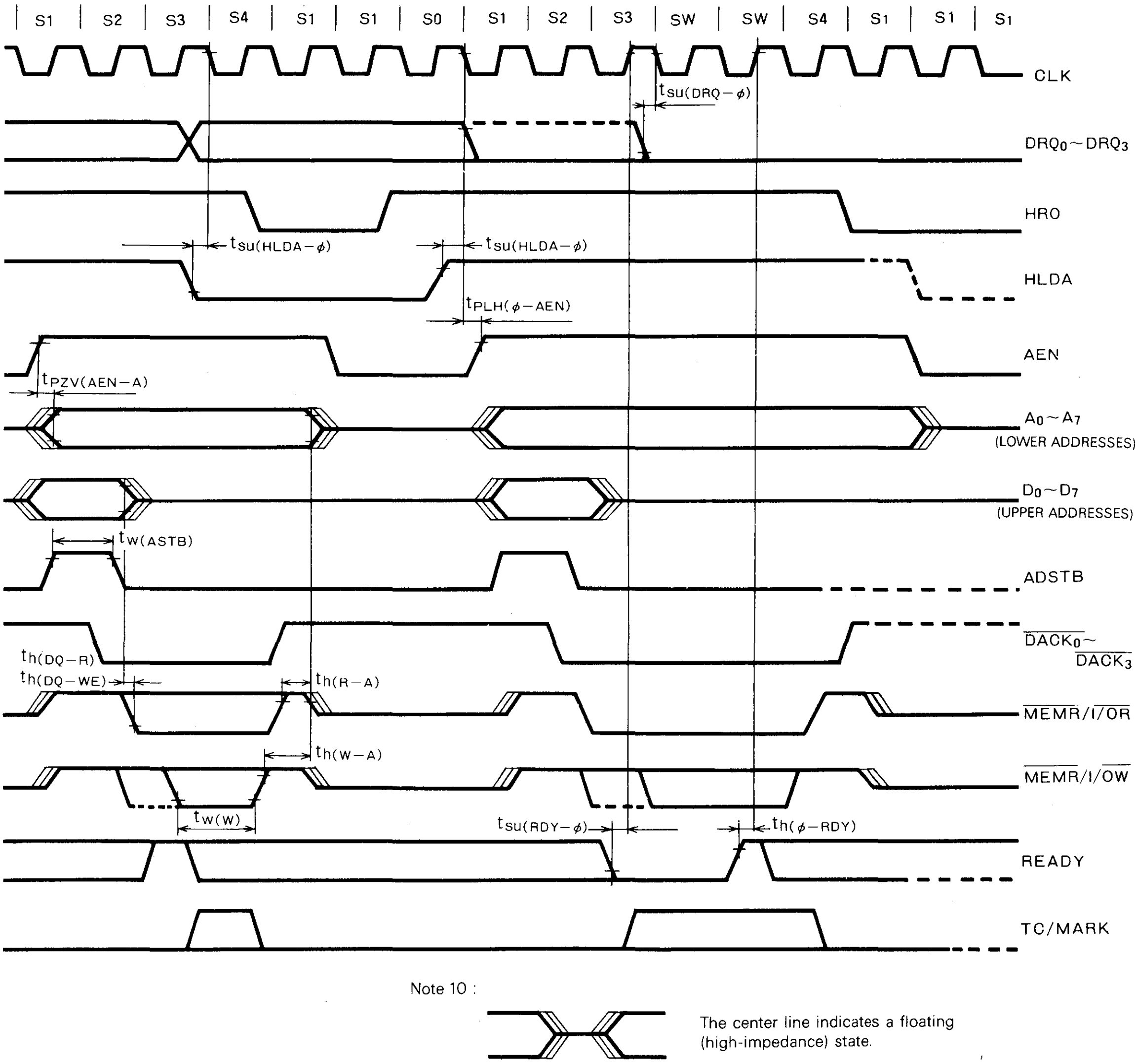
Slave Mode (Reference voltage: "H" = 2V "L" = 0.8V)

Write

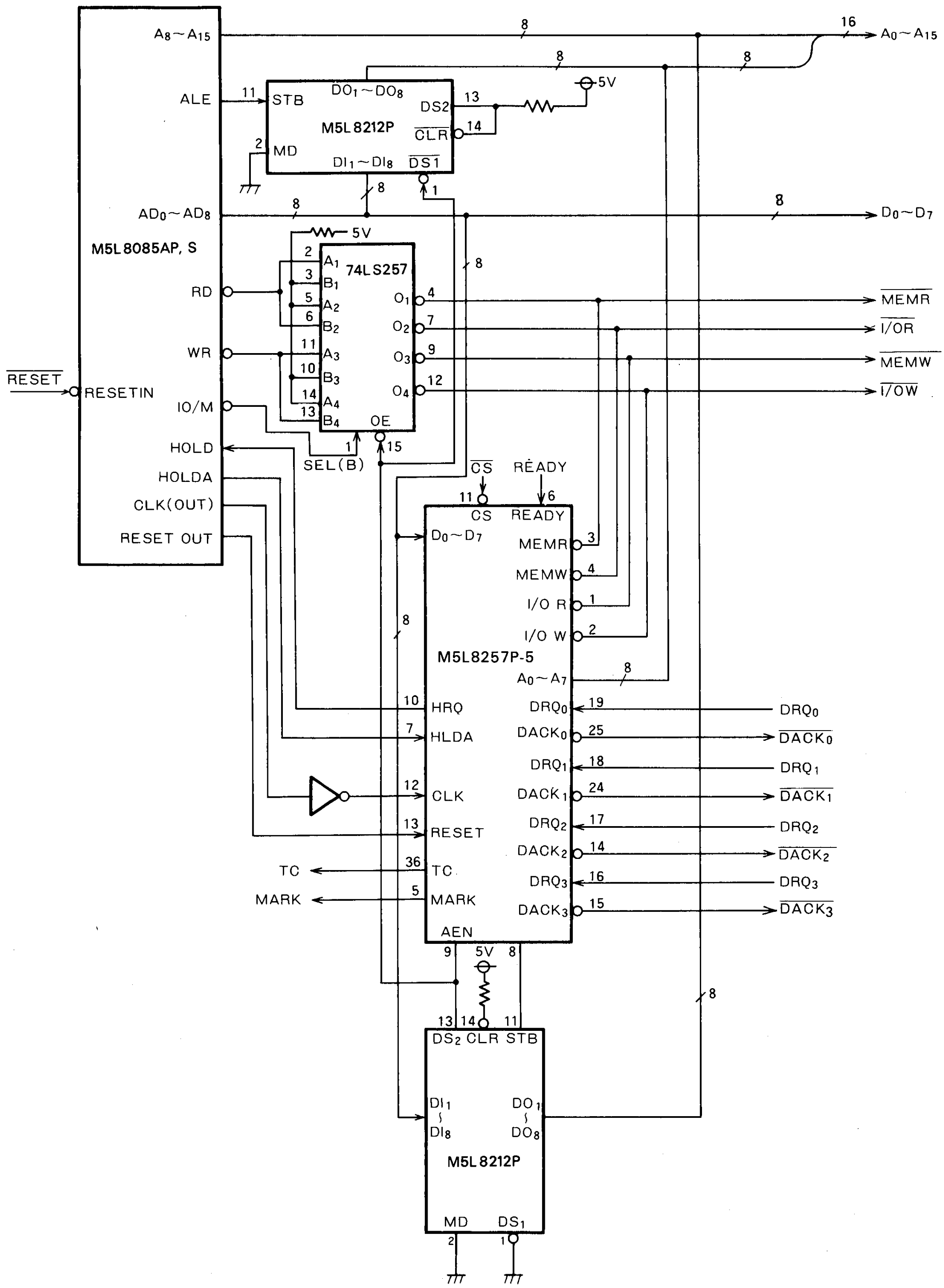


Read




S1
S2
S3
S4
S1
S1
S0
S1
S2
S3
SW
SW
S4
S1
S1
S1
CLK
tsu(DRQ−ϕ)
DRQ0~DRQ3
HRQ
tsu(HLDA−ϕ)
tsu(HLDA−ϕ)
HLDA
tPLH(ϕ−AEN)
AEN
tPZV(AEN−A)
A0~A7
(LOWER ADDRESSES)
D0~D7
(UPPER ADDRESSES)
tW(ASTB)
ADSTB
DACK0~DACK3
th(DQ−R)
th(DQ−WE)
th(R−A)
MEMR/I/OR
th(W−A)
MEMR/I/OW
tW(W)
tsu(RDY−ϕ)
th(ϕ−RDY)
READY
TC/MARK
Note 10 :
The center line indicates a floating (high-impedance) state.

APPLICATION EXAMPLE



MITSUBISHI ELECTRIC

PROGRAMMABLE INTERRUPT CONTROLLER

DESCRIPTION

The M5L8259AP is a programmable LSI for interrupt control. It is fabricated using N-channel silicon-gate ED-MOS technology and is designed to be used easily in connection with an M5L8080AP, M5L8085AP or M5L8086S.

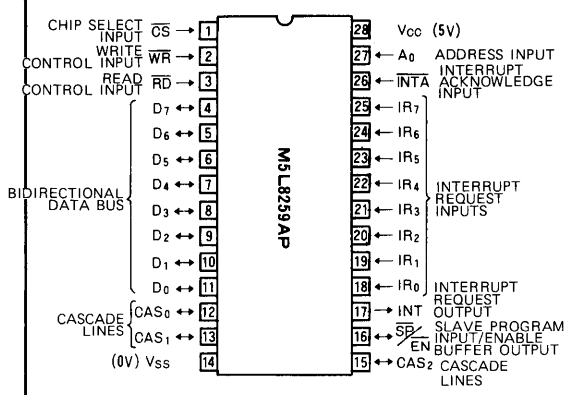
FEATURES

- Single 5V power supply
- CALL instruction to the CPU is generated automatically
- Priority, interrupt mask and vectored address for each interrupt request input are programmable
- Up to 64 levels of interrupt requests can be controlled by cascading with M5L8259AP
- Polling functions
- TTL compatible
- Interchangeable with Intels P8259A in pin configuration and electrical characteristics.

APPLICATIONS

- The M5L8259AP can be used as an interrupt controller for CPUs M5L8080AP, M5L8085AP and M5L8086S

PIN CONFIGURATION (TOP VIEW)



Outline 28P4

FUNCTIONS

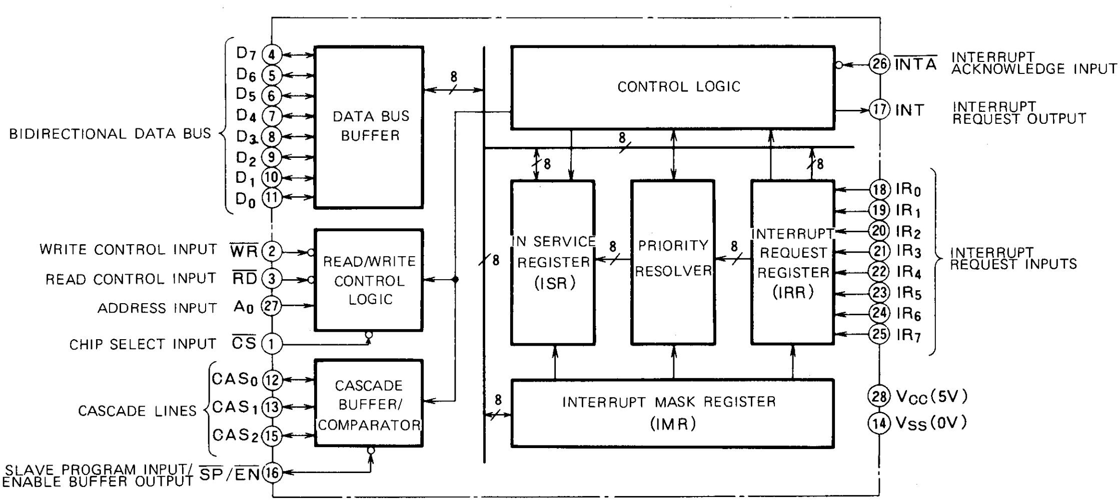
The M5L8259AP is a device specifically designed for use in real time, interrupt driven microcomputer systems. It manages eight levels or request and has built-in features for expandability to other M5L8259AP's.

The priority and interrupt mask can be changed or reconfigured at any time by the main program.

When an interrupt is generated because of an interrupt request at 1 of the pins, the M5L8259AP based on the mask and priority will output an INT to the CPU. After that, when an $\overline{INTA}$ signal is received from the CPU or the system controller, a CALL instruction and a programmed vector address is released onto the data bus.

8

BLOCK DIAGRAM



PROGRAMMABLE INTERRUPT CONTROLLER

PIN DESCRIPTION

Symbol	Pin name	Input or output	Functional significance
$\overline{CS}$	Chip select input	Input	This input is active at low-level, but may be at high-level during interrupt request input and interrupt processing.
$\overline{WR}$	Write control input	Input	Command write control input from the CPU
$\overline{RD}$	Read control input	Input	Data read control input for the CPU
$D_7 \sim D_0$	Bidirectional data bus	Input/ output	Data and commands are transmitted through this bidirectional data bus to and from the CPU.
$CAS_2 \sim CAS_0$	Cascade lines	Input/ output	These pins are outputs for a master and inputs for a slave. And these pins of the master will be able to address each individual slave. The master will enable the corresponding slave to release the device routine address during bytes 2 and 3 of $\overline{INTA}$.
$\overline{SP}/\overline{EN}$	Slave program input/ Enable buffer output	Input/ output	SP: In normal mode, a master is designated when $\overline{SP}/\overline{EN}$=1 and a slave is designated when $\overline{SP}/\overline{EN}$=0. EN: In the buffered mode, whenever the M5L8259AP's data bus output is enabled, its $\overline{SP}/\overline{EN}$ pin will go low.
INT	Interrupt request output	Output	This pin goes high whenever a valid interrupt is asserted.
$IR_7 \sim IR_0$	Interrupt request input	Input	The asynchronous interrupt inputs are active at high-level. The interrupt mask and priority of each interrupt input can be changed at any time. When using edge triggered mode, the rising edge (low to high) of the interrupt request and the high-level must be held until the first $\overline{INTA}$. For level triggered mode, the high-level must be held until the first $\overline{INTA}$.
$\overline{INTA}$	Interrupt acknowledge input	Input	When an interrupt acknowledge ($\overline{INTA}$) from the CPU is received, the M5L8259AP releases a CALL instruction or vectored address onto the data bus.
A_0	A_0 address input	Input	This pin is normally connected to one of the address lines and acts in conjunction with the $\overline{CS}$, $\overline{WR}$ and $\overline{RD}$ when writing commands or reading status registers.

OPERATION

The M5L8259AP is interfaced with a standard system bus as shown in Fig. 1 and operates as an interrupt controller.

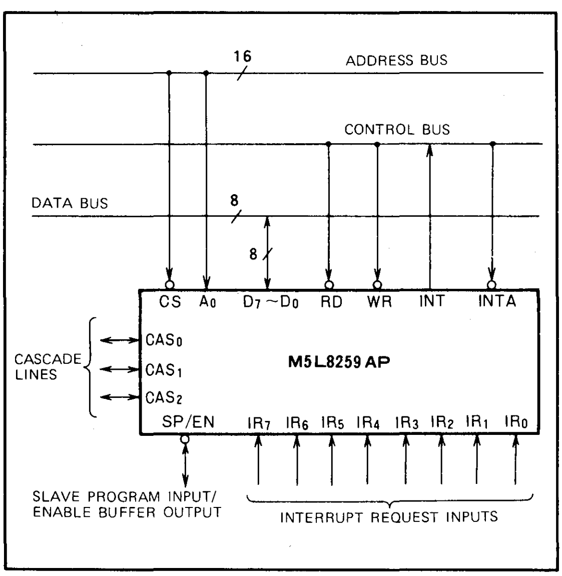


Fig. 1 The M5L8259AP interfaces to standard system bus.

Table 1 M5L8259AP basic operation

A_0	D_4	D_3	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	Input operation (read)
0			0	1	0	IRR, ISR or interrupting level→data bus
1			0	1	0	IMR→Data bus
						Output operation (write)
0	0	0	1	0	0	Data bus→OCW2
0	0	1	1	0	0	Data bus→OCW3
0	1	X	1	0	0	Data bus→ICW1
1	X	X	1	0	0	Data bus→OCW1, ICW2, ICW3, ICW4
						Disable function
X	X	X	1	1	0	Data bus → High-impedance
X	X	X	X	X	1	Data bus → High-impedance

Interrupt Sequence

1. When the CPU is an M5L8080AP or M5L8085AP:
   (1) When one or more of the interrupt request inputs are raised high, the corresponding IRR bit(s) for the high-level inputs will be set.
   (2) Mask state and priority levels are considered and, if appropriate, the M5L8259AP sends an INT signal to the CPU.
   (3) The acknowledgement of the CPU to the INT signal, the CPU issues an $\overline{\text{INTA}}$ pulse to the M5L-8259AP.
   (4) The ISR bit corresponding to the interrupt request input is set upon receiving an $\overline{\text{INTA}}$ from the CPU, and the corresponding IRR bit is reset. A CALL instruction is released onto the data bus.
   (5) A CALL is a 3-byte instruction, so additional $\overline{\text{INTA}}$ pulses are issued to the M5L8259AP from the CPU.
   (6) These two $\overline{\text{INTA}}$ pulses allow the M5L8259AP to release the program address onto the data bus. The low-order 8-bit vectored address is released at the second $\overline{\text{INTA}}$ pulse and the high-order 8-bit vectored address is released at the third $\overline{\text{INTA}}$ pulse.
   (7) This completes the 3-byte CALL instruction and the interrupt routine will be serviced. The ISR bit is reset at the end of the third $\overline{\text{INTA}}$ pulse in the AEOI mode. In the other modes the ISR bit is not reset until an EOI command is issued.
2. When the CPU is an M5L8086S:
   (1) When one or more of the interrupt request inputs are raised high, the corresponding IRR bit(s) for the high-level inputs will be set.
   (2) Mask state and priority levels are considered and if appropriated, the M5L8259AP sends an INT signal to the CPU.
   (3) As an acknowledgement to the INT signal, the CPU issues an $\overline{\text{INTA}}$ pulse to the M5L8259AP.
   (4) The ISR bit corresponding to the interrupt request input is set upon receiving the first $\overline{\text{INTA}}$ pulse from the CPU, and the corresponding IRR bit is reset. The M5L8259AP does not drive the data bus, and the data bus goes to high-impedance state.
   (5) When the second $\overline{\text{INTA}}$ pulse is issued from the CPU an 8-bit pointer is released onto the data bus.
   (6) This completes the interrupt cycle and the interrupt routine will be serviced. The ISR bit is reset at the end of the second $\overline{\text{INTA}}$ pulse in the AEOI mode. In the other modes the ISR bit is not reset until an EOI command is issued from the CPU.

The interrupt request input must be held at high-level until the first $\overline{\text{INTA}}$ pulse is issued. If it is allowed to return to low-level before the first $\overline{\text{INTA}}$ pulse is issued, an interrupt request in IR_7 is executed. However, in this case the ISR bit is not set.

Interrupt sequence outputs

1. When the CPU is a M5L8080AP or M5L8085AP:
   A CALL instruction is released onto the data bus when the first $\overline{\text{INTA}}$ pulse is issued. The low-order 8 bits of the vectored address are released when the second $\overline{\text{INTA}}$ pulse is issued, and the high-order 8 bits are released when the third $\overline{\text{INTA}}$ pulse is issued. The format of these three outputs is shown in Table 2.

Table 2 Formats of interrupt CALL instruction and vectored address

First $\overline{\text{INTA}}$ pulse (CALL instruction)

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	1	0	0	1	1	0	1

Second $\overline{\text{INTA}}$ pulse (low-order 8-bit of vectored address)

IR	Interval = 4							
	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
IR_7	A_7	A_6	A_5	1	1	1	0	0
IR_6	A_7	A_6	A_5	1	1	0	0	0
IR_5	A_7	A_6	A_5	1	0	1	0	0
IR_4	A_7	A_6	A_5	1	0	0	0	0
IR_3	A_7	A_6	A_5	0	1	1	0	0
IR_2	A_7	A_6	A_5	0	1	0	0	0
IR_1	A_7	A_6	A_5	0	0	1	0	0
IR_0	A_7	A_6	A_5	0	0	0	0	0

IR	Interval = 8							
	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
IR_7	A_7	A_6	1	1	1	0	0	0
IR_6	A_7	A_6	1	1	0	0	0	0
IR_5	A_7	A_6	1	0	1	0	0	0
IR_4	A_7	A_6	1	0	0	0	0	0
IR_3	A_7	A_6	0	1	1	0	0	0
IR_2	A_7	A_6	0	1	0	0	0	0
IR_1	A_7	A_6	0	0	1	0	0	0
IR_0	A_7	A_6	0	0	0	0	0	0

Third $\overline{\text{INTA}}$ pulse (high-order 8 bits of vectored address)

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8

2. When the CPU is a M5L8086S:
   The data bus goes to a high-impedance state when the first $\overline{INTA}$ pulse is issued. Then the pointer $T_7 \sim T_0$ is released when the next $\overline{INTA}$ pulse is issued. The content of the pointer $T_7 \sim T_0$ is shown in Table 3. The $T_2 \sim T_0$ are a binary code corresponding to the interrupt request level, $A_{10} \sim A_5$ are unused and ADI mode control is ignored.

Table 3 Contents of interrupt pointer

Second $\overline{INTA}$ pulse (8-bit pointer)

	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
IR_7	T_7	T_6	T_5	T_4	T_3	1	1	1
IR_6	T_7	T_6	T_5	T_4	T_3	1	1	0
IR_5	T_7	T_6	T_5	T_4	T_3	1	0	1
IR_4	T_7	T_6	T_5	T_4	T_3	1	0	0
IR_3	T_7	T_6	T_5	T_4	T_3	0	1	1
IR_2	T_7	T_6	T_5	T_4	T_3	0	1	0
IR_1	T_7	T_6	T_5	T_4	T_3	0	0	1
IR_0	T_7	T_6	T_5	T_4	T_3	0	0	0

Interrupt Request Register (IRR), In-service Register (ISR)

As interrupt requests are received at inputs $IR_7 \sim IR_0$, the corresponding bits of IRR are set and as an interrupt request is serviced the corresponding bit of ISR is set. The IRR is used to store all the interrupt levels which are requesting service, and the ISR is used to store all the interrupt levels which are being serviced. The status of these two registers can be read. These two registers are connected through the priority resolver.

An interrupt request received by IR_n is acknowledged on the leading edge when in the edge triggered mode or it is acknowledged on the level when in the level triggered mode. After that an INT signal is released and the interrupt request signal is latched in the corresponding IRR bit if the high-level is held until the first $\overline{INTA}$ pulse is issued. It is important to remember that the interrupt request signal must be held at high-level until the first $\overline{INTA}$ pulse is issued.

The interrupt request latching in the IRR causes a signal to be sent to the priority resolver unless it is masked out. When the priority resolver receives the signals it selects the highest priority interrupt request latched in IRR. The ISR is set when the first $\overline{INTA}$ pulse is issued while the corresponding bit of IRR is reset and the other bits of IRR are unaffected.

The bit of ISR that was set is not reset during the interrupt routine, but is reset at the end of the routine by the EOI command (end of interrupt) or by the last $\overline{INTA}$ pulse in AEOI mode.

Priority Resolver

The priority resolver examines all of the interrupt requests set in IRR to determine and selects the highest priority. The ISR bit corresponding to the selected (highest priority) request is set by the $\overline{INTA}$ pulse.

Interrupt Mask Register (IMR)

The contents of the interrupt mask register are used to mask out (disable) interrupt requests of selected interrupt request pins. Each terminal is independently masked so that masking a high priority interrupt does not influence the lower or higher priority interrupts. Therefore the contents of IMR selectively enable reading.

Interrupt Request Output (INT)

The interrupt request output connects directly to the interrupt input of the CPU. The output level is compatible with the input level required for the CPUs.

Interrupt Acknowledge Input ($\overline{INTA}$)

The CALL instruction and vectored address are released onto the data bus by the $\overline{INTA}$ pulse.

Data Bus Buffer

The data bus buffer is a 3-state bidirectional data bus buffer that is used to interface with the system bus. Write commands to the M5L8259AP, CALL instructions, vectored addresses, status information, etc. are transferred through the data bus buffer.

Read/Write Control Logic

The read/write control logic is used to control functions such as receiving commands from the CPU and supplying status information to the data bus.

Chip Select ($\overline{CS}$)

The M5L8259AP is selected (enabled) when $\overline{CS}$ is at low-level, but during interrupt request input or interrupt processing it may be high-level.

Write Control Input ($\overline{WR}$)

When $\overline{WR}$ goes to low-level the M5L8259AP can then write.

Read Control Input ($\overline{RD}$)

When $\overline{RD}$ goes low status information in the internal register of the M5L8259AP can be read through the data bus.

Address Input (A_0)

The address input is normally connected with one of the address lines and is used along with $\overline{WR}$ and $\overline{RD}$ to control write commands and reading status information.

Cascade Buffer/Comparator

The cascade buffer/comparator stores or compares identification codes. The three cascade lines are output when the M5L8259AP is a master or input when it is a slave. The identification code on the cascade lines select it as master or slave.

PROGRAMMING THE M5L8259AP

The M5L8259AP is programmed through the Initialization Command Word (ICW) and the operational command word (OCW). The following explains the functions of these two commands.

Initialization Command Words (ICWs)

The initialization command word is used for the initial setting of the M5L8259AP. There are 4 commands in this group and the following explains the details of these four commands.

ICW1

The meaning of the bits of ICW1 is explained in Fig. 3 along with the functions. ICW1 contains vectored address bits $A_7 \sim A_5$, a flag indicating whether interrupt input is edge triggered or level triggered, CALL address interval, whether a single M5L8259AP or the cascade mode is used, and whether ICW4 is required or not.

Whenever a command is issued with $A_0=0$ and $D_4=1$, this is interpreted as ICW1 and the following will automatically occur.

(a) The interrupt mask register (IMR) is cleared.
(b) The interrupt request input IR_7 is assigned the lowest priority.
(c) The identification code for slave mode is set to 7.
(d) The special mask mode is cleared and the status read is set to the interrupt request register (IRR).
(e) When IC4=0 all bits in ICW4 are set to zero.

ICW2

ICW2 contains vectored address bits $A_{15} \sim A_8$ or interrupt type $T_7 \sim T_3$, and the format is shown in Fig. 3.

ICW3

When SNGL=1 it indicates that only a single M5L8259AP is used in the system, in which case ICW3 is not valid. When SNGL=0, ICW3 is valid and indicates cascade connections with other M5L8259AP devices. In the master mode, a "1" is set for each slave.

When the CPU is an M5L8080AP or M5L8085AP the CALL instruction is released from the master at the first $\overline{INTA}$ pulse and the vectored address is released onto the data bus from the slave at the second and third $\overline{INTA}$ pulses.

When the CPU is a M5L8086S the master and slave are in high-impedance at the same time and the pointer is released onto the data bus from the slave at the next $\overline{INTA}$ pulse.

The master mode is specified when $\overline{SP}/\overline{EM}$ pin is high-level or BUF=1 and M/S=1 in ICW4, and slave mode is specified when $\overline{SP}/\overline{EM}$ pin is low-level or BUF=1 and M/S=0 in ICW4. In the slave mode, three bits $ID_2 \sim ID_0$ identify the slave. And then when the slave code released on the cascade lines from the master, matches the assigned ID code, the vectored address is released by it onto the data bus at the next $\overline{INTA}$ pulse.

ICW4

Only when IC4=1 in ICW1 is ICW4 valid. Otherwise all bits are set to zero. When ICW4 is valid it specifies special fully nested mode, buffer mode master/slave, automatic EOI and microprocessor mode. The format of ICW4 is shown in Fig. 3.

8

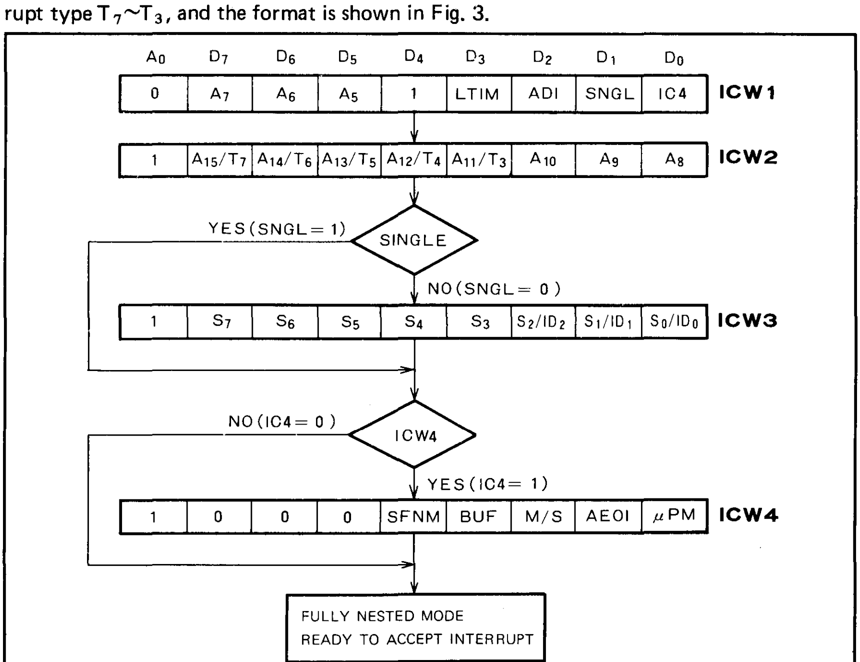


Fig. 2 Initialization sequence

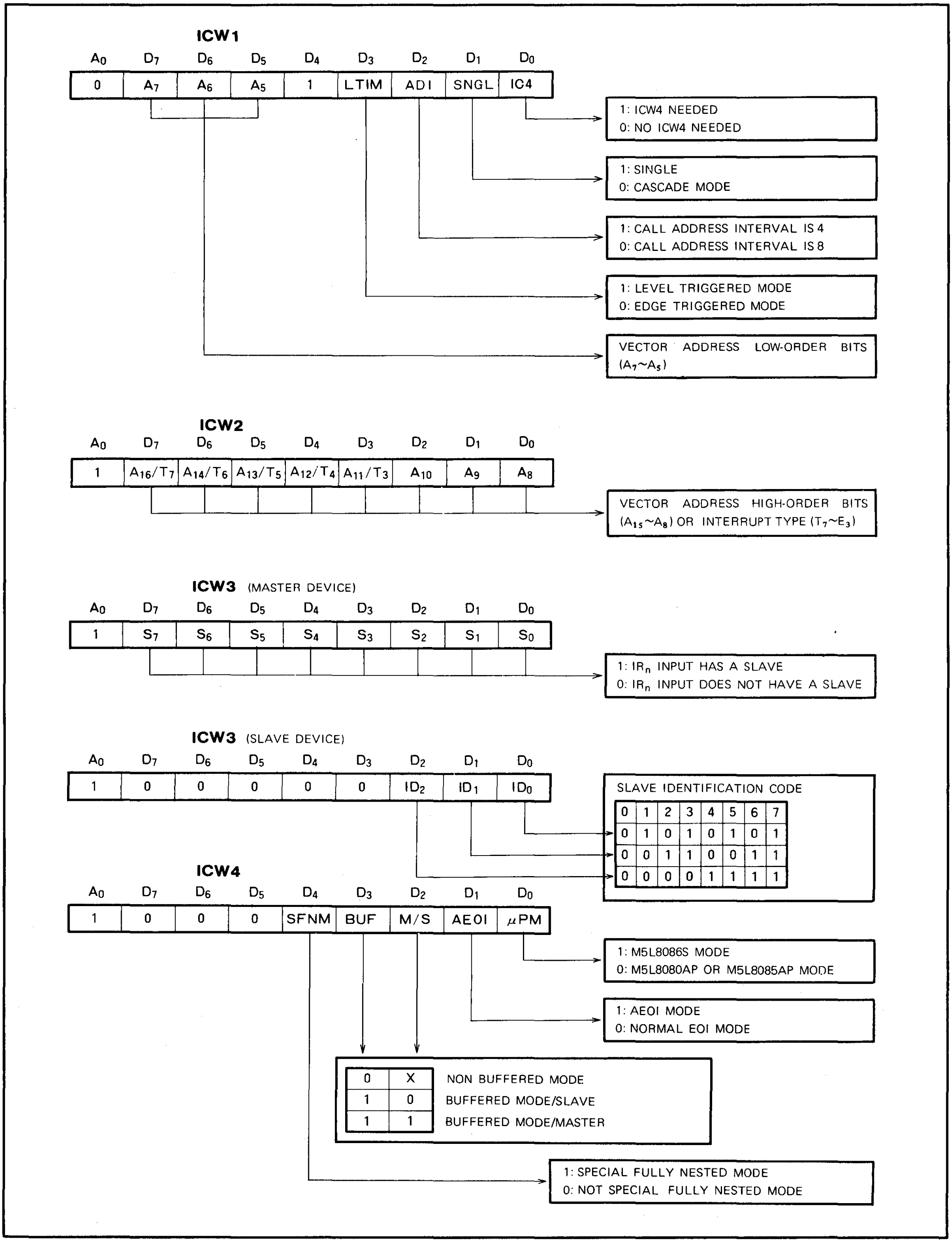


Fig. 3 Initialization command word format

Operation Command Words (OCWs)

The operation command words are used to change the contents of IMR, the priority of interrupt request inputs and the special mask. After the ICW are programmed into the M5L8259AP, the device is ready to accept interrupt requests. There are three types of OCWs; explanation of each follows, and the format of OCWs is shown in Fig. 4.

OCW1

The meaning of the bits of OCW1 are explained in Fig. 4 along with their functions. Each bit of IMR can be independently changed (set or reset) by OCW1.

OCW2

The OCW2 is used for issuing EOI commands to the M5L-8259AP and for changing the priority of the interrupt request inputs.

OCW3

The OCW3 is used for specifying special mask mode, poll mode and status register read.

OCW1

A_0	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	M_7	M_6	M_5	M_4	M_3	M_2	M_1	M_0

1: INTERRUPT MASK SET
0: INTERRUPT MASK RESET

OCW2

A_0	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
0	R	SL	EOI	0	0	L_2	L_1	L_0

ID LEVEL TO BE ACTED UPON

	0	1	2	3	4	5	6	7
L_0	0	1	0	1	0	1	0	1
L_1	0	0	1	1	0	0	1	1
L_2	0	0	0	0	1	1	1	1

R	SL	EOI		
0	0	1	NON-SPECIFIC EOI	EOI
0	1	1	SPECIFIC EOI (RESETS ISR BITS $L_2 \sim L_0$)	EOI
1	0	1	ROTATE ON NON-SPECIFIC EOI	AUTOMATIC ROTATION
1	0	0	SETS AUTOMATIC ROTATION FLIP-FLOP	AUTOMATIC ROTATION
0	0	0	RESET AUTOMATIC ROTATION FLIP-FLOP	AUTOMATIC ROTATION
1	1	1	ROTATE ON SPECIFIC EOI (RESETS ISR BIT $L_2 \sim L_0$)	SPECIFIC ROTATION
1	1	0	SETS PRIORITY COMMAND (SET LOWEST PRIORITY BIT $L_2 \sim L_0$)	SPECIFIC ROTATION
0	1	0	NO OPERATION	

OCW3

A_0	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
0	—	ESMM	SMM	0	1	P	RR	RIS

RR	RIS	
0	X	NO OPERATION
1	0	SETS STATUS READ REGISTER IN IRR
1	1	SETS STATUS READ REGISTER IN ISR

P — 1: POLL COMMAND; 0: NO POLL COMMAND

ESMM	SMM	
0	X	NO OPERATION
1	0	RESET SPECIAL MASK MODE
1	1	SETS SPECIAL MASK MODE

Fig. 4 Operation command word format

FUNCTION OF COMMAND

Interrupt masks

The mask register contains a mask for each individual interrupt request. These interrupt masks can be changed by programming using OCW1.

Special mask mode

When an interrupt request is acknowledged and the ISR bit corresponding to the interrupt request is not reset by EOI command (which means an interrupt service routine is executing) lower priority interrupt requests are ignored.

In special mask mode interrupt requests received at interrupt request inputs which are masked by OCW1 are disabled, but interrupts at all levels that are not masked are possible. This means that in the mask mode all level of interrupts are possible or individual inputs can be selectively programmed so all interrupts at the selected inputs are disabled. The masks are stored in IMR and special mask is set/reset by executing OCW3.

Buffered mode

The buffered mode will structure the M5L8259AP to send an enable signal on $\overline{SP/EN}$ to enable the data bus buffer, when the data bus requires the data bus buffer or when cascading mode is used. In this mode, when data bus output of the M5L8259AP is enabled, the $\overline{SP/EN}$ output becomes active. This allows the M5L8259AP to be programmed whether it is a master or a slave by software. The buffered mode is set/reset by executing ICW4.

Fully nested mode

The fully nested mode is the mode when no mode is specified and is the usual operational mode. In this mode, the priority of interrupt request terminals is fixed from the lowest IR_7 to the highest IR_0. When an interrupt request is acknowledged the CALL instruction and vectored address are released onto the data bus. At the same time the ISR bit corresponding to the accepted interrupt request is set. This ISR bit remains set until it is reset by the input of an EOI command or until the trailing edge of last $\overline{INTA}$ pulse in AEOI mode. While an interrupt service routine is being executed, interrupt requests of same or lower priority are disabled while the bit of ISR remains set. The priorities can be changed by OCW2.

Special fully nested mode

The special fully nested mode will be used when cascading is used and this mode will be programmed to the master by ICW4. The special fully nested mode is the same as the fully nested mode with the following two exceptions.

1. When an interrupt from a certain slave is being serviced, this slave is not locked out from the master priority logic. Higher priority interrupts within the slave will be recognized by the master and the master will initiate an interrupt request to the CPU. In general in the normal fully nested mode, a serviced slave is locked out from the master's priority, and so higher priority interrupts from the same slave are not serviced.
2. When an interrupt from a certain slave is being serviced the software must check ISR to determine if there are additional interrupts requests to be serviced. If the ISR bit is 0 the EOI command may be sent to the master too. But if it is not 0 the EOI command should not be sent to the master.

Poll mode

The poll mode is useful when the internal enable flip-flop of the microprocessor is reset, and interrupt input is disabled. Service to the device is achieved by a programmer initiative using a poll command. In the poll mode the M5L-8259AP at the next $\overline{RD}$ pulse puts 8 bits on the data bus which indicates whether there is an interrupt request and reads the priority level. The format of the information on the data bus is as shown below.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
I	—	—	—	—	W_2	W_1	W_0

I — 1: there is an interrupt request. 0: there is no interrupt request.

W_2–W_0 — Binary code of the highest priority level requesting services.

When I=0 (no interrupt request), $W_2 \sim W_0$ is 111. The poll is valid from $\overline{WR}$ to $\overline{RD}$ and interrupt is frozen. This mode can be used for processing common service routines for interrupts from more than one line and does not require any $\overline{INTA}$ sequence. Poll command is issued by setting P=1 in OCW3.

End of interrupt (EOI) and specific EOI (SEOI)

An EOI command is required by the M5L8259AP to reset the ISR bit. So an EOI command must be issued to the M5L8259AP before returning from an interrupt service routine.

When AEOI is selected in ICW4, the ISR bit can be reset at the trailing edge of the last $\overline{INTA}$ pulse. When AEOI is not selected the ISR bit is reset by the EOI command issued to the M5L8259AP before returning from an interrupt service routine. When programmed in the cascade mode the EOI command must be issued to the master once and to corresponding slave once.

There are two forms of EOI command, specific EOI and non-specific EOI. When the M5L8259AP is used in the fully nested mode, the ISR bit being serviced is reset by the EOI command. When the non-specific EOI is issued the M5L-8259AP will automatically reset the highest ISR bit of those that are set. Other ISR bits are reset by a specific EOI and the bit to be reset is specified in the EOI by the program. The SEOI is useful in modes other than free nested mode. When the M5L8259AP is in special mask mode ISR bits masked in IMR are not reset by EOI. EOI and SEOI are selected when OCW2 is executed.

Automatic EOI (AEOI)

In the AEOI mode the M5L8259AP executes non-specific EOI command automatically at the trailing edge of the last $\overline{INTA}$ pulse. The AEOI mode is not required within a single M5L8259AP, but it is useful when a nested multilevel interrupt structure is expected. When AEOI=1 in ICW4, the M5L8259AP is put in AEOI mode continuously until reprogrammed in ICW4.

Automatic rotation

The automatic rotation mode is used in applications where many interrupt requests of the same level are expected such as multichannel communication systems. In this mode when an interrupt request is serviced, that request is assigned the lowest priority so that if there are other interrupt requests they will have higher priorities. This means that the next request on the interrupt request being serviced must wait until the other interrupt requests are serviced (worst case is waiting for all 7 of the other controllers to be serviced). The priority and serving status are rotated as shown in Fig. 5.

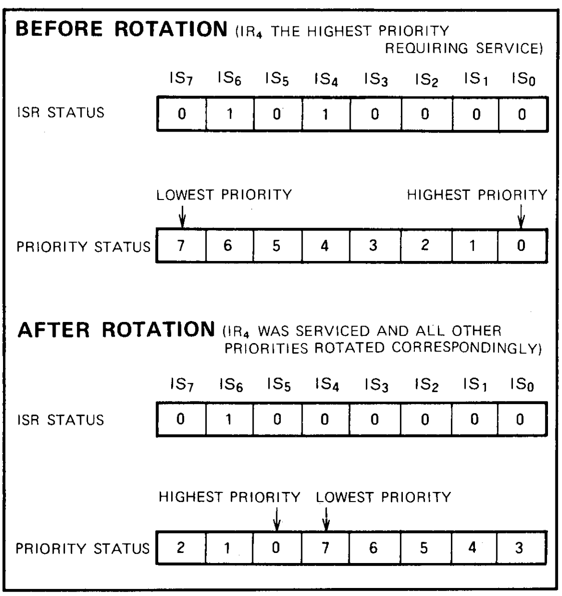


Fig. 5 An example of priority rotation

Automatic rotation mode is selected when R=1, EOI=1, SL=0 in OCW2. The internal priority status is changed by EOI or AEOI commands. The rotation priority A flip-flop is set by R=1, EOI=0 and SL=0 which is useful when the M5L8259AP is used in the AEOI mode.

Specific rotation

Specific rotation gives the user versatile capabilities in interrupt controlled operations. It serves in those applications in which a specific device's interrupt priority must be altered. As opposed to automatic rotation which automatically sets priorities, specific rotation is completely user controlled. That is, the user selects the interrupt level that is to receive lowest or highest priority. Priority changes can be executed during an EOI command.

Level triggered mode/Edge triggered mode

Selection of level or edge triggered mode of the M5L-8259AP is made by ICW1. When using edge triggered mode not only is a transition from low to high required, but the high-level must be held until the first $\overline{INTA}$. If the high-level is not held until the first $\overline{INTA}$, the interrupt request will be treated as if it were input on IR_7, except that the ISR bit is not set. When level triggered mode is used the functions are the same as edge triggered mode except that the transition from low to high is not required to trigger the interrupt request.

In the level triggered mode and using AEOI mode together, if the high-level is held too long the interrupt will occur immediately. To avoid this situation interrupts should be kept disabled until the end of the service routine or until the IR input returns low. In the edge triggered mode this type of mistake is not possible because the interrupt request is edge triggered.

Reading the M5L8259AP internal status

The contents of IRR and ISR can be read by the CPU with status read. When an OCW3 is issued to the M5L8259AP and an $\overline{RD}$ pulse issued the contents of IRR or ISR can be released onto the data bus. A special command is not required to read the contents of IMR. The contents of IMR can be released onto the data bus by issuing an $\overline{RD}$ pulse when A_0=1. There is no need to issue a read register command every time the IRR or ISR is to be read. Once a read register command is received by the M5L8259AP, it remains valid until it is changed. Remember that the programmer must issue a poll command every time to check whether there is an interrupt request and read the priority level. Polling overrides status read when P=1, RR=1 in OCW3.

Cascading

The M5L8259AP can be interconnected in a system of one master with up to eight slaves to handle up to 64 priority levels. A system of three units that can be used with the M5L8080AP or M5L8085AP is shown in Fig. 6.

The master can select a slave by outputting its identification code through the three cascade lines. The INT output of each slave is connected to the master interrupt request inputs. When an interrupt request of one of the slaves is to be serviced the master outputs the identification code of the slave through the cascade lines, so the slave will release the vectored address on the next $\overline{\text{INTA}}$ pulse.

The cascade lines of the master are nomally low, and will contain the slave identification code from the trailing edge of the first $\overline{\text{INTA}}$ pulse to the leading edge of the last $\overline{\text{INTA}}$ pulse. The master and slave can be programmed to work in different modes. ICWs, and EOI commands must be issued twice: once for the master and once for the corresponding slave. Each $\overline{\text{CS}}$ of the M5L8259AP requires an address decoder.

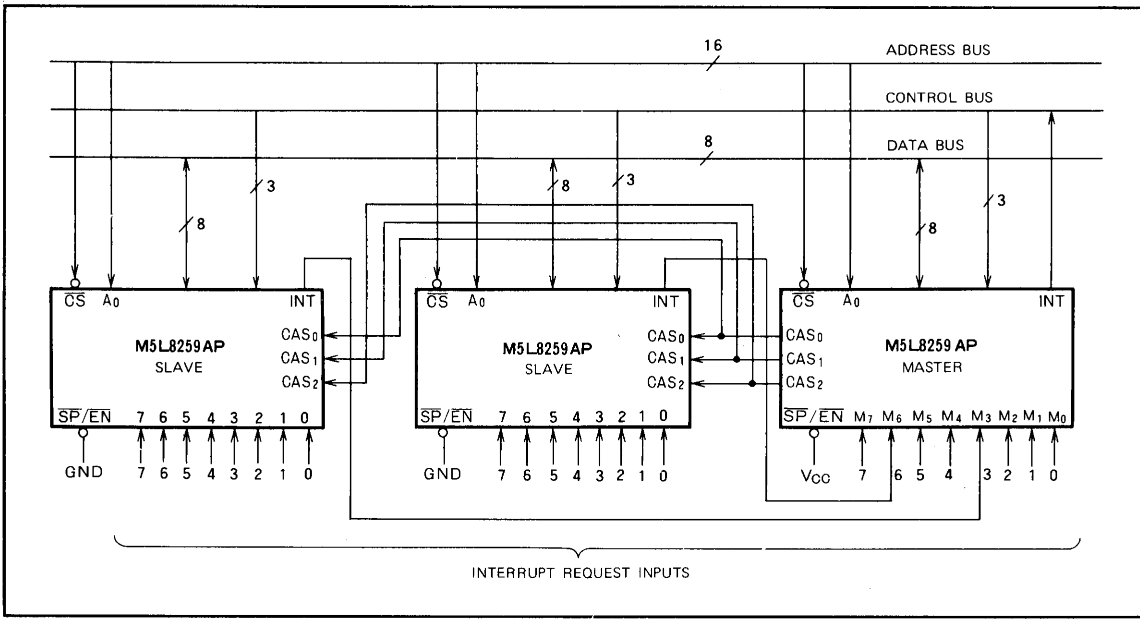


Fig. 6 Cascading the M5L8259AP

INSTRUCTION SET

Number \ Item	Mnemonic	Instruction code									Function			
		A_0	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	ICW4 required?	Interval	Single	Trigger
1	ICW1 A	0	A_7	A_6	A_5	1	0	1	1	0	N	4	Y	E
2	ICW1 B	0	A_7	A_6	A_5	1	1	1	1	0	N	4	Y	L
3	ICW1 C	0	A_7	A_6	A_5	1	0	1	0	0	N	4	N	E
4	ICW1 D	0	A_7	A_6	A_5	1	1	1	0	0	N	4	N	L
5	ICW1 E	0	A_7	A_6	0	1	0	0	1	0	N	8	Y	E
6	ICW1 F	0	A_7	A_6	0	1	1	0	1	0	N	8	Y	L
7	ICW1 G	0	A_7	A_6	0	1	0	0	0	0	N	8	N	E
8	ICW1 H	0	A_7	A_6	0	1	1	0	0	0	N	8	N	L
9	ICW1 I	0	A_7	A_6	A_5	1	0	1	1	1	Y	4	Y	E
10	ICW1 J	0	A_7	A_6	A_5	1	1	1	1	1	Y	4	Y	L
11	ICW1 K	0	A_7	A_6	A_5	1	0	1	0	1	Y	4	N	E
12	ICW1 L	0	A_7	A_6	A_5	1	1	1	0	1	Y	4	N	L
13	ICW1 M	0	A_7	A_6	0	1	0	0	1	1	Y	8	Y	E
14	ICW1 N	0	A_7	A_6	0	1	1	0	1	1	Y	8	Y	L
15	ICW1 O	0	A_7	A_6	0	1	0	0	0	1	Y	8	N	E
16	ICW1 P	0	A_7	A_6	0	1	1	0	0	1	Y	8	N	L
17	ICW2	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	8-bit vectored address			
18	ICW3 M	1	S_7	S_6	S_5	S_4	S_3	S_2	S_1	S_0	Slave connections (master mode)			
19	ICW3 S	1	0	0	0	0	0	ID_2	ID_1	ID_0	Slave identification code (slave mode)			
											SFNM	BUF	AEOI	8086
20	ICW4 A	1	0	0	0	0	0	0	0	0	N	N	N	N
21	ICW4 B	1	0	0	0	0	0	0	0	1	N	N	N	Y
22	ICW4 C	1	0	0	0	0	0	0	1	0	N	N	Y	N
23	ICW4 D	1	0	0	0	0	0	0	1	1	N	N	Y	Y
24	ICW4 E	1	0	0	0	0	0	1	0	0	N	N	N	N
25	ICW4 F	1	0	0	0	0	0	1	0	1	N	N	N	Y
26	ICW4 G	1	0	0	0	0	0	1	1	0	N	N	Y	N
27	ICW4 H	1	0	0	0	0	0	1	1	1	N	N	Y	Y
28	ICW4 I	1	0	0	0	0	1	0	0	0	N	Y S	N	N
29	ICW4 J	1	0	0	0	0	1	0	0	1	N	Y S	N	Y
30	ICW4 K	1	0	0	0	0	1	0	1	0	N	Y S	Y	N
31	ICW4 L	1	0	0	0	0	1	0	1	1	N	Y S	Y	Y
32	ICW4 M	1	0	0	0	0	1	1	0	0	N	Y M	N	N
33	ICW4 N	1	0	0	0	0	1	1	0	1	N	Y M	N	Y
34	ICW4 O	1	0	0	0	0	1	1	1	0	N	Y M	Y	N
35	ICW4 P	1	0	0	0	0	1	1	1	1	N	Y M	Y	Y
36	ICW4 NA	1	0	0	0	1	0	0	0	0	Y	N	N	N
37	ICW4 NB	1	0	0	0	1	0	0	0	1	Y	N	N	Y
38	ICW4 NC	1	0	0	0	1	0	0	1	0	Y	N	Y	N
39	ICW4 ND	1	0	0	0	1	0	0	1	1	Y	N	Y	Y
40	ICW4 NE	1	0	0	0	1	0	1	0	0	Y	N	N	N
41	ICW4 NF	1	0	0	0	1	0	1	0	1	Y	N	N	Y
42	ICW4 NG	1	0	0	0	1	0	1	1	0	Y	N	Y	N
43	ICW4 NH	1	0	0	0	1	0	1	1	1	Y	N	Y	Y
44	ICW4 NI	1	0	0	0	1	1	0	0	0	Y	Y S	N	N
45	ICW4 NJ	1	0	0	0	1	1	0	0	1	Y	Y S	N	Y
46	ICW4 NK	1	0	0	0	1	1	0	1	0	Y	Y S	Y	N
47	ICW4 NL	1	0	0	0	1	1	0	1	1	Y	Y S	Y	Y
48	ICW4 NM	1	0	0	0	1	1	1	0	0	Y	Y M	N	N
49	ICW4 NN	1	0	0	0	1	1	1	0	1	Y	Y M	N	Y
50	ICW4 NO	1	0	0	0	1	1	1	1	0	Y	Y M	Y	N
51	ICW4 NP	1	0	0	0	1	1	1	1	1	Y	Y M	Y	Y
52	OCW1	1	M_7	M_6	M_5	M_4	M_3	M_2	M_1	M_0	Interrupt mask			
53	OCW2 E	0	0	0	1	0	0	0	0	0	EOI			
54	OCW2 SE	0	0	1	1	0	0	L_2	L_1	L_0	SEOI			
55	OCW2 RE	0	1	0	1	0	0	0	0	0	Rotate on Non-Specific EOI command (Automatic rotation)			
56	OCW2 RSE	0	1	1	1	0	0	L_2	L_1	L_0	Rotate on Specific EOI command (Specific rotation)			
57	OCW2 R	0	1	0	0	0	0	0	0	0	Rotate in AEOI Mode (SET)			
58	OCW2 CR	0	0	0	0	0	0	0	0	0	Rotate in AEOI Mode (CLEAR)			
59	OCW2 RS	0	1	1	0	0	0	L_2	L_1	L_0	Set priority without EOI			
60	OCW3 P	0	0	0	0	0	1	1	0	0				
61	OCW3 RIS	0	0	0	0	0	1	0	1	1				
62	OCW3 RR	0	0	0	0	0	1	0	1	0				
63	OCW3 SM	0	0	1	1	0	1	0	0	0				
64	OCW3 RSM	0	0	1	0	0	1	0	0	0				

Note: Y: yes, N: no, E: edge, L: level, M: master, S: slave

8

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.5~7	V
V_I	Input voltage		−0.5−7	V
V_O	Output voltage		−0.5~7	V
Pd	Power dissipation		1000	mW
Topr	Operating free-air temperature range		0~70	°C
Tstg	Storage temperature range		−65~150	°C

RECOMMENDED OPERATING CONDITIONS (Ta=0~70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage	2		V_{CC}+0.5	V
V_{IL}	Low-level input voltage	−0.5		0.8	V

ELECTRICAL CHARACTERISTICS (Ta=0~70°C, V_{CC}, 5V±10%, V_{SS}=0V, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	I_{OH}=−400μA	2.4			V
$V_{OH(INT)}$	High-level output voltage, interrupt request output	I_{OH}=−100μA	3.5			V
V_{OL}	Low-level output voltage	I_{OL}=2.2mA			0.45	V
I_{CC}	Supply current from V_{CC}				85	mA
I_{IH}	High-level input current	V_I=V_{CC}	−10		10	μA
I_{IL}	Low-level input current	V_I=0V	−10		10	μA
I_{OZ}	Off-state output current	V_{SS}=0, V_I=0.45~5.5V	−10		10	μA
$I_{IH(IR)}$	High-level input current, interrupt request inputs	V_I=V_{CC}			10	μA
$I_{IL(IR)}$	Low-level input current, interrupt request inputs	V_I=0V	−300			μA
Ci	Output capacitance	V_{CC}=V_{SS}, f=1MHz, 25mVrms, Ta=25°C			10	pF
$Ci_{/O}$	Input/output capacitance	V_{CC}=V_{SS}, f=1MHz, 25mVrms, Ta=25°C			20	pF

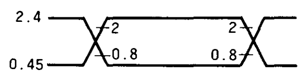
TIMING REQUIREMENTS (Ta=0~70°C, V_{CC}=5V±10%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Limits			Unit
			Min	Typ	Max	
$t_{w(W)}$	Write pulse width	t_{WLWH}	290			ns
$t_{su(A-W)}$	Address setup time before write	t_{AHWL}	0			ns
$t_{h(W-A)}$	Address hold time after write	t_{WHAX}	0			ns
$t_{su(DQ-W)}$	Data setup time before write	t_{DVWH}	240			ns
$t_{h(W-DQ)}$	Data hold time after write	t_{WHDX}	0			ns
$t_{w(R)}$	Read pulse width	t_{RLRH}	235			ns
$t_{su(A-R)}$	Address setup time before read	t_{AHRL}	0			ns
$t_{h(R-A)}$	Address hold time after read	t_{RHAX}	0			ns
$t_{w(IR)}$	Interrupt request input width, low-level time, edge triggered mode	t_{JLJH}	100			ns
$t_{su(CAS-INTA)}$	Cascade setup time after $\overline{INTA}$ (slave)	t_{CVIAL}	55			ns
$t_{rec(W)}$	Write recovery time	t_{WHRL}	190			ns
$t_{rec(R)}$	Read recovery time	t_{RHRL}	160			ns

SWITCHING CHARACTERISTICS ($T_a = 0 \sim 70°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted)

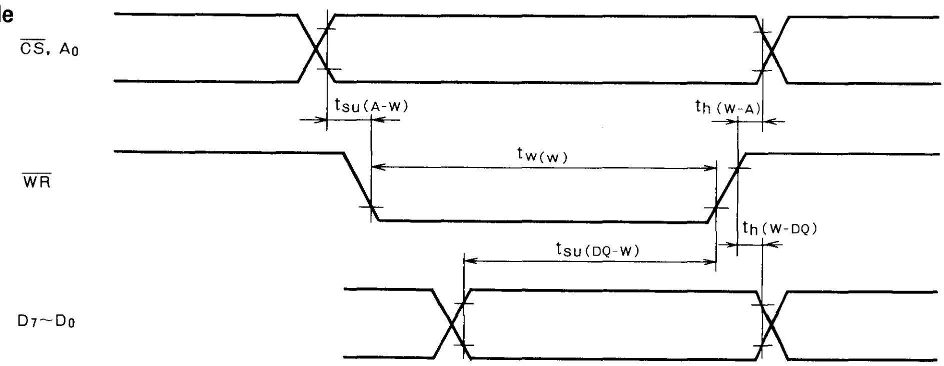
Symbol	Parameter	Alternative symbol	Limits			Unit
			Min	Typ	Max	
$t_{PZV(R-DQ)}$	Data output enable time after read	t_{RLDV}			200	ns
$t_{PVZ(R-DQ)}$	Data output disable time after read	t_{RHDZ}			100	ns
$t_{PZV(A-DQ)}$	Data output enable time after address	t_{AHDV}			200	ns
$t_{PHL(R-EN)}$	Propagation time from read to enable signal output	t_{RLEL}			125	ns
$t_{PLH(R-EN)}$	Propagation time from read to disable signal output	t_{RHEH}			150	ns
$t_{PLH(IR-INT)}$	Propagation time from interrupt request input to interrupt request output	t_{JHIH}			350	ns
$t_{PLV(INTA-CAS)}$	Propagation time from $\overline{INTA}$ to cascade output (master)	t_{IALCV}			565	ns
$t_{PZV(CAS-DQ)}$	Data output enable time after cascade output (slave)	t_{CVDV}			300	ns

Note 1: $\overline{INTA}$ signal is considered read signal
$\overline{CS}$ signal is considered address signal
Input pulse level 0.45~2.4V
Input pulse rise time 20ns
Input pulse fall time 20ns
Reference level input V_{IH}=2V, V_{IL}=0.8V
output V_{OH}=2V, V_{OL}=0.8V
Load capacitance C_L=100pF, where $\overline{SP}/\overline{EN}$ pin is 15pF

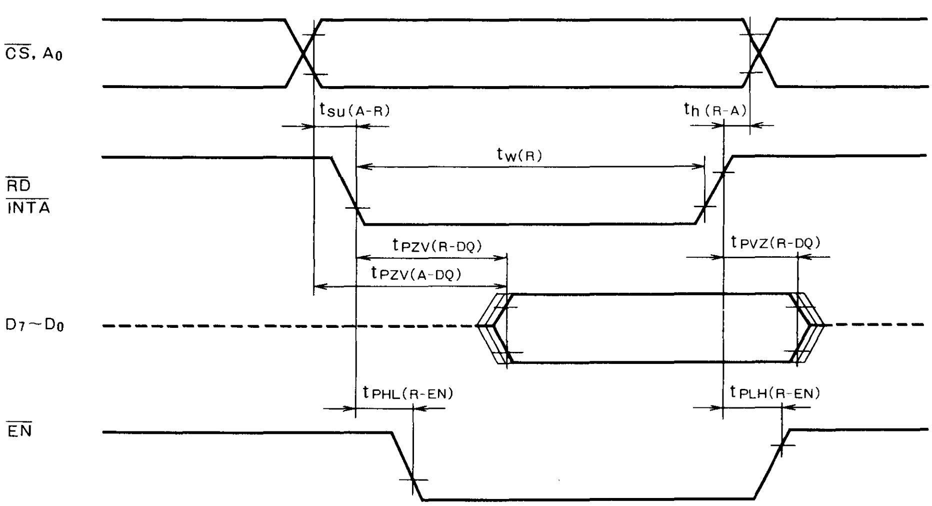


TIMING DIAGRAM

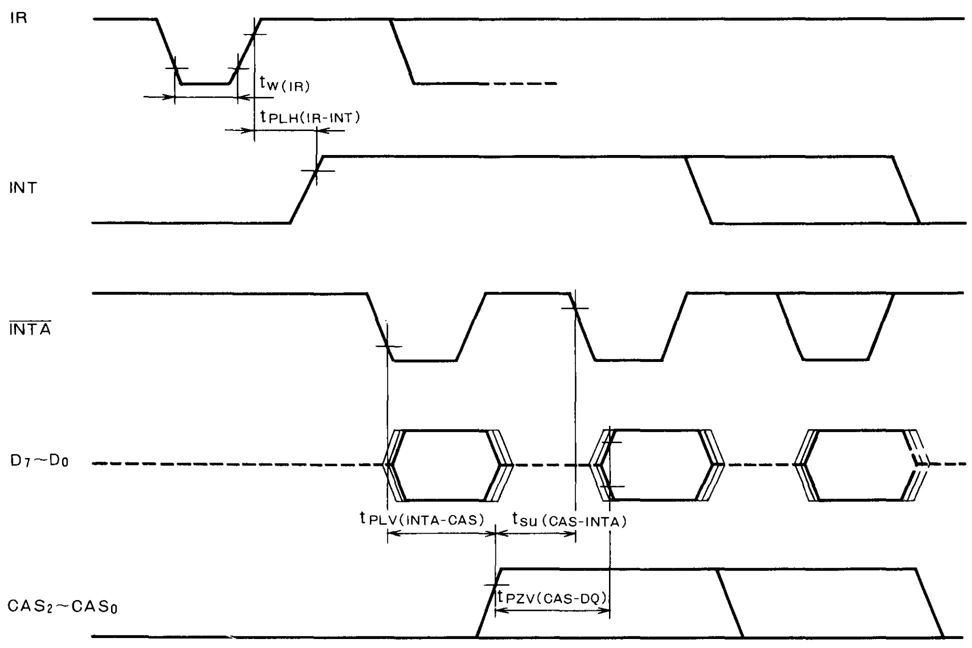
Write Mode



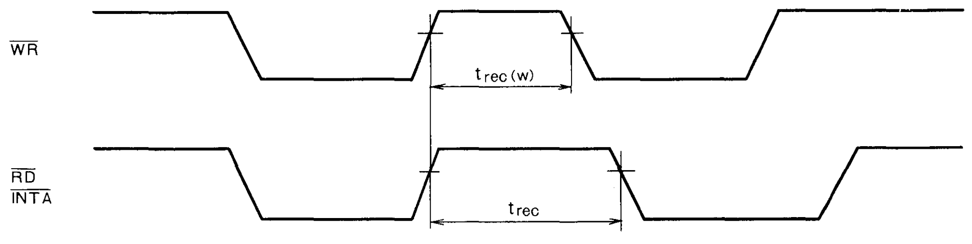
Read Mode



Interrupt Sequence



Other Timing



Note 1 M5L8086S mode
2 M5L8080AP/M5L8085AP mode
3 M5L8086S mode is in high-impedance state, pointer is released during the next $\overline{INTA}$.
When in single M5L8080AP/M5L8085AP mode, data is released by all $\overline{INTA}$s. When master, CALL instruction is released during the first $\overline{INTA}$, high impedance state during the second and third $\overline{INTA}$. When slave, high impedance state during the first $\overline{INTA}$, vectored address is released during the second and third $\overline{INTA}$.

PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE

DESCRIPTION

The M5L8279P-5 is a programmable keyboard and display interface device that is designed to be used in combination with an 8-bit microprocessor such as the Mitsubishi MELPS 8 CPUs. This device is fabricated with N-channel silicon-gate technology and is packed in a 40-pin DIL package. It needs only single 5V power supply.

FEATURES

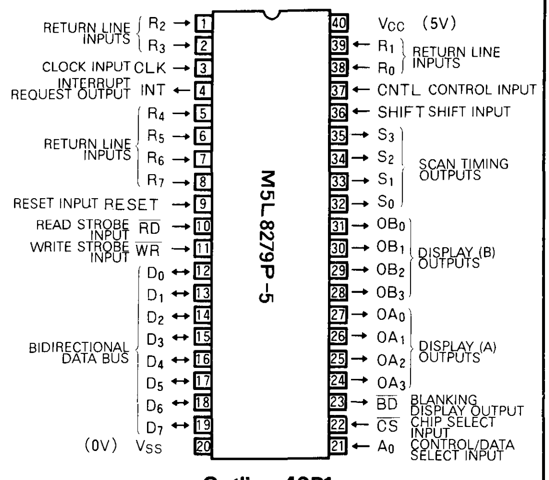
Parameter	M5L 8279P-5
Output enable time after read (max)	200ns
Output enable time after address (max)	250ns
Clock cycle time (min)	320ns

- Single 5V power supply
- Keyboard mode
- Sensor mode
- Strobed entry mode
- Internally provided key bounce protection circuit
- Programmable debounce time
- 2-key/N-key rollover
- 8-character keyboard FIFO
- Internally contained 16 × 8-bit display RAM
- Programmable right and left entry
- Interchangeable with Intel's 8279/8279-5 in pin configuration and electrical characteristics

APPLICATIONS

- Microcomputer I/O device
- 64 contact key input device for such items as electronic cash registers
- Dual 8- or single 16-alphanumeric display

PIN CONFIGURATION (TOP VIEW)



Outline 40P1

FUNCTION

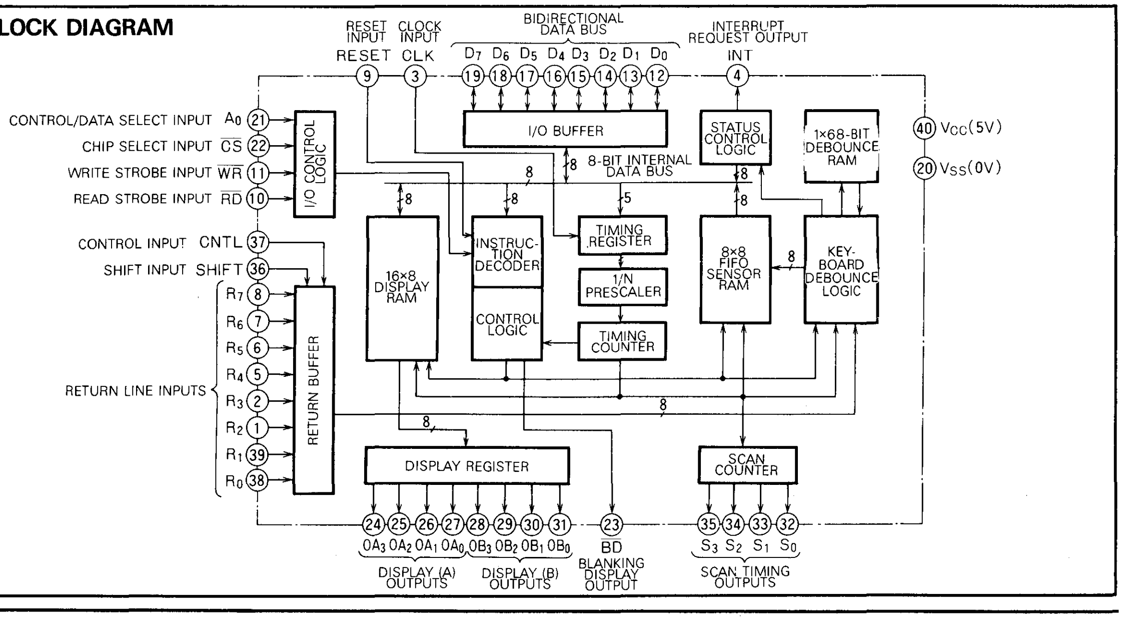
The total chip, consisting of a keyboard interface and a display interface, can be programmed by eight 8-bit commands.

The keyboard portion is provided with a 64-bit key debounce buffer and an 8 × 8-bit FIFO. It operates in any one of the scanned keyboard mode, scanned sensor mode or strobed entry mode.

The display portion is provided with a 16 × 8-bit display RAM that can be organized into a dual 16 × 4 configuration. Also, an 8-digit display configuration is possible by means of programming.

8

BLOCK DIAGRAM



PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE

PIN DISCRIPTON

Pin	Name	Input or output	Functions
$D_0 \sim D_7$	Bidirectional data bus	In/out	All data and commands between the CPU and the chip are transferred through these lines.
CLK	Clock input	In	Clock signal from the system which is used to generate internal timing.
RESET	Reset input	In	Resets the chip when this signal is high. After the reset it assumes 8-digit, left-entry, encode display, and 2-key rollover mode, and the prescale value of the clock becomes 31. The display RAM, however, is not cleared.
$\overline{CS}$	Chip select input	In	Chip select is enabled when this signal is low.
A_0	Control/data select input	In	When this signal is high, it indicates that the signals in and out are either command (in) or status (out). When low, it indicates they are data (in/out).
$\overline{RD}$	Read strobe input	In	Functions to control data transfer to the data bus.
$\overline{WR}$	Write strobe input	In	Functions to control command/data transfer from the data bus.
INT	Interrupt request output	Out	When there is any data in the FIFO during the keyboard mode or the strobed mode, this signal turns high-level so as to request interrupt to the CPU. It turns low each time data is read, but if any data remains in the FIFO it will turn high again and request interrupt to the CPU.
$S_0 \sim S_3$	Scan timing outputs	Out	These signals are used to scan the key switch, the sensor matrix, or the display digit. They can be either decoded or encoded, but it requres an external decoder in the encode mode. Signals $S_0 \sim S_3$ are all turned to low-level when RESET is high.
$R_0 \sim R_7$	Return line inputs	In	These are the return lines which are connected with the scan lines through the keys or sensor switches, and are used for 8-bit input in the strobed entry mode. They are provided with internal pullups to maintain them high until a switch closure pulls one low. They become active at low-level.
SHIFT	Shift input	In	In the keyboard mode, the shift input becomes the second highest bit of the key input information and is stored in the FIFO. This input is ignored in the other modes. It is constantly kept at high-level by an internal pull resistor.
CNTL	Control input	In	In the keyboard mode, the control input becomes the most significant bit of the key input information and is stored in the FIFO. The signal is active at high-level. In the strobed entry mode, it becomes the strobe signal and stores the return input data in the FIFO at the rising edge of the input. It affects nothing internal in the sensor mode. It is constantly kept at high-level by an internal pullup resistor.
$OA_0 \sim OA_3$ $OB_0 \sim OB_3$	Display (A) and (B) outputs	Out	These output ports can be used either as a dual 4-bit port or a single 8-bit port depending on an application, and the contents of the display RAM are output synchronizing with the scan timing signals. These two 4-bit ports may be blanked independently. Blanking may be activated with either high- or low-level signal by means of clear command.
$\overline{BD}$	Blanking display output	Out	This signal is used in preventing overlapped display during digit switching. It also may be brought to low-level by display blanking command.

OPERATION

Of the three operating modes, the keyboard mode is the most common, and allows programmed 2-key lockout and N-key rollover. Encoded timing signals corresponding with key input are stored in the FIFO through the key-debounce logic, and the debouncing time of the key is also programmable. In the sensor mode, the contents of the 8 × 8 key contacts are constantly stored in the FIFO/sensor RAM, generating an interrupt signal to the CPU each time there is a change in the contents. In the strobed entry mode, the CNTL input signal is used as a strobe for storing the 8 return line inputs to the FIFO/sensor RAM.

The display portion is provided with a 16 × 8-bit display RAM that can be organized into a dual 16 × 4-bit configuration. Also, an 8-digit display configuration is possible by means of programming. Input to the register can be performed by either left or right entry modes. In the auto increment mode, read and write can be carried out after designating the starting address only.

Both the keyboard and display sections are scanned by common scan timing signals that are derived from the basic clock pulse. This frequency-dividing ratio is changeable by means of programming. There are decode and encode modes for the scanning mode; timing signals that are decoded from the lower 2 bits of the scan counter are output in the decode mode, while the 4-bit binary output from the scan counter is decoded externally in the encode mode.

COMMAND DESCRIPTION

There are eight commands provided for programming the operating modes of the M5L8279P. These commands are sent on the data bus with the signal $\overline{CS}$ in low-level and the signal A_0 in high-level and are stored in the M5L8279P at the rising edge of the signal $\overline{WR}$.

1. Mode Set Command

Code : (MSB → LSB)

MSB							LSB
0	0	0	D	D	K	K	K

DD (Display mode set command)

- 0 0 8—8-bit character display—left entry
- 0 1 16—8-bit character display—left entry[1]
- 1 0 8—8-bit character display—right entry
- 1 1 16—8-bit character display—right entry

KKK (Keyboard mode set command)

- 0 0 0 Encoded display keyboard mode — 2-key lockout[1]
- 0 0 1 Decoded display keyboard mode — 2-key lockout
- 0 1 0 Encoded display keyboard mode — N-key rollover
- 0 1 1 Decoded display keyboard mode — N-key rollover
- 1 0 0 Encoded display, sensor mode
- 1 0 1 Decoded display, sensor mode
- 1 1 0 Encoded display, strobed entry mode
- 1 1 1 Decoded display, strobed entry mode

Note 1 : Default after reset.

2. Program Clock Command

Code :

MSB							LSB
0	0	1	P	P	P	P	P

The external clock is divided by the prescaler value PPPPP designated by this command to obtain the basic internal frequency.

When the internal clock is set to 100kHz, it will give a 5.1ms keyboard scan time and a 10.3ms debounce time. The prescale value that can be specified by PPPPP is from 2 to 31. In case PPPPP is 00000 or 00001, the prescale is set to 2. Default after a reset pulse is 31, but the prescale value is not cleared by the clear command.

3. Read FIFO Command

Code :

MSB							LSB
0	1	0	AI	X	A	A	A

X = Don't care

This command is used to specify that the following data readout ($CS \cdot \overline{A_0} \cdot RD$) is from the FIFO. As long as data is to be read from the FIFO, no additional commands are necessary.

AI and AAA are used only in the sensor mode. AAA designates the address of the FIFO to be read, and AI is the auto-increment flag. Turning AI to "1" makes the address automatically incremented after the second read operation. This auto-increment bit does not affect the auto-increment of the display RAM.

4. Read Display RAM Command

Code :

MSB							LSB
0	1	1	AI	A	A	A	A

This command is used to specify that the following data readout ($CS \cdot \overline{A_0} \cdot RD$) is from the display RAM. As long as data is to be read from the display RAM, no additional commands are necessary.

The data AAAA is the value with which the display RAM read/write counter is set, and it specifies the address of the display RAM to be read or written next.

AI is the auto-increment flag. Turning AI to "1" makes the address automatically incremented after the second read/write operation. This auto-increment bit does not affect the auto-increment of FIFO readout in the sensor mode.

5. Write Display RAM Command

Code :

MSB							LSB
1	0	0	AI	A	A	A	A

With this command, following display RAM read/write addressing is achieved without changing the data readout source (FIFO or display RAM). Meaning of AI and AAAA are identical with read display RAM command.

6. Display Write Inhibit/Blanking Command

Code :

MSB							LSB
1	0	1	X	IW	IW	BL	BL
				A	B	A	B

X = Don't care

The IW is a write inhibit bit to the display RAM that corresponds with the output A or B. Inhibit is activated by turning the IW "1".

The BL is used in blanking the out A or B. Blanking is activated by turning the BL "1". Setting both BL flags makes the signal $\overline{BD}$ low so that it can be used in 8-bit display mode.

Resetting the flags makes all IW and BL turn "0".

7. Clear command

Code :

MBS							LSB
1	1	0	C_D	C_D	C_D	C_F	C_A

C_D: Clears the display RAM.

C_D	C_D	C_D	
0	X	X	No specific performance
1	0	X	Entire contents of the display RAM are turned "0".
1	1	0	The contents of the display RAM are turned 20H ($00100000 = 0A_30A_20A_10A_0$ $0B_30B_20B_10B_0$).
1	1	1	Entire contents of the display RAM are turned "1".

C_F: Clears the status word and resets the interrupt signal (INT).

C_A: Clears the display RAM and the status word and resets the interrupt signal (INT).

Clearing condition of the display RAM is determined by the lower 2 bits of the C_D.

Clearing the display RAM needs a whole display scan cycle and causes the display-unavailable status (DU) in the status word to be "1". The display RAM is not accessible for the duration of a scan cycle (scan time for 16 digits), even if the display mode was in 8-digit display mode or a decoded mode.

As both C_F and C_A function to reset the internal key-debounce counter, the key input under counting is ignored, and the internal FIFO counter is reset to make the interrupt signal low-level.

C_A resets the internal timing counter, forcing $S_0 \sim S_3$ to start from $S_3S_2S_1S_0$ = 0000 after the execution of the command.

8. End Interrupt/Error Mode Set Command

Code (MSB → LSB):

1	1	1	E	X	X	X	X

X = Don't care

In the sensor matrix mode, an interrupt signal is generated at the beginning of the next key scan time to inhibit further writing to the FIFO when there is a change in the sensor switch, but execution of this command makes the interrupt signal released so as to allow writing to the FIFO.

When E is kept in "0", depression of any sensor makes the second highest bit of the status word "1". When E is kept in "1", the status is kept "0" all the time.

When E is programmed to "1" in the N-key rollover mode, the execution of this command makes the chip operate in special error mode, during which time depression of more than two keys in a key scan time causes an error and sets the second highest bit of the status word "1".

Status word

MSB → LSB

DU	S/E	O	U	F	N	N	N

NNN: Indicates the number of characters in the FIFO during the keyboard and strobed entry modes.

F: Indicates that the FIFO is filled up with 8 characters.
The number of characters existing in the FIFO (0~8 characters) can be known by means of the bits NNN and F (FNNN = 0000~FNNN = 1000).

U: Underrun error flag
This flag is set when a master CPU tries to read an empty FIFO.

O: Overrun error flag
This flag is set when another character is strobed into a full FIFO.
The bits U and O cannot be cleared by status read. They will be cleared by the clear command.

S/E: Sensor closure/multiple error flag
When "111EXXXX" is executed by turning E = 0, the bit S/E in the status word is set when there is at least one sensor closure.
When "111EXXXX" is executed by turning E = 1 (special error mode), the bit S/E is set when there are more than two key depressions made in a key scan time.

DU: Display unavailable
This flag is set during a whole display scan cycle when a clear display command is executed, and announces that the display RAM is not accessible.

Note: It is necessary to execute the clear command (C_F=1) to reset the underrun, overrun, and special error flags.

CPU INTERFACE

1. Command Write

A command is written on the rising edge of the signal $\overline{WR}$ with $\overline{CS}$ low and A_0 high.

2. Data Write

Data is written to the display RAM on the rising edge of the signal $\overline{WR}$ with $\overline{CS}$ and A_0 low.

The address of the display RAM is also incremented on the rising edge of the signal WR if AI is set for the display RAM.

3. Status Read

The status word is read when $\overline{CS}$ and $\overline{RD}$ are low and A_0 is high. The status word appears on the data bus as long as the signal $\overline{RD}$ is low.

4. Data Read

Data is read from either the FIFO or the display RAM with $\overline{CS}$ = $\overline{RD}$ = 0 and A_0 = 1. The source of the data (FIFO or display RAM) is decided by the latest command (read display or read FIFO). The data read appears on the data bus as long as the signal $\overline{RD}$ is low.

The trailing edge of the signal $\overline{RD}$ increments the address of the FIFO or the display RAM when AI is set. After the reset, data will be read from the FIFO, however.

$\overline{CS}$	A_0	$\overline{RD}$	$\overline{WR}$	Operation
0	1	1	0	Command write
0	0	1	0	Data write
0	1	0	1	Status read
0	0	0	1	Data read
1	X	X	X	No operation

KEYBOARD INTERFACE

Keyboard interface is done by the scan timing signals (S_0~S_3), the return line inputs (R_0~R_7), the SHIFT and the CNTRL inputs.

In the decoded mode, the low order of two bits of the internal scan counter are decoded and come out on the timing pins (S_0~S_3). In the encoded mode, the four binary bits of the scan counter are directly output on the timing pins, thus a 3-to-8 decoder must be employed to generate keyboard scan timing.

The return line inputs (R_0~R_7), the SHIFT and the CNTL inputs are pulled up high by internal pullup transistors until a switch closure pulls one low.

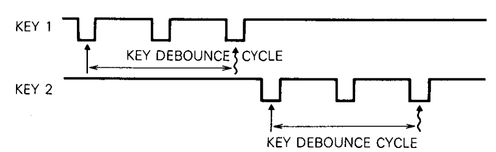
The internal key debounce logic works for a 64-key matrix that is obtained by combining the return line inputs with the scan timing.

For the keyboard interface, M5L8279P-5 has four distinctive modes that allow various kinds of applications. In the following explanation, a "key scan cycle" is the time needed to scan a 64-key matrix, and a "key debounce cycle" needs a duration of two "key scan" cycles. (In the decoded mode 32 keys, unlike 64 keys in the encoded mode, can be employed for a maximum key matrix due to the limit of timing signals. However, both the key scan cycle and the key debounce cycle are the same as in the encoded mode.)

1. 2-Key Lockout (Scanned Keyboard mode)

The detection of a new key closure resets the internal debounce counter and starts counting. At the end of a key debounce cycle, the key is checked and entered into the FIFO if it is still down. An entry in the FIFO sets the IRQ output high. If any other keys are depressed in a key debounce cycle, the internal key debounce counter is reset each time it encounters a new key. Thus only a single-key depression within a key debounce duration is accepted, but all keys are ignored when more than two keys are depressed at the same time.

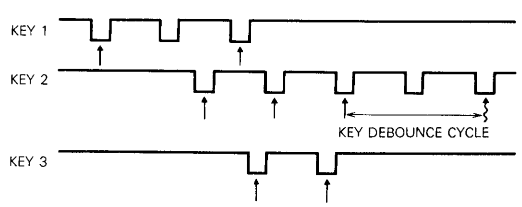
Example 1: Accepting two successive key depressions



Note 2 : ↑ : Debounce counter reset

⌇ : Key input

Example 2: Overlapped depression of three keys

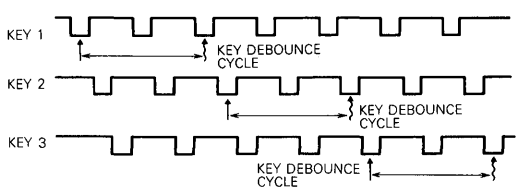


Note 3 : Only key 2 is acceptable.

8

2. N-Key Rollover (Scanned Keyboard Mode)

Each key depression is treated independently from all others so as to allow overlapped key depression. Detection of a new key depression makes the internal key debounce counter reset and start to count in a same manner as in the case of 2-key rollover. But, in N-key rollover, other key closures are entirely ignored within a key debounce cycle so that depression of any other keys would not reset the key debounce counter. In this way, overlapped key depression is allowed so as to enable the following key input:

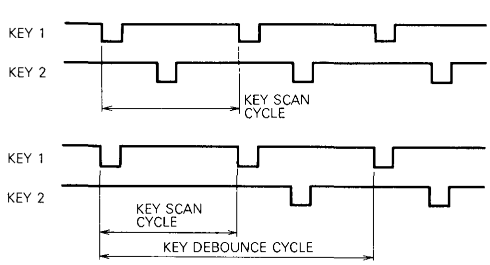


The scanned key input signal does not always reflect the actual key depressing action, as the key matrix is scanned by the timing signal.

With N-key rollover, there is a mode provided with which error is caused when there are more than two key inputs in a key scan cycle, which can be programmed by using the end interrupt/error mode set command. In this mode (special error mode), recognition of the above error sets the IRQ signal to "1" and sets the bit S/E in the status word.

In case two key entries are made separately in more than a debounce cycle, there would be no problem, as key depression is clearly identified. And no problem exists for 2-key lockout, as the both keys are recognized invalid.

Example of error



3. Sensor Matrix Mode

The key debounce logic is disabled in this mode. As the image of the sensor switch is kept in the FIFO, any change in this status is reported to the CPU by means of the interrupt signal INT. Although a debounce circuit is not used in this mode, it has an advantage in that the CPU is able to know how long and when the sensor was depressed.

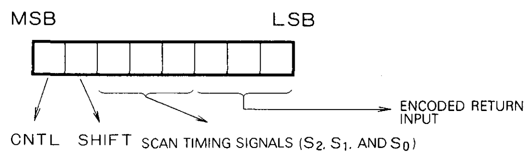
In the sensor matrix mode with the bit E = 0 of the end interrupt/error mode set command, the second most significant bit of the status word (S/E bit) is set to "1" when any sensor switch is depressed.

4. Strobe Mode

The data is entered into the FIFO from the return lines ($R_0 \sim R_7$) at the rising edge of a CNTL pulse. The INT goes high while any data exists in the FIFO, in the same manner as in the keyboard mode. The key debounce circuit will not operate.

Formats of data entered into the FIFO in each of the above modes are described in the following:

Keyboard matrix



Sensor matrix mode

MSB　　　　　　　　　LSB

R_7	R_6	R_5	R_4	R_3	R_2	R_1	R_0

CNTL AND SHIFT INPUTS ARE IGNORED

Strobe mode

MSB　　　　　　　　　LSB

R_7	R_6	R_5	R_4	R_3	R_2	R_1	R_0

CNTL AND SHIFT INPUTS ARE IGNORED.

DISPLAY INTERFACE

The display interface is done by eight display outputs ($OA_0 \sim OA_3$, $OB_0 \sim OB_3$), a blanking signal ($\overline{BD}$), and scan timing outputs ($S_0 \sim S_3$).

The relation between the data bus and the display outputs is as shown below:

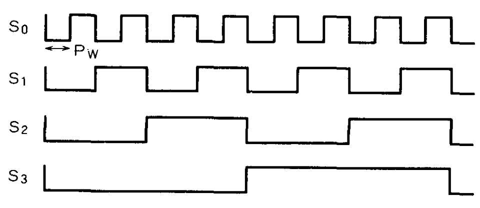
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
↓	↓	↓	↓	↓	↓	↓	↓
OA_3	OA_2	OA_1	OA_0	OB_3	OB_2	OB_1	OB_0

Clearing the display RAM is achieved by the reset signal (9-pin) but requires the execution of the clear command.

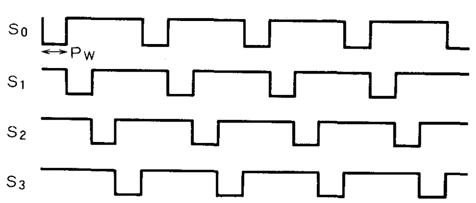
The timing diagrams for both the encoded and decoded modes are shown below.

For the encoded mode, a 3-to-8 or 4-to-16 decoder is required, according to whether eight or sixteen digit display used.

(1) Encoded mode

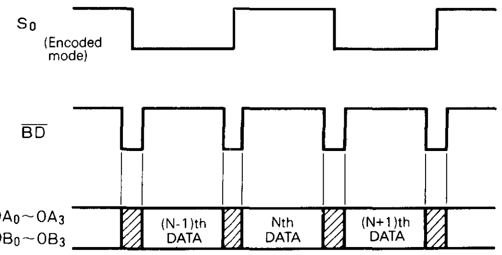


(2) Decoded mode



Note 4 : Here P_W is 640μs if the internal clock frequency is set to 100kHz.

Timing relations of S, $\overline{BD}$, and display outputs ($OA_0 \sim OA_3$, $OB_0 \sim OB_3$) are shown below:



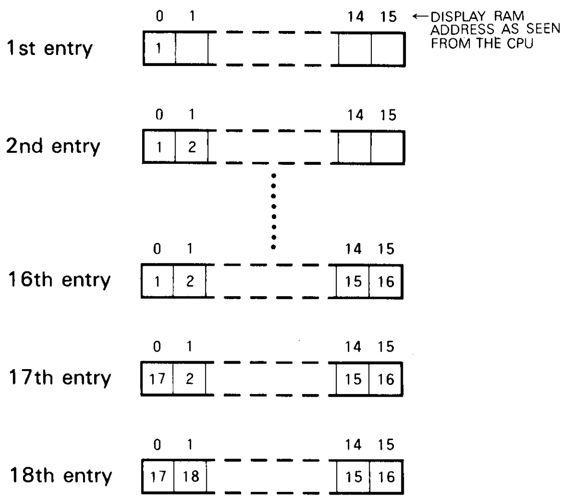
Note 5 : Values of the output data shown in the slantd line areas are decided upon the clear command executed last to become the value of the display RAM after the reset. The values in the slanted areas after reset will go low. In the same manner, the values $OA_0 \sim OA_3$, $OB_0 \sim OB_3$ are dependent on the clear command executed last. When the both A and B are blanked, the signal $\overline{BD}$ will be in low-level.

8

KEY ENTRY METHODS

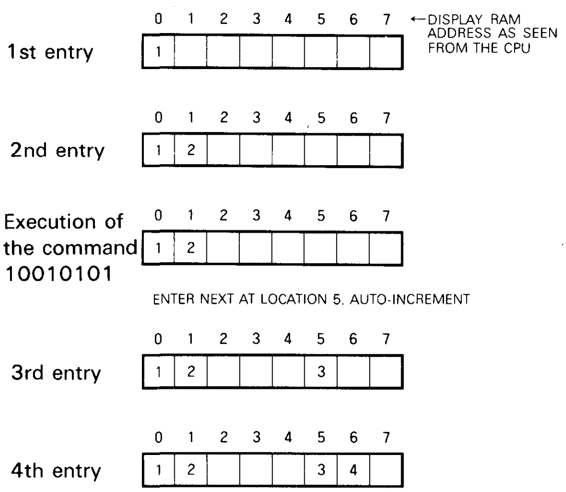
1. Left Entry

Address 0 in the display RAM corresponds to the leftmost position ($S_3S_2S_1S_0$ = 0000) of a display and address 15 (or address 7 in 8-character display) to the rightmost position ($S_3S_2S_1S_0$ = 111 or $S_2S_1S_0$ = 111). The 17th (9th) character is entered back into the leftmost position.



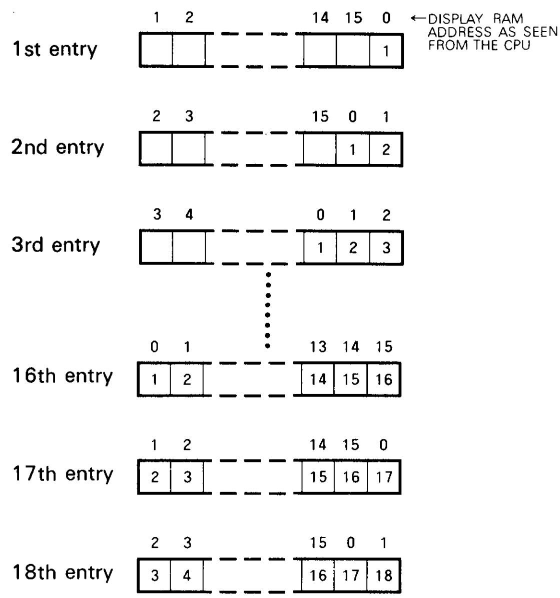
LEFT ENTRY

Auto-increment mode

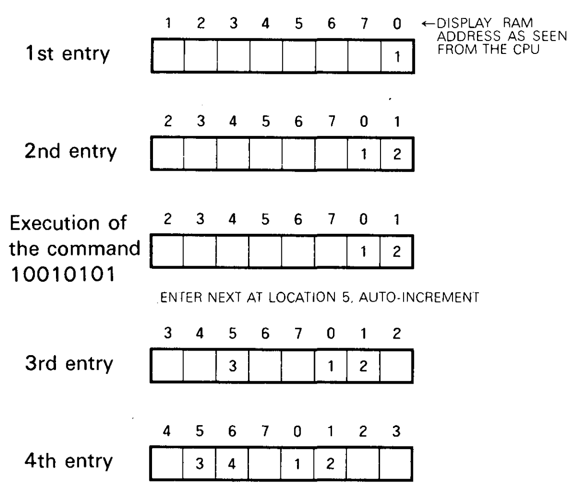


2. Right Entry

The first data is entered in the rightmost position ($S_3S_2S_1S_0$ = 0000 in 16-character display) of a display. From the next entry, the display is shifted left one character and the new data is placed in the rightmost position. A display position and a register address as viewed from the CPU change each each time and do not correspond.



Auto-increment mode



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.5~7	V
V_I	Input voltage		−0.5~7	V
V_O	Output voltage		−0.5~7	V
P_d	Maximum power dissipation	Ta=25℃	1000	mW
T_{opr}	Operating free-air temperature range		0~70	℃
T_{stg}	Storage temperature range		−60~150	℃

RECOMMENDED OPERATING CONDITIONS (Ta=0~70℃, unless otherwise noted.)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage		0		V
$V_{IH(RL)}$	High-level input voltage, for return line inputs	2.2			V
V_{IH}	High-level input voltage, all others	2			V
$V_{IL(RL)}$	Low-level input voltage, for return line inputs	V_{SS}−0.5		1.4	V
V_{IL}	Low-level input voltage, all others	V_{SS}−0.5		0.8	V

ELECTRICAL CHARACTERISTICS (Ta=0~70℃ V_{CC} = (Note 6), V_{SS}=0V, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	I_{OH}=−400μA	2.4			V
$V_{OH(INT)}$	Low-level output voltage, interrupt request output	I_{OH}=−300μA	3.5			V
V_{OL}	Low-level output voltage	I_{OL}=2.2mA			0.45	V
I_{CC}	Supply current from V_{CC}				120	mA
$I_{I(RL)}$	Input current, return line inputs, shift input and control input	$V_I=V_{CC}$			10	μA
		V_I=0V	−100			μA
I_I	Input current, all others	$V_I=V_{CC}$~0V	−10		10	μA
I_{OZ}	Off-state output current	$V_I=V_{CC}$~0V	−10		10	μA
C_i	Input capacitance	$V_I=V_{CC}$	5		10	pF
C_O	Output capacitance	$V_O=V_{CC}$	10		20	pF

TIMING REQUIREMENTS ($T_a = 0 \sim 70$°C, V_{CC} = (Note 6), $V_{SS} = 0V$, unless otherwise noted.)

Read Cycle

Symbol	Parameter	Alternative symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{C(R)}$	Read cycle time	t_{RCY}	(Note 6)	1000			ns
$t_{W(R)}$	Read pulse width	t_{RR}		250			ns
$t_{SU(A-R)}$	Address setup time before RD	t_{AR}		0			ns
$t_{h(R-A)}$	Address setup time after RD	t_{RA}		0			ns

Write Cycle

Symbol	Parameter	Alternative symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{W(W)}$	Write pulse width	t_{WW}	(Note 6)	250			ns
$t_{SU(A-W)}$	Address setup time before WR	t_{AW}		0			ns
$t_{h(W-A)}$	Address hold time after WR	t_{WA}		0			ns
$t_{SU(DQ-W)}$	Data input setup time before WR	t_{DW}		150			ns
$t_{h(W-DQ)}$	Data input hold time after WR	t_{WD}		0			ns

Other Timings

Symbol	Parameter	Alternative symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{C(\phi)}$	Clock cycle time	t_{CY}	(Note 6)	320			ns
$t_{W(\phi)}$	Clock pulse width	$t_{\phi W}$		120			ns

For an internal clock frequency of 100kHz

- Key scan cycle time: ~ 5.1ms
- Key debounce cycle time: ~ 10.3ms
- Single-key scan time: 80μs
- Display scan time: ~ 10.3ms
- Single digit display time: 490μs
- Blanking time: 150μs
- Internal clock cycle: 10μs

Note 6 : Test conditions:
Input pulse level: 0.45~2.4V
Input pulse rise time: 20ns
Input pulse fall time: 20ns
High-level input reference level: 2V
Low-level input reference level: 0.8V
$C_L = 150pF$

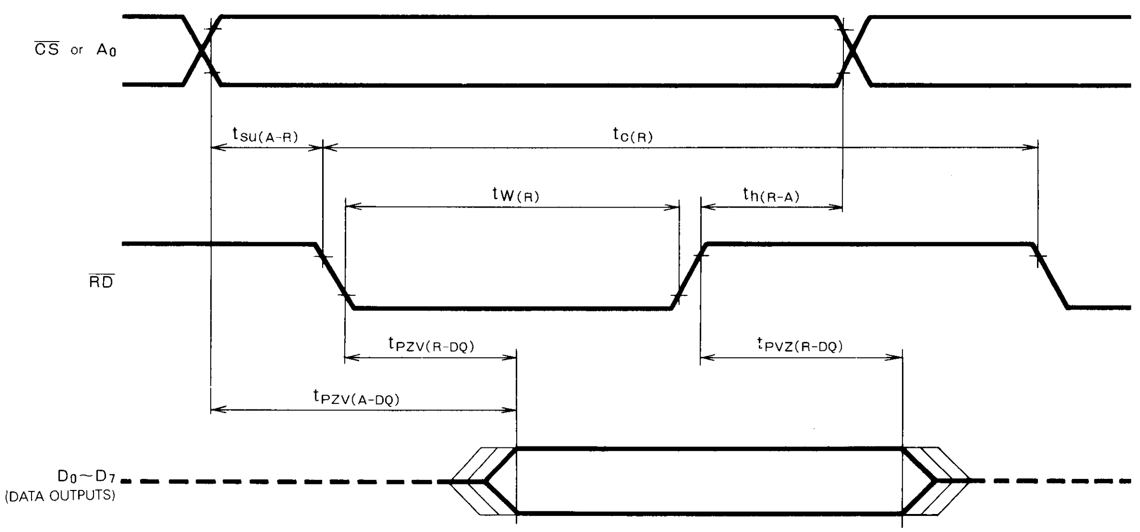
SWITCHING CHARACTERISTICS ($T_a = 0 \sim 70$°C, V_{CC} = (Note 1), $V_{SS} = 0V$, unless otherwise noted.)

Symbol	Parameter	Alternative symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{PZV(R-DQ)}$	Output enable time after read	t_{RD}	(Note 7)			200	ns
$t_{PZV(A-DQ)}$	Output enable time after address	t_{AD}				250	ns
$t_{PVZ(R-DQ)}$	Output disable time after read	t_{DF}		10		100	ns

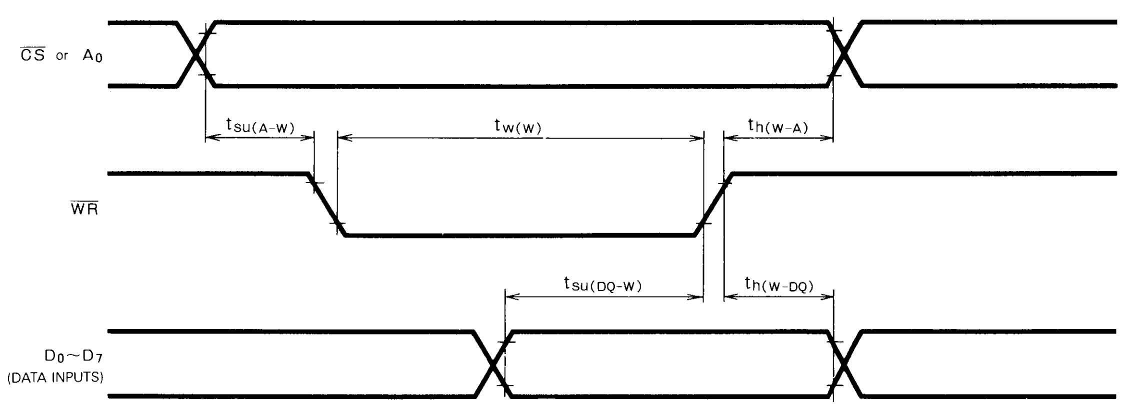
Note 7 : Test conditions
Input pulse level: 0.45~2.4V
Input pulse rise time: 20ns
Input pulse fall time: 20ns
High-level input reference voltage: 2V
Low-level input reference voltage: 0.8V
High-level output reference voltage: 2V
Low-level output reference voltage : 0.8V
$C_L = 150pF$

TIMING DIAGRAM

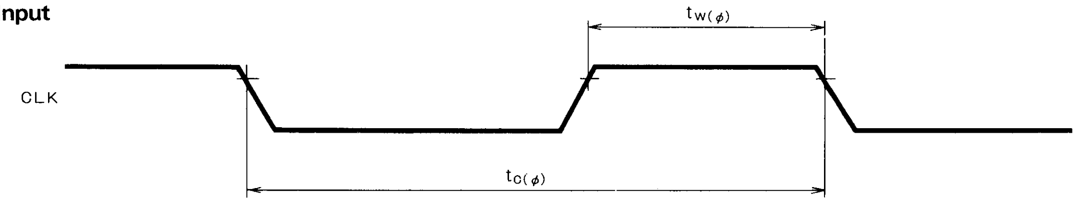
Read Mode



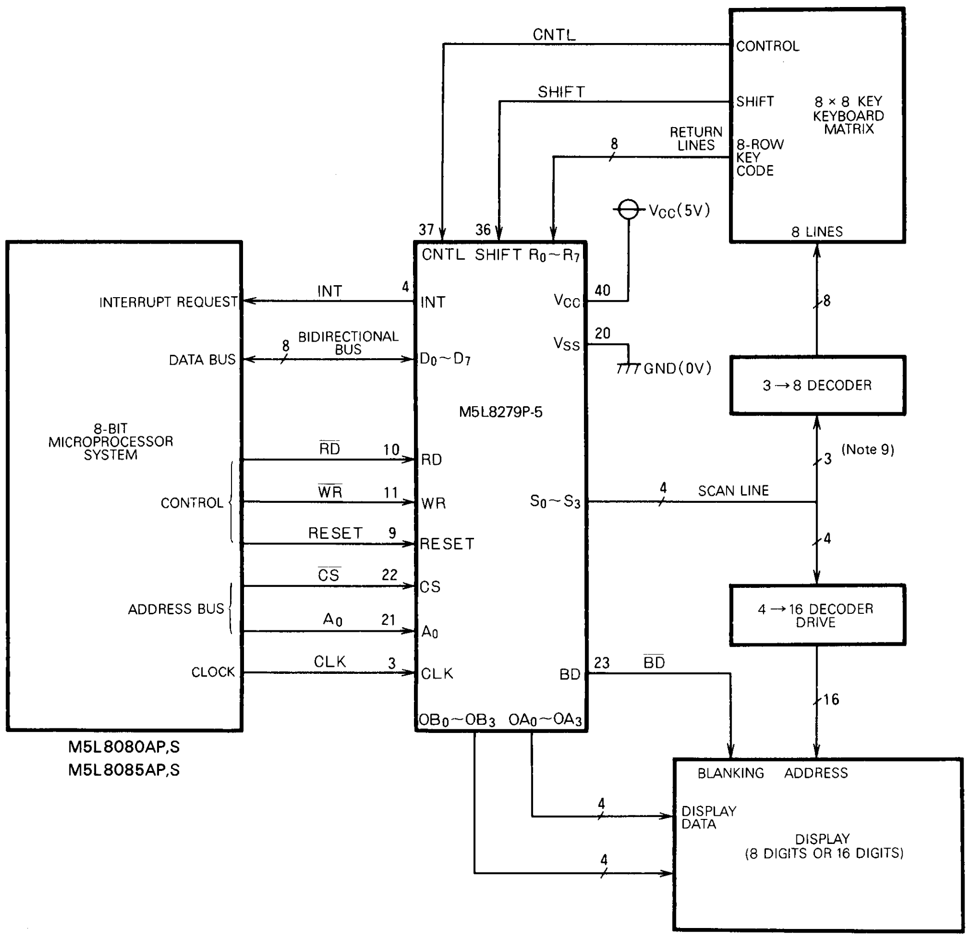
Write Mode



Clock Input



APPLICATION EXAMPLE



Note 8 When using an 8-bit character display of more than 9 digits for the decoder display, it is necessary to provide a separate decoder (for example 4→10 decoder, 4→16 decoder) and key scan 3→8 decoder. Only S_0, S_1 and S_2 may be used as inputs to the key scan 3→8 decoder.

9: Don't drive the keyboard decoder with the MSB of the scan line.

FLOPPY DISK FORMATTER/CONTROLLER

DESCRIPTION

The M5W1791-02P is a floppy disk formatter/controller device which accommodates single and double density formats. The device is designed for use with microprocessors or microcomputers. The device is fabricated with the N-channel silicon gate ED-MOS technology and is packaged in a 40-pin DIL package.

FEATURES

- Single 5V power supply
- Accommodate single and double density formats
  IBM 3740 single density format
  IBM system 34 double density format
- Selectable sector length (128,256,512 or 1024 bytes/sector)
- Side select compare
- Single/multiple sector read or write with automatic sector search
- Selectable track to track stepping time
- Write precompensation
- DMA or programmed data transfers
- Window extension
- Interchangeable with Western Digital's FD1791-02 in function except for V_{DD} power supply and pin configuration

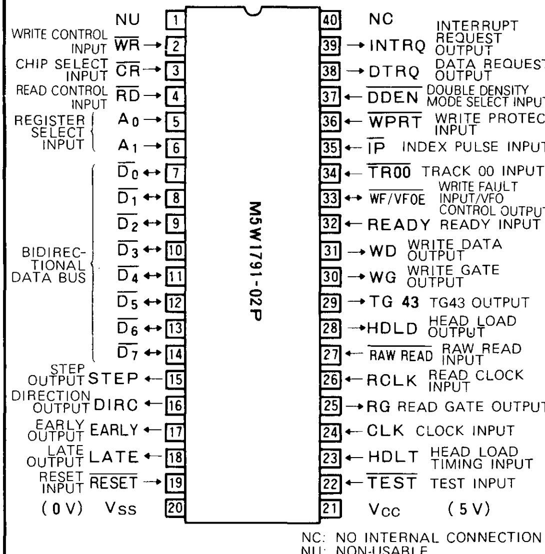
APPLICATIONS

- Single or double density floppy disk drive formatter/controller
- 8-inch or mini floppy disk interface

FUNCTION

The M5W1791-02P is a floppy disk formatter/controller that can be used with most microprocessor or microcomputer systems. The hardware of the M5W1791-02P consists of a floppy disk interface, a CPU interface and a PLA control logic. The total chip can be programmed by eleven 8-bit commands. The floppy disk interface portion performs the communication with the floppy disk drive under control of the PLA control logic. The CPU interface portion has five registers — command, dara, status, track and sector register — and communicates with the CPU through the data bus. These functions are also controlled by the PLA.

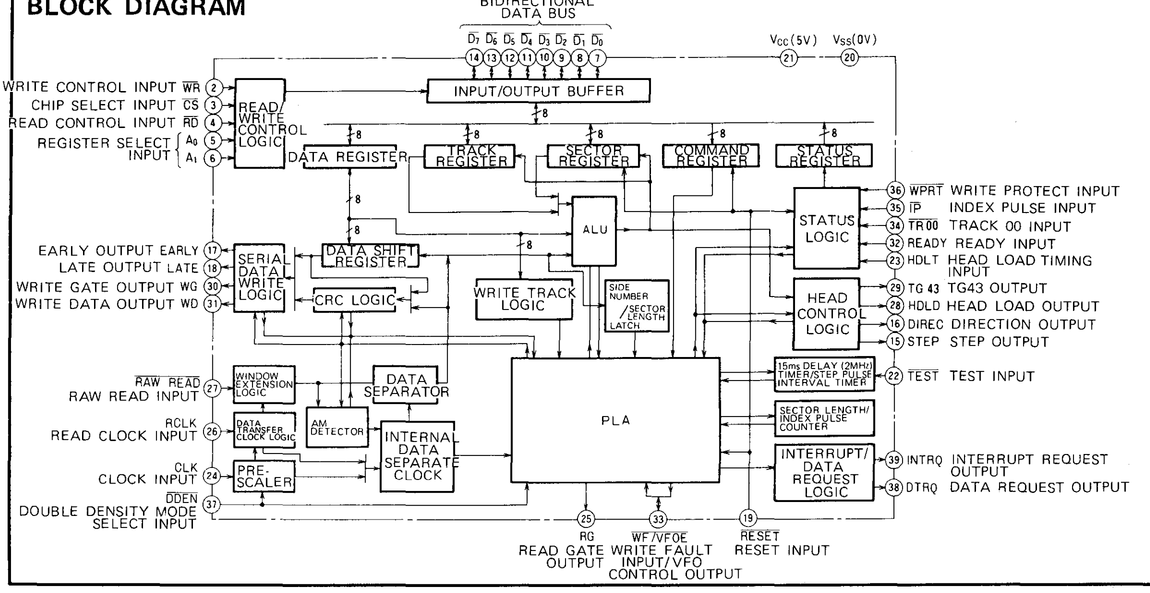
PIN CONFIGURATION (TOP VIEW)



NC: NO INTERNAL CONNECTION
NU: NON-USABLE

Outline 40P1

BLOCK DIAGRAM



PIN DESCRIPTION

Pin	Name	Input or output	Functions
NU	Non-usable terminal		NU (pin 1) is internally connected to the back gate bias generater, so it must remain open.
NC	No internal connection		NC (pin 40) is not internally connected.
$\overline{RESET}$	Reset input	Input	Reset input (Active low). The device is reset by this signal and automatically loads 0316 into the command register. The not-ready-status bit is also reset by this signal. When reset input is made to be high, the device executes restore command unless READY is active and the device loads 0116 to the sector register.
$\overline{WR}$	Write control input	Input	Write signal from a master CPU (Active low).
$\overline{CS}$	Chip select input	Input	Chip select (Active low).
$\overline{RD}$	Read control input	Input	Read signal from a master CPU (Active low).
A_0, A_1	Register select input	Input	Register select inputs. These inputs select the register under the control of the $\overline{RD}$ and $\overline{WR}$. A_1 / A_0 / $\overline{RD}$ / $\overline{WR}$ 0 / 0 / STATUS REGISTER / COMMAND REGISTER 0 / 1 / TRACK REGISTER / TRACK REGISTER 1 / 0 / SECTOR REGISTER / SECTOR REGISTER 1 / 1 / DATA REGISTER / DATA REGISTER
$\overline{D_0}$~$\overline{D_7}$	Bidirectional data bus	In/Out	Three-state, inverted bidirectional data bus.
CLK	Clock input	Input	Clock input to generate internal timing, 2MHz for 8-inch drives, 1MHz for mini drives.
DTRQ	Data request output	Output	DTRQ is an open drain output, so pull up to V_{CC} by the 10k resistor. In the disk read mode, DTRQ indicates that data is assembled in the data register. In the disk write mode, it indicates that the data register is empty. DTRQ is reset by the read data or write data operation.
INTRQ	Interrupt request output	Output	INTRQ is also a open drain output, so pull up to V_{CC} by the 10k resistor. INTRQ becomes active at the completion of any command and is reset when the CPU reads the status or writes the command.
STEP	Step output	Output	Step pulse output (Active high).
DIRC	Direction output	Output	Direction output. High level means the head is stepping in and low level means the head is stepping out.
EARLY	Early output	Output	This signal is used for write precompensation. It indicates that the write data pulse should be shifted early.
LATE	Late output	Output	This signal is also used for write precompensation. It indicates that the write data pulse should be shifted late.
$\overline{TEST}$	Test input	Input	This input is only used for test purposes, so user must tie it to V_{CC} or leave it open unless using voice coil actuated motors.
HDLT	Head load timing input	Input	When the device finds high level on this input, the device assumes that the head is engaged on the media. Active high.
RG	Read gate output	Output	This signal shows the external data separator that the synchfield is detected.

Pin	Name	Input or output	Functions
RCLK	Read clock input	Input	This signal is internally used for the data window. Phasing relation to raw read data is specified but polarity (RCLK high or low) is not important.
$\overline{\text{RAW READ}}$	Raw read input	Input	This input signal from the drive shall be low for each recorded flux transition.
HDLD	Head load output	Output	This output signal controls the loading of the head of the drive. The head must be loaded on the media by this high-level output.
TG43	TG 43 output	Output	This output is valid only during disk read/write operation and it shows the position of the head. High level on this output indicates that the head is positioned between track 44 to 76
WG	Write gate output	Output	This signal becomes active before disk write operations are to occur.
WD	Write data output	Output	This signal consists of data bits and clock bits. It becomes active for every flux transition. Active high.
READY	Ready input	Input	This signal shows the device the drive is ready. In the disk read/write operation except for TYPE 1 command operation, low-level input terminates current operation and the device generates the INTRQ. In the TYPE 1 command operation, this signal is neglected. Not ready bit in the status register is the inverted form of this input.
$\overline{\text{WF}}/\overline{\text{VFOE}}$	Write fault input/ VFO enable output	In/Out	This is a bidirectional signal. It becomes write fault input when WG is active. In the disk write operation, low level signal on this input terminates the write operation and makes INTRQ active. This signal also appears in the status register as the write fault bit. When WG is inactive, this signal works as VFO enable output. $\overline{\text{VFOE}}$ output is also an open drain type, so pull it up to V_{CC} and never input active write fault signal while WG is inactive.
$\overline{\text{TR00}}$	Track 00 input	Input	This signal indicates that the head is located on the track 00 to the device. Active low.
$\overline{\text{IP}}$	Index pulse input	Input	This input indicates to the device that an index hole of the diskette has been encountered.
$\overline{\text{WPRT}}$	Write protect input	Input	Low level signal on this input informs the device that the drive is in the write protected state. Before disk write operations, this signal is sampled and an active low signal will terminate the current command and set INTRQ. The write protect status bit in the status register is also set.
$\overline{\text{DDEN}}$	Double density mode select input	Input	This input determines the device operation mode. When DDEN=0, double density mode is selected. When DDEN=1, single density mode is selected.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.5~7	V
V_I	Input voltage		−0.5~7	V
V_O	Output voltage		−0.5~7	V
Pd	Power dissipation	Ta=25°C	1000	mW
Topr	Operating free-air temperature range		0~70	°C
Tstg	Storage temperature range		−65~150	°C

RECOMMENDED OPERATING CONDITIONS (Ta=0~70°C, unless otherwise noted)

Symbol	Paramter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage	2			V
V_{IL}	Low-level input voltage			0.8	V

ELECTRICAL CHARACTERISTICS (Ta=0~70°C, V_{CC}=5V ±5%, unless otherwise noted)

Symbol	Parameter	Test condition	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	$I_{OH}=-100\mu A$	2.4			V
V_{OL}	Low-level output voltage	$I_{OL}=1.6mA$			0.45	V
I_{CC}	Supply current				100	mA
I_I	Input current, other inputs	$V_I=V_{CC}\sim 0V$	−10		10	μA
I_{OZ}	Off-state output current	$V_I=V_{CC}\sim 0V$	−10		10	μA

TIMING REQUIREMENTS (Ta=0~70°C, V_{CC}=5V±5%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{su\ (A-R)}$ $t_{su\ (CS-R)}$	Address setup time before read and chip select	TSET		50			ns
$t_{h\ (R-A)}$ $t_{h\ (R-CS)}$	Address hold time after read and chip select	THLD		10			ns
$t_{w\ (R)}$	Read pulse width	TRE	C_L=50pf	400			ns
$t_{su\ (A-W)}$ $t_{su\ (CS-W)}$	Address setup time before write and chip select	TSET		50			ns
$t_{h\ (W-A)}$ $t_{h\ (W-CS)}$	Address hold time after write and chip select	THLD		10			ns
$t_{w\ (W)}$	Write pulse width	TWE		350			ns
$t_{su\ (DQ-W)}$	Data setup time before write	TDS		250			ns
$t_{h\ (W-DQ)}$	Data hold time after write	TDH		70			ns
$t_{w\ (RR)}$	Raw read pulse width	Tpw	(Note 1, 2)	100	200		ns
$t_{c\ (RR)}$	Raw read cycle time	tbc	(Note 3)		1500	1800	ns
$t_{w\ (RCLK)}$	Read clock high-level width	Ta	(Note 4, 5)	800			ns
$t_{w\ (RCLK)}$	Read clock low-level width	Tb	(Note 4, 5)	800			ns
$t_{c\ (RCLK)}$	Read clock cycle time	Tc			1500	1800	ns
$t_{h\ (RCLK-RR)}$	Read clock hold time before raw read	T_{X1}		40			ns
$t_{h\ (RR-RCLK)}$	Read clock hold time after raw read	T_{X2}	(Note 1)	40			ns
$t_{w\ (WD)}$	Write data pulse width	Twp	FM	450	500	550	ns
			MFM	150	200	250	ns
$t_{c\ (WD)}$	Write data cycle time	Tbc			2,3,4		μs
$t_{w\ (\phi)}$	Clock high-level pulse width	TCD_1		230	250	20000	ns
$t_{w\ (\phi)}$	Clock low-level pulse width	TCD_2		200	250	20000	ns
$t_{w\ (RESET)}$	Reset pulse width	TMR		50			μs
$t_{w\ (IP)}$	Index pulse width	TIP	(Note 5)	10			μs
$t_{w\ (WF)}$	Write fault pulse width	TWF	(Note 5)	10			μs

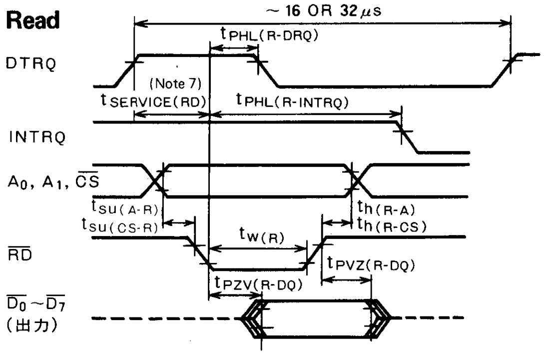
SWITCHING CHARACTERISTICS (Ta=0~70°C, V_{CC}=5V±5%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{PLH\ (WG-WD)}$	Propagation time from write gate to write data	Twg	FM		2		μs
			MFM		1		μs
$t_{PLH\ (E-WD)}$ $t_{PLH\ (L-WD)}$	Propagation time from early or late to write data	Ts	MFM	125			ns
$t_{PHL\ (WD-E)}$ $t_{PHL\ (WD-L)}$	Propagation time from write data to early or late	Th	MFM	125			ns
$t_{PHL\ (WD-WG)}$	Propagation time from write data to write gate	Twf	FM		2		μs
			MFM		1		μs
$t_{PZV\ (R-DQ)}$	Output enable time after read	TDACC	C_L=50pF			350	ns
$t_{PVZ\ (R-DQ)}$	Output disable time after read	TDOH	C_L=50pF	50		150	ns
$t_{PHL\ (R-DRQ)}$	Propagation time from read to DRQ	TDRR(RD)			400	500	ns
$t_{PHL\ (R-INTRQ)}$	Propagation time from read to INTRQ	TIRR(RD)	(Note 5)		500	3000	ns
$t_{PHL\ (W-DRQ)}$	Propagation time from write to DRQ	TDRR(WR)			400	500	ns
$t_{PHL\ (W-INTRQ)}$	Propagation time from write to INTRQ	TIRR(WR)	(Note 5)		500	3000	ns
$t_{w\ (STP)}$	Step pulse width	TSTP	(Note 5)	2 or 4			μs
$t_{PLH(DIR-STP)}$	Propagation time from direction to step	TDIR	(Note 5)	12			μs

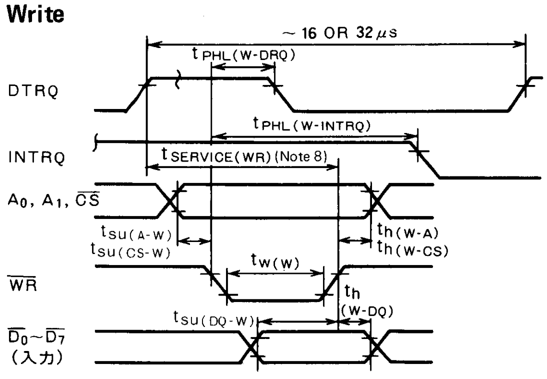
Note 1 : The pulse of $\overline{RAW}$ $\overline{READ}$ may be any width if pulse is entirely within RCLK When the pulse occurs in the RCLK window, $\overline{RAW}$ $\overline{READ}$ pulse width must be less than 300 ns for MFM mode and 600 ns for FM mode at CLK=2MHz. Times double for 1MHz.
2 : 100 ns pulse width is recommended for the $\overline{RAW}$ $\overline{READ}$ pulse in 8 MFM mode.
3 : $\overline{RAW\ READ}$ cycle time $T_{C(RR)}$ and WD cycle time $T_{C(WD)}$ is normally 2μs in MFM and 4μs in FM. Times double when CLK=1MHz.
4 : The polarity of RCLK during Raw $\overline{READ}$ is not important.
5 : When $\overline{VFOE}$=1, RCLK must be low level.
6 : Times double when CLK=1MHz.

TIMING DIAGRAM

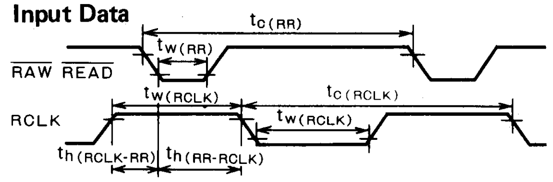
Read



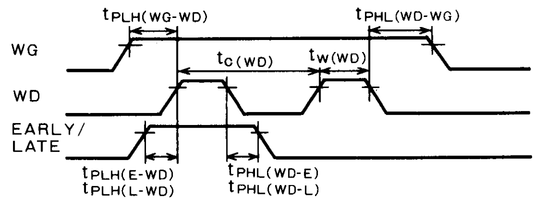
Write



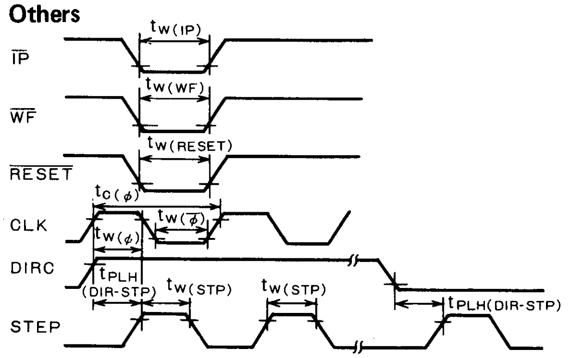
Input Data



Write Data



Others



Note 7: $t_{SERVICE(RD)}$ maximum value; FM: 27.5μs, MFM: 13.5μs
8: $t_{SERVICE(WR)}$ maximum value; FM: 28μs, MFM: 14μs

MELPS 86 MICROPROCESSORS

16-BIT PARALLEL MICROPROCESSOR

DESCRIPTION

The M5L8086S is a 16-bit parallel microprocessor fabricated using high-speed N-channel silicon-gate ED-MOS techology. It requires a single 5V power supply and has a maximum basic clock rate of 5MHz.

The M5L8086S is upward compatible, both in hardware and software, with the M5L8080AP, S and M5L8085AP, S therefore it can replace either of these devices. It has higher performance because of additional and more powerful operation and addressing functions and instructions.

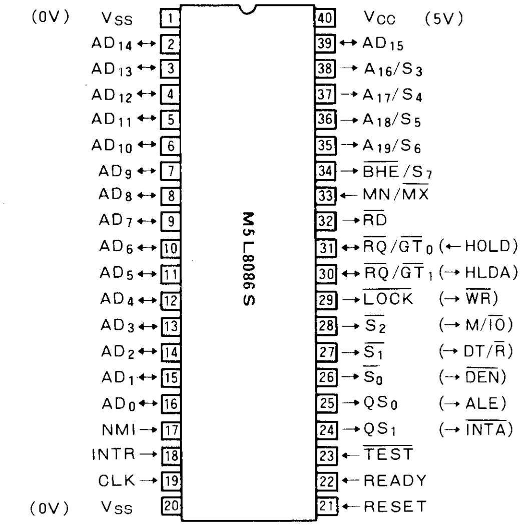
FEATURES

- Direct addressing: 1M byte
- Instruction set upward compatible with that of M5L–8080AP, S
- Enlarged powerful addressing: 24 modes
- On chip 16-bit registers: 14 registers
- Arithmetic operations include multiplication and division, signed or unsigned and 8-bit or 16-bit operands.
- Basic clock rate: 5MHz (max.)
- Multi-CPU functions
- Single 5V power supply
- Interchangeable with the Intel 8086 in pin configuration and electrical characteristics

APPLICATIONS

Central processing unit for 16-bit microcomputer and control units

PIN CONFIGURATION (TOP VIEW)



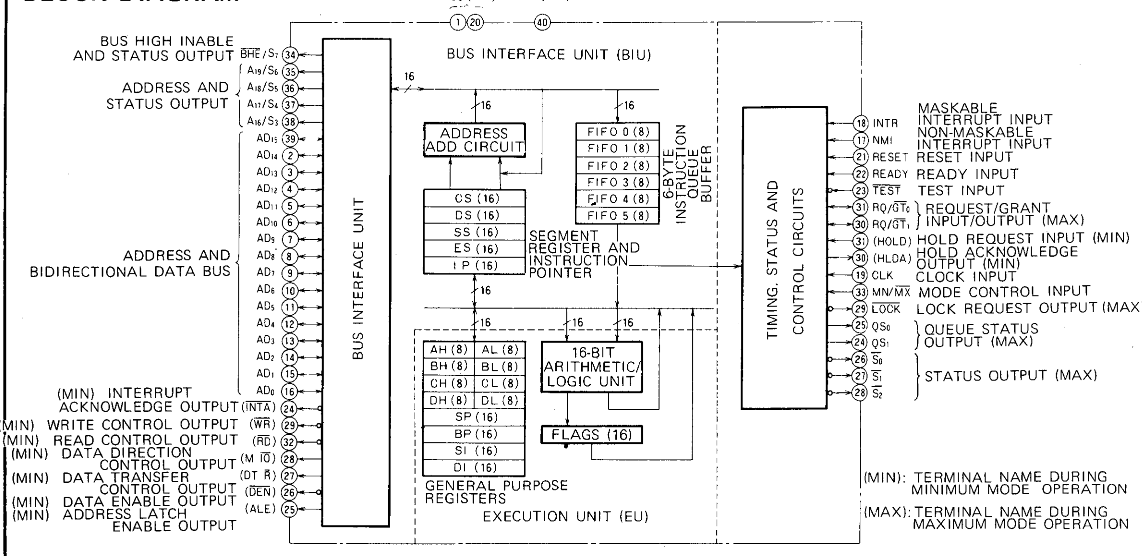
( ) terminal name during minimum mode operation

Outline 40 S 1

FUNCTIONS

The M5L8086S has a minimum and maximum mode, which allows the composition to be selected to match the scale of the system in which it is used. The internal function consists of execution unit (EU) and bus interface unit (BIU). The BIU controls the 6-byte instruction queue, while generating addresses, and decodes instructions to be executed by the EU. Each unit operates asynchronously and can

BLOCK DIAGRAM



access the instruction queue.

The pipeline architecture increased the throughput of the system. The ability to select 8-bit bytes or 16-bit words by using terminals A_0 and $\overline{BHE}$, allows more efficient use of memory. This along with a large direct addressable memory (up to 1 M bytes) makes it practical to process large complicated programs. Two kinds of external interrupt input are provided. The INTR is a maskable interrupt input for the normal interrupt applications, while the NMI is a nonmaskable interrupt for the use of a higher priority interrupt such as power down. In addition to external interrupts, internal interrupts can be initiated by software with the overflow and so on.

PIN DESCRIPTIONS

Pins which have the same functions in minimum or maximum mode

Pin	Name	Input or Output	Functional description
$AD_0 \sim AD_{15}$	Address and data bas	Input/output	$AD_0 \sim AD_{15}$ is used as both an address bus ($A_0 \sim A_{15}$) and a data bus ($D_0 \sim D_{15}$). Though time sharing it outputs addresses during T_1 state and outputs data during T_2, T_3, T_W, T_4 states.
A_{19}/S_6 ∫ A_{16}/S_3	Address and status	Output	The high-order 4 bits ($A_{16} \sim A_{19}$) and status ($S_3 \sim S_6$) are output using time sharing techniques. The address bits are output during T_1 state and data are output during T_2, T_3, T_W, T_4 states. The status bits S_3 and S_4 determine which segment register is used in the bus cycle as follows: S_4 S_3 0 0 extra segment 0 1 stack segment 1 0 code segment or none 1 1 data segment while S_5 shows the interrupt enable flag and starts the beginning of a clock cycle. Status bit S_6 is aways 0. .
$\overline{BHE}/S_7$	Bus high enable and status	Output	Bus high enable ($\overline{BHE}$) and status are output using time sharing techniques. Bus high enable is output during T_1 state and status is output during T_2, T_3, T_W, T_4 states. $\overline{BHE}$ along with A_0 is used to select byte or word unit processing. The selection is as shown below. $\overline{BHE}$ A_0 0 0 word processing (16 bits) 0 1 high-order byte processing (8 bits) 1 0 low-order byte processing (8 bits) 1 1 undefined This pin goes to low-level during the first clock cycle of an interrupt acknowledge cycle. S_7 is a spare status bit.
$\overline{RD}$	Read control	Output	An active "L" signal indicates read timing from memory or an I/O port.
READY	Ready	Input	Signal indicating data transfer to or from memory and I/O device. When the READY signal is at low level the CPU waits for the signal to go high level. When the signal is at high level the CPU ends the read or write.
INTR	Maskable interrupt request	Input	This signal is sampled at the final clock cycle of each instruction for its level. Enable can be masked by software to inhibit interrupts. An interrupt vector of 256 types can be made using an M5L8259A.
$\overline{TEST}$	Test	Input	The CPU samples this pin while in the wait state. As the result of executing a WAIT instruction this pin is at high level. If the pin is still at high level when sampled the CPU continues to idle until it goes to low level and when that happens the CPU will resume operation.
NMI	Non-maskable interrupt request	Input	This signal is sampled during the final clock cycle of an instruction execution cycle. It is used for urgent interrupts such as power down. A type 2 interrupt is generated by this signal.
RESET	Reset	Input	This signal is used to initialized the CPU. When used it must be maintained at high level for 4 clock cycles to be effective.
CLK	Clock	Input	This signal is used for internal clocking. It is normally attached to the clock output of a M5L8284P or similar device.

Pin Description During Minimum Mode

Pin	Name	Input or output	Functional description
$M/\overline{IO}$	Data direction control	Output	This pin indicates whether the CPU is accessing memory or an I/O device at the time.
$\overline{WR}$	Write control	Output	This signal is used for timing when writing data to external memory or I/O device.
$\overline{INTA}$	Interrupt acknowledge	Output	This pin is used as the read strobe for the interrupt vector on the data bus during the interrupt acknowledge cycle.
ALE	Address latch enable	Output	This signal is the output strobe from the CPU for write address. This is output using time sharing techniques to an external latch.
$DT/\overline{R}$	Data transfer control	Output	This signal indicates the direction of data transfer between the data bus buffer and an external device.
$\overline{DEN}$	Data enable	Output	This signal enables the external data bus buffer.
HOLD	Hold request	Input	When a hold request is received by the CPU it will enter the hold state and surrender control of the data bus at the end of the current instruction execution cycle.
HLDA	Hold acknowledge	Output	This signal shows that the CPU bus accepted a hold request from a peripheral device and that control of the data bus has been surrendered to the peripheral device.

Pin Description During Maximum Mode

Pin	Name	Input or Output	Function description
$\overline{S_0}$ $\overline{S_2}$, $\overline{S_1}$,	Status	Output	$\overline{S_2}$ $\overline{S_1}$ $\overline{S_0}$ 0 0 0 Interrupt acknowledge 0 0 1 Read I/O port 0 1 0 Write I/O port 0 1 1 Hold 1 0 0 Instruction fetch 1 0 1 Read memory 1 1 0 Write memory 1 1 1 Passive cycle
$\overline{RQ}/\overline{GT_0}$ $\overline{RQ}/\overline{GT_1}$	Request/Grant	Input/output	This pin is used by other local bus masters to input a hold request to the CPU and then used to output acknowledge. $\overline{RQ}/\overline{GT_0}$ has higher priority than $\overline{RQ}/\overline{GT_1}$.
$\overline{LOCK}$	Lock request	Output	This signal forbids the use of the system bus by any other system bus masters when the CPU is using the system bus.
QS_1, QS_0	Queue status	Output	The status signal is used for indicating queue operations. QS_0 QS_1 0 0 No operation 0 1 fetch first byte (operation code) of the instruction 1 0 clear the contents of queue 1 1 fetch the next byte of the instruction

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	−1.0～7	V
V_I	Input voltage	With respect to GND	−1.0～7	V
Pd	Maximum power dissipation	Ta=25°C	2.5	W
Topr	Operating free-air ambient temperature range		0～70	℃
Tstg	Storage temperature range		−65～150	℃

RECOMMENDED OPERATING CONDITIONS (Ta=0～70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{IH}	High-level input voltage	2.0		V_{CC}+0.5	V
V_{IL}	Low-level input voltage	−0.5		0.8	V
$V_{IH(\phi)}$	High-level clock input voltage	3.9		V_{CC}+1.0	V
$V_{IL(\phi)}$	Low-level clock input voltage	−0.5		0.6	V

ELECTRICAL CHARACTERISTICS (Ta=0～70°C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OL}	Low-level output voltage	I_{OL}=2mA			0.45	V
V_{OH}	High-level output voltage	I_{OH}=−400μA	2.4			V
I_{CC}	Supply current	Ta=25°C			340	mA
I_{LI}	Input leak current	0V<V_I<V_{CC}			±10	μA
I_{LO}	Output leak current	0.45≦V_O≦V_{CC}			±10	μA
Ci	Input capacitance	f_C=1MHz			10	pF
Co	Output capacitance	f_C=1MHz			20	pF

TIMING REQUIREMENTS DURING MINIMUM MODE (Ta=0～70°C, V_{CC}=5V±10%, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{C(\phi)}$	Clock cycle time	TCLCL		200		500	ns
$t_{W(\phi L)}$	Clock input low-level pulse width	TCLCH		$\frac{2}{3}t_{C(\phi)}$-15			ns
$t_{W(\phi H)}$	Clock input high-level pulse width	TCHCL		$\frac{1}{3}t_{C(\phi)}$+2			ns
$t_{r(\phi)}$	Clock input rise time	TCH1CH2	V_{IL}=1.0V, V_{IH}=3.5V			10	ns
$t_{f(\phi)}$	Clock input fall time	TCL2CL1	V_{IL}=1.0V, V_{IH}=3.5V			10	ns
$t_{su(DQ-\phi)}$	Data input setup time before clock	TDVCL		30			ns
$t_{h(\phi-DQ)}$	Data input hold time after clock	TCLDZ		10			ns
$t_{su(RDY-\phi)}$	RDY setup time before clock (Note 1, 2)	TRIVCL		35			ns
$t_{h(\phi-RDY)}$	RDY hold time after clock (Note 1, 2)	TCLRIX		0			ns
$t_{su(READY-\phi)}$	READY setup time before clock	TRYHCH		$\frac{2}{3}t_{C(\phi)}$-15			ns
$t_{h(\phi-READY)}$	READY hold time after clock	TCHRYX		30			ns
$t_{su(READY-\phi)}$	READY data invalid setup time before clock (Note 3)	TRYLCL		−8			ns
$t_{su(HOLD-\phi)}$	HOLD setup time before clock	THVCH		35			ns
$t_{su(INTR-\phi)}$ $t_{su(NMI-\phi)}$ $t_{su(TEST-\phi)}$	INTR, NMI, $\overline{TEST}$ setup time before clock (Note 2)	TINVCH		30			ns

SWITCHING CHARACTERISTICS DURING MINIMUM MODE ($T_a=0\sim70°C$, $V_{CC}=5V\pm10\%$, $C_L=20\sim100pF$, unless otherwise noted)

Symbol	Parameter	Alternative	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{PXV}(\phi\text{-}A)$	Propagation time, clock to address valid	TCLAV		15		110	ns
$t_{PVX}(\phi\text{-}A)$	Address hold time after clock	TCLAX		10			ns
$t_{PVZ}(\phi\text{-}A)$	Propagation time, clock to address float	TCLAZ		$t_{PVX(A\text{-}\phi)}$		80	ns
$t_{W}(ALE)$	Address latch enable pulse width	TLHLL		$t_{W(\phi L)}-20$			ns
$t_{PLH}(\phi\text{-}ALE)$	Propagation time, clock to address latch enable	TCLLH				80	ns
$t_{PHL}(\phi\text{-}ALE)$	Propagation time, clock to address latch enable	TCHLL				85	ns
$t_{PVZ}(ALE\text{-}A)$	Propagation time, address latch enable to address float	TLLAZ		$t_{W(\phi H)}-10$			
$t_{PXV}(\phi\text{-}DQ)$	Propagation time, clock to data valid	TCLDV		15		110	ns
$t_{PVZ}(\phi\text{-}DQ)$	Propagation time, clock to data float	TCHDZ		$t_{h(A\text{-}\phi)}$		85	ns
$t_h(\overline{WR}\text{-}DQ)$	Data hold time after write	TWHDZ		$t_{W(\phi L)}-30$			ns
$t_{PHL}(\phi\text{-}DEN)$ $t_{PHL}(\phi\text{-}\overline{WR})$ $t_{PHL}(\phi\text{-}INTA)$	Propagation time, clock to data enable, clock to write, clock to $\overline{INTA}$	TCVCTV		10		110	ns
$t_{PHL}(\phi\text{-}DT/\overline{R})$ $t_{PLH}(\phi\text{-}DT/\overline{R})$ $t_{PHL}(\phi\text{-}M/\overline{IO})$ $t_{PHL}(\phi\text{-}M/\overline{IO})$	Propagation time, clock to data send and return control signal, clock to data transfer control signal	TCHCTV		15		110	ns
$t_{PLH}(\phi\text{-}\overline{DEN})$ $t_{PLH}(D\text{-}\overline{WR})$	Propagation time, clock to data enable and write	TCVCTX		10		110	ns
$t_{PHL}(A\text{-}\overline{RD})$	Propagation time, address float to read	TAZRL		0			ns
$t_{PHL}(\phi\text{-}\overline{RD})$	Propagation time, clock to read	TCLRL		10		165	ns
$t_{PLH}(\phi\text{-}\overline{RD})$	Propagation time, clock to read	TCLRH		10		150	ns
$t_{PZV}(\overline{RD}\text{-}A)$	Next cycle address propagation time after read	TRHAV		$t_{C(\phi)}-45$			ns
$t_{PLH}(\phi\text{-}HLDA)$	HLDA propagation time after clock.	TCLHAV		10		160	ns

Note 1: Signal at M5L8284P is shown for reference.
2: Setup time required to be recognized at next clock
3: Requirement during T2 state

9

TIMING REQUIREMENTS DURING MAXIMUM MODE ($T_a=0\sim70°C$, $V_{CC}=5V\pm10\%$, unless otherwise noted)

Symbol	Parameter	Alternative	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_C(\phi)$	Clock cycle	TCLCL		200		500	ns
$t_W(\phi L)$	Clock input low-level pulse width	TCLCH		$\frac{2}{3}t_{c(\phi)}-15$			ns
$t_W(\phi H)$	Clock input high-level pulse width	TCHCL		$\frac{1}{3}t_{c(\phi)}+2$			ns
$t_r(\phi)$	Clock input rise time	TCH1CH2	$V_{IL}=1.0V$ $V_{IH}=3.5V$			10	ns
$t_f(\phi)$	Clock input fall time	TCL2CL1	$V_{IL}=1.0V$ $V_{IH}=3.5V$			10	ns
$t_{su}(DQ\text{-}\phi)$	Data input setup time before clock	TDVCL		30			ns
$t_h(\phi\text{-}DA)$	Data input hold time after clock	TCLDZ		10			ns
$t_{su}(READY\text{-}\phi)$	Ready setup time before clock	TRYHCH		35			ns
$t_h(\phi\text{-}READY)$	Ready hold time after clock	TCHRYX		0			ns
$t_{su}(READY\text{-}\phi)$	Ready invalid setup time before clock (Note 6)	TRYLCL		$\frac{2}{3}t_{c(\phi)}-15$			ns
$t_{su}(RDY\text{-}\phi)$	RDY setup time before clock (Note 4, 5)	TR1VCL		35			ns
$t_h(\phi\text{-}RDY)$	RDY hold time after clock (Note 4, 5)	TCLRIX		40			ns
$t_{su}(INTR\text{-}\phi)$ $t_{su}(NMI\text{-}\phi)$ $t_{su}(\overline{TEST}\text{-}\phi)$	INTRO, NMI, $\overline{TEST}$ setup time before clock	TINVCH		30			ns
$t_{su}(\overline{RQ}/\overline{GT}\text{-}\phi)$	$\overline{RQ}/\overline{GT}$ setup time before clock	TGVCH		30			ns
$t_h(\phi\text{-}\overline{RQ})$	$\overline{RQ}$ hold time after clock	TCHGX		30			ns

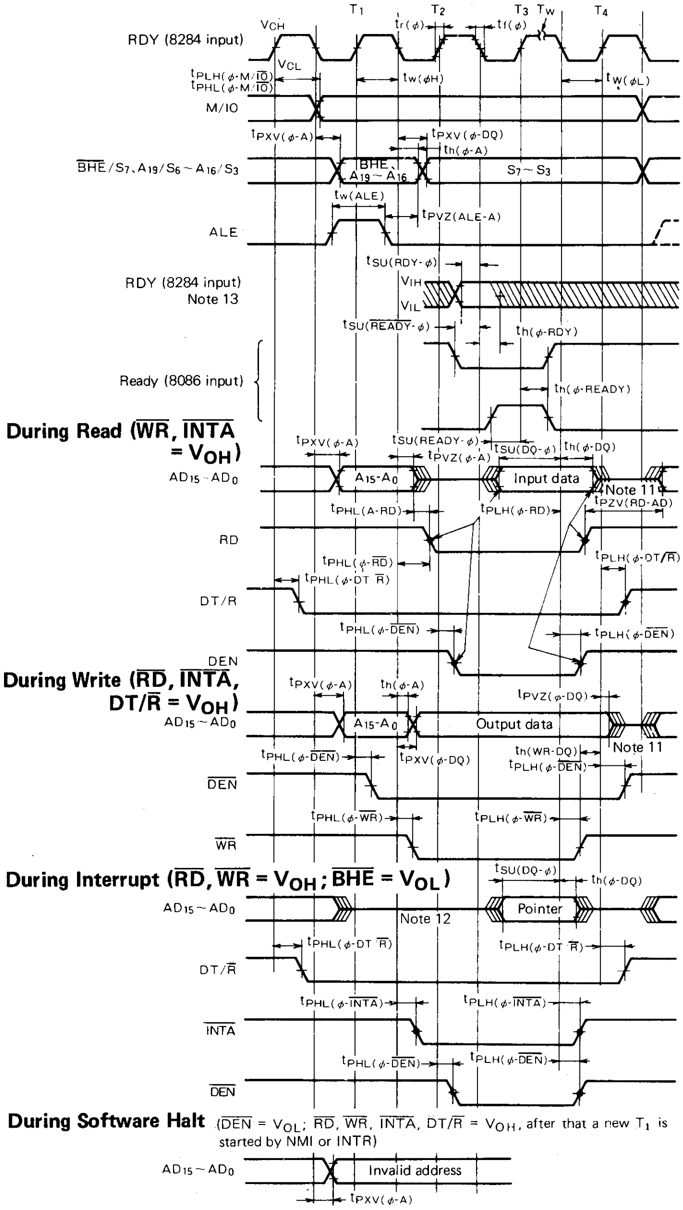
SWITCHING CHARACTERISTICS

MAXIMUM MODE (2) (Ta = 0 ~ 70°C, V_{CC} = 5V ± 10%, C_L = 20 ~ 100pF, unless otherwise noted)

Symbol	Parameter	Alternative	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{PHL(\phi-\overline{MRDC})}$ $t_{PHL(\phi-\overline{IORC})}$ $t_{PHL(\phi-\overline{AIOWC})}$ $t_{PHL(\phi-\overline{AMWC})}$ $t_{PHL(\phi-\overline{INTA})}$ $t_{PHL(\phi-\overline{MWTC})}$ $t_{PHL(\phi-\overline{IOWC})}$	Propagation time, clock to $\overline{MRDC}$, $\overline{IORC}$, $\overline{AIOWC}$, $\overline{AMWC}$, $\overline{INTA}$, $\overline{MWTC}$, $\overline{IOWC}$ (Note 4)	TCLML	C_L = 80pF	10		35	ns
$t_{PLH(\phi-\overline{MRDC})}$ $t_{PLH(\phi-\overline{IORC})}$ $t_{PLH(\phi-\overline{AIOWC})}$ $t_{PLH(\phi-\overline{AMWC})}$ $t_{PLH(\phi-\overline{INTA})}$ $t_{PLH(\phi-\overline{MWTC})}$ $t_{PLH(\phi-\overline{IOWC})}$	Propagation time, clock to $\overline{MRDC}$, $\overline{IORC}$, $\overline{AIOWC}$, $\overline{AMWC}$, $\overline{INTA}$, $\overline{MWTC}$, $\overline{IOWC}$ (Note 4)	TCLMH		10		35	ns
$t_{PVZ(RDY-S)}$	Propagation time, RDY to status 3~7 float (Note 6)	TRYHSH				110	ns
$t_{PHL(\phi-\overline{S})}$		TCHSV		10		110	ns
$t_{PLH(\phi-\overline{S})}$	Propagation time, clock to status 0~2	TCLSH		10		130	ns
$t_{PLH(\phi-QS)}$ $t_{PHL(\phi-QS)}$ $t_{PXV(\phi-A)}$ $t_{PHL(\phi-\overline{LOCK})}$ $t_{PLH(\phi-\overline{LOCK})}$	Propagation time, clock to queue status, address lock	TCLAV		15		110	ns
$t_{PVX(\phi-A)}$	Propagation time, clock to address	TCLAX		10			ns
$t_{PVZ(\phi-A)}$	Propagation time, clock to address float	TCLAZ		$t_{PVX(\phi-A)}$		80	ns
$t_{PLH(\overline{S}-ALE)}$	Propagation time, status 0~2 to address latch enable (Note 4)	TSVLH				15	ns
$t_{PLH(\overline{S}-\overline{MCE})}$ $t_{PLH(\overline{S}-\overline{PDEN})}$	Propagation time, status 0~2 to MCE, $\overline{PDEM}$ (Note 4)	TSVMCH				15	ns
$t_{PLH(\phi-ALE)}$	Propagation time, clock to address latch enable (Note 4)	TCLLH				15	ns
$t_{PLH(\phi-\overline{MCE})}$ $t_{PLH(\phi-\overline{PDEN})}$	Propagation time, clock to MCE, $\overline{PDEN}$ (Note 4)	TCLMCH				15	ns
$t_{PHL(\phi-ALE)}$	Propagation time, clock to address latch enable (Note 4)	TCHLL				15	ns
$t_{PHL(\phi-MCE)}$ $t_{PHL(\phi-PDEN)}$	Propagation time, clock to MCE, $\overline{PDEN}$ (Note 4)	TCLMCL				15	ns
$t_{PXV(\phi-DQ)}$ $t_{PXV(\phi-S)}$	Propagation time, clock to data and status 3~7 valid	TCLDV		15		110	ns
$t_{PVZ(\phi-DQ)}$ $t_{PVZ(\phi-S)}$	Propagation time, clock to data and status 3~7 float	TCHDZ		$t_{PVX(\phi-A)}$		85	ns
$t_{PLH(\phi-\overline{DEN})}$	Propagation time, clock to $\overline{DEN}$ (Note 4)	TCVNV		5		45	ns
$t_{PHL(\phi-\overline{DEN})}$	Propagation time, clock to $\overline{DEN}$ (Note 4)	TCVNX		10		45	ns
$t_{PHL(A-\overline{RD})}$	Propagation time, address float to read	TAZRL		0			ns
$t_{PHL(\phi-\overline{RD})}$	Propagation time, clock to read	TCLRL		10		165	ns
$t_{PLH(\phi-\overline{RD})}$	Propagation time, clock to read	TCLRH		10		150	ns
$t_{PZV(\overline{RD}-A)}$	Propagation time, invalid read to next address	TRHAV		$t_{C(\phi)}$- 45			ns
$t_{PHL(\phi-DT/\overline{R})}$	Propagation time, clock to data S/R control (Note 4)	TCHDTL				50	ns
$t_{PLH(\phi-DT/\overline{R})}$	Propagation time, clock to data S/R control (Note 4)	TCHDTH				30	ns
$t_{PHL(\phi-\overline{GT})}$	Propagation time, clock to data S/R control (Note 4)	TCLGL	C_L = 30pF			85	ns
$t_{PLH(\phi-\overline{GT})}$	Propagation time, clock to data S/R control (Note 4)	TCLGH	C_L = 30pF			85	ns

Note 4: Signal of M5L8284P is shown for reference.
5: Setup time required to be recognized at next clock.
6: Applies only to T_3 and wait states.
7: Applies only to T_2 states.

TIMING DIAGRAM (During Minimum Mode)



Note 8: [symbol] the center line indicates floating (high-impedance) state.

9: Input signal is entered within the range of $V_{OL} \sim V_{OH}$ unless otherwise noted.

10: When the T_W state is entered the RDY signal is sampled near the end of T_2, T_3 and T_W.

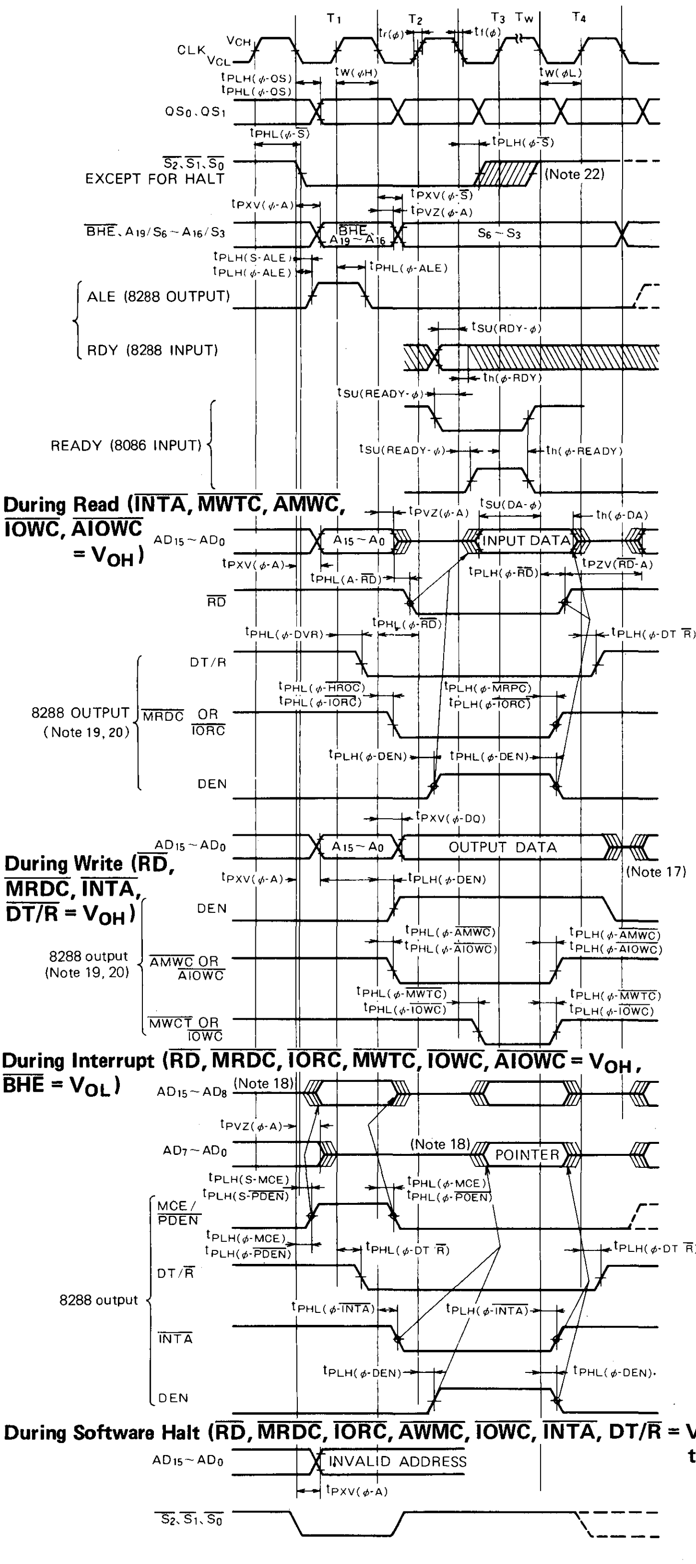
11: Only when the M5L8086S enters a hold acknowledge cycle does the local bus go to a floating state after a write cycle.

12: An interrupt cycle requires 2 clock cycles. The AD bus goes to a floating state during the second cycle of an interrupt.

13: Signals of the M5L8284P are shown for reference.

14: All timing signals are tested at 1.5V unless otherwise noted.

TIMING DIAGRAM (During Maximum Mode)



Note 15: Input signals are entered within the range V_{OL}~V_{OH} unless otherwise noted.

16: When the T_W state is entered the RDY signal is sampled near the end of T_2, T_3 and T_W.

17: Only when the M5L8086S enters a hold acceptance cycle does the local bus go to a floaling state after a write cycle.

18: An interrupt cycle requires 2 cycles. The AD bus goes to a floating state during the second cycle of an interrupt.

19: Signals of the M5L8284P and M5L8288 are shown for reference.

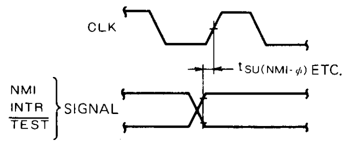
20: The M5L8288 sends a command and a control signal soon after the CEN signal.

21: All timing signals are tested at 1.5V unless otherwise noted.

22: Status is invalid just before T_4 state.

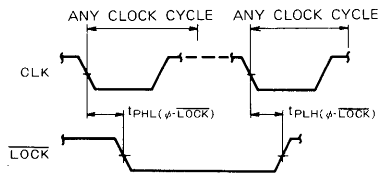
TIMING DIAGRAM

Asynchronous Signal Recognition Timing

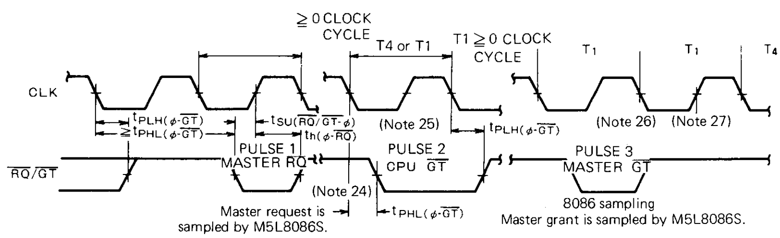


Note 23: Setup time required for being recognized in the next cycle.

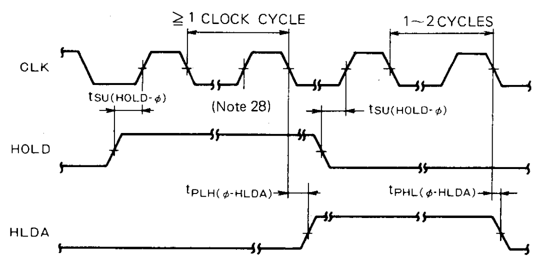
Bus Lock Signal Timing (For Maximum Mode Only)



Request/Grant Sequence Timing (For Maximum Mode Only)



Hold Acknowledge Timing (For Minimum Mode Only)



Note 24: S_2, S_1 and S_0 are changed to floating from the states of (1, 1, 1) at this edge.

25: AD bus, $\overline{RD}$ and $\overline{LOCK}$ are changed to floating at this edge.

26: S_2, S_1 and S_0 of the other master are changed to floating from the states of (1, 1, 1) at this edge.

27: AD bus, $\overline{RD}$ and $\overline{LOCK}$ of the other master are changed to floating at this edge.

28: Bus is changed to floating at this edge.

MACHINE INSTRUCTIONS

Type of instruction		Mnemonic	Instruction code D7 D6 D5 D4 D3 D2 D1 D0	Instruction code D7 D6 D5 D4 D3 D2 D1 D0	Hexadecimal notation
Data transfers	General transfers	[MOV EA1/r1, EA2/r2]	1 0 0 0 1 0 d W	MOD REG R/M	88~8B
			[DISP-L]	[DISP-H]	
		MOV r1, r2	1 0 0 0 1 0 d W	1 1 REG R/M	
		MOV r1, EA2	1 0 0 0 1 0 1 W	MOD REG R/M	
			[DISP-L]	[DISP-H]	
		MOV EA1, r2	1 0 0 0 1 0 0 W	MOD REG R/M	
			[DISP-L]	[DISP-H]	
		[MOV EA1/r1, DATA]	1 1 0 0 0 1 1 W	MOD 0 0 0 R/M	C6~C7
			[DISP-L]	[DISP-H]	
			[DATA-L]	[DATA-H]	
		MOV r1, DATA	1 1 0 0 0 1 1 W	1 1 0 0 0 R/M	
			[DATA-L]	[DATA-H]	
		MOV EA1, DATA	1 1 0 0 0 1 1 W	MOD 0 0 0 R/M	
			[DISP-L]	[DISP-H]	
			[DATA-L]	[DATA-H]	
		MOV r1, DATA	1 0 1 1 W REG	[DATA-L]	B0~BF
			[DATA-H]		
		MOV Acc, ADDR	1 0 1 0 0 0 0 W	[ADDR-L]	A0~A1
			[ADDR-H]		
		MOV ADDR, Acc	1 0 1 0 0 0 1 W	[ADDR-L]	A2~A3
			[ADDR-H]		
		[MOV SEG, EA2/r2]	1 0 0 0 1 1 1 0	MOD 0 SR R/M	8E
			[DISP-L]	[DISP-H]	
		MOV SEG, r2	1 0 0 0 1 1 1 0	1 1 0 SR R/M	
		MOV SEG, EA2	1 0 0 0 1 1 1 0	MOD 0 SR R/M	
			[DISP-L]	[DISP-H]	
		[MOV r1/EA1, SEG]	1 0 0 0 1 1 0 0	MOD 0 SR R/M	8C
			[DISP-L]	[DISP-H]	
		MOV r1, SEG	1 0 0 0 1 1 0 0	1 1 0 SR R/M	
		MOV EA1, SEG	1 0 0 0 1 1 0 0	MOD 0 SR R/M	
			[DISP-L]	[DISP-H]	
		[XCHG r1, EA2/r2]	1 0 0 0 0 1 1 W	MOD REG R/M	86~87
			[DISP-L]	[DISP-H]	
		XCHG r1, r2	1 0 0 0 0 1 1 W	1 1 REG R/M	
		XCHG r1, EA2	1 0 0 0 0 1 1 W	MOD REG R/M	
			[DISP-L]	[DISP-H]	
		XCHG AX, r2	1 0 0 1 0 REG		90~97
		XLAT m	1 1 0 1 0 1 1 1		D7
	Stack operations	[PUSH EA1/r1]	1 1 1 1 1 1 1 1	MOD 1 1 0 R/M	FF
			[DISP-L]	[DISP-H]	
		PUSH r1	1 1 1 1 1 1 1 1	1 1 1 1 0 R/M	
		PUSH EA1	1 1 1 1 1 1 1 1	MOD 1 1 0 R/M	
			[DISP-L]	[DISP-H]	
		PUSH r1	0 1 0 1 0 REG		50~57
		PUSH SEG	0 0 0 SR 1 1 0		06, 0E, 16, 1E
		[POP EA1/r1]	1 0 0 0 1 1 1 1	MOD 0 0 0 R/M	8F
			[DISP-L]	[DISP-H]	
		POP r1	1 0 0 0 1 1 1 1	1 1 1 0 0 R/M	
		POP EA1	1 0 0 0 1 1 1 1	MOD 0 0 0 R/M	
			[DISP-L]	[DISP-H]	
		POP r1	0 1 0 1 1 REG		58~5F
		POP SEG	0 0 0 SR 1 1 1		07, 0F, 17, 1F
	Input/output	IN Acc, Port	1 1 1 0 0 1 0 W	[PORT]	E4~E5
		IN Acc, DX	1 1 1 0 1 1 0 W		EC~ED
		OUT Port, Acc	1 1 1 0 0 1 1 W	[PORT]	E6~E7
		OUT DX, Acc	1 1 1 0 1 1 1 W		EE~EF
	Address transfers	LEA r1, EA2	1 0 0 0 1 1 0 1	MOD REG R/M	8D
			[DISP-L]	[DISP-H]	
		LDS r1, EA2	1 1 0 0 0 1 0 1	MOD REG R/M	C5
			[DISP-L]	[DISP-H]	
		LES r1, EA2	1 1 0 0 0 1 0 0	MOD REG R/M	C4
			[DISP-L]	[DISP-H]	
	Flag transfers	LAHF	1 0 0 1 1 1 1 1		9F
		SAHF	1 0 0 1 1 1 1 0		9E
		POPF	1 0 0 1 1 1 0 1		9D
		PUSHF	1 0 0 1 1 1 0 0		9C

Clock cycles	Bytes in the code	Bus cycles	Function description	OF	DF	IF	TF	SF	ZF	AF	PF	CF
2	2	0	(r1)←(r2)	X	X	X	X	X	X	X	X	X
8+EA	2~4	1	(r1)←(EA2) MOD≠11									
9+EA	2~4	1	(EA1)←(r2) MOD≠11									
4	3~4	0	(r1)←DATA	X	X	X	X	X	X	X	X	X
10+EA	3~6	1	(EA1)←DATA MOD≠11									
4	2~3	0	(r1)←DATA	X	X	X	X	X	X	X	X	X
10	3	1	(A_{CC})←(ADDR)	X	X	X	X	X	X	X	X	X
10	3	1	(ADDR)←(A_{CC})	X	X	X	X	X	X	X	X	X
			When SR = 01: undefined	X	X	X	X	X	X	X	X	X
2	2	0	(SEG)←(r2)									
8+EA	2~4	1	(SEG)←(EA2) MOD≠11									
2	2	0	(r1)←(SEG)	X	X	X	X	X	X	X	X	X
9+EA	2~4	1	(EA1)←(SEG) MOD≠11									
4	2	0	(r1)←→(r2)	X	X	X	X	X	X	X	X	X
17+EA	2~4	1	(r1)←→(EA2) MOD≠11									
3	1	0	(AX)←→(r2)	X	X	X	X	X	X	X	X	X
11	1	1	(AL)←((BX)+(AL)) : (m)	X	X	X	X	X	X	X	X	X
11	2	1	(SP)←(SP)−2, ((SP)+1 : (SP))←(r1)	X	X	X	X	X	X	X	X	X
16+EA	2~4	2	(SP)←(SP)−2, ((SP)+1 : (SP))←(EA1) MOD≠11									
10	1	1	(SP)←(SP)−2, ((SP)+1 : (SP))←(r1)	X	X	X	X	X	X	X	X	X
10	1	1	(SP)←(SP)−2, ((SP)+1 : (SP))←(SEG) When SR = 01: undefined	X	X	X	X	X	X	X	X	X
8	2	1	(r1)←((SP)+1 : (SP)), (SP)←(SP)+2	X	X	X	X	X	X	X	X	X
17+EA	2~4	2	(EA1)←((SP)+1 : (SP)), (SP)←(SP)+2 MOD≠11									
8	1	1	(r1)←((SP)+1 : (SP)), (SP)←(SP)+2	X	X	X	X	X	X	X	X	X
8	1	1	(SEG)←((SP)+1 : (SP)), (SP)←(SP)+2 When SR = 01 : undefined	X	X	X	X	X	X	X	X	X
10	2	1	(A_{CC})←(Port)	X	X	X	X	X	X	X	X	X
8	1	1	(A_{CC})←((DX))	X	X	X	X	X	X	X	X	X
10	2	1	(Port)←(A_{CC})	X	X	X	X	X	X	X	X	X
8	1	1	((DX))←(A_{CC})	X	X	X	X	X	X	X	X	X
2+EA	2~4	0	(r1)←EA2 When MOD = 11: undefined	X	X	X	X	X	X	X	X	X
16+EA	2~4	2	(r1)←(EA2), (DS)←(EA2+2) When MOD = 11: undefined	X	X	X	X	X	X	X	X	X
16+EA	2~4	2	(r1)←(EA2), (ES)←(EA2+2) When MOD = 11: undefined	X	X	X	X	X	X	X	X	X
4	1	0	(AH)←(SF) : (ZF) : X : (AF) : X : (PF) : X : (CF)	X	X	X	X	X	X	X	X	X
4	1	0	(SF) : (ZF) : X : (AF) : X : (PF) : X : (CF)←(AH)	X	X	X	X	○	○	○	○	○
8	1	1	(FR)←((SP)+1 : (SP)), (SP)←(SP)+2	○	○	○	○	○	○	○	○	○
10	1	1	(SP)←(SP)−2, ((SP)+1 : (SP))←(FR)	X	X	X	X	X	X	X	X	X

9

Type of instruction		Mnemonic	D7 D6 D5 D4	D3 D2 D1 D0	D7 D6	D5 D4 D3	D2 D1 D0	Hexadecimal notation
Arithmetic instructions	Addition and related	〔ADD EA1/r1, EA2/r2 〕	0 0 0 0	0 0 d W	MOD	REG	R/M	00~03
			〔DISP-L	〕	〔DISP-H		〕	
		ADD r1,r2	0 0 0 0	0 0 d W	1 1	REG	R/M	
		ADD r1, EA2	0 0 0 0	0 0 1 W	MOD	REG	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		ADD EA1,r2	0 0 0 0	0 0 0 W	MOD	REG	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		〔ADD EA1/r1, DATA 〕	1 0 0 0	0 0 S W	MOD	0 0 0	R/M	80~83
			〔DISP-L	〕	〔DISP-H		〕	
			〔DATA-L	〕	〔DATA-H		〕	
		ADD r1, DATA	1 0 0 0	0 0 S W	1 1	0 0 0	R/M	
			〔DATA-L	〕	〔DATA-H		〕	
		ADD EA1, DATA	1 0 0 0	0 0 S W	MOD	0 0 0	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
			〔DATA-L	〕	〔DATA-H		〕	
		ADD Acc, DATA	0 0 0 0	0 1 0 W	〔DATA-L		〕	04~05
			〔DATA-H	〕				
		〔ADC EA1/r1, EA2/r2 〕	0 0 0 1	0 0 d W	MOD	REG	R/M	10~13
			〔DISP-L	〕	〔DISP-H		〕	
		ADC r1,r2	0 0 0 1	0 0 d W	1 1	REG	R/M	
		ADC r1, EA2	0 0 0 1	0 0 1 W	MOD	REG	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		ADC EA1,r2	0 0 0 1	0 0 0 W	MOD	REG	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		〔ADC EA1/r1, DATA 〕	1 0 0 0	0 0 S W	MOD	0 1 0	R/M	80~83
			〔DISP-L	〕	〔DISP-H		〕	
			〔DATA-L	〕	〔DATA-H		〕	
		ADC r1, DATA	1 0 0 0	0 0 S W	1 1	0 1 0	R/M	
			〔DATA-L	〕	〔DATA-H		〕	
		ADC EA1, DATA	1 0 0 0	0 0 S W	MOD	0 1 0	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
			〔DATA-L	〕	〔DATA-H		〕	
		ADC Acc, DATA	0 0 0 1	0 1 0 W	〔DATA-L		〕	14~15
			〔DATA-H	〕				
		〔INC EA1/r1 〕	1 1 1 1	1 1 1 W	MOD	0 0 0	R/M	FE~FF
			〔DISP-L	〕	〔DISP-H		〕	
		INC r1	1 1 1 1	1 1 1 W	1 1	0 0 0	R/M	
		INC EA1	1 1 1 1	1 1 1 W	MOD	0 0 0	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		INC r1	0 1 0 0	0 REG				40~47
		AAA	0 0 1 1	0 1 1 1				37
		DAA	0 0 1 0	0 1 1 1				27

Clock cycles	Bytes in the code	Bus cycles	Function description	Flags								
				OF	DF	IF	TF	SF	ZF	AF	PF	CF
				○	X	X	X	○	○	○	○	○
3	2	0	(r1)←(r1)+(r2)									
9+EA	2~4	1	(r1)←(r1)+(EA2) MOD≠11									
16+EA	2~4	2	(EA1)←(EA1)+(r2) MOD≠11									
			When S:W = 01 then DATA is DATA-L and DATA-H	○	X	X	X	○	○	○	○	○
			When S:W = 11 then DATA is DATA-L and DATA-H is filled with the sign of DATA-L (sign extended)									
4	3~4	0	(r1)←(r1)+DATA									
17+EA	3~6	2	(EA1)←(EA1)+DATA MOD≠11									
4	2~3	0	(Acc)←(Acc)+DATA	○	X	X	X	○	○	○	○	○
				○	X	X	X	○	○	○	○	○
3	2	0	(r1)←(r1)+(r2) +1									
9+EA	2~4	1	(r1)←(r1)+(EA2) +1 MOD≠11									
16+EA	2~4	2	(EA1)←(EA1)+(r2) +1 MOD≠11									
			When S:W = 01 then DATA is DATA-L and DATA-H	○	X	X	X	○	○	○	○	○
			When S:W = 11 then DATA is DATA-L and DATA-H is filled with the sign of DATA-L (sign extended)									
4	3~4	0	(r1)←(r1)+DATA +1									
17+EA	3~6	2	(EA1)←(EA1)+DATA +1									
4	2~3	0	(Acc)←(Acc)+DATA +1	○	X	X	X	○	○	○	○	
				○	X	X	X	○	○	○	○	
3	2	0	(r1)←(r1)+1									
15+EA	2~4	2	(EA1)←(EA1)+1 MOD≠11									
2	1	0	(r1)←(r1)+1	○	X	X	X	○	○	○	○	X
4	1	0	When (AL) $0F_{16}$ or (CF) = 1: (AL) ← (AL) + (AL) + 60_{16}	△	X	X	X	△	△	○	△	○
			(AH)←(AH)+1, (AF)←1, (CF)←(AF)									
			(AL)←(AL) ∧ $0F_{16}$									
4	1	0	When (AL) $0F_{16}$ or (CF) = 1: (AL) ← (AL) + (AL) + 60_{16}	△	X	X	X	○	○	○	○	○
			(CF)←(AF)V(CF), (AF)←1									
			When (AL) > $9F_{16}$ or (CF) = 1: (AL) ← (AL) + 60_{16}									
			(CF)←1									

9

Type of instruction		Mnemonic	Instruction code D7 D6 D5 D4	D3 D2 D1 D0	D7 D6	D5 D4 D3	D2 D1 D0	Hexadecimal notation
Arithmetic instructions (Cont'd)	Subtraction and related	[SUB EA1/r1, EA2/r2]	0 0 1 0	1 0 d W	MOD	REG	R/M	28~2B
			[DISP-L]		[DISP-H]			
		SUB r1, r2	0 0 1 0	1 0 d W	1 1	REG	R/M	
		SUB r1, EA2	0 0 1 0	1 0 1 W	MOD	REG	R/M	
			[DISP-L]		[DISP-H]			
		SUB EA1, r2	0 0 1 0	1 0 0 W	MOD	REG	R/M	
			[DISP-L]		[DISP-H]			
		[SUB EA1/r1, DATA]	1 0 0 0	0 0 S W	MOD	1 0 1	R/M	80~83
			[DISP-L]		[DISP-H]			
			[DATA-L]		[DATA-H]			
		SUB r1, DATA	1 0 0 0	0 0 S W	1 1	1 0 1	R/M	
			[DATA-L]		[DATA-H]			
		SUB EA1, DATA	1 0 0 0	0 0 S W	MOD	1 0 1	R/M	
			[DISP-L]		[DISP-H]			
			[DATA-L]		[DATA-H]			
		SUB Acc, DATA	0 0 1 0	1 1 0 W	[DATA-L]			2C~2D
			[DATA-H]					
		[SBB EA1/r1, EA2/r2]	0 0 0 1	1 0 d W	MOD	REG	R/M	18~1B
			[DISP-L]		[DISP-H]			
		SBB r1, r2	0 0 0 1	1 0 d W	1 1	REG	R/M	
		SBB r1, EA2	0 0 0 1	1 0 1 W	MOD	REG	R/M	
			[DISP-L]		[DISP-H]			
		SBB EA1, r2	0 0 0 1	1 0 0 W	MOD	REG	R/M	
			[DISP-L]		[DISP-H]			
		[SBB EA1/r1, DATA]	1 0 0 0	0 0 S W	MOD	0 1 1	R/M	80~83
			[DISP-L]		[DISP-H]			
			[DATA-L]		[DATA-H]			
		SBB r1, DATA	1 0 0 0	0 0 S W	1 1	0 1 1	R/M	
			[DATA-L]		[DATA-H]			
		SBB EA1, DATA	1 0 0 0	0 0 S W	MOD	0 1 1	R/M	
			[DISP-L]		[DISP-H]			
			[DATA-L]		[DATA-H]			
		SBB Acc, DATA	0 0 0 1	1 1 0 W	[DATA-L]			1C~1D
			[DATA-H]					
		[DEC EA1/r1]	1 1 1 1	1 1 1 W	MOD	0 0 1	R/M	FE~FF
			[DISP-L]		[DISP-H]			
		DEC r1	1 1 1 1	1 1 1 W	1 1	0 0 1	R/M	
		DEC EA1	1 1 1 1	1 1 1 W	MOD	0 0 1	R/M	
			[DISP-L]		[DISP-H]			
		DEC r1	0 1 0 0	1 REG				48~4F
		[NEG EA1/r1]	1 1 1 1	0 1 1 W	MOD	0 1 1	R/M	F6~F7
			[DISP-L]		[DISP-H]			
		NEG r1	1 1 1 1	0 1 1 W	1 1	0 1 1	R/M	
		NEG EA1	1 1 1 1	0 1 1 W	MOD	0 1 1	R/M	
			[DISP-L]		[DISP-H]			
		[CMP EA1/r1, EA2/r2]	0 0 1 1	1 0 d W	MOD	REG	R/M	38~3B
			[DISP-L]		[DISP-H]			
		CMP r1, r2	0 0 1 1	1 0 d W	1 1	REG	R/M	
		CMP r1, EA2	0 0 1 1	1 0 1 W	MOD	REG	R/M	
			[DISP-L]		[DISP-H]			
		CMP EA1, r2	0 0 1 1	1 0 0 W	MOD	REG	R/M	
			[DISP-L]		[DISP-H]			
		[CMP EA1/r1, DATA]	1 0 0 0	0 0 S W	MOD	1 1 1	R/M	80~83
			[DISP-L]		[DISP-H]			
			[DATA-L]		[DATA-H]			
		CMP r1, DATA	1 0 0 0	0 0 S W	1 1	1 1 1	R/M	
			[DATA-L]		[DATA-H]			
		CMP EA1, DATA	1 0 0 0	0 0 S W	MOD	1 1 1	R/M	
			[DISP-L]		[DISP-H]			
			[DATA-L]		[DATA-H]			
		CMP Acc, DATA	0 0 1 1	1 1 0 W	[DATA-L]			3C~3D
			[DATA-H]					
		AAS	0 0 1 1	1 1 1 1				3F
		DAS	0 0 1 0	1 1 1 1				2F

Clock cycles	Bytes in the code	Bus cycles	Function description	Flags OF	DF	IF	TF	SF	ZF	AF	PF	CF
3	2	0	(r1)←(r1)−(r2)	○	X	X	X	○	○	○	○	○
9+EA	2~4	1	(r1)←(r1)−(EA2) MOD≠11									
16+EA	2~4	2	(EA1)←(EA1)−(r2) MOD≠11									
			When S:W = 01 then DATA is DATA-L and DATA-H When S:W = 11 then DATA is DATA-L and the signs of DATA-L are extended to form 16-bit operand.	○	X	X	X	○	○	○	○	○
4	3~4	0	(r1)←(r1)−DATA									
17+EA	3~6	2	(EA1)←(EA1)−DATA MOD≠11									
4	2~3	0	(Acc) ←(Acc)−DATA	○	X	X	X	○	○	○	○	○
3	2	0	(r1)←(r1)−(r2) -1 when (CF) = 1	○	X	X	X	○	○	○	○	○
9+EA	2~4	1	(r1)←(r1)−(EA2) -1 when (CF) = 1 MOD≠11									
16+EA	2~4	2	(EA1)←(EA1)−(r2) -1 when (CF) = 1 MOD≠11									
			When S:W = 01 then DATA is DATA-L and DATA-H When S:W = 11 then DATA is DATA-L and the sign of DATA-L is extended to form a 16-bit operand.	○	X	X	X	○	○	○	○	○
4	3~4	0	(r1)←(r1)−DATA -1 when (CF) = 1									
17+EA	3~6	2	(EA1)←(EA1)−DATA -1 when (CF) = 1 MOD≠11									
4	2~3	0	(Acc)←(Acc)−DATA when (CF) = 1	○	X	X	X	○	○	○	○	○
3	2	0	(r1)←(r1)−1	○	X	X	X	○	○	○	○	X
15+EA	2~4	2	(EA1)←(EA1)−1 MOD≠11									
2	1	0	(r1)←(r1)−1	○	X	X	X	○	○	○	○	X
			When W=0 (SRC)=FFH When W=1 (SRC)=FFFFH	○	X	X	X	○	○	○	○	1
3	2	0	(r1)←(SRC)−(r1), (r1)←0−(r1)									
16+EA	2~4	4	(EA1)←(SRC)−(EA1), (EA1)←0−(SRC) MOD≠11									
3	2	0	(r1)−(r2)	○	X	X	X	○	○	○	○	○
9+EA	2~4	1	(r1)−(EA2) MOD≠11									
16+EA	2~4	2	(EA1)−(r2) MOD≠11									
			When S:W = 01 then DATA is DATA-L and DATA-H When S:W = 11 then DATA is DATA-L and the signs of DATA-L are extended to form 16-bit operand.	○	X	X	X	○	○	○	○	○
4	3~4	0	(r1)−DATA									
17+EA	3~6	2	(EA1)−DATA MOD≠11									
4	2~3	0	(Acc)−DATA	○	X	X	X	○	○	○	○	○
4	1	0	When ((AL) V $0F_{16}$) or (AF) = 1: When (AL) V $0F_{16}$ or (AF) = 1: (AL) ← (AL) -6 (AH)←(AH)−1, (AF)←1, (CF)←(AF) (AL) ← (AL) V $0F_{16}$	△	X	X	X	△	△	○	△	○
4	1	0	When ((AL) V $0F_{16}$) or (AF) = 1: When (AL) V $0F_{16}$ or (AF) = 1: (AL) ← (AL) -6 (CF)←(AF)V(CF), (AF)←1 When (AL) > $9F_{16}$ or (CF) = 1: (AL) ← (AL) -60_{16} (CF)←1	△	X	X	X	○	○	○	○	○

9

Type of instruction		Mnemonic	D7 D6 D5 D4	D3 D2 D1 D0	D7 D6	D5 D4 D3	D2 D1 D0	Hexadecimal notation
Arithmetic instructions (cont'd)	Multiplication and related	〔MUL EA1/r1 〕	1 1 1 1	0 1 1 W	MOD	1 0 0	R/M	F6～F7
			〔DISP-L	〕	〔DISP-H		〕	
		MUL r1	1 1 1 1	0 1 1 0	1 1	1 0 0	R/M	
			1 1 1 1	0 1 1 1	1 1	1 0 0	R/M	
		MUL EA1	1 1 1 1	0 1 1 0	MOD	1 0 0	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
			1 1 1 1	0 1 1 1	MOD	1 0 0	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		〔IMUL EA1/r1 〕	1 1 1 1	0 1 1 W	MOD	1 0 1	R/M	F6～F7
			〔DISP-L	〕	〔DISP-H		〕	
		IMUL r1	1 1 1 1	0 1 1 0	1 1	1 0 1	R/M	
			1 1 1 1	0 1 1 1	1 1	1 0 1	R/M	
		IMUL EA1	1 1 1 1	0 1 1 0	MOD	1 0 1	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
			1 1 1 1	0 1 1 1	MOD	1 0 1	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		AAM	1 1 0 1	0 1 0 0	0 0	0 0 1	0 1 0	D4
	Division and related	〔DIV EA1/r1 〕	1 1 1 1	0 1 1 W	MOD	1 1 0	R/M	F6～F7
			〔DISP-L	〕	〔DISP-H		〕	
		DIV r1	1 1 1 1	0 1 1 0	1 1	1 1 0	R/M	
			1 1 1 1	0 1 1 1	1 1	1 1 0	R/M	
		DIV EA1	1 1 1 1	0 1 1 0	MOD	1 1 0	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
			1 1 1 1	0 1 1 1	MOD	1 1 0	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		〔IDIV EA1/r1 〕	1 1 1 1	0 1 1 W	MOD	1 1 1	R/M	F6～F7
			〔DISP-L	〕	〔DISP-H		〕	
		IDIV r1	1 1 1 1	0 1 1 0	1 1	1 1 1	R/M	
			1 1 1 1	0 1 1 1	1 1	1 1 1	R/M	
		IDIV EA1	1 1 1 1	0 1 1 0	MOD	1 1 1	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
			1 1 1 1	0 1 1 1	MOD	1 1 1	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		AAD	1 1 0 1	0 1 0 1	0 0	0 0 1	0 1 0	D5
		CBW	1 0 0 1	1 0 0 0				98
		CWD	1 0 0 1	1 0 0 1				99

Clock cycles	Bytes in the code	Bus cycles	Function description	Flags OF	DF	IF	TF	SF	ZF	AF	PF	CF
			(EXT) ← overflow digit of operation; when (EXT) = 0: (CF) ← 0 When (EXT) ≠0: (OF) ← (CF), (CF) ← 1 − (CF)	○	X	X	X	△	△	△	△	○
70 ~ 77	2	0	When W = 0: (EXT) = (AH) (AX) ← (AL) * (r1)									
118 ~ 133	2	0	When W = 1: (EXT) = (DX) (DX : AX) ← (AX) * (r1)									
(76 ~ 83) + EA	2 ~ 4	1	When W = 0: (EXT) = (AH) MOD ≠ 11 (AX) ← (AL) * (EA1)									
(124 ~ 139) + EA	2 ~ 4	1	When W = 1: (EXT) = (DX) MOD ≠ 11 (DX : AX) ← (AX) * (EA1)									
			(EXT) ← overflow digit of operation; when (LOW) changes to (EXT) by extending the sign bit of (LOW): (CF) ← 0 Otherwise: (OF) ← (CF), (CF) ← 1	○	X	X	X	△	△	△	△	○
80 ~ 98	2	0	When W = 0: (EXT) = (AH), (LOW) = (AL) (AX) ← (AL) * (r1)									
128 ~ 154	2	0	When W = 1: (EXT) = (DX), (LOW) = (AX) (DX : AX) ← (AX) * (r1)									
(86 ~ 104) + EA	2 ~ 4	1	When W = 0: (EXT) = (AH), (LOW) = (AL) MOD ≠ 11 (AX) ← (AL) * (EA1)									
(134 ~ 160) + EA	2 ~ 4	1	When W = 1: (EXT) = (DX), (LOW) = (AX) MOD ≠ 11 (DX : AX) ← (AX) * (EA1)									
83	2	0	(AH) ← (AL) ÷ $0A_{16}$, (AL) ← remainder	△	X	X	X	○	○	△	○	△
			(temp) ← dividend; when W = 0: MAX = FF_{16}; When W = 1: MAX = $FFFF_{16}$; (temp) ÷ (EA1/r1) When results of the division are larger than MAX, an interrupt of TYPE = 0 is generated, (SP) ← (SP) − 2, ((SP) + 1: (SP)) ← flag (IF) ← 0, (TF) ← 0, (SP) ← (SP) −2, ((SP) + 1: (SP)) ← (CS) (CS) ← contents of address 2, (SP) ← (SP) −2, ((SP) +1: (SP)) ← IP (IP) ← contents of address 0, the result of the division is undefined.	△	X	X	X	△	△	△	△	△
80 ~ 90	2	0	(AL) ← (AX) ÷ (r1), (AH) ← Remainder									
144 ~ 162	2	0	(AX) ← (DX : AX) ÷ (r1), (DX) ← Remainder									
(86 ~ 96) + EA	2 ~ 4	1	(AL) ← (AX) ÷ (EA1) (AH) ← Remainder MOD ≠ 11									
(150 ~ 168) + EA	2 ~ 4	1	(AX) ← (DX : AX) ÷ (EA1), (DX) ← Remainder MOD ≠ 11									
			(temp) ← dividend; when W = 0: MAX = $7F_{16}$ When W = 1: MAX = $7FFF_{16}$; MIN = 81_{16} when the result of the division is positive and over MAX or when negative and more negative than MIN an interrupt of TYPE = 0 is generated and the result of the division is undefined.	△	X	X	X	△	△	△	△	△
101 ~ 112	2	0	(AL) ← (AX) ÷ (r1), (AH) ← Remainder									
165 ~ 184	2	0	(AX) ← (DX : AX) ÷ (r1), (DX) ← Remainder									
(107 ~ 118) + EA	2 ~ 4	1	(AL) ← (AX) ÷ (EA1), (AH) ← Remainder MOD ≠ 11									
(171 ~ 190) + EA	2 ~ 4	1	(AX) ← (DX : AX) ÷ (EA1), (DX) ← Remainder MOD ≠ 11									
60	2	0	(AL) ← (AH) * $0A_{16}$ + (AL), (AH) ← 0	△	X	X	X	○	○	△	○	△
2	1	0	When (AL) < 80_{16}: (AH) ← 0 When (AL) ≧ 80_{16}: (AH) ← FF_{16} (extended sign bit)	X	X	X	X	X	X	X	X	X
5	1	0	When (AX) < 8000_{16}: (DX) ← 0 When (AX) ≧ 8000_{16}: (DX) ← $FFFF_{16}$ (extended sign bit)	X	X	X	X	X	X	X	X	X

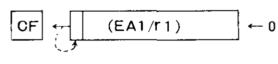
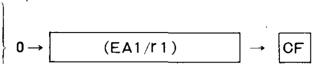
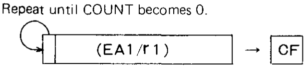
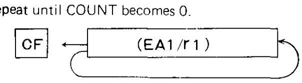
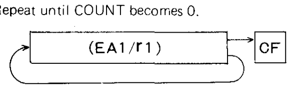
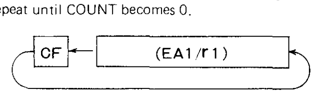
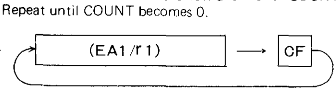
Type of instruction		Mnemonic	Instruction code D7 D6 D5 D4	D3 D2 D1 D0	D7 D6	D5 D4 D3	D2 D1 D0	Hexadecimal notation
Bit manipulation	Logic	〔NOT EA1/r1 〕	1 1 1 1	0 1 1 W	MOD	0 1 0	R/M	F6～F7
			〔DISP-L	〕	〔DISP-H		〕	
		NOT r1	1 1 1 1	0 1 1 W	1 1	0 1 0	R/M	
		NOT EA1	1 1 1 1	0 1 1 W	MOD	0 1 0	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		〔AND EA1/r1, EA2/r2 〕	0 0 1 0	0 0 d W	MOD	REG	R/M	20～23
			〔DISP-L	〕	〔DISP-H		〕	
		AND r1, r2	0 0 1 0	0 0 d W	1 1	REG	R/M	
		AND r1, EA2	0 0 1 0	0 0 1 W	MOD	REG	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		AND EA1, r2	0 0 1 D	0 0 d W	MOD	REG	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		〔AND EA1/r1, DATA 〕	1 0 0 0	0 0 0 W	MOD	1 0 0	R/M	80～81
			〔DISP-L	〕	〔DISP-H		〕	
			〔DATA-L	〕	〔DATA-H		〕	
		AND r1, DATA	1 0 0 0	0 0 0 W	1 1	1 0 0	R/M	
			〔DATA-L	〕	〔DATA-H		〕	
		AND EA1, DATA	1 0 0 0	0 0 0 W	MOD	1 0 0	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
			〔DATA-L	〕	〔DATA-H		〕	
		AND Acc, DATA	0 0 1 0	0 1 0 W	〔DATA-L		〕	24～25
			〔DATA-H	〕				
		〔TEST EA1/r1, EA2/r2 〕	1 0 0 0	0 1 0 W	MOD	REG	R/M	84～85
			〔DISP-L	〕	〔DISP-H		〕	
		TEST r1, r2	1 0 0 0	0 1 0 W	1 1	REG	R/M	
		TEST r1, EA2 }	1 0 0 0	0 1 0 W	MOD	REG	R/M	
		TEST EA1, r2 }	〔DISP-L	〕	〔DISP-H		〕	
		〔TEST EA1/r1, DATA 〕	1 1 1 1	0 1 1 W	MOD	0 0 0	R/M	F6～F7
			〔DISP-L	〕	〔DISP-H		〕	
			〔DATA-L	〕	〔DATA-H		〕	
		TEST r1, DATA	1 1 1 1	0 1 1 W	1 1	0 0 0	R/M	
			〔DATA-L	〕	〔DATA-H		〕	
		TEST EA1, DATA	1 1 1 1	0 1 1 W	MOD	0 0 0	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
			〔DATA-L	〕	〔DATA-H		〕	
		TEST Acc, DATA	1 0 1 0	1 0 0 W	〔DATA-L		〕	A8～A9
			〔DATA-H	〕				
		〔OR EA1/r1, EA2/r2 〕	0 0 0 0	1 0 d W	MOD	REG	R/M	08～0B
			〔DISP-L	〕	〔DISP-H		〕	
		OR r1, r2	0 0 0 0	1 0 d W	1 1	REG	R/M	
		OR r1, EA2	0 0 0 0	1 0 1 W	MOD	REG	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		OR EA1, r2	0 0 0 0	1 0 0 W	MOD	REG	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		〔OR EA1/r1, DATA 〕	1 0 0 0	0 0 0 W	MOD	0 0 1	R/M	80～81
			〔DISP-L	〕	〔DISP-H		〕	
			〔DATA-L	〕	〔DATA-H		〕	
		OR r1, DATA	1 0 0 0	0 0 0 W	1 1	0 0 1	R/M	
			〔DATA-L	〕	〔DATA-H		〕	
		OR EA1, DATA	1 0 0 0	0 0 0 W	MOD	0 0 1	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
			〔DATA-L	〕	〔DATA-H		〕	
		OR Acc, DATA	0 0 0 0	1 1 0 W	〔DATA-L		〕	0C～0D
			〔DATA-H	〕				
		〔XOR EA1/r1, EA2/r2 〕	0 0 1 1	0 0 d W	MOD	REG	R/M	30～33
			〔DISP-L	〕	〔DISP-H		〕	
		XOR r1, r2	0 0 1 1	0 0 d W	1 1	REG	R/M	
		XOR r1, EA2	0 0 1 1	0 0 1 W	MOD	REG	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		XOR EA1, r2	0 0 1 1	0 0 0 W	MOD	REG	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
		〔XOR EA1/r1, DATA 〕	1 0 0 0	0 0 0 W	MOD	1 1 0	R/M	80～81
			〔DISP-L	〕	〔DISP-H		〕	
			〔DATA-L	〕	〔DATA-H		〕	
		XOR r1, DATA	1 0 0 0	0 0 0 W	1 1	1 1 0	R/M	
			〔DATA-L	〕	〔DATA-H		〕	
		XOR EA1, DATA	1 0 0 0	0 0 0 W	MOD	1 1 0	R/M	
			〔DISP-L	〕	〔DISP-H		〕	
			〔DATA-L	〕	〔DATA-H		〕	
		XOR Acc, DATA	0 0 1 1	0 1 0 W	〔DATA-L		〕	34～35
			〔DATA-H	〕				

Clock cycles	Bytes in the code	Bus cycles	Function description	OF	DF	IF	TF	SF	ZF	AF	PF	CF
3 16+EA	2 2~4	0 1	When W = 0: (SRC) = FF_{16} When W = 1: (SRC) = $FFFF_{16}$ (r1)←(SRC)−(r1) (EA1) ←(SRC)−(EA1) MOD≠11	X	X	X	X	X	X	X	X	X
3 9+EA 16+EA	2 2~4 2~4	0 1 2	After execution of the instruction (CF)←0, (OF)←0 (r1)←(r1) ∧ (r2) (r1)←(r1) ∧ (EA2) MOD≠11 (EA1)←(EA1) ∧ (r2) MOD≠11	0	X	X	X	○	○	△	○	0
4 17+EA	3~4 3~6	0 2	After execution of the instruction (CF)←0, (OF)←0 (r1)←(r1) ∧ DATA (EA1)←(EA1) ∧ DATA MOD≠11	0	X	X	X	○	○	△	○	0
4	2~3	0	(Acc)←(Acc) ∧ DATA After execution of the instruction (CF)←0, (OF)←0	0	X	X	X	○	○	△	○	0
3 9+EA	2 2~4	0 1	After execution of the instruction (CF)←0, (OF)←0 (r1)← (r1) ∧ (r2) (r1)← (r1) ∧ (EA2) (EA1)←(EA1) ∧ (r2) MOD≠11	0	X	X	X	○	○	△	○	0
5 11+EA	3~4 3~6	0 2	After execution of the instruction (CF)←0, (OF)←0 (r1)←(r1) ∧ DATA (EA1)←(EA1) ∧ DATA MOD≠11	0	X	X	X	○	○	△	○	0
4	2~3	0	(Acc) ∧ DATA, (CF)←0, (OF)←0	0	X	X	X	○	○	△	○	0
3 9+EA 16+EA	2 2~4 2~4	0 1 2	After execution of the instruction (CF)←0, (OF)←0 (r1)←(r1) ∨ (r2) (r1)←(r1) ∨ (EA2) MOD≠11 (EA1)←(EA1) ∨ (r2) MOD≠11	0	X	X	X	○	○	△	○	0
4 17+EA	3~4 3~6	0 2	After execution of the instruction (CF)←0, (OF)←0 (r1)←(r1) ∨ DATA (EA1)←(EA1) ∨ DATA MOD≠11	0	X	X	X	○	○	△	○	0
4	2~3	0	(Acc)←(Acc) ∨ DATA, (CF)←0, (OF)←0	0	X	X	X	○	○	△	○	0
3 9+EA 16+EA	2 2~4 2~4	0 1 2	After execution of the instruction (CF)←0, (OF)←0 (r1)←(r1) ⊻ (r2) (r1)←(r1) ⊻ (EA2) MOD≠11 (EA1)←(EA1) ⊻ (r2) MOD≠11	0	X	X	X	○	○	△	○	0
4 17+EA	3~4 3~6	0 2	After execution of the instruction (CF)←0, (OF)←0 (r1)←(r1) ⊻ DATA (EA1)←(EA1) ⊻ DATA MOD≠11	0	X	X	X	○	○	△	○	0
4	2~3	0	(Acc)←(Acc) ⊻ DATA, (CF)←0, (OF)←0	0	X	X	X	○	○	△	○	0

9

MACHINE INSTRUCTIONS

Type of instruction		Mnemonic	Instruction code D7 D6 D5 D4	D3 D2 D1 D0	D7 D6	D5 D4 D3	D2 D1 D0	Hexadecimal notation
Bit manipulation (cont'd)	Shifts	[SHL/SAL EA1/r1, 1/CL]	1 1 0 1	0 0 V W	MOD	1 0 0	R/M	D0~D3
			[DISP-L	]	[DISP-H		]	
		SHL/SAL r1, 1	1 1 0 1	0 0 0 W	1 1	1 0 0	R/M	
		SHL/SAL r1, CL	1 1 0 1	0 0 1 W	1 1	1 0 0	R/M	
		SHL/SAL EA1, 1	1 1 0 1	0 0 0 W	MOD	1 0 0	R/M	
			[DISP-L	]	[DISP-H		]	
		SHL/SAL EA1, CL	1 1 0 1	0 0 1 W	MOD	1 0 0	R/M	
			[DISP-L	]	[DISP-H		]	
		[SHR EA1/r1, 1/CL]	1 1 0 1	0 0 V W	MOD	1 0 0	R/M	D0~D3
			[DISP-L	]	[DISP-H		]	
		SHR r1, 1	1 1 0 1	0 0 0 W	1 1	1 0 1	R/M	
		SHR r1, CL	1 1 0 1	0 0 1 W	1 1	1 0 1	R/M	
		SHR EA1, 1	1 1 0 1	0 0 0 W	MOD	1 0 1	R/M	
			[DISP-L	]	[DISP-H		]	
		SHR EA1, CL	1 1 0 1	0 0 1 W	MOD	1 0 1	R/M	
			[DISP-L	]	[DISP-H		]	
		[SAR EA1/r1, 1/CL]	1 1 0 1	0 0 V W	MOD	1 1 1	R/M	D0~D3
			[DISP-L	]	[DISP-H		]	
		SAR r1, 1	1 1 0 1	0 0 0 W	1 1	1 1 1	R/M	
		SAR r1, CL	1 1 0 1	0 0 1 W	1 1	1 1 1	R/M	
		SAR EA1, 1	1 1 0 1	0 0 0 W	MOD	1 1 1	R/M	
			[DISP-L	]	[DISP-H		]	
		SAR EA1, CL	1 1 0 1	0 0 1 W	MOD	1 1 1	R/M	
			[DISP-L	]	[DISP-H		]	
	Rotates	[ROL EA1/r1, 1/CL]	1 1 0 1	0 0 V W	MOD	0 0 0	R/M	D0~D3
			[DISP-L	]	[DISP-H		]	
		ROL r1, 1	1 1 0 1	0 0 0 W	1 1	0 0 0	R/M	
		ROL r1, CL	1 1 0 1	0 0 1 W	1 1	0 0 0	R/M	
		ROL EA1, 1	1 1 0 1	0 0 0 W	MOD	0 0 0	R/M	
			[DISP-L	]	[DISP-H		]	
		ROL EA1, CL	1 1 0 1	0 0 1 W	MOD	0 0 0	R/M	
			[DISP-L	]	[DISP-H		]	
		[ROR EA1/r1, 1/CL]	1 1 0 1	0 0 V W	MOD	0 0 1	R/M	D0~D3
			[DISP-L	]	[DISP-H		]	
		ROR r1, 1	1 1 0 1	0 0 0 W	1 1	0 0 1	R/M	
		ROR r1, CL	1 1 0 1	0 0 1 W	1 1	0 0 1	R/M	
		ROR EA1, 1	1 1 0 1	0 0 0 W	MOD	0 0 1	R/M	
			[DISP-L	]	[DISP-H		]	
		ROR EA1, CL	1 1 0 1	0 0 1 W	MOD	0 0 1	R/M	
			[DISP-L	]	[DISP-H		]	
		[RCL EA1/r1, 1/CL]	1 1 0 1	0 0 V W	MOD	0 1 0	R/M	D0~D3
			[DISP-L	]	[DISP-H		]	
		RCL r1, 1	1 1 0 1	0 0 0 W	1 1	0 1 0	R/M	
		RCL r1, CL	1 1 0 1	0 0 1 W	1 1	0 1 0	R/M	
		RCL EA1, 1	1 1 0 1	0 0 0 W	MOD	0 1 0	R/M	
			[DISP-L	]	[DISP-H		]	
		RCL EA1, CL	1 1 0 1	0 0 1 W	MOD	0 1 0	R/M	
			[DISP-L	]	[DISP-H		]	
		[RCR EA1/r1, 1/CL]	1 1 0 1	0 0 V W	MOD	0 1 1	R/M	D0~D3
			[DISP-L	]	[DISP-H		]	
		RCR r1, 1	1 1 0 1	0 0 0 W	1 1	0 1 1	R/M	
		RCR r1, CL	1 1 0 1	0 0 1 W	1 1	0 1 1	R/M	
		RCR EA1	1 1 0 1	0 0 0 W	MOD	0 1 1	R/M	
			[DISP-L	]	[DISP-H		]	
		RCR EA1, CL	1 1 0 1	0 0 1 W	MOD	0 1 1	R/M	
			[DISP-L	]	[DISP-H		]	

Clock cycles	Bytes in the code	Bus cycles	Function description	Flags OF	DF	IF	TF	SF	ZF	AF	PF	CF
2 8+4/bit 15+EA 20+EA+4/bit	2 2 2~4 2~4	0 0 2 2	When V = 0: COUNT ← 1 if the high-order bit of (EA1/r1) = (CF) : (OF) ← 0 if the high-order bit of (EA1/r1) ≠ (CF) : (OF) ← 1 When V = 1: COUNT ← (CL), (OF) is undefined Shift one bit as indicated below and reduce COUNT by 1, Repeat until COUNT becomes 0.  	○	X	X	X	X	X	X	X	○
2 8+4/bit 15+EA 20+EA+4/bit	2 2 2~4 2~4	0 0 2 2	When V = 0: COUNT ← 1 if the high-order bits of (EA1/r1) are equal : (OF) ← 0 if the high-order bits of (EA1/r1) are not equal: (OF) ← 1 When V = 1: COUNT ← (CL), (OF) is undefined Shift one bit as indicated below and reduce COUNT by 1, Repeat until COUNT becomes 0.  	○	X	X	X	X	X	X	X	○
2 8+4/bit 15+EA 20+EA+4/bit	2 2 2~4 2~4	0 0 2 2	When V = 0: COUNT ← 1, (OF) ← 0 When V = 1: COUNT ← (CL), (OF) is undefined Shift one bit as indicated below and reduce COUNT by 1, Repeat until COUNT becomes 0.  	○	X	X	X	X	X	X	X	○
2 8+4/bit 15+EA 20+EA+4/bit	2 2 2~4 2~4	0 0 2 2	When V = 0: COUNT ← 1 if the high-order bit of (EA1/r1) = (CF) : (OF) ← 0 if the high-order bit of (EA1/r1) ≠ (CF) : (OF) ← 1 When V = 1: COUNT ← (CL), (OF) is undefined Rotate one bit as indicated below and reduce COUNT by 1. Repeat until COUNT becomes 0.  	○	X	X	X	X	X	X	X	○
2 8+4/bit 15+EA 20+EA+4/bit	2 2 2~4 2~4	0 0 2 2	When V = 0: COUNT ← 1 if the high-order bits of (EA1/r1) are equal: (OF) ← 0 if the high-order bits of (EA1/r1) are not equal: (OF) ← 1 When V = 1: COUNT ← (CL), (OF) is undefined Rotate one bit as indicated below and reduce COUNT by 1. Repeat until COUNT becomes 0.  	○	X	X	X	X	X	X	X	○
2 8+4/bit 15+EA 20+EA+4/bit	2 2 2~4 2~4	0 0 2 2	When V = 0: COUNT ← 1 if the high-order bits of (EA1/r1) are equal: (OF) ← 0 if the high-order bits of (EA1/r1) are not equal: (OF) ← 1 When V = 1: COUNT ← (CL), (OF) is undefined Rotate one bit as indicated below and reduce COUNT by 1. Repeat until COUNT becomes 0.  	○	X	X	X	X	X	X	X	○
2 8+4/bit 15+EA 20+EA+4/bit	2 2 2~4 2~4	0 0 2 2	When V = 0: COUNT ← 1 if the high-order bits of (EA1/r1) are equal: (OF) ← 0 if the high-order bits of (EA1/r1) are not equal: (OF) ← 1 When V = 1: COUNT ← (CL), (OF) is undefined Rotate one bit as indicated below and reduce COUNT by 1. Repeat until COUNT becomes 0.  	○	X	X	X	X	X	X	X	○

9

Type of instruction		Mnemonic	Instruction code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Hexadecimal notation
String manipulations	Repeat prefix	[REP] REPE/REPZ REPNE/REPNZ	1 1 1 1 0 0 1 Z 1 1 1 1 0 0 1 1 1 1 1 1 0 0 1 0		F2 ~ F3
	Transmission	MOVS MEM1, MEM2	1 0 1 0 0 1 0 W		A4 ~ A5
	Comparison	CMPS MEM1, MEM2	1 0 1 0 0 1 1 W		A6 ~ A7
	Scan	SCAS MEM	1 0 1 0 1 1 1 W		AE ~ AF
	Load	LODS MEM	1 0 1 0 1 1 0 W		AC ~ AD
	Store	STOS MEM	1 0 1 0 1 0 1 W		AA ~ AB

Clock cycles (Note 29)	Bytes in the code	Bus cycles (Note 29)	Function description	OF	DF	IF	TF	SF	ZF	AF	PF	CF
2 2	1 1	0 0	Register CX becomes a counter. The prefixed instruction is executed and CX is counted down by 1. The execution of the prefixed instruction and counting down of CX is repeated until CX becomes 0. The instructions SCAS and CMPS, which may alter some flags will not repeat when the value of Z ≠ (ZF)	X	X	X	X	X	X	X	X	X
18 9+17/LOOP	1	2 2/LOOP	(DI) = MEM1, (ST) = MEM2 When W = 0: ((DI)) ← ((SI)) if (DF) = 0: (DI) ← (DI)+2, (SI) ← (SI)+2 if (DF) = 1: (DI) ← (DI)−2, (SI) ← (SI)−2 When W = 1: ((DI+1:DI)) ← ((SI+1:SI)) if (DF) = 0: (DI) ← (DI)+2, (SI) ← (SI)+2 if (DF) = 1: (DI) ← (DI)−2, (SI) ← (SI)−2	X	X	X	X	X	X	X	X	X
22 9+22/LOOP	1	2 2/LOOP	(SI) = MEM1, (DI) = MEM2 When W = 0: ((SI)) ← ((DI)) if (DF) = 0: (DI) ← (DI)+1, (SI) ← (SI)+1 if (DF) = 1: (DI) ← (DI)−1, (SI) ← (SI)−1 When W = 1: ((SI+1:SI)) ← ((DI+1:DI)) if (DF) = 0: (DI) ← (DI)+2, (SI) ← (SI)+2 if (DF) = 1: (DI) ← (DI)−2, (SI) ← (SI)−2	○	X	X	X	○	○	○	○	○
15 9+15/LOOP	1	1 1/LOOP	(DI) = MEM When W = 0: (AL) ← ((DI)) if (DF) = 0: (DI) ← (DI)+1 if (DF) = 1: (DI) ← (DI)−1 When W = 1: (AX) ← ((DI+1:DI)) if (DF) = 0: (DI) ← (DI)+2 if (DF) = 1: (DI) ← (DI)−2	○	X	X	X	○	○	○	○	○
12 9+13/LOOP	1	1 1/LOOP	(SI) = MEM When W = 0: (AL) ← ((SI)) if (DF) = 0: (SI) ← (SI)+1 if (DF) = 1: (SI) ← (SI)−1 When W = 1: (AX) ← ((SI+I:SI)) if (DF) = 0: (SI) ← (SI)+2 if (DF) = 1: (SI) ← (SI)−2	X	X	X	X	X	X	X	X	X
11 9+10/LOOP	1	1 1/LOOP	(DI) = MEM When W = 0: ((DI)) ← (AL) if (DF) = 0: (DI) ← (DI)+1 if (DF) = 1: (DI) ← (DI)−1 When W = 1: ((DI+1:DI)) ← (AX) if (DF) = 0: (DI) ← (DI)+2 if (DF) = 1: (DI) ← (DI)−2	X	X	X	X	X	X	X	X	X

Note 29: The number of clock and bus cycles depend on the number of time an instruction is repeated.
The numbers shown are the number per loop and to determine the number of cycles the listed figures must be multiplied by the number of times the instruction is repeated.

Type of instruction		Mnemonic	Instruction code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Hexadecimal notation
Control transfer	Unconditional jump	JMP DISP16	1 1 1 0 1 0 0 1 (DISP-H)	(DISP-L)	E9
		JMP DISP8	1 1 1 0 1 0 1 1	(DISP-L)	EB
		JMP EA1/r1	1 1 1 1 1 1 1 1 (DISP-L)	MOD 1 0 0 R/M (DISP-H)	FF
		JMP FAR-LABEL	1 1 1 0 1 0 1 0 (Offset-H) (Seg-H)	(Offset-L) (Seg-L)	EA
		JMP EA1/r1	1 1 1 1 1 1 1 1 (DISP-L)	MOD 1 0 1 R/M (DISP-H)	FF
	Call	CALL NEAR-PROC	1 1 1 0 1 0 0 0 (DISP-H)	(DISP-L)	E8
		CALL EA1/r1	1 1 1 1 1 1 1 1 (DISP-L)	MOD 0 1 0 R/M (DISP-H)	FF
		CALL FAR-PROC	1 0 0 1 1 0 1 0 (Offset-H) (Seg-H)	(Offset-L) (Seg-L)	9A
		CALL EA1/r1	1 1 1 1 1 1 1 1 (DISP-L)	MOD 0 1 1 R/M (DISP-H)	FF
	Return	RET	1 1 0 0 0 0 1 1		C3
		RET DATA	1 1 0 0 0 0 1 0 (DATA-H)	(DATA-L)	C2
		RET	1 1 0 0 1 0 1 1		CB
		RET DATA	1 1 0 0 1 0 1 0 (DATA-H)	(DATA-L)	CA

Clock cycles	Bytes in the code	Bus cycles	Function description	Flags								
				OF	DF	IF	TF	SF	ZF	AF	PF	CF
15	3	0	Jump within current segment (DEST)←(IP)+DISP	X	X	X	X	X	X	X	X	X
15	2	0	Jump within current segment	X	X	X	X	X	X	X	X	X
18+EA	2~4	1 0	When MOD = 11: (IP) ← (r1) When MOD ≠ 11: (IP) ← (EA1)	X	X	X	X	X	X	X	X	X
15	5	0	Jump to other segment (IP) ← Offset, (CS) ← Seg	X	X	X	X	X	X	X	X	X
24+EA	2~4	2	Jump to other segment (IP)←(EA1/r1) (CS)←(EA1+1/r1+1)	X	X	X	X	X	X	X	X	X
11	3	1	Call within current segment (SP) ← (SP) −2 ((SP) +1 : (SP)) ← (IP) (IP) ← (IP) +DISP	X	X	X	X	X	X	X	X	X
13+EA	2~4	2	Call within current segment (SP) ← (SP) −2 ((SP) +1 : (SP)) ← (IP) (IP) ← (EA1)/(r1)	X	X	X	X	X	X	X	X	X
20	5	2	Call to other segment (SP) ← (SP) −2 ((SP)+1 : (SP)) ← (CS) (CS) ← Segment (SP) ← (SP) −2 ((SP)+1 : (SP)) ← (IP) (IP) ← Offset	X	X	X	X	X	X	X	X	X
29+EA	2~4	4	Call to other segment (SP)←(SP)−2 ((SP)+1 : (SP))←(CS) (CS)←(EA1+2) (SP)←(SP)−2 ((SP)+1 : (SP)) ←(IP) (IP)←(EA1)	X	X	X	X	X	X	X	X	X
8	1	1	Return within current segment (IP)←((SP)+1 : (SP)), (SP)←(SP)+2	X	X	X	X	X	X	X	X	X
12	3	1	Return within current segment (IP)←((SP)+1 : (SP)), (SP)←(SP)+DATA	X	X	X	X	X	X	X	X	X
18	1	2	Return to other segment (IP)←((SP)+1 : (SP)), (SP)←(SP)+2 (CS)←((SP)+1 : (SP)), (SP)←(SP)+2	X	X	X	X	X	X	X	X	X
17	3	2	Return to other segment (IP)←((SP)+1 : (SP)), (SP)←(SP)+2 (CS)←((SP)+1 : (SP)), (SP)←(SP)+DATA	X	X	X	X	X	X	X	X	X

Type of instruction		Mnemonic	Instruction code D7 D6 D5 D4	D3 D2 D1 D0	D7 D6 D5 D4 D3 D2 D1 D0	Hexadecimal notation
Control transfer (cont'd)	Condition jump	JE/JZ LABEL	0 1 1 1	0 1 0 0	[DISP]	74
		JL/JNGE LABEL	0 1 1 1	1 1 0 0	[DISP]	7C
		JLE/JNG LABEL	0 1 1 1	1 1 1 0	[DISP]	7E
		JB/JNAE LABEL	0 1 1 1	0 0 1 0	[DISP]	72
		JBE/JNA LABEL	0 1 1 1	0 1 1 0	[DISP]	76
		JP/JPE LABEL	0 1 1 1	1 0 1 0	[DISP]	7A
		JO LABEL	0 1 1 1	0 0 0 0	[DISP]	70
		JS LABEL	0 1 1 1	1 0 0 0	[DISP]	78
		JNE/JNZ LABEL	0 1 1 1	0 1 0 1	[DISP]	75
		JNL/JGE LABEL	0 1 1 1	1 1 0 1	[DISP]	7D
		JNLE/JG LABEL	0 1 1 1	1 1 1 1	[DISP]	7F
		JNB/JAE LABEL	0 1 1 1	0 0 1 1	[DISP]	73
		JNBE/JA LABEL	0 1 1 1	0 1 1 1	[DISP]	77
		JNP/JPO LABEL	0 1 1 1	1 0 1 1	[DISP]	7B
		JNO LABEL	0 1 1 1	0 0 0 1	[DISP]	71
		JNS LABEL	0 1 1 1	1 0 0 1	[DISP]	79
	Internal control	LOOP LABEL	1 1 1 0	0 0 1 0	[DISP]	E2
		LOOPZ/LOOPE LABEL	1 1 1 0	0 0 0 1	[DISP]	E1
		LOOPNZ/LOOPNE LABEL	1 1 1 0	0 0 0 0	[DISP]	E0
		JCXZ LABEL	1 1 1 0	0 0 1 1	[DISP]	E3

Clock cycles	Bytes in the code	Bus cycles	Function description	Flags O F	D F	I F	T F	S F	Z F	A F	P F	C F
16 4	2 2	0 0	When (ZF) = 1: (IP) ← (IP) + DISP (extends sign bit) When (ZF) = 0: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (SF) ∀ (OF) = 1: (IP) ← (IP) + DISP (extends sign bit) When (SF) ∀ (OF) = 0: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (SF) ∀ (OF) V (ZF) = 1: (IP) ← (IP) + DISP (extends sign bit) When (SF) ∀ (OF) V (ZF) = 0: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (CF) = 1: (IP) ← (IP) + DISP (extends sign bit) When (CF) = 0: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 6	2 2	0 0	When (CF) V (ZF) = 1: (IP) ← (IP) + DISP (extends sign bit) When (CF) V (ZF) = 0: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (PF) = 1: (IP) ← (IP) + DISP (extends sign bit) When (PF) = 0: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (OF) = 1: (IP) ← (IP) + DISP (extends sign bit) When (OF) = 0: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (SF) = 1: (IP) ← (IP) + DISP (extends sign bit) When (SF) = 0: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (ZF) = 0: (IP) ← (IP) + DISP (extends sign bit) When (ZF) = 1: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (SF) ∀ (OF) = 0: (IP) ← (IP) + DISP (extends sign bit) When (SF) ∀ (OF) = 1: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (SF) ∀ (OF) V (ZF) = 0: (IP) ← (IP) + DISP (extends sign bit) When (SF) ∀ (OF) V (ZF) = 1: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (CF) = 0: (IP) ← (IP) + DISP (extends sign bit) When (CF) = 1: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (CF) V (ZF) = 0: (IP) ← (IP) + DISP (extends sign bit) When (CF) V (ZF) = 1: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (PF) = 0: (IP) ← (IP) + DISP (extends sign bit) When (PF) = 1: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (OF) = 0: (IP) ← (IP) + DISP (extends sign bit) When (OF) = 1: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
16 4	2 2	0 0	When (SF) = 0: (IP) ← (IP) + DISP (extends sign bit) When (SF) = 1: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
17 5	2 2	0 0	(CX) ← (CX) −1 When (CX) = 0: (IP) ← (IP) + DISP (extends sign bit) When (CX) = 1: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
18 6	2 2	0 0	(CX) ← (CX) −1 When (ZF) = 1 and (CX) ≠ 0: (IP) ← (IP) + DISP (extends sign bit) When (ZF) = 0 or (CX) = 0: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
19 5	2 2	0 0	(CX) ← (CX) −1 When (ZF) = 0 and (CX) ≠ 0: (IP) ← (IP) + DISP (extends sign bit) When (ZF) = 1 or (CX) = 0: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X
18 6	2 2	0 0	When (CX) = 0: (IP) ← (IP) + DISP (extends sign bit) When (CX) = 1: (IP) ← (IP) + 2 (executes the next inst.)	X	X	X	X	X	X	X	X	X

9

Type of instruction		Mnemonic	Instruction code D7 D6 D5 D4	D3 D2 D1 D0	D7 D6	D5 D4 D3	D2 D1 D0	Hexadecimal notation
Control transfer (cont'd)	Interrupt	[INT type]	1 1 0 0	1 1 0 V	[When V = 1: Specify type]			CC~CD
		INT type 3	1 1 0 0	1 1 0 0				
		INT (any)	1 1 0 0	1 1 0 1	[type]			
		INTO	1 1 0 0	1 1 1 0				CE
		IRET	1 1 0 0	1 1 1 1				CF
Processor control	Flag	CLC	1 1 1 1	1 0 0 0				F8
		CMC	1 1 1 1	0 1 0 1				F5
		STC	1 1 1 1	1 0 0 1				F9
		CLD	1 1 1 1	1 1 0 0				FC
		STD	1 1 1 1	1 1 0 1				FD
		CLI	1 1 1 1	1 0 1 0				FA
		STI	1 1 1 1	1 0 1 1				FB
	Miscellaneous	HLT	1 1 1 1	0 1 0 0				F4
		WAIT	1 0 0 1	1 0 1 1				9B
		ESC	1 1 0 1	1 X X X	MOD	X X X	R/M	D8~DF
			[DISP-L	]	[DISP-H		]	
		LOCK	1 1 1 1	0 0 0 0				F0
		NOP	1 0 0 1	0 0 0 0				90

Note 30: The preceding tables are summaries of details of the instructions of the M5L8086S. Basic instructions with variation are shown in brackets "[ ]" in the mnemonic column followed by the variations.
Instructions are from 1 to 6 bytes in length. Details of the first byte are given in the left half of the instruction code column, the second byte in the right half, the third byte below the first, the fourth byte below the second, the fifth byte below the third and the sixth byte below the fourth.
The hexadecimal column shows the value of the first byte of an instruction. When in has a single value the single value is shown. When it has a range of values, the range is shown.

Clock cycles	Bytes in the code	Bus cycles	Function description	Flags								
				OF	DF	IF	TF	SF	ZF	AF	PF	CF
52 51	1 2	5 5	When V = 0: Type = 3 When V = 1: Type = type (0 ~ 255) (SP)←(SP)−2, ((SP)+1 : (SP))← Flag (IF)←0, (TF)←0, (SP)←(SP)−2 ((SP)+1 : (SP))←(CS), (CS)←(type *4+2) (SP)←(SP)−2, ((SP)+1 : (SP))←(IP) (IP)←(type *4)	X	X	0	0	X	X	X	X	X
4 53	1 1	0 5	When (OF) = 0: No operation When (OF) = 1: (SP)←(SP)−2, ((SP)+1 : (SP))← Flag (IF)←0, (TF)←0 (SP)←(SP)−2, ((SP)+1 : (SP))←(CS) (CS)← 12_{16} (SP)←(SP)−2, ((SP)+1 : (IP)← 10_{16}	X	X	0	0	X	X	X	X	X
24	1	3	Return from interrupt routine (IP)←((SP)+1 : (SP)), (SP)←(SP)+2 (CS)←((SP)+1 : (SP)), (SP)←(SP)+2 (Flag)←((SP)+1 : (SP)), (SP)←(SP)+2	○	○	○	○	○	○	○	○	○
2	1	0	(CF)←0	X	X	X	X	X	X	X	X	0
2	1	0	When (CF) = 0: (CF) ← 1 (complement CF) When (CF) = 1: (CF) ← 0	X	X	X	X	X	X	X	X	○
2	1	0	(CF)←1	X	X	X	X	X	X	X	X	1
2	1	0	(DF)←0	X	1	X	X	X	X	X	X	X
2	1	0	(DF)←1	X	1	X	X	X	X	X	X	X
2	1	0	(IF)←0	X	X	0	X	X	X	X	X	X
2	1	0	(IF)←1	X	X	1	X	X	X	X	X	X
2	1	0	When this instruction is executed the CPU is put in the HALT state. An interrupt or RESET will take the CPU out of the HALT state.	X	X	X	X	X	X	X	X	X
3	1	0	The CPU is kept in the wait state until the TEST pin is V_{OL}. When	X	X	X	X	X	X	X	X	X
2 8+EA	2 2~4	0 1	AAA and BBB are not specified When MOD = 11: No operation When MOD ≠ 11: (DATA BUS) ← (EA1)	X	X	X	X	X	X	X	X	X
2	1	0	When this is prefixed to an instruction, during the execution of the instruction a bus lock is output through the LOCK pin.	X	X	X	X	X	X	X	X	X
3	1	0	NO OPERATION	X	X	X	X	X	X	X	X	X

SYMBOLS USED AND THEIR MEANING

Symbol	Contents	Symbol	Contents
A_{CC}	Accumulator (AX, AL or AH)	LABEL	Label name
ADDR	Memory address	∨	Inclusive OR
DATA	Immediate data (data which is part of the instruction)	⊻	Exclusive OR
DISP	Displacement	∧	Logical AND
d	When d = 0 source is REG, destination is EA (R/M) When d = 1 source is EA (R/M0, destination is REG	−	Subtraction
		+	Addition
EA	Effective address	*	Multiplication
EA1, EA2	First effective address, second effective address	÷	Division
Port	I/O port	←	Direction of data transfer
r1, r2	First register, second register	↔	Exchange of data
SR	Segment register code	( )	Contents of register or memory
SEG	Segment register (CS, DS, SS and ES)	V1:V2	Pair of register or data treated as one unit
W	When 0: byte processing When 1: word processing	×	Does not affect the flag or instruction
−L	Suffix indicating low-order 8 bits	○	Flag may be changed by the instruction
−H	Suffix indicating high-order 8 bits	△	Flag is undefined after execution of the instruction
DEST	Transfer address (Destination address)	FR	Flag register

ADDRESSING MODE AND REGISTER

1. Instruction Format

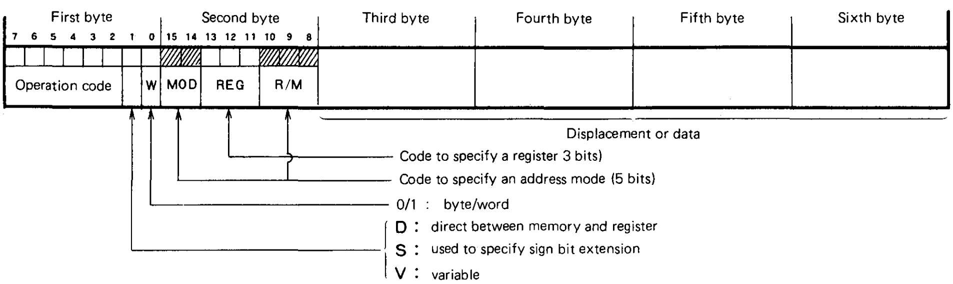


Table 1 EA Determination based on R/M and MOD

MOD / R/M	Memory mode			Register mode	
	Calculation for EA (effective address)			11	
	00	01	10	W=0	W=1
0 0 0	(BX)+(SI)	(BX)+(SI)+D8	(BX)+(SI)+D16	AL	AX
0 0 1	(BX)+(DI)	(BX)+(DI)+D8	(BX)+(DI)+D16	CL	CX
0 1 0	(BP)+(SI)	(BP)+(SI)+D8	(BP)+(SI)+D16	DL	DX
0 1 1	(BP)+(DI)	(BP)+(DI)+D8	(BP)+(DI)+D16	BL	BX
1 0 0	(SI)	(SI)+D8	(SI)+D16	AH	SP
1 0 1	(DI)	(DI)+D8	(DI)+D16	CH	BP
1 1 0	Direct address	(BP)+D8	(BP)+D16	DH	SI
1 1 1	(BX)	(BX)+D8	(BX)+D16	BH	DI

Note 31: D8: 8-bit displacement variable, D16: 16-bit displacement variable

Table 2 Register code

REG	W=0	W=1
0 0 0	AL	AX
0 0 1	CL	CX
0 1 0	DL	DX
0 1 1	BL	BX
1 0 0	AH	SP
1 0 1	CH	BP
1 1 0	DH	SI
1 1 1	BH	DI

Segment override prefix

0	0	1	SR	1	1	0

Segment register code

SR	SEG
0 0	ES
0 1	CS
1 0	SS
1 1	DS

2. Effective Address (EA) Calculation Time

EA configuration		Segment register used	Computing time
Displacement only	Direct address	DS	6 (clocks)
Base or index register	BP	SS	5
	BX, SI, DI	DS	
Displacement + base or index register	(BP + D8 or D16)	SS	9
	(BX + D8 or D16, S1 + D8 or D16, D1 + D8 or D16)	DS	
Base register + index register	BP+DI	SS	7
	BX+SI	DS	
	BP+SI	SS	8
	BX+DI	DS	
Displacement + base register + index register	BP+DI+D8 or D16	SS	11
	BX+SI+D8 or D16	DS	
	BP+SI+D8 or D16	SS	12
	BX+DI+D8 or D16	DS	

Note 32: When the segment override prefix is used the segment register used (column 2) is changed to the segment register specified by | 0 | 0 | 1 | SR | 1 | 1 | 0 | and 2 clock cycles must be added to the time (column 3) above.

3. Flag Register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X	X	X	X	OF	DF	IF	TF	SF	ZF	X	AF	X	PF	X	CF

High-order 8 bits (bits 15–8); Low-order 8 bits (bits 7–0)

Table 3 Flag code and name

OF :	overflow flag	When an arithmetic overflow occurs, when 2 operands are exclusive ORed up to the high-order bit and the high-order bit bit is 1, OF is set.
DF :	direction flag	
IF :	interrupt enable flag	
TF :	trap flag	
SF :	sign flag	When the high-order bit is 1, SF is set.
ZF :	zero flag	When the result is zero, ZF is set.
AF :	auxiliary flag	When there is a borrow from the low-order 4 bits, AF is set.
PF :	parity flag	When the number of 1's in the low-order 8 bits is even, PF is set.
CF :	carry flag	When a carry is generated from the high-order bit, CF is set.

OF, SF, ZF, AF, PF and CF are set/reset after the operation is completed.

8086 INSTRUCTION SET MATRIX

$D_3 \sim D_0$ \ $D_7 \sim D_4$ (Hexadecimal notation)		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	ADD b,ear	ADC b,ear	AND b,ear	XOR b,ear	INC AX	PUSH AX	—	JO	See table below	NOP	MOV AL←m	MOV AL←i	—	See table below	LOOPNZ /LOOPNE	LOCK
0001	1	ADD w,ear	ADC w,ear	AND w,ear	XOR w,ear	INC CX	PUSH CX	—	JNO		XCHG CX	MOV AX←m	MOV CL←i	—		LOOPZ/ LOOPE	—
0010	2	ADD b,rea	ADC b,rea	AND b,rea	XOR b,rea	INC DX	PUSH DX	—	JB/ JNAE		XCHG DX	MOV AL←m	MOV DL←i	RET (i+SP)		LOOP	REP z=0
0011	3	ADD w,rea	ADC w,rea	AND w,rea	XOR w,rea	INC BX	PUSH BX	—	JNB/ JAE		XCHG BX	MOV AX←m	MOV BL←i	RET		JCXZ	REP z=1
0100	4	ADD b,ia	ADC b,i	AND b,i	XOR b,i	INC SP	PUSH SP	—	JE/ JZ	TEST b,ea	XCHG SP	MOVS b	MOV AH←i	LES	AAM	IN b	HLT
0101	5	ADD w,ia	ADC w,i	AND w,i	XOR w,i	INC BP	PUSH BP	—	JNE/ JNZ	TEST w,ea	XCHG BP	MOVS w	MOV CH←i	LDS	AAD	IN w	CMC
0110	6	PUSH ES	PUSH SS	SEG ES	SEG SS	INC SI	PUSH SI	—	JBE/ JNA	XCHG b,ea	XCHG SI	CMPS b	MOV DH←i	MOV b,ea,i	—	OUT b	See table below
0111	7	POP ES	POP SS	DAA	AAA	INC DI	PUSH DI	—	JNBE/ JA	XCHG w,ea	XCHG DI	CMPS w	MOV BH←i	MOV w,ea,i	XLAT	OUT w	
1000	8	OR b,ear	SBB b,ear	SUB b,ear	CMP b,ear	DEC AX	POP AX	—	JS	MOV b,ear	CBW	TEST b,i,a	MOV AX←i	—	ESC 0	CALL d	CLC
1001	9	OR w,ear	SBB w,ear	SUB w,ear	CMP w,ear	DEC CX	POP CX	—	JNS	MOV w,ear	CWD	TEST w,i,a	MOV CX←i	—	ESC 1	JMP d	STC
1010	A	OR b,rea	SBB b,rea	SUB b,rea	CMP b,rea	DEC DX	POP DX	—	JP/ JPE	MOV b,rea	CALL l,d	STOS b	MOV DX←i	RET l,(i+SP)	ESC 2	JMP l,d	CLI
1011	B	OR w,rea	SBB w,rea	SUB w,rea	CMP w,rea	DEC BX	POP BX	—	JNP/ JPO	MOV w,rea	WAIT	STOS w	MOV BX←i	RET l	ESC 3	JMP si,d	STI
1100	C	OR b,i	SBB b,i	SUB b,i	CMP b,i	DEC SP	POP SP	—	JL/ JNGE	MOV easr	PUSHF	LODS b	MOV SP←i	INT type 3	ESC 4	IN b,v=1	CLD
1101	D	OR w,i	SBB w,i	SUB w,i	CMP w,i	DEC BP	POP BP	—	JNL/ JGE	LEA	POPF	LODS w	MOV BP←i	INT (any)	ESC 5	IN w,v=1	STD
1110	E	PUSH CS	PUSH DS	SEG CS	SEG DS	DEC SI	POP SI	—	JLE/ JNG	MOV srea	SAHF	SCAS b	MOV SI←i	INTO	ESC 6	OUT b,v=1	See table below
1111	F	—	POP DS	DAS	AAS	DEC DI	POP DI	—	JNLE /JG	POP ea	LAHF	SCAS w	MOV DI←i	IRET	ESC 7	OUT w,v=1	

TABLE GROUP INSTRUCTION CODE LIST

mod□r/m	000	001	010	011	100	101	110	111
immed	ADD	OR	ADC	SBB	AND	SUB	XOR	CMP
Shift	ROL	ROR	RCL	RCR	SHL/ SAL	SHR	—	SAR
Grp 1	TEST	—	NOT	NEG	MUL	IMUL	DIV	IDIV
Grp 2	INC	DEC	CALL id	CALL l,id	JMP d	JMP l,id	PUSH	—

Note 33: Special symbols used only in the "Instruction set matrix" and the "Group Instruction Code List".
EA➡effective address (including register mode), REG⬅register

- b : byte operation
- w : Word operation
- – : this code should not be used because the result and function are undefined
- a : accumulator
- d : direct address
- ea : calculation of EA
- ear : processing results of EA and REG are transferred to EA
- easr : (EA) ← (SR)
- i : immediate data
- ia : immediate data and accumulator
- id : indirect address
- is : immediate data in sign extended form
- l : segment is included in the jump
- m : memory
- rea : processing results of REG and EA are transferred to REG
- si : sign of 8 byte displacement is extended
- srea : (SR) ← (EA)
- v : variable
- z : z bit
- ← : shows direction of transfer.

Note 34: The length of instructions varies from 1 byte (8 bits) to 6 bytes. The "Instruction Set Matrix" is ordered by the hexadecimal value of the first byte of the instruction. The instruction and its operands (an instruction may have no operand) are listed in mnemonic or symbolic form. The group instructions (those instructions with different functions depending on bit D_5, D_4, D_3 in the second word of the instruction) are shown in the "Group Instruction Code List".

OCTAL LATCH

DESCRIPTION

The M5L8282P and M5L8283P are semiconductor integrated circuits consisting of sets of eight 3-state latches for use with various types of microprocessors.

FEATURES

- 3-state, high-fanout output
  .... (I_{OL} = 32mA, I_{OH} = –5mA)
- Pin and electrical compatibility with the Intel 8282 and 8283

APPLICATION

Data latches for various microcomputer systems

FUNCTION

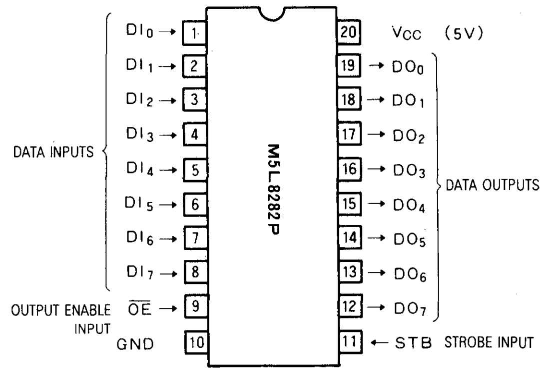
The M5L8282P and M5L8283P are latches with non-inverted and inverted outputs, respectively.

When the strobe input STB is high, the data inputs $DI_0 \sim DI_7$ are passed through the data outputs $DO_0 \sim DO_7$ (M5L8282P) or to the data outputs $\overline{DO_0} \sim \overline{DO_7}$ (M5L8283P), changes in the $DI_0 \sim DI_7$ signals being reflected in the data outputs.

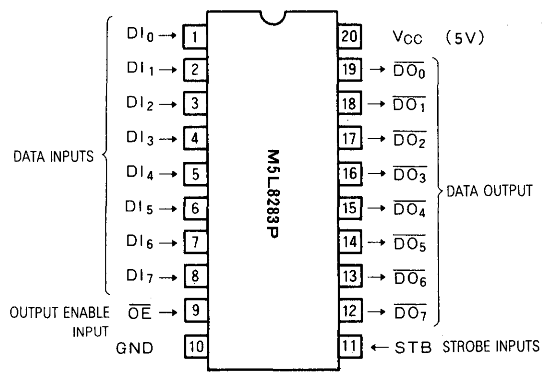
If the STB is changed from high to low, the data $DI_0 \sim DI_7$ just before the change is latched. If the DI data is changed while STB is low, this change is not reflected in the data outputs.

When $\overline{OE}$ is made high, all the data outputs go into the high-impedance state, the data latched prior to $\overline{OE}$ going high being held.

PIN CONFIGURATION (TOP VIEW)

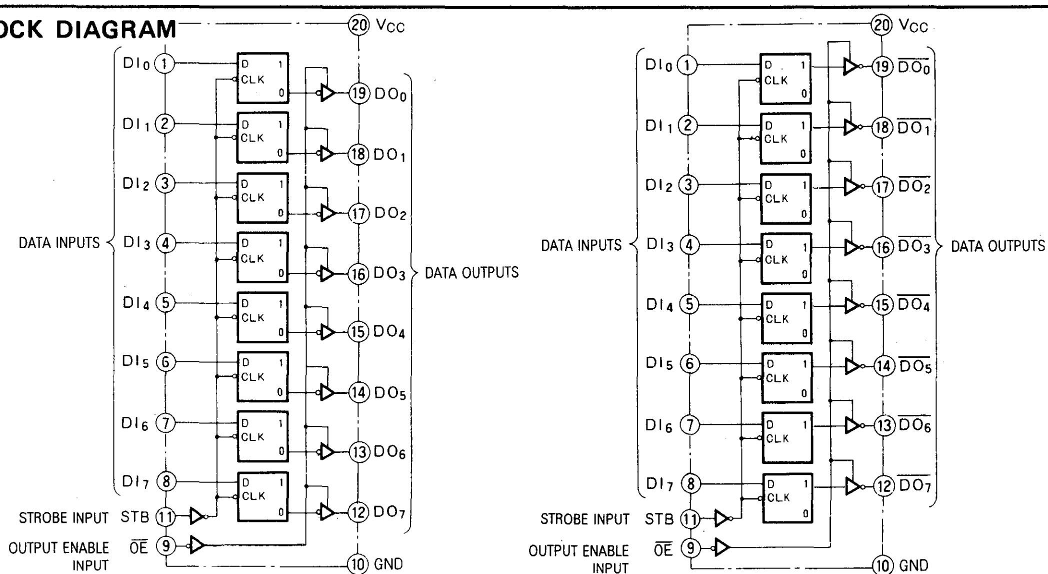


Outline 20P4



Outline 20P4

BLOCK DIAGRAM



ABSOLUTE MAXIMUM RATINGS (T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage		−0.5 ~ +7	V
V_I	Input voltage		−0.5 ~ +5.5	V
V_O	Output voltage		−0.5 ~ V_{CC}	V
Topr	Operating free-air temperature range		0 ~ +75	°C
Tstg	Storage temperature range		−65 ~ +150	°C

RECOMMENDED OPERATING CONDITIONS (T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min	Nom	Max	
V_{CC}	Supply voltage		4.5	5	5.5	V
I_{OH}	High-level output current	$V_{OH} \geqq 2.4V$	0		−5	mA
I_{OL}	Low-level output current	$V_{OL} \leqq 0.45V$	0		32	mA

ELECTRICAL CHARACTERISTICS (T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{IH}	High-level input voltage		2			V
V_{IL}	Low-level input voltage				0.8	V
V_{IC}	Input clamp voltage	$V_{CC}=4.5V$, $I_{IC}=-5mA$			−1	V
V_{OH}	High-level output voltage	$V_{CC}=4.5V$, $I_{OH}=-5mA$	2.4			V
V_{OL}	Low-level output voltage	$V_{CC}=4.5V$, $I_{OL}=32mA$			0.45	V
I_{OZH}	Off-state output current, high-level applied to the output	$V_{CC}=5.5V$, $V_I=2V$, $V_O=5.25V$			50	μA
I_{OZL}	Off-state output current, low-level applied to the output	$V_{CC}=5.5V$, $V_I=2V$, $V_O=0.4V$			−50	μA
I_{IH}	High-level input current	$V_{CC}=5.5V$, $V_I=5.25V$			50	μA
I_{IL}	Low-level input current	$V_{CC}=5.5V$, $V_I=0.45V$			−0.2	mA
I_{CC}	Supply current	$V_{CC}=5.5V$			160	mA
C_{IN}	Input capacitance	F = 1MHz, $V_{BIAS}=2.5V$ $V_{CC}=5V$, Ta = 25°C			12	pF

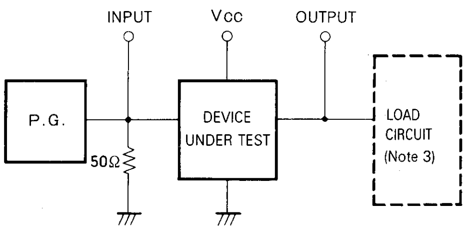
SWITCHING CHARACTERISTICS (V_{CC} = 5V ±10%, T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter	Alternate symbol	Test conditions	M5L8282P			M5L8283P			Unit
				Limits			Limits			
				Min	Typ	Max	Min	Typ	Max	
t_{PLH} t_{PHL}	Propagation time from DI input to DO or $\overline{DO}$ for low-to-high or high-to-low change	T_{IVOV}	(Note 2)	5		30	5		22	ns
t_{PLH} t_{PHL}	Propagation time from STB input to DO or $\overline{DO}$ for low-to-high and high-to-low change	T_{SHOV}	(Note 2)	10		45	10		40	ns
t_{PZH} t_{PZL}	Propagation time from $\overline{OE}$ input to DO or $\overline{DO}$ output when output is enabled	T_{ELOV}	(Note 2)	10		30	10		30	ns
t_{PHZ} t_{PLZ}	Propagation time from $\overline{OE}$ input to DO or DO output when the output is disabled	T_{EHOV}	(Note 2)	5		18	5		18	ns
t_r	Output risetime	T_{OLOH}	From 0.8V to 2V			20			20	ns
t_f	Output falltime	T_{OHOL}	From 2V to 0.8V			12			12	ns

TIMING REQUIREMENTS (V_{CC} = 5V ±10%, T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter	Alternate symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
$t_{w(STBH)}$	Strobe STB high pulse width	T_{SHSL}		15			ns
t_{su}	Strobe STB setup time for DI_0 ~ DI_7	T_{IVSL}		0			ns
t_h	STB hold time for DI_0 ~ DI_7	T_{SLIX}		25			ns
t_r	Input risetime	T_{ILIH}	From 0.8V to 2V			20	ns
t_f	Input falltime	T_{ILIH}	From 2V to 0.8V			12	ns

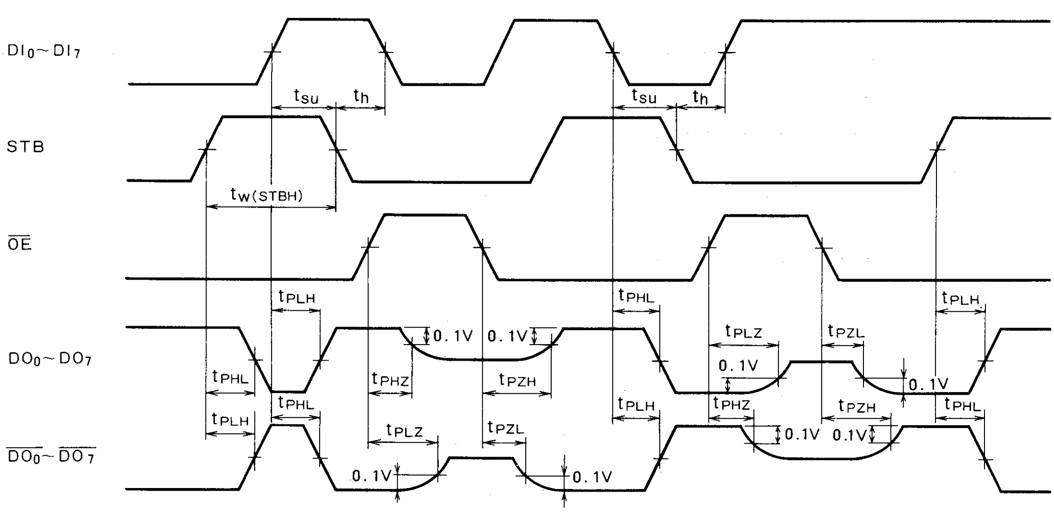
Note 2. Test Circuit



Note 3.

TEST ITEM	t_{PLH}, t_{PHL}	t_{PLZ}, t_{PZL}	t_{PHZ}, t_{PZH}
LOAD CIRCUIT	2.14V, 52.7Ω, 300pF	1.5V, 33Ω, 300pF	1.5V, 180Ω, 300pF

TIMING DIAGRAM (Reference voltage = 1.5V)

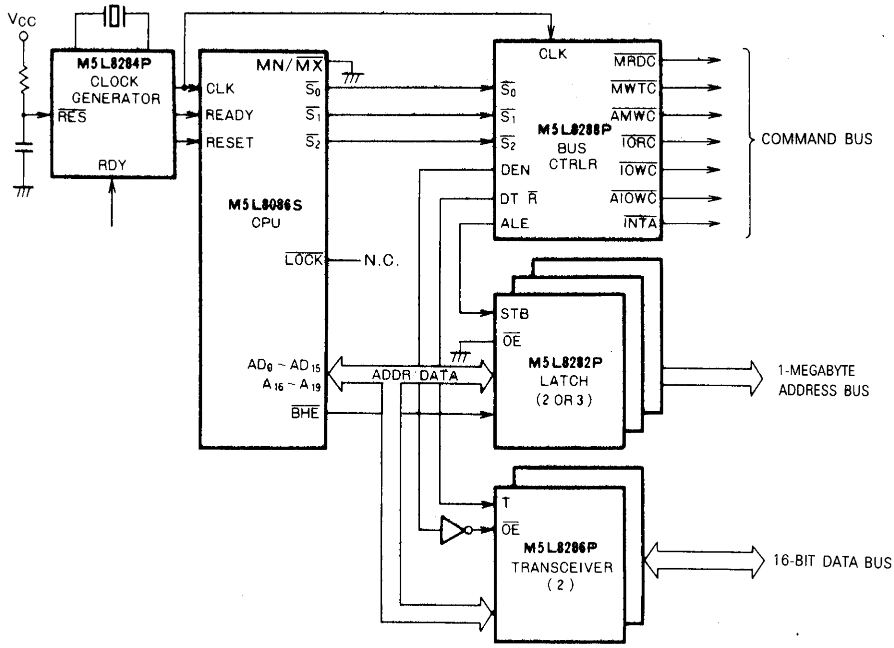


PRECAUTIONS FOR USE

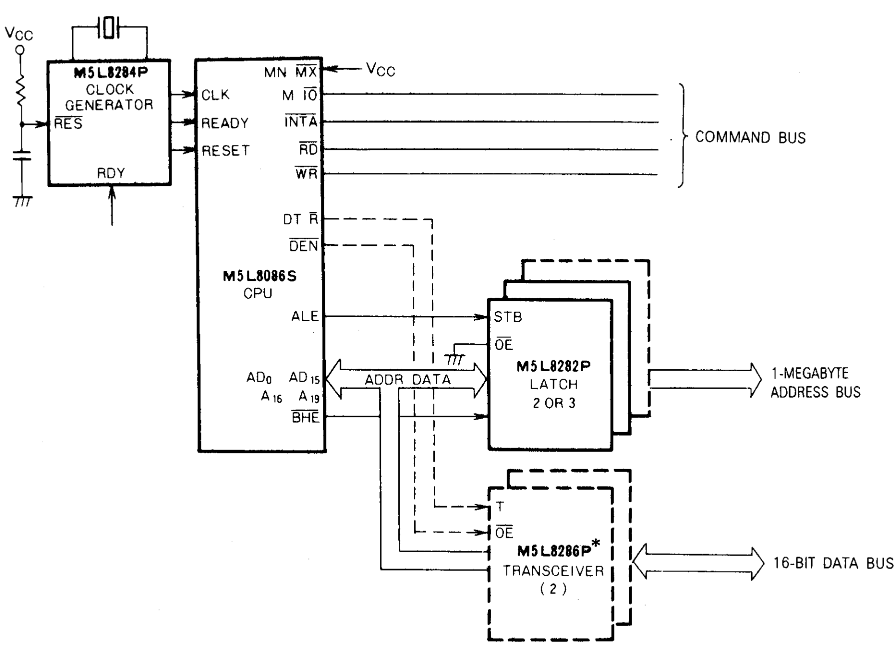
Care should be taken to accommodate the glitch that is generated when STB goes from low to high with the output low for the M5L8283P.

APPLICATION EXAMPLES

(1) Use in the maximum mode



(2) Use in the minimum mode



* : Option
Required when the number of devices driving the bus increases

CLOCK GENERATOR AND DRIVER FOR 8086, 8088, 8089 PROCESSORS

DESCRIPTION

The M5L8284P is a semiconductor integrated circuit consisting of a clock generator for use with the 8086 and 8088 16-bit microprocessors.

It has a synchronous delay circuit and synchronous reset circuit capable of controlling two Multibus (Intel trade mark) circuits.

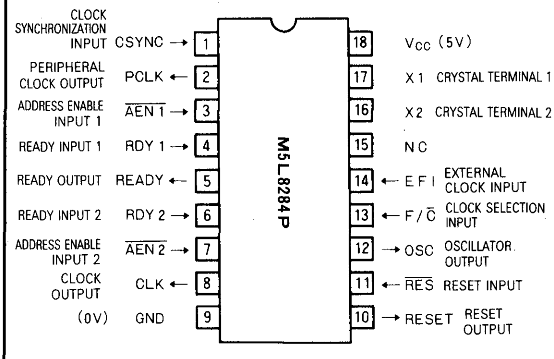
FEATURES

- Stable, crystal controlled output frequency
- Synchronous operation of several M5L8284Ps is possible
- External clock input
- By means of an external capacitor and resistance a power-on reset signal can be generated
- Pin and electrical compatibility with the Intel 8284

APPLICATION

Clock driver and generators for the 8086 or 8088.

PIN CONFIGURATION (TOP VIEW)



Outline 18P4

NC: NO CONNECTION

FUNCTION

The M5L8284P is a clock generator/driver for the 8086, 8088 or 8089 processors.

Internally the crystal oscillator signal is divided by three to provide the clock output CLK, and by two to provide the peripheral clock output PCLK. In addition, a reset circuit and ready circuit are provided to ensure synchronization to the CLK signal.

The reset input $\overline{RES}$ is used to generate the reset output $\overline{RESET}$ as the CPU reset signal synced to the CLK signal. A Schmitt trigger circuit is used at the input side.

Thus, a reset signal can be output at power on by connecting a capacitor and resistor to the $\overline{RES}$ input.

The clock selection input $F/\overline{C}$ can be used to select the crystal oscillator circuit output or an external clock input as the input for the divide by three circuit.

By using these pins, the M5L8284P output can be used to drive multiple M5L8284P devices.

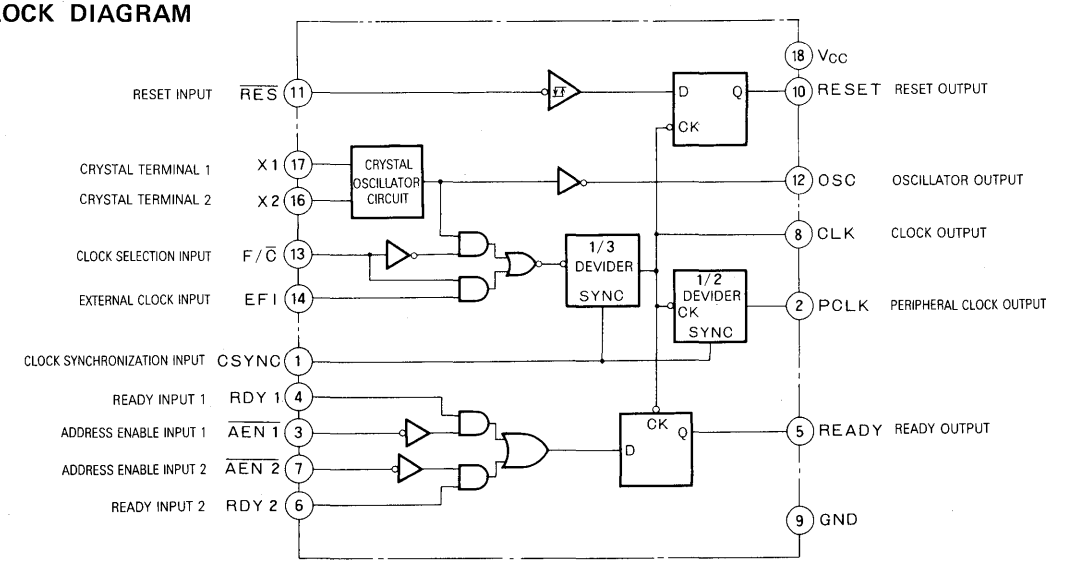
The clock synchronization input CSYNC is used to operate multiple M5L8284Ps in synchronous.

The ready inputs RDY1 and RDY2 are used to generate the ready output READY. These ready inputs are valid when the address enable inputs $\overline{AEN1}$ or $\overline{AEN2}$ respectively are low.

The PCLK, RESET, and READY signals operate internally synchronized to the CLK signal.

9

BLOCK DIAGRAM



CLOCK GENERATOR AND DRIVER FOR 8086, 8088, 8089 PROCESSORS

PIN DESCRIPTIONS

Pin	Name	Input or output	Function
$\overline{AEN1}$, $\overline{AEN2}$	Address enable input	Input	When $\overline{AEN1}$ and $\overline{AEN2}$ are made low, RDY1 and RDY2 are made effective respectively. By using these two inputs separately, the CPU can be used to access two Multibusses. When not used as a multimaster, $\overline{AEN}$ should be set to low. These inputs are active low.
RDY1, RDY2	Bus ready input	Input	These inputs are connected to the signal indicating the completion of data reception from a system bus device or a signal output indicating that data is valid. RDY1 and RDY2 are effective when $\overline{AEN1}$ and $\overline{AEN2}$ are low respectively. These inputs are active high.
READY	Ready output	Output	The state of RDY appears at this output in synchronization with the CLK output. This is done to synchronize the READY output to the M5L8284P internal CLK because the RDY input changes in an asynchronous fashion with respect to CLK. This pin is normally connected to the CPU ready input and cleared after the required CPU hold time, t_h.
X_1, X_2	Crystal element terminals	Input	These pins are used to make connections to the crystal. The crystal should have a frequency such that the period is three times the CPU cycle time. The crystal should be chosen in the range 12~25MHz and have as low as possible a series resistance. Care should be taken not to ground these pins.
F/$\overline{C}$	Clock selection input	Input	When this input is set to low, the CLK, and PCLK outputs are driven from the crystal oscillator output and when it is set to high, they are driven from the EFI input.
EFI	External clock input	Input	When F/$\overline{C}$ is high, the signal input at this pin is used to drive CLK and PCLK. The input is a rectangular TTL level signal of frequency such that the period is three times the CPU cycle time.
CLK	Clock output	Output	This output is connected to the CPU and the clock inputs of the surrounding ICs connected to the local bus. The output waveform is 1/3 the frequency of the crystal connected to X_1 and X_2 or the FEI input frequency and has a duty cycle of 1/3. Since for V_{CC} = 5V, $V_{OH} \geqq 4.5V$, this output can be directly connected to the CPU clock input.
PCLK	Peripheral clock output	Output	This output is used as the clock signal for peripheral devices. The output waveform is a 50% duty cycle TTL level rectangular waveform having a frequency of 1/2 the CLK output frequency.
OSC	Oscillator output	Output	This output is a TTL level crystal oscillator circuit output. The frequency is the same as that of the crystal connected to X_1 and X_2 but care should be taken as the frequency will be unstable if these pins are left open.
$\overline{RES}$	Reset input	Input	This active low input is used to generate the reset output signal for the CPU. The input uses a Schmitt trigger circuit so that by connecting a capacitor and resistance, the CPU power-up reset can be generated.
RESET	Reset output	Output	Connected to the CPU RESET input. The $\overline{RES}$ is synchronized to the CLK signal. This output is active high.
CSYNC	Clock synchronization input	Input	For using multiple M5L8284P devices, this input is used as a clock synchronization input. When CSYNC is made high, the internal counter of the M5L8284P is reset and when it is made low it begins operation. CSYNC must be synchronized with EFI. Refer to the Section on using this device.

ABSOLUTE MAXIMUM RATINGS (T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage		−0.5 ~ +7	V
V_I	Input voltage		−0.5 ~ +5.5	V
V_O	Output voltage		−0.5 ~ +V_{CC}	V
Topr	Operating free-air temperature range		0 ~ +75	°C
Tstg	Storage temperature range		−65 ~ +150	°C

CLOCK GENERATOR AND DRIVER FOR 8086, 8088, 8089 PROCESSORS

RECOMMENDED OPERATING CONDITIONS (T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter		Limits Min	Limits Nom	Limits Max	Unit
V_{CC}	Supply voltage		4.5	5	5.5	V
I_{OH}	High-level output current	CLK $V_{OH} \geqq 4V$ Other outputs $V_{OH} \geqq 2.4V$	0		−1	mA
I_{OL}	Low-level output current	$V_{OL} \leqq 0.45V$	0		5	mA

ELECTRICAL CHARACTERISTICS (T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits Min	Limits Typ	Limits Max	Unit
V_{IH}	High-level input voltage	$\overline{RES}$		2.6			V
		Other inputs		2			V
V_{IL}	Low-level input voltage					0.8	V
$V_{T+} \cdot V_{T-}$	Hysteresis width	$\overline{RES}$	$V_{CC}=5V$	0.25			V
V_{IC}	Input clamp voltage		$V_{CC}=4.5V$, $I_{IC}=-5mA$			−1	V
V_{OH}	High-level output voltage	CLK	$V_{CC}=4.5V$, $I_{OH}=-1mA$	4			V
		Other outputs		2.4			V
V_{OL}	Low-level output voltage		$V_{CC}=4.5V$, $I_{OL}=5mA$			0.45	V
I_{IH}	High-level input current		$V_{CC}=5.5V$, $V_I=5.25V$			50	μA
I_{IL}	Low-level input current		$V_{CC}=5.5V$, $V_I=0.45V$			−0.5	mA
I_{CC}	Supply current		$V_{CC}=5.5V$			140	mA

SWITCHING CHARACTERISTICS (V_{CC} = 5V ±10%, T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter	Alternate symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
T_C	CLK repetition period	TCLCL	(Note 5.)	125			ns
$T_{W(CLKH)}$	CLK high pulse width	TCHCL		(1/3TC)+2			ns
$T_{W(CLKL)}$	CLK low pulse width	TCLCH		(2/3TC)−15			ns
t_{TLH}	CLK low-level to high-level transition time	TCH1CH2	1V~3.5V			10	ns
t_{THL}	CLK high-level to low-level transition time	TCL2CL1	3.5V~1V			10	ns
$T_{W(PCLKH)}$	PCLK high pulse width	TPHPL		TC-20			ns
$T_{W(PCLKL)}$	PCLK low pulse width	TPLPH		TC-20			ns
td_{IV}	READY invalid time with respect to CLK (Note 1)	TRYLCL		−8			ns
td_V	READY valid time with respect to CLK (Note 2)	TRYHCH		(2/3TC)−15			ns
$T_{DHL(CLK\text{-}RESET)}$	High-level to low-level delay time From CLK to RESET	TCLIL		40			ns
$T_{DLH(CLK\text{-}PCLK)}$	Low-level to high-level delay time From CLK to PCLK	TCLPH				22	ns
$T_{DHL(CLK\text{-}PCLK)}$	High-level to low-level delay time From CLK to PCLK	TCLPL				22	ns
$T_{DLH(OSC\text{-}CLK)}$	Low-level to high-level delay time From OSC to CLK	TOLCH		−5		12	ns
$T_{DHL(OSC\text{-}CLK)}$	High-level to low-level delay time From OSC to CLK	TOLCL		2		20	ns

9

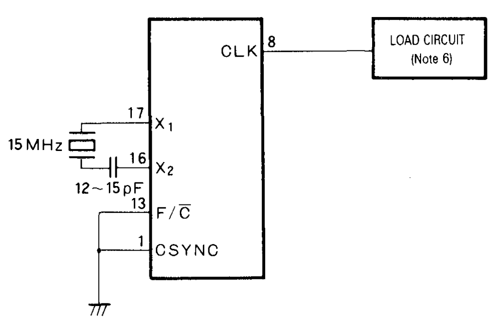
CLOCK GENERATOR AND DRIVER FOR 8086, 8088, 8089 PROCESSORS

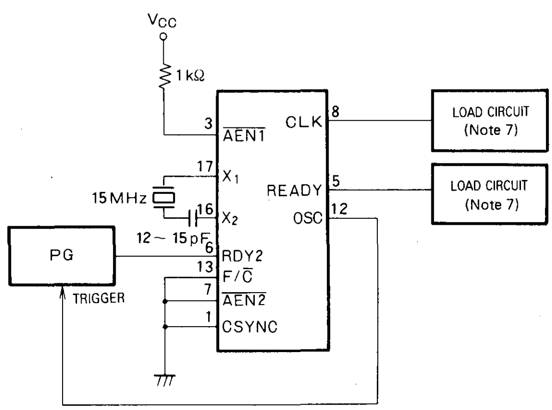
TIMING REQUIREMENTS (V_{CC} = 5V ±10%, T_a = 0~75°C, unless otherwise noted)

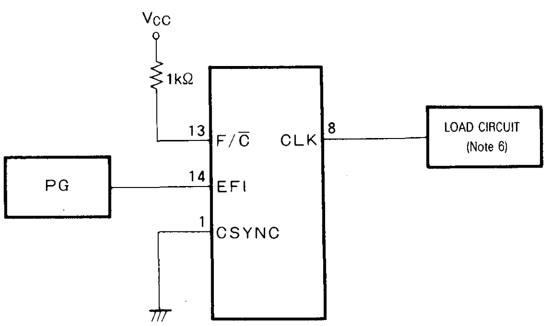
Symbol	Parameter	Alternate symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$f_{(X'tal)\,max}$	Crystal frequency			12		25	MHz
$t_{W(EFIH)}$	EFI high pulse width	TEHEL	V_I (90% − 90%)	13			ns
$t_{W(EFIL)}$	EFI low pulse width	TELEH	V_I (10% − 10%)	13			ns
$T_{C(FEI)}$	EFI repetition period (Note 3)	TELEL		$t_{W(EFIH)}$ $t_{W(EFIL)}$			ns
$t_{SU(RDY)}$	RDY1 and RDY2 setup time with respect to CLK	TR1VCL		35			ns
$t_{h(RDY)}$	RDY1 and RDY2 hold time with respect to CLK	TCLR1X		0			ns
$t_{SU(AEN)}$	$\overline{AEN1}$ and $\overline{AEN2}$ setup time with respect to RDY1 and RDY2	TA1R1V	(Note 5)	15			ns
$t_{h(AEN)}$	$\overline{AEN1}$ and $\overline{AEN2}$ hold time with respect to CLK	TCLA1X		0			ns
$t_{SU(CSYNC)}$	CSYNC setup time with respect to EFI	TYHEH		20			ns
$t_{h(CSYNC)}$	CSCYNC hold time with respect to EFI	TEHYL		20			ns
$t_{W(CSYNC)}$	CSYNC pulse width	TYHYL		$2T_{C(EFI)}$			ns
$t_{SU(\overline{RES})}$	$\overline{RES}$ setup time with respect to CLK (Note 4)	TI1HCL		65			ns
$t_{h(\overline{RES})}$	$\overline{RES}$ hold time with respect to CLK (Note 4)	TCLI1H		20			ns

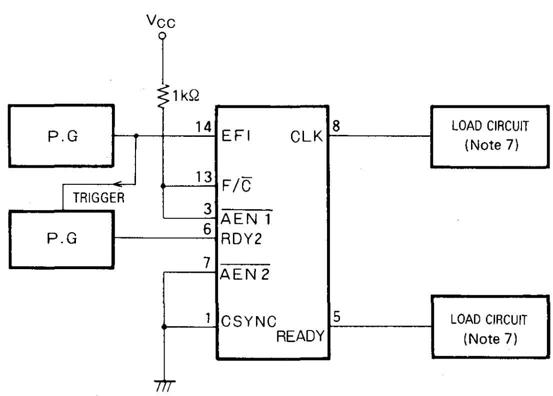
Note 1. Applies to T2 state time
2. Applies to T3 and TW state times
3. δ = EFI t_R (5ns max) + EFI t_F (5ns max)
4. $t_{SU(\overline{RES})}$ and $t_{h(\overline{RES})}$ are required only to guarantee the next clock period

Note 5. Test circuit





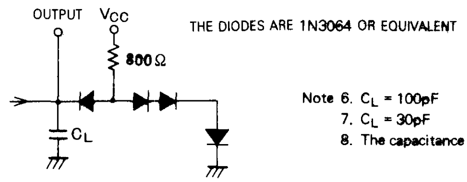




MITSUBISHI ELECTRIC

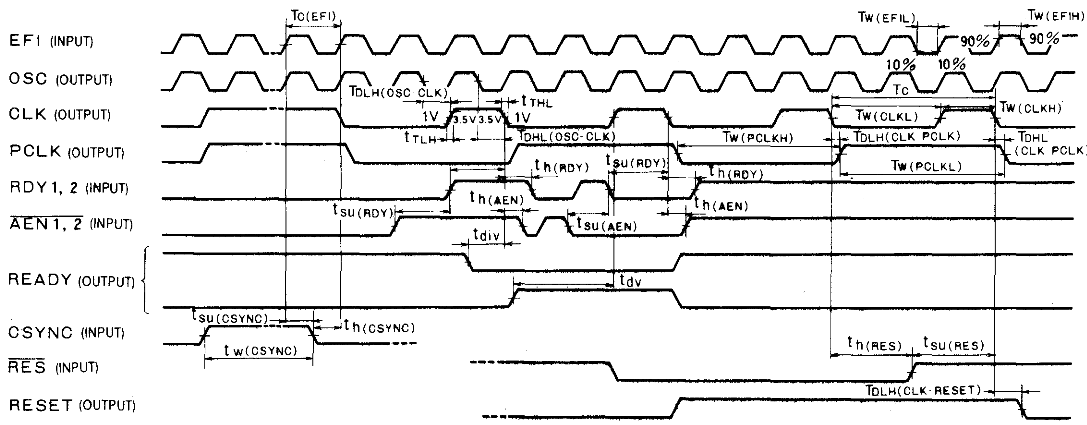
CLOCK GENERATOR AND DRIVER FOR 8086, 8088, 8089 PROCESSORS

LOAD CIRCUIT



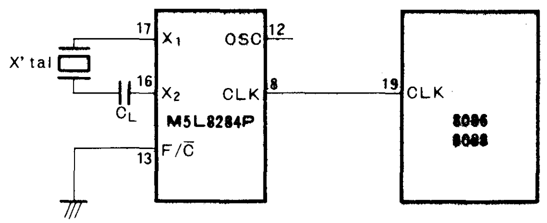
Note 6. C_L = 100pF
7. C_L = 30pF
8. The capacitance C_L includes all the coupling and floating capacitances of the probe input circuit.

TIMING DIAGRAM (Reference level = 1.5V)



APPLICATION NOTES

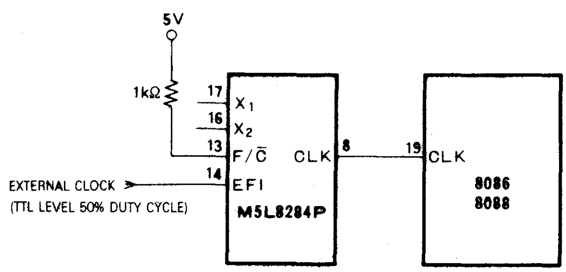
(1) Connecting the crystal



The crystal frequency should be chosen such that the period is three times the cycle time of the 8086 or 8088 and when connecting it care should be taken that it is located as close as possible to the M5L8284P.

To ensure stable oscillations, a ceramic capacitor should be placed in series with the crystal at pin X_2. Suitable values of capacitance for 15MHz, 12MHz, and 22MHz, are 12~15pF, 24pF, and 8pF respectively.

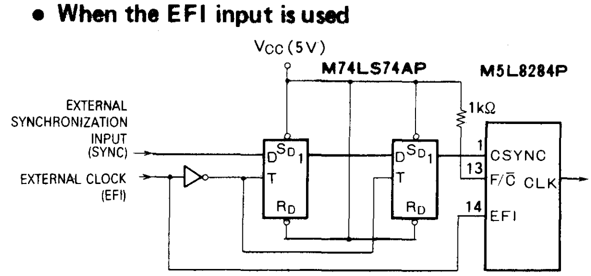
(2) External clock connections



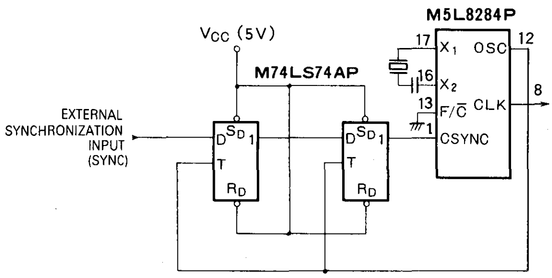
The maximum frequency of the external clock is 25MHz and while there is no lower limit on the frequency it should be set at least three times the CPU minimum clock frequency.

(3) Synchronizing using the CSYNC input

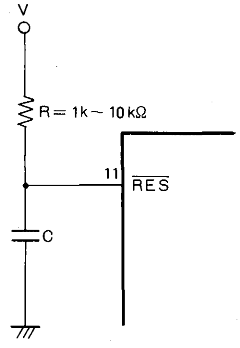
- **When the EFI input is used**



- **When the EFI input is not used**



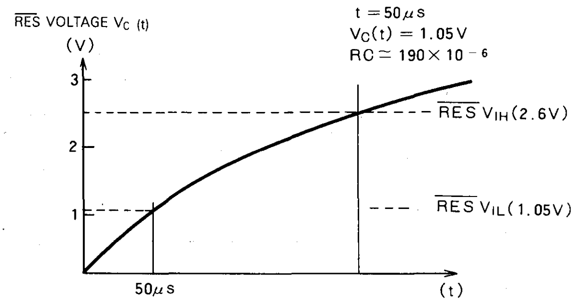
(4) Power-on reset circuit



Since the 8086, 8088 and 8089 require a reset pulse over 50μs after V_{CC} reaches 4.5V upon power up, the capacitor value should be determined by the graph shown below. Note that the time for V_{CC} to reach 4.5V has not been considered so that it is necessary to use a value of C larger than in the power supply used.

$$V_C(t) = V\left(1 - e^{\frac{-t}{RC}}\right)$$

V = 4.5V
t = 50μs
$V_C(t)$ = 1.05V
RC ≃ 190 × 10^{-6}



CHANGE IN RES INPUT VOLTAGE

PRECAUTIONS FOR USE

If noise is allowed to enter the XTAL1 and XTAL2 or V_{CC} pins, the oscillator frequency will be pulled off the parallel resident frequency and the stray capacitance between XTAL1 and XTAL2 may cause the circuit to go into relaxation oscillation. To prevent this, care should be given to the following points.

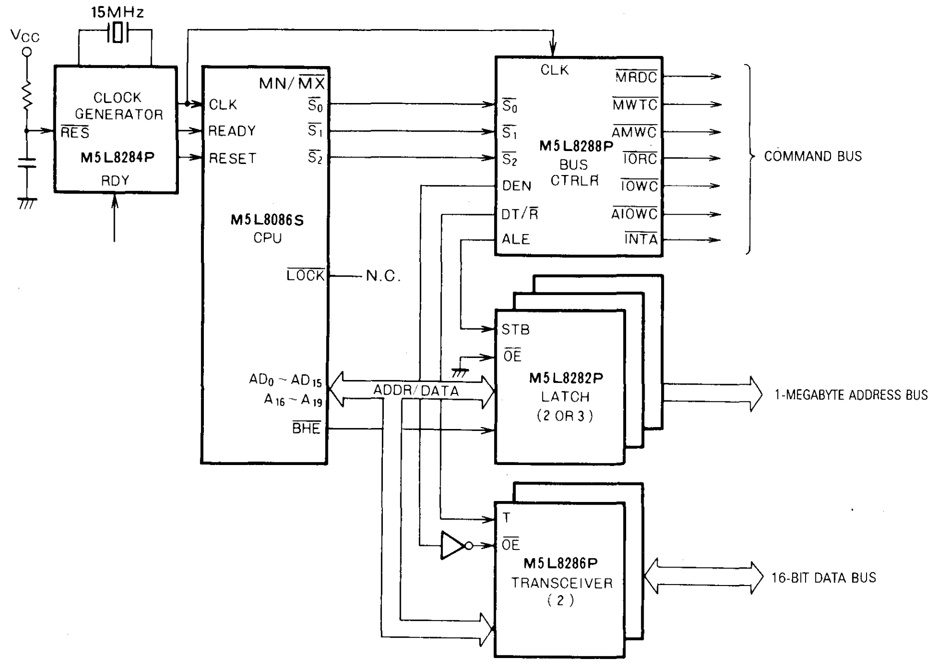
(1) The crystal should be one with a small parallel capacitance.

(2) A capacitor of value 0.01 to 0.1μF with good high frequency characteristics should be connected between V_{CC} and the ground. This capacitor should be mounted as close as possible to the IC.

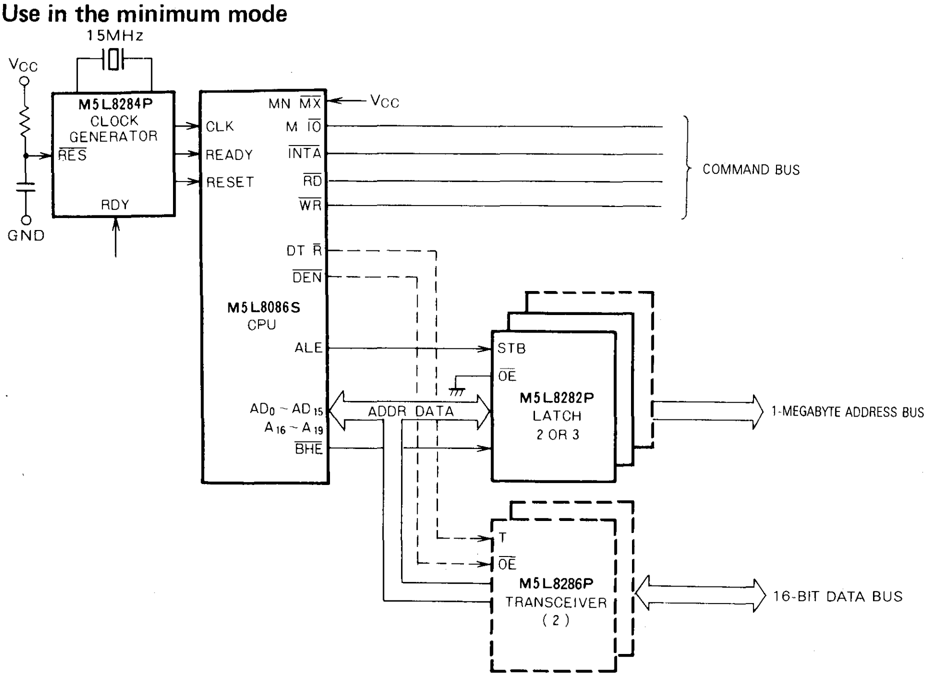
CLOCK GENERATOR AND DRIVER FOR 8086, 8088, 8089 PROCESSORS

APPLICATION EXAMPLES

(1) Use in the maximum mode



(2) Use in the minimum mode



OPTION:
REQUIRED WHEN THE NUMBER OF DEVICES DRIVING THE BUS INCREASES

OCTAL BUS TRANSCEIVER

DESCRIPTION

The M5L8286P and M5L8287P are semiconductor integrated circuits consisting of a set of eight 3-state output bus transceivers for use with a variety of microprocessor systems.

FEATURES

- 3-state, high-fanout outputs (I_{OL} = 16mA, I_{OH} = −1mA for the A outputs and I_{OL} = 32mA, I_{OH} = −5mA for the B outputs)
- Electrical and pin compatibility with the Intel 8286 and 8287

APPLICATION

Two-way bus transceivers for microcomputer systems

FUNCTION

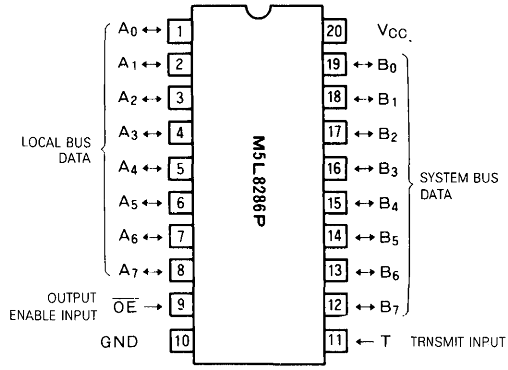
The M5L8286P and M5L8287P are two-way bus transceivers with non-inverted and inverted outputs respectively.

When the output enable input $\overline{OE}$ is high, the local bus data pins $A_0 \sim A_7$ and system data pins $B_0 \sim B_7$ are both placed in the high-impedance state.

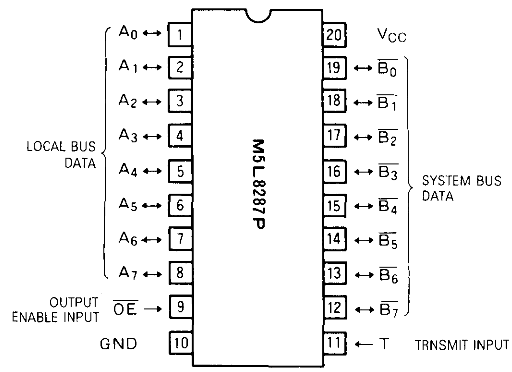
When the output enable input $\overline{OE}$ is low, the input and output states are controlled by the transmit input T.

When T is high, $A_0 \sim A_7$ are input pins and $B_0 \sim B_7$ are output pins. When T is low, $B_0 \sim B_7$ are input pins and $A_0 \sim A_7$ are output pins.

PIN CONFIGURATIONS (TOP VIEW)

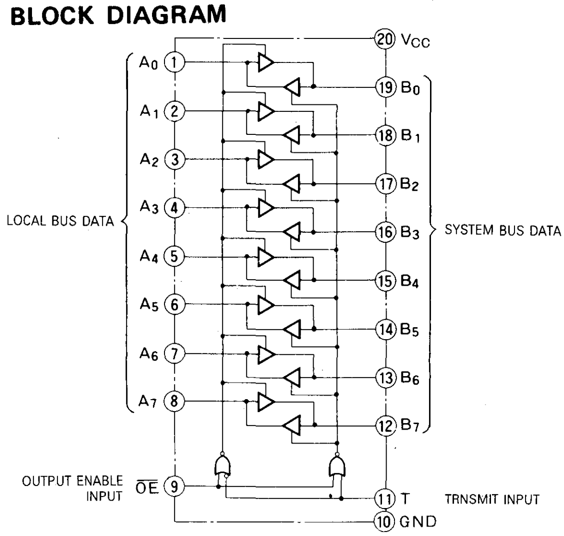


Outline 20P4

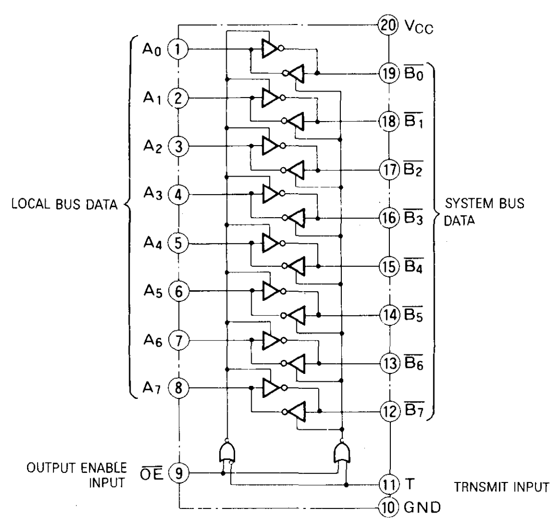


Outline 20P4

BLOCK DIAGRAM



M5L8286P



M5L8287P

OCTAL BUS TRANSCEIVER

FUNCTION TABLES (Note 1)

M5L8286P

$\overline{OE}$	T	A	B
L	L	O	I
L	H	I	O
H	X	Z	Z

M5L8287P

$\overline{OE}$	T	A	B
L	L	$\overline{O}$	I
L	H	I	$\overline{O}$
H	X	Z	Z

Note 1. I: Input pin
O, $\overline{O}$: Output pin (non-inverted for the M5L8286P and inverted for the M5L8287P)
Z: Indicated the high-impedance state (A and B are separated)
X: Either high or low

ABSOLUTE MAXIMUM RATINGS (T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage		−0.5 ~ +7	V
V_I	Input voltage		−0.5 ~ +5.5	V
V_O	Output voltage		−0.5 ~ V_{CC}	V
Topr	Operating free-air temperature range		0 ~ +75	°C
Tstg	Storage temperature range		−65 ~ +150	°C

RECOMMENDED OPERATING CONDITIONS (T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter			Limits			Unit
				Min	Nom	Max	
V_{CC}	Supply voltage			4.5	5	5.5	V
I_{OH}	High-level output current	$V_{OH} \geqq 2.4V$	A output	0		−1	mA
			B output	0		−5	mA
I_{OL}	Low-level output current	$V_{OL} \leqq 0.45V$	A output	0		16	mA
			B output	0		32	mA

ELECTRICAL CHARACTERISTICS (T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter		Test conditions		Limits			Unit
					Min	Typ	Max	
V_{IH}	High-level input voltage				2			V
V_{IL}	Low-level input voltage	A input					0.8	V
		B input					0.9	V
V_{IC}	Input clamp voltage		V_{CC}=4.5V, I_{IC}=−5mA				−1	V
V_{OH}	High-level output voltage	A output	V_{CC}=4.5V, I_{OH}=−1mA		2.4			V
		B output	V_{CC}=4.5V, I_{OH}=−5mA		2.4			V
V_{OL}	Low-level output voltage	A output	V_{CC}=4.5V, I_{OL}=16mA				0.45	V
		B output	V_{CC}=4.5V, I_{OL}=32mA				0.45	V
I_{OZH}	Off-state output current, with high-level applied at the output	A output	V_{CC}=5.5V, V_I=2V	V_I=0.8V			50	μA
		B output	V_O=5.25V	V_I=0.9V				
I_{OZL}	Off-state output current, with low-level applied the output	A output	V_{CC}=5.5V, V_I=2V	V_I=0.8V			−0.2	mA
		B output	V_O=0.45V	V_I=0.9V				
I_{IH}	High-level input current		V_{CC}=5.5V, V_I=5.25V				50	μA
I_{IL}	Low-level input current		V_{CC}=5.5V, V_I=0.45V				−0.2	mA
I_{CC}	Supply current	M5L8286P	V_{CC}=5.5V				160	mA
		M5L8287P					130	mA
C_{IN}	Input capacitance		F=1MHz, V_{BIAS}=2.5V V_{CC}=5V, Ta=25°C				12	pF

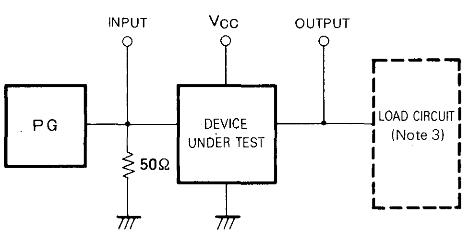
SWITCHING CHARACTERISTICS (V_{CC} = 5V ±10%, T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter	Alternate symbol	Test conditions	M5L8286P Limits Min	M5L8286P Limits Typ	M5L8286P Limits Max	M5L8287P Limits Min	M5L8287P Limits Typ	M5L8287P Limits Max	Unit
t_{PLH} t_{PHL}	Low-level to high-level and high-level and low-level transition time from input A. B to outputs B, A	TIVOV	(Note 2)	5		30	5		22	ns
t_{PZH} t_{PZL}	Output enable time from $\overline{OE}$ input to A or B output	TELOV	(Note 2)	10		30	10		30	ns
t_{PHZ} t_{PLZ}	Output disable time from $\overline{OE}$ input to A or B output	TEHOZ	(Note 2)	5		18	5		18	ns
t_r	Output risetime	TOLOH	From 0.8V to 2V			20			20	ns
t_f	Output falltime	TOHOL	From 2V to 0.8V			12			12	ns

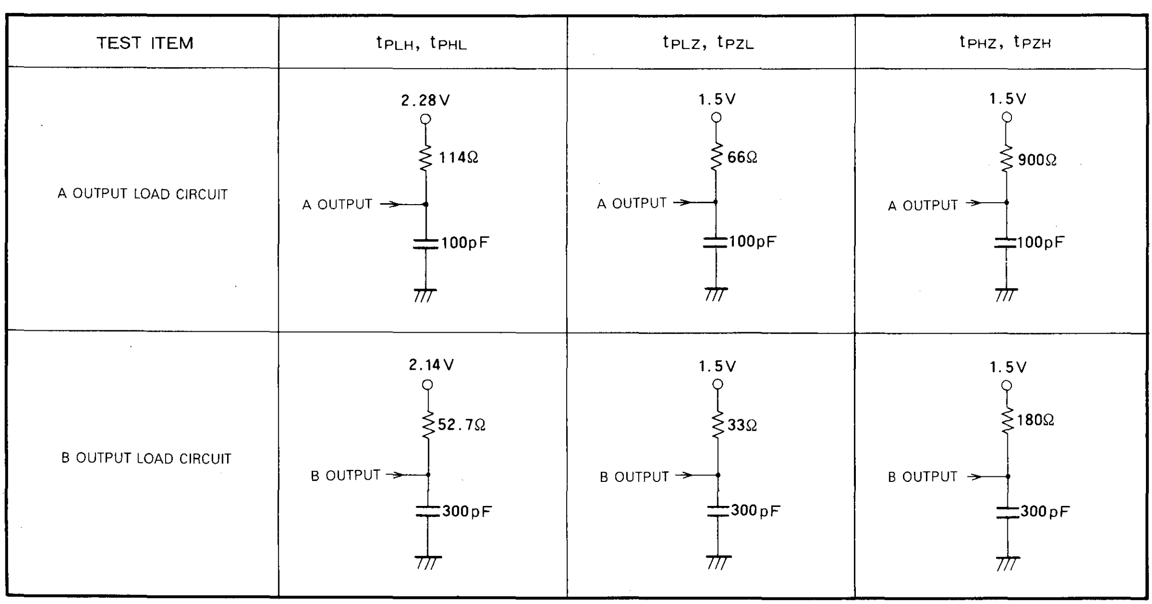
TIMING REQUIREMENTS (V_{CC} = 5V ±10%, T_a = 0~75°C, unless otherwise noted)

Symbol	Parameter	Alternate symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
t_{SU}	T setup time with respect to $\overline{OE}$	T_{TVFL}		10			ns
t_h	T hold time with respect to $\overline{OE}$	T_{EHTV}		5			ns
t_r	Input risetime	T_{ILIH}	From 0.8V to 2V			20	ns
t_f	Input falltime	T_{ILIL}	From 2V to 0.8V			12	ns

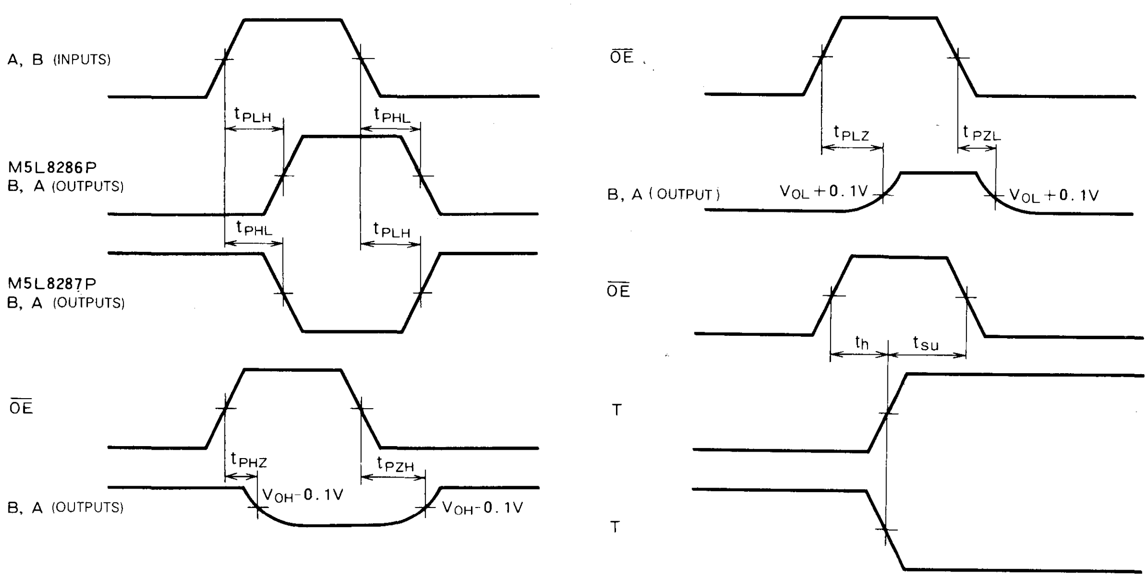
Note 2. Test Circuit



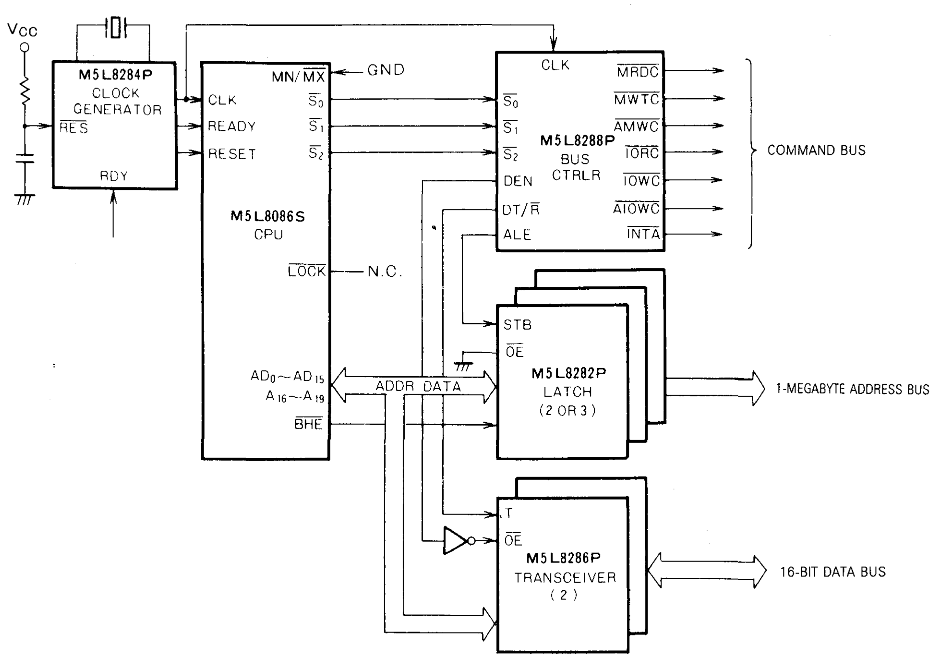
Note 3.



TIMING DIAGRAM (Reference voltage = 1.5V)



APPLICATION EXAMPLE



M5L8288P

BUS CONTROLLER FOR 8086, 8088, 8089 PROCESSORS

DESCRIPTION

The M5L8288P is a semiconductor integrated circuit consisting of a bus controller and bus driver for the 8086 and 8088, 16-bit microprocessors. By using the status signals from the CPU a Multibus (Intel trademark) control signal is generated.

FEATURES

- High-fanout outputs
  Command output I_{OL}=32mA, I_{OH}=–5mA
  Control output I_{OL}=16mA, I_{OH}=–1mA
- Advanced command outputs ($\overline{AIOWC}$ and $\overline{AMWC}$ outputs)
- Pin and electrical compatibility with the Intel 8288

APPLICATION

Bus controller and bus driver for maximum mode operation of the 8086 and 8088

FUNCTION

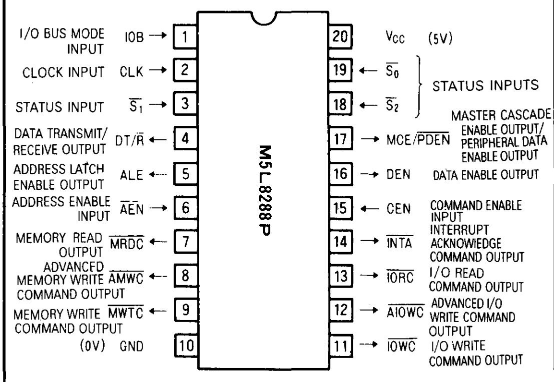
The M5L8288P is a bus controller and driver for maximum mode operation of the 8086 and 8088 processors.

The command signals and control signals are decoded by means of the $\overline{S_0}$~$\overline{S_2}$ outputs from the CPU and the control signals for I/O devices and memory are output.

The device can be used in the Multimaster mode in which several CPUs acting as masters are connected to one data bus. An input pin for the control signal $\overline{AEN}$ from an 8289 bus arbiter is provided.

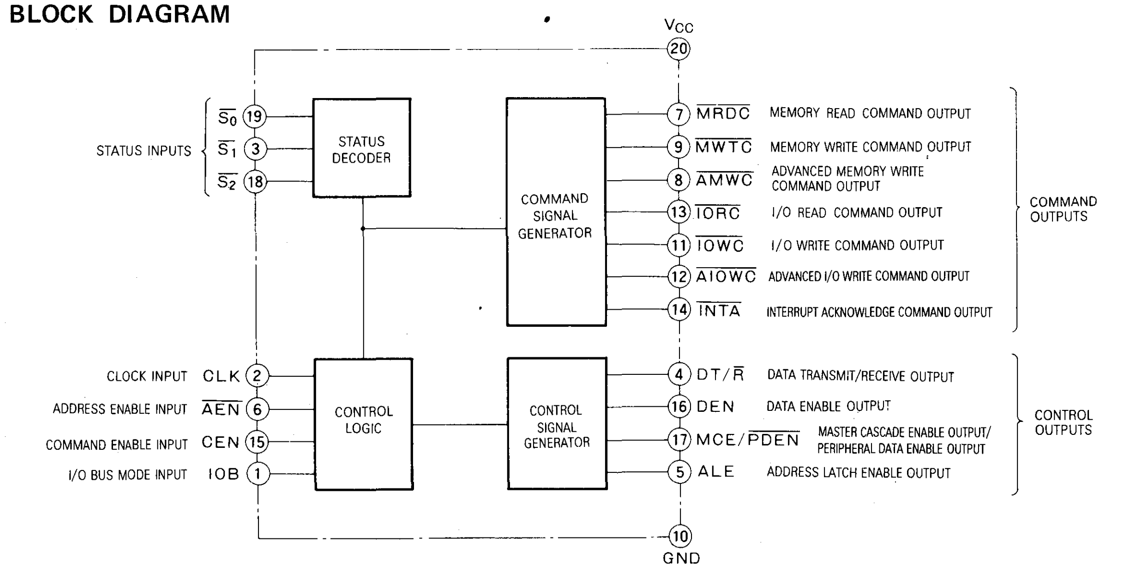
By using the M5L8288P as a bus controller, a high-performance 16-bit microcomputer system can be configured.

PIN CONFIGURATION (TOP VIEW)



Outline 20P4

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

PIN DESCRIPTIONS

Pin	Name	Input or output	Functions
$\overline{S_0}$, $\overline{S_1}$, $\overline{S_2}$	Status input	Input	These are connected to the CPU status output $\overline{S_0}$~$\overline{S_2}$. The M5L8288P uses these signals to generate the proper timing command signals and control signals. All pins are provided with internal pull-up resistors.
CLK	Clock input	Input	Used to connect the clock generator M5L8284P clock output CLK. All outputs of the M5L8288P change in synchronization with the clock input.
ALE	Address latch enable output	Output	Provides the strobe signal output for the address latches This pin is connected to the STB pin of the M5L8282P or M5L8283P and used to latch the address from the CPU. When using any other address latch, the following conditions must be satisfied. 1. The enable input must be active high. 2. Data reading is always performed while the enable input is high. 3. The latching operation is performed as the enable input goes from high to low.
DEN	Data enable	Output	Provides the data enable signal for the local bus or a data transceiver on the system bus. Operates in active high mode.
DT/$\overline{R}$	Data transmit/receive control output	Output	Controls the flow of data between CPU and memory or peripheral I/O devices. When this pin is high, the CPU can write data to the peripheral devices. When it is low, it can read data from the peripheral devices. It is connected to the transmit input T of the M5L8286P or M5L8287P bus transceivers.
$\overline{AEN}$	Address enable input	Input	When the IOB input is low and the $\overline{AEN}$ input is set to high, all command outputs are put in the high-impedance state. When the IOB input is high, there is no effect on the $\overline{IORC}$, $\overline{IOWC}$, $\overline{AIOWC}$, and $\overline{INTA}$ outputs, the command output other than these four going into the high-impedance state. None of the command outputs will go low until at least 115ns after $\overline{AEN}$ transits from high to low.
CEN	Command enable input	Input	When this pin is set to low, all command outputs and DEN are prohibited by the $\overline{PDEN}$ control output (not high-impedance state). When set to high, the above outputs are enabled.
IOB	Input/output bus mode input	Input	When this pin is set to high, the M5L8288P functions in the I/O bus mode, and when set to low it functions in the system bus mode. (The I/O bus mode and system bus mode are described in the functional description)
$\overline{AIOWC}$	Advanced I/O write command output	Output	The $\overline{AIOWC}$ issues an I/O Write Command earlier in the machine cycle to give I/O devices an early indication of a write instruction. Its timing is the same as a read command signal. Active low.
$\overline{IOWC}$	I/O write command output	Output	Instructs an I/O device to read the data on the data bus. Active low.
$\overline{IORC}$	I/O read command output	Output	Instructs an I/O device to drive its data onto the data bus. Active low.
$\overline{AMWC}$	Advanced write command output	Output	The $\overline{AMWC}$ issues a memory write command earlier in the machine cycle to give memory devices an early indication of a write instruction. Its timing is the same as a read command signal. Active low.
$\overline{MWTC}$	Memory write command output	Output	Provides a write instruction to memory for the current data on the bus. Active low.
$\overline{MRDC}$	Memory read command output	Output	Provides an output instruction to memory for the present data on the bus. Active low.
$\overline{INTA}$	Interrupt acknowledge command output	Output	This output informs an interrupting device that it has accepted the interrupt, outputting a vector address output instruction to the data bus. $\overline{IORC}$ operates in the same manner for interrupt cycles. Active low.

Pin	Name	Input or output	Functions
MCE/$\overline{PDEN}$	Master cascade Enable output/ Peripheral data Enable output	Output	This output pin has two functions. 1. When the IOB input is set to low: The MCE function is enabled. The signal acts as the enable signal which allows a slave PIC (M5L8259AP) to read the cascade address output to the bus by the master PIC during an interrupt sequence. Active high. 2. When the IOB input is set to high: The $\overline{PDEN}$ function is enabled. This output provides the enable signal to the data bus transceiver connected to the I/O interface bus when an instruction occurs ($\overline{IORC}$, $\overline{IOWC}$, $\overline{AIOWC}$, $\overline{INTA}$). Operates the same way as DEN with respect to the system bus.

FUNCTIONAL DESCRIPTION

The state of the command outputs and control outputs are determined by the CPU status outputs $\overline{S_0}$~$\overline{S_2}$. The table summarizes the states of the outputs $\overline{S_0}$~$\overline{S_2}$ and their corresponding valid command output names.

STATUS INPUTS AND COMMAND OUTPUTS RELATIONSHIPS

$\overline{S_2}$	$\overline{S_1}$	$\overline{S_0}$	8086, 8088 status	Valid command output name
L	L	L	Interrupt acknowledge	$\overline{INTA}$
L	L	H	Data read from an I/O port	$\overline{IORC}$
L	H	L	Data write to an I/O port	$\overline{IOWC}$, $\overline{AIOWC}$
L	H	H	Halt	—
H	L	L	Instruction fetch	$\overline{MRDC}$
H	L	H	Read data from memory	$\overline{MRDC}$
H	H	L	Write data to memory	$\overline{MWTC}$, $\overline{AMWC}$
H	H	H	Passive state	—

Depending upon whether the M5L8288P is in the I/O bus mode or system bus mode, the command output sequence will vary.

1. I/O bus mode operation

When IOB is high, the M5L8288P function in the I/O bus mode.

In the I/O Bus mode all I/O command lines ($\overline{IORC}$, $\overline{IOWC}$, $\overline{AIOWC}$, $\overline{INTA}$) are always enabled (i.e., not dependent on $\overline{AEN}$). When an I/O command is initiated by the processor, the 8288 immediately activates the command lines using $\overline{PDEN}$ and DT/$\overline{R}$ to control the I/O bus transceiver. The I/O command lines should not be used to control the system bus in this configuration because no arbitration is present. This mode allows one 8288 Bus Controller to handle two external busses. No waiting is involved when the CPU wants to gain access to the I/O bus. Normal memory access requires a "Bus Ready" signal ($\overline{AEN}$ LOW) before it will proceed. It is advantageous to use the IOB mode if I/O or peripherals dedicated to one processor exist in a multi-processor system.

2. System bus mode operation

When IOB is set to low, the M5L8288P enters the system bus mode. In this mode no command is issued until 115 ns after the $\overline{AEN}$ Line is activated (LOW). This mode assumes bus arbitration logic will inform the bus controller (on the $\overline{AEN}$ line) when the bus is free for use. Both memory and I/O commands wait for bus arbitration. This mode is used when only one bus exists. Here, both I/O and memory are shared by more than one processor.

3. $\overline{AMWC}$ and $\overline{AIOWC}$ outputs

With respect to the normal write control signals $\overline{MWTC}$ and $\overline{IOWC}$, the advanced-write command signals $\overline{AMWC}$ and $\overline{AIOWC}$ transit low one clock cycle earlier and remain low for two clock cycles.

These signals are used with peripheral devices or static RAM devices which require a long write pulse, so that the CPU does not go into an unnecessarily wait cycle.

BUS CONTROLLER FOR 8086, 8088, 8089 PROCESSORS

ABSOLUTE MAXIMUM RATINGS (T_a=0~75°C, unless otherwise noted)

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage		−0.5~+7	V
V_I	Input voltage		−0.5~+5.5	V
V_O	Output voltage		−0.5~V_{CC}	V
P_d	Power dissipation		1.5	W
T_{opr}	Operating free-air temperature range		0~75	°C
T_{stg}	Storage temperature range		−65~+150	°C

RECOMMENDED OPERATING CONDITIONS (T_a=0~75°C, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min	Nom	Max	
V_{CC}	Supply voltage		4.5	5	5.5	V
I_{OH}	High-level output current	Command outputs			−5	mA
		Control outputs			−1	
I_{OL}	Low-level output current	Command outputs			32	mA
		Control outputs			16	

ELECTRICAL CHARACTERISTICS (T_a=0~75°C, unless otherwise noted)

Symbol	Parameter		Test conditions		Limits			Unit
					Min	Typ	Max	
V_{IH}	High-level input voltage				2			V
V_{IL}	Low-level input voltage						0.8	V
V_{IC}	Input clamp voltage						−1	V
V_{OH}	High-level output voltage	Command outputs	V_{CC}=4.5V, V_I=2V	I_{OH}=−5mA	2.4			V
		Control outputs	V_I=0.8V	I_{OH}=−1mA	2.4			
V_{OL}	Low-level output voltage	Command outputs	V_{CC}=4.5V, V_I=2V	I_{OL}=32mA			0.5	V
		Control outputs	V_I=0.8V	I_{OL}=16mA			0.5	
I_{IH}	High-level input voltage		V_{CC}=5.5V, V_I=5.5V				50	μA
I_{IL}	Low-level input voltage		V_{CC}=5.5V, V_I=0.45V				−0.7	mA
I_{OZH}	Off-state output current with high-level applied to output		V_{CC}=5.5V, V_O=5.25V				100	μA
I_{OZL}	Off-state output current with low-level applied to output		V_{CC}=5.5V, V_O=0.4V				−100	μA
I_{CC}	Supply current		V_{CC}=5.5V				230	mA

9

BUS CONTROLLER FOR 8086, 8088, 8089 PROCESSORS

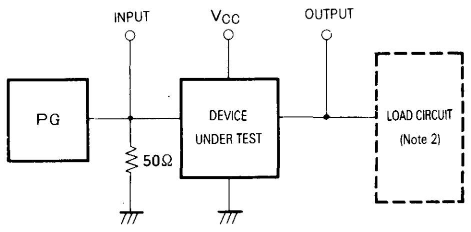
SWITCHING CHARACTERISTICS (V_{CC}=5V±10%, T_a=0~75°C, unless otherwise noted)

Symbol	Parameter	Alternate symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
t_{PLH}	Output low-level to high-level propagation time From CLK input to DEN output	TCVNV	(Note 1)	5		45	ns
t_{PHL}	Output high-level to low-level propagation time From CLK input to $\overline{PDEN}$ output						
t_{PLH}	Output low-level to high-level propagation time From CLK input to DEN output	TCVNX		10		45	ns
t_{PHL}	Output high-level to low-level propagation time From CLK input to $\overline{PDEN}$ output						
t_{PLH}	Output low-level to high-level propagation time From CLK input to ALE output	TCLLH				20	ns
t_{PLH}	Output low-level to high-level propagation time From CLK input to MCE output	TCLMCH				20	ns
t_{PLH}	Output low-level to high-level propagation time From $\overline{S_0}$~$\overline{S_1}$ inputs to ALE output	TSVLH				20	ns
t_{PLH}	Output low-level to high-level propagation time From $\overline{S_0}$~$\overline{S_1}$ inputs to MCE output	TSVMCH				20	ns
t_{PHL}	Output high-level to low-level propagation time From CLK input to ALE output	TCHLL		4		15	ns
t_{PHL}	Output high-level to low-level propagation time From CLK input to $\overline{MRDC}$, $\overline{IORC}$, $\overline{INTA}$, $\overline{AMWC}$, $\overline{MWTC}$, $\overline{AIOWC}$, and $\overline{IOWC}$ outputs	TCLML		10		35	ns
t_{PLH}	Output low-level to high-level propagation time From CLK input to $\overline{MRDC}$, $\overline{IORC}$, $\overline{INTA}$, $\overline{AMWC}$, $\overline{MWTC}$, $\overline{AIOWC}$, and $\overline{IOWC}$ outputs	TCLMH		10		35	ns
t_{PHL}	Output high-level to low-level propagation time From CLK input to DT/$\overline{R}$ output	TCHDTL				50	ns
t_{PLH}	Output low-level to high-level propagation time From CLK input to DT/$\overline{R}$ output	TCHDTH				30	ns
t_{PZH}	High-level output enable time From $\overline{AEN}$ input to $\overline{MRDC}$, $\overline{IORC}$, $\overline{INTA}$, $\overline{AMWC}$, $\overline{MWTC}$, $\overline{AIOWC}$, and $\overline{IOWC}$ outputs	TAELCH				40	ns
t_{PHZ}	High-level output disable time From $\overline{AEN}$ input to $\overline{MRDC}$, $\overline{IORC}$, $\overline{INTA}$, $\overline{AMWC}$, $\overline{MWTC}$, $\overline{AIOWC}$, and $\overline{IOWC}$ outputs	TAEHCZ				40	ns
t_{PHL}	Output high-level to low-level propagation time From $\overline{AEN}$ input to $\overline{MRDC}$, $\overline{IORC}$, $\overline{INTA}$, $\overline{AMWC}$, $\overline{MWTC}$, $\overline{AIOWC}$, and $\overline{IOWC}$ outputs	TAELCV		115		200	ns
t_{PLH} t_{PHL}	Output low-level to high-level and high-level to low-level propagation time From $\overline{AEN}$ input to DEN output	TAEVNV				20	ns
t_{PLH} t_{PHL}	Output low-level to high-level and high-level to low-level propagation time From CEN input to DEN and PDEN outputs	TCEVNV				25	ns
t_{PLH} t_{PHL}	Output low-level to high-level and high-level to low-level propagation time From CEN input to $\overline{MRDC}$, $\overline{IORC}$, $\overline{INTA}$, $\overline{AMWC}$, $\overline{MWTC}$, $\overline{AIOWC}$, and $\overline{IOWC}$ outputs	TCELRH		10		35	ns

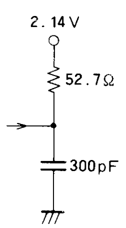
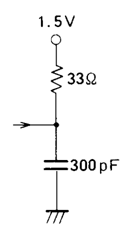
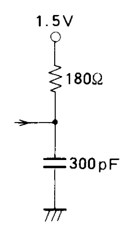
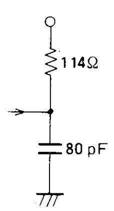
TIMING REQUIREMENTS (V_{CC}=5V±10%, T_a=0~75°C, unless otherwise noted)

Symbol	Parameter	Alternate symbol	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
t_C	Clock CLK cycle time	TCLCL		100			ns
$t_{W(CLKL)}$	Clock CLK low pulse width	TCLCH		50			ns
$t_{W(CLKH)}$	Clock CLK high pulse width	TCHCL		30			ns
$t_{SU(\overline{S_0}-\overline{S_2})}$	$\overline{S_0}$~$\overline{S_2}$ setup time with respect to T for the T_1 state	TSVCH		35			ns
$t_{h(\overline{S_0}-\overline{S_2})}$	$\overline{S_0}$~$\overline{S_2}$ hold time with respect to T for the T_4 state	TCHSV		10			ns
$t_{SU(\overline{S_0}-\overline{S_2})}$	$\overline{S_0}$~$\overline{S_2}$ setup time with respect to T for the T_3 state	TSHCL		35			ns
$t_{h(\overline{S_0}-\overline{S_2})}$	$\overline{S_0}$~$\overline{S_2}$ hold time with respect to T for the T_3 state	TCLSH		10			ns
t_r	Input risetime	TILIH				20	ns
t_f	Input falltime	TIHIL				12	ns

Note 1. Test Circuit



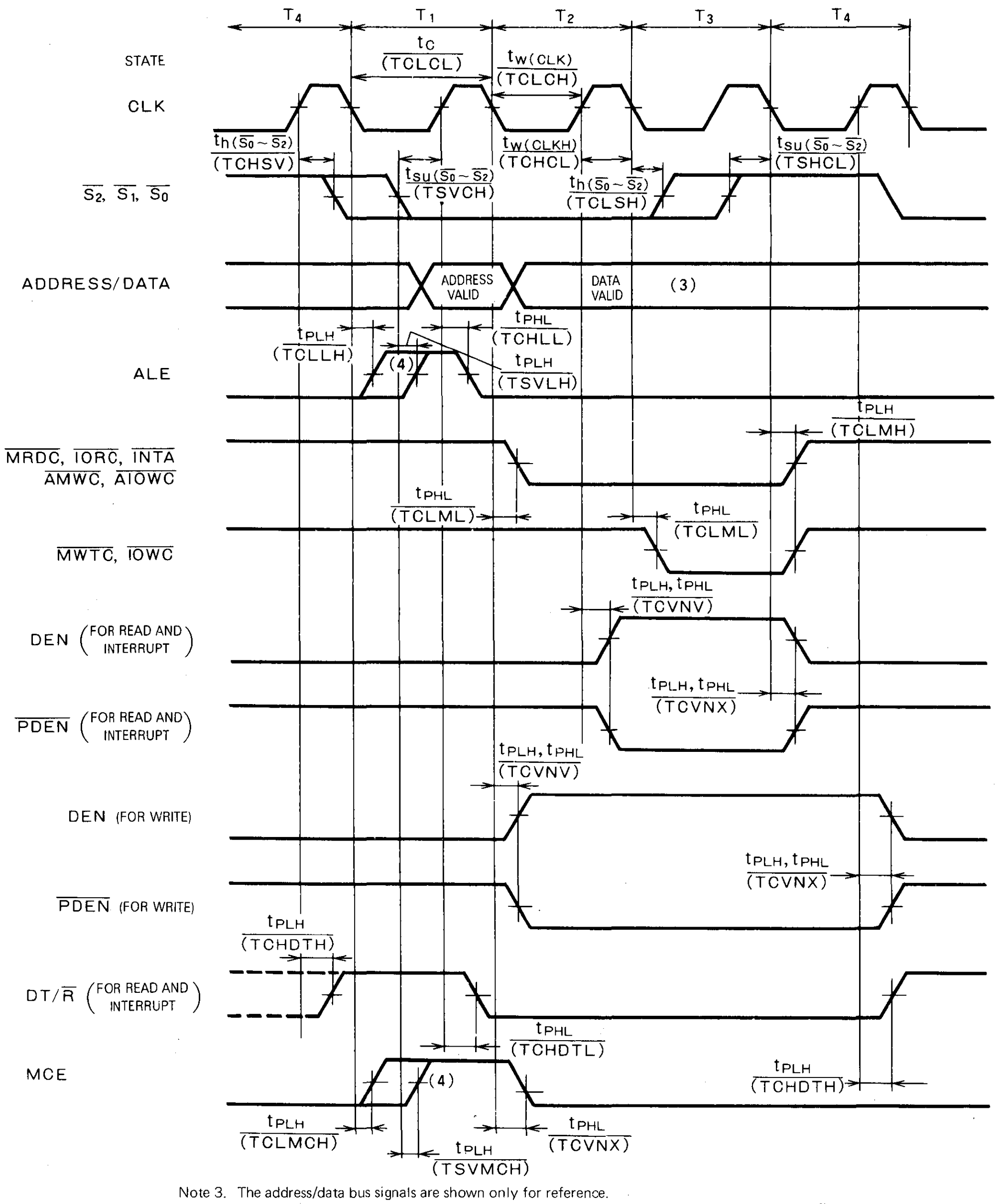
Note 2.

Load circuit	t_{PLH}, t_{PHL}	t_{PLZ}, t_{PZL}	t_{PHZ}, t_{PZH}
Command output load circuit			
Control output load circuit		—	—

9

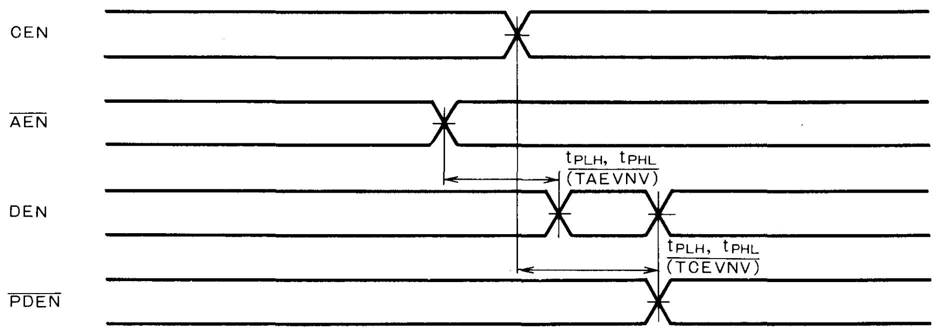
TIMING DIAGRAM

1. Command output timing

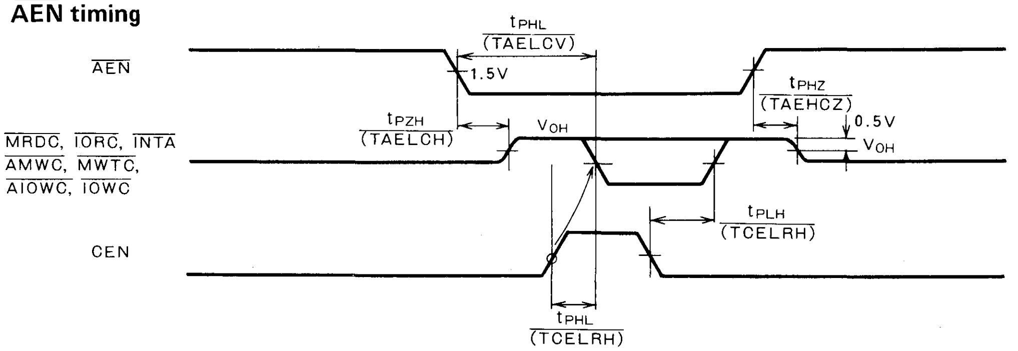


Note 3. The address/data bus signals are shown only for reference.
4. The ALE and MCE leading edge occurs in synchronization with the falling edge of CLK or $\overline{S_0}$~$\overline{S_2}$, whichever is later.
5. Unless otherwise noted, the timing of all signals is respect to 1.5V.

2. DEN and $\overline{\text{PDEN}}$ timing

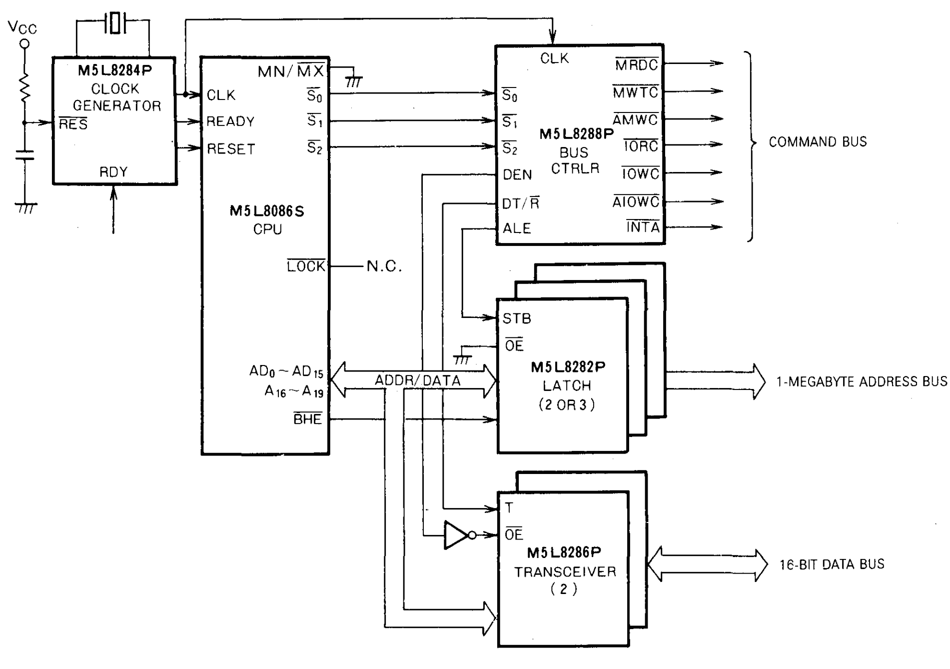


3. $\overline{\text{AEN}}$ timing



Note 6. CEN must be low or valid prior to T_2 to prevent the command from being generated.

APPLICATION EXAMPLE



SPEECH SYNTHESIS LSIs (PARCOR SYSTEM)

10

DESCRIPTION

The M58817AP is a p-channel MOS speech synthesizer making use of the LPC (PARCOR) method.

By using the device with one type M58818-XXXP device, approximately 100 seconds (maximum) of speech output can be achieved.

The M58819S EPROM is also available for use with the M58817AP.

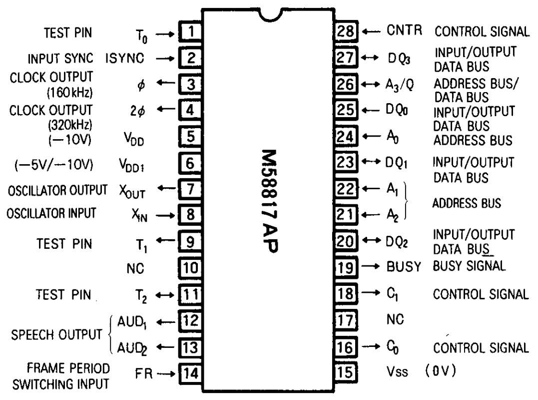
FEATURES

- Single −10V power supply
- May be used with 5V microcomputers by use of a dual −5V/−10V supply
- Selectable characteristic parameter compression density
  Low audio quality: .... 1.96K-bit/s (max)
  High audio quality: .... 3.92K-bit/s (max)
- Male and female voices or sound effects mixed
- Usable with up 16 phrase ROMs
- Direct speaker drive is possible
- By use of just one masked ROM up to 100 seconds of speech output is possible

APPLICATIONS

Clocks, educational equipment, toys, electronic cash registers.

PIN CONFIGURATION (TOP VIEW)

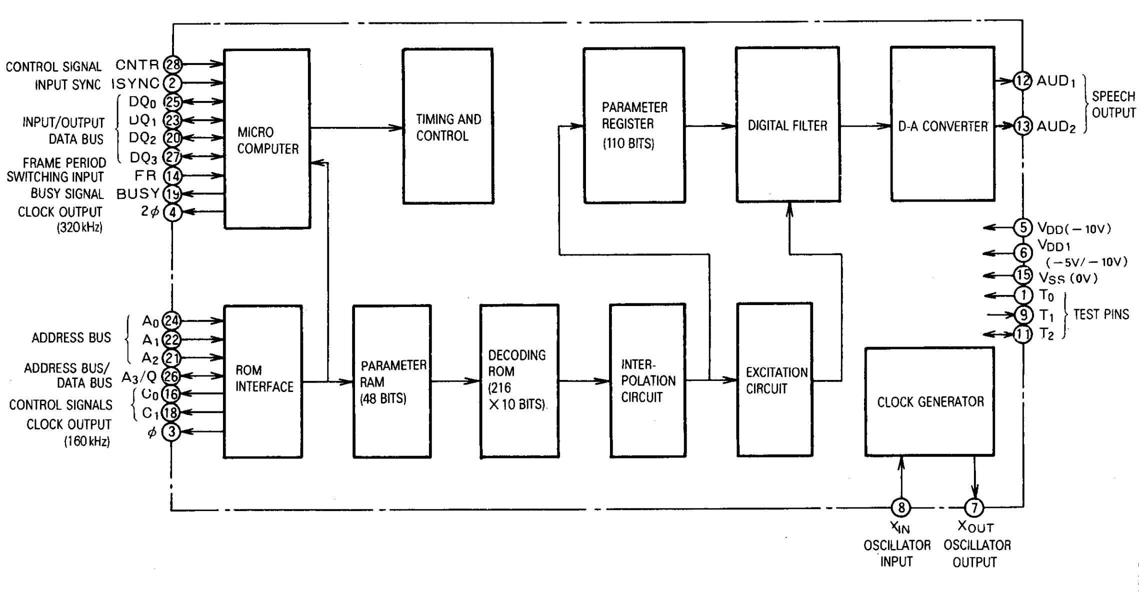


Outline 28P4 NC : NO CONNECTION

FUNCTION

The M58817AP is an LPC (PARCOR) speech synthesizer consisting of a microcomputer interface, parameter storage RAM, decoding ROM, interpolation logic, parameter register, excitation circuit, ROM interface, lattice-type digital filter (pipeline multiplier, adder, stack and registers, etc.), D-A converter, timing logic circuitry, and clock oscillator.

BLOCK DIAGRAM



BASIC FUNCTION BLOCKS

Function	Operational description
Microcomputer interface	Signal exchange with an external controller
Timing logic	Internal timing control
ROM interface	Data exchange with ROM
Parameter storage RAM (48 bits)	Temporary storage of one frame of voice parameters (K-parameter, pitch and amplitude)
Decoding ROM (216 x 10 bits)	Decoding of non-linearly coded parameters stored in RAM
Interpolation logic	Linear interpolation of K-parameters, pitch and amplitude every 3.125ms
Excitation circuit	Consists of a white noise and pulse generator used to generate voiced and unvoiced sounds
Parameter register (110 bits)	A register used to temporarily store linearly interpolated data
Digital lattice filter	A 14-bit 10-stage lattice filter used to control the spectral shape, producing linearly approximated data.
D-A converter	8-bit (including sign) D-A converter
Clock generator	Generates a clock by means of an externally connected ceramic element

FUNCTIONAL DESCRIPTION

The M58817AP can be controlled by an external system by means of eight instructions.

1. Addressing instruction: Used to set phrase ROM addresses
2. Indirect addressing instruction: Used to indirectly set phrase ROM addresses
3. Bit read instruction: Used to shift into a 4-bit shift buffer 1-bit of contents from the phrase ROM
4. Data transmission instruction: Outputs the contents of the 4-bit shift buffer
5. Test instruction: Test whether speech generation has been completed
6. Male speaker instruction: Start instruction for voice generation (male)
7. Female speaker instruction: Start instruction for voice generation (female)
8. Stop instruction: Stop the speech generating

Operation begins with the setting of the phrase ROM address counter to the address specified by an addressing instruction or indirect addressing instruction.

Next, upon generation of a male speech or female speech instruction from the controller, the synthesizer enters the speech start-mode and accesses the phrase ROM every 25ms to receive one frame of voice characteristic parameters. In response to demands from the synthesizer, parameters are sent to the synthesizer in bit-serial form. For this transmission, the phrase ROM address counter is automatically incremented with the 8-bit ROM words converted to serial format.

The transmitted parameters are then expanded and interpolated by the synthesizer, whereupon PARCOR voice generation is performed with a cycle time of 125μs (8kHz rate), and D-A conversion to analog speech is performed to generate the output speech signal.

At the last frame of speech parameters stored in the phrase ROM, an end-code is written to signify the end of the parameter stream. When this code is detected, speech generation is halted.

To indicate whether speech generation is in progress, a test instruction may be generated or the synthesizer busy signal may be used (high for speech generation).

When a bit read instruction is generated, the 1-bit contents of the address specified by the phrase ROM address counter is sent to the synthesizer's 4-bit shift

PIN DISCRIPTION

Pin	Name	Input or output	Function
ISYNC	Input sync	Input	Use as a sync signal for commands and data from an external controller
ϕ	Clock output	Output	160kHz clock output
2ϕ	Clock output	Output	320kHz clock output
X_{IN}	Oscillator input	Input	Used to set the frequency of the internal clock generator by means of an external RC circuit or IF-type ceramic filter connected between X-out and this pin.
X_{OUT}	Oscillator output	Output	Output of the internal clock generator (640kHz)
$T_0 \sim T_2$	Test pin	Input/output	Test pin
AUD_1 AUD_2	Speech output	Output	Speech output
FR	Frame period switching signal	Input	Used to set the frame length, 25ms when open and 12.5ms when grounded
C_0 C_1	Control signals	Output	Control of external ROM
BUSY	Busy signal	Output	Used to verify the presence of voice output, high during voice output
$DQ_0 \sim DQ_3$	Input data bus	Input/output	Two-way bus used to manage commands and data from an external controller
$A_0 \sim A_2$	Address bus	Output	Used for external memory addresses
A_3/Q	Address bus/data bus	Input/output	Used to accept addresses and data from external memory
CNTR	Control signal	Input	Used for external control. When high, the external controller effects a command using the sync signal and the signals DQ_0 through DQ_3.

buffer, shifting in serial fashion. The address counter is automatically incremented.

The data transmission instruction causes the synthesizer's 4-bit shift buffer contents to be transmitted in parallel. Thus, by using the bit read and data transmission instructions any arbitrary address contents from the phrase ROM may be read.

The STOP instruction can be used to halt speech generation.

TABLE 1. INSTRUCTION CODES

Instruction	Data bus line DQ				Transmission direction	I_{DB}
	DQ3	DQ2	DQ1	DQ0	Controller C ⋮ Synthesizer S	
Stop	0	0	0	—	C ⟶ S	CIR0
Address setting	0	0	1	—		CIR1
Data transmission	0	1	0	—		CIR2
Female speaker	0	1	1	—		CIR3
Bit read	1	0	0	—		CIR4
Male speaker	1	0	1	—		CIR5
Indirect address	1	1	0	—		CIR6
Test	1	1	1	—		CIR7

Note: 1 & 0 refer to high- and low-level signals
"—" indicates a non-defined level I_{DB}
refers to the data bus lines

Addressing is performed by sending address setting instructions to the data bus line DQ, and as shown in Table 2, is achieved in the order from lower bits to higher bits in groups of 4 bits. After the address has been set, for direct addressing the bit read instruction is used while for indirect addressing an indirect address instruction is sent to the data bus line.

TABLE 2. ADDRESS SETTING

Data bus line DQ				Transmission direction	I_{DB}
DQ3	DQ2	DQ1	DQ0	Controller C ⋮ Synthesizer S	
a_3	a_2	a_1	a_0	C ⟶ S	CAD0
a_7	a_6	a_5	a_4		CAD1
a_{11}	a_{10}	a_9	a_8		CAD2
a_{15}	a_{14}	a_{13}	a_{12}		CAD3
—	—	a_{17}	a_{16}		CAD4

Note that a_0 and a_{17} refer to the least significant and most significant bits respectively

When data transmission instructions are sent, the data from the synthesizer's 4-bit shift buffer is sent to the external controller.

TABLE 3. DATA TRANSMISSION

Data bus line DQ				Transmission direction	I_{DB}
DQ3	DQ2	DQ1	DQ0	Controller C ⋮ Synthesizer S	SRD
S_3	S_2	S_1	S_0	C ⟵ S	

S_0: Least significant bit
S_3: Most significant bit

When a test instruction is generated, a signal is output via the external controller data bus line DQ_0.

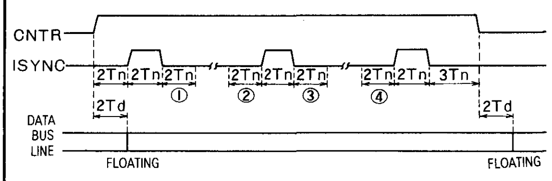
TABLE 4. TEST INSTRUCTION

Data bus line DQ				Transmission direction	I_{DB}
DQ3	DQ2	DQ1	DQ0	Controller C ⋮ Synthesizer S	STK
—	—	—	(SM)	C ⟵ S	

SM = 0: Non-speech generating mode
1: Speech generation
-: Non-defined state

INSTRUCTION EXECUTION TIME

It is necessary to satisfy the timing conditions for all eight instructions as shown in Fig. 1.



Note 1. T_n : Period within which it must be greater than T_ϕ
T_d : Period within which it must be smaller than T_ϕ
T_ϕ : Clock phase (160kHz), i.e. 6.25μs
2. There are cases when 1 & 2 or 3 & 4 may coincide.

When the CNTR signal is set to high, using the ISYNC signal and the data bus line, synthesizer instructions may be specified to control data transmission and reception. When the CNTR signal is made low, the synthesizer cannot be controlled by the controller, the data bus line being floated.

Fig. 2~9 show the bus line timing relationships and execution times for the eight types of instructions.

In the diagrams codes surrounded by a rectangle represent instructions and the times in which they are generated.

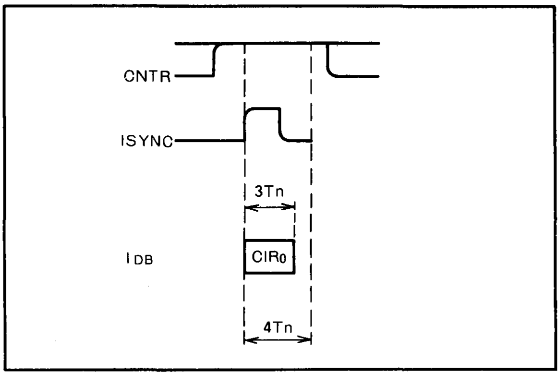


Fig. 2 Stop instruction

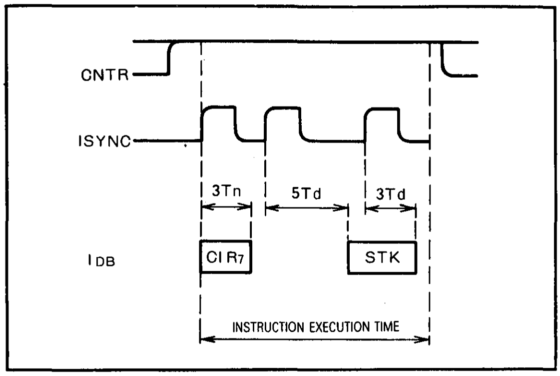


Fig. 3 Test instruction

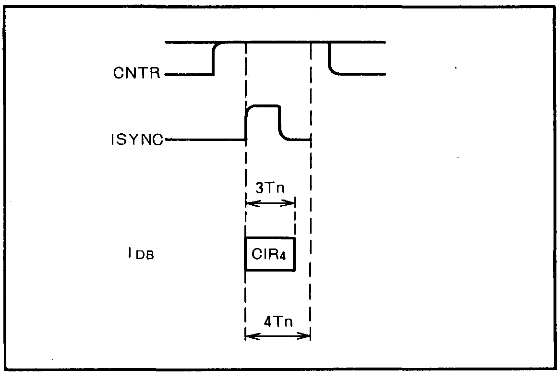


Fig. 4 Bit read instruction

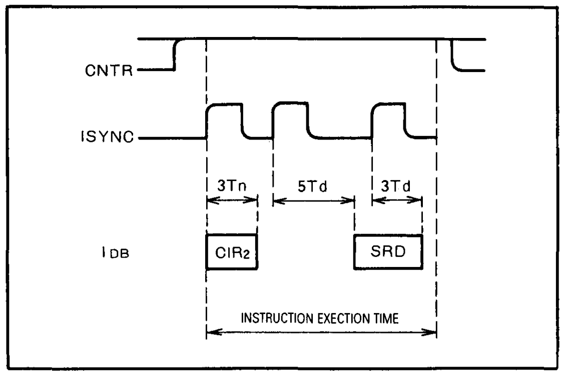


Fig. 5 Data transmission instruction

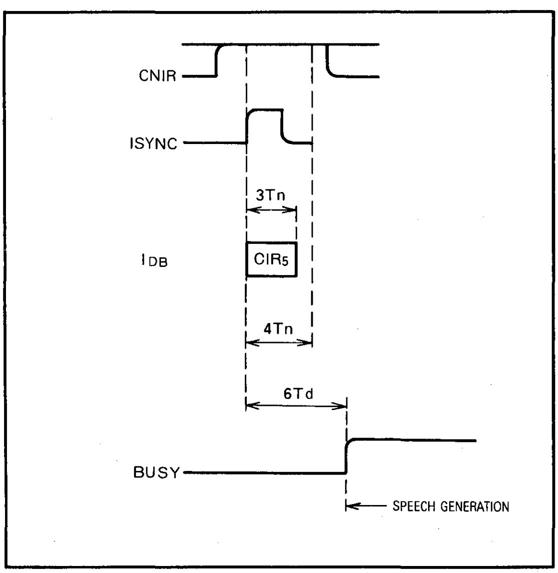


Fig. 6 Male speaker instruction

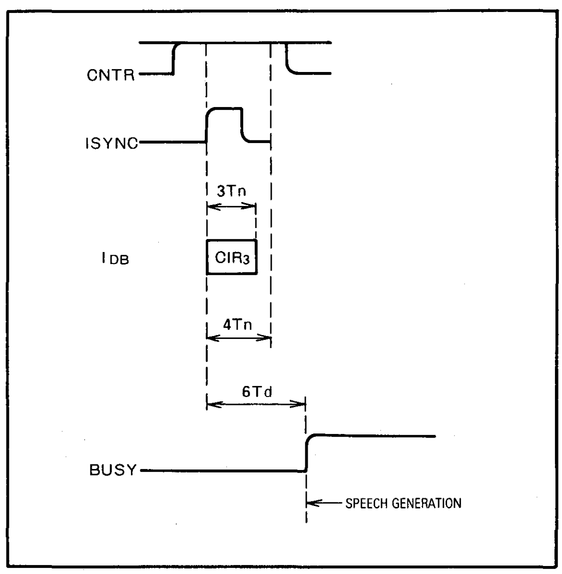


Fig. 7 Female speaker instruction

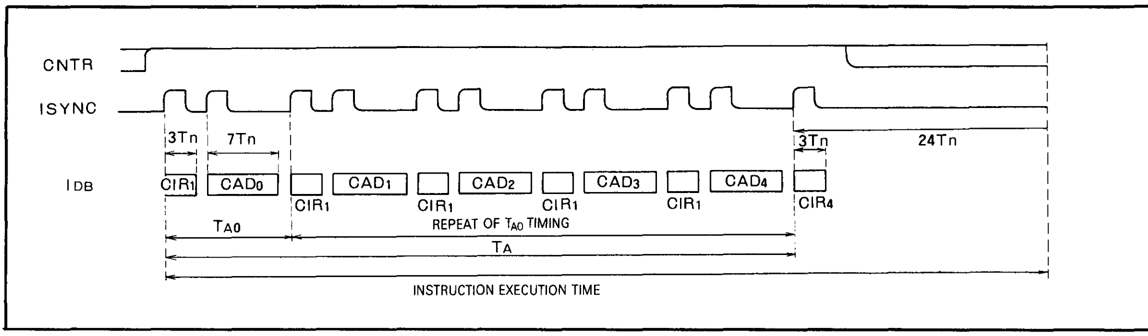


Fig. 8 Direct addressing instruction

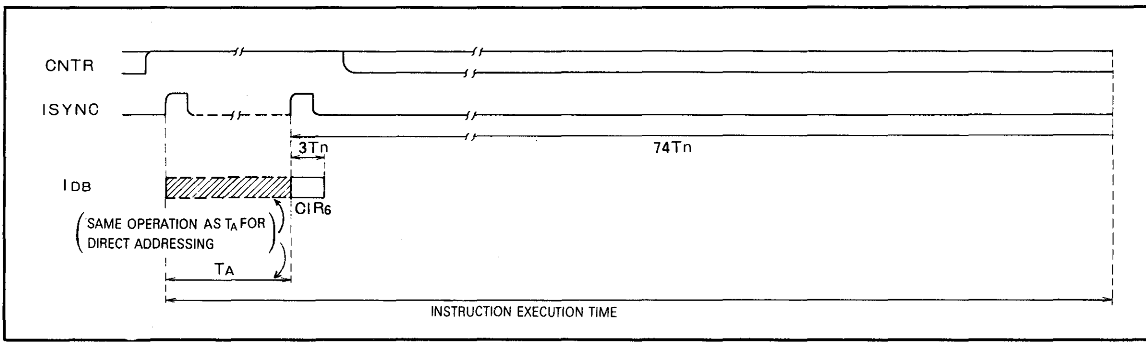


Fig. 9 Indirect addressing instruction

INSTRUCTION SET
DEFINITION OF SYMBOLS

DQ: Interface data bus line
DQ_i (i = 0~3): i-th bit of DQ
a: 18-bit data used for addressing
a_k (k = 0~17): k-th bit of a
A: 18-bit phrase ROM address
A_k (k = 0~17): k-th bit of A
S: 4-bit shift buffer
S_i (i = 0~3): i-th bit of S
R (A): Contents of ROM at the address specified at A
R_j (A): (j = 0~7): Value of the i-th bit of R (A)

USING THE INSTRUCTION SET
1. Direct and Indirect Addressing Instruction
(1) Direct addressing

Instruction code:

DQ_3	DQ_2	DQ_1	DQ_0
0	0	1	—
a_3	a_2	a_1	a_0
0	0	1	—
a_7	a_6	a_5	a_4
0	0	1	—
a_{11}	a_{10}	a_9	a_8
0	0	1	—
a_{15}	a_{14}	a_{13}	a_{12}
0	0	1	—
—	—	a_{17}	a_{16}
1	0	0	—

① ② ③ ④

As shown above data is input to the synthesizer from top to bottom in sequence from the external controller.

10

Function equations: $(A) \xleftarrow{③} a$

Instruction Description

This instruction directly sets the phrase ROM address.

By using this instruction, the 1st bit, i.e. R_0 (A) = R_0 (a) of the 8-bit contents of the phrase ROM at the specified address is specified. Note that the upper order bits $a_{14} \sim a_{17}$ are specified by the phrase ROM chip select data.

When the contents of the address counter previous to this instruction are known and it is not necessary to change the upper order bits $a_{14} \sim a_{17}$, the appropriate setting step from 1 to 4 may be eliminated.

After outputting of data a_{16} and a_{17}, this instruction requires a minimum time of $24T_\phi$ during which bit read instructions must be output as dummies. T_ϕ is the clock period.

1. Indirect Address

Instruction Code:

DQ3	DQ2	DQ1	DQ0
0	0	1	—
a_3	a_2	a_1	a_0
0	0	1	—
a_7	a_6	a_5	a_4
0	0	1	—
a_{11}	a_{10}	a_9	a_8
0	0	1	—
a_{15}	a_{14}	a_{13}	a_{12}
0	0	1	—
—	—	a_{17}	a_{16}
1	1	0	—

① ② ③ ④

As shown above data is input to the synthesizer starting from the top sequentially.

Function equations:

First, (A) ← a, where (A_e) = a

Next, (A_j) ← R_j (a) ≡ R_j ((A_E)) where j = 0~7

(A_8 + j) ← R_j (a + 1) ≡ R_j ((A_e) + 1) where j = 0~5

(A_j) = a_j where j = 14~17

Note that the above excludes the case for which $a_0 \sim a_{13}$ of (A_e) are all ones.

Instruction Description

This instruction indirectly specifies the phrase ROM address.

By using this instruction, the first bit, i.e. R_0 ((A)) = R_0 (a), of the 8-bit phrase ROM contents at the specified address is specified.

First, this instruction sets the address counter A to the address data a. This address is used as (A_e). Next, the ROM output R_0 ((A_e)) ~ R_7 ((A_e)) at the specified address (A_e) is sent to the address counter $A_0 \sim A_7$, and the ROM output R_0 ((A_e)+ 1) ~ R_5 ((A_e)+ 1) at the next address from a(A_e), (A_e) + 1 is sent to the address counter $A_8 \sim A_{13}$. The upper-order 4-bits $A_{14} \sim A_{17}$ of (A_e) do not change, however. The value of the set address data a must not be the individual phrase ROM last address.

If the contents of the address counter prior to this instruction are known and there is no necessity to change higher order address bits, the appropriate step 1 through 4 may be eliminated.

After outputting the data 4_{16} and a_{17}, this instruction requires a minimum of $74T_\phi$.

2. Bit Read Instruction

Instruction code:

DQ3	DQ2	DQ1	DQ0
1	0	0	—

Function equations:

First (S_0 ← (S_1)

(S_1) ← (S_2)

(S_2) ← (S_3)

(S_3) ← R_j ((A)) where j = 0~7 and j = 0 immediately after the instruction.

Next, j ← j + 1 where j ← 0 if j = 8

(A) ← (A) + 1

Instruction Description

This instruction shifts into the 4-bit shift buffer S the contents of the phrase ROM. In addition, it acts to initialize the ROM after direct addressing.

By using this instruction, the 1st bit of the 8-bit contents of ROM specified by the address counter A is sent to the 4-bit shift buffer S. Immediately after execution, R_0 ((A)) is sent, and when this instruction is executed continuously, the address counter is automatically incremented in the sequence R_0 ((A)), R_1 ((A)) ... R_7 ((A)), R_0 ((A) + 1), thereby performing a serial conversion on the ROM data automatically. In addition, the address counter A is incremented by bits allowing all 16 chips to be addressed.

3. Data Transmission Instruction

Instruction code:

DQ3	DQ2	DQ1	DQ0
0	1	0	—

Function equations:
$(DQ_0) \leftarrow (S_0)$
$(DQ_1) \leftarrow (S_1)$
$(DQ_2) \leftarrow (S_2)$
$(DQ_3) \leftarrow (S_3)$

Instruction Description

This instruction outputs to the interface data bus line DQ the 4-bit contents of the shift buffer S.

By using this instruction, the 4-bit data contents of the phrase ROM can be read externally.

4. Test Instruction

Instruction code:

DQ3	DQ2	DQ1	DQ0
1	1	1	—

Function equations: $(DQ_0) \leftarrow (SM)$

SM=1 for speech generation and 0 for non-generation speech, with $DQ_1 \sim DQ_3$ non-defined.

Instruction Description

This instruction is used to test whether speech generation is being performed and output the result at DQ_0.

Use of this instruction allows the speech mode (SM) status to be output as a DC level at the busy pin as well. SM and BUSY are of the same polarity, 1 for periods of speech generation and 0 for non-generation periods.

5. Speech Instruction

Instruction code:

DQ3	DQ2	DQ1	DQ0	
1	0	1	—	Male speaker
0	1	1	—	Female speaker

Function equations:

(1) Speech synthesizer $\leftarrow R_j$ ((A))
j = 0~7 and j = 0 directly after an addressing instruction

(2) $j \leftarrow j + 1$, however $j \leftarrow 0$ if $j = 8$
$(A) \leftarrow (A) + 1$
(1) and (2) are repeated.

Instruction Description

These instructions are used to begin and execute speech generation by the speech generating equipment.

Using these instructions, speech characteristic parameters stored in the phrase ROM are sent to the synthesizer continuously at a rate of one frame every 25ms, the synthesizer processing them with a cycle time of 125μs. The results of this process are D/A converted every 125μs to create the AUD1 and AUD2 speech outputs. At each of AUD1 and AUD2 pins, an analog speech signal is output corresponding to the appropriate sign codes, one being positive and the other negative.

Use of these instructions automatically increments the 18-bit phrase ROM address counter A, and performs a serialization of ROM data.

As the last voice parameter frame in the phrase ROM, an end code is used so that the synthesizer, sending this end code is reset, ending the speech synthesis process cycle and setting the speech mode SM to 0.

Thus, with the exception of phrase ROM memory capacity, speech of any arbitrary length may be generated.

6. Stop Instruction

Instruction code:

DQ3	DQ2	DQ1	DQ0
0	0	0	—

INSTRUCTION FUNCTION

This instruction stops the speech output of the synthesizer.

Instruction Description

This instruction stops the speech generated by the synthesizer.

By using this instruction, speech generation is halted, and the speech mode SM is reset to 0.

The phrase ROM program counter is not modified.

Upon power up, it is always required to generate a stop instruction.

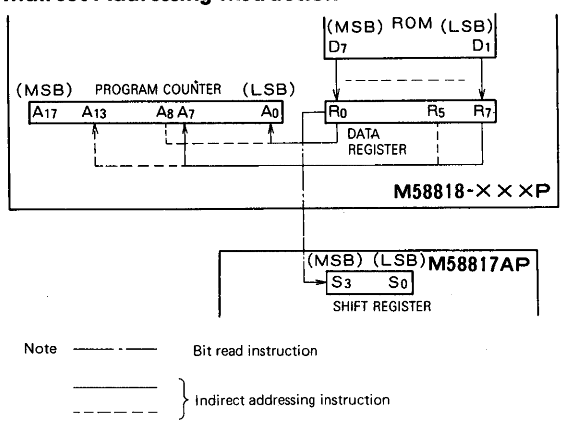
Example

This example generates female speech by sending the ROM program counter to the first address $0013F_{16}$.

DQ3	DQ2	DQ1	DQ0	
0	0	1	0	Addressing instruction
1	1	1	1	F
0	0	1	0	Addressing instruction
0	0	1	1	3
0	0	1	0	Addressing instruction
0	0	0	1	1
0	0	1	0	Addressing instruction
0	0	0	0	0
0	0	1	0	Addressing instruction
0	0	0	0	0
1	0	0	0	Bit read instruction
0	1	1	0	Female speaker instruction

Data is transferred to the synthesizer (M58817AP) from the controller as shown above. Refer to the Timing Diagram for input timing.

Data Transfer Using the Bit Read Instruction and Indirect Addressing Instruction



(1) Using the bit read instruction

If a bit read instruction is output by the synthesizer, the upper order bit of the address specified by the program counter is transferred to the upper order bit (DQ_3) of the synthesizer register. By repetitively sending this instruction, the previously sent bits are shifted towards the lower order bits in the register. Thus, to observe data stored in ROM using the bit read instruction, the data must be stored after doing bit conversion in groups of 4 bits.

(2) Using the indirect addressing instruction

As can be seen in the Figure, the upper order and lower order bits for the contents of ROM and those of the program counter are reversed, requiring care when performing ROM storage operations.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	0.3∼−15	V
V_I	Input voltage		0.3∼−15	V
Pd	Power dissipation		700	mW
Topr	Operating temperature		−10∼70	℃
Tstg	Storage temperature		−40∼125	℃

RECOMMENDED OPERATING CONDITIONS (T_a = -10∼70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	−11	−10	−9	V
V_{DD1}	Supply voltage (Note 1)	−11		−4.75	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage	−1		0	V
$V_{IH(\phi)}$	High-level clock input voltage	−1		0	V
V_{IL}	Low-level input voltage	V_{DD}		−4	V
$V_{IL(\phi)}$	Low-level clock input voltage	V_{DD}		$V_{DD}+2$	V
$f(\phi)$	Oscillation frequency	620		660	kHz

Note 1. V_{DD1} is the controller and interface power supply. For single power supply operation V_{DD1} = V_{DD}.

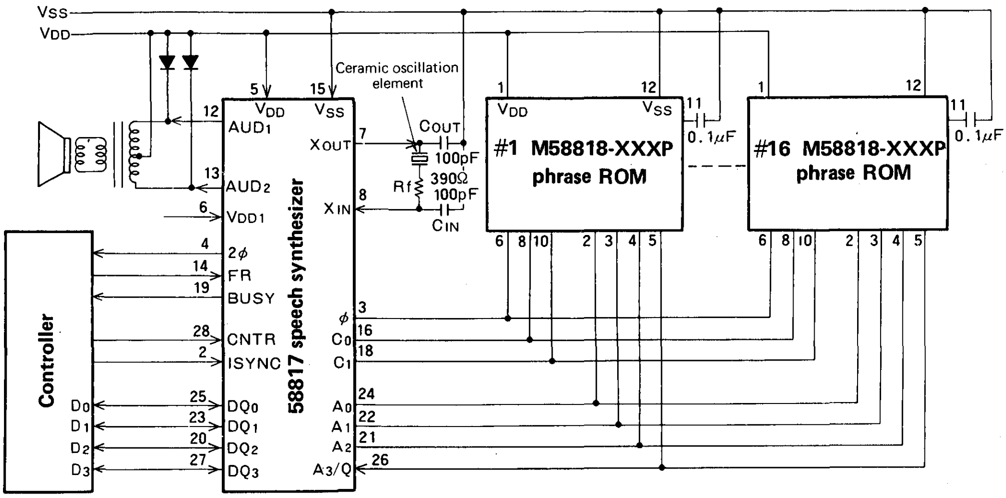
ELECTRICAL CHARACTERISTICS (T_a = -10∼70°C, V_{DD} = -10±1V, f(φ) = 640±20kHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH1}	High-level output voltage for clock output (2φ), busy signal, input/output data bus (V_{DD1}-related outputs)	$V_{DD}=V_{DD1}$, $I_{OH}=-50\mu A$	−1			V
		$V_{DD1}=-5\pm0.25V$, $I_{OH}=-50\mu A$	−1			V
V_{OH2}	High-level output voltage for clock output (φ), control signals, address bus (V_{DD}-related outputs)	$I_{OH}=-50\mu A$	−0.8			V
V_{OL1}	Low-level output voltage for clock output (2φ), busy signal, input/output data bus (V_{DD1}-related outputs)	$V_{DD}=V_{DD1}$, $I_{OL}=50\mu A$			−4.5	V
		$V_{DD1}=-5\pm0.25V$, $I_{OL}=50\mu A$			$V_{DD1}+0.6$	V
V_{OL2}	Low-level output voltage for clock output (φ), control signals, address bus (V_{DD}-related outputs)	$I_{OL}=50\mu A$			−5	V
I_{DD}	Supply current	$V_{DD}=V_{DD1}$, un-loaded output			−30	mA
I_{DD1}	Supply current	Un-loaded output			−1	mA
I_{DA}	Maximum D/A output current	RL = 100Ω			30	mA
Ci	Input capacitance	$V_{DD}=V_{DD1}=V_{SS}$, f=1MHz, 250mVrms		7	10	pF
I_I	Input current, with the exception of the frame period switching input	$V_I=0\sim V_{DD}$			±1	μA
	Input current for the frame period switching input (built-in pull down resistor)	$V_I=0\sim V_{DD}$			−50	μA

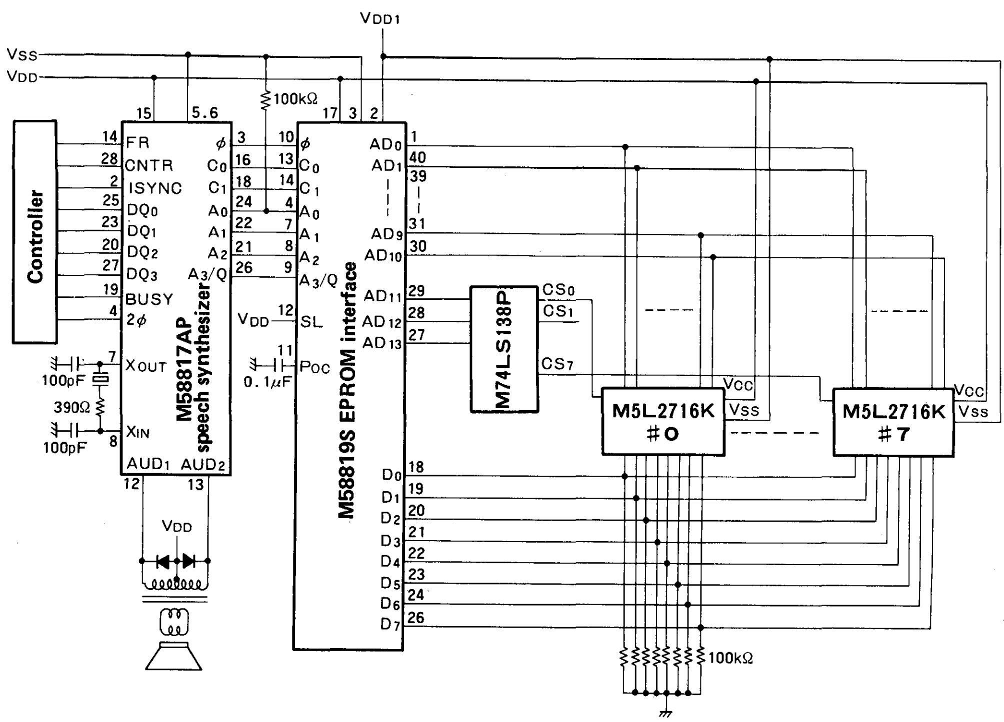
10

APPLICATION EXAMPLES

(1) Use with M58818-XXXP



(2) Use with an EPROM



128K-BIT PHRASE ROM

DESCRIPTION

The M58818-XXXP is a p-channel MOS phrase ROM having a capacity of 16K bytes (131072 bits), intended for use in storage of speech parameters and other data. The device includes also an 18-bit address counter/register, an 8-bit output sense amplifier/buffer, address control circuitry, and read control circuitry.

Used in conjunction with the M58817AP speech synthesizer, up to 100 seconds of speech output can be obtained.

FEATURES

- Single −10V power supply
- Stores parameters for up to 100 seconds of speech output
- Large memory capacity (128K bits)
- Built-in 18-bit address counter/register

APPLICATIONS

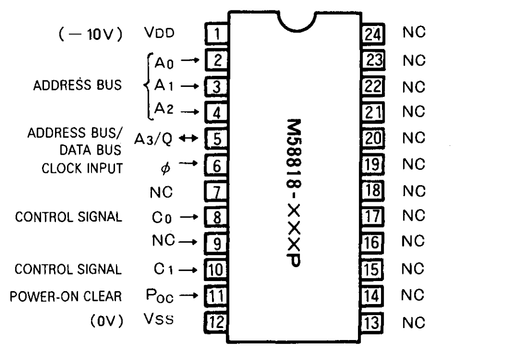
Clocks, educational equipment, toys, electronic cash registers.

FUNCTION

The M58818-XXXP is a 16K x 8-bit masked ROM consisting of a 14-bit address counter, address latch, chip select latch, data register, and control logic circuitry.

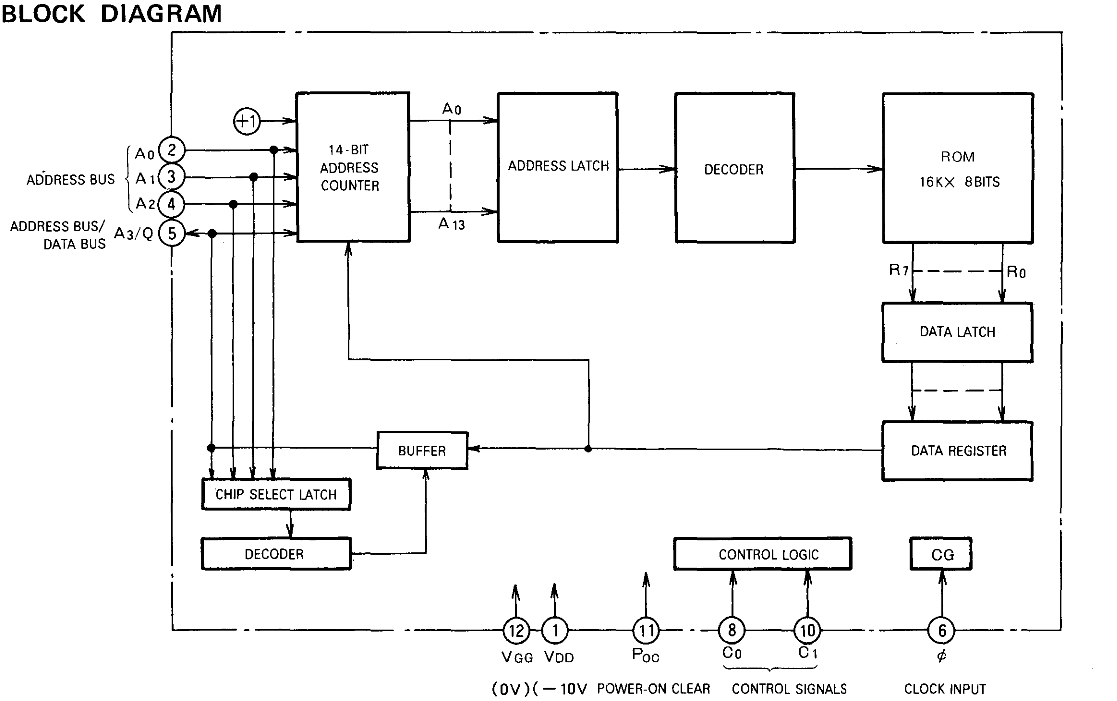
Addressing is done by means of pins A_0 through A_3 in five steps.

PIN CONFIGURATION (TOP VIEW)



Outline 24P1 NC : NO CONNECTION

BLOCK DIAGRAM



BASIC FUNCTIONAL BLOCKS

Function	Operational description
Address counter	For continuous addressing of ROM data for output, this 14-bit pure binary address counter is automatically incremented by 1
Data register	This 8-bit register is used for temporary storage of ROM data
Control logic	Provides internal control by means of the control signals C_0 and C_1
Data memory	16K x 8 bit masked ROM

PIN DESCRIPTION

Pin	Name	Input or output	Function
$A_0 \sim A_2$	Address bus	Input	Input pins for the external input of address into the address counter and chip select latch. This data must be input in five steps.
A_3/Q	Address bus/data bus	Input/ output	In the address and chip select data input mode this pin acts as an input pin to accept address and data. This data must be input in five steps. In the data output mode this pin acts as an output pin, outputting ROM data 1 bit at a time in serial fashion. However, for address overflow and when the chip is not selected, this output pin is floating.
ϕ	Clock input	Input	Clock input pin
C_0 C_1	Control signals	Input	Determine internal operation status. Input pins for control signals.
P_{OC}	Power-on clear	Input	Upon power-on, an internal power-on clear circuit automatically clears the internal status. When this pin is set to high, A_3/Q output is prohibited and the internal states are cleared.

FUNCTIONAL DESCRIPTION

The M58818-XXXP is controlled by the following four instructions.

Instruction	Effect	Instruction code	
		C_0	C_1
Address instruction	Data is loaded into the internal address counter and chip select latch of the M58818-XXXP	0	1
Read instruction	Serially converted data from the ROM is output 1 bit at a time. This may also be used as an addressing instruction capable of directly addressing ROM.	1	0
Indirect addressing instruction	Indirectly sets the address of ROM	1	1
NOP instruction	No operation	0	0

These instructions are described below.

1. Direct addressing

The timing is shown in Fig. 1. First, by using the addressing instruction, A_0, A_1, A_2, and A_3/Q inputs are used to read-in in 4-bit parallel format the starting address and chip select data. Then, five addressing instructions are used to completely set the contents of the address counter and chip select latch.

Next, if a read instruction is input, ROM data is read from the address specified by the sequence above.

The read instruction acts as a direct addressing instruction, and 24 clock cycles after the input of the read instruction ROM data is available at the A_3/Q pin. In addition, by inputting a read instruction ROM data can be serially output 1 bit at a time.

The address counter is automatically incremented by 1 every 8 read instructions.

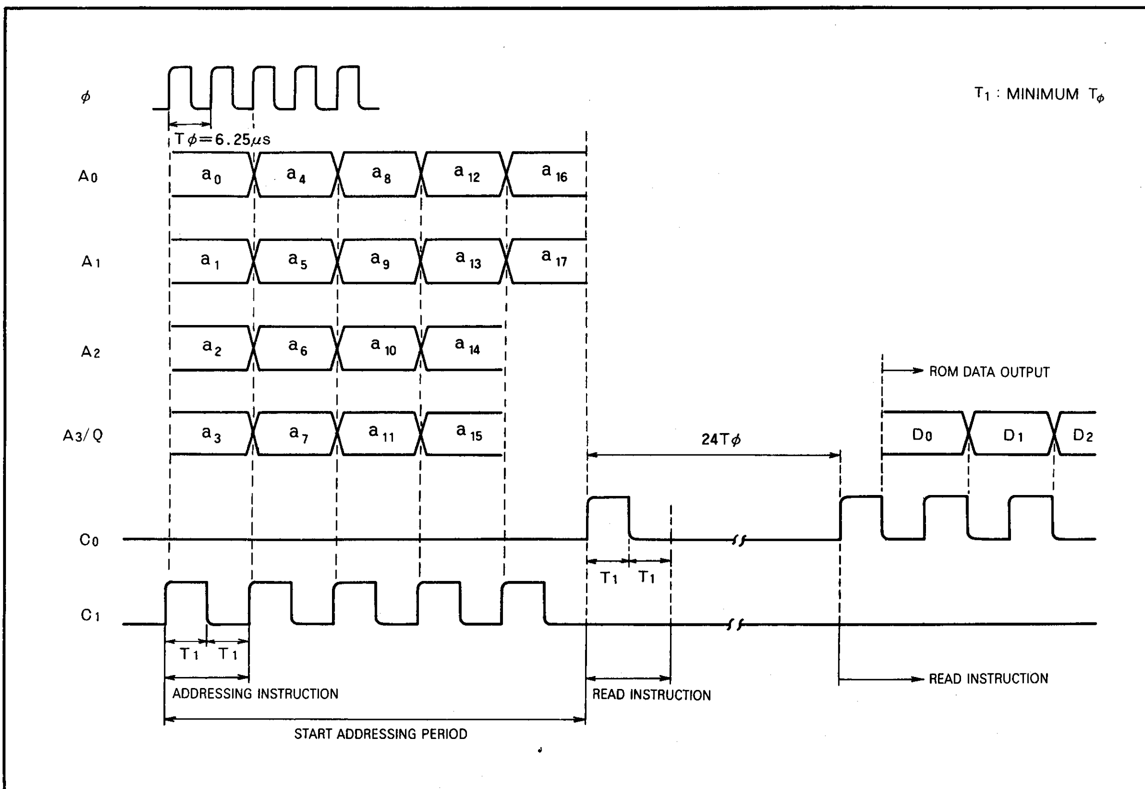


Fig. 1 Direct addressing timing

2. Indirect addressing

The timing is shown in Fig. 2. As with direct addressing, an indirect addressing instruction is input after setting of the address.

By using an indirect addressing instruction, ROM data at the address in the address counter is output and shifted into the address counter.

In addition, when two words of ROM data are shifted into the address counter, 74 clock cycles after the input of an indirect addressing instruction the ROM data at the set address is available as an output at the A_3/Q pin.

By inputting a read instruction, ROM data can be read 1 bit at a time in serial fashion.

The relationship between the two words of ROM data and the address shifted into the address counter is as follows.

If the address counter address is set to (00000_{16}) when an indirect addressing instruction is input, the ROM data at that address $(D_0\ D_1\ D_2\ D_3\ D_4\ D_5\ D_6\ D_7)_2$ and the ROM data at the address incremented 1 (00001_{16}) $(D'_0\ D'_1\ D'_2\ D'_3\ D'_4\ D'_5\ D'_6\ D'_7)_2$ are output and shifted into the address counter, the overall address being set to $(D'_5\ D'_4\ D'_3\ D'_2\ D'_1\ D'_0\ D_7\ D_6\ D_5\ D_4\ D_3\ D_2\ D_1\ D_0)_2$.

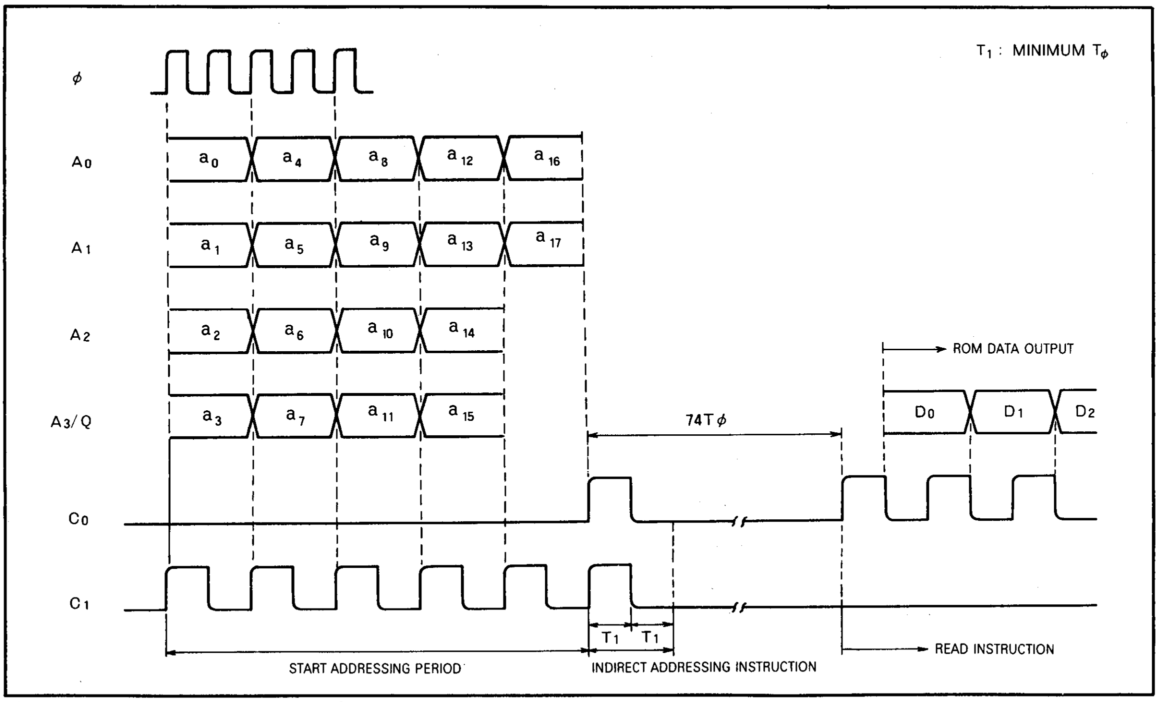


Fig. 2 Indirect addressing timing

3. Relationship of phrase ROM contents and address setting

Up to a maximum of 16 M58818-XXXP phrase ROMs may be connected to a single M58817AP synthesizer. Therefore, since the capacity of a single M58818-XXXP is 16K bytes (131072 bits), it is necessary to be able to address 256K bytes. That is, an 18-bit address a_i (i=0~17) is required the relationship between a_i and the ROM chip being as follows.

$a_{17}\ a_{16}\ a_{15}\ a_{14}\ a_{13}\ a_{12}\ a_{11}\ a_{10}\ a_9\ a_8\ a_7\ a_6\ a_5\ a_4\ a_3\ a_2\ a_1\ a_0$

Chip select (a_{17}–a_{14}) | Addressing bits for 1 chip (a_{13}–a_0)

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	0.3～－15	V
V_I	Input voltage		0.3～－15	V
Pd	Power dissipation	Ta＝25°C	130	mW
Topr	Operating temperature		－10～70	°C
Tstg	Storage temperature		－40～125	°C

RECOMMENDED OPERATING CONDITIONS (T_a=-10~70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	－11	－10	－9	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage	－1.5		0	V
V_{IL}	Low-level input voltage	V_{DD}		－4.5	V
f(φ)	External clock frequency	155	160	165	kHz
C_P	Capacitance connected to P_{OC} pin		0.1		μF

ELECTRICAL CHARACTERISTICS (T_a=-10~70°C, V_{DD}=-10V±1V, V_{SS}=0V, f(φ)=160±5kHz, unless otherwise noted)

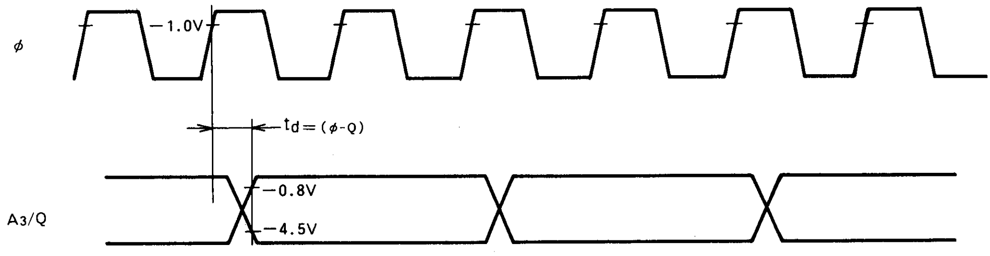
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	I_{OH}＝－0.5mA	－0.8			V
V_{OL}	Low-level output voltage	I_{OL}＝0.5mA			－4.5	V
I_{DD}	Supply current	Un-loaded output			－10	mA
I_I	Input current for all pins except power-on clear	V_I＝0～V_{DD}			±1	μA
I_I	Input current for power-on clear input	V_I＝0～V_{DD}			－50	μA
Ci	Input capacitance	V_{DD}＝Vss, f＝1MHz, 250mVrms		7	10	pF

10

SWITCHING CHARACTERISTICS (T_a=-10~70°C, V_{DD}=-10±1V, $f_{(\phi)}$=160±5kHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{d(\phi-Q)}$	Data output delay time	C_L=100pF			3	μs

TIMING DIAGRAM



DESCRIPTION

The M58819S is a p-channel MOS EPROM interface LSI device capable of addressing 16K bytes of EPROM devices.

The device is housed in a 40-pin ceramic DIL package. Speech output is possible by using the M58819S in conjunction with the M58817AP and EPROMs.

FEATURES

- Single −10V power supply
- Extendable to control up to 16K bytes of EPROM
- Built-in 14-bit address counter

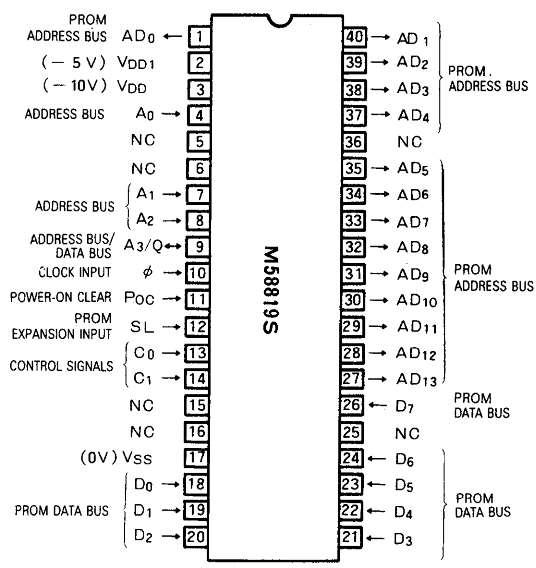
APPLICATION

Small-scale speech output devices.

FUNCTION

The M58819S is an interface LSI which includes a 14-bit address counter, chip select latch, control logic circuitry, and a data register. It is usable in conjunction with EPROMs and the M58817AP speech synthesizer.

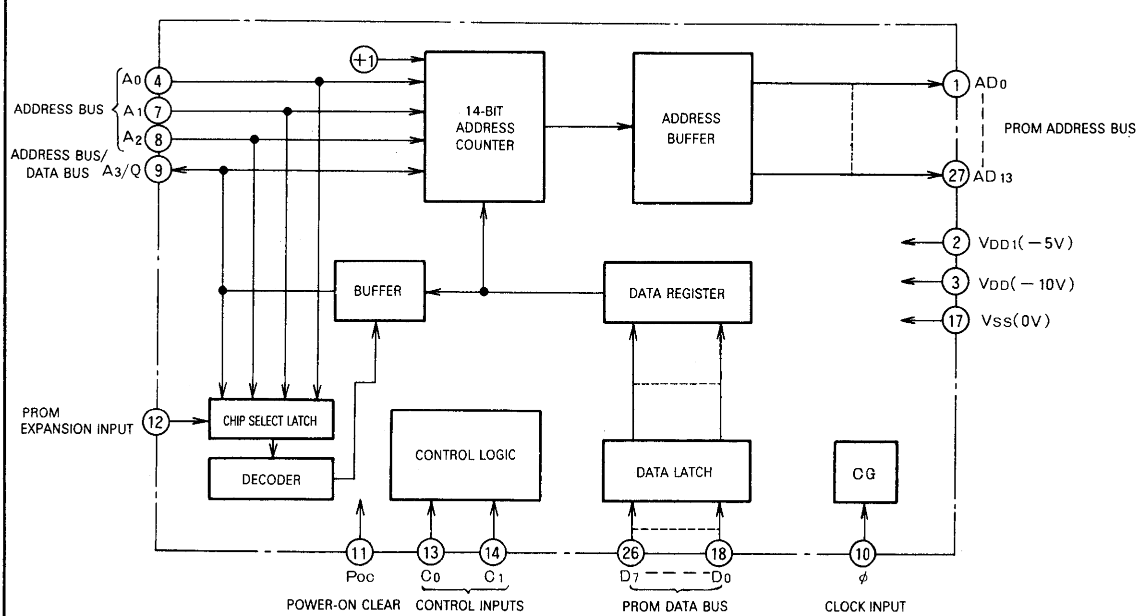
PIN CONFIGURATION (TOP VIEW)



Outline 40S1

NC : NO CONNECTION

BLOCK DIAGRAM



10

BASIC FUNCTIONAL BLOCKS

Function	Operational description
Address counter	This counter is used to specify the address for external EPROMs. When this 14-bit binary counter is used to output continuous addresses for data output, it is automatically incremented by one.
Data register	This 8-bit register is used for temporary storage of EPROM data
Control logic	Provides internal control by means of the control signals C_0 and C_1

PIN DESCRIPTION

Pin	Name	Input or output	Function
$AD_0 \sim AD_{13}$	PROM address bus	Output	Output pin which sends the 14-bit address to the EPROM devices
$A_0 \sim A_2$	Address bus	Input	Input pins for the external input of address into the address counter and data into the chip select latch. This data must be input in five steps.
A_3/Q	Address bus/ data bus	Input/ output	In the address and chip select data input mode this pin acts as an input pin to accept address and data. This data must be input in five steps. In the data output mode this pin acts as an output pin, outputting EPROM data 1 bit at a time in serial fashion. However, for address overflow and when the chip is not selected, this output pin is floating.
ϕ	Clock input	Input	Clock input pin
C_0 C_1	Control signals	Input	Input pins for the control signals which determine the internal operating states
Poc	Power-on clear	Input	Upon power-on, an internal power-on clear circuit automatically clears the internal status. When this pin is set to high, A_3/Q output is prohibited and the internal states are cleared.
$D_0 \sim D_7$	PROM data bus	Input	8-bit parallel input for EPROM data
SL	PROM expansion input	Input	Normally the SL pin is set to low, enabling addressing of 16K bytes of EPROM. For addressing of 32K bytes of EPROM with two M58819S devices, one of the SL pins is set to high. The chip select data is used to switch between the two devices.

FUNCTIONAL DESCRIPTION

The M58819S is controlled by the following four instructions.

Instruction	Effect	Instruction code	
		C_0	C_1
Addressing instruction	Data is loaded into the internal address counter and chip select latch of the M58819S.	0	1
Read instruction	Serially converted data from the EPROM is output 1 bit at a time. This may also be used as an addressing instruction capable of directly addressing EPROM'	1	0
Indirect addressing instruction	Indirectly sets the address of EPROM	1	1
NOP instruction	No operation	0	0

These instructions are described below.

1. Direct addressing

The timing is shown in Fig. 1. First, by using the addressing instruction, A_0, A_1, and A_3/Q inputs are used to read-in in 4-bit parallel format the starting address and chip select data. Then, five addressing instructions are used to completely set the contents of the address counter and chip select latch.

Next, if a read instruction is input, EPROM data is read from the address specified by the sequence above.

The read instruction acts as a direct addressing instruction, and 24 clock cycles after the input of the read instruction EPROM data is available at the A_3/Q pin. In addition, by inputting a read instruction EPROM data can be serially output 1 bit at a time. The address counter is automatically incremented by 1 every 8 read instructions.

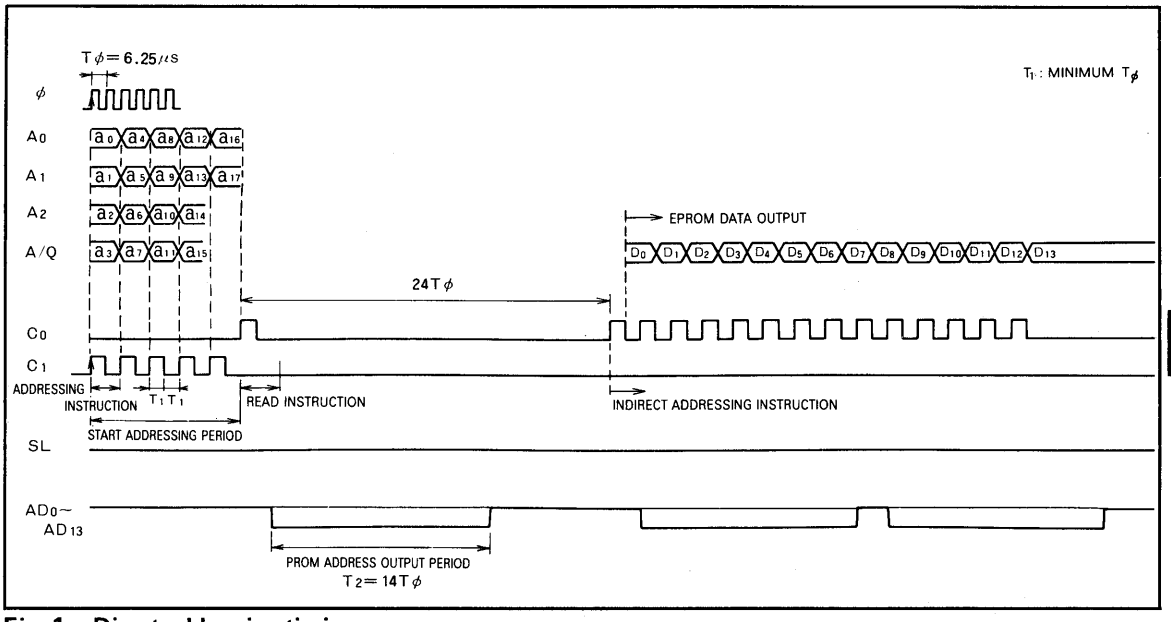


Fig. 1 Direct addressing timing

10

2. Indirect addressing

The timing is shown in Fig. 2. As with direct addressing, an indirect addressing instruction is input after setting of the address.

By using an indirect addressing instruction, EPROM data at the address in the address counter is output and shifted into the address counter.

In addition, when two words of ROM data are shifted into the address counter, 74 clock cycles after the input of an indirect addressing instruction the EPROM data at the set address is available as an output at the A_3/Q pin. By inputting a read instruction, ROM data can be read 1 bit at a time in serial fashion.

The relationship between the two words of ROM data and the address shifted into the address counter is as follows.

If the address counter address is set to 00000_{16} when an indirect addressing instruction is input, the EPROM data at that address $(D_0\ D_1\ D_2\ D_3\ D_4\ D_5\ D_6\ D_7)_2$ and the EPROM data at the address incremented 1 $(00001)_{16}$ $(D'_0\ D'_1\ D'_2\ D'_3\ D'_4\ D'_5\ D'_6\ D'_7)_2$ are output and shifted into the address counter, the overall address being set to $(D'_5\ D'_4\ D'_3\ D'_2\ D'_1\ D'_0\ D_7\ D_6\ D_5\ D_4\ D_3\ D_2\ D_1\ D_0)_2$.

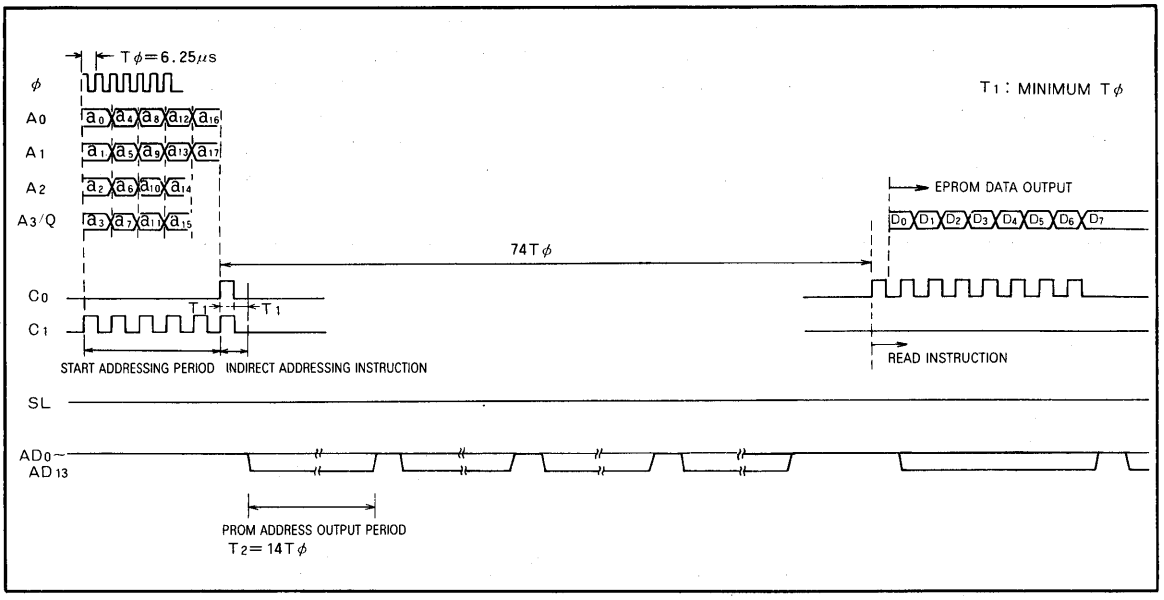


Fig. 2 Indirect addressing timing

3. Interface addressing

The M58819S is capable of addressing up to a maximum of 16K bytes of EPROM. The chip select data is $(0000)_2$.

Therefore, it is necessary to be able to specify an 18-bit address a_i (i=0~17), the relationship between a_i and the PROM address bus being as follows.

a_{17}	a_{16}	a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0
0	0	0	0	AD_{13}	AD_{12}	AD_{11}	AD_{10}	AD_9	AD_8	AD_7	AD_6	AD_5	AD_4	AD_3	AD_2	AD_1	AD_0

Chip select (a_{17}–a_{14}) | Addressing bits for the EPROM (a_{13}–a_0)

MITSUBISHI ELECTRIC

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	0.3~−15	V
V_I	Input voltage	With respect to V_{SS}	0.3~−15	V
Pd	Power dissipation	Ta = 25°C	180	mW
Topr	Operating temperature		−10~70	°C
Tstg	Storage temperature		−40~125	°C

RECOMMENDED OPERATING CONDITIONS (T_a=-10~70°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	−11		−9	V
V_{DD1}	Supply voltage	−5.25	−5	−4.75	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage	−1		0	V
V_{IL}	Low-level input voltage for D_0~D_7 inputs	V_{DD1}		−4.3	V
V_{IL}	Low-level input voltage for A_0~A_3, ϕ, SL, C_0, and C_1 inputs	V_{DD}		−4.5	V
f(ϕ)	External clock frequency	155	160	165	kHz

ELECTRICAL CHARACTERISTICS (T_a=-10~70°C, V_{DD}=-10±1V, V_{DD1}=-5±0.25V, V_{SS}=0V, f(ϕ)=160±5kHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH1}	High-level output voltage for Q output	I_{OH} = 0.5mA	−0.8			V
V_{OH2}	High-level output voltage for AD_0~AD_{10} output	I_{OL} = 0.5mA	−1			V
V_{OH3}	High-level output voltage for AD_{11}~AD_{13} outputs	I_{OL} = 0.5mA	−1			V
V_{OL1}	Low-level output voltage for Q output	I_{OL} = 0.5mA			−4.5	V
V_{OL2}	Low-level output voltage for AD_0~AD_{10} outputs	I_{OL} = 0.1mA			V_{DD1}+0.6	V
V_{OL3}	Low-level output voltage for AD_{11}~AD_{13} outputs	I_{OL} = 0.4mA			V_{DD1}+0.6	V
I_{DD}	Supply current	Un-loaded output			−10	mA
I_{DD1}	Supply current	Un-loaded output			−1	mA
I_I	Input current for all inputs other than P_{OC}	V_I = 0~V_{DD}			±1	μA
I_I	Input current for P_{OC} input	V_I = 0~V_{DD}			−50	μA
Ci	Input capacitance	V_{DD} = V_{DD1} = V_{SS}, f = 1MHz, 250mVrms		7	10	pF

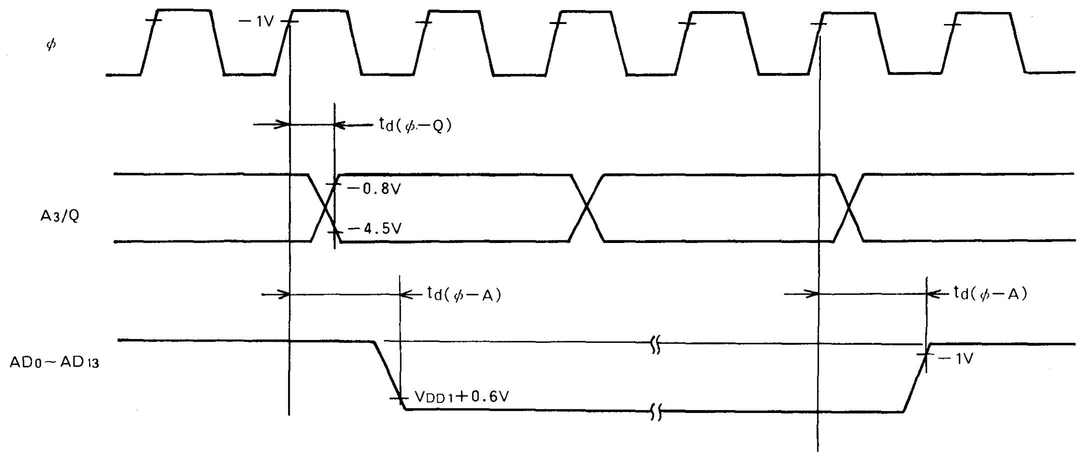
SWITCHING CHARACTERISTICS (T_a=-10~70°C, V_{DD}=-10±1V, V_{LD}=-5±0.25V, f=160±5kHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{d(\phi-Q)}$	Data output delay time (Note 1)	C_L = 100pF			3	μs
$t_{d(\phi-A)}$	Address delay time (Note 2)	C_L = 200pF			6	μs

Note 1. Applies to pin 9 (V_{DD}-related output)

2. Applies to pins 1, 27, 28, 29, 31, 32, 33, 34, 35, 37, 38, 39, and 40 (V_{LD1}-related outputs)

TIMING DIAGRAMS



GENERAL-PURPOSE MOS LSIs

11

M50110XP, M50115XP

30～120-FUNCTION REMOTE-CONTROL TRANSMITTERS

DESCRIPTION

The M50110XP and M50115XP are remote-control transmitter circuits manufactured by aluminum-gate CMOS technology for use in television receivers, audio equipment and other devices using infrared for transmission. The M50110XP conveys 30 different commands on the basis of a 10-bit PCM code, while the M50115XP conveys 120 different commands. These transmitters are intended to be used in conjunction with an M50111XP, M50116XP or M50117XP receiver. The X in each type corresponds to blank, A, B or C, which are respectively used for audio equipment, TV and VTR, air conditioners and other applications, or video-disk equipment.

FEATURES

Type	Remote-control function
M50110XP	30
M50115XP	120

- Single power supply
- Wide supply voltage range: 2.2V～8V
- Low power dissipation:
  Idle state (V_{DD}=3V): 3mW (typ)
  3μW (max)
- Has many functions and various uses
- Low-cost LC or ceramic oscillator used for reference frequency
- Low external component count
- Low transmitter duty cycle for minimal power consumption
- High-speed transmission

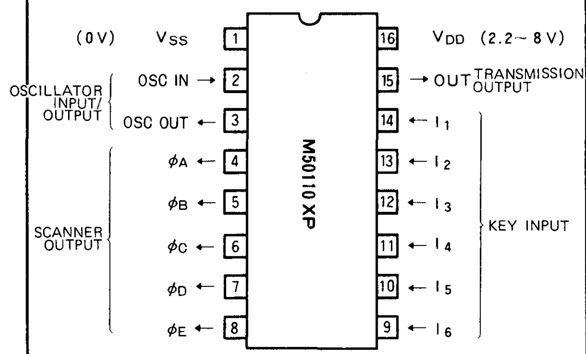
APPLICATION

- Remote-control transmitter for audio equipment, TV, VTR, air conditioners and video-disk equipment

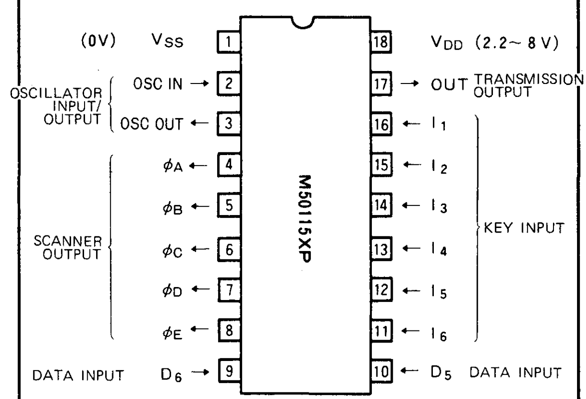
FUNCTION

The M50110XP and M50115XP transmitter circuits for infrared remote-control systems consist of an oscillator, a timing generator, a scanner, a key-in encoder, an instruction decoder, a code modulator and an output buffer. In M50110XP with a 6x5 keyboard matrix 30 commands can be transmitted by 10-bit PCM codes. In M50115XP, with a 6x5 keyboard matrix and two data inputs 120 commands can be transmitted. Oscillation is stopped when none of the keys are depressed in order to minimize power consumption.

PIN CONFIGURATIONS (TOP VIEW)



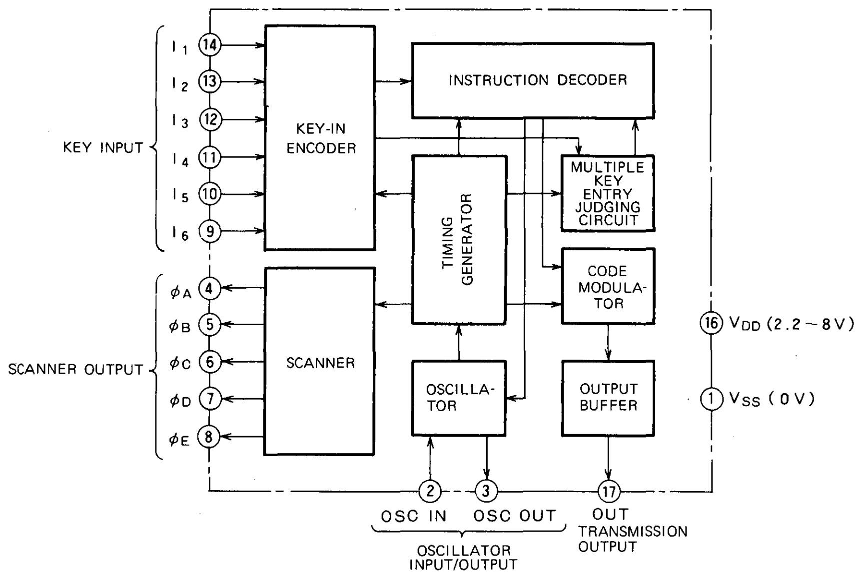
Outline 16P4



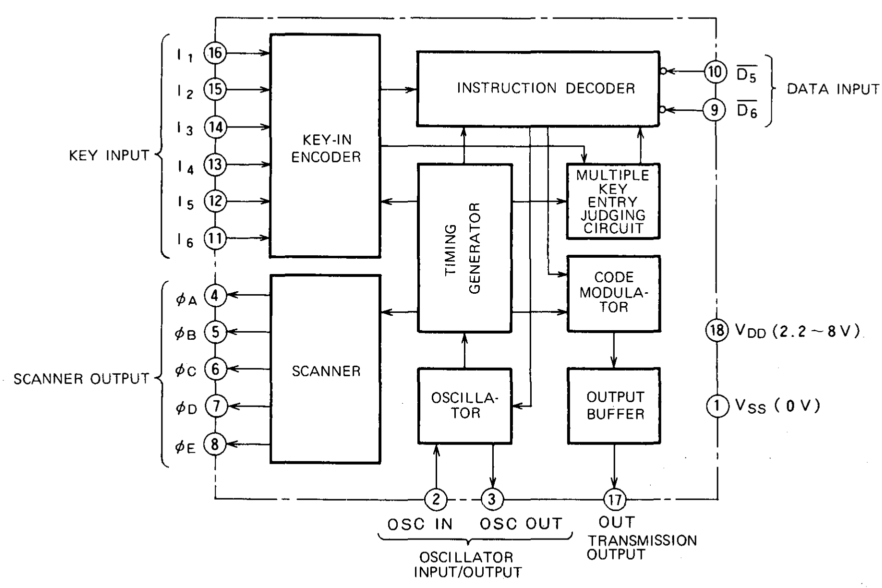
Outline 18P4

BLOCK DIAGRAM

M50110XP



M50115XP



FUNCTION

Oscillator

As the oscillator is on chip, oscillation frequency is easily obtained by connecting an external LC network or ceramic resonator between the OSC IN and OSC OUT terminals. Fig. 1 and Fig. 2 show examples of typical oscillation circuits.

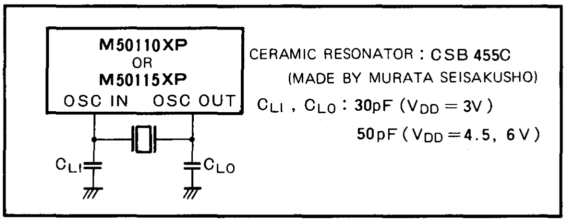


Fig. 1 An example of an oscillator (using a ceramic resonator)

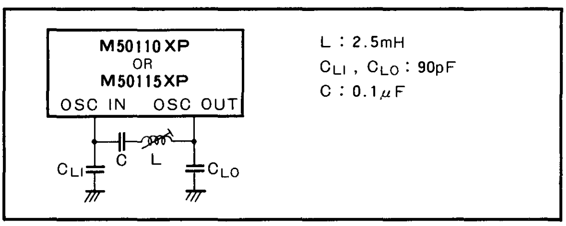


Fig. 2 An example of an oscillator (using an LC network)

Setting the oscillation frequency to 480kHz (or 455kHz) will also set the signal transmission carrier wave to 400kHz (or 38kHz).

Power consumption is minimized by stopping oscillation in the oscillator when none of the keys is depressed.

Key input and data input

In the M50110XP, 30 different commands can be sent through a 6x5 keyboard matrix, consisting of inputs I_1~I_6 and scanner outputs I_A~I_E. In the M50115XP, 120 different commands can be sent because two data inputs, $\overline{D_5}$ and $\overline{D_6}$, are also used.

Table 2 shows the relationship between the keyboard matrix and the transmission code.

Table 1 Key code, type number and use

Key code			Type number	Use
K_0	K_1	K_2		
0	0	0	M50110P M50115P	Remote control for audio equipment
1	0	0	M50110AP M50115AP	Remote control for TV and VTR
0	1	0	M50110BP M50115BP	Remote control for air conditioners and other application
0	0	1	M50110CP M50115CP	Remote control for video-disk equipment

Table 2 Relation between the keyboard matrix and the transmission code names

	ϕ_A	ϕ_B	ϕ_C	ϕ_D	ϕ_E
I_6	A−1	A−2	A−3	A−4	A−5
I_5	A−6	A−7	A−8	A−9	A−10
I_4	A−11	A−12	A−13	A−14	A−15
I_3	B−0	B−1	B−2	B−3	B−4
I_2	B−5	B−6	B−7	B−8	B−9
I_1	B−10	B−11	B−12	B−13	B−14

Table 3 Relation between the transmission code names and the transmission codes

Transmission code name	Transmission				
	D_0	D_1	D_2	D_3	D_4
A−1	1	0	0	0	0
A−2	0	1	0	0	0
A−3	1	1	0	0	0
A−4	0	0	1	0	0
A−5	1	0	1	0	0
A−6	0	1	1	0	0
A−7	1	1	1	0	0
A−8	0	0	0	1	0
A−9	1	0	0	1	0
A−10	0	1	0	1	0
A−11	1	1	0	1	0
A−12	0	0	1	1	0
A−13	1	0	1	1	0
A−14	0	1	1	1	0
A−15	1	1	1	1	0
B−0	0	0	0	0	1
B−1	1	0	0	0	1
B−2	0	1	0	0	1
B−3	1	1	0	0	1
B−4	0	0	1	0	1
B−5	1	0	1	0	1
B−6	0	1	1	0	1
B−7	1	1	1	0	1
B−8	0	0	0	1	1
B−9	1	0	0	1	1
B−10	0	1	0	1	1
B−11	1	1	0	1	1
B−12	0	0	1	1	1
B−13	1	0	1	1	1
B−14	0	1	1	1	1

Transmission Commands

In the M50110XP, 30 commands can be transmitted by 10-bit PCM codes (K_0~K_2, D_0~D_6), and in the M50115XP, 120 commands can be transmitted. The first three bits K_0~K_2, which are key codes between transmitters and receivers, correspond to type numbers and uses. Relation between key codes, type numbers and uses of remote control systems is shown in Table 1.

The next five bits D_0~D_4 correspond to the 6x5 keyboard matrix. Relation between transmission codes and their name is shown in Table 2.

The last two bits, D_5 and D_6, are controlled by the data inputs D_5 and D_6. When terminal D_5 or D_6 is open or high level, data code D_5 or D_6 becomes "0", and when terminal D_5 or D_6 is low level, code data D_5 or D_6 becomes "1".

In the M50110XP, the data bits D_5 and D_6 are fixed in "0." To prevent spurious operation, the codes are designed so that there is no transmission code whose data bits D_0~D_6 are all "0" or "1."

Transmission Coding

When oscillation frequency f_{osc} is 480kHz, transmission of data code is executed as follows: when f_{osc} is other than 480kHz, the period is multiplied by 480kHz/f_{osc} and its frequency by f_{osc}/480kHz.

A single pulse is amplitude-modulated by a carrier of 40kHz, and the pulse width is 0.25ms. Therefore a single pulse consists of 10 clock pulses of 40kHz (see Fig. 3).

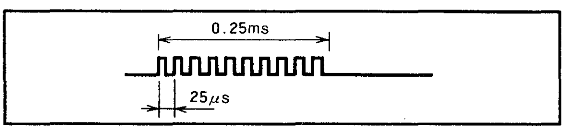


Fig. 3 A single pulse modulated onto carrier (40kHz)

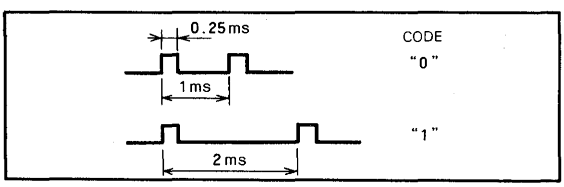


Fig. 4 Distinction between the bits "1" and "0"

The distinction between "0" and "1" bits is made by the pulse interval between two pulses, with an 1ms interval corresponding to "0", and a 2ms interval representing "1" (see Fig. 4).

One command word is composed of 10 bits, that is, of 11 pulses, and it is transmitted in the 24ms cycle while a matrix switch is depressed (see Fig. 5).

As mentioned above, adopting of this code means that the period during which output is high level (i.e. signal emitting LED is lit) is shorter than in continuous wave transmission. Indeed the LED is on for only half of the 11-pulse period or 1.375ms, which is 5.7% of the 24ms entire cycle. This not only saves in total power consumption, but it also improves LED reliability. That is to say, emission can be increased on the same power consumption.

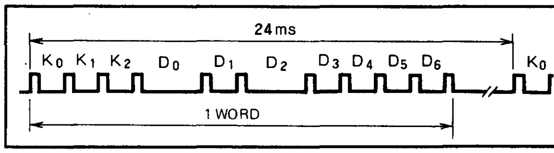


**Fig. 5 Synthesis of one word
(the code below shows 0001010000)**

MITSUBISHI ELECTRIC

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to GND	$-0.3 \sim 9$	V
V_I	Input voltage		$V_{SS} \leqq V_I \leqq V_{DD}$	V
V_O	Output voltage		$V_{SS} \leqq V_O \leqq V_{DD}$	V
Pd	Maximum power dissipation	Ta = 25°C	300	mW
Topr	Operating free-air temperature range		$-30 \sim 70$	°C
Tstg	Storage temperature range		$-40 \sim 125$	°C

RECOMMENDED OPERATING CONDITIONS (Ta = −30～70°C, unless otherwise noted)

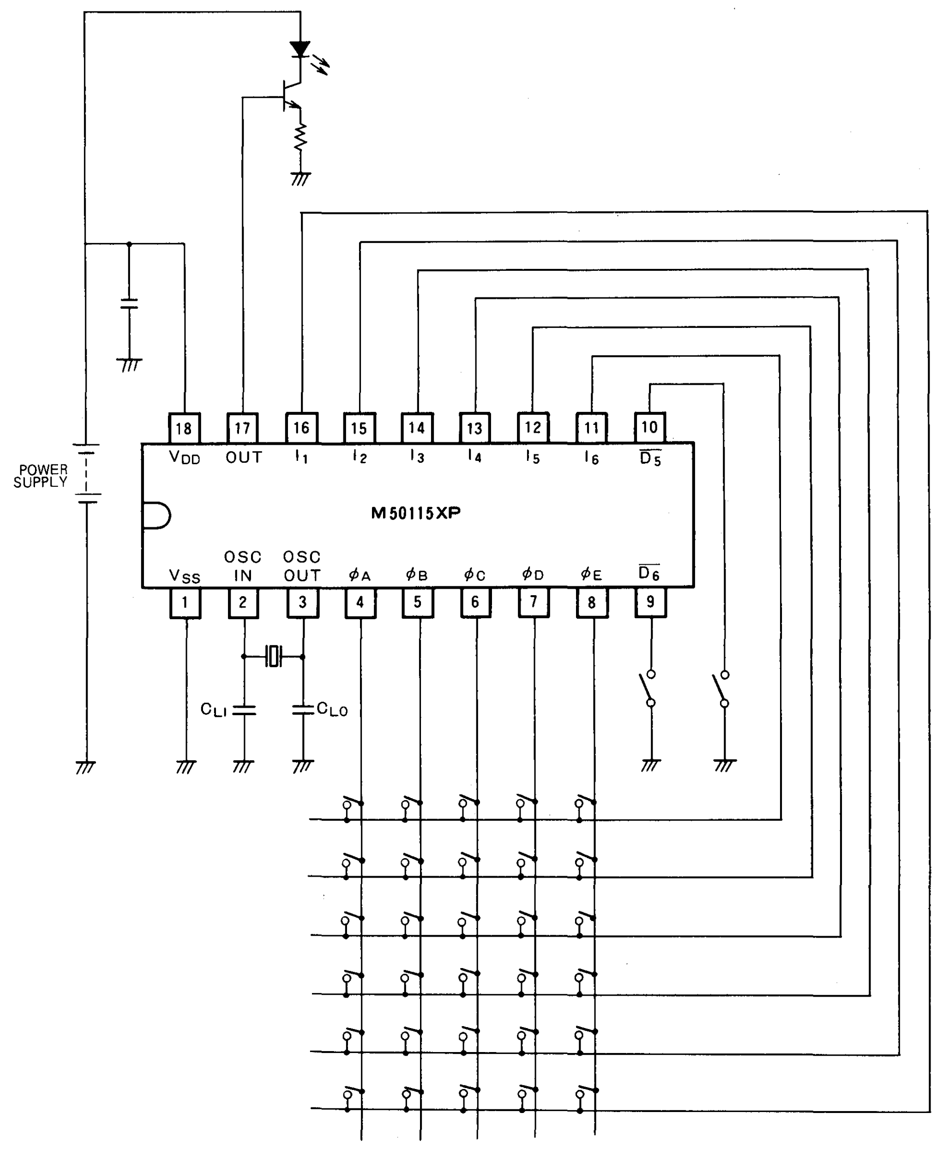
Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	2.2		8	V
V_{IH}	High-level input voltage	$0.7 \times V_{DD}$		V_{DD}	V
V_{IL}	Low-level input voltage	0		$0.3 \times V_{DD}$	V
f_{OSC}	Oscillation frequency		455		kHz
			480		kHz

ELECTRICAL CHARACTERISTICS (Ta = 25°C, unless otherwise noted)

Symbol	Parameter	Test conditions		Limits			Unit
				Min	Typ	Max	
V_{DD}	Operational supply voltage	Ta = −30～70°C, f_{OSC} = 455kHz		2.2		8	V
I_{DD}	Supply voltage during operation	f_{OSC} = 455kHz	V_{DD} = 3 V		0.1	0.5	mA
			V_{DD} = 6 V		0.5	2	mA
I_{DD}	Supply voltage during non-operation	V_{DD} = 3 V				1	μA
		V_{DD} = 8 V				5	μA
R_I	Pull-up resistances, $I_1 \sim I_6$				20		kΩ
I_{OL}	Low-level output currents, $\phi_A \sim \phi_E$	V_{DD} = 3 V, V_{OL} = 0.9V		0.18	0.6		mA
		V_{DD} = 6 V, V_{OL} = 1.8V		0.7	3		mA
I_{OH}	High-level output current, OUT	V_{DD} = 3 V, V_{OH} = 2 V		−2	−5		mA
		V_{DD} = 6 V, V_{OH} = 4 V		−8	−16		mA

11

APPLICATION EXAMPLE (M50115XP)



30 ~ 120-FUNCTION REMOTE-CONTROL RECEIVER

DESCRIPTION

The M50111XP, M50116XP and M50117XP are remote control receiver circuits manufactured by aluminum-gate CMOS technology for use in television receivers, audio equipment and other applications using infrared transmission. The systems can receive 30~120 different 10-bit PCM code commands by remote control.

The M50111XP, M50116XP and M50117XP are designed for use with an M50110XP or M50115XP transmitter. The X in each type number corresponds to blank, A, B or C, which are respectively used for audio equipment, TV and VTR, air conditioner and other applications, or video-disk equipment.

FEATURES

Type	Remote-control function		Parallel outputs
	Serial data	Parallel data	
M50111XP	120	30	D_0~D_3,$\overline{STA}$,$\overline{STB}$
M50116XP	120	60	D_0~D_3,$\overline{STA}$~$\overline{STD}$
M50117XP	120	120	D_0~D_7, FF

- Single power supply
- Wide power supply voltage range: 4.5V~8V
- Low power dissipation
- Low-cost LC or ceramic oscillator used for frequency reference
- Information is transmitted by pulse code modulation
- High speed reception
- Superior noise immunity — instructions are not executed unless the same code is received two or more times in succession
- Single transmission frequency (40kHz or 38kHz) for carrier wave
- Many functions and various uses
- Large tolerance in operating frequency between the transmitter and the receiver
- Can be simply connected to a microcomputer

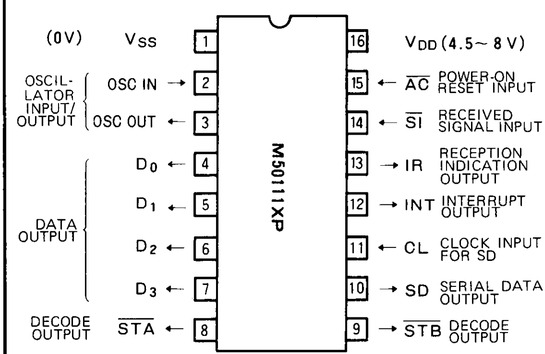
APPLICATION

- Remote control receivers for audio equipment, TV, VTR, air conditioners, video-disk equipment and similar devices

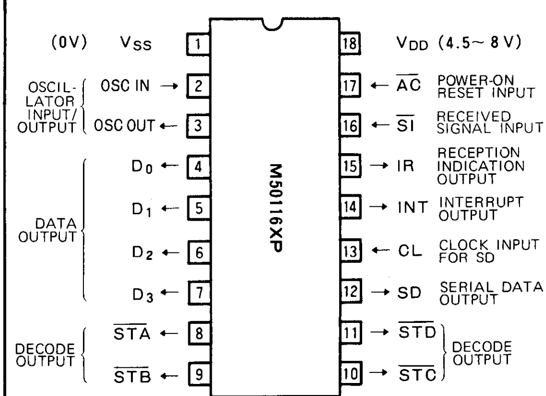
FUNCTION

The M50111XP, M50116XP and M50117XP receivers for infrared remote control systems consist of an oscillator, a timing generator, a demodulator, an error prevention circuit, a reception state decision circuit, a serial data processor, a shift register, a received signal input circuit, power-on reset circuit and other circuits. The M50111XP, M50116XP and M50117XP are designed to decode and execute instructions after 2 successive receptions of the identical instruction code. This provides positive assurance that noise will not be executed as instructions.

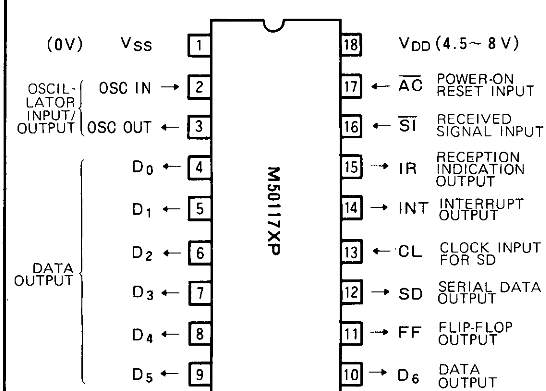
PIN CONFIGURATIONS (TOP VIEW)



Outline 16P4



Outline 18P4



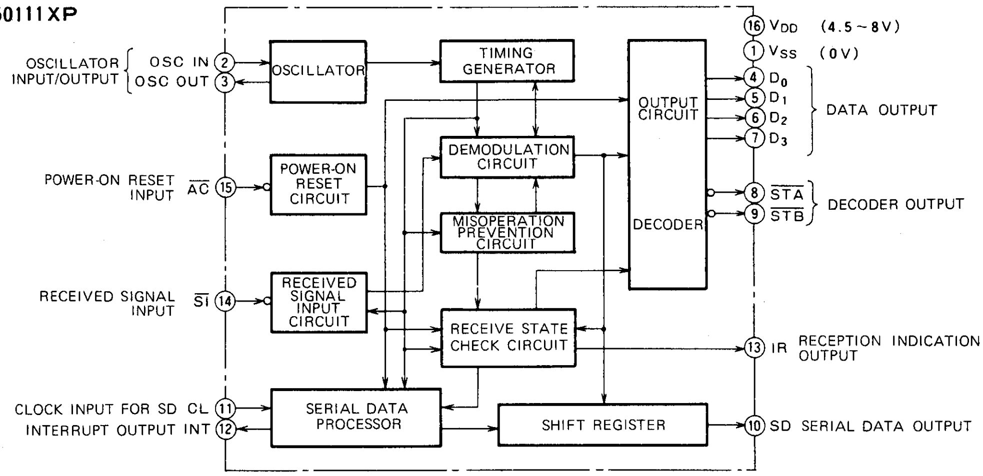
Outline 18P4

With the data outputs D_0~D_6 and the decode outputs $\overline{STA}$~$\overline{STD}$, M50111XP can process 30 different instructions, the M50116XP can process 60 different instructions and the M50117XP can process 120 different instructions. With a serial data output SD, 120 different instructions can be processed by any of the receivers.

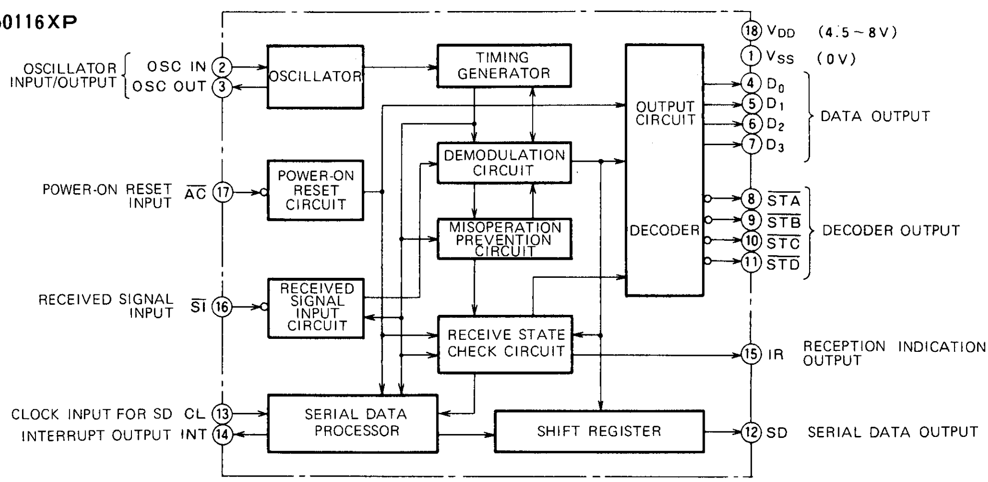
30~120-FUNCTION REMOTE-CONTROL RECEIVER

BLOCK DIAGRAM

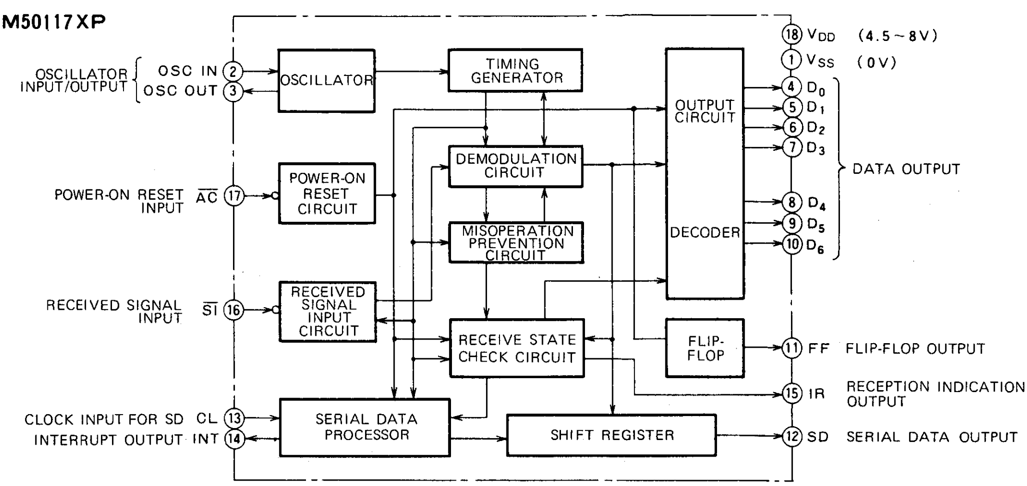
M50111XP



M50116XP



M50117XP



30 ~ 120-FUNCTION REMOTE-CONTROL RECEIVER

FUNCTION

Oscillator

As the oscillator is on chip, oscillation frequency is easily obtained by connecting an external LC network or ceramic resonator between the OSC IN and OSC OUT terminals. Fig. 1 and Fig. 2 show examples of typical oscillation circuit.

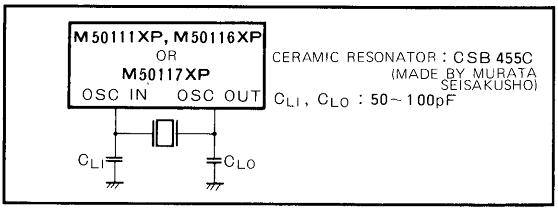


Fig. 1 An example of an oscillator (using a ceramic resonator)

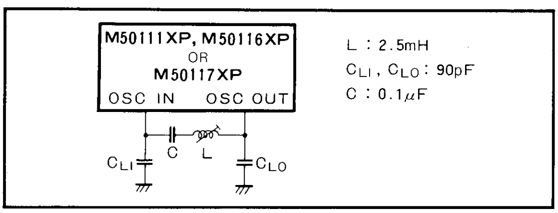


Fig. 2 An example of an oscillator (using an LC network)

When oscillation frequency f_{OSC} is 480kHz, execution is as follows:

Received Signal Input Circuit and Demodulation Circuit

The received signal, sensed by the photo detector, is amplified and an integrated signal is supplied through $\overline{SI}$ to be processed by the received signal input circuit, and then it is sent to the demodulation circuit. In the demodulation circuit, the pulse interval of the signal is analyzed and then converted to a digital code. Fig. 3 shows the relationship between the $\overline{SI}$ input wave form, codes and data.

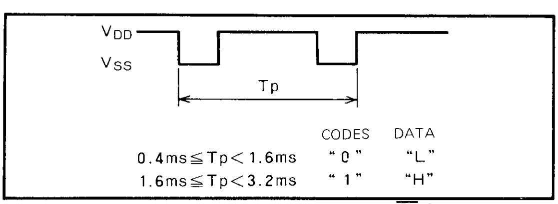


Fig. 3 The relationship between the $\overline{SI}$ input wave form, code and data

When the input pulse interval to the $\overline{SI}$ input is 3.2 ms or longer, it will be assumed to be the end of a word, but if the interval is finally 50 ms or longer it will be accepted as the end of the command transmission and the device will be put in the idle state. In the idle state, the data outputs D_0~D_6 and the reception indication output IR goes to low-level and the decoder outputs $\overline{STA}$~$\overline{STD}$ go to high-level.

Misoperation Prevention Circuit

Any signal whose low-level interval at $\overline{SI}$ input is less than 50~100 μs is not accepted as a transmission signal.

When a pulse interval T_p is less than 0.4 ms, the misoperation prevention circuit resets to idle state to prevent an error. When all data codes D_0~D_4 are supplied as 0 or 1, it resets to idle state.

Receive State Check Circuit

The reception indication output IR becomes high-level after receiving the same transmission code 2 or more times in succession. Therefore reception states of an instruction from the transmitter can be indicated by an LED connected to the output IR.

Reception Code, Data Output, Decode Output and Flip-flop Output

Data outputs D_0~D_6 correspond to D_0~D_6 of the transmission codes. When a code is 0, the data output will be low-level, and when a code is 1, the data output will be high-level, while decode outputs $\overline{STA}$~$\overline{STD}$ correspond to transmission codes D_4, D_5 as shown in Table 1. When the transmission codes D_0~D_6 are 1010000, the flip-flop output FF will go to high-level, and when the codes are 0101000, the output FF will go to low-level.

Table 2 shows the relationship between key codes and type numbers, and examples of their use.

Table 1 The relationship between the decode outputs and the transmission codes D_4, D_5

Transmission code		Decode output			
D_4	D_5	$\overline{STA}$	$\overline{STB}$	$\overline{STC}$	$\overline{STD}$
0	0	L	H	H	H
1	0	H	L	H	H
0	1	H	H	L	H
1	1	H	H	H	L

Table 2 The relationship between be key codes, types numbers examples of their use

Key code			Type number	Use
K_0	K_1	K_2		
0	0	0	M50111P M50116P M50117P	Remote control for audio equipment
1	0	0	M50111AP M50116AP M50117AP	Remote control for TV, VTR
0	1	0	M50111BP M50116BP M50117BP	Remote control for air conditioners and others
0	0	1	M50111CP M50116CP M50117CP	Remote control for video-disk equipment

Serial Data Processor

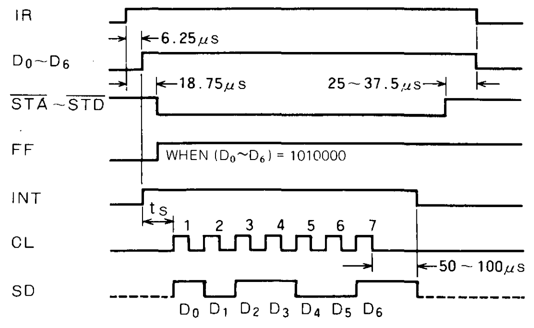
When an identical code is received twice, the reception indication output IR is turned from low-level to high-level and then after 6.25μs delay the interrupt output INT is turned from low-level to high-level (see the timing diagram). When pulses are supplied to the clock input CL for SD while the INT output is high-level, the received data are sent from the serial data output SD. These data are synchronized with the rising edge of the CL input pulses. Thus the contents of the transmission code can be read, if the SD output is decided at the falling edge of the CL input pulses.

The SD output is a three-state output, which is usually in the disabled state (high impedance). After an interrupt output INT goes to high-level, the disabled state is absolved at the first low-to-high transmission of a CL input pulses. And then the data D_0~D_6 is serially sent, and after 50~100μs from the seventh high-to-low transmission of CL input pulses, the SD output is again put in the disabled state and at the same time the INT output goes to low-level.

Power-on Reset Circuit

Attaching a capacitor to the terminal $\overline{AC}$, the power-on reset function can be activated when power supply is applied to the IC. When the $\overline{AC}$ input is turned to low-level, the data outputs D_0~D_6, the reception indication output IR, the interrupt output INT and the flip-flop output FF go to low-level, the decode outputs $\overline{STA}$~$\overline{STD}$ go to high-level and the serial data output SD is put in disabled state.

Timing Diagram



After the INT output becomes high-level, when the received code is not identical to the previously received code before the first fall of the CL input, the INT output is returned to low-level; at the same time the $\overline{STA}$~$\overline{STD}$ outputs become high level and the SD output become a disabled state. After the INT output goes to high-level, when received codes are not identical after the first fall of the CL input, the data D_0~D_6 are sent and then the INT output goes to low-level after 50~100μs from the seventh fall of CL input pulses and the SD output is put in the disabled state.

The time t_s from the rising edge of the INT output to the rising edge of the CL input must be at least 6.25μs.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	−0.3 ~ 9	V
V_I	Input voltage		$V_{SS} \leqq V_I \leqq V_{DD}$	V
V_O	Output voltage		$V_{SS} \leqq V_O \leqq V_{DD}$	V
Pd	Maximum power dissipation	Ta = 25°C	300	mW
Topr	Operating free-air temperature range		−30 ~ 70	°C
Tstg	Storage temperature range		−40 ~ 125	°C

RECOMMENDED OPERATING CONDITIONS (Ta = −30 ~ 70°C, unless otherwise noted)

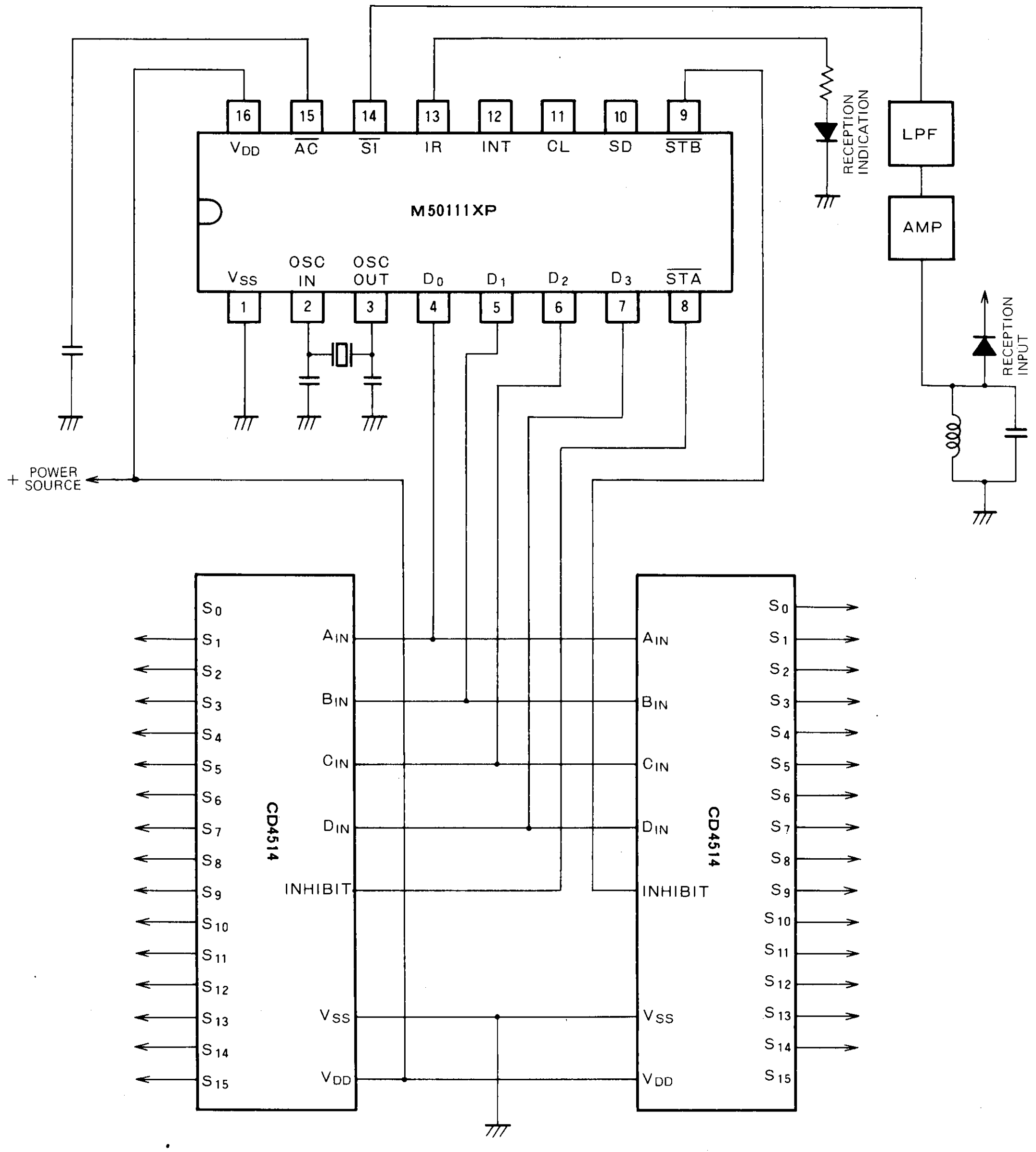
Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	4.5		8	V
V_{IH}	High-level input voltage	$0.7 \times V_{DD}$		V_{DD}	V
V_{IL}	Low-level input voltage	0		$0.3 \times V_{DD}$	V
f_{OSC}	Oscillation frequency		455		kHz
			480		kHz

ELECTRICAL CHARACTERISTICS (Ta = 25°C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{DD}	Operational supply voltage	Ta = −30 ~ 70°C, f_{OSC} = 455kHz	4.5		8	V
I_{DD}	Supply current from V_{DD}	V_{DD} = 5V, f_{OSC} = 455kHz		0.3	1.0	mA
I_{OH}	High-level output current, SD	V_{DD} = 4.5V, V_{OH} = 2.4V	−2	−6		mA
I_{OH}	High-level output current, INT, IR	V_{DD} = 4.5V, V_{OH} = 2.4V	−1	−3		mA
I_{OH}	High-level output current, D_0~D_6, $\overline{STA}$~$\overline{STD}$, FF	V_{DD} = 4.5V, V_{OH} = 2.4V	−0.5	−1.5		mA
I_{OL}	Low-level output current, D_0~D_6, $\overline{STA}$~$\overline{STD}$, FF, SD, INT, IR	V_{DD} = 4.5V, V_{OL} = 0.4V	1.6	3.2		mA
R_I	Pull-up resistance, $\overline{SI}$			20		kΩ
R_I	Pull-up resistance, $\overline{AC}$			48		kΩ
R_I	Pull-down resistance, CL			63		kΩ

11

APPLICATION EXAMPLE (M50111XP)



MITSUBISHI ELECTRIC

REFRIGERATOR CONTROLLER

DESCRIPTION

The M50250P is a semiconducter integrated circuit which uses aluminum-gate CMOS technology. It is designed for use as a refrigerator controller and provides compressor and defrosting control as well as a door alarm function.

Commercial power frequencies of 50/60Hz can be used as the reference signal.

FEATURES

- Excellent noise immunity
- Wide operating voltage range
- Built-in voltage regulating Zener diode
- Low power dissipation
- Defrosting control
- 6-minute compressor stop control
- Compressor startup control
- Door alarm control

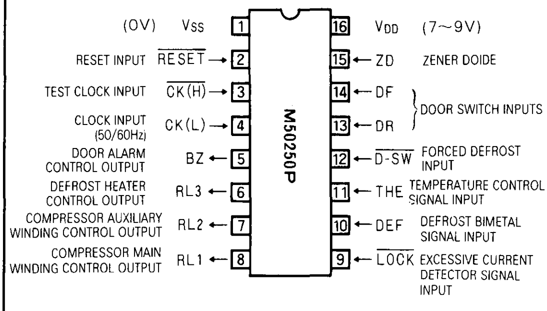
APPLICATIONS

Refrigerator and compressor control

FUNCTION

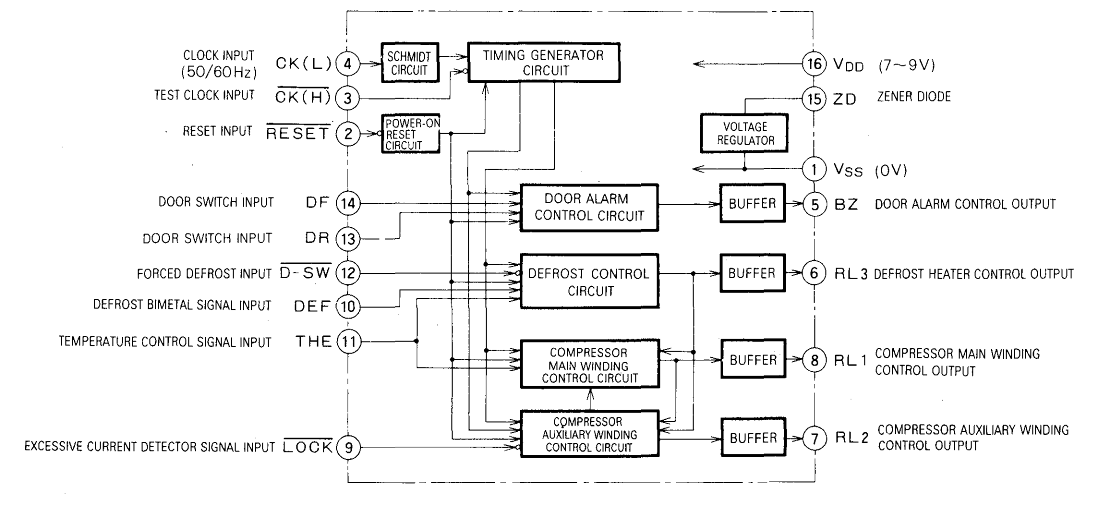
The M50250P is a refrigerator control IC which uses commercial power frequencies (50/60Hz) as a reference signal to provide defrost control, compressor 6-minute stop control, compressor startup control, and door alarm control functions.

PIN CONFIGURATION (TOP VIEW)



Outline 16P4

BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

For the purposes of the description below, the times shown apply if a 60Hz signal is applied to the CK(L) pin. These times are multiplied by 1.2 for a 50Hz input frequency.

DEFROST CONTROL

The amount of time that the temperature control signal input THE is high is counted until the total reaches 7.40 hours, at which time if the defrost bimetal signal input DEF is low, the defrost heater control output RL3 goes high. When the defrost operation is completed, the DEF input goes high and the RL3 output goes low.

In addition, by setting the forced defrost input $\overline{\text{D-SW}}$ to low, forced defrosting is performed even if 7.40 hours have not elapsed.

COMPRESSOR CONTROL

1) 6-Minute Stop Function

As a protective feature for the compressor, if the compressor is temporarily stopped, a 6-minute timer operates. Until this 6-minute time limit is reached, the compressor remains in the stopped condition. However, when power of M50250P is supplied the compressor operates immediately.

2) Startup Control

Whenever power is supplied or the 6-minute timer period elapses, when the temperature control signal input THE is high and the RL3 output is low, the main winding control output RL1 goes high and the compressor operates. At this startup time, if the excessive current detector signal input $\overline{\text{LOCK}}$ goes low, the auxiliary winding control output RL2 becomes high and the compressor goes into two-phase operation.

When the $\overline{\text{LOCK}}$ input goes from low to high and the high period lasts for 0.5 seconds (more precisely 0.495 ~ 0.5 seconds), the RL2 output goes from high to low, and the compressor goes into single-phase operation.

3) Excessive Current Detector Signal

When the compressor is started-up, a 10-second timer is reset, and the clock is started. Within 5 seconds (or more precisely 4 ~ 5s) of this startup, the $\overline{\text{LOCK}}$ input goes low and the RL2 output goes high whereupon two-phase operation begins. The 10-second timer is again reset and the clock begins running. If the $\overline{\text{LOCK}}$ input does not remain high for at least 0.5s, after 5 seconds the 6-minute stop function operates and the compressor is stopped.

If the $\overline{\text{LOCK}}$ input goes low within 5 to 10 seconds of the compressor startup, the 6-minute stop function operates.

If the $\overline{\text{LOCK}}$ input goes low after more than 10 seconds after the compressor startup, the RL2 output goes high, the 10-second timer is reset and the clock begins running. That is, the initial startup condition is restored.

DOOR ALARM CONTROL

When either the DF or DR door switch inputs or both the inputs are high for more than 15 seconds (more precisely 14.9 ~ 14.95), the buzzer alarm control output BZ goes high for 0.15 seconds and until both DF and DR go low, this output alternates between 1.5 seconds low and 0.15 seconds high.

POWER ON RESET FUNCTION

By connecting a capacitor to the $\overline{\text{RESET}}$ pin, an automatic reset can be performed when power is applied to the M50250P.

When this automatic reset operates, the timers and flip-flops controlling the outputs are reset or set appropriately.

However, to ensure proper power on reset operation, the supply voltage V_{DD} rising edge should be at least 10ms removed from the rising edge of the $\overline{\text{RESET}}$ input.

REGULATED POWER SUPPLY

An internal Zener diode is used to provide simple regulated power supply. In addition, the Zener diode pin (ZD) is independent of the supply voltage pin (V_{DD}) making it useful as the overall system regulated power supply.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	$-0.3\sim9.5$	V
V_I	Input voltage		$V_{SS} \leqq V_I \leqq V_{DD}$	V
V_O	Output voltage		$V_{SS} \leqq V_O \leqq V_{DD}$	V
P_d	Power dissipation	Ta = 25°C	250	mW
T_{opr}	Operating free-air temperature range		$-30\sim75$	°C
T_{stg}	Storage temperature range		$-40\sim125$	°C

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	7		9	V
P_d	Zener power dissipation			100	mW
f_{IN}	Input frequency, CK(L)		50		Hz
			60		Hz
V_{IH}	High-level input voltage, CK(L)	$0.9\times V_{DD}$	V_{DD}	V_{DD}	V
V_{IL}	Low-level input voltage, CK(L)	0	0	$0.1\times V_{DD}$	V
V_{IH}	High-level input voltage for inputs other than CK(L)	$0.7\times V_{DD}$	V_{DD}	V_{DD}	V
V_{IL}	Low-level input voltage for inputs other than CK(L)	0	0	$0.3\times V_{DD}$	V

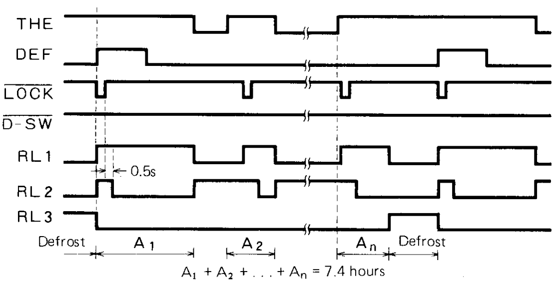
ELECTRICAL CHARACTERISTICS (T_a=25°C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{DD}	Operating supply voltage	$f_{CK(L)}$=60Hz, Ta = $-30\sim75$°C	4.5		9	V
V_{ZD}	Zener voltage	I_{ZD}= 2 mA	7		9	V
		I_{ZD}= 10mA	7		9	V
I_{DD}	Supply current	V_{DD}= 9 V, $f_{CK(L)}$=60Hz Input and output terminals open			1	mA
I_{OH}	High-level output current	V_{DD}= 7 V, V_O= 5 V	−2			mA
I_{OL}	Low-level output current	V_{DD}= 7 V, V_O= 2 V	2			mA
R_I	Pull-up resistance, inputs other than CK(L)			20		kΩ

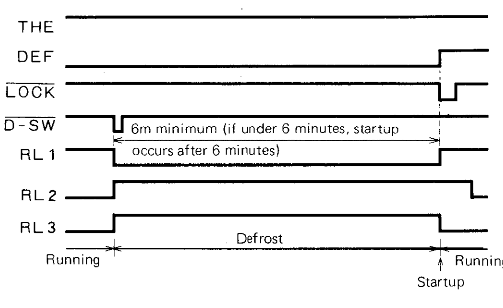
11

TIMING DIAGRAMS

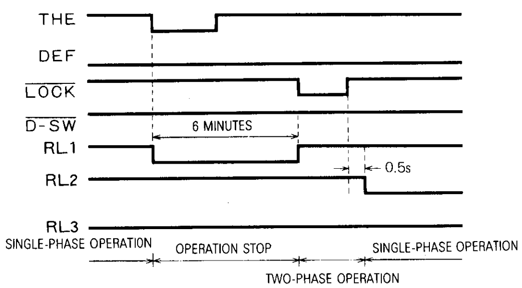
Defrost control



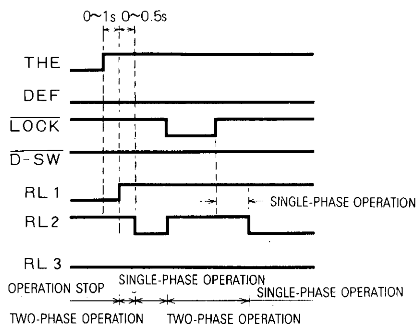
Forced defrost



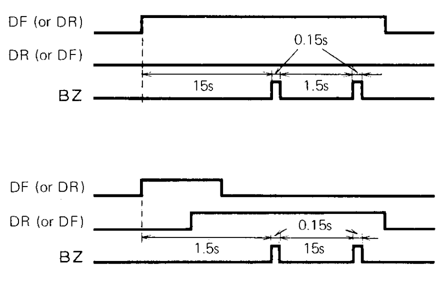
Compressor 6-minute stop function



Compressor startup

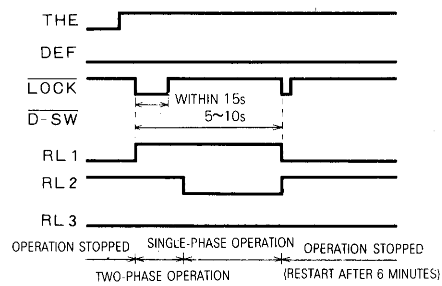


Door alarm control

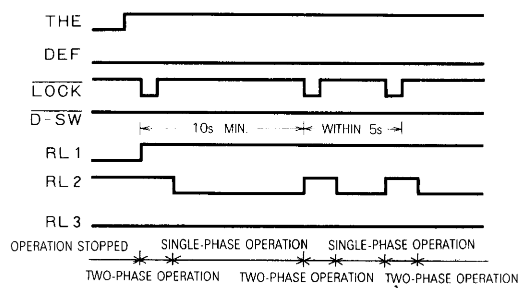


Receipt of the excessive current detector signal input

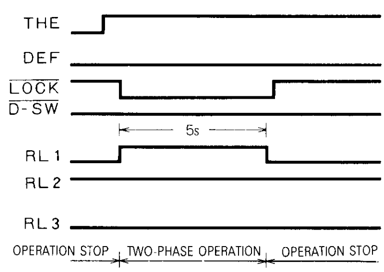
(1) When LOCK input goes low within 5 to 10 seconds from startup



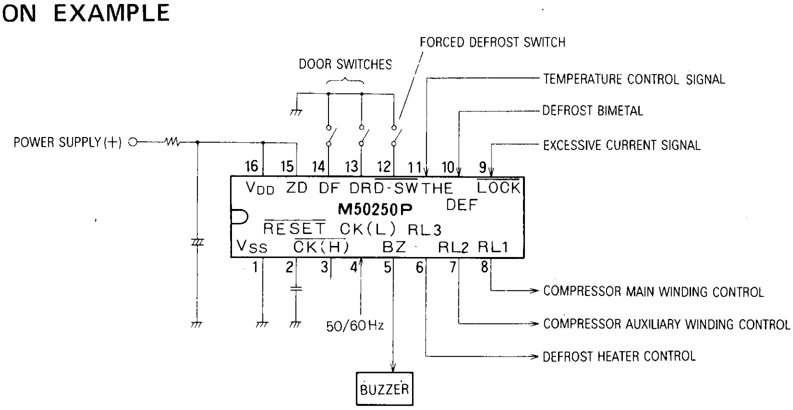
(2) When LOCK input goes low after more than 10 seconds after startup



(3) When LOCK input is low for more than 5 seconds after startup (or when lock input goes low 10 seconds after startup)



APPLICATION EXAMPLE



M50401P, M50402P
M50403P, M50404P, M50405P

CMOS ANALOG CLOCK CIRCUITS

DESCRIPTION

The M50401P, M50402P, M50403P, M50404P and M50405P comprise a family of CMOS circuits designed for use with quartz crystal oscillator elements in clock applications.

TYPE	PROCESS	QUARTZ CRYSTAL	MOTOR	ALARM SOUND
M50401P M50402P	SILICON GATE CMOS	4.1943MHz	31ms	2048×2×1Hz 2048×8×1Hz
M50403P M50404P	SILICON GATE CMOS	32.768kHz	31ms	2048×2×1Hz 2048×8×1Hz
M50405P	SILICON GATE CMOS	32.768kHz	16ms	2048×16×1Hz

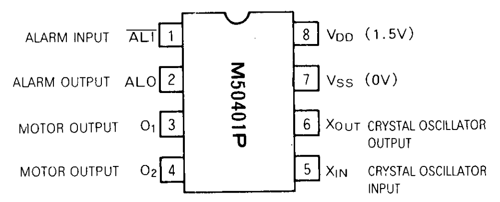
FEATURES

- Low power consumption
  M50401P, M50402P .... 25μA (typ)
  M50403P, M50404P, M50405P ..... 2μA (typ)
- Low operating voltage .... 1.1V (min)

APPLICATIONS

- Alarm clocks
- Precision clocks for electronic equipment
- Frequency dividers for electronic equipment

PIN CONFIGURATION (TOP VIEW)

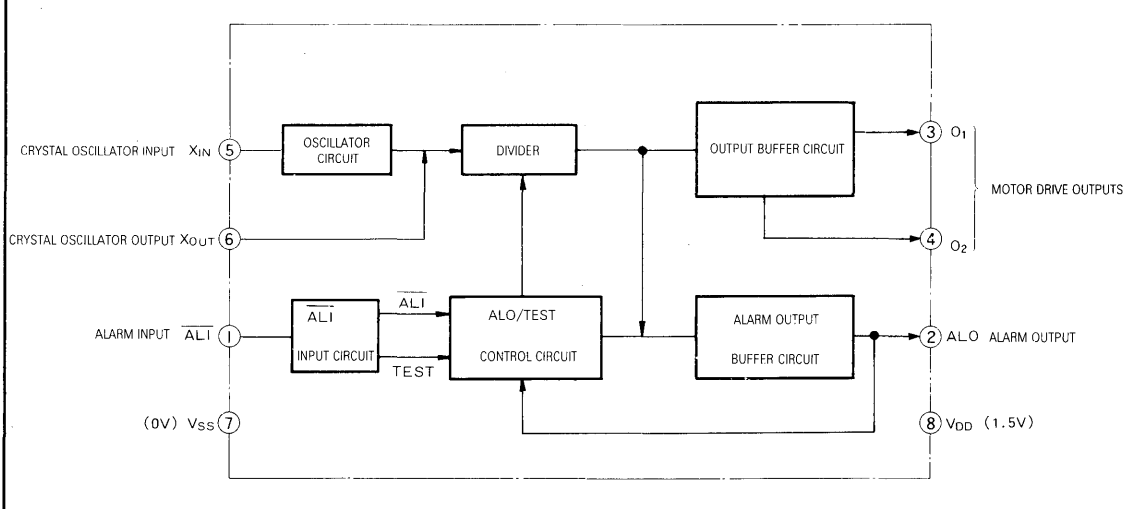


Outline 8P4

FUNCTION

The M50401P, M50402P, M50403P, M50404P and M50405P are CMOS ICs designed for use in crystal-controlled clocks. They consist of a crystal controlled oscillator, dividers, alarm bell drive output buffer circuit, and motor drive output buffer circuits. They are designed for use with standard 4,1943MHz or 32.768kHz clock reference crystals, and perform the required divisions to drive a stepping motor.

BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

OSCILLATOR CIRCUIT

A crystal oscillator element is connected between X_{IN} (oscillator input) and X_{OUT} (oscillator output) pins and capacitors are connected to ground from each of these pins.

OUTPUT BUFFER CIRCUIT

The output buffer circuit is an amplifier capable of producing a drive current from the output of the last divider stage. The O_1 output is 1-second shifted from the O_2 output. By connecting these to a stepping motor, the 1-second hand steps can be generated.

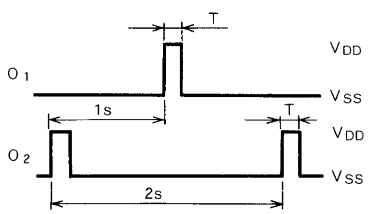
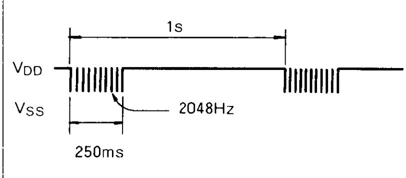
ALARM INPUT ($\overline{ALI}$)

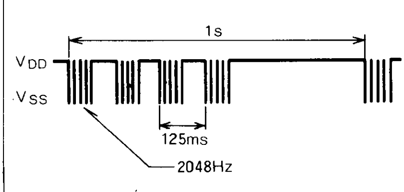
By setting the $\overline{ALI}$ pin to the value of V_{SS}, an alarm waveform is output from ALO. In addition, by setting the $\overline{ALI}$ pin to an intermediate value, a test state is achieved, and a 2048Hz signal is continuously output from ALO. If an intermediate level is applied to $\overline{ALI}$ while X_{IN} and X_{OUT} are set to V_{SS}, the last 12 stages of dividers can be fed from the ALO signal.

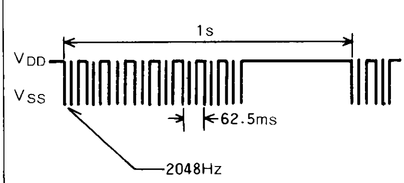
ALARM OUTPUT BUFFER CIRCUIT

The alarm output buffer circuit generates a drive signal for a piezoelectric element or magnetic speaker. The alarm output signals are 50% duty cycle 2048Hz, 2Hz, and 1Hz signals for the M50401P and M50403P, and 50% duty cycle 2048Hz, 8Hz, and 1Hz signals for the M50402P and M50404P. And 50% duty cycle 2048Hz, 16Hz and 1Hz signals for the M50405P.

Table 1 O_1, O_2, and ALARM OUT Pin Output Waveforms

<table>
<tr><th>Type</th><th>O_1 and O_2 output waveform</th><th>Pulse width T(ms)</th><th>ALARM OUT pin output waveform</th></tr>
<tr><td>M50401P
M50403P</td><td rowspan="3">
</td><td rowspan="2">31</td><td>

50% duty cycle at 2048Hz, 2Hz and 1Hz</td></tr>
<tr><td>M50402P
M50404P</td><td>

50% duty cycle at 2048Hz, 8Hz and 1Hz</td></tr>
<tr><td>M50405P</td><td>16</td><td>

50% duty cycle at 2048Hz, 16 Hz and 1Hz</td></tr>
</table>

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	−0.3~4	V
P_d	Power dissipation	Ta=25℃	300	mW
T_{opr}	Operating temperature		−20~70	℃
T_{stg}	Storage temperature		−65~150	℃

RECOMMENDED OPERATING CONDITIONS (T_a = 25°C, unless otherwise noted)

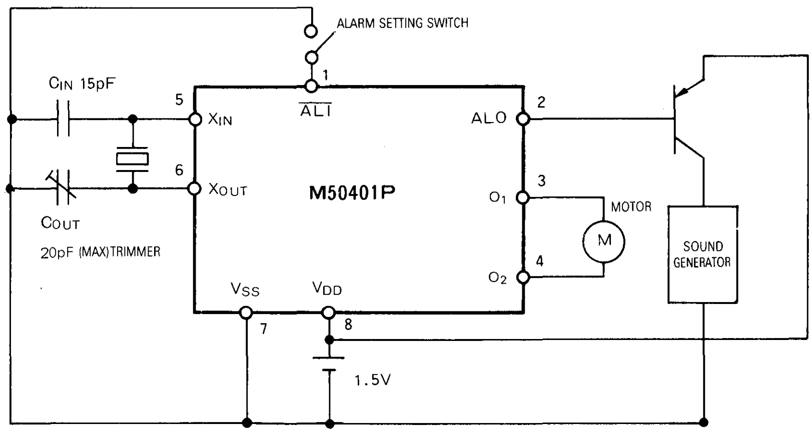
Symbol	Parameter		Limits Min	Limits Nom	Limits Max	Unit
V_{DD}	Supply voltage			1.5		V
V_{SS}	Supply voltage (GND)			0		V
f_{OSC}	Quartz crystal frequency	M50401P M50402P		4.1943		MHz
		M50403P M50404P M50405P		32.768		kHz
R_O	Quartz crystal impedance	M50401P M50402P		30	60	Ω
		M50403P M50404P M50405P		20		kΩ
C_{IN}	External input capacitor			15		pF
C_{OUT}	External output capacitor			15		pF

ELECTRICAL CHARACTERISTICS (T_a = 25°C, V_{SS} = 0V, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits Min	Limits Typ	Limits Max	Unit
V_{DD}	Supply voltage	M50401P M50402P	$C_{IN}=C_{OUT}=15pF$, $R_O=30\Omega$	1.1	1.5	1.8	V
		M50403P M50404P M50405P	$C_{IN}=C_{OUT}=15pF$, $R_O=20k\Omega$	1.1	1.5	1.8	V
I_{DD}	Supply current	M50401P M50402P	$V_{DD}=1.5V$, $C_{IN}=C_{OUT}=15pF$, $R_O=30\Omega$		25	35	μA
		M50403P M50404P M50405P	$V_{DD}=1.5V$, $C_{IN}=C_{OUT}=15pF$, $R_O=20k\Omega$		2	5	μA
$R_{ON(P+N)}$	Motor drive output saturation resistance (p-channel saturation resistance + n-channel saturation resistance)		$V_{DD}=1.2V$, $R_L=180\Omega$		55	75	Ω
$R_{ON(AL)}$	Alarm output saturation resistance (n-channel)		$V_{DD}=1.2V$, $I_{OUT}=1mA$		200	300	Ω
	Alarm output saturation resistance (p-channel)		$V_{DD}=1.2V$, $I_{OUT}=0.1mA$		1	4	kΩ
V_{INH}	Input voltage		$\overline{ALI}$ pin	V_{DD} −0.15		V_{DD}	V
V_{INL}	Input voltage		$\overline{ALI}$ pin	V_{SS}		0.15	V
R_{AH}	Input pullup resistance		$\overline{ALI}$ pin	5	15		kΩ
$\Delta f/f_0$	Oscillator frequency supply voltage dependence	M50401P M50402P	$\frac{f_{1.2}-f_{1.7}}{f_{1.5}}$ ALO pin			±1	ppm
		M50403P M50404P M50405P			±2		ppm

11

APPLICATION EXAMPLE



QUARTZ CRYSTAL
4.1943MHz
(32,768kHz)

CMOS LCD DIGITAL ALARM CLOCK CIRCUITS

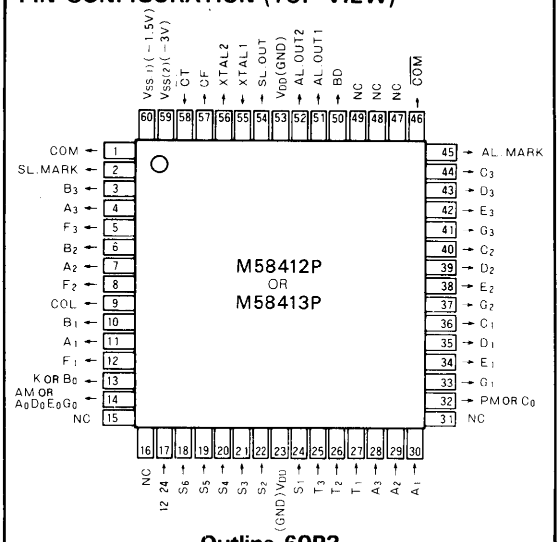
DESCRIPTION

The M58412P and M58413P CMOS aluminum-gated LSIs serve 4-digit liquid-crystal display (LCD) digital alarm clocks employing quartz oscillators of 4.2 MHz and 32 kHz respectively.

FEATURES

- Low current consumption. Under ordinary conditions, M58412P consumes 30μA at an oscillator frequency of 4.2 MHz and $V_{SS(1)}$ level of −1.5V, while M58413P consumes 2μA at 32 kHz and $V_{SS(1)}$ of −1.5V.
- The 12-hour clock-display function shows AM or PM hours and minutes; the 24-hour system shows hours and minutes alone.
- Separate switches enable independent setting of hours and minutes.
- Five alarm output signals are provided: a continuous alarm-bell signal, intermittent alarm-bell signal, external bell-oscillator-circuit-drive signal, external electronic-apparatus switching signal, and 12 min or 120 min DC signal.
- The alarm bell output can continue for up to 12 min.
- A 10 min 'snooze' function is incorporated.
- The LSI causes the whole display to flash on and off when battery voltage drops below the specified level.
- Two LCD mark outputs are provided: alarm and sleep. The display offers immediate indication of the function in current operation.
- The LSIs enable sleep and auto-recording timers to be set at any time during a 59-minute period. A 120-minute output mode is also available with auto-recording timers.

PIN CONFIGURATION (TOP VIEW)



Outline 60P2

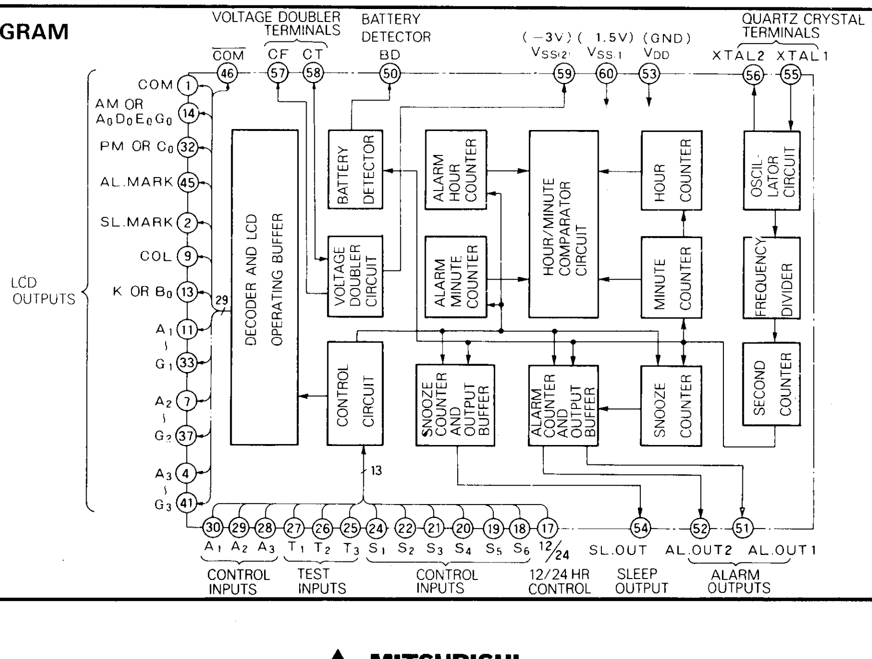
APPLICATIONS

- Alarm clocks with a 'snooze' function
- Sleep timers
- Travel watches
- Switching timers for electronic apparatus
- Auto-recording timers for audio equipment

FUNCTION

Normal clock, alarm clock, 'snooze' timer, sleep timer, electronic-apparatus switching timer, and audio-equipment auto-recording timer functions are provided by the oscillator and frequency divider (4.2 MHz for M58412P and 32 kHz for M58413P).

BLOCK DIAGRAM



OPERATION

The following figures and tables show the LCD-electrode arrays on the LCD panel, the segment codes, and the display modes.

An LCD-electrode arrangement and names of segments and marks (12-hour clock display)

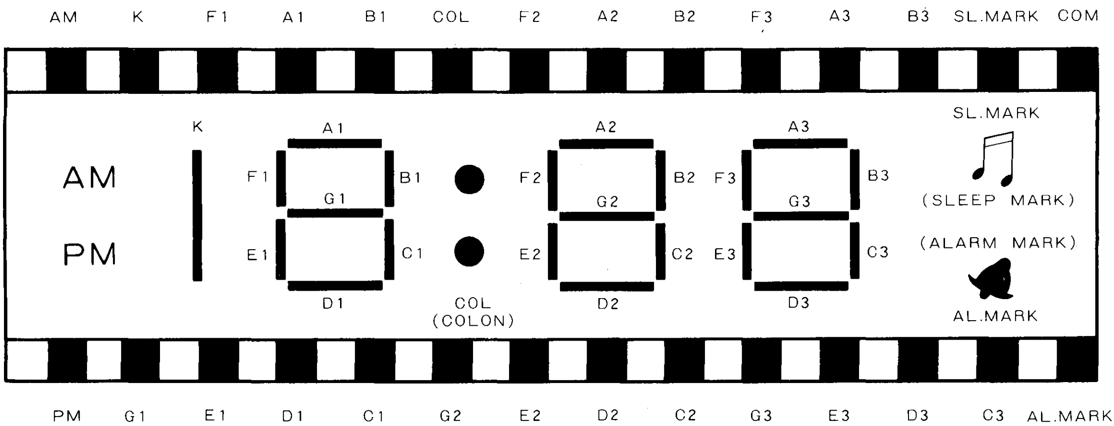
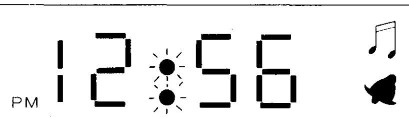
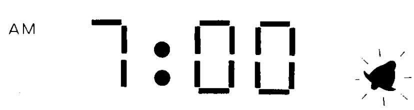
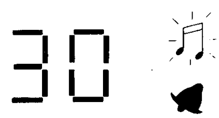


Table 1 12-Hour Clock Display

Mode	Display	Meaning of mark display
Normal clock ordinary display		Sleep timer is in operation. Alarm timer is being set.
Alarm		Alarm time is displayed; adjustment possible
Sleep-time		Sleep time is displayed; adjustment possible. Alarm timer is being set.

Note 1. The symbol ☼ indicates a 2sec on-off flash.

MITSUBISHI ELECTRIC

An LCD-electrode arrangement and names of segments and marks (24-hour clock display)

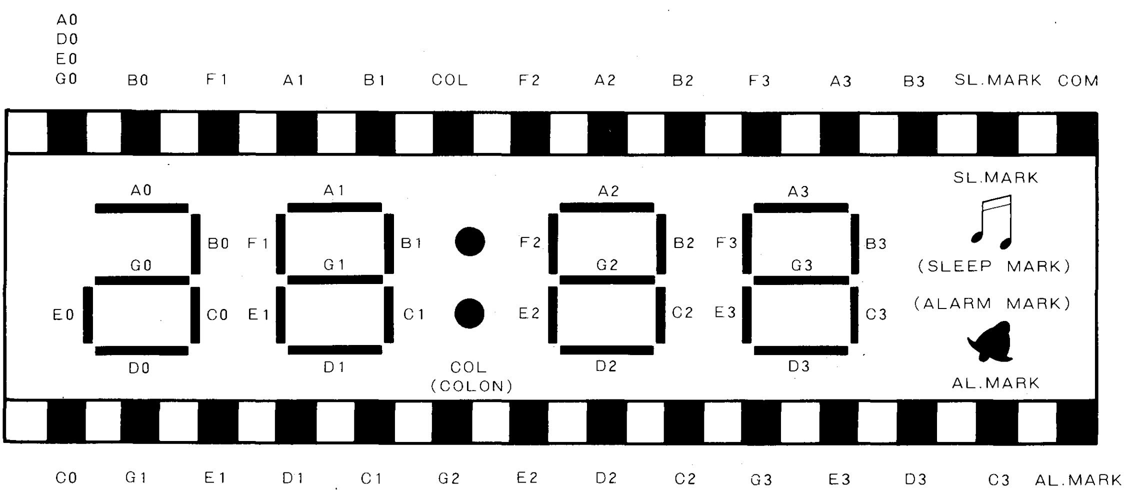


Table 2 24-Hour Clock Display

Mode	Display	Meaning of mark display
Normal display	20:56 (colon flashing), sleep mark, alarm mark	→ Sleep timer is in operation. → Alarm timer is being set.
Alarm display	7:00, alarm mark (flashing)	→ Alarm time is displayed; adjustment possible.
Sleep-time display	30, sleep mark (flashing), alarm mark	→ Sleep time is displayed; adjustment possible. → Alarm timer is being set.

Note 1 : The symbol ⁘ indicates a 2sec on-off flash.

FUNCTIONS OF INPUT AND OUTPUT PINS

Input Pins

The potential drop to the level of $V_{SS(1)}$ (−1.5V) or $V_{SS(2)}$ (−3V) achieved inside the LSI ensures that all the input pins are used in the floating-state condition. The input pins A_1, A_2, A_3, S_1, and S_6 have the potential of $V_{SS(1)}$, while all the other input pins have $V_{SS(2)}$. Signal input requires the use of the $V_{DD(GND)}$ level for all input pins.

S_1 Pin

Every push of the S_1 push-button switch advances 1 minute in normal clock-time adjustment, alarm-time setting and sleep-time setting. Raising to the hour digit is prohibited in this operation. In the normal-clock ordinary-display mode, the S_1 pin also serves as a start/stop input pin for the sleep timer. A sleep mark flashing on and off displays the sleep timer's operation. It will disappear when the sleep timer stops operation, or as soon as the time initially set on the sleep timer is reached.

S_2 Pin

Every push of the S_2 push-button switch advances 1 hour in normal clock-time adjustment and alarm-time setting. In the normal-clock ordinary-display mode, the S_2 pin also serves as an input pin to bring the sleep output to the $V_{SS(1)}$ level. This function makes it possible to switch off a radio or other electronic apparatus before the time initially set on the sleep timer is reached.

S_3 Pin

When the A_3 pin is held at the V_{DD} level, a momentary switch should be used to enable the input with the S_3-pin potential to change momentarily to the V_{DD} level. Pushing the S_3 switch changes the mode cyclically in the sequence: normal-clock ordinary display; alarm-time display (alarm-time setting is possible); and sleep-time display (sleep-time setting is possible). However, when the A_3 pin is in the floating state (the inside-LSI potential is −1.5V), it is recommended to use a lock switch to retain the V_{DD} level at the S_3 pin. While S_3–pin potential is kept at V_{DD}, the alarm-time display mode is effective (alarm-time setting is possible). Disconnecting the S_3 pin restores the normal-clock ordinary-display mode. The sleep-timer mode cannot be used when the A_3 pin is in the floating state. In this case, however, there are convenient applications (for travel watches, etc.) free from the problem of alarm-time lags behind the set time which sometimes arise from the use of momentary switches due to their accidental operation.

S_4 Pin

When the normal-clock normal-display mode of the basic clock is effective, maintaining this pin at the V_{DD} level causes entry to the normal-clock time-adjustment mode. After time adjustment with the S_1 and S_2 pins, clock operation starts with the '00' second of the adjusted time as soon as S_4 is disconnected from the V_{DD} level.

S_5 Pin

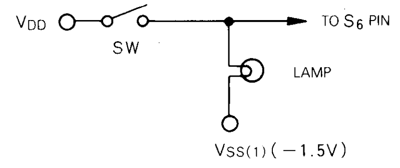
This pin is used to provide alarm-timer set input. Maintaining this input pin at the V_{DD} level causes the alarm mark to stay on. When the normal clock time coincides with the alarm time, two types of alarm output, AL. OUT1 and AL. OUT2, generate alarm signals. (The alarm signal with a pulse width of 250ms is generated only once after the coincidence takes place.) When cancellation of the alarm signal is desired, disconnecting the S_5 pin from the V_{DD} level causes both the alarm mark and alarm signal to disappear. No alarm signals will be generated when the normal clock time coincides with the alarm time, unless the S_5 pin is at the V_{DD} level.

S_6 Pin

This pin has three functions: 'snooze' timer-setting input, sleep-timer resetting input, and LCD lamp switching at night. When an alarm signal is generated in the normal-clock ordinary-display mode, bringing the S_6 potential momentarily to V_{DD} stops the alarm signal for a moment and generates it again after 9∼10 minutes. (The 1-pulse alarm signal with a 250ms pulse width cannot be generated again.) The 'snooze' function can be repeated at every signal input made to the S_6 pin. However, it does not operate after an alarm signal has continued for 12 minutes. This function is useful for 'snooze' clocks and other applications.

When no alarm signals are generated in the normal-clock ordinary-display mode, or when the 'snooze' function is not in operation, bringing the S_6 potential momentarily to V_{DD} makes it possible to reset the sleep time to 59 minutes and to make the sleep output level V_{DD}. This means that when a stereo or other apparatus connected is to be switched off after 59∼60 minutes, it is unnecessary to use the 59 minute setting in the sleep-time display mode: It is only necessary to push the S_6 push-button switch and then push the S_1-pin start button, giving great ease of operation. The S_6 pin, at a potential level of −1.5V, also serves as an LCD lamp power terminal at night (See Fig. 1). Care should be taken, however, over the fact that every time the LCD lamp is turned on a 'snooze' timer set input or sleep-timer/reset input is entered.

Fig. 1 An LCD lamp circuit



A_1 and A_2 Pins

AL. OUT1 pin alarm output in different modes is generated in accord with a combination of the A_1- and A_2-pin potentials. Table 3 shows four modes and their applications.

Table 3 AL. OUT1 Pin Alarm Output

A_1	A_2	Output waveform of AL. OUT1	Main applications
N.C.	N.C.	V_{DD}, $V_{SS(1)}$; 1024Hz; 1s, 1s (intermittent 1024Hz bursts)	Operation of bells, buzzers and other sound sources without oscillator circuits (intermittent sound)
V_{DD}	N.C.	V_{DD}, $V_{SS(1)}$; 1s, 1s (intermittent DC level)	Operation of sound sources with oscillator circuits (intermittent sound)
N.C.	V_{DD}	V_{DD}, $V_{SS(1)}$; 1024Hz (continuous)	Operation of sound sources without oscillation circuits (continuous sound)
V_{DD}	V_{DD}	V_{DD}, $V_{SS(1)}$; 250ms (single pulse)	Switching of electronic apparatus

A_3 Pin

This pin controls mode shifts between normal-clock ordinary-display mode, alarm-time display mode, and sleep-time display mode by the S_3 pin. When the A_3 pin is not connected (N.C.), it operates in the alarm-time display mode so long as the S_3 pin is at the V_{DD} level. When the S_3 pin is disconnected from the V_{DD} contact, the S_3 pin enters the normal-clock ordinary-display mode, but not the sleep-time display mode. When the A_3 pin is at the V_{DD} level, mode shifts occur cyclically in the sequence: normal-clock ordinary display; alarm-time display; sleep-time display; and normal-clock ordinary display, each time the S_3 pin is momentarily at the V_{DD} level.

12/24 Pin

Bringing the 12/24-hour pin to the V_{DD} level turns the 12-hour cycle display into the 24-hour cycle display.

T_1, T_2 and T_3 Pins

The T_3 pin is a clock-input pin for high-speed test use. Combinations of T_1 and T_2 pin potentials control the test mode and options, as shown in Table 4.

Table 4 Test Mode

T_1	T_2	Mode
N.C.	N.C.	Normal operation
V_{DD}	N.C.	Normal-clock ordinary display with the colon kept ON (without colon on-off flash)
N.C.	V_{DD}	The counter is reset and 12:00 AM (0:00 for 24-hour cycle) is displayed in the normal-clock ordinary-display mode. Here, the alarm time is 12:00 AM (0:00 for 24-hour cycle) and the sleep time is 59 minutes.
V_{DD}	V_{DD}	Carryover from the minute to the hour digits is prohibited. The common output is held at the $V_{SS(2)}$ level, and the segment and mark output for display is at the V_{DD} level. High-speed testing is possible.

OUTPUT PINS

Output Pins for Segments, COM, $\overline{COM}$ AL. MARK, and SL. MARK

The COM output pin common signal has a frequency of 32Hz. Segments and mark output pins which are not displayed give common signals, while segments and mark output pins displayed give inverse-phase signals of common signals. The $\overline{COM}$ output is used for permanently-displayed segments or marks.

AL. OUT1 (Alarm output 1) Pin

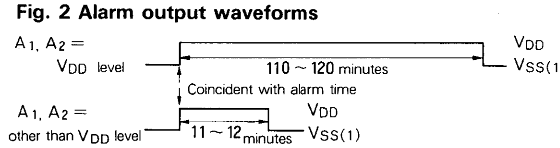
When the normal-clock ordinary-display time coincides with the alarm time, alarm signals with the waveforms shown in Table 3 are generated at the AL. OUT1 pin for 12 minutes. The 250 ms pulse-width alarm output, however, is given only once after the coincidence.

Coincidence in the alarm-time display mode causes the AL. OUT1 to be given for one minute. When they coincide in the normal-clock time-adjustment mode, continuous alarm signals are generated until the time is advanced.

AL. OUT2 (Alarm Output 2) Pin

When both the A_1 and A_2 pins are at the V_{DD} level (when the AL. OUT1 is the 250 ms pulse-width alarm output), the AL. OUT2 pin gives a DC output for 110~120 min. In cases of the alarm-time minute digit set to integral multiples of 10 minutes from 10 to 50 min., a DC output is sent out for 120 min. This signal is useful for controlling electronic apparatus for 2-hour auto-recording. When both the A_1 and A_2 pins are at other than the V_{DD} level, a DC output is given for 11~12 min by the AL. OUT2 pin.

Fig. 2 Alarm output waveforms

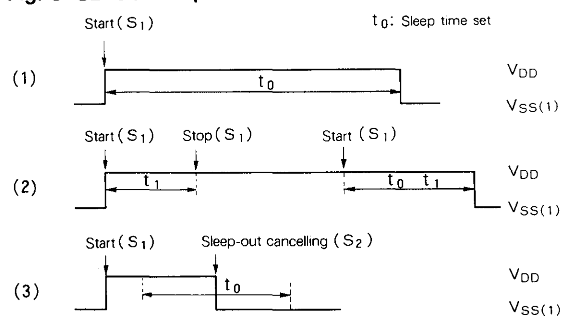


11

SL. OUT (Sleep Output) Pin

This pin can be used not only for sleep timers but also for turning on and off radios, TVs, cassette decks, and VTRs. In the normal-clock ordinary-display mode, the SL. OUT pin can be brought to the V_{DD} level, i.e., the switch-on stage, by starting the sleep timer with the S_1 pin, or by bringing the S_6 pin potential momentarily to V_{DD} after 12 minutes' issuance of the alarm signal or when the 'snooze' function is not in operation. As soon as the sleep time becomes 59 minutes in the normal-clock ordinary-display mode (the sleep timer does not display the time elapsed), or by bringing the S_2-pin potential momentarily to V_{DD} in the normal-clock ordinary-display mode, the switch-off state, i.e., the $V_{SS(1)}$ level, holds. Fig. 3 shows SL. OUT-pin output waveforms: (1) when the switched-off state is entered at the sleep time set; (2) when the timer is stopped after the start of the sleep timer and started again; and (3) when the switched-off state is entered before the sleep time set. Input pins to be used are shown in parentheses. When the sleep output is turned to the V_{DD} level by using the S_6 pin, this level is maintained unless the sleep timer is started with the S_1 pin. Use of the SL. OUT pin as a maximum 60-minute auto-recording pin requires that both the A_1 and A_2 pin potentials are set to V_{DD} and the AL. OUT1 pin is connected with the S_1 pin as shown in Fig. 7. In this case, sleep output assumes the V_{DD} level when the alarm time coincides with the normal time.

Fig. 3 SL. OUT output waveforms



POWER CIRCUITS

V_{DD}, $V_{SS(1)}$, $V_{SS(2)}$, CF, and CT Pins

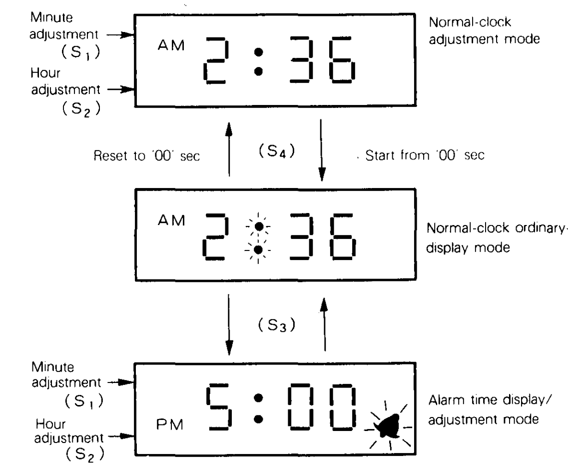
The electrical power supply is a 1.5V battery ($=V_{DD}-V_{SS(1)}$). Use of $0.1\mu F$ condensers between the CF and CT pins and between the $V_{SS(2)}$ and $V_{DD(GND)}$ pins gives voltage about double the power voltage, making possible direct operation of the LCD.

BD (Battery Detector) Pin

By connecting a resistor between the BD and $V_{SS(1)}$ pins which has a proper temperature characteristic and a resistance between $15k\Omega$ and 750Ω, the segments and marks displayed flash on and off in a 2sec period, a visual reminder of the necessity to replace the battery, when the battery voltage drops to any specified level in the detectable voltage range of $V_{DD}=-1.2\sim-1.5V$. This flashing can be stopped by making the S_6 potential momentarily V_{DD}. However, it will start again at the next sampling time (max. one minute later) until the battery is replaced.

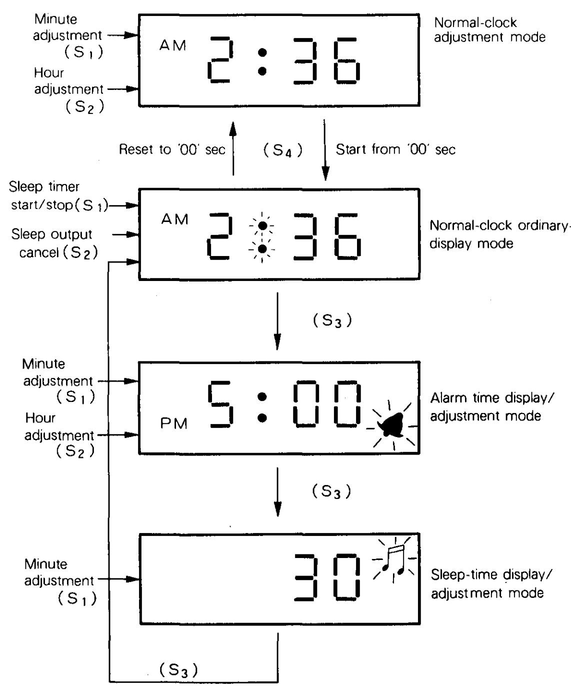
OPERATIONAL METHODS

Fig. 4 Operation when the A_3 Pin is N.C.



Note 2 : The symbol ※ shows a 2sec-period on-off flash.
3 : Lock switches are used in the S_3, S_4 and S_5 pins.

Fig. 5 Operation when the A_3, pin is at the V_{DD} level



Note 4 : The symbol ※ shows a 2sec-period on-off flash.
5 : Lock switches are used in the S_4 and S_5 pins.

MITSUBISHI ELECTRIC

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
$V_{SS(1)}$	Supply voltage	V_{DD} = GND Ta = 25°C	0.1 ~ −3	V
$V_{SS(2)}$	Supply voltage		0.1 ~ −7	V
$V_{I(1)}$	Input voltage for $V_{SS(1)}$ supply		$V_{SS(1)}$ ~ V_{DD}	V
$V_{I(2)}$	Input voltage for $V_{SS(2)}$ supply		$V_{SS(2)}$ ~ V_{DD}	V
Topr	Operating free-air ambient temperature range		−20 ~ 65	°C
Tstg	Storage temperature range		−30 ~ 80	°C

RECOMMENDED OPERATING CONDITIONS (Ta = 25°C, unless otherwise noted)

Symbol	Parameter		Conditions (Note 6)	Limits			Unit
				Min	Nom	Max	
$V_{SS(1)}$	Supply voltage	M58412P	C_{IN} = 15pF, C_{OUT} = 10pF, R_O = 20 Ω	−1.2	−1.5	−1.9	V
		M58413P	C_{IN} = 15pF, C_{OUT} = 30pF, R_O = 30k Ω	−1.1	−1.5	−2	V
$V_{SS(2)}$	Supply voltage	M58412P	C_{IN} = 15pF, C_{OUT} = 10pF, R_O = 20 Ω	−2.4	−3	−3.8	V
		M58413P	C_{IN} = 15pF, C_{OUT} = 30pF, R_O = 30k Ω	−2.2	−3	−4	V

ELECTRICAL CHARACTERISTICS

(Ta = 25°C, V_{DD} = GND, M58412P : f = 4.1943MHz, M58413P : f = 32.768kHz, unless otherwise noted)

Symbol	Parameter		Test conditions (Note 6)	Limits			Unit
				Min	Typ	Max	
I_{DD}	Supply current from V_{DD}	M58412P	$V_{SS(1)}$ = −1.5V, C_{IN} = 15pF, C_{OUT} = 10pF C_1 = C_2 = 0.1μF, R_O = 20 Ω		30	80	μA
		M58413P	$V_{SS(1)}$ = −1.5V, C_{IN} = 15pF, C_{OUT} = 30pF C_1 = C_2 = 0.1μF, R_O = 30k Ω		2	5	μA
$V_{I(OSC)}$	Oscillator input voltage	M58412P	C_{IN} = 15pF, C_{OUT} = 10pF, R_O = 20 Ω within 1sec of oscillation			−1.2	V
		M58413P	C_{IN} = 15pF, C_{OUT} = 30pF, R_O = 30k Ω within 5sec of oscillation			−1.2	V
$I_{OL(COM)}$	Low-level output current (common)		$V_{SS(2)}$ = −3V, V_{OL} = −2.9V	30			μA
$I_{OH(COM)}$	High-level output current (common)		$V_{SS(2)}$ = −3V, V_{OH} = −0.1V	−30			μA
$I_{OL(SEG)}$	Low-level output current (segment)		$V_{SS(2)}$ = −3V, V_{OL} = −2.9V	5			μA
$I_{OH(SEG)}$	High-level output current (segment)		$V_{SS(2)}$ = −3V, V_{OH} = −0.1V	−5			μA
$I_{OL(AL)}$	Low-level output current (alarm, sleep)		$V_{SS(1)}$ = −1.5V, V_{OL} = −1V	100			μA
$I_{OH(AL)}$	High-level output current (alarm, sleep)		$V_{SS(1)}$ = −1.5V, V_{OH} = −0.5V	−100			μA
I_{IL}	Low-level input current		$V_{SS(1)}$ = −3V, V_{IL} = −3V except for test input terminals			−0.2	μA
I_{IH}	High-level input current		$V_{SS(2)}$ = −3V, V_{IH} = 0V except for test input terminals			0.2	μA
$V_{O(2)}$	Doubler output voltage		$V_{SS(1)}$ = −1.5V, C_1 = C_2 = 0.1μF I_O = 2μA	−2.8			V
$V_{I(BD)}$	Battery detector voltage range		15k Ω ≦ R_{BD} ≦ 750k Ω	−1.2		−1.5	V

Note 6 : R_O refers to a crystal impedance.

APPLICATION EXAMPLES

Fig. 6 An alarm clock with 'snooze' and sleep functions

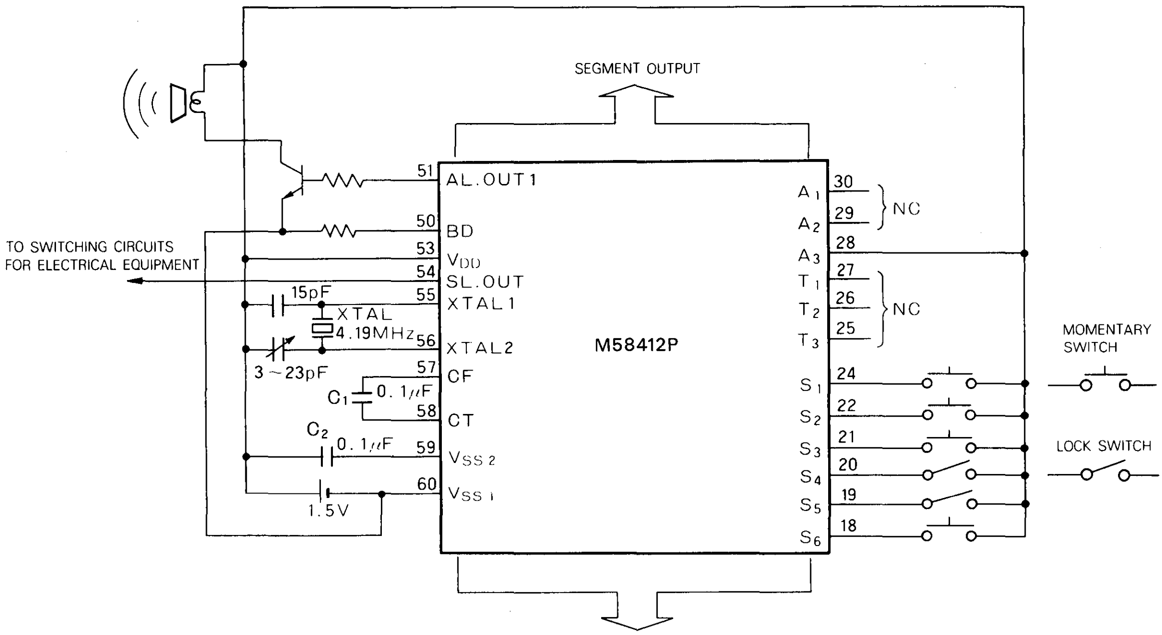
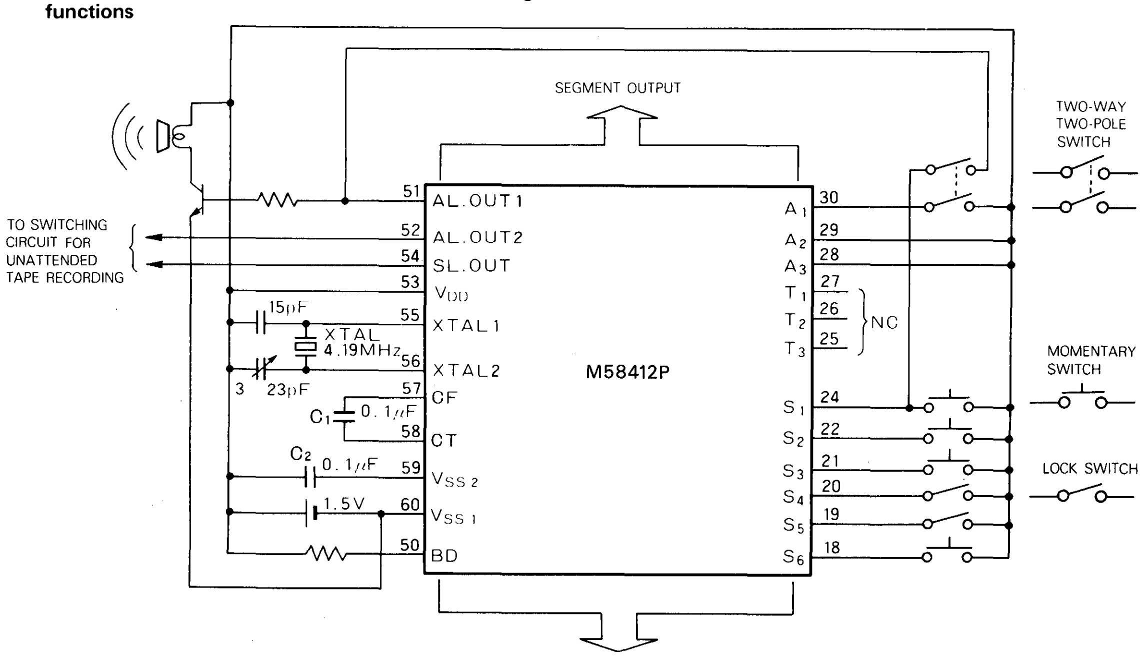


Fig. 7 An alarm clock with 'snooze' and auto-recording functions



Note 7 : The circuit of Fig. 6 gives intermittent alarm-bell tones.
8 : The circuit of Fig. 7 gives continuous alarm-bell tones.
9 : Use of Type M58413P in Fig. 6 and Fig. 7 requires the employment of a 32kHz quartz oscillator and a 5 ~ 35pF variable condenser.

Note 10 : Use is made of AL. OUT2 for 110 ~ 120 minute fixed-time auto-recording output and of the SL. OUT pin for maximum 60' minute non-fixed-time auto-recording output

M58435P
M58437-001P

CMOS ANALOG CLOCK CIRCUITS

DESCRIPTION

This family of CMOS circuits is particularly suited for crystal-controlled clocks where induction motors or stepping motors are used.

Type	Process	Crystal oscillator	Motor	Alarm sound
M58435P	Silicon-gate CMOS	4.1943MHz	Stepping motor	1024Hz
M58437 −001P	Aluminum-gate CMOS	32.768KHz	Stepping motor	4096×8×1Hz

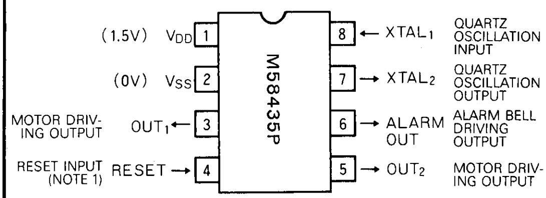
FEATURES

- Low power dissipation:
  M58435P: ········· 30μA (typ)
  M58437-001P: ········· 2μA (typ)
- Low voltage operation:
  M58435P: ········· 1.2V (min)
  M58437-001P: ········· 1.1V (min)
- Direct drive of ceramic resonator (M58437-001P only)

APPLICATIONS

- Crystal-controlled alarm clock
- Precision timepiece for electronic apparatus
- Frequency divider for electronic apparatus

PIN CONFIGURATION (TOP VIEW)



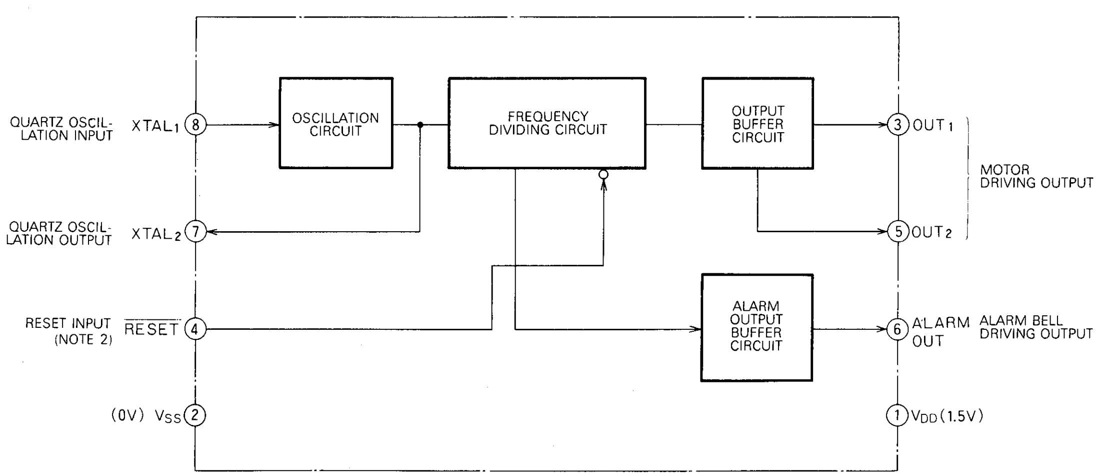
Note 1 : This pin is non-connected for M58434P.

Outline 8P1 (M58435P) (M58437-001P)

FUNCTION

Circuitry consists of an oscillator, frequency divider, bridge-type driver circuit for an induction motor or a stepping motor (M58435P, M58437-001P), and an alarm bell driver circuit. The oscillator frequency is 32.768kHz for the M58437-001P and 4.1943 MHz for the other types.

BLOCK DIAGRAM



Note 2 : M58434P has no reset input pin.

FUNCTIONAL DESCRIPTION

Oscillation Circuit

This circuit is completed by connecting a crystal between $XTAL_1$ (oscillation input) and $XTAL_2$ (oscillation output) and capacitances between both terminals and GND.

Motor Driver Circuit

This circuit amplifies motor driving current at the output frequency of the last divider. In M58435P,Outputs OUT_1 and OUT_2 are always in a mutually reversed phase, while in the M58437-001P, OUT_1 has a wave-form delayed 1sec from OUT_2. It is realized by continuous movement or stepped movement when the M58434P is connected to an induction motor (M58434), to a stepping motor with series-connected capacitance (M58435P) or a stepping motor (M58437-001P). The size of the capacitance for M58435P is determined by the total current consumption and the required motor torque, and with a 47μF capacitor, SUM-2 manganese dry cells will last for about one year.

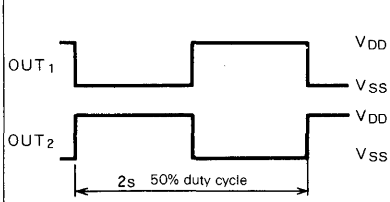
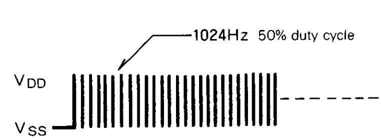
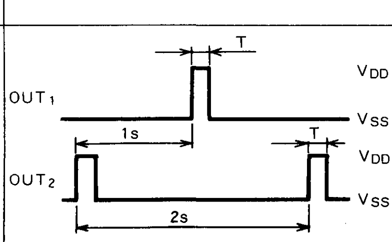
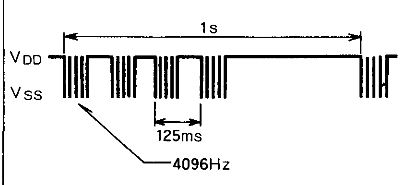
Reset Input ($\overline{\text{RESET}}$)

When the $\overline{\text{RESET}}$ terminal of the M58435P is held at V_{SS} level, outputs OUT_1 and OUT_2 hold their current states of that time, and invert 0.97~1.0sec after the reset terminal is released from the V_{SS} level. In the M58436-001P and M58437-001P, OUT_1 and OUT_2 go to the V_{SS} level, and 0.97~1.0sec after the reset terminal is released from V_{SS} level, a 31ms pulse is generated from the output opposite to the one that emitted a 31ms pulse immediately before the reset. If the $\overline{\text{RESET}}$ terminal is connected with the V_{SS} during the 31ms pulse, the reset will be started completely after the pulse ends. This prevents inadvertent interruption of complete action of the motor owing to the reset function. The M59434P has no reset function.

Alarm Output Buffer Circuit

This circuit consists of an N-channel open-drain MOS transistor and generates a signal to drive a ceramic resonator or magnetic speaker (see p. 10-14). The alarm output is a 1024 Hz signal, with a duty cycle of 50% for M5843P and M58435P, and burst signals of 4096Hz, 8Hz, and 1Hz, each of 50% duty, for M58437-001P. Direct drive of the ceramic resonator by M58437-001P is possible because of the high alarm output breakdown voltage.

Table 1 Output Waveforms on the OUT_1, OUT_2, and ALARM OUT terminals

Type	OUT_1 and OUT_2 waveform	Pulse width (ms)	ALARM OUT waveform
M58435P		——	
M58437-001P		31	 50% duty cycle for 4096Hz, 8Hz, and 1Hz

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	−0.3~5	V
Pd	Maximum power dissipation	Ta=25℃	300	mW
Topr	Operating free-air ambient temperature range		−20~70	℃
Tstg	Storage temperature range		−40~125	℃

RECOMMENDED OPERATING CONDITIONS (Ta=25℃, unless otherwise noted)

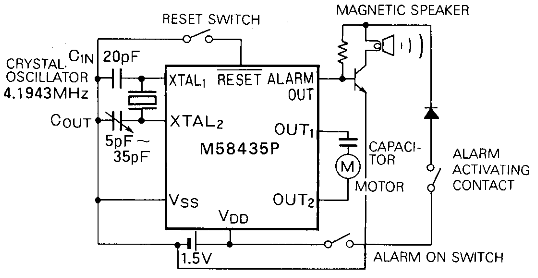
Symbol	Parameter		Limits			Unit
			Min	Nom	Max	
V_{DD}	Supply voltage			1.5		V
V_{SS}	Supply voltage (GND)			0		V
fosc	Crystal oscilation frequency	M58435P		4.1943		MHz
		M58437-001P		32.768		kHz
R_O	Crystal impedance of crystal oscillator	M58435P		30	60	Ω
		M58437-001P		20	30	kΩ
C_{IN}	External input capacity			20		pF
C_{OUT}	External output capacity			20		pF

ELECTRICAL CHARACTERISTICS (Ta=25℃, V_{SS}=0V, unless otherwise noted)

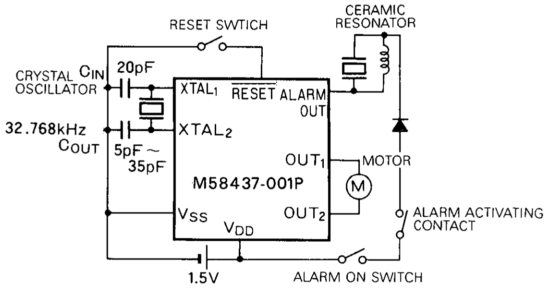
Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
V_{DD}	Supply voltage	M58435P	$C_{IN}=C_{OUT}=20pF$, $R_O=30\Omega$	1.2	1.5	1.9	V
		M58437-001P	$C_{IN}=C_{OUT}=20pF$, $R_O=20k\Omega$	1.1	1.5	1.9	V
I_{DD}	Supply current	M58435P	$V_{DD}=1.5V$, $C_{IN}=C_{OUT}=20pF$, $R_O=30\Omega$		30	50	μA
		M58437-001P	$V_{DD}=1.5V$, $C_{IN}=C_{OUT}=20pF$, $R_O=20k\Omega$		2	5	μA
$R_{ON(P+N)}$	Motor driving output saturation resistance (P-channel + N-channel)	M58435P	$V_{DD}=1.5V$, $I_{OUT}=\pm 3mA$		150	300	Ω
		M58437-001P	$V_{DD}=1.5V$, $I_{OUT}=\pm 3mA$		100	200	Ω
$R_{ON(AL)}$	Alarm bell driving output saturation resistance (N-channel)	M58435P	$V_{DD}=1.5V$, $I_{OUT}=3mA$		0.5	1	kΩ
		M58437-001P	$V_{DD}=1.5V$, $I_{OUT}=3mA$		100	200	Ω
I_{SW}	Reset input current	M58435P	$V_{DD}=1.5V$			1	μA
		M58437-001P				1	μA

APPLICATION EXAMPLES

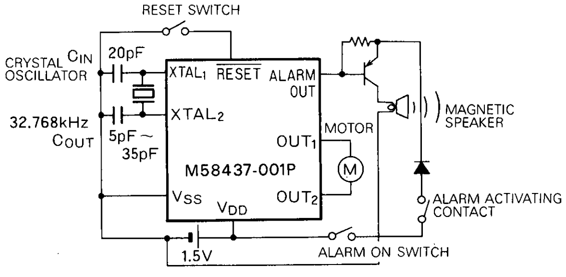
(1) Magnetic speaker with M58435P



(2) Ceramic buzzer with M58437-001P



(3) Magnetic speaker with M58437-001P



MITSUBISHI ELECTRIC

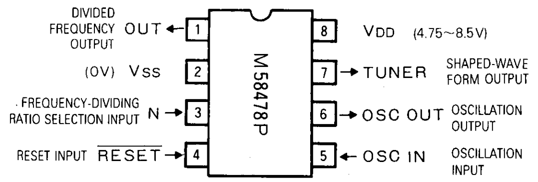
DESCRIPTION
The M58478P, M50121P, and M50122P are semiconductor integrated circuits which use aluminum-gate CMOS technology. The M58478P produces a frequency of 1/59719 or 1/88672, the M50121P produces a frequency of 1/58239 or 1/61425, and the M50122P produces a frequency of 1/86118 or 1/92077 of the input frequency.

FEATURES
- Usable as a crystal oscillator circuit
- Capable of handling small-amplitude input signals as low as $0.3V_{pp}$
- Frequency-dividing ratio selected through pin N
- Reset function
- Produces a shaped-waveform output of the same frequency as the input signal or oscillation output
- Derives a vertical scanning frequency from TV color subcarrier

APPLICATION
Frequency divider for VTR equipment.

FEATURES
The M58478P, M50121P, and M50122P have a programmable counter consisting of a 17-stage binary frequency divider which provides one of two frequency-dividing ratios as selected by the state of the N input.

PIN CONFIGURATION (TOP VIEW)

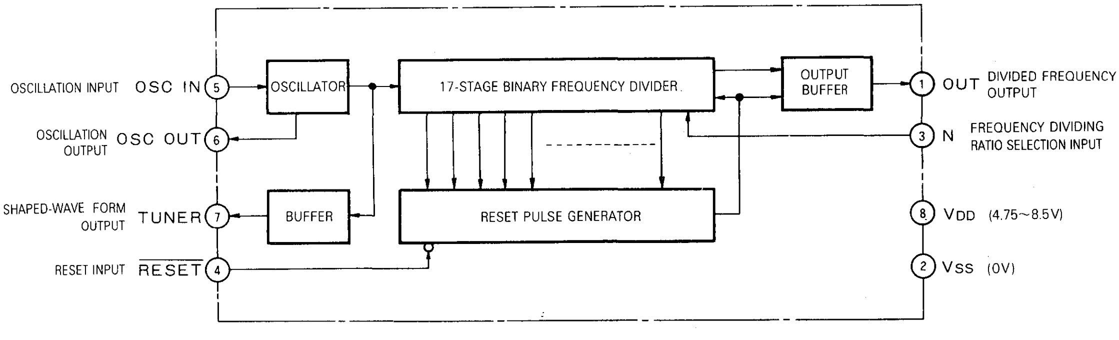


Outline 8P4 (M58478P)
(M50121P)
(M50122P)

Table 1 Input versus output frequencies

Type	Input frequency (MHz)	State of the N input	Output frequency (Hz)
M58478P	3.579545	H(open)	59.94
	4.433618	L	50.00
M50121P	3.579545	H(open)	61.46
		L	58.28
M50122P	4.433618	H(open)	51.48
		L	48.15

BLOCK DIAGRAM



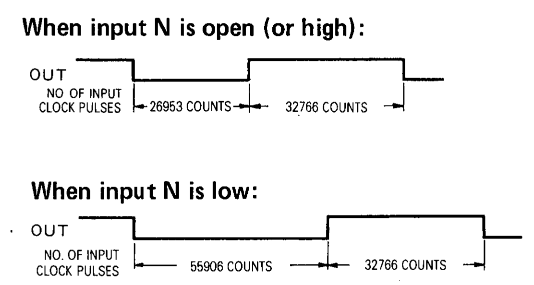
FUNCTIONAL DESCRIPTION

Crystal Oscillator

A crystal oscillator is configured by connecting a quartz resonator element between pins OSC IN and OSC OUT, and capacitances C_{L1} and C_{L0} between the two pins and V_{SS} (the feedback resistor included, on the chip). A built-in amplifier at the OSC IN pin enables even small amplitude signals to be input through a coupling capacitor.

Output Frequency

The frequency dividing ratio depends on the state of the N input. Table 2 summarizes the frequency dividing ratios and duty cycles as they are related to this N input. An example of a divided frequency output waveform is shown in Fig. 1.



Note 1. The frequency-dividing ration in the following cycle is determined by the state of N input in just before the output OUT changes from high to low.

Fig. 1 Waveforms of divided-frequency output (for the M58478P)

A shaped-waveform output of the same frequency as the input signal or oscillation frequency is available at the TUNER output.

Reset Function

When the $\overline{\text{RESET}}$ input is changed from high to low (edge triggered, active low input), the output OUT changes to low.

Pull-up Resistance

There are resistors at the N and $\overline{\text{RESET}}$ inputs, eliminating the need for external resistors. The standard resistance of the pull-up resistor is 20kΩ.

Frequency Dividing Ratio

The frequency-dividing ratio is determined by the data input of the programmable counter consisting of a 17-stage binary divider.

Special Frequency Dividing Ratios

It is possible to modify the frequency dividing ratios on special order. By changing one of the manufacturing processes, the data input of the programmable counter consisting of a 17-stage binary divider can be changed to enable any frequency-dividing ratio from 5 to 131071 (= $2^{17}-1$).

Table 2 Frequency-Dividing Ratios

Type	State of the N input	Frequency-dividing ratio	Divided frequency output low-level period	Divided frequency output high-level period
M58478P	H	59719	26953	32766
	L	88672	55906	32766
M50121P	H	58239	25473	32766
	L	61425	28659	32766
M50122P	H	86118	53352	32766
	L	92077	59311	32766

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	−0.3～9	V
V_I	Input voltage		$V_{SS} \leqq V_I \leqq V_{DD}$	V
Pd	Power dissipation	Ta=25°C	250	mW
Topr	Operating free-air temperature range		−30～70	°C
Tstg	Storage temperature range		−40～125	°C

RECOMMENDED OPERATING CONDITIONS (T_a=-30~70°C, unless otherwise noted)

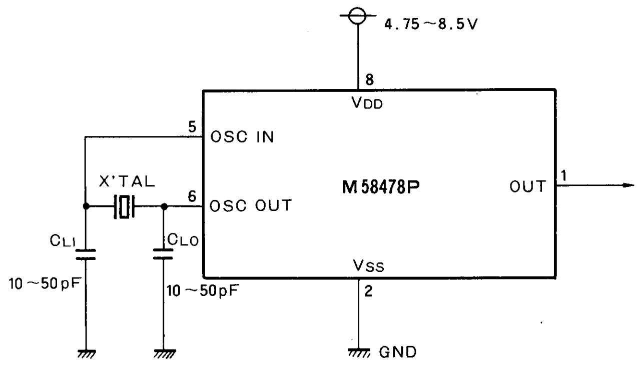
Symbol	Parameter	Limits Min	Limits Nom	Limits Max	Unit
V_{DD}	Supply voltage	4.75		8.5	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage	V_{DD}-0.5			V
V_{IL}	Low-level input voltage			0.5	V
V_I	Oscillation input amplitude voltage	0.3			V_{PP}
f	Input frequency with input N open		3.58	5.5	MHz
f	Input frequency with input N low		4.43	5.5	MHz

ELECTRICAL CHARACTERISTICS (T_a=25°C, V_{DD}=6.5V, V_{SS}=0V, f_{IN}=4.5MHz, unless otherwise noted)

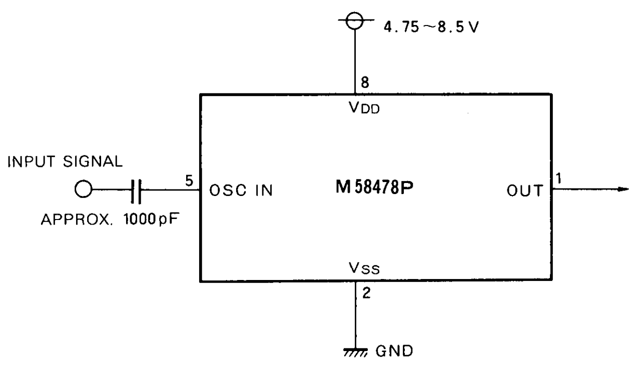
Symbol	Parameter	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
V_{DD}	Supply voltage	Ta=−30～70°C	4.75		8.5	V
I_{DD}	Supply current	N and $\overline{RESET}$ inputs and outputs open			5	mA
V_{IH}	High-level input voltage		V_{DD}-0.5			V
V_{IL}	Low-level input voltage				0.5	V
V_{OH}	High-level output voltage		V_{DD}-0.5			V
V_{OL}	Low-level output voltage				0.5	V
I_{OH}	High-level output current	$V_O=V_{SS}$	−2			mA
I_{OL}	Low-level output current	$V_O=V_{DD}$	2			mA
R_I	Pull-up resistance, N and $\overline{RESET}$ inputs			20		kΩ
V_I	Oscillation input amplitude voltage	V_{DD}=4.75V	0.3			V_{PP}
f_{MAX}	Maximum operating frequency	V_{DD}=4.75V	5.5			MHz

APPLICATION EXAMPLES

(1) Crystal Oscillator (with built-in feedback resistance)



(2) External Input Signal Connections



CMOS COUNTER/TIMERS

DESCRIPTION

The M58479P and M58482P are electronic timer ICs developed by aluminum-gate CMOS technology. Use of these ICs makes possible timer devices without mechanical elements, which have reduced power dissipation, superior reliability, and higher noise immunity. The M58479P is specifically designed for high noise immunity while the M58482P particularly features low power dissipation.

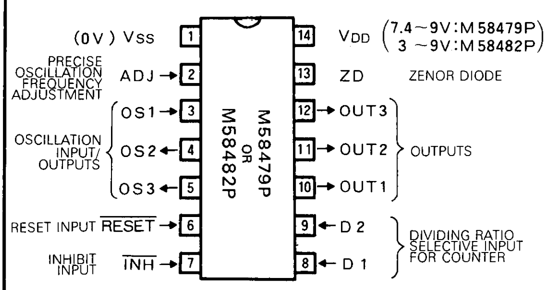
FEATURES

- Low power dissipation
  M58479P: 2mW (typ), 7.5mW (max)
  M58482P: 200μW (typ), 750μW (max)
- Superior noise immunity
- Single power supply with a zenor diode
- Internal RC oscillator
- Precise oscillation frequency regulating capability
- Extremely broad time-delay range (50ms~4800h)
- Time-delay settable to 10, 60, or 600 times fundamental time (1024 times oscillation period)
- M58479P has automatic-reset function during power engagement
- Built-in reset and inhibit functions
- Residual time display possible by adding Mitsubishi's M53290P and M53242P IC

APPLICATIONS

- Electronic timer or counter with broad time-delay range (50ms~4800h)

PIN CONFIGURATION (TOP VIEW)



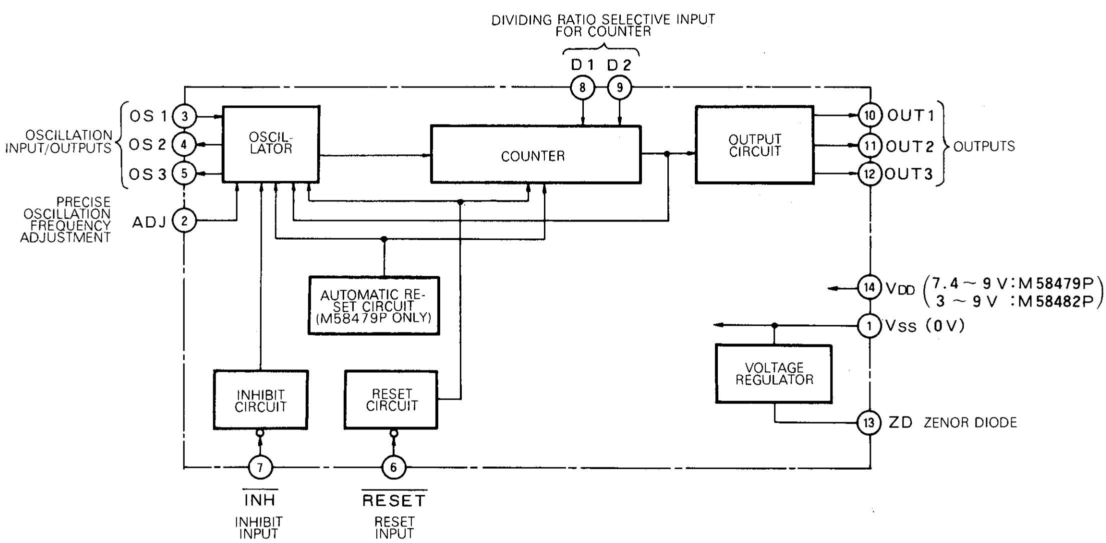
Outline 14P4

FUNCTION

These devices make possible extremely long clock performance, by counting pulse signals from the RC oscillator. It has precise oscillation frequency adjustment, automatic-reset, reset, and inhibit functions.

There are three outputs. When the time duration is up, OUT1 turns from low to high and OUT2 from high to low. OUT3 can be connected to M53290P and M53242P TTLs for residual time display.

BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

Voltage Regulator

A zenor diode is on-chip, making it easy to obtain a constant voltage regulator circuit. Since the zenor diode terminal (ZD) is independent of the power terminal (V_{DD}), it can be used as a constant voltage power supply for the total system.

Oscillator

Oscillation is obtained by connecting an external resistor (feedback resistor R_{FC}) between terminals OS1 and OS3 and an external capacitor (oscillation capacitor C_{FC}) between terminals OS1 and OS2. The values of the external resistor and capacitor can then be changed to vary the oscillation period and thus change the time delay. Oscillation period T_0 is obtained by the following equation:

$$T_0 = -R_{FC} \cdot C_{FC} \left\{ \ln \frac{V_{TR}}{V_{DD}+V_{BE}} + \ln \frac{V_{DD}-V_{TR}}{V_{DD}+V_{BE}} \right\} \cdots (1)$$

Where,

- R_{FC} : Resistance of external resistor
- C_{FC} : Capacitance of external capacitor
- V_{TR} : Transition voltage of the first inverter in the oscillation circuit
- V_{DD} : Supply voltage
- V_{BE} : Forward rising voltage of the diode in terminal OS1 (0.3~0.7V)

Automatic-Reset Function

The M58479P has a power-supply voltage-detection circuit on-chip, so that the counter is automatically reset by the rising edge of the supply voltage when power is turned on. The reset is then released, making the oscillator ready to function and the counter ready to start counting.

The M58482P can also be provided with the same automatic-reset function by connecting capacitor between terminals $\overline{\text{RESET}}$ and V_{SS}.

Reset Function

When the $\overline{\text{RESET}}$ input turns low (V_{SS}), oscillation of the oscillator can be stopped and the counter reset.

Inhibit Function

When terminal $\overline{\text{INH}}$ turns low (V_{SS}) while the timer is in action, the oscillation halts. When input $\overline{\text{INH}}$ is turned high or returned to OPEN afterwards, it starts to count residual time.

Counter

This counter consists of an 11-stage 1/2 frequency divider, a 2-stage 1/10 frequency divider and a 1-stage 1/6 frequency divider. As shown in the table below, timer duration can be changed by varying the number of pulses counted according to the combination of the input levels on terminals D1 and D2.

D1	D2	Number of pulses counted	Time delay	Typical time delay applied
H	H	1024	T_1	1 min
L	H	1024×10	$T_1 \times 10$	10 min
H	L	$1024 \times 10 \times 6$	$T_1 \times 10 \times 6$	1h
L	L	$1024 \times 10 \times 6 \times 10$	$T_1 \times 10 \times 6 \times 10$	10h

Where, $T_1 = T_0 \times 1024$

T_0 is the value obtained from equation (1)

Output Circuits

The chips have three outputs: OUT1 changes from low to high and OUT2 from high to low as soon as the time duration is up. Either can be used to drive a transistor by connecting it to the transistor base. OUT1 can drive a thyristor when connected to the thyristor gate.

OUT3 is an open-drain output with period 1/8 of the time delay, and can be used to drive a TTL in a separate (5V) power supply line. Thus, if a M53290P counter and a M53242P binary-to-decimal decoder are connected to OUT3, with their output connected to a light-emitting diode, residual time will be displayed on the LED. When not in use, OUT3 should be connected to V_{SS}.

Fine Adjustment of Oscillation Period

A variable resistor can be connected between terminals ADJ and V_{SS}, enabling precise adjustment of the period of the oscillator. However, when not used for fine adjustment, ADJ should be connected to V_{SS}.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	−0.3∼9.5	V
V_I	Input voltage		$V_{SS} \leq V_I \leq V_{DD}$	V
Pd	Maximum power dissipation	Ta=25°C	250	mW
Topr	Operating free-air temperature range		−30∼75	°C
Tstg	Storage temperature range		−40∼125	°C

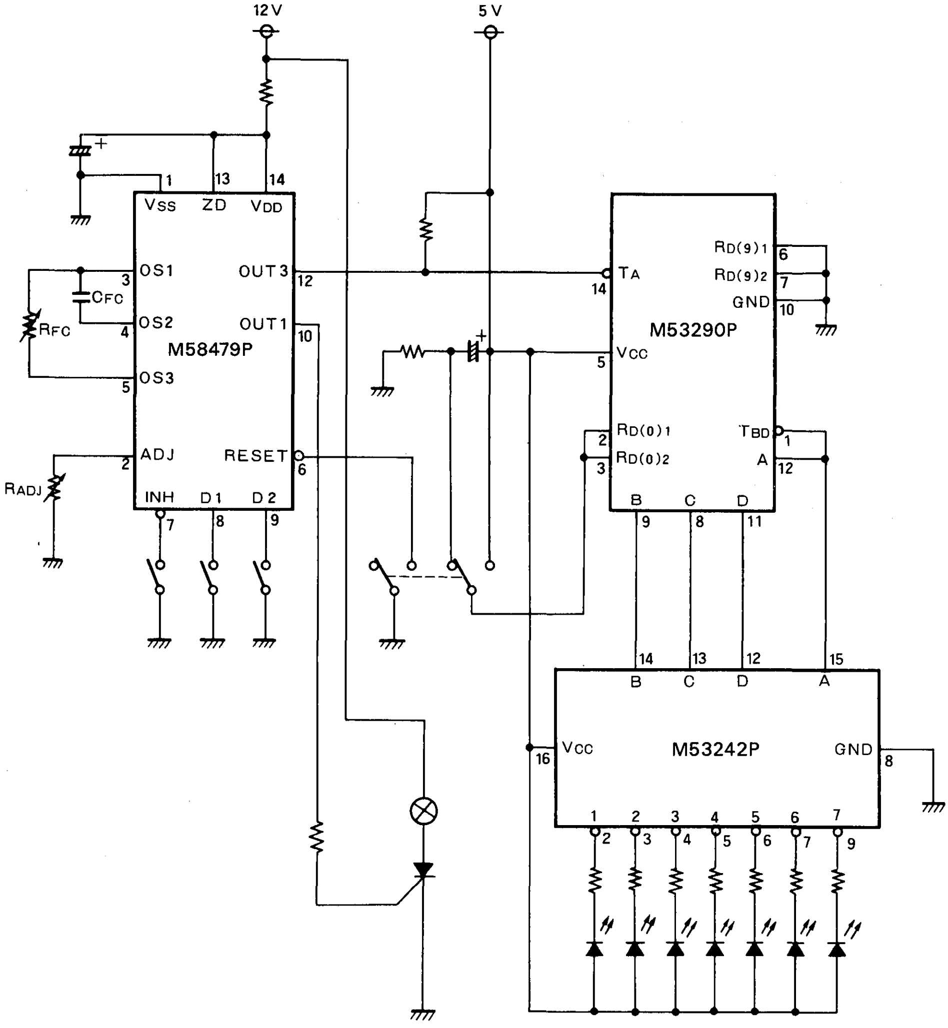
RECOMMENDED OPERATING CONDITIONS (Ta = −30∼75°C, unless otherwise noted.)

Symbol	Parameter		Limits			Unit
			Min	Nom	Max	
V_{DD}	Supply voltage	M58479P	7.4		9	V
		M58482P	3		9	V
I_{ZD}	Zenor current				10	mA
R_{FC}	Feedback resistance		0.005		10	MΩ
C_{FC}	Oscillation capacitance		0.001		1	μF
R_{FC}	Resistance for fine-adjustment of oscillation frequency		0		100	kΩ
V_{IH}	High-level input voltage, $\overline{RESET}$, $\overline{INH}$, D_1, D_2		$0.7 \times V_{DD}$	V_{DD}	V_{DD}	V
V_{IL}	Low-level input voltage, $\overline{RESET}$, $\overline{INH}$, D_1, D_2		0	0	$0.3 \times V_{DD}$	V

ELECTRICAL CHARACTERISTICS (Ta=25°C, unless otherwise noted.)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
V_{ZD}	Zenor voltage		I_{ZD}=2mA	7.4	8.2	9	V
			I_{ZD}=10mA	7.5	8.2	9	V
I_{DD}	Supply current	M58479P	V_{DD}=7.5V, C_{FC}=0.01μF, R_{FC}=1MΩ R_{ADJ}=0Ω, Input/output open		0.25	1	mA
		M58482P	V_{DD}=7.5V, C_{FC}=0.01μF, R_{FC}=1MΩ R_{ADJ}=0Ω, Input/output open		25	100	μA
V_{RE}	Supply voltage at the time of automatic-reset release	M58479P		3.1		5.4	V
V_{TR}	Transition voltage of first inverter in the oscillator		V_{DD}=7.5V, R_{ADJ}=0Ω	2.9		4.8	V
R_I	Pull-up resistance: $\overline{RESET}$, $\overline{INH}$, D1, D2 inputs	M58479P		10	20	30	kΩ
		M58482P		25	50	75	kΩ
I_{OH}	High-level output current, OUT1 and OUT2 outputs		V_{DD}=7.5V, Vo=0V	5	10		mA
I_{OL}	Low-level output current, OUT1, OUT2, and OUT3 outputs		V_{DD}=7.5V, Vo=7.5V	10	20		mA
I_{OZH}	Off-state output current, OUT3 output		V_{DD}=7.5V, Vo=7.5V			1	μA
I_{OL}	Low-level output current: OUT1, OUT2, and OUT3 outputs		V_{DD}=7.5V, Vo=0.4V	1.6			mA
I_{OL}	Low-level output current: OUT1, OUT2, and OUT3 outputs	M58482P	V_{DD}=4.5V, Vo=0.4V	1.6			mA
V_{OL}	Low-level output voltage: OUT1, OUT2, and OUT3 outputs		V_{DD}=7.5V			0.1	V

APPLICATION EXAMPLE



M58480P, M58484P

30-FUNCTION REMOTE-CONTROL TRANSMITTERS

DESCRIPTION

The M58480P and M58484P are 30-function remote-control transmitter circuits manufactured by aluminum-gate CMOS technology for use with in television receivers, audio equipment and the like, using infrared for transmission. They convey 30 different commands on the basis of a 6-bit PCM code. In the M58480P, entry priority is given to the first key pushed, while in the M58484P each key has an assigned priority. These transmitters are intended to be used in conjunction with an M58481, M58485P or M58487P receiver.

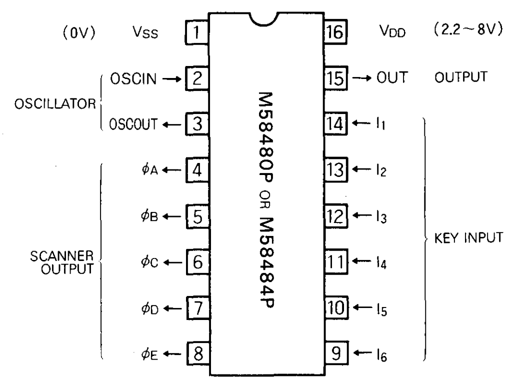
FEATURES

- Single power supply
- Wide supply voltage range: ························· 2.2V~8V
- Low power dissipation:
  Non-operating condition (V_{DD}=3V) :······3nW (typ)
  :······ 3μW (max)
- On-chip oscillator
- Low-cost LC/L or ceramic oscillator used in determining reference frequency (480 kHz or 455 kHz)
- Low external component count
- Low transmitter duty cycle (3.6%) for minimal power consumption

APPLICATIONS

- Remote-control transmitters for TV and other applications

PIN CONFIGURATION (TOP VIEW)

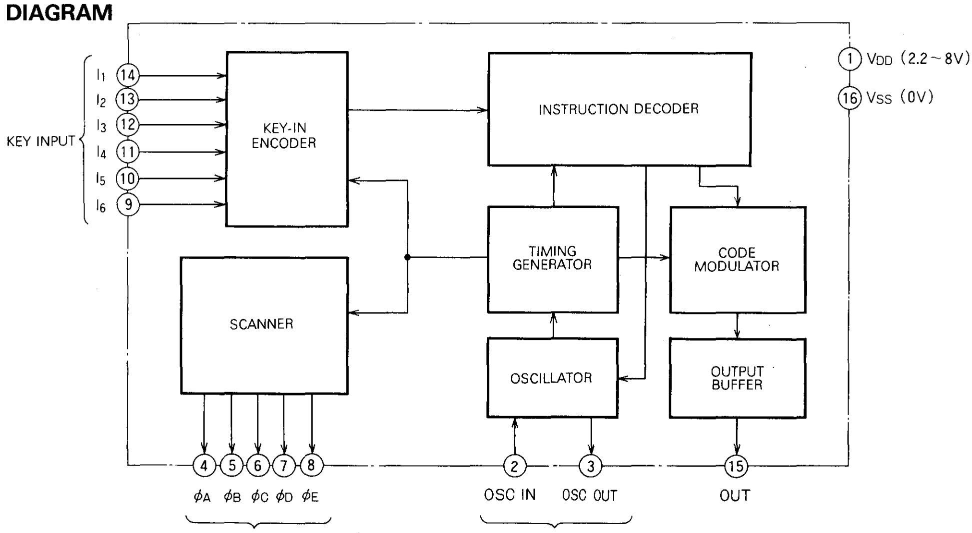


Outline 16P4

FUNCTION

The M58480P and M58484P transmitter circuits for infrared remote-control systems consist of an oscillator, a timing generator, a scanner, a key-in encoder, an instruction decoder, a code modulator, and an output buffer. With a 6 x 5 keyboard matrix, 30 commands can be transmitted by 6-bit PCM code. Oscillation is stopped when none of the keys are depressed, to minimize power consumption.

BLOCK DIAGRAM

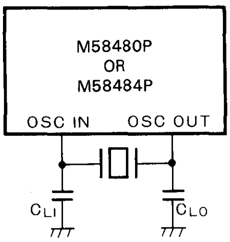


FUNCTIONAL DESCRIPTION

Oscillator

As the oscillator is on chip, oscillation frequency is easily obtained by connecting an external LC network or ceramic resonator between the OSC IN and OSC OUT terminals. Figs. 1 and 2 show typical oscillators.

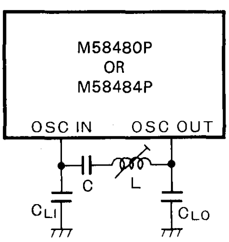
Fig. 1 An example of an oscillator (using a ceramic resonator)



CERAMIC RESONATOR : SFB455R OR CSB455A (MADE BY MURATA SEISAKUSHO)

C_{LI}, C_{LO} : 30pF (V_{DD}=3V)
50pF (V_{DD}=4.5, 6V)

Fig. 2 An example of an oscillator (using an LC network)



L : 2.5mH
C_{LI}, C_{LO} : 90pF
C : 0.1μF

Setting the oscillation frequency to 480 kHz (or 455kHz) will also set the signal transmission carrier wave to 40 kHz (or 38 kHz).

Power consumption is minimized by stopping oscillation in the oscillator when none of the keys are depressed.

Key Input

Thirty different commands can be input by a 6 x 5 keyboard matrix consisting of inputs $I_1 \sim I_6$ and scanner outputs ϕA~ϕE.

In the M58480P, key with first-key entry is given priority, and next-key entry is not allowed unless all keys are released.

In the M58484P, with assigned priority, simultaneous depression of more than two keys makes the key with higher priority effective. Order of key priority for scanner outputs is ϕA, ϕB, ϕC, ϕD, and ϕE, and in the same scanner output, I_1, I_2, I_3, I_4, I_5, and I_6.

When more than two keys are depressed at the same time, however, commands may not function due to short-circuiting among scanner outputs.

Table 1 shows the relationship between the keyboard matrix and the transmission commands.

Table 1 Relation between the keyboard matrix and the transmission commands

Key input \ Scanner output	ϕ_E	ϕ_D	ϕ_C	ϕ_B	ϕ_A
I_1	CH1	CH2	CH3	CH4	POWER ON/OFF
I_2	CH5	CH6	CH7	CH8	CH UP
I_3	CH9	CH10	CH11	CH12	CH DOWN
I_4	CH13	CH14	CH15	CH16	VO UP
I_5	BR UP	BR DOWN	BR 1/2	MUTE	VO DOWN
I_6	CS UP	CS DOWN	CS 1/2	CALL	VO 1/3

Transmission Commands

Table 2 shows the 30 commands that can be transmitted by 6-bit PCM codes ($D_1 \sim D_6$).

The code 000000 is not assigned for preventing error operations.

Table 2 Relation between the commands and the transmission codes

D_1	D_2	D_3	D_4	D_5	D_6	Function	Remarks
1	0	0	0	0	0	CH UP	
0	1	0	0	0	0	CH DOWN	
1	1	0	0	0	0	VO UP	
0	0	1	0	0	0	VO DOWN	Analog control
1	0	1	0	0	0	BR UP	
0	1	1	0	0	0	BR DOWN	
1	1	1	0	0	0	CS UP	
0	0	0	1	0	0	CS DOWN	
1	0	0	1	0	0	MUTE	
0	1	0	1	0	0	VO(1/3)	
1	1	0	1	0	0	BR(1/2)	Normalization of analog
0	0	1	1	0	0	CS(1/2)	
1	0	1	1	0	0	CALL	
0	1	1	1	0	0	POWER ON/OFF	
0	0	0	0	1	0	CH 1	
1	0	0	0	1	0	CH 2	
0	1	0	0	1	0	CH 3	
1	1	0	0	1	0	CH 4	
0	0	1	0	1	0	CH 5	
1	0	1	0	1	0	CH 6	
0	1	1	0	1	0	CH 7	
1	1	1	0	1	0	CH 8	Channels selected directly
0	0	0	1	1	0	CH 9	
1	0	0	1	1	0	CH 10	
0	1	0	1	1	0	CH 11	
1	1	0	1	1	0	CH 12	
0	0	1	1	1	0	CH 13	
1	0	1	1	1	0	CH 14	
0	1	1	1	1	0	CH 15	
1	1	1	1	1	0	CH 16	

Transmission Coding

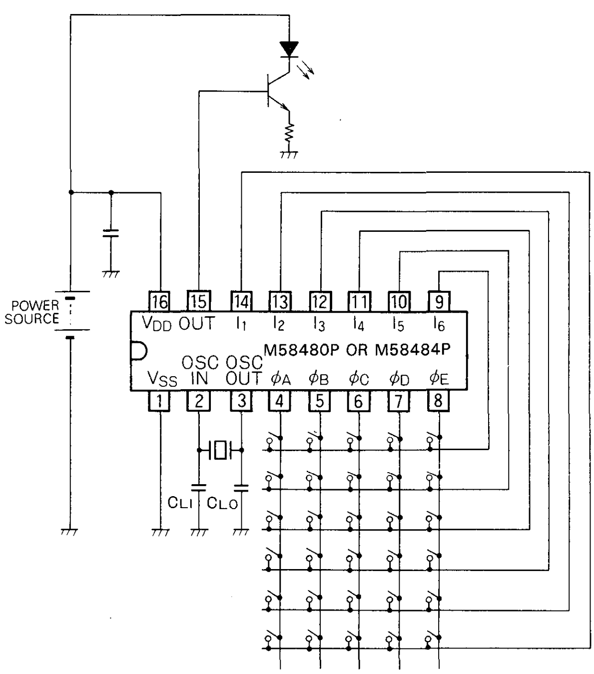
When oscillation frequency f_{OSC} is 480kHz, transmission of data code is executed as follows: when f_{OSC} is other than 480kHz, period is multiplied by 480kHz/f_{OSC} and its frequency by f_{OSC}/480kHz.

A single pulse is amplitude-modulated by a carrier of 40kHz, and the pulse width is 0.5ms. Therefore a single pulse consists of 20 clock pulses of 40kHz (see Fig. 3).

The distinction between "0" and "1" bits is made by the pulse interval between pulses, with a 2msec interval corresponding to "0", and a 4msec interval representing "1" (Fig. 4).

One command word is composed of 6 bits, that is, of 7 pulses, and it is transmitted in the 48ms cycle while a matrix switch is depressed.

APPLICATION EXAMPLE



As mentioned above, adoption of this code means that the period during which output is high (i.e. signal emitting LED is lit) is shorter than in continuous wave transmission. Indeed the LED is on for only half the 7-pulse period or 1.75ms, which is 3.6% of the 48ms entire cycle. This not only saves in total power consumption, but it also improves LED reliability. Put another way, emission can be increased on the same power consumption.

Fig. 3 A single pulse modulated onto carrier (40kHz)

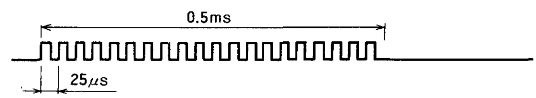


Fig. 4 Distinction between the bits "1" and "0"

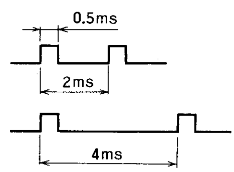
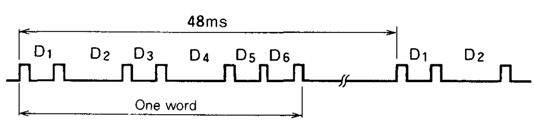


Fig. 5 Synthesis of one word (the code below shows 010100)



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	$-0.3 \sim 9$	V
V_I	Input voltage		$V_{SS} \leqq V_I \leqq V_{DD}$	V
V_O	Output voltage		$V_{SS} \leqq V_O \leqq V_{DD}$	V
P_d	Maximum power dissipation	Ta = 25℃	300	mW
T_{opr}	Operating free-air temperature range		$-30 \sim 70$	℃
T_{stg}	Storage temperature range		$-40 \sim 125$	℃

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	2.2		8	V
f_{OSC}	Oscillation frequency		455		kHz
			480		kHz
V_{IH}	High-level input voltage, $I_1 \sim I_6$	$0.7 \times V_{DD}$	V_{DD}	V_{DD}	V
V_{IL}	Low-level input voltage, $I_1 \sim I_6$	0	0	$0.3 \times V_{DD}$	V

ELECTRICAL CHARACTERISTICS (Ta = 25℃, unless otherwise noted)

Symbol	Parameter	Test conditions		Limits			Unit
				Min	Typ	Max	
V_{DD}	Operational supply voltage	Ta = −30 ~ 70℃, f_{OSC} = 455kHz		2.2		8	V
I_{DD}	Supply voltage during operation	f_{OSC} = 455kHz	V_{DD} = 3 V		0.1	0.5	mA
			V_{DD} = 6 V		0.5	2	mA
I_{DD}	Supply voltage during non-operation	V_{DD} = 3 V				1	μA
		V_{DD} = 8V				5	μA
R_I	Pull-up resistances, $I_1 \sim I_6$				20		kΩ
I_{OL}	Low-level output currents, $\phi_A \sim \phi_E$	V_{DD} = 3 V, V_O = 3 V		0.2	0.5		mA
		V_{DD} = 6 V, V_O = 6 V		1	2		mA
I_{OH}	High-level output current, OUT	V_{DD} = 3 V, V_O = 0 V		−5	−10		mA
		V_{DD} = 6 V, V_O = 0 V		−15	−30		mA

M58481P

30 FUNCTION REMOTE-CONTROL RECEIVER

DESCRIPTION

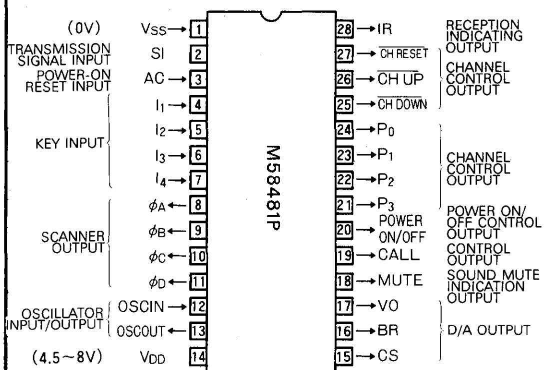
The M58481P is a 30-function remote-control receiver circuit manufactured by aluminum-gate CMOS technology for use in television receivers, audio equipment, and the like using infrared for transmission. It enables direct control of 16 functions at the receiver.

The M58481P is intended for use with an M58480P or M58484P transmitter.

FEATURES

- Single power supply
- Wide supply voltage range: 4.5V~8V
- Low power dissipation
- On-chip oscillator
- Low-cost LC or ceramic oscillator used in determining reference frequency (480 kHz or 455 kHz)
- Information is transmitted by pulse code modulation
- Good noise immunity—instructions are not executed unless same code is received three or more times in succession
- Single transmission frequency (40 kHz or 38 kHz) for carrier wave
- 16 TV channels selected directly
- Three analog functions—volume, brightness and color saturation—are independently controlled to 64 stages by three 6-bit D/A converters.
- 16 commands are controlled at the M58481P receiver as well
- Has large tolerance in operating frequency between the transmitter and the receiver
- Can be connected with an M51231P or equivalent touch-control channel selector.

PIN CONFIGURATION (TOP VIEW)



Outline 28P4

APPLICATION

- Remote-control receiver for TV or other applications

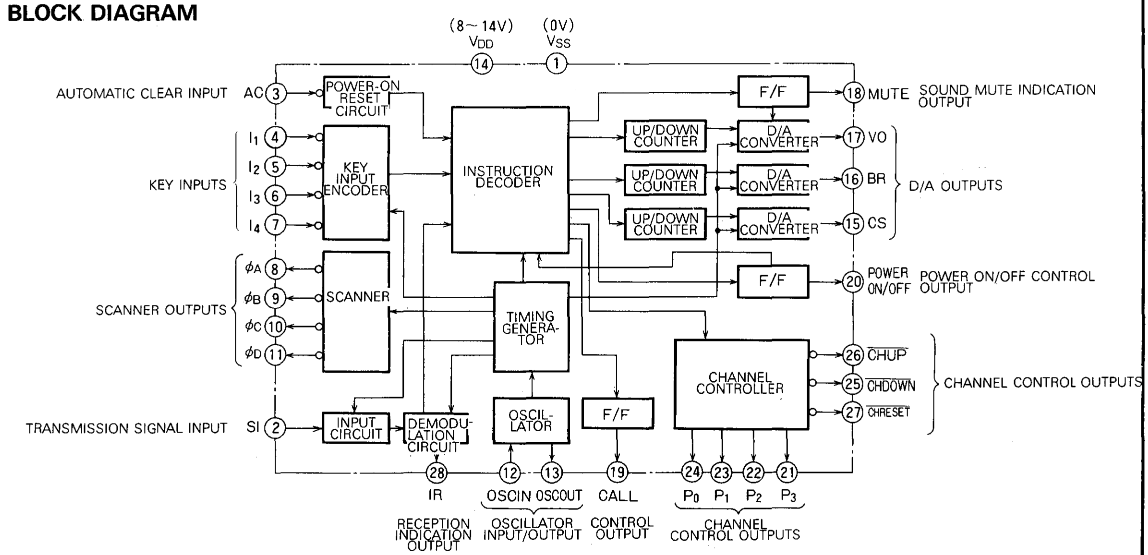
FUNCTION

The M58481P is designed to decode and execute instructions after three successive receptions of the identical instruction code, providing a good noise immunity.

Instructions comprise direct selection of 16 channels, channel position high and low, volume high and low, brightness high and low, color saturation high and low, normalization of volume, brightness and color saturation, sound mute on and off, TV main power on and off, and output CALL on and off.

In addition, 16 functional instructions can be entered from the receiver.

BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

Oscillator

As the oscillator is on-chip, oscillation frequency is easily obtained by connecting an external LC network or ceramic resonator between the OSC IN and OSC OUT terminals. Figs. 1 and 2 show typical oscillators.

Fig. 1 An example of an oscillator (using ceramic resonator)

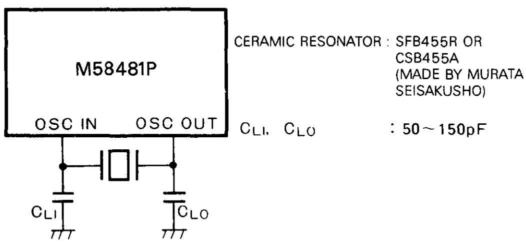
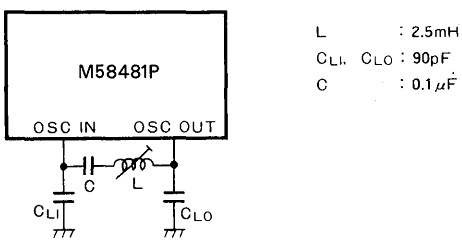


Fig. 2 An example of an oscillator (using LC network)



Reception Signal Input Circuit and Demodulation Circuit

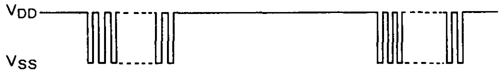
The reception signal caught by the photo detector is amplified in the amplifier and added to the SI, where it is converted into a pulse signal in the input circuit to be sent to the demodulation circuit. In the demodulation circuit, the pulse interval of the pulse signal is judged and then converted into the digital code to be sent to the instruction decoder.

SI is applied as amplified, either through a capacitor coupling (Fig. 3) or directly as a pulse signal (Figs. 4 and 5). A Schmitt trigger circuit is provided in the SI input circuit for preventing spurious operation due to noise.

Fig. 3 SI input waveform (when applied through a capacitor coupling)

Fig. 4 SI input waveform (when applied directly)

Fig. 5 SI input waveform (when applied directly)



Instruction Decoder

The instruction decoder starts to function after receiving the same instruction code three or more times in succession from the demodulation circuit.

Table 1 shows the relations between the reception code and instruction function. To prevent spurious operation, there is no code 000000.

Table 1 Relations between reception codes and instructions

Reception code						Function	Remarks
D_1	D_2	D_3	D_4	D_5	D_6		
1	0	0	0	0	0	CH UP	Channel up
0	1	0	0	0	0	CH DOWN	Channel down
1	1	0	0	0	0	VO UP	
0	0	1	0	0	0	VO DOWN	
1	0	1	0	0	0	BR UP	Analog control
0	1	1	0	0	0	BR DOWN	
1	1	1	0	0	0	CS UP	
0	0	0	1	0	0	CS DOWN	
1	0	0	1	0	0	MUTE	Sound mute on/off
0	1	0	1	0	0	VO(1/3)	
1	1	0	1	0	0	BR(1/2)	Normalization of analog control
0	0	1	1	0	0	CS(1/2)	
1	0	1	1	0	0	CALL	Output CALL on/off
0	1	1	1	0	0	POWER ON/OFF	Power on/off
0	0	0	0	1	0	CH 1	
1	0	0	0	1	0	CH 2	
0	1	0	0	1	0	CH 3	
1	1	0	0	1	0	CH 4	
0	0	1	0	1	0	CH 5	
1	0	1	0	1	0	CH 6	
0	1	1	0	1	0	CH 7	
1	1	1	0	1	0	CH 8	Channels selected directly
0	0	0	1	1	0	CH 9	
1	0	0	1	1	0	CH 10	
0	1	0	1	1	0	CH 11	
1	1	0	1	1	0	CH 12	
0	0	1	1	1	0	CH 13	
1	0	1	1	1	0	CH 14	
0	1	1	1	1	0	CH 15	
1	1	1	1	1	0	CH 16	

Key Inputs

16 different instructions can be input by a 4 x 4 keyboard matrix consisting of inputs $I_1 \sim I_6$ and scanner outputs $\phi A \sim \phi E$. Protection is also available against chattering within 10ms.

Entry priority is given to the first key depressed, and subsequent key entry is not allowed unless all keys are released. When two or more keys are depressed at the same time, scanner outputs may short-circuit, disabling all functions.

While one of the keys is depressed, instructions from the transmitter are ignored.

Table 2 Relations between keyboard matrix and instructions

Scanner output / Key input	ϕD	ϕC	ϕB	ϕA
I_1	CH RESET	CH DOWN	CH UP	POWER ON/OFF
I_2	MUTE	VO DOWN	VO UP	VO(1/3)
I_3	VO(1/3) BR(1/2) CS(1/2)	BR DOWN	BR UP	BR(1/2)
I_4	CALL	CS DOWN	CS UP	CS(1/2)

Indication of Reception

As soon as an identical code is received three times, output IR turns from low-level to high-level. Thus reception of an instruction from the transmitter can be indicated by an LED connected to output IR. Table 2 shows the relations between the keyboard matrix and the instructions.

Analog Outputs (VO, BR, CS)

As three 6-bit D/A converters are contained internally, three kinds of analog values can be controlled to 64 stages independently. The D/A converters are pulse-width modulator, the repetition frequency is 1.25kHz (when f_{OSC}= 480kHz) and minimum pulse width is 12.5μs.

Analog values can be incremented/decremented at a rate of about 1 step/0.1sec through the remote control or key input. The time required for increasing the analog value from the minimum to the maximum is about 6.6 seconds (when f_{OSC}=480kHz).

It is also possible to set the analog values to 1/3 (VO), 1/2 (BR, CS) of these maximum values by means of the remote control or the key input (normalization).

Sound Mute

Sound mute on/off is controlled through the remote control or the key input. When sound mute is on, output VO goes low, and output MUTE goes high.

Sound mute is automatically released from ON when VO is either incremented or decremented by remote control or the key input.

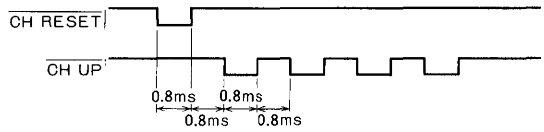
Channel Control

It is possible to employ either of two channel-control methods: parallel control by outputs $P_0 \sim P_3$, and serial control by outputs $\overline{\text{CH UP}}$, $\overline{\text{CH DOWN}}$, and $\overline{\text{CH RESET}}$.

In parallel control, a 4-bit address corresponding to a selected channel number appears at output $P_0 \sim P_3$. Table 3 shows the relation between channel numbers and outputs $P_0 \sim P_3$.

In serial control, a single pulse appears on the output $\overline{\text{CH RESET}}$ first, and then the pulses whose number is deducted by one from the selected channel number appear on the output $\overline{\text{CH UP}}$, as shown in Fig. 6. Up and down

Fig. 6 Timing chart of serially controlled channel selection (when fosc =480kHz)



channel switching, is controlled by a single pulse appearing at output $\overline{\text{CH UP}}$ or $\overline{\text{CH DOWN}}$, allowing connection to the M51231P or equivalent touch-control channel selector IC.

During direct channel selection or up-down channel switching, output VO goes low for 25~50ms.

Table 3 Relations between channel number and address output $P_0 \sim P_3$.

Channel number	Address outputs			
	P_0	P_1	P_2	P_3
1	0	0	0	0
2	1	0	0	0
3	0	1	0	0
4	1	1	0	0
5	0	0	1	0
6	1	0	1	0
7	0	1	1	0
8	1	1	1	0
9	0	0	0	1
10	1	0	0	1
11	0	1	0	1
12	1	1	0	1
13	0	0	1	1
14	1	0	1	1
15	0	1	1	1
16	1	1	1	1

Power On/Off

The remote control or the key input makes it possible to turn the POWER ON/OFF output from low to high or vice versa, effecting on/off control of the TV set.

While POWER ON/OFF is low, all channel and analog controls through the remote control are disabled, as are all through the keyboard, except CH RESET ($\phi D \sim I_1$), VO (1/3), BR (1/2), and CS (1/2) ($\phi D \sim I_3$).

Output CALL

The output CALL is turned high or low by remote control or the key input. This output effects on/off control of channel number indication or change of receiving modes of multi-channel broadcasting.

Power-on Reset

Attaching a capacitor to terminal AC activates the power-on reset function when power is on to the M58481P.

Activation of the power-on reset function sets outputs VO, BR, and CS to 1/3, 1/2, and 1/2, respectively, of their maximum value, turns POWER ON/OFF and CALL outputs low, and turns outputs $P_0 \sim P_3$ to 0000.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	−0.3∼9	V
V_I	Input voltage		$V_{SS} \leq V_I \leq V_{DD}$	—
V_O	Output voltage		$V_{SS} \leq V_O \leq V_{DD}$	—
P_d	Maximum power dissipation	Ta=25℃	300	mW
T_{opr}	Operating free-air temperature range		−30∼70	℃
T_{stg}	Storage temperature range		−40∼126	℃

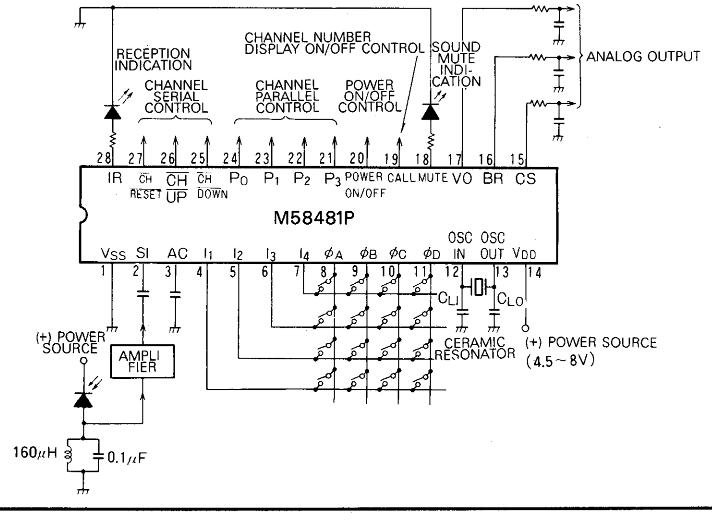
RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	4.5		8	V
fosc	Oscillation frequency		455		kHz
			480		kHz
V_I	Input voltage, SI	3			V_{P-P}
V_{IH}	High-level input voltage, I_1∼I_4	0.7×V_{DD}	V_{DD}	V_{DD}	V
V_{IL}	Low-level input voltage, I_1∼I_4	0	0	0.3×V_{DD}	V

ELECTRICAL CHARACTERISTICS (Ta=25℃, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{DD}	Operating supply voltage	Ta=−30∼70℃, f_{OSC}=455kHz	4.5		8	V
I_{DD}	Supply current	V_{DD}=5V, f_{OSC}=455kHz		0.4	1	mA
		V_{DD}=8V, f_{OSC}=455kHz		1.5	3	mA
R_I	Pull-up resistors, I_1∼I_4			20		kΩ
I_{OL}	Low-level output currents, ϕ_A∼ϕ_D	V_{DD}=8V, V_O=8V	3			mA
I_{OL}	Low-level output currents, $\overline{CH\ UP}$, $\overline{CH\ DOWN}$, $\overline{CH\ RESET}$	V_{DD}=8V, V_O=8V	15			mA
I_{OZH}	Off-state output currents, $\overline{CH\ UP}$, $\overline{CH\ DOWN}$, $\overline{CH\ RESET}$	V_{DD}=8V, V_O=8V			1	μA
I_{OH}	High-level output currents, P_0∼P_3	V_{DD}=8V, V_O=0V	−0.5			mA
I_{OL}	Low-level output currents, P_0∼P_3	V_{DD}=8V, V_O=8V	15			mA
I_{OH}	High-level output currents, VO, BR, CS	V_{DD}=8V, V_O=0V	−5			mA
I_{OL}	Low-level output currents, VO, BR, CS	V_{DD}=8V, V_O=8V	10			mA
I_{OH}	High-level output currents, POWER ON/OFF, CALL, MUTE	V_{DD}=8V, V_O=0V	−15			mA
I_{OL}	Low-level output currents, POWER ON/OFF, CALL, MUTE	V_{DD}=8V, V_O=8V	3			mA
I_{OH}	High-level output current, IR	V_{DD}=8V, V_O=0V	−10			mA
I_{OL}	Low-level output current, IR	V_{DD}=8V, V_O=8V	3			mA

APPLICATION EXAMPLE



MITSUBISHI ELECTRIC

29-FUNCTION REMOTE-CONTROL RECEIVER

DESCRIPTION

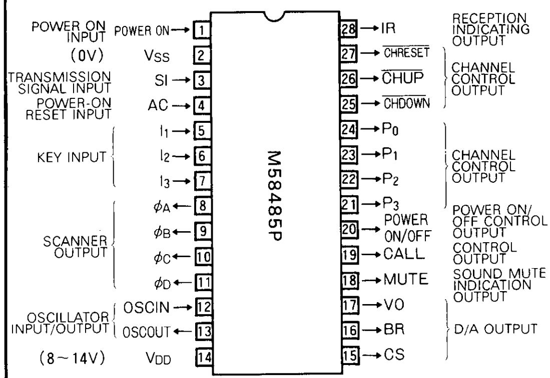
The M58485P is a 29-function remote-control receiver circuit manufactured by aluminum-gate CMOS technology for use in television receivers, audio equipment, and the like using infrared for transmission. It enables direct control of 12 functions at the receiver.

The M58485P is intended for use with an M58480P or M58484P transmitter.

FEATURES

- Single power supply
- Wide supply voltage range: 8V ~ 14V
- Low power dissipation
- On-chip oscillator
- Low-cost LC or ceramic oscillator used in determining reference frequency (480 kHz or 455 kHz)
- Information is transmitted by pulse code modulation
- Good noise immunity—instructions are not executed unless the same code is received three or more times in succession
- Single transmission frequency (40 kHz or 38 kHz) for carrier wave
- 16 TV channels selected directly
- Three analog functions—volume, brightness, and color saturation—are independently controlled to 64 stages by three 6-bit D/A converters.
- 12 instructions are controlled at the M58485P receiver, as well.
- Has large tolerance in operating frequency between the transmitter and the receiver
- Can be connected with an M51231P or equivalent touch-control channel selector

PIN CONFIGURATION (TOP VIEW)



Outline 28P4

APPLICATION

- Remote-control receiver for TV or other applications

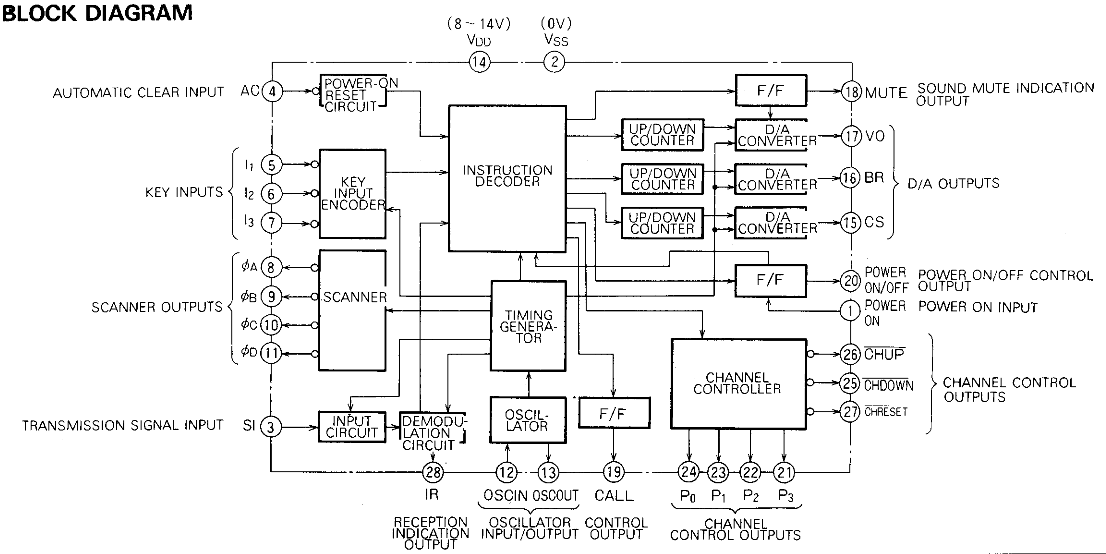
FUNCTION

The M58485P is designed to decode and execute instructions after three successive receptions of the identical instruction code, providing a good noise immunity.

Instructions comprise direction selection of 16 channels, channel position high and low, volume high and low, brightness high and low, color saturation high and low, normalization of volume, brightness and color saturation, sound mute on and off, TV main power on and off, and output CALL on and off.

In addition, 12 functional instructions can be entered from the receiver.

BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

Oscillator

As the oscillator is on-chip, oscillation frequency is easily obtained by connecting an external LC network or a ceramic resonator between the OSC IN and OSC OUT terminals. Figs. 1 and 2 show typical oscillators.

Fig. 1 An example of an oscillator (using ceramic resonator)

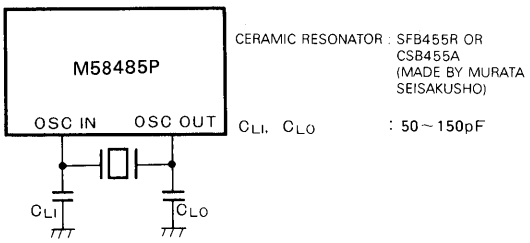
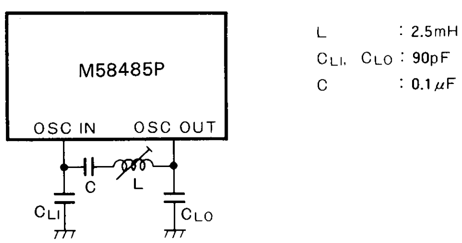


Fig. 2 An example of an oscillator (using LC network)



Reception Signal Input Circuit and Demodulation Circuit

The reception signal caught by the photo detector is amplified in the amplifier and added to the SI, where it is converted into a pulse signal in the input circuit to be sent to the demodulation circuit. In the demodulation circuit, the pulse interval of the pulse signal is judged and then converted into the digital code to be sent to the instruction decoder.

SI is applied as amplified either through a capacitor coupling (Fig. 3) or directly as a pulse signal (Figs. 4 and 5). A Schmitt trigger circuit is provided in the SI input circuit for preventing spurious operation due to noise.

Fig. 3 SI input waveform (when applied through a capacitor coupling)

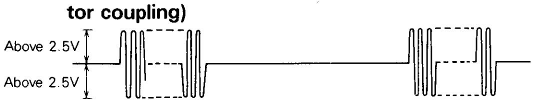


Fig. 4 SI input waveform (when applied directly)

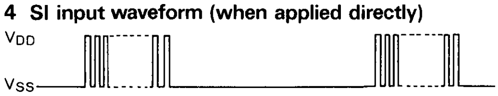
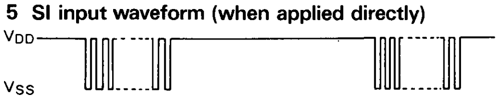


Fig. 5 SI input waveform (when applied directly)



Instruction Decoder

The instruction decoder starts to function after receiving the same instruction code three or more times in succession from the demodulation circuit.

Table 1 shows the relations between the reception code and instruction function. To prevent spurious operation, there is no code 000000.

Table 1 Relations between reception codes and instructions

Reception code						Function	Remarks
D_1	D_2	D_3	D_4	D_5	D_6		
1	0	0	0	0	0	CH UP	Channel up
0	1	0	0	0	0	CH DOWN	Channel down
1	1	0	0	0	0	VO UP	
0	0	1	0	0	0	VO DOWN	
1	0	1	0	0	0	BR UP	Analog control
0	1	1	0	0	0	BR DOWN	
1	1	1	0	0	0	CS UP	
0	0	0	1	0	0	CS DOWN	
1	0	0	1	0	0	MUTE	Sound mute on/off
0	1	0	1	0	0	VO(1/3)	Normalization of analog control
1	1	0	1	0	0	BR(1/2), CS(1/2)	
1	0	1	1	0	0	CALL	Output CALL on/off
0	1	1	1	0	0	POWER ON/OFF	Power on/off
0	0	0	0	1	0	CH 1	
1	0	0	0	1	0	CH 2	
0	1	0	0	1	0	CH 3	
1	1	0	0	1	0	CH 4	
0	0	1	0	1	0	CH 5	
1	0	1	0	1	0	CH 6	
0	1	1	0	1	0	CH 7	
1	1	1	0	1	0	CH 8	
0	0	0	1	1	0	CH 9	Channels selected directly
1	0	0	1	1	0	CH 10	
0	1	0	1	1	0	CH 11	
1	1	0	1	1	0	CH 12	
0	0	1	1	1	0	CH 13	
1	0	1	1	1	0	CH 14	
0	1	1	1	1	0	CH 15	
1	1	1	1	1	0	CH 16	

Key Inputs

It is possible to input 12 different instructions by the 3 x 4 keyboard matrix consisting of inputs $I_0 \sim I_3$ and scanner outputs $\phi A \sim \phi D$. Protection is also available against chattering within 10ms.

As entry priority is given to each key, depression of more than two keys at the same time makes the key with higher priority effective. For the scanner output, priority is given in the order of ϕA, ϕB, ϕC, and ϕD, and in the order of I_1, I_2, and I_3 if scanner output is the same. When two or more keys are depressed at the same time, scanner outputs may short-circuit, disabling all functions.

While one of the keys is depressed, instructions from the transmitter are ignored.

Table 2 shows the relations between the keyboard matrix and the commands.

Table 2 Relations between keyboard matrix and instructions

Key input \ Scanner output	ϕ_D	ϕ_C	ϕ_B	ϕ_A
I_1	CH UP	VO UP	BR UP	CS UP
I_2	CH DOWN	VO DOWN	BR DOWN	CS DOWN
I_3	POWER ON/OFF	MUTE	VO(1/3) BR(1/2) CS(1/2)	CALL

Indication of Reception

As soon as an identical code is received three times, the output IR turns from low-level to high-level. Thus reception of a command from the transmitter can be indicated by an LED connected to output IR.

Analog Outputs (CO, BR, CS)

As three 6-bit D/A converters are contained internally, three kinds of analog values can be controlled to 64 stages independently. The D/A converters are pulse-width modulator, and the repetition frequency is 1.25 kHz (when f_{OSC}= 480 kHz) and minimum pulse width is 12.5μs.

Analog values can be incremented/decremented at a rate of about 1 step/0.1 sec through the remote control or the key input. The time required for increasing the analog value from the minimum to the maximum is about 6.6 seconds (when f_{OSC}=480 kHz).

It is also possible to set the analog values to 1/3 (VO), 1/2 (BR, CS) of these maximum values by means of the remote control or the key input (normalization).

Sound Mute

Sound mute on/off is controlled through the remote control or the key input. When sound mute is on, output VO goes low, and output MUTE goes high.

Sound mute is automatically released from ON when VO is either incremented or decremented by remote control or the key input.

Channel Control

It is possible to employ either of two channel control methods: parallel control by outputs $P_0 \sim P_3$, and serial control by outputs $\overline{\text{CH UP}}$, $\overline{\text{CH DOWN}}$, and $\overline{\text{CH RESET}}$.

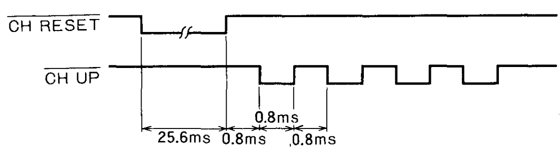
In parallel control, a 4-bit address corresponding to a selected channel number appears at output $P_0 \sim P_3$. Table 3 shows the relations between channel numbers and outputs $P_0 \sim P_3$.

In serial control, a single pulse appears on the output $\overline{\text{CH RESET}}$ first, and then the pulses whose number is deducted by one from the selected channel number appear on the output $\overline{\text{CH UP}}$, as shown in Fig. 6. Up and down channel switching is controlled by a single pulse appearing at output $\overline{\text{CH UP}}$ or $\overline{\text{CH DOWN}}$, allowing connection to the M51231P or equivalent touch-control channel selector IC.

Table 3 Relations between channel number and address output $P_0 \sim P_3$.

Channel number	Address outputs			
	P_0	P_1	P_2	P_3
1	0	0	0	0
2	1	0	0	0
3	0	1	0	0
4	1	1	0	0
5	0	0	1	0
6	1	0	1	0
7	0	1	1	0
8	1	1	1	1
9	0	0	0	1
10	1	0	0	1
11	0	1	0	1
12	1	1	0	1
13	0	0	1	1
14	1	0	1	1
15	0	1	1	1
16	1	1	1	1

Fig. 6 Timing chart of serially controlled channel selection (when fosc =480kHz)



During direct channel selection or up-down channel switching, output VO goes low for 25~50ms.

Outputs, $\overline{\text{CH UP}}$, $\overline{\text{CH DOWN}}$, $\overline{\text{CH RESET}}$, and $P_0 \sim P_3$, are the open-drain type of N-channel transistor.

Power on/off

The remote control or the key input makes it possible to turn the POWER ON/OFF output from low to high or vice versa, and it is possible to change the POWER ON/OFF output from low to high by means of the POWER ON input.

While POWER ON/OFF is low, all channel and analog controls through the remote control are disabled, as are all through the keyboard.

Output CALL

The output CALL is turned high or low by remote control or the key input. This output effects on/off control of channel number indication or change of receiving modes of multi-channel broadcasting.

Power-on Reset

Attaching a capacitor to terminal AC activates the power-on reset function when power is on to the M58485P.

Activation of the power-on reset function sets outputs VO, BR, and CS to 1/3, 1/2, and 1/2, respectively, of their maximum value, turns POWER ON/OFF and CALL outputs low and turns outputs $P_0 \sim P_3$ to 0000.

29-FUNCTION REMOTE-CONTROL RECEIVER

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	−0.3∼15	V
V_I	Input voltage		$V_{SS} \leq V_I \leq V_{DD}$	—
V_O	Output voltage		$V_{SS} \leq V_O \leq V_{DD}$	—
P_d	Maximum power dissipation	Ta=25℃	300	mW
T_{opr}	Operating free-air temperature range		−30∼70	℃
T_{stg}	Storage temperature range		−40∼125	℃

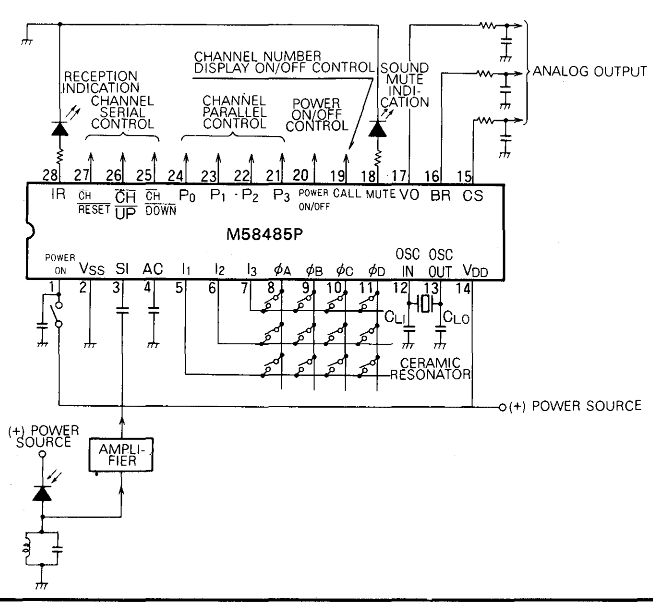
RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	8	12	14	V
f_{OSC}	Oscillation frequency		455		kHz
			480		kHz
V_I	Input voltage	5			V_{P-P}
V_{IH}	High-level input voltage, I_1∼I_3	$0.7 \times V_{DD}$	V_{DD}	V_{DD}	V
V_{IL}	Low-level input voltage, I_1∼I_3	0	0	$0.3 \times V_{DD}$	V

ELECTRICAL CHARACTERISTICS (Ta=25℃, V_{DD}=12V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{DD}	Supply voltage	Ta=−30∼70℃, f_{OSC}=455kHz	8	12	14	V
I_{DD}	Supply current	f_{OSC}=455kHz		2	5	mA
R_I	Pull-up resistance, I_1∼I_3			20		kΩ
I_{OL}	Low-level output currents, ϕ_A∼ϕ_D	V_O=12V	5			mA
I_{OL}	Low-level output currents, $\overline{CH\ UP}$, $\overline{CH\ DOWN}$, $\overline{CH\ RESET}$	V_O=12V	20			mA
I_{OZH}	Off-state output currents, $\overline{CH\ UP}$, $\overline{CH\ DOWN}$, $\overline{CH\ RESET}$	V_O=12V			1	μA
I_{OL}	Low-level output currents, P_0∼P_3	V_O=12V	20			mA
I_{OZH}	Off-state output currents, P_0∼P_3	V_O=12V			1	μA
I_{OH}	High-level output currents, VO, BR, CS	V_O=0V	−7			mA
I_{OL}	Low-level output currents, VO, BR, CS	V_O=12V	7			mA
I_{OH}	High-level output currents, POWER ON/OFF, CALL, MUTE	V_O=0V	−20			mA
I_{OL}	Low-level output currents, POWER ON/OFF, CALL, MUTE	V_O=12V	5			mA
I_{OH}	High-level output current, IR	V_O=0V	−15			mA
I_{OL}	Low-level output current, IR	V_O=12V	5			mA

APPLICATION EXAMPLE



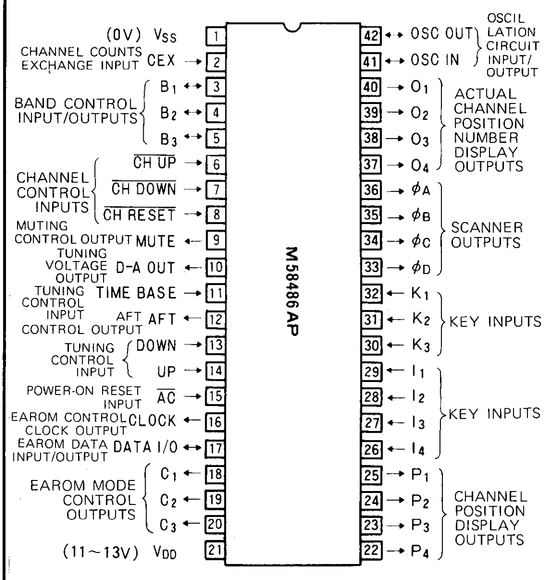
MITSUBISHI ELECTRIC

VOLTAGE SYNTHESIZER

DESCRIPTION
The M58486AP is an aluminum gate CMOS integrated circuit. It has a fully automatic search function capable of writing into an EAROM the tuning voltages corresponding to all receivable stations and a sequentially automatic search function which presets any arbitrary channel. Used in conjunction with the M51251P linear sensor and M5G1400P EAROM, it is possible to configure a fully electronic tuning system for use in TVs or VTR equipment.

FEATURES
- Fully automatic search and sequentially automatic search functions
- The channel display provides channel position tab display, channel position number display, and actual channel number display
- Automatic bandswitching
- Band skip function
- Digital AFT (Automatic Fine Tuning) function
- Frequency fine adjustment function
- AFT on/off data is memorized in EAROM for each channel position
- Direct connection with a remote controller LSI such as the M58485P or M58487AP
- Direct 16 (or 12) channel selection
- Last channel memory function

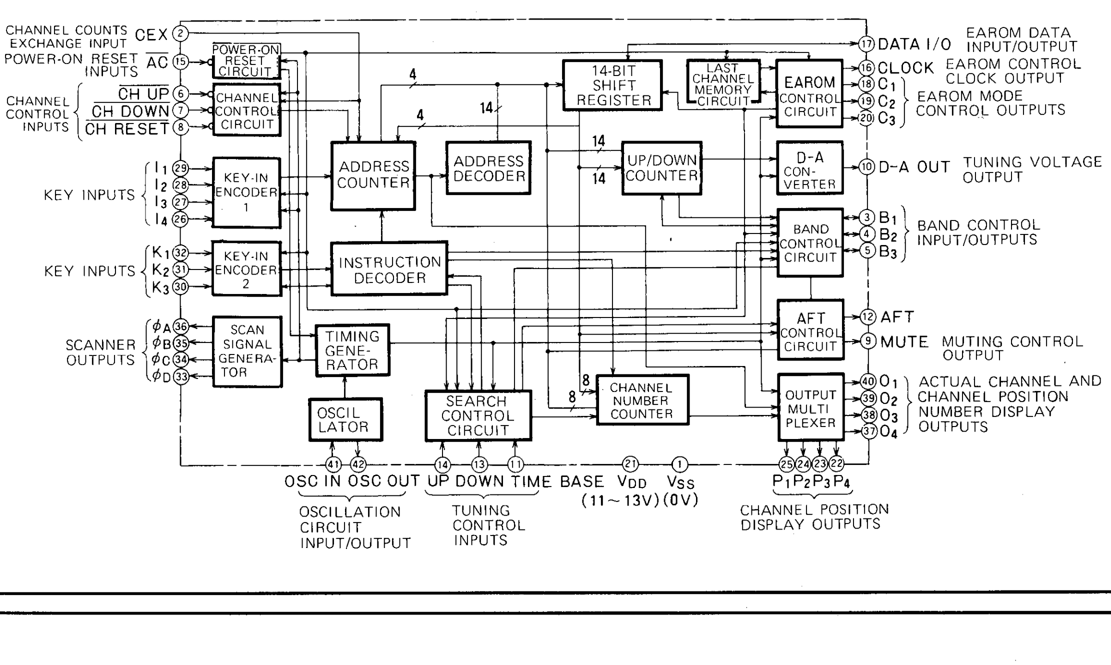
PIN CONFIGURATION (TOP VIEW)



Outline 42P1

APPLICATIONS
Electronic tuning systems for TVs, VTRs, and other electronic equipment.

BLOCK DIAGRAM



FUNCTION

The M58486AP voltage synthesizer, when used in conjunction with the M51251P linear sensor and M5G1400P EAROM, enables the configuration of a completely electronic tuning system without the use of any mechanical parts.

The main functions include fully automatic search, sequentially automatic search, direct selection of either 12 or 16 channels, automatic bandswitching, a band skip function, digital AFT (Automatic Fine Tuning), fine tuning, last channel memory, channel position tab display, channel position number display, and actual channel number display functions.

In addition, direct and sequential channel selection from a remote controller as possible.

FUNCTIONAL DESCRIPTION

Oscillator Circuit

As the oscillator is on-chip, an oscillator frequency is easily obtained by connecting an external LC network or ceramic resonator between the OSC IN and OSC OUT terminals. Fig. 1 and 2 show typical examples.

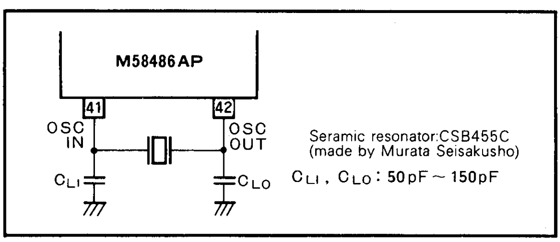


Fig. 1 An example of an oscillator (using a ceramic resonator)

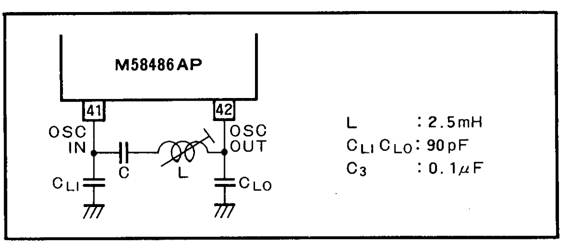


Fig. 2 An example of an oscillator (using an LC network)

Key Inputs

The M58486AP is provided with scanner outputs $\phi_A \sim \phi_D$, key inputs $I_1 \sim I_4$ and $K_1 \sim K_3$. 16-channel position selection can be achieved by using the 4x4 matrix formed by $\phi_A \sim \phi_D$ and $I_1 \sim I_4$. In addition, the 4x3 matrix formed by $\phi_A \sim \phi_D$ and $K_1 \sim K_3$ enables the input of 12 commands.

If two or more of the keys are depressed simultaneously, no commands will be input. However, it is possible to input FAM or CH LOCK in combination with another key.

Table 1 shows the relationships between these matrices and the command functions.

Table 1 Matrix and Command Functions

I \ ϕ	ϕ_A	ϕ_B	ϕ_C	ϕ_D
I_1	CHP 1	CHP 5	CHP 9	CHP 13
I_2	CHP 2	CHP 6	CHP 10	CHP 14
I_3	CHP 3	CHP 7	CHP 11	CHP 15
I_4	CHP 4	CHP 8	CHP 12	CHP 16

K \ ϕ	ϕ_A	ϕ_B	ϕ_C	ϕ_D
K_1	U-SEARCH	D/A UP	CHN 10	CHP-UP
K_2	V-SEARCH	D/A DOWN	CHN 1	CHP-DOWN
K_3	SEARCH	CH LOCK	FAM	STORE

Tuning Voltage Output (D/A OUT)

As a 14-bit D-A converter is built into the M58486AP, tuning voltage can be controlled to 16384 stages. The D-A converter is a pulse-width modulator, and the repetition frequency is 28Hz and the minimum pulse width is 2.2 μs.

By applying this output signal to the electronic tuner through an RC network, the desired tuning frequency can be achieved.

Tuning Control Inputs (UP, DOWN, TIME BASE)

These inputs are required for tuning in the search mode or channel selection mode and are supplied by the M51251P.

As shown in Fig. 3, UP and DOWN inputs are controlled by the AFC signal. The UP input is changed to a high level when the AFC signal exceeds a threshold voltage (V_H) and the DOWN input is changed to a high level when the AFC signal falls below a threshold voltage (V_L).

The TIME BASE input is high when a normal video signal is captured.

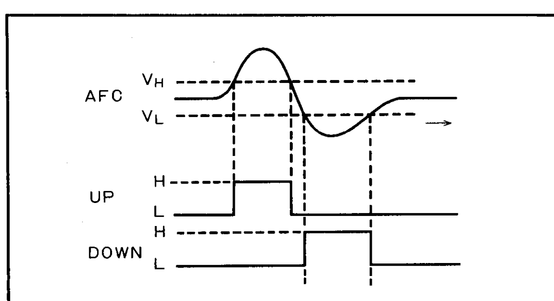


Fig. 3 Relationship of AFC signal to UP and DOWN inputs

Band Input/Outputs (B1 ~ B3)

The M58486AP system is provided with three bands. An electronic tuner is controlled by these three band inputs/outputs and D-A OUT.

As shown in Table 2, these bands correspond to the TV broadcast frequency bands.

Band	Broadcast frequency band
B1	VHF low band
B2	VHF high band
B3	UHF

Three band inputs (B1 ~ B3) are provided on the M58486AP, the output corresponding to the currently selected band being high, with all other band outputs low. Thus, by connecting transistor and LED with currents to these outputs a display of the selected band can be implemented.

If a particular band pin is shorted to V_{SS} that band will be skipped during the search (band skip function).

Search Modes

The search is the function searching automatically the video signal and writing of the required data into the EAROM.

The search function is controlled by the UP, DOWN, and TIME BASE tuning control inputs. Search functions will be described using Fig. 3, 4, and 5.

When search is begun, as up signal is applied to the 14-bit up/down counter, and the analog output of the D-A converter increases (sweeps).

As shown in Fig. 3 and 4, when the signal reaches a certain point, the UP input changes to high. Next, if the DOWN input goes high within 50ms after the UP input goes low, the sweep is ended and the digital AFT is enabled. If DOWN doesn't go high within 50ms, this is taken as an indication that the signal was not a video signal, and the sweep is continued.

Digital AFT is controlled by both the UP and DOWN inputs. When UP is high, the up signal is applied to the up/down counter and the analog output of the D-A converter increases. When DOWN is high, the down signal is applied to the up/down couner and the analog output of the D-A converter decreases. The up/down speed of digital AFT is 1/16 of the up sweep speed.

Digital AFT is ended after 200ms, after which the TIME BASE input is examined. If TIME BASE is low, it is taken as an indication that the signal is not a video signal and the sweep operation is restarted. If TIME BASE is high, the signal is taken as a video signal and the required data is written into the EAROM at the specified address. For this operation, the EAROM address is determined by the channel position and the data written is as follows.

Note, however, that for automatic writing of data into EAROM in the search mode, AFT deta is on.

14-bit up/down counter data	14 bits
2-digit BCD data of channel number counter	8 bits
Band control binary data	2 bits
AFT on/off control data	1 bit

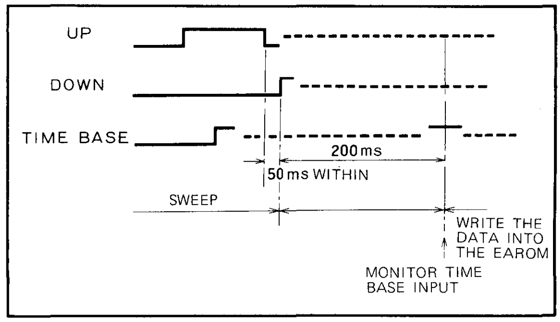


Fig. 4 UP, DOWN, TIME BASE inputs in the search mode

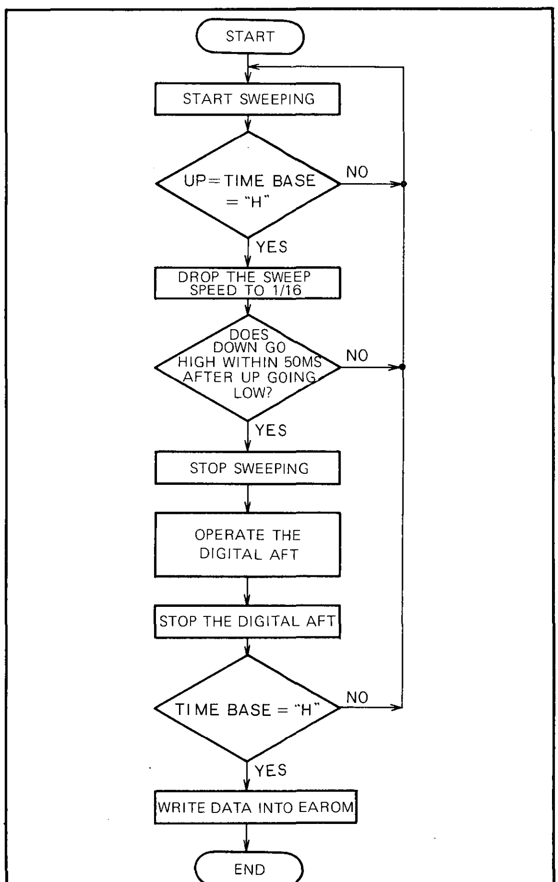


Fig. 5 A flowchart of the search method

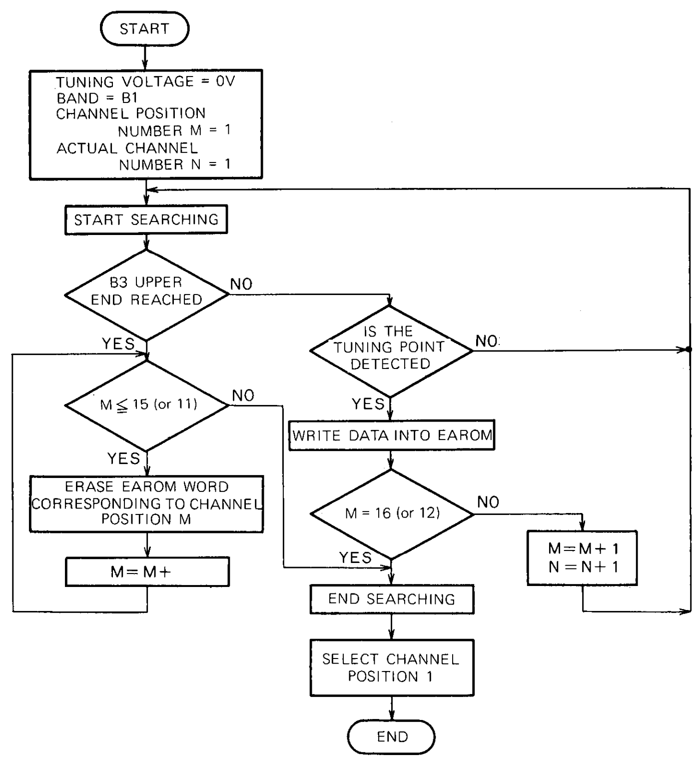


Fig. 6 A flowchart of fully automatic search (SEARCH or V-SEARCH)

Fully Automatic Search

Fig. 6 shows the flowchart of the fully automatic search.

When SEARCH or V-SEARH key is input, the D-A converter analog output is set to the lower end of B1 and the channel position number and actual channel number are both initialized to 1.

After initialization, search begins and when a video signal is captured, the required data is automatically written into the EAROM, the channel position number and actual channel number being incremented by 1, after which the search is restarted.

In this manner, when tuning voltage goes to the upper end of band B3 or when all 16 (or 12) channel positions are written, the EAROM data corresponding to channel position number 1 (channel position 1 is selected) is read, and the fully automatic search operation is completed. If the upper end of band B3 is reached before all 16 (or 12) channels have been searched, the data at the EAROM addresses corresponding to reset channel position is erased. If these erased channel positions are selected, the D-A converter analog output is set to the lower end of band B1, the actual channel number is set to 0, and the AFT function is turned off.

When U-SEARCH key is input, the operation is exactly the same as the above described SEARCH or V-SEARCH except that initialization to the lower end of band B3 is performed and the search ends at the upper edge band B2.

Also, during fully automatic search, no key command can be input.

Sequentially Automatic Search

For sequentially automatic search, the channel position and actual channel number are the currently selected channel position.

When V-SEARCH key is input, search begins from the current position if the current band is B1 or B2, and from the lower end of band B1 if the current band is B3.

The search begins and when a video signal has been captured, the required data is automatically written into the EAROM, the search mode is cancelled, and the search is completed. When the upper end of the B2 band is reached, the tuning voltage output returns to the lower end of the B1 band and search continues.

When U-SEARCH key is input, search begins at the present location if the current band is B3. If it is B1 or B2, it begins at the lower end of band B3. The search method is exactly the same as for the above described V-SEARCH except that when the upper end of band B3 is reached, the tuning voltage returns to the lower end of band B3.

When SEARCH key is input, search begins from the current location. For SEARCH, when the upper end of band B3 is reached, the tuning voltage returns to the lower end of band B1.

During sequentially automatic search, pressing channel selector keys cancels the search mode, ending the search and resulting in input of the channel selection command.

Search Speed

The tuning voltage rate of change varies between bands and within bands such that the search speed with respect to frequency is virtually constant over the entire range.

Because of the time constant associated with the integration circuit connected to the D/A OUT output, time delays occurs during the sweep. However, to compensate for this when UP and TIME BASE inputs are both high, the search speed is dropped to 1/16 of the sweep speed.

Table 3 shows the search speed for all bands without this reduced speed mode.

Table 3 Search Speed for Each Band

Tuning voltage \ Band	B1	B2	B3, B4
0 ~ 1/4	1.16s	2.31s	9.22s
1/4 ~ 1/2	0.58	1.16	
1/2 ~ 1	0.58	1.16	4.61
Total	2.32	4.63	13.83

Note 1. The reference oscillator frequency is 455kHz.
2. The tuning voltage is given normalized to a value of 1.

Switching between fully automatic search and sequentially automatic search is accomplished by the FAM command as shown in Table 1. By using a switch, connecting the ϕ_C pin, with the K_3 pin results in fully automatic search while opening this connection results in switching to sequencially automatic search.

If the FAM command is attempted during a search, the command will not immediately be executed. After the search mode has been cancelled it will be input and the appropriate search mode, either fully automatic or automatic sequencial search, will be selected.

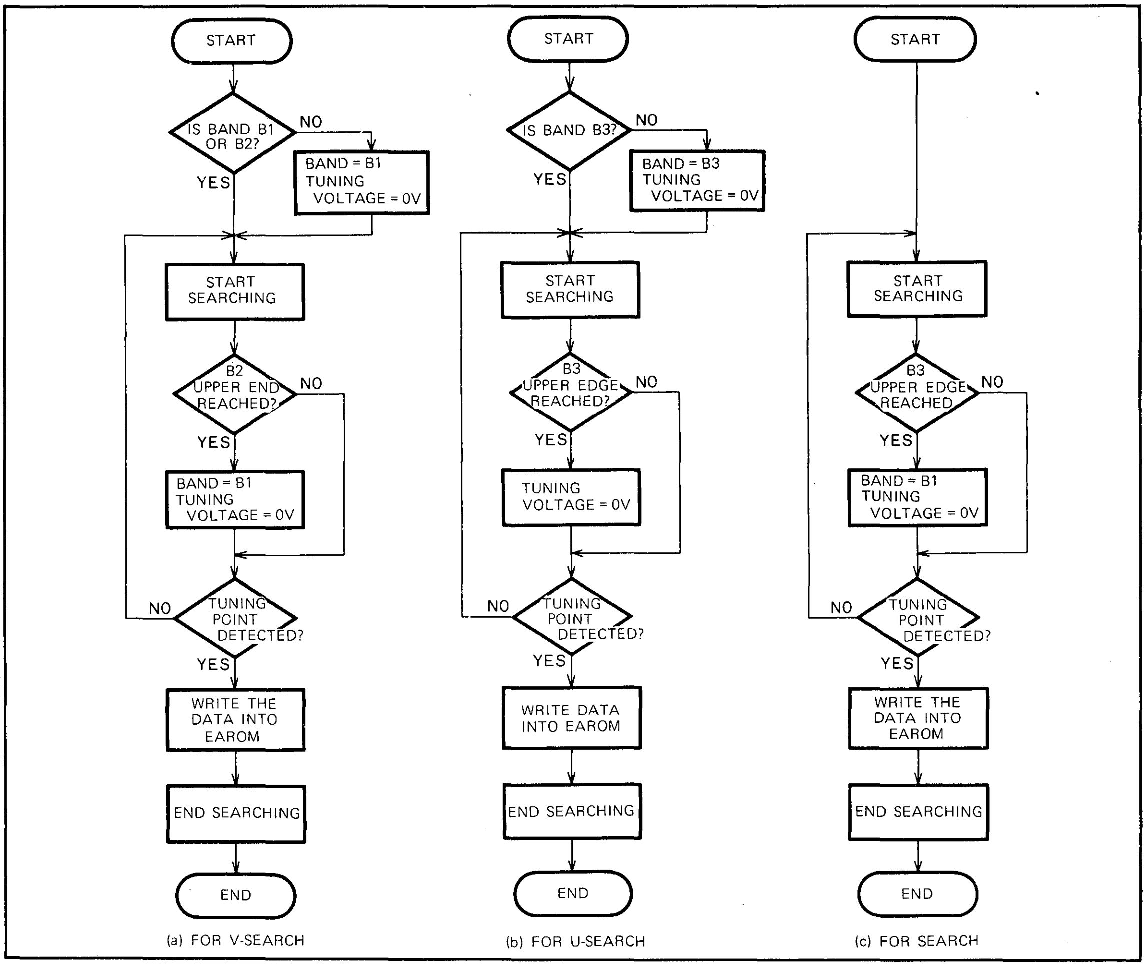


Fig. 7 Shows the flowchart of sequentially of search.

Channel Selection Mode

When either a channel selection key is depressed or a channel selection command is input from a remote control receiver (described below), the data at the EAROM address corresponding to the selected channel position is read.

After the read data is set in the up/down counter, if the AFT control data read is on, 16 down pulses are applied, causing the up/down counter to count down and cause a corresponding output from the D-A converter. This is to enable pull-in at the optimum position of the video signal, using the digital AFT and linear AFT to be descfibed next.

If the AFT control data read is on, after 100 ms digital AFT is enabled. In addition, when 100ms has elapsed, the AFT output goes high and linear AFT is enabled. When the AFT control data is off, both digital and linear AFT functions are disabled.

Tuning Voltage Fine Adjustment (D-A UP, D-A DOWN)

By pressing the D-A UP and D-A DOWN key, it is possible to adjust the D-A converter analog output (that is the tuning voltage).

After channel selection, pressing the D-A UP or D-A DOWN keys turns AFT off and disables both digital and linear AFT functions. After this, the up or down signals are applied to the up/down counter and the D-A converter analog output changes. The rate of this change is 1/128 of the sweep speed, allowing sufficient fine adjustment.

When the key is released writing into the EAROM begins. At this time, the AFT on/off data is written as off.

EAROM Input/Output (CLOCK, C_1, C_2, C_3, DATA I/O)
This system makes use of an M5G1400P as an EAROM.

To control the M5G1400P, the M58486AP is provided with a reference clock (~14kHz) output clock, outputs C_1, C_2, and C_3 used to specify the mode, and a data input/output DATA I/O.

These inputs and outputs are controlled by the EAROM control circuit. The clock output is fixed at the V_{DD} level at all times except during memory read and write operations.

AFT Output
The AFT pin is connected to the AFT on/off pin (pin 15) of the M51251P, and is used to on/off control linear AFT. When the AFT output is high, linear AFT is enabled. When it is low or high impedance (open) linear AFT is disabled. Table 4 summarizes the AFT output for the various states.

Table 4 AFT Outputs for the Various Modes

Mode	AFT output level
Search mode (during sweep)	L
Search mode (during the 200ms that digital AFT is enabled)	Z
Channel selection mode (with linear AFT on)	H
Channel selection mode (with linear AFT off)	Z

Note 1. 'Z' indicates high-impedance (open)

Last Channel Memory
In this system when the power supply is applied, a last channel memory function selects the last channel position that was selected before the power supply was last removed.

This function is controlled by the last channel memory circuit such that when a channel is selected the data for the selected channel position is written into a specified address in the EAROM. Each time a channel is selected the data contents are updated so that the last channel selected before power is removed is always stored. When the power is applied, this data is read from the EAROM and used as the initial channel position selected.

Channel Position Display (P_1 ~ P_4)
By connecting transistors and LEDs to the 4x4 matrix formed by the P_1 ~ P_4 and ϕ_A ~ ϕ_D outputs, a 16-channel position display can be configured.

The display repetition frequency is 45Hz and the duty cycle is 23.5%. Fig. 8 gives an example of output timings. Note that when not used pins should be connected to V_{DD}.

Channel Number Display (O_1 ~ O_2)
The output O_1 ~ O_4 provide a two-digit (0 ~ 99) BCD output of the actual channel number. The upper and lower digits are output under the control of the ϕ_D and ϕ_B scan signals. Thus, by using a BCD seven-segment decoder (for example, the M53247P or equivalent), it is possible to display the actual channel number using a two-digit seven-segment display. Fig. 9 shows an example of timing for the outputs O_1 ~ O_4 used to display the actual channel number.

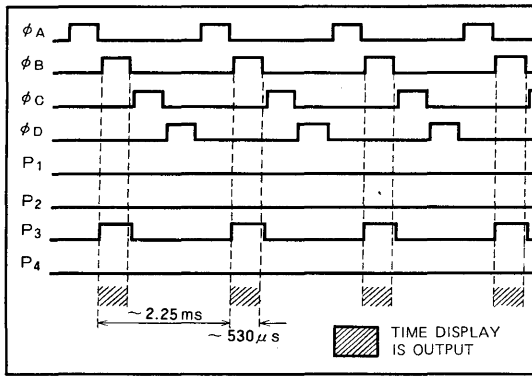


Fig. 8 Timing example for outputs P_1~P_4 and ϕ_A~ϕ_D (channel position = 7)

When the O_1 ~ O_4 outputs are used to display the channel position number in binary form the ϕ_A and ϕ_C scan signals are used for timing of the otuputs. For the channel positions 1 ~ 16, the O_1 ~ O_4 outputs are 0 ~ 15. Therefore, the channel position number can be displayed using seven-segment display elements. Fig. 9 shows a timing example of the outputs O_1 ~ O_4 used to display the channel position number. The outputs O_1 through O_4 use N-channel transistors in open drain configuration. When not used they should be connected to the V_{SS} pin.

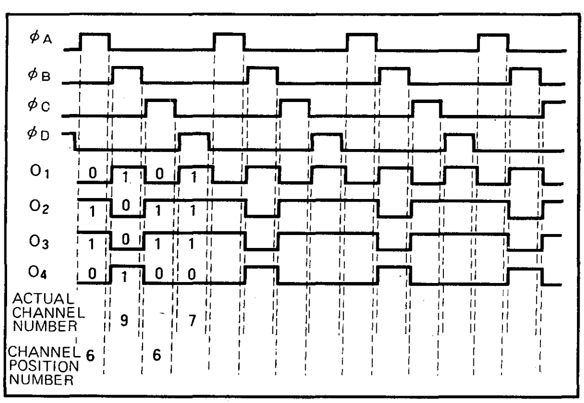


Fig. 9 Timing example for outputs O_1~O_4 and ϕ_A~ϕ_D (Actual channel number setting line 79 Channel position number 7)

Actual Channel Number Control Inputs (CHN 10, CHN 1)
When CHN 10 key is input, the upper digit of the actual channel number is incremented by 1, cycling back to 0 after reaching 9. In the same manner, when CHN 1 key is input, the lower digit is incremented by 1. Therefore, by using these inputs, the actual channel number can be changed with respect to the channel position, and by using the STORE command described below, the proper corresponding channel numbers can be selected.

Channel Position Control Inputs (CHP-UP, CHP-DOWN)
When either CHP-UP or CHP-DOWN key is input, the contents of the address counter are incremented or decremented by 1, the channel position display changing accordingly. But data is not read from the EAROM, so the D-A converted analog output, band, AFT output and actual channel do not change.

When these commands are input, the channel position is changed, and the STORE command described below is input, data is written into the EAROM at the address corresponding to the displayed channel position. This enables, for example, such copying operations as writing the same data in position 3 as stored in position 1.

EAROM Write Command (STORE)
When the STORE command is input, data is written into the EAROM at the address corresponding to the currently displayed channel position. This STORE command is used to change the actual channel number and to perform memory copying operations.

Audio Control Output (MUTE)
In the search mode or channel selection mode, the MUTE output changes to a high level, enabling the muting function which lowers the sound level to the minimum level. This output is normally low.

Power-on reset ($\overline{AC}$)
By connecting a capacitor between the $\overline{AC}$ pin and the V_{SS} pin, the power-on reset function is enabled upon applying power to the M58486AP.

When the power-on reset operates, the last channel memory function is enable the channel position selected before the power was removed, is selected.

Remote Control Inputs ($\overline{\text{CH UP}}$, $\overline{\text{CH DOWN}}$, $\overline{\text{CH RESET}}$)
If the $\overline{\text{CH UP}}$, $\overline{\text{CH DOWN}}$, and $\overline{\text{CH RESET}}$ inputs are connected to the corresponding pins on, for example, a remote control receiver device such as the M58485P or M58487AP, direct remote control of channel selection, channel up, and channel down functions is possible.

Channel Lock Input (CH LOCK)
By using the input combination of the key input K_3 and the scan signal ϕ_B, the CH LOCK command is input. This command prohibits the CHP1 ~ CHP16, CHP-UP, AND CHP-DOWN keys commands as well as the remote control $\overline{\text{CH-UP}}$, $\overline{\text{CH-DOWN}}$, and $\overline{\text{CH-RESET}}$. This command is independent of any other key commands and can be input simultaneously input with any command except the channel selection commands CHP1~CHP16.

Number of Channels Selection Input (CEX)
The CEX input is provided with a built-in pull-up resistance and when at the high level (or open), the M58486AP for 16 channels. When it is at the low level the M58486AP accommodates 12 channels.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	$-0.3 \sim 15$	V
V_I	Input voltage		$V_{SS} \leqq V_I \leqq V_{DD}$	V
V_O	Output voltage		$V_{SS} \leqq V_O \leqq V_{DD}$	V
P_d	Power dissipation	Ta=25°C	300	mW
T_{opr}	Operating temperature		$-30 \sim 70$	°C
T_{stg}	Storage temperature		$-40 \sim 125$	°C

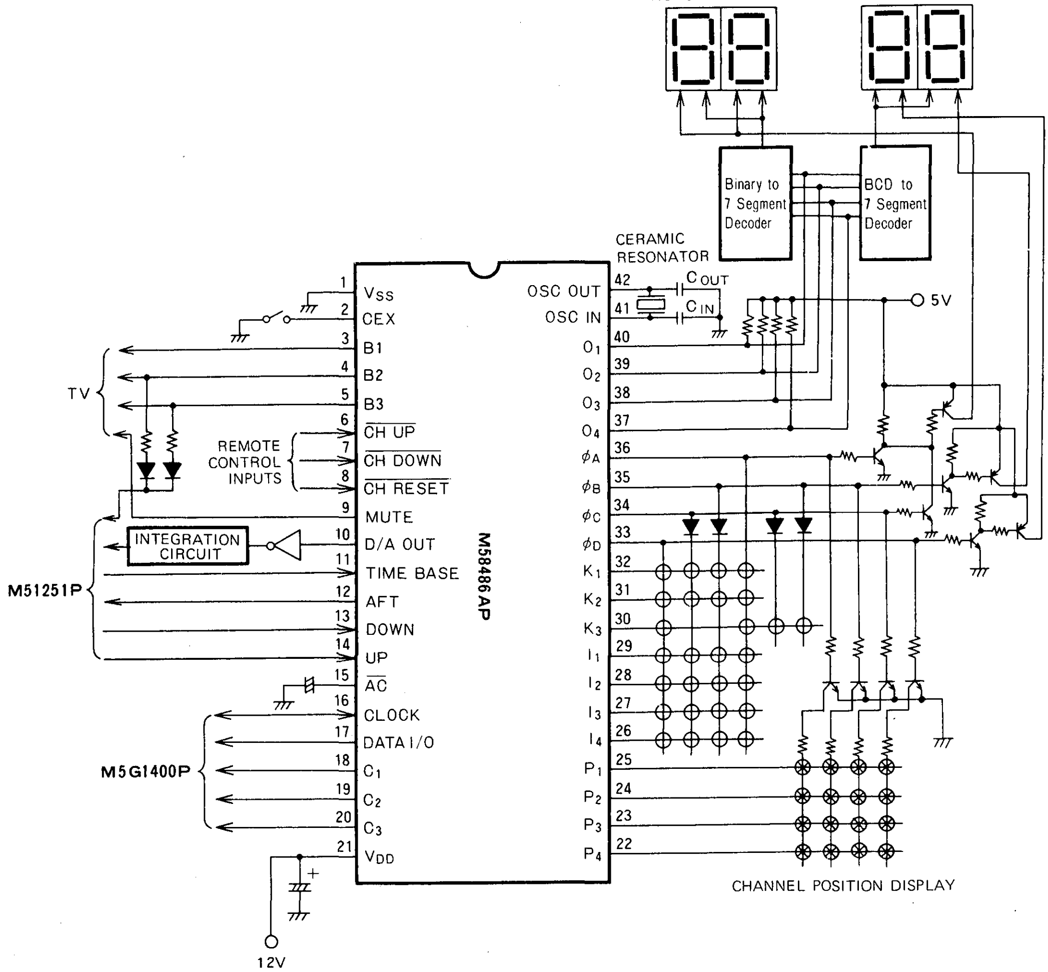
RECOMMENDED OPERATING CONDITIONS

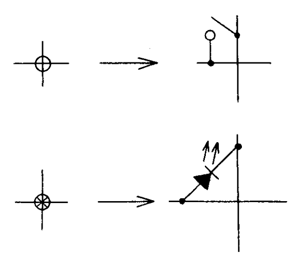
Symbol	Parameter	Limits			Unit
		Min	Typ	Max	
V_{DD}	Supply voltage	11	12	13	V
V_{IH}	High-level input voltage	$V_{DD}-3$	V_{DD}	V_{DD}	V
V_{IL}	Low-level input voltage	0	0	3	V
f_{OSC}	Oscillation frequency		455		kHz
			480		kHz

ELECTRICAL CHARACTERISTICS (Ta=25°C, V_{DD}=12V, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{DD}	Operational supply voltage	Ta=$-30 \sim 70$°C, f_{OSC}=455kHz	11	12	13	V
I_{DD}	Supply current	f_{OSC}=455kHz		0.5	6	mA
R_I	Pull-up resistance , $\overline{\text{CH UP}}$, $\overline{\text{CH DOWN}}$, $\overline{\text{CH RESET}}$, UP, DOWN, TIME BASE, CEX			50		kΩ
R_I	Pull-up resistance, $\overline{\text{AC}}$			100		kΩ
R_I	Pull-down resistance, $I_1 \sim I_4$, $K_1 \sim K_3$			50		kΩ
I_{OH}	High-level output current, $\phi_A \sim \phi_D$	V_O=10V	−5			mA
I_{OH}	High-level output current, B1~B3, MUTE	V_O=10V	−1			mA
I_{OL}	Low-level output current, B1~B3, MUTE	V_O=2V	2			mA
I_{OH}	High-level output current, AFT, D/A OUT	V_O=10V	−1.5			mA
I_{OL}	Low-level output current, AFT	V_O=2V	1			mA
I_{OL}	Low-level output current, D/A OUT	V_O=2V	1.5			mA
I_{OZH}	Off-state output current, AFT	V_O=12V			1	μA
I_{OZL}	Off-state output current, AFT	V_O=0V			−1	μA
V_{OH}	High-level output voltage, CLOCK, C1~C3	I_{OH}=−0.5mA	11			V
V_{OL}	Low-level output voltage, CLOCK, C1~C3	I_{OL}=1mA			2	V
V_{OH}	High-level output voltage, DATA I/O	I_{OH}=−0.2mA	11			V
V_{OL}	Low-level output voltage, DATA I/O	I_{OL}=0.5mA			2	V
I_{OZH}	Off-state output current, DATA I/O	V_O=12V			1	μA
I_{OZL}	Off-state output current, DATA I/O	V_O=0V			−1	μA
V_{OH}	High-level output voltage, $P_1 \sim P_4$	I_{OH}=−40mA	10			V
I_{OZL}	Off-state output current, $P_1 \sim P_4$	V_O=2V			10	μA
I_{OL}	Low-level output current, $O_1 \sim O_4$	V_O=0.4V	1.6			mA
I_{OZH}	Off-state output current, $O_1 \sim O_4$	V_O=10V			1	μA

APPLICATION EXAMPLE





MITSUBISHI ELECTRIC

24-FUNCTION REMOTE-CONTROL RECEIVER

DESCRIPTION

The M58487AP is a 24-function remote-control receiver circuit manufactured by aluminum-gate CMOS technology for use in television receivers, audio equipment, and the like, using infrared for transmission. It enables direct control of 8 functions at the receiver.

The M58487AP is intended for use with an M58480P or M58484P transmitter.

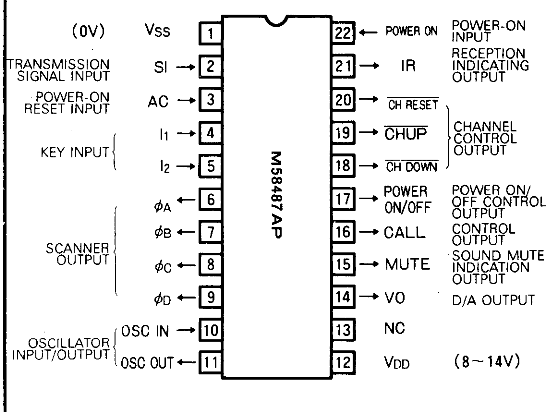
FEATURES

- Wide supply voltage range: 8V~14V
- Single power supply
- Low power dissipation
- On-chip oscillator
- Low-cost LC or ceramic oscillator used in determining the reference frequency (480kHz or 455kHz)
- Information is transmitted by means of pulse code modulation
- Good noise immunity—instructions are not executed unless same code is received three or more times in succession.
- Single transmission frequency (40kHz or 38kHz) for carrier wave
- 16 TV channels selected directly
- Three analog functions—volume, brightness, and color situration—are independently controlled to 64 stages by three 6-bit D/A converters
- 8 commands are controlled at the M58487AP receiver
- Has large tolerance in operating frequency between the transmitter and the receiver.
- Can be connected with an M51231P or equivalent touch control channel selector IC

APPLICATION

- Remote-control receiver for TV or other applications

PIN CONFIGURATION (TOP VIEW)



Outline 22P1

NC : NO CONNECTION

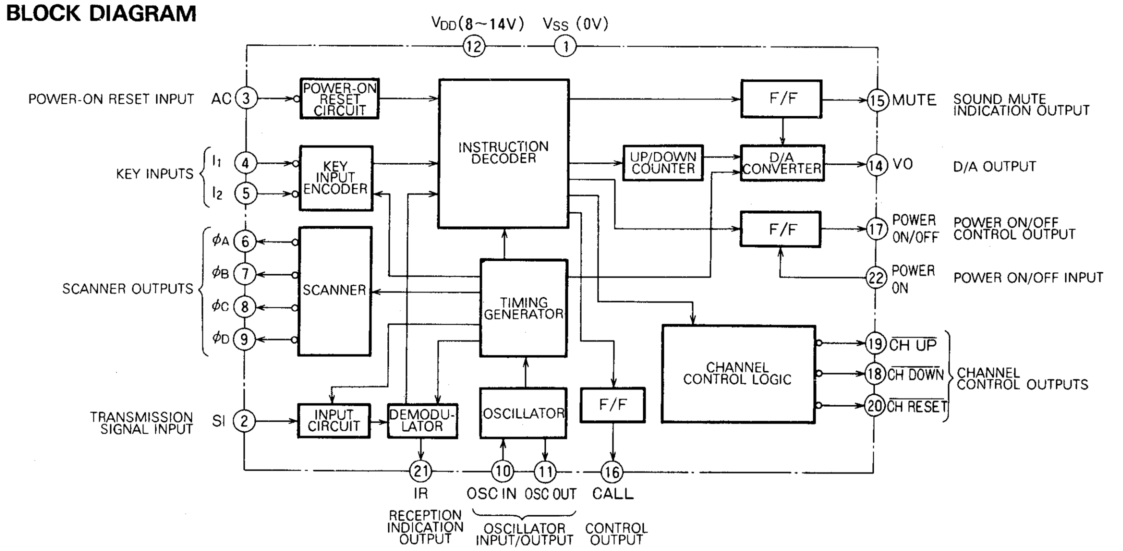
FUNCTION

The M58487AP is designed to decode and execute instructions after three successive receptions of the identical instruction code, providing a good noise immunity.

Instructions comprise direct selection of 16 channels, channel position up and down, volume up and down, brightness up and down, color saturation up and down, normalization of volume, brightness and color saturation, sound mute on and off, TV main power on and off, and output CALL on and off.

In addition, 8 functional instructions can be entered from the receiver side.

BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

Oscillator

As the oscillator is on-chip, oscillation frequency is easily obtained by connecting an external LC network or ceramic resonator between the OSC IN and OSC OUT terminals. Figs. 1 and 2 show typical oscillators.

Fig. 1 An example of an oscillator (when a ceramic resonator is used)

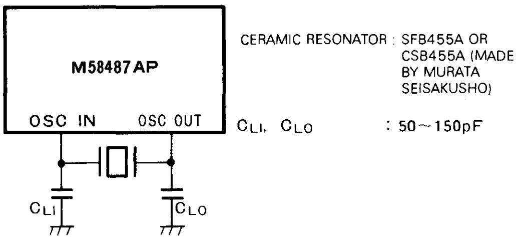
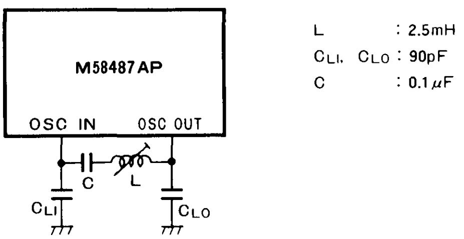


Fig. 2 An example of an oscillator (when a LC network is used)



Reception Signal Input Circuit and Demodulation Circuit

The reception signal caught by the photo detector is amplified in the amplifier and added to the SI, where it is converted into a pulse signal in the input circuit to be sent to the demodulation circuit. In the demodulation circuit, the pulse interval of the pulse signal is judged and then converted into the digital code to be sent to the instruction decoder.

SI is applied as amplified, either through a capacitor coupling (Fig. 3) or directly as a pulse signal (Figs. 4 and 5). A Schmitt trigger circuit is provided in the SI input circuit for preventing spurious operation due to noise.

Fig. 3 SI input waveform (when applied through a capacitor coupling)

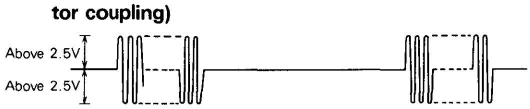


Fig. 4 SI input waveform (when applied directly)

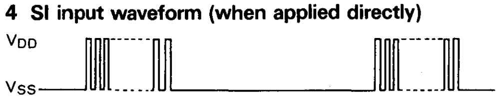
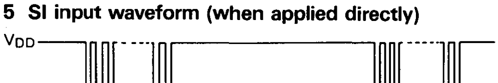


Fig. 5 SI input waveform (when applied directly)



Instruction Decoder

The instruction decoder starts to function after receiving the same instruction code three or more times in succession from the demodulation circuit.

Table 1 shows the relations between the reception code and instruction function. To prevent spurious operation, there is no code 000000.

Table 1 Relations between reception codes and instructions

Reception code						Function	Remarks
D_1	D_2	D_3	D_4	D_5	D_6		
1	0	0	0	0	0	CH UP	Channel up
0	1	0	0	0	0	CH DOWN	Channel down
1	1	0	0	0	0	VO UP	Volume up
0	0	1	0	0	0	VO DOWN	Volume down
1	0	0	1	0	0	MUTE	Sound mute on/off
0	1	0	1	0	0	VO(1/3)	Normalization of volume
1	0	1	1	0	0	CALL	Output CALL on/off
0	1	1	1	0	0	POWER ON/OFF	Power on/off
0	0	0	0	1	0	CH 1	Direct channel selection (Direct access)
1	0	0	0	1	0	CH 2	
0	1	0	0	1	0	CH 3	
1	1	0	0	1	0	CH 4	
0	0	1	0	1	0	CH 5	
1	0	1	0	1	0	CH 6	
0	1	1	0	1	0	CH 7	
1	1	1	0	1	0	CH 8	
0	0	0	1	1	0	CH 9	
1	0	0	1	1	0	CH 10	
0	1	0	1	1	0	CH 11	
1	1	0	1	1	0	CH 12	
0	0	1	1	1	0	CH 13	
1	0	1	1	1	0	CH 14	
0	1	1	1	1	0	CH 15	
1	1	1	1	1	0	CH 16	

Key Inputs

8 different instructions are input by a 2 x 4 keyboard matrix consisting of inputs $I_1 \sim I_2$ and scanner outputs $\phi A \sim \phi D$, Protection is also available against chattering within 10ms.

As entry priority is given to each key, depression of more than two keys at the same time makes the key with higher priority effective. For the scanner output, priority is given in the order of ϕA, ϕB, ϕC, and ϕD, and I_1 takes precedence over I_2 if the scan output is the same. When two or more keys are depressed at the same time, scanner outputs may short-circuit, disabling all functions.

While one of the keys is depressed, instructions from the transmitter are ignored.

Table 2 shows the relations between the keyboard matrix and the instructions.

Table 2 Relations between keyboard matrix and instructions

Key input \ Scanner output	ϕ_D	ϕ_C	ϕ_B	ϕ_A
I_1	POWER ON/OFF	VO UP	MUTE	CH UP
I_2	CALL	VO DOWN	VO(1/3)	CH DOWN

MITSUBISHI ELECTRIC

Indication of Reception

As soon as an identical code is received three times, output IR turns from low-level to high-level. Thus reception of an instruction from the transmitter can be indicated by an LED connected to the output IR.

Output VO

As the 6-bit D/A converter is contained internally, analog value can be controlled to 64 stages independently. The D/A converter is pulse-width modulator, the reception frequency is 1.25kHz (when f_{OSC} = 480kHz) and minimum pulse width is 12.5μs.

Analog value can be incremented/decremented at a rate of about 1 step/0.1 second through the remote control or the key input. The time required for increasing the analog value from the minimum to the maximum is about 6.6 seconds (when f_{OSC} = 480kHz).

It is also possible to set the analog value to 1/3 of its maximum value by means of the remote control or the key input (normalization).

Sound Mute

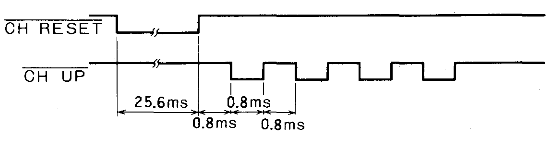
Sound mute on/off is controlled through the remote control or the key input. When sound mute is on, output VO goes low, and output MUTE goes high.

Sound mute is automatically released from ON when the output VO is either incremented or decremented by remote control or the key input.

Channel Control

Channel control is attained through outputs $\overline{\text{CH UP}}$, $\overline{\text{CH DOWN}}$ and $\overline{\text{CH RESET}}$. With respect to direct channel selection by the remote-control operation, a single pulse appears on output $\overline{\text{CH RESET}}$ first, and then the pulses whose number is deducted by one from the selected channel appear on the output $\overline{\text{CH UP}}$. Up and down channel switching is controlled by presenting a single pulse on the output $\overline{\text{CH UP}}$ or $\overline{\text{CH DOWN}}$. Thus it can be connected with an M51231P or equivalent touch-control channel selector IC.

Fig. 6 Timing chart of channel control (when f_{osc} = 480kHz)



During direct channel selection, up or down, output VO goes low for 50~100ms.

Outputs, $\overline{\text{CH UP}}$, $\overline{\text{CH DOWN}}$, and $\overline{\text{CH RESET}}$ are the open-drain type of N-channel transistor.

Power On/Off

The remote control or the key input makes it possible to turn the POWER ON/OFF output from low to high or vice versa. While the POWER ON/OFF output is low, all channel and analog controls through the remote control are disabled, as are all through the keyboard.

Output CALL

The output CALL is turned high or low by remote control or the key input. This output effects on/off control of channel number indication or change of receiving modes of multi-channel broadcasting.

Power-on Reset

Attaching a capacitor to terminal AC activates the power-on reset when power is on to the M58487AP.

Activation of the power-on reset function sets output VO to 1/3 of its maximum value and turns the POWER ON/OFF and CALL outputs to low-level.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	−0.3〜15	V
V_I	Input voltage		$V_{SS} \leqq V_I \leqq V_{DD}$	V
V_O	Output voltage		$V_{SS} \leqq V_O \leqq V_{DD}$	V
Pd	Maximum power dissipation	Ta=25℃	300	mW
Topr	Operating temperature		−30〜70	℃
Tstg	Storage temperature		−40〜125	℃

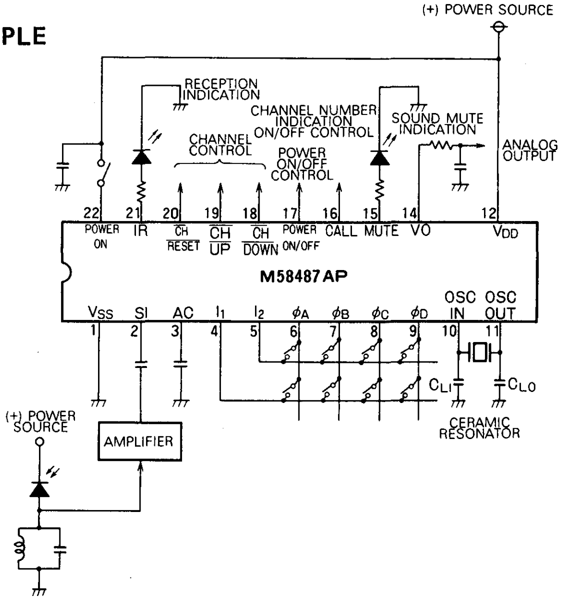
RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	8	12	14	V
f_{OSC}	Oscillation frequency		455		kHz
			480		kHz
V_I	Input voltage, SI	5			V_{P-P}
V_{IH}	High-level input voltages, I_1, I_2	$0.7 \times V_{DD}$	V_{DD}	V_{DD}	V
V_{IL}	Low-level input voltages, I_1, I_2	0	0	$0.3 \times V_{DD}$	V

ELECTRICAL CHARACTERISTICS (Ta=25℃, V_{DD}=12V, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{DD}	Operating supply voltage	Ta=−30〜70℃, f_{OSC}=455kHz	8	12	14	V
I_{DD}	Supply current	f_{OSC}=455kHz		2	5	mA
R_I	Pull-up resistances, I_1, I_2			20		kΩ
I_{OL}	Low-level output currents, $\phi_A \sim \phi_D$	V_O=12V	5			mA
I_{OL}	Low-level output currents, $\overline{\text{CH RESET}}$, $\overline{\text{CH UP}}$, $\overline{\text{CH DOWN}}$	V_O=12V	20			mA
I_{OZH}	Off-state output currents, $\overline{\text{CH RESET}}$, $\overline{\text{CH UP}}$, $\overline{\text{CH DOWN}}$	V_O=12V			1	μA
I_{OH}	High-level output current, VO	V_O= 0 V	−7			mA
I_{OL}	Low-level output current, VO	V_O=12V	7			mA
I_{OH}	High-level output currents, POWER ON/OFF, CALL, MUTE	V_O= 0 V	−20			mA
I_{OL}	Low-level output currents, POWER ON/OFF, CALL, MUTE	V_O=12V	5			mA
I_{OH}	High-level output current, IR	V_O= 0 V	−15			mA
I_{OL}	Low-level output current, IR	V_O=12V	5			mA

APPLICATION EXAMPLE



MITSUBISHI ELECTRIC

MICROCOMPUTER SYSTEMS

12

MELCS 85/2 SINGLE-BOARD COMPUTER

DESCRIPTION

The PCA8501 is a general-purpose single-board computer that is composed of a memory and an I/O interface around the M5L8085AP 8-bit microprocessor and fabricated on a single 125 x 145mm printed circuit board. As it has been designed so compactly in its dimensions, it may be easily attached to the board currently used. There are two types available: the PCA8501G01, which is implemented with the M5L2114LP NMOS RAMs, and the PCA8501G02, is implemented with the M58981S CMOS RAMs.

FEATURES

Type	Contents
PCA8501G01	Consists of the single-board computer only. Two M5L2114LP NMOS RAMs are mounted for its RAM, excluding both a battery backup circuit and a wait signal generation circuit, and one M5L2716K EPROM is attached separately.
PCA8501G02	Consists of the single-board computer only. Two M58981S CMOS RAMs are mounted for its RAM, including a battery backup circuit and a wait signal generation circuit, and one M5L2716K EPROM is attached separately.

- A single-board computer comprised of the CPU, memory, I/O interface and a timer.
- Capacity of EPROM: 4K bytes (max)
- Capacity of RAM: 1K bytes
- Programmable I/O port: 48 bits (8-bit x 6)
- Internally contained I^2L timer: One of the following 8 timer outputs can be selected (1.6μs, and 0.1, 0.2, 0.4, 0.8, 1.6, 3.3 and 6.6ms).
- Single 5V power supply
- Compact dimensions (L x W x H): 125 x 145 x 17mm

APPLICATIONS

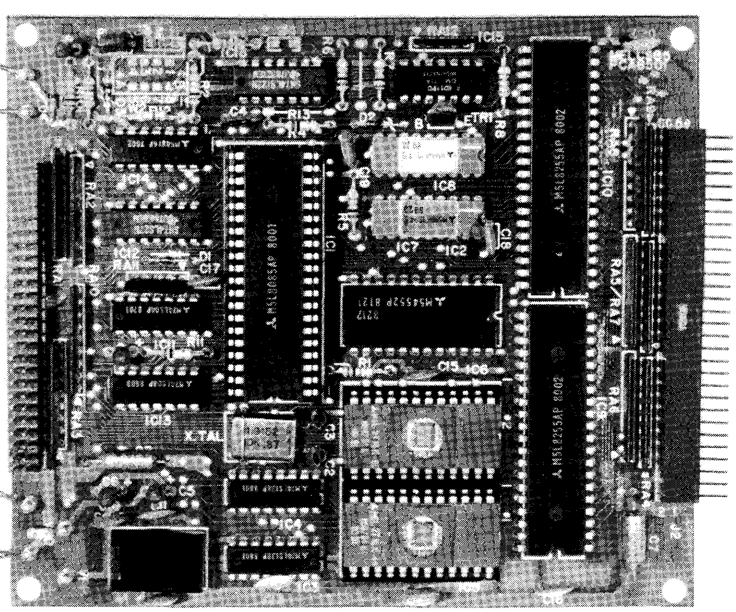
- Personal computers
- Small automatic testing or control equipment
- Data-communication terminal equipment
- Data loggers and data-collection equipment
- Process-control equipment
- Instrument monitoring controllers

FUNCTION

The PCA8501 is a highly reliable single-board computer designed around Mitsubishi's M5L8085AP CPU (equivalent to Intel's 8085A) and its LSI family. The 8-bit parallel CPU is fabricated by the N-channel silicon-gate ED-MOS process. The PCA8501 comes with 4K bytes of electrically programmable read-only memory (EPROM) in the form of two M5L2716Ks and 1K bytes of random-access memory in the form of two M5L2114LPs or two M58981Ss.

For its I/O ports, the PCA8501 contains two M5L8255AP programmable peripheral interfaces (PPI) providing 8 bits $\times$ 6 $=$ 48 bits programmable ports.

A timer circuit and a battery backup circuit (which is available only for the PCA850G02) are mounted on the board, allowing timer interrupt and memory backup.



12

OPERATIONS

As soon as the power is applied, the M5L8085A CPU is reset by the power-on reset circuit and starts to execute the program from the address 0000_{16}.

The low-order 8 bits of the address are multiplexed with data and sent out through the CPU terminals. They are latched in the address latch circuit so as to compose an address bus with the high-order 8 bits of the address.

If an external extension signal is used, it makes easy the external expansion of memory capacity for both ROMs and RAMs.

Use of CMOS RAMs enables memory backup by means of the battery backup circuit and batteries so that the contents of the RAM are maintained even after the power is turned off.

Either of the ROMs, M5L2708K (1K bytes) or M5L2716K (2K bytes) can be used by using a jumper socket, but the standard version is arranged for the use of the M5L2716K.

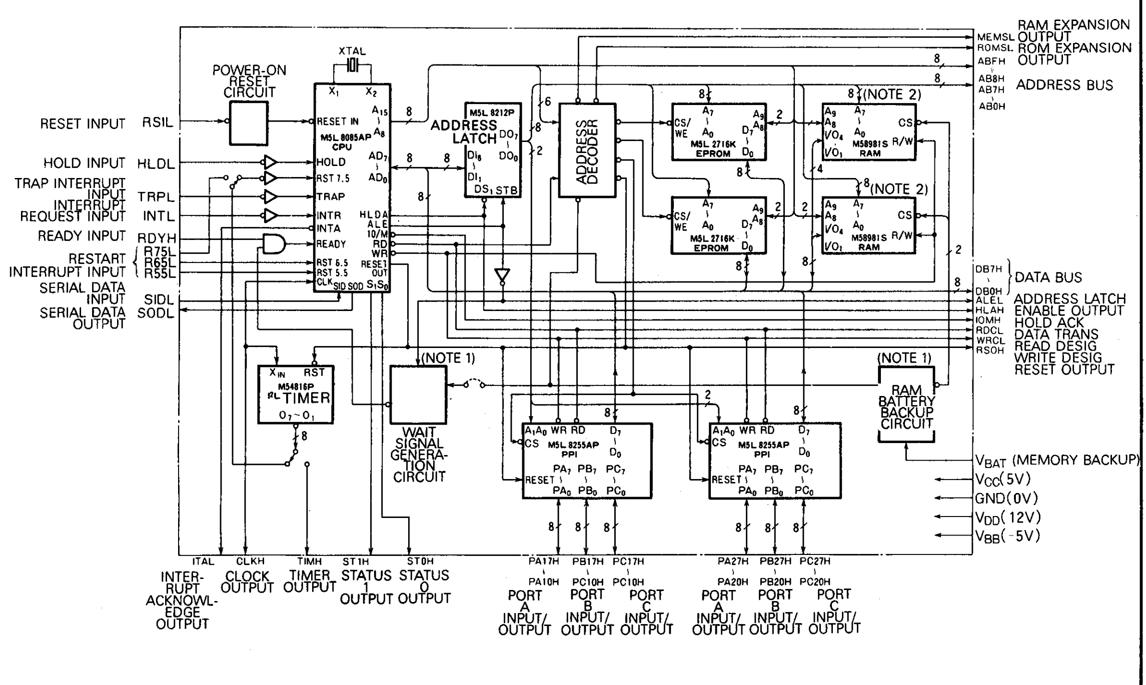
Parallel data can be read/written through the PPIs, and serial data through the SID and SOD of the M5L8085AP CPU.

As the timer IC is provided on the board, it enables timer interrupt by means of the RST 7.5 or timer output to the external circuit.

BLOCK DIAGRAM NOTATION

Name of block	Function
CPU power-on reset	Execution is carried out in accordance with the contents of a program. System reset signal is generated when the power is turned on.
Address latch circuit	Latches the low-order 8-bit address signal on the multiplexed data bus.
Address decoder	Decodes the high-order bits of the address, and generates the memory and I/O chip select signals.
EPROM	Both erasable M5L2716K and M5L2708K can be used.
RAM	Allows the use of the M58981S CMOS RAMs other than the M5L2114LPs, which enables battery backup.
RAM battery backup circuit	Enable maintaining the contents of the memory by the backup circuit with batteries in use, when the CMOS RAMs are used.
I/O port (PPI)	It is a programmable I/O interface consisting of 48-bit I/O signal terminals, corresponding to six 8-bit I/O ports.
Timer	This generates 7 different signals after dividing the clock signal from the CPU, allowing RST 7.5 interrupt by using a jumper wire.
Wait signal generation circuit	When the CMOS RAMs are in use, wait signal is generated to wait one clock time. (This feature is not available in the PCA8501G01.)

BLOCK DIAGRAM



Note 1 : The wait signal generation circuit and the RAM battery backup circuit are not mounted on the PCA8501G01.
2 : The M5L21114LPs are mounted on the PCA8501G01, instead of M58981Ss.

MITSUBISHI ELECTRIC

SPECIFICATIONS

Processing Method

Method: 8-bit parallel operation
CPU: M5L8085AP
Word length:
  Instruction: 8, 16, 24 bits
  Data: 8 bits
Cycle time:
  Basic cycle time: 1.6μs
CPU clock frequency:
  2.4576 MHz ±1% (Ta=0~55°C, V_{CC}=5V ±5%)
  (Quartz oscillation frequency: 4.9152 MHz ±1%)

Memory Address and Memory Capacity

EPROM (M5L2716K)
  Memory address:
    #1: 0000_{16}~$07FF_{16}$
    #2: 0800_{16}~$0FFF_{16}$
  Memory capacity:
    #1: 2K bytes (An EPROM is fitted to the standard product)
    #2: 2K bytes (Only a socket is provided on the standard product)
  RAM (M5L2114LP x 2 or M58981S x 2)
    Memory address:
      4000_{16}~$43FF_{16}$
    Memory capacity:
      1K bytes
      Externally expandable up to a maximum of 64K bytes

I/O Address and I/O Capacity

I/O address:
  PPI (M5L8255AP)

I/O port		Signal description	Address
PPI #1	PA	PA10H~PA17H	6000_{16}
	PB	PB10H~PB17H	6001_{16}
	PC	PC10H~PC17H	6002_{16}
	C.W.	Control word	6003_{16}
PPI #2	PA	PA20H~PA27H	7000_{16}
	PB	PB20H~PB27H	7001_{16}
	PC	PC20H~PC27H	7002_{16}
	C.W.	Control word	7003_{16}

As two PPIs (Programmable Peripheral Interfaces) are provided on the board, the PCA8501 has I/O ports of 48 bits (8-bit x 6) in total.

Interrupt

5 Interrupts:
Five interrupts such as TRAP, RST 7.5, RST 6.5, RST 5.5, and INTR are provided. The TRAP has the highest priority, while the INTR has the lowest priority. The RST 7.5 will enable timer interrupt by means of a jumper wire.

Connectors

For bus extension (connector J1):
  Straight pin header, T-type, 50 pins
For I/O port connection (connector J2):
  Angle pin header, L-type, 60 pins

PIN CONFIGURATIONS

Connector J1

12V	V_{DD}	2	1	V_{BB}	5V
	GND	4	3	GND	
	GND	6	5	GND	
5V	V_{CC}	8	7	V_{CC}	5V
DATA BUS	DB1H ↔	10	9	↔ DB0H	(LSB) DATA BUS
	DB3H ↔	12	11	↔ DB2H	
	DB5H ↔	14	13	↔ DB4H	
(MSB)	DB7H ↔	16	15	↔ DB6H	
READ CONTROL OUTPUT	RDCL ←	18	17	→ ST0H	STATUS OUTPUT
WRITE CONTROL OUTPUT	WRCL ←	20	19	→ ST1H	
INTERRUPT ACK. OUTPUT	ITAL ←	22	21	→ IOMH	DATA TRANSFER CONTROL OUTPUT
ROM EXPANSION OUTPUT	ROMSL ←	24	23	← INTL	INTERRUPT REQUEST INPUT
READY INPUT	RDYH →	26	25	← RSIL	RESET INPUT
ADDRESS LATCH ENABLE OUTPUT	ALEH ←	28	27	→ RSOH	RESET OUTPUT
HOLD ACKNOWL-EDGE OUTPUT	HLAH ←	30	29	→ CLKH	CLOCK OUTPUT
HOLD INPUT	HLDL →	32	31	→ MEMSL	RAM EXPANSION OUTPUT
RESTART INTER-RUPT INPUT	R75L →	34	33	← TRPL	TRAP INTERRUPT INPUT
ADDRESS BUS	AB1H ←	36	35	→ AB0H	(LSB) ADDRESS BUS
	AB3H ←	38	37	→ AB2H	
	AB5H ←	40	39	→ AB6H	
	AB7H ←	42	41	→ AB6H	
	AB9H ←	44	43	→ AB8H	
	ABBH ←	46	45	→ ABAH	
	ABDH ←	48	47	→ ABCH	
(MSB)	ABFH ←	50	49	→ ABEH	

Connector J2

(5V)	V_{CC}	2	1	V_{CC}	(5V)
	GND	4	3	GND	
SERIAL DATA OUTPUT	SODL ←	6	5	← SIDL	SERIAL DATA INPUT
RESTART INTER-RUPT INPUT	R55L →	8	7	← R65L	RESTART INTER-RUPT INPUT
	NC	10	9	→ TIMH	TIMER OUTPUT
PPI #1 I/O PORT A	PA11H ↔	12	11	↔ PA10H	PPI #1 I/O PORT A
	PA13H ↔	14	13	↔ PA12H	
	PA15H ↔	16	15	↔ PA14H	
	PA17H ↔	18	17	↔ PA16H	
PPI #1 I/O PORT B	PB11H ↔	20	19	↔ PB10H	PPI #1 I/O PORT B
	PB13H ↔	22	21	↔ PB12H	
	PB15H ↔	24	23	↔ PB14H	
	PB17H ↔	26	25	↔ PB16H	
PPI #1 I/O PORT C	PC11H ↔	28	27	↔ PC10H	PPI #1 I/O PORT C
	PC13H ↔	30	29	↔ PC12H	
	PC15H ↔	32	31	↔ PC14H	
	PC17H ↔	34	33	↔ PC16H	
PPI #2 I/O PORT A	PA21H ↔	36	35	↔ PA20H	PPI #2 I/O PORT A
	PA23H ↔	38	37	↔ PA22H	
	PA25H ↔	40	39	↔ PA24H	
	PA27H ↔	42	41	↔ PA26H	
PPI #2 I/O PORT B	PB21H ↔	44	43	↔ PB20H	PPI #2 I/O PORT B
	PB23H ↔	46	45	↔ PB22H	
	PB25H ↔	48	47	↔ PB24H	
	PB27H ↔	50	49	↔ PB26H	
PPI #2 I/O PORT C	PC21H ↔	52	51	↔ PC20H	PPI #2 I/O PORT C
	PC23H ↔	54	53	↔ PC22H	
	PC25H ↔	56	55	↔ PC24H	
	PC27H ↔	58	57	↔ PC26H	
	GND	60	59	GND	

Note 3 : NC indicates no connection.

12

Memory and I/O Addresses

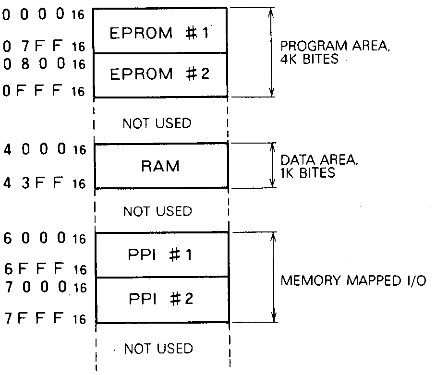
As memory and I/O addresses are fixed in this single-board computer, it is necessary to designate extra addresses besides those already assigned, if any additional external memory or I/O devices are to be employed.

I/O Address

	PPI #1				PPI #2			
	Port A	Port B	Port C	C.W.	Port A	Port B	Port C	C.W.
Memory mapped I/O address	6000_{16}	6001_{16}	6002_{16}	6003_{16}	7000_{16}	7001_{16}	7002_{16}	7003_{16}

The following addresses are inhibited from expanding externally, because there is no perfect redundancy in the decode of the PPIs: $6000_{16} \sim 6FFF_{16}$
$7000_{16} \sim 7FFF_{16}$

Memory Address Map



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	0~7	V
V_{BB}	Supply voltage		0.3~−15	V
V_{DD}	Supply voltage		−0.3~20	V
V_I	Input voltage		5.5	V
V_O	Output voltage		0~5.5	V
Topr	Operating free-air ambient temperature range		0~55	℃
Tstg	Storage temperature range		−30~70	℃

RECOMMENDED OPERATING CONDITIONS (Ta=0~55℃, unless otherwise noted.)

Symbol	Parameter	Limits Min	Nom	Max	Unit
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{BB}	Supply voltage	−4.75	−5	−5.25	V
V_{DD}	Supply voltage	11.6	12	12.6	V
V_{IH}	High-level input voltage	2			V
V_{IL}	Low-level input voltage			0.8	V

ELECTRICAL CHARACTERISTICS (Ta=0~55℃, Vcc=5V±5%, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits Min	Typ	Max	Unit
V_{OH}	High-level output voltage, PA11H~PC27H outputs	$I_{OH}=-50\mu A$	2.4			V
V_{OH}	High-level output voltage, AB0H~AB7H outputs	$I_{OH}=-900\mu A$	3.65			V
V_{OH}	High-level output voltage, AB8H~ABFH outputs	$I_{OH}=-300\mu A$	2.4			V
V_{OH}	High-level output voltage, RSOH, HLAH, CLKH and ALEL output	$I_{OH}=-300\mu A$	2.4			V
V_{OH}	High-level output voltage, other outputs	$I_{OH}=-400\mu A$	2.4			V
V_{OL}	Low-level output voltage, PA11H~PC27H outputs	$I_{OL}=1.8mA$			0	V
V_{OL}	Low-level output voltage, AB0H~AB7H outputs	$I_{OL}=16mA$			0.5	V
V_{OL}	Low-level output voltage, AB8H~ABFH outputs	$I_{OL}=1.9mA$			0.45	V
V_{OL}	Low-level output voltage, CLKH and HLAH output	$I_{OL}=4mA$			0.4	V
V_{OL}	Low-level output voltage, ALEL output	$I_{OL}=0.8mA$			0.4	V
V_{OL}	Low-level output voltage, other outputs	$I_{OL}=1.9mA$			0.45	V
I_{CC}	V_{CC} supply current				0.9	A
f_{CKL}	CPU clock frequency		4.866	4.9152	4.965	MHz

MELCS 85/2 MEMORY AND PARALLEL I/O EXPANSION BOARD

DESCRIPTION

The PCA8506 memory and parallel I/O expansion board is designed to be used with the PCA8501 or PCA8540 single-board computer. Memory, parallel I/O ports, and a timer are assembled on a 145 x 125 mm printed circuit board. The PCA8506 can easily be attached to the PCA8501 or PCA8540 single-board computer by using a bus-extension connector.

FEATURES

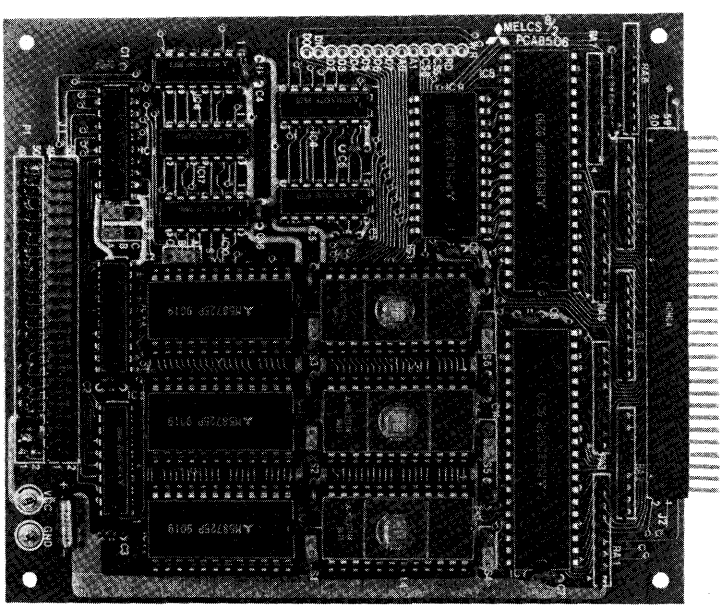
- Expansion board consisting of memory, parallel I/O ports, and a timer
- Memory capacity: 12K bytes (expandable in units of 2K bytes RAM or 2K bytes ROM)
- Programmable ports: 48 bits (8 bits x 6 ports)
- Programmable timer: 16 bits x 3
- Power supplied from the PCA8501 or PCA8540
- Compact dimensions (LxWxH): 125x145x17 mm

APPLICATIONS

- Personal computer expansion module
- Control equipment module

FUNCTION

The PCA8506 expansion board consists of up to 12K bytes memory, six 8-bit parallel I/O ports, along with 3 16-bit counters for timer application. The memory can easily be expanded in units of 2K bytes up to 12K bytes using any combination of M5L2716K EPROMs or M58725P static RAMs. The parallel I/O ports consist of 2 (programmable peripheral interfaces) (PPIs) each composed of 3 8-bit I/O ports. The timer consists of a programmable interval timer (PIT) which has 3 16-bit timer counters.



12

MELCS 85/2 MEMORY AND PARALLEL I/O EXPANSION BOARD

OPERATION

The address bus of the CPU is connected to other boards through the address bus buffer. The data bus is connected to the data input/output pins of memory, I/O, and a timer through the bidirectional data bus buffer. The data bus buffer is in an active state only when an IC device on the board is selected. The buffer is ready for output to external units only when the read signal RDCL from the CPU goes low.

Six 24-pin sockets are provided for memory, which are designed for M5L2716K EPROMs. Since the M5L2716K is compatible with the M58725P except for Vpp/WR (pin 21), if pin 21 is switched on the connector corresponding to a socket, a M58725P static RAM can be used in place of a M5L2716K EPROM in that socket. It is therefore possible to mix ROMs and RAMs in any order desired by the user.

Since addresses have been allocated on the memory map for the 2 parallel I/O ports and a timer the contents can be read and written in the same way memory is accessed.

All ports of PPIs are initiated to the input mode after the power is turned on, and remain in this mode until a control word is written. As soon as the counter is set by the control word, following the operation mode, the timer will begin to count.

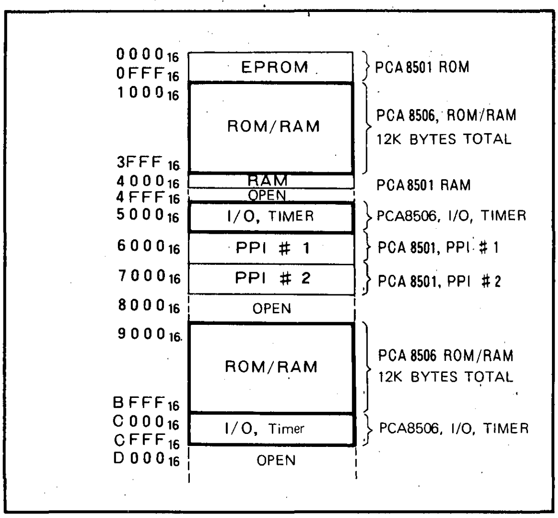
DIMENSIONS

(LxWxH) 125x145x17 mm

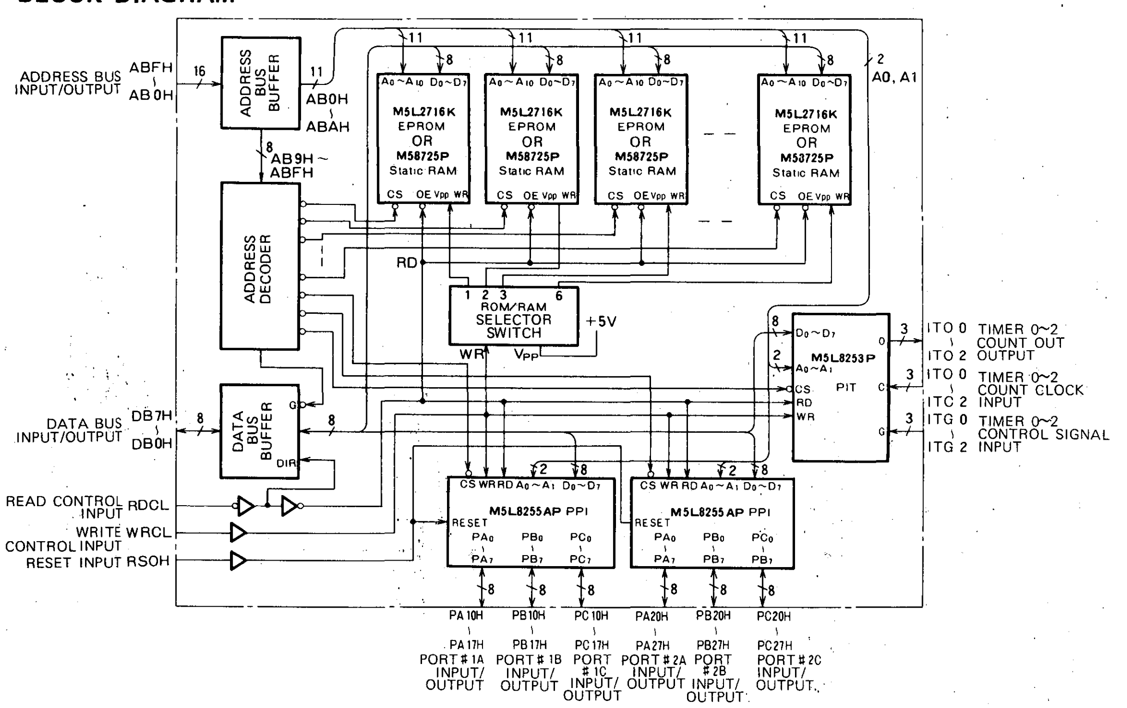
MEMORY AND I/O ADDRESSING

Both memory and I/O addresses can select two areas. When this board is added, different address areas from those of the main board should be selected.

MEMORY MAP



BLOCK DIAGRAM



MITSUBISHI ELECTRIC

MELCS 85/2 MEMORY AND PARALLEL I/O EXPANSION BOARD

SPECIFICATIONS

Memory Address and Memory Capacity

Memory Address (Note 1)

- # 1 : 1000_{16}~$17FF_{16}$
- # 2 : 1800_{16}~$1FFF_{16}$
- # 3 : 2000_{16}~$27FF_{16}$
- # 4 : 2800_{16}~$2FFF_{16}$
- # 5 : 3000_{16}~$37FF_{16}$
- # 6 : 3800_{16}~$3FFF_{16}$

Memory Capacity

- #1: 2K bytes (only the socket is supplied)
- #2: 2K bytes (only the socket is supplied)
- #3: 2K bytes (only the socket is supplied)
- #4: 2K bytes (only the socket is supplied)
- #5: 2K bytes (only the socket is supplied)
- #6: 2K bytes (only the socket is supplied)

Either the M5L2716K EPROM or M58725P RAM can be used in the sockets.

I/O and Timer Addresses and I/O Capacity

I/O and timer addresses (Note 1)

Name		Signal designation	Address
Port #1	PA	PA10H~PA17H	5000_{16}
	PB	PB10H~PB17H	5001_{16}
	PC	PC10C~PC17H	5002_{16}
	CW	Control Word	5003_{16}
Port #2	PA	PA20H~PA27H	5100_{16}
	PB	PB20H~PB27H	5101_{16}
	PC	PC20H~PC27H	5102_{16}
	CW	Control Word	5103_{16}
Timer	COUNTER 0	Interval timer 0	5200_{16}
	COUNTER 1	Interval timer 1	5201_{16}
	COUNTER 2	Interval timer 2	5202_{16}
	CW	Control Word	5203_{16}

Note 1: The address area can be altered by using an inline connector as follows:
Memory 9000_{16}~$BFFF_{16}$
I/O and timer $CXXX_{16}$

I/O Capacity

Port #1 : 8 bits x 3 ports = 24 bits
Port #2 : 8 bits x 3 ports = 24 bits

Interface

Bus : All signals are TTL compatible (fanout LS TTL 1 gate).

I/O and Timer : All signals are TTL compatible.

Connectors

1. P1 (for bus) : Straight dip-type, 50 pins.
2. J1 (for bus) : Straight pin header, T-type, 50 pins.
3. J2 (for I/O) : Angle pin header, L-type, 60 pins.

Power

5V, 1A maximum (when six M5L2716Ks are loaded).

PIN CONFIGURATION

Connectors P1 and J1

		P1, J1			
		2	1		
	GND	4	3	GND	
	GND	6	5	GND	
5V	Vcc	8	7	Vcc	5V
DATA BUS	DB1H ↔	10	9	↔ DB0H	DATA BUS
DATA BUS	DB3H ↔	12	11	↔ DB2H	DATA BUS
DATA BUS	DB5H ↔	14	13	↔ DB4H	DATA BUS
DATA BUS	DB7H ↔	16	15	↔ DB6H	DATA BUS
READ CONTROL INPUT	RDCL →	18	17	NC	
WRITE CONTROL INPUT	WRCL →	20	19	NC	
	NC	22	21	NC	
	NC	24	23	NC	
	NC	26	25	NC	
	NC	28	27	← RSOH	RESET INPUT
	NC	30	29	NC	
	NC	32	31	NC	
	NC	34	33	NC	
ADDRESS BUS	AB1H →	36	35	← AB0H	ADDRESS BUS
ADDRESS BUS	AB3H →	38	37	← AB2H	ADDRESS BUS
ADDRESS BUS	AH5H →	40	39	← AB4H	ADDRESS BUS
ADDRESS BUS	AB7H →	42	41	← AB6H	ADDRESS BUS
ADDRESS BUS	AB9H →	44	43	← AB8H	ADDRESS BUS
ADDRESS BUS	ABBH →	46	45	← ABAH	ADDRESS BUS
ADDRESS BUS	ABDH →	48	47	← ABCH	ADDRESS BUS
ADDRESS BUS	ABFH →	50	49	← ABEH	ADDRESS BUS

NC : NO CONNECTION

Connector J2

		J2			
I/O PORTS #1A	PA 16H ↔	2	1	↔ PA 17 H	I/O PORTS #1A
I/O PORTS #1A	PA 14H ↔	4	3	↔ PA 15 H	I/O PORTS #1A
I/O PORTS #1A	PA 12H ↔	6	5	↔ PA 13 H	I/O PORTS #1A
I/O PORTS #1A	PA 10H ↔	8	7	↔ PA11 H	I/O PORTS #1A
#1B	PB 16H ↔	10	9	↔ PB 17 H	#1B
#1B	PB 14H ↔	12	11	↔ PB 15 H	#1B
#1B	PB 12H ↔	14	13	↔ PB 13 H	#1B
#1B	PB 10H ↔	16	15	↔ PB 11H	#1B
#1C	PC 16H ↔	18	17	↔ PC 17 H	#1C
#1C	PC 14H ↔	20	19	↔ PC 15H	#1C
#1C	PC 12H ↔	22	21	↔ PC 13H	#1C
#1C	PC 10H ↔	24	23	↔ PC 11H	#1C
I/O PORTS #2A	PA26H ↔	26	25	↔ PA 27H	I/O PORTS #2A
I/O PORTS #2A	PA24H ↔	28	27	↔ PA 25H	I/O PORTS #2A
I/O PORTS #2A	PA22A ↔	30	29	↔ PA 23H	I/O PORTS #2A
I/O PORTS #2A	PA20H ↔	32	31	↔ PA 21H	I/O PORTS #2A
#2B	PB26H ↔	34	33	↔ PB27 H	#2B
#2B	PA24H ↔	36	35	↔ PB 25H	#2B
#2B	PB22H ↔	38	37	↔ PB 23H	#2B
#2B	PB20H ↔	40	39	↔ PB 21H	#2B
#2C	PC26H ↔	42	41	↔ PC 27H	#2C
#2C	PC24H ↔	44	43	↔ PC 25H	#2C
#2C	PC22H ↔	46	45	↔ PC 23H	#2C
#2C	PC20H ↔	48	47	↔ PC 21H	#2C
	CS 4 L ←	50	49	→ CS 3L	
TIMERS	ITG 0 →	52	51	NC	
TIMERS	ITC 1 →	54	53	→ ITO 0	TIMERS
TIMERS	ITO 1 ←	56	55	← ITG 1	TIMERS
TIMERS	ITG 2 →	58	57	← ITC 2	TIMERS
TIMERS	ITC 0 →	60	59	→ ITO 2	TIMERS

MELCS 85/2 MEMORY AND PARALLEL I/O EXPANSION BOARD

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	0～6.5	V
V_I	Input voltage	With respect to GND	5.5	V
V_O	Output voltage	With respect to GND	5.5	V
Topr	Operating free-air ambient temperature range		0～55	°C
Tstg	Storage temperature range		−30～70	°C

RECOMMENDED OPERATING CONDITIONS (Ta=0～55°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{IH}	High-level input voltage	2			V
V_{IL}	Low-level input voltage			0.8	V

ELECTRICAL CHARACTERISTICS (Ta=0～55°C, V_{CC}=5V±5%, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage PA11H～PC27H output	$I_{OH}=-50\mu A$	2.4			V
V_{OH}	High-level output voltage, IT00～IT02 output	$I_{OH}=-150\mu A$	2.4			V
V_{OL}	Low-level output voltage, PA11H～PC27H output	$I_{OL}=1.6mA$			0.4	V
V_{OL}	Low-level output voltage, IT00～IT02 output	$I_{OL}=1.6mA$			0.4	V

MELCS 85/2 MEMORY AND SERIAL I/O EXPANSION BOARD

DESCRIPTION

The PCA8507 memory and serial I/O expansion board is designed to be used with the PCA8501 or PCA8540 single-board computer. Memory, a serial I/O port and a timer are assembled on a 145 x 125 mm printed circuit board. The PCA8507 can easily be attached to the PCA8501 or PCA-8540 single-board computer by using a bus-extension connector.

FEATURES

- Expansion board consists of memory, a serial I/O and a timer
- Memory capacity: 12K bytes (expandable in units of 2K bytes RAM or 2K bytes ROM)
- Serial I/O port and TTL, RS-232-C interface
- Programmable timer, 16 bits x 3
- Power supply from the PCA8501 or PCA8540
- Compact, dimensions (LxWxH): 125x145x17mm

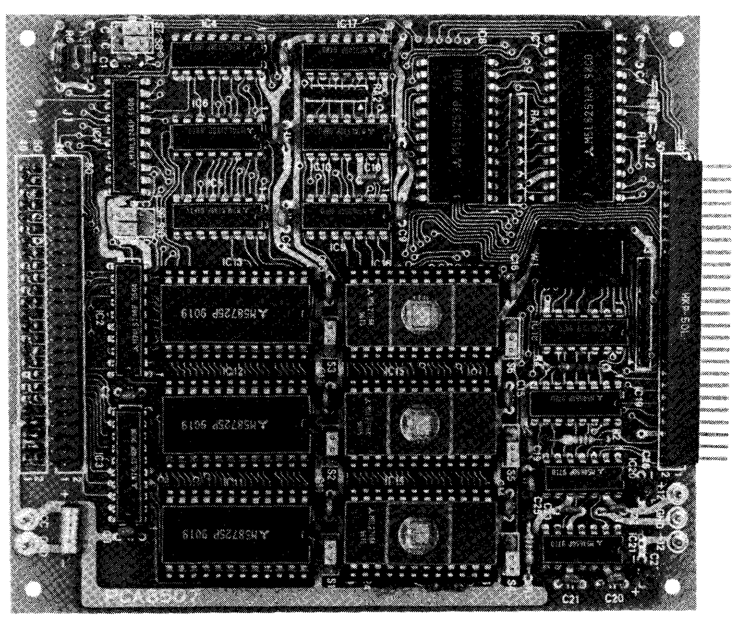
APPLICATIONS

- Personal computer expansion module
- Control equipment module
- Data terminal module

FUNCTION

The PCA8507 expansion board consists of up to 12K bytes of memory, serial I/O port, interface for TTL level and RS-232-C output, along with 3 16-bit counters for timer application.

The memory can easily be expanded in units of 2K bytes up to 12K bytes using any combination of M5L-2716K EPROMs and M5825P static RAMs. The serial I/O port consists of a universal synchronous asynchronous receiver transmitter (USART) for changing parallel/serial and formatting the string in the specified format. Interfaces are provided between the USART and the TTL level or RS-232-C output. The interface is selected by a jumper connection. The timer consists of a programmable interval timer (PIT) which has 3 16-bit timer counters. One of the timers is used by the USART for controlling the baud rate of serial data transfer.



MELCS 85/2 MEMORY AND SERIAL I/ O EXPANSION BOARD

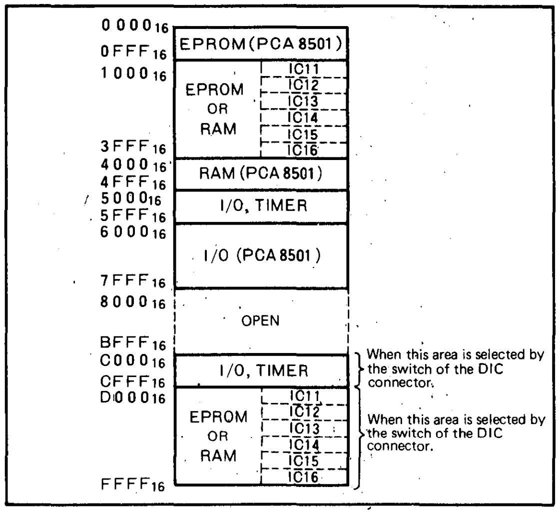
OPERATION

The address bus of the CPU is connected to other boards through the address bus buffer. The data bus is connected to the data input/output pins of memory, I/O, and a timer through the bidirectional data bus buffer. The data bus buffer is in an active state only when an IC device on the board is selected. The buffer is ready for output to external units only when the read signal RDCL from the CPU goes low.

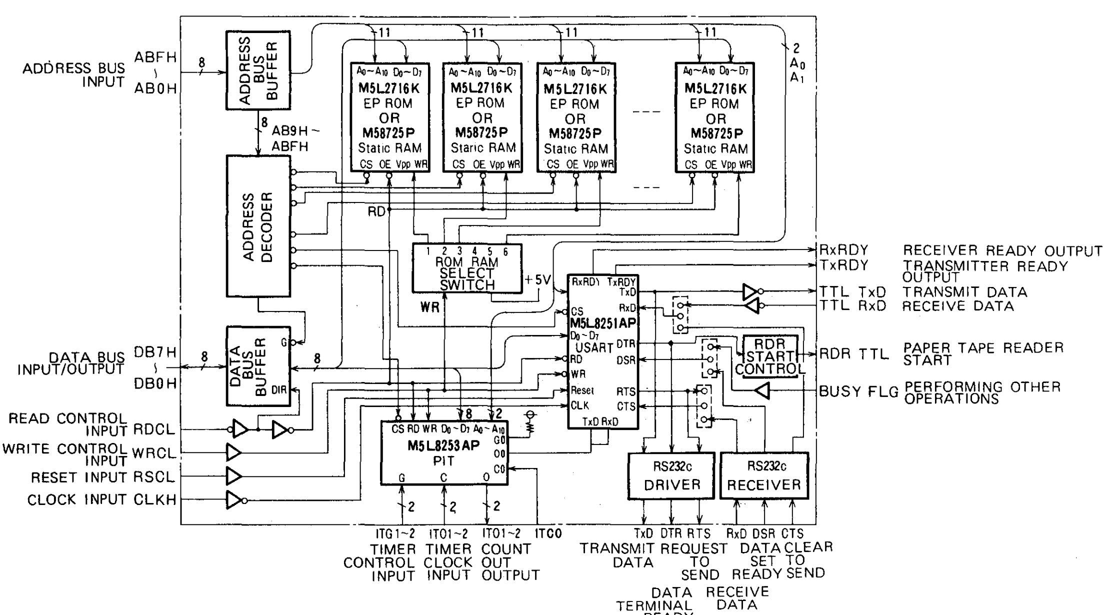
Six 24-pin sockets are provided for memory, which are designed for M5L2716K EPROMs. Since the M5L2716K is compatible with the M58725P except for V_{PP}/WR (pin 21), if pin 21 is switched on the connector corresponding to a socket, an M58725P static RAM can be used in place of an M5L2716K EPROM in that socket. It is therefore possible to mix ROMs and RAMs in any order desired by the user.

Since addresses have been allocated on the memory map for the USART as a serial I/O port and a timer the contents can be read and written in the same way memory is accessed. TTL and standard RS-232-C interfaces are built on the board to interface between the serial I/O port and peripheral devices. Selection of one or the other interface is done by a jumper in the jumper socket. The timer consists of 3 16-bit counters. One of the counters is used as a clock by the USART in setting the baud rate.

MEMORY MAP



BLOCK DIAGRAM



MITSUBISHI ELECTRIC

MELCS 85/2 MEMORY AND SERIAL I/O EXPANSION BOARD

SPECIFICATIONS

Memory Address and Memory Capacity

Memory Address (Note 1)

- # 1 : $1000_{16} \sim 17FF_{16}$
- # 2 : $1800_{16} \sim 1FFF_{16}$
- # 3 : $2000_{16} \sim 27FF_{16}$
- # 4 : $2800_{16} \sim 2FFF_{16}$
- # 5 : $3000_{16} \sim 37FF_{16}$
- # 6 : $3800_{16} \sim 3FFF_{16}$

Memory Capacity

- #1: 2K bytes (only the socket is supplied)
- #2: 2K bytes (only the socket is supplied)
- #3: 2K bytes (only the socket is supplied)
- #4: 2K bytes (only the socket is supplied)
- #5: 2K bytes (only the socket is supplied)
- #6: 2K bytes (only the socket is supplied)

Either the M5L2716K EPROM or M58725P RAM can be used in the sockets.

I/O and Timer Addresses and I/O Capacity

I/O and timer addresses (Note 1)

Name		Signal designation	Address
Serial port	TD	Parallel data	5000_{16}
	CW	Control word	5001_{16}
Timer	COUNTER 0	Interval timer 0	5100_{16}
	〃 1	Interval timer 1	5101_{16}
	〃 2	Interval timer 2	5102_{16}
	CW	Control word	5103_{16}

Note 1 The address area can be altered by using an inline connector as follows:
Memory $D000_{16} \sim FFFF_{16}$
I/O and Timer $CXXX_{16}$

INTERFACE

Bus : All signals are TTL compatible (fanout LS TTL 1 gate).

Timer : All signals are TTL compatible (fanout TTL 1 gate).

Serial I/O : TTL level or RS-232-C standard interface.

CONNECTORS

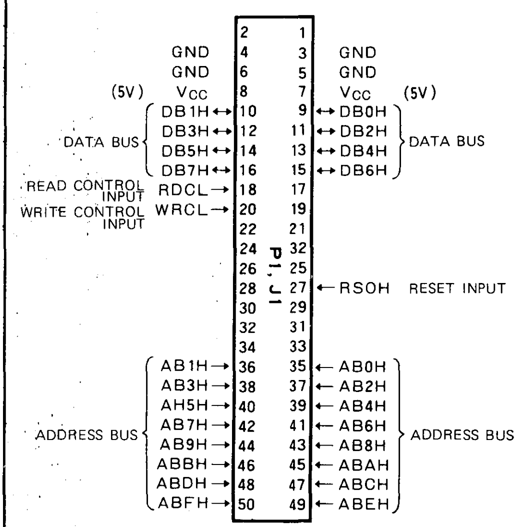
1. P1 (for bus): Straight dip-type, 50 pins
2. J1 (for bus): Straight pin header, T-type, 50 pins
3. J2 (for I/O): Angle pin header, L-type, 50 pins

POWER

5V, 1A maximum (when six M5L2716Ks are loaded)
±12V (when used as an RS-232-C interface)

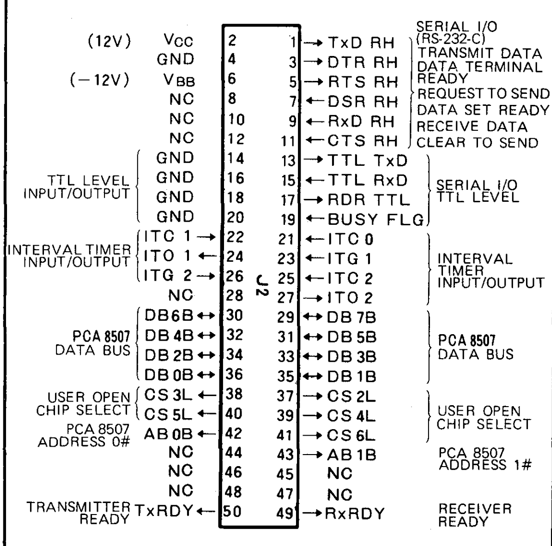
PIN CONFIGURATIONS

Connectors P1 and J1



In the above diagram when a name is not shown for a pin, then that pin is not used, only connected to the same pin number of P1 and J1.

Connector J2



NC: NO CONNECTION

12

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	0～6.5	V
V_{DD}	Supply voltage (plus supply for RS-232-C)		15	V
V_{BB}	Supply voltage (minus supply for RS-232-C)		−15	V
V_I	Input voltage		5.5	V
V_O	Output voltage		5.5	V
Topr	Operating free-air ambient temperature range		0～55	V
Tstg	Storage temperature range		−30～70	°C

RECOMMENDED OPERATING CONDITIONS (Ta=0～55°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{IH}	High-level input voltage	3		V_{CC}	V
V_{IL}	Low-level input voltage	0		0.65	V
V_{DD}	Supply voltage (plus supply for RS-232-C)	10.8	12	13.2	V
V_{BB}	Supply voltage (minus supply for RS-232-C)	−13.2	−12	−10.8	V

ELECTRICAL CHARACTERISTICS (Ta=0～55°C, V_{CC}=5V±5%, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
V_{OH}	High-level output voltage	DB0B～DB7B	I_{OH}=− 3 mA	2.4			V
	High-level output voltage	AB0B, AB1B	I_{OH}=− 3 mA	2.4			V
	High-level output voltage	IT01, IT02	I_{OH}=−150μA	2.4			V
	High-level output voltage	CS2L～CS6L	I_{OH}=−400μA	2.7			V
	High-level output voltage	TxDRH, DTRRH, RTSRH	V_{CC+}=10.8V, V_{CC-}=−13.2V V_{IL}=0.8V, R_L=3～7kΩ	5			V
V_{OL}	Low-level output voltage	DB0B～DB7B	I_{OL}=12mA			0.4	V
	Low-level output voltage	AB0B, AB1B	I_{OL}=12mA			0.4	V
	Low-level output voltage	IT01, IT02	I_{OL}=1.6mA			0.45	V
	Low-level output voltage	CS2L～CS6L	I_{OL}= 4 mA			0.4	V
	Low-level output voltage	TxDRH, DTRRH, RTSRH	V_{CC+}=10.8V, V_{CC-}=−10.8V V_{IH}= 2 V, R_L=3～7kΩ			−5	V

MELCS 85/3 VOICE GENERATING SINGLE-BOARD COMPUTER

DESCRIPTION

The PCA8520 is a voice generating single-board computer. It consists of an 8-bit M5L8085AP microprocessor, memory, I/O interface, voice reproducing IC, and is fabricated on a single 125 x 145 mm printed circuit board. Voice data is first recorded in EPROMs and is changed into voice data through delta modulation system.

FEATURES

Type	Contents
PCA8520G01	Single-board computer only
PCA8520G02	PCA8520G01 single-board computer .... 1 pc. M5L2716K (007) EPROM for control program storage .... 1 pc. (008~014) EPROMs for voice data storage .... 7 pcs. Speaker .... 1 pc. Instruction manual .... 1 vol.

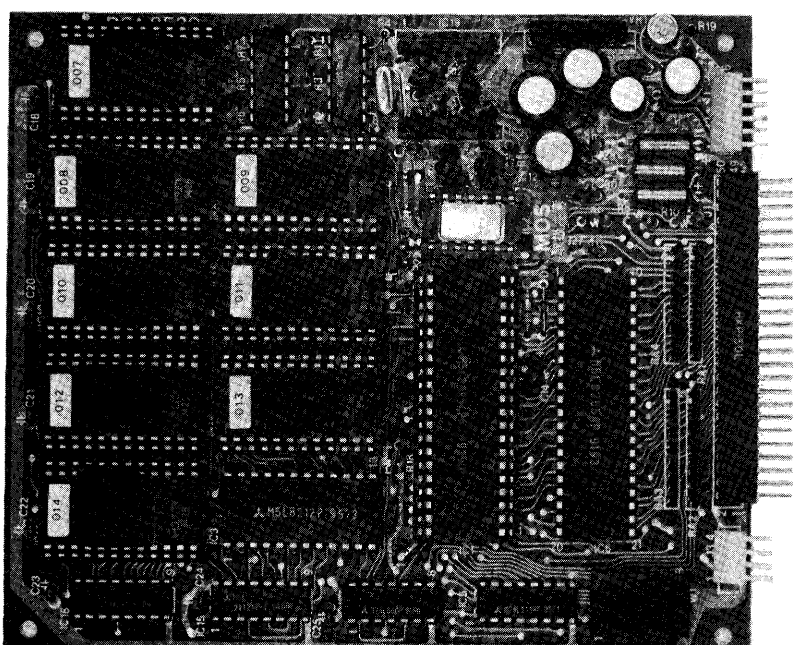
- A single-board computer complete with CPU, memory, I/O interface and voice reproducing IC.
- Storage capacity of the EPROM:
  using M5L2716K: 16K bytes (max)
  using M5L2732K: 32K bytes (max)
- Storage capacity of the RAM: 256 bytes
- I/O interface: 24 bits (8 bits x 3)
- Voice recording time:
  using M5L2716K: 9 seconds (max)
  using M5L2732K: 18 seconds (max)
- Voice maximum output power (at V_{CC2} = 9V): 1W (typ)
- Compact dimensions (LWH) 125x145x20 mm

APPLICATIONS

- An alarm device to be used in factories, offices, etc.
- A recorded sales message device
- An audio output information device
- A device to give voice operation instructions
- Numerical value response for measurement instruments and calculators

FUNCTION

The PCA8520 is a single-board computer with a voice generating function, and is designed around Mitsubishi's M5L8085AP CPU, its LSI family, and voice reproducing IC. It comes with 16K bytes (M5L2716K x 8) or 32K bytes (M5L2732K x 8) of read-only memory and 256 bytes (M5L2112AP) of random-access memory. The



12

MELCS 85/3 VOICE GENERATING SINGLE-BOARD COMPUTER

PCA8520 has 1 M5L8255AP programmable peripheral interface (PPI) which offers 24 bits (8 bits x 3) of programmable I/O port.

Voice reproducing is performed through an IC for delta demodulation, a low-pass filter, and a power amplifier. Voice data in the ROM can be sent to the voice reproducing circuit by program control, and then output with 1W of power.

A nine-second message can be output when using 8 M5L2716KS. An eighteen-second message is possible when using 8 M5L2732Ks. Voice data can be output at both syllable- and word-levels, and can be edited under program control.

OPERATION

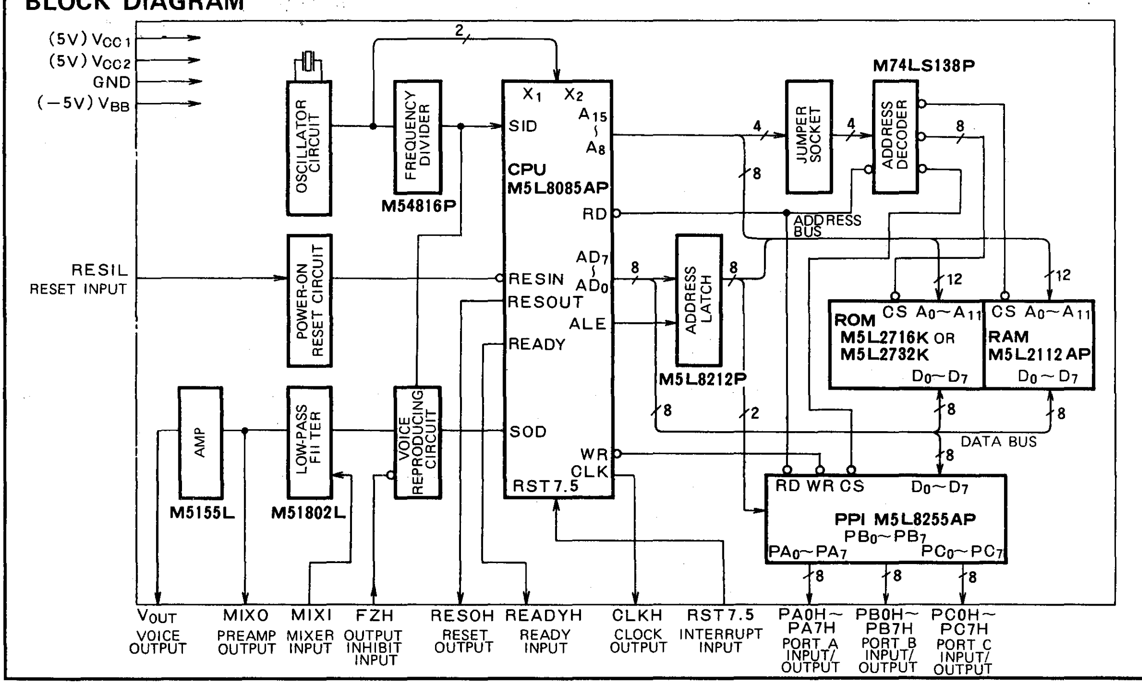
The M5L8085AP CPU executes programs stored in the ROM synchronizing with a quartz ocillator clock. The frequency of this clock is divided by 256 and is supplied to the SID terminal of the CPU and the input of the IC for delta demodulation. The voice can be generated using voice data. This voice data, which is stored in the ROM, is converted parallel to serial and is sent to the SOD terminal in sequence from the most-significant bit.

The M5L2112AP RAM can be used as a data stack, etc. The M5L8255AP PPI can be used for external data inputs or outputs.

BLOCK DIAGRAM NOTATION

Name	Function
Reset circuit	A system reset signal is generated when power is turned on.
Oscillator circuit	The clock is supplied to the CPU and the frequency divider circuit.
Frequency divider circuit	The frequency of the oscillator clock is devided by 256 and is supplied to the voice reproducing circuit.
CPU	Executes the program
Address latch	As the data and low-order address signals are sent from $AD_0 \sim AD_7$ terminals of the CPU using time-sharing technique, only the address signal is latched into the address latch circuit by the ALE timing signal.
Jumper socket	M5L2716Ks or M5L2732Ks can be selected by simply changing the jumper wire in the jumper socket.
Address decoder	Generates the selection signal of a ROM, RAM, and PPI decoding the high-order bits of the address signal.
ROM	Memory to store program and voice data
RAM	Memory to store data stack, temporaly data, etc.
PPI	Is used for external data inputs and outputs
Voice reproducing circuit	Reproduces voice waveform from the digital signal which is sent from the SOD terminal of the CPU
Low-pass filter	This filter only passes low-frequency voice signals.
Amplifier	Voice signal passed through the low-pass filter is amplified to 1W.

BLOCK DIAGRAM



MELCS 85/3 VOICE GENERATING SINGLE-BOARD COMPUTER

SPECIFICATIONS

Item		Contents
Method		CVSD Method
CPU device		Mitsubishi M5L8085AP
Cycle time		Basic instruction time 2.2μs (at 3.6MHz crystal oscillator frequency)
Memory	ROM	16K bytes (max) using M5L2716Ks address 0000_{16}~$3FFF_{16}$
		32K bytes (max) using M5L2732Ks address 0000_{16}~$7FFF_{16}$
	RAM	256 bytes address $C000_{16}$~$C0FF_{16}$
I/O interface		Programmable I/O ports: 8 bits x 3 ports (PPI M5L8255AP) address 8000_{16}~8003_{16}
Voice recording time		9 seconds using M5L2716Ks (max) 18 seconds using M5L2732Ks (max)
Voice maximum output power		1W (V_{CC2}=5V, THD=10% f=1kHz)
Interrupt		1 interrupt, 1 level
Auxillary units		MELCS 85/1 microcomputer console voice data unit.
Power supply		5V (Two power sources: V_{CC1}, V_{CC2}), -5V
Connectors		Angle pin, header type 50 pins (for the I/O ports) Angle pin, header type 6 pins (for voice output) Angle pin, header type 4 pins (for power)
Physical dimensions		(LxWxH): 125x145x20mm

CONNECTORS

I/O Ports: (Connector J1)
angle pin header L-type 50 pins
Power: (Connector J2)
angle pin header L-type 4 pins
Voice Output: (Connector J3)
angle pin header L-type 6 pins

PIN CONFIGURATIONS

Connector J1

Function	Signal	Pin	Pin	Signal	Function
(5V)	V_{SS1}	2, 4	1, 3	V_{CC1}	(5V)
	GND	6, 8	5, 7	GND	
	NC	10	9	←FZH	VOICE OUTPUT INHIBIT
	NC	12	11	←RST75H	INTERRUPT INPUT
I/O PORT A	PA1H↔	14	13	↔PA0H	I/O PORT A
I/O PORT A	PA3H↔	16	15	↔PA2H	I/O PORT A
I/O PORT A	PA5H↔	18	17	↔PA4H	I/O PORT A
I/O PORT A	PA7H↔	20	19	↔PA6H	I/O PORT A
I/O PORT B	PB1H↔	22	21	↔PB0H	I/O PORT B
I/O PORT B	PB3H↔	24	23	↔PB2H	I/O PORT B
I/O PORT B	PB5H↔	26	25	↔PB4H	I/O PORT B
I/O PORT B	PB7H↔	28	27	↔PB6H	I/O PORT B
I/O PORT C	PC1H↔	30	29	↔PC0H	I/O PORT C
I/O PORT C	PC3H↔	32	31	↔PC2H	I/O PORT C
I/O PORT C	PC5H↔	34	33	↔PC4H	I/O PORT C
I/O PORT C	PC7H↔	36	35	↔PC6H	I/O PORT C
RESET OUTPUT	RESOH←	38	37	↔RESIL	RESET INPUT
READY INPUT	READYH→	40	39	↔CLKH	CLOCK OUTPUT
	NC	42	41	NC	
(−5V)	V_{BB}	44	43	V_{BB}	(−5V)
(5V)	V_{CC2}	46	45	V_{CC2}	(5V)
	GND	48, 50	47, 49	GND	

NC: NO CONNECTION

Connector J2

Pin	Signal	Description	
1	V_{OUT}	VOICE OUTPUT PIN	VOICE OUTPUT
2	GND		VOICE OUTPUT
3	MIXI	MIXING INPUT PIN	VOICE OUTPUT
4	GND		VOICE OUTPUT
5	MIXO	PREAMP OUTPUT PIN	VOICE OUTPUT
6	GND		VOICE OUTPUT

Connector J3

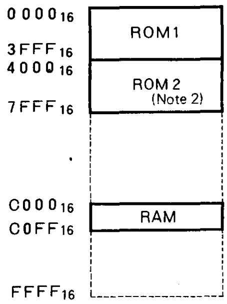
Pin	Signal	Voltage	
1	V_{CC1}	5V	POWER SUPPLY
2	V_{CC2}	5V	POWER SUPPLY
3	GND		POWER SUPPLY
4	V_{BB}	−5V	POWER SUPPLY

I/O ADDRESS

	PPI			
	Port A	Port B	Port C	C, W
I/O Address	80_{16}	81_{16}	82_{16}	83_{16}

MELCS 85/3 VOICE GENERATING SINGLE-BOARD COMPUTER

MEMORY ADDRESS MAP



Note 2: ROM2 is additional storage area when 8 M5L2732Ks are used.
3: ROM is fully decoded, but RAM and PPI are not.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC1}	Supply voltage	With respec to GND	0~7	V
V_{CC2}	Supply voltage		0~15	V
V_{BB}	Supply voltage		−15~0	V
V_I	Input voltage		5.5	V
V_O	Output voltage		0~5.5	V
Topr	Operational free-air ambient temperature range		0~55	℃
Tstg	Storage temperature range		−30~70	℃

RECOMMENDED OPERATING CONDITIONS (Ta = 0 ~ 55℃, unless otherwise noted.)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC1}	Supply voltage	4.75	5	5.25	V
V_{CC2}	Supply voltage	4	5	12	V
V_{BB}	Supply voltage	−18	−5	−4	V
V_{IH}	High-level input voltage	2			V
V_{IL}	Low-level input voltage			0.8	V

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 55℃, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage, RESOH, CLKH	$I_{OH} = -400\mu A$	2.4			V
V_{OL}	Low-level output voltage, RESOH, CLKH	$I_{OL} = 2mA$			0.45	V
V_{OH}	High-level output voltage, PA0H~PC7H	$I_{OH} = -200\mu A$	2.4			V
V_{OL}	Low-level output voltage, PA0H~PC7H	$I_{OL} = 1.7mA$			0.45	V
V_{IH}	High-level input voltage, RESIL		2.4		V_{CC} +0.5	V
V_{IL}	Low-level input voltage, RESIL		−0.3		0.8	V
V_{IH}	High-level input voltage, READY, RST75H		2.2		V_{CC} +0.5	V
V_{IL}	Low-level input voltage, READY, REST75H		−0.3		0.8	V
P_O	Voice maximum output power	THD = 10%, f = 1kHz, V_{CC2} = 9V, R_L = 8Ω, Ta = 25°C	0.7	1		W
I_{CC1}	Supply current from V_{CC1}	When 8 M5L2716K EPROMs are used.		450	900	mA
I_{CC2}	Supply current from V_{CC2}				400	mA
I_{BB}	Supply current from V_{BB}				100	mA

MELCS 85/2 COLOR TV DISPLAY SINGLE-BOARD COMPUTER

DESCRIPTION

The PCA8540 is a single-board computer of the MELPS 85 LSI family. The TV interface is fabricated on a single 125 x 145 mm printed circuit board. It provides for screen displaying with a resolution of 256 x 192 elements maximum in 2 colors, up to 8 colors in semigraphic 4, or up to 64 ASCII coded characters. A simple connection to the antenna terminal allows it to be used with a home color TV receiver. The PCA8540 also produces composite video signals that can be connected directly to the video monitor.

FEATURES

Type	Function
PCA8540G01	For home-use TV with output of NTSC system signals for Japan Channel 1 or 2 Contains no EPROMs Contains only one M58725P for screen memory
PCA8540G02	For video monitor TV with monochrome video monitor signals Contains no EPROMs Contains only one M58725P for screen memory

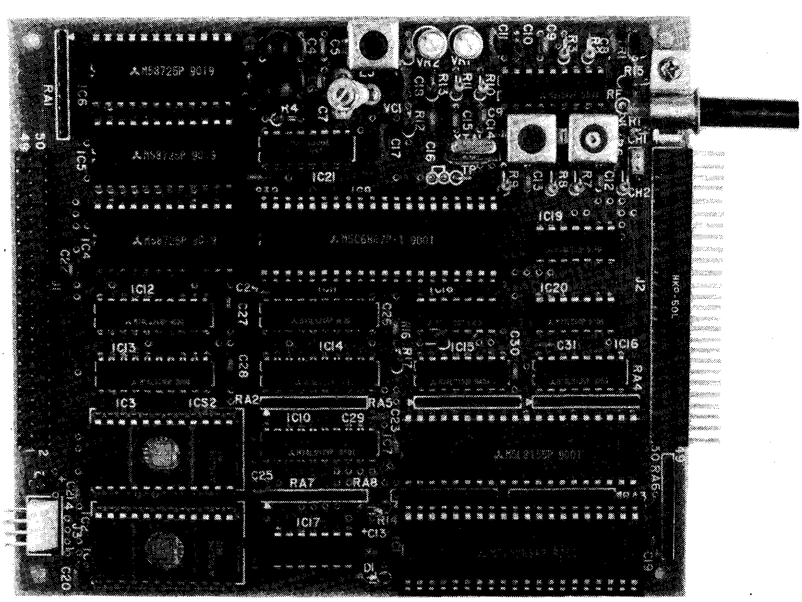
- A single-board computer complete with CPU, memory, I/O and TV interface
- Enables up to 256(H) x 192(V) elements graphic display on a home color TV receiver (or monochrome video monitor)
- Up to 64 characters can be displayed
- The 64 ASCII characters are stored on an internal character generator ROM and can be displayed together with semigraphics 4 mode
- Provide 9 colors on screen: green, yellow, blue, red, light gray, cyan, magenta, orange and black
- ROM 4K bytes (max) + RAM 256 bytes or ROM 2K bytes + RAM 2.25K bytes
- Programmable I/O port with timer: 22 bits
- Compact: dimensions (LxWxH): 125x145x20 mm
- Expandable memory and I/O (using memory I/O expansion board PCA8506 or PCA8507)

APPLICATIONS

- TV games
- Personal computer
- Data terminal with graphic capability
- Display terminal for microcomputer systems
- Commercial advertising display
- Slave computer for a MELCS 85/2 system

FUNCTION

The PCA8540 is a single-board computer, with color TV display capabilities designed to be compatible with the Mitsubishi M5L8085AP CPU and its LSI family as well as the VDG (video display generator) LSI M5C6847P-1. The PCA8540 comes with 4K bytes of ROM + 256 bytes



12

MELCS 85/2 COLOR TV DISPLAY SINGLE-BOARD COMPUTER

of RAM or 2K bytes of ROM + 2,304 bytes of RAM along with 3 I/O ports (22 bits). ROM, RAM and I/O are provided by the M5L2716K x 2 + M5L8155A or M5L2716K + M58725P + M5L8155A.

The TV interface of the G01 system consists of a VDG LSI M5C6847P-1 (TV interface), an M51342P RF modulator IC and 3 M58725P 16K static RAMs which are used for screen display memory. The G02 system has a video amp circuit instead of an M51342P.

As the various display modes can be programmed using an M5C6847P-1, the following can be displayed.

- Character display, pattern stored on internal ROM
- Reverse character display (one character)
- Semigraphics 4 (up to 8 colors)
- Semigraphics 6 (up to 4 colors)
- 64 x 64 4 colors ● 128 x 64 2 colors
- 128 x 64 4 colors ● 128 x 96 2 colors
- 128 x 96 4 colors ● 128 x 192 2 colors
- 128 x 192 4 colors ● 256 x 192 2 colors

The PCA8506 and PCA8507 are used, for memory I/O expansion boards, to expand to a maximum of 16K bytes of ROM or RAM, an RS-232-C serial interface can be used.

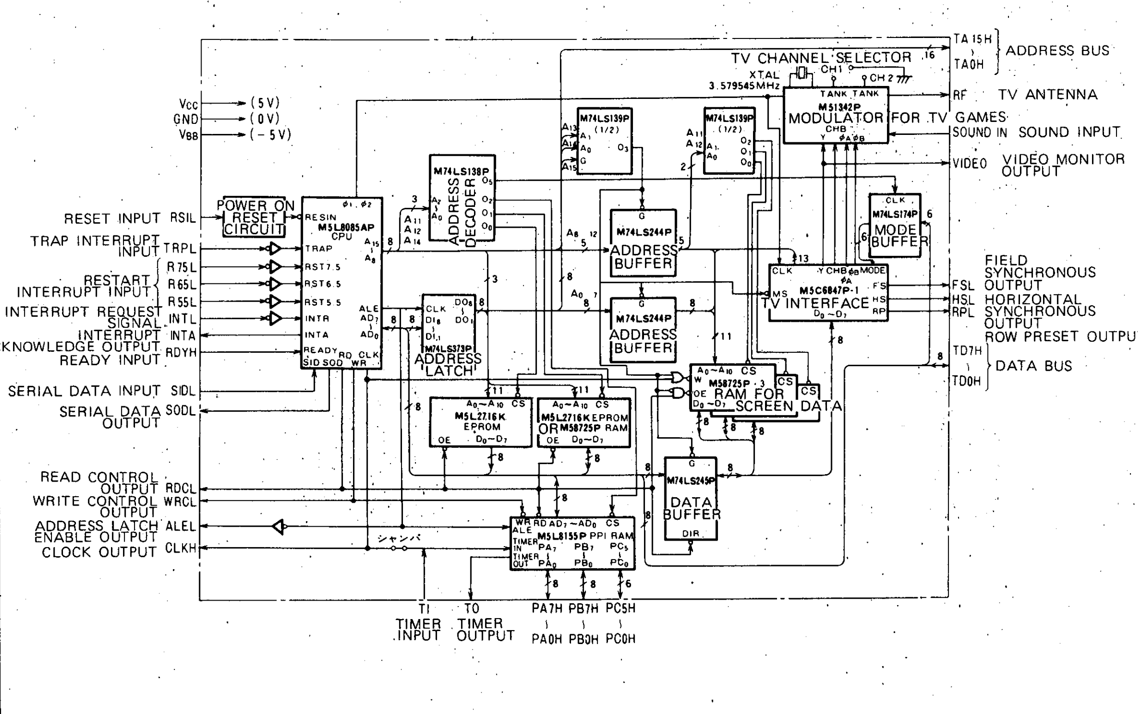
OPERATIONS

The program for the M5L8085AP CPU is normally stored on 2 M5L2716K EPROMs (2 x 2K bytes) and an M5L-8155P RAM (256 bytes) but 1 M5L2716K EPROM can be replaced by an M58725P RAM (2K bytes). Data transmission to and from external sources is done through the ports of the M5L8155P.

There is a data buffer between the M5C6847P-1 and the CPU on the address and data bus. This allows the M5C-6847P-1 to operate independently of the CPU when reading information from the M58725P RAM for screen data. It adds synchronous signal before it is output serially to the M51342P TV game modulator. The signal includes the intensity and color signals which are modulated by the M51342P into NTSC system TV signals for channel 1 or 2. The M5C6847P-1's composite video signal can be used for input to the monochrome video monitor.

When the CPU accesses the RAM, addresses 6000_{16}~$77FF_{16}$ for screen data, $\overline{MS}$ of the M5C6847P-1 will be at low-level and the address output will be in high-impedance state. During this period the CPU can change the contents of the RAM for screen data. The CPU can also change the display mode of the M5C6847P-1 through the data bus by accessing mode set address 4800_{16}.

BLOCK DIAGRAM (PCA8540G01)



MITSUBISHI ELECTRIC

MELCS 85/2 COLOR TV DISPLAY SINGLE-BOARD COMPUTER

SPECIFICATIONS

Item	Description
Method	8-bit parallel operation
CPU Component	Mitsubishi's M5L8085AP (equivalent to the intel 8085A)
Cycle time	Basic instruction time 2.23μs (at clock frequency 1.79 MHz)
Memory	EPROM 4K bytes (M5L2716K x 2) Address 0000_{16}~$0FFF_{16}$ or 2K bytes (M5L2716K x 1) (Note 1) Address 0000_{16}~$07FF_{16}$ RAM 256 bytes (M5L8155P) Address 4000_{16}~$40FF_{16}$ or 2304 bytes (M5L8155P + M58725P) Address 08000_{16}~$0FFF_{16}$ (Note 1) 4000_{16}~$40FF_{16}$ Screen Memory (Note 2) 6K bytes (M5872P x 3) Address 6000_{16}~$77FF_{16}$
I/O	Programmable port 22 bits (M5L8155P) Address 4100_{16}~4105_{16} Serial input/output Opens SID, SOD of CPU
Video output	G01: NTSC system, Japan, channel 1 or 2 G02: Monochrome composite video monitor signal
Display method	Priority CPU
Interrupt	5-level (INTR, RST55, RST65, RST75, TRAP)
Support device	PCA0803 (program checker) can be used PC8500 (portable microcomputer console) can be used.
Power supply	G01: 5V ±5%, -5V ±5% G02: 5V ±5%
Applicable connector	Straight pin header 50 pins (for bus extension) Angle pin header 50 pins (for I/O port)
Physical dimensions	(L x W x H): 125 x 145 x 20 mm

Note 1 By switch of ROM/RAM connector.
2 0.5K bytes are used for screen data and 5.5K bytes can be used for data.

PIN CONFIGURATIONS

Connector J1

Function	Signal	Pin	Pin	Signal	Function
	NC	2	1	VBB	(−5 V)
	GND	4	3	GND	
	GND	6	5	GND	
(5V)	Vcc	8	7	Vcc	(5 V)
DATA BUS	DB1H ↔	10	9	↔ DB0H	(LSB) DATA BUS
DATA BUS	DB3H ↔	12	11	↔ DB2H	DATA BUS
DATA BUS	DB5H ↔	14	13	↔ DB4H	DATA BUS
(MSB) DATA BUS	DB7H ↔	16	15	↔ DB6H	DATA BUS
READ CONTROL OUTPUT	RDCL ←	18	17	→ STOH	STATUS OUTPUT
WRITE CONTROL OUTPUT	WRCL ←	20	19	→ ST1H	STATUS OUTPUT
INTERRUPT ACKNOWLEDGE OUTPUT	ITAL ←	22	21	→ IOMH	DATA TRANSMISSION CONTROL OUTPUT
ROM EXPANSION OUTPUT	ROMSEL 1 ←	24	23	← INTL	INTERRUPT REQUEST INPUT
READY INPUT	RDYH →	26	25	← RSIL	RESET INPUT
ADDRESS LATCH ENABLE OUTPUT	ALEL ←	28	27	→ RSOH	RESET OUTPUT
HOLD RECEIVE INPUT	HLAH ←	30	29	→ CLKH	CLOCK OUTPUT
HOLD INPUT	HLDL →	32	31	→ ROMSEL 2	ROM EXPANSION OUTPUT
RESTART INTERRUPT INPUT	R 75L →	34	33	← TRPL	TRAP INTERRUPT INPUT
ADDRESS BUS	AB1H ←	36	35	→ AB0H	(LSB) ADDRESS BUS
ADDRESS BUS	AB3H ←	38	37	→ AB2H	ADDRESS BUS
ADDRESS BUS	AB5H ←	40	39	→ AB4H	ADDRESS BUS
ADDRESS BUS	AB7H ←	42	41	→ AB6H	ADDRESS BUS
ADDRESS BUS	AB9H ←	44	43	→ AB8H	ADDRESS BUS
ADDRESS BUS	ABBH ←	46	45	→ ABAH	ADDRESS BUS
ADDRESS BUS	ABDH ←	48	47	→ ABCH	ADDRESS BUS
(MSB) ADDRESS BUS	ABFH ←	50	49	→ ABEH	ADDRESS BUS

NC: NO CONNECTION

Connector J2

Function	Signal	Pin	Pin	Signal	Function
SOUND INPUT	SOUND IN →	2	1	Vcc	(5 V)
	GND	4	3	GND	
SERIAL DATA OUTPUT	SODL ←	6	5	← SIDL	SERIAL DATA INPUT
RESTART INTERRUPT INPUT	R55L →	8	7	← R65L	RESTART INTERRUPT INPUT
VERTICAL SYNCHRONOUS SIGNAL OUTPUT	FSL ←	10	9	→ HSL	HORIZONTAL SYNCHRONOUS SIGNAL OUTPUT
I/O PORT A	PA1H ↔	12	11	↔ PA0H	I/O PORT A
I/O PORT A	PA3H ↔	14	13	↔ PA2H	I/O PORT A
I/O PORT A	PA5H ↔	16	15	↔ PA4H	I/O PORT A
I/O PORT A	PA7H ↔	18	17	↔ PA6H	I/O PORT A
I/O PORT B	PB1H ↔	20	19	↔ PB0H	I/O PORT B
I/O PORT B	PB3H ↔	22	21	↔ PB2H	I/O PORT B
I/O PORT B	PB5H ↔	24	23	↔ PB4H	I/O PORT B
I/O PORT B	PB7H ↔	26	25	↔ PB6H	I/O PORT B
I/O PORT C	PC1H ↔	28	27	↔ PC0H	I/O PORT C
I/O PORT C	PC3H ↔	30	29	↔ PC2H	I/O PORT C
I/O PORT C	PC5H ↔	32	31	↔ PC4H	I/O PORT C
	NC	34	33	NC	
	NC	36	35	NC	
	NC	38	37	NC	
	NC	40	39	NC	
	NC	42	41	NC	
ROW PRESET OUTPUT	RPL ←	44	43	NC	
TIMER OUTPUT	TO ←	46	45	← TI	TIMER INPUT
RAM EXPANSION OUTPUT	RAMSEL 2 ←	48	47	→ RAMSEL 1	RAM EXPANSION OUTPUT
	NC	50	49	NC	

NC: NO CONNECTION

Connector J3

Pin	Signal	Function
1	VBB (−5V)	POWER SUPPLY
2	GND (0V)	POWER SUPPLY
3	Vcc (5V)	POWER SUPPLY
4	NC	

NC: NO CONNECTION

INTERRELATION BETWEEN EACH MODE AND THE SCREEN

$\overline{A}$/S (D_7)	INV (D_6)	$\overline{A}$/G (D_5)	$\overline{I}$/E (D_4)	CSS (D_3)	GM2 (D_2)	GM1 (D_1)	GM0 (D_0)	Color (Note 5) Character	Color (Note 5) Back-ground	Color (Note 5) Border	Display	Data	Mode	Memory Capacity (Bytes)
0	0	0	0	0	X	X	X	Green	Black	Black	5 x 7 Dot matrix 1 char.	$D_7D_6D_5D_4D_3D_2D_1D_0$ (Note 4) (Note 3)	Alpha numeric 32 characters x 16 lines	0.5K
	1							Black	Green					
0	0	0	0	1	X	X	X	Orange	Black	Black				
	1							Black	Orange					
1	X	0	0	X	X	X	X	8 Color ①		Black	D_3 D_2 / D_1 D_0	(Note 4) Color — Luminance	Semigraphic 4 64 x 32 picture elements	0.5K
X	X	0	1	0	X	X	X	4 Color ②		Black	D_5 D_4 / D_3 D_2 / D_1 D_0	Color Luminance	Semigraphic 6 64 x 48 picture elements	0.5K
X	X	0	1	1	X	X	X	4 Color ③						
X	X	1	X	0	0	0	0	4 Color ④		Green	E_3~E_0	Color E_3 E_2 E_1 E_0	64×64 Color graphic	1K
X	X	1	X	1	0	0	0	4 Color ⑤		Dark gray				
X	X	1	X	0	0	0	1	2 Color ⑥		Green	D_7~D_0	Luminance	128×64 Graphic	1K
X	X	1	X	1	0	0	1	2 Color ⑦		Dark gray				
X	X	1	X	0	0	1	0	4 Color ④		Green	E_3~E_0	Color E_3 E_2 E_1 E_0	128×64 Color graphic	2K
X	X	1	X	1	0	1	0	4 Color ⑤		Dark gray				
X	X	1	X	0	0	1	1	2 Color ⑥		Green	D_7~D_0	Luminance	128×96 Graphic	2K
X	X	1	X	1	0	1	1	2 Color ⑦		Dark gray				
X	X	1	X	0	1	0	0	4 Color ④		Green	E_3~E_0	Color E_3 E_2 E_1 E_0	128×96 Color graphic	3K
X	X	1	X	1	1	0	0	4 Color ⑤		Dark gray				
X	X	1	X	0	1	0	1	2 Color ⑥		Green	D_7~D_0	Luminance	128×192 Graphic	3K
X	X	1	X	1	1	0	1	2 Color ⑦		Dark gray				
X	X	1	X	0	1	1	0	4 Color ④		Green	E_3~E_0	Color E_3 E_2 E_1 E_0	128×192 Color graphic	6K
X	X	1	X	1	1	1	0	4 Color ⑤		Dark gray				
X	X	1	X	0	1	1	1	2 Color ⑥		Green	D_7~D_0	Luminance	256×192 Graphic	6K
X	X	1	X	1	1	1	1	2 Color ⑦		Dark gray				

Note 3 INV (reverse of character) is determined by D_6 of data (when D_6 1→ INV 1, 0→ INV 0)

4 When I/E (D_4) = 0, $\overline{A}$/S is determined by D_7 of data (1 = Semigraphics 4 mode, 0 = Character mode)

5 Details regarding color are on the next page.

MELCS 85/2 COLOR TV DISPLAY SINGLE-BOARD COMPUTER

COLOR DETAILS

	Color data				
	D_6	D_5	D_4	$D_3 \sim D_0$	
8 Colors ①	X	X	X	0	Black
	0	0	0	1	Green
	0	0	1	1	Yellow
	0	1	0	1	Blue
	0	1	1	1	Red
	1	0	0	1	Dark gray
	1	0	1	1	Cyan
	1	1	0	1	Magenta
	1	1	1	1	Orange
	CSS	D_7	D_6	$D_5 \sim D_0$	
4 Colors ②	0	X	X	0	Black
	0	0	0	1	Green
	0	0	1	1	Yellow
	0	1	0	1	Blue
	0	1	1	1	Red
4 Colors ③	1	X	X	0	Black
	1	0	0	1	Dark gray
	1	0	1	1	Cyan
	1	1	0	1	Magenta
	1	1	1	1	Orange
	CSS	D_7	D_6	(D_5D_4, D_3D_2, D_1D_0)	
4 Colors ④	0	0	0		Green
	0	0	1		Yellow
	0	1	0		Blue
	0	1	1		Red
4 Colors ⑤	1	0	0		Dark gray
	1	0	1		Cyan
	1	1	0		Magenta
	1	1	1		Orange
2 Colors ⑥	CSS	D_7	$(D_6 \sim D_0)$		
	0	0			Black
	0	1			Green
2 Colors ⑦	1	0			Black
	1	1			Dark gray

MEMORY ADDRESS MAP

Address	Contents	Notes
0000_{16} – $07FF_{16}$	EPROM IC2	PROGRAM AREA 2K BYTES
0800_{16} – $0FFF_{16}$	EPROM IC3 (OR RAM)	PROGRAM OR DATA AREA 2K BYTES
1000_{16} – $3FFF_{16}$	OPEN	ROMSEL 1($1000 \sim 17FF_{16}$) ROMSEL 2($1800 \sim 1FFF_{16}$)
4000_{16} – $40FF_{16}$	RAM IC7	DATA AREA 256 BYTES
4100_{16} – 4105_{16}	PPI IC7	PPICW (4100_{16}) PPIA (4101_{16}) PPIB (4102_{16}) PPIC (4103_{16}) PPITL (4104_{16}) PPITH (4105_{16})
4106_{16} – $47FF_{16}$	RESERVED	
4800_{16} – $4FFF_{16}$	VDG MODE SET	
5000_{16} – $5FFF_{16}$	OPEN	RAMSEL 1($5000 \sim 57FF_{16}$) RAMSEL 2($5800 \sim 5FFF_{16}$)
6000_{16} – $77FF_{16}$	VDG FIELD RAM (6K)	
7800_{16} – $7FFF_{16}$	OPEN	
8000_{16} – $FFFF_{16}$	OPEN	

MEMORY CAPACITY AND I/O EXPANSION

The capacity of the PCA8540 can be easily expanded by the addition of other boards such as the PCA8506 or PCA8507.

PCA8506 (ROM, RAM and Parallel I/O Extension)

- Memory capacity .... 12K bytes (Note 6)
- Programmable ports 48 bits .... 8 bits x 6
- Programmable timers .... 16 bits x 3
- Small size, dimensions (LxWxH) .... 145x125x17 mm

PCA8507 (ROM, RAM and Serial I/O Extension)

- Memory capacity .... 12K bytes (Note 6)
- Serial port (RS-232-C or TTL Level) .... 1 port
- Programmable timers .... 16 bits x 3
- Small size, dimensions (LxWxH) .... 145x125x17 mm

Note 6: The memory can easily be expanded in units of 2K bytes up to 12K bytes using any combination of M5L2716K EPROMs and M58725P static RAMs.

MELCS 85/2 COLOR TV DISPLAY SINGLE-BOARD COMPUTER

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	0~7	V
V_{BB}	Supply voltage		0.3~−6.5	V
V_I	Input voltage		5.5	V
Topr	Operational free-air ambient tempera ture range		5~40	°C
Tstg	Storage temperature range		−10~70	°C

RECOMMENDED OPERATING CONDITIONS (Ta=5~40°C, unless otherwise noted.)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{BB}	Supply voltage	−5.25	−5	−4.75	V
V_{IH}	High-level input voltage	2			V
V_{IL}	Low-level input voltage			0.8	V

ELECTRICAL CHARACTERISTICS (V_{CC}=5V+5%, V_{BB}=−5V+5%, Ta=25°C, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage, PA0H~PC5H	$I_{OH}=-50\mu A$	2.4			V
V_{OH}	High-level output voltage, AB0H~AB7H	$I_{OH}=-900\mu A$	3.65			V
V_{OH}	High-level output voltage, AB8H~ABFH	$I_{OH}=-300\mu A$	2.4			V
V_{OH}	High-level output voltage, RS0H, CLKH, HLAH, ALEL	$I_{OH}=-300\mu A$	2.4			V
V_{OH}	High-level output, other outputs	$I_{OH}=-400\mu A$	2.4			V
V_{OL}	Low-level output voltage, PA0H~PC5H	$I_{OL}=1.8mA$			0.4	V
V_{OL}	Low-level output voltage, AB0H~AB7H	$I_{OL}=16mA$			0.5	V
V_{OL}	Low-level output voltage, AB8H~ABFH	$I_{OL}=1.9mA$			0.45	V
V_{OL}	Low-level output voltage, CLKH, ALEL	$I_{OL}=8mA$			0.4	V
V_{OL}	Low-level output, other outputs	$I_{OL}=1.9mA$			0.4	V
I_{CC}	Supply current from V_{CC}	When 2 EPROMs are loaded.		0.6	1.3	A
I_{BB}	Supply current from V_{BB}	When 2 EPROMs are loaded.		0.05	0.2	A
f_{CLK}	CPU clock frequency			1.79		MHz
f_{CH1}	RF output frequency 1			91.25		MHz
f_{CH2}	RF output frequency 2			97.25		MHz
f_{SUB}	Color sub-carrier frequency			3.579545		MHz

MELCS 70/2 SPEECH SYNTHESIZER SINGLE-BOARD COMPUTER

DESCRIPTION

The PCA7002 consists of a controller (single-chip microcomputer) and speech synthesizing LSI housed on a compact 125 x 145mm printed circuit board, capable of generating speech output. Using the PARCOR method, previously analyzed data is stored in EPROM memory and used as the voice data basis for speech synthesis.

Two versions of the single-board computer are available, the PCA7002G02 consisting of a controller and EPROM, and the PCA7002G01 consisting of the basic board alone without a controller and EPROM.

Type	Configuration
PCA7002G01	Board only. No instruction manual supplied. (Note 1)
PCA7002G02	The basic board plus a speech sample stored in a single M5L2732K and the standard program stored in a single M5L8049-005P. In addition, an instruction manual is provided. (Note 2)

Note 1. The PCA7002G01 controller and speech data memory section consists of IC sockets only. However, the synthesizer and other speech output circuits are provided.

2. The PCA7002G02 is provided with the following sample voice in Japanese (female speaker).
   (1) This is the Mitsubishi Electric Corporation.
   (2) This voice has been synthesized with the use of the Single-chip synthesizer.
   (3) Welcome.
   (4) May I help you?
   (5) Please wait a moment.

FEATURES

- Single-board computer capable of PARCOR system speech generating using LSI devices.
- Speech data memory .... 16K bytes (max)
- Speech recording time .... 100s (max)
- Speech output power .... 5.5W (max)
- Small package .... 125(W)x145(L)x30(H) mm

APPLICATIONS

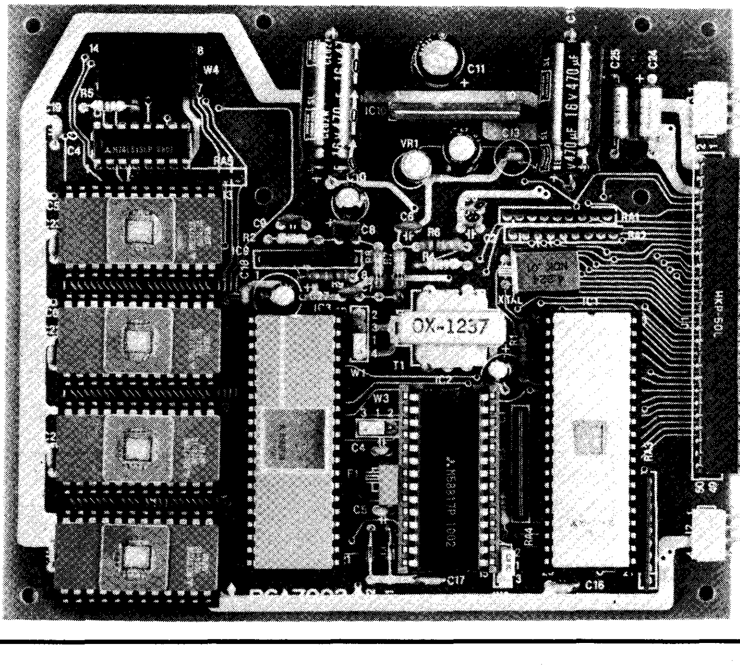
Clocks, control equipment, vending machines, bus annunciators, copying machines, alarm devices.

FUNCTION

The PCA7002 is a speech generating and output singleboard computer consisting of a controller, speech synthesizer LSI, speech data memory, filter, and amplifier circuits. An M5L8048-XXXP 8-bit single-chip controller can be used, and while an M5L8049-XXXP or M5L8748S may be used, an M5L8049-005P which contains the standard program has been provided.

An M5L2716K or M5L2732K may be used as a speech data memory and four M5L2732K devices will allow 60 to 100 seconds of speech output.

In addition, a filter and amplifier circuits are provided on the printed circuit board as well, enabling a maximum speech output power of 5.5W.



12

MELCS 70/2 SPEECH SYNTHESIZER SINGLE-BOARD COMPUTER

FUNCTIONAL DESCRIPTION

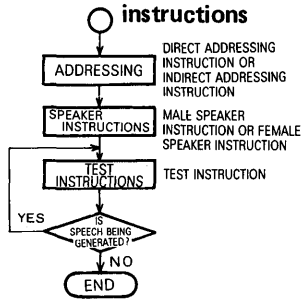
The M58817AP may be controlled by using the following eight instructions which are set up using four data bus lines ($DQ_0 \sim DQ_3$) and one strobe line (ISYNC).

1. Addressing instruction: Used to set phrase ROM addresses
2. Indirect addressing instruction: Used to indirectly set phrase ROM addresses
3. Bit read instruction: Used to shift into a 4-bit shift buffer 1-bit of contents from the phrase ROM
4. Data transmission instruction: Output the contents of the 4-bit shift buffer
5. Test instruction: Test whether speech generation has been completed
6. Male speaker instruction: Start instruction for voice generation (male)
7. Female speaker instruction: Start instruction for voice generation (female)
8. Stop instruction: Halts the generation of speech

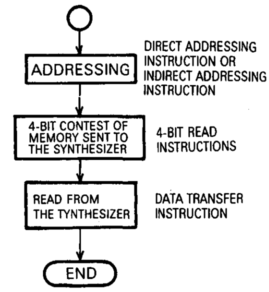
In accordance with the above instructions, the controller reads speech generation instructions for the synthesizer and data from the speech data memory.

The diagram below illustrates the flow of speech synthesis instructions and data.

Speech synthesis instructions



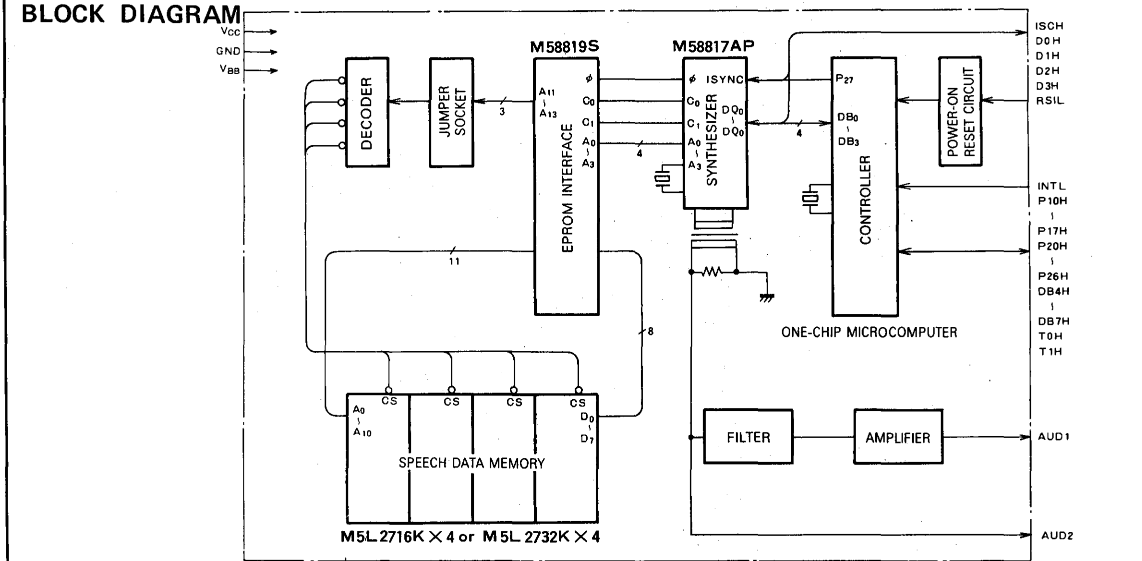
Dater read



OPERATION OF BASIC BLOCKS

Name	Function
Controller	Provides speech processing in accordance with the program
Power-on reset	Generates a controller reset signal upon power-on
Synthesizer	Performs data transfer and speech generation under the control of the controller
EPROM interface	Reads speech data from the speech data memory and sends it to the synthesizer
Jumper socket	Set to specify M5L2716K or M5L2732K devices
Decoder	Outputs the speech data memory selection signal based on the address signals output by the EPROM interface
Speech data memory	Memory used to store speech data
Filter	Eliminates high-frequency noise components from the speech output of the synthesizer
Amplifier	Provides power amplification to an output of 5.5W, maximum, of the speech signal from the filter

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

MELCS 70/2 SPEECH SYNTHESIZER SINGLE-BOARD COMPUTER

SPECIFICATIONS

Parameter	Specification
Control method	8-bit parallel processing
Control LSI	M5L8048-XXXP, M5L8049-XXXP, and M5L8748K (the PCA7002G02 includes an M5L8049-005P containing the standard program)
Cycle time	Basic cycle 4.19μs (3.579545MHz crystal)
Speech synthesis method	PARCOR
Speech synthesis LSI	M58817AP
Speech data memory	16K-byte, maximum (M5L2732Kx4) Addressable 00000_{16}~$03FFF_{16}$ 8K-byte, maximum (M5L2716Kx4) Addressable 00000~$01FFF_{16}$
Data rate	1.96K bit/s (maximum) 3.92K bit/s (maximum) (Selectable by means of on-board jumpers)
Speech synthesis output time	60 seconds (minimum) using 1.96K bit/s transfer rate (Memory capacity/1960) 30 seconds (minimum) using 3.92K bit/s transfer rate (Memory capacity/3920)
Speech output power	5.5W (4Ω speaker) 2.75W (8Ω speaker)
Interrupt	1 factor, 1 level
Power supply	+5V ±5%, −5V ±5% When using an M5L8049-005P controller and four M5L2732K data memories: 800mA (maximum) @+5V 270mA (maximum) @−5V
Connectors	50-pin angled-pin header (I/O port) 4-pin angled-pin header (power supply) 4-pin angled-pin header (speech output)
Dimensions	125(W) x 145(L) x 30(H)mm

PIN CONFIGURATIONS

Connector J1

V_{CC}		2	1		V_{CC}
GND		4	3		GND
V_{BB}		6	5		V_{BB}
ISCH	→	8	7	↔	D0H
BSYH	←	10	9	↔	D1H
SCTH	→	12	11	↔	D2H
CLKH	←	14	13	↔	D3H
N.U.		16	15	←	RSIL
INTL	→	18	17		N.U.
SRST	→	20	19		N.U.
(Note 1) N.U.		22	21	←	IRST
P11H	↔	24	23	↔	P10H
P13H	↔	26	25	↔	P12H
P15H	↔	28	27	↔	P14H
P17H	↔	30	29	↔	P16H
P21H	↔	32	31	↔	P20H
P23H	↔	34	33	↔	P22H
P25H	↔	36	35	↔	P24H
DB4H	↔	38	37	↔	P26H
DB6H	↔	40	39	↔	DB5H
(Note 2) N.C.		42	41	↔	DB7H
T1H	↔	44	43	↔	T0H
V_{BB}		46	45		V_{BB}
GND		48	47		GND
V_{CC}		50	49		V_{CC}

J1

Note 1. NU: Non-usable
2. NC: No connection
3. Pin numbers conform to silk-secreened numbers on the printed circuit board.

Connector J2

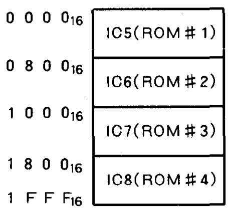
1	V_{CC}
2	GND
3	V_{BB}
4	NC

Connector J3

1	AUD1
2	SIG
3	AUD2
4	SIG

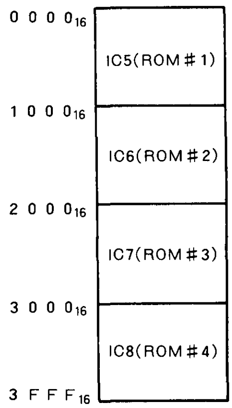
SPEECH DATA MEMORY MAP

Using M5L2716K



(Maximum 8K bytes)

Using M5L2732K



(Maximum 16K bytes)

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to the GND	−0.3~7	V
V_{BB}	Supply voltage	With respect to the GND	V_{CC}+0.3~V_{CC}−15	V
V_I	Input voltage	With respect to the GND	−0.3~7	V
Topr	Operating temperature		0~70	°C
Tstg	Storage temperature		−40~125	°C

RECOMMENDED OPERATING CONDITIONS (T_a=25±5°C, V_{CC}=5V±5%, V_{BB}=−5V±5%, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{BB}	Supply voltage	−4.75	−5	−5.25	V

ELECTRICAL CHARACTERISTICS (T_a=0~55°C, V_{CC}=5V±5%, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Typ	Max	
V_{IH}	RSIL	3.8		V_{CC}	V
V_{IH}	INTL, PIOH~P26H, DB4H~DB7H, TOH, TIH	2		V_{CC}	V
V_{IH}	DOH~D3H, ISCH, SCTH	4.0		V_{CC}	V
V_{IL}	Inputs other than DOH~D3H, ISCH, and SCTH	−0.3		0.8	V
V_{IL}	DOH~D3H, ISCH, SCTH	−0.3		1	V
V_{OH}	DB4H~DB7H (I_{OH}=-100μA)	2.4			V
V_{OH}	PIOH~P26H, TOH, TIH (I_{OH}=-50μA)	2.4			V
V_{OH}	DOH~D3H, BSYH, CLKH (I_{OH}=-0.1mA)	4.0			V
V_{OL}	DB4H~DB7H (I_{OL}=2mA)			0.45	V
V_{OL}	PIOH~P26H, TOH, TIH (I_{OL}=1.6mA)			0.45	V
V_{OL}	DOH~D3H, BSYH, CLKH (I_{OL}=50μA)			0.6	V
I_{CC}	With one M5L8049-005P and four M5L2732K devices mounted		500	800	mA
I_{BB}	With one M5L8049-005P and four M5L2732K devices mounted		200	270	mA

MITSUBISHI ELECTRIC

MICROCOMPUTER SUPPORT SYSTEM

13

MELCS 8/2 PROGRAM CHECKER

DESCRIPTION

The PCA0803 program checker is simple to use, and is suitable for testing the functioning of equipment that employs the PCA0801 single-board computer and the PCA0802 memory and I/O expansion board without requiring any extra software monitor program.

The PCA0803 program checker is useful both in design evaluation and system troubleshooting in field maintenance.

FEATURES

- Single-step function: After halting the CPU at any designated address, allows step-by-step execution of the program instructions in successive single machine cycles.
- Breakpoint function: Halts the CPU at any designated address. Program execution can then be started from this address.
- Memory read/write function: Enables data to be read or written from/to any desired memory location.
- Reset function: Can reset the M5L8080AP CPU.
- Complete with bus cable: A special bus cable, approx. 800 mm long, is provided for connection.
- Supply voltage: 5V ±5%
- Supply current: 0.6A (typ)
- Compact dimensions (L x W x H): 170 x 200 x 27mm

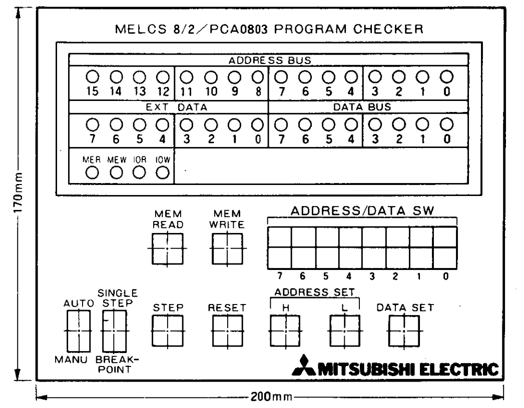
APPLICATIONS

- For design and evaluation of MELCS 8/2 and MELCS 85/2 Board Computer application systems.

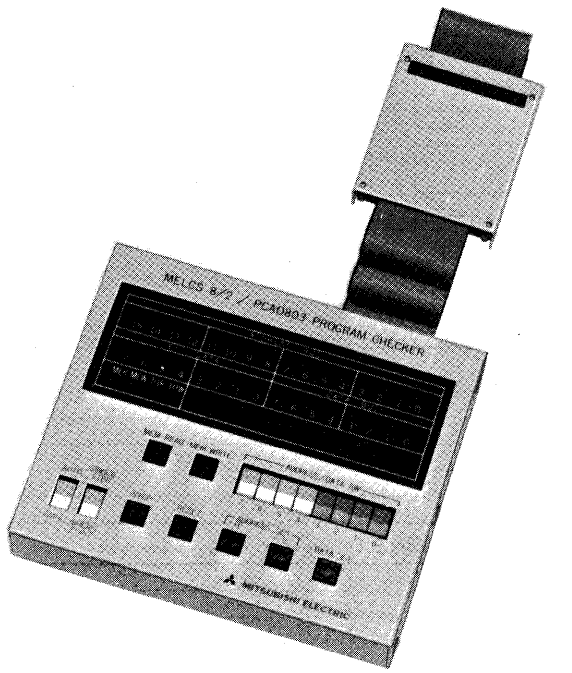
FUNCTION

Software and hardware debugging can be readily achieved by simply connecting the PCA0803 program checker to the equipment tested. Because the PCA0803 is a hardware device, it does not require any extra software monitor programs. The PCA0803 program checker is capable of performing single-step program execution, breakpoint operation, CPU resetting, and memory read/write operations.

APPEARANCE



Mitsubishi PCA0803 program checker



MELCS 8/2 PROGRAM CHECKER

FUNCTION

1. Display Panel

The display panel indicates the operating status of the address bus, data bus and control signals.

2. Address/Data Switches

The ADDRESS/DATA switches are used in setting the address and data for the designated RAM area.

3. H/L Address Set Switch

The H/L ADDRESS SET switch is used in latching the address to the address/data latch circuit. The address is latched to the address/data latch circuit in two operations, the most significant 8 bits and then the least significant 8 bits.

4. Data Set Key

This key is used for data setting.

5. MEM Read/MEM Write Keys

These keys are used in reading or writing data from/to the designated memory location.

6.`Manu/Auto Selection Switch

In the AUTO position, the system executes sequential program instructions. In single-step or breakpoint operation, this switch should be set to the MANU position.

7. Single Step/Breakpoint Selection Switch

In the SINGLE STEP position, depression of the STEP key causes step-by-step execution of the program instructions during successive single machine cycles. When the switch is set to the BREAKPOINT position, the program execution halts at the designated address.

8. Step Key

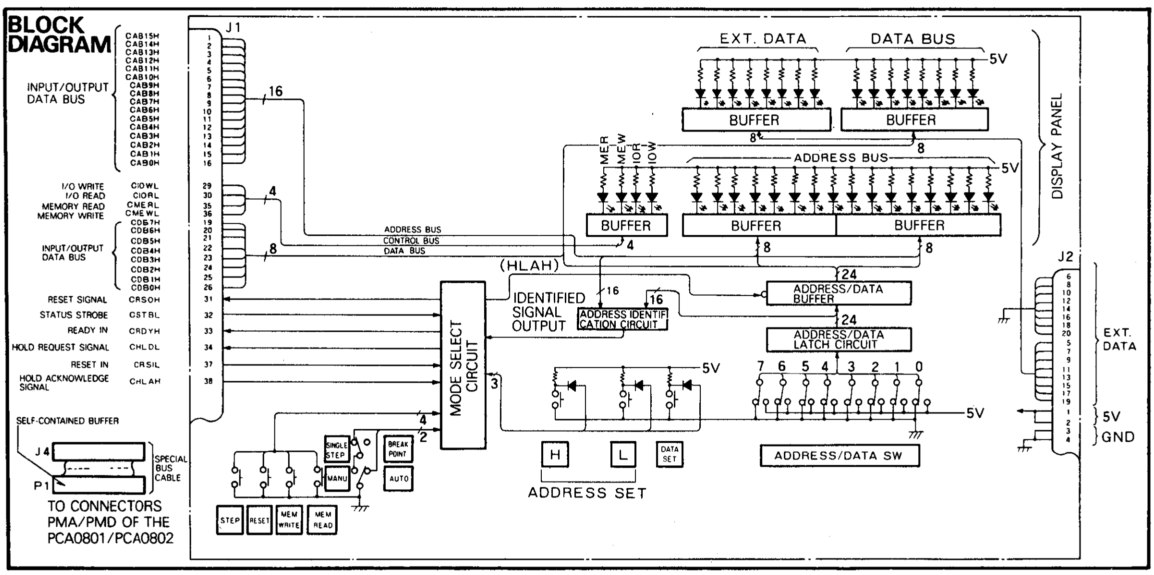
Each time this key is depressed, it executes one program step.

9. Reset Key

This key resets the M5L8080AP CPU. The program counter is cleared to '0', and both the data bus and the address bus are kept in the floating state.

10. Mode! Selection Circuit

This circuit receives various signals from each of the operational switches and sends out selected signals corresponding to the mode assigned.



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage		7	V
V_I	Input voltage		5.5	V
Topr	Operating free-air ambient temperature range		0~55	℃
Tstg	Storage temperature range		−30~70	℃

RECOMMENDED OPERATING CONDITIONS (Ta = 0 ~ 55℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{IH}	High-level input voltage	3		V_{CC}	V
V_{IL}	Low-level input voltage	0		0.65	V

MITSUBISHI ELECTRIC

DESCRIPTION

The PC4000 is a debugging machine for use with single-chip microcomputers. It is intended for use as a general purpose debugging machine for support of single-chip microcomputer hardware and software.

FEATURES

- Usable for RAM-based program debugging
- Connectable to the user system via a DIL socket or connector
- Built-in EPROM (2716, 2732) writer function
- Uses serial data transfer for two-way data transfer with the host machine (e.g. PC9000 cross assembler machine)
- Usable with a variety of single-chip microcomputers by simply replacing a single board
- Print out of internal memory contents is possible by means of an external printer
- Easy-to-carry-about in its compact case, provided with an angle stand

APPLICATIONS

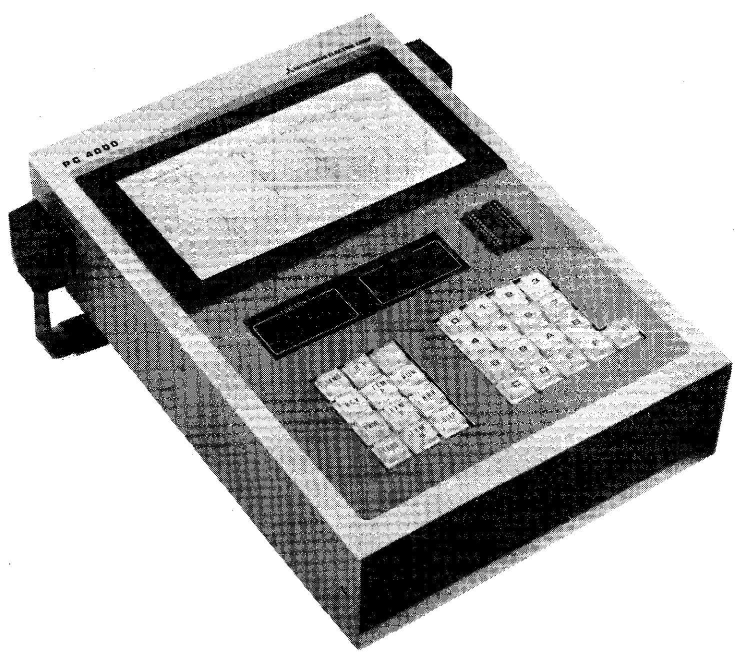
Hardware and software development and program debugging for single-chip microcomputer systems.

CONFIGURATION

The PC4000, as shown in the block diagram, consists of the following hardware elements.

(1) M5L8085AP monitor CPU
(2) Serial data input/output interface circuit
(3) EPROM writer circuit
(4) Program RAM (10 bits x 4K)
(5) Keyboard and LED display circuits
(6) Power supply

The PC4000 is used in conjunction with a dedicated board which allows interface of the PC4000 with the object microcomputer under development. The dedicated board insertion access window is located on the right side of the PC4000. In addition, each dedicated board stores the control program for the monitor CPU. Therefore, when the microcomputer type is changed, the PC4000 can be modified to suit the new type by merely changing the single dedicated board.



DEBUGGING MACHINE

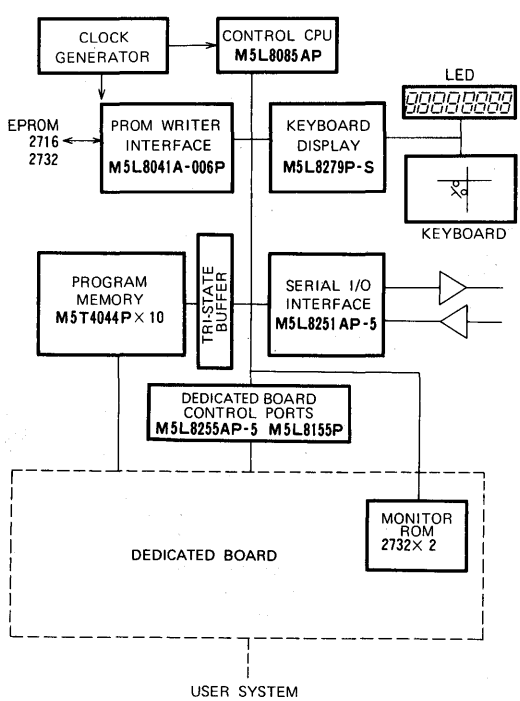
FUNCTIONAL DESCRIPTION

Object programs developed on such devices as the PC9000 cross assembler machine are sent to the PC4000 via the serial input/output interface. The serial data transmission rate can be selected from 1200bps to 9600bps and the interface is a 20mA current loop type. The transmission format is Intel-compatible hexadecimal.

The data in the program memory is executed by the evaluation CPU on the dedicated board. In addition, this memory contents can be written into 2716 or 2732 EPROM devices or data can be read out of such devices via a 24-pin DIL socket.

The keyboard consists of 12 function keys and 16 numerical keys as well as a single entry key. The LED display is an 8-digit display of 7-segment LED elements used to display data for reference while processing is performed.

BLOCK DIAGRAM



KEY FUNCTIONS (BASIC FUNCTIONS ONLY)

Symbol	Name	Function
SEND	Data transmit key	Converts program memory data to serial data and transmits to an external device
RCV	Data receive key	Receives serial data and writes this data into program memory
PROG	(EPROM) Program key	Writes program memory data into the EPROM inserted in the socket
LOAD	(EPROM) Load key	Sends data from the EPROM inserted in the socket to program memory
PRT	Print key	Data transmit to the optional printer
EXM P	Examine program memory key	Verification/correction of program memory contents
EXM R	Examine register key	Verification/correction of register contents
EXM M	Examine memory key	Verification/correction of RAM contents
RES	Reset key	Reset of program counter
RUN	Run (execute) key	Re-start of program execution at the specified address (real time)
BRK	Break point set key	Sets the break point address
STEP	Single step key	Excutes the program one step at a time
0～F	Numerical keys	Used for input of address and data
ENT	Entry key	Effectively enters input numerical data

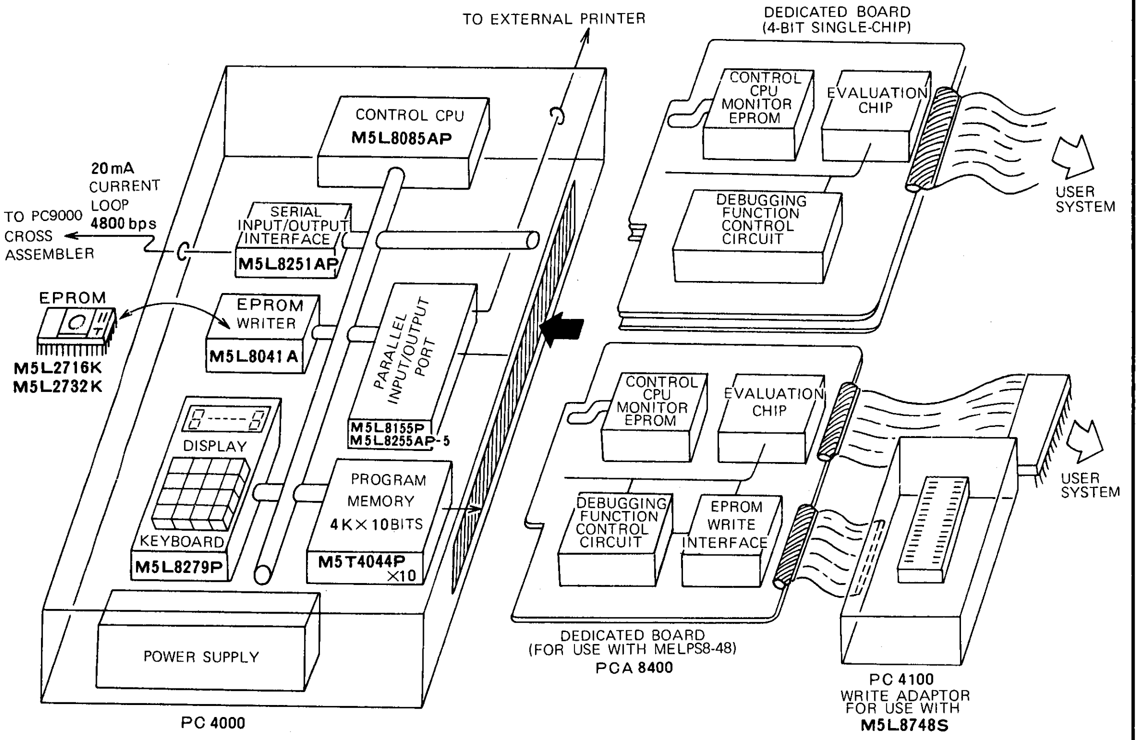
SPECIFICATIONS

Item	Specification
Method	The system is used with a dedicated board which includes the evaluation chip to perform in-circuit emulation
Applicable microcomputers	M58840-XXXP M58494-XXXP M58496-XXXP M5L8048-XXXP M5L8049-XXXP and all other Mitsubishi single-chip microcomputers
Program RAM	Built-in, 4K x 10 bits (250ns access time)
Control CPU	M5L8085AP
Built-in EPROM writer circuit	Usable with 2716 or 2732 devices
Display	7-segment LED, 8 digits
Input	Key switches: Commands: 12 keys Numerical: 16 keys Entry: 1 key
Interface	① 20mA current loop serial input/output interface 4800bps, full deplex, one line (Selectable from 1200 to 9600bps) ② Centronix-compatible parallel interface, one line
Monitor function	Monitor programs for the appropriate object microcomputers are written into the two M5L2732K devices mounted on the dedicated board. Basic Functions • Transfer of RAM data with an external system • Read and write of EPROM data • Verification/correction of the built-in program memory (RAM) contents • Execution and halt at any arbitrary program address • Single-step execution of programs • Verification/correction of internal registers, memory, flags
User system connection	Input/output connections to the dedicated board by means of a cable
Dimensions	364 × 257 × 85mm (excluding handle and key switch tops)
Power supply	AC 100V 100VA
Operating temperature	5～40°C
Storage temperature	−20～60°C

13

DEBUGGING MACHINE

PC4000 CONFIGURATION



SPEECH SYNTHESIS EVALUATION UNIT

DESCRIPTION
The PC7000 is a device intended for use in evaluating speech synthesized from voice data by the PARCOR method under uniform conditions and enables complete evaluation of Mitsubishi speech synthesis products.

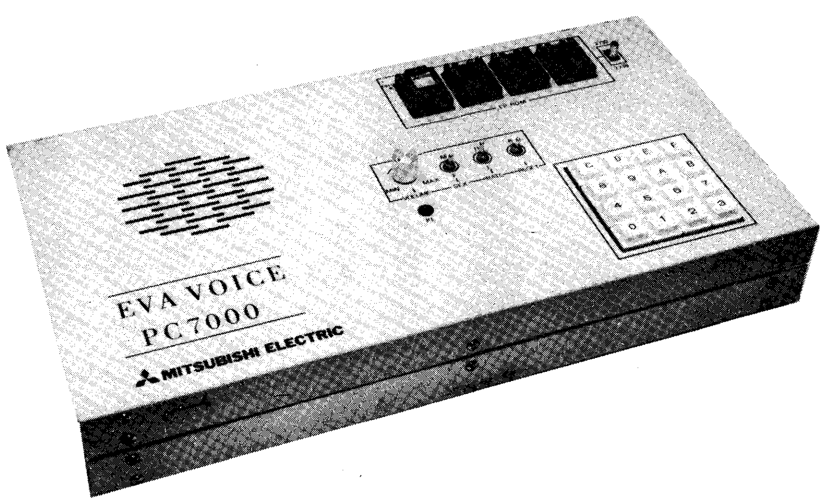
FEATURES
- Enables speech synthesis in response to keyed-in information
- Allows free selection of the number of repetitions of speech
- Enables up to a maximum of seven speeches to be continuously generated in response to one keyed input
- Spacing between words is freely settable
- Up to four words may be generated continuously with the programmed spacing
- EPROM data modification may be used to easily change the above programmed sequences
- Built-in power supply and speaker allow immediate use
- A line output is provided for test listening by means of external filters, amplifiers and speakers
- An external switch may be used to easily switch between male and female speakers
- An external switch may be used to select low or high bit rate

FUNCTION
The PC7000 includes a 16-key panel which allows programmed voice sequences to be called up from EPROM memory. In response to a single key up to eight voice instruction steps may be specified, enabling the generation of up to seven words continuously, four words continuously at the specified interval and the generation of repetitions of the entire sequence. The EPROM may be removed by means of a window provided on the PC7000.

Because the PC7000 is provided with a built-in power supply, audio amplifier, and speaker, it can be powered from the commercial power source and used for test listening immediately. In addition, a line output is provided for use with other amplifiers and speakers making connections with other audio equipment simple.

SPECIFICATIONS

Item	Specification
Speech generation section	Low passfilter, highpass filter, main amplifier
Keyboard section	16 input keys (0~F)
Volume control	Controls output volume
Speech output power	1W (Max)
Line output	Output before and after filtering
Speaker	12cm single cone, 8Ω
EPROM	M5L2716K or M5L2732K (the EPROM may be removed from the PC7000 by means of an access window)
External switches	(1) Male/female speaker switch (2) Low-high bit rate switch (3) 2716/2732 EPROM switch

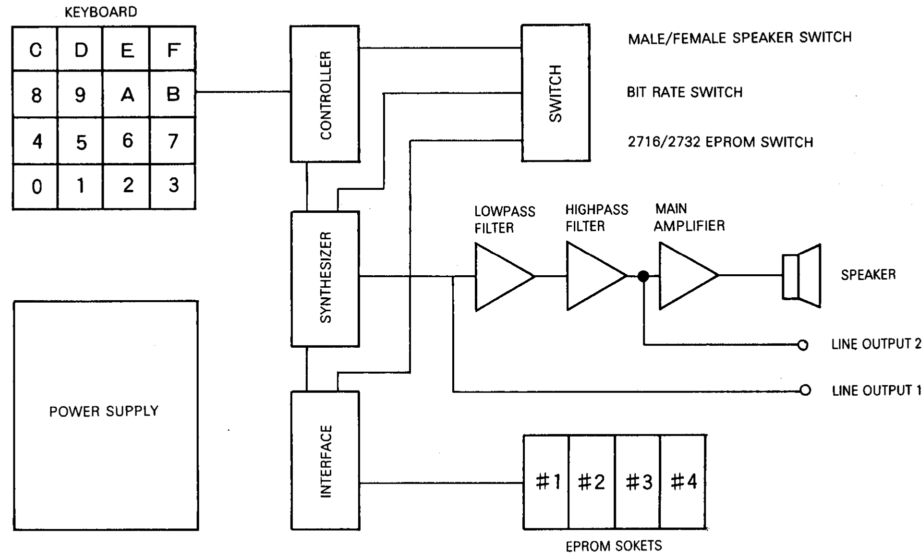


SOFTWARE

The PC7000 has a configuration similar to the PCA7002 single-board computer, and uses the same M5L8049-005P CPU as does the PCA7002. On the PCA7002, corresponding to the speech input to the 18 ports, speech is generated in accordance with speech sequences and speech addresses previously recorded in EPROM, whereas in the PC7000, of the 18 ports 1 through 16 correspond to the keyboard inputs 0 through F. Therefore, it is necessary to store in the PC7000 EPROM addresses 0000_{16} through $007F_{16}$ voice sequences which correspond to the keyboard 0 through F.

For details of the method of writing voice sequence data and starting addresses, refer to the manual for the PCA7002.

BLOCK DIAGRAM

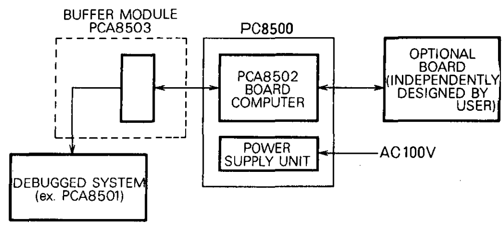


MELCS 85/1 PORTABLE MICROCOMPUTER CONSOLE

DESCRIPTION

The PC8500 portable microcomputer console is a microcomputer system embodying the PCA8502 board computer. Not only it does operate as a general-purpose microcomputer, but it also can be used as a debugging system, in which the M5L8085AP MELPS 85 8-bit microprocessor (identical with Intel's 8085A) is used. The PCA8503 is a buffer module that interfaces the debugged system with the PC8500 through an IC socket of the M5L8085AP, S, when the PC8500 is used as a debugging system.

MELCS 85/1 SYSTEM CONFIGURATION

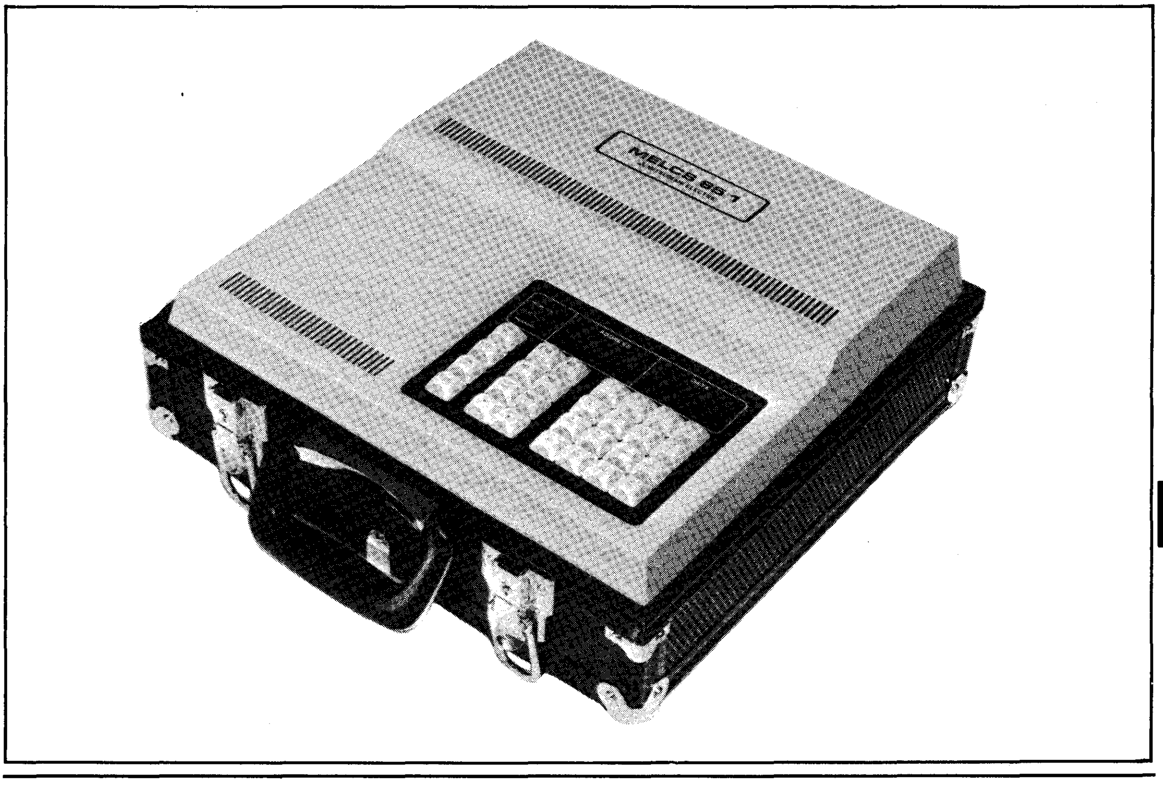


FEATURES

- Can be used as a debugging system in which a microprocessor identical with the M5L8085AP is used.
- Interfacing of the PC8500 with the debugged system through an IC socket of the microprocessor on the debugging system.
- The PCA8503 is provided for the interface.
- Feasible to use the PC8500 as a customized unit by adding an optional board to the general-purpose microcomputer PC8500.
- The 24-key keyboard and the eight 7-segment LED display are furnished as input/output devices.
- Contains a circuit for a system typewriter on the board.
- The PC8500 is housed in a portable carrying case.

APPLICATIONS

- Debugging unit
  Hardware and software development of a system in which an 8-bit microprocessor identical with the M5L8085A is used.
  Testing for board computer.
  Maintenance and inspection systems that use a board computer.
- General-purpose microcomputer
  Application system that is customized by the user (e.g. PROM writer, data logger, board checking system, etc).



MELCS 85/1 PORTABLE MICROCOMPUTER CONSOLE

FUNCTION

The PC8500 is composed of the board computer PCA8502 and the power supply unit, as shown in the block diagram. The functions of the PCA8502 comprise the following hardware functional blocks:

(1) CPU
(2) Program memory
(3) RAM
(4) Keyboard display interface
(5) Parallel I/O interface
(6) Serial I/O interface
(7) Special logical circuit designed for the debugging system

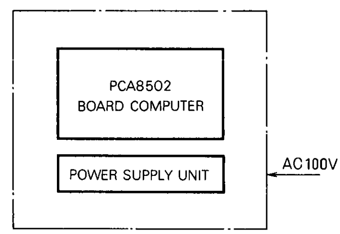
The PCA8502 offers 1K bytes of EPROM and 4K bytes of RAM and also releases the M5L8255AP PPI (8-bit × 3 programmable I/O ports) for a parallel I/O interface.

Program monitoring is provided by a monitor that controls the keyboard and the LED display of the PCA8502 and a monitor that controls the system typewriter

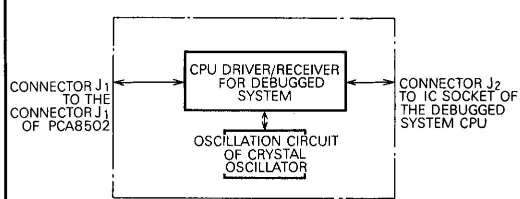
The PCA8503 is a buffer module employed in interfacing the PC8500 (PCA8502) with a user system (debugged board), as shown in the block diagram, and supplied as an optional board.

BLOCK DIAGRAMS

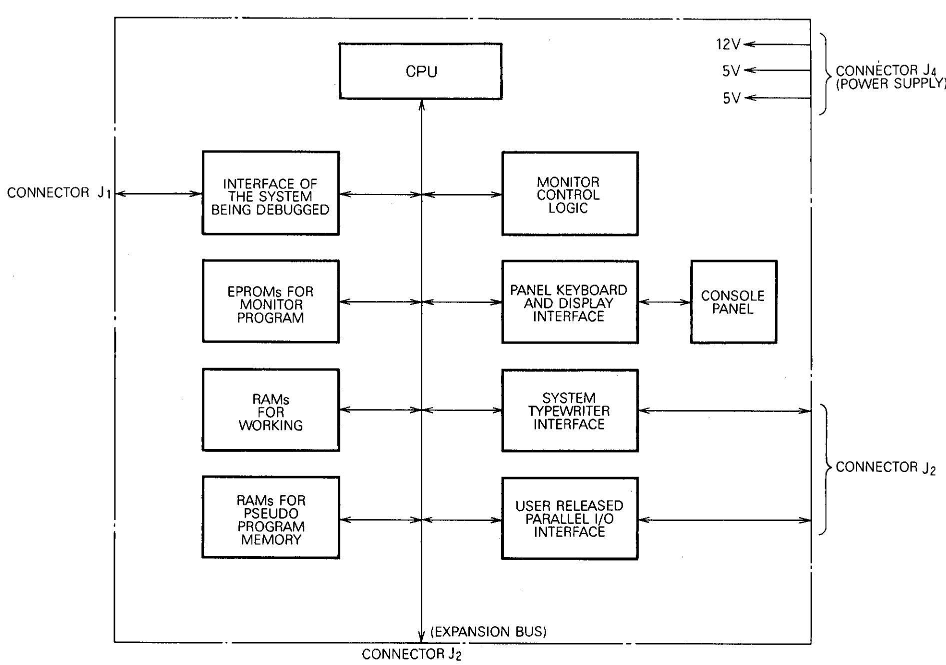
PC8500



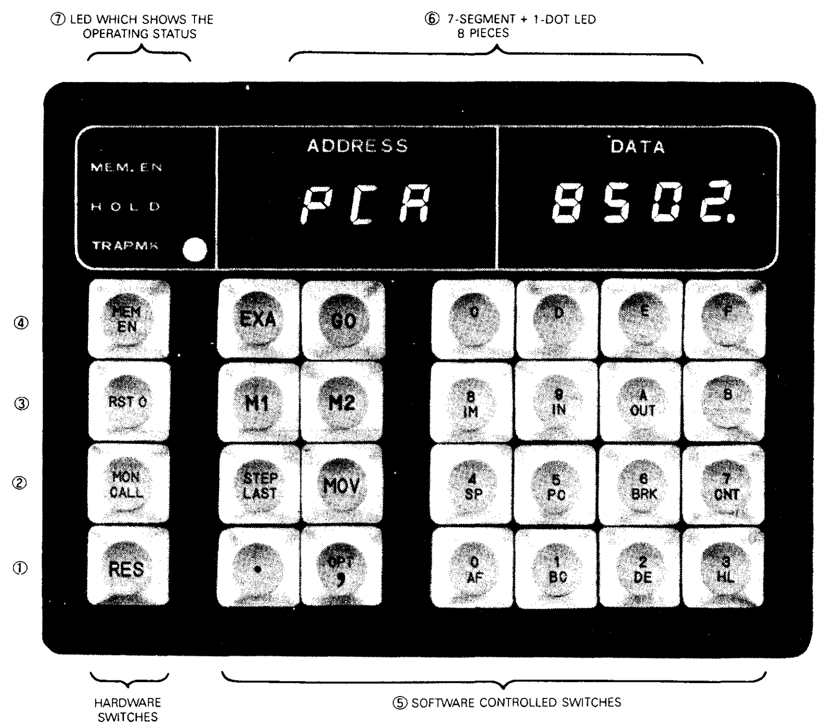
PCA8503



PCA8502 BLOCK DIAGRAM



HOW TO OPERATE THE PANEL



1 RES (RESET)
Resets the I/O controllers of the system, including the CPU, and the CPU enters the WAIT state.

2 MON CALL (MONITOR CALL)
With this switch, the control of the CPU is removed to the monitor area. As this switch was depressed following the depression of the "RES" switch, the CPU enters the monitor command request state after executing the monitor program.

3 RST 0 (RESTART 0)
As this switch was depressed after depressing the "RES" switch, it makes the CPU perform from the address 0_{16}.

4 MEN EN (MEMORY ENABLE)
Depression of this switch enables the pseudo program memory, and the RAM address provided in the system is changed to the area of 0000_{16}~$0FFF_{16}$ superseding the ROM area. While it is disabled, it can be used as an ordinary RAM that will have addresses designated by the mini-switches provided in the system.

5 Software control keyboard
This keyboard consists of 24 2-key rollover scanning keys, and is used for entering commands for the monitor program. It can also serve as a user-specified input device, when a user's program is prepared for it.

6 7-segment LED display
It is composed of 8 pieces of 7-segment LEDs and used as an output device for the monitor. It can also serve as a user-specified output device when a user's program is prepared for it.

7 Status indicating LEDs
The MEN EN indicator LED displays the state of the pseudo program memory; it indicates that the pseudo program is enabled when the LED is on.

The HOLD indicator LED shows that the CPU is in the HOLD state.

The TRAPMK indicator LED lights to show that the TRAP interrupt signal is being masked. It remains lit as long as the monitor program is in execution or the command designating TRAP interrupt is valid.

MELCS 85/1 PORTABLE MICROCOMPUTER CONSOLE

SPECIFICATIONS OF THE PCA8502

	Item	Description
	Method	8-bit parallel processing unit
	CPU	M5L8085AP
	Cycle time	1.3μs basic cycle at crystal oscillator 6.144MHz
Memory	System use area	ROM: $F800_{16} \sim FFFF_{16}$ = 2K bytes for a monitor program ROM: $F400_{16} \sim F7FF_{16}$ = 1K bytes for user released area RAM: $F300_{16} \sim F3FF_{16}$ = 256 bytes for monitor used area Inhibited area: $F000_{16} \sim F2FF_{16}$
Memory	User released area	RAM: $*000_{16} \sim *FFF_{16}$, max 4K bytes. Where * indicates any number from $0_{16} \sim E_{16}$ Can be used as a pseudo program memory.
	I/O interface	Keyboard display interface: $F0_{16}$, $F1_{16}$: interface for panel switch data command indication USART: $F4_{16}$, $F5_{16}$: system typewriter interface data command Parallel port: $F8_{16} \sim FB_{16}$: system control interface I/O address of the area, $F0_{16} \sim FF_{16}$: other than the above are inhibited from use. Programmable I/O port released for user's purpose: $*0_{16}$, $*1_{16}$, $*2_{16}$, $*3_{16}$ Where * indicates any number from $0_{16} \sim E_{16}$.
	Keyboard display	Keyboard: 24 keys, with 2-key rollover scanning method Display: 7-segment LED × 8 pcs
	System typwriter interface	Driver/receiver: 20mA current loop (with source power supply) TTL level (I/O under negative logic) Signal lines: Serial data input, serial data output, and reader start signal lines Applicable transfer speed: 110, 1200, 2400, and 4800 baud Capable of connection with ASR-33, Casio Typuter, etc.
	User released I/O port	8-bit × 3 I/O programmable ports
Functions as a debugging system	Applicable CPU	M5L8085AP (identical with Intel's 8085A)
Functions as a debugging system	CPU clock	Can be operated with clock from the user's system (3.125MHz max)
Functions as a debugging system	Interface with user system	To be connected with the IC socket of the CPU of user's system through the buffer module (PCA8503)
Functions as a debugging system	User's address area	All the address areas except those below, which must be used by the debugging unit, are released to users. Address area: $F000_{16} \sim FFFF_{16}$ I/O address area: $F0_{16} \sim FF_{16}$
Functions as a debugging system	Interrupt	All interrupt signals to the CPU are released for users. As for TRAP interrupt, it is possible to mask it by the monitor command.
Functions as a debugging system	Pseudo program memory	It is possible to substitute the address area $0000_{16} \sim 0FFF_{16}$ of the user's system with the RAM in the debugging system.
Function as a debugging system	System monitor	As a system monitor, there are two types of monitors; the keyboard monitor, which uses the keyboard and the LED display as I/O device, and a TTY monitor, which uses the system typewriter as I/O device. Functions of the monitor are: (1) Verifying the contents of the memory (2) Verifying registers of the CPU (3) Execution of user's program (4) Executing a program after setting breakpoint address (5) Step-by-step execution of program (6) Verifying I/O registers (7) Block transferring of data (8) Setting and resetting interrupt mask (9) Data dump and load to the memory (hexadecimal notation is available in the case of the TTY monitor)
	Optional board	Available for expansion of the CPU bus A single optional board (approx. 140 × 310mm) can be added.
	Connectors	J1 (50 pins): for the interface with the user's system J2 (50 pins): for the parallel I/O port and the system typewriter interface J3 (50 pins): for CPU bus J4 (10 pins): for power supply connection
	Power supply	5V, 2.5A (typ) 12V, 150mA (typ) −5V, 90mA (typ)
	Dimensions	(W × L × H): 310 × 300 × 22mm

SPECIFICATIONS OF THE PCA8503

Item	Description
Function	Interfaces the board computer PCA8502 with a user's system which has a CPU identical to the M5L8085A. Furnished with the driver/receiver and an extension cable.
Connectors	50 pins and 40 pins
Power supply	Supplied from the PCA8502, 5V/350mA (typ)
Cable	Approx. 1m long
Dimensions	(W × L × H): 120 × 100 × 25mm
Operating free-air temperature	0~50℃

SPECIFICATIONS OF THE PC8500

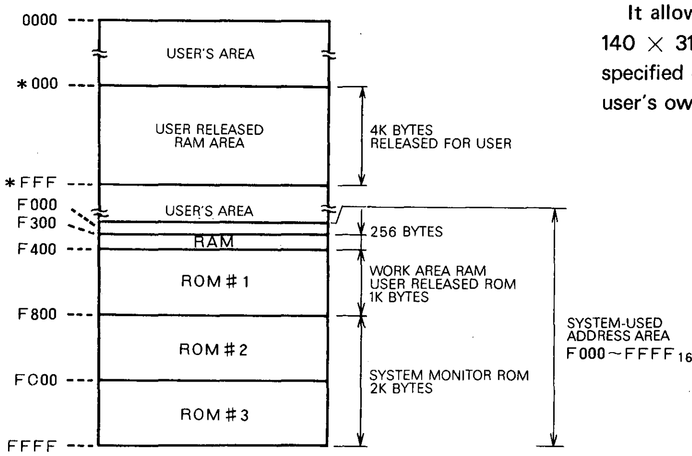
Item	Description
Function	In compliance with function of the PCA8502 implemented
Supply power input	AC100V±10%, 50Hz/60Hz
Internal supply power	5V/5A, 12V/300mA, −5V/300mA Those used in the board are 5V/2.5A, 12V/150mA, −5V/90mA (typ).
Operating free-air temperature	10~40℃
Dimensions (carrying case)	(W × L × H): 370 × 350 × 140mm
Weight	7kg

SYSTEM ADDRESS AREAS

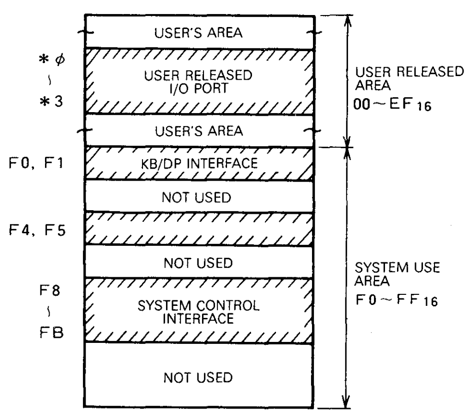
Among the address areas used by the system, the memory addresses 0000_{16}~$EFFF_{16}$ and the I/O device addresses 00_{16}~EF_{16} are all released for the user, and the rest of the areas are used by the system. So the user should stay within the prescribed areas.

Furthermore, the RAM area released for the user may be switched over of its address in the unit of 4K bytes using the mini-switches.

MEMORY ADDRESS AREA



I/O ADDRESS AREA



PSEUDO PROGRAM MEMORY

Among the address areas in the user's system, it is possible to substitute the area 0000_{16}~$OFFF_{16}$ with the RAM within the system.

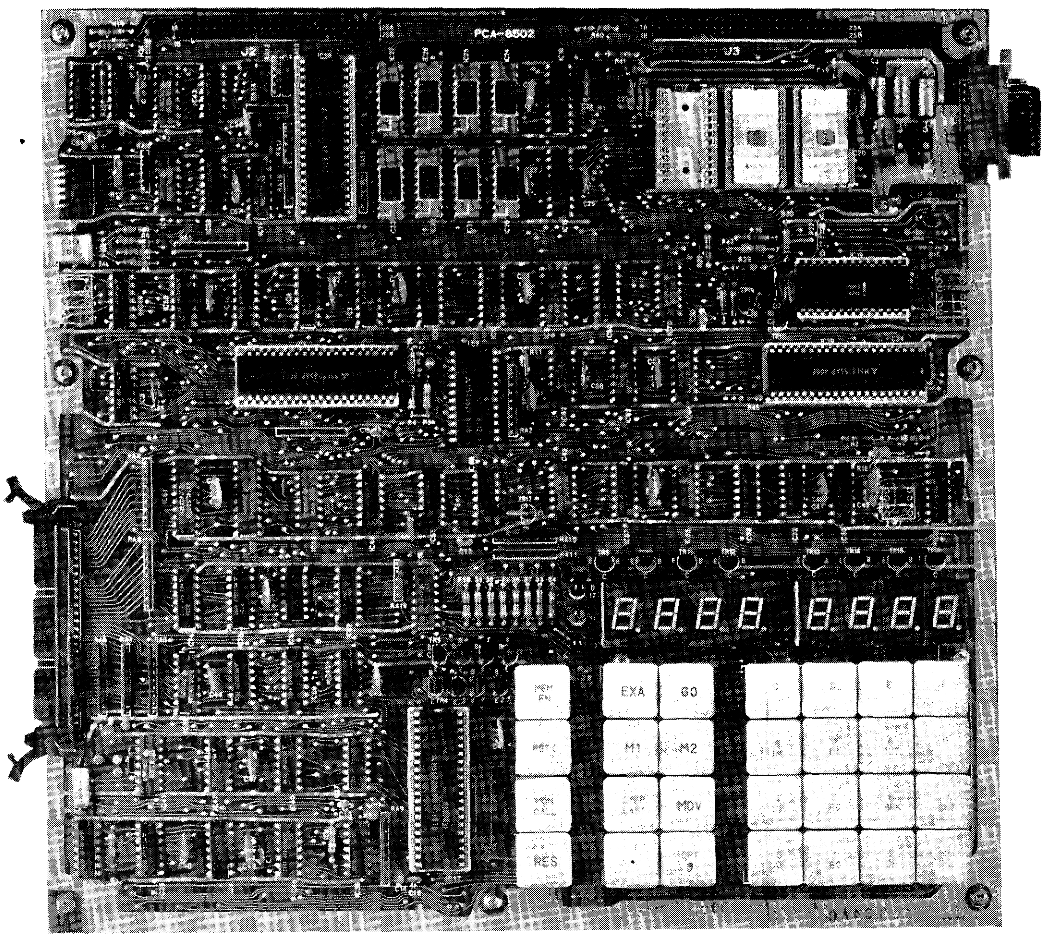
Substitution is enabled by depressing the [MEM EN] key, which allows the RAM to access the area 0000_{16}~$OFFF_{16}$, enabling the execution of a user's program, and altering the contents of the RAM.

OPTIONAL BOARD

It allows expansion of the system as the bus lines of the CPU are extended to the connector J_3.

It allows addition of one extra board whose size is about 140 × 310mm as an optional board with which a user-specified device may be obtained by preparing it with the user's own design.

PCA8502



DESCRIPTION

The PC9000 is a cross assembler machine. It is capable of converting programs for the Mitsubishi single-chip microcomputers written in assembler language to machine language. In addition, it can perform such debugging functions as disassembly and act as an EPROM writer.

FEATURES

- Input of the source program from the keyboard
- An efficient screen editor allows editing of source programs
- Program dump and load to the mini-floppy disk
- Object data write/read for 2708, 2716 and 2732 EPROM devices
- Listing using a Centronix-compatible printer is possible
- Data transmission is possible to the PC4000 debugging machine
- Usable with all types of Mitsubishi single-chip microcomputers
- Compact, desk-top design

APPLICATION

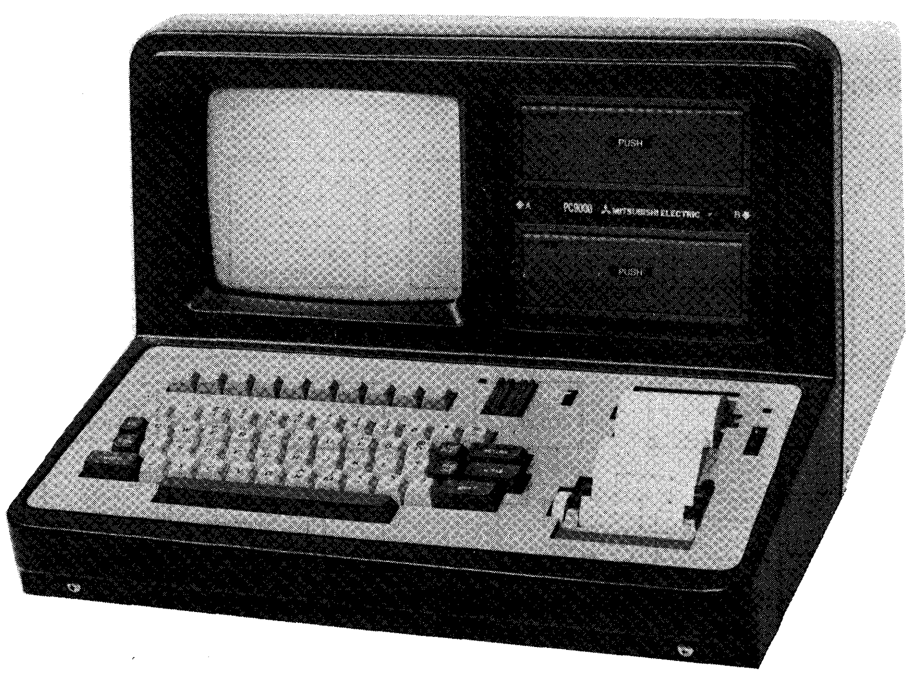
Software development support for Mitsubishi single-chip microcomputers.

FUNCTION

The PC9000 as shown in the configuration diagram consists of the following hardware

(1) Control CPU and bootstrap ROM
(2) 48K byte RAM
(3) 2K byte display screen RAM
(4) 9-inch CRT display circuit
(5) EPROM writer circuit
(6) ASCII keyboard
(7) Hardcopy output by means of an internal mini-printer circuit or an external printer interface circuit
(8) Floppy disk controller (two mini floppy disk drives)
(9) Parallel input/output interface circuit (two lines)
(10) Power supply

An M5L8085AP is used as the control CPU. The keyboard, CRT, mini-floppy disk drives, and printer interfaces are connected by means of a bus line. The keyboard is used for input of commands to the monitor and source program data verification. The 9-inch green CRT display screen is capable of displaying 24 lines of 80 characters. As a printer a 20 column mini-printer is built-in to the PC9000 in addition to the ability to use an 80 column printer having Centronix compatibility via an interface which is available. The built-in mini-printer may be used to output



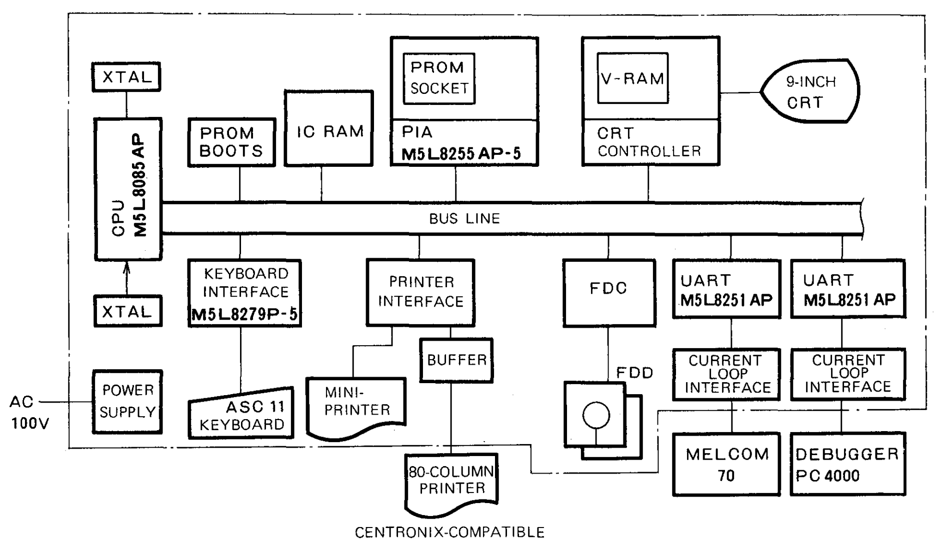
the disassembly results while the external printer may be used to output the assembly listing as well as disassembly listing.

FUNCTIONAL DESCRIPTION

The PC9000 contains the assembler, disassembler, source editor, and EPROM writer functions required for software support of microcomputers. These functions are summarized in the Table.

Function	Effect	Applicable devices
Assemble	Source input: keyboard output: printer, EPROM, data transfer (with debugging unit)	All 4-bit single-chip PMOS, and CMOS microcomputers M5L8048, M5L8049 and M5L8041A 8-bit single-chip microcomputers
Disassemble	Disassembly of the specified file Output: 20 column printer, external printer	Same as above
Source editor	Deletion, insertion, modification, character search, and screen editing	Same as above
PROM writer	EPROM erase check, write, verification, read	M5L2708K, M5L2716K, M5L2732K

PC9000 CONFIGURATION



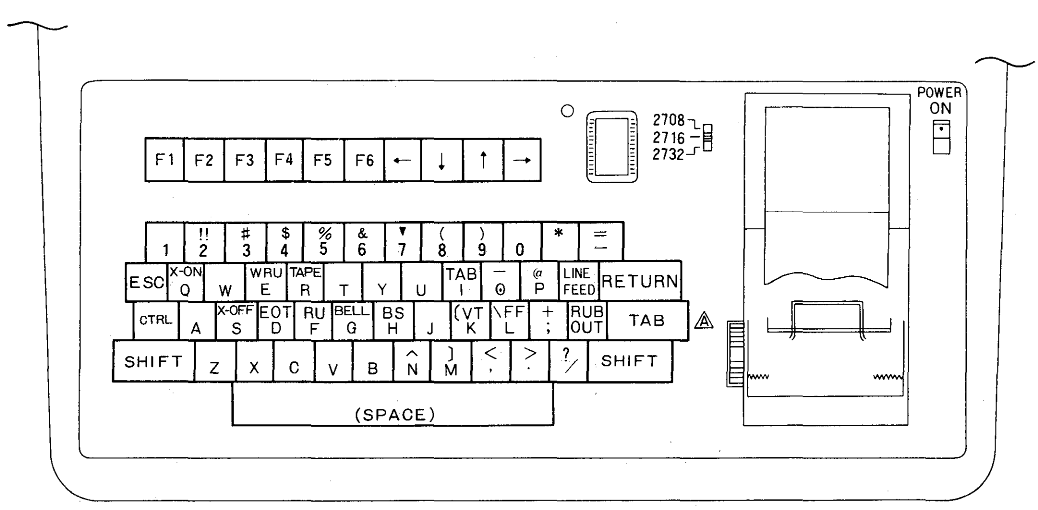
MITSUBISHI ELECTRIC

CROSS ASSEMBLER MACHINE

SPECIFICATIONS

Item	Specification
Structure	Desktop-type, single cabinet
C P U	Mitsubishi M5L8085AP (2.45 MHz clock)
IC memory	2K byte ROM (bootstrap area), 48K-byte DRAM, 2K-byte VRAM
Memory device	Mini floppy disk x 2 drives, double-sided, double-density
Display	9-inch green CRT display, 80 lines x 25 characters
Keyboard	Modified ASCII specifications, 2-key lockout
Dedicated printer	5 x 7 dot Matrix thermal printer, 20 columns. 2 lines/s. Paper width: 60mm.
Printer interface	Centronix, parallel interface Interface connector: 36-pin DDK Amphenol
Serial input/output interface	20mA current loop (2 lines)
Data transfer format	MELPS 85 Hexadecimal (equivalent to Intel Hexadecimal)
Applicable microcomputers	MELPS 8−48 (M5L8048-XXXP, M5L8049-XXXP and others) MELPS 4 (M58840-XXXP and others) MELPS 41 (M58494-XXXP) MELPS 42 (M58496-XXXP and others)
Outer dimensions and weight	Desk top-type 470(W) x 290(H) x 490(D), 17kg
Power supply	AC 100V ± 10% 50/60Hz

KEYBOARD ARRANGEMENT



13

DESCRIPTION

The dedicated MELPS 4 PCA4001 board is for use with the PC4000 debugging machine for the M58840-XXXP and M58841-XXXSP single-chip 4-bit microcomputers and is used by inserting the board in the PC4000 cabinet.

FEATURES

- Connection to user's systems by a flat cable
- Single-step operation and breakpoint operation capability from the PC4000 debugging machine keyboard. Debugging functions such as confirmation of internal register contents
- Can be used with external and internal clocks

APPLICATIONS

The development of hardware, and software for systems using the MELPS 4 (M58840-XXXP, M58841-XXXSP) single-chip microcomputers.

CONFIGURATION

As can be seen in the block diagram, the PCA4001 consists of the following hardware:

(1) Evaluation chip (M58842S) and peripheral circuitry
(2) EPROM with the PC4000 monitor program
(3) Single-step and breakpoint control circuit
(4) Program memory interface circuit
(5) Input/output buffer/latch circuit

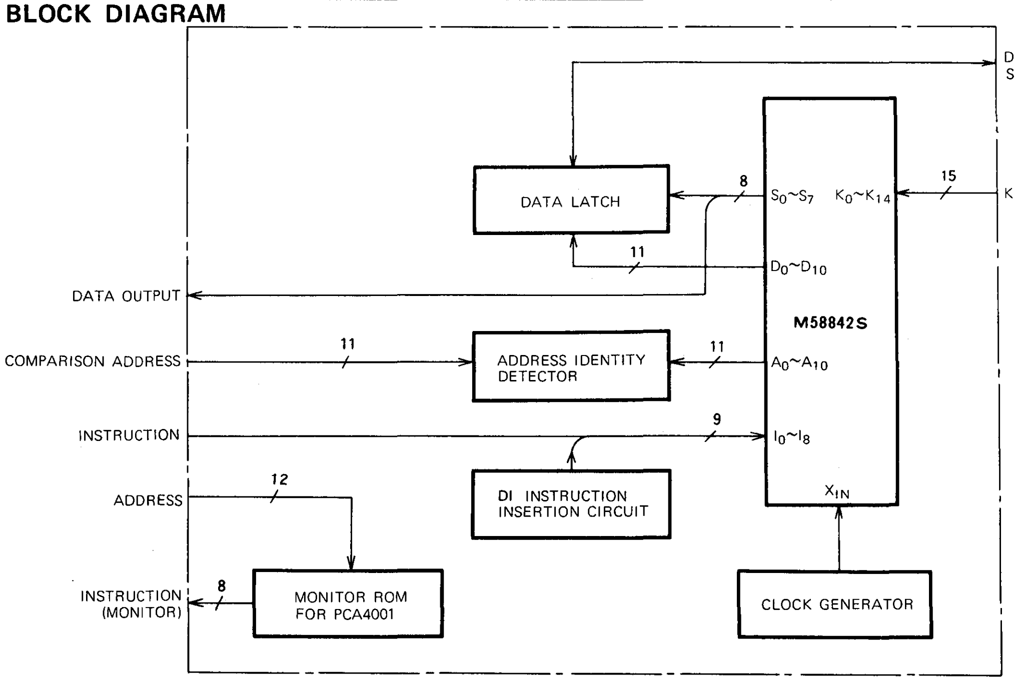
The board and the user's systems can be connected by means of an accessory cable.

FUNCTION

The debugging machine PC4000 operates as a debugging machine for MELPS 4 microcomputers using the monitor program contents of a ROM mounted on the dedicated board. The evaluation chip (M58842S) loaded on the board executes program stored in the program memory in the PC4000 debugging machine.

The internal status of the evaluation chip is read out under monitor CPU control when single-step operation and breakpoint operation are halted.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

SPECIFICATIONS

Item		Specification
Applicable micro-computers		M58840-XXXP, M58841-XXXSP
Clock frequency	Package	600kHz
	Variable range	300～600kHz
Applicable debugging machine		PC4000 (connected by a cartridge connector)
Power supply		Supplied by the PC4000
Connection to user's systems		By an accessory cable
Debugging functions (contents of monitor EPROM)		● Program execution from any address and halt ● Single-step operation ● Data writing to EPROM and reading ● Confirmation and change of the contents of program RAM ● Serial data transfer to an external device ● Confirmation and change of the RAM in the evaluation chip and the contents of the following registers and flags ● Program counter ● Data pointer ● Accumulator ● B register ● H/L registers

DESCRIPTION

The PCA4003 is a dedicated MELPS 4 board for use with the PC4000 debugging machine for the M58843-XXXP 4-bit single-chip microcomputers and is used by inserting the board in the PC4000 cabinet.

FEATURES

- Connection to user's system by a flat cable
- Single-step operation and breakpoint operation capability from the PC4000 debugging machine keyboard. Debugging functions such as confirmation of internal register contents.
- Can be used with external and internal clocks

APPLICATIONS

The development of hardware and software for systems using the MELPS 4 (M58843-XXXP) single-chip microcomputers.

CONFIGURATION

As can be seen in the block diagram, the PCA4003 consists of the following hardware.

(1) Evaluation chip (M58842S) and peripheral circuitry
(2) EPROM with the PC4000 monitor program
(3) Single-step and breakpoint control circuit
(4) Program memory interface circuit
(5) Input/output buffer/latch circuit

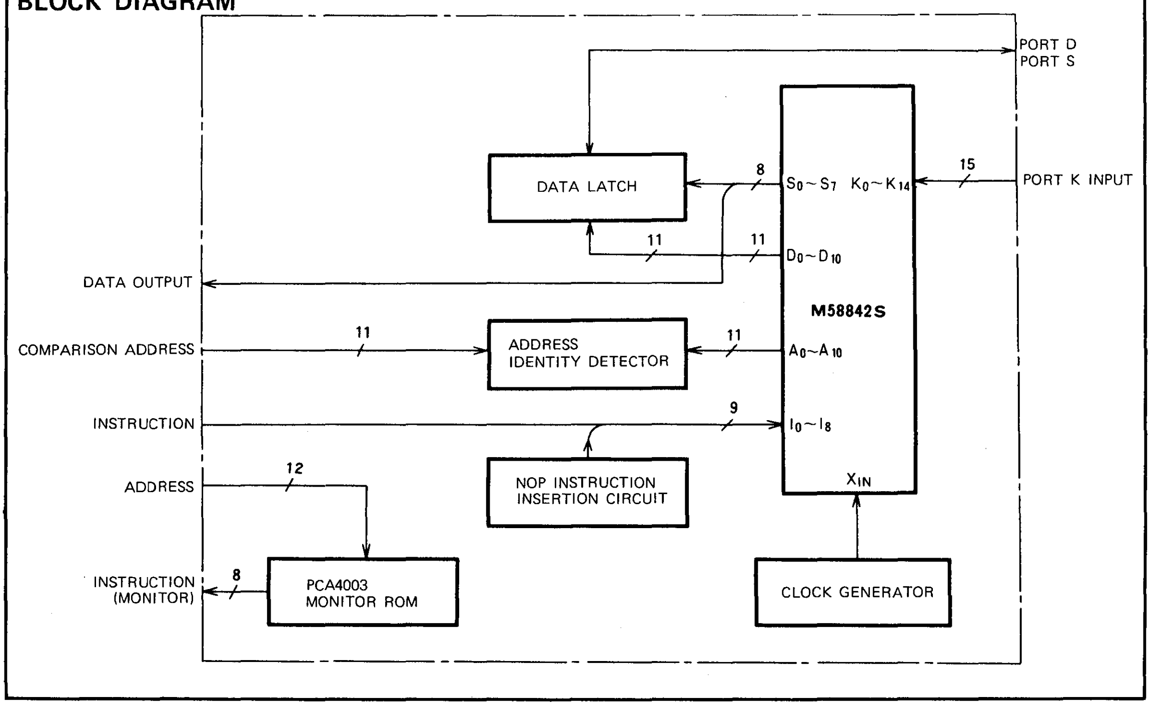
The board and user system can be connected by means of an accessory cable.

FUNCTION

The debugging machine PC4000 operates as a debugging machine for MELPS 4 microcomputers using the monitor program contents of a ROM mounted on the dedicated board. The evaluation chip (M58842S) loaded on the board executes the program stored in the program memory in the PC4000 debugging machine.

The internal status of the evaluation chip is read out under monitor CPU control when single-step operation and breakpoint operation are halted.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

SPECIFICATIONS

Item		Specification
Applicablemicrocomputers		M58843-XXXP
Clock frequency	Package	455kHz
	Variable range	300～600kHz
Applicable debugging machine		PC4000 (connected by a card edge connector)
Power supply		Supplied by the PC4000
Connection to user's system		By an accessory cable
Debugging functions (contents of monitor EPROM)		• Program execution from any address and halt • Single-step operation • Data writing to EPROM and reading • Confirmation and change of the contents of program RAM • Serial data transfer to an external device • Confirmation and modification of the RAM data in the evaluation chip and verification and modification of the following registers and flags: • Program counter • Data pointer • Accumulator • B register • H/L registers

MELPS 4 DEDICATED BOARD

DESCRIPTION

The PCA4004 is a dedicated MELPS 4 board for use with the PC4000 debugging machine for the M58844-XXXSP 4-bit single-chip microcomputers and is used by inserting the board in the PC4000 cabinet.

FEATURES

- Connection to user's system by a flat cable
- Single-step operation and breakpoint operation capability from the PC4000 debugging machine keyboard. Debugging functions such as confirmation of internal register contents.
- Can be used with external and internal clocks

APPLICATIONS

The development of hardware and software for system using the MELPS 4(M58844-XXXSP) single-chip microcomputers.

CONFIGURATION

As can be seen in the block diagram, the PCA4004 consists of the following hardware.

(1) Evaluation chip (M58842S) and peripheral circuitry
(2) EPROM with the PC4000 monitor program
(3) Single-step and breakpoint control circuit
(4) Program memory interface circuit
(5) Input/output buffer/latch circuit

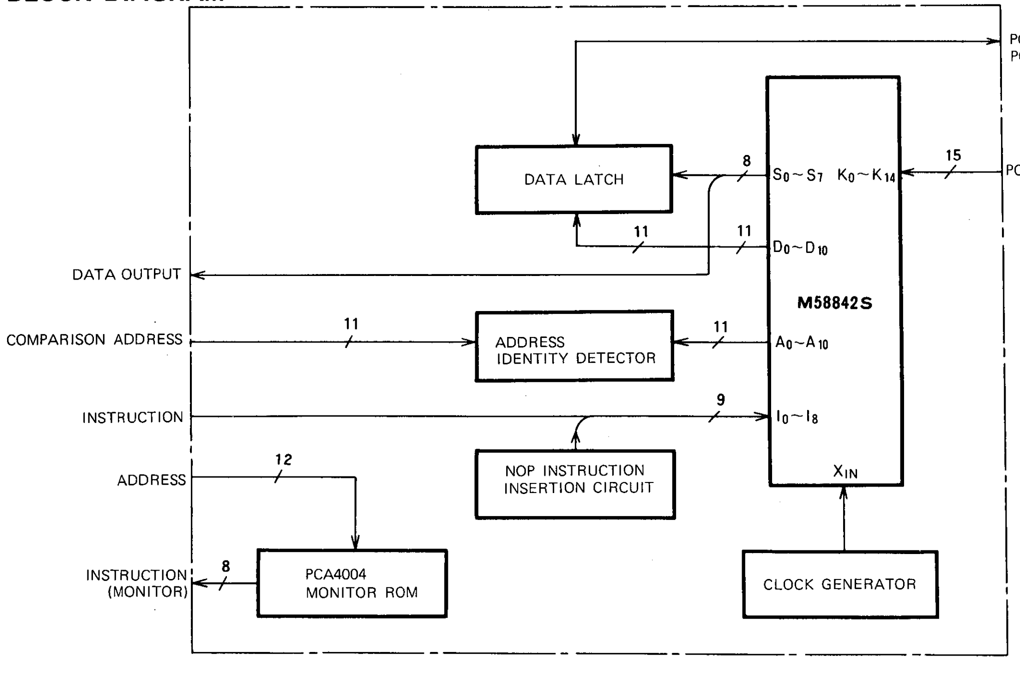
The board and user system can be connected by means of an accessory cable.

FUNCTION

The debugging machine PC4000 operates as a debugging machine for MELPS 4 microcomputers using the monitor program contents of a ROM mounted on the dedicated board. The evaluation chip (M58842S) loaded on the board executes the program stored in the program memory in the PC4000 debugging machine.

The internal status of the evaluation chip is read out under monitor CPU control when single-step operation and breakpoint operation are halted.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

SPECIFICATIONS

Item		Specification
Applicable microcomputers		M58844-XXXSP
Clock frequency	Package	455kHz
	Variable range	300 ~ 600kHz
Applicable debugging machine		PC4000 (connected by a card edge connector)
Power supply		Supplied by the PC4000
Connection to user's system		By an accessory cable
Debugging functions (contents of monitor EPROM)		● Program execution from any address and halt ● Single-step operation ● Data writing to EPROM and reading ● Confirmation and change of the contents of program RAM ● Serial data transfer to an external device ● Confirmation and modification of the RAM data in the evaluation chip and verification and modification of the following registers and flags: ● Program counter ● Data pointer ● Accumulator ● B register ● H/L registers

DESCRIPTION

The PCA4005 is a dedicated MELPS 4 board for use with the PC4000 debugging machine for the M58845-XXXSP 4-bit single-chip microcomputers and is used by inserting the board in the PC4000 cabinet.

FEATURES

- Connection to user's system by a flat cable
- Single-step operation and breakpoint operation capability from the PC4000 debugging machine keyboard. Debugging functions such as confirmation of internal register contents.
- Can be used with external and internal clocks

APPLICATIONS

The development of hardware and software for systems using the MELPS 4 (M58845-XXXSP) single-chip microcomputers.

CONFIGURATION

As can be seen in the block diagram, the PCA4005 consists of the following hardware.

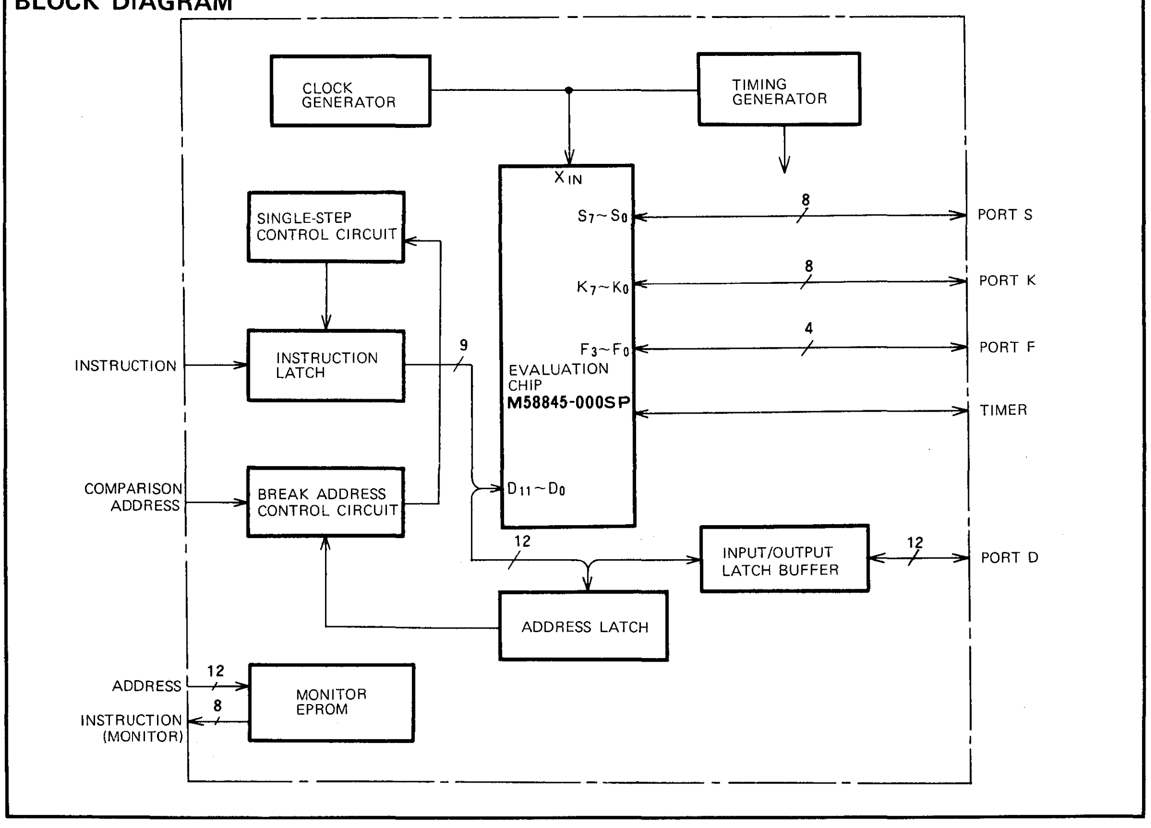
(1) Evaluation chip (M58845-000SP) and peripheral circuitry
(2) EPROM with the PC4000 monitor program
(3) Single-step and breakpoint control circuit
(4) Program memory interface circuit
(5) Input/output buffer/latch circuit

The PC4000 is connected to this board using a card edge connector and this board is connected to the user system by means of an accessory cable.

FUNCTION

The debugging machine PC4000 operates as a debugging machine for the M58845-XXXSP using the contents of the monitor ROM mounted on the dedicated board. The evaluation chip (M58845-000SP) loaded on the board executes the program stored in the program memory in the PC4000 debugging machine.

BLOCK DIAGRAM



SPECIFICATIONS

Item		Specification
Applicable microcomputers		M58845-XXXSP
Clock frequency	Package	455kHz
	Variable range	300～600kHz
Applicable debugging machine		PC4000 (connected by a card edge connector)
Power supply		Supplied by the PC4000 when inserted into debugging machine
Connection to user's system		By an accessory cable
Debugging functions (contents of monitor EPROM)		● Program execution from any address and halt ● Data writing to EPROM and reading ● Confirmation and change of the contents of program RAM ● Serial data transfer to an external device ● Confirmation and modification of the evaluation chip RAM data and register contents

13

DESCRIPTION

The PCA4011 is a dedicated MELPS 41 board for use with the PC4000 debugging machine for the M58494-XXXP 4-bit single-chip microcomputers and is used by inserting the board in the PC4000 cabinet.

FEATURES

- Connection to user's system by a flat cable
- Single-step operation and breakpoint operation capability from the PC4000 debugging machine keyboard. Debugging functions such as confirmation of internal register contents.
- Can be used with external and internal clocks

APPLICATIONS

The development of hardware and software for systems using the MELPS 41 (M58494-XXXP) single-chip microcomputers.

CONFIGURATION

As can be seen in the block diagram, the PCA4011 consists of the following hardware:

(1) Evaluation chip (M58494-000P) and peripheral circuitry
(2) ROM with the PC4000 monitor program
(3) Single-step and breakpoint control circuit
(4) Program memory interface circuit
(5) Input/output buffer/latch circuit

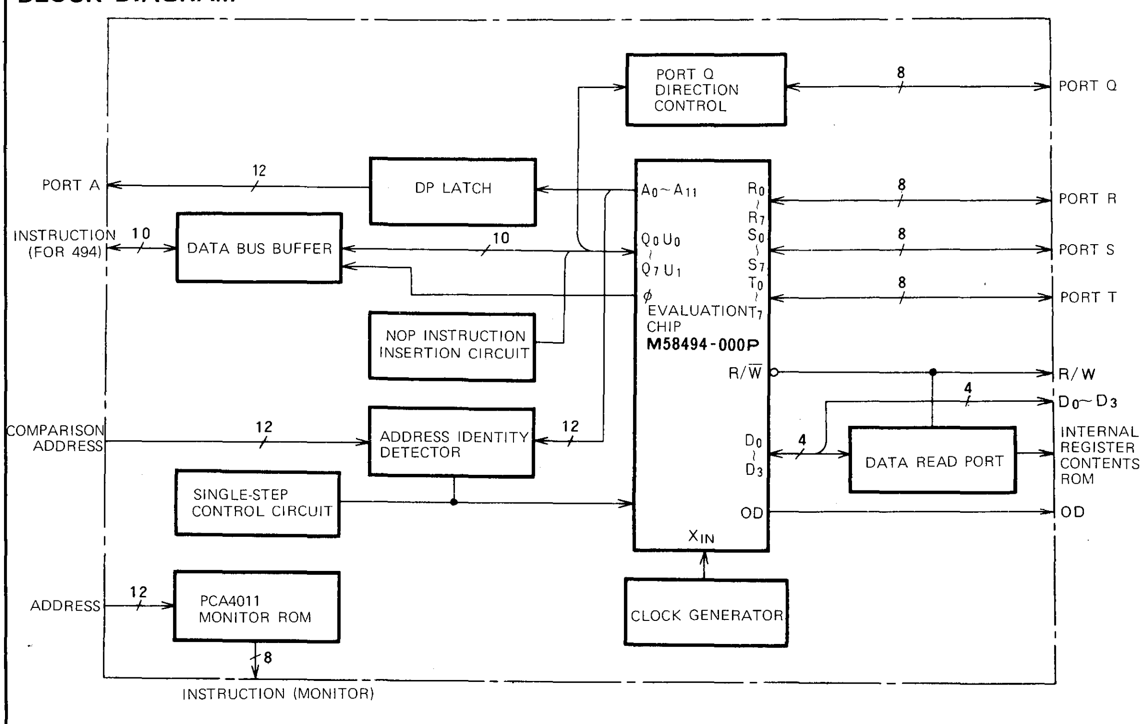
The PC4000 is connected to this board using a card edge connector and this board is connected to the user system by means of an accessory cable.

FUNCTION

The debugging machine PC4000 operates as a debugging machine for the MELPS 41 using the contents of the monitor ROM mounted on the dedicated board. The evaluation chip (M58494-000P) loaded on the board executes the program stored in the program memory in the PC4000 debugging machine.

The internal status of the evaluation chip is read out under monitor CPU control when single-step operation and breakpoint operation are halted.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

SPECIFICATIONS

Item		Specification
Applicable microcomputers		M58494-XXXP
Clock frequency	Package	250kHz
	Variable range	100～350kHz (externally connected)
Applicable debugging machine		PC4000 (connected by a card edge connector)
Power supply		Supplied by the PC4000
Connection to user's system		By an accessory cable
Debugging functions (contents of monitor EPROM)		● Program execution from any address and halt ● Data writing to EPROM and reading ● Confirmation and change of the contents of program RAM ● Serial data transfer to an external device ● Confirmation and modification of the RAM data in the evaluation chip (M58494-000P) and the contents of the following registers and flags: ● Program counter ● S register ● Data pointer ● T register ● Stack pointer ● CY flag ● Accumulator ● B register ● Q register ● R register

MELPS 42 DEDICATED BOARD

DESCRIPTION

The PCA4012 is a dedicated MELPS 42 board for use with the PC4000 debugging machine for the M58496-XXXP 4-bit single-chip microcomputers and is used by inserting the board in the PC4000 cabinet.

FEATURES

- Connection to user's system by flat cables
- Single-step operation and breakpoint operation capability from the PC4000 debugging machine keyboard. Debugging functions such as confirmation of internal register contents.
- Can be used with external and internal clocks

APPLICATIONS

The development of hardware and software for systems using the MELPS 42 (M58496-XXXP) single-chip microcomputers.

CONFIGURATION

As can be seen in the block diagram, the PCA4012 consists of the following hardware:

(1) Evaluation chip (M58496-000P) and peripheral circuitry
(2) ROM with the PC4000 monitor program
(3) Single-step and breakpoint control circuit
(4) Program memory interface circuit
(5) Input/output buffer/latch circuit

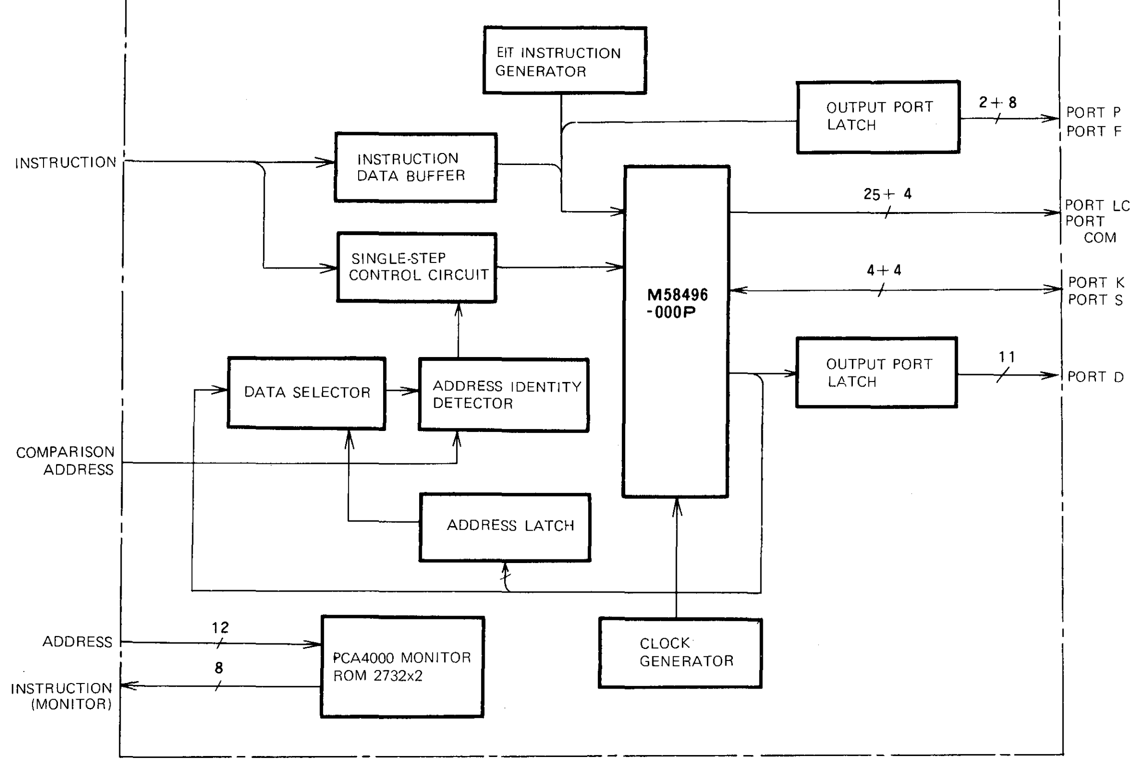
The PC4000 is connected to this board using a card edge connector and this board is connected to the user system by means of an accessory cable.

FUNCTION

The debugging machine PC4000 operates as a debugging machine for the MELPS 42 using the contents of the monitor ROM mounted on the dedicated board. The evaluation chip (M58496-000P) loaded on the board executes the program stored in the program memory in the PC4000 debugging machine.

The internal status of the evaluation chip is read out under monitor CPU control when single-step operation and breakpoint operation are halted.

BLOCK DIAGRAM



SPECIFICATIONS

Item		Specification
Applicable microcomputers		M58496-XXXP
Clock frequency	Package	4.194304MHz
	Variable range	2 ~ 4.2MHz
Applicable debugging machine		PC4000 (connected by a card edge connector)
Power supply		Supplied from the PC4000 when inserted into the debugging machine
Connection to user's system		By an accessory cable
Debugging functions (contents of monitor EPROM)		● Program execution from any address and halt ● Single-step operation ● Data writing to EPROM and reading ● Confirmation and change of the contents of program RAM ● Serial data transfer to an external device ● Confirmation and change of the RAM in the evaluation chip (M58496-000P) and the contents of the following registers and flags: ● Data pointer ● Accumulator ● B register ● CY flag

MELPS 42 DEDICATED BOARD

DESCRIPTION

The PCA4014 is a dedicated MELPS 42 board for use with the PC4000 debugging machine for the M58497-XXXP 4-bit single-chip microcomputers and is used by inserting the board in the PC4000 cabinet.

FEATURES

- Connection to user's system by flat cables.
- Single-step operation and breakpoint operation capability from the PC4000 debugging machine keyboard. Debugging functions such as confirmation of internal register contents.
- Can be used with external and internal clocks.

APPLICATIONS

The development of hardware and software for systems using the MELPS 42 (M58497-XXXP) single-chip microcomputers.

CONFIGURATION

As can be seen in the block diagram, the PCA-4014 consists of the following hardware:

(1) Evaluation chip (M59497-000P) and peripheral circuitry
(2) EPROM with the PC4000 minotor program
(3) Single-step and breakpoint control circuit
(4) Program memory interface circuit
(5) Input/output buffer/latch circuit

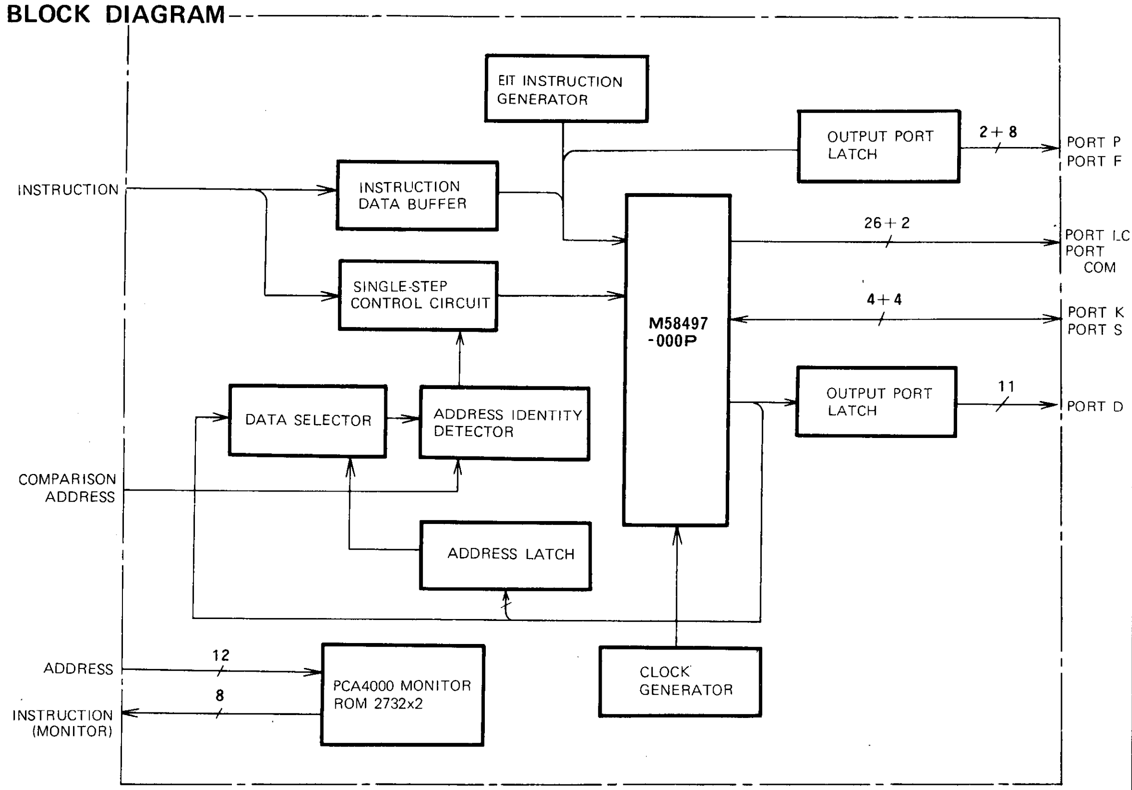
The PC4000 is connected to this board using a card edge connector and this board is connected to the user system by means of an accessory cable.

FUNCTION

The debugging machine PC4000 operates as a debugging machine for the MELPS 42 using the contents of the monitor ROM mounted on the dedicated board. The evaluation chip (M58497-000P) loaded on the board executes the program stored in the program memory in the PC4000 debugging machine.

The internal status of the evaluation chip is read out under monitor CPU control when single-step operation and breakpoint operation are halted.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

SPECIFICATIONS

Item		Specification
Applicable microcomputers		M58497-XXXP
Clock frequency	Package	480KHz
	Variable range	2 ~ 4.2MHz 32KHz
Applicable debugging machine		PC4000 (connected by a card edge connector)
Power supply		Supplied from the PC4000 when insweted into the debugging machine
Connector to user's system		Bv an accessory cable
Debugging functions (contents of monitor EPROM)		● Program execution from any address and halt ● Single-step operation ● Data writing to EPROM and reading ● Confirmation and change of the contents of program RAM ● Serial data transfer to an external device ● Confirmation and change of the RAM in the evaluation chip (M58497-000P) and the contents of the following registers and flags. ● Data pointer ● Accumulator ● B register ● CY flag

DESCRIPTION

The PCA8400 is a dedicated MELPS 8-48 board for use with the PC4000 debugging machine for the 8-bit single-chip microcomputers and is used by inserting the board in the PC4000 cabinet.

FEATURES

- Connection to user's system by means of a 40-pin DIL plug
- Control circuits and connectors for the M5L8748S writing adaptor (PC4100)

APPLICATIONS

The development of hardware and software for systems using the MELPS 8-48 8-bit single-chip microcomputers.

CONFIGURATION

As can be seen in the block diagram, the PCA8400 consists of the following hardware:

(1) Evaluation chip (M5L8039P-6) and peripheral circuitry
(2) ROM with the PC4000 monitor program
(3) Single-step and breakpoint control circuit
(4) Program memory interface circuit
(5) Input/output buffer/latch circuit

The PC4000 is connected to this board using a card edge connector and this board is connected to the user system by means of an accessory cable.

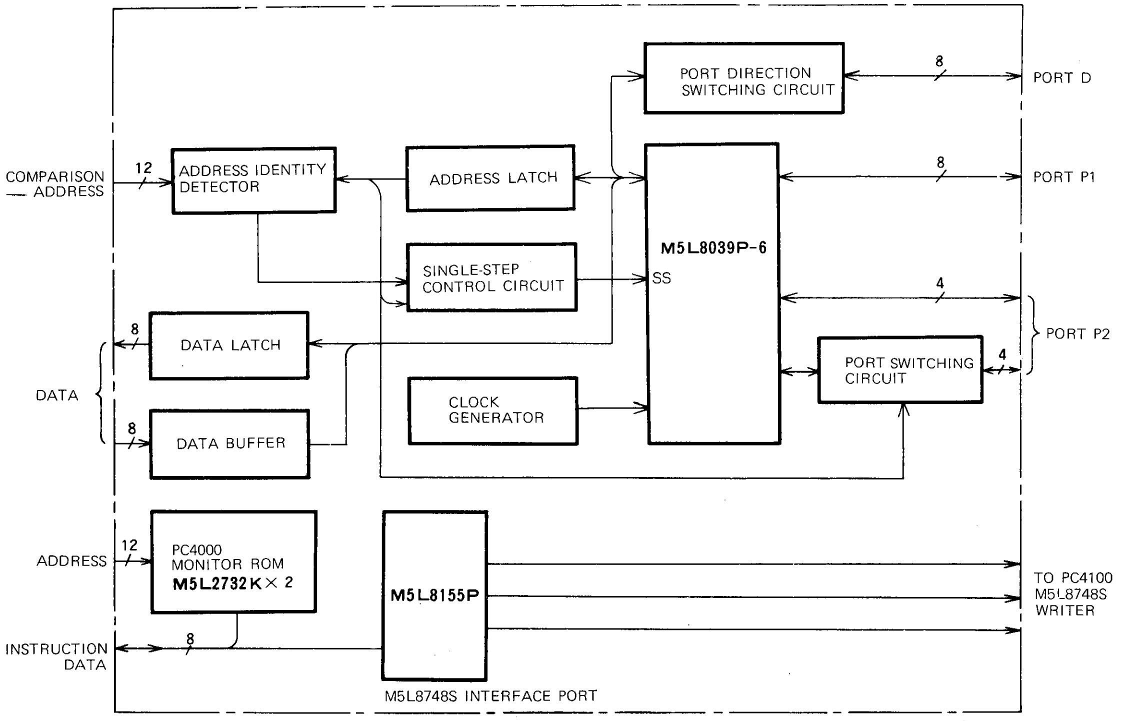
FUNCTION

The debugging machine PC4000 operates as a debugging machine for the MELPS 8-48 using the contents of the monitor ROM mounted on the dedicated board. The evaluation chip (M5L8039P-6) loaded on the board executes the program stored in the program memory in the PC4000 debugging machine.

The internal status of the evaluation chip is read out under monitor CPU control when single-step operation and breakpoint operation are halted.

An interface and connector to enable connection to the M5L8748S writing adaptor PC4100 has been provided, allowing programs to be written and read from the M5L8748S.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

SPECIFICATIONS

Item		Specification
Applicable microcomputers		M5L8048-XXXP M5L8049-XXXP M5L8748S (PC4100) M5L8039P-6
Clock frequency	Package	6.144MHz
	Variable range	1~6.144MHz (By changing the oscillator crystal)
Applicable debugging machine		PC4000 (connected by a card edge connector)
Power supply		Supplied from the PC4000 when inserted into the debugging machine
Connection to user's system		By an accessory cable
Debugging functions (contents of monitor EPROM)		● Program execution from any address and halt ● Data writing to EPROM and reading ● Confirmation and change of the contents of program RAM ● Serial data transfer to an external device ● Confirmation and modification of the RAM data in the evaluation chip ● (M5L8039P-6) and the contents of the following registers and flags: ● Program counter ● Accumulator ● PSW
Other		By connecting the PC4100, read and write operations to the M5L8748S can be performed.

M5L8748S PROGRAMMING ADAPTOR

DESCRIPTION
The PC4100 is a programming adaptor for use in writing a program developed on a MELPS 8-48 dedicated PCA8400 board in a PC4000 debugging machine into the internal EPROM of the M5L8748S single-chip microcomputer.

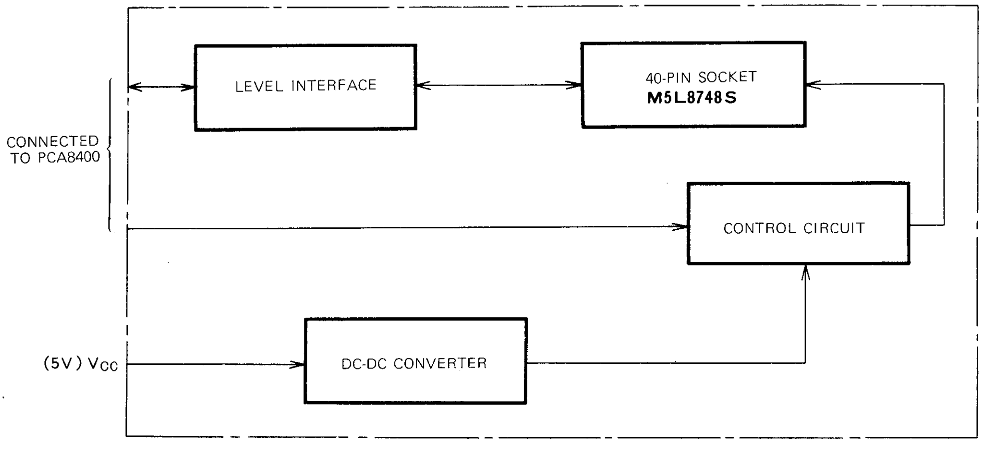
FEATURES
- Interfaced to the PC4000 debugging machine and MEL-PS 8-48 board by means of a connector.
- A protective circuit prevents misinsertion of the device to be writtin into
- No external power supplv is required

APPLICATIONS
Data writing into the M5L8748S

SPECIFICATIONS

Item	Specification
Applicable microcomputer	M5L8748S
Matching boards	PCA8400 (MELPS 8-48 dedicated board) (Connected by a flat cable)
Power supply	Supplied from the PC4000 debugging machine when connected to the PCA8400
Connector	Textool zero insertion force 40-pin socket
Misinsertion protection	Power supply is switched on and off by means of a read relay
Outer dimension	200 (L) x 130 (W) x 38 (H)mm

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

MELPS 4 EVALUATION BOARD

DESCRIPTION

The PCA4301 evaluation board is used as an evaluation board for MELPS 4 4-bit single-chip microcomputers.

When used in the external ROM mode, this board consists of the evaluation chip (M58842S) and the program EPROM (M5L2716K), possessing equivalent functions to the masked ROM M58840-XXXP and M58841-XXXSP. When creating the mask for a developed program, this board is suitable for program verification and running tests.

FEATURES

- Board computer equivalent to the M58840-XXXP and M58841-XXXSP
- Simple program modification using an EPROM
- Connection to user's system by means of a cable
- Built-in clock generator

APPLICATIONS

Program and applications equipment development for MELPS 4 4-bit single-chip microcomputers.

FUNCTIONS

The evaluation chip (M58842S) outputs the value of the program counter, and reads and executes the instruction stored in the appropriate EPROM address.

It is possible to have this board emulate the operation of a single-chip microcomputer.

CONFIGURATION

As can be seen in the block diagram, the PCA4301 consists of the following hardware.

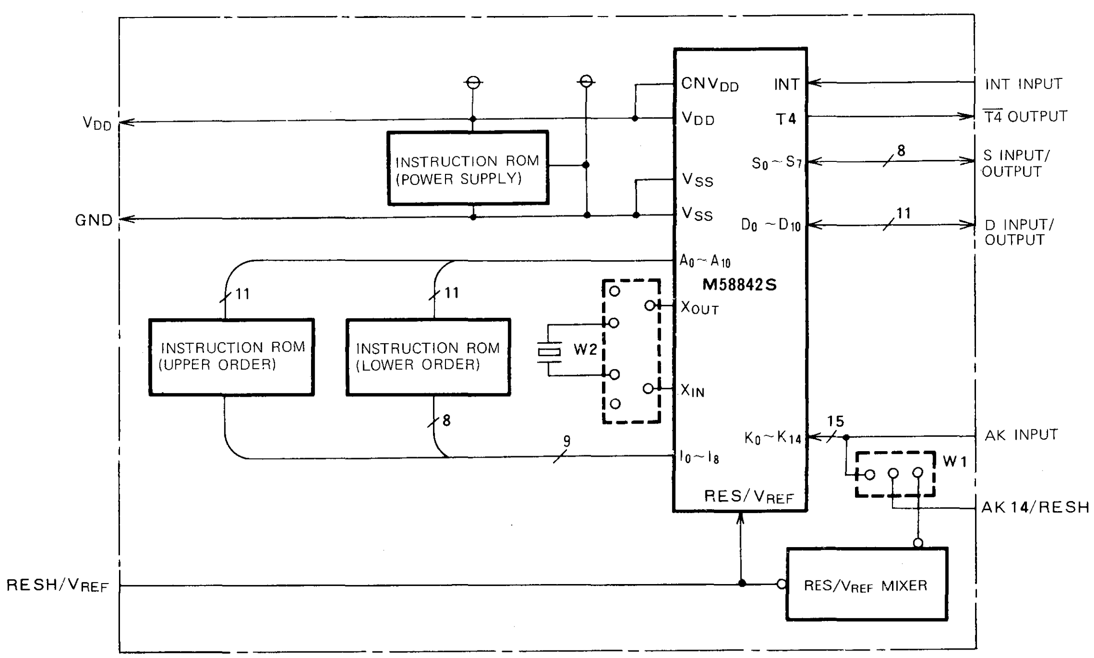
(1) Evaluation chip and peripheral circuitry
(2) Program EPROM socket
(3) EPROM power supply circuit

The board and user system can be connected by means of an accessory cable.

SPECIFICATIONS

Item		Specification
Type		4-bit parallel processor
CPU		Mitsubishi Electric M58842S
Cycle time		10μs (when using a CF600kHz)
Memory	EPROM(H)	2K byte Upper order 1 bit (M5L2716K)
	ERROM(L)	2K byte Lower order 8 bits (M5L2716K)
Built-in clock		CR 300～600kHz CF 600kHz
Interrupts		1 level, 1 factor
Connector used		50-pin straight header
Supporting devices		PC4000 (single-chip microcomputer debugging machine) with PCA4001 (M58840-XXXP board) mounted
Power supply		• −15V±10% (single supply) • Power supply current: 250mA (max) (during execution a NOP instruction)
Outer dimensions		106.7 (L) x 125 (W) x 15 (H)mm

BLOCK DIAGRAM



MELPS 4 EVALUATION BOARD

DESCRIPTION

The PCA4303 evaluation board is used as an evaluation board for MELPS 4 4-bit single-chip microcomputers.

When used in the external ROM mode, this board consists of the evaluation chip (M58842S) and the program EPROM (M5L2716K) possessing equivalent functions to the masked ROM M58843-XXXP. When creating the mask for a developed program, this board is suitable for program verification and running tests.

FEATURES

- Board computer equivalent to the M58843-XXXP
- Simple program modification using an EPROM
- Connection to user's system by means of a cable
- Built-in clock generator

APPLICATIONS

Program and applications equipment development for MELPS 4 4-bit single-chip microcomputers.

FUNCTION

The evaluation chip (M58842S) outputs the value of the program counter, and reads and executes the instruction stored in the appropriate EPROM address.

It is possible to have this board emulate the operation of a single-chip microcomputer.

CONFIGURATION

As can be seen in the block diagram, the PCA4303 consists of the following hardware.

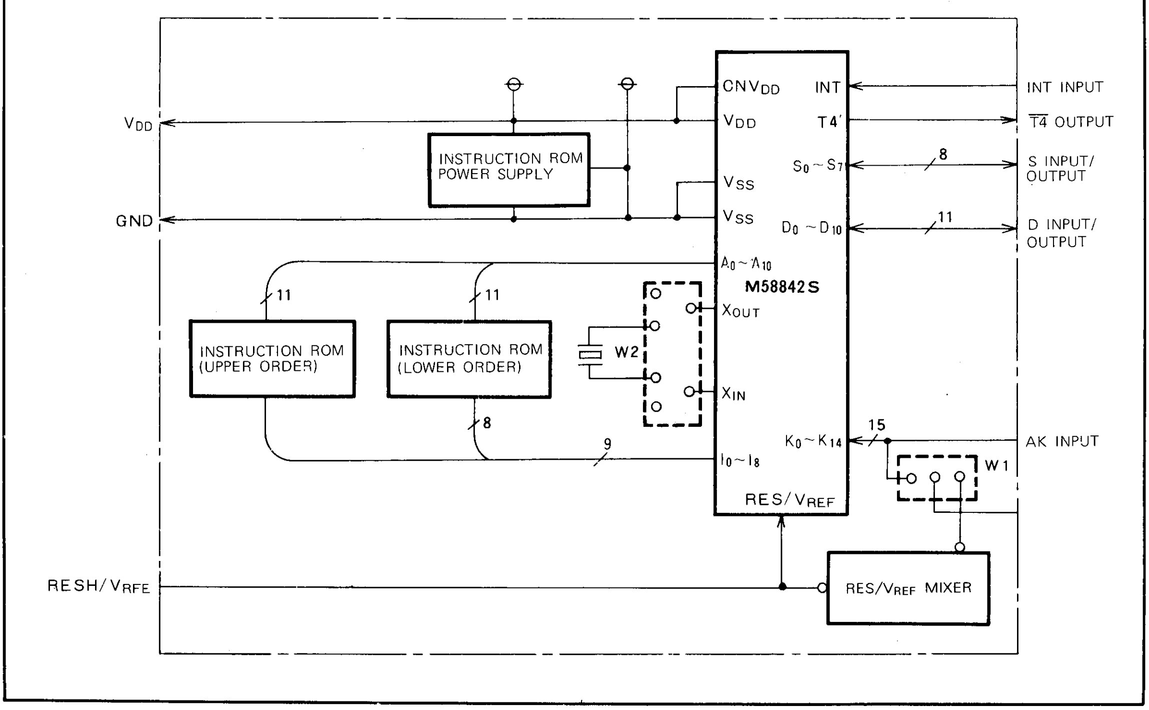
(1) Evaluation chip and peripheral circuitry
(2) Program EPROM socket
(3) EPROM power supply circuit

The board and user system can be connected by means of an accessory cable.

SPECIFICATIONS

Item		Specification
Type		4-bit parallel processor
CPU		Mitsubishi Electric M58842S
Cycle time		10μs (when using a CF600kHz)
Memory	EPROM(H)	2K byte Upper order 1 bit (M5L2616K)
	ERROM(L)	2K byte Lower order 8 bits (M5L2716K)
Built-in clock		CR 300～600kHz CF 600kHz
Interrupts		1 level, 1 factor
Connector used		50-pin straight header
Supporting devices		PC 4000 (single-chip microcomputer debugging machine) with PCA4003 (M 58843-XXXP board) mounted
Power supply		• −15V ±10% (single supply) • Power supply current: 250mA (max) (during execution an NOP instruction)
Outer dimensions		106.7 (L) x 125 (W) x 15 (H)mm

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

MELPS 4 EVALUATION BOARD

DESCRIPTION

The PCA4304 evaluation board is used as an evaluation board for MELPS 4 4-bit single-chip microcomputers.

When used in the external ROM mode, this board consists of the evaluation chip (M58842S) and the program EPROM (M5L2716K), possessing equivalent functions to the masked ROM M58844-XXXSP. When creating the mask for a developed program, this board is suitable for program verification and running tests.

FEATURES

- Board computer equivalent to the M58844-XXXSP
- Simple program modification using an EPROM
- Connection to user's system by means of a cable
- Built-in clock generator

APPLICATIONS

Program and applications equipment development for MELPS 4 4-bit single-chip microcomputers.

FUNCTION

The evaluation chip (M58842S) outputs the value of the program counter, and reads and executes the instruction stored in the appropriate EPROM address.

It is possible to have this board emulate the operation of a single-chip microcomputer.

CONFIGURATION

As can be seen in the block diagram, the PCA4304 consists of the following hardware.

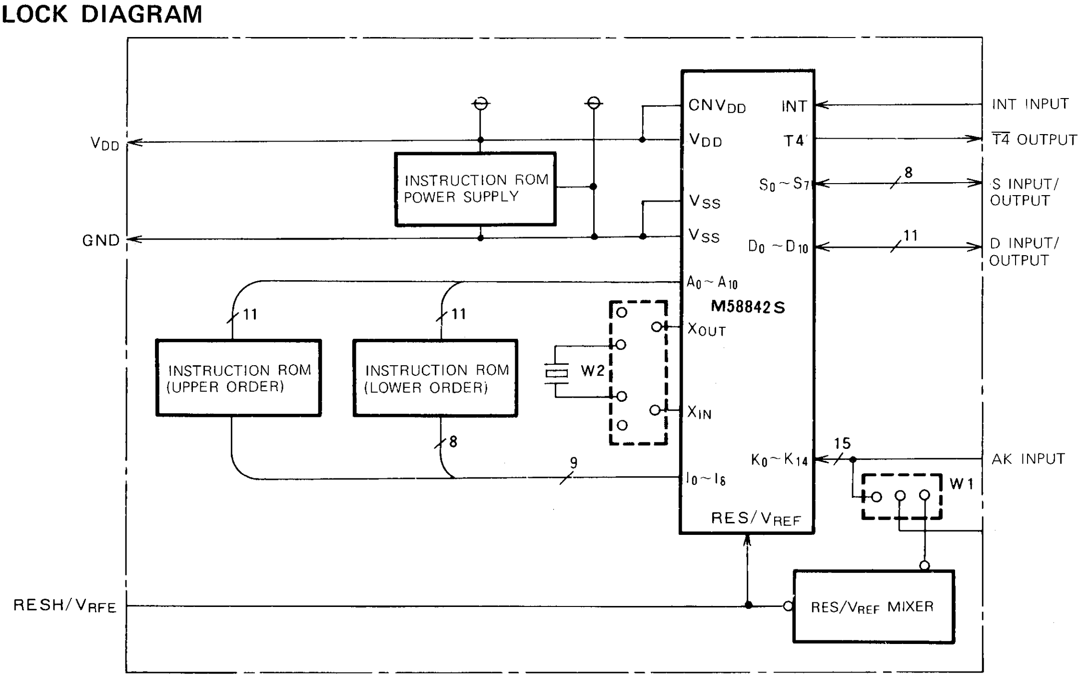
(1) Evaluation chip and peripheral circuitry
(2) Program EPROM socket
(3) EPROM power supply circuit

The board and user system can be connected by means of an accessory cable.

SPECIFICATIONS

Item		Specification
Type		4-bit parallel processor
CPU		Mitsubishi Electric M58842S
Cycle time		10μs (when using a CF600kHz)
Memory	EPROM(H)	2K byte Upper order 1 bit (M5L2616K)
	ERROM(L)	2K byte Lower order 8 bits (M5L2716K)
Built-in clock		CR 300～600kHz CF 600kHz
Interrupts		1 level, 1 factor
Connector used		50-pin straight header
Supporting devices		PC 4000 (single-chip microconputer debugging machine) with PCA 4004 (M58844-XXX SP dedicated board) mounted
Power supply		• −15V ±10% (single supply) • Power supply current: 250mA (max) (during execution an NOP instruction)
Outer dimensions		106.7 (L) x 125 (W) x 15 (H)mm

BLOCK DIAGRAM



13

DESCRIPTION

The PCA4305 evaluation board is used as an evaluation board for MELPS 4 4-bit single-chip microcomputers.

When used in the external ROM mode, this board consists of the evaluation chip (M58845-000SP) and the program EPROM (M5L2716K), possessing equivalent functions to the masked ROM M58845-XXXSP. When creating the mask for a developed program, this board is suitable for program verification and running tests.

FEATURES

- Board computer equivalent to the M58845-XXXSP
- Simple program modification using an EPROM
- Connection to user's system by means of a cable
- Built-in clock generator

APPLICATIONS

Program and applications equipment development for MELPS 4 4-bit single-chip microcomputers.

CONFIGURATION

As can be seen in the block diagram, the PCA4305 consists of the following hardware.

(1) Evaluation chip and peripheral circuitry
(2) Program EPROM socket
(3) EPROM power supply circuit

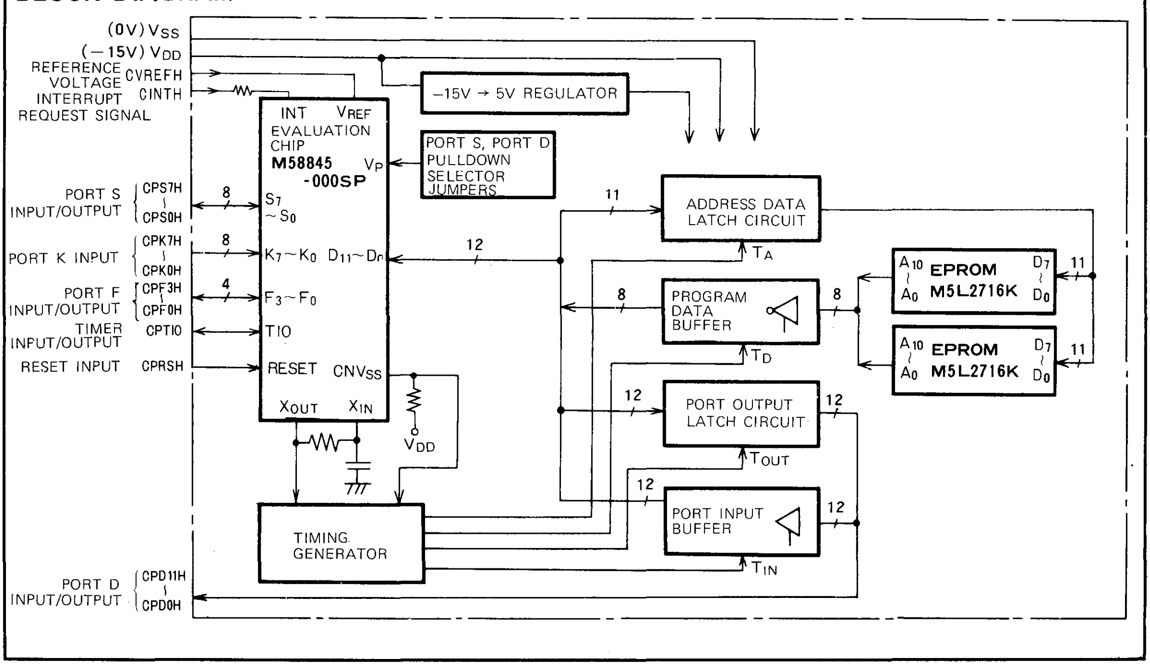
The board and user system can be connected by means of an accessory cable.

FUNCTION

The evaluation chip (M58845-000SP) outputs the value of the program counter, and reads and executes the instruction stored in the appropriate EPROM address.

It is possible to have this board emulate the operation of a single-chip microcomputer.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

SPECIFICATIONS

Item		Specification	
Type		4-bit parallel processor	
CPU		M58845-000SP	
Cycle time		Built-in 455kHz clock	
Memory		Program memory: 2048 words x 9 bits (M5L2716K x2) Data memory: 128 words x 4 bits (Built-in M58845-XXXSP)	
I/O	K	Analog input	1 bit x 8
	S	Output	8 bits x 1
		Input	4 bits x 2
	D	Output	1 bit x 12
		Sense input	1 bit x 12
	F	Output	4 bits x 1
		Input	4 bits x 1
A-D converter		Built-in, ±3 LSB ± accuracy	
Touchkey interface		Built-in	
Subroutine nesting		3 levels (including 1 level of interrupt)	
Clock generator		Built-in (ceramic filter or RC circuit)	
Timers		2	
Interrupts		INT pin	
Power supply		– 15V ±5%, 500mA (max)	
Connector used		68 conductor (34 each side) card-edge connector	
Outer dimensions		190 (L) x 180 (W) x 20 (H)mm	

MELPS 41 EVALUATION BOARD

DESCRIPTION

The PCA4101 evaluation board is used as an evaluation board for MELPS 41 4-bit single-chip microcomputers.

When used in the external ROM mode, this board consists of the evaluation chip (M58494-000P) and the program EPROM (M5L2716K), possessing equivalent functions to the masked ROM M58494-XXXP. When creating the mask for a developed program, this board is suitable for program verification and running tests.

FEATURES

- Board computer equivalent to the M58494-XXXP
- Simple program modification using an EPROM
- Connection to user's system by means of a cable
- Built-in clock generator

APPLICATIONS

Program and applications equipment development for MELPS 41 4-bit single-chip microcomputers.

CONFIGURATION

As can be seen in the block diagram, the PCA4101 consists of the following hardware.

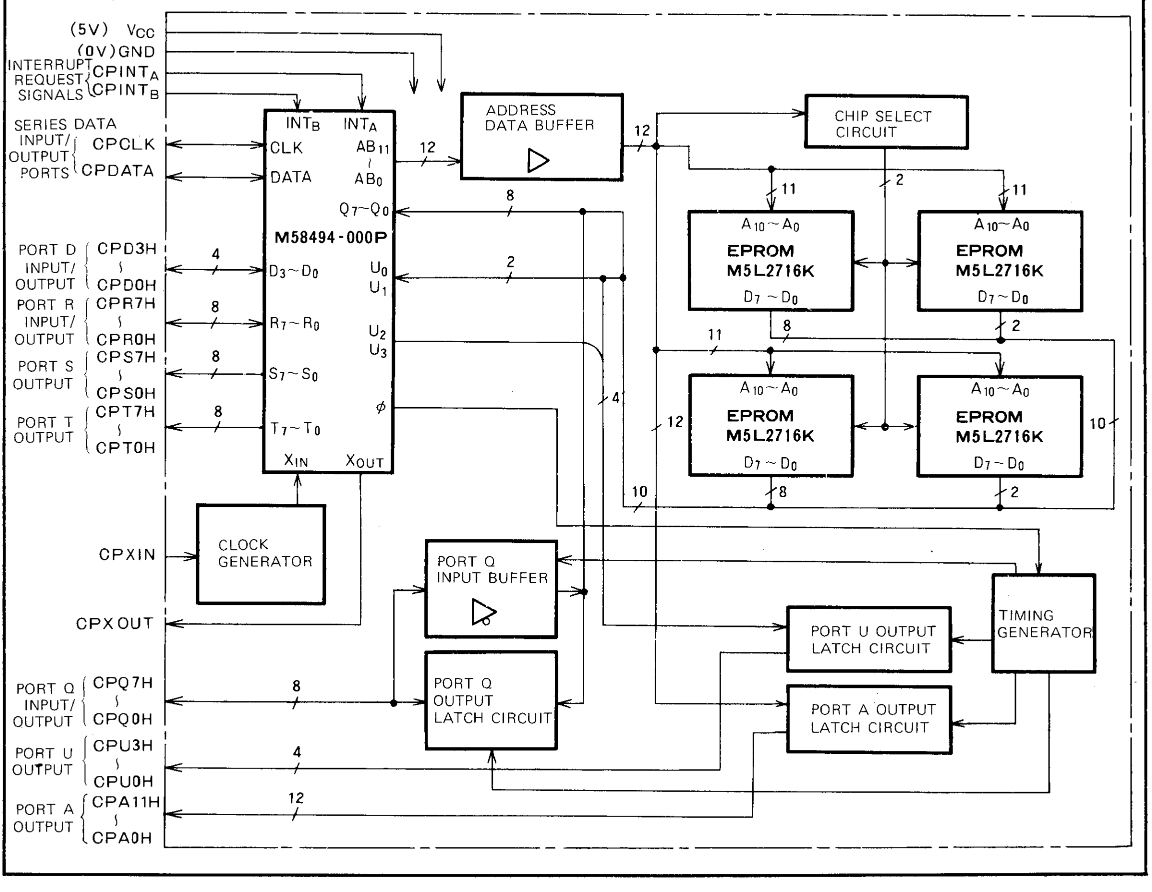
(1) Evaluation chip and peripheral circuitry
(2) Address data latch circuit
(3) Timing generator
(4) Program EPROM socket
(5) Port input/output latch buffers

The board and user system can be connected by means of an accessory cable.

FUNCTION

The evaluation chip (M58494-000P) performs time division of part of the port contents, outputs the value of the program counter, and reads in instructions from the EPROM and executes them. Port output buffer and latch control timing is derived from either an internal or an external clock generator. This board is usable to emulate the operation of the single-chip microcomputer.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

SPECIFICATIONS

Item		Specification
Type		4-bit parallel processor
Evaluation chip		M58494-000P
Clock frequency	Internal	250kHz
	Range	100～350kHz
Memory		Program memory: 4K words x 10 bits (M5L2716K x 4) Data memory: Internal: 32 x 4 bits External: 1024 words x 4 bits
I/O		Input/output ports: for a total of 20 lines R_0～R_7, Q_0～Q_7, D_0～D_3 Output ports: for a total of 32 lines A_0～A_{11}, U_0～U_3, S_0～S_7, T_0～T_7
Interrupts		INT_A INT_B Timer
Power supply		5 V ± 5 %
Connector used		Two 50-pin angled headers

DESCRIPTION

The PCA4201 evaluation board is used as an evaluation board for MELPS 42 4-bit single-chip microcomputers.

When used in the external ROM mode, this board consists of the evaluation chip (M58496-000P) and the program EPROM (M5L2716K), possessing equivalent functions to the masked ROM M58496-XXXP. When creating the mask for a developed program, this board is suitable for program verification and running tests.

FEATURES

- Board computer equivalent to the M58496-XXXP
- Simple program modification using an EPROM
- Connection to user's system by means of a cables
- Built-in clock generator

APPLICATIONS

Program and applications equipment development for MELPS 42 4-bit single-chip microcomputers.

FUNCTION

The evaluation chip (M58496-000P) performs time division of part of the port contents, outputs the value of the program counter, and reads in instructions from the EPROM and executes them. Port output buffer and latch control timing is derived from either an internal or an external clock generator. This board is usable to emulate the operation of the single-chip microcomputer.

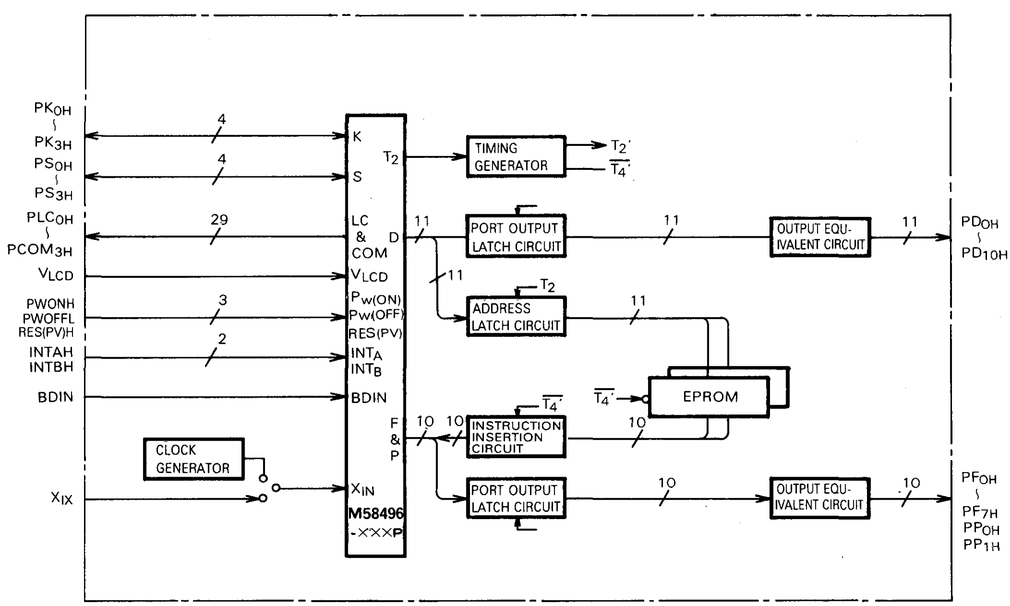
CONFIGURATION

As can be seen in the block diagram, the PCA4201 consists of the following hardware.

(1) Evaluation chip and peripheral circuitry
(2) Address data latch circuit
(3) Timing generator
(4) Program EPROM socket
(5) Port/output latch

The board and user system can be connected by means of accessory cables.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

SPECIFICATIONS

Item		Specification		
Type		4-bit parallel processor		
Evaluation chip		M58496－000P		
Clock frequency	Internal	4.19MHz		
	Range	2～4.2MHz		
Memory		Program memory: 2K words x 10 bits (M5L2716K x2) Data memory: 128 words x 4 bits		
I/O		Dedicated LCD ports	LC_0～LC_{24} COM_0～COM_3	Total of 28 lines
		Input ports	K_0～K_3 S_0～S_3	Total of 8 lines
		Output ports	F_0～F_7 D_0～D_{10} P_0, P_1	Total of 21 lines
Interrupts		INT_A, INT_B, clock interrupt		
Power supply		5 V ± 10%		
Connector used		Two 50-pin angled headers		

MELPS 42 EVALUATION BOARD

DESCRIPTION

The PCA4202 evaluation board is used as an evaluation board for MELPS 42 4-bit single-chip microcomputers.

When used in the external ROM mode, this board consists of the evaluation chip (M58497-000P) and the program EPROM (M5L2716K), prossessing equivalent functions to the masked ROM M58497-XXXP. When creating the mask for a developed program, this board is suitable for program verification and running tests.

FEATURES

- Board computer equivalent to the M58497-XXXP
- Simple program modification using an EPROM
- Connection to user's system by means of flat cables
- Built-in clock generator

APPLICATIONS

Program and applications equipment development for MELPS 42 4-bit single-chip microcomputers.

FUNCTION

The evaluation chio (M58497-000P) performs time division of part of the port contents, outputs the value of the program counter, and reads in instructions from the EPROM and executes them. Port output buffer and latch control timing is derived from either an internal or an external clock generator. This board is usable to emulate the operation of the single-chip microcomputer.

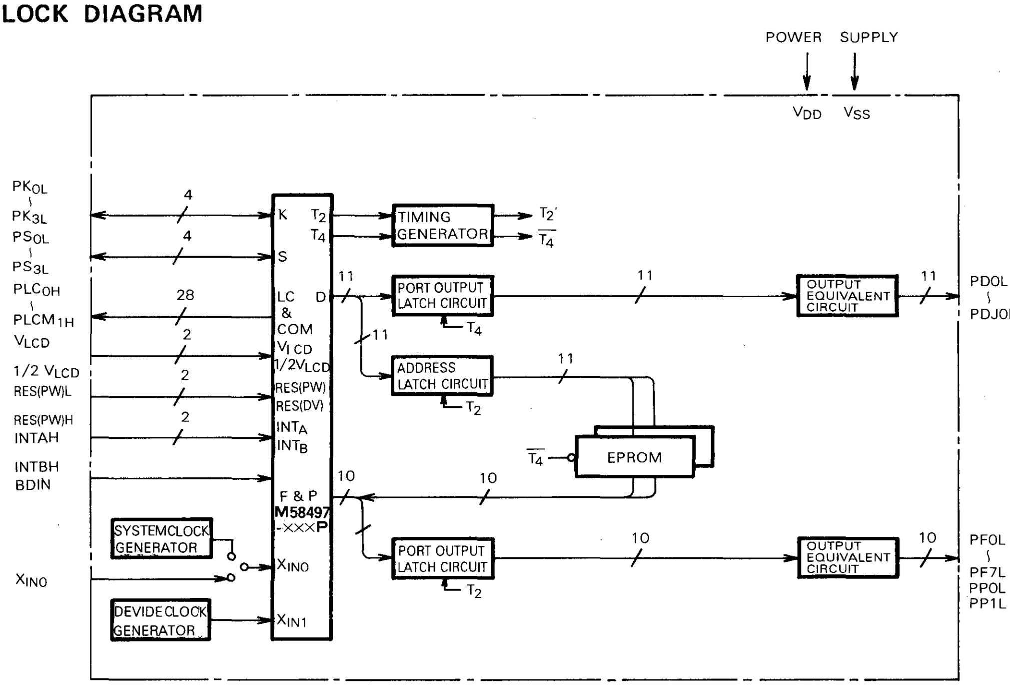
CONFIGURATION

As can be seen in the Block Diagram, the PCA4201 consist of the following hardware.

(1) Evaluation chip and peripheral circuitry
(2) Address data latach circuit
(3) Timing generator
(4) Program EPROM socket
(5) Port input/output latch buffers

The board and user system can be connected by means of an accessory cable.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

SPECIFICATIONS

Item		Specification
Type		4-bit parallel processor
Evaluation chip		M58497–000P
Clock frequency	Package	480KHz
	Divider	32KHz
Memory		Program meory: 2K words x 10 bits (M5L2716K x 2) Data memory: 128 words x 4 bits
I/O		Dedicated LCD ports: $LC_0 \sim LC_{25}$, $COM_0 \sim COM_1$ } Total of 28 lines Input/output ports: $K_0 \sim K_3$, $S_0 \sim S_3$ } Total of 8 lines Output ports: $F_0 \sim F_7$, $D_0 \sim D_{10}$, P_0, P_1 } Total of 21 lines
Interrupts		INT_A, INT_B, clock interrupt
Power supply		5V ± 10%
Connector used		Two 50-pin angled headers

MELPS 8-48 EVALUTION BOARD

DESCRIPTION

The PCA8402 evaluation board is used as an evaluation board for MELPS 8-48 8-bit single-chip microcomputers.

This board consists basically of the external ROM chip (M5L8039P-6) and EPROM (M5L2716K), possessing equivalent functions to the masked ROM M5L8048-XXXP and M5L8049-XXXP. When creating the mask for a developed program, this board is suitable for program verification and running tests.

FEATURES

- Board computer equivalent to the M5L8048-XXXP and M5L8049-XXXP
- Simple program modification using an EPROM
- Connection to user's system socket by means of a 40-pin DIL plug
- Built-in clock generator

APPLICATIONS

Program and applications equipment development for MELPS 8-48 8-bit single-chip microcomputers.

FUNCTION

The evaluation chip (M5L8039P-6) outputs the value of the program counter and reads in instructions from ERROM and executes them.

The board is equivalent in operation to a single-chip microcomputer.

CONFIGURATION

As can be seen in the block diagram, the PCA8402 consists of the following hardware:

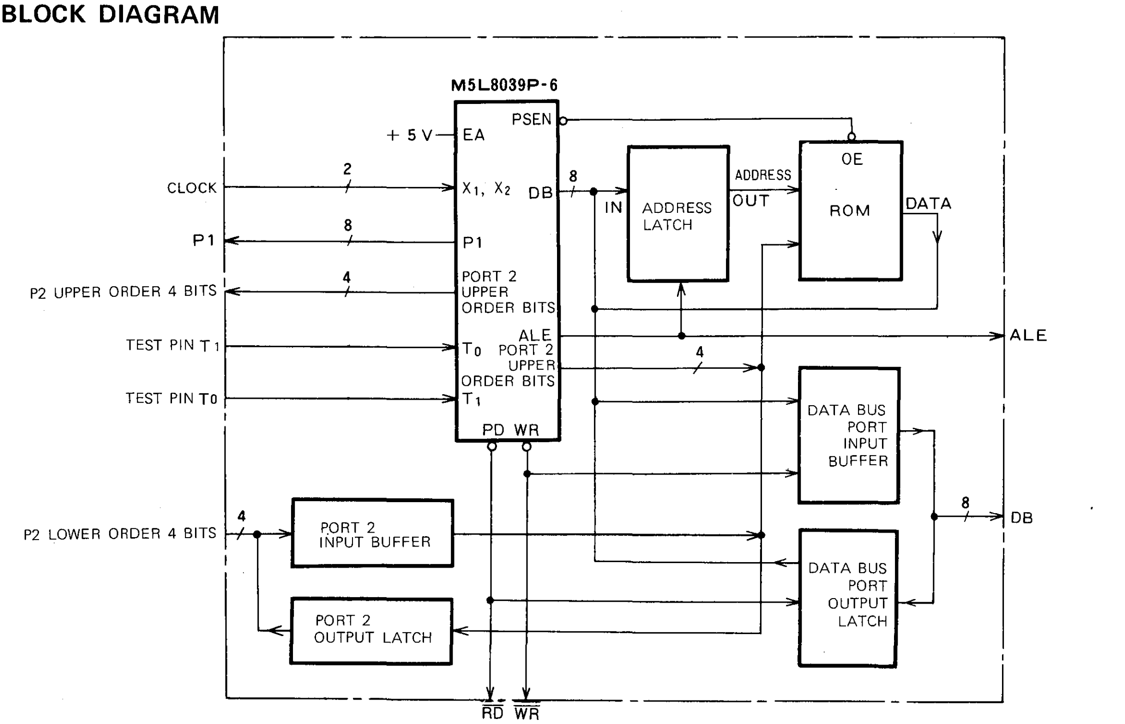
(1) Evaluation chip and peripheral circuitry
(2) Program EPROM socket
(3) EPROM power supply circuit

The board and user system can be connected by means of an accessory cable.

SPECIFICATIONS

Item	Specification
Type	80bit parallel processor
CPU	M5L8039P-6 (equivalent to Intel 8039-6)
Cycle time	Clock supplied by user system (maximum 6MHz)
Memory	Program memory: 2K bytes (M5L2716K) Data memory: 128 bytes (built-in M5L8039P-6)
I/O	8-bit parallel port x3 Test pin x2
Interrupts	INT pin
Power supply	5V ±5%, 500mA (max)
Connector used	40-pin DIL IC socket
Outer dimensions	60 (L) x 65 (W) x 30 (H)mm

BLOCK DIAGRAM



MICROCOMPUTER SOFTWARE

14

SOFTWARE CODES

Software products for Mitsubishi's MELPS microprocessors are designated by the following alphanumeric codes.

1. PROGRAMS

Example: G A 1 AS 0101

- G: Mitsubishi MELPS microprocessor software
- Kind of microprocessor
  - A: MELPS 8/85 8-bit parallel CPU
  - B: Single-chip 4-bit microcomputer
  - C: Single-chip 8-bit microcomputer
  - Z: General
- Operation system
  - 1: For host computer systems
  - 2: For target computer systems
- Kind of program
  - AP: Application program
  - AS: Assembler
  - CR: Control program
  - DP: Diagnostic program
  - OS: operation system
  - SB: Subroutine
  - SM: Simulator
  - SP: Service program
  - TL: Compiler
  - TS: Test program
- Identifying serial number

2. MANUALS AND SUPPORT MATERIALS

Example: G A M – SR 00 – 01 A <6 2 B 0>

- G: Mitsubishi MELPS microprocessor software
- Kind of microprocessor
  - A: MELPS 8/85 8-bit parallel CPU
  - B: Single-chip 4-bit microcomputer
  - C: Single-chip 8-bit microcomputer
  - Z: General
- Language
  - E: English
  - M: Japanese
- Kind of Material
  - HR: Hardware manual
  - PS: Program manual
  - SH: Consumables
  - SR: Software manual or operating manual
  - SS: Software support materials
- Identifying serial number
- Availability
  - A: Unrestricted
- Year of issue—last digit, starting from 1976 = 6
- Month of issue
  - 1: January
  - ⋮
  - 9: September
  - X: October
  - Y: November
  - Z: December
- Fully revised edition
  - A: First
  - B: Second
  - etc.
- Partial revision (advance of full revision resets to 0)

14

AVAILABLE MATERIALS

HOST PROGRAMS

Program	Program code number	Normal shipping media	Source language
MELPS 4 Cross Assembler–MELCOM 70	GBIAS0001	Magnetic tape	FORTRAN (part in assembler)
MELPS 4 Cross Assembler–MELCOM 7000 or COSMO 700	GBIAS0002	Magnetic tape	FORTRAN
MELPS 4 Simulator–MELCOM 70	GBISM0001	Magnetic tape	FORTRAN (part in assembler)
MELPS 4 Paper-Tape Generation Program for PROM Writers–MELCOM 70	GBISP0001	Magnetic tape	FORTRAN (part in assembler)
MELPS 4 Paper-Tape Generation Program for PROM Writers–MELCOM 7000 and COSMO 700	GBISP0002	Magnetic tape	FORTRAN (part in assembler)
MELPS 41 Cross Assembler–MELCOM 70	GBIAS0003	Magnetic tape	FORTRAN (part in assembler)
MELPS 41 Simulator–MELCOM 70	GBISM0002	Magnetic tape	FORTRAN (part in assembler)
MELPS 41 Paper-Tape Generation Program for PROM Writers–MELCOM 70	GBISP0003	Magnetic tape	FORTRAN (part in assembler)
MELPS 42 Cross Assembler–MELCOM 70	GBIAS0010	Magnetic tape	FORTRAN (part in assembler)
MELPS 42 Simulator–MELCOM 70	GBISM0006	Magnetic tape	FORTRAN (part in assembler)
MELPS 42 Paper-tape Generation Program for PROM Writers–MELCOM 70	GBISP0006	Magnetic tape	FORTRAN (part in assembler)

Manuals (in Japanese)	Manual number	Number of pages

MELPS 4 CROSS ASSEMBLER MANUALS

MELPS 4 Assembler Language Manual	GBM–SR00–01A	127
MELPS 4 Cross Assembler Manual–MELCOM 70	GBM–SR00–02A	68
MELPS 4 Cross Assembler Operating Manual–MELCOM 70	GBM–SR00–03A	16

MELPS 4 SIMULATOR MANUALS

MELPS 4 Simulator Manual–MELCOM 70	GBM–SR00–04A	102
MELPS 4 Simulator Operating Manual–MELCOM 70	GBM–SR00–05A	23

MELPS 4 PAPER-TAPE GENERATION PROGRAM MANUALS FOR PROM WRITERS

MELPS 4 Paper-Tape Generation Program Manual for PROM Writers–MELCOM 70	GBM–SR00–06A	17
MELPS 4 Paper-Tape Generation Program Operating Manual for PROM Writers–MELCOM 70	GBM–SR00–07A	8

MELPS 4 HANDBOOK

MITSUBISHI MELPS 4 Single-Chip 4-Bit Microcomputer Handbook–Support Software (Note 1)	GBM–SR10–01A	200

Note 1 : Includes contents of all above manuals concerning MELPS 4 software.

MELPS 41 CROSS ASSEMBLER MANUALS

MELPS 41 Assembly Language Manual	GBM–SR00–08A	162
MELPS 41 Cross Assembler Manual–MELCOM 70	GBM–SR00–09A	75
MELPS 41 Cross Assembler Operating Manual–MELCOM 70	GBM–SR00–10A	8

MELPS 41 SIMULATOR MANUALS

MELPS 41 Simulator Manual	GBM–SR00–11A	93
MELPS 41 Simulator Operating Manual	GBM–SR00–12A	9

MELPS 41 PAPER-TAPE GENERATION PROGRAM MANUALS FOR PROM WRITERS

MELPS 41 Paper-Tape Generation Program Manual for PROM Writers– MELCOM 70	GBM–SR00–13A	8
MELPS 41 Paper-Tape Generation Program Operating Manual for PROM Writers–MELCOM 70	GBM–SR00–14A	11

AVAILABLE MATERIALS

MELPS 42 CROSS ASSEMBLER MANUALS

MELPS 42 Cross Assembler Manual—MELCOM 70	GBM-SR00-31A	34

MELPS 42 SIMULATOR MANUALS

MELPS 42 Simulator Manual	GBM-SR00-32A	86

MELPS 42 PAPER-TAPE GENERATION PROGRAM MANUALS FOR PROM WRITERS

MELPS 42 Paper-Tape Generation Program Manual for PROM Writers—MELCOM 70	GBM-SR00-33A	21

GENERAL DESCRIPTION

MELPS 4/41 software is the name used to designate a software series provided by Mitsubishi for development application programs for equipment in which single-chip microcomputers are used.

MELPS 4/41 software is used as a tool to develop application programs, and comprises all the programs —assembly, PROM programming and mask making— necessary to the manufacture of single-chip microcomputers.

MELPS 4/41 SOFTWARE CONFIGURATION

	Language processor	Program debug	ROM programming
	Cross assembler	Simulator	Paper tape generation program for PROM writers
Host programs	Translates a symbolic source program written in assembly language and produces as output an object program in machine language. There are many kinds of control data and instruction codes and other functions can be changed easily. In MELPS 41, the coding format of is free and a number of input media can be used to input the source program.	Executes and checks a user's program on the pseudo CPU of the host computer. This allows more efficient program debugging and provides more extensive checking than can be accomplished by hardware. Both MELPS 4 and MELPS 41 simulators feature: ●Many flexible control commands ●Trace output, halt table and deleting ●Interrupt operations capable of cyclic interruptions ●Assignment of I/O ports and data The MELPS 41 simulator also has: ●Reverse assembler ●Setting execution time count ●Assignment of memory protect region	Translates assembler binary object programs and outputs paper tape in hexadecimal form. Generates paper tape for PROM writers of Minato Electronics or Takeda Riken. **Automatic mask ROM design program for single-chip microcomputer** M58840-XXXP and M58494-XXXP single-chip 4-bit microcomputers can be automatically programmed to customer's specifications. The plotter instructions for automatic mask production and the program to test the production ROMs are automatically generated from the object program provided by the customer.

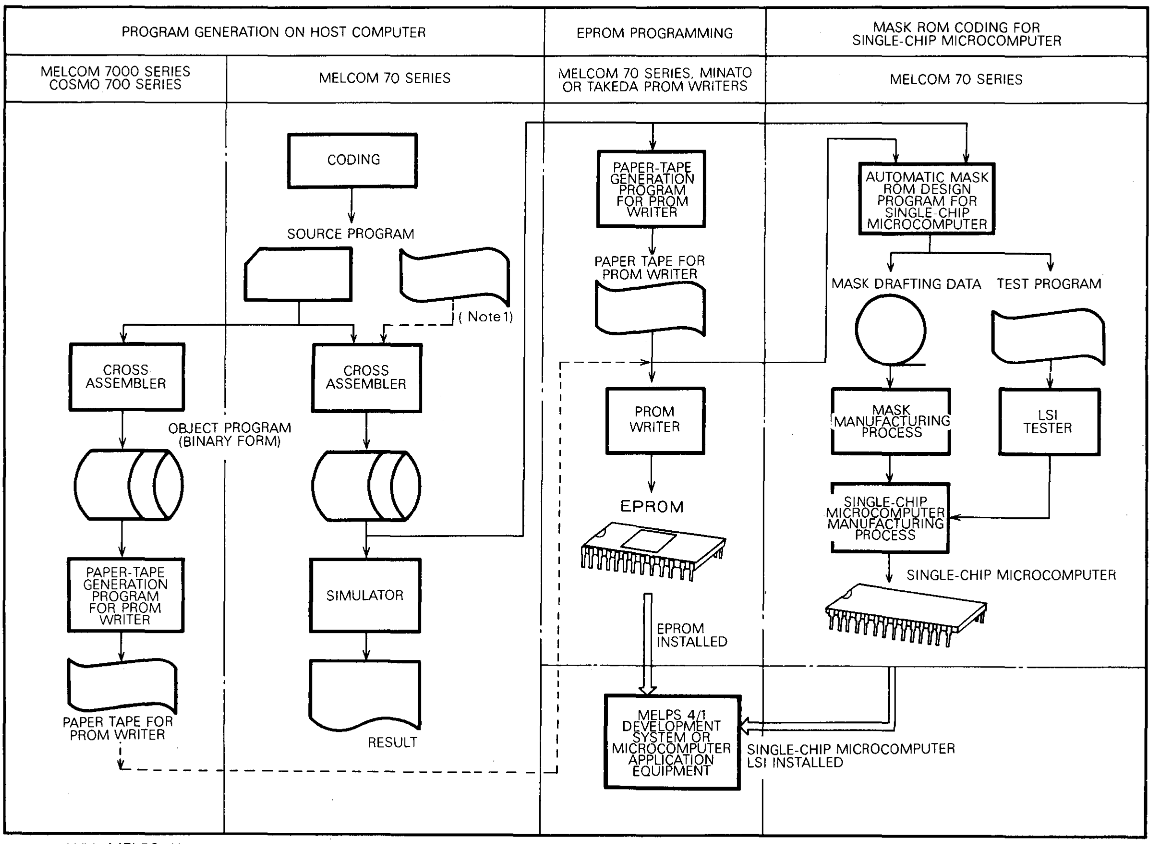
DEVELOPMENT OF APPLICATION PROGRAMS

The user can develop his application programs using MELPS 4/41 software as follows:

The cross assembler is used for object-program generation, and the simulator is used for program debugging. When the application program is finalized, the paper-tape generation program for PROM writers is used to generate a paper tape for the PROM writer.

1. EPROM: Newly developed application programs are programmed in EPROMs, using the PROM writer; then these EPROMs are ready to be installed in sockets of an evaluation breadboard computer or other single-chip microcomputer.
2. Mask-programmable single-chip microcomputer: Mitsubishi Electric has developed a system to produce a mask-programmable single-chip microcomputer to the user's specifications. The object program can be in the PROM-writer format of either Minato Electronics or Takeda Riken.

PROGRAM DEVELOPMENT



Note 1 : With MELPS 41, paper-tape can also be used for source program input.

AVAILABLE MATERIALS

Program	Program code number	Normal shipping media	Source language
HOST PROGRAMS			
MELPS 8/85 PL/Iμ Cross Compiler—MELCOM 7000 (B-version)	GA1TL0100	Magnetic tape	FORTRAN IV
MELPS 8/85 Cross Assembler—MELCOM 70 (A-version)	GA1AS0100	Magnetic tape	FORTRAN IV (part in assembler)
MELPS 8/85 Simulator—MELCOM 70 (B-version)	GA1SM0100	Magnetic tape	FORTRAN IV (part in assembler)
MELPS 8/85 Paper Tape Generation Program for PROM Writers— MELPS 70	GA1SP0100	Magnetic tape	FORTRAN IV (part in assembler)
MELPS 8-48 Cross Assembler—MELCOM 70 (A-verison)		Magnetic tape	FORTRAN IV (part in assembler)
TARGET PROGRAMS			
MELPS 8/85 Self assembler	GA2AS0100	Paper tape	MELPS 8/85 assembler
MELPS 8/85 Editor	GA2SP0103	Paper tape	MELPS 8/85 assembler
MELPS 8 BOM-PTS Basic Operating Monitor	GA2OS0100	Paper tape	MELPS 8/85 assembler
MELPS 8 BOM-B Basic Operating Monitor	GA2OS0101	Paper tape	MELPS 8/85 assembler
MELPS 8/85 Subroutine 1 : Integer Arithmetic Operations	GA2SB0100	Paper tape	MELPS 8/85 assembler

Manuals (in Japanese)	Manual number	Number of pages
MELPS 8/85 PL/Iμ CROSS COMPILER MANUALS		
MELPS 8/85 PL/Iμ Compiler Summary (B-version)	GAM-SR00-07A	74
MELPS 8/85 PL/Iμ Compiler Language Manual (B-version)	GAM-SR00-08A	80
MELPS 8/85 PL/Iμ Cross Compiler Operating Manual (B-version)	GAM-SR00-09A	52
MELPS 8/85 PL/Iμ Cross Compiler Operating Manual—MELCOM 7000	GAM-SR00-10A	28
MELPS 8/85 CROSS ASSEMBLER MANUALS		
MELPS 8/85 Assembly Language Manual (A-version)	GAM-SR00-01A	90
MELPS 8/85 Cross Assembler Operating Manual (A-version)	GAM-SR00-02A	40
MELPS 8/85 Cross Assembler and Simulator Operating Manual—MELCOM 7000	GAM-SR00-04A	16
MELPS 8/85 SIMULATOR MANUAL		
MELPS 8/85 Simulator Operating Manual (B-version)	GAM-SR00-03A	40
MELPS 8/85 SELF ASSEMBLER MANUALS		
MELPS 8/85 Self Assembly Language Manual (B-version)	GAM-SR00-25A	84
MELPS 8/85 Self Assembler Manual—PTS	GAM-SR00-19A	22
MELPS 8/85 Self Assembler Operating Manual	GAM-SR00-24A	32
MELPS EDITOR MANUALS		
MELPS Editor Manual—PTS	GAM-SR00-26A	20
MELPS Editor Operating Manual—PTS	GAM-SR00-27A	32
MELPS 8 BASIC OPERATING MONITOR MANUALS		
MELPS 8 BOM-PTS Basic Operating Monitor Manual	GAM-SR00-18A	18
MELPS 8 BOM-B Basic Operating Monitor Manual	GAM-SR00-23A	14
MELPS 8/85 SUBROUTINE MANUALS		
MELPS 8/85 Subroutine 1 (Integer Arithmetic Operations) Manual	GAM-SR00-17A	18
PAPER-TAPE GENERATION PROGRAM MANUAL FOR PROM WRITERS		
Paper-Tape Generation Program Manual for PROM Writers—MELCOM 70	GAM-SR00-32A	32
MELPS 8-48 CROSS ASSEMBLER MANUALS		
MELPS 8-48 Assembly Language Manual (A-version)	GCM-SR00-01A	148
MELPS 8-48 Cross Assembler Manual (A-version)	GCM-SR00-02A	24
MELPS 8-48 Cross Assembler Operating Manual-MELCOM 70	GCM-SR00-03A	5

GENERAL DESCRIPTION

MELPS 8/85 software is the name used to designate a software series provided by Mitsubishi for developing application programs or operating systems for equipment in which MELPS 8/85 CPUs are used.

MELPS 8/85 software is divided into two parts. The first is that used as a tool to develop application programs, and the second is that used as a part of application programs for MELPS 8/85 CPUs. MELPS 8/85 software can also be divided into two classifications: the first, host programs, which are developed to run on a host computer; and the second, target programs, which are developed to run on a MELPS 8/85 microcomputer.

SOFTWARE CONFIGURATION

	Language processor	Program debug	ROM Programming
Host programs	**PL/Iμ cross compiler** Compiles a source program written in PL/Iμ language and produces as output an object program in machine language. The complete Intel PL/M language is a subset of PL/Iμ. Therefore, any program written in PL/M can be compiled using a PL/Iμ compiler. Additional functions have been included in PL/Iμ that make it easy to use.	**Simulator** Executes and checks a user's program on the pseudo CPU in a host computer. This allows more efficient program debugging and provides more extensive checking than can be accomplished by hardware. FEATURES: • Provides traces and other debugging aids • Provides simulated I/O operations • Provides simulated interrupt operations • Simplifies program modifications • Provides flexibility for symbolic addresses • Provides data for evaluation of execution time • Batch or conversational processing can be used	**Paper-tape generation program for PROM writers** Paper tapes for PROM writers can be generated by a cross compiler or a cross assembler. The tapes contain translated absolute object programs. Many kinds of paper tapes can be generated for the PROM writers of Takeda Riken, Minato Electronics, DATA I/O and PRO-LOG.
Host programs	**Cross assembler** Translates a symbolic source program written in assembly language and produces as output an object program in machine language. Parts of a program can be translated and tested, after which they can be combined and linked because the individual outputs are relocatable. This makes it easy to develop modules and then combine them to form a complete program.		**Mask-programmable ROM** M58735-XXXP 4K-byte mask-programmable ROMs can be automatically programmed to customer's specifications. The plotter instructions for automatic mask production and the program to test the production ROMs are automatically generated from the object program provided by the customer.
Target programs	**Self assembler** Translates a source program written in assembly language into an object program written in machine language for execution on the microcomputer. Paper-tape is used as the source-program input medium. The assembled object program is in MELPS 8/85 binary object format and is punched out on paper tape. Functions and language specifications of the assembler are included in the specifications of the cross assembler.	**BOM-B and BOM-PTS Basic operating monitors** This is a basic operating monitor program to control execution of a program as well as to facilitate debugging a program. This program has a structure that makes it easy to expand or reduce the functions. The monitor can be used for a MELPS 8/85 CPU with any memory arrangement or organization. FUNCTIONS • Program execution control • Program debugging • Input/output control • Program loading • Memory readout MEMORY CAPACITY BOM-B : 2K byte BOM-PTS : 7.5K byte	**Editor** Facilitates editing of source programs and increases the efficiency of program development. FUNCTION Loading the text from a keyboard or a paper-tape reader to the work area, editing the text by means of commands from a keyboard and controlling I/Os. **Integer Arithmetic Operation Subroutines** 10 subroutines are provided that can perform arihmetic operations with binary or decimal integers and logical operations. These subroutines facilitate handling of information of 16 bits or 32 bits for expressions of larger value.

14

DEVELOPMENT OF APPLICATION PROGRAMS

The user can develop his application programs using MELPS 8/85 software in any of three ways.

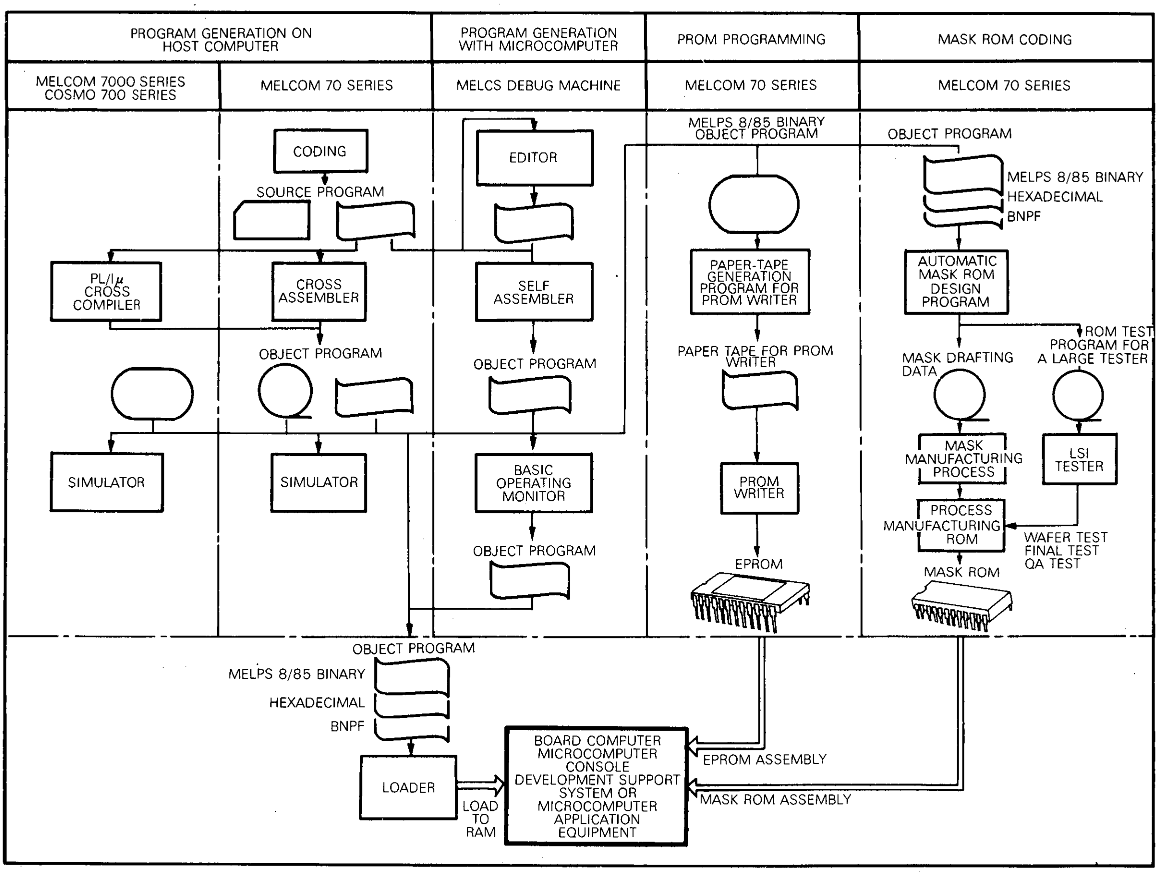
1. On a host computer: the MELPS 8/85 cross compiler or cross assembler is used for object-program generation, and the simulator is used for program debugging.
2. On a microcomputer: the MELPS 8/85 assembler is used for object-program generation, and the microcomputer is used for execution and implementation of programs.
3. On a combination of host computer and microcomputer: object programs are produced by the MELPS 8/85 cross compiler and/or the MELPS 8/85 cross assembler on a host computer. The object programs are debugged and implemented on a MELPS 8/85 microcomputer under control of the basic operating monitor.

The user can develop MELPS 8/85 programs using general-purpose subroutines for functions such as arithmetic operations, input/output control and logical operations.

Full utilization of these subroutines can facilitate program development, debugging and implementation. The final media of a developed program can be any of the following:

1. Paper tape: There are four basic forms of object programs on paper-tape: MELPS 8/85 binary, simple (IPL) binary, hexadecimal and BNPF. Object programs on paper tape are stored in RAMs and are loaded by the appropriate loader.
2. PROM: The developed program is programmed in a PROM using the PROM writer; then this PROM is installed in the appropriate PROM socket of the microcomputer.
3. Mask ROM: Mitsubishi Electric is ready to produce a mask ROM to a user's specifications. The object program can be in MELPS 8/85 binary, hexadecimal or BNPF form.

PROGRAM DEVELOPMENT



MITSUBISHI ELECTRIC

CROSS ASSEMBLER

DESCRIPTION

The MELPS 4 cross assembler has been prepared for the development of application programs suitable for equipment using the M58840-XXXP, M58841-XXXSP, M58842S, M58843-XXXP, M58844-XXXSP, M58846-XXXSP and M58847-XXXSP single-schip 4-bit microcomputer.

This cross assembler not only provides many pseudo instructions, control commands, and control data for improving programming efficiency, but it also provides program versatility for changing instruction codes and functions.

FEATURES OF THE CROSS ASSEMBLER

- 21 types of control data
- Instruction codes and functions easily changed
- Catalogs the control data in disk storage
- Constants can also be expressed in non-decimal notations
- Expandability using pseudo instructions
- Printouts available from the tables and cross-reference lists
- Execution computer: MELCOM 70 (memory capacity more than 24K words, monitor BDOS)
- Implementation language: FORTRAN IV (parts are written in assembly language)

FEATURES OF THE ASSEMBLY LANGUAGE

- 6 pseudo instructions
- 10 simulator control commands
- 68 machine instructions
- Decimal numbers can be used to define the constants of the machine instruction operand field.

INPUT/OUTPUT MEDIA

- Source input : Punched cards and magnetic disk
- Control data input : Punched cards and magnetic disk
- Control data command : Punched cards
- Execution command : System typewriter keyboard
- Object output : Magnetic disk
- Output lists : Line printer

FUNCTION

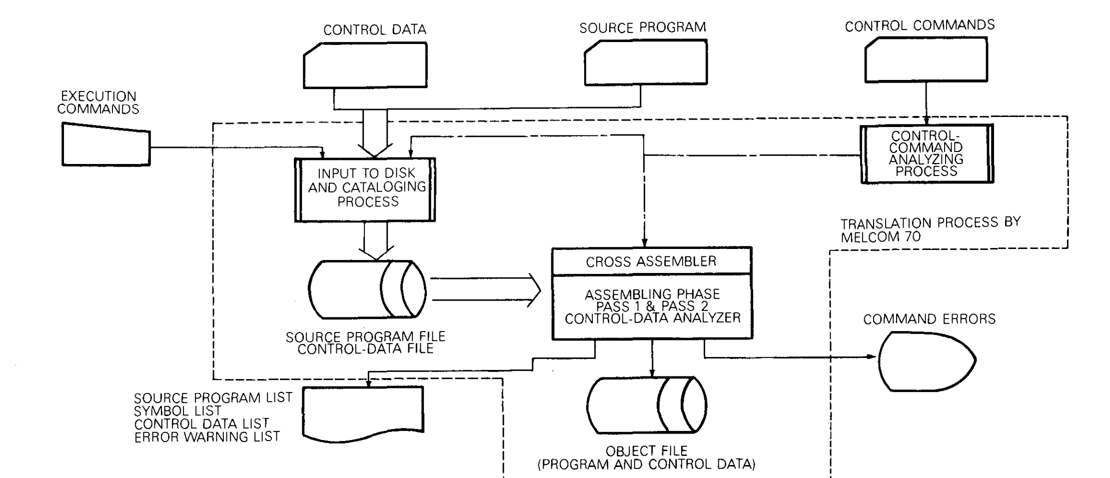
This cross assembler converts source programs written in the MELPS 4 assembly language to machine instruction codes that are filed in disk storage in the form of binary absolute object codes.

The MELPS 4 cross assembler is a 2-pass translator that provides data and control command analysis along with cataloging functions.

Modifying the number of bits in an instruction code and setting mnemonic tables and numeric tables to constants can easily be accomplished by means of the control data. In this way, programming versatility is provided for changing functions, allowing the user free selection in defining the mnemonics of the machine instructions, etc.

The standard version of the MELPS 4 assembly language has 7 assembler control commands (see Table 1). In addition 6 pseudo instructions and 10 system simulator control commands (Table 2) can all be used in the source language program.

PROCESSING SYSTEM



PROGRAM ORDERING INFORMATION

Program name	Ordering number	Program and software manuals included
MELPS 4 cross assembler	GBIAS0001	MELPS 4 Assembler Language Manual MELPS 4 Cross Assembler Manual MELPS 4 Cross Assembler Operating Manual

GBM-SR10-01A<93A0>

14

CROSS ASSEMBLER

This cross assembler facilitates assembly by the use of the control commands shown in Table 1. Basically, it only requires the source program and control commands input by punched cards with control data being utilized only when necessary. All input is stored and filed in disk storage. The control data is processed by the control command analyzing processor, and the symbol table is created in pass 1. This is followed by pass 2, where each instruction is converted to machine language, while control data, labels and the assembly list are printed out as specified by the control commands. On the assembly list, the control commands, sequence numbers, location numbers and addresses are printed out, along with error and warning messages, followed by the ROM page list and the cross-reference list.

OBJECT LANGUAGE

The object file is composed of a name section and a text section.

The name section is filed on sector 0 of the object file and stores overall information such as the total number of instructions in the text section, control data, file name, source program file name, size of a single page and the module name.

The text section contains the data that controlled the conversion of the source program to instruction codes and other related data necessary for execution by the simulator.

ASSEMBLY LANGUAGE

The assembly language that the MELPS 4 cross assembler accepts consists of machine instructions and pseudo instructions.

1. Machine Instructions

There are 68 basic machine instructions. These are converted to their corresponding machine codes and then assembled into an object program. For the mnemonics, instruction codes and their functional descriptions, please refer to the data sheet provided for the M58840-XXXP single-chip 4-bit microcomputer.

2. Pseudo Instructions

Although the pseudo instructions are written in the source program together with machine instructions, they are not converted to instruction codes but are used to control the assembler and the simulator. The instruction codes will be written in the ROM.

The system simulation control instructions are among the pseudo instructions along with assembler-control instructions, numeric symbols defining instructions and list-control instructions. The pseudo instructions are shown in Table 2.

Table 1 Assembler control commands

Command		Format	Function
Execution start		/ / / **RUN**	Starts execution of the cross assembler
Execution end		/ / / **END**	Terminates execution of the cross assembler
Input/output function assignment		/ / / **ASMB4, x, y, z**	Assignment of assembly execution and control data and assembly listings $x=\begin{pmatrix}A\\P\end{pmatrix}$ $y=\begin{pmatrix}L\\N\end{pmatrix}$ $z=\begin{pmatrix}L\\N\end{pmatrix}$ **x** : Assembly control **A** : Assembly needed **P** : Designation of cataloging function **y** : Assembly listing **z** : Control data listing **L** : Listing needed **N** : No listing needed
File assignment control	Control data	/ / / **CDISK, XXXXX**	Assignment of the control file name (max. 6 characters)
	Source program	/ / / **SDISK, XXXXX**	Assignment of the source program file name (max. 6 characters)
	Object	/ / / **BDISK, XXXXX**	Assignment of the object file name (max. 6 characters)
Input/output device assignment		/ / / **INPUT, x, y**	Assignment of input device for the control data and source program $x=\begin{pmatrix}C\\D\end{pmatrix}$ $y=\begin{pmatrix}C\\D\end{pmatrix}$ **x** : Control data input **y** : Source program input **C** : Punched card input **D** : Disk input

Table 2 Pseudo instructions

Classification	Mnemonic	Instruction	Function
Assembler control instructions	TTL	Program title declaration	Declares the program title
	PAGE	Program counter paging	Sets the counter to the top address of the next page
	ORG	Program counter setting	Sets the counter to the top address of the program
	END	End declaration	Declares the end of the program
Symbol value equivalence instruction	EQU	Symbol value setting	Sets a numeral value to the specific numeral symbol
List control instruction	EJE	Page eject declaration	Advances the printout form to the next page during output
System simulator control instructions	SIN	Data input	Reads the input data
	RIN	Mode cancellation	Cancels "✱ ✱" mode input
	SDIS	Display content printout	Prints out the contents of the display
	RDIS	SDIS presetting	Enables execution of the SDIS instruction
	SSC	Step counter selection	Selects the step counters
	RSC	SSC presetting	Enables execution of the SSC instruction
	WSC	Step counter content printout	Prints out the contents of the step counters
	RWSC	WSC presetting	Enables execution of the WSC instruction
	SINT	Terminal input	Starts input from the terminal assigned
	RINT	SINT presetting	Validates execution of the SINT instruction

Note 1. Validation refers to the execution of one command before another to enable its function. For example, to execute the SSC instruction the RSC instruction must be executed first.

3. Language Format

The following format should be used in coding programs in this cross assembler.

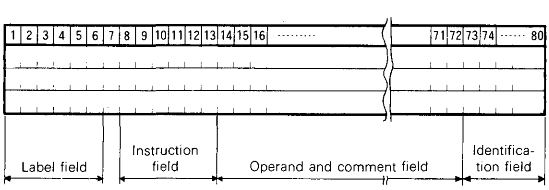
The single-line statement is composed of the label, instruction, operand, comment, and identification fields. The format of the source statement is fixed as indicated in Fig. 1. Although the constant is usually a decimal number, it may be expressed by hexadecimal notation and other nation when defined by pseudo instructions and control data.

An asterisk (*) in the first column of a line indicates that the entire statement is used as a comment field.

The following are valid characters for use in statements:

Alphabetics: A~Z
Numerics: 0~9
Special characters: ; = , ▼ @ $ + − * / ! & ( ) . # % < > ? (space)

Fig. 1 Source statement format



(1) Label field

The value of the program counter at that time is set to the label. The number of characters used for a label is limited to a maximum of 6, and any of the alphanumerics and special characters specified above can be used. However, an asterisk (*) cannot be used in the first column of the label field.

(2) Instruction field

Mnemonic codes are written in this field, left-justified. For pseudo instructions, any of the mnemonics among the assembler-control instructions, numeric symbol definition instructions, list-control instructions, and system simulator control instructions may be used.

(3) Operand field

Parameters of the instruction are specified in this field. This field contains the label, defined symbol, or numerical value. The operand is stated from the 14th column, left-justified.

(4) Comment field

Whenever the operand is followed by more than one space to the end of the statement, the successive columns may be used for comments.

(5) Identification field

The use of this field is optional. Many find it convenient to use this field for a sequential identification card number.

ASSEMBLY LIST FORMAT

A source program prepared and assembled in the format indicated in the preceding paragraph may produce source, symbol table, cross reference, and ROM page list printouts. The format of an assembly list produced as an example is shown in Fig. 2. Please note that pages, locations, and object codes are indicated in hexadecimal notation.

MESSAGE FORMAT

Error and warning messages are printed out on the assemble list. In the case of errors, the message is printed out under the respective statement in the following format:

$ $ $ $ $ $ $ E R R O R   x x x $

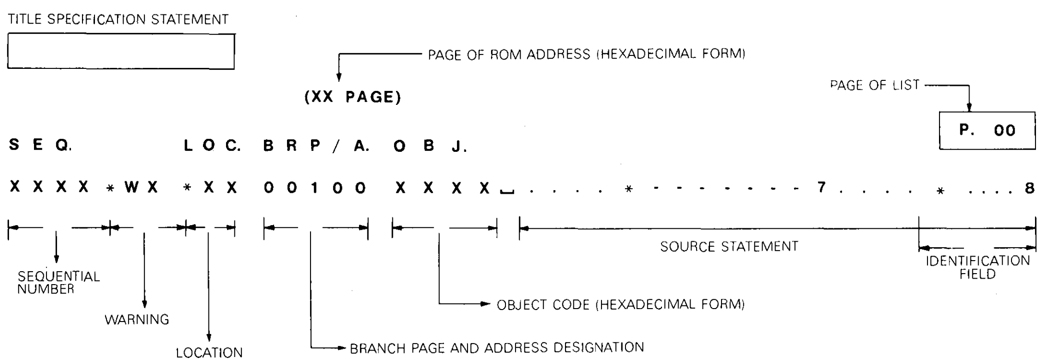
where "**x x x**" indicates the type of error by a numerical code.

In the case of warnings, the following message is printed between SEQ (sequential number) and LOC (location number):

* **Wx** * (where "**x**" indicates the degree of warning)

In addition the total numbers of errors and warnings are printed on the last line of the assembly list. The cross-reference list, however, will not be produced when any errors are indicated.

Fig. 2 Assembly list format



Example of an assembly list

An actual example of an assembly list, for an assembly made with the MELPS 4 cross assembler, is shown in Fig. 3.

Fig. 3 Example of an assembly list

```
EXAMPLE PROGRAM                                                                                   P.  1
                                  ( 00 PAGE )
SEQ.   LOC.BRP/A. OBJ.  ....*....1....*....2....SOURCE STATEMENT....*....5....*....6....*....7....*....8

  1                           TTL    EXAMPLE PROGRAM---------------------------------(1)  EXA00010
  2                           ORG    0,0         --------------------------------(2)  EXA00020
  3                     * FILE DATA EXCHANGE     --------------------------------(3)  EXA00030
  4                     *                                                             EXA00040
  5                     DIGMAX EQU   13          DIGMAX=13 ----------------------(4)  EXA00050
  6    00         04A         LZ     0                                                EXA00060
  7    01 0E/00   100         BM     XCG02       EXCHANGE F0 & F2 ---------------(5)  EXA00070
  8    02 0E/01   101         BM     XCG13       EXCHANGE F1 & F3 ---------------(6)  EXA00008
  9    03 0E/07   107         BM     XCG23       EXCHANGE F2 & F3 ---------------(7)  EXA00009
 10    04         000         NOP                                                     EXA00100
 11                     *                        --------------------------------(8)  EXA00100

 12                           ORG    E,0                                              EXA00110
 13                     * SUBROUTINE FILE EXCHANGE                                    EXA00120
 14                     *                                                             EXA00130
 15                     *     EXCHANGE FILE M(2,X,0-DIGMAX)                           EXA00140
 16                     *                                                             EXA00150
 17    00         0CD   XCG02 LXY    0,DIGMAX    EXCHANGE F0 (0-DIGMAX) & F2(0-DIGMAX)--(9)  EXA00160
 18    01         0DD   XCG13 LXY    1,DIGMAX    EXCHANGE F1 (0-DIGMAX) & F3(0-DIGMAX)--(10) EXA00170
 19                     *                                                             EXA00180
 20    02         066   LBL4  TAM    2                                                EXA00190
 21    03         062         XAM    2                                                EXA00200
 22    04         068         XAMD   0                                                EXA00210
 23*W0*05         102         BM     LBL4        BM IS EQUIVALENT WITH B ON PAGE 14-----(11) EXA00220
 24    06         044         RT                                                      EXA00230
 25                     *                                                             EXA00930
 26    07         0ED   XCG23 LXY    2,13        EXCHANGE F2 (0-DIGMAX) & F3(0-DIGMAX)      EXA01010
 27                     *     COMMON ROUTINE START                                    EXA01020
 28                     *                                                             EXA01030
 29    08         065   LBL5  TAM    1                                                EXA01040
 30    09         061         XAM    1                                                EXA01050
 31    0A         068         XAMD   0                                                EXA01060
 32*W0*08         108         BM     LBL5                                             EXA01070
 33    0C         044         RT                                                      EXA01080
 34                           END                                                     EXA01100
```

① The program name is declared as "EXAMPLE PROGRAM"

② It shows that the start of the program was set to page 0 address 0 by means of the program counter setting instruction.

③ An asterisk (*) in the first column indicates that the entire statement is a comment.

④ Numeric value 13 (decimal number) is assigned to the symbol DIGMAX by means of the symbol value equivalence instruction.

⑤ The label XCG02 is assigned by means of the BM instruction during the assembly process, and calls the subroutine starting at page 14 address 00.

⑥ The label XCG13 is assigned by means of the BM instruction during the assembly process, and calls the subroutine starting at page 14 address 01.

⑦ The label XCG23 is assigned by means of the BM instruction during the assembly process, and calls the subroutine starting at page 14 address 07.

⑧ This whole statement line is used as a comment field.

⑨ The numerical value 0 is loaded in register X of the data pointer and 13 (decimal number) in register Y by means of the LXY instruction. As written, the results of this LXY instruction are nullified by the results of the following LXY instruction.

⑩ The numerical value 1 is loaded in register X of the data pointer and 13 (decimal number) in register Y by means of the LXY instruction.

⑪ The BM instruction in this case assigns the branch address of the label LBL4 to address 02 of page 14.

SIMULATOR

DESCRIPTION

The MELPS 4 simulator software has been prepared for facilitating program debugging for application programs suitable to equipment using single-chip 4-bit microcomputers. It also allows a significant saving of program-development time.

With this simulator, each instruction of the microcomputer is executed on a host computer just as though the program were being executed on an actual microcomputer system. This allows confirmation that the operations and sequences of a program are correct before the microcomputer system is built. Various control commands such as traces and halt tables are available for use during program development. The program, which was assembled and stored in disk storage by the MELPS 4 cross assembler, can then use this simulator to simulate its execution. The results of the simulation are printed out along with other helpful information for verification and debugging of a program under development.

FEATURES

- Trace and halt tables
- 20 control commands
- 10 control instructions that can be used along with pseudo instructions during source-program preparation
- Selective printout of input data for verification
- Selection of display digits (1~12 digits)
- Indication of each register, I/O port and memory file, etc

INPUT/OUTPUT MEDIA

- Object input: Cartridge disk storage
- Control commands: Punched card/keyboard and input data
- Execution commands: Keyboard input
- Trace dump: Input/output of the table by paper tape
- Simulation: Output of the result through the line printer or the system typewriter
- Messages: System typewriter

APPLICATIONS

- In conjunction with the MELPS 4 cross assembler as a tool for developing application programs for 4-bit microcomputers
- Especially useful for debugging programs prepared for the M58840-XXXP

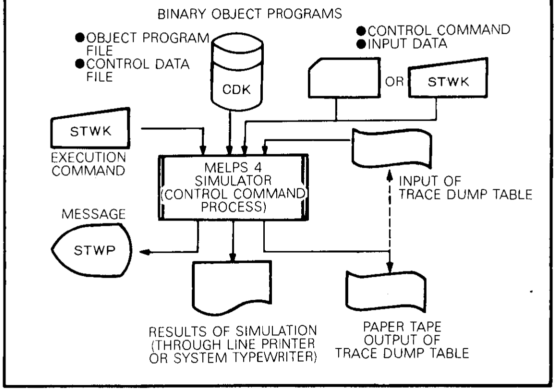
FUNCTIONS

Various control commands are provided by the MELPS 4 simulator to help determine if the program is operating properly according to original specifications. These control commands can set operating conditions and halt program process, while indicating the system status, CPU state, and memory contents.

This simulator has 20 control commands, including trace dump, trace and halt tables, clear, printout of tables, set registers or stack pointers and carry flags, selective printout of input data, input/output of paper tape, etc. that may be used to facilitate debugging.

It also has 10 simulator-control pseudo instructions that may be assigned during the preparation of the source program.

MELPS 4 SIMULATOR PROCESSING SYSTEM



Simulator

Binary object codes stored in the disk file (BDISK), generated by the MELPS 4 cross assembler, are processed in this program according to the conditions given by the control commands, and the result can be selectively printed out on the line printer or the system typewriter. Input and control data can be input through the card reader or the keyboard depending on the mode selected.

Input data format

Input data format of the simulator is shown below:

$XXYY\sqcup n_1 n_1 n_1 n_1 n_1 \sqcup n_2 n_2 n_2 n_2 n_2$

where, "XX" (or *, $) indicates the input symbol, and "YY" (or ••, **• ON, OFF) the input mode symbol.
"$n_1 n_1 n_1 n_1 n_1$" the analog value (digital) under on-state.
"$n_2 n_2 n_2 n_2 n_2$" the analog value (digital) under off-state.

Control command input format

Normally, the control commands are expressed in the following format, but its relation with control commands is described in Table 1.

$///XX\sqcup$(parameter)

where, "XX" indicates the mnemonic of the control command.

There are two input modes: type-in mode or batch mode. But the simulator start ST command should be entered from the keyboard.

PROGRAM ORDERING INFORMATION

Program name	Ordering number	Software manuals included
MELPS 4 simulator	GB1SM 0001	MELPS 4 Simulator Manual MELPS 4 Simulator Operating Manual

GBM-S10-01A<93AO>

Table 1 Simulator control commands

Functional classification	Item	Control commands: Action	Control commands: Mnemonic	Functions
Simulator control commands	Start condition setup	Starts simulation	ST	Designates the control command input device and the simulation result output device and assigns control data output.
	Load program	Loads the object program	LO	Loads the absolute object program
	Command input reassignment	Reassigns the control command input	CM	Changes the command input device to another device.
	Finish simulation	Stops simulation	FN	Terminates the program execution, and control is returned to the monitor.
Execution control commands	Trace	Trace region assignment started	TS	Assigns trace regions where the contents of the program counter, registers, and memory file will be printed out while being executed.
		Trace region assignment discontinued	TD	Discontinues trace region assignment.
		Printout of the trace table	PT	Prints out the trace table
	Halt	Halt-point assignment started	HS	Assigns halt points by page number, address and times of execution.
		Halt-point assignment discontinued	HD	Discontinues halt-point assignment
		Printout of the halt-point table.	PH	Prints out the halt-point table
	Data setup	Initialization of the program counter, registers, memory file, etc.	MM	Sets the initial data to the program counter, registers, I/O ports, memory file, etc.
		Resets of the program counter, registers, memory file, etc.	CL	Resets the program counter, registers, I/O ports, memory file, etc.
	Data printout	Printout of the data in the program counter, register, memory file, etc.	DM	Dumps the contents of the program counter, registers, I/O ports, memory file, etc.
	Dump table	Dumps the trace and halt-point tables.	DT	Outputs the contents of the trace and halt-point tables on paper tape.
		Reads the trace and halt-point tables	RT	Inputs the data of the trace and halt-point tables from paper tape.
	Data input	Printout of the input data	PK	Prints out the contents of the periodical input data while the program is in execution.
		Release of input data printout	NK	Releases printout of the contents of the periodical input data while the program is in execution.
		Device assignment for data input	DV	Assigns the input device for the input data
	Program start	Continuance of program execution	RN	Starts to execute the program without changing the contents of the program counter, registers, I/O ports, memory file, etc.
		Program execution	RS	Starts to execute the program after initializing the contents of the program counter, registers, I/O ports, memory file, etc.

APPLICATION EXAMPLE

Once the command ST and its parameter are typed in through the system typewriter keyboard, successive commands may be entered through punched cards or the system typewriter keyboard. The command input device may be changed at any time by using the CM command.

Simulation is started on the object file in the disk storage that was stored there, after assembling, by the MELPS 4 cross assembler. When the MELCOM 70 is used, the simulator program should be called by the command EXEC SIML4 to start simulating operation.

The following commands must be assigned when tracing and executing the simulated program. Assignment of the input and printer devices, along with selection of the desired list printout, is entered by the ST command in the format ST, X, Y, Z. Here X represents the input device (S for the system typewriter and C for the card reader). Y represents the output device on which the simulation result is printed out (L for the line printer device and S for the system typewriter and W for both). Z represents the need for the control data list output (L to output the control data list and N for omitting output).

The stored object program (BDISK file) is loaded by the simulator with the LO command in the format: LO file name.

The CL and MM commands should be used when initialization is required. When the program counter, registers, I/O ports, and memory file are to be cancelled, the command CL may be used. The format of the MM command is:

MM XXXX = nnnn

It is used in setting initial values. XXXX represents the symbol or numerical figure by which the program counter, registers, I/O ports or memory files are designated. And nnnn represents a parameter to be given.

Entry of the HS command:

HS pp: aa, nnnn

will make the machine halt at address aa of page pp after that instruction has been executed nnnn times.

Entry of the TS command:

TS p_1p_1 : a_1a_1, p_2p_2: a_2a_2 [,R] [,M] sets flags to make a trace effective from address a_1a_1 of page p_1p_1 to address a_2a_2 of page p_2p_2. R designates the output of the contents of the registers and M the memory file.

When the DM command is executed, the contents of each register and memory file at the time are printed out.

Program execution is commenced with the RS or RN command and continues until a location is reached that has been designated by the parameter of an HS command to print out its result. The RS command designates a start after cancelling the contents of the program counter, registers, I/O ports and memory file.

The assignment of the trace region is discontinued with the TD command, and the halt-point assignment with the HD command. The trace table is printed out with the PT command, and halt-point table with the command PH, whenever required. Paper-tape dump and input of the trace and halt-point table is assigned with the DT and RT commands. To set up for input data, there are the PK command, which prints out the contents during the execution of selected data input, and the NK command, which discontinues printing. For the assignment of the input device, the DV command is provided.

Besides the above commands, 10 simulator-control pseudo instructions are used during source-program preparation and during simulation.

A typical example of the use of the MELPS 4 simulator control commands is shown in Table 2, and the results of a simulation example of the assembled program of Fig. 1, is shown in Fig. 2.

Table 2 Example of the use of simulator control commands

A typical example of control commands	Function of the control command and its parameter(s)
ST S, L, N	To start simulation, the I/O are assigned and control data is omitted or output. In this example, command input is assigned to the system typewriter, printout is assigned to the line printer, and the list of the control data is omitted.
LO BFILE	The file stored in the disk (BDISK) whose file name is BFILE is loaded.
CL	The program counter, registers, I/O ports and file memory are cleared and set to intial values.
HS 0 : 5 , 2	This assigns a halt-point. In this example it will halt after the second execution of the instruction in address 5 of page 0.
TS 0 : 1 ,E : F ,R ,	This command designates a trace from address 1 of page 0 to address F of page E, and orders display of the contents of the program counter, registers, and I/O ports after completing tracing.
PT	This command prints out the trace-dump table. Assignments made by TS commands can be verified by this command.
PH	This command prints out the halt-point table. Assignments made by HS commands can be verified by this command.
MM 0, 1=2	The contents of column 1 of the memory file F_0 are set to 2.
DM	The contents of the program counter, registers, I/O ports and memory file at the time this command is executed are printed out.
RN	Program is started without changing the contents of the program counter, registers, I/O ports and memory file

Fig. 1 Example of assembled program

```
EXAMPLE PROGRAM                                                                                P. 1
                                    ( 00 PAGE )
SEQ.   LOC.BRP/A. OBJ. ....*....1....*....2....SOURCE STATEMENT....*....5....*....6....*....7....*....8

 1                            TTL    EXAMPLE PROGRAM-------------------------------------------①  EXA00010
 2                            ORG    0,0             ------------------------------------------②  EXA00020
 3                     * FILE DATA EXCHANGE          ------------------------------------------③  EXA00030
 4                     *                                                                          EXA00040
 5                     DIGMAX EQU    13              DIGMAX=13 --------------------------------④  EXA00050
 6     00         04A         LZ     0                                                            EXA00060
 7     01 0E/00   100         BM     XCG02           EXCHANGE F0 & F2 -------------------------⑤  EXA00070
 8     02 0E/01   101         BM     XCG13           EXCHANGE F1 & F3 -------------------------⑥  EXA00008
 9     03 0E/07   107         BM     XCG23           EXCHANGE F2 & F3 -------------------------⑦  EXA00009
10     04         000         NOP
11                     *                             ------------------------------------------⑧  EXA00100

12                            ORG    E,0                                                          EXA00110
13                     * SUBROUTINE FILE EXCHANGE                                                 EXA00120
14                     *                                                                          EXA00130
15                     *      EXCHANGE FILE M( 2,X,0-DIGMAX)                                      EXA00140
16                     *                                                                          EXA00150
17     00         0CD  XCG02  LXY    0,DIGMAX        EXCHANGE F0 (0-DIGMAX) & F2(0-DIGMAX)--⑨  EXA00160
18     01         0DD  XCG13  LXY    1,DIGMAX        EXCHANGE F1 (0-DIGMAX) & F3(0-DIGMAX)--⑩  EXA00170
19                     *                                                                          EXA00180
20     02         066  LBL4   TAM    2                                                            EXA00190
21     03         062         XAM    2                                                            EXA00200
22     04         068         XAMD   0                                                            EXA00210
23*W0*05          102         BM     LBL4            BM IS EQUIVALENT WITH B ON PAGE 14-----⑪  EXA00220
24     06         044         RT                                                                  EXA00230
25                     *                                                                          EXA00930
26     07         0ED  XCG23  LXY    2,13            EXCHANGE F2 (0-DIGMAX) & F3(0-DIGMAX)       EXA01010
27                     *     COMMON ROUTINE START                                                 EXA01020
28                     *                                                                          EXA01030
29     08         065  LBL5   TAM    1                                                            EXA01040
30     09         061         XAM    1                                                            EXA01050
31     0A         068         XAMD   0                                                            EXA01060
32*W0*08          108         BM     LBL5                                                         EXA01070
33     0C         044         RT                                                                  EXA01080
34                            END                                                                 EXA01100
```

14

Fig. 2 Example of simulation results

```
** START SIMULATOR OF MELPS 4 **


///LO OFILE                    ------------------------------------------------------ ①


 EXAMPLE PROGRAM               ------------------------------------------------------ ②


CONTROL DATA FILE=CFILE
SOURCE FILE=SFILE
OBJECT FILE=OFILE


///CL                          ------------------------------------------------------ ③

///MM PROG=0:0                 ------------------------------------------------------ ④

///TS 0:0,0:4,R,M              ------------------------------------------------------ ⑤

///HS 0:4,1                    ------------------------------------------------------ ⑥

///TD                          ------------------------------------------------------ ⑦

///TS 0:0,0:5,R,M              ------------------------------------------------------ ⑧

///PT                          ------------------------------------------------------ ⑨

          *** TRACE DUMP TABLE ***

     NO. 1 --- 00:00   00:05   R,M

///DM                          ------------------------------------------------------ ⑩

     *** DUMP OF MEMORY ***

PC = 00:00  INST. 00:00  = TYA   SK0 = 00:00 SK1 = 00:00 SK2 = 00:00 -------- ⑪

CPS = 0  ACC = 0  CY1 = 0  DP1 Z,X,Y = 0,0,0 }
                  CY2 = 0  DP2 Z,X,Y = 0,0,0 } ------------------------------ ⑫

PORT. J 0-E  = 000000000000000  D 0-A  = 00000000000 }
      E 0-7  = 0000 0000        S 0-7  = 0000 0000   } ---------------------- ⑬

INT = 0  INTEF = 0  INTHL = H           ------------------------------------- ⑭

REG. B = 0  H = 0000  L = 0000  C = 0   ------------------------------------- ⑮

              F  E  D  C  B  A  9  8  7  6  5  4  3  2  1  0

        F0 =  5  4  3  D  0  0  0  0  0  2  0  0  0  0  0  1
        F1 =  A  9  8  0  0  0  0  0  0  7  0  0  0  0  0  6
        F2 =  F  E  D  0  0  0  0  0  0  C  0  0  0  0  0  B   }-------------- ⑯
        F3 =  8  7  6  0  0  0  0  0  0  5  0  0  0  0  0  4
        F4 =  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
        F5 =  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
        F6 =  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
        F7 =  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0

///RN                          ------------------------------------------------------ ⑰

///DM                          ------------------------------------------------------ ⑱

     *** DUMP OF MEMORY ***

PC = 00:05  INST. 00:04  = NOP   SK0 = 00:00 SK1 = 00:00 SK2 = 00:00

CPS = 0  ACC = 1  CY1 = 0  DP1 Z,X,Y = 0,2,F
                  CY2 = 0  DP2 Z,X,Y = 0,0,0

PORT. J 0-E  = 000000000000000  D 0-A  = 00000000000
      E 0-7  = 0000 0000        S 0-7  = 0000 0000

INT = 0  INTEF = 0  INTHL = H

REG. B = 0  H = 0000  L = 0000  C = 0

              F  E  D  C  B  A  9  8  7  6  5  4  3  2  1  0

        F0 =  5  4  D  0  0  0  0  0  0  C  0  0  0  0  0  B
        F1 =  A  9  6  0  0  0  0  0  0  5  0  0  0  0  0  4
        F2 =  F  E  8  0  0  0  0  0  0  7  0  0  0  0  0  6   }-------------- ⑲
        F3 =  8  7  3  0  0  0  0  0  0  2  0  0  0  0  0  1
        F4 =  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
        F5 =  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
        F6 =  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
        F7 =  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0

///FN                          ------------------------------------------------------ ⑳

** END SIMULATOR OF MELPS 4 **
```

① The file stored in the BDISK whose file name is OFILE is loaded.

② The program title that was declared at the time of source program preparation is printed out.

③ The contents of the program counter, stack pointer, registers, I/O ports and memory file are cancelled and set to initial conditions.

④ The contents of address 0 of page 0 of the object program is printed out on the system typewriter by means of the data setup command in order to allow alteration in the program code and to identify correct loading of the program.

⑤ Tracing is directed from address 0 of page 0 to address 4 of page 0 and the contents of the registers are displayed.

⑥ This assigns a halt after once executing the program in address 4 of page 0.

⑦ Tracing designation in step ⑤ is discontinued by releasing the trace-region assignment.

⑧ A new trace region is assigned. In this example, tracing is directed from address 0 of page 0 to address 5 of page 0, and the contents of the registers and memory file are to be printed out.

⑨ The trace table is printed out in order to confirm that the trace has been registered correctly.

⑩ This prints out the contents, in their initial states, of the program counter, registers, stack pointer, I/O ports and memory file.

⑪ The contents of the program counter, stack pointer, etc., are printed out.

⑫ The contents of CPS, (either one of a pair of the data pointers or the carry is selected), ACC (accumulator), CY1 and CY2 (carry), DP1 and DP2 (data pointer), Z (file assignment) and Y (digit designation in the file) are printed out.

⑬ This indicates each bit in the contents of the ports D and S and the registers J and E.

⑭ This indicates the contents of the interrupt request INT, interrupt acknowledge flag INTEF and the condition given for the INTHL.

⑮ This indicates the contents of the registers, B, H, L and C, respectively.

⑯ The contents of the memory file before the execution of the program are printed.

⑰ The program execution is started, and trace is carried out in accordance with the assignment given at step ⑧ and continues to indicate the contents of registers and memory file and then to halt at the halt point designated in step ⑥.

⑱ The DM command is entered to print the contents of registers, etc., at the time the execution is terminated.

⑲ This shows that each file in F_0 and F_2 is exchanged with F_1 and F_3 after executing the program shown in Fig. 1.

⑳ The simulation is now terminated, and control has returned to the monitor.

PAPER-TAPE GENERATION PROGRAM FOR PROM WRITERS

DESCRIPTION

MELPS 4 PROM writer paper tape generation programs are used to convert the absolute binary object program generated by the MELPS 4 cross assembler into another format that can be used in a PROM writer. The program is output on paper tape in the new format.

With this program, a binary object program can easily be converted to hexadecimal object format that can be programmed directly into a PROM. It can produce paper tapes that meet the requirements for various types of PROMs and PROM writers because of its functional versatility.

FEATURES

- Outputs the binary object program in the disk storage to paper tape in hexadecimal format
- Paper tape output can be partitioned with a simple control command
- May be used in conjunction with the MELPS 4 cross assembler
- Execution computer: MELCOM 70 Minicomputer (memory capacity more than 16K words, monitor BDOS)
- Programming language: FORTRAN IV (parts are written in assembly language)

INPUT/OUTPUT MEDIA

- Input: Cartridge disk storage
- Output: Paper tape (ASCII code, even parity)
- Control command input: Through the keyboard of the system typewriter
- Messages: System typewriter printout

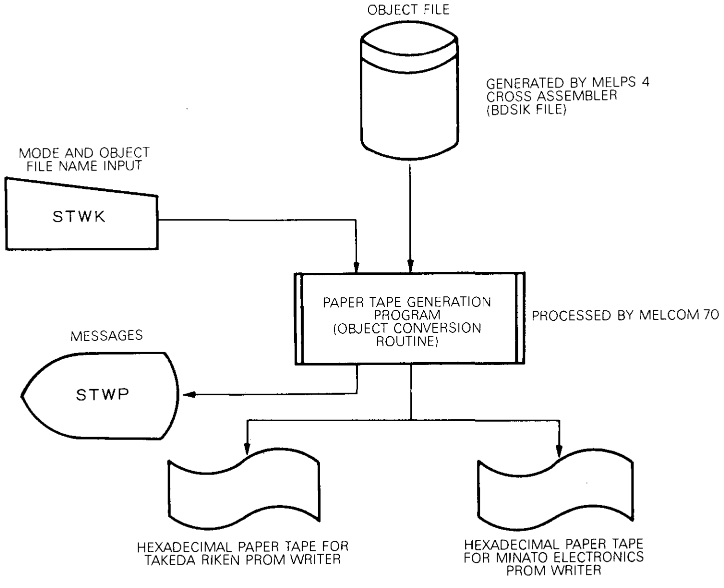
APPLICATIONS

- For preparing programs for 1K words × 8-bit EPROMs (M5L2708S), etc., which are to be programmed by PROM writers supplied by Takeda Riken or Minato Electronics.

FUNCTION

This program is used for converting the absolute binary object format programs generated by the MELPS 4 cross assembler to hexadecimal object format compatible with the PROM writers manufactured by Takeda Riken (T310) and Minato Electronics (model 1830 and 1802). The paper-tape output is partioned in accordance with PROM capacity (number of bytes).

PROCESSING SYSTEM



14

PROGRAM ORDERING INFORMATION

Program name	Ordering number	Program and software manuals included
MELPS 4 paper tape generation program for PROM writer	GBISP0001	MELPS 4 paper tape generation program for PROM writer manual MELPS 4 paper tape generation program for operating manual GBM-SR10-01A<93A0>

PROGRAM PROCESSING

The program has conversion routines for Takeda Riken's and Minato Electronics' PROM writer. Select T_1 mode (for Takeda Riken's PROM writer) or M_1 mode (for Minato Electronics' PROM writer) through the system typewriter keyboard. Then the object program is converted to paper tape compatible with the selected PROM writer. When a (BDISK file) file name is called, a paper tape is output for the PROM writer. When a number of programs are to be converted from the same file, successive calls can be made until all the programs are converted. Termination of the job is directed with the E command, and control is then returned to the monitor.

The object file consists of name and text segments. The data to be converted is contained in the text segment. Instruction codes stored after sector 1 of the disk that correspond to machine instructions are converted to hexadecimal codes and output to paper tape.

Example of Hexadecimal Paper Tape Format

This program can generate paper tapes for Takeda Riken's PROM writer and Minato Electronics' PROM writer. Examples of both formats are shown in Figs. 1 and 2.

Example of Object Conversion

The program at present can output 1K-word units of paper tape up to a total of 4K words. An example is shown in Fig. 3.

Error Processing

When an error is encountered during object conversion, a message will be printed out in the following format:

$ $ $ $ $ $ $ X X X $

where, XXX indicates the error code.

Fig. 3 Example of object conversion

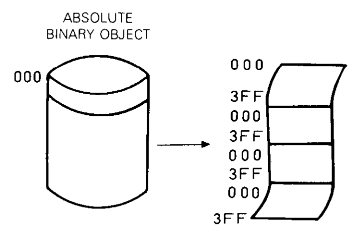


Fig. 1 Example of hexadecimal paper tape format of Takeda Riken

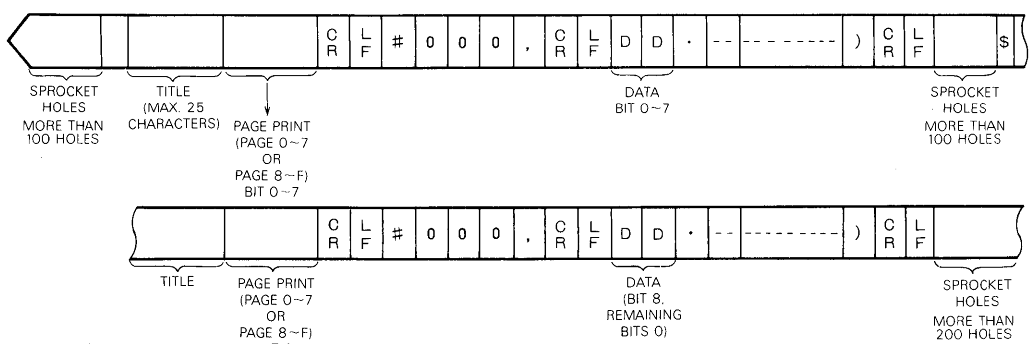
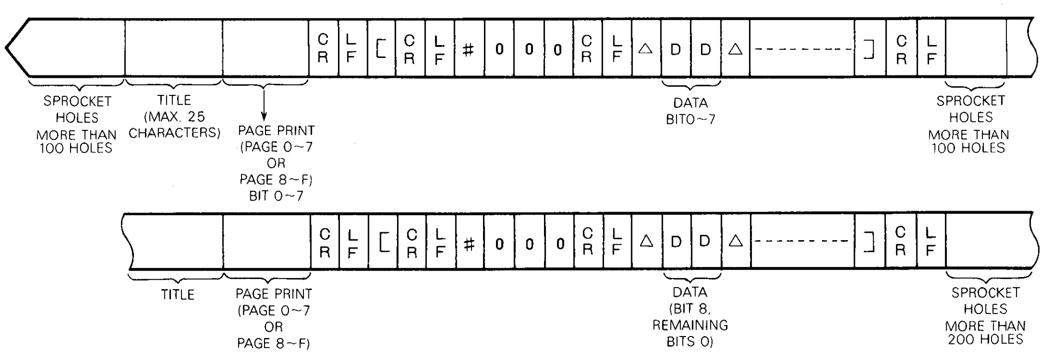


Fig. 2 Example of hexadecimal paper tape format of Minato Electronics



CROSS ASSEMBLER

DESCRIPTION

The MELPS 41 cross assembler has been prepared for the development of application programs suitable to equipment using the M58494-XXXP single-chip 4-bit CMOS micro-computer.

This cross assembler allows coding in free formats for improved programming efficiency and permits the use of various input media. It also provides program versatility for changing instruction codes and functions thanks to the control commands and control data employed.

FEATURES

Of the Cross Assembler

- Free-format coding.
- Various source-input media available
- Instruction codes and functions easily changed
- Catalogues the control data in disk storage
- Constants can also be expressed in non-decimal formats
- Numerical formula in the operand field can be processed
- Printouts available from the tables and cross-reference lists
- Execution computer: MELCOM 70 (memory capacity more than 24K-words, monitor BDOS)
- Implementation language: FORTRAN IV (parts are written in assembler language)

Of the Assembly Language

- 9 pseudo instructions
- 1 macro instruction
- 93 machine instructions
- The constants of the machine-instruction operand field can be defined using decimal numbers.

INPUT/OUTPUT MEDIA

- Source input: Punched cards, punched tapes, magnetic disk, and magnetic tape
- Control-data input: Punched cards, punched tapes and magnetic disk
- Control-data command: Punched cards and system-typewriter keyboard
- Object output: Magnetic disk and punched tapes
- Output lists: Line printer and system type-writer

FUNCTION

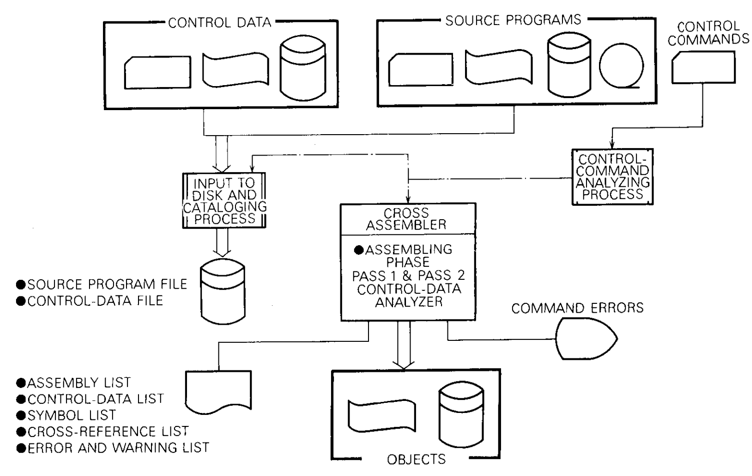
This cross assembler converts source programs written in the MELPS 41 assembly language to machine instruction codes, which are filed in disk storage in the form of binary absolute object codes.

The MELPS 41 cross assembler is a 2-pass translator that provides data and control command analysis along with cataloging functions.

Modifying the number of bits in an instruction code and setting mnemonic tables and numeric tables to constants can easily be accomplished by means of the control data. In this way, programming versatility is provided for changing functions, allowing the user free selection in defining the mnemonics of the machine instructions, etc. Codes corresponding to the MELPS 41 mnemonics are displayed in a 10-bit form.

The MELPS 41 assembler language has 9 assembler

THE PROCESSING SYSTEM



PROGRAM ORDERING INFORMATION

Program name	Ordering number	Program and software manuals included
MELPS 41 cross assembler	GB1AS0003	MELPS 41 Assembler Language Manual MELPS 41 Cross Assembler Manual MELPS 41 Cross Assembler Operating Manual

control commands listed in Table 1 and 9 pseudo instructions (see Table 2).

CROSS ASSEMBLER

This cross assembler facilitates assembly by using the commands listed in Table 1. The source program and control data can be input by punched cards, punched tapes, magnetic tapes and magnetic disks. The control data can also be input by using these types of media. It is very convenient to prepare standard control data and store it in the magnetic disk if it rarely needs changes. The control data is processed by the control-command analyzing processor, and the symbol table is created in pass 1. This is followed by pass 2, where each instruction is converted to machine language, while control data, labels and assembly lists are printed out as specified by the control commands. In this case, the object codes in the assembly list are displayed in hexadecimal form. The control commands, sequence numbers, location numbers for all pages, locations of the pages to be jumped to, and source statements are printed out. In addition, error and warning messages are displayed, followed by the output of ROM-page and cross-reference lists.

OBJECT LANGUAGE

The disk object file is composed of a name section and a text section. In the case of punched tapes, the file consists of a name section, text section and an end-of-tape section.

The name section of a disk object file is filed on sector 0, and stores information such as the total number of instructions in the text section and control data.

The text section is filed from sector 1, and contains the data that controlled the conversion of the source program into instruction codes and other related data necessary for execution by the simulator.

ASSEMBLEY LANGUAGE

The assembly language that the MELPS 41 cross assembler accepts consists of machine instructios, pseudo instructions, and macro instructions.

1. Machine Instructions

There are 93 basic machine instructions. These are converted to their corresponding machine codes and then assembled into an object program. For the mnemonics, instruction codes and their functional descriptions, please refer to the machine-instruction list of the M58494-XXXP.

2. Pseudo Instructions

Although the pseudo instructions are written in the source program together with machine instructions, they are not converted to instruction codes but are used to control the assembler. The instruction codes will be written in the ROM.

The pseudo instructions include assembler-control numeric-symbol defining, list-control, and memory-address setting instructions.

3. Macro Instructions

These instructions give one-word expressions for combined used of several machine instructions. When internal or external memory-address setting is to be carried out by using 'LXx,' 'LYy,' and 'LZz' RAM address instructions, for instance, the macro instruction (LZXY symbol) can be used instead.

Table 1 Assembler control commands

Command		Format	Function
Input/output function assignment		**///ASM41, X, Y, U, Z**	Assignment of assembly execution, object output, and control-data and assembly listings X=(A/P) X: Designation of assembly execution A: assembly needed P: Cataloging function U=(O/N) U: Designation of object output O: Object output needed Y=(N/L) Y: Designation of assembly listing L: Listing needed N: No listing needed Z=(L/N) Z: Designation of control-data listing
Input-device assignment control		**///INPUT, X, Y, Z**	Assignment of input devices for the control data and source program and of magnetic-tape codes. X=(C/D/P/N) X: Designation of control-data input C: Card reader D: Disk P: Punched-tape reader M: Magnetic tape N: No input Y=(C/D/P/M) Y: Designation of source-program input Z=(A/E) Z: Code assignment E: EBCDIC code A: ASCII code
Output-device assignment control		**///OUTPUT, X, Y**	Assignment of object-output device and character selection for the single output-list line X=(D/P) X: Designation of object-output device D: Disk P: Punched tape Y=(C/Blank) Y: Designation of the no. of characters in a line C: 80 characters Blank: 120 characters
File assignment control	Control date	**///CDISK, XXXXXX**	Assignment of the control-data file name (max. 6 characters)
	Program	**///SDISK, XXXXXX**	Assignment of the source-program file name (max. 6 characters)
	Object	**///BDISK, XXXXXX**	Assignment of the oject file name (max. 6 characters)
Date assignment control		**///CDATE, YY, MM, DD**	Assignment of the year, month and day YY: Year (2 digit) MM: Month (2 digit) DD: Day (2 digit)
Execution-start control		**///RUN**	Starts execution of the cross assembler
Execution-end control		**///END**	Terminates execution of the cross assembler

Table 2 Pseudo instructions

Classification	Mnemonic	Instruction	Function
Assembler-control instructions	TTL	Program title declaration	Declares the program title
	ORG	Program counter setting	Sets the counter to the top address of the following program
	PAGE	Program counter paging	Sets the counter to the top address of the next page
	PAUSE	Assembly pausing	Stops the assembly for a short time (effective only for pass 1 execution)
	END	End declaration	Declares the end of the program
Symbol value equivalence instruction	EQU	Symbol value setting	Sets a predetermined value to a specific numeral symbol
List-control instruction	EJE	Page eject declaration	Advances the printout form to the next page during output
Memory address setting	INTM	Internal-memory-address setting	Sets the internal-memory address to the specified symbol
	EXTM	External-memory-address setting	Sets the external-memory address to the specified symbol

Table 3 Macro instructions

Instruction	Function
LZXY $\ell\pm n$[1]	(1) When the ℓ is set by the INTM instruction, expansion is made into LXx and LYy instructions (2) When the ℓ is set by the EXTM instruction, expansion is made into LZz, LXx and LYy instructions

Note 1 : ℓ is specified by the INTM or EXTM instruction; symbol n is hexadecimal and $0 \leqq n \leqq 4095$

4. Language Format

The following format should be used in coding programs in this cross assembler.

The single-line statement of the source program is composed of the label, instruction, operand, comment, and identification fields. Format of the source statement is free, as indicated in Fig. 1. Although the constant is usually a decimal number, it may be expressed by hexadecimal notation when defined by pseudo instructions and control data.

The following are valid characters for use in statements.

Alphabetics: A~Z
Numerics: 0~9
Special characters: ; = , ▼ @ $ + − * / ! & ( ) . # % < > ? (blank)

(1) Label field

The value of the program counter at that time is set to the label. Any of the alphanumerics and special characters specified above can be used. The character : (colon) is placed at the rear end of the label field. However, an asterisk (*) cannot be used in the first column of the label field.

(2) Instruction field

Mnemonic codes are written in this field. In addition to the machine instructions, use can be made of pseudo instructions such as the assembler-control, numeric-symbol definition, list-control, and memory-address setting instructions.

(3) Operand field

Parameters of the instruction are specified in this field. The field contains the label, defined symbol, or numerical value. It is usually necessary to leave a blank of one character or more behind the instruction.

(4) Comment field

This field is used for writing notes for the statement and is not converted to an object in the process of changing the source statement into its corresponding object.

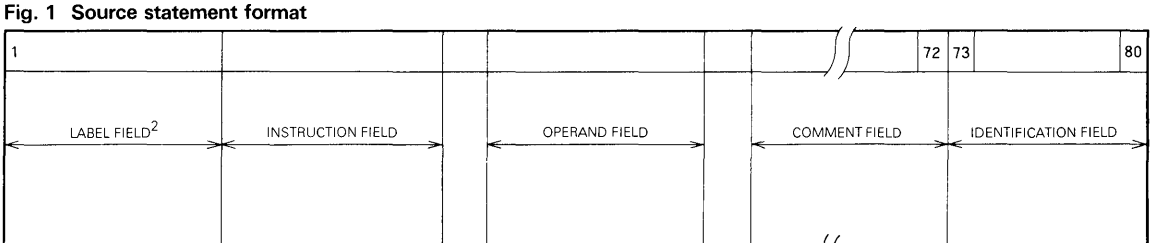
Writing an asterisk(*) in the first column of the source statement enables the whole statement to be used as a comment.

Whenever the instruction or operand field is followed by more than one space, the successive characters may be regarded as comments.

(5) Identification field

The use of this field is optional. Many operators find it convenient to use it for the sequential identification card number.

Fig. 1 Source statement format



Note 2 : A colon (:) is placed behind the label.

14

ASSEMBLY LIST FORMAT

A source program coded and assembled in the format indicated in the preceding paragraph may produce assembly-list, symbol-table-list, cross-reference-list, and ROM-page-list printouts. The format of an assembly list produced as an example is shown in Fig. 2. Please note that pages, addresses, locations, and object codes are indicated in hexadecimal notation.

MESSAGE FORMAT

Error and warning messages are printed out on the assembly list. In the case of errors, the message is printed out under the respective statement in the following format:

$$$$$$$ERROR␣xxx␣$␣ (Error message)

where 'xxx' indicates the type of error by a numerical code. The total number of errors is printed on the last line of the assembly list. The cross-reference list, however, will not be produced when any error is indicated.

Fig. 2 Example of an assembly list

```
MELPS 41 EXAMPLE PROGRAM                                     P. 1
                                      ( 00 PAGE )
SEQ.  LOC. BRP/A. OBJ. ....*....1....*....2....SOURCE STATEMENT....*....5....*....6....*....7....*....8
  1                              TTL   MELPS 41 EXAMPLE PROGRAM..............................① EXA00000
  2                       *                  ........................................................② EXA00010
  3                       A:     EQU   10         SYMBOL 'A' IS EQUAL TO 10......................③ EXA00020
  4                       REG1:  EXTM  200        Z:X:Y=2:0:0 ; TOP ADDRESS OF REGISTER NO.1....④ EXA00030
  5                       REG2:  INTM  10         X:Y=1:0     ; TOP ADDRESS OF REGISTER NO.2....⑤ EXA00040
  6                       *                                                                       EXA00050
  7                              ORG   0,10                   ...................................⑥ EXA00055
  8    10            19A         LA    A                      ...................................⑦ EXA00060
  9    11            036         SMR1             RESET BUS FLOUTING MODE SET(MR1)                EXA00070
 10    12            1C1         LP    1                                                          EXA00080
 11    13            1A2         LZXY  REG1                   ...................................⑧ EXA00090
       14            180                                      ...................................⑨
       15            180
 12    16    01/03   383         BM    INTEX      REGISTER NO.1 SAVE OUT EXTERNAL MEMORY.........⑩ EXA00100
 13    17            1B1         LZXY  REG2                   ...................................⑪ EXA00110
       18            180
 14    19    01/00   380         BM    EXTIN      REGISTER NO.2 RESTORE IN INTERNAL MEMORY.......⑫ EXA00120
 15    1A            000         NOP                                                              EXA00130
 16                       *                                                                       EXA00140
 17                              PAGE             ;; = ORG 1,0 = ;;..............................⑬ EXA00150
 18                       * ;; MEMORY DATA TRANSFER SUBROUTINE  ;;                                EXA00160
 19                       *      NOTE. MEMORY FROM EXTERNAL TO INTERNAL                           EXA00170
 20    00            0FC  EXTIN: TSMI             (Y)=(Y)+1 , IF ((Y).EQ.0) RETURN SUBROUTINE     EXA00180
 21    01    01/00   100         B     EXTIN                                                      EXA00190
 22    02            0F8         RT                                                               EXA00200
 23                       *      NOTE. MEMORY FROM INTERNAL TO EXTERNAL                           EXA00210
 24    03            0FE  INTEX: TMSI             (Y)=(Y)+1 , IF ((Y).EQ.0) RETURN SUBROUTINE     EXA00220
 25    04    01/03   103         B     INTEX                                                      EXA00230
 26    05            0F8         RT                                                               EXA00240
 27                       *                                                                       EXA00250
 28                              END                                                              EXA00260
```

① The program name is declared as "MELPS 41 EXAMPLE PROGRAM."
② An asterisk (*) in the first column indicates that the entire statement is a comment.
③ Numeric value 10 (decimal number) is assigned to the symbol A by means of the symbol-value equivalence instruction.
④ The value Z : X : Y = 2 : 0 : 0 is assigned to the symbol REG 1 by means of the external-memory address-setting instruction.
⑤ The value X : Y : 1 : 0 is assigned to the symbol REG 2 by means of the internal-memory address-setting instruction.
⑥ The following program is assigned to address 10 (hexadecimal number) of page 0 by means of the program-counter setting instruction.
⑦ The numerical value 10 (decimal number) assigned to the symbol A is loaded in register A.
⑧ The numerical value 1 is loaded in the page register.
⑨ The value assigned to symbol REG 1 is expanded in LZ, LX and LY instructions.
⑩ The label INTEX is assigned by means of the BM instruction during assembly process and calls the subroutine starting at page 1 address 3.
⑪ The value assigned to symbol REG2 is expanded in LX and LY instructions.
⑫ The label EXTIN is assigned by means of the BM instruction during assembly process and calls the subroutine starting at page 1 address 0.
⑬ The program-counter page number is advanced to that of the next page.

DESCRIPTION

The MELPS 41 simulator software has been prepared for facilitating program debugging of application programs suitable to equipment using the M58494-XXXP CMOS single-chip 4-bit microcomputer or microprocessors. It also allows a significant saving of program-development time.

With this simulator, each instruction of the microcomputer is executed just as though the program were being executed on an actual microcomputer system. This allows confirmation that the operations and sequences of a program are correct from a software point of view before the microcomputer system is built. Simulations using various simulator control commands are possible, and the results of the simulations are printed out along with other helpful information for verification and debugging of the program under development.

FEATURES

- An ample 26 control commands
- Production and deletion of trace and halt tables are possible
- Interruption-generation setting and periodical interruption are possible
- I/O port -setting function
- Data-setting function
- Execution-time counting function
- Reverse assembly is possible
- Memory-protection area-setting function
- Execution computer: MELCOM 70 (memory: 24K-words or larger)
- Implementation language: FORTRAN IV (parts are written in assembler language)

INPUT/OUTPUT MEDIA

- Object input: Cartridge-disk storages, punched tapes
- Control commands: Punched cards and system-typewriter keyboard
- Intermediate results: Punched tapes
- Simulation results: Line printers and system typewriters
- Messages: Line printers and system typewriters

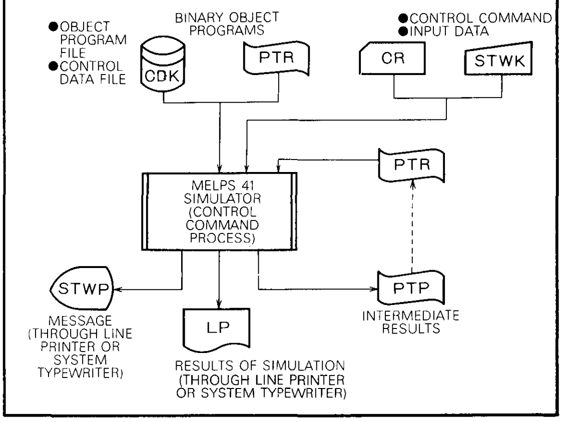
APPLICATIONS

In conjunction with the MELPS 41 cross assembler as a series of tools for developing application programs for single-chip 4-bit microcomputers. Especially useful for debugging programs prepared for the M58494-XXXP CMOS microcomputer.

PROGRAM ORDERING INFORMATION

Program name	Ordering number	Software manuals included
MELPS 41 simulator	GB1SM0002	MELPS 41 Simulator Manual MELPS 41 Simulator Operating Manual

MELPS 41 SIMULATOR PROCESSING SYSTEM



FUNCTION

Various simulator control commands are provided by the MELPS 41 simulator to help determine if the program is operating properly according to original specifications. These control commands can set operating conditions and halt program processes, while indicating the system status, CPU state and memory contents in a trace mode. Interruption-generation setting is also possible.

This simulator allows the production and deletion of trace and halt tables, table printouts, and the setting and printed indication of registers, stack pointers, carry flags, memories, and I/O ports. The 26 control commands can also be used for interruption generation, timer setting and reverse assembly.

SIMULATOR

Binary object codes stored in the disk file (BDISK), generated by the MELPS 41 cross assembler, are processed in this program, and a simulation is carried out according to the conditions given by the simulator control commands. The results of the simulation can be selectively displayed on a line printer or system typewriter. It is also possible to output or input intermediate results by means of punched tapes.

The simulator control commands are classified into (1) simulator control instructions for starting and ending simulations, loading and saving programs, and changing I/O devices, and (2) execution-control instructions for controlling simulation-execution status.

Control-Command Input Format

///XX␣ (parameter)

XX: Specified by a 2-character symbol (26 kinds).

Parameter: A required parameter can be selected from those which have been defined in the control

Table 1 Simulator control commands

Function classification	Item	Control commands: Action	Control commands: Mnemonic	Functions
	Simulator start-up	Specification of simulation-start conditions	ST	Designates the control-command input devices and the simulation-result output device, and sets them to start status.
Simulator control commands	Execution-program setting	Execution-program loading	LO	Loads the absolute object program (designates input-device file name).
Simulator control commands	Program saving	Execution-program saving	SV	Outputs intermediate results of the program content, register, port, F/F, Timer, and memory in punched tape.
Simulator control commands	Designation of simulation output device	Selection of command-input and simulation-result output devices	DV	Designates the command-input device and simulation-result output device by using the device symbol.
Simulator control commands	Simulator termination	Simulation-termination designation	FN	Terminates the program execution, and control is returned to the monitor.
Execution control commands	Trace	Trace-region assignment started	HS	Sets starting and termination addresses to the trace region, traces, and executes while printing out the contents of the registers, ports, timer and memory as specified.
Execution control commands	Trace	Trace-region assignment discontinued	TD	Discontinues trace-region assignment by table-number designation.
Execution control commands	Trace	Printout of the trace table	PT	Prints out the trace table.
Execution control commands	Halt	Halt-point assignment started	HS	Assigns halt points by page number, address and times of execution.
Execution control commands	Halt	Halt-point assignment discontinued	HD	Discontinues halt-point assignment.
Execution control commands	Halt	Printout of the halt-point table	PH	Prints out the halt-point table.
Execution control commands	Data setting	Initialization of the program counter, registers, memory, file, etc.	MM	Sets the initial data to the program counter, registers, I/O ports, memory file, etc.
Execution control commands	Data setting	Reset of the program counter, registers, memory file, etc.	CL	Resets the program counter, registers, I/O ports, memory file, etc.
Execution control commands	Data printout	Printout of the data in the program counter, registers, ports, flip-flop devices, memory, timer, etc.	DM	Dumps the contents of the program counter, registers, I/O ports, memory, flip-flop device, timer, etc.
Execution control commands	Port control	Input-port control	IN	Controls the input-port data read-in device and the input port by print-mode designation.
Execution control commands	Port control	Export-port control	OT	Designates an output device for the data obtained from the output port.
Execution control commands	Interruption	Interruption-generation assignment started	IT	Sets interruption conditions such as interruption type, interruption generation, head address, and generation cycle number.
Execution control commands	Interruption	Interruption-generation assignment discontinued	ID	Deletes the interruption-generation table.
Execution control commands	Interruption	Printout of the interruption-generation table	PI	Prints out the interruption-generation table.
Execution control commands	Execution step time	Execution-timer setting and printout	TI	Sets the execution timer and prints out the number of execution steps.
Execution control commands	Memory protection	Memory-protection-region assignment started	PS	Designates the kind of memory, starting and termination addresses of the protected region, and inhibits write-in steps.
Execution control commands	Memory protection	Memory-protection-region assignment discontinued	PD	Discontinues the memory-protection assignment by the memory-protection table number.
Execution control commands	Memory protection	Printout of the memory-protection region	PP	Dumps the contents of the memory-protection table.
Execution control commands	Execution start	Program-execution start-up	RN	Starts simulation execution. Termination by executing the halt point and the execution-limit step number.
Execution control commands	Execution start	Program execution	GO	Starts simulation execution. Termination by halt-point execution. Trace-region assignment is invalid here.
Execution control commands	Reverse assembly	Reverse assembly control	PA	Reverse-assembles the specified region and prints out the source list.

command. A comma (,) is used to divide one parameter from another.

The following are parameter-configuration examples: reserved word, address indication, numerical-value setting, numerical-value indication, and time setting.

1. Reserved word

This symbol is classified according to its function in the simulator, and specifies a predetermined character symbol, program counter (PC), memory, register, and port.

///MM␣REGS A = 9

2. Address indication

Address indications for the internal memory, external memory and ROM are possible.

///DM␣EXTM, 0:1:E, 0:A:5 External memory address indication

///DM␣INTM, 0:0, 1:0 Internal memory address indication

///MM␣PROG, OF:23 ROM address indication

3. Numerical value setting

A numerical value is set for each function parameter.

///MM␣FFLG, CY = 1

4. Numerical value indication

Decimal or hexadecimal notation is used.

///MM␣TIME, T1 = E

5. Time setting

The specified time is set.

///TI␣SET, 8:15:3

(Note : This parameter means 8ms, 15, 3μ sec.)

APPLICATION EXAMPLES

Once the command ST (this is used for specifying simulation-start conditions) and its parameter are typed in through the system-typewriter keyboard, successive commands may enter through punched cards or the system-typewriter keyboard. It is also possible to designate command-input and result-printer devices by setting the DV-command parameter.

Simulation is started on the object file in the disk storage that was stored there, after assembling, by the MELPS 41 cross assembler. When the MELCOM 70 is used, the simulator program should be called by the command //EXEC␣SIM41 to start simulating operation. The following are examples of command assignment in the case of tracing and execution during system-application program simulation.

Assignment of the input and printer devices is entered by the ST command in the format ST␣X, Y, where X represents the input device (S for the system typewriter, and C for the card reader; no designation equals the S designation in effect), and Y represents the output device on which the simulation result is printed out (L for the line printer and S for the system typewriter; no designation equals the L designation in effect).

The stored object program (BDISK file) is loaded by the simulator with the LO command in the format LO␣ file name. The CL command should be used for clearing the initial values and the MM command for setting initial values.

When the program counter, registers, I/O ports and memory file are to be cancelled, the command CL may be used. The MM command in the format of MM␣XXXX,nnnn can be used for setting their values. Here XXXX represents the symbol or numerical figure by which the program counter, registers, I/O ports or memory files are designated, while nnnn represents a parameter to be assigned.

Designating the halt command HS PP: aa nnnn will make the machine halt at address aa of page PP after that instruction has been executed nnnn times.

Entry of the TS command

TS␣P_1P_1:a_1a_1, p_2p_2:a_{22}, R, P, I. X_1:Y_1, X_2:Y_2(, E, Z_1: X_1:Y_1, Z_2:X_2:Y_2)

makes possible the assignment that a trace is to be carried out from address a_1a_1 of page p_1p_1 to address a_2a_2 of page p_2p_2. Here R designates the output of the contents of the registers and F/F print; P designates the ports and timer print; and I and E respectively designate print modes for the internal and external memories.

When the DM command is executed, the contents of each register, port, flip-flop device, memory, timer, and program counter are printed out.

Interruption can be carried out by the IT, ID and PI commands. The IT command designates the kind of interruption, the head address of interruption generation, and the number of generation cycles. ID discontinues the interruption-generation assignment, and PI effects interruption-generation table prinouts.

The TI command can be used for execution timer setting and printouts. The PS, PD and PP commands are provided for memory protection. PS designates the memory-protection-region assignment, PD discontinues the memory-protection-region assignment in accordance with the memory-protection table number, and PP prints out the contents of the memory-protection-region.

Table 2 Examples of the use of simulator control commands

Application exmaples of control command	Function of the conrol command and its parameters
///ST␣S, L	To start simulation, the command-input and simulation-result printout devices are assigned. In this example, command input S is assigned to the system typewriter, and printout L to the line printer.
///LO␣D, BFILE	The file stored in the disk (BDISK) whose file name is BFILE is loaded.
///CL␣INTM, O:O, O:F	The designated internal memory is cleared from digit 0 to digit F of the 0 file.
///HS␣5:F, 2	This assigns a halt point: in this example it will halt after the second execution of the instruction in address F of page 5.
///TS␣O:5, E:F, R, P	This command designates a trace from address 5 of page 0 to address F of page E, and orders display of the contents of each register, flip-flop device, port and timer after completing tracing.
///IT␣INTA, O:F, 5	This effects the generation of interruption A starting at address F of page 0 and after every 5 steps after that.
///PT INTA, O:F, 5	This command prints out the trace table. Assignments made by TS commands can be verified by this command.
///PH	This command prints out the halt-point table. Assignments made by HS commands can be verified by this command.
///MM␣PORT, Q=A5	This sets A5 to port Q.
///MM␣INTM, O:O **O:O O=1** **O:1 O=•**	This command changes the value 0 in digit 0 of file 0 to 1.
///DM	The contents of the program counter, registers, I/O ports, flip-flop devices, memory, and timer at the time this command is executed are printed out.
///RN 50	This starts the execution of simulation, which is stopped when the halt-point address is reached or when the number of execution steps reaches 50.

14

Fig. 1 Example of simulation results

```
*** START SIMULATOR OF MELPS 41 ***

///ST                          ........................①

///LO D,BFILE                  ........................②

///MM INTM,0:0                 ........................③
 0:0   0=1
 0:1   0=2
 0:2   0=.

///MM INTM,0:F
 0:F   0=3
 1:0   0=.

///MM EXTM,2:1:0
 2:1:0   0=A                   }.......................④
 2:1:1   0=.

///MM EXTM,2:1:E
 2:1:E   0=B
 2:1:F   0=C
 2:2:0   0=.

///MM REGS,PC=10               ........................⑤

///TS 00:16,00:17,E,2:0:0,2:1:F ......................⑥

///TS 00:16,00:16,R            ........................⑦

///TS 00:19,00:1A,I,0:0,1:F

///TS 00:19,00:19,R

///HS 00:1A,1                  ........................⑧

///DM REGS                     ........................⑨

        (REGS)
          PC   PREG ACC TR Q  R  S  T  U   Z X Y   SP   K
         00:10(00)  0   0  00 00 00 00 00  0 0 0   F   00:00

///PT                          ........................⑩

        *****        TRACE  DUMP  TABLE        *****
          TBL.NO   TRACE.ADD        TRACE.MODE
             1     00:16 00:17        E(2:0:0,2:1:F)
             2     00:16 00:16   R
             3     00:19 00:1A        I(0:0,1:F)
             4     00:19 00:19   R

          * CATALOG COUNT =  4 *

///PH                          ........................⑪

        *****  HALT POINT TABLE  *****
          TBL.NO  HALT.ADD  EXEC.NUM


             1       00:1A        1

          * CATALOG COUNT =  1 *

///TI SET                      ........................⑫

///RN 100                      ........................⑬

 00/16   383      BM        03

        (REGS)
          PC   PREG ACC TR Q  R  S  T  U   Z X Y   SP   K
         01:03(02)  A   0  00 00 00 00 00  2 0 0   0   00:00

        (FFLG)
          CY      INTF             INT        MF    MR    MR1   MR2  IOA
              (A) (T) (B)   (A) (T) (B)        (21LQ)(SRBT)(--AI)
          0    0   0   0     0   0   0     1    0000  1010  0000   0

           (EXTM)
          2:0:0   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   }..........⑭
          2:1:0   A 0 0 0 0 0 0 0 0 0 0 0 0 0 B C

 00/17   1B1      LX        01

           (EXTM)
          2:0:0   1 2 0 0 0 0 0 0 0 0 0 0 0 0 0 3   }..........⑮
          2:1:0   A 0 0 0 0 0 0 0 0 0 0 0 0 0 B C

 00/19   380      BM        00

        (REGS)
          PC   PREG ACC TR Q  R  S  T  U   Z X Y   SP   K
         01:00(02)  A   0  00 00 00 00 00  2 1 0   0   00:00

        (FFLG)
          CY      INTF             INT        MF    MR    MR1   MR2  IOA
              (A) (T) (B)   (A) (T) (B)        (21LQ)(SRBT)(--AI)
          0    0   0   0     0   0   0     1    0000  1010  0000   0

           (INTM)
           0:0    1 2 0 0 0 0 0 0 0 0 0 0 0 0 0 3   }..........⑯
           1:0    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

 00/1A   000      NOP

           (INTM)
           0:0    1 2 0 0 0 0 0 0 0 0 0 0 0 0 0 3   }..........⑰
           1:0    A 0 0 0 0 0 0 0 0 0 0 0 0 0 B C

///TI DIS

          ***    EXECUTION  COUNTER    ***          }..........⑱
             # STEP #            # TIME #



                     75.          0 M 547..................⑲

///FIN
```

Execution is started by the RN and GO commands, and it is continued to the point specified by the HS parameter. In the case of RN the machine is stopped by executing the limit-step number and the IDLE instruction. In the case of GO, termination is effected after the execution of the IDLE instruction, and the trace-region assignment becomes invalid.

The assignment of the trace region is discontinued with the TD command, and the halt-point assignment with the HD command. The trace table is printed out with the PT command, and halt-point table with the command PH, whenever required.

The IN and OT commands can be used for I/O-port control. The DV command is provided for designating devices for command input and simulation-result printout.

Application examples of the use of the MELPS41 simulator control commands are listed in Table 2, and the results of a simulation example are shown in Fig. 1.

① The operation start-up of the MELPS 41 simulator is designated. At this time, it is specified that the control command is to be input through the system typewriter and the results of simulation output on the line printer.
② The program is loaded from the file BFILE of the disk.
③ Data is set from the 0:0 of the internal memory.
  The value 0 of the memory 0:0 is set to 1.
  The value 0 of the memory 0:1 is set to 2.
  The assignment is ended without changing the value 0 of memory 0:2.
④ Data is set from 2:1:0 of the external memory.
  The value 0 of the memory 2:1:0 is set to A (hexadecimal number)
  The assignment is ended without changing the value 0 of memory 2:1:1.
⑤ The address inside the program-counter page is set to 10 (decimal number).
⑥ Tracing of the region from address 16 of page 0 to address 17 of page 0 is designated, and the contents of the region from 2:0:0 to 2:1:F of the external memory are printed out.
⑦ When address 16 of page 0 is executed, the contents of the flip-flop device and register are printed out.
⑧ With the halt point set to address 1A of page 0, termination after a single execution is specified.
⑨ The contents of the registers and flip-flop device are printed out.
⑩ The trace-region table is printed out.
⑪ The halt-point table is printed out.
⑫ The execution counter is initialized.
⑬ The program execution is started. Trace is carried out in accordance with the assignment given by steps ⑥ and ⑦, and is terminated at the halt point designated by step ⑧. Otherwise, termination is entered when 100 steps are executed.
⑭ The contents of the external memory before transferring the contents of the internal memory to it are printed.
⑮ This shows that the 16-digit data from 0:0 to 0:F in the internal memory have been transferred to the region 2:0:0-2:0:F of the external memory.
⑯ The contents of the internal memory before transferring the contents of the external memory to it are printed.
⑰ This shows that 16-digit data from 2:1:0 to 2:1:F in the external memory have been transferred to the region 1:0 - 1:F.
⑱ The number of steps executed and the execution time are printed.

PAPER-TAPE GENERATION PROGRAM FOR PROM WRITERS

DESCRIPTION

The MELPS 41 PROM writer paper-tape generation program is used to convert the absolute binary object programs generated by the MELPS 41 cross assembler into another format that can be used in a PROM writer. The program is output on paper tape in the new format. This program also allows converting paper tapes in hexadecimal form into binary objects.

With the MELPS 41 program, a binary object program can easily be converted to hexadecimal object format that can be programmed directly into an EPROM. It can produce paper tapes that meet the requirements for various types of EPROMs and PROM writers because of its functional versatility.

FEATURES

- Outputs the binary object program in the disk storage to paper tapes in hexadecimal format
- Converts hexadecimal-form paper tapes into binary objects
- Comparison function
- Outputs PROM-writer format selectively
- Paper-tape output can be partitioned with a simple control command
- May be used in conjunction with the MELPS 41 cross assembler
- Execution computer: MELCOM 70 minicomputer (memory capacity, more than 16K-words; monitor, BDOS)
- Programming language: FORTRAN IV (parts are written in assembly language)

INPUT/OUTPUT MEDIA

- Input: Cartridge-disk units, paper tapes (ASCII code, even parity)
- Output: Paper tapes (ASCII code, even parity), cartridge-disk units
- Control-command outputs: Through system-typewriter keyboard
- Message: System-typewriter printout

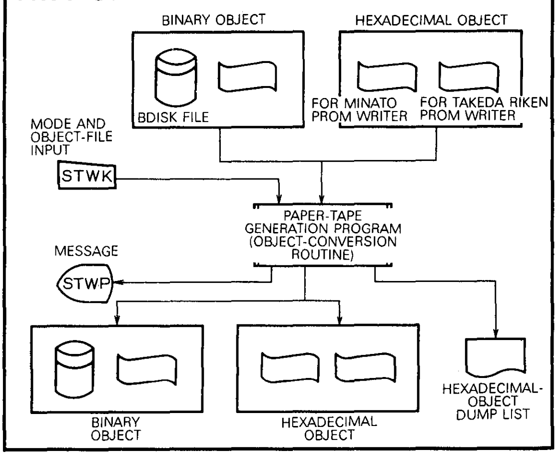
APPLICATIONS

For preparing programs for EPROMs (M5L2708K, S M5L2716K, etc.) that are to be programmed by PROM writers supplied by Takeda Riken (T310) or Minato Electronics (Models 1830 and 1802).

FUNCTION

This program is used for converting the absolute binary object programs that were generated by the MELPS 41 cross assembler in the disk area to a hexadecimal object format compatible with Minato Electronics Models 1830 and 1802 and Takeda Riken (T310). The paper-tape output is partitioned in accordance with EPROM capacity (number of bytes). The program also permits the processing of hexadecimal-format object paper tapes for input conversion and storage in the disk in a binary object format. Outputs on paper tapes are also available.

PROCESSING SYSTEM



PROGRAM PROCESSING

The program has routines for selectively converting binary objects processed with the MELPS 41 cross assembler into paper tapes for Takeda Riken's and Minato Electronics' PROM writers.

Object conversion can be carried out by designating the input and output modes. For example, select BD mode for input and TI mode for output (for Takeda Riken's PROM writer) or BD mode for input and MI mode for output (for Minato Electronics' PROM writer) through the system-typewriter keyboard. Then the object program is converted to paper tapes compatible with the selected PROM writer only by calling the object file (BDISK file) into which it is to be converted and then putting in the number of paper-tape outputs. By putting the paper tapes after conversion into the disk of file 1 when the original data are in file 2, their contents can be compared with each other. The file-comparison function allows easy checking of the validity of the converted paper tapes.

It is also possible to input hexadecimal-format paper tapes, to store them in the disk as a binary-object file, and to output them on paper tapes. In this case, the binary-object paper tapes are composed of name, text and end segments. After completion of the conversion, control can be returned to the monitor by the EN command.

PROGRAM ORDERING INFORMATION

Program name	Ordering number	Program and software manuals included
MELPS 41 Paper-tape generation program for PROM writers	GB1SP0003	MELPS 41 paper-tape generation program for PROM writer manual MELPS 41 paper-tape generation program for PROM writer operating manual

PAPER-TAPE GENERATION PROGRAM FOR PROM WRITERS

The object disk file consists of name and text segments. The data to be converted is contained in the text segment. Instruction codes stored after section 1 of the disk that correspond to the machine instructions are converted into hexadecimal codes and output to paper tapes.

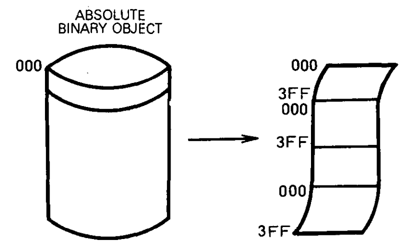
Example of Hexadecimal Tape Output

This program can generate paper tapes for Minato Electronics' and Takeda Riken's PROM writers. It can output 8 paper tapes in 1K-byte units at maximum. Examples are shown in Figs. 1—3.

Example of Object Conversion

This program can output 1K-word units of paper tape up to a total of 8K-words. Fig. 4 shows an example in which conversion is made from an absolute binary object (disk) to paper tape.

Fig. 4 Example of object conversion



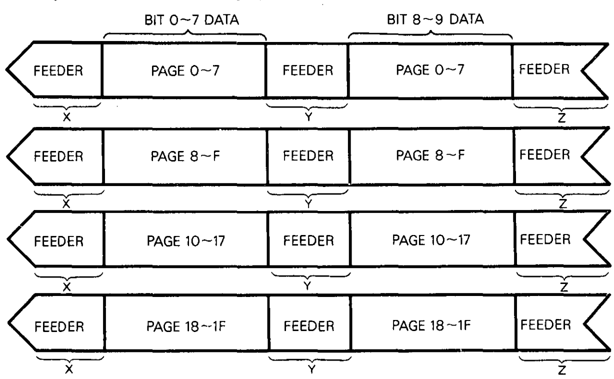
Error Processing

When an error is encountered during object conversion, an error message will be printed out in the following format:

$ $ $ $ $ $ $ E R R O R␣X X X $

where XXX indicates the error code.

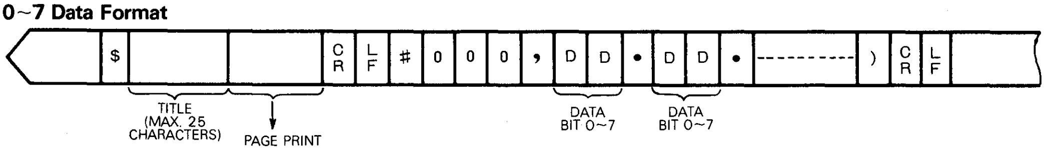
Fig. 1 Example of hexadecimal paper-tape output



Note : X, Y and Z denote the numbers of the sprocket holes: X: 100 or more Y: 200 or more Z: 200 or more.

Fig. 2 Example of hexadecimal paper-tape format of Takeda Riken

●Bit 0~7 Data Format



●Bit 8~9 Data Format

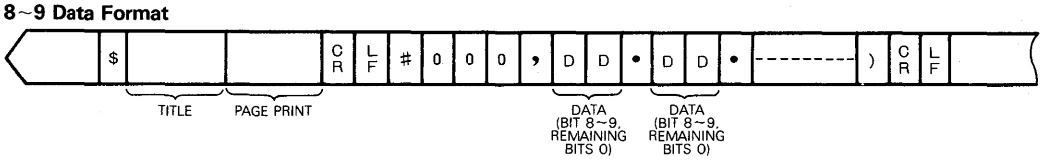
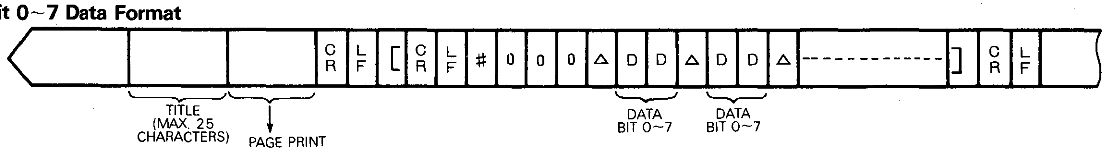
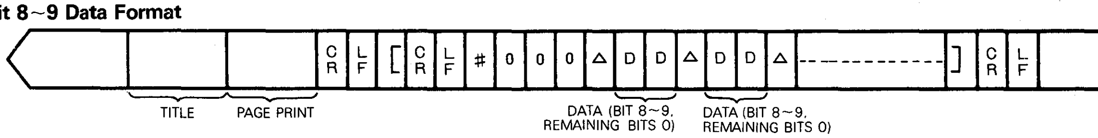


Fig. 3 Example of hexadecimal paper-tape format of Minato Electronics

●Bit 0~7 Data Format



●Bit 8~9 Data Format



CROSS ASSEMBLER

DESCRIPTION

The MELPS 42 cross assembler has been prepared for the development of application programs suitable for equipment using the M58496-XXXP single-chip 4-bit microcomputer.

This cross assembler not only provides many pseudo instructions, control commands, and control data for improving programming efficiency, but it also provides program versatility for changing instruction codes and functions.

FEATURES OF THE CROSS ASSEMBLER

- 3 types of control data
- Instruction codes and functions easily changed
- Catalogs the control data in disk storage
- Constants can also be expressed in non-decimal notations
- Printouts available from the tables and cross-reference lists
- Execution computer: MELCOM 70 (memory capacity more than 24K words, monitor BDOS)
- Implementation language: FORTRAN IV (parts are written in assembly language)

FEATURES OF THE ASSEMBLY LANGUAGE

- 6 pseudo instructions
- 77 machine instructions
- Decimal numbers can be used to define the constants of the machine instruction operand field.

INPUT/OUTPUT MEDIA

- Source input : Punched cards and magnetic disk
- Control data input : Punched cards and magnetic disk
- Control data command : Punched cards
- Execution command : System typewriter keyboard
- Object output : Magnetic disk
- Output lists : Line printer

FUNCTION

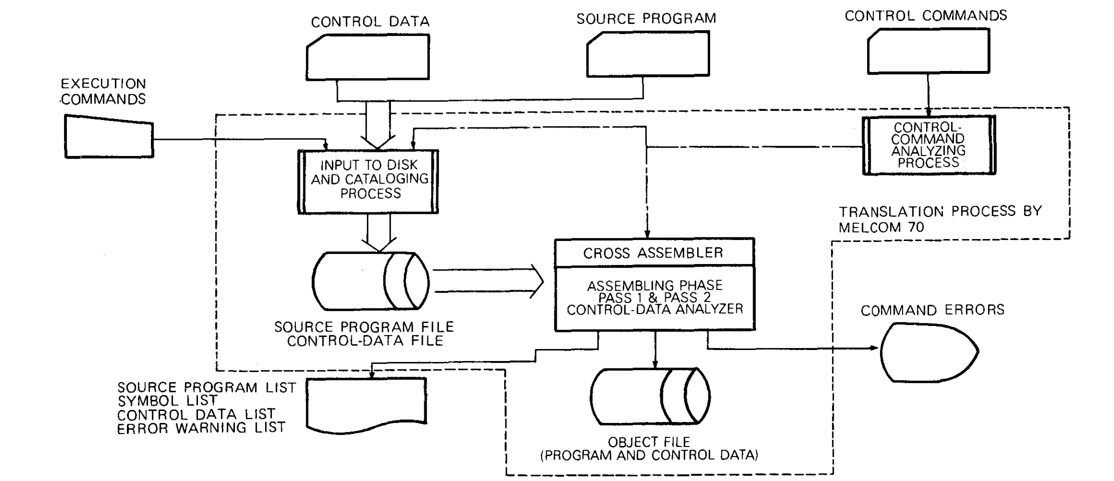
This cross assembler converts source programs written in the MELPS 42 assembly language to machine instruction codes that are filed in disk storage in the form of binary absolute object codes.

The MELPS 42 cross assembler is a 2-pass translator that provides data and control command analysis along with cataloging functions.

Modifying the instruction code and setting mnemonic tables and numeric tables to constants can easily be accomplished by means of the control data. In this way, programming versatility is provided for changing functions, allowing the user free selection in defining the mnemonics of the machine instructions, etc.

The standard version of the MELPS 42 assembly language has 7 assembler control commands (see Table 1). In addition 6 pseudo instructions (Table 2) can be used in the source language program.

PROCESSING SYSTEM



PROGRAM ORDERING INFORMATION

Program name	Ordering number	Program and software manuals included
MELPS 42 cross assembler	GBIAS0010	MELPS 42 Cross Assembler Manual GBM-SR 00-31A<03A0>

CROSS ASSEMBLER

This cross assembler facilitates assembly by the use of the control commands shown in Table 1. Basically, it requires only the source program and control commands input by punched cards with control data being utilized only when necessary. All input is stored and filed in disk storage. The control data is processed by the control command analyzing processor, and the symbol table is created in pass 1. This is followed by pass 2, where each instruction is converted to machine language, while control data, labels and the assembly list are printed out as specified by the control commands. On the assembly list, the control commands, sequence numbers, location numbers and addresses are printed out, along with error and warning messages, followed by the ROM page list and the cross-reference list.

OBJECT LANGUAGE

The object file is composed of a name section and a text section.

The name section is filed on sector 0 of the object file and stores overall information such as the total number of instructions in the text section, control data, file name, source program file name, size of a single page and the module name.

The text section contains the data that controlled the conversion of the source program to instruction codes and other related data necessary for execution by the simulator.

ASSEMBLY LANGUAGE

The assembly language that the MELPS 42 cross assembler accepts consists of machine instructions and pseudo instructions.

1. Machine Instructions

There are 77 basic machine instructions. These are converted to their corresponding machine codes and then assembled into an object program. For the mnemonics, instruction codes and their functional descriptions, please refer to the data sheet provided for the M58496-XXXP single-chip 4-bit microcomputer.

2. Pseudo Instructions

Although the pseudo instructions are written in the source program together with machine instructions, they are not converted to instruction codes but are used to control the assembler. The instruction codes will be written in the ROM.

The assembler-control instructions, numeric symbols defining instructions and list control instructions are among the pseudo instructions.

The pseudo instructions are shown in Table 2.

Table 1 Assembler control commands

Command		Format	Function
Execution start		/ / / **RUN**	Starts execution of the cross assembler
Execution end		/ / / **END**	Terminates execution of the cross assembler
Input/output function assignment		/ / / **ASMB4, x, y, z**	Assignment of assembly execution and control data and assembly listings x = (A / P) y = (L / N) z = (L / N) **x** : Assembly control **A** : Assembly needed **P** : Designation of cataloging function **y** : Assembly listing **z** : Control data listing **L** : Listing needed **N** : No listing needed
File assignment control	Control data	/ / / **CDISK, XXXXX**	Assignment of the control file name (max. 6 characters)
	Source program	/ / / **SDISK, XXXXX**	Assignment of the source program file name (max. 6 characters)
	Object	/ / / **BDISK, XXXXX**	Assignment of the object file name (max. 6 characters)
Input/output device assignment		/ / / **INPUT, x, y**	Assignment of input device for the control data and source program x = (C / D / N) y = (C / D / N) **x** : Control data input **y** : Source program input **C** : Punched card input **D** : Disk input **N** : Control data no input

Table 2 Pseudo instructions

Classification	Mnemonic	Instruction	Function
Assembler control instruction	TTL	Program title declaration	Declares the program title
	PAGE	Program counter paging	Sets the counter to the top address of the next page
	ORG	Program counter setting	Sets the counter to the top address of the program
	END	End declaration	Declares the end of the program
Symbol value equivalence instruction	EQU	Symbol value setting	Sets a numeral value to the specific numeral symbol
List control instruction	EJE	Page eject declaration	Advances the printout form to the next page during output

3. Language Format

The following format should be used in coding programs in this cross assembler.

The single-line statement is composed of the label, instruction, operand, comment, and identification fields. The format of the source statement is fixed as indicated in Fig. 1. Although the constant is usually a decimal number, it may be expressed by hexadecimal notation when defined by pseudo instructions.

An asterisk (*) in the first column of a line indicates that the entire statement is used as a comment field.

The following are valid characters for use in statements:

Alphabetics: A~Z

Numberics: 0~9

Special characters: ; = , ▾ @ $ + − * / ! & ( ) . # % < > ? (space)

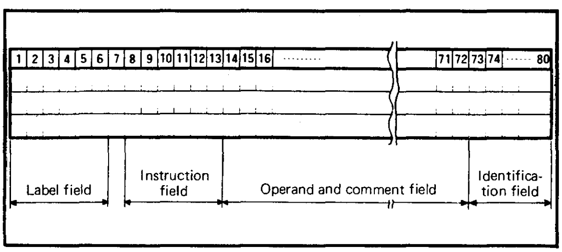


Fig. 1 Source statement format

(1) Label field

The value of the program counter at that time is set to the label. The number of characters used for a label is limited to a maximum of 6, and any of the alphanumerics and special characters specified above can be used. However, an asterisk (*) cannot be used in the first column of the label field.

(2) Instruction field

Mnemonic codes are written in this field, left-justified. For pseudo instructions, any of the mnemonics among the assembler-control instructions, numeric symbol definition instructions and list-control instructions may be used.

(3) Operand field

Parameters of the instruction are specified in this field. This field contains the label, defined symbol, or numerical value. The operand is stated from the 14th column, left-justified.

(4) Comment field

Whenever the operand is followed by more than one space to the end of the statement, the successive columns may be used for comments.

(5) Identification field

The use of this field is optional. Many find it convenient to use this field for a sequential identification card number.

ASSEMBLY LIST FORMAT

A source program prepared and assembled in the format indicated in the preceding paragraph may produce source, symbol table, cross reference, and ROM page list printouts. The format of an assembly list produced as an example is shown in Fig. 2. Please note that pages, locations, and object codes are indicated in hexadecimal notation.

MESSAGE FORMAT

Error and warning messages are printed out on the assembly list. In the case of errors, the message is printed out under the respective statement in the following format.

$$$$$$$ERROR xxx$

where "x x x" indicates the type of error by a numerical code.

In the case of warnings, the following message is printed between SEQ (sequential number) and LOC (location number):

*** Wx *** (where "x" indicates the degree of warning)

In addition the total number of errors and warnings are printed on the last line of the assembly list. The cross-reference list, however, will not be produced when any errors are indicated.

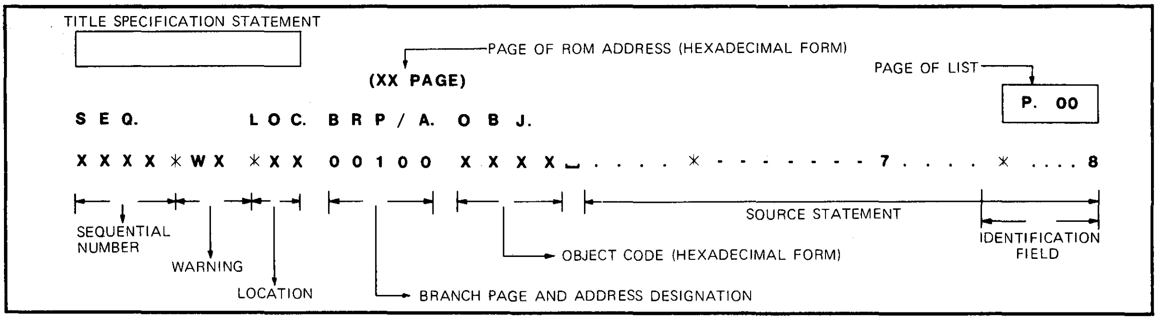


Fig. 2 Assembly list format

Example of an assembly list

An actual example of an assembly list for an assembly made with the MELPS 42 cross assembler is shown in Fig. 3.

```
EXAMPLE PROGRAM                                                                                    P. 1
                                      ( 00 PAGE )
SEQ.   LOC.BRP/A. OBJ. ....*....1....*....2....SOURCE STATEMENT....*....5....*....6....*....7....*....8

  1                           TTL    EXAMPLE PROGRAM-----------------------------------------① EXA00010
  2                           ORG    0,0             ----------------------------------------② EXA00020
  3                    * FILE DATA EXCHANGE          ----------------------------------------③ EXA00030
  4                    *                                                                       EXA00040
  5                    DIGMAX EQU    13              DIGMAX=13 ------------------------------④ EXA00050
  7    01 0E/00  100          BM     XCG02           EXCHANGE F0 & F2 -----------------------⑤ EXA00070
  8    02 0E/01  101          BM     XCG13           EXCHANGE F1 & F3 -----------------------⑥ EXA00008
  9    03 0E/07  107          BM     XCG23           EXCHANGE F2 & F3 -----------------------⑦ EXA00009
 10    04        000          NOP
 11                    *                             ----------------------------------------⑧ EXA00100

 12                           ORG    E,0                                                       EXA00110
 13                    * SUBROUTINE FILE EXCHANGE                                              EXA00120
 14                    *                                                                       EXA00130
 15                    *      EXCHANGE FILE M(2,X,0-DIGMAX)                                    EXA00140
 16                    *                                                                       EXA00150
 17    00        0CD   XCG02  LXY    0,DIGMAX        EXCHANGE F0 (0-DIGMAX) & F2(0-DIGMAX)--⑨ EXA00160
 18    01        0DD   XCG13  LXY    1,DIGMAX        EXCHANGE F1 (0-DIGMAX) & F3(0-DIGMAX)--⑩ EXA00170
 19                    *                                                                       EXA00180
 20    02        066   LBL4   TAM    2                                                         EXA00190
 21    03        062          XAM    2                                                         EXA00200
 22    04        068          XAMD   0                                                         EXA00210
 23*W0*05        102          BM     LBL4            BM IS EQUIVALENT WITH B ON PAGE 14-----⑪ EXA00220
 24    06        044          RT                                                               EXA00230
 25                    *                                                                       EXA00930
 26    07        0ED   XCG23  LXY    2,13            EXCHANGE F2 (0-DIGMAX) & F3(0-DIGMAX)     EXA01010
 27                    *      COMMON ROUTINE START                                             EXA01020
 28                    *                                                                       EXA01030
 29    08        065   LBL5   TAM    1                                                         EXA01040
 30    09        061          XAM    1                                                         EXA01050
 31    0A        068          XAMD   0                                                         EXA01060
 32*W0*08        108          BM     LBL5                                                      EXA01070
 33    0C        044          RT                                                               EXA01080
 34                           END                                                              EXA01100
```

① The program name is declared as "EXAMPLE PROGRAM".
② It shows that the start of the program was set to page 0 address 0 by means of the program counter setting instruction.
③ An asterisk (*) in the first column indicates that the entire statement is a comment.
④ Numeric value 13 (decimal number) is assigned to the symbol DIGMAX by means of the symbol value equivalence instruction.
⑤ The label XCG02 is assigned by means of the BM instruction during the assembly process, and calls the subroutine starting at page 14 address 00.
⑥ The label XCG13 is assigned by means of the BM instruction during the assembly process, and calls the subroutine starting at page 14 address 01.
⑦ The label XCG23 is assigned by means of the BM instruction during the assembly process, and calls the subroutine starting at page 14 address 07.
⑧ This whole statement line is used as a comment field.
⑨ The numerical value 0 is loaded in register X of the data pointer and 13 (decimal number) in register Y by means of the LXY instruction. As written, the results of this LXY instruction are nullified by the results of the following LXY instruction.
⑩ The numerical value 1 is loaded in register X of the data pointer and 13 (decimal number) in register Y by means of the LXY instruction.
⑪ The BM instruction in this case assigns the branch address of the label LBL4 to address 02 of page 14.

PAPER-TAPE GENERATION PROGRAM FOR PROM WRITERS

DESCRIPTION
MELPS 42 PROM writer paper-tape generation programs are used to convert the absolute binary object program generated by the MELPS 42 cross assembler into another format that can be used in a PROM writer. The program is output on paper tape in the new format.

With this program, a binary object program can easily be converted to hexadecimal object format that can be programmed directly into a PROM. It can produce paper tapes that meet the requirements of Takeda Riken's and Minato Electronics' PROM writers.

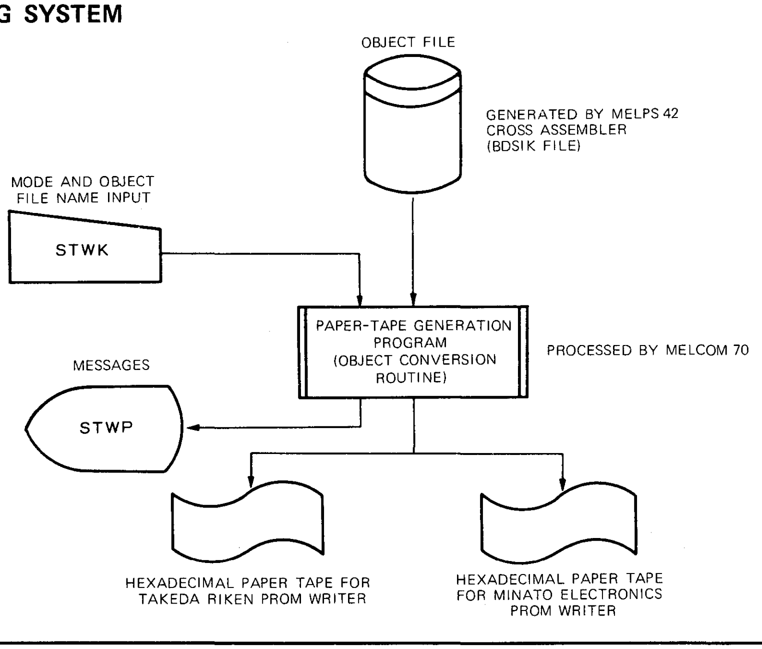
FEATURES
- Outputs the binary object program in the disk storage to paper tape in hexadecimal format
- Paper-tape output can be partitioned with a simple control command
- May be used in conjunction with the MELPS 42 cross assembler
- Execution computer: MELCOM 70 Minicomputer (memory capacity more than 16K words, monitor BDOS)
- Programming language: FORTRAN IV (parts are written in assembly language)

INPUT/OUTPUT MEDIA
- Input : Cartridge disk storage
- Output : Paper tape (ASCII code, even parity)
- Control command input : Through the keyboard of the system typewriter
- Messages : System typewriter printout

APPLICATIONS
For preparing programs for 1K words X 8-bit EPROMs (M5L2708K, S), etc., which are to be programmed by PROM writers supplied by Takeda Riken or Minato Electronics.

FUNCTION
This program is used for converting the absolute binary object format programs generated by the MELPS 42 cross assembler to hexadecimal object format compatible with the PROM writers manufactured by Takeda Riken (T310) and Minato Electronics (model 1830 and 1802). The paper-tape output is partitioned in accordance with PROM capacity (number of bytes).

PROCESSING SYSTEM



PROGRAM ORDERING INFORMATION

Program name	Ordering number	Program and software manuals included
MELPS 42 paper-tape generation program for PROM writer	GBISP 0006	MELPS 42 paper-tape generation program for PROM writer manual GBM-SR 00-33A <03A0>

PAPER-TAPE GENERATION PROGRAM FOR PROM WRITERS

PROGRAM PROCESSING

The program has conversion routines for Takeda Riken's and Minato Electronics' PROM writers. Select T_1 mode (for Takeda Riken's PROM writer) or M_1 mode (for Minato Electronics' PROM writer) through the system typewriter keyboard. Then the object program is converted to paper tape compatible with the selected PROM writer. When a (BDISK file) file name is called, a paper tape is output for the PROM writer. When a number of programs are to be converted from the same file, successive calls can be made until all the programs are converted. Termination of the job is directed with the E command, and control is then returned to the monitor.

The object file consists of name and text segments. The data to be converted is contained in the text segment. Instruction codes, stored after sector 1 of the disk, that corresponds to machine instructions are converted to hexadecimal codes and output to paper tape.

Example of Hexadecimal Paper-Tape Format

This program can generate paper tapes for Takeda Riken's PROM writer or Minato Electronics' PROM writer. Examples of both formats are shown in Figs. 1 and 2.

Example of Object Conversion

The program at present can output 1K-word units of paper tape up to a total of 4K words. An example is shown in Fig. 3.

Error Processing

When an error is encountered during object conversion, a message will be printed out in the following format:

$$$$$$$ xxx$

where, XXX indicates the error code.

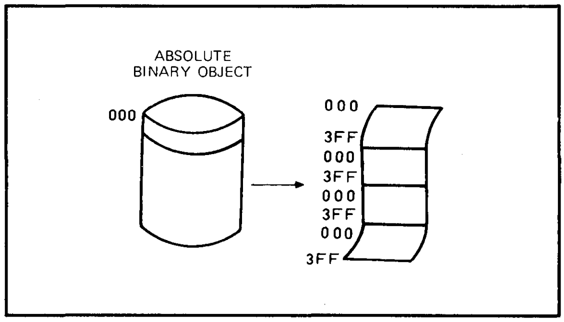


Fig. 3 Example of object conversion

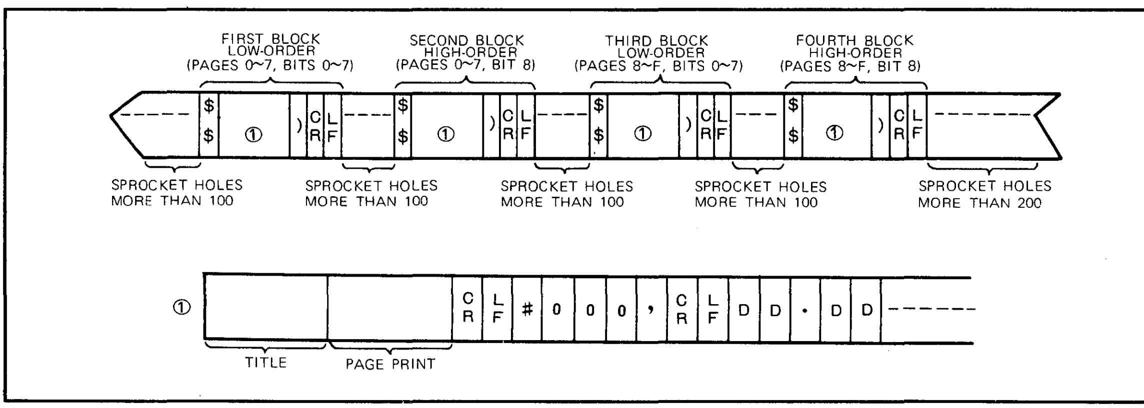


Fig. 1 Example of hexadecimal paper-tape format of Takeda Riken

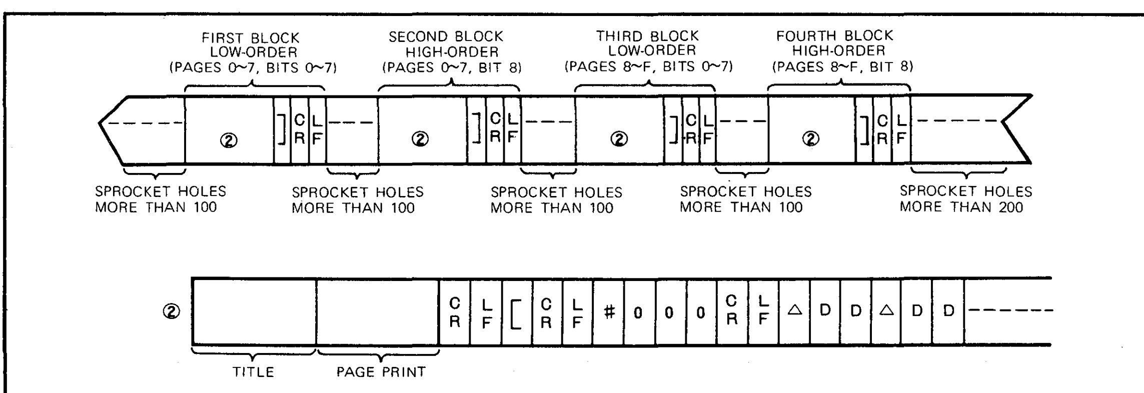


Fig. 2 Example of hexadecimal paper-tape format of Minato Electronics

CROSS ASSEMBLER

DESCRIPTION

The MELPS 8-48 cross assembler has been prepared for aiding the development of application programs suitable for equipment using the M5L8041-XXXP, M5L8048-XXXP and M5L8049-XXXP MITSUBISHI single-chip 8-bit micro-computers.

This cross assembler allows conversion of source programs written in the MELPS 8-48 assembler language by using a host computer into objects in the MELPS 8 binary language.

The assembler language has machine pseudo and macro instructions. The full equipment of pseudo instructions and control commands ensures high programming and debugging efficiency. Coding can be carried out in a free format.

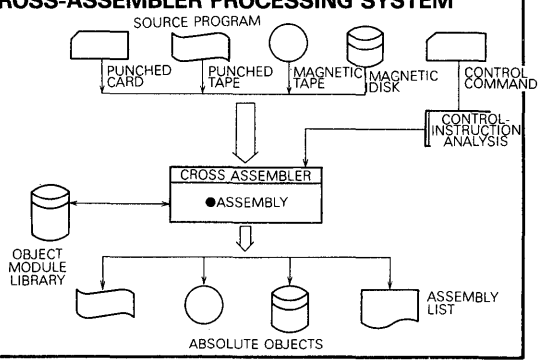
FEATURES OF THE CROSS ASSEMBLER

- Flexibility in assembler-language changing
- Various input/output media available
- Free-format coding
- A symbol table is output as part of the object code.
- Executed on a MELCOM 70 minicomputer (with 24 K words of memory capacity or more, BDOS monitor)
- FORTRAN IV programming language (with some assembler language)

FEATURES OF THE ASSEMBLER LANGUAGE

- 10 pseudo instructions
- 6 Macro instructions
- Numerical formula used
- Character constants and strings used

CROSS-ASSEMBLER PROCESSING SYSTEM



- In addition to decimal notation as the standard format, binary, octal and hexadecimal notations can be used
- Machine-instruction compatibility with Intel Corporation's cross assembler

INPUT/OUTPUT MEDIA

- Source input: Punched cards, punched tapes, magnetic tapes, magnetic disks
- Control-command input: Punched cards
- Object code output : Magnetic disk

FUNCTION

This cross assembler converts source programs written in the MELPS 8-48 assembler language to machine-instruction codes, which are output as absolute objects.

The MELPS 8-48 cross assembler functions in two phases: control-command analyzing phase and assembly phase (intermediate-language-generation and listing phases).

The assembly-control commands listed in Table 1 are available. They cover use for execution start-up, termination assignment, I/O assignment, file assignment, link control and relocation assignment.

This cross assembler permits the use of the machine-instruction codes applicable to Intel's Models 8041, 8048 and 8049 and of the 10 pseudo instructions listed in Table 3.

CROSS ASSEMBLER

With various control commands and pseudo instructions, the MELPS 8-48 cross assembler ensures easy program debugging.

Source programs can be input by means of punched cards, punched tapes, magnetic tapes, and magnetic disks. When the control commands are read in, parameters to control assembly processing are generated by designating the assembly-control command.

In the assembly-processing stage, the source program is read in, and the intermediate language is generated in phase 1. This intermediate language and the source program are stored in the disk, and the absolute object is then produced. That can be output on punched tape, magnetic tape, magnetic disk or other media as specified.

PROGRAM ORDERING INFORMATION

Program name	Ordering No.	Program and software manuals included
MELPS 8-48 cross assembler	GC1AS0200	Source program MELPS 8-48 Assembler Language Manual GCM-SR00-01A MELPS 8-48 Cross-Assembler Manual GCM-SR00-02A MELPS 8-48 Cross-Assembler Operating Manual GCM-SR00-03A

CROSS-ASSEMBLER OBJECT LANGUAGE

The objects produced by this cross assembler basically consist of name, symbol and text sections. An end section is placed at the rear end of an object. Fig. 1 shows the object-module configuration. Each name section is placed at the head of each object module, and serves for recording information such as the name of the object module, ROM/ RAM information, and the number of symbols. The symbol sections are used to record information concerning the numeric symbols (labels) written in the source program. The text sections have data on the conversion of the source program to the instruction code. The end section specifies the termination of one object program.

ASSEMBLY LANGUAGE

Machine instructions pseudo and macro instructions can be used in the MELPS 8-48 cross assembler.

1. Machine Instructions

A total of 96 basic machine instructions are available. They are converted to their corresponding machine codes and then assembled into an object program. A classification of these instructions is given in Table 2.

For the mnemonics, instruction codes and their functions, please refer to the data sheet provided for the single-chip 8-bit microcomputers M5L 8041-XXXP, M5L8048-XXXP and M5L8049-XXXP.

2. Pseudo Instructions

Although the pseudo instructions are written in the source program together with machine instructions, they control the cross-assembler execution during assembly processing. That is, they are not converted into instruction codes to be written in the ROM but are used to control the assembler.

These instructions include those used for assembly control, numeric-symbol and memory-content definition, area securing, and list control. Table 3 lists the pseudo instructions.

3. Macro Instructions

These instructions consist of groups of several machine language instructions or simple instructions used to specify parameters. They consist of the 6 instructions MACRO, LOCAL, REPT, IRP, IRPC, and ENDM.

Fig. 1 Object-module configuration

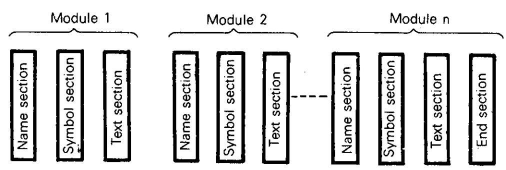


Table 1 Control Commands and their Functions

Command	Format	Function
Execution-start control	**///RUN**	Starts execution of the cross assembler in accordance with the command designated.
Execution-end control	**///END**	Terminates execution of the cross assembler.
Mode designation	**///ASMB48, X, Y, Z**	Specification of the processor for which assembly is to be performed and the ROM size MM: Mode designation MM = 48 8048 mode (including 8049) MM = 41 8041 mode X: Maximum ROM size designation X = 1 ~ 4 for 1 ~ 4K byte
Output option designation	**///OPTIN, XX, XX**	Selection of assembly listing, reference listing, and symbol listing for the object code including number of characters per line (same as ///OPTIN, LS, XL when abbreviated) XX: Output option selection XX = LS Assembly listing output XX = XL Reference listing output XX = SO Symbol object file output XX = PL 80 characters per line (132 maximum when no specifications made)
Source input device designation	**///INPUT, X〔(Y)〕**	Source input device designation (sames as ///INPUT, C when abbreviated) X: Source program input device X = C Paper card X = D Magnetic disk X = P Paper tape X = M Magnetic tape Y: Magnetic tape character code (abbreviated as ASCII code) Y = A ASCII code Y = E EBCDEC code
File designation	**///MDISK, XXXXX**	Macro source working file name (maximum of 5 characters)
	///SDISK, XXXXX	Source program file name (maximum of 5 characters)
	///BDISK, XXXXX	Absolute object program file name (maximum of 5 characters)

4. Language Format

The following free format should be used in coding programs in this cross assembler.

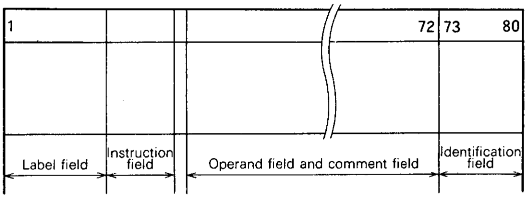
The single-line statement of the source program is composed of label, instruction, operand, comment, and identification fields. The format of the source statement is free, as shown in Fig. 2, allowing easy coding and punching without the fear of dislocated columns. The following characters can be used in statements.

- Alphanumeric .... A~Z
- Numerics .... 0~9
- Special Characters .... : ; = , ▼ @ $ + − * / ! & ( ) . # % < > ? (blank)

Table 2 A Classification of Machine Instructions

Function classification	Functions of the instruction
Data-transfer instruction	Direct data setting Between registers Between memories and registers
Adding logic operation	Addition, AND, OR, EXOR logic operations. Accumulator increase and decrease, clear and rotation shift, decimal correction
Register increase and decrease	Register increment, register decrement data-memory increment
Flag control	Carry clear, carry correction, clear-flag 0, 1 and flag 0, 1 correction
Subroutine control	Subroutine jump, return from subroutine return and status restore
Interruption control	External interruption possible External interruption prohibited Register-bank and memory-bank selection Clock-output marble
Input/output control	Between port and accumulator Port and immediate-data OR and AND Between bus and accumulator Bus and immediate-data OR and AND Between expander port and accumulator Expander-port and accumulator OR and AND
Jump instructions	Unconditional jump Indirect jump Register decrement skip Jump by carry 0, 1 Jump by accumulator 0 or non-zero Jump by T0 = 0 or 1 Jump by T1 = 0 or 1 Jump by F0 = 1 or F1 = 1 Jump at the time of timer flag Jump at the time of $\overline{INT}$ = 0 Jump by accumulator bit
Timer-counter control	Timer/counter read Timer/counter load Timer start, counter start Timer/counter stop Timer/counter interruption allowed Timer/counter interruption prohibited
Others	No operations

Fig. 2 Source-Statement Format



1. **Label field**
   The value of the program counter at that time is set to the label. The number of characters used for a label is limited to a maximum of 6. The character : is placed at the back of this field. However, a semicolon (;) cannot be used in the first column of the label field.
2. **Instruction field**
   Mnemonic codes of the machine and pseudo instructions are written in this field.
3. **Operand field**
   Arguments (formula, data, parameters, etc.) for the instructions are written in this field. The label, defined symbol, formula, or numerical value is contained within it.
4. **Comment field**
   This field is used for writing notes for the statement and is not converted to an object. Placing a semicolon (;) in the first column makes the whole statement a comment. When a semicolon (;) is placed halfway through the statement, the section after the semicolon is regarded as a comment.

Table 3 Pseudo Instructions

Classification \ Item	Mnemonic	Instruction
Assembler-control instructions	NAM	Program-name declaration instruction
	ORG	Program-counter setting instruction
	EOT	
	END	End-declaration instruction
Numeric-symbol and memory-content definition instructions	EQU	Numeric-symbol definition instruction
	SET	
	DB	Data-setting instruction
	DW	Address-setting instruction
Region-securing instruction	DS	Region-securing instruction
List-control instruction	EJE	Page-feed declaration instruction

Table 4 Macro Instructions

Instruction	Description
Macro	Macro symbol definition instruction
ENDM	End instruction for a macro definition
LOCAL	Symbol replacement instruction
REPT	Repeat instruction
IRP	Infinite repeat instruction
IRPC	Direct number infinite repeat instruction

Table 5 Expression Formats for Numeric Values, Character Constants and Formulae

Item		Expression
Numeric values	Binary	—
	Octal	**nQ**
	Decimal	**n**
	Hexadecimal	**nH**
Character constants	A (1-byte)	**"A"**
	AB (2-byte)	**"AB"**
	A'B (3-byte)	**"A " B"**
Formulae	4 Arithmetic-rule operations	**+, −, *, /**
	Logic formula	—
Others	Program counter	**$**

5. Identification field

The use of this field is optional. Many operators find it convenient to use it for the sequential identification card number.

CODING FORMAT

Programs written in the MELPS 8-48 assembler language can be coded in free formats.

General formats for using the control commands and program coding for this cross assembler are described below, together with a citation of a few examples.

1. Control Commands

1. Control-command general format

```
LABEL  | STATEMENT AND
1 2 3 4 5 6 7 8 9 10 ... 33
///XYZ, P1, P2, P3,
```

XYZ : Assembly-control-command symbols
P1, P2, P3 ··············· : Assembly-control parameters

2. Example of assembly control

```
LABEL  | STATEMENT AND
1 2 3 4 5 6 7 8 9 10 ... 33
///ASM48,L,C,N
///SDISK,XXXXX



///RUN
```

The source program is input through the card reader, and an assembly list is output.

2. MELPS 8/48 Assembler-Language Program

1. General format of program coding

```
LABEL  |                      S   IDENTIFICATION SEQUENCE
1 2 3 4 5 6 7 8 9 ... 22          72 73 ... 80
LABEL:MNEM OPE ;COMM              SEQ    000
                                         010
                                         020
```

LABEL: ········· A colon (:) is always placed behind the label name.
MNEM ··········· Instruction symbol (mnemonic)
OPE ············ Operand (1 or more blanks must always be included.)
;COMM ········· Comment (A semicolon (;) is always placed at the head)
SEQ ············ Sequential No. (columns 73-80)

2. Example of program

```
LABEL  | STATEMENT AND COMME
1 2 3 4 5 6 7 8 9 10 ... 33
;
;  ***PROGRAM EXAMPLE ***      ①
;  *DECIMAL ADDITION
       NAM   EXAM              ②
       ROM                     ③
     X EQU   10
     Y EQU   50                ④
   CNT EQU   10
       ORG   500H              ⑤
       MOV   R0,#X
       MOV   R1,#Y             ⑥
       MOV   R2,#CNT
       CLR   C     ;CLEAR CARRY ⑦
   BR: MOV   A,@R0             ⑧
       ADDC  A,@R1             ⑨
       DA    A                 ⑩
       MOV   @R0,A             ⑪
       INC   R0
       INC   R1                ⑫
       DJNZ  R2,BR             ⑬
       OUTL  P1,A              ⑭
       END                     ⑮
```

① The lines having a semicolon (;) in the first column are regarded as comments.
② The program name is declared as "EXAM" by the NAM pseudo instruction.
③ The following lines are regarded as a ROM region.
④ The decimal numbers 10, 50, and 10 are assigned respectively to symbols X, Y, and CNT.
⑤ The program start address is address 500 in hexadecimal notation.
⑥ The values #X, #Y, and #CNT are respectively put into registers R0, R1, and R2.
⑦ The carry is cleared.
⑧ The contents of label BR memory at R0 (at the address to be jumped to; the colon (:) shows a label) are put into accumulator A.
⑨ The contents of the carry and data memory at R1 are added to each other, and put into the accumulator.
⑩ The accumulator contents are decimal-corrected.
⑪ The accumulator contents (decimal-corrected results of the addition) are put into the memory data at R0.
⑫ The contents of registers R0 and R1 are incremented.
⑬ Register R2 is decremented and if the contents are not 0 (zero), branching to BR follows. If 0, execution proceeds to next step.
⑭ The contents of the accumulator are output in port 1.
⑮ The end of program is declared.

PAPER-TAPE GENERATION PROGRAM FOR PROM WRITERS

DESCRIPTION

This program is used to convert absolute binary object formatted programs, which are produced by the MELPS 8-48 cross assembler, into other language formats and then produce a paper tape that can be used as input for a PROM writer.

The functional configuration of this program offers automatic conversion of object programs from one format to another format as well as comparison processing. In addition, it provides extensions suitable to various applications.

FEATURES

- It selectively produces partitioned, punched paper tapes with simple control commands.
- It converts MELPS 8 binary object programs stored on disks into various hexadecimal formats on paper tape.
- It converts various hexadecimal-format paper tapes into MELPS 8 hexadecimal format.
- Comparison-matching control functions for MELPS 8 hexadecimal format paper tape as well as other formats.
- Output of various block sizes as specified by the block size (i.e., paper-tape partition) parameter.
- Sorting capability to put files in address sequence.
- Execution computer: MELCOM 70 minicomputer (memory capacity: more than 24K-words; program: about 5,000 steps).
- Implementation language: FORTRAN IV (parts are written in assembler language).

INPUT/OUTPUT MEDIA

- Conversion of MELPS 8 binary to hexademimal paper tapes
  Input: cartridge disk units
  Output: paper tapes (even-parity ASCII code)
- Conversion of other hexadecimal paper tapes to MELPS 8 hexadecimal paper tapes
  Input: paper tapes in other hexadecimal format (even-parity ASCII code)
  Output: paper tapes in MELPS 8 hexadecimal format (even-parity ASCII code)
- Comparison of MELPS 8 hexadecimal with other hexadecimal paper-tape formats and self comparison
  Input: paper tapes (even-parity ASCII code)
  Output: printed on system typewriter
- Control-command input
  Through system-typewriter keyboard

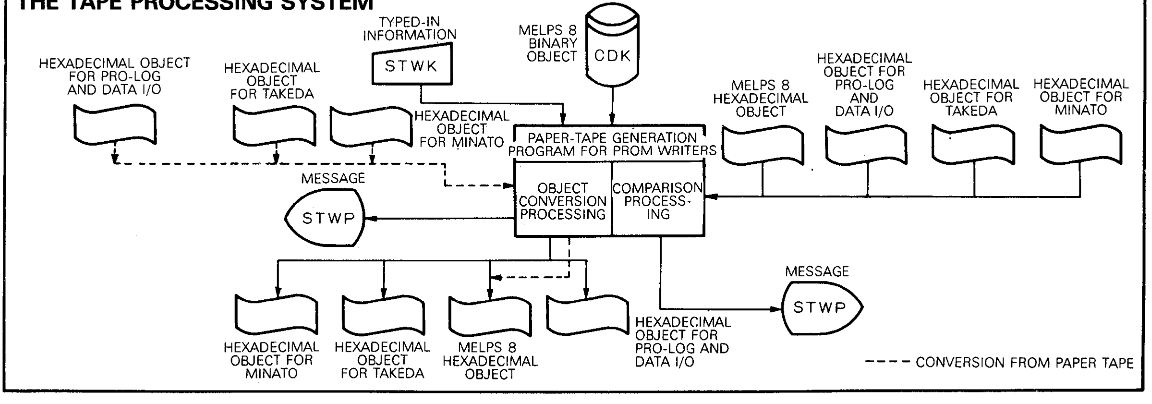
APPLICATIONS

Programs are applicable to the M5L2708K and -S (1K-word by 8-bit), M5L2716K (2K-word by 8-bit), and other similar ROMs when prepared by a PROM writer produced by Takeda Riken, Minato Electronics, Pro-log, and Data I/O.

FUNCTION

This program converts absolute binary object programs (abbreviated MELPS 8 binary), created in the disk area by the MELPS 8-48 cross assembler, into hexadecimal object programs. These hexadecimal object programs can be used to program PROMs on PROM writers produced by Takeda Riken (T310), Minato Electronics (Type 1830), Pro-log Ltd. (Series 90), and Data I/O (abbreviated hereafter as Takeda, Minato, Pro-log, and Data I/O). This program also converts absolute binary object programs into the MELPS 8 hexadecimal format and creates paper tapes with blocks of suitable size. The program can also convert paper tapes of Takeda, Minato, Pro-log and Data I/O into MELPS 8 hexadecimal format and compare the object paper tapes.

THE TAPE PROCESSING SYSTEM



PROGRAM ORDERING INFORMATION

Program	Program code no.	Program and software manuals included	
Paper-tape preparation program for MELPS 8/48 PROM writers	GA1SP0110	Paper-Tape Preparation Program for MELPS 8/48 PROM Writers Manual	GAM-SR00-50A

PAPER-TAPE GENERATION PROGRAM FOR PROM WRITERS

PAPER-TAPE PROCESSING

The program provides for both conversion and comparison of various object programs. Table 1 shows a summary of the conversion processing indicating various combinations of object programs and media that the program is capable of processing. Table 2 shows a summary of the comparison processing indicating the various combinations of object programs and media that the program is capable of processing. Examples of all the object conversions listed in Table 1 are illustrated in Fig. 1.

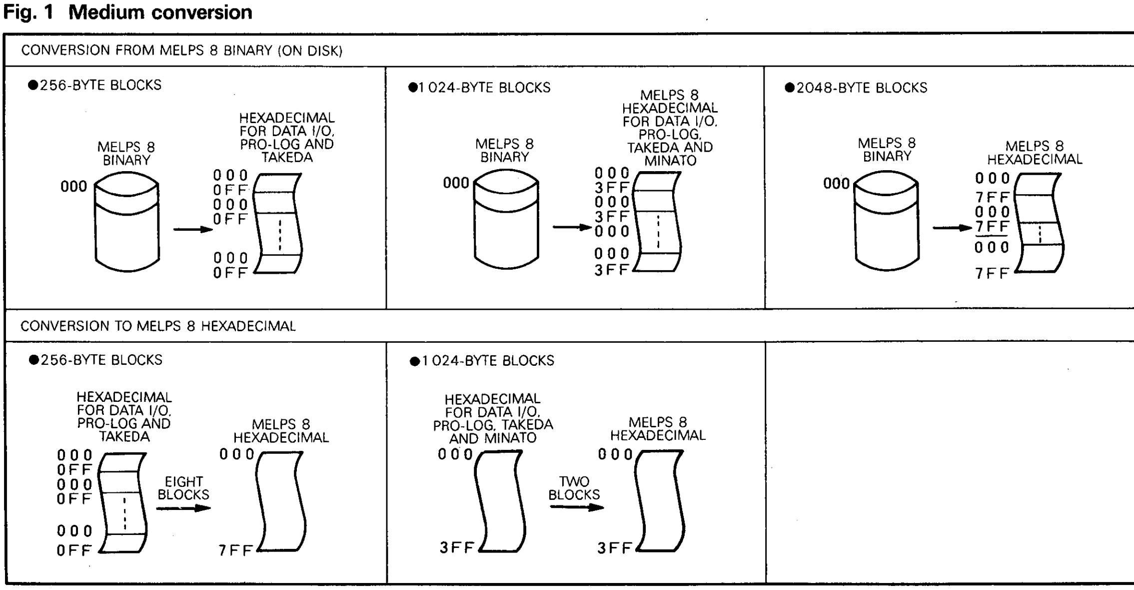
Table 1 Object conversions

Conversion processing for each company's PROM writer / Paper tape block size	Hexadecimal paper tapes for PROM writers that can be converted from MELPS 8 binary (on disk)	Hexadecimal paper tapes for PROM writer that can be converted into MELPS 8 hexadecimal paper tape
256 bytes	Data I/O, Pro-log, Takeda	Conversion from eight blocks of Data I/O, Pro-log or Takeda to one 2048-byte block
1024 bytes	Data I/O, Pro-log, Takeda, Minato, TDA-80	Conversion from one block of Data I/O, Pro-log, Takeda or Minato to one 1024- or 2048-byte block
2048 bytes	MELPS 8 hexadecimal (for mask ROM)	

Table 2 Comparison processing of object paper tapes

Objects compared / Comparison	MELPS 8 hexadecimal		Comparison object	
	Object	Media	Object	Media
MELPS 8 hexadecimal self comparison	MELPS 8 absolute hexadecimal	Paper tape ●1 024-byte block ●2048-byte block	MELPS 8 absolute hexadecimal	Paper tape ●1 024-byte block ●2048-byte block
Comparison of MELPS 8 hexadecimal with Minato hexadecimal	MELPS 8 absolute hexadecimal	Paper tape ●1 024-byte block ●2048-byte block	Hexadecimal for Minato	Paper tape ●1 024-byte block ●Two 2048-byte blocks
Comparison of MELPS 8 hexadecimal with Takeda hexadecimal	MELPS 8 absolute hexadecimal	Paper tape ●1 024-byte block ●2048-byte block	Hexadecimal for Takeda	Paper tape ●Eight 256-byte blocks ●One 1 024-byte block ●Two 1 024-byte blocks
Comparison of MELPS 8 hexadecimal with Pro-log hexadecimal	MELPS 8 absolute hexadecimal	Paper tape ●2048-byte block	Hexadecimal for Pro-log	Paper tape ●Eight 256-byte blocks ●Two 1 024-byte blocks
Comparison of MELPS 8 hexadecimal with Data I/O hexadecimal	MELPS 8 absolute hexadecimal	Paper tape ●2048-byte block	Hexadecimal for Data I/O	Paper tape ●Eight 256-byte blocks ●Two 1 024-byte blocks

Fig. 1 Medium conversion



PL/1μ CROSS COMPILER

DESCRIPTION

This cross compiler is supplied on magnetic tape to users of MELPS 8/85 CPUs. It is written in FORTRAN IV for execution of the MELCOM 7000 and can be easily run on other host computers with a FORTRAN IV compiler.

The PL/Iμ language gives MELPS 8/85 microcomputer users the same advantages that users of mini and large computer systems have with the high level programming languages that are currently available. It has the same language structure as PL/I and has been designed to take advantage of the system architecture of the microprocessor. System designers can use PL/Iμ to quickly and easily implement new applications. In addition, programs written in PL/Iμ are self-documenting; so they can be easily changed and maintained. PL/Iμ is recognized as one of the best suited languages for programming microcomputer applications because the user retains the control and efficiency of an assembly language.

FEATURES

Of the PL/Iμ Cross Compiler

- Conditional compile with preprocessor
- Inline assembly
- Source program editing at compile time
- Assignment of programs to ROM or RAM regions
- Generates a relocatable object program
- Linking function
- Easily understood error messages
- Flexibility in input/output media
- Execution computer: MELCOM 7000 (UTS/VS)
- Implementation computer: MELPS 8/85 microcomputer
- Implementation language: FORTRAN IV

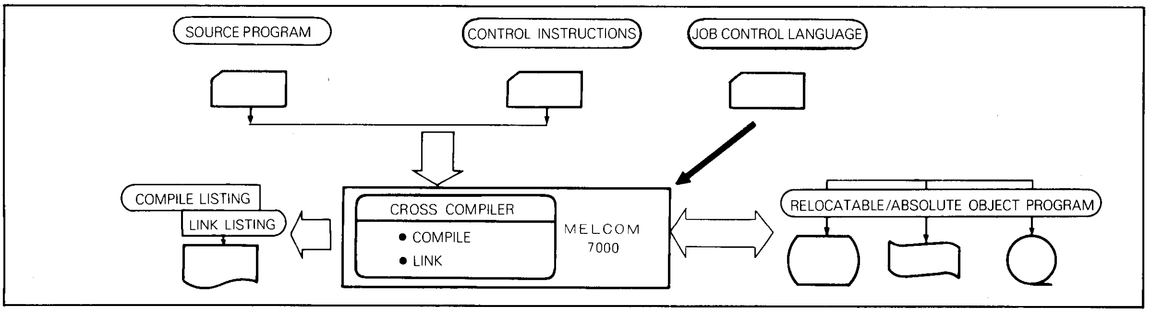
Of the PL/Iμ Language

- Bit operations
- Three-level structure
- One-dimensional arrays
- Allocation of variables to specified absolute addresses
- Multi-entry function
- Interrupt function

FUNCTION

PL/Iμ has a preprocessor that allows user to modify programs under development at compile time through the use of conditional compile, exchange, exclude and include functions. A program is divided into fixed and variable segments, and these segments are automatically assigned to the appropriate memory (RAM or ROM) during compiling. The link editor can link up to 20 object programs (files).

Fig. 1 PL/Iμ cross compiler processing system



PROGRAM ORDERING INFORMATION

Program name	Ordering number	Program and software manuals included	
MELPS 8/85 PL/Iμ cross compiler	GAITL0110	Source Program	
		MELPS 8/85 PL/Iμ Compiler Summary Manual (C-version)	GAM-SR00-07A
		MELPS 8/85 PL/Iμ Compiler Language Manual (C-version)	GAM-SR00-08A
		MELPS 8/85 PL/Iμ Cross Compiler Operating Manual (C-version)	GAM-SR00-09A
		MELPS 8/85 MELCOM 7000 PL/Iμ Cross Compiler Operating Manual	GAM-SR00-10A

MANUALS

Manual name	Manual number
MELPS 8/85 PL/Iμ Compiler Summary Manual (C-version)	GAM-SR00-07A
MELPS 8/85 PL/Iμ Compiler Language Manual (C-version)	GAM-SR00-08A
MELPS 8/85 PL/Iμ Cross Compiler Operating Manual (C-version)	GAM-SR00-09A
MELPS 8/85 Assembly Language Manual (A-version)	GAM-SR00-34A
MELPS 8/85 Cross Assembler Operating Manual (A-version)	GAM-SR00-02A
MELPS 8/85 Simulator Operating Manual (B-version)	GAM-SR00-35A
MELPS 8 Hardware Manual	GAM-HR00-01A

14

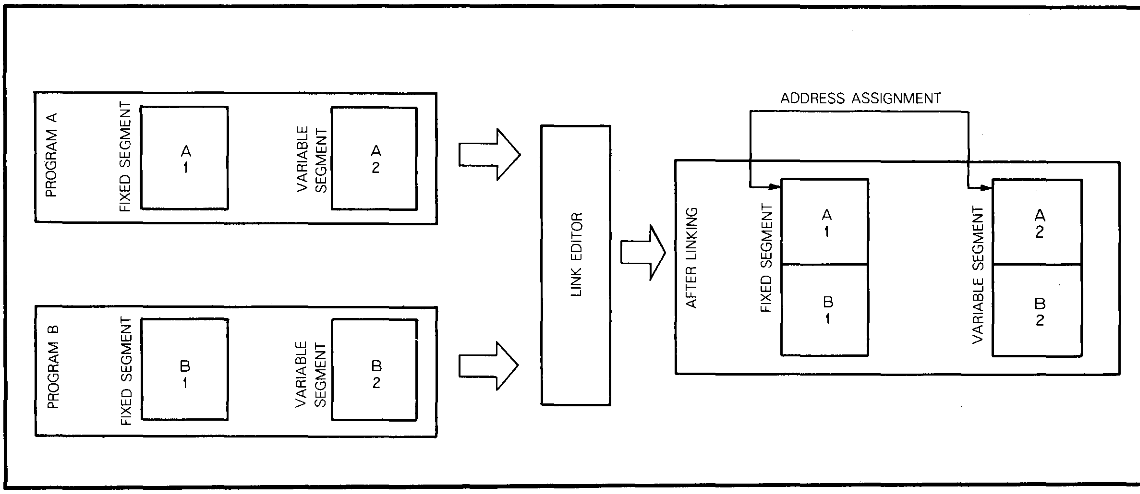
OPERATIONS

Users of PL/Iμ will find it flexible and easy to use because of its many special features such as the preprocessor, the link editor and the memory manager.

The preprocessor has 10 statements that can be used at compile time to edit a PL/Iμ source program. These can generate, exchange or delete program text, as well as modify definitions, references and macro instructions.

The link editor is able to link up to 20 object programs that have been generated by MELPS 8/85 software. The memory manager divides PL/Iμ programs into fixed and variable segments and assigns the segments to the appropriate memory. A fixed segment is assigned to a non-write area (ROM) while a variable segment is assigned to a write area (RAM) during compiling; at the same time, the starting address of each segment is recorded for linking (see Fig. 2).

Fig. 2 Linking of two programs



PL/I μ LANGUAGE

The PL/Iμ language is a subset of the popular PL/I language with the addition of special functions to take advantage of the microprocessor's architecture. The main features of the PL/Iμ language are as follows:

Easy to Read and Write

The statements are written in free format and are independent of columns and lines. The statements are formatted in natural language. It is easy to express, read and understand the programs. Programs written in PL/Iμ are self-documenting.

Block-Structured Language

Programs written in PL/Iμ consist of one or more blocks that are called procedures. A procedure (block) can be thought of as a subroutine. The block structure of PL/Iμ simplifies modular programming. Each procedure can be conceptually simple and therefore easy to formulate and debug.

BASIC LANGUAGE SPECIFICATIONS

1. Statements

The basic unit of the PL/Iμ language is called a statement. A procedure (block) is composed of one or more statements, and a program is composed of one or more procedures. The statements are categorized as follows:

Statements	– Procedure definition:	PROCEDURE statement
	– Declaration:	DECLARATIVE statement
	– Condition:	IF statement
	– Non-condition:	Assignment statement, DO group, and others

The last character of a statement must be a semicolon (;). A statement may have a label (identifier) that is the name of the statement.

Example **EXAMPLE: X=Y+Z;**

2. Identifiers

PL/Iμ identifiers are used to name variables, procedures, macro instructions and statements. An identifier may be up to 31 characters in length, and the first character must be an @, ? or alphabetic (A~Z) character. The remaining 30 characters may be alphanumeric (A~Z, 0~9), @ or ?.

Reserved words may not be used as identifiers in the PL/Iμ language.

3. Data Elements

The PL/Iμ data elements represent constants or variables (1~16 bits in length), arrays (1 dimension) and 3-level structure. Constants can be expressed in several different ways in PL/Iμ. PL/Iμ accepts constants in binary, octal, decimal and hexadecimal bases and character strings (ASCII or ISO code).

Example of a PL/Iμ program

```
     /*  THIS IS A SAMPLE FOR A CATALOG  */ ..................... ①
       DECLARE I          BINARY (7),
                FOREVER BINARY (7)  INITIAL (1); ................ ②

            /*  IF VARIABLE FOREVER IS TRUE, THE STATEMENTS
                UP TO THE CORRESPONDING END ARE EXECUTED   */

          DO WHILE FOREVER; ..................................... ③
            /*  READ INPUT PORT 10 AND SAVE IN VARIABLE I   */

             I=INPUT (10); ...................................... ④
            /*  THIS VALUE I IS USED TO SELECT ONE OF
                THE STATEMENTS OF DO-CASE TO EXECUTE  */

             DO CASE I ; ........................................ ⑤

            /*  I=0  */

             DO;

            /*  WRITE 8 AT OUTPUT PORT 5 AND HALT  */

             OUTPUT (5)=08H; .................................... ⑥
             HALT;
             END;

            /*  I=1  */

             OUTPUT (5)=80H;

            /*  I=2  */

             OUTPUT (5)=40H;
          END;

       END;
```

①. Comments are preceded by '/*' and followed by '*/'.

②. The initial value of a type declared variable 'FOREVER' is 1.

③. DO-WHILE group.

④. The device number of an input instruction is expressed using a number.

⑤. DO-CASE group.

⑥. 08H used in the output instruction indicates a hexadecimal number of value 08_{16}.

14

LANGUAGE SPECIFICATIONS

Item	Specification
Character set	55-character set Alphabetic: **A～Z**, Currency unit (**$**), Numeric: **0～9** Special: = + − * / , . : ; < > % ' ( ) @ ? (blank)
Comments	/* */
Identifiers	31 or less alphanumeric characters
Reserved words	**IF DO GO TO OR BY ON EOF END XOR AND NOT MOD HALT THEN ELSE CASE CALL GOTO DATA BYTE PLUS MAIN LABEL BASED MINUS WHILE ENTRY ENABLE RETURN BINARY DISABLE DECLARE ADDRESS INITIAL ALIGNED OPTIONS INTERNAL EXTERNAL RELOCATE GENERATE INTERRUPT PROCEDURE LITERALLY UNALIGNED**
Constant types	Binary, octal, decimal, hexadecimal character string
Variable declaration option	**BINARY(n)** $1 \leqq n \leqq 15$, **BIT(m)** $1 \leqq m \leqq 16$ **LABEL INITIAL BASED DATA BYTE ADDRESS EXTERNAL INTERNAL ALIGNED UNALIGNED**
Operators	* / **MOD** + − **PLUS MINUS** < <= <> = >= > **NOT AND OR XOR**
Arrays	One-dimensional, 1～255 elements
Structures	Three-level, array structure
Expressions	Arithmetical expression, logical expression, structured expression
Statements	Insert statement, **GOTO** statement, **IF** statement, **CALL** statement, **GENERATE** statement, **RETURN** statement, **HALT** statement, **DECLARE** statement, **ON** statement, **PROCEDURE** statement, **DO** group, **ENTRY** statement, **NULL** statement, **RELOCATE** statement, **ENABLE** statement, **DISABLE** statement,
DO group	**DO WHILE**, repeat **DO**, **DO CASE**
Library functions	**TIME MEMORY SHL SHR ROL ROR INPUT OUTPUT DEC HIGH LOW LENGTH LAST CARRY ZERO SIGN PARITY**
Preprocessor statements	% insert statement, %**ACTIVATE** statement, %**DEACTIVATE** statement, %**END** statement, %**EXCLUDE** statement, %**GOTO** statement, %**IF** statement, %**INCLUDE** statement, %**MACRO** statement, %**NULL** statement

SIMULATOR

DESCRIPTION

A pseudo CPU and a pseudo memory are modeled in the host computer by the simulator, and programs in the pseudo memory are executed by the pseudo CPU to debug and test programs.

The simulator contains a powerful set of 26 control commands for efficient program debugging.

FEATURES

- Set of 26 powerful control commands
- Batch and conversational processing
- Symbolic addressing
- Execution time calculations
- Intermediate results saved in specified format
- Binary, octal, decimal and hexadecimal numbers are selectable
- Assignment of program segments to ROM or RAM region
- Memory protection
- Interrupt function
- Flexibility in input/output media
- Continuous processing of input/output data
- Execution minicomputer: MELCOM 70 (memory capacity more than 24K words, monitor BDOS)
- Programming language: FORTRAN IV (parts are written in assembly language)

FUNCTION

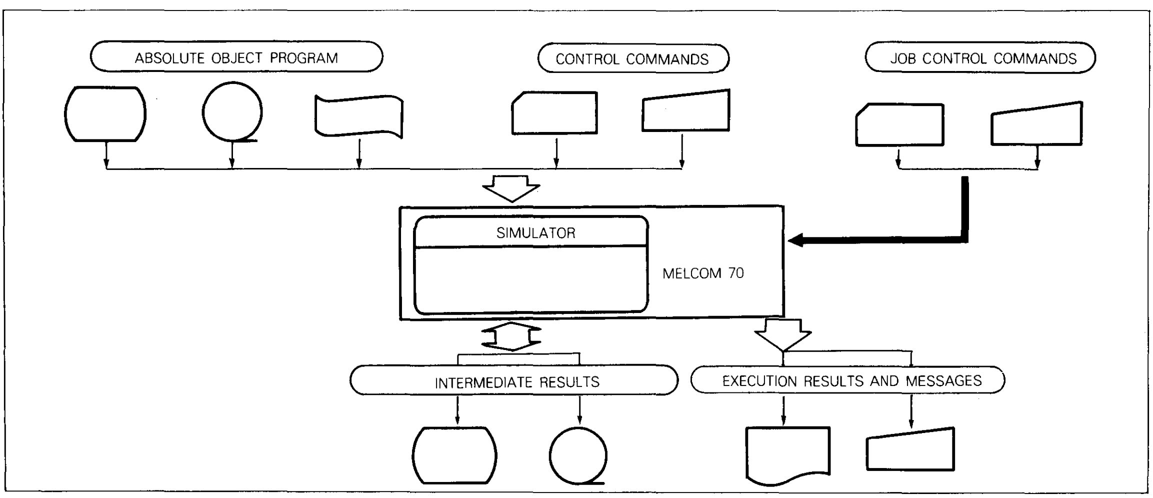
The trace command function assigns a specific trace region so that it traces only the specified program steps. Execution of the simulation can be halted by a breakpoint that can be assigned to any location. Program debugging efficiency can be expected to increase by the use of these functions.

Memory protect and ROM regions are simulated. This means the simulator will not allow writing in a ROM region and will not allow either reading or writing in a memory protect region. Therefore, the program under simulation is completely simulated, including the state of the memory in the object computer system.

Input/output media

- Object program input: Paper tape, magnetic tape and magnetic disk
- Control command input: Punched card and keyboard
- Simulation intermediate results output: Magnetic tape and magnetic disk
- Simulation result output: List
- Input/output data: Punched card, keyboard, paper tape and magnetic tape

SIMULATOR PROCESSING SYSTEM



PROGRAM ORDERING INFORMATION

Program name	Ordering number	Program and software manuals included
MELPS 8/85 simulator (B-version)	GA1SM0110	Source Program MELPS 8/85 Simulator Operating Manual (B-version) GAM-SR00-35A MELPS 8/85 Cross Assembler & Simulator Operating Manual (on MELCOM 70) GAM-SR00-04A

MANUALS

Manual name	Manual number
MELPS 8/85 Assembly Language Manual (A-version)	GAM-SR00-34A
MELPS 8/85 Cross Assembler Operating Manual (A-version)	GAM-SR00-02A
MELPS 8/85 Simulator Operating Manual (B-version)	GAM-SR00-35A
MELPS 8 Hardware Manual	GAM-HR00-01A

CROSS ASSEMBLER FUNCTIONS

The control commands and pseudo instructions in this cross assembler give the user flexibility and improve the efficiency of programming. The cross assembler allows linking, multi-assembly and conditional assembly.

The control commands are shown in Table 1, and the features and their limitations are shown in Table 2.

Table 1 List of control commands

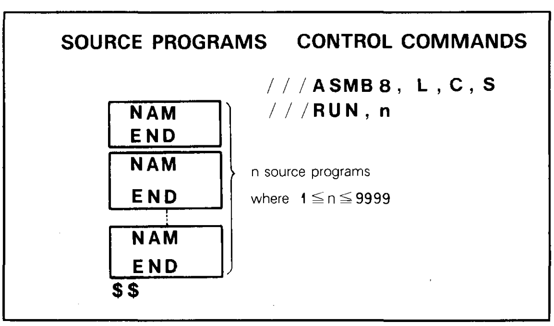
Classification		Control command name	Mnemonic
Assembler control		Execution start	RUN
		End	END
Execution control	Assembly control command	Input/output assignment	ASMB8
		Block assignment	BLOCK
		File assignment	SDISK
			ODISK
			BDISK
	Link control command	Link assignment	LINKG
		Link location assignment	LKLOC

Table 2 Cross assembler features and their limitations

Features	Limitations
Relocatable object programs	
Link editor	Maximum 20 programs on the disk
Program segmented to non-write area (ROM) and write area (RAM)	
Multi-assembly	Maximum 9999 programs
Conditional assembly	Maximum 20 blocks
Flexibility in I/O media selection	Card, disk, paper tape, magnetic tape

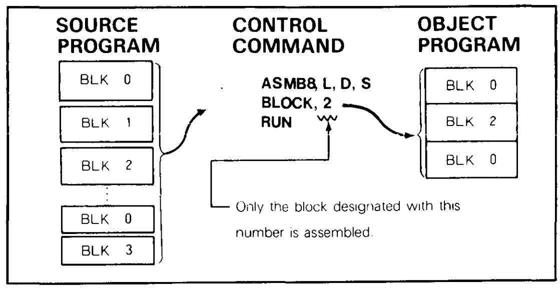
1. Multi-Assembly

Many programs can be batch-assembled in one run.



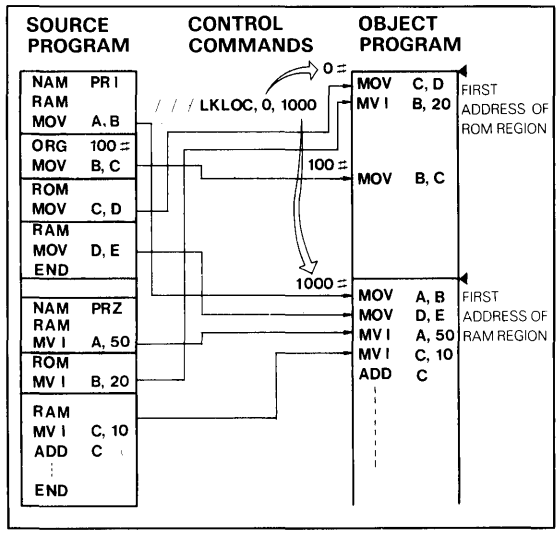
2. Conditional Assembly

Only the designated blocks of a source program are assembled.



3. Linking of ROM/RAM Regions

ROM and RAM regions are linked separately.



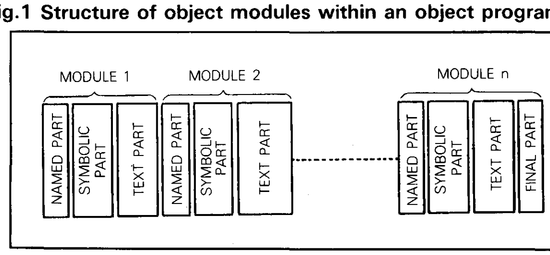
CROSS-ASSEMBLER OBJECT PROGRAM

The cross-assembler object program is composed of many object modules, and each module is composed of a name, a symbolic part and a text part. A final part ends each object program.

The symbolic part contains the symbolic name corresponding to symbols. It is possible to program using symbolic names because each module contains a symbolic part.

The object is composed of an 8-bit binary code, and one byte of the instruction code is expressed with one character (8 bits).

Fig.1 Structure of object modules within an object program



ASSEMBLY LANGUAGE FUNCTIONS

The assembly language consists of mnemonic instructions (each corresponding to a machine language instruction), pseudo instructions and macro instructions.

Pseudo instructions are executed by the cross assembler when a source program is being assembled, and they modify the object program. Macro instructions are converted to small segments of machine instructions that are then inserted in the object program. These inserted segments execute the functions of the macro instruction.

Algebraic expressions, alphanumeric constants, character strings, octal numbers, decimal numbers, hexadecimal numbers and symbols may be used as an operand in instructions.

1. Machine Instructions

There are 78 basic machine instructions. These are converted to their corresponding machine language instructions and then inserted in the object program.

A summary of the machine instructions is given in Table 3.

Table 3 Summary of machine instructions

Classification	Instruction functions
Data transfer instructions	Direct data set Between registers Between memory and registers
Addition, subtraction, logical operations and compare instructions	Addition, subtraction, comparing and logical operations using the accumulator together with registers, memory or carry flag
Increment and decrement instructions	Registers, register pairs and memory incremented or decremented
Circulate and shift instructions	Circulate or shift the accumulator's contents
Accumulator adjust instructions	Complement, decimal adjust
Carry instructions	Complement, set
Jump instructions	Unconditional jump Conditional jump
Subroutine call instructions	Unconditional subroutine call Conditional subroutine call
Return instructions	Unconditional return Conditional return
Input/output control instructions	Input and output control
Interrupt control instructions	Enable interrupts Disable interrupts
Stack operation instructions	Saves the contents of registers Restores the contents of registers
Others	CPU halt No operation.

2. Pseudo Instructions

Pseudo instructions control the execution of the cross assembler while source programs are being assembled. They are not assembled as instructions in the object programs. As shown in Table 4, there are 13 pseudo instructions.

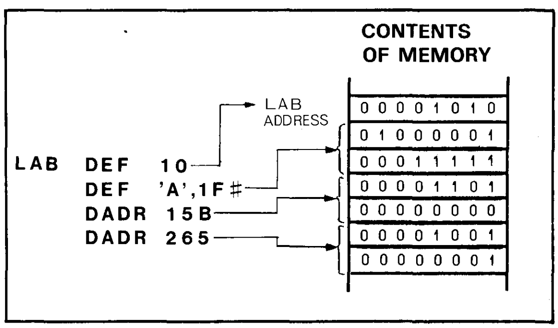
Table 4 List of pseudo instructions

Classification	Mnemonic symbol	Name
Assembler control instructions	NAM	Program name declaration
	ORG	Program counter setting
	ROM	ROM region declaration
	RAM	RAM region declaration
	BLK	Block declaration
	END	End declaration
Link symbol assignment instructions	ENT	Entry name declaration
	EXT	External reference symbol declaration
Memory contents definition instructions	EQU	Value symbol setting
	DEF*	Data setting
	DADR*	Address setting
Storage allocation instructions	BSS**	Storage allocation
List control instructions	EJE	Page eject declaration

* DEF and DADR pseudo instructions set the data or the address in the memory location where the instruction is. See Fig. 2.

**BSS pseudo instruction sets the program counter to the value of the operand.

Fig. 2 Example of DEF and DADR pseudo instructions



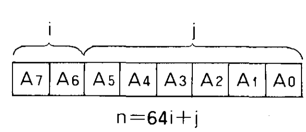
3. Macro Instructions

Macro instructions are converted to object program segments in machine language that executes the macro instruction functions. The following two macro instructions are included in this cross assembler.

Table 5 Macro instructions

Instructions	Name	Corresponding statement
GET i , j	Data input instruction	**IN n**
PUT i , j	Data output instruction	**OUT n**

where



n=64i+j

CODING EXAMPLES

Examples of coding using control commands and the assembler language of the cross assembler follow.

1. Control Commands

1. Control commands are in the following general form:

```
///Symbol [, Parameter List]
```

2. Two source programs are read in from the card reader, and the assembly lists are printed.

```
///ASMB85, L, C, S
///RUN, 2
```

3. Four object programs (files) F11, F12, F13 and F14 on the disk are linked together, and a relocatable object program is generated and filed in RF11 on the disk.

```
///LINKG, R, N, D, F11, F12, F13, F14
///BDISK, RF11
///RUN
```

2. Assembly Language

1. A statement is of the following general form:

```
[Symbol]␣Mnemon␣[Operand]
```

where ␣ indicates a blank, and [ ] defines a field that is optional.

2. This example evaluates the data in address INDATA against the table at address TA01. It then jumps to the appropriate processing program according to the evaluation. The first address of the corresponding processing program is located at address SENS.

```
*                                          }
*  ***  EXAMPLE  OF  CODING  ***           } ①
*                                          }
        NAM     PROM                         ②
        EXT     TAB1, TAB2, TAB3           }
        EXT     TAB4, TAB5, TAB6           } ③
        ENT     MAIN                         ④
        ROM                                  ⑤
MAIN    EQU     *                            ⑥
        LXI     H, TA01                      ⑦
        MVI     C, 0
        MVI     B, 6B                        ⑧
        LDA     INDATA
LOOP1   CMP     M
        JZ      LOOP2
        INX     H
        INR     C
        DCR     B
        JNZ     *-7                          ⑨
        JMP     1000    ERR, SHORT           ⑩
*
LOOP2   MVI     H, 0
        MOV     L, C
        LXI     B, SENS
        DAD     H
        DAD     B
        PCHL
*
        RAM                                  ⑪
INDATA  DEF     45#                          ⑫
*
        ROM
TA01    DEF     'MELPS8'                     ⑬
SENS    DADR    TAB1                         ⑭
        DADR    TAB2
        DADR    TAB3, TAB4
        DADR    TAB5, TAB6
*
        END
```

① An asterisk in the first column indicates that the entire statement is a comment.
② The program name is declared as 'PROM'.
③ The external programs referenced by this program are declared.
④ The external programs that reference this program are declared.
⑤ The program segment from here to the next RAM pseudo instruction is regarded as a ROM region.
⑥ The symbol MAIN refers to the value in the program location counter at this source program statement.
⑦ Locations can be referred to by symbols.
⑧ Octal numbers can be used.
⑨ Expressions can be used in the operand field.
⑩ The statement following a blank after an operand field is a comment.
⑪ Declares the start of a RAM region.
⑫ Hexadecimal numbers can be used.
⑬ Character constants in ASCII code can be used.
⑭ The address of symbol TAB 1 is set to the location of address SENS and SENS+1.

CROSS ASSEMBLER

DESCRIPTION

This cross assembler is used to convert source programs in assembly language to object programs in MELPS 8/85 format (8-bit binary format) on a host computer. The assembly language consists of mnemonic instructions (each mnemonic instruction corresponds to a machine language instruction), pseudo instructions and macro instructions. It is obvious that the assembly language makes programming and modification of programs easy. The pseudo instructions and control commands in this cross assembler give the user flexibility and improve programming efficiency.

FEATURES

Of the Cross Assembler

- Generates a relocatable object program
- Linking function
- Multi-assembly
- Conditional assembly
- Flexibility in input/output media
- Output of symbolic table of the object program
- Execution computer: MELCOM 70 (memory capacity more than 24K words, monitor BDOS)
- Implementation language: FORTRAN IV (parts are written in assembly language)

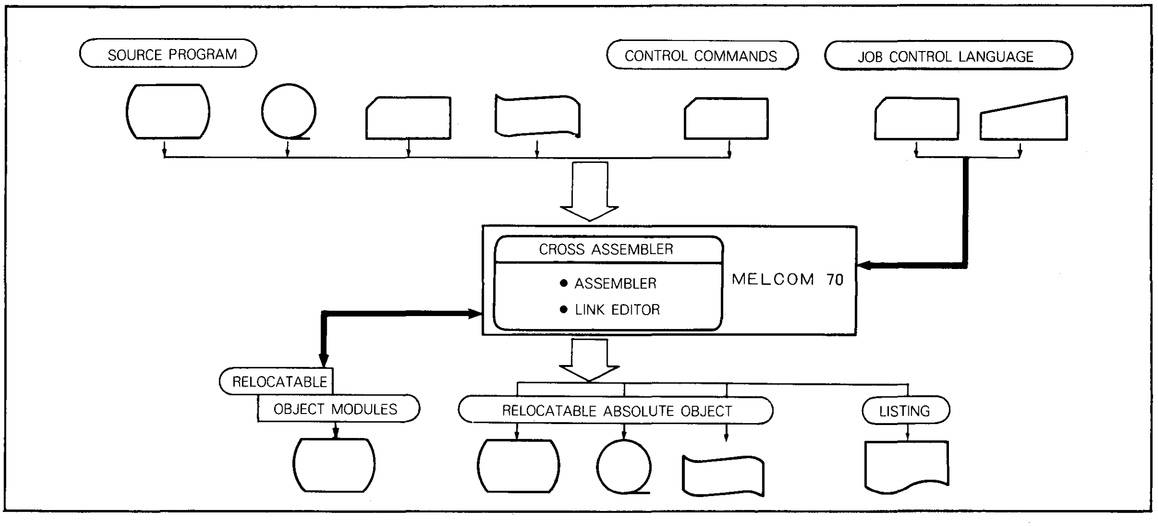
Of the Assembly Language

- 13 pseudo instructions
- Algebraic expressions
- Character constants and strings
- Octal, decimal and hexadecimal numbers
- The mnemonic codes of the machine instructions are the same as Intel's

INPUT/OUTPUT MEDIA

- Source input: Punched card, paper tape, magnetic tape and magnetic disk
- Object input: Magnetic disk
- Control command input: Punched card
- Object output: Paper tape, magnetic tape and magnetic disk

CROSS ASSEMBLER PROCESSING SYSTEM



PROGRAM ORDERING INFORMATION

Program name	Ordering number	Program and software manuals included
MELPS 8/85 cross assembler	GA1A0110	Source Program MELPS 8/85 Assembly Language Manual (A-version) MELPS 8/85 Cross Assembler Operating Manual (A-version) MELPS 8/85 Cross Assembler & Simulator Operating Manual (on MELCOM 70)

MANUALS

Manual name	Manual number
MELPS 8/85 Assembly Language Manual (A-version)	GRM-SR00-34A
MELPS 8/85 Cross Assembler Operating Manual (A-version)	GAM-SR00-02A
MELPS 8/85 Simulator Operating Manual (B-version)	GAM-SR00-35A
MELPS 8 Hardware Manual	GAM-HR00-01A

METHOD OF CODING CONTROL COMMANDS

The input formats for control commands are shown in Fig. 1.

Fig. 1 Input formats for control commands

Column no.	1					72	73 80
Contents	Blank	Command	Blank	Parameter list	Blank	Comment	Sequence number
No. of columns	1 or more columns	The number of characters in the command	1 or more columns	The number of characters in the parameter list	1 or more columns	Free	8 columns
Remarks			The command, parameter list and comment must be less than 73 columns.				Not required if the command is typed in from the system typewriter

CONTROL COMMANDS

The simulator includes 26 control commands as shown in Table 1.

Table 1 List of control commands and their functions

	Functions \ Item	Control commands: Action	Control commands: Mnemonic command	Comments
Simulator control commands	Start	Start simulation	START	Starts simulation and designates the input unit for control commands.
		Reinitialize	REINIT	Sets the state to the same state it was after the START command execution was completed.
	End	End simulation	END	Returns to the monitor when executed during simulation.
	Program loading or saving intermediate results	Load object program	LOAD	The absolute object program or the saved intermediate partially executed program is loaded.
		Save intermediate results	SAVE	All information such as executed commands, contents of registers and flags, and so forth, are saved in external memory.
	Changing control command input unit	Changes to card reader	BATCH	The command input unit is changed to the card reader.
		Changes to system typewriter	TYPE	The command input unit is changed to the system typewriter.
Executive control commands	Start	Starts execution of the object program	GO	The stop point can be designated by either an address or the number of instructions to be executed.
		Starts execution of the object program	RUN	Continues execution until a HLT instruction is encountered.
	Stop	Assigns a breakpoint	BREAK	A breakpoint is assigned by an address or a range.
		Releases an assigned breakpoint	NOBREAK	A breakpoint assigned is released.
		Steps	STEP	Breakpoints are assigned after every specified number of machine instructions.
	Assigning memory regions	Assigns a ROM region	ROM	It is declared that region assigned with this command is the ROM region.
		Releases an assigned ROM region	NOROM	The assigned ROM region is released.
		Assigns a memory protection region	PROT	A memory protect (unaccessible) region is assigned.
		Releases an assigned memory protect region	NOPROT	An assigned memory protect region is released.
	Trace	Assigns a trace region	TRACE	Printing out the contents of registers, the program counter and flip-flops along with the executed instruction codes while executing the instructions in a trace region.
		Releases an assigned trace region	NOTRACE	The assigned trace region is released.
	Set data		SET	Registers, stack pointers, program counter, flag flip-flops, I/O ports and the contents of memory are set.
	Interrupt		INTER	If interrupt is enabled, within 3-byte instruction associated with this command is executed.
	Counts the number of cycles		TIME	Counts the total number of cycles of the machine instructions executed before this command is encountered.
	Printing out	Assigns a base	BASE	A base for printing is assigned.
		Prints out	DISPLAY	The contents of registers, stack pointers, program counter, flag flip-flops, I/O ports, and memory are printed according to the assigned base.
	Conversion of values		CONV	The current program counter or the assigned value is printed out in binary, octal, decimal or hexadecimal.
I/O commands	Input/output simulation	Input simulated	IP	Defines an input string for a machine instruction IN.
		Output simulated	OP	Defines an output string for a machine instruction OUT.

Note 1 : The underlined part of the mnemonic command can be used as a short mnemonic.
2 : The control command 'START' is the first command, and its input unit must be the card reader.

EXAMPLE OF SIMULATION

The program shown in Fig. 2 is simulated using the control command in the sequence shown in Table 4. The program in Fig. 2 is named 'CON102'. It converts a decimal integer (0~65,535) to a binary number.

The decimal number to be converted is stored in addresses DED1~DED5 in ASCII code, and the converted result is stored in addresses BID and BID+1 (see Table 2). Further, if characters other than 0~9 are found in addresses DED1~DED5, the A register is set to '1' as an error flag; and if the converted result is more than 65,535, the carry flip-flop is set to '1' as an error flag.

The simulation is executed in three segments as follows:

1. The test values are set in memory addresses DED1~DED5.
2. The program is executed.
3. The simulator confirms that the contents of addresses BID and BID+1 are the correct value for the conversion of data in addresses DED1 (address 9113)~DED5 (address 9117). At the same time, it confirms that the contents of register A and the carry flip-flop are correct.

The objective program listing is shown in Fig. 2, and explanations of the simulation control commands using this example are shown in Table 4.

Table 2 Memory location and contents

Address	Contents	Explanation of contents
DED1	a	The 5-digit decimal integer is $a\times10^4+b\times10^3+c\times10^2+d\times10+e$ and a, b, c, d and e are set in ASCII code.
DED2	b	
DED3	c	
DED4	d	
DED5	e	
BID	Converted results	Low-order 8 bits are stored in BID and high-order 8 bits in BID+1.
BID+1		

Table 3 Error flags for conversion

Item / Number to be converted	Error and no error display: A register	Error and no error display: Carry flip-flop	Converted result
Integer 0~65,535	0	0	Correct
More than 65,535	0	1	Not correct
Character other than decimal digits	1	0	Not converted

Note 3 Overflow is displayed by being carry flip-flop as 1, and error is so A resister as 1.

Fig. 2 Assembly listing of the objective program "CON102"

```
**CROSS ASSEMBLER OF 8-BIT MICROPROCESSOR
*                      *
*       CON102         *
*                      *
2328                      ORG  9000
2328 219923 CON102   LXI  H',DED1
232B 0605            MVI  B,5
232D 7E     CO100    MOV  A,M
232E FE3B            CPI  48
2330 DA9423          JC   ER
2333 FE3B            CPI  59
2335 D29423          JNC  ER
2338 23              INX  H
2339 05              DCR  B
233A C22C23 D23      JN2  CO100
233D 3A9D23 CO000    LDA  DED5
2340 D630            SUI  48
2342 2600            MVI  H,0
2344 6F              MOV  L,A
2345 3A9C23 CO001    LDA  DED4
2348 D630            SUI  48
234A 110A00          LXI  D,10
234D CA5523 CO101    JZ   CO002
2350 19              DAD  D
2351 3D              DCR  A
2352 C34D23          JMP  CO101
2355 3A9B23 CO002    LDA  DED3
2358 D630            SUI  48
235A 116400          LXI  D,100
235D CA6523 CO102    JZ   CO003
2360 19              DAD  D
2361 3D              DCR  A
2362 C35D23          JMP  CO102
2365 3A9A23 CO003    LDA  DED2
2368 D630            SUI  48
236A 11E803          LXI  D,1000
236D CA7523 CO103    JZ   CO004
2370 19              DAD  D
2371 3D              DCR  A
2372 C36D23          JMP  CO103
2375 3A9923 CO004    LDA  DED1
2378 FE37            CPI  37#
237A D29023          JNC  OV
237D D630            SUI  48
237F 111027          LXI  D,10000
2382 CABA23 CO104    JZ   CO005
2385 19              DAD  D
2386 3D              DCR  A
2387 C38223          JMP  CO104
238A 229E23 CO005    SHLD BID
238D C39723          JMP  CO006
2390 37     OV       STC
2391 C39723          JMP  CO006
2394 3E01   ER       MVI  A,11
2396 A7              ANA  A
2397 00     CO006    NOP
2398 76              HLT
2399 00     DED1     DEF  0
239A 00     DED2     DEF  0
239B 00     DED3     DEF  0
239C 00     DED4     DEF  0
239D 00     DED5     DEF  0
239E 0000   BID      DADR
2328                 END
```

14

Table 4 Example of the use of simulation control commands

START M70, CARD	MELCOM 70 is used as the host computer, and the input unit for the control commands is selected to be the card reader.
LOAD START, 5	The object program is input from the paper-tape reader (device number 5).
SET CPU SP=10000 PC=9000	The stack pointer is set to the value 10,000, and the program counter is set to the value 9,000.
SET MEMORY, DED1=31# SE M, DED2:DED5=32#, 33#, 35#, 37#	Data is set in memory. 31# is stored in location DED1, 32# in DED2, 33# in DED2 +1, 35# in DED2 +2, and 37# in DED5.
BREAK C0002, C0003, C0004, C0005	Breakpoints are assigned.
DISPLAY CPU, SP, PC	Displays the contents of the stack pointer (SP) and the program counter (PC) for confirmation.
D M, DED1:DED5	Confirms whether or not the correct value is set in memory. Here, D is the abbreviated command for DISPLAY and M for MEMORY.
GO *	The program is executed until the machine instruction HLT is encountered, printing out the contents of the PC and SP registers and flip-flops at each breakpoint that was assigned by BREAK above.
D M, 9119:9120(@)	Confirms whether the conversion is correct or not, displaying the result of the conversion in binary form. It can also be confirmed by finding the change of the contents of registers H and L in the list that is printed out during execution.
TIME	The number of cycles executed is counted.
NOBR C0002, C0003, C0004, C0005	The breakpoints assigned with BREAK are released.
S M, DED1=36# S M, DED2:DED5=35# S M, DED4=43#	36# is set in address DED1, 35# in addresses DED2 ~ DED5 and 43# in address DED4.
S CP, PC=9000	9,000 is set in the program counter.
GO	Executes until a HLT instruction is encountered.
D M, 9113:9120	The data and the result are printed in the hexadecimal because the BASE command is not used. In this case, including a character other than 0 ~ 9 confirms whether or not a '1' is set in the A register after execution.
SAVE 2, SAV1	Intermediate results are saved in file SAV1 of the disk.

START M70, C	MELCOM 70 is used as the host computer, and the input unit for the control commands is selected to be the card reader.
LO CONT, 2, SAV1 TYPE	The intermediate results that were saved are loaded from the disk. The file name is 'SAV1'. The input unit for control commands is changed from the card reader to the keyboard.
S CUP, SP=10000, PC=9000	The program counter and the stack pointer are set.
S M, DED1:DED5=37#, 35#	37# is set in address DED1, 35# in DED1 +1, 37# in DED1 +2, 35# in DED 1 +3 and 37# in DED5.
GO	Executes until a HLT instruction is encountered. Confirms whether or not a '1' is set in the carry flip-flop because the data exceeded 65,535.
S CPU, PC=9000	The start address is set.
S M, DED1:DED5=30#	30# is set in addresses DED1 ~ DED5.
GO	Executes until an HLT instruction is encountered.
D M, 9113:9120	Confirms the conversion result.
S CPU, PC=9000	The start address is set.
S M, 9113=36# S M, 9115=35#	36# is set in address 9113, 35# in address 9115.
GO	Execution starts. Executes until an HLT instruction is encountered.
D M, 9113:9120	Confirms the conversion result.
END	Declares the end of simulation.

PAPER-TAPE GENERATION PROGRAM FOR PROM WRITERS

DESCRIPTION

This program is used to convert absolute binary object formatted programs, which are produced by the MELPS 8/85 cross-assembler, into other language formats and then produce a paper tape that can be used as input for a PROM writer.

The functional configuration of this program provides for automatic conversion of object programs from one format to another format. In addition, it provides extensions suitable to various applications.

FEATURES

- Producing, selectively, punched paper tapes with simple control commands
- Converting MELPS 8 binary object programs stored on disks into various hexadecimal formats on paper tape
- Converting various hexadecimal formatted paper tapes into MELPS 8 hexadecimal format
- Matching control functions for MELPS 8 hexadecimal formatted paper tape as well as other formats
- Output of various block sizes as specified by the block-size parameter
- Sorting capability to put files in address sequence
- Executing computer is a MELCOM 70 minicomputer
- Implementation language: FORTRAN IV (parts are written in assembler language)

INPUT/OUTPUT MEDIA

- Converts MELPS 8 binary to hexadecimal paper tape.
  Input: cartridge disk
  Output: paper tape (even-parity ASCII code)
- Converts other hexadecimal paper tapes to MELPS 8 hexadecimal paper tapes.
  Input: Paper tape in other hexadecimal format (even-parity ASCII code)
  Output: Paper tape in MELPS 8 hexadecimal format (even-parity ASCII code)
- Compares MELPS 8 hexadecimal with other hexadecimal paper-tape formats.
  Input: Paper tape (even-parity ASCII code)
  Output: Printed on system typewriter.
- Inputs system commands.
  Input using the keyboard of the system typewriter

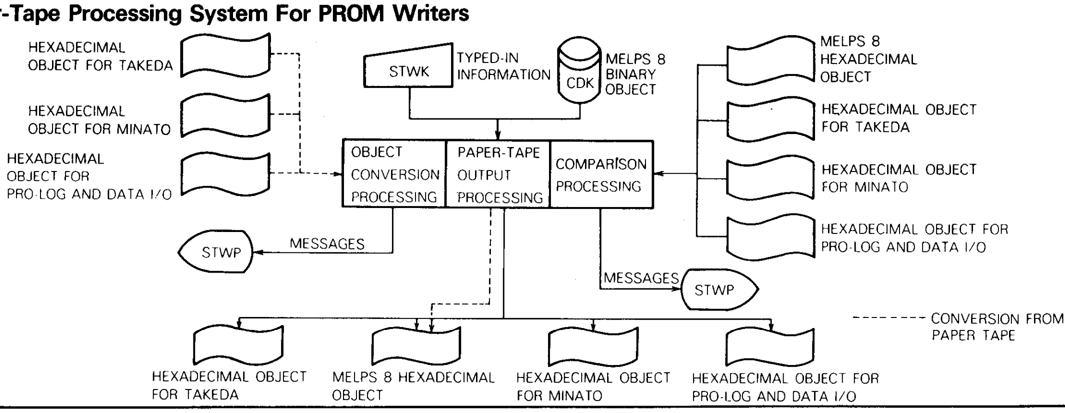
APPLICATIONS

- Programs are applicable to the M58563S (256-word by 8-bit), M5L2708K, S (1024-word by 8-bit, M5L2716K (2048-word by 8-bit) or other similar ROMs when being programmed by a PROM writer made by Data I/O, Prolog, Takeda or Minato Electronics.

FUNCTION

This program converts absolute binary object programs (abbreviated MELPS 8 binary), created on the disk by the MELPS 8/85 cross assembler, into hexadecimal object programs. These hexadecimal object programs can be used to program PROMs on PROM writers such as those made by Data I/O, the Series 90 made by Pro-log Ltd., the T-310 made by Takeda Riken and the 1830 made by Minato Electronics (abbreviated elsewhere to Data I/O, Pro-log, Takeda and Minato). This program also converts absolute binary object programs into MELPS 8 hexadecimal format and creates a paper tape with blocks of suitable size. The program can also convert paper tapes of Data I/O, Pro-log, Takeda and Minato into MELPS 8 hexadecimal format and compare the functions of each.

Paper-Tape Processing System For PROM Writers



PROGRAM ORDERING INFORMATION

Program	Program code number	Program and software manuals included	
Paper tape preparation program for PROM writers	GA1SP0100	Paper-Tape Preparation Program for PROM Writers Manual	GAM-SR00-32A

14

MITSUBISHI ELECTRIC

PAPER-TAPE GENERATION PROGRAM FOR PROM WRITERS

PAPER-TAPE PROCESSING SYSTEMS FOR PROM WRITERS

The program provides for both conversion and comparison of various object programs. Table 1 shows a summary of the conversion processing indicating various combinations of object programs and media that the program is capable of processing. Table 2 shows a summary of the comparison processing indicating the various combinations of object programs and media that the program is capable of processing. Examples of all the object conversions listed in Table 1 are illustrated in Fig. 1.

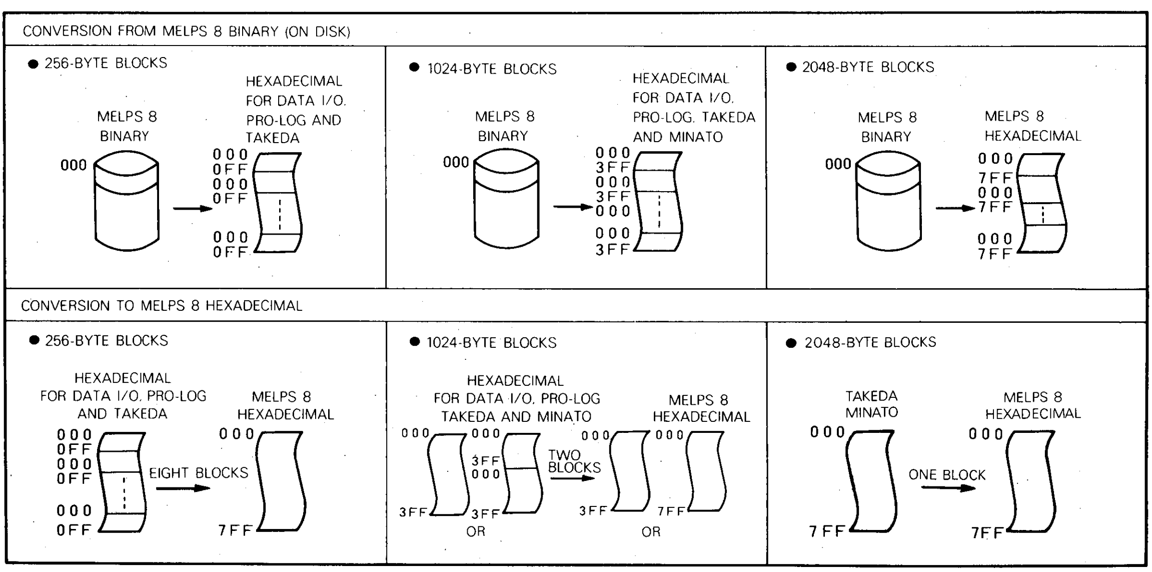
Table 1 Object conversions

Conversion processing for each company's PROM writer / Paper tape block size	Hexadecimal paper tapes for PROM writers that can be converted from MELPS binary (on disk)	Hexadecimal paper tapes for PROM writers that can be converted into MELPS 8 hexadecimal paper tape
256 bytes	Data I/O, Pro-log, Takeda.	Conversion from eight blocks of Data I/O, Pro-log or Takeda to one 2048-byte block
1024 bytes	MELPS 8 hexadecimal (for mask ROM), Data I/O, Pro-log, Takeda, Minato	Conversion from one block of Data I/O, Pro-log, Takeda or Minato to one 1024-byte block or two blocks to 2048-byte block
2048 bytes	MELPS 8 hexadecimal, Takeda, Minato (for mask ROM)	Conversion from one block Takeda, Minato to 2048-byte block

Table 2 Comparison processing of object paper tapes

Objects compared / Comparison	MELPS 8 hexadecimal		Comparison object	
	Object	Media	Object	Media
MELPS 8 hexadecimal self comparison	MELPS 8 absolute hexadecimal	Paper tape ●1024-byte block ●2048-byte block	MELPS 8 absolute hexadecimal	Paper tape ●1024-byte block ●2048-byte block
Comparison of MELPS 8 hexadecimal with Minato	MELPS 8 absolute hexadecimal	Paper tape ●1024-byte block ●2048-byte block	Hexadecimal for Minato	Paper tape ●1024-byte block ●2048-byte block
Comparison of MELPS 8 hexadecimal with Takeda	MELPS 8 absolute hexadecimal	Paper tape ●1024-byte block ●2048-byte block	Hexadecimal for Takeda	Paper tape ●eight 256-byte blocks ●two 1024-byte blocks ●2048-byte block
Comparison of MELPS 8 hexadecimal with Pro-log	MELPS 8 absolute hexadecimal	Paper tape ●1024-byte block ●2048-byte block	Hexadecimal for Pro-log	Paper tape ●eight 256-byte blocks ●two 1024-byte blocks ●2048-byte block
Comparison of MELPS 8 hexadecimal with Data I/O	MELPS 8 absolute hexadecimal	Paper tape ●1024-byte block ●2048-byte block	Hexadecimal for Data I/O	Paper tape ●eight 256-byte blocks ●two 1024-byte blocks ●2048-byte block

Fig. 1 Medium conversion



SELF ASSEMBLER

DESCRIPTION

The MELPS 8/85 self assembler is a target program that has been prepared for the development of application programs suitable to microcomputers using the MELPS 8/85 CPU and devices utilizing microprocessors.

The PTS-A version of the MELPS 8/85 self assembler requires fewer control commands than the cross assembler, and is capable of assembly, even without a host minicomputer, using an inexpensive debug machine.

The coding for this self assembler is easy, since input data in the MELPS 8/85 self assembler language (B-version) may be handled in free format.

FEATURES

Of the Self Assembler

- May be used on either 3-pass or 2-pass system
- Source input may be in free format
- Source input may be prepared either with paper tape or from the keyboard
- The number of symbols can be increased in accordance with memory capacity expansion
- The execution computer is the MELCS 8/1 and MELCS 85/1 debug machine (with memory more than 8K-bytes and using the BOM-PTS monitor)
- The MELPS 8/85 assembler language (A-version) is used as the implementation language

Of the Self Assembler Language

- 8 pseudo instructions
- Algebraic expressions
- Character constants
- Octal, decimal, and hexadecimal numbers
- The mnemonic codes of the machine instructions are the same as those for the MELPS 8/85 cross assembler and Intel's.

INPUT/OUTPUT MEDIA

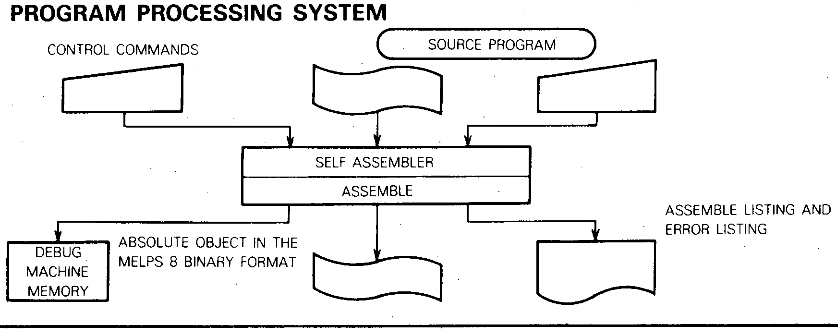
- Source input: Keyboard or paper tape
- Control command input: Keyboard
- Object output: Paper tape or debug machine memory
- Program supply media: Paper tape (object)

FUNCTION

This self assembler converts source programs written in the MELPS 8/85 self assembly language (B-version) into absolute objects in the MELPS 8 binary format utilizing the debug machine.

This self assembler can handle 4 control commands for input device assignment, object output device assignment, assembly execution control, and end designation control, and can use both machine and pseudo instructions. The machine instructions, in one-to-one correspondence with machine language, consist of 80 basic instructions (the same as the MELPS 8/85 cross assembler) that are to be subject to object conversion. The pseudo instructions are divided into assembly control, data setting and storage allocation instructions, and consist of eight instructions.

SELF ASSEMBLER PROGRAM PROCESSING SYSTEM



PROGRAM ORDERING INFORMATION

Program name	Ordering number	Program and software manuals included	
MELPS 8/85 self assembler	GA2AS0100	Self Assembly Language Manual (B-version)	GAM-SR00-25A
		Self Assembler Manual (PTS-A-version)	GAM-SR00-19A
		Self Assembler Operating Manual (PTS-A-version)	GAM-SR00-24A

MANUALS

Manual name	Manual number
MELPS 8 Editor Manual (PTS-A-version)	GAM-SR00-26A
MELPS 8 Editor Operating Manual (PTS-A-version)	GAM-SR00-27A
MELPS 8 Basic Operating Monitor (BOM-B) Manual	GAM-SR00-23A
MELPS 8 Basic Operating Monitor (BOM-PTS) Manual	GAM-SR00-18A
MELPS 8 Hardware Manual	GAM-HR00-01A

This self assembler facilitates assembly by the use of the control commands shown in Table 1. The assembly consists of the creation of the symbol table in pass 1, where source programs are read from the keyboard or paper tapes, the creation of the assembly list in pass 2, where source programs are read from paper tapes and each instruction is converted into machine language, and the output of absolute objects in pass 3.

SELF ASSEMBLER OBJECT LANGUAGE

The cross assembler is composed of many object modules, and each module is composed of a name part, a symbolic part, a text part and a final part. This self assembler outputs only the text part and the final part in response to the object output control command.

ASSEMBLY LANGUAGE FUNCTIONS

The assembly language that this self assembler accepts consists of the following machine instructions and pseudo instructions.

1. Machine Instructions

There are 80 basic machine instructions. These are converted to their corresponding machine codes and then inserted in the object program. The mnemonic and all the other instructions are the same as for the MELPS 8/85 cross assembler; for these please refer to the Cross Assembler Manual.

2. Pseudo Instructions

The pesudo instructions that this self assembler accepts consist of ORG, NAM, PAUS, and END as assembler-control instructions; EQU, DB, and DW as data-setting instructions; and DS as storage allocation instruction. These instructions are summarized in Table 2.

3. Language Format

The Self Assembler Language Manual (B-version) is applicable to the language formats for the MELPS 8/85 self assembler; these are equivalent to those for the MELPS 8/85 cross assembler, with some restrictions, and may be handled in a similar manner. In the source program, a statement starts with CR (carriage return) and ends with CR (carriage return), consisting of label, command, operand, comment, and identification fields.

Table 1 List of control commands for the self assembler

Functional classification	Mnemonic	Function
Input device assignment command	///SP ␣ {ST / SK}.	Input device assignment for pass 1 ST : Paper tape reader SK : Keyboard
Object output device assignment	/// OB ␣ {ST / DM}.	Object output device assignment ST : Paper tape punch DM : Debug machine memory
Assembly execution control	/// OP ␣ {LS / LC / LE / None} {, AN / None}. ① ② ① Listing control ② Object output control	Assembly execution start assignment and control of source listing and of object output (1) Listing control LS : Source listing needed LC : Commentless condensed listing needed LE : Listing of error statements only needed None : Source listing unnecessary (2) Object output control AN : Output of absolute objects without symbol parts None : No object output
End designation control command	/// ED.	End of assembly execution designated

Table 2 List of pseudo instructions

Functional classification \ Item	Instruction mnemonic symbol	Name of instruction
Assembly-control instructions	ORG	Program counter setting
	NAM	Program name declaration
	PAUS	Assemble stop
	END	End declaration
Data-setting instructions	EQU	Value symbol setting
	DB	Data setting
	DW	Address setting
Storage allocation instruction	DS	Storage allocation

Table 3 Labels, characters, numerals, and expressions

Sort	Item	Symbol
Label	Label expression	L :
	Initial characters for labels	A ~Z , @, ?
	Characters, except the initial ones, for labels	A ~Z , @, ?, 0 ~9
	Number of label characters	From one to five (e.g. LABL1:)
Character constant	A 1 byte	▼A▼
	AB 2 bytes	▼AB▼
	A▼B 3 bytes	▼A▼▼B▼
Numeral	Octal number	n O
	Decimal number	n
	Hexadecimal number	n H
Expression	Add	+
	Subtract	−
	Multiply	*
	Divide	/
Others	Program counter	$
	Operational order	From left to right

A comment is preceded by a semicolon (;). Since the format is free, any column may be used if the delimiters are properly placed. (Note that the printout for columns 35~72 and 81 and over are neglected.)

Table 3 summarizes the labels, characters, numerals, and expressions, etc.

1. **Label field**

   One~five characters may be used. Only A~Z, @, and ? may be used as the first character and A~Z, @, ?, and 0~9 may be used as the remaining characters. A colon is to be added at the end of the character string.

   Label example **L1:MOV A, B**
   LABL5:
   @ABCD:
   A123?:
   ?AB01:

2. **Instruction field**

   Instruction mnemonic codes are placed in this field. Machine instructions are formed with the same codes as in the MELPS 8/85 cross assembler. The pseudo instructions available are, ORG, NAM, PAUS, and END as assembler-control instructions; EQU, DB, and DW as data-setting instructions; and DS as storage allocation instruction.

3. **Operand field**

   Operands 1 and 2, the first and second operands of the instruction parameters, may be written. When both the operands 1 and 2 are necessary, a comma as a delimiter should be written.

   Octal, decimal, and hexadecimal numbers may be used as numerals, formats such as ▼A▼, ▼AB▼, ▼A▼ ▼B▼, etc. as character constants, expressions combined with operators (+, −, *, /) as expressions, and $ as the program counter.

4. **Comment field**

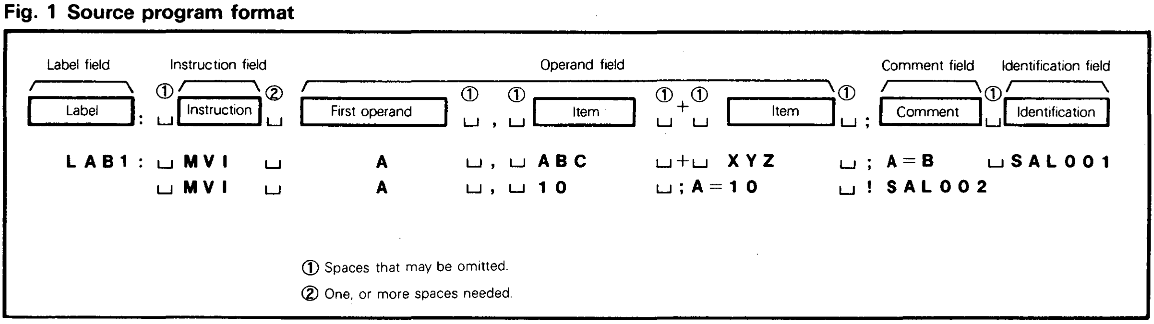
   A line preceded by a semicolon (;) and a character string following a semicolon (;) placed at the end of a command or at an arbitrary position along a line are regarded as comments.

Comment examples; **THIS LINE IS COMMENT**
;COMMENT
;
LI: MOV A, B;COMMENT;ABC

5. **Identification field**

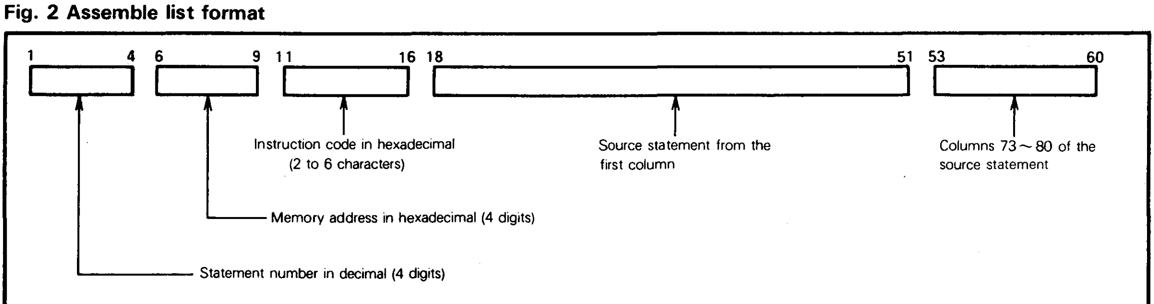
   The field is composed of the characters in columns 73~80 or from 1 to 8 characters following !. This field is placed at the end of one statement and may be omitted.

Fig. 1 Source program format



Note : Mark ␣ denotes space.

Fig. 2 Assemble list format



FORMATS FOR SOURCE PROGRAM AND ASSEMBLE LIST

The coding of source programs is in free format like that shown in Fig. 1.

The format of assemble lists is shown in Fig. 2 and an example of the list is given in Fig. 3.

OBJECT TAPE FORMAT

The object program which is generated in pass 3 is an absolute object program in MELPS 8 binary format.

ASSEMBLE EXAMPLES

Examples of execution of passes 1, 2, and 3 are given in Fig. 4 for paper-tape input and in Fig. 5 for keyboard input.

ERROR MESSAGE FORMAT

Error messages are divided into two types: one for control commands and the other for assemble.

Errors for control commands ··· ＊Q＊

Errors for assemble ···? ␣＊x＊   x: Error code

Fig. 3 Example of assemble list

```
1   4 6    9 11       1618                                    5153      60
0001                   ;    SUBROUTINE (1) ···········MULT AT-02000
0002                   ;                                   AT-02020
0003                   ;        *  DATA  (A)········MULT AT-02030
0004                   ;                 (B)········MULT AT-02040
0005                   ;        * RESULT (H) (L)···· PROD AT-02060
0006                   ;                                   AT-02070
0007 0000 0E08     L@001: MVI   C, 8                       AT-02080
0008 0002 210000          LXI   H, 0                       AT-02090
0009 0005 110000          LXI   D, 0                       AT-02100
0010 0008 57              MOV   D, A                       AT-02110
0011                                                       AT-02120
0012 0009 7A       L@002: MOV   A, D                       AT-02130
0013 000A 0F              RRC   ;                     RI G AT-02140
0014 000B 57              MOV   D, A                       AT-02150
0015 000C 7C              MOV   A, H                       AT-02160
0016 000D D21100          JNC   L@003                      AT-02170
0017 0010 80              ADD   B ;                    (A) AT-02180
0018                   ;                                   AT-02190
0019 0011 1F       L@003: RAR   ;                     R-S  AT-02200
0020 0012 67              MOV   H, A                       AT-02210
0021 0013 7D              MOV   A, L ;                 (A) AT-02220
0022 0014 1F              RAR   ;                     R-S  AT-02230
0023 0015 6F              MOV   L, A                       AT-02240
0024                   ;                                   AT-02250
0025 0016 79              MOV   A, C ;                 (A) AT-02260
0026 0017 D601            SUI   1 ;                    (A) AT-02270
0027 0019 4F              MOV   C, A                       AT-02280
0028 001A C20900          JNZ   L@002                      AT-02290
0029 001D 7A              MOV   A, D                       AT-02300
0030 001E C9              RET                              AT-02310
0031                   ;                                   AT-02320
0032 0000                 END                              AT-03330
```

Fig. 4 Paper tape input

```
:///SP.       Input from a tape reader
:///OB.
:///OP LS,AN.
P1 START
P1 END
:///GO.       Continue pass 2
P2 START      Assemble listing start
0001                       NAM  EXAMP1
0002 03E8                  ORG  1000
0003 03E8 79        L001:MOV  A,C
0004 03E9 3E02      L002:MVI  A,2
0005 03EB 48        L003:MOV  C,B
0006 03EC 00               NOP
0007 0000                  END
P2 END
:///GO.
P3 START
```

Fig. 5 Keybord input

```
:///SP SK.········· Input from a keyboard
:///OB.
:///OP LS,AN.
P1 START
        NAM EXAMP1 |
        ORG  1000  |
L001:MOV A,C       | Statements that are
L002:MVI A,2       | typed in from a keyboard
L003:MOV C,B       |
        NOP        |
        END        |
P1 END
:///GO.
P2 START ··········· Assemble listing start
0001                       NAM  EXAMP1
0002 03E8                  ORG  1000
0003 03E8 79        L001:MOV  A,C
0004 03E9 3E02      L002:MVI  A,2
0005 03EB 48        L003:MOV  C,B
0006 03EC 00               NOP
0007 0000                  END
P2 END
:///GO.
P3 START
```

DESCRIPTION

The MELPS editor program was developed to make modifications of programs at the source language level easy. This design feature also makes it a useful tool in program development for microcomputers and micro-processors.

FEATURES

- Fifteen easy-to-use control commands
- Convenient loading from the keyboard or by paper tape
- Variable work area to match the application requirements
- Versatile input control
- Easy-to-use buffer-pointer control
- Flexible output control
- Data editing made easy
- String command is possible
- The repetition function of commands shortens input commands
- Editor complete control
- The command format is similar to that used in the MELCOM 70 editor
- Debugging and execution are done on a MELCS 8/1 and MELCS 85/1 (memory 8K-bytes, monitor BOM-PTS)
- The programming language is MELPS assembler (A version).

INPUT/OUTPUT MEDIA

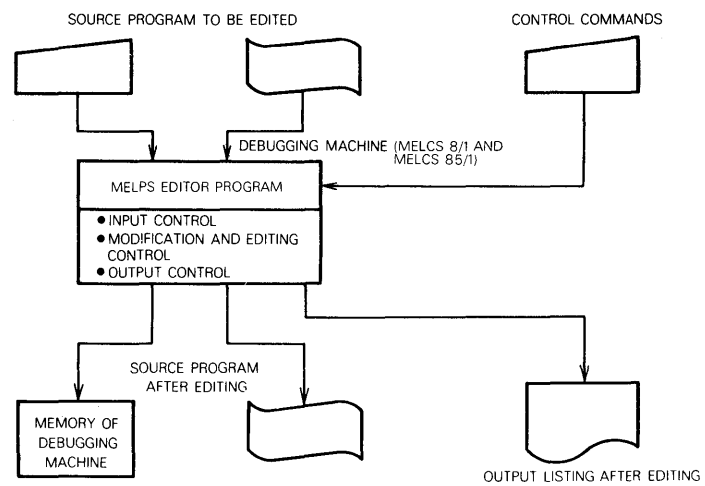
- Programs for editing: Keyboard or paper tape
- Control commands input: Keyboard
- Output after editing: Printer or paper tape

FUNCTION

The MELPS editor loads text from paper tape or keyboard into the work area where the text is modified and edited. Control commands for the editor are entered through the keyboard. The edited text is punched out on paper tape, and at the same time the copy can be printed.

The powerful control commands are divided into five functions as shown in Table 1. There are a total of 15 easy-to-use control commands. One instruction can delete, insert or replace from one character to a number of lines. This is facilitated by the flexible control provided for the buffer pointer. The edited results can be punched on paper tape and printed simultaneously.

MELPS Editor Processing System



PROGRAM ORDERING INFORMATION

Program	Ordering number	Program and software manuals included	
MELPS 8 Editor	GA2SP0103	Source Program	
		MELPS Editor Manual (PTS-A version)	GAM-SR00-26A
		MELPS Editor Operating Manual (PTS-A version)	GAM-SR00-27A

MANUALS

Manual name	Manual number
MELPS 8 Self Assembler Language Manual (B-version)	GAM-SR00-25A
MELPS 8 Self Assembler Manual (PTS-A version)	GAM-SR00-19A
MELPS 8 Self Assembler Operating Manual (PTS-A version)	GAM-SR00-24A
MELPS 8 Basic Operating Monitor (BOM-B) Manual	GAM-SR00-23A
MELPS 8 Basic Operating Monitor (BOM-PTS) Manual	GAM-SR00-18A
MELPS 8 Hardware Manual	GAM-HR00-01A

FUNCTIONAL OPERATIONS

The MELPS editor is designed to increase the effectiveness of modifying, editing, and debugging programs. There are five groups of control functions: input control, buffer-pointer control, output control, data-editing control and editor end control. There are a total of fifteen control commands listed in Table 1. An explanation of the action of each control command is also given in Table 1. The general format of a control command for input is shown in Fig. 1.

1. String commands

   The control commands can be used independently or they can be combined into a string as shown in the example that follows.

   ///BP$5TW$2CP$3DL$RPA$B$$

2. Command repetition

   The format for repetition of a command is as follows:

   n <command string <command string < ···· <··· >>>

Where n is a decimal number and $|n| \leqq 255$, if n is negative, it is converted to a positive number. The command string between <and> will be repeated n times. Repetition command nesting of <and> is limited to eight levels.

An example of command formats and how they can be stringed follows. The contents of the work area before and after execution are also shown in Fig. 2.

Fig. 1 General format of input commands

```
///[     ]$  [     ]$$
 ①   ②  ③    ④   ⑤
```

① This is an input command message. Editor program is ready to accept a command.
②④ Editor control commands and necessary arguments.
③ Delimiter (one ESC) (ESC = Escape)
⑤ End of command delimiter (two ESC).

Fig. 2 Typical editor command

An editor command

///BP$RP1$10$2CP$DLBPFPPOP D$IN;C$$

(② = RP1$10, ③ = 2CP$DL, ⑤ = FPPOP D$IN;C)

Before the above command is executed and the modifications are made.			After the above command is executed and the modifications are made.		
PUSH	**H**	①	**PUSH**	**H**	①
MVI	**D, 1**	②	**MVI**	**D, 10**	②
MOV	**E, A**	③	**SHLD**	**XY**	④
SHLD	**XY**	④	**POP**	**D ; C**	⑦
POP	**D**	⑤			

① Not modifying.
② Replace 1 with 10.
③ Delete one line (MOV E,A)
④ Not modifying.
⑤ Search the string for the data (POP D) and set the buffer pointer to the end location of the data.
Insert the assigned string (;C) in the work area starting at the location indicated by the buffer pointer.

Table 1 Editor control commands and an explanation of their actions

Control function	Control command	Mnemonic	Action
Input control	Source load	**LD**	Assign the input device for text load and load text.
Buffer-pointer control	Buffer-pointer initial setting	**BP**	Set the buffer pointer to the first address of the work area.
	Buffer-pointer character setting	**CP**	Move the buffer pointer n characters.
	Buffer-pointer line setting	**LP**	Move the buffer pointer n lines.
	Buffer-pointer end setting	**EP**	Move the buffer pointer to the end of the work area.
Output control	Print typewriter	**TW**	Print n lines.
	Line punch	**PN**	Punch n lines from the first line of the work area.
	Punch work area	**PP**	Punch all the contents of the work area.
	Punch sprocket holes	**PS**	Punch sprocket holes for n bytes.
Data-editing control	Delete character	**DC**	Delete n characters.
	Find and buffer-pointer setting	**FP**	Search the string for the data and set the buffer pointer to the end location of the data.
	Replace	**RP**	Locate data to be replaced and replace with the new data.
	Delete line	**DL**	Delete n lines.
	Insert	**IN**	Insert the assigned string in the work area starting at the location indicated by the buffer pointer.
Editor end control	End	**EN**	End of editor processing.

BASIC OPERATING MONITOR—PAPER-TAPE SYSTEM

DESCRIPTION

The BOM-PTS basic operating monitor was developed for microcomputers that use the M5L8080A 8-bit parallel CPU. It controls execution and debugging of the user's program. The BOM-PTS has a program capacity of 7.5K-bytes and drives the system typewriter (Casio Typuter, Model 500 or 501) as its I/O unit.

FEATURES

- Has 3 macro instructions and 22 monitor commands
- Provides trace, snapshot, and address halt commands for effective program development and debugging
- Has pseudo I/O and PROM write functions

FUNCTION

The BOM-PTS 22 monitor commands and 3 macro instructions provide the following functions:

1. Program execution control
2. Program loading
3. Memory punching
4. Program debugging (trace, snapshot, and halt commands)
5. I/O control and pseudo I/O processing
6. Memory and register data display, and data alteration
7. PROM writing function

Starting BOM-PTS Execution

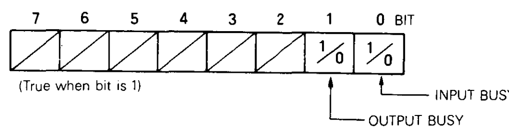
When the BOM start switch on the panel of the debugging machine MELCS 8/1 is turned on, the following message is printed out. After the printout, monitor commands can be entered.

BOM-PTS A00 `READY'
//

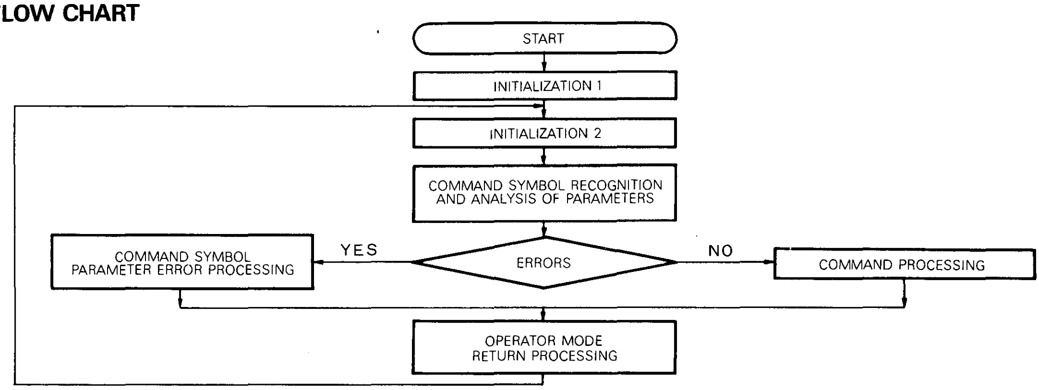
Hardware Limitations

1. Memory Configuration
   Memory locations in the ROM are:
   $E000_{16} \sim FCFF_{16}$
   In addition to the ROM, the following 78 bytes of RAM area are required:
   $F000_{16} \sim EDFF_{16}$
2. Input/Output Device Addresses
   PTR, for keyboard input: $7B_{16}$ **(IN 7B#)**
   PTP, for printout: $7B_{16}$ **(OUT 7B#)**
   Status input: $7B_{16}$ **(IN 7B#)**

The structure of the status bits is as follows:



FLOW CHART



PROGRAM ORDERING INFORMATION

Program name	Ordering number	Program and software manuals included
MELPS 8 basic operatimg monitor (BOM-PTS)	GA2OS0100	Source program, Object program Basic Operating Monitor Manual (BOM-PTS) GAM-SR00-18A

MANUALS

Manual name	Manual number
MELPS 8 Basic Operating Monitor Manual (BOM-B version)	GAM-SR00-23A
MELPS 8/85 Self-Assembler Language Manual (B version)	GAM-SR00-25A
MELPS 8/85 Self-Assembler Manual (PTS-A version)	GAM-SR00-19A
MELPS 8/85 Self-Assembler Operating Manual (PTS-A version)	GAM-SR00-24A
MELPS 8 Hardware Manual	GAM-HR00-01A

MONITOR COMMANDS AND MACRO INSTRUCTIONS FOR BOM-PTS

Name		Function	Command designation, parameter input format, and calling sequence	Parameter
Command	G	Start program	//G para1(4)␣[para 2(4)] CR LF	para1(4): Starting address para2(4): Altered starting address
	R	Restart program	//R CR LF	—
	U	User pseudo I/O processing	//U para1(4) CR LF	para1(4): First address of the user pseudo I/O processing routine
	LM	MELPS 8 binary loader	//LM para1(4), para2(4) CR para3(2) CR LF	para1(4): ROM start address(when relocatable) para2(4): RAM start address(when relocatable) para3(4): LE(Load End indicating key word)
	DM	Dump memory data, MELPS 8 binary test portion (to paper tape punch)	//DM para1(1), para2(4), para3(4) CR LF	para1(1): para1(1)=T para2(4): Start address para3(4): End address
		Dump MELPS 8 binary end portion (to paper tape punch)	//DM parp1(1) [para4(4)] CR LF	para1(1): para1(1)=E para4(4): Starting address
	PR	Printout register data in hexadecimal form	//PR CR LF	—
	PM	Printout memory data in hexadecimal form	//PM para1(4), para2(4) CR LF	para1(4): Starting address para2(4): End address
	PA	Reverse assembler	//PA para1(4), para2(4), para3(1) CR LF	para1(4): Starting address para2(4): End address para3(1): No reverse assembly is done to the operand when para 3(1)=1.
	MR	Alter the register data	//MR CR LF	—
	MM	Alter the memory data	//MM para1(4) CR LF	para1(4): Starting address
	MC	Complement the memory data	//MC para1(4), para2(4) CR LF	parp1(4): Starting address para2(4): End address
	MS	Set up constants in memory	//MS para1(4), para2(4), para3(2) CR LF	para1(4): Starting address para2(4): End address para3(2): The constant
	MT	Transfer memory data in blocks	//MT para1(4), para2(4), para3(4) CR LF	para1(4): Starting address para2(4): End address ⌈which transfer is made para3(4): Starting address of the memory to
	I	Enable machine interrupt	//I para1(1) CR LF	para1(1): Enables machine interrupt when para 1(1)=1, and disables interrupt when para 1(1)=1.
	PT	Print debug table	//PT CR LF	—
	C	Clear debug table	//C CR LF	—
	H	Prepare halt and debug table	//H para1(1), para2(4), para3(4) CR LF	para1(1): para1(1)=S para2(4): Halt address para3(4): Number of passes before halt is active
		Cancel halt and debug table	//H para1(1), para2(1) [, ···, para9(1)] CR LF	para1(1): para1(1)=D para2(1)~para9(1): 0~7 (table number), W (whole table)
	S	Prepare snapshot and debug table	//S para1(1), para2(4), para3(6), para4(4), para5(4) [, para6(1)] CR LF	para1(1): para1(1)=S para2(4): Snapshot executing address para3(6): Snapshot symbol para4(4): Memory data display starting address para5(4): Memory data display end address para6(1): para6(1)=R
		Cancel snapshot and debug table	//S para1(1), para2(1) [, ···, para9(1)] CR LF	para1(1): para1(1)=D para2(1)~para9(1): 0~7 (table number), W (whole table)
	T	Prepare trace and debug table	//T para1(1), para2(4), para3(4), para4(4), para5(4) [, para6(1) [, para7(1)]] CR LF	para1(1): para1(1)=S para2(4): Trace region starting address para3(4): Trace region end address para4(4): Memory data display starting address para5(4): Memory data display end address para6(1): para 6(1)=R specifies register data display. para 7(1)=B specifies to trace only while the debug para7(1) instruction is in execution.
		Cancel trace and debug table	//T para1(1), para2(1) [, ···, para5(1)] CR LF	para1(1): para1(1)=D para2(1)~para5(1): 0~3 (table number), W (whole table)
	FP	Write PROM	//FP para1(4), para2(4), para3(2) CR LF	para1(4): Starting address para2(4): End address para3(2): PROM writing address
	FT	Transfer writing address	//FT para1(4) CR LF	para1(4): Starting address
	FC	Compare PROM data with main memory data	//FC para1(4) CR LF	para1(4): Starting address
Macro instruction	EXIT	End declaration of program	CALL F015#	
	PAUSE	Temporary stop of program execution	CALL F012#	
	EXIO	I/O control	(see calling sequence below)	

EXIO calling sequence:

```
         :
       CALL  FOOC #---- Execution of the EXIO macro instruction
       DADR DCB1 ------- Starting address of the data control block
         :                (DCB)
DCB1   DEF   IOD---- Designation of I/O operation; PTR (10D=52#), PTP (=50#),
                     keyboard (=4B#), printout (=44#)
       DADR DA ---- Setup of the I/O data-storing memory starting address
       DADR DL ---- Setup of the I/O data-storing memory length
         :
DA     [ I/O data-storing memory ]  ↕ DL byte
```

Note 1 : paran (m): n = the nth parameter (input by the operator or printout by the monitor) in a command, and is a hexadecimal parameter 1~m digits. If the number of digits in the parameter exceeds m, only the first m digits are valid.

2 : ——(underlining); Represents and input by the operator.

3 : [ ] (blocking); Represents input by the operator that can be omitted.

4 : #; Indicates hexadecimal number in the assembler language.

MITSUBISHI ELECTRIC

BASIC OPERATING MONITOR—BASIC SYSTEM

DESCRIPTION

The MELPS 8 BOM-B basic operating monitor was developed for microcomputers that use the M5L8080A 8-bit parallel CPU. It controls execution and debugging of the user's program. It is contained in 2K-bytes of memory and drives the system typewriter (Casio Typuter Model 500) as its I/O unit.

FEATURES

- Available as a standard mask ROM (M58731-001S) It can also be programmed into a ROM for a microcomputer configuration that incorporates program debugging functions.
- Has 3 macro instructions and 9 monitor commands
- Allows addition of user's monitor commands
- Cannot be destroyed by a user's program

FUNCTION

The 9 monitor commands and 3 macro instructions provide the following functions:

1. Program execution control
2. Program loading
3. Memory punching
4. Program debugging
5. I/O control

Starting BOM-B Program Execution

When program execution is started at address 6800_{16}, the following message is printed out.

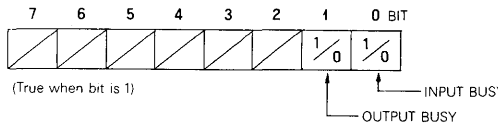
//**MELPS 8 BOM-B A01**
//

After the printout, monitor commands can be entered.

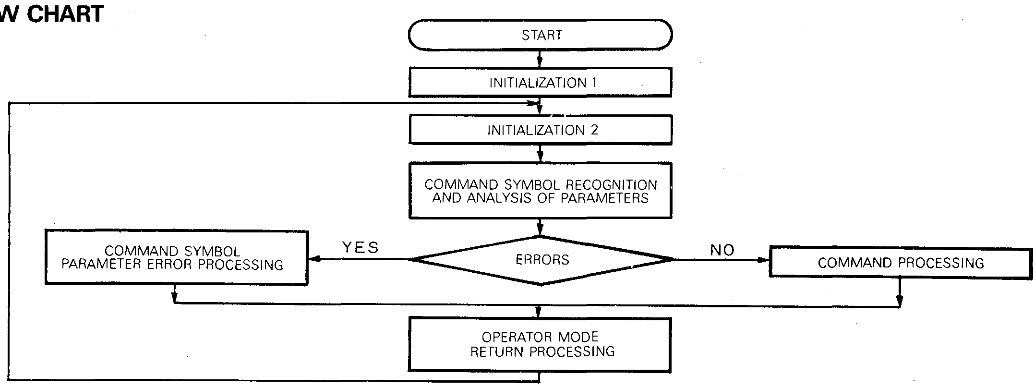
Hardware Limitations

1. Memory Configuration
   Memory locations in the ROM are:
   6800_{16}~$6FFF_{16}$
   In addition to the ROM, the following 78 bytes of RAM area are required:
   $3F80_{16}$~$3FCD_{16}$
2. Input/Output Device Addresses
   PTR, for keyboard input: $7B_{16}$ **(In 7B#)**
   PTP, for printout: $7B_{16}$ **(OUT 7B#)**
   Status input: $7B_{16}$ **(IN 7B#)**

The structure of the status bits is as follows:



FLOW CHART



PROGRAM ORDERING INFORMATION

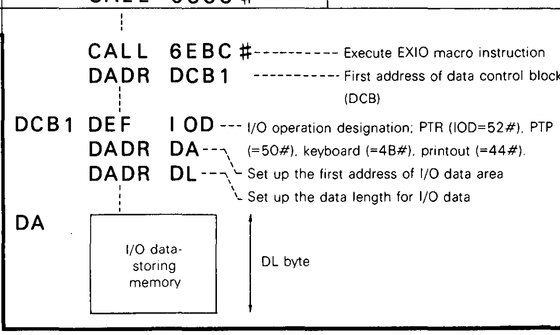
Program name	Ordering number	Program and software manuals included
MELPS 8 basic operating monitor (BOM-B)	GA20S0101	Source program, Object program Basic Operating Monitor Manual (BOM-B version) GAM-SR00-23A

MANUALS

Manual name	Manual number
MELPS 8 Basic Operating Monitor Manual (BOM-B version)	GAM-SR00-18A
MELPS 8/85 Self-Assembler Language Manual (B version)	GAM-SR00-25A
MELPS 8/85 Self-Assembler Manual (PTS-A version)	GAM-SR00-19A
MELPS 8/85 Self-Assembler Operating Manual (PTS-A version)	GAM-SR00-24A
MELPS 8 Hardware Manual	GAM-HR00-01A

Monitor commands and macro instructions for BOM-B.

Name		Function	Command designation and parameter input format or calling sequence	Parameter
Commands	G	Change start address	//G para1(4)␣[para2(4)] CR LF	para1(4): Start address para2(4): Change start address
	R	Restart of program	//R CR LF	—
	L	MELPS 8 binary loader	//L CR LF	—
	H	MELPS 8 hexadecimal loader	//H CR LF	—
	T	Punch MELPS 8 binary text block of the memory data	//T para1(4), para2(4) CR LF	para1(4): First address para2(4): End address
	E	Punch MELPS 8 binary end block	//E [para1(4)] CR LF	para1(4): Start address
	P	Print hexadecimal test block of the memory data	//P para1(4), para2(4) CR LF	;para1(4): First address para2(4): End address
	S	Substitute memory	//S para1(4) CR LF	para1(4): Change address
	M	Print and modify register data in hexadecimal format	//M CR LF	—
Macro instruction	EXIT	End of program	CALL 6806#	
	PAUSE	Pause program execution	CALL 6803#	
	EXIO	Input/output control		



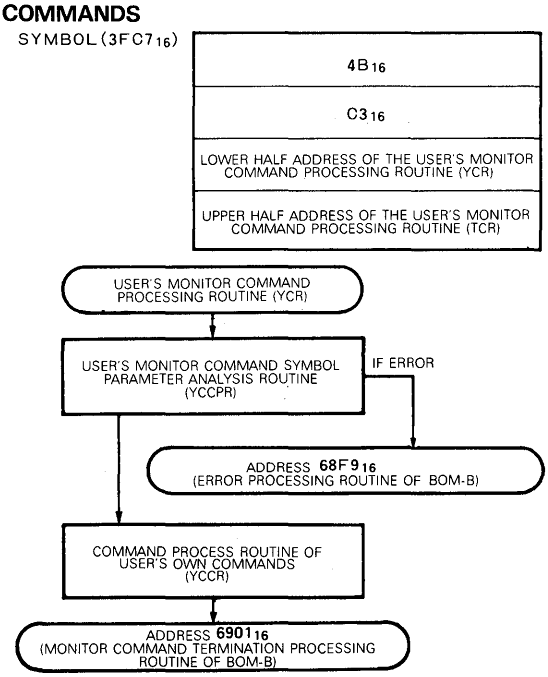
Note 1 : para n (m): This designation shows the nth parameter in a command (operator input or monitor printout), and also shows it to be a hexadecimal parameter (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F) of which the significant digits are 1~m. If the length exceeds m, the least significant digits are valid.
2 : ___(underline): Indicates input by an operator.
3 : [ ] (blocking): Indicates input by an operator that can be omitted.
4 : #: Indicates a hexadecimal number in assembler language.

HOW TO IMPLEMENT USER'S OWN MONITOR COMMANDS

It is feasible to implement new monitor commands, which are prepared by a user for his own need, by correcting four bytes ($3FC7_{16}$~$3FCA_{16}$) of the record in the RAM. The user's monitor commands are then added as follows:

1. Set the data in "$4B_{16}$" to SYMBOL.
2. Set the data in "$C3_{16}$" to SYMBOL + 1.
3. Set the starting address of the user's monitor command processing routine (YCR) low-order into SYMBOL + 2 and high-order into SYMBOL + 3.
4. Then a symbol parameter analysis routine and command processing routine are prepared as required for the user's command.
5. Command symbols used for the user's monitor commands should not be identical with any of the 9 command symbols used in BOM-B.
6. Both command symbol and parameter errors are checked in the YCR, and a jump is executed to address $68F9_{16}$, where the error processing routine of the BOM-B is residing, when an error is found. A question mark (?) will be printed out in case an error is found.
7. The last step of the YCR must be a jumped to address 6901_{16}, where the monitor command termination processing routine is stored.

PROCESS FLOW OF USER'S MONITOR COMMANDS



APPLICATIONS

15

MEMORY DEVELOPMENT APPROACHES

1. LARGE-SCALE SEMICONDUCTOR MEMORY DEVELOPMENT APPROACHES

Introduction

The IC age which began in 1965 progressed into the 1970's to see the beginning of a full-swing push towards LSI and in 1978 to the advent of VLSI, the 64K-bit RAM. Going into the 1980's, the 64K-bit RAM is being used practically in systems, and in February 1980 two announcements of 256K-bit RAM development in Japan were made at the International Solid State Circuit Conference (ISSCC) held at San Francisco. The VLSI Joint Research Laboratory has pointed out the possibility of megabit memories, a development which may materialize in the form of a 1M-bit RAM by the mid 80's.

The ever-increasing level of integration in MOS memory devices has been supported by three areas of advancements.

- Circuit engineering advancements
- Process accuracy and micro-process advancements
- Increase in chip size

Using the ISSCC report material as a basis we will examine some practical approaches to the development of megabit memory devices and some of the problems involved in this effort.

Approaches to Megabit Memory Development

Table 1.1 summarizes some typical approaches that have been proposed for the development of megabit memory. As is clear from this outline, areas for potential study include improvements in cell structure and development of methods to overcome the presence of failed bits in large-scale memories as well as the obvious aim of development of smaller and smaller memory cells.

Table 1.1 Approaches to Megabit Memory Development

Approach		Prototype results: Capacity (bits)	Prototype results: Chip size	Description	Announced by	Year	Month
Device	Micro-process	256k	4.8×8.6mm	Single 5V operation 1 transistor and 1 CMOS to form the cells. L = 1.5μm	NTIS	80	2
		256k	5.8×5.8mm	Same as above L_{eff}=1.2μm, molybdenum gate with spare cell.	NTT	80	2
	New structure	0.5M	*(10×10mm)	Same as above, QSA (quadruply self-aligned), L = 2μm	Joint Research Laboratory	80	2
		64k	24mm²	Same as above, 3-layer polysilicon	NS	79	2
		64k	25.2mm²	Same as above, VMOS, L ≥ 4μm	Siemens	79	2
		64k	30mm²	MOS 1 transistor, 1 capacitor bipolar configuration, 8K-word x 8 bits	IBM	80	2
Logic/circuitry	Failed bit saving	4M		Masked PROM using the full wafer	NTT	80	2
		64k	4.3×9.4mm	With spare decoder	Bell Labs	79	2
				Partial (half-good)	Teradyne	80	1
	Redundancy	1M		RAM using full wafer, 2 x 32K-word x 20 bits with 2 spare bits	NTT	79	2
		64k		3-layer connections, Al gate, 8-bit register built-in	IBM	80	2

*Forecast value

Memory Cell Structure Overview

The remarkable progress that has been made in the development of highly integrated MOS memories has centered around the development of static devices using structural elements of six, four, three, and finally two elements and ultimately the transition to dynamic devices with improvements in the surrounding circuitry and the development of micro-patterns as well as the use of larger and larger chips. From the standpoint of the number of cells used in large-scale memories, since the development of two-element cells using one MOS transistor and one MOS capacitor, devices have progressed from the originally developed 4K dynamic RAM through 16K, 64K, and today 256K RAM devices. Several proposals have been made for a 1-element memory cell, placing the megabit memory within the realm of possibility. We will discuss here several examples of 1-element and 2-element cells and the approach to fabrication of megabit memories.

1. Polysilicon Bit Line Cells

To increase the read voltage for 2-element cells, the ratio of the MOS capacitance to the bit line floating capacitance must be made as high as possible in a small area. Another reason for making the MOS capacitance as high as possible is the reduction of soft errors caused by minute amounts of a particles included inside the package material. For this reason, polysilicon has been replacing n^+ defused material for the fabrication of bit lines. Fig. 1.1 illustrates the example of a Mostek 64K RAM. By making the second polysilicon layer (POLY II) the bit line, the activated region area is most effectively used and the MOS capacitor value is increased while the bit line floating capacitance is decreased. Fabricating the bit line from polysilicon is an effective means of reducing soft error rates as well (in the depletion layer between n^+ bit line region and p substrate, electrons generated by a particles collect in the bit lines and generate noise but this is of a decreased level). The process for this type of cell is characterized by a reduced resistivity for the polysilicon layer and the fabrication of a burried contact between the polysilicon bit line and the n^+ fused layer.

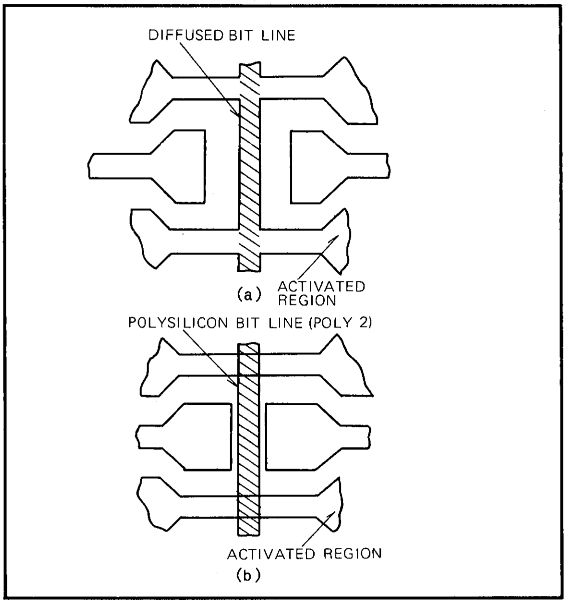


Fig. 1.1 Diffused bit line (a), and polysilicon bit line (b)

2. Hi-C RAM Cell

For 2-element cells the junction capacitance associated with and parallel to the MOS capacitor can be used effectively to form a Hi-C RAM cell. Fig. 1.2 (a) illustrates a conventional cell in which the sum of the MOS capacitance C_{OX} formed by the inverted layer and storage gate and the junction capacitance C_D formed by the inverted layer and the p substrate is used as the capacitance for data storage. In conventional dynamic RAMs, the p-type impurity density is $1 \sim 2 \times 10^{15}/cm^2$ so that C_D is $<< C_{OX}$ and so C_D is ignored. In the Hi-C cell illustrated in Fig. 1.2 (b), a n^+-p^+-p junction is formed just below the storage gate so that the spreading of the depletion layer between the n^+ region and p substrate is reduced, increasing the value of C_D. A simple approximation expression for C_D is given below.

$$C_D = \sqrt{\frac{\epsilon_{Si} \cdot q \cdot N_A}{2(V_S + |V_{BB}|)}}$$

In this expression, ϵ_{Si} is the dielectric constant for silicon, q is the charge of an electron, N_A is the impurity density of the p-type semiconductor, V_S is the potential of the n^+ region, and V_{BB} is the potential of the p-type substrate. For the Hi-C cell, the value of N_A is made large. For a storage gate oxide layer 50nm, the ion implant of impurities required for the formation of a p^+ region is $1 \times 10^{13}/cm^2$ making the expected value of $(C_{OC} = C_D)/C_{OX}$ 1.63 (actual measurements have shown this to be 1.67). In this manner, using the same area, the data storage capacitance is increased more than 60% allowing the bias potential to be made equal to the ground potential. In previously used cell structures, to form the inverted layer it was necessary to provide a bias voltage, V_{DD}. For this reason, variations in the value of V_{DD} created the danger of read errors. With the Hi-C cell structure this danger is completely eliminated. One process difficulty arises however in that the Hi-C region may not self-align with the storage gate and transfer gate.

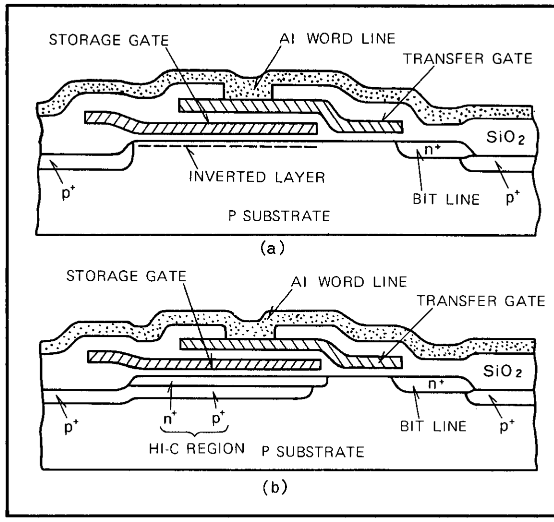


Fig. 1.2 Conventional cell (a), and Hi-C RAM cell (b) cross-sections

3. 3-Layer Polysilicon Cell

While the Hi-C cell is improved by correction of the internal vertical structure of the substrate, the 3-layer polysilicon cell and stacked capacitor RAM cell (STC RAM) represent the approach of correction of the vertical structure of the surface of the substrate. Fig. 1.3 illustrates the cross-section of a National Semiconductor 3-layer polysilicon cell. The capacitance formed by Poly 1 and Poly 2 is used for data storage, while Poly 1 is connected through a burried contact to the source of the self-aligned transfer gate formed by Poly 3. Poly 2 is biased to ground potential. The 3-layer polycell features a data storage capacitance formed by a polysilicon to polysilicon structure and an n^+-p junction capacitance C_D formed by a very small surface area (in the example shown in Fig. 1.3, the n^+-p junction takes up only 1/15 of the total storage node surface area). For this reason, the leakage current is small, enabling good data hold properties, while reducing α particle related soft error rates. The process is characterized by the necessity to fabricate a third polysilicon layer, reduce the resistance of the polysilicon material, and fabricate the burried contact.

MITSUBISHI ELECTRIC

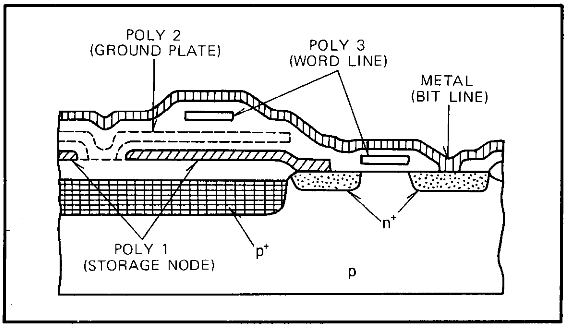


Fig. 1.3 3-Layer polysilicon cell example

4. VMOS RAM Cell

Semiconductor technology has focused on the use of silicon in a planar structure. VMOS technology has been developed however to increase the level of integration by means of forming vertical channels. VMOS uses anisotropic etching to form V-shaped grooves in the silicon surface, forming a channel along the wall of the grooves. Fig. 1.4 illustrates the cross section structure of a dynamic RAM cell introduced recently by AMI. In the VMOS cell, the junction capacitance formed by the burried n^+ region and the p^+ substrate is used for data storage. A p^- layer is epitaxially grown on the p region of the p^+ substrate surface by means of autodoping of impurity atoms. The p region along the surface of the V-shaped groove forms the n^+ drain (bit line), n^+ source (junction data storage capacitance), metal gate (word line) which form the VMOS transistor and determines its effective channel length. As is clear from the figure, the data storage capacitance is included in the area of the VMOS transistor (transfer gate), so that the level of integration is double that of a planar-type 2-element cell device. Siemens has produced a prototype of a VMOS 64K RAM (4μm grid device), the cell area being 215μm^2 and the chip area being 25.2mm^2. VMOS technology was first used to produce static circuits as well as dynamic memory and EPROM. However, the gates formed are susceptible to voltage breakdown and formation of the V-shaped groove caused process difficulties.

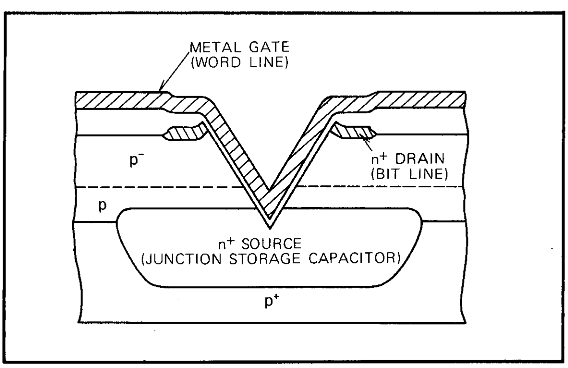


Fig. 1.4 VMOS dynamic RAM cell cross-section

5. Stacked Capacitor RAM Cell

To form 2-element cells, the data storage capacitance can be implemented as a MOS capacitor, MIS capacitor (using for example silicon nitride, Si_3N_4, as a dielectric), or the junction capacitance. Fig. 1.5 shows the example of an MIS capacitor being used. The capacitor is formed by a layer of tantalum (Ta) and a layer of molybdenum (Mo) between which is a layer of tantalum oxide (Ta_2O_5). The dielectric coefficient of Ta_2O_5 being 22, much higher than that of silicon at 3.9, enables a very small cell area to be used to store a large electric charge. The transfer gate is formed by the shallow junction between the gate polysilicon and source drain and the deep n^+ region and contact hole which form a quadruply self-aligned (QSA)-MOST, enabling an improvement in level of integration. For devices designed on a 2μm grid, the space required for a 512K-bit and 1M-bit RAM is 4.8 x 9.5 and 9.2 x 9.5mm respectively, placing megabit memory devices in the realm of possibility at last. The device process involves heretofore undeveloped technologies such as those required to process Ta, Ta_2O_5 and Mo, placing some stumbling blocks in the way of development of this new technology. As with other techniques, however, the common problem of lowering the resistivity of polysilicon exists.

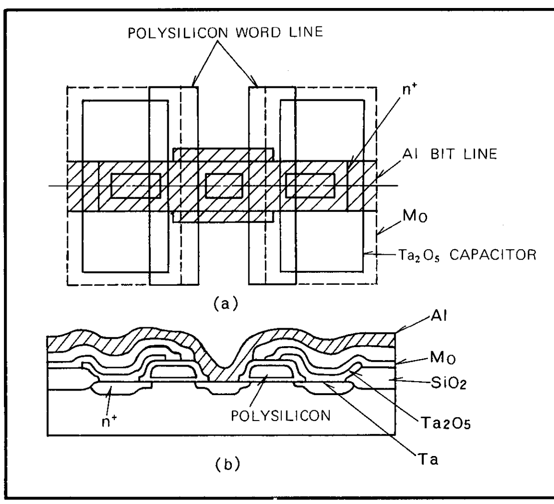


Fig. 1.5 Stacked capacitor top view (a), and cross-section (b)

6. CCRAM (Charge Coupled RAM) Cell

With the exception of the VMOS cell which makes use of the junction capacitance only for data storage, 2-element cells require three lines; storage gate, bit, and word lines. The allowance of space for these three lines results in a lowering of space usage efficiency. To overcome this disadvantage, the CCRAM cell uses one gate for both the storage gate and transfer gate functions. Fig. 1.6 (a) shows an example of the cross-section structure of a CCRAM cell, illustrating the shallow n^+ and relatively deep p^+ regions directly under the storage gate resembling a Hi-C region in the silicon substrate surface. The n^+ region is designed to shift the flat band voltage V_{FB} so as to separate the transfer gate and n^+ bit lines by the stored data (or potential void) with the transfer gate (word line) off and data stored. When the transfer gate is turned on (for read or write), the potential directly below the gate becomes more positive than the potential on the surface directly below the storage gate (due to the p^+ region of the storage gate), so that electrons freely flow in the bit line (Fig. 1.6 (b)). This operation is exactly the same as previously developed 2-element cells. Process problems, similar to those of the Hi-C cell, include the self-alignment of the n^+-p^+ regions, the setting of the flat band voltage, and operating margin sensitivity to process parameter variations.

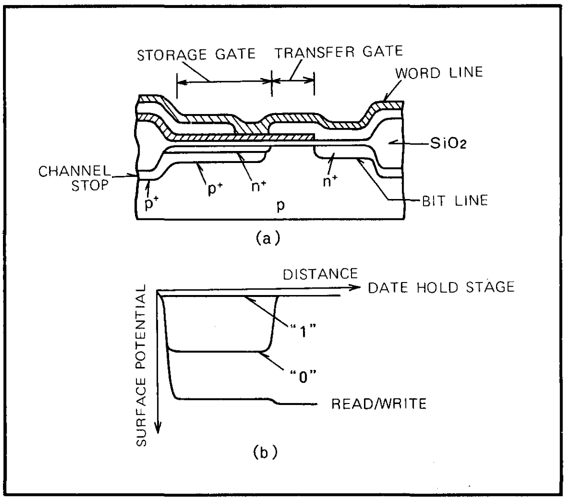


Fig. 1.6 CCRAM cell cross-section (a), and surface potential (b)

7. MCM (Merged Charge Memory) Cell

Whereas the CCRAM cell makes use of a combined storage gate and transfer gate, the MCM cell is one in which the bit line and storage gates are combined. In the structure shown in Fig. 1.7, the charge representing data is stored in the form of minority carriers in the potential void directly below the intersection of the word line and BSS (bit storage/sense) line. Depending upon whether the write data is zero or one, the DSS line potential V_{BH} will be set to a high value (forming a deep potential void) or a middle value of $V_{BH}/2$ (forming a shallow potential void). By setting the word line potential to approximately $V_{BH}/2$, the charge from the n^+ diffused line is stored as a zero or a one in the potential void. The holding of zero and one data from thereon with BSS line potential as V_{BH} is done with the potential void filled either half way or virtually empty. For reading, the BSS line is floated, and the n^+ diffused line is set to ground potential. The word line is then set from a negative potential (the potential for data hold) to a value of $V_{BH}/2$. The charge flowing out of the n^+ diffused line fills its potential void, while reading is done by means of a source follower circuit used to output as one or zero the voltage drop caused by the BSS line capacitor junction corresponding to the originally empty void. For this structural reason, the MCM cell has no contact hole and does not necessitate the alignment of pn junctions, such that if W is the minimum line width, the surface area for one cell becomes approximately $4W^2$ as the theoretically smallest value. Difficulties arise however from the fact that the charge per unit area is low, the ratio of BSS line to storage capacitance is large, and the pitch is narrow, creating problems in the implementation of sense amplifier and decoding circuits (at present, sense circuits fitting within a 2W pitch have not been devised).

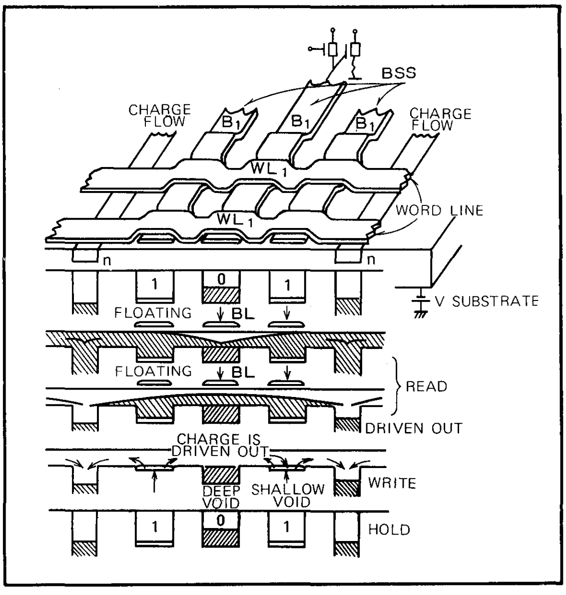


Fig. 1.7 MCM cell conceptual diagram

8. Taper Isolated RAM Cell

The cell previously described are combinational forms using basically two elements to store one piece of data and requiring one transfer gate. Texas Instruments have developed the taper isolated dynamic gain (TIDG) RAM in which the threshold levels for one and zero of a single MOS transistor have been changed, eliminating the requirements for a capacitor for writing and hold. Operation is similar to dynamic RAM, with reading done by detection of differences in current drive capabilities of the transistor caused by threshold value differences in comparison with fixed charge reading used with previously developed cells. Fig. 1.8 shows the equivalent circuit for a TIDG cell as well as the cross-section cut in the channel width direction and the surface potential. The p and n implanted levels and diffusion depth are chosen such that a potential higher than that directly below the gate oxide layer is formed in the taper between the thick oxide layer and the thin gate oxide layer, so that a potential barrier is formed as shown in Fig. 1.8. The result of this is that, directly below the gate, sufficient holes accumulate to form a channel (low threshold level) or that the majority of holes are driven out such that a channel does not form (high threshold level), data storage being determined as one or zero depending on which situation exists. Data writing (zero) is performed by driving holes out of the substrate and channel stop regions by applying a high potential to the gate and low potential to the source and drain. Because this type of cell requires only one polysilicon layer, it is thought to have great applications in the implementation of megabit scale memories. Variations in characteristics, however, are seen to be difficult to control.

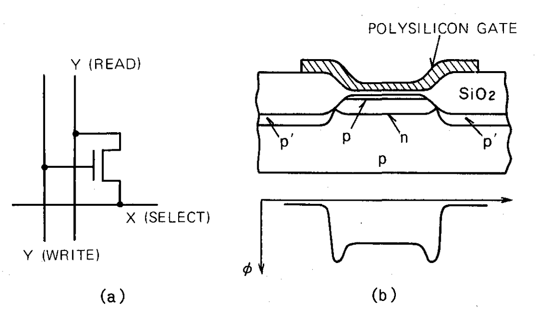


Fig. 1.8 Equivalent circuit (a) and cross-section (b) of the Texas Instruments TIDG cell

9. Mostek ROM Cell

While it is not a dynamic RAM cell, the 64K ROM cell of the Mostek Company is of some interest. Fig. 1.9 shows the top view of the device. The transistor threshold is programmed as either high or low level. The column line is high-level for the standby state, and changes to a virtual ground for the decoded condition, outputing the precharge bit line potential as is or at a lower level as data. This cell and the associated circuit configuration was announced at the ISSCC simultaneously as an EPROM application by Motorola, Mostek, and Texas Instruments. For the EPROM application, a floating gate is fabricated at the upper part of the transistor shown in Fig. 1.9. This cell structure and surrounding circuit configuration is usable in such applications as the previously described taper isolated cell and is expected to yield advancements in that area.

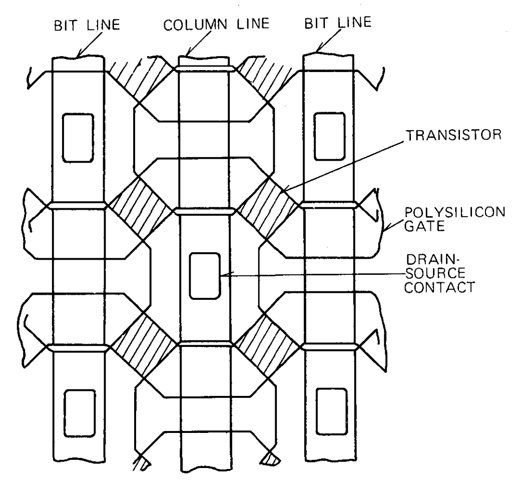


Fig. 1.9 Mostek 64K ROM cell

Saving Failed Bits

As a separate but related field to memory cell structure, the saving of failed memory bits is being studied and several methods have been attempted. One method, developed by device users, results in the effective usage of a device within which several failed bits or even one half of the entire device has been detected to be bad. The other relies on the provision of spare bits to form a redundant backup system for failed bits. Here we will discuss several examples of the latter method.

1. Bell Laboratories — Fault Tolerant 64K RAM

This method relies on the external creation of shorts and open circuits between spare bits and failed bits. The Bell Laboratories example consists of the use of a laser to open circuits by cutting the polysilicon material. The addresses of failed bits detected during a wafer test are recorded. A decoding circuit is then used to process addresses. Fig. 1.10 shows a possible laser program decoding circuit. The laser is used to cut the polysilicon links which are imbedded below the passivation layer. The decoding circuit is used to select the spare bit lines. The laser is programmed such that the stage subsequent to the selected line output is clamped to a low level. Since this method relies on laser cutting, it presents some practical difficulties. The Mostek method to be discussed next eliminates these difficulties by electrically saving the failed bits and is expected to be a useful technique for the future.

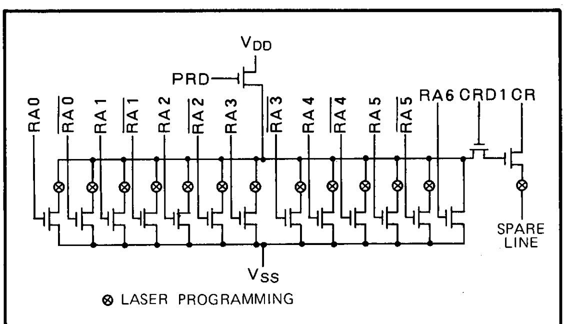


Fig. 1.10 Bell Laboratories failed bit saving decoder circuit

2. Mostek Example — 5V 64K EPROM

Mostek applies failed bit saving techniques to EPROM devices. To improve yield, 8K of spare cells, spare column decoder, spare column selector, spare sense amplifier, and spare data input buffer are provided in addition to the 64K-bits of memory cells. When a failed bit is detected during a wafer test, the $\overline{CE}$ input is driven at 25V, the result of which is that (see Fig. 1.11) RPR becomes 25V and $\overline{RPR}$ goes from 5V to 0V. Transistor Q_1 turns on, and the polysilicon resistance R is cut by the voltage V_{PP} = 25V. In this way, the repair buffer selects the spare matrix. The advantage of this method is, of course, that the entire process is handled electrically.

Another example of electrical saving of failed bits was disclosed by Japan Telephone and Telegraph. In this example a failed bit is detected during the wafer test and the connection to the spare decoder corresponding to this address is made by forming a short circuit by destroying a polysilicon p-n junction. These examples are methods that can be used to increase yield of complete 64K and 256K devices. Another technique which might be used is that which provides the device with an onboard correction circuit which can save the device from even soft errors. Whichever technique is adopted, it is clear that technological breakthroughs are going to be necessary if progress in large-scale memory development is to continue.

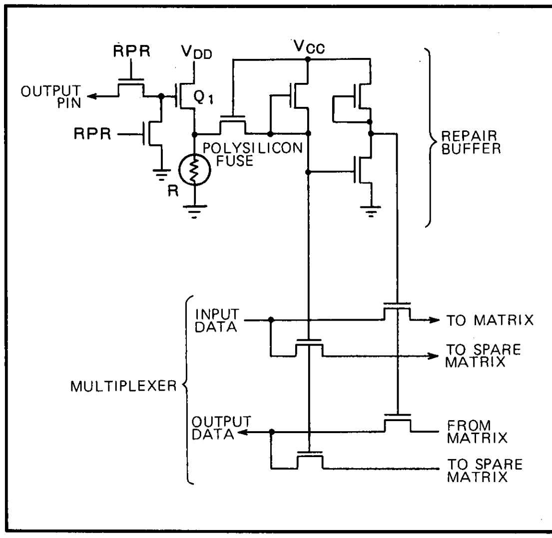


Fig. 1.11 Mostek EPROM failed bit saving method

Summary

As can be seen, various approaches are being attempted to the development of megabit memory devices, including study in the area of effective usage of such devices, device structure, and recovery from soft errors, all of which is progressing in spite of inherent process limitations. By the year 2000 it is expected that 4M-bit devices with access times of 1μs will be producible in an area of 4cm^2. To provide this new technology required by our modern society, the semiconductor industry will have to mobilize its circuit device and process technologies in a concerted effort.

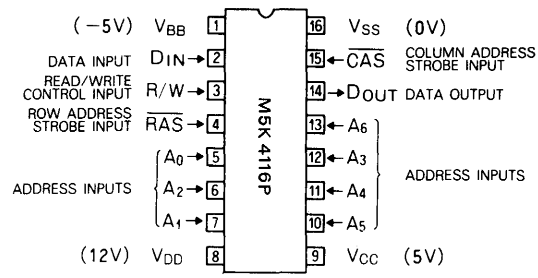
2. 16K-BIT DYNAMIC RAM

2.1 M5K4116P,S TECHNOLOGY

INTRODUCTION

The M5K4116P, S are 16 384-word by 1-bit dynamic RAMs, fabricated using the n-channel silicon-gate MOS process, and ideal for large-capacity memory systems where high speed, low power dissipation, and low costs are essential. The use of double-layer polysilicon process technology and single-transistor dynamic storage cells provide high circuit density at reduced costs, and the use of dynamic circuitry including sense amplifiers assures low power dissipation. Multiplexed address input permits both a reduction in pins to the standard 16-pin package configuration and an increase in system densities.

Table 2 compares the M5K4116P, S 16 384-bit dynamic RAM with a 4096-bit static RAM.



Outline 16P1 (M5K4116P)
16S1 (M5K4116S)

Fig. 2.1 Pin configuration (top view)

Table 2.2 compares that the requirements of the two RAM types when a 16K-byte memory system is constructed.

Table 2.1 Comparison of the 16,384 bit dynamic RAM and 4K static RAM

Characteristics \ Device	16K dynamic RAM	(Note 1) 4K static RAM
Total power	462mW max	440mW max
Power/bit	28.2μW	107.4μW
Speed	t_a=150ns	t_a=200ns
Power x speed/bit	4.23pJ	21.5pJ

Note 1 : M5L2114P-2

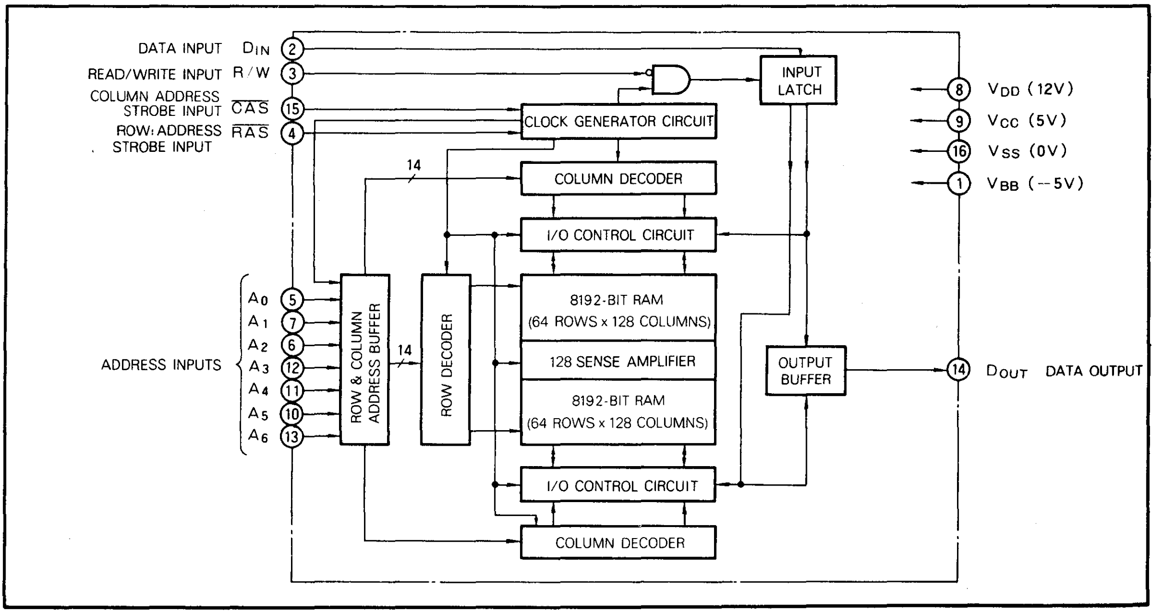
As can be seen, the power × speed per bit of the 16K dynamic RAM is 4.23pJ, only 1/5 that of the 4K static RAM.

Table 2.2 Requirements for a 16K-byte memory system

Device	Number of RAMS	Voltage	Current	Over-all power	Relative power	Relative size
4K-bit static RAM	32	5V	@2.56A	12.8W	1	1
16K-bit dynamic RAM	8	5V 12V −5V	* @0.28A @2mA	3.37W	0.26	0.25

*Current from V_{CC} is neglected because V_{CC} is only connected to output buffer.

Fig. 2.2 Block diagram



FUNCTION

In addition to normal read, write, and read-modify-write operations, the M5K4116P, S provide a number of other functions, e.g., page-mode, $\overline{RAS}$-only refresh, and delayed-write. The input conditions for each are shown in Table 2.3

If you interchange address pins as shown in Fig. 2.3, you can get a sequential location map for the 16,384 memory bits.

Table 2.3 Input conditions for each mode

Operation	Input						Output	Re-fresh	Remarks
	$\overline{RAS}$	$\overline{CAS}$	R/W	D_{IN}	Row address	Column address	D_{OUT}		
Read	ACT	ACT	NAC	DNC	APD	APD	VLD	YES	Page mode is identical except refresh is NO.
Write	ACT	ACT	ACT	VLD	APD	APD	OPN	YES	
Read-mondify-write	ACT	ACT	ACT	VLD	APD	APD	VLD	YES	
$\overline{RAS}$-only refresh	ACT	NAC	DNC	DNC	APD	DNC	OPN	YES	
Standby	NAC	DNC	DNC	DNC	DNC	DNC	OPN	NO	

Note 2 : ACT. active NAC ; non-active DNC; don't care VLD ; valid APD: applied OPN; open

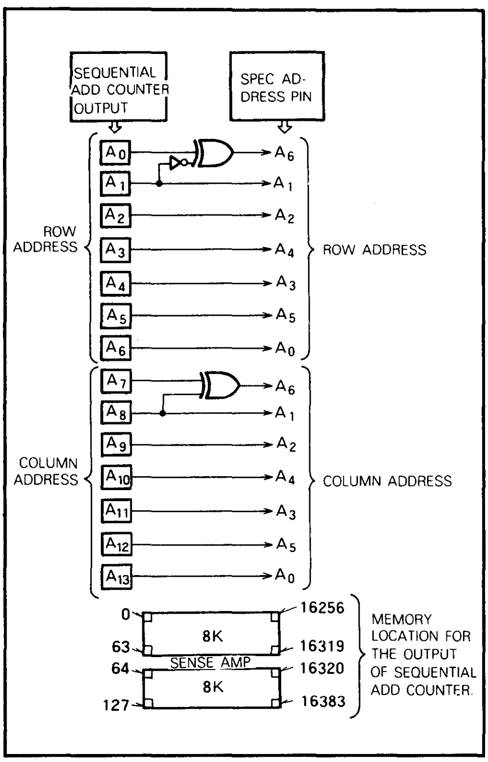


Fig. 2.3 Method for converting sequential address

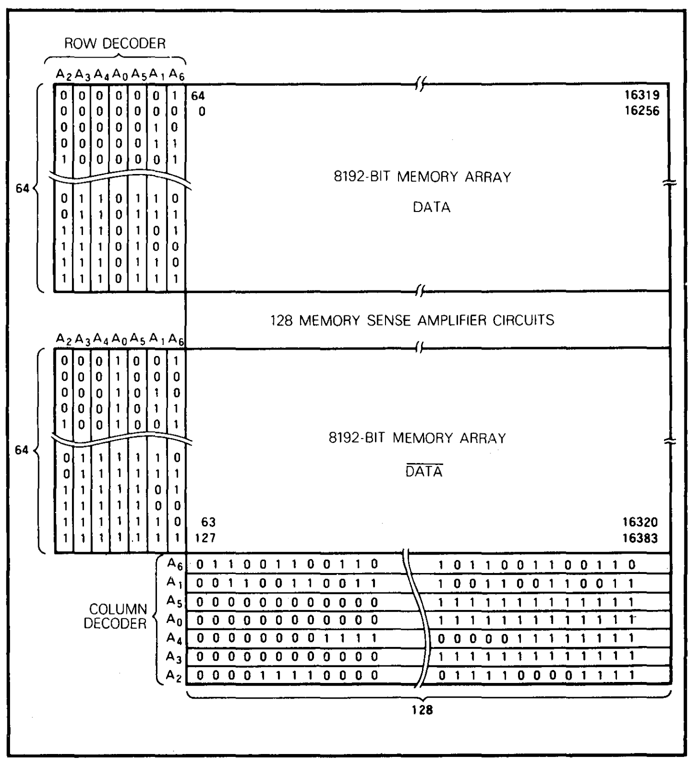


Fig. 2.4 M5K4116P, S memory map

N-CHANNEL DOUBLE-LAYER POLY SILICON GATE MOS PROCESS

In order to fabricate the M5K4116P, S series, single transistor memory cells and the N-channel double-layer polysilicon gate MOS process are used. There is no diffusion area between switching transistor Q and the data-storage memory capacitor because of the use of the double polysilicon gate MOS process, so that the memory cell area is reduced by 75% from that of the previous process.

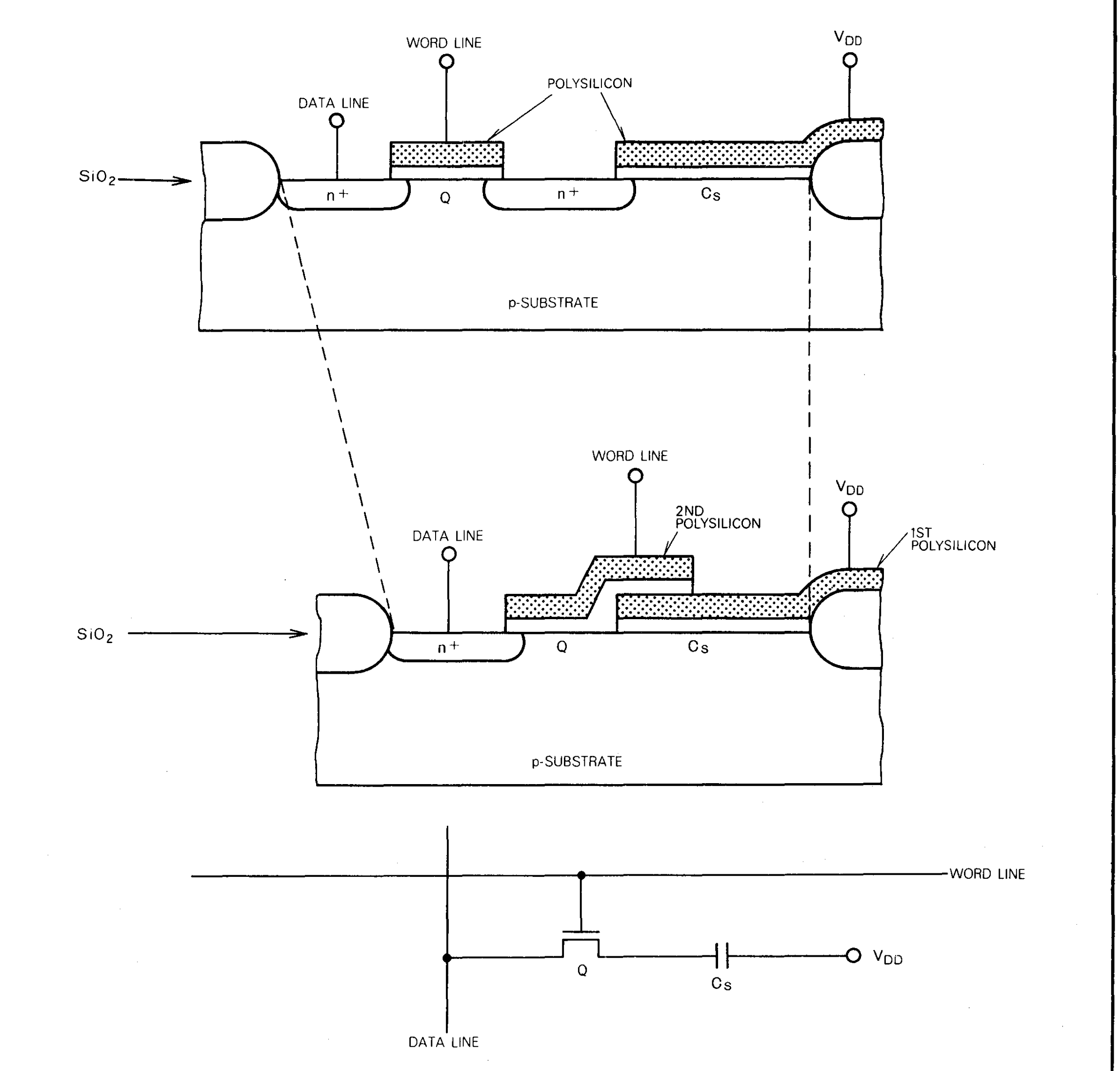
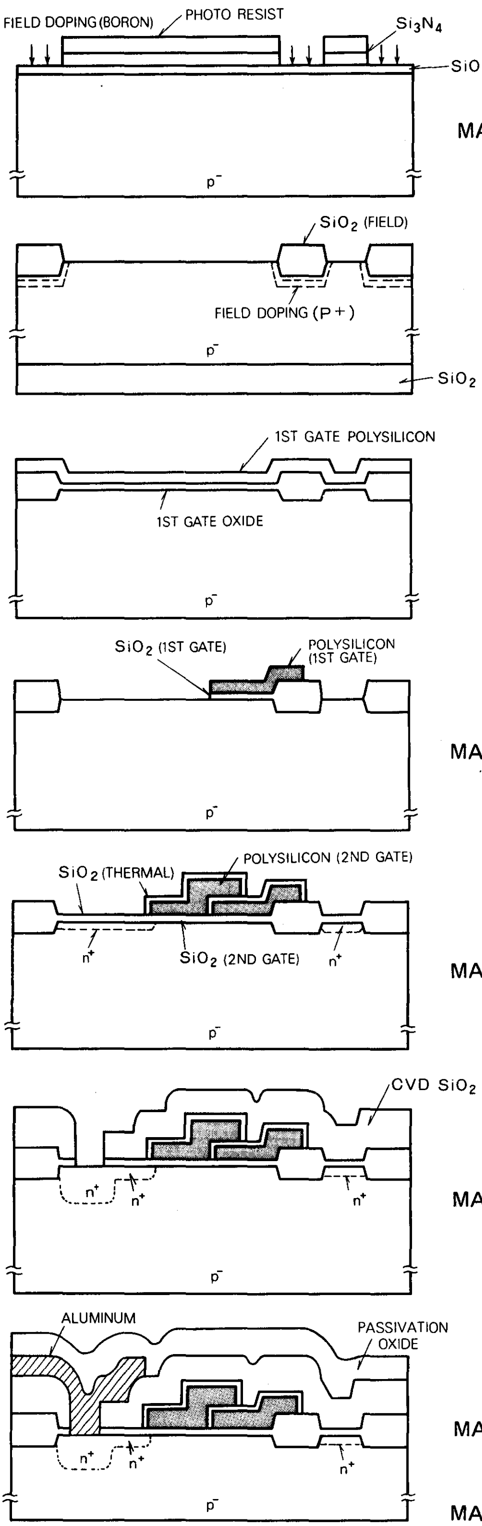


Fig. 2.5 Memory cell structure



MASK 1 — Deposit silicon nitride (Si_3N_4) on wafer and coat with photo resist. Define active area with MASK 1 and implant fields.

Remove photo resist and grow field oxide using nitride as mask.

Grow 1st gate oxide and deposit 1st polysilicon.

MASK 2 — Define 1st polysilicon gate using MASK 2.

MASK 3 — Define 2nd polysilicon gate and diffuse phosphorus N^+ dopant for source and drain.

MASKs 4, 5 — Deposit oxide and define aluminum contact with MASKs 4 and 5.

MASK 6 — Deposit aluminum and define bonding pad metal interconnect with MASK 6.

MASK 7 — Deposit passivation oxide and define bonding pad openings with MASK 7.

Fig. 2.6 Water manufacturing process

SUMMARY OF OPERATIONS

Addressing

To select one of the 16 384 memory cells in the M5K 4116P, S, the 14-bit address signal must be multiplexed into 7 address signals, which are then latched into the on-chip latch by two externally applied clock pulses. First, the negative-going edge of the row-address-strobe pulse ($\overline{RAS}$) latches the 7 row address bits; next, the negative-going edge of the column-address-strobe pulse ($\overline{CAS}$) latches the 7 column-address bits. Timing of the $\overline{RAS}$ and $\overline{CAS}$ clocks can be selected by either of the following two methods.

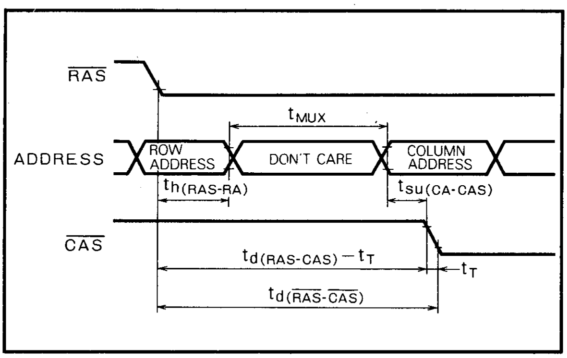


Fig. 2.7 Address multiplex

1. The delay time from $\overline{RAS}$ to $\overline{CAS}$ $t_{d(RAS-CAS)}$ is set between the minimum and maximum values of the limits. In this case, the internal $\overline{CAS}$ control signals are inhibited until almost $t_{d(RAS-CAS)max}$ ('gated $\overline{CAS}$' operation). The external $\overline{CAS}$ signal can be applied with a margin not affecting the on-chip circuit operations (e.g. access time), and the address inputs can easily be changed from row address to column address. This interval is called the 'multiplex time'. Eq. 1 gives the multiplex time.

   $t_{MUX} = t_{d(RAS-CAS)} - t_T - t_{h(RAS-RA)} - t_{SU(CA-CAS)}$ .... Eq. 1

   In the next conditions, the multiplex time (t_{MUX}) is maximized.

   $t_{d(RAS-CAS)} = \max$

   $t_{h(RAS-RA)} = \min$

   $t_{su(CA-CAS)} = \min$

   Table 4 shows the maximum multiplex time in the case where the access time is not greater than $t_{a(RAS)MAX}$.

Table 2.4 Maximum multiplex time

Type number	t_{MUX}	$t_{d(RAS-CAS)}$	$t_{h(RAS-RA)}$	$t_{su(CA-CAS)}$
M5K4116P, S-2	35 ns	50 ns	20 ns	− 10 ns
M5K4116P, S-3	45 ns	65 ns	25 ns	− 10 ns
M5K4116P, S-4	55 ns	85 ns	35 ns	− 10 ns

Note 3 : t_T =5ns

2. The delay time $t_{d(RAS-CAS)}$ is set greater than the maximum value of the limits. In this case the internal inhibition of $\overline{CAS}$ has already been released, so that the internal $\overline{CAS}$ control signals are controlled by the externally applied $\overline{CAS}$, which also controls the access time.

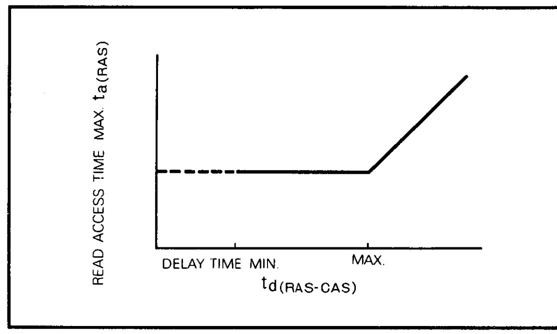


Fig. 2.8 Read access time vs delay time

Data Input

Data to be written into a selected cell is strobed by the later of the two negative transitions R/W input and $\overline{CAS}$ input. Thus, when the R/W input makes its negative transition prior to the $\overline{CAS}$ input (early write), the data input is strobed by the $\overline{CAS}$, and the negative transition of the $\overline{CAS}$ is set as the reference point for setup and hold times. In the read-write or read-modify-write cycles, however, when the R/W input makes its negative transition after the $\overline{CAS}$, the R/W negative transition is set as the reference point for set-up and hold times.

Data Output Control

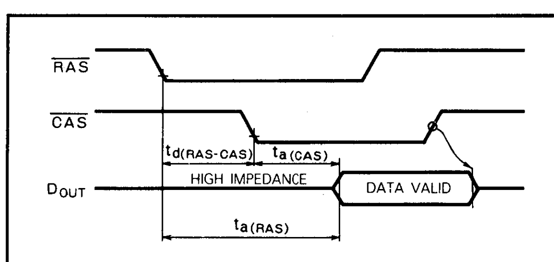


Fig. 2.9 Read cycle

The output of the M5K4116P, S is in the high-impedance state when the $\overline{CAS}$ is high. When the memory cycle in progress is a read, read-modify-write, or a delayed-write cycle, the data output will go from the high-impedance state to the active condition, and the data in the selected cell will be read. This data output will have the same polarity as the input data. Once the output has entered the active condition, this condition will be maintained until the $\overline{CAS}$ goes high, irrespective of the condition of the $\overline{RAS}$ (to a maximum of 10μs).

The output will remain in the high-impedance state throughout the entire cycle in an early-write cycle.

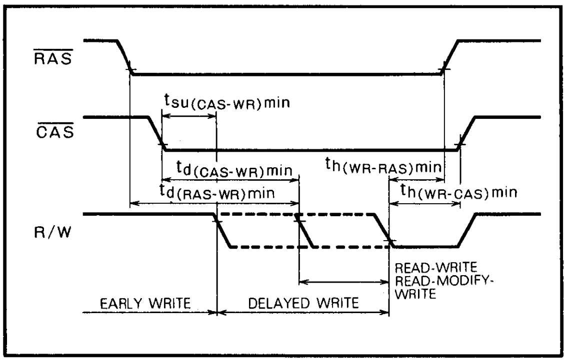


Fig. 2.10 Write cycle

Table 2.5 Output state in write cycle

Operation mode	Output state
Early write	High impedance
Read-write, read-modify-write	Data valid
Others	Unspecified

These output conditions of the M5K4116P, S, which can readily be changed by controlling the timing of the write pulse in a write cycle, and the width of the $\overline{CAS}$ pulse in a read cycle, offer capabilities for a number of applications, such as the following.

1. Common I/O Operation

If all write operations are performed in the early-write mode, input and output can be connected directly to give a common I/O data bus.

2. Data Output Hold

The data output can be held between read cycles, without lengthening the cycle time, until the next cycle commences. This enables extremely flexible clock-timing settings for $\overline{RAS}$ and $\overline{CAS}$.

3. Two Methods of Chip Selection

Since the output is not latched, the $\overline{CAS}$ is not required to maintain the output of selected chips in the matrix in a high-impedance state. This means that the $\overline{CAS}$ and/or the $\overline{RAS}$ can both be decoded for chip selection.

4. Extended-Page Boundary

By decoding $\overline{CAS}$, the page boundary can be extended beyond the 128 column locations on a single chip. In this case, the $\overline{RAS}$ must be applied to all devices.

Page-Mode Operation

This operation allows for multiple-column addressing at the same row address, and eliminates the power dissipation associated with the negative-going edge of the $\overline{RAS}$ because once the row address has been strobed, the $\overline{RAS}$ is maintained. Also, the time required to strobe in the row address for the second and subsequent cycles is eliminated, thereby decreasing access and cycle times.

Refresh

Refreshing of the dynamic cell matrix is accomplished by performing a memory operation at each of the 128 row-address locations within a 2ms time interval. Any normal memory cycle will perform the refreshing, and the $\overline{RAS}$-only refresh offers a significant reduction in operating power.

Power Dissipation

Most of the circuitry in the M5K4116P, S is dynamic, and most of the power is dissipated when the addresses are strobed. Both the $\overline{RAS}$ and the $\overline{CAS}$ are decoded and applied to the M5K4116P, S as chip-select in the memory system, but if the $\overline{RAS}$ is decoded, all unselected devices go into stand-by independent of the $\overline{CAS}$ condition, minimizing system power dissipation.

Stand-By Current-Refresh Only

The I_{DDSB} (stand-by current of V_{DD}) and the (stand-by current of V_{BB}) are calculated by the following equations.

1. $\overline{RAS}/\overline{CAS}$ refresh

$$I_{DDSB}=I_{DD1(AV)} \times \left\{128 \times \frac{t_C}{t_{C(REF)}}\right\} + I_{DD2} \times \left\{1-(128 \times \frac{t_C}{t_{C(REF)}})\right\} \quad \text{.... Eq. 2}$$

$$I_{BBSB}=I_{BB1(AV)} \times \left\{128 \times \frac{t_C}{t_{C(REF)}}\right\} + I_{BB2} \times \left\{1-(128 \times \frac{t_C}{t_{C(REF)}})\right\} \quad \text{.... Eq. 3}$$

Assuming that t_C=375ns, $I_{DD1(AV)}$=35mA, $I_{BB1(AV)}$=200μA, I_{DD2}=1.5mA, I_{BB2}=100μA, $t_{C(REF)}$=2 ms, we can obtain following results:

I_{DDSB}=35mA x 0.024 + 1.5mA x 0.976=2.3mA

I_{BBSB}=200μA x 0.024 + 100μA x 0.976=102μA

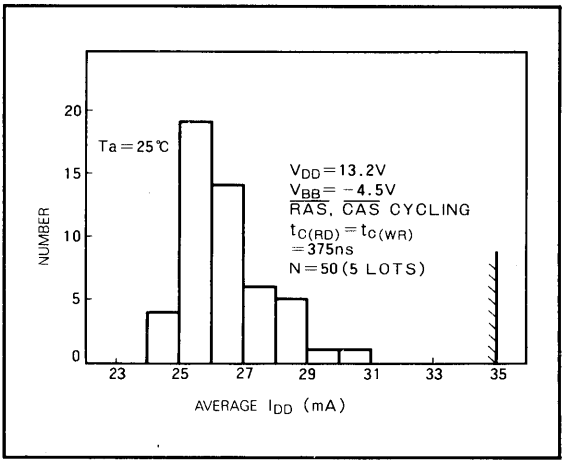


Fig. 2.11 Distribution of average I_{DD}

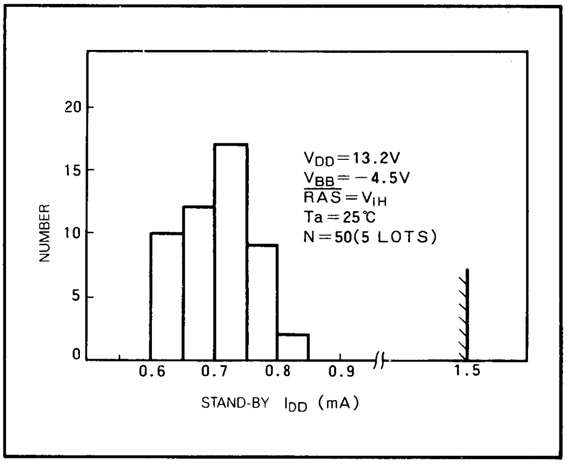


Fig. 2.12 Distribution of stand-by I_{DD}

2. $\overline{RAS}$-only refresh

$$I_{DDSB}=I_{DD3(AV)}\times\left\{128\times\frac{t_C}{t_{C(REF)}}\right\}+$$
$$I_{DD2}\times\left\{1-(128\times\frac{t_C}{t_{C(REF)}})\right\}\ \text{.... Eq. 4}$$
$$I_{BBSB}=I_{BB3(AV)}\times\left\{128\times\frac{t_C}{t_{C(REF)}}\right\}+$$
$$I_{BB2}\times\left\{1-(128\times\frac{t_C}{t_{C(REF)}})\right\}\ \text{.... Eq. 5}$$

Assuming that $I_{DD3\ (AV)}$=27mA, $I_{BB3(AV)}$=200μA, we obtain the following results:

I_{DDSB}=27mA x 0.024 + 1.5mA x 0.0976=2.1mA

I_{BBSB}=200μA x 0.024 + 100μA x 0.0976=102μA

Stand-by current is about 2.1mA. Therefore, by using low-power refresh and external circuits, it is possible to use a battery back-up system.

Power Supplies

Although the M5K4116P, S require no particular power-supply sequencing so long as the devices are used within the limits of the absolute maximum ratings, it is recommended that the V_{BB} supply be applied first and removed last. V_{BB} should never be more positive than V_{SS} when power is applied to V_{DD}. Generally, when V_{DD} is applied and V_{BB} is not applied, stand-by current is larger than that in the normal state. Table 6 shows this effect.

Some eight dummy cycles are necessary after power is applied to the device before memory operation is achieved. Dummy cycles must be executed by the $\overline{RAS}/\overline{CAS}$ refresh cycles or $\overline{RAS}$-only refresh cycles.

Table 2.6 Change of stand-by current

Condition \ Device	#1		#2		#3		#4		Unit
	$I_{DD1(AV)}$	I_{DD2}	$I_{DD1(AV)}$	I_{DD2}	$I_{DD1(AV)}$	I_{DD2}	$I_{DD1(AV)}$	I_{DD2}	
$V_{BB}=-5V$	25.3	0.71	26.0	0.73	25.9	0.69	24.9	0.72	mA
$V_{BB}=0V$	28.0	0.76	28.8	0.78	28.7	0.74	27.6	0.76	mA
Change +%	+10.7	+7.0	+10.8	+6.8	+10.8	+7.2	+10.8	+5.6	%

2.2 16K-BIT DYNAMIC RAM APPLICATION

APPLICATIONS FOR DYNAMIC RAM

Dynamic RAM (Ramdom-Access Memory)s can be very effective components in the implementation of reliable, high-performance, low-cost memory systems. However, these devices have several requirements that should be considered.

Bit-Cell Structure

First, consider the dynamic memory bit cell, which is quite unlike the cell of a static RAM. Fig. 2.13 shows a typical single-transistor memory bit cell. The bit cell consists of a transistor and a capacitor that constitute a "sample and hold" circuit.

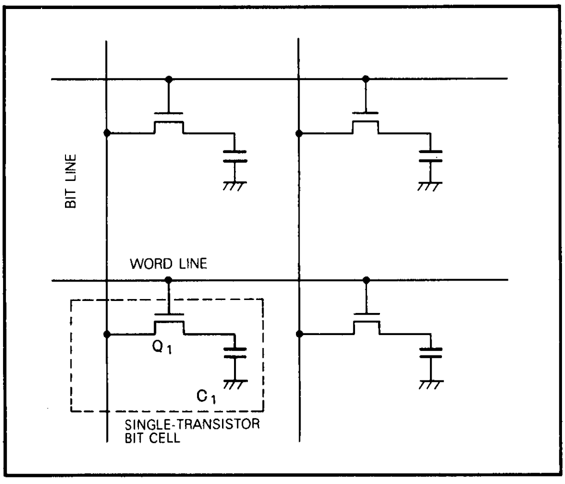


Fig. 2.13 Single-transistor memory bit cell

During the write operation, the selected word line is brought to an active state (high). This causes the bit cell transistor Q_1 to turn "On" and the data that is placed on the bit line is stored in the capacitor C_1. The stored data is retained even if transistor Q_1 turns "Off".

During the read operation, the selected line is brought to an active state (high) again, and the capacitor voltage is placed on the bit line. At this time, the read-out data is amplified and rewritten on the capacitor internally.

Because of the theory governing dynamic memory storage, capacitor charge in the cell will gradually leak off, and the stored data will be lost.

For example, a 1nA leakage current discharging a 1pF capacitor results in a voltage change of 1V per ms. The storage time of M5K4116P, S is shown in Fig. 2.14. If data is to be retained for longer than the self-discharge time of the cell storage capacitor, typically 2ms, the data must be sensed before it is lost and then restored to its original voltage level.

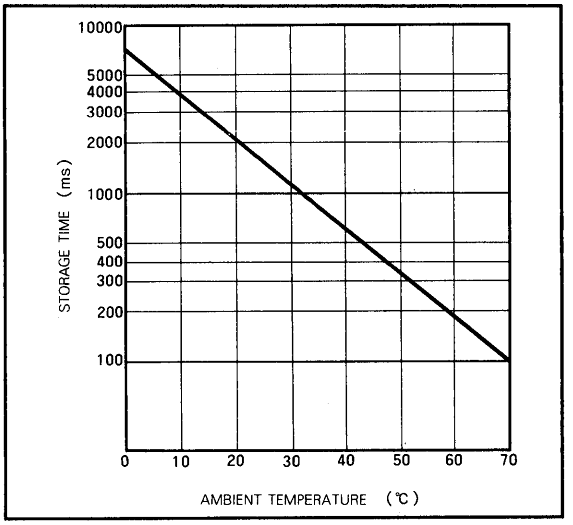


Fig. 2.14 Storage time vs. ambient temperature

Refresh

Thus one can see that the refresh function is a very important requirement for a charge-storage memory, i.e., a dynamic RAM. The dynamic memory controller must assure that every bit cell is refreshed periodically enough to maintain data integrity. The refresh interval is specified by the vendor, and a typical requirement is that each bit cell be refreshed every 2ms.

The M5K4116P, S are 16 384-bit memories constructed with 128 rows and 128 columns. All columns in a single row in an array are refreshed simultaneously. This means that the user must supply 128 refresh cycles each 2ms.

In order to supply the refresh row address, a refresh counter (7 bits) is required and is incremented after each refresh cycle. A "two inputs to one output" multiplexer is also used to multiplex either the system-supplied memory address or the refresh counter-supplied address onto the dynamic memory row address inputs.

Refresh Techniques

In most memory systems it is difficult to guarantee that normal memory operations will cause all the rows within a memory to be sensed within the specified refresh interval. For this reason, most dynamic memory systems have special circuitry that will cause all rows of memory cells to be sensed within the 2ms interval.

There are three commonly used techniques for refreshing the memories. The first is "burst mode refresh" where all memory accesses are inhibited for a fixed period of time while all rows are continuously accessed. This mode is shown in Fig. 2.15 (a). The second is "cycle steal mode," where a single memory cycle is periodically stolen from the processor in order to refresh a single row. This mode is shown in Fig. 2.15 (b). The third is called "invisible or transparent mode," where refresh cycles are introduced at the times when the memory is not being accessed and thus refresh is invisible to the processor. (The processor sees no delay due to the refresh function.) This mode is shown in Fig. 2.15 (c). The memory cycle of the invisible refresh mode is generally longer than that of the first or second method because single memory access continues after single memory access.

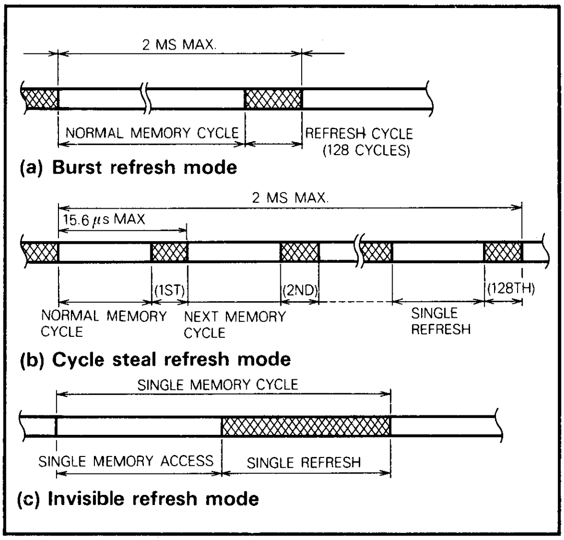


Fig. 2.15 Refresh techniques

Design Example

In designing dynamic memory systems, it is important to decide whether the memory refresh will be synchronous with the processor or asynchronous. In synchronous refresh, the designer uses a system clock to trigger the refresh logic. In asynchronous refresh, however, the designer must provide for a local timer to trigger the refresh and memory access arbiter.

This example illustrates the asynchronous refresh method which is more popular than the synchronous refresh in of interfacing dynamic RAMs to microprocessors. The memory controller block diagram is shown in Fig. 2.16. There are two controllers which access the memory. One is the microprocessor, and the other is refresh timer which requests a memory refresh every 15.6 μs (MAX). The memory access arbiter decides to which request the memory cycle is allocated. If the two controllers generate the request simultaneously, the arbiter allocates the memory cycle to the refresh timer.

Memory timing logic generates the memory clock timing (i.e. $\overline{RAS}$, $\overline{CAS}$, R/W) in accodance with the memory cycles. This timing is shown in Fig. 2.18. Three multiplexers are used in the circuit of Fig. 2.17. In the normal memory cycle, the row address (ADR0~ADR6) or column address (ADR7~ADRD) is multiplexed by MPXCNT and $\overline{CPU\ ADREN}$. In the refresh cycle, the refresh address is present at MA0~MA6, which is gated by $\overline{REFADREN}$.

The refresh controller in Fig. 2.17 is also used in 64K dynamic RAM applications by changing the refresh address counter and microprocessor address multiplexer.

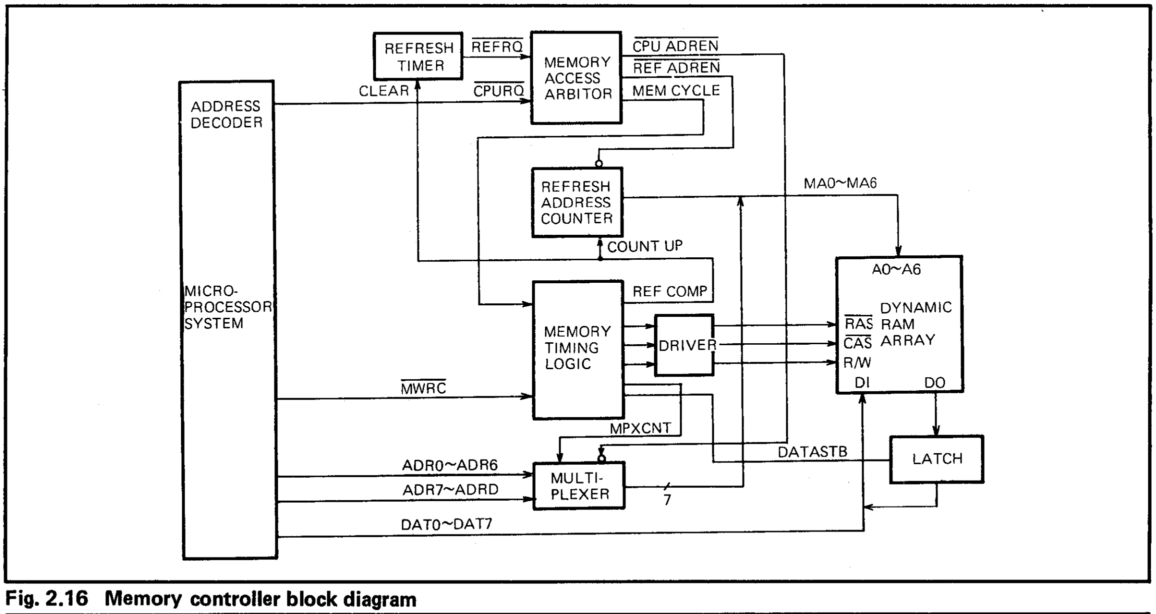


Fig. 2.16 Memory controller block diagram

MITSUBISHI ELECTRIC

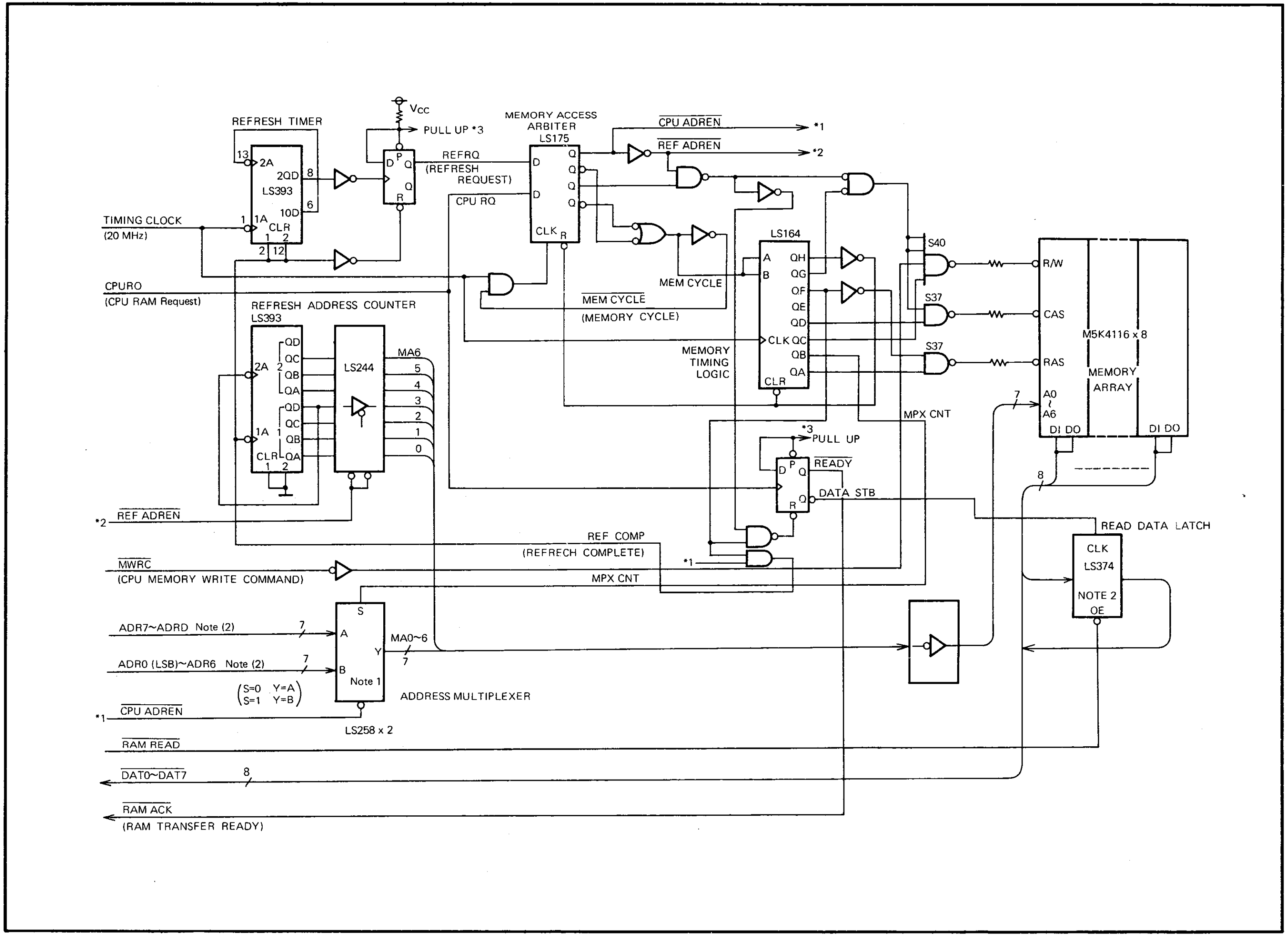


Fig. 2.17 Refresh controller logic

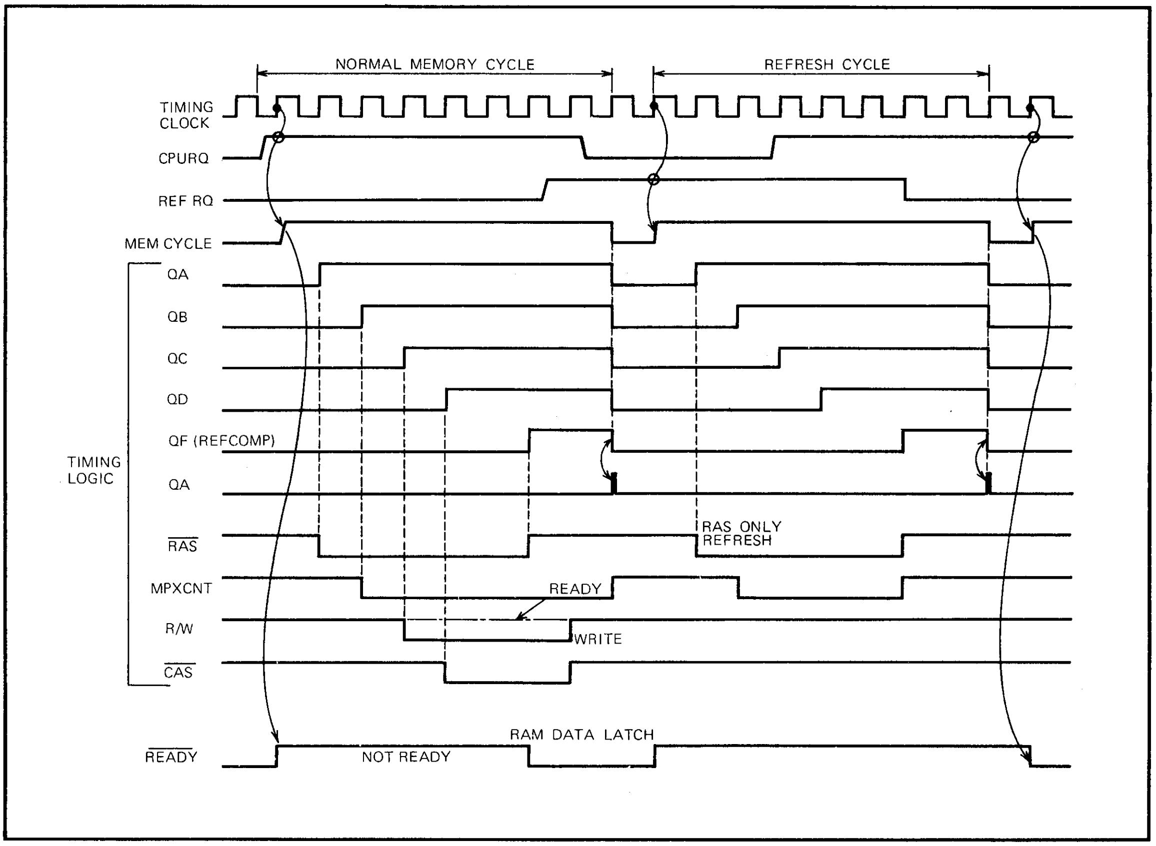


Fig. 2.18 Example of memory timing

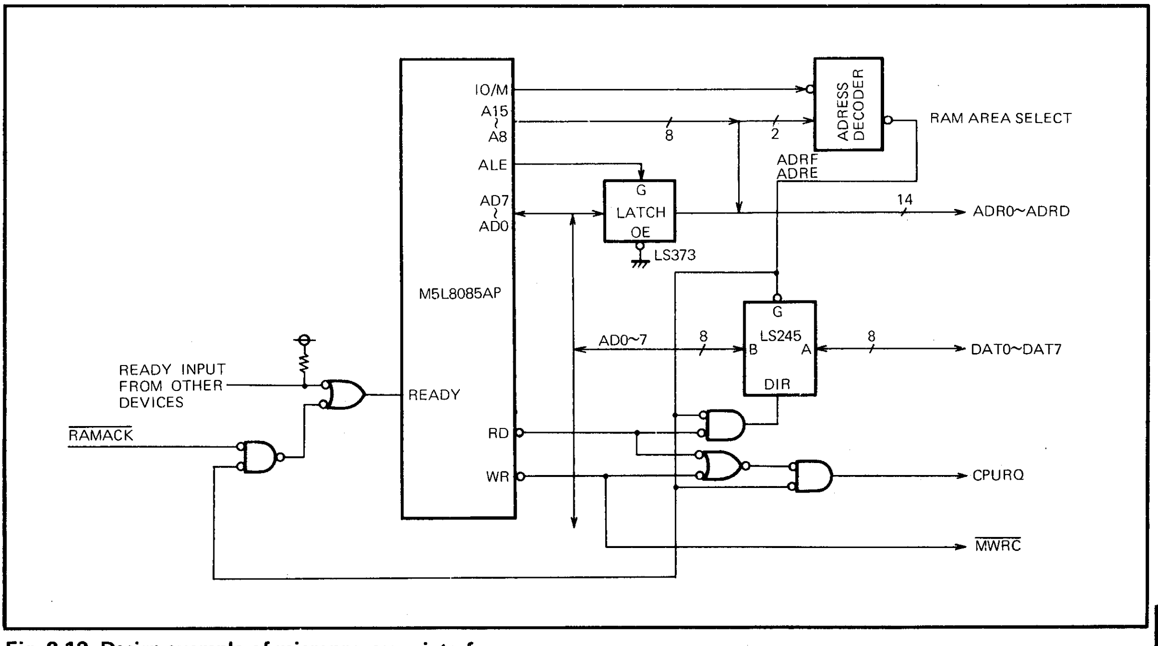


Fig. 2.19 Design example of microprocessor interface

15

Power Distribution and Decoupling Techniques

It should always be remembered that dynamic memories, while appearing to be rather simple digital devices, are in fact highly complex analog systems. They include differential sensing amplifiers that must detect deci volt signals buried in noise and must operate in tens of nanoseconds. For these reasons, the designer should respect the complexity involved and take the steps necessary to ensure a trouble-free design.

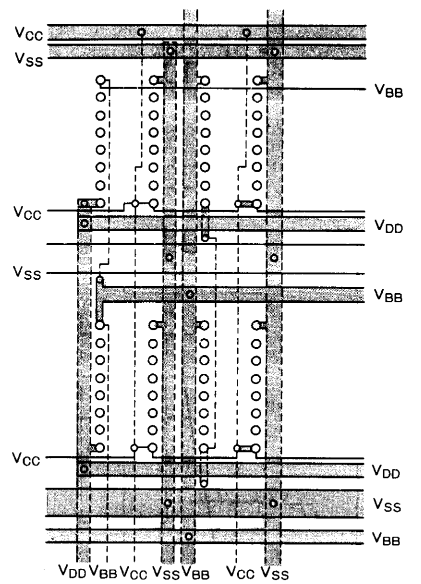
The layout of dynamic memories is of special importance. Typical I_{DD}, I_{BB} and I_{SS} current waveforms for the M5K4116P, S are shown in our data sheets. Distribution and decoupling techniques must be used to suppress these noises, which can cause data loss.

The layout should have an effectively gridded power-supply distribution network to supply adequate current and to minimize inductive effects. The distribution of circuit grounding is most important in reducing ground noise and inductive effects, and to provide a ground plane for the signal lines. An example of the power grid of the M5K 4116P, S is shown in Fig. 2.20, in which the decoupling capacitors are not shown.

In order to increase the effectiveness of the power grid, decoupling capacitors should be used. The capacitors required fall into two categories. The first consists of capacitors of small size and low inductance such as monolithic and other ceramic capacitors, which are adequate for suppression of transient noise. The second type consists of larger bulk capacitors used to prevent power supply drop. These also should be included within the memory array for good distribution.

The decoupling capacitors used in the memory array should be of a type that exhibits good high-frequency characteristics. It is recommended that a 0.1μF ceramic capacitor be connected between V_{DD} and V_{SS} at every other device in the memory array. It is also recommended that a 0.1μF ceramic capacitor be connected between V_{BB} and V_{SS} at every other device in the array, preferably the devices alternate to the V_{DD} decoupling. Decoupling of the V_{CC} is fairly noncritical. The capacitors are connected at the top and bottom of each column of memories.

In addition to the ceramic capacitor, it is recommended that a $2\sim5\mu$F tantalum or equivalent capacitor be connected between V_{DD} and V_{SS} adjacent to the array for each group of 16 memory devices. Use of a slightly smaller-value bulk capacitor is also recommended between V_{BB} and V_{SS}. An example of capacitor placement is shown in Fig. 2.21.



Note 7 : The dotted lines show the soldered side of the P.C. board.

Fig. 2.20 Suggested power grid for M5K41116P, S

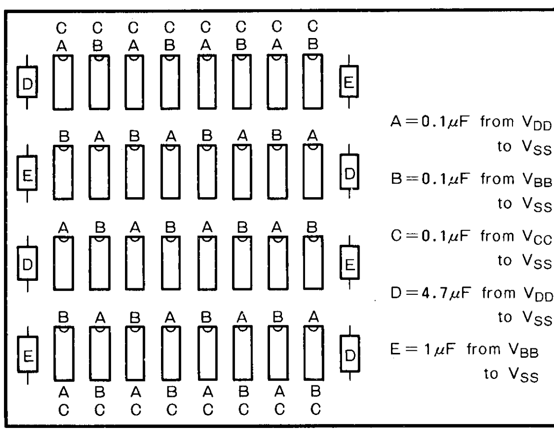


Fig. 2.2 Effective capacitor placement for the M5K4116P, S

Signal Lines Effects

By carefully laying out the circuit to minimize signal path length, one can reduce effects due to the transmission-line properties of the PC board. However, this may not be sufficient. It is necessary to add a series-terminating resistor to the output of the clock driver in order to match line impedances and damp out reflections caused by mismatching between the driver's source impedance and the characteristic impedance of the line.

In order to avoid to cross talk problems, all signal lines should be kept as short as possible. This implies that the signal drivers and receivers should be physically close to the memory array.

3. 64K-BIT DYNAMIC RAM

3.1 Technology

Since the introduction of the IK RAM in 1970, the development of dynamic RAM devices has progressed at a rate which has seen capacities multiplied by four in approximately two years, the latest stage of development being the 64K RAM.

Today's modern RAM devices take the user into consideration, and 64K dynamic RAMs which operate off a single 5V power supply are common.

We will describe here the new technology which made possible the development of a highly integrated, high-performance 64K RAM (Type M5K4164S) which operates from a single 5V supply.

Fig. 3.1 shows the cross-section of the cell structure with Table 3.1 summarizing a comparison of the basic parameters of the device with the 16K RAM.

Cell Structure and Process Technology

The M5K4164S 64K RAM makes use of the same two-level n-channel polysilicon gate process and one-transistor cell structure used in the triple power supply 16K RAM (M5K4116 P/S) which has been used in large quantities.

To achieve a high-density RAM, the masks are manufactured using electron beam technology.

In addition, the geometries on several critical levels of the M5K4164S are 2.5 to 3.0μm, necessitating the use of positive photo-resist (for resolution and delineation control) as well as dry-plasma processing at these critical levels.

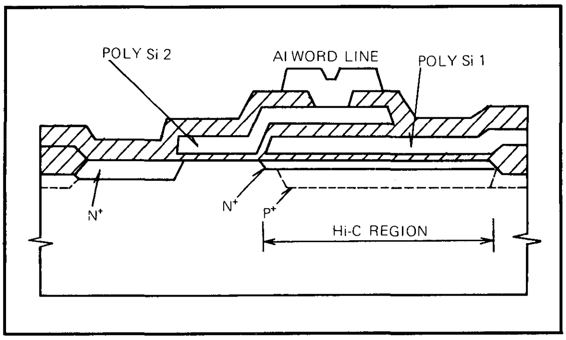


Fig. 3.1 Hi-C structure memory cell cross-section

Table 3.1 Main Parameters

Parameter	16K RAM	64K RAM
Memory cell area	350μm²	200μm²
Chip area	16.3mm²	31.3mm²
Effective channel length	4 μm	2 μm
Gate oxide thickness	850Å	430Å
Diffused layer depth	1.0μm	0.5μm
Diffused layer with	4.0μm	3.0μm
Aluminum width	4.0μm	3.0 μm

Substrate Bias Circuit

In order to facilitate the operation from a single 5V supply, the M5K4164S makes use of an on-chip substarate bias circuit. This bias circuit consists of a ring oscillator, driver circuit, charge pump circuit, and decoupling capacitors. The circuit supplies a bias to the substrate of approximately -3.5V for V_{CC}= 5V (Refer to Fig. 3.2).

The substrate bias circuit has the following functions.

1. It prevents destruction of storage data and disturbance of bipolar transistor operation caused by input undershoot which causes an injection of electrons from the input terminals to the substrate.
2. A reduction in the capacitance of the pn junction formed by the substrate and internal circuit nodes enables an increase in circuit operation speed.
3. The transistor threshold voltage (V_{TH}) modulation due to a bias substrate is reduced, resulting in increased circuit operating speed and stability.

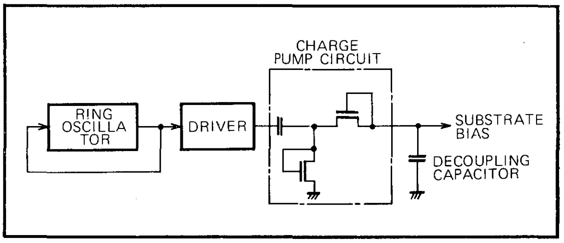


Fig. 3.2 Substrate bias circuit

As shown in Fig. 3.3, the substrate bias for high values of V_{CC} is lower than for the standby mode due to the effect of increased impact ionization current. Adequate margin, however, is maintained against a value of V_{IL} min of −2V.

Reduced Power Consumption and Noise

For operation from a 5V supply, it is necessary to reduce the transistor threshold voltage, V_{TH}. This however invites error operation due to noise. For this reason, circuits required to operate from low voltages only make use of transistors with a low V_{TH}, while those requiring noise immunity are implemented with transistors having a high value of V_{TH}. This scheme insures stable operation.

To lower the peak circuit current, a significant problem in memory system design, and provide for high speed operation, the ratioless driver circuit shown in Fig. 3.4 was used.

With this circuit, the current flowing in transistors Q_1 and Q_2 for changes in the output waveform is practically zero.

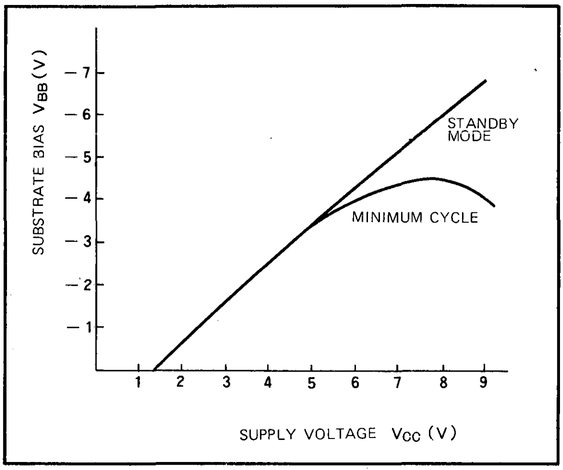


Fig. 3.3 Substrate bias vs supply voltage

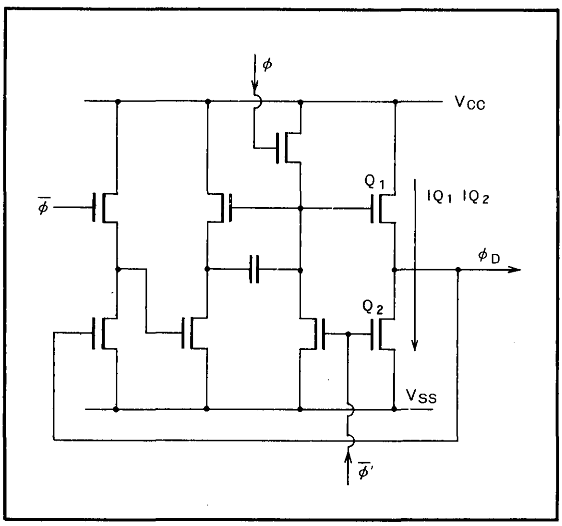


Fig. 3.4 Driver circuit (1) I_{Q1Q2}

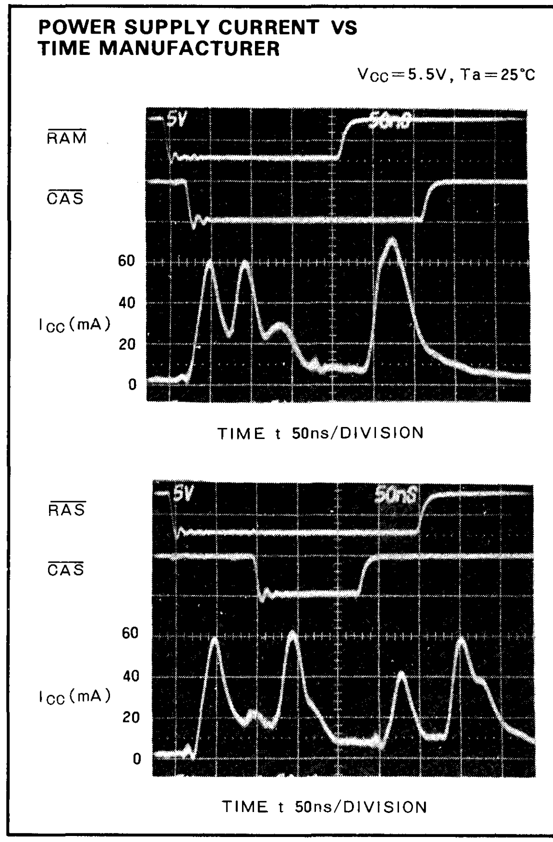


Fig. 3.5 Peak current waveforms

Soft Error Reduction

Reduced pattern sizes and lower supply voltages for 64K RAM devices which result in smaller storage charges result in a higher susceptibility to alpha particles caused soft errors.

These soft errors are caused by alpha particles from minute amounts of uranium and thorium which are present in the IC package and decay. These particles cause the formation of electron-hole pairs in the substrate which collect on the surface and can destroy data.

All floating nodes of dynamic circuits are susceptible to such radiation caused errors and for RAM operation, errors can occur when such phenomena occur in the memory cells and bit lines (including the sense amplifier).

To prevent such soft errors, three approaches are possible.

1. Increase the stored charge in the memory cells.
2. Increase the sense amplifier sensitivity and the bit line signal level.
3. Prevent alpha particles from reaching the chip circuits.

As described below the M5K4164S makes use of these techniques to reduce the effects of alpha radiation.

Bootstrapped Word Line Voltage

Designs of 64K dynamic RAM devices which must operate on 5V supplies must strive to write data into memory with the voltage V_{CC} as well as increase the charge stored in the memory cells in order to reduce the effects of soft errors. This in effect means raising the word line voltage to above the value of $V_{CC} + V_{TH}$ for write and read operations.

Previously this increase in voltage was accomplished by means of the coupling capacitance between the word line and the delay circuit. However, the increased capacitance resulted in a slow risetime of the word line voltage to V_{CC}, as well as increased power consumption. To eliminate these problems a circuit design such as that shown in Fig. 3.6 is used. The transistor Q_2 is kept off until the word line voltage reaches V_{CC}. This has the word line charge capacity. C_2 is then charged by means of transistor Q_3 after which Q_2 is turned on to connect the word line and C_2. The use of this circuit enables increase of the word line voltage without sacrificing operating speed and power consumption, thereby cutting soft error rates by 90%.

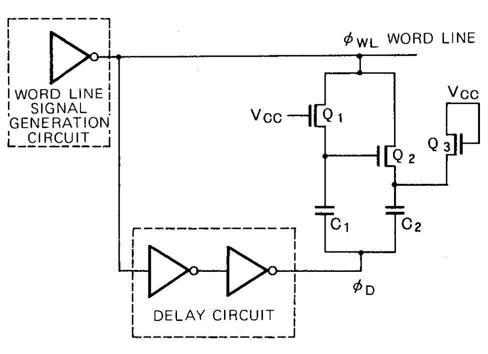


Fig. 3.6 Bootstrapped word line voltage generation circuit

High Capacity (Hi-C) Memory Cell

The increase of memory cell stored charge requires an increase in the memory cell capacitance C_S. Limited chip area, however, places restrictions on the size of the memory cell itself. For this reason the Hi-C structure shown in Fig. 3.1 was used. This cell structure makes use of the normal silicon oxide layer and the p^+ and n^+ junction capacitance. The process for Hi-C memory cell structure requires two additional ion implantation steps and involved the risk of deterioration of the refresh time, an important characteristic of a dynamic RAM device. By selecting the ion implantation level properly, the junction capacitance can be increased without deterioration in the refresh time characteristic. For Hi-C structured cells, a portion of the minority carriers formed in the p^+ layer are recombined, resulting in an effective reduction in soft errors. Such ion implantation has achieved a 30% increase in the memory cell capacitance and a reduction in soft error rate to 1/12 of the error rate of a normally structured cell, as shown in Fig. 3.7.

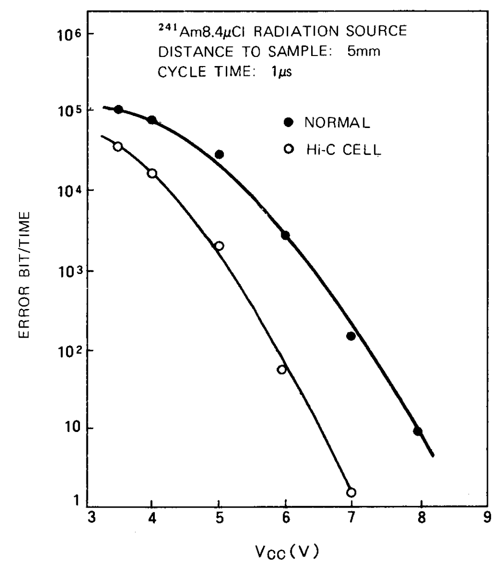


Fig. 3.7 Soft error rate dependency on supply voltage (V_{CC})

Sense Amplifier Circuit

Increasing the sensitivity of the sense amplifier circuit is another effective method of reducing soft errors. Fig. 3.8 shows part of the sense amplifier circuit used by Mitsubishi Electric. High sensitivity with respect to the control signals ϕ_1, ϕ_2, and ϕ_3 plays an important role in this amplifiers operation. After the data read from the memory cell is passed to the sense amplifier, the ϕ_3 signal is controlled to separate the bit line and cut off the noise that is present on the bit line when sensing begins. Smooth sensing begins with the signal ϕ_1 applied so that the minute potential difference is amplified. Next, ϕ_2 is applied and amplified at high speed. By careful adjustment of the timing of the three control signals ϕ_1, ϕ_2, and ϕ_3, detection of potential differences as low as 30mV can be achieved without sacrificing speed in this sense amplifier circuit.

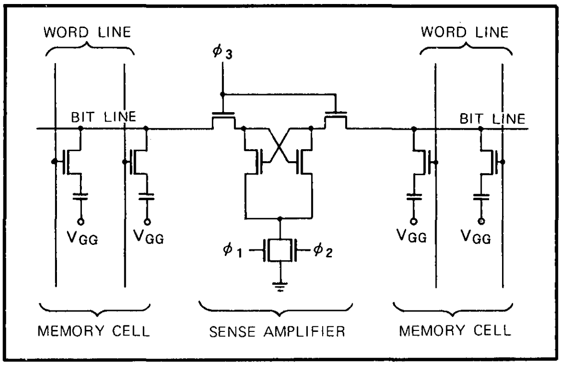


Fig. 3.8 Sense amplifier circuit

128 Refresh Method

When the sense amplifiers sensitivity (offset) and other factors are considered, it is clear that it is important to maximize the read voltage applied from the memory cell to the bit line. The electrical charge, Q, read from the memory cell determines the voltage change ΔV by the following relationship.

$$\Delta V \approx Q/C_B \ (\text{for } C_B \gg C_S)$$

where C_B is the bit line and C_S is the memory cell capacitance.

From this relationship it is seen that to make ΔV large C_B must be made small. To satisfy this condition the 128 refresh method is used to implement a single bit line with 64 memory cells, a technique which reduces the length of the bit line. Fig. 3.9 shows the chip layout. The memory cells are broken into 64x256 bit units which are narrow, long blocks. The column decoders are located in three blocks totalling 256 decoders at the end of the bit line.

Using this arrangement, the bit line capacitance can be minimized.

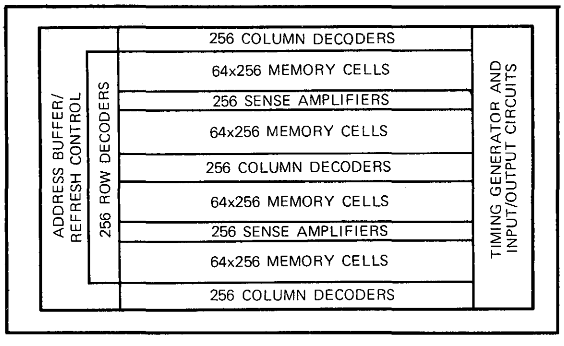


Fig. 3.9 M5K4164S Chip arrangement

Chip Coating

In addition to circuit and device structure improvements aimed at reducing soft errors, the design goal of 10^{-6}/(device hours) required further improvements. The effective range of travel of 5MeV alpha particles in organic resin is a quite short 30~50μm, enabling almost all alpha particles to be shut out by coating the silicon chip surface with such a resin to a thickness of about 40μm.

When this is done, however, alpha particles emitted from the coating material itself cause errors, making material selection critical. The polyimide resin chosen exhibits an alpha radiation level of 0.005α/(cm²•hour), below the measurement sensitivity of an ion chamber. This is low enough that the resulting alpha particle generation level is 1/10 or less that of the package material itself.

Before ceiling the package, this material is coated to a depth of 40μm resulting in at least an expected 90% reduction in alpha particles over non-coated chips.

System evaluations of the M5K4164S treated in such a manner indicate that 10^{-7}/(device hours) for soft error has been achieved.

3.2 Functional Description

Introduction

The M5K4164S is a 64K-bit dynamic RAM which operates off a single 5V supply and has a refresh function built in by means of pin 1. It can be used in a wide range of applications from large mainframes to microcomputers.

This section presents a functional description of the M5K4164S and examines how it can be used in design of a memory system.

Block Diagram

Fig. 3.10 shows the block diagram of the M5K4164S. To preserve the refresh cycle used for 16k dynamic RAM devices, two 32k (two 32k [128 rows (refresh address) x 256 columns] blocks were arranged one on top of the other.

In the center of each block is located 256 sense amplifiers making a total of 512 amplifiers in all.

On one end of each of these two array blocks, is located one row of row decoder.

To prevent crosstalk between the column address lines and bit lines, the column decoder are located at the ends of the bit lines on the opposite side from the sense amplifiers. A total of three rows of column decoders are used.

The central column decoder is used commonly by the two blocks.

Memory Cell

As show in Fig. 3.11, the memory cell consists of one transistor and one capacitor. Data is stored as a one or zero depending upon the amount of electrical charge stored in the capacitor through the transistor Q.

Because leakage current would result in the stored charge of the cell being reduced with time, the data must be refreshed within 2ms.

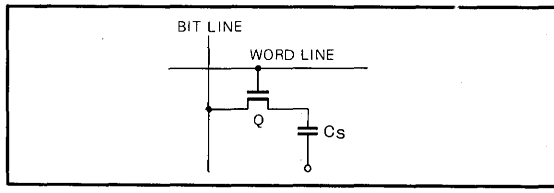


Fig. 3.11 Memory cell

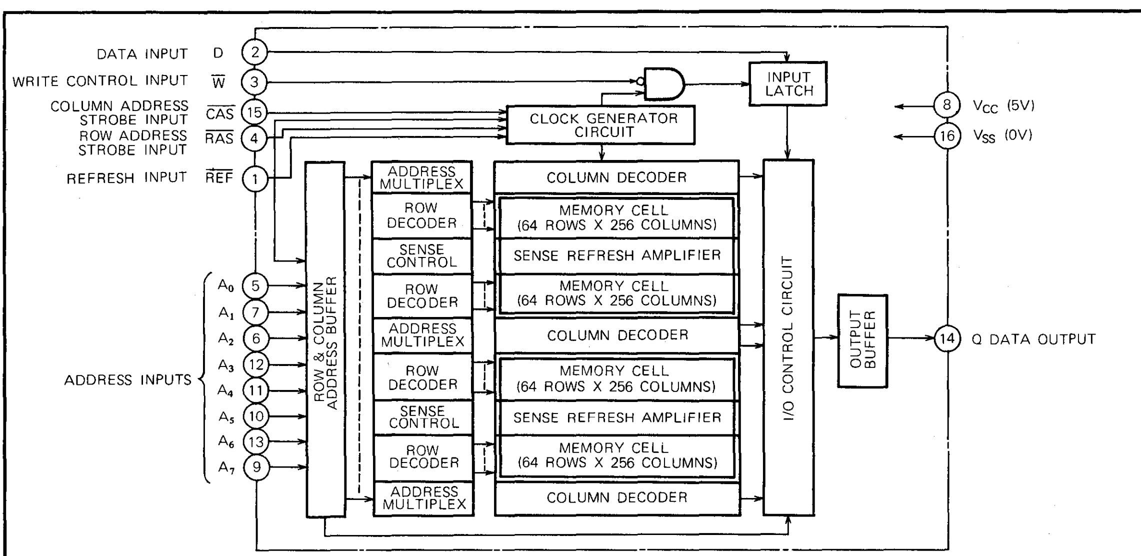


Fig. 3.10 Block diagram

Clock Timing

The M5K4164S has four clock inputs; $\overline{\text{RAS}}$, $\overline{\text{CAS}}$, $\overline{\text{W}}$, $\overline{\text{REF}}$. Among these, $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ are the basic clock inputs for the memory operation. The $\overline{\text{RAS}}$ input is generally used for memory cell data amplification and refresh operation while the $\overline{\text{CAS}}$ is used for data read and write operations only.

To enable the design of a memory system with a large timing margin, it is necessary to know the timing relationships between these two clock inputs and the internal clock signals generated by these clocks.

Fig. 3.13 shows the timing parameters of the $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ clocks while Fig. 3.14 and 3.15 show their relationships to the internal clock timing.

For read or write operations, $\overline{\text{RAS}}$ goes low after which the falling edge of $\overline{\text{CAS}}$ initiates the cycle.

After the read or write is completed, both signals return to a high level and the precharging operation is performed for the next cycle.

For this timing relationship to work, the external $\overline{\text{RAS}}$ clock must follow the changes of the internally generated $\overline{\text{RAS}}$ clock. To simplify the setting of the timing relationships of the external $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ clocks, the internal $\overline{\text{CAS}}$ clock is controlled by the external $\overline{\text{RAS}}$ clock.

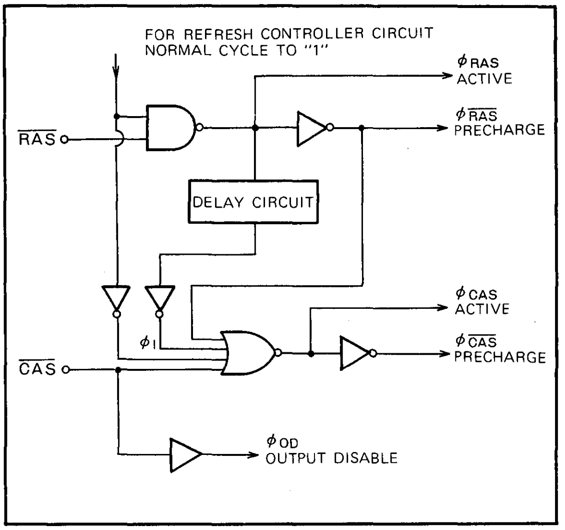


Fig. 3.12 Internal clock generator of $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$

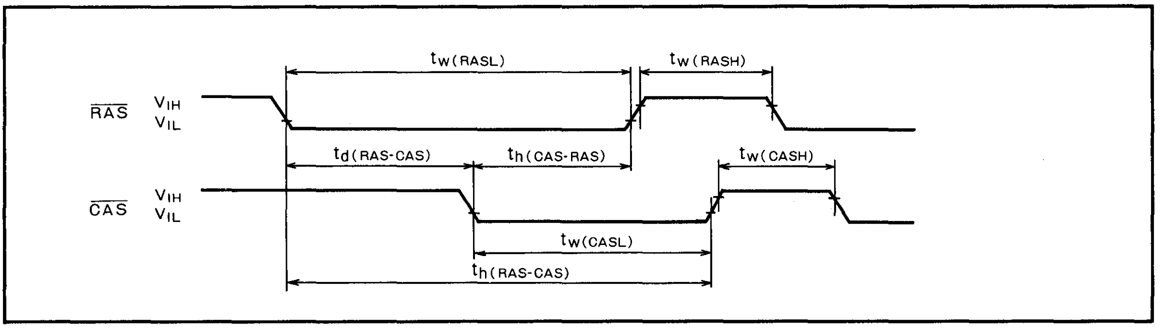


Fig. 3.13 $\overline{\text{RAS}}$, $\overline{\text{CAS}}$ timing

(1) $\overline{CAS}$ Falling Edge Timing (Fig. 3.14)

The memory system design must be such that the falling edge timing of $\overline{CAS}$ does not critically affect the access time. In other words as shown by the solid line in Fig. 3.14, the internal ϕ_{CAS} phase is prevented by the delay phase ϕ_I from approaching $t_{d(RAS\text{-}CAS)}$ max. This type of operation is referred to as gated $\overline{CAS}$.

This gated $\overline{CAS}$ feature permits $\overline{CAS}$ to be activated at anytime between the minimum and maximam value of $t_{d(RAS\text{-}CAS)}$ without affecting access time $[t_{a(RAS)}]$.

For gated $\overline{CAS}$ operation, if the generation of internal clock phase ϕ_{CAS} is delayed, the effective pulse width of ϕ_{CAS} is reduced. For this reason, the rising edge of $\overline{CAS}$ is specified by $t_{h(RAS\text{-}CAS)}$ which is reference to $\overline{RAS}$ rather that $t_{w(CASL)}$. This applies to the column address, $\overline{W}$ and D inputs hold time as well.

As shown by the dotted line in Fig. 3.14, if $\overline{CAS}$ falls to a low level after $t_{d(RAS\text{-}CAS)}$ max, the ϕ_{CAS} phase is generated upon the falling edge of $\overline{CAS}$.

The minimum and maximum values of $t_{d(RAS\text{-}CAS)}$, the delay time $\overline{RAS}$ to $\overline{CAS}$, are specified for the M5K4164S. Operation within the $t_{d(RAS\text{-}CAS)}$ max limit ensures that the access time for the device is guaranteed. This value may be exceeded without causing data storage or reading errors but the access time will be increased.

(2) $\overline{CAS}$ Rising Edge Timing (Fig. 3.15)

As shown in Fig. 3.15, the internally generated $\overline{CAS}$ circuit precharge signal $\phi_{\overline{CAS}}$ is generated with a timing that is related to the relationship between $\overline{RAS}$ and the $\overline{CAS}$ rising edge.

For a $\overline{CAS}$ rising edge occurring before the $\overline{RAS}$ rising edge, $\phi_{\overline{CAS}}$ is generated with the $\overline{CAS}$ rising edge as a reference point (as shown in Fig. 3.15 as a solid line). If however the $\overline{CAS}$ rising edge occurs after that of $\overline{RAS}$, $\phi_{\overline{CAS}}$ is generated with the $\overline{RAS}$ rising edge as a reference (shown as dotted line in Fig. 3.15).

However, the data in the output buffer is cleared upon the occurrence of the rising edge of $\overline{CAS}$ regardless of the state of $\overline{RAS}$. The required pulse width for clear is $t_{w(CASH)}$.

In this manner, the output data can be maintained for a long period while the internal precharge width is made large.

As described above, if the $\overline{CAS}$ rising edge occurs after that of $\overline{RAS}$, the internal $\overline{CAS}$ pulse width becomes not $t_{w(CASL)}$ but $t_{h(CAS\text{-}RAS)}$. Consideration should be given to this point in system design.

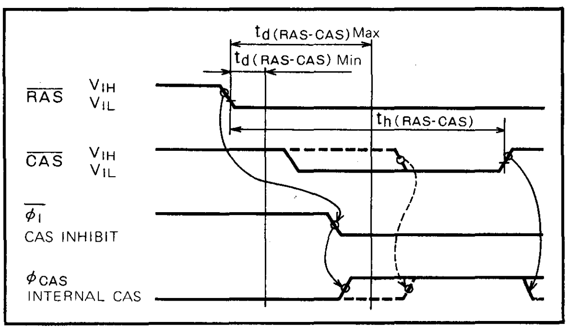


Fig. 3.14 The timing relationship of $\overline{RAS}$ and $\overline{CAS}$ falling edges to internal clock signals (gated $\overline{CAS}$ operation)

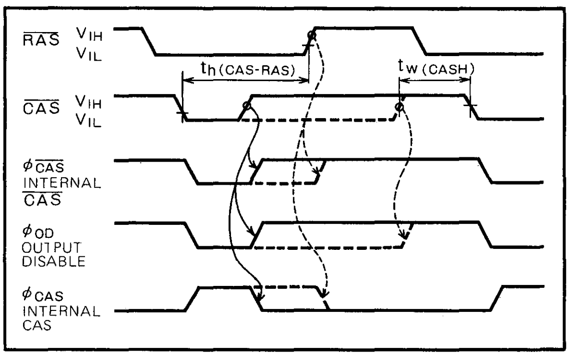


Fig. 3.15 Relationship of $\overline{RAS}$ and $\overline{CAS}$ rising edges to internal clock timing

Address Timing

Addressing of any one of the 65,536 memory cells of the M5K4164S requires the internal latching of two 8-bit multiplexed address (A_0 to A_7) by means of clocks $\overline{RAS}$ and $\overline{CAS}$. First, the row address is latched by the falling edge of $\overline{RAS}$. This selects 512 memory cells from the total of 65,536 memory cells. Fig. 3.16 shows the timing relationships for this operation.

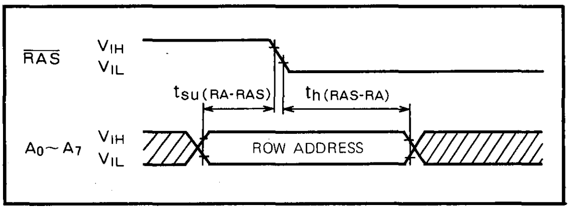


Fig. 3.16 Row address latching timing

The setup time $t_{su(RA-RAS)}$ and hold time $t_{h(RAS-RA)}$ are specified with the $\overline{RAS}$ falling edge as a reference point.

The falling edge of $\overline{CAS}$ latches the column address. This selects one cell from among the 512 cells selected by $\overline{RAS}$. Fig. 3.17 shows the timing relationships for this operation. The setup time $t_{su(CA-CAS)}$ and the hold time $t_{h(CAS-CA)}$ are specified with the falling edge of $\overline{CAS}$ as a reference, while the hold time $t_{h(RAS-CA)}$ is specified with the falling edge of $\overline{RAS}$ as a reference point.

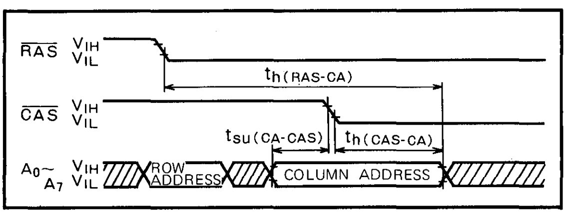


Fig. 3.17 Column address latching timing

For these operations two timing parameters must be considered. One is the column address setup time $t_{su(CA-CAS)}$ which is specified as minus 5 ns, minimum. This means that the column address may be input anytime up to 5 ns after the $\overline{CAS}$ falling edge.

The other parameter is the column address hold time $t_{h(RAS-CA)}$. For the previously described gated $\overline{CAS}$ operation, if $\overline{RAS}$ to $\overline{CAS}$ delay time $t_{d(RAS-CAS)}$ is set between the specified minimum and maximum values, the time from $\overline{RAS}$, $t_{h(RAS-CA)}$ and time from $\overline{CAS}$, $t_{h(CAS-CA)}$ must both be satisfied as the column address hold time. This applies to both the $\overline{W}$ and D signals to be described later.

The time required to switch from row address to column address is referred to as the multiplex time (t_{mux}). This timing is shown in Fig. 3.18.

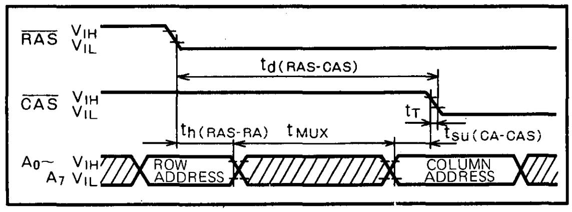


Fig. 3.18 Address multiplex timing

The multiplex time t_{mux} is given by the following expression:

$$t_{mux} = t_{d(RAS-CAS)} - t_T - t_{h(RAS-RA)} - t_{su(CA-CAS)} \quad \cdots(1)$$

As long as the access time, $t_{a(RAS)}$ from $\overline{RAS}$ does not exceed the maximum value, the following expression determines the maximum value of t_{mux} is achieved by the following conditions.

$t_{d(RAS-CAS)}$ = maximum
$t_{a(RAS-RA)}$ = minimum
$t_{su(CA-CAS)}$ = minimum

Table 3.2 shows actual values of t_{mux} maximum for t_T = 5ns.

Table 3.2 Maximum Multiplex Time

Parameter / Device	t_{MUX} max	$t_{d(RAS-CAS)}$	$t_{h(RAS-RA)}$	$t_{su(CA-CAS)}$
M5K4164S -15	55ns	75ns	20ns	− 5ns
M5K4164S -20	75ns	100ns	25ns	− 5ns

If the timing is set to satisfy the above described, operation is guaranteed for both read and write functions. To simplify the following description, the timing parameters for address inputs has been eliminated unless absolutely required.

Read Cycle

Fig. 3.19 shows the timing parameters for the read cycle.

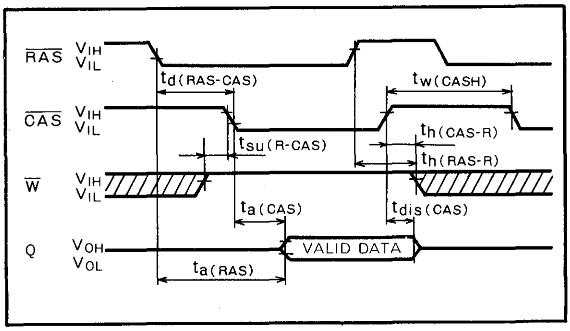


Fig. 3.19 Read cycle timing

In this read cycle, $\overline{RAS}$ and $\overline{CAS}$ are made active, and the $\overline{W}$ input is set to a high level. The setup time, $t_{su(R\text{-}CAS)}$ before $\overline{CAS}$, resulting in output of the data stored in the memory cell at pin Q. The time for the falling edge of $\overline{RAS}$ and $\overline{CAS}$ to the output is defined as the $\overline{RAS}$ access time $t_{a(RAS)}$ and the $\overline{CAS}$ access time $t_{a(CAS)}$ respectively.

The $\overline{RAS}$ access time depends on the $\overline{RAS}$ to $\overline{CAS}$ delay time, $t_{d(RAS\text{-}CAS)}$. The relationship for the $t_{a(RAS)}$ and $t_{d(RAS\text{-}CAS)}$ is shown in Fig. 3.19.

As can be seen from this figure, by setting $t_{d(RAS\text{-}CAS)}$ before t_d for gated $\overline{CAS}$ operation, $t_{a(RAS)}$ does not depend on the value of $t_{d(RAS\text{-}CAS)}$ and is constant.

For $t_{d(RAS\text{-}CAS)}$ set after t_d, $t_{a(RAS)}$ depends upon the value of $t_{d(RAS\text{-}CAS)}$. For this condition, $t_{a(RAS)}$ is given by the following expression.

$$t_{a(RAS)} = t_{d(RAS\text{-}CAS)} + t_{a(CAS)} \quad \text{... (2)}$$

Equation (2) expresses only the electrical characteristics of the RAM device, the guaranteed access time being given by the following expression.

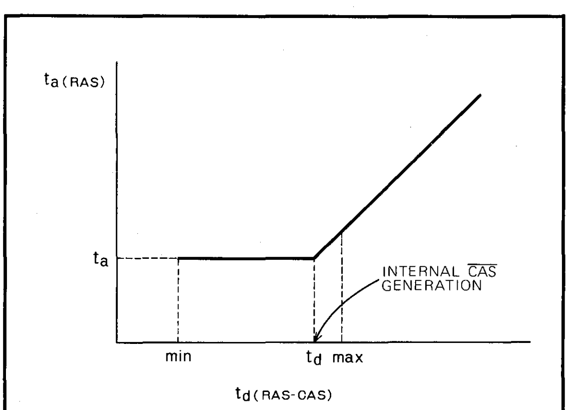


Fig. 3.20 Dependency of $t_{a(RAS)}$ on $t_{d(RAS\text{-}CAS)}$

$$t_{a(RAS)} \leqq t_{d(RAS\text{-}CAS)}\,max + t_{a(CAS)}\,max \quad \text{... (3)}$$

In equation (3), for a value of $t_{d(RAS\text{-}CAS)}$ greater than the maximum value, $t_{a(RAS)}$ increases by the increased amount only.

During a read operation when the output is active, inputs $\overline{RAS}$ and $\overline{W}$ have no effect on the output. Only raising $\overline{CAS}$ to a high level will put the output in the high-impedance state.

The time from the rising edge of $\overline{CAS}$ until the output goes into the high-impedance state is defined as the output disable time ($t_{dis(CAS)}$). This time, $t_{dis(CAS)}$ is the period for the RAM output to go to the open state and should be distinguished from that time the output states to go to V_{OH} and V_{OL}.

The read cycle parameters $t_{h(CAS-R)}$ and $t_{h(RAS\text{-}R)}$ determine the read cycle ending time. Operation is guaranteed if either of these parameters are satisfied.

Write Cycle

Three types of write cycles are specified; early write, read write and read modify write.

(1) Early Write Cycle

Fig. 3.21 Illustrates the timing relationship for this cycle.

This cycle is selected for applications such as I/O common applications in which the output is held at high impedance during the writing of data into the memory cell.

This cycle is executed by causing the $\overline{W}$ input to fall before $\overline{CAS}$.

The $\overline{W}$ and D inputs are latched by $\overline{CAS}$, then the writing of data is executed, the $\overline{W}$ and D input timing parameters $t_{su(W\text{-}CAS)}$, $t_{su(D\text{-}CAS)}$, $t_{h(CAS\text{-}W)}$, and $t_{h(CAS\text{-}D)}$ are determined by the falling edge of $\overline{CAS}$ as a reference point.

Two points here are worthy of consideration. First is the write pulse setup time $t_{su(W\text{-}CAS)}$. This parameter is specified as minus 10ns, minimum.

The significance of this is that $\overline{W}$ input may occur anytime within after 10ns of the falling ege $\overline{CAS}$.

However, should the $\overline{W}$ input falling edge occur after $\overline{CAS}$, the rising edge of $\overline{W}$ is determined not by $t_{h(CAS\text{-}W)}$, but by $t_{W(W)}$.

The other point for consideration is setting $t_{d(RAS\text{-}CAS)}$ between the minimum and maximum values. For this condition, gated $\overline{CAS}$ operation requires that as hold time the time from $\overline{RAS}$ for the $\overline{W}$ and D input, $t_{h(RAS-W)}$ and $t_{s(RAS\text{-}D)}$ and time from $\overline{CAS}$, $t_{h(CAS\text{-}W)}$ and $t_{h(CAS\text{-}D)}$ both must be satisfied.

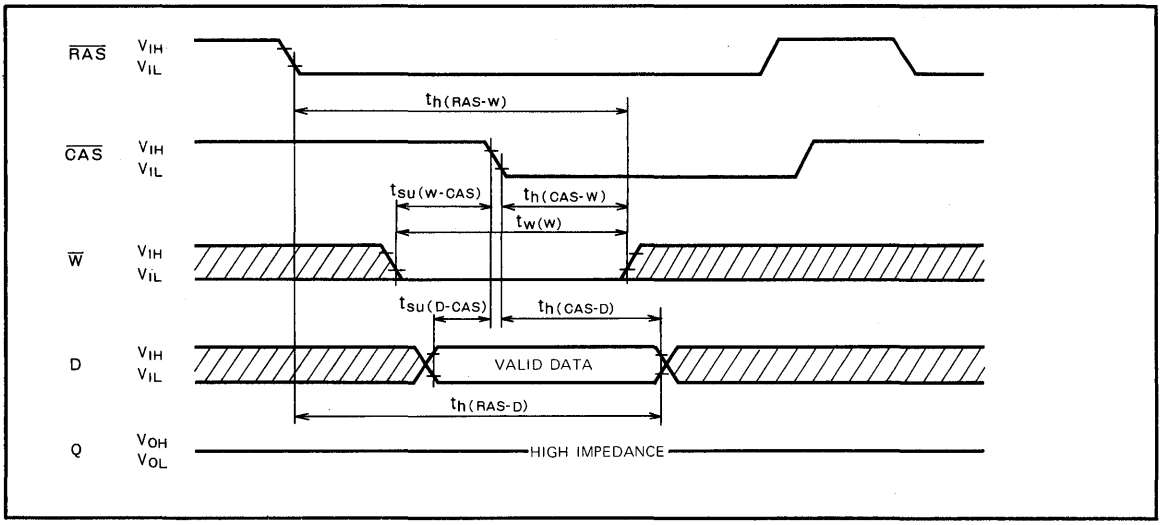


Fig. 3.21 Early write cycle timing

(2) Read Write Cycle Timing

This cycle is used in applications where data is to be read out of memory while new data is being written into a memory cell.

The timing parameters for this read write cycle are shown in Fig. 3.22.

For this type of cycle, the $\overline{W}$ input signal falls after $t_{d(RAS-W)}$ min and $t_{d(CAS-W)}$ min.

The data read timing is the same as the read cycle. Since the read data is latched into an output buffer, $\overline{W}$ input can be made active without disabling the output.

Since the D input is latched by the falling edge of the $\overline{W}$ input, the $\overline{W}$ input falling edge is determined as a reference point for the D input setup time $t_{su(D-W)}$ and hold time $t_{h(W-D)}$.

Date is written into the memory cell between the time the $\overline{W}$ input signal falls and $\overline{RAS}$ and $\overline{CAS}$ rise. This time is specified as $t_{h(W-RAS)}$ and $t_{h(W-CAS)}$ and both of these must be satisfied.

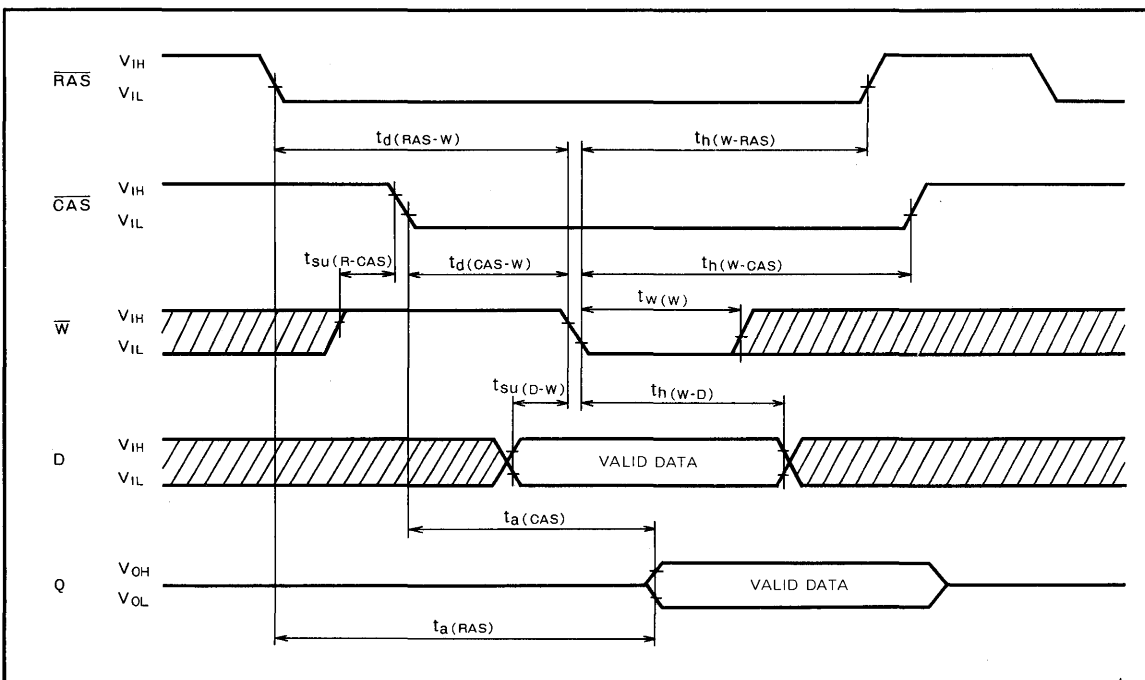


Fig. 3.22 Read write cycle timing

(3) Read Modify Write Cycle

This cycle is used in applications such as ECC (see section on ECC) on which memory cell data is read and verified for correctness, the correct data being written into the cell if an error is detected. Fig. 3.23 shows the timing parameters of the read modify write cycle.

The RAM operation is the same as the previously described read write cycle except that after the data is read, data is written so the cycle is slightly extended.

The minimum time for the read modify write cycle is given by the following expression.

$$t_{CRMW}\ \text{min} = t_{a(RAS)}\ \text{max} + t_{MOD} + t_{h(W\text{-}RAS)}\ \text{min} + t_{w(RASH)}\ \text{min} + 3t_T \quad \text{... (4)}$$

In equation (4), t_{MOD} is the time required for incorrect data to be rewritten correctly, and is a function of system design. In the device specifications $t_{CRMW\ min}$ is specified for $t_{MOD} = 0$.

As previously described, the M5K4164S write cycle mode is determined by the $\overline{W}$ input falling edge timing. This falling edge timing does not limit the operation of the RAM but merely controls the output state. If the $\overline{W}$ input falling edge does not satisfy the conditions described for the three write modes, data will be written but the output state will be indeterminate.

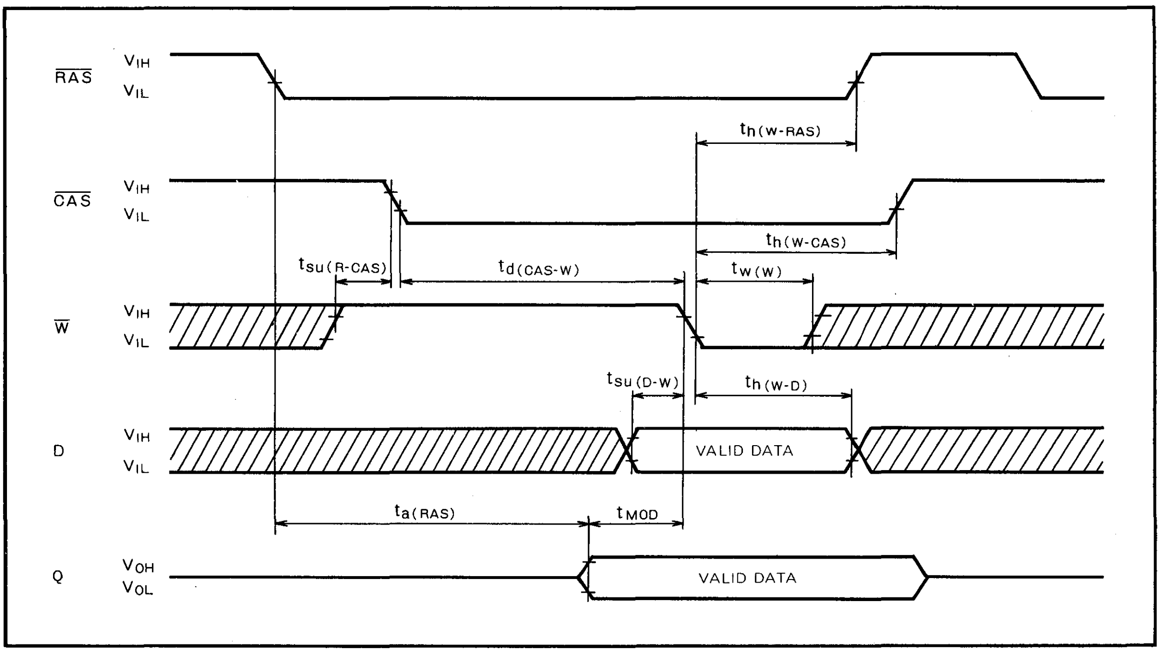


Fig. 3.23 Read modify write cycle timing

Page Mode Timing

Page mode operation is successive memory operations at multiple column locations within the same row address.

As with normal operation, page mode operation can be carried out in the read, early write, read write or read modify write modes. The timing parameters particular to the page mode of operation are shown in Fig. 3.23. The other parameters are the same as for normal cycles.

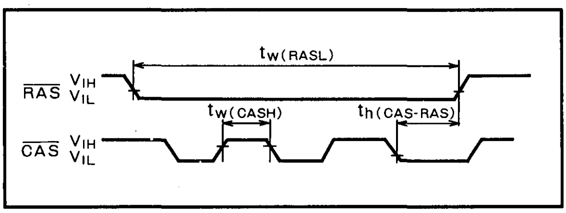


Fig. 3.24 Page mode cycle timing

To perform page mode read and write operations, the $\overline{RAS}$ low-level pulse width, $t_{w(RASL)}$ must be increased, the maximum value being 10μs. The high-level $\overline{CAS}$ pulse width, $t_{w(CASH)}$ is specified separately for the normal mode cycle and page mode. For the page mode, the pulse width must be increased. For details refer to the specifications.

For page mode operation the hold time $t_{h(CAS-RAS)}$ must be satisfied for even the last cycle, as shown in Fig. 3.24. This applies to $\overline{W}$ as well.

Refresh

Referring to the block diagram of Fig. 3.10, for each $\overline{RAS}$ cycle, one word line is selected for each of the upper and lower blocks, enabling access to 512 memory cells. Next, the 512 sense amplifiers operate to amplify and refresh the cell data. Address signal A_7 (Row) has no connection with this refresh operation since it is used as a block select address for data read and write operations.

$\overline{RAS}$ Only Refresh Timing

$\overline{RAS}$ only refresh is performed by setting $\overline{CAS}$ to high which sets the output to high-impedance while refresh is performed.

Both distributed and burst mode refresh can be performed.

Fig. 3.25 shows the timing parameters for $\overline{RAS}$ only refresh operation.

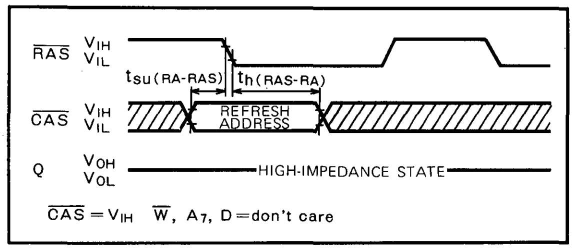


Fig. 3.25 $\overline{RAS}$ only refresh timing

Hidden Refresh Timing

Hidden refresh is accomplished by setting $\overline{CAS}$ to low after a read cycle to hold the data in the valid state while refresh is performed.

Both distributed and burst mode refresh are possible. Fig. 3.26 shows the timing parameters for hidden refresh operations.

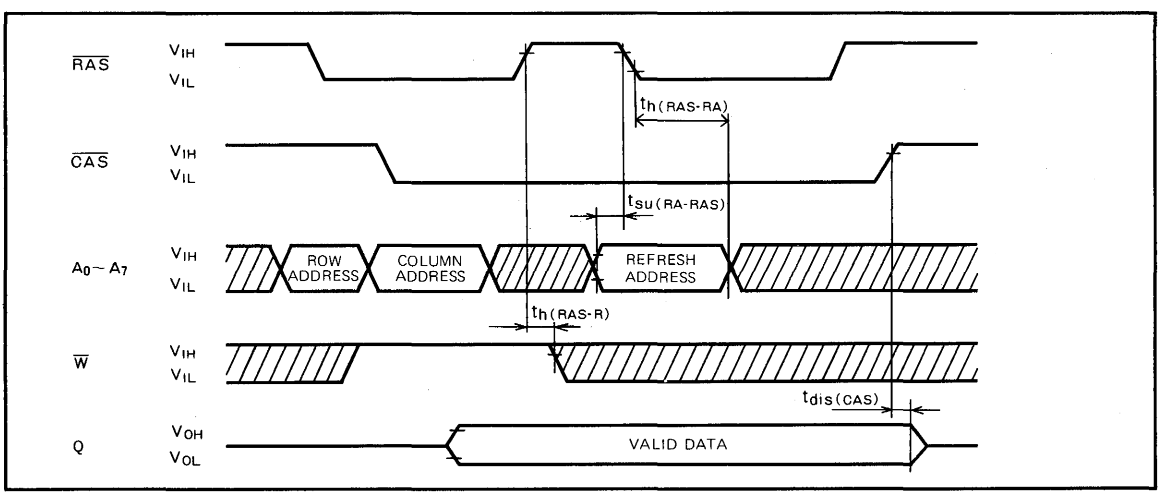


Fig. 3.26 Hidden refresh timing

Data latched in the output buffer by the read cycle is refreshed during the hidden refresh cycles by $\overline{RAS}$. Therefore output data is held indefinitely as long as hidden refreshing is continued.

Timing design is simplified because the $\overline{W}$ may be changed in any state during hidden refreshing.

Refresh Operations Using Pin 1

To simplify the refresh operation, a function absolutely essential to dynamic RAM operation, two special refresh functions easier to use than the conventional $\overline{\text{RAS}}$ clock refresh have been provided.

These functions are automatic refresh and self refresh.

These special functions are implemented by an on-chip refresh counter, address multiplexer, and timer, along with the associated control circuitry. The following is an operational description of these circuits.

(1) Refresh Control Circuit

Fig. 3.27, is a block diagram of the refresh circuit which makes use of Pin 1. The control circuit controls not only the refresh counter, address multiplexer and timer as shown in Fig. 3.26, but $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ circuits as well.

Pin 1 refreshing is controlled externally by the $\overline{\text{REF}}$ input and internally by the $\overline{\text{RAS}}$ signal which is generated by the refresh control circuit.

During pin 1 refresh operations, the $\overline{\text{CAS}}$ circuit with the exclusion of the output buffer is inhibited to prevent data writing and reading.

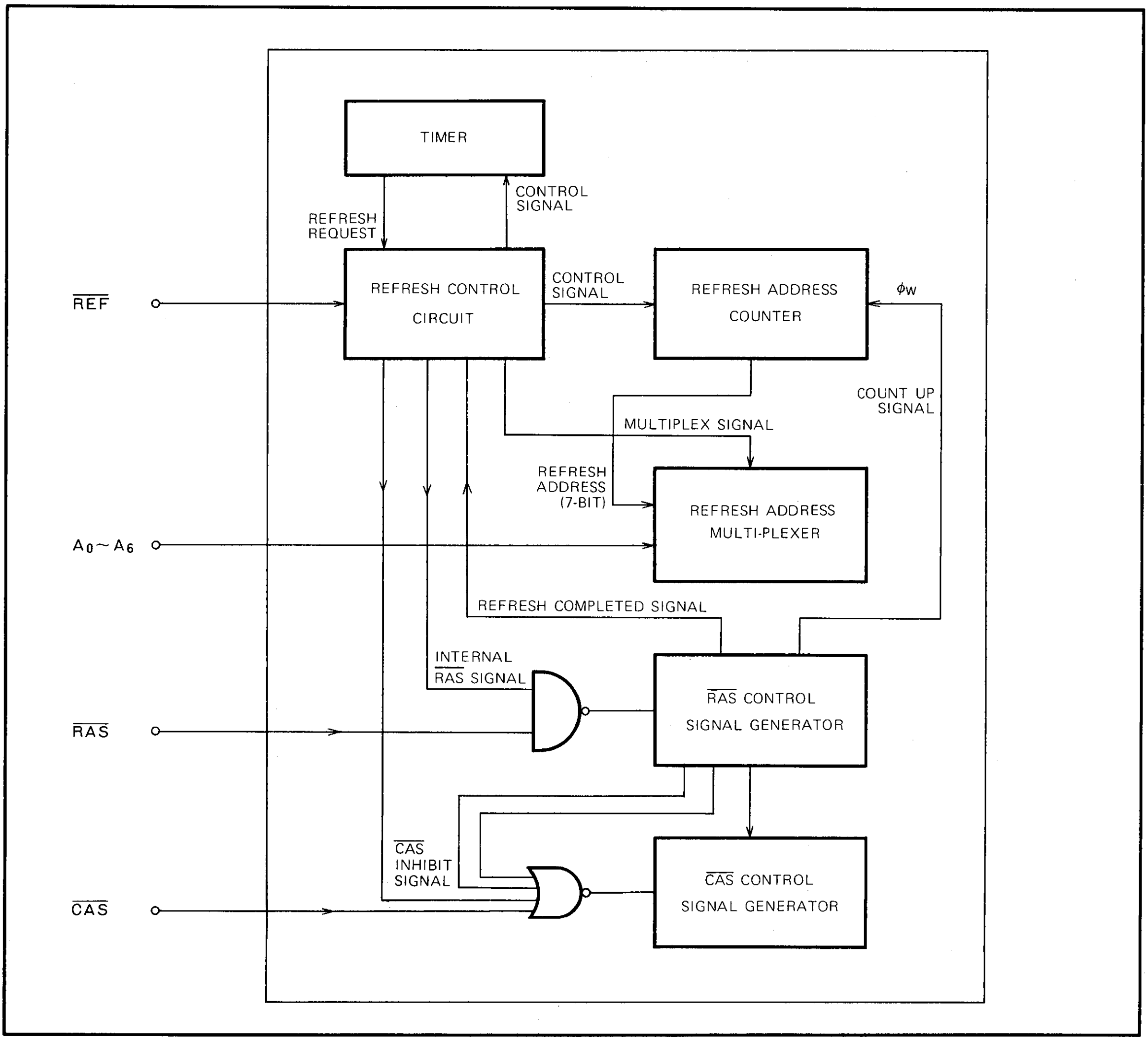


Fig. 3.27 Refresh circuit block diagram

(2) Refresh Counter Circuit

The M5K4164S on-chip refresh counter consists of a 7-bit toggle-type counter, the individual counter output being used as the refresh address bit.

For automatic refresh operations, the refresh counter counts up synchronized to the internal clock signal ϕ_W which is synchronized in turn to the $\overline{REF}$ input, 128 $\overline{REF}$ pulses required to cycle to the original state. For self refresh operation, the refresh counter is free-running with a period of from 12 to 16μs, counting up in synchronous to the refresh request signal REFREQ (described afterwards). A complete cycle and return to the original state requiring that the $\overline{REF}$ input be held low for 16μs x 128 = 2 ms.

The above described counting operation is performed approximately in synchronous with the refresh operation completion. The output of the refresh counter, Q_0 to Q_6 (refresh address) is held until the next refresh cycle, forming the address for the next cycle.

The refresh counter outputs are initialized by approximately 8 dummy cycles of $\overline{RAS}$, $\overline{RAS}/\overline{CAS}$, or $\overline{REF}$. Therefore, no special initialization is required for this refresh counter.

However, the contents of the counter, that is the toggle counter flip-flop states, cannot be reset or preset externally during power up or dummy cycles.

For this reason, using both normal $\overline{RAS}$ and pin 1 refresh will cause the specified refresh time to be exceeded, and therefore should be avoided.

(3) Address Multiplexer

Fig. 3.28 shows the M5K4164S onchip address multiplexer.

The address multiplexer consists of two MOS transistors at the address buffer inputs and the associated control signals (MUX, $\overline{MUX}$).

During a normal cycle, $\overline{MUX}$ is high and MUX is low, so that only the external address A_0 to A_6 is input.

For pin 1 refresh operations, $\overline{MUX}$ is low and MUX is high so that the refresh counter output signals Q_0 to Q_6 only are input to the address buffer.

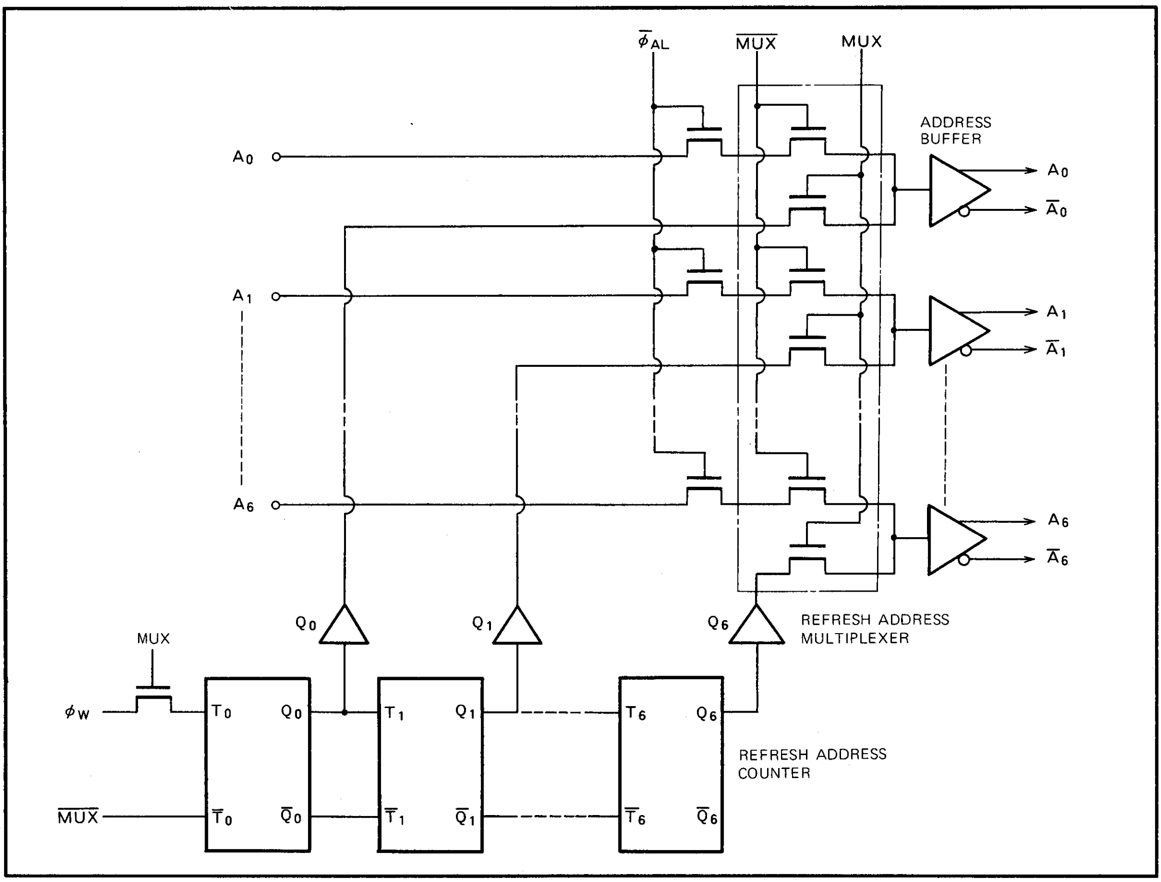


Fig. 3.28 Refresh address counter and multiplexer circuits

(4) Timer Circuit

Fig. 3.29 shows the timer circuit block diagram while Fig. 3.30 illustrates its timing.

In the circuit of Fig. 3.29, the oscillator provides the substrate bias as well. The other circuits are active when $\overline{REF}$ is low.

When $\overline{REF}$ goes low, transistor Q_1 turns on, Q_2 turns off and the charge stored in C_2 passes through the rectifying circuit C_2 and Q_1 to discharge upon the falling edge of the oscillator output signal. The charge for one cycle of the oscillator output is proportional to the ratio of the capacitance of C_1 and C_2.

The ratio of C_1 to C_2 is chosen such that the voltage across C_2 for an oscillator repetition period of 12 to 16μs is approximately equal to the threshold voltage of the next gate. When the C_2 voltage reaches V_{TH}, the refresh request signal REFREQ goes active, causing the RAM refresh operation similar to the autoamtic refresh external signal $\overline{REF}$. When C_2 is charged by REFREQ, REFREQ goes low, causing a repetition of the above described timer operation.

As long as the $\overline{REF}$ signal is kept low, this operation will automatically continue refreshing all memory cells every 2 ms.

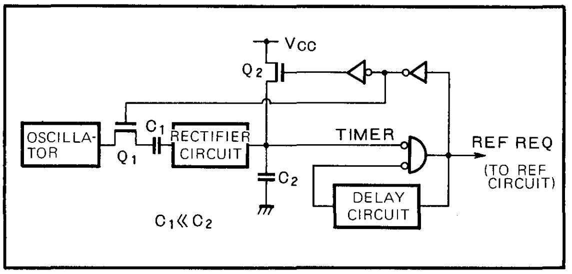


Fig. 3.29 Timer circuit block diagram

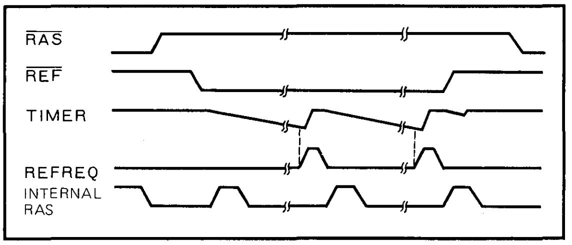


Fig. 3.30 Timer circuit timing

Automatic Refresh Timing

Automatic refresh is accomplished in the same manner as $\overline{RAS}$ only refresh but without providing the refresh address data.

Fig. 3.31 shows the timing of the automatic refresh operation.

Automatic refresh begins when $\overline{REF}$ is set to low $t_{w(RASH)}$ after $\overline{RAS}$ goes high.

Shortly after the refresh cycle begins the internal $\overline{RAS}$ signal begins to operate to strobe the refresh counter output and perform the refresh.

The $\overline{REF}$ input is internally latched. When the refresh operation is complete an internal refresh complete signal causes the chip to be precharged. Therefore, it is not necessary to use the $\overline{REF}$ input to determine the precharge time greatly simplifying the timing design of $\overline{REF}$.

It is also possible to perform hidden refreshing by holding the $\overline{CAS}$ input low after a read cycle. The timing is very similar to the $\overline{RAS}$ hidden refresh operation timing.

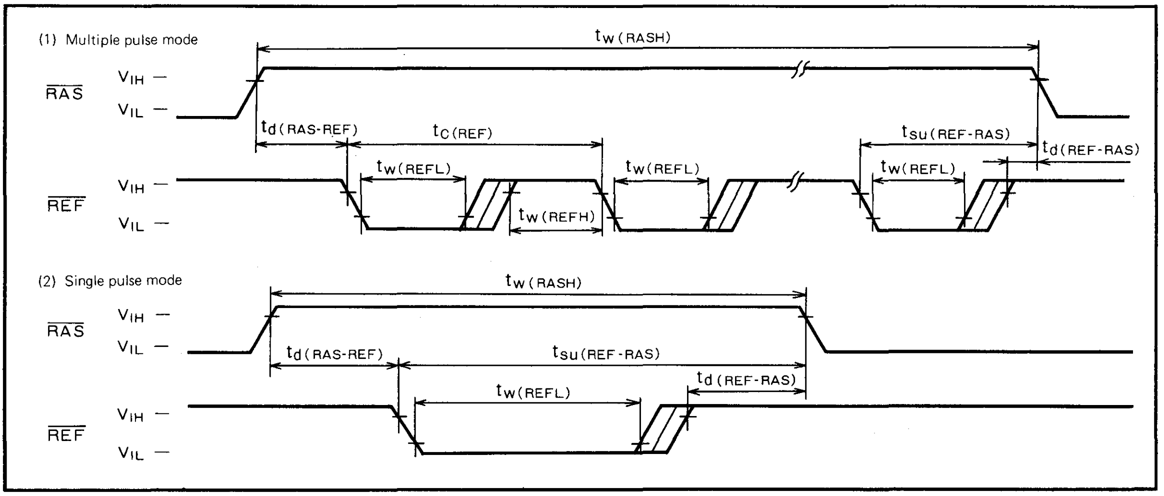


Fig. 3.31 Automatic refresh timing

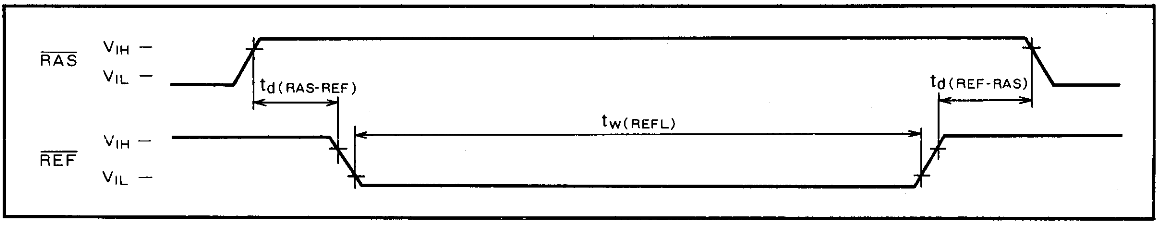


Fig. 3.32 Self refresh timing

For details refer to the specifications.

Self Refresh Timing

Self refresh is generally used for battery backup of memory contents.

Fig. 3.32 shows the self refresh timing relationships from which it can be seen that they are quite similar to those of automatic refresh. Self refresh begins when $\overline{REF}$ is set to low $t_{w(RASH)}$ after $\overline{RAS}$ is set to high.

Shortly after the beginning of the refresh cycle, the internal $\overline{RAS}$ signal begins to operate to strobe the refresh counter and perform the refresh operation.

As long as $\overline{RAS}$ is high and $\overline{REF}$ is low, the RAM will be automatically refreshed. This operation is repeated with a period of from 12 to 16μs. After 2ms, the refresh counter has gone through all of the row address, refreshing all of the memory cells. Self refresh ends when $\overline{REF}$ is set to high but setting $\overline{REF}$ to high may not terminate the internal operation of the circuit (refer to Fig. 3.30) so that one cycle time of $t_{d(REF-RAS)}$ is required between setting $\overline{REF}$ to high and $\overline{RAS}$ to low.

As with automatic refresh, hidden refreshing is possible. For details refer to the specifications.

Notes on the Use of Pin 1

When pin 1 is not to be used to refresh the chip, it should be handled in the following manner.

(1) Since pin 1 refresh is inhibited by setting the $\overline{REF}$ input to high, the input should be set between $V_{IH\ min}$ and $V_{IH\ max}$. (The pin 1 input leakage current for V_{IN} = 6.5V is guaranteed to be below 10μA.)

(2) When the above method is not possible, pin 1 should be left open. Since as shown in Fig. 3.33 as MOS transistor is used to connect a pull-up resistor between the input terminals and V_{cc}, the terminal will be held to a high level (V_{cc}) when left open.
However, when the input is set low in order to perform a refresh operation, a current flows from V_{cc} to the input terminal. This resistance is made a very high value (approximately 3MΩ) in order to guarantee the specified leakage current of 10μA maximum for V_{IN} = 0V.
This high resistance results in pin 1 being susceptible to the effects of external noise so that if pin 1 is to be left open, such noise should be considered carefully.

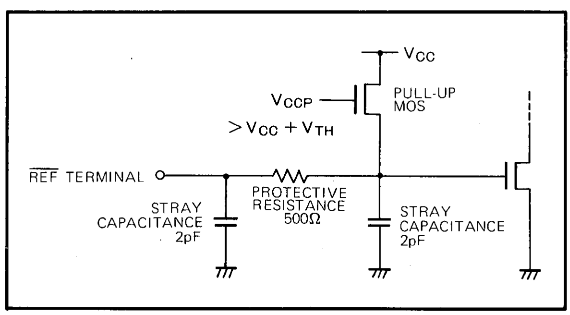


Fig. 3.33 $\overline{REF}$ input equivalent circuit

3.3 M5K4164S Bit Map

Introduction

To facilitate the generation of worst-case pattern checking and the optimization of test sequences, it is necessary to know the internal topology of a memory device. This section will examine the internal topology of the M5K-4164S.

Memory Array

Fig. 3.34 shows the dual in-line package as viewed from above with pin 1 to the upper right. It illustrates the memory cell layout.

The row decoder are located to the left of the memory cells while the colum decoder are located parallel to the cells.

Address Decoder

Fig. 3.35 shows the address decoder. To optimize pattern layout, the decoder is arranged as shown in Fig. 3.35. For this reason, with A_0 (row) as the least significant bit and A_7 (column) as the most significant bit, sequential binary addresses will not address adjacent cells in order.

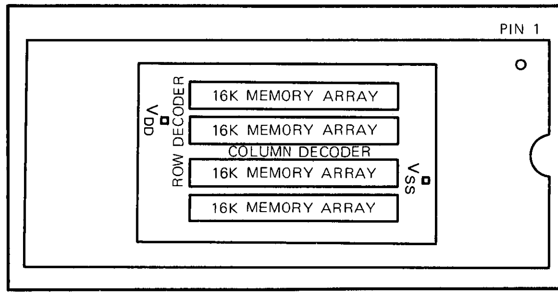


Fig. 3.34 Memory array location

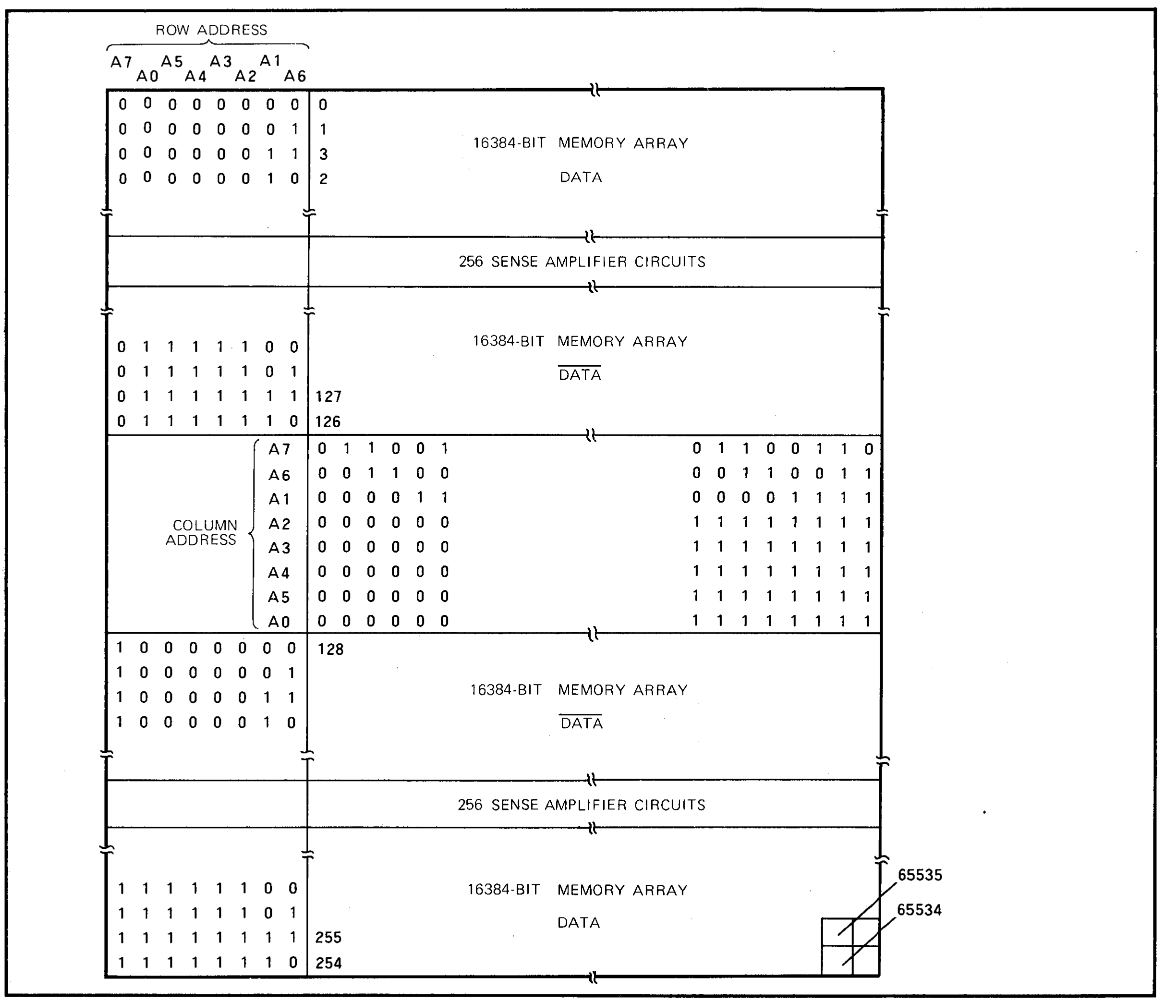


Fig. 3.35 Address decoder location

For the arrangement of Fig. 3.35, Table 3.3 shows the addresses that will be accessed for sequentially incremented binary addresses if A_6 (row) is the least significant bit and A_0 (column) is the most significant bit.

Bit Topology

For the purposes of simplified explanation, we have assumed thus far that the memory cells are located in an orderly fashion in a matrix. For actual devices, however, techniques required to increase the density on the chip dictate that an arrangement such as shown in Fig. 3.36 is used.

For this reason, this layout must be considered carefully when designing tests which detect interference between adjacent cells.

Data Polarity

Because the sense amplifiers are located in the center of the bit lines of the M5K4164S, half of the data matrix is stored in inverted form. While this has absolutely no effect on actual operation, it must be considered if a test is to be devised which will test all cells in the charged state. This bit inversion pattern is given in Table 3.4.

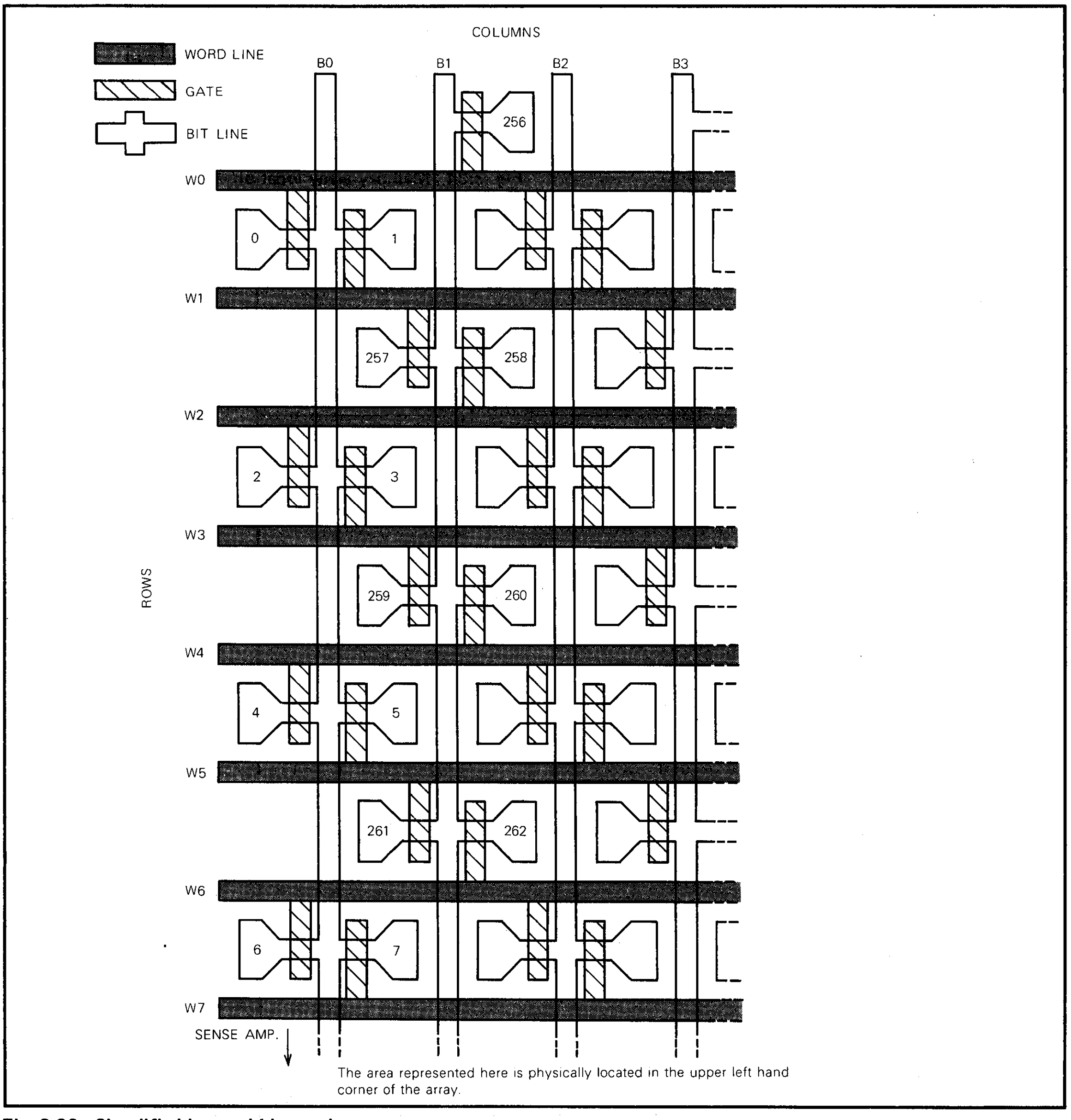


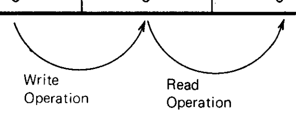
Fig. 3.36 Simplified internal bit topology

Table 3.3 Address Coding

	Column (MSB)								Row							(LSB)
Cell No	A_0	A_5	A_4	A_3	A_2	A_1	A_6	A_7	A_7	A_0	A_5	A_4	A_3	A_2	A_1	A_6
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
⋮	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
⋮					⋮								⋮			
32767	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
⋮					⋮								⋮			
65535	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Table 3.4 Data Polarity Arrangement

A7 (row)	A0 (row)	Input data	Memory cell data	Output data
0	0	1	1	1
		0	0	0
	1	1	0	1
		0	1	0
1	0	1	0	1
		0	1	0
	1	1	1	1
		0	0	0



3.4 Memory System Design Considerations

New memory systems designs are making use of dynamic RAM, static RAM, EPROM and other semiconductor memory devices. All of these devices have some general design considerations in common. This application note will examine some of the delicate timing considerations involved in the design of a dynamic RAM board.

Power Distribution

Fig. 3.37 shows the current waveform of an M5K4164S dynamic RAM. Note that when $\overline{RAS}$ and $\overline{CAS}$ go low, the row or column address latch and buffer are charged, and that when $\overline{RAS}$ and $\overline{CAS}$ go high, the row or column address latch and buffer are precharged, resulting in a transient current waveform. The 60 to 80mA current pulse of approximate width 50ns and a risetime of 5~10ns represents the risetime which is observed at 50ns per division. With rise and fall times of this magnitude harmonic noise components above 10MHz are generated. It is therefore necessary when designing the board power distribution to suppress such noise and provide the device with a clean supply voltage. Decoupling capacitors should be used which are capable of charging a small loop. For a 0.1μF capacitor value used with a 250ns cycle RAM, the spike voltage is given by the following expression.

$$v = \frac{1}{c} \int idt = \frac{80mA}{0.1\mu F} \times 50ns = 40mV$$

This yields an acceptable value of spike voltage.

It is recommended that ceramic capacitors with good high frequency characteristics are used as the decoupling capacitors in memory arrays. The decoupling capacitor is connected between the memory V_{CC} and the ground with as short a lead dressing as possible. In addition, as bulk decoupling a solid tantalum capacitor is required. This type of capacitor has a better transient response than other large value capacitors and can be used with one capacitor per 16-memory devices between V_{CC} and the ground.

The power supply traces for a memory array should be made as wide as possible and it is recommended that they be arranged in a grid. Fig. 3.39 shows an example of such an arrangement.

As another method, the use of multi-layer boards is possible, and is an effective method in simplifying power distribution.

Fig. 3.38 (a) shows the lumped constant equivalent circuit for a PC board. L_S and R_S represent the PC board inductance and resistance respectively. If we let the L_S and R_S of a 10mil wide 2-ounce copper pattern be 10nH/inch and 4mΩ/inch, then the generated spike voltage is given by the following expression.

$$L_S \cdot \frac{di}{dt} = 10nH \times \frac{80mA}{50nS} = 16mV$$

$$R_S I = 4m\Omega \times 80mA = 0.32mV$$

Since the effect of the series resistance R_S compared to that of the series inductance is very small, it may be neglected. The series resistance of L_S is frequency dependent, increasing with increasing frequencies.

To reduce the level of the spike voltage, as shown in Fig. 3.38 (a), a decoupling capacitor is used to decrease the series resistance. This is done by shortening the PC board current loop.

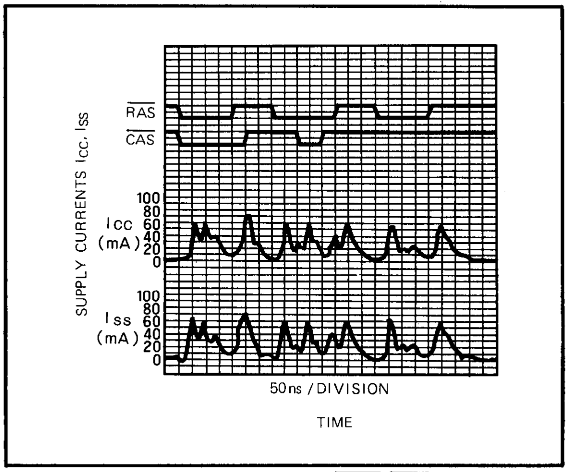


Fig. 3.37 Supply current vs time $\overline{RAS}/\overline{CAS}$ cycle $\overline{RAS}$ only cycle

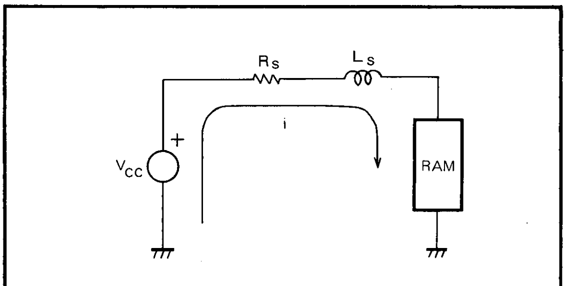


Fig. 3.38 (a) PC board trace equivalent circuit

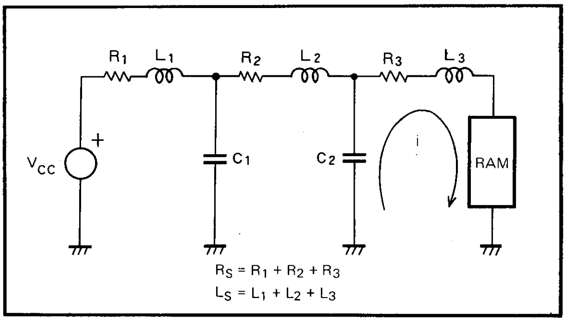


Fig. 3.38(b) PC board trace equivalent circuit with decoupling capacitors

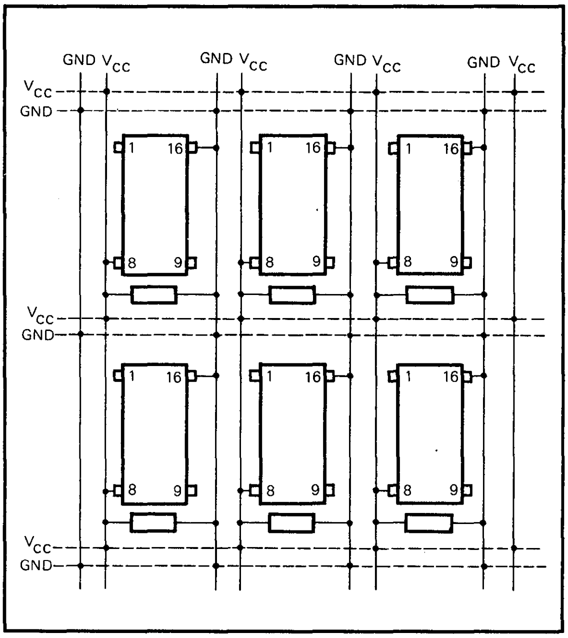


Fig. 3.39 Gridded power distribution and decoupling capacitors

Signal Distribution

The next most important consideration in the design of a memory system is the design of memory signal (address, data, and control signals) distribution.

For the case of the M5K4164S dynamic RAM, two types of chip enable signals exist; $\overline{RAS}$ and $\overline{CAS}$. If these are to be driven by TTL circuits, it is very important to keep the driving TTL device as close as possible to the RAM array. This minimizes the transmission line impedance mismatch between the RAM array loaded line and the TTL driver. Another technique is the use of a damping resistor located close to the driver. The value of this resistor is selected to provide a good waveform at the RAM input, the usual values being in the range 10 to 51Ω. This technique brings the output impedance of the driver close to the line impedance which minimizes waveform overshoot and undershoot.

To eliminate crosstalk from $\overline{RAS}$ and $\overline{CAS}$, the $\overline{RAS}$ and $\overline{CAS}$ signal lines should be kept at 90° to the traces for other signals. If this is impossible, they should be kept as far as possible from traces of other signals. In addition the address and data signal traces should be kept as short as possible.

Logical Considerations

For memory systems with critical timing, it is necessary to consider the propergation delay to surrounding ICs. To minimize signal delay, gate selection and the use of the same IC package for related signals are effective in reducing the difference in delays between signals. To reduce the capacitive loading on drivers, it is necessary to limit the number of drivers per memory array. For $\overline{RAS}$ and $\overline{CAS}$, 8 memory/drivers and for address 16 memories/driver are recommended.

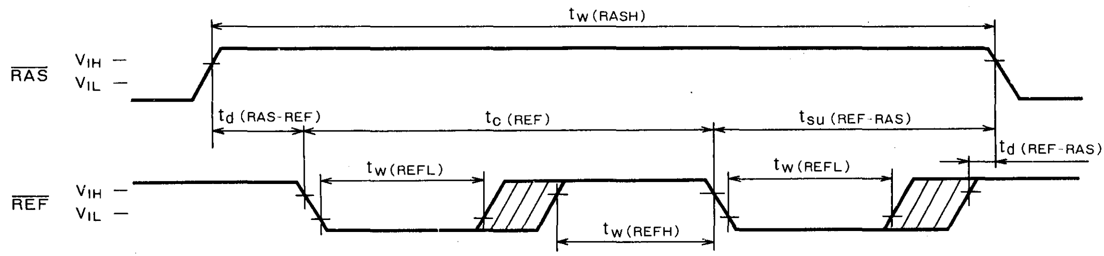
3.5 M5K4164S Refresh Methods

Introduction

The refreshing of the M5K4164S cell matrix requires the refreshing of 128 row addresses at least every 2ms. In addition to the previously available $\overline{RAS}$-only refresh method, the M5K4164S provides $\overline{REF}$ (pin 1) automatic refreshing, and self-refreshing. This section will cover the application of $\overline{REF}$ refresh operations.

Automatic Refresh

Automatic refresh begins after $\overline{RAS}$ precharge ($\overline{RAS} = V_{IH}$) upon setting $\overline{RAF}$ (pin 1) to low. This method is quite similar to the $\overline{RAS}$-only refresh with the refresh address counter output present as a 7-bit word for automatic refreshing, the refresh counter being automatically incremented at the end of the refresh cycle. Fig. 3.40 shows the automatic refresh timing.



Note: $\overline{CAS}$, Address, D and $\overline{W}$ are don't cares.

Symbol	Parameter	Alternative Symbol	M5K4164S-15 Limits Min	M5K4164S-15 Limits Max	M5K4164S-20 Limits Min	M5K4164S-20 Limits Max	Unit
$t_{c(REF)}$	Automatic Refresh Cycle Time	t_{FC}	260		330		ns
$t_{w(RASH)}$	$\overline{RAS}$ high pulse width	t_{RP}	395		480		ns
$t_{d(RAS-REF)}$	Delay time, $\overline{RAS}$ to $\overline{REF}$	t_{RFD}	100		120		ns
$t_{w(REFL)}$	$\overline{REF}$ low pulse width	t_{FP}	80	8000	100	8000	ns
$t_{w(REFH)}$	$\overline{REF}$ high pulse width	t_{FI}	135		150		ns
$t_{d(REF-RAS)}$	Delay time, $\overline{REF}$ to $\overline{RAS}$	t_{FSR}	30		30		ns
$t_{su(REF-RAS)}$	$\overline{REF}$ pulse setup time before $\overline{RAS}$	t_{FRD}	295		360		ns

Fig. 3.40 Automatic Refresh Timing

Automatic refresh has many advantages over the $\overline{RAS}$-only refresh method generally used previously. As shown in Fig. 3.41, $\overline{RAS}$-only refresh generally requires logic circuitry. This consists of the row-address, column-address and refresh address multiplexer and refresh address counter. With automatic refresh, the dotted area shown in Fig. 3.41 may be eliminated.

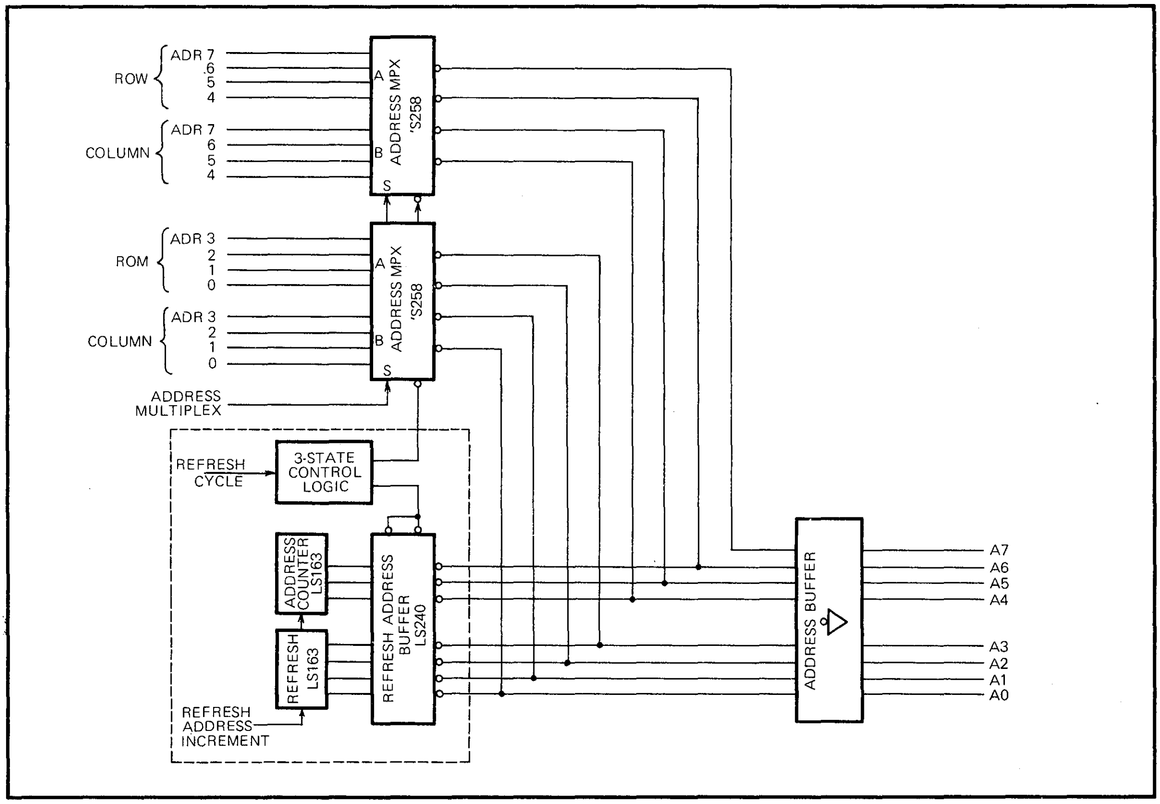


Fig. 3.41 Address multiplexer and refresh address counter

By decoding $\overline{RAS}$, one bank of a complex memory system may be selected, while for $\overline{RAS}$-only refresh $\overline{RAS}$ is fed to all portions of the memory requiring the decoder as shown in Fig. 3.42 (a). With automatic refresh, $\overline{RAS}$ is used during the memory cycle and $\overline{REF}$ for the refresh cycle independently so that the game shown in Fig. 3.42 (b) can be eliminated.

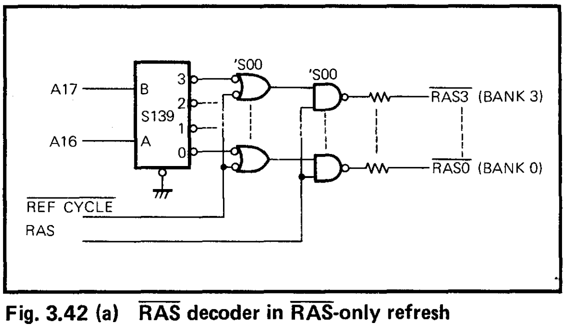


Fig. 3.42 (a) $\overline{RAS}$ decoder in $\overline{RAS}$-only refresh

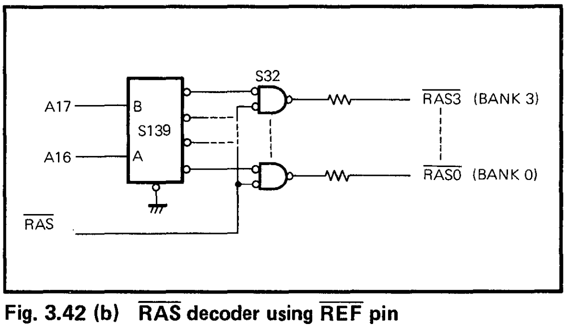


Fig. 3.42 (b) $\overline{RAS}$ decoder using $\overline{REF}$ pin

Another feature of automatic refresh is that the timing of the refresh controller is simplified. The timing for $\overline{RAS}$-only refresh and automatic refresh is shown in Fig. 3.43 (a) and Fig. 3.43 (b) respectively.

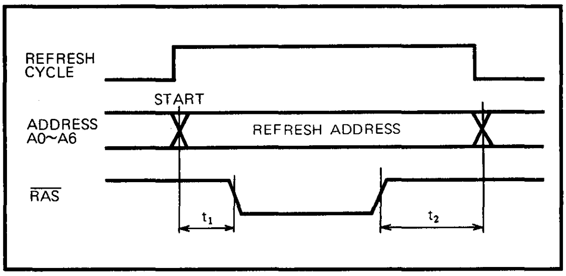


Fig. 3.43 (a) $\overline{RAS}$-only refresh timing

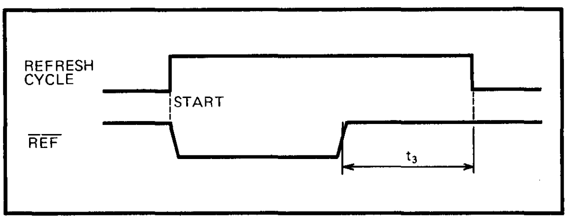


Fig. 3.43 (b) Automatic refresh timing

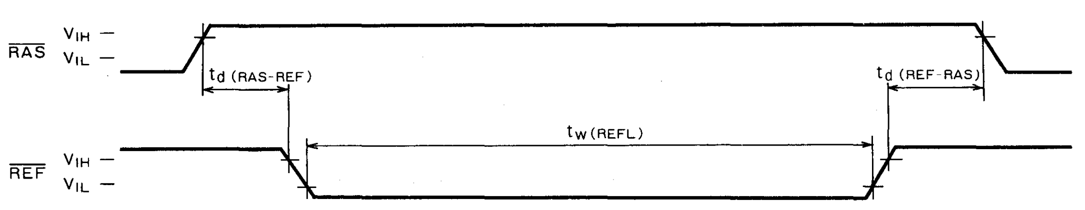
For $\overline{RAS}$-only, the controller disables the address multiplexer upon entering the memory cycle while it enables the refresh counter output. Next, after a delay time of t_1 (required because of the address buffer delay time and row address setup time t_{su}(RA-RAS)), $\overline{RAS}$ is set to low. The refresh cycle ends at the time t_2 that RAS is precharging.

In contrast to this, the automatic refresh controller sets $\overline{REF}$ to low simultaneously with the beginning of the refresh cycle, and after the $\overline{RAS}$ precharge time t_3, the refresh cycle ends. For this reason, there is no necessity to consider the settling time for address selection.

Self Refresh

Self refresh, similar to automatic refresh, sets $\overline{REF}$ low after $\overline{RAS}$ precharge occurs, beginning the internal refresh cycle. This method of refresh ignores all other inputs as long as $\overline{RAS}$ is high and $\overline{REF}$ is low, making use of an internal timer to automatically refresh the row addresses every 12~16μs which enables all cells to be refreshed within 2ms. The rising edge of $\overline{REF}$ terminates the refresh operation and after one cycle ($t_{d(REF\text{-}RAS)}$) a normal read-write cycle is entered. Fig. 5 shows the timing for the self refresh cycle. Self refresh is an extremely effective method of providing memory backup by means of a secondary power supply. As shown in Fig. 3.44, most of the required functions are implemented within the chip for the $\overline{RAS}$-only refresh with a simplified external circuit. This results in low power consumption and a long life for the secondary power supply.

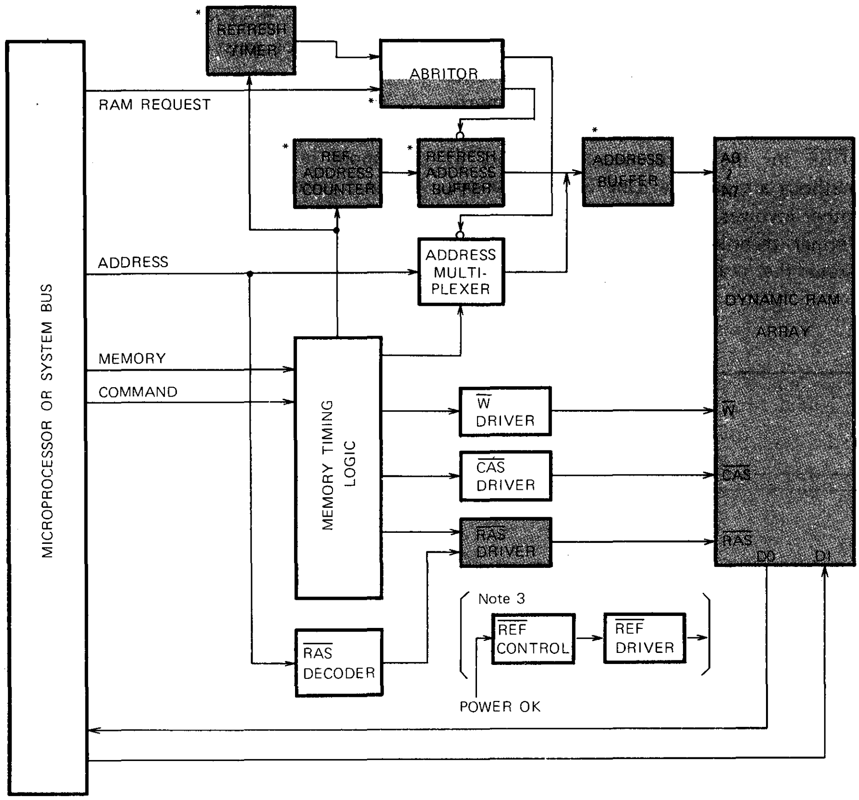
As described above, self refresh may not be used in the RAS only refresh mode. In designs using two refresh counters (internal and external) which operate independentry, guaranteeing the refresh (2ms) time is difficult.



Note: Addresses, $\overline{CAS}$, D and $\overline{W}$ are don't care.

Symbol	Parameter	Alternative Symbol	M5K4164S-15 Limits Min	M5K4164S-15 Limits Max	M5K4164S-20 Limits Min	M5K4164S-20 Limits Max	Unit
$t_{d(RAS\text{-}REF)}$	Delay time, $\overline{RAS}$ to $\overline{REF}$	t_{RFD}	100		120		ns
$t_{w(REFL)}$	$\overline{REF}$ low pulse width	t_{FBP}	8000	∞	8000	∞	ns
$t_{d(REF\text{-}RAS)}$	Delay time, $\overline{REF}$ to $\overline{RAS}$	t_{FBR}	295		360		ns

Fig. 3.44 Self-Refresh Timing



Note 1. block shows the device which should be backed up during power down.

2. * block can be powered down using $\overline{\text{REF}}$ (PIN1)

3. During self-refresh, $\overline{\text{REF}}$ driver and controller have to be backed up.

Fig. 3.45 Typical dynamic RAM system with battery back up

Design Example

The design example shows the increased effectiveness of $\overline{\text{REF}}$ (pin 1) refresh when the M5K4164S is used as the memory for a microprocessor. This design example illustrates the interface between the M5K4164S and the microprocessor.

When using $\overline{\text{REF}}$ for the microprocessor memory in interface, two methods are possible. One is asynchronous refresh and the other involves synchronously refreshing the memory. The former technique is not affected by the microprocessor status (i.e. reset, wait state, $\overline{\text{DMA}}$, and cpu clock). However, control logic is somewhat complex. While the second method makes use of simple control logic, the microprocessor must satisfy the refresh operation timing conditions.

Fig. 3.46 through 3.48 show the block diagram, schematic diagram, and timing diagram for the asynchronous refresh example. For this example, the refresh cycle counter refresh request ($\overline{\text{REFREQ}}$) starts the refresh cycle independently of the microprocessor operation. The arbiter determines whether the microprocessor (RAMREQ) or refresh cycle counter ($\overline{\text{REFREQ}}$) has access to the RAM.

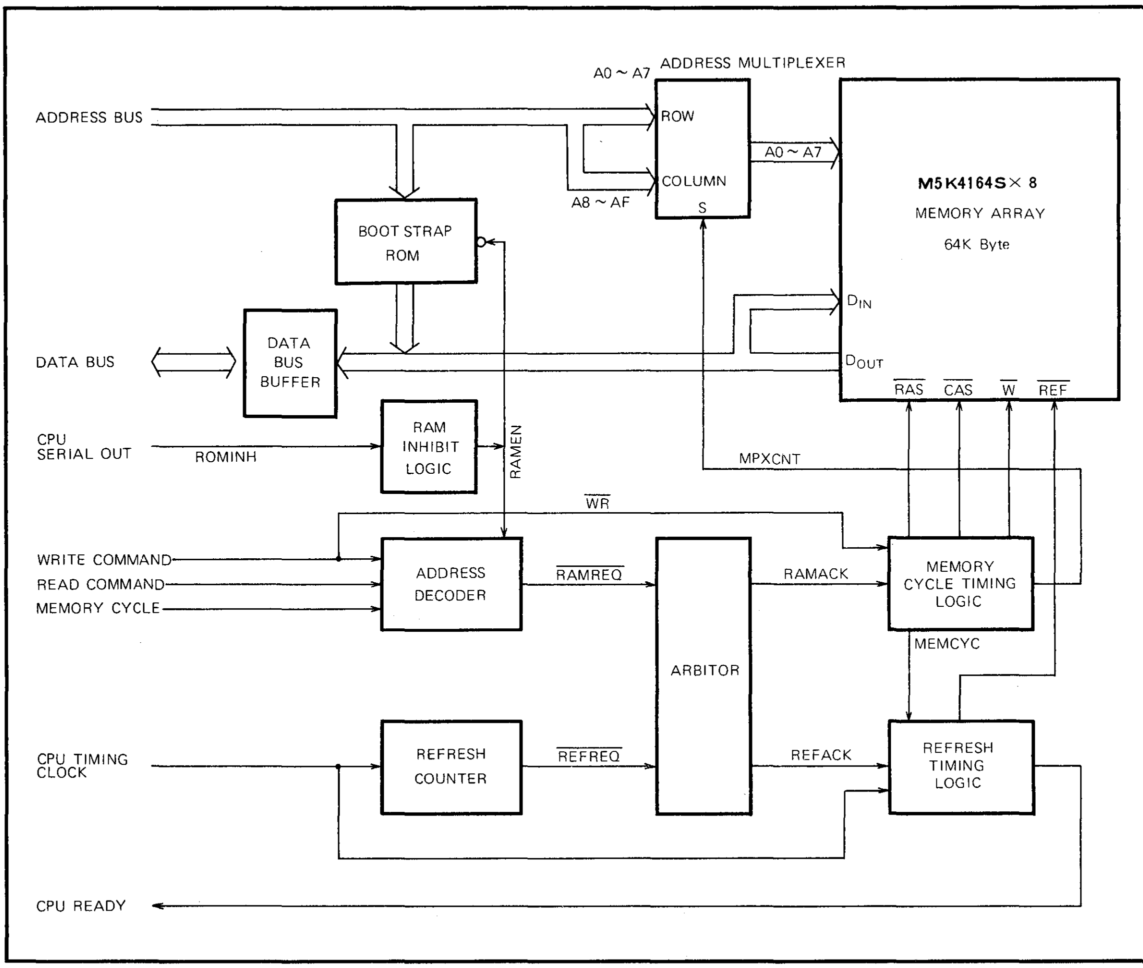
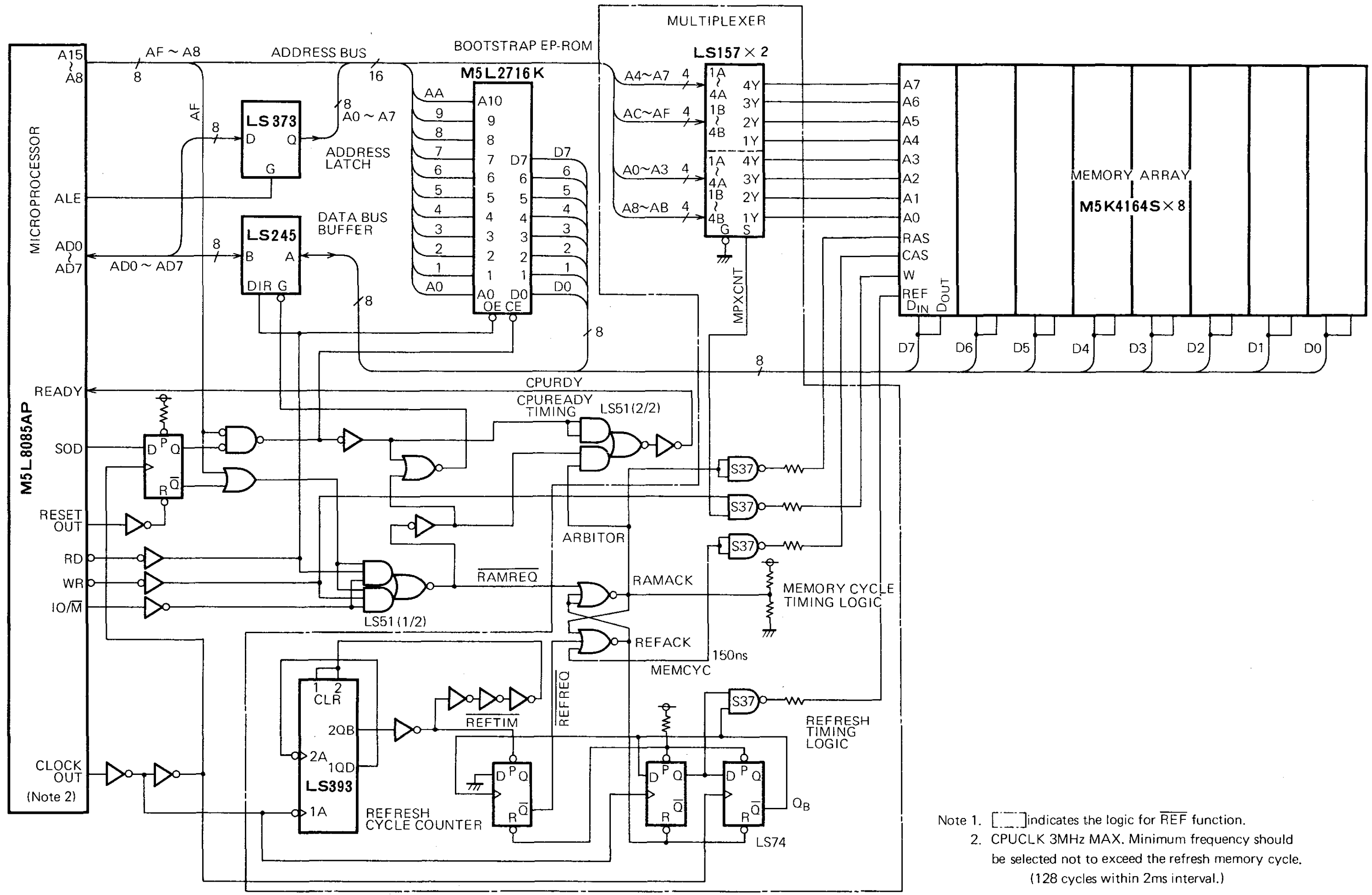
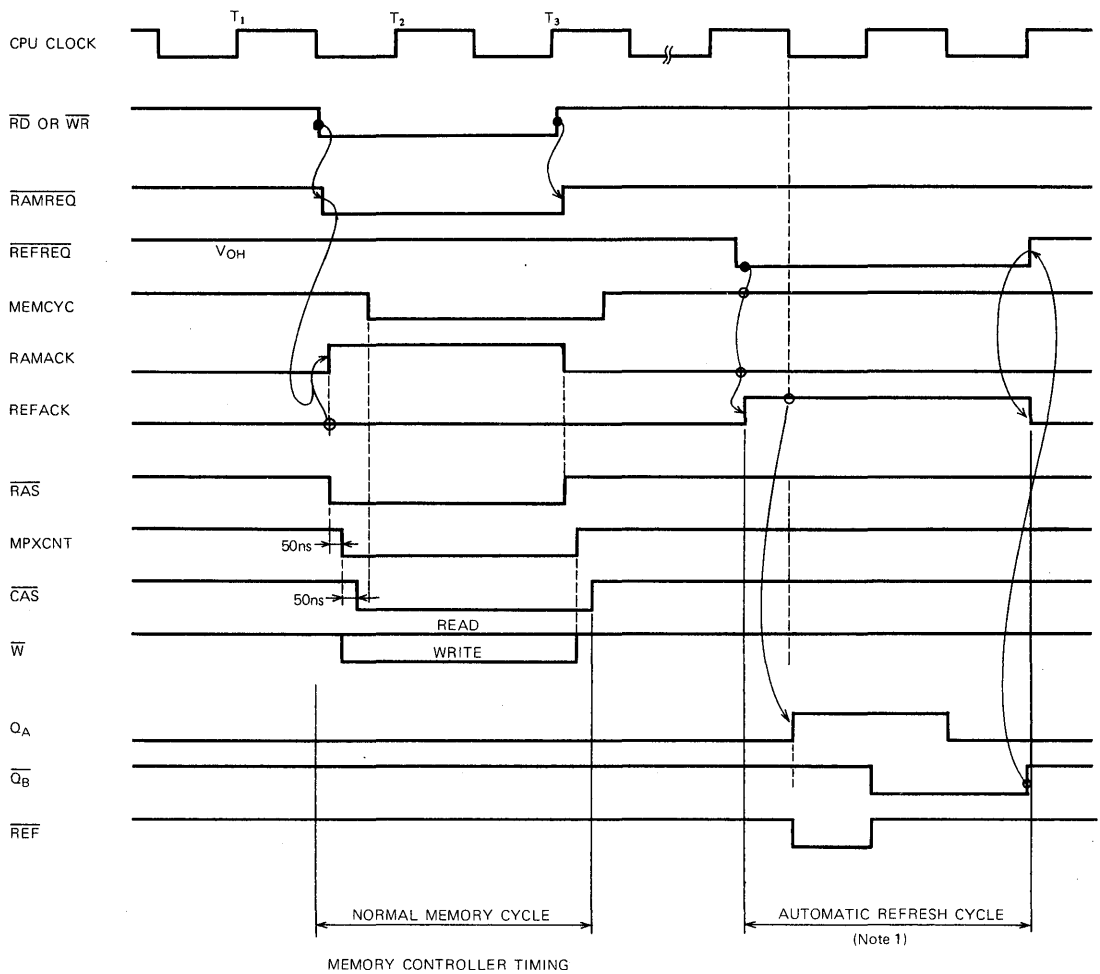


Fig. 3.46 Block diagram of the design example



Note 1. [ ] indicates the logic for $\overline{REF}$ function.
2. CPUCLK 3MHz MAX. Minimum frequency should be selected not to exceed the refresh memory cycle. (128 cycles within 2ms interval.)

Fig. 3.47 Design example of microprocessor interface (Ansynchronous)



Note 1: If the microprocessor attempts to access the RAM during this period, it must wait until the refresh is over.

Fig. 3.48 Memory and refresh timing (Anshronous)

A bootstrap ROM, commonly used in this type of memory system, is shown in the example. This is used to load the initial program of a RAM-based system into RAM from disk, for system initialization. In this example, the SOD (Serial Output Data) of the M5L8085AP is used to select either the bootstrap ROM or RAM as shown in Fig. 3.49.

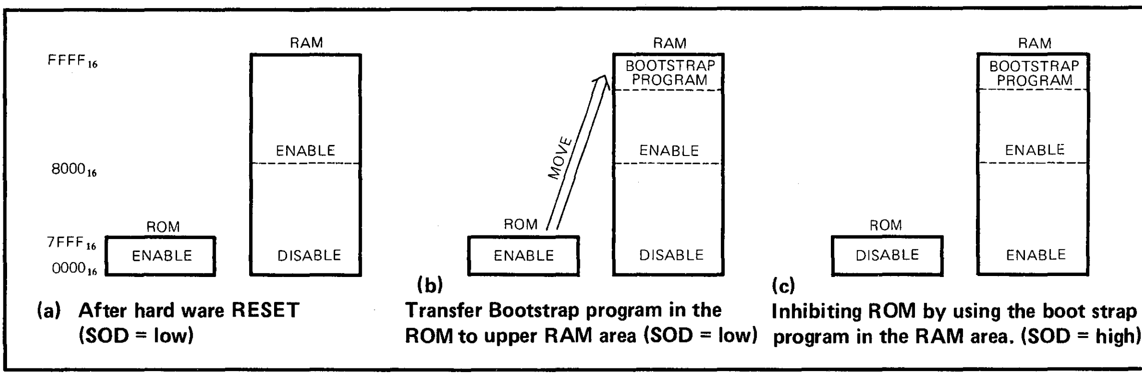


Fig. 3.49 Start-up procedure of the memory overlaped system

Fig. 3-50 and 3.51 show the schematic diagram and timing for the synchronous refresh example. In this example a Z80 microprocessor is used with synchronous refresh. As shown in Fig. 3.51, after the Z80 fetch instruction, the refresh operation is performed (T_3 and T_4 state).

In this manner refresh is performed synchronously with microprocessor operation. As mentioned previously, this type of operation involves a variety of limitations which must be considered carefully when designing such a system. (i.e. wait state, DMA, reset and CPU clock cycle)

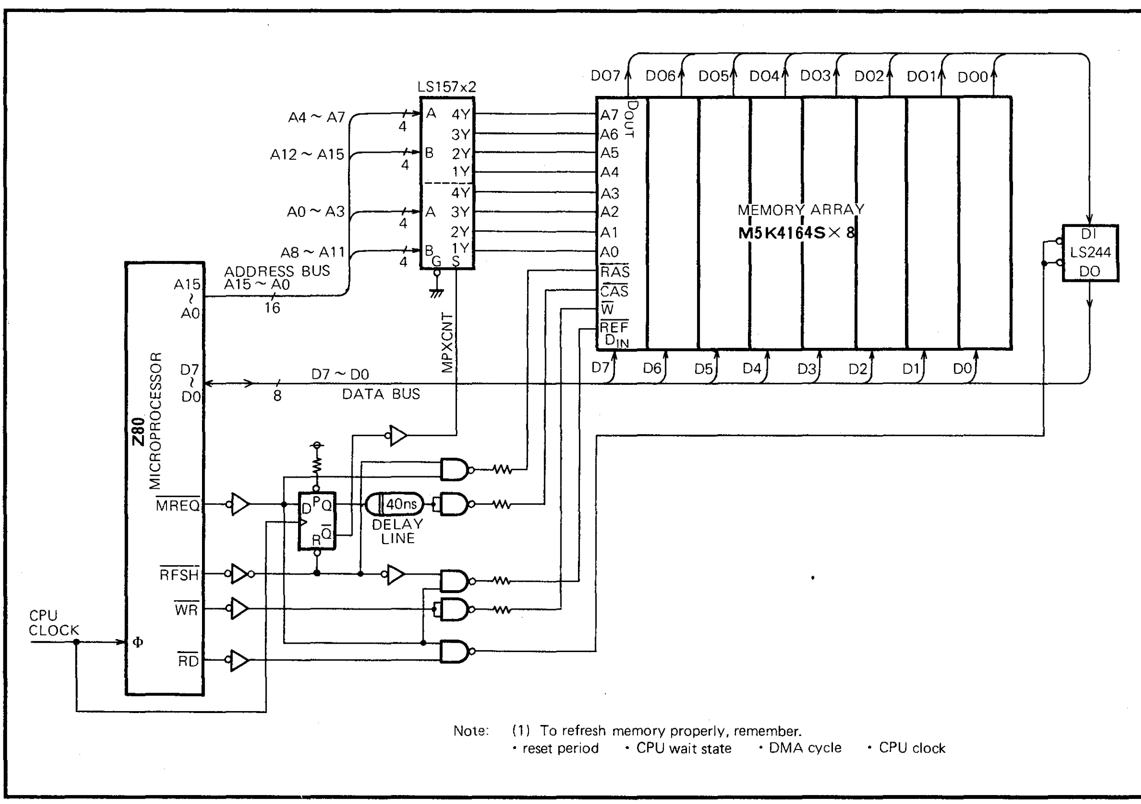


Fig. 3.50 Design example of microprocessor interface (Synchronous)

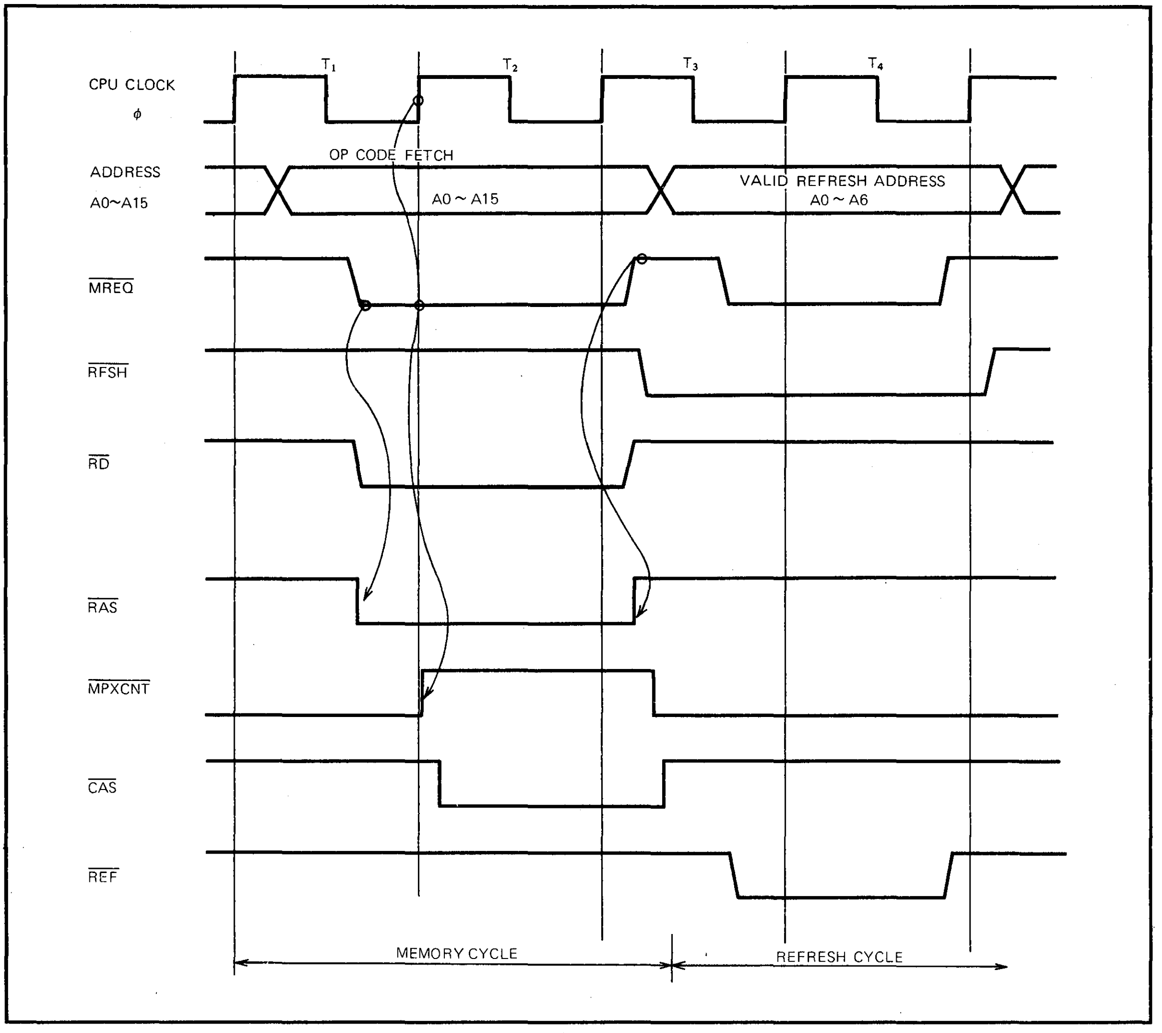


Fig. 3.51 Op code fetch and refresh timing

4. STATIC RAM

4.1 M58725P Technology

M58725P Features

The M58725P is a 2K by 8-bit static RAM making use of the latest n-channel MOS technology. Beginning with the 256 bit p-channel RAM, this technology has progressed at a rate which has seen the bit capacity quadruple in 3 to 4 years and the M58725P is the latest stage of development of this technology. A high-resistivity polysilicon material is used to lower power consumption and a chip select feature is used to implement a power cut function as the major features of the new device in addition to an overall improvement in performance. Fig. 4.1 shows the progress that has been made in the field of static RAMs, using the product of per bit power consumption and access time as an index of performance. From this the significant progress that has been made with these devices in recent years becomes apparent.

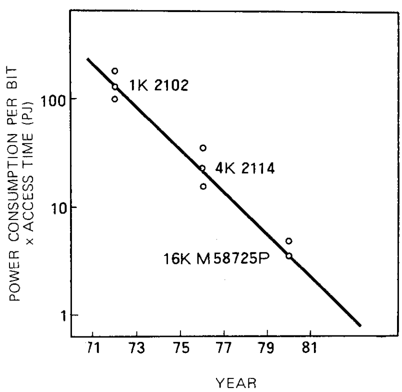


Fig. 4.1 Performance improvements in MOS RAMs

Memory Cell Structure

Fig. 4.2 shows the memory cell structure which consists of six elements; four transistors and two resistors. These resistors are implemented by means of a high-resistivity polysilicon which has enabled a significant reduction of power consumption within the memory cell. The holding current in a static RAM is basically that current which is required to cancel the p-n junction leakage current, an extremely small current. However, in previous devices the implementation of high resistances has been difficult, resulting in unnecessarily large power consumption. The M58725P using a two-layer polysilicon process has implemented this resistance by making the resistance of the second layer high (approximately 100MΩ), thereby also resulting in a reduction in cell area.

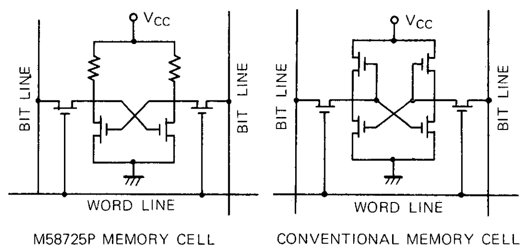


Fig. 4.2 Memory cell structures

Power Cutting with the Chip Select Signal

In order to take full advantage of the inherently low power memory cells of the M58725P, a chip select signal ($\overline{S}$) is used to provide a power cutting function. In the standby mode the power consumption to the peripheral circuitry is almost completely eliminated, leaving a very small part of the consumption of the peripheral circuit and the inherently low consumption of the memory cells. As shown in Fig. 4.3, when S is made high the power consumption is 1/10 of the normal level. This feature is particularly effective in lowering overall consumption in systems which use several RAM devices.

When the $\overline{S}$ signal changes levels, the circuit current changes rapidly so that decoupling capacitors in the power supply line are recommended to prevent noise generation.

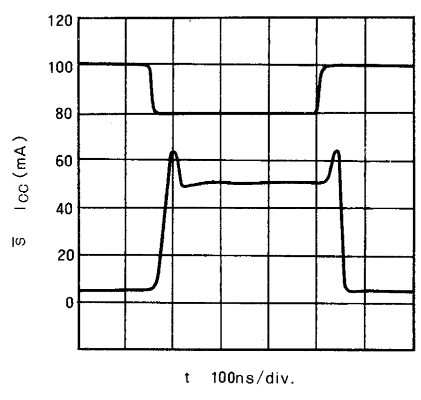


Fig. 4.3 Power cutting feature

$\overline{OE}$ Signal

The output enable signal $\overline{OE}$ controls the output for use on two-way buses to prevent several devices from bus contention. For writing, this signal is made high, thereby disabling the output. The output is enabled by setting it low as for reading operations. When using the M58725P with the 8085, 8086 or similar microprocessors, the CPU $\overline{RD}$ signal can be connected to the $\overline{OE}$ input to effectively prevent bus contention. Another method does not rely on use of the $\overline{OE}$ signal. Refer to the section on writing operations for details.

FUNCTIONAL DESCRIPTION

Read Operations

Fig. 4.4 shows the timing relationships for read operations. The section at (a) illustrates the access time from address selection. Since the M58725P is a completely asynchronous device, reading data at the DQ outputs requires only that the address be selected with the $\overline{OE}$ input low. If the setting of $\overline{OE}$ is late, the access time will depend on the timing of the falling edge of the $\overline{OE}$ signal. If it is sufficiently fast, the access time will be determined by the time t_a (A).

Fig. 4.4 (b) illustrates the access time from the operation of chip select. With previously available RAMs without the power cutting feature, the chip select access time t_a (S) was quite a bit faster than the address access time t_a (A) in most cases. With the M58725P however, these two access times are the same careful consideration must be given to timing relationships when laying out PC boards.

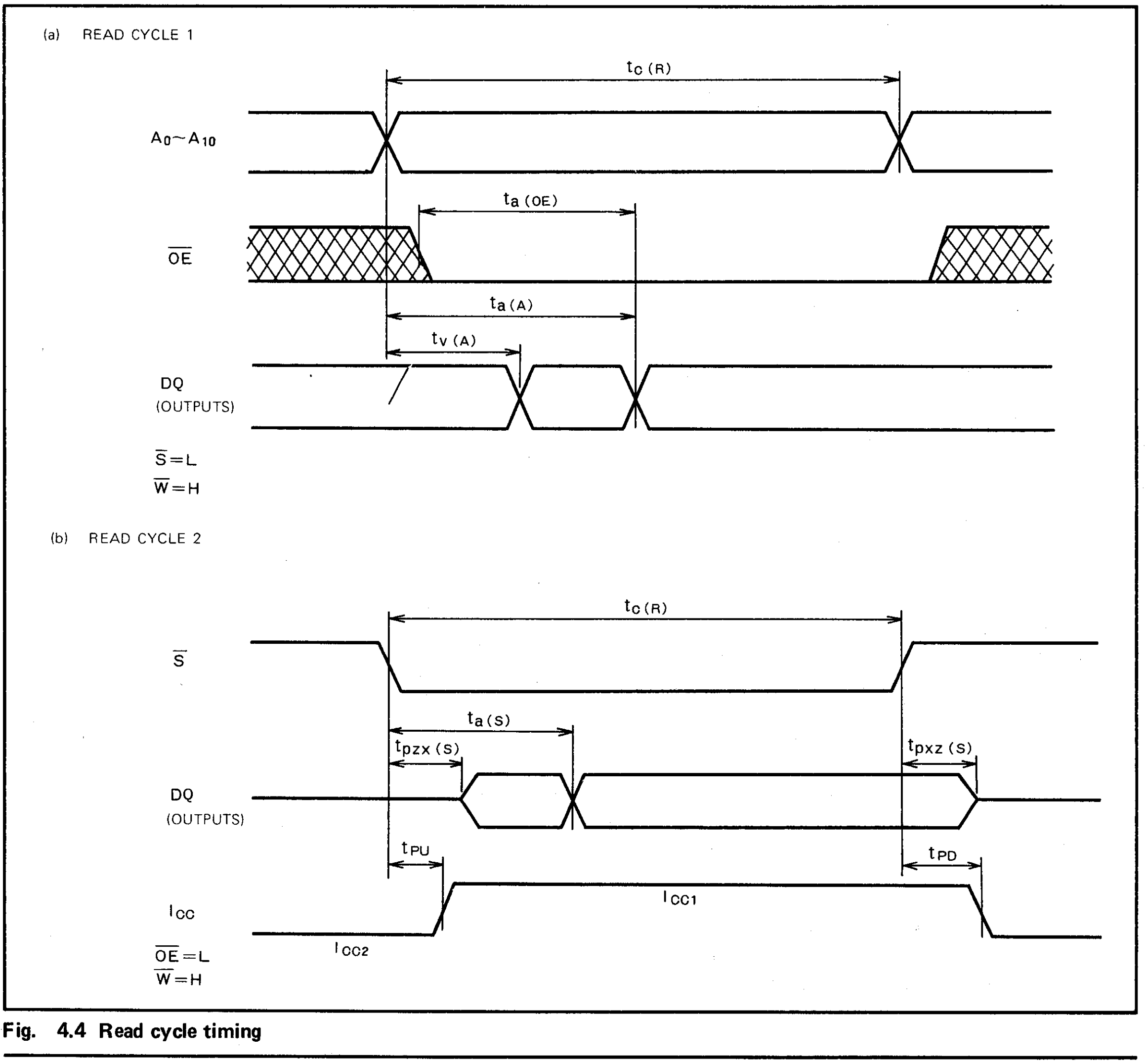


Fig. 4.4 Read cycle timing

MITSUBISHI ELECTRIC

Write Operations

Fig. 4.5 illustrates the timing relationships for write operations. The data at the DQ inputs will be written into memory upon the rising edge of either W or S but this data should be kept stable in the period between t_{su} (D) and t_h (D). To prevent bus contention before the data to be written is applied, the input $\overline{OE}$ should be set to high. If the $\overline{OE}$ signal is not used, the falling edge of the $\overline{W}$ signal must be made simultaneous with or earlier than the falling edge of the $\overline{S}$ signal. This will keep the DQ pins in the high-impedance state and prevent bus contention. For use with the 8085, 8086 or similar processors, the $\overline{S}$ signal can be gated with the $\overline{WR}$ and $\overline{RD}$ signals to achieve this effect. This is required to prevent data from being written into the address of the cycle previous to t_{su} (A) or following t_{wr}.

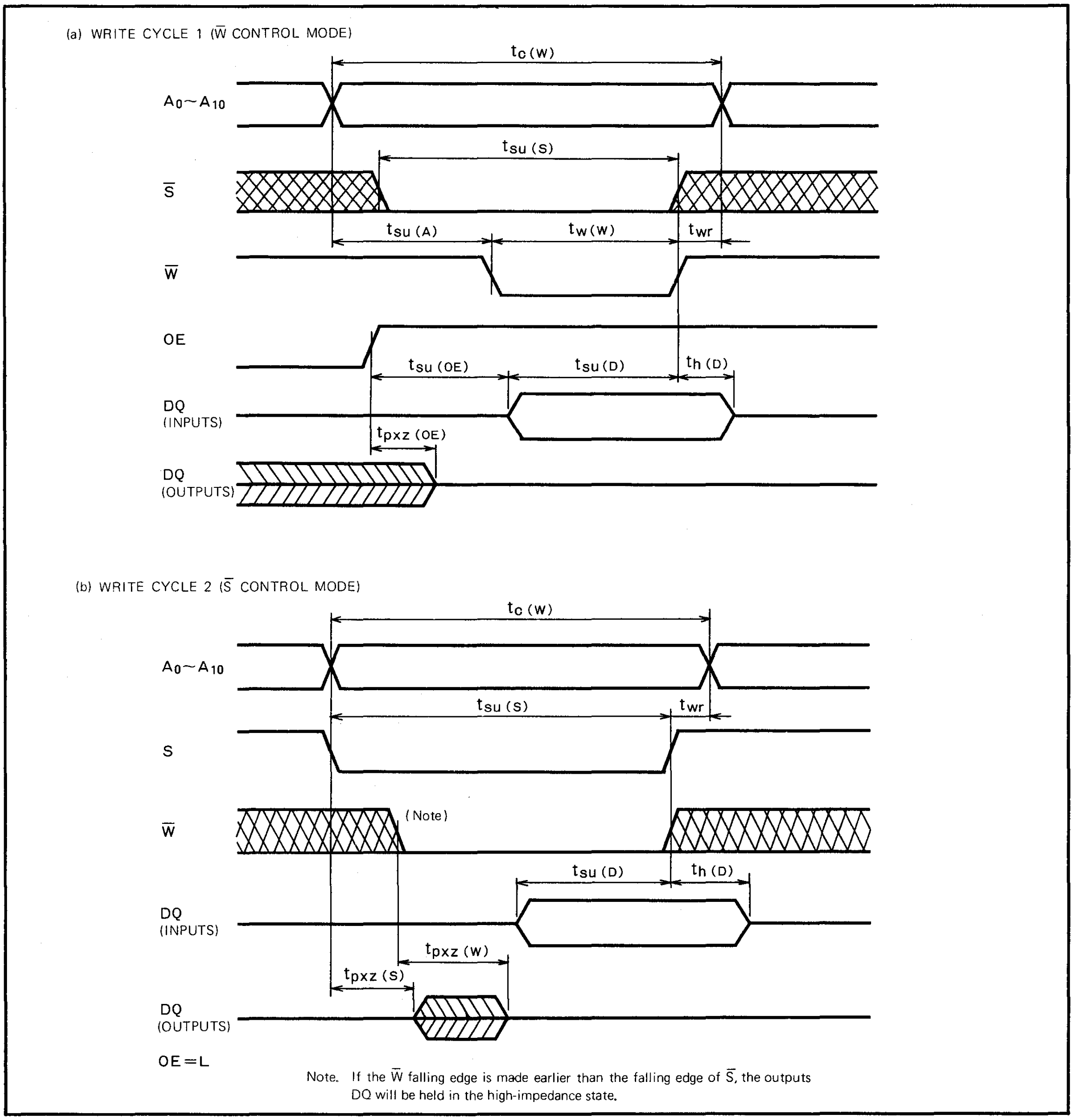


Fig. 4.5 Write cycle timing

4.2 Static RAM Application

Mainframe memories tend to be large in bit capacity of various word widths. Microprocessor memories, in contrast, are typically smaller in size with a fixed word width of 8 or 16 bits. Static RAM, which do not require special refresh or timing circuitry, are suitable for microprocessor application which demands low cost and ease of use in a 4- or 8-bit memory organization.

M5L2114L

When using a static RAM like the M5L2114 which has common input/output and no $\overline{OE}$ (output Enable), the problem of bus contention should be considered. Fig. 4.6 shows the decoding technique for the M5L2114L.

Controlling the chip select with the READ or WRITE command prevents the M5L2114L from driving against the transceiver prior to the command. The limitations of this method are access time to read or write and a limitation of $\overline{CS}$ to write setup and hold time. The timing diagram is shown in Fig. 4.7.

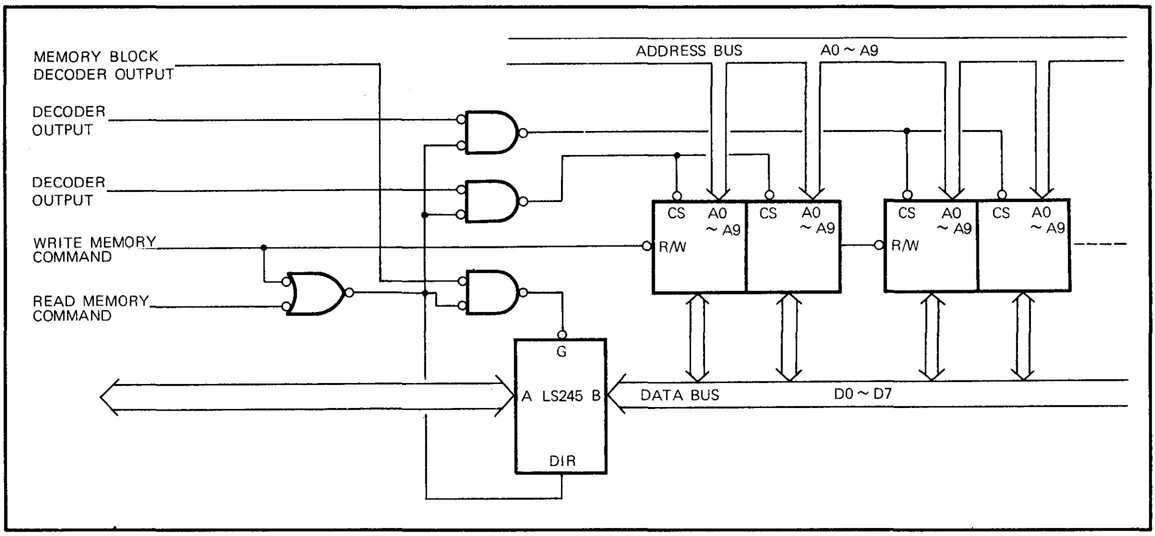


Fig. 4.6 M5L2114 decoding example

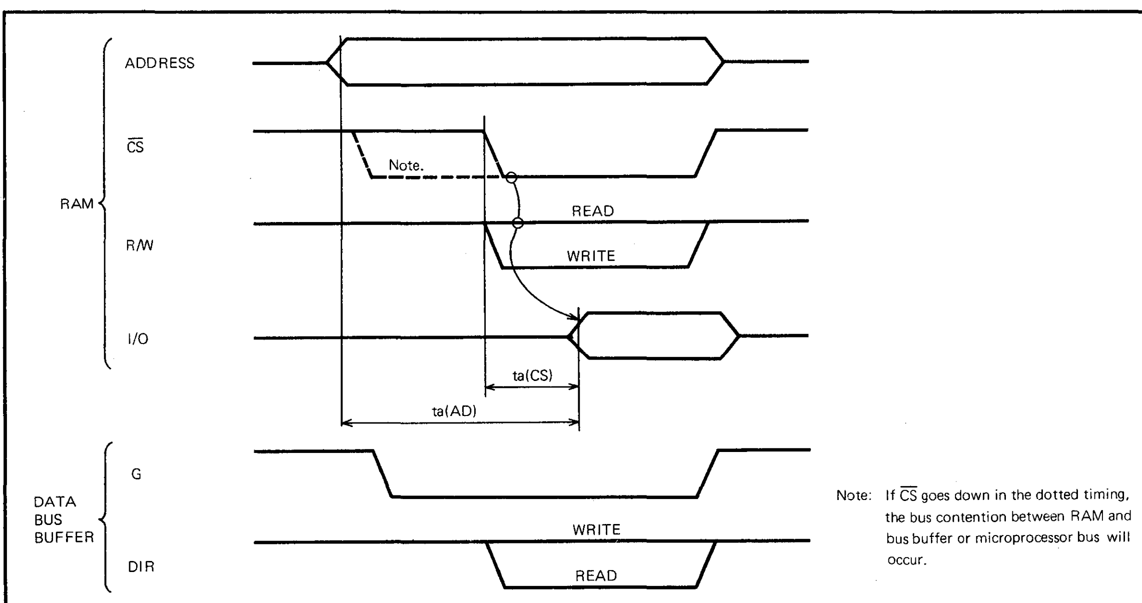


Fig. 4.7 M5L2114L timing diagram

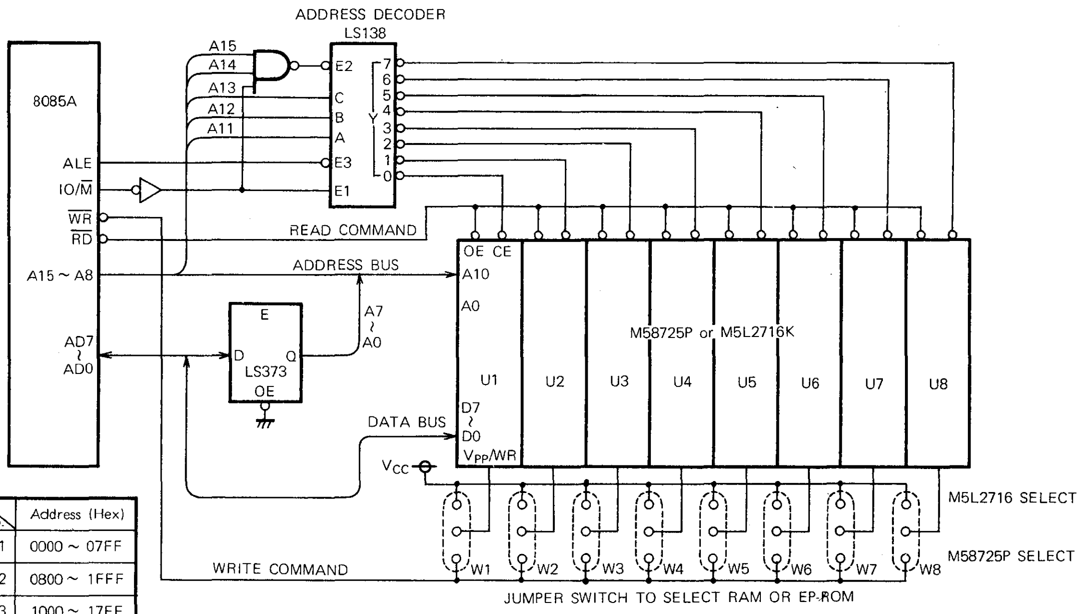
MITSUBISHI ELECTRIC

M58725

The 2K x 8 bit M58725 is more suitable for microprocessor applications, because the pin configuration of the M58725 is similar to that of the M5L2716K EPROM and the M58725 has byte memory organization. Fig. 4.8 shows an example of the M58725 interfaced to an 8085A microprocessor. RAM (M58725) or EPROM (M5L2716K) can be selected respectively by eight jumper switches. The interchangeability makes the system flexible and is convenient at the development stage of a microprocessor system which needs to use RAM.

I/O loading

There is capacitive loading at each memory's I/O pins and signal path line of the print current board. If many memories are used on a common bus, this loading prevents the guaranteeing of AC characteristics. For example, the M5L2114L has drive capacity of 2.1mA under capacitive loading of 100pF to guarantee the specified AC characteristics. A separate bus technique such as shown in Fig. 4.6 should be used in these cases.



No.	Address (Hex)
U1	0000 ~ 07FF
U2	0800 ~ 1FFF
U3	1000 ~ 17FF
U4	1800 ~ 1FFF
U5	2000 ~ 27FF
U6	2800 ~ 2FFF
U7	3000 ~ 37FF
U8	3800 ~ 3FFF

Fig. 4.8 Mixed EPROM and RAM application

5. CMOS STATIC RAM

5.1 M58981P Technology

Introduction

Since the low power consumption of CMOS RAM can be utilized to implement a battery backed-up non-volatile memory, the demand for these devices is increasing for use in equipment such as electronic cash registers and point-of-sales equipment. The M58981P was designed as a 4K-bit CMOS RAM to fill this demand. It utilizes silicon gate CMOS technology to achieve high speed and high density. The M58981P is an asynchronous CMOS RAM configured in 1,024-word by 4-bit words. It is pin-compatible with the Mitsubishi Electric M5L2114LP n-channel 4K static RAM and the Intel 2114. An advanced micro-pattern process has enabled the formation of 26,000 devices in an area of $20mm^2$, packaged in an 18-pin DIL plastic package.

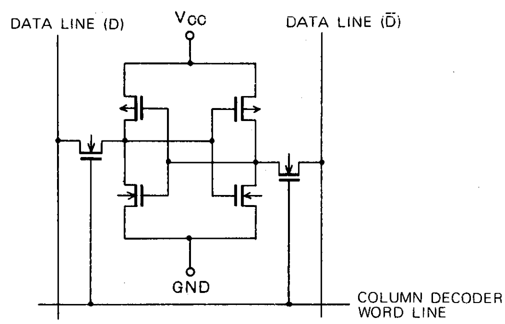


Fig. 5.1 Memory cell circuit

Memory Cell

The memory cell used in the M58981P is shown in Fig. 5.1. It consists of a CMOS circuit (with n-channel transfer gate) of 6 transistors.

Circuit Configuration and Operation

Fig. 5.2 shows the block diagram of the M58981P, with the timing diagram shown in Fig. 5.3.

(1) Write Operations

The address signals $A_0 \sim A_9$ select the address and when the R/W signal goes low, the I/O data at that time is written into memory. The write operation is performed during the overlap time when $\overline{CS}$ and R/W are low. When the I/O pin is in the output mode, care should be taken to prevent the data output from causing bus contention.

(2) Read Operations

The address is specified by the $A_0 \sim A_9$ signals. When the R/W signal goes high, the data from the specified address appears at the I/O pin.

(3) $\overline{CS}$ Signal

The chip select signal ($\overline{CS}$) selects the chip when set to low, and removes the chip from the bus when set to high. The output is floating enabling wired-OR connection with other chips. When the $\overline{CS}$ signal is high, no current flows in the input buffer (refer to block diagram). All word lines become low, and all memory cells are "unselected", the DC current flowing between V_{CC} and ground falling to a very low leakage current level. This is the standby condition, one of the major features of a CMOS RAM.

(4) Power Down Operation

Data hold can be accomplished with V_{CC} to 2V or greater. $\overline{CS}$ is made common with V_{CC} for (2V∿2.2V power down) or greater than 2.2V (power down greater than 2.2V) to hold memory contents.

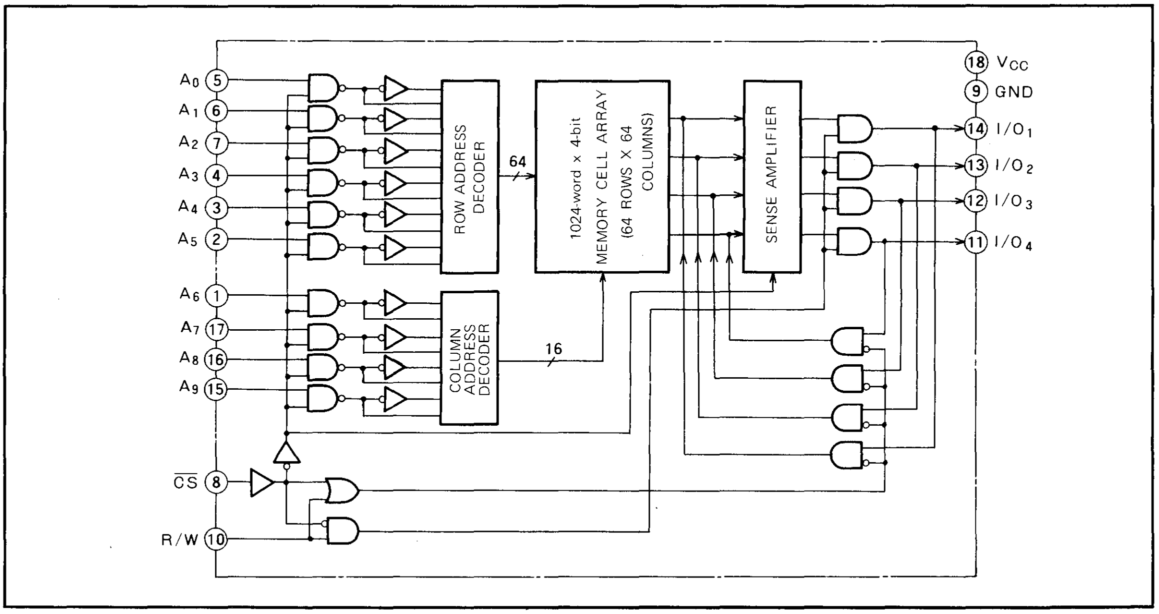


Fig. 5.2 Block diagram

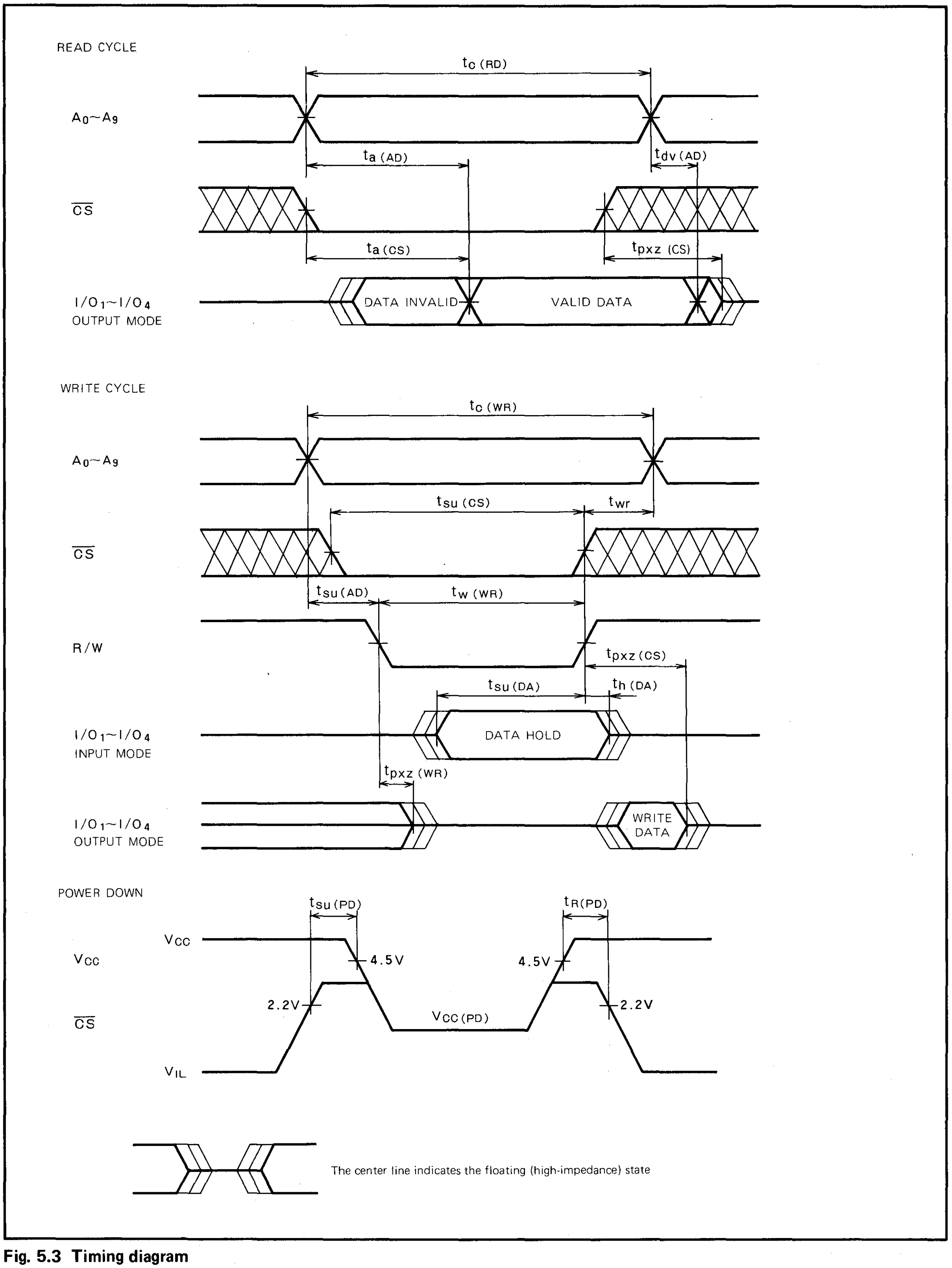
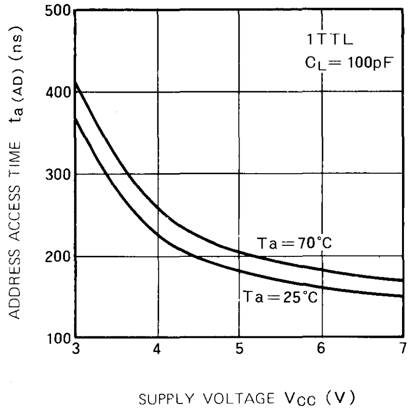


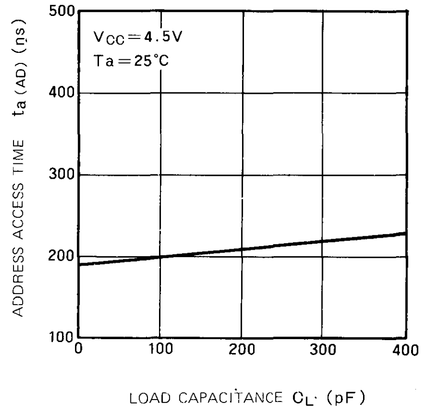
Fig. 5.3 Timing diagram

TYPICAL CHARACTERISTICS

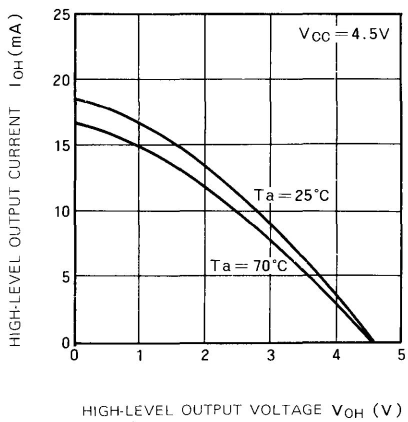
ADDRESS ACCESS TIME VS SUPPLY VOLTAGE



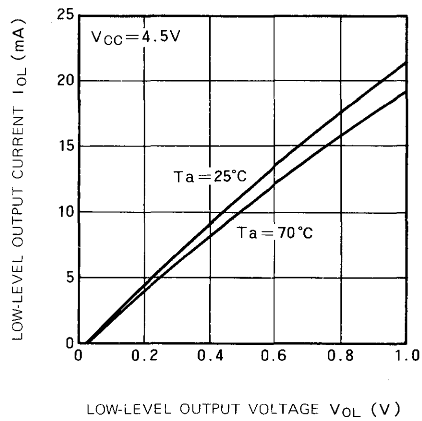
ADDRESS ACCESS TIME VS LOAD CAPACITANCE



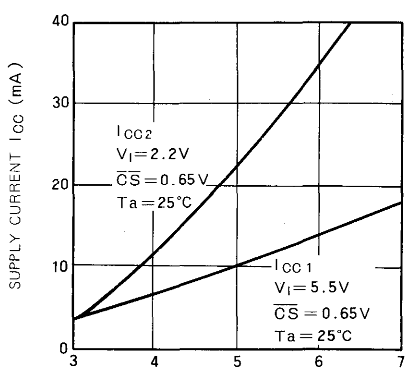
HIGH-LEVEL OUTPUT CURRENT VS HIGH-LEVEL OUTPUT VOLTAGE



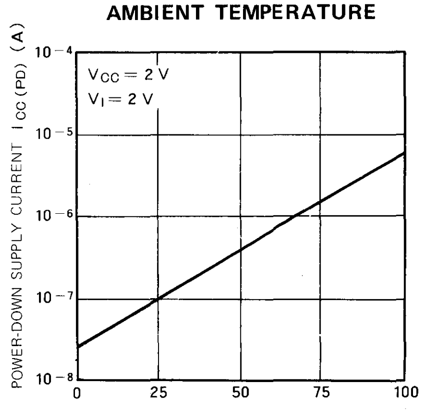
LOW-LEVEL OUTPUT CURRENT VS LOW-LEVEL OUTPUT VOLTAGE



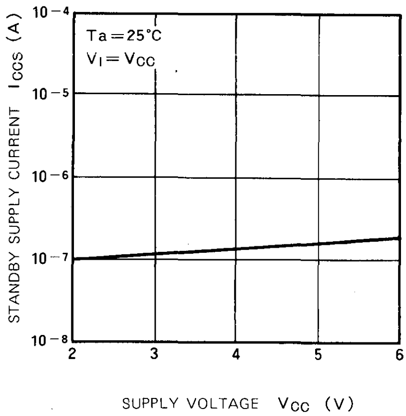
SUPPLY CURRENT VS SUPPLY VOLTAGE



POWER-DOWN SUPPLY CURRENT VS AMBIENT TEMPERATURE



STANDBY SUPPLY CURRENT VS SUPPLY VOLTAGE



Memory Map

Fig. 5.4 shows the M58981P memory map.

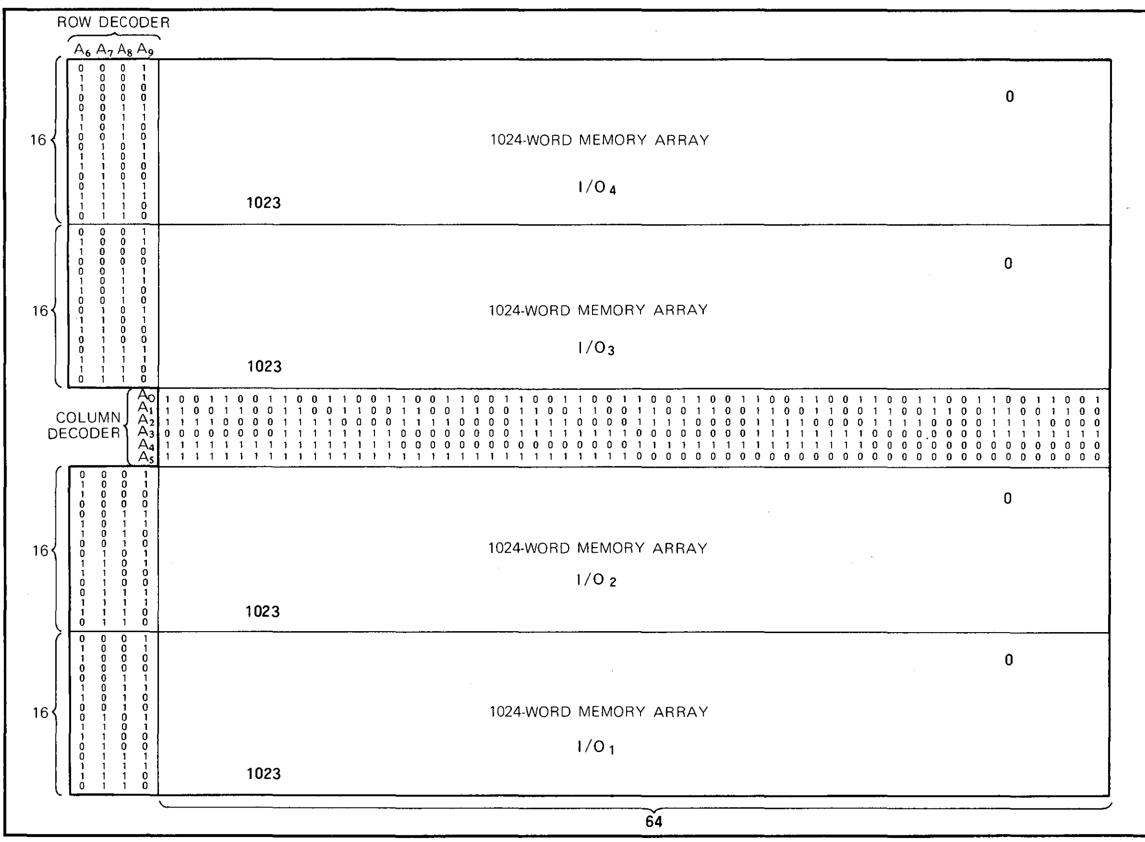


Fig. 5.4 Memory map

5.2 CMOS STATIC RAM APPLICATIONS

INTRODUCTION

Mitsubishi M5L 5101LP and M58981P are static RAMs that are fabricated with a CMOS technology. The M5L 5101LP is organized as 256 words of 4 bits, and the M58981P is organized as 1024 words of 4 bits. They are fully TTL-compatible, and use only a single 5V supply voltage V_{CC}.

The purpose of this application note is to describe the various circuit techniques for battery-supported non-volatile memory systems. Electrical characteristics for the two RAMs can be found on the previous pages.

NON-VOLATILE MEMORY SYSTEM

We can relatively easily design a large non-volatile memory system with little additional interface logic by using CMOS RAMs. The block diagram of a basic computer system that uses CMOS RAMs is shown in Fig. 5.5, and the power supply on-off timing of the system are shown in Fig. 5.6. It is usually necessary to have advanced warning that AC power has been lost. This warning signal produced by the power-fail-detect circuit interrupts the processor, which stores the volatile data in the non-volatile area (CMOS RAMs) before the system's DC source drops. And after the RAMs have been protected, their V_{CC} power source is replaced by V_{BAT}, as shown in Fig. 5.6.

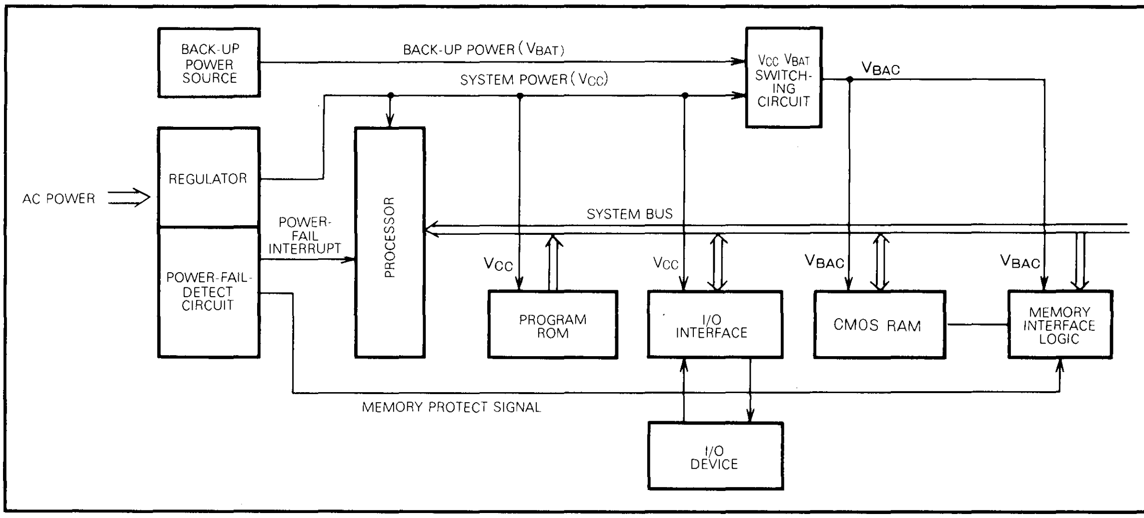


Fig. 5.5 Non-volatile memory system

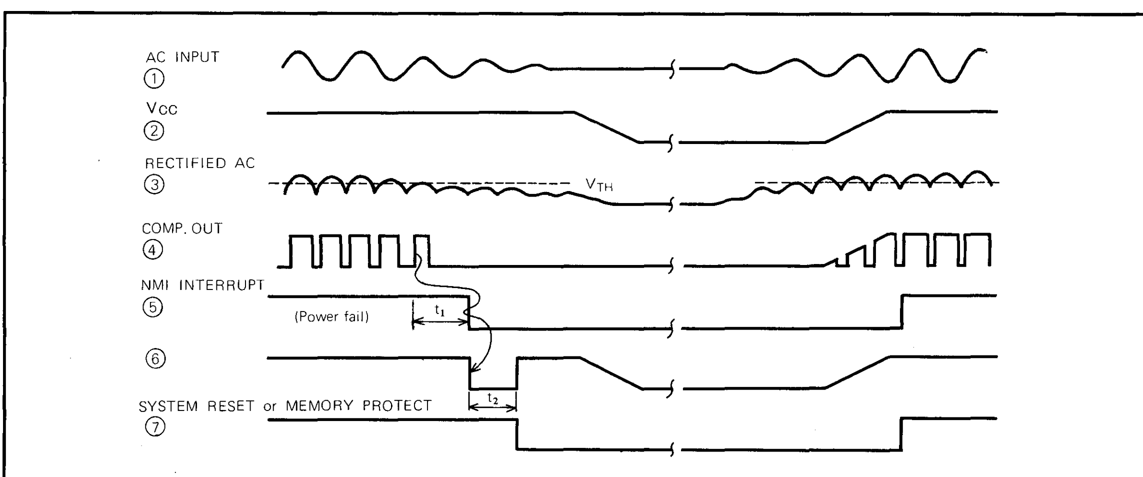


Fig. 5.6 Power on-off timing

EXAMPLE OF CMOS NON-VOLATILE MEMORY SYSTEM

Power-Failure Detection

The power-fail-detect circuit watches a separate power supply point to provide an advanced warning of power failure. As described before, this warning signal (power fail) can interrupt the processor or merely protect the CMOS RAMs.

Fig. 5.7 is a simplified diagram of the power-fail-detect circuit. This shows that the power failure is detected from the secondary transformer output, which is not regulated. The Zener-diode voltage and RC time constant should be well selected to prevent AC power failure from shutting down the memory system.

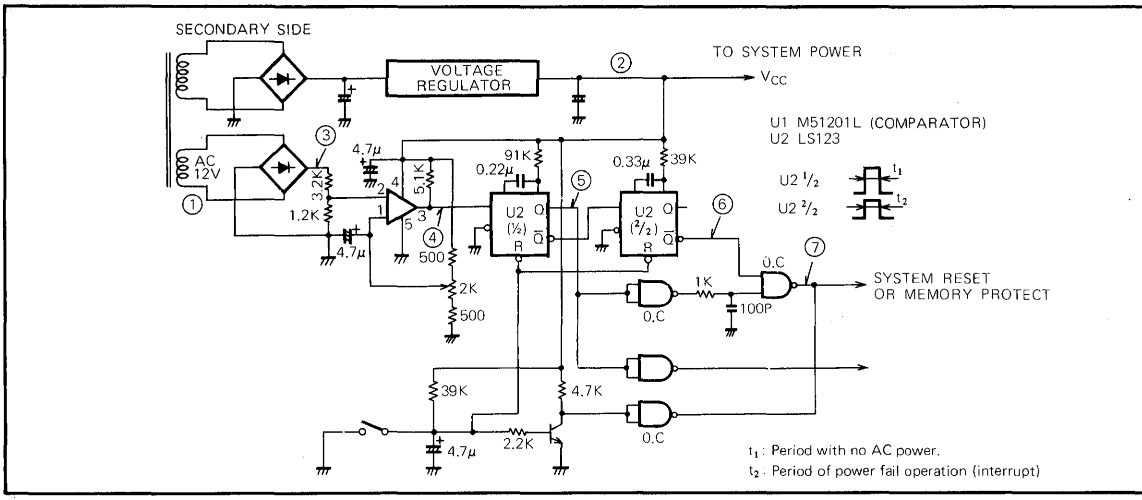


Fig. 5.7 Power-fail-detect circuit

Power-Switching Circuit

The power-switching circuit replaces the main source V_{CC} by the back-up power source V_{BAT} when V_{CC} drops, and replaces the V_{BAT} by the V_{CC} when the V_{CC} voltage rises enough to enable normal operation.

Two types of power-switching circuit are shown in Fig. 5.8 and Fig. 5.9. The diode-coupled circuit in Fig. 5.8 requires the main DC supply V_{CC} to be above the required V_{BAC} voltage by the amount of drop through the diode (about 0.6~0.7V). Fig. 5.9 shows a transistor-coupled circuit, which has better performance than the circuit in Fig. 4. In this case it is recommended to use a transistor with low collector-base saturation for Q1.

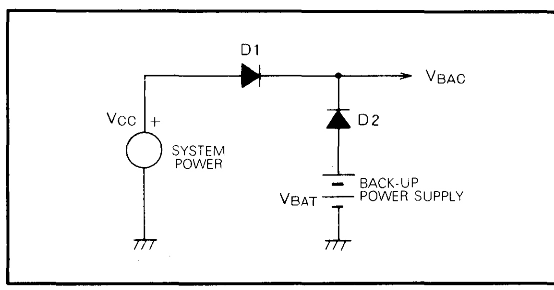


Fig. 5.8 Diode-coupled switching circuit

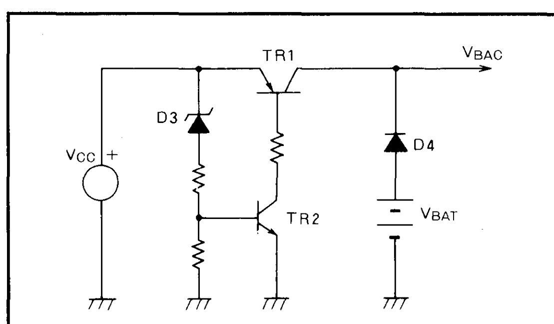


Fig. 5.9 Transistor-coupled switching circuit

TYPICAL APPLICATION CIRCUIT

An Example of M5L5101LP-1 Application

An example of a 1K-byte non-volatile M5L5101LP-1 memory system is shown in Fig. 5.10. In this case, the memory-protect signal is detected from the voltage of power source V_{CC}. But it is better to watch the unregulated voltage (see Fig. 5.7) to produce the memory-protect signal that protects RAMs at the time when V_{CC} is dropping or rising as shown in Fig. 5.6. The CE2 pin is used for decoding the RAM array. When the RAMs are not selected (i.e. CE2 = low-level), they enter a stand-by mode, and the power supply current is extremely low.

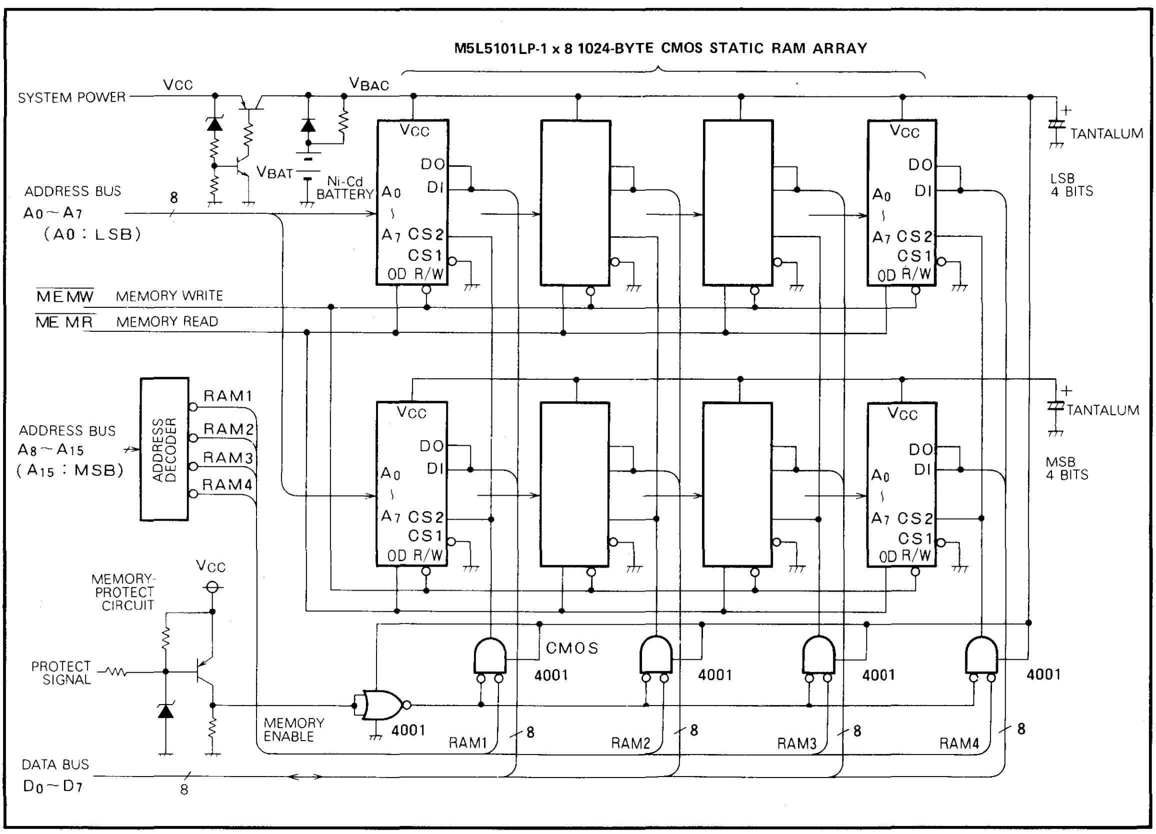


Fig. 5.10 Example of M5L5101LP

An Example of M58981P

The M58981P is a CMOS RAM which is fully pin compatible with M5L2114LP is organized as 1024 words of 4 bits. The M58981P has two control inputs, $\overline{CS}$ and R/W. The $\overline{CS}$ can control normal memory operation and stand-by operation. When the RAM is in the stand-by mode (i.e. $\overline{CS} \geq 2.2V$), the power supply current is extremely low.

Fig. 5.11 shows the memory signal timings at the time when AC power turns on and off. An example of 4K-byte non-volatile memory system using M58981P is shown in Fig. 5.12.

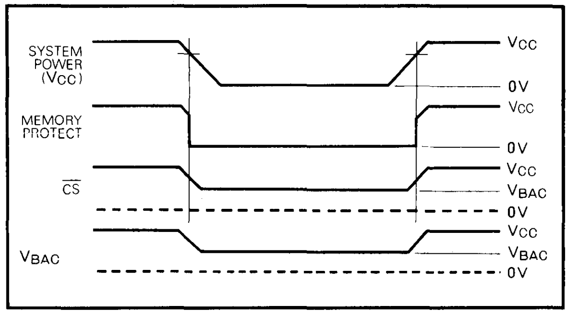


Fig. 5.11 Power on-off timing (M58981P)

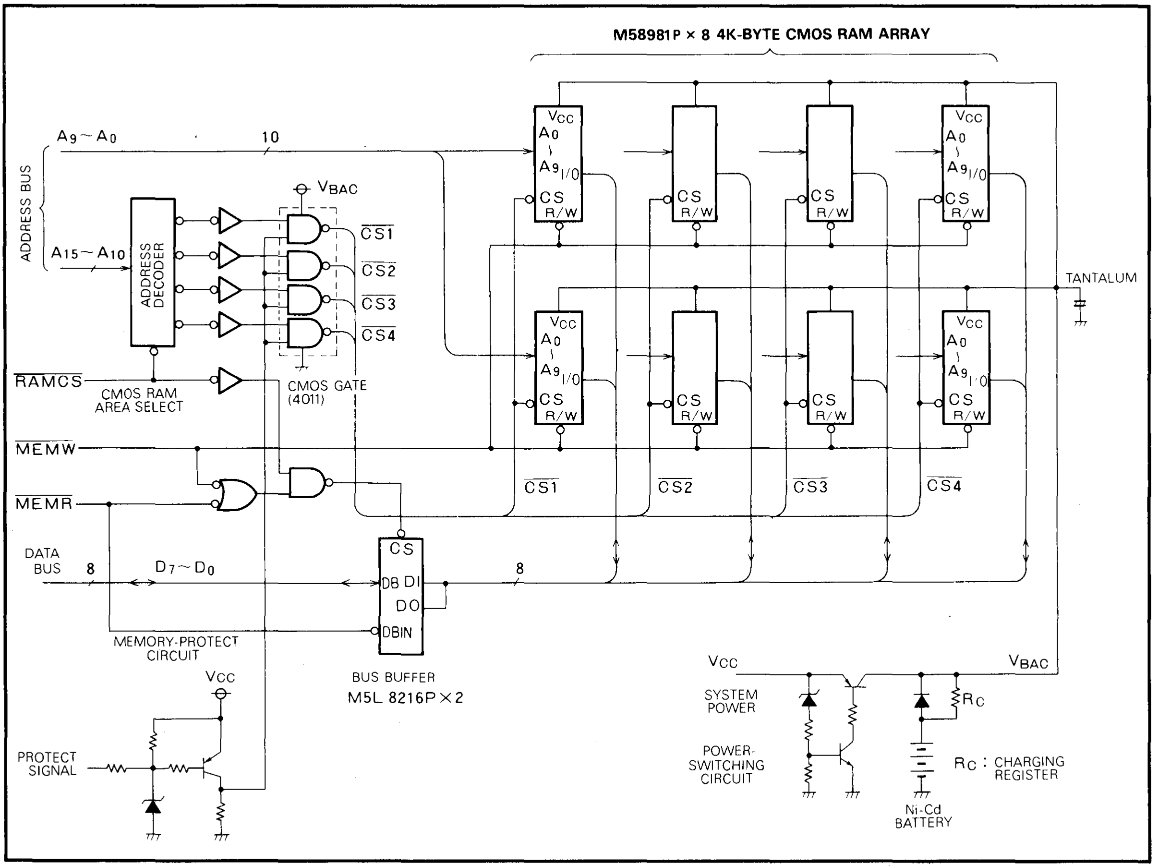


Fig. 5.12 Example of M58981P

Other Recommendations

1. Nickel-cadmium batteries are available for the memory back-up power source because of its rechargeable operation and wide variety of capacities, sizes and styles. For details, see related articles.
2. In order to decrease the DC power-source impedance, decoupling capacitors whose leak currents are small should be used. It is also necessary to use 0.01~0.1μF monolithic-type capacitors and 2~5μF tantalum types effectively.
3. When CMOS gates are used for decoding logic as shown in Fig. 5.10 and Fig. 5.12, it should be carefully ascertained that the propagation time of CMOS gates does not exceed the access time of memory, and also that the stand-by voltage of the gates does not drop below 3V. (It is possible to reduce the propagation time of CMOS gate using high-speed CMOS gate version TC40HXXX series.)

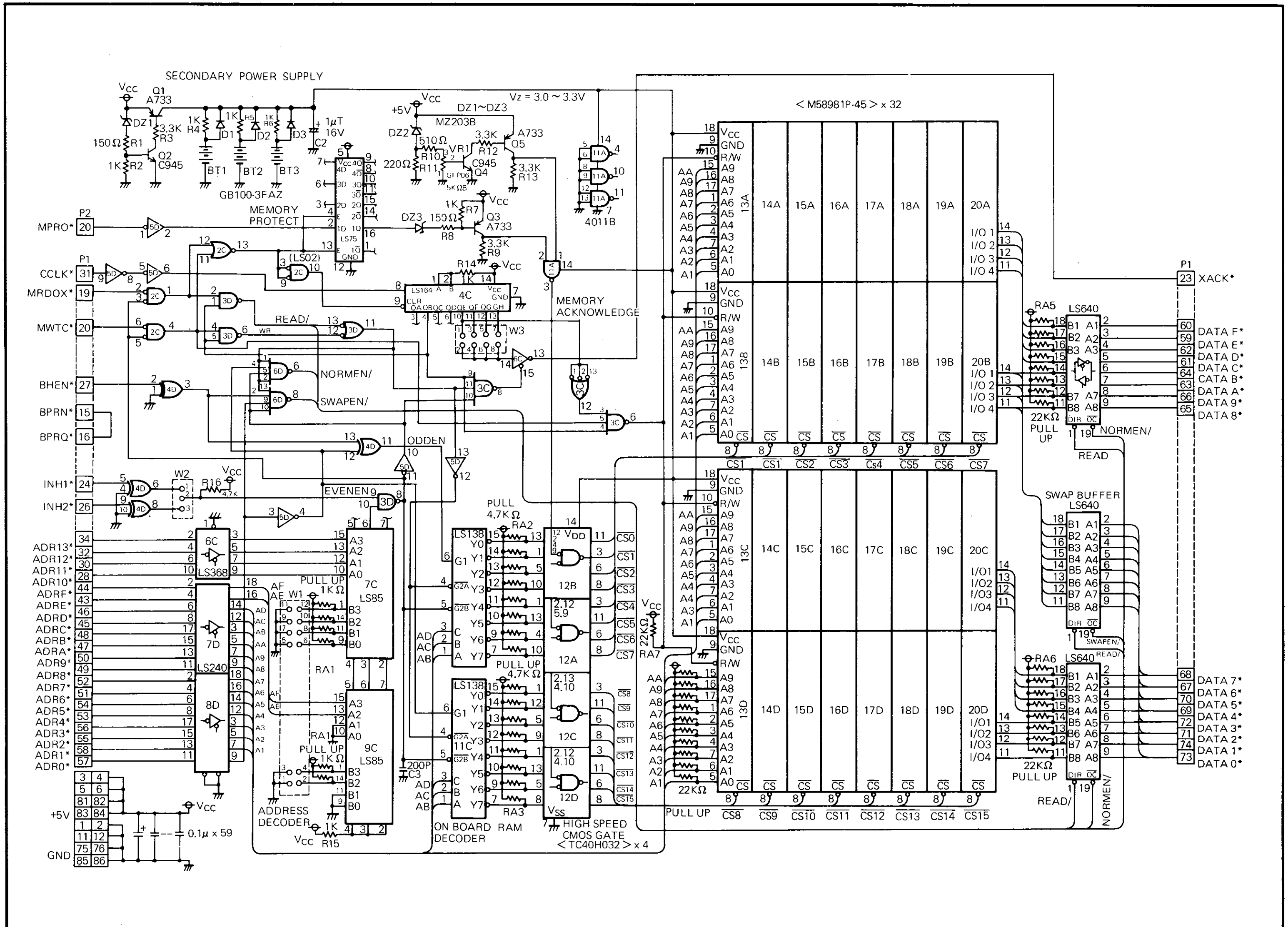


Fig. 5.13 Schematic diagram of 16K byte CMOS RAM board

MITSUBISHI ELECTRIC

Design example of *MULTI-BUS board

Design example of CMOS RAM board is shown in Fig. 5.13, and the block diagram is shown in Fig. 5.14.
This board is compatible with the proposed IEEE 796 bus standard, called *MULTIBUS

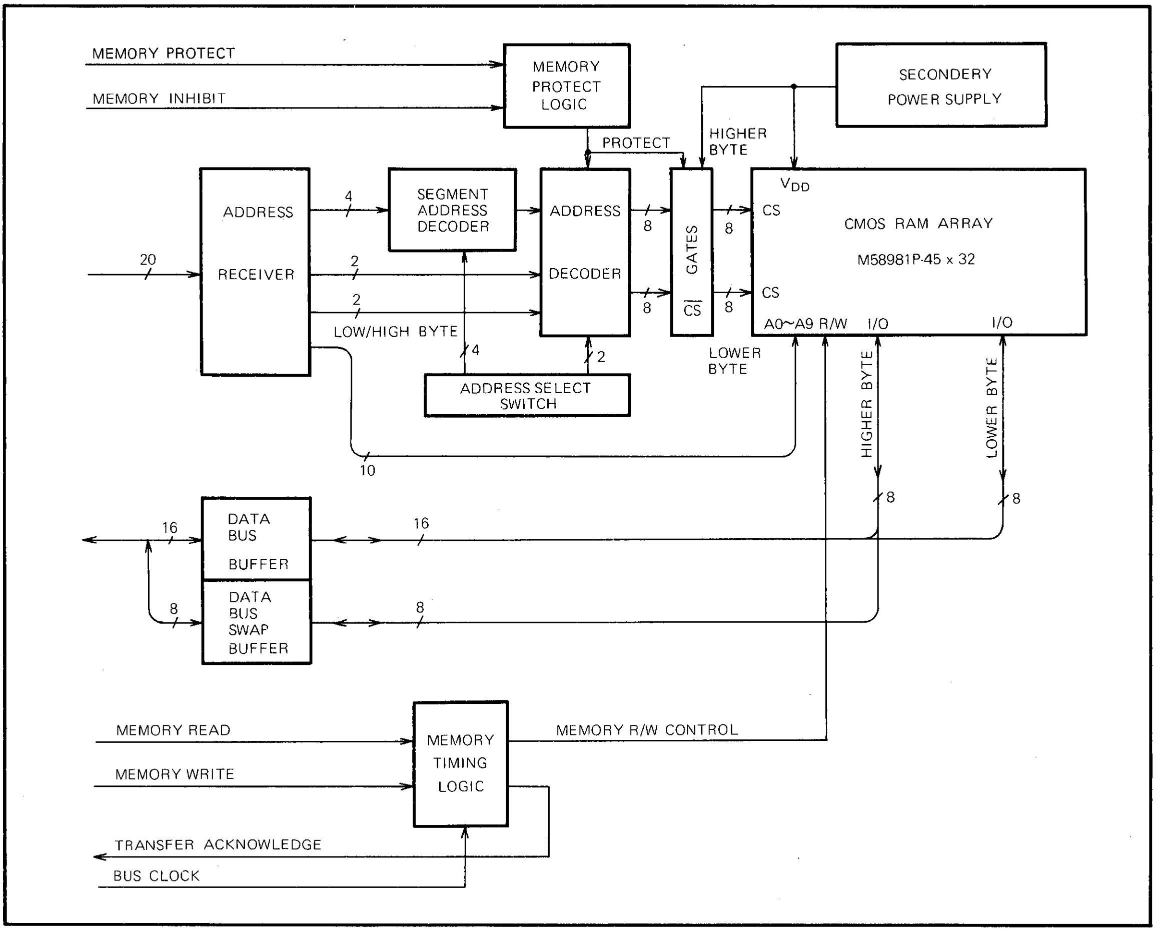


Fig. 5.14 CMOS RAM Board Block Diagram

* MULTIBUS is trademark of Intel corp.

6. EPROM

6.1 EPROM Technology

INTRODUCTION

With their ability to be electrically programmed and erased with ultraviolet light, EPROM (Erasable and Programmable Read Only Memory) devices have achieved high popularity for their ease-of-use and are retaining their position as the target for memory development.

Although the EPROM was originally developed for use as a microprocessor system debugging ROM, the device has undergone significant improvements in density, reliability, and basic process technology as well as cost per bit which have extended its usefulness beyond microprocessors into such equipment as cash registers, point-of-sale equipment, household appliances, entertainment equipment, and a variety of other fields. Since the introduction by Mitsubishi Electric of a p-channel 2K-bit EPROM, the development of n-channel devices has enabled remarkable improvements in access time, and density in the form of an 8K-bit device. The development of devices operable from a single power supply greatly improved ease-of-use of the 16K- and 32K-bit devices which were to follow. This section will briefly outline the progress made in EPROM technology including a description of circuit configuration and notes on applications.

The Structure and Basic Operation of a Memory Transistor

As shown in Fig. 6.1, increasing EPROM capacity has been accompanied by changes in the memory transistor structure. The 2K-bit device made use of a P-channel MOS transistor to form an insulated single-layer polysilicon floating gate. In contrast to this, devices of 8K-bit capacity and greater make use of n-channel transistors and two-layer gate structure with a control gate to which a voltage may be applied placed over the floating gate. A capacitance between the control gate and the floating gate form an acceleration field for electron injection to the floating gate. Programming is performed in the following manner. For programming operations a high voltage is applied to the drain and control gate. By virtue of the control gate, capacitance between control gate and floating gate a channel is formed between the source and drain through which a current flows. As a result, for high drain voltages current induced breakdown occurs. The hot-electrons produced as a result of this breakdown phenomena exceed the high energy barrier and are injected into the floating gate. By imparting a voltage to these injected electrons the control gate can have higher threshold voltage than before injection (refer to Fig. 6.2), and the read voltage may be applied to the control gate while maintaining an open circuit. This ends the write operation. This applies to the memory transistors used in presently available EPROM devices of 8K-bit capacity and over. Fig. 6.3 shows the programming characteristics (dependency of the threshold value on the write pulse width) for 16K- and 32K-bit memory transistors.

The injected charge is located on the floating gate which is surrounded by a 1,000Å thick silicon oxide layer of good insulating characteristics, and is therefore retained for a long period. It is the retention of this charge which holds the written data. A significant feature of two-layer gate structure is the associated increase in density. As shown in Fig. 6.1, whereas in the single-layer gate an additional row selection transistor is required, the two-layer memory transistor eliminates this necessity by having the control gate serve two functions.

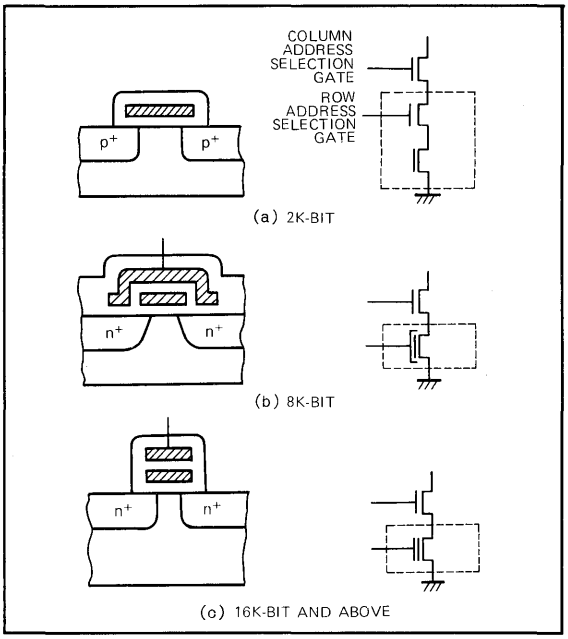


Fig. 6.1 Memory transistor construction

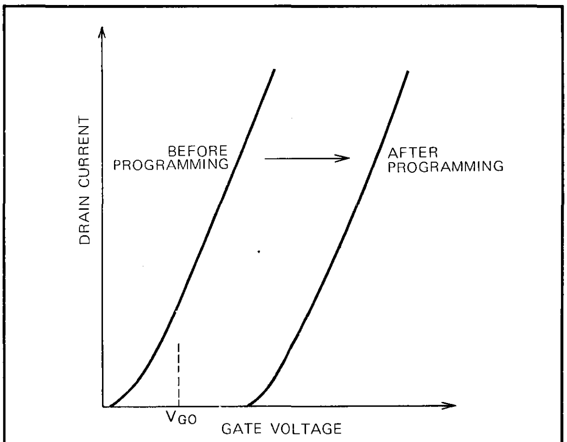


Fig. 6.2 Variation in memory transistor threshold voltage (V_{GO}: Read gate voltage, both vertical and horizontal scales are arbitrary)

The introduction of 8K- and 16K-bit devices and greater was accompanied by improvements of control gate structure. As shown in Fig. 6.1, whereas for the 8K-bit device the side of the folating gate is completely covered by the control gate, this is not true of devices of 16K-bit capacity and greater. It should be noted that while significant improvements in overall capacity has been made, chip size remains essentially unchanged, the 16K-bit chip size being merely 8.2% greater than that for the 8K-bit device.

Erasing is done by exposing the device to ultraviolet light. The electrons on the floating gate receive the ultraviolet energy, pass through the oxide layer and escape. The transmittivity of ultraviolet radiation from a low pressure mercury lamp through polysilicon is low compared to silicon oxide. For this reason, the ultraviolet energy reaching the floating gate of 16K-bit and greater memory devices using transistors without polysilicon sides is larger than the 8K-bit structure. This results in shorter erase times for 16K-bit devices and over. Fig. 6.4 illustrates the change in threshold value by exposure of ultraviolet energy.

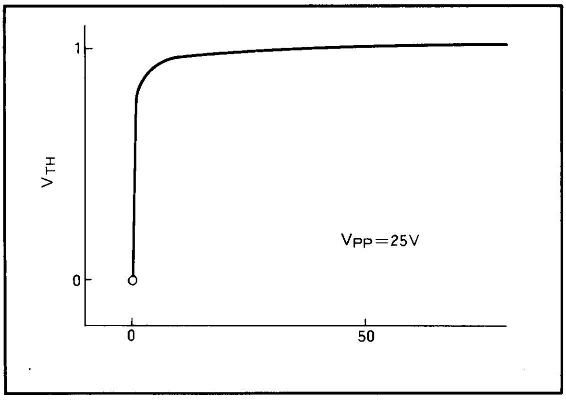


Fig. 6.3 Dependency of V_{TH} on write pulse width (16K-bit and 32K-bit)

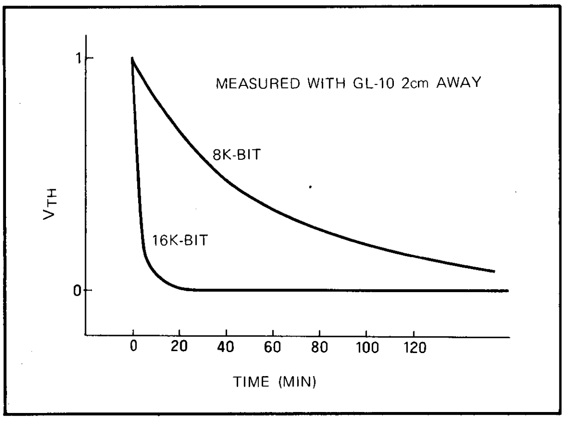


Fig. 6.4 Variation in V_{TH} with erasure time

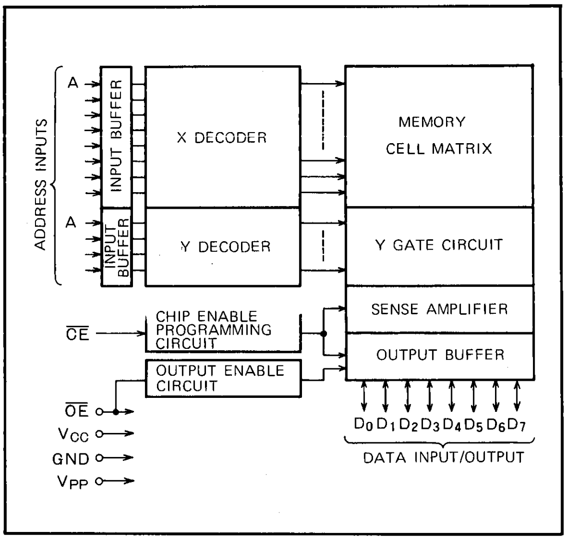


Fig. 6.5 EPROM Block diagram

EPROM Circuit Configuration and Characteristics

Circuit Configuration

Fig. 6.5 shows the block diagram of an ultraviolet light erasable EPROM. Currently available devices are configured in 8-bit words with the memory cells arranged in eight blocks. Input and output is performed in parallel by means of the signal lines $D_0 \sim D_7$ connected to these eight blocks. The address signals are divided into column decoder inputs and row decoder inputs. For a 32K-bit EPROM, the $A_0 \sim A_3$ (four lines) address signals are input to the column decoder while the $A_4 \sim A_{11}$ (eight lines) address signals are input to the row decoder, the memory being arranged as a matrix of 2^4 (=16) columns by 2^8 (=256) rows.

After decoding, the column signals are input to the column selection transistor gate which is connected to the memory cell drain. Finally, the decoder row inputs are connected to the memory control gates. Sense amplifiers and data input/output buffers used in read and program operations are connected to the drains (data lines) of the memory cells controlled by the column selector transistors. Almost all of the chip area is taken up by the memory cells, address circuits, decoders, and data circuits, the remaining area being allotted to the important control circuits.

These control circuits consist of the chip enable and output enable circuits. The former controls the power down operation or programming operations. The latter circuit controls the enabling or disabling of the output signal by means of the OEsignal. 16K-bit devices and over are provided with these two select/unselect control circuits. The two line control method is very effective for OR-connecting of multiple devices. If only one signal were allowed to control chip select and unselect, cases could arise where one chip is enabled for output before the previous chip goes into the floating state.

As shown in Fig. 6.6, this results in excessive current flowing and the generation of power supply noise. In addition, data on the bus is unstable before and after address changes. This condition is called "the bus contention problem" and can be eliminated by using the $\overline{CE}$ as the chip enable and $\overline{OE}$ as the output enable signal in a two-line control mode.

EPROM Operation, Characteristics, and Application Notes.
The basic operations possible with an EPROM are programming, read, and erase. These operations will be discussed with respect to 16K- and 32K-bit devices along with some precautions for use. Table 1 summarizes a comparison of the characteristics of EPROM devices currently available.

(1) Programming Operations

The normal state of all cells for an EPROM device when shipped or after erasure is "1", programming operations change the memory cell contents to 0. Programming operations are performed in groups of 8 bits (one word). After applying the programming voltage to the programming pin and selecting the program mode, the address data is set up. Next, a programming pulse of the required width is input. The active state of this pulse depends on the device (for instance, for 16K-bit devices the pulse is active high while for 32K-bit devices it is active low), so that care should be taken when generating this pulse. Although it is often thought that the higher the programming voltage and the wider the programming pulse, the more effective the programming operation will be, the device characteristics dictate that the best programming will be achieved by setting these values to the central specification values. In particular, the maximum allowable voltage for programming that may be applied to the Vpp pin is 26V. Care must be taken that the Vpp supply doesn't overshoot the 26-volt maximum specification. Programming for both 16K- and 32K-bit devices can be performed in any arbitrary order, further simplifying the programming operation.

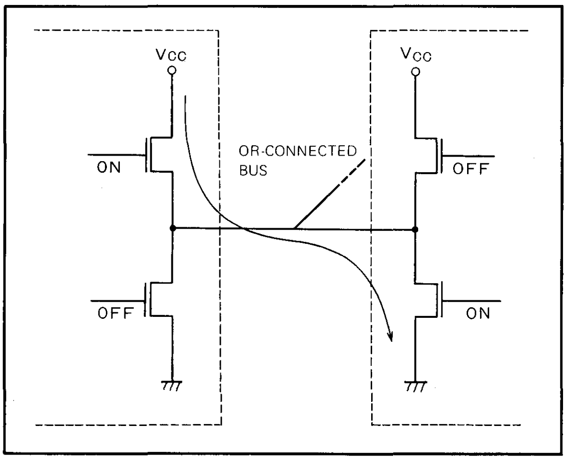


Fig. 6.6 Fighting for an OR-connected bus

Table 6.1 Comparison of Available EPROM Devices

MEMORY CAPACITY (BITS)	2K (256×8)	3K (1024×8)	16K (2048×8)	32K (4096×8)
TYPE	M5L1702AS	M5L2708K	M5L2716K	M5L2732K
CHANNEL TYPE	p	n	n	n
CHIP AREA	14.2mm^2	17.8mm^2	19.3mm^2	22.5mm^2
ADDRESS ACCESS TIME (MAX)	1000ns	450ns	450ns	450ns
POWER DISSIPATION (MAX)	600mW	800mW	525mW	787mW
POWER DISSIPATION PER BIT	0.3mW	0.1mW	0.03mW	0.02mW
SUPPLY VOLTAGES	+5, −9V	+5, −5, +12V	+5V	+5V

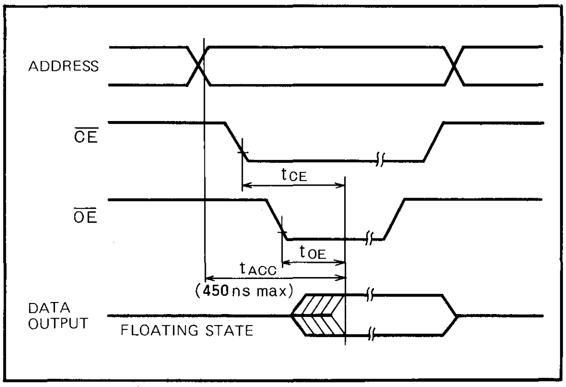


Fig. 6.7 Read timing diagram

(2) Read Operation

The read mode is enabled by lowering the program voltage and using the chip enable signal to select the chip, and the output control signal to enable the output of the memory contents at the selected address. The chip enable signal serves also as the power down signal, enabling an extreme limitation on power consumption for the non-selected periods. Access time is specified in terms of chip enable, address, and output enable access times, the power down feature making the chip enable access time generally the longest. Operating conditions and output timing should be carefully considered as high temperatures and excessive output loads have an adverse affect on access time. Fig. 6.7 shows the read timing for 16K-bit and 32K-bit devices with Fig. 6.8 and Fig. 6.9 giving the chip enable access time dependency on temperature and load capacitance.

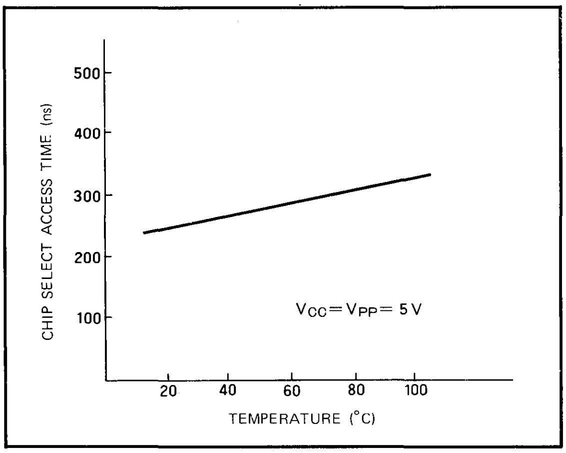


Fig. 6.8 2716 Chip select access time temperature characteristics example

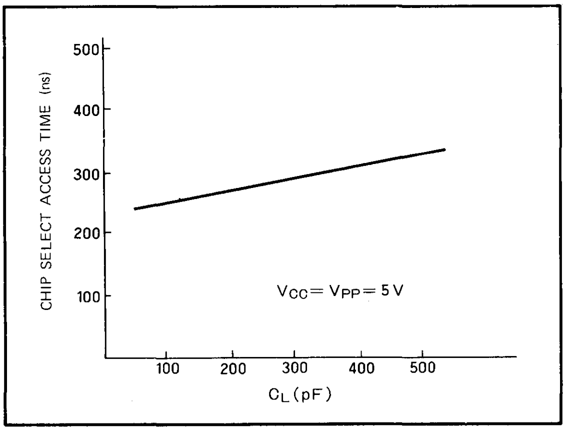


Fig. 6.9 2716 Chip select access time load capacitance dependency

(3) Erasure

Erasure is performed by exposing the chip to ultraviolet light. Fig. 6.4 shows the change in memory transistor threshold value with relationship to ultraviolet radiation. The erasure time should be selected to allow for variations in the memory transistor characteristics. Fig. 6.10 shows the relationship between the ultraviolet radiation time and the number of bits erased. Verification of erasure by means of a PROM should not be assumed to indicate that the EPROM is sufficiently erased. While the required erasure time depends upon factors such as the type and condition of the lamp used and the distance to the device being erased, the actual erasure procedure should be continued for a period of five times the time required to erase all cells as verified by a PROM programmer. Generally, for 16K- and 32K-bit EPROMs, the erasure time for a GL-10 lamp 2.5cm away from the device is between 15 and 20 minutes.

The erasure characteristics for 8K-bit EPROMs differs from those for 16K-bit and greater capacity for structural reasons, with the differences extending to the degree of influence of sunlight and fluorescent lighting on the inadvertent erasure of data. To prevent such long term ambient radiation from affecting electrical characteristics, the use of a seal to cut out such radiation for normal use is required.

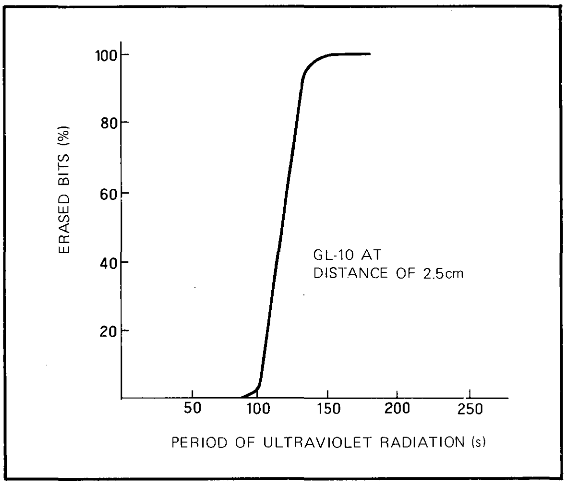


Fig. 6.10 Erasure characteristics example for 2716 and 2732

MITSUBISHI
ELECTRIC

6.2 APPLICATION OF EPROMS

EPROM control functions

EPROM control functions are provided to simplify interface and allow full utilization of performance. A new generation of dual-control function EPROMs has become popular which has both a chip enable ($\overline{CE}$) and an output enable ($\overline{OE}$) control input.

$\overline{CE}$ (Chip Enable, active low)

The falling edge of $\overline{CE}$ activates the address input buffers and latches the address in preparation for the address decoders and the sense amplifiers to perform their function. This acts also as a power control function, allowing the device to enter a low-power standby mode when the $\overline{CE}$ input is disabled.

$\overline{OE}$ (Output Enable, active low)

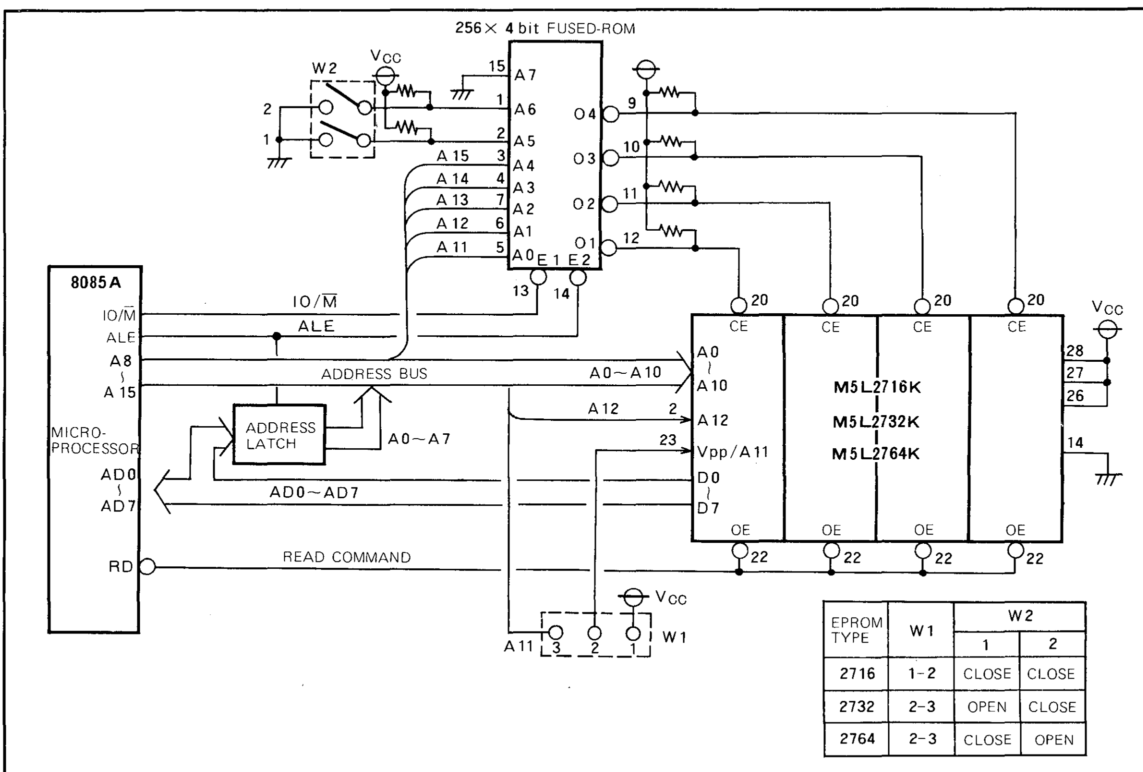
OE controls the device's output buffer, and is used to avoid bus contention since the device's output can be turned on and off directly by the processor. The $\overline{CE}$ and $\overline{OE}$ control functions are ANDed inside the device. This means that only the simultaneous application of $\overline{CE}$ and $\overline{OE}$ will activate the output of the device. (When either of $\overline{CE}$ or $\overline{OE}$ is not active, the output buffer of the device goes into the high-impedence state.)

Microprocessor interface

As described above, EPROMs can be interfaced easily to microprocessors using the $\overline{CE}$ and $\overline{OE}$ functions. A typical example is shown in Fig. 6.11. The address from the microprocessor is decoded by the bipolar PROM which generates the primary decoded signal $\overline{CE}$. Next, read memory command from the microprocessor enables $\overline{OE}$. This decoding method makes possible a substantial power saving.

EPROM package compatibility and design technique

As the density of EPROMs increase, more address pins will be needed for higher density devices. But the EPROM family has similar pin configurations maintain which keeps the compatibility with each memory size devices. The pinouts of the family are shown in Fig. 6.12 which have different signals at the dotted pinouts only. It may seem as though the 28-pin package is not compatible with the 24-pin devices, but the lower 24 pins are indential to the 24-pin package of 2716 or 2732 as shown in Fig. 6.12.



EPROM TYPE	W1	W2	
		1	2
2716	1-2	CLOSE	CLOSE
2732	2-3	OPEN	CLOSE
2764	2-3	CLOSE	OPEN

Fig. 6.11 Microprocessor-EPROM interface example

The pinouts of the EPROM family enable the memory design to support 2K-, 4K-, and 8K-byte EPROMs, which require some techniques of address decoding and print circuit board layout. Fig. 6.11 shows now the EPROM family may be connected to the very popular M5L8085A microprocessor. The high-order microprocessor address bits are fed to a 256x4 bipolar PROM for address spatial decoding. The PROM allows the address space to be re-defined at any time so that various EPROMs can be used. The jumpers W1 and W2 are used to define the type of EPROM according to the table in Fig. 6.11. The address map of the PROM is shown in Table 6.2.

2764	2732	2716	Pin	Pin	2716	2732	2764	
V_{PP}			1	28			V_{CC}	
A_{12}			2	27			$\overline{PGM}$	
A_7	A_7	A_7	3(1)	(24)26	V_{CC}	V_{CC}	NC	
A_6	A_6	A_6	4(2)	(23)25	A_8	A_8	A_8	
A_5	A_5	A_5	5(3)	(22)24	A_9	A_9	A_9	
A_4	A_4	A_4	6(4)	(21)23	V_{PP}	A_{11}	A_{11}	*
A_3	A_3	A_3	7(5)	(20)22	$\overline{OE}$	$\overline{OE}/V_{PP}$	$\overline{OE}$	*
A_2	A_2	A_2	8(6)	(19)21	A_{10}	A_{10}	A_{10}	
A_1	A_1	A_1	9(7)	(18)20	$\overline{CE}$	$\overline{CE}$	$\overline{CE}$	
A_0	A_0	A_0	10(8)	(17)19	D_7	D_7	D_7	
D_0	D_0	D_0	11(9)	(16)18	D_6	D_6	D_6	
D_1	D_1	D_1	12(10)	(15)17	D_5	D_5	D_5	
D_2	D_2	D_2	13(11)	(14)16	D_4	D_4	D_4	
GND	GND	GND	14(12)	(13)15	D_3	D_3	D_3	

Fig. 6.12 EPROM Family pinouts

Table 6.2 PROM Address Map

Input signal / Decoder address	GND	W2-2	W2-1	Microprocessor's address A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	Decoder outputs			
	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	O_1	O_2	O_3	O_4
2716 mode	0	0	0	△	△	△	0	0	0	1	1	1
							0	1	1	0	1	1
							1	0	1	1	0	1
							1	1	1	1	1	0
2732 mode	0	0	1	△	△	0	0	*	0	1	1	1
						0	1	*	1	0	1	1
						1	0	*	1	1	0	1
						1	1	*	1	1	1	0
2764 mode	0	1	0	△	0	0	*	*	0	1	1	1
					0	1	*	*	1	0	1	1
					1	0	*	*	1	1	0	1
					1	1	*	*	1	1	1	0
Not used	0 ↓ 1	1 ↓ 1	1 ↓ 1	* ↓ 1	* ↓ 1	* ↓ 1	* ↓ 1	* ↓ 1	1	1	1	1

Note: * indicates 0 or 1 △ indicates 0 or 1 which defines the EPROM's page address

MITSUBISHI ELECTRIC

Functional description of M5L8041A-006P

General

M5L8041A-006P is a slave computer LSI which is designed for EPROM writer control using a mask-programmed M5L8041A-XXXP. The operation mode of the PRPG is defined by the master microprocessor. So it is programmed by the system's software as an I/O peripheral.

Command description

There are 7 commands provided for programming the PRPG. These commands are sent on the data bus with the signal $\overline{CS}$ at low and the signal A_0 at high and are stored in the PRPG at the rising edge of the signal $\overline{WR}$.

The summary of PRPG's commands and status is shown in Table 6.3.

PRPG Timing and interfacing

PRPG's operation timing are triggored by the commands are shown Table 6.3. There are two operation modes, 2716 mode and 2732 mode, whose timing are shown in Fig. 6.13 and Fig. 6.14.

Application for EPROM writer

Introduction

M5L8041A-006P is one of the applications for EPROM writer controller which can interface to microprocessors (e.g. 8080A, 8085A, 8086). EPROM writer design is simplified by using M5L8041A-006P.

Features of the M5L8041A-006P;

- EPROM write controll for the 2716 or 2732
- Fully compatible with Mitsubishi microprocessors
- Reduces the master microprocessor's program for EPROM writing.

PRPG interface and timing

An example of PRPG interfacing is shown in Fig. 6.15. Using the PRPG, the design of the EPROM writer is simplified. M5L8243P is used for the port expander of PRPG.

PRPG's operational timing is managed by the commands shown in Table. 6.3. There are two operation modes (i.e., 2716 and 2732 mode) whose timing is shown in Fig. 6.13 and Fig. 6.14 respectively. If the mode set command is not equal to the hardware switch to select 2716 or 2732 in the Fig. 6.15, the mode will not be set and the FAIL LED will light.

Design example of EPROM/RAM board

Fig. 6.16 presents the design example of EPROM/RAM board which is fully compatible with the proposed IE^3-P796 bus standard. The M5L2716K, M5L2732K or M5L2764K can be used in this board, and also 2Kx8 bit of RAM (M58725 P) can be mixed with M5L2716K.

Table 6.3 M5L8041A-006P (PRPG) Function Table

	COMMAND/ STATUS	DESCRIPTION	CODE	LED DISPLAY PASS	LED DISPLAY FAIL	NOTES
COMMAND	1 Mode set	Defines the operation mode of PRPG	MSB … LSB: M 0 0 0 0 0 F1 F0 ; A0: 1 M → EPROM Selection: 0 = 2716 Mode, 1 = 2732 Mode F1 F0 → Selection of clock input: 0 0 = 3.58 MHz, 0 1 = 4.194303 MHz, 1 0 = 6 MHz	PASS = ON When mode is set correctly. (Mode request LED turns off.)	FAIL = ON When programmed mode is not coincident with hardware switch.	It is necessary to provide mode set command after power on. After mode is set, the operation mode is available until the other mode is set.
COMMAND	2 Address set	Defines the start address (SPA) and the end address (EPA) of programing EPROM	MSB … LSB: 1 1 1 1 0 0 0 0 ; A0: 1 After providing Address set command, it is necessary to provide following 4 bytes of SPA and EPA continually. 1st byte: (SPA) HIGHER ; A0: 0 2nd byte: (SPA) LOWER ; A0: 0 3rd byte: (EPA) HIGHER ; A0: 0 4th byte: (EPA) LOWER ; A0: 0		FAIL = ON 1 SPA > SPB 2 SPA EPA 2716 ≧ 800_{16} ≧ 800_{16} 2732 ≧ 1000_{16} ≧ 1000_{16}	A bytes commands after address set command should provide with A0 input low-level.
COMMAND	3 Blank check	Checks the programming EPROM if it is erased.	MSB … LSB: 1 0 1 1 0 0 0 0 ; A0: 1	PASS = ON If EPROM is erased. (i.e. All data are FF_{16})	FAIL = ON If EPROM is not erased.	
COMMAND	4 EPROM write	Writes the data to the EPROM which are sent from master CPU and rerifies the written data if they are correctly programmed.	MSB … LSB: 1 0 1 0 0 0 0 0 ; A0: 1 After providing this command, PRPG generates EPROM programming timing pulse.	PASS = ON When programming finishes completely.	FAIL = ON If it is not able to write correctly in the EPROM address. (SPA ~ EPA)	
COMMAND	5 Verify	Checks the programmed EPROM if it is written correctly.	MSB … LSB: 1 1 0 1 0 0 0 0 ; A0: 1 After providing this command, PRPG compares the programed data with the source data of master CPU.	PASS = ON When the programed data are equal to the source data.	FAIL = ON If the programed data are not equal to the source data	
COMMAND	6 Automatic write	Executes the commands 3 , 4 , 5 automatically.	MSB … LSB: 1 1 0 0 0 0 0 0 ; A0: 1	PASS = ON When there is no error in 3 and 5 command.	FAIL = ON When there are any errors in 3 and 5 command.	
COMMAND	7 Copy	Transfers the EPROM's data where are difined SCA and ECA to the master CPU.	MSB … LSB: 1 1 1 0 0 0 0 0 ; A0: 1 After providing Copy command, it is necessary to provide following 4 bytes of SCA and ECA. 1st byte: (SCA) HIGHER ; A0: 0 2nd byte: (SCA) LOWER ; A0: 0 3rd byte: (ECA) HIGHER ; A0: 0 4th byte: (ECA) LOWER ; A0: 0	PASS = ON When data transfer finishes.	FAIL = ON Same condition as command 2	SCA = start address of copy data ECA = start address of copy data
STATUS	8 Write data buffer full	Indicates if write data buffer in the PRPG are full or not.	MSB … LSB: – – – – – – F – ; A0: 1 F: 0 PRPG data buffers: full 1 PRPG data buffers: not full			The status is used by 4 or 6 command.

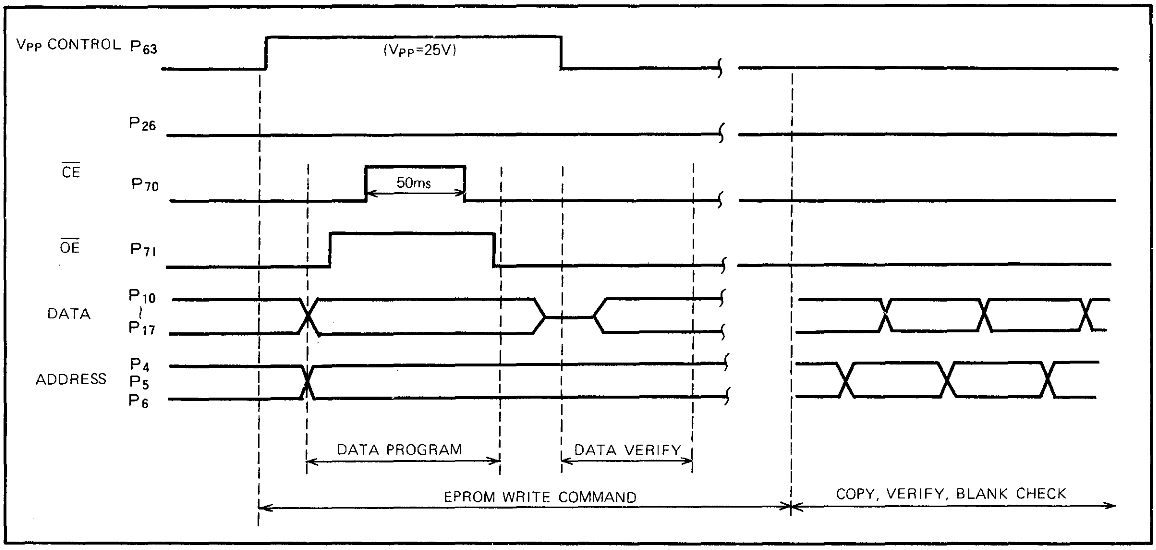


Fig. 6.13 2716 Programming timing

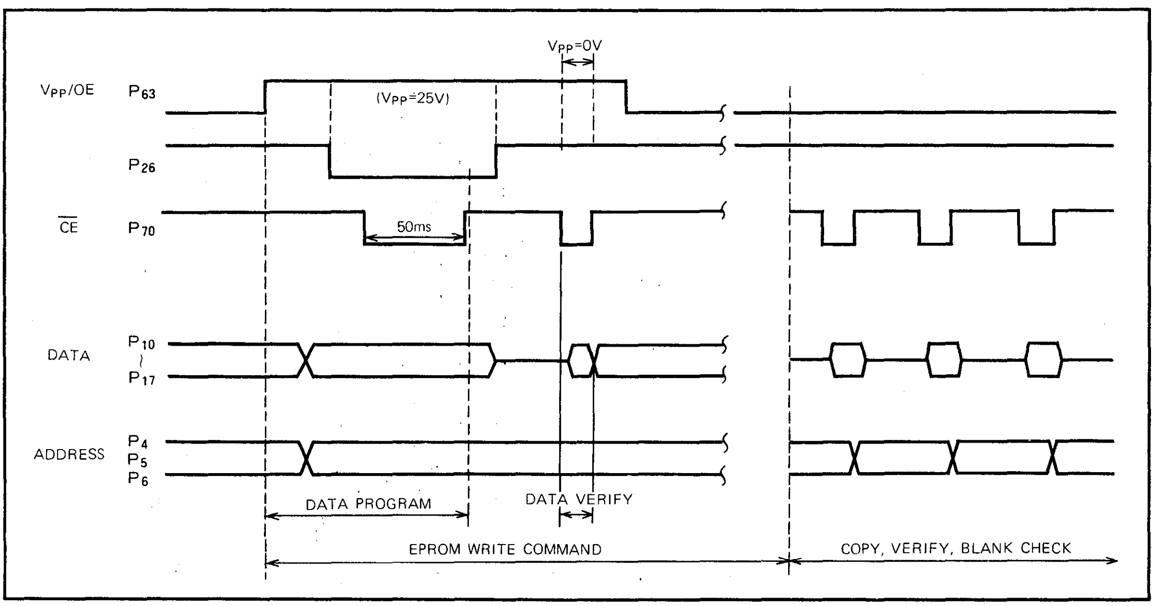
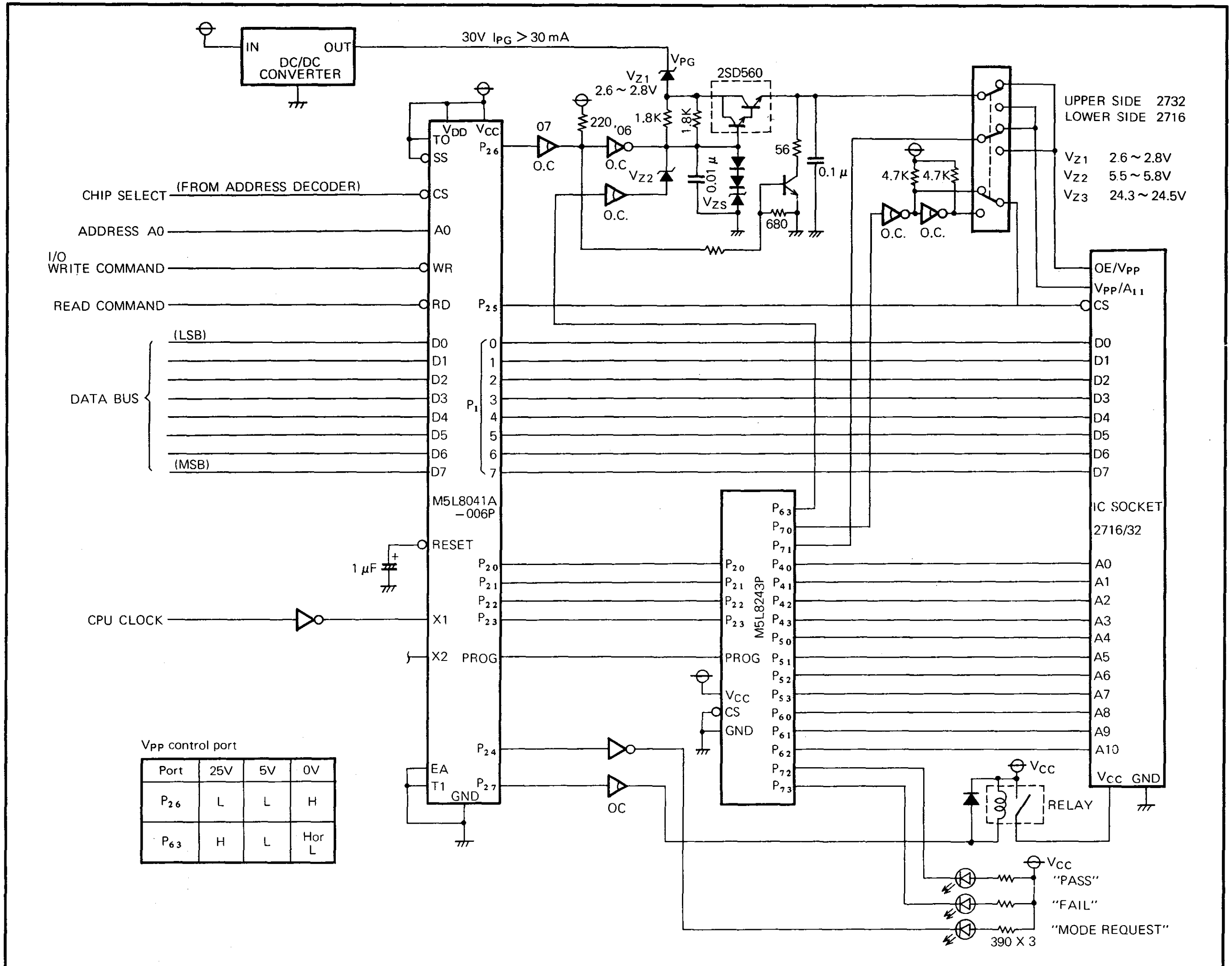


Fig. 6.14 2732 Programming timing



V_{PP} control port

Port	25V	5V	0V
P_{26}	L	L	H
P_{63}	H	L	H or L

Fig. 6.15 Design example of PRPG.

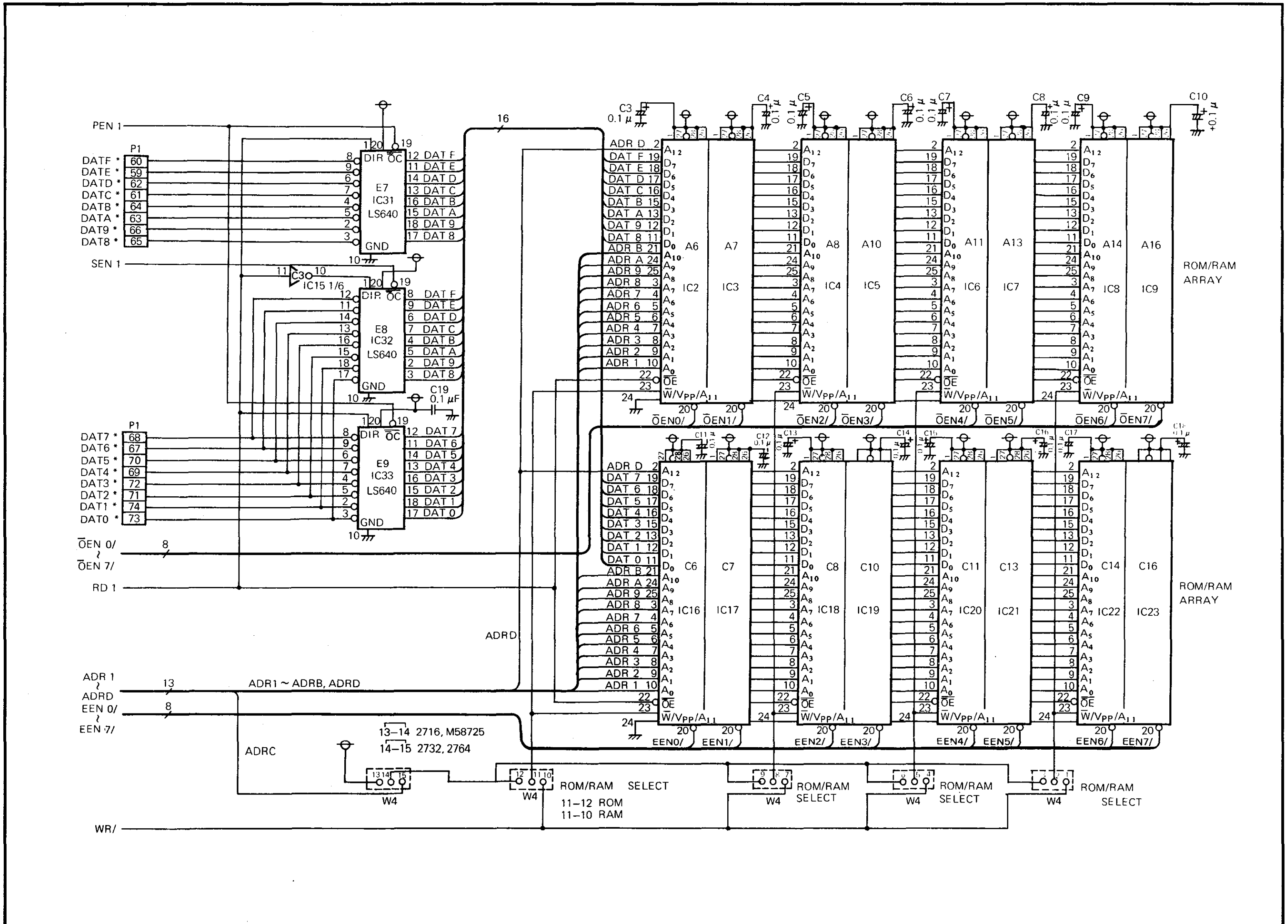
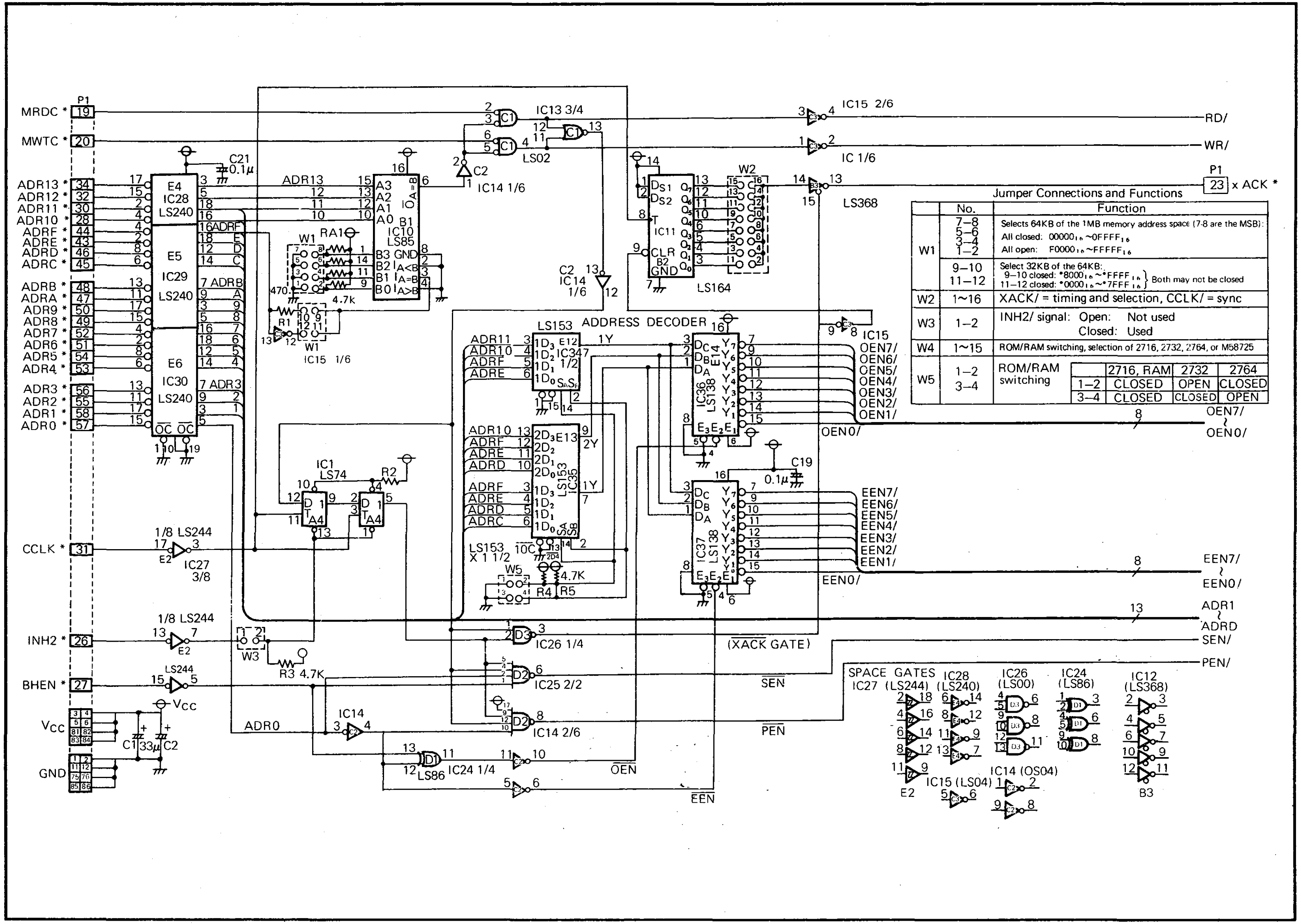


Fig. 6.16 Design example of EPROM/RAM board (1/2)



Jumper Connections and Functions

	No.	Function
W1	7–8 5–6 3–4 1–2	Selects 64KB of the 1MB memory address space (7-8 are the MSB): All closed: 00000_{16}~$0FFFF_{16}$ All open: $F0000_{16}$~$FFFFF_{16}$
	9–10 11–12	Select 32KB of the 64KB: 9–10 closed: $*8000_{16}$~$*FFFF_{16}$ 11–12 closed: $*0000_{16}$~$*7FFF_{16}$ Both may not be closed
W2	1~16	XACK/ = timing and selection, CCLK/ = sync
W3	1–2	INH2/ signal: Open: Not used Closed: Used
W4	1~15	ROM/RAM switching, selection of 2716, 2732, 2764, or M58725
W5	1–2 3–4	ROM/RAM switching (see below)

W5	2716, RAM	2732	2764
1–2	CLOSED	OPEN	CLOSED
3–4	CLOSED	CLOSED	OPEN

Fig. 6.16 Design example of EPROM/RAM board (2/2)

7. Error Detecting and Correcting

1. Introduction

The reliability of semiconductor memory devices is extremely high, so that for normal operation performance can be guaranteed without the use of special correcting circuits. However, for applications requiring even greater reliability, or in which the large number of ICs used in the system causes a reduction of the overall MTBF, some form of data error correction is requried. This section will examine errors in memory devices and techniques for detecting and correcting them.

2. Error Checking Concept and the Redundancy Code

A code which can be used for error checking consists of the following code of added bits to the normal data word.

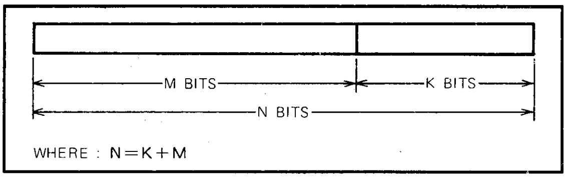


Fig. 7.1 Error check code format

The first M bits represent data, and will be referred to as the data word. The remaining K bits are not directly related to data and are called redundancy bits or the redundancy word. The combination of these two portions form a N bits total word. Such a combination of a data word and a redundancy word is known as a redundancy code.

As an example, let us take the case of M=2 and K=1. As shown in Table 7.1, there are four possible combinations of data word.

Table 7.1 Redundancy Code Example

DW		RW
D_1	D_0	R_0
0	0	0
0	1	1
1	0	1
1	1	0

DW : data word
RW : redandancy word

To this data word is added a redundancy word as shown in Table 7.1, the redundancy bit R_0 being given by the following expression.

$$R_0 = D_0 \oplus D_2 \quad \text{......................} (1)$$

where $\oplus$ is the exclusive-OR operator

Next, let us consider the case for which one of the bits is different, that is, the case of a single bit error. Since the word has three bits, for each word there are three possible error words making a total of 12 possible error words. This is summarized in Table 7.2.

Table 7.2 Valid Word and Error Word Combinations

Valid word	Error words		
0 0 0	0 0 1	0 1 0	1 0 0
0 1 1	0 1 0	0 0 1	1 1 1
1 0 1	1 0 0	1 1 1	0 0 1
1 1 0	1 1 1	1 0 0	0 1 0

Table 7.3 Decimal Notation Valid and Error Words

Valid word	Error words		
0	1	2	4
3	2	1	7
5	4	7	1
6	7	4	2

A valid word is one with no errors.

From Table 7.2 and Table 7.3 we see the following

(1) The set of valid words contains no error words.

(2) Any valid word differs from other valid words by not less than two bits.

Basically the above two statements are the same. Specifically, valid words differ from each other in not less than two bits so that single bit errors are not found in valid data. In this manner error check is to detect a (h-1) bit error by making valid words differ from each other by h bits to select proper redundancy words. (Seleting thus the set of valid words which does not include error words differing not less than (h-1) bits, so we can distinguish these error words from valid words.) The value h for such an error check code is known as the Hamming distance. When this relationship is expressed in set theory notation we have the representation of Fig. 7.2.

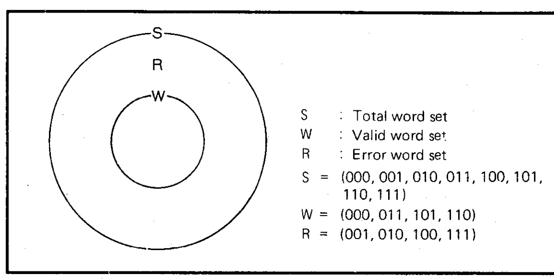


Fig. 7.2 Parity codes classification

Next let us examine the error word. For the error words shown in Table 7.3, the difference from the valid word is in order across the row the first bit, the second bit, and finally third bit. While the same word exists for the error words of any particular row, it is impossible to know which row a particular error word belongs to or phrased differently, although the fact that the word is an error word is determinable, which bit is in error must be known or else the code cannot be used to correct the error. This type of error check is known as error detection.

As the next example, let us take the case where M=1 and K=2. For this example the data words are shown in Table 7.4 and it can be seen that there are only two types of data words with the redundancy word determined by equation (2).

Table 7.4 Error Correcting Code Example

DW	RW
D_0	K_1 K_0
0	0 0
1	1 1

$$K_0 = K_1 = D_0 \text{ .... } (2)$$

If as in the previous example, we consider a single bit error word, we can derive Table 7.5 and Table 7.6 as shown below.

Table 7.5 Single Bit Error Word Combinations

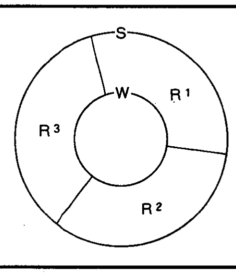
Valid word	Error words		
0 0 0	0 0 1	0 1 0	1 0 0
1 1 1	1 1 0	1 0 1	0 1 1

Table 7.6 Decimal Notation Single Bit Error Words

Valid word	Error words		
0	1	2	4
7	6	5	3

As can be seen from the table, this time there is no ambiguity in the error words. Therefore, the valid word can be obtained merely by looking at the error word. For example, if the word were 001, 010, or 100, the valid data would be clearly 000. However, if the error word were 110, 101, or 011, the original valid word would have been 111. Approaching this differently, let us assume that the word is 001 or 110. Since the error word has the K_0 bit different, reversing the differing bit turns 001 into 000 and 110 into 111 which are the valid data words. The same procedure applies to the other error words which are possible.

In this manner, a redundancy word is used to eliminate ambiguities in the error word and enable the derivation of the valid word by examining the error word. This procedure is known as error correction. Expressed in terms of set theory symbols we have the relationships shown in Fig.7.3



S : total word set
W : valid word set
R^1 : bit 1 error word set
R^2 : bit 2 error word set
R^3 : bit 3 error word set
W = (000, 111)
R^1 = (001, 110)
R^2 = (010, 101)
R^3 = (100, 011)

Fig. 7.3 Error correction code relationships

An actual circuit implementation of such an error correction code would involve determining whether the word belonged to W, R^1, R^2, or R^3. If the word was found to belong to the R set, the error correction procedure involves simply inverting the bit corresponding to the error word set.

3, Parity

As discussed in the previous section, some error check techniques involve error words which have duplication and can only be used to detect errors. One particular type of error detection scheme used to detect single bit errors is called parity. The parity error detection method uses a redundancy word one bit long (R_0) regardless of the length of the data word. R_0 is derived in the following manner.

$$R_0 = D_0 \oplus D_1 \oplus D_2 \oplus \ \dots \oplus D_{m-1} \text{ .... } (3)$$

or

$$R_0 = \overline{D_0 \oplus D_1 \oplus D_2 \oplus \ \dots \oplus D_{m-1}} \text{ .... } (4)$$

The single bit derived in equation (3) represents the number of 1's in the total word is even and is termed even parity while equation (4) is referred to as odd parity.

Fig. 7.4 shows the example of an 8-bit data word, using an M74LS280P (SN74LS280) as the parity generator. The M74LS280P has Σ_E and Σ_O as parity outputs and inputs A through I. The input to output relationship being defined by the following expressions.

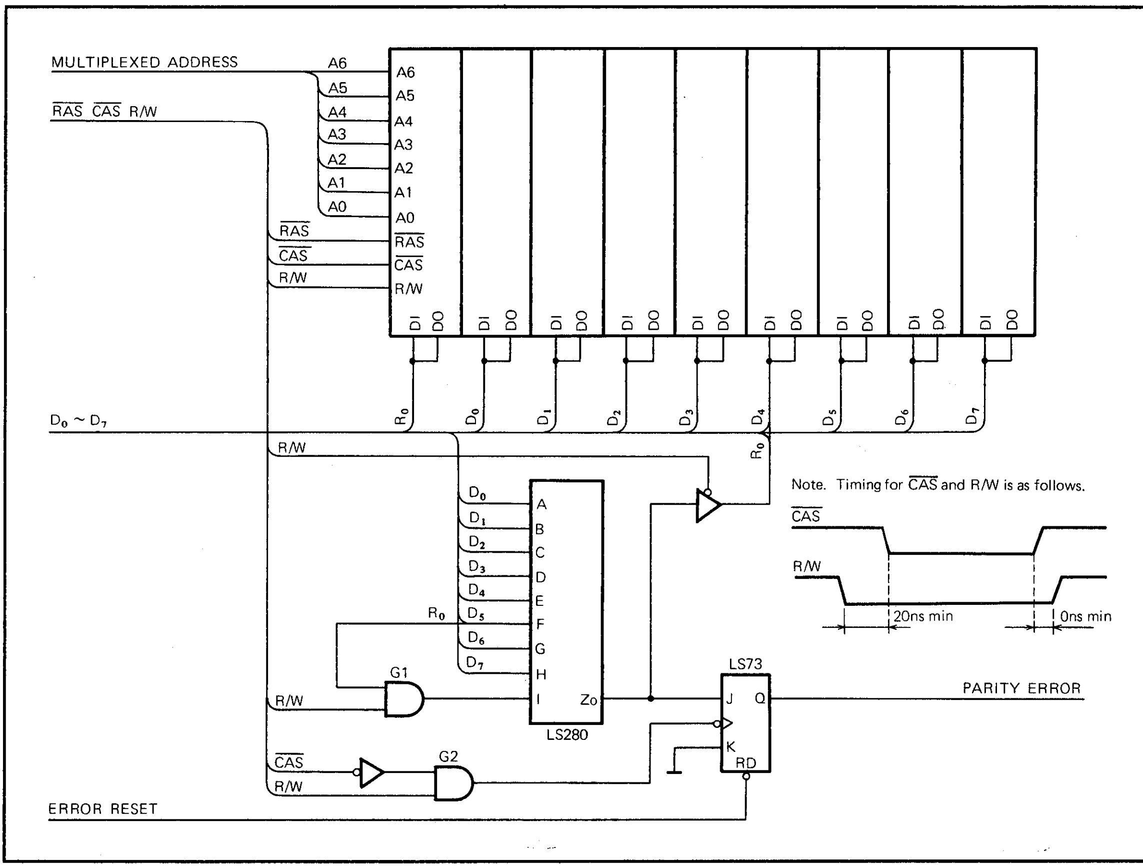


Fig. 7.4 Parity circuit example

$$\Sigma_E = \overline{A \oplus B \oplus C \oplus D \oplus E \oplus F \oplus G \oplus H \oplus I} \quad \text{..... (5)}$$

$$\Sigma_O = A \oplus B \oplus C \oplus D \oplus E \oplus F \oplus G \oplus H \oplus I \quad \text{..... (6)}$$

For write operations, since G1 makes the I input 0, the parity bit R_{OR} is written into memory with a value given by the following expression.

$$R_{OW} = \Sigma_{OW} = D_{OW} \oplus D_{1W} \oplus \cdots\cdots \oplus D_{1W} \quad \text{..... (7)}$$

where $D_{OW} \sim D_{7W}$ is the write data

Next, for read operations, all of the read data (including parity) is input to the M74LS280P. Therefore the output is given by the following expression.

$$\Sigma_{OR} = D_{OR} \oplus D_{1R} \oplus \cdots\cdots \oplus D_{7R} \oplus R_{OR} \quad \text{..... (8)}$$

If no errors are present in $D_{0r} \sim D_{7R}$, then they become equal to $D_{0w} \sim D_{7W}$ and R_{OW}, equation (8) replacing (7).

$$\Sigma_{OR} = R_{OR} + R_{OR} = 0 \quad \text{..... (9)}$$

The JK flip-flop does not change states at the rising edge of CAS. If, however, there is any single bit error in $D_{OR} \sim D_{7R}$ or R_{OR}, the following expression is derived from equation (7).

$$R_{OR} = \overline{D_{OR} \oplus D_{1R} \oplus \cdots\cdots \oplus D_{7R}} \quad \text{..... (10)}$$

When this is substituted for equation (8), we have the following.

$$\Sigma_{OR} = \overline{R}_{OR} + R_{OR} = 1 \quad \text{..... (11)}$$

Upon the rising edge of $\overline{\text{CAS}}$, the JK flip-flop output goes to 1 indicating the presence of an error.

4. Single-Bit Error Correction

The detection and reversal (correction) of a single bit error is referred to as single-bit error correction (SEC). For SEC to be possible the Hamming distance for the total word must be a minimum of 3-bits and the following equation must be satisfied. Examples of allowable values of M are given in Table 7.7.

$$2^K - 1 \geqq N$$
$$\text{where } N = M + K \qquad \text{.....(12)}$$

Table 7.7 Allowable M values

K	≦ M ≦
4	4～11
5	12～26
6	27～57
7	58～120
8	121～245

SEC is an extension of the parity concept. For instance, let us take the case of M = 16. From the Table we see that K = 5 from which N = 16 + 5 = 21. In single-bit error correction, each bit of the redundancy word is given a check bit weight, from which the location of the error bit can be determined. To implement this scheme a 21-bit total word is generated.

Bit number		21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Data word		D_{15}	D_{14}	D_{13}	D_{12}	D_{11}		D_{10}	D_9	D_8	D_7	D_6	D_5	D_4		D_3	D_2	D_1		D_0		
Redundancy word							R_4								R_3				R_2		R_1	R_0
Weight of redundancy bit	R_0	×		×		×		×		×		×		×		×		×		×		×
	R_1			×	×			×	×			×	×			×	×			×	×	
	R_2	×	×					×	×	×	×					×	×	×	×			
	R_3							×	×	×	×	×	×	×	×							
	R_4	×	×	×	×	×	×															

Fig. 7.5 SEC Weighting

Whereas for parity the R_0 has the weight for all bits, SEC, as shown in Fig. 7.5, assigns weights (2 bits) R_0～R_4 in a binary coded fashion. Specifically the weights are given in equations (13) through (17) below.

$$R_0 = D_0 \oplus D_1 \oplus D_3 \oplus D_4 \oplus D_6 \oplus D_8 \oplus D_{10} \oplus D_{11} \oplus D_{13} \oplus D_{13} \oplus D_{15} \qquad \text{..... (13)}$$

$$R_1 = D_0 \oplus D_2 \oplus D_3 \oplus D_5 \oplus D_6 \oplus D_9 \oplus D_{10} \oplus D_{12} \oplus D_{13} \qquad \text{..... (14)}$$

$$R_2 = D_1 \oplus D_2 \oplus D_3 \oplus D_7 \oplus D_8 \oplus D_9 \oplus D_{10} \oplus D_{14} \oplus D_{15} \qquad \text{..... (15)}$$

$$R_3 = D_4 \oplus D_5 \oplus D_6 \oplus D_7 \oplus D_8 \oplus D_9 \oplus D_{10} \qquad \text{..... (16)}$$

$$R_4 = D_{11} \oplus D_{12} \oplus D_{13} \oplus D_{14} \oplus D_{15} \qquad \text{..... (17)}$$

In this manner the redundancy bits R_0 through R_4 are generated and written into memory along with the data bits D_0 to D_{15}. For read operations, Σ_{0R0}～Σ_{0R4} are examined as was explained for parity checking. Should there be no errors, $\Sigma_{0R0} = \Sigma_{0R1} = \Sigma_{0R2} = \Sigma_{0R3} = \Sigma_{0R4} = 0$. Let us assume that a single-bit error occurs, for instance at the 11th bit (D_{W6}). R_0, R_1, and R_3 are used as check bits for the 11th bit so that from equation (13) through (17) we have $\Sigma_{0R0} = \Sigma_{0R1} = \Sigma_{0R3} = 1$ and $\Sigma_{0R2} = \Sigma_{0R4} = 0$.

Σ_{0R0}	Σ_{0R1}	Σ_{0R2}	Σ_{0R3}	Σ_{0R4}	
0	1	0	1	1	→ decimal 11

This indicates the presence of an error at the 11th bit. Therefore, the error can be corrected by merely reversing this bit (to 1 if it were 0 and 0 if it were 1) to obtain correct data. In a similar manner, if an error occurs at other bits the position will be indicated in binary code by Σ_{0R0}～Σ_{0R4}, allowing reversal of the bit after decoding of the position. These Σ_{0R0} through Σ_{0R4} are known as the syndromes.

While we have examined single-bit error correction, for a variety of reasons actual systems make use of double error detection and single-bit error correction.

5. Single-Bit Error Correction/Double-Bit Error Detection
The correction of 1-bit errors and detection of 2-bit errors is known as single-bit error correction/double-bit error detection (SEC-DED). To implement such a SEC-DED scheme, the total word Hamming distance must be at least 4-bits and the following equation must be satisfied.
Table 7.8 gives examples of K and M values.

$$2^{K-1} - 1 \geqq N$$
where $N = M + K$ ..... (18)

Table 7.8 Allowable M values

K	≦ M ≦
4	1 ~ 3
5	4 ~ 10
6	11 ~ 25
7	26 ~ 56
8	57 ~ 119

In SEC, weights are assigned so that the syndrome bits $R_0 \sim R_{K-1}$ are expressed in binary code to indicate directly the position of the error bit. In SEC-DED however, for a number of reasons, the weighting is assigned as shown in Fig. 7.6 For this reason, decoding of the syndrome is required for error position determination.

Bit number		22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Data word		D_{15}	D_{14}	D_{13}	D_{12}	D_{11}	D_{10}	D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0						
Redundancy word																		R_0	R_1	R_2	R_3	R_4	R_5
Weight of redundancy bit	R_0	×				×	×	×		×		×			×		×	×					
	R_1		×	×				×	×		×		×			×	×		×				
	R_2	×	×		×		×		×					×	×	×				×			
	R_3	×	×	×	×	×				×		×	×	×							×		
	R_4			×	×	×	×	×	×	×	×											×	
	R_5									×	×	×	×	×	×	×	×						×

Fig. 7.6 SEC-DED Weighting

$$R_0 = D_0 \oplus D_2 \oplus D_5 \oplus D_7 \oplus D_9 \oplus D_{10} \oplus D_{11} \oplus D_{15} \quad ...(19)$$
$$R_1 = D_0 \oplus D_1 \oplus D_4 \oplus D_6 \oplus D_8 \oplus D_9 \oplus D_{13} \oplus D_{14} \quad ...(20)$$
$$R_2 = D_1 \oplus D_2 \oplus D_3 \oplus D_8 \oplus D_{10} \oplus D_{12} \oplus D_{14} \oplus D_{15} ...(21)$$
$$R_3 = D_3 \oplus D_4 \oplus D_5 \oplus D_{11} \oplus D_{12} \oplus D_{13} \oplus D_{14} \oplus D_{15} ..(22)$$
$$R_4 = D_6 \oplus D_7 \oplus D_8 \oplus D_9 \oplus D_{10} \oplus D_{11} \oplus D_{12} \oplus D_{13} ...(23)$$
$$R_5 = D_0 \oplus D_1 \oplus D_2 \oplus D_3 \oplus D_4 \oplus D_5 \oplus D_6 \oplus D_7 \quad ...(24)$$

From equations (19) through (24) we have the following expression.

$$C_D = R_0 \oplus R_1 \oplus R_2 \oplus R_3 \oplus R_4 \oplus R_5$$
$$= D_0 \oplus D_1 \oplus D_2 \oplus D_3 \oplus D_4 \oplus D_5 \oplus D_6 \oplus D_7 \oplus D_8$$
$$\oplus D_9 \oplus D_{10} \oplus D_{11} \oplus D_{12} \oplus D_{13} \oplus D_{14} \oplus D_{15} \quad ......(25)$$

The righthand side of equation (25) is the same as the righthand side of the parity expression. Therefore, for read operations, examination of C_{DR} is given by the following.

$$C_{DR} = \Sigma_{OR0} \oplus \Sigma_{OR1} \oplus \Sigma_{OR2} \oplus \Sigma_{OR3} \oplus \Sigma_{OR4} \oplus \Sigma_{OR5}$$

Examination of this value indicates whether or not a 2-bit error is present (the parity can only indicate whether the number of error bits is odd or even. Therefore it is ineffective for more than 2-bits of error. In contrast to this, the SEC-DED scheme provides a no error/1-bit error syndrome detection capability which detects an even number of error bits as a 2-bit error).

Table 7.9 Summary of error types

Σ_{OR5}	Σ_{04}	Σ_{03}	Σ_{OR2}	Σ_{OR1}	Σ_{OR0}	C_{DR}	Remark
1	0	0	0	0	0	0	No error
0	1	0	0	0	0	1	R_{R5} error
0	0	0	0	0	0	1	R_{R4} error
0	0	1	0	0	0	1	R_{R3} error
0	0	0	1	0	0	1	R_{R2} error
0	0	0	0	1	0	1	R_{R1} error
0	0	0	0	0	1	1	R_{R0} error
1	0	0	0	1	1	1	D_{R0} error
1	0	0	1	1	0	1	D_{R1} error
1	0	0	1	0	1	1	D_{R2} error
1	0	1	1	0	0	1	D_{R3} error
1	0	1	0	1	0	1	D_{R4} error
1	0	1	0	0	1	1	D_{R5} error
1	1	0	0	1	0	1	D_{R6} error
1	1	0	0	0	1	1	D_{R7} error
0	1	0	1	1	0	1	D_{R8} error
0	1	0	0	1	1	1	D_{R9} error
0	1	0	1	0	1	1	D_{R10} error
0	1	1	0	0	1	1	D_{R11} error
0	1	1	1	0	0	1	D_{R12} error
0	1	1	0	1	0	1	D_{R13} error
0	0	1	1	1	0	1	D_{R14} error
0	0	1	1	0	1	1	D_{R15} error
Varies depending upon the error bit (all zeroes cannot occur)						0	2 bits error

Table 7.9 shows a summary of read error detection for various types of errors.

Table 7.9 applies to errors of 2 bits or less.

Fig. 7.7 is the circuit implementation for the expressions of equation (19) through (25). In actual systems, the detection of an error results in data being latched and corrected, whereupon it is rewritten into memory and the control circuits for these operations would be required as well.

The decoding of Σ_{ORO} through Σ_{OR5} by the SN74S138 and SN74S02 is shown in the following Table 7.10.

Table 7.10 Σ_{OR0} ~ Σ_{OR5} Decoding

Σ_{OR5}	Σ_{OR4}	Σ_{OR3}	Σ_{OR2}	Σ_{OR1}	Σ_{OR0}	Σ_{OR3} ~ Σ_{OR5} Octal	Σ_{OR0} ~ Σ_{OR2} Octal	Data word
1	0	0	0	1	1	4	3	D_0
1	0	0	1	1	0	4	6	D_1
1	0	0	1	0	1	4	5	D_2
1	0	1	1	0	0	5	4	D_3
1	0	1	0	1	0	5	2	D_4
1	0	1	0	0	1	5	1	D_5
1	1	0	0	1	0	6	2	D_6
1	1	0	0	0	1	6	1	D_7
0	1	0	1	1	0	2	6	D_8
0	1	0	0	1	1	2	3	D_9
0	1	0	1	0	1	2	5	D_{10}
0	1	1	0	0	1	3	1	D_{11}
0	1	1	1	0	0	3	4	D_{12}
0	1	1	0	1	0	3	2	D_{13}
0	0	1	1	1	0	1	6	D_{14}
0	0	1	1	0	1	1	5	D_{15}

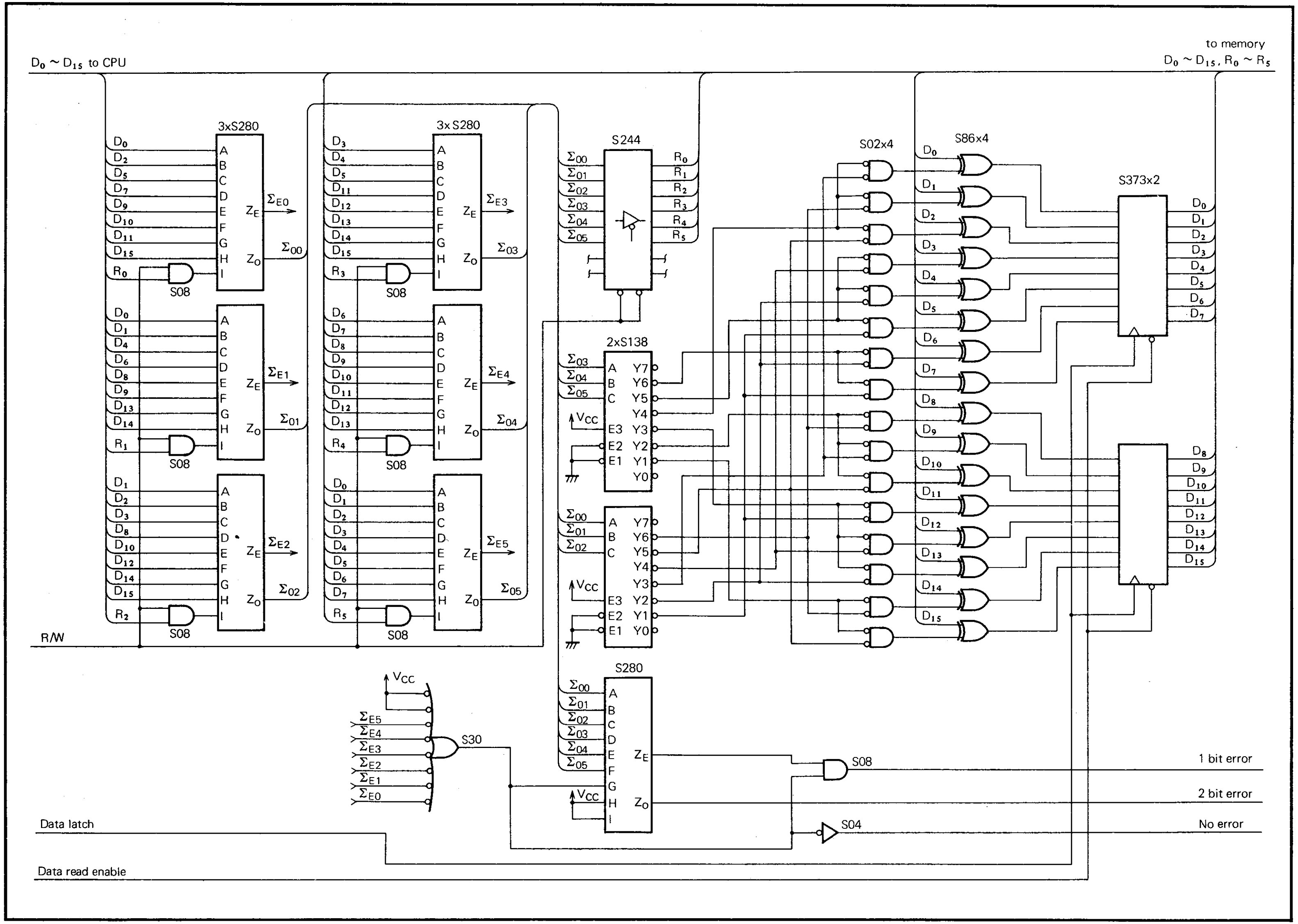


Fig. 7.7 SEC-DED circuit example

DESCRIPTION

Examples of subroutines for the MELPS 4 single-chip 4-bit microcomputer are described below. The subroutine calling sequence is also explained.

Subroutine	Mnemonic	Program list reference
● A-D conversion by successive approximation.	ADC1	Fig. 4
● A-D conversion by sequential comparisons.	ADC2	Fig. 5
● Clear file.	CF,CFM	Fig. 11
● Right-shift file.	RSF	Fig. 11
● Left-shift file.	LSF	Fig. 11
● Transfer of file.	TF	Fig. 12
● Exchange of file.	EXF	Fig. 13
● Increment memory.	INM	Fig. 13
● Decrement memory.	DEM	Fig. 13
● Skip non-zero memory.	SNM	Fig. 13
● Skip non-zero file.	SNFMA,SNFMI	Fig. 16
● BCD addition of files.	ADF	Fig. 17
● BCD subtraction of file.	SBF	Fig. 17
● Sign change of file.	SCF	Fig. 17

1. Effective Subroutine Program Procedures

These procedures are effective in reducing memory size of the program and increasing program execution speed. Convenient instructions that are used in subroutines are discussed.

1.1 Subroutine call instructions

The following four instructions can be used as subroutine call instructions:

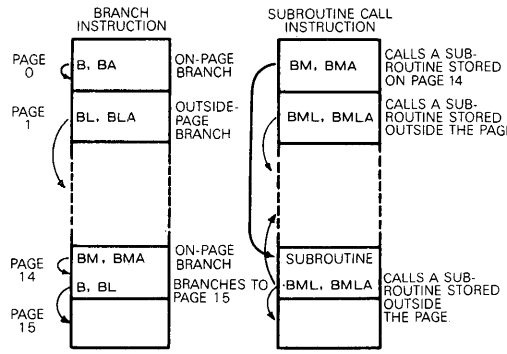
BM, BMA, BML, BMLA

The BM and BMA instructions are one-word instructions that can call all the subroutine stored in page 14. These instructions are designed to designate page 14 automatically by hardware action. If the entrance of a subroutine is programmed on page 14, the subroutine can be called by these one-word instructions, which reduces programming memory requirements.

When the BM, BMA, B or BA instruction is executed on page 14 (in other words, when any of these instructions are used on page 14) the BM and BMA instructions will operate as a branch on page 14 and the B and BA instructions will operate as a branch on page 15. When any of the RT, RTS, BL, BLA, BML and BMLA instructions is executed, this special function is cancelled and BM, BMA, B and BA no longer have a special function. That is, the BM and BMA instructions operate as subroutine call instructions on page 14 and the B and BA instructions as on-page branch instructions. Details of these functions are explained in Fig. 1.

In case the whole subroutine cannot be stored on page 14, only the entrance to the subroutine should be stored on page 14. The balance of the subroutine programs should be stored on another page and branched to. Page 14 can be used without any problems for programs other than subroutines.

Fig. 1 Subroutine call instructions



Note 1 : The B and BA instruction will branch on page 15, and the BM and BMA instruction will branch on page 14 if executed without executing an RT, RTS, BL, BLA, BML or BMLA instruction after the execution of a BM or BMA instruction.

1.2 Consecutively described skip instructions

If either arithmetic LA or RAM addressing LAX instructions appear in sequence, only the first instruction will be executed and the successive same instructions are skipped. It is useful for clearing files as shown in Fig. 7.

1.3 In-RAM file designation changing instructions

The following four instructions:

TAM j (where, j = 0~3)
XAM j (where, j = 0~3)
XAMD j (where, j = 0~3)
XAMI j (where, j = 0~3),

automatically change the contents of the X register depending on the contents of the Z register. File designation is made by the immediate modifier j (j = 0~3). Its designating rules are shown in Table 1. These instructions are very useful for shifting and transferring data within files.

Table 1 In-RAM file designation changing rules using the TAM, SAM, SAMD and SAMI instructions.

Contents of the Z register / Value of j	(Z)=0	(Z)=1
0	No change	No change
1	F0⇄F1 F2⇄F3	F4⇄F5 F6⇄F7
2	F0⇄F2 F1⇄F3	F4⇄F6 F5⇄F7
3	F0⇄F3 F1⇄F2	F4⇄F7 F5⇄F6

2. A-D Conversion Programs

A-D conversion is performed by comparing the input voltage of the analog input port K with V_{ref}, which is generated by the D-A converter, and checking the contents of the H-L register until they at are the same level. Register Y designates the port K input. For example, the input K_y is selected when the contents of the Y register are y.

There are two methods, successive approximation and sequential comparison, for A-D conversion. Either is selected by means of the program.

2.1 Successive approximation method

Program Operation

In this method, the input voltage in the analog input port $K_{(Y)}$ is converted to an 8-bit digital value using the successive approximation technique, and the result is stored in the H-L register.

Its program flow is shown in Fig. 2. The H-L register is first cleared, and then the C register is set to designate the most significant bit (MSB) of the H-L register. When the instruction CPA is executed after "1" has been set in the MSB, the input voltage in the analog input port $K_{(Y)}$ is compared with the D-A conversion output V_{ref}.

When

$|V_{ref}| > |V_{K(Y)}|$

is met during the execution of the next instruction (during the execution of the NOP instruction), $J_{(Y)}$ is set to "1". Otherwise it will be reset to "0"

If

$|V_{ref}| > |V_{K(Y)}|$ i.e. $J_{(Y)} = 1$

the MSB of the H-L register is reset to "0".

If

$|V_{ref}| < |V_{K(Y)}|$ i.e. $J_{(Y)} = 0$

the MSB will remain as "1". Then (C) is decremented by 1, and the above procedure is repeated eight times until reaching the least significant bit (LSB).

When using ports other than those used for analog quantity measurements such as when using port K as a key input port and applying a low level from D_{REF}, program statements 21 and 22 shown in Fig. 4 must be changed as shown below. The original program functions are retained.

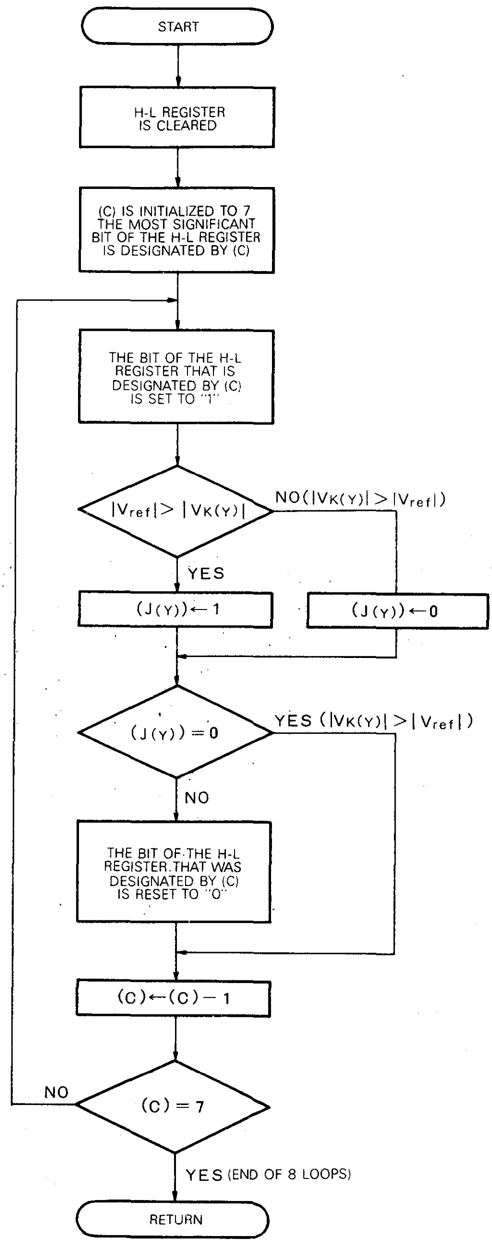
CPA	**CPAS**
NOP	**NOP**
	CPAE

This successive approximation method has a constant conversion speed—approximately 0.6ms at 600kHz—and thus it is suitable for examining analog value with large variations and detecting different analog values from multiple channels.

Subroutine Call

The subroutine is called after designating the terminal of the analog input port K and the bit position of the J register with the Y register. A-D conversion is performed

Fig. 2 A-D conversion subroutine flow chart for the successive approximation method



for the port K_0 in the following example.

```
LXY  0, 0
BM   ADC1
```

2.2 Sequential comparison method

Program Operation

In this method, the input voltage in the analog input port $K_{(Y)}$ is converted to an 8-bit digital value using the sequential comparison technique, and the result is stored in the H-L register.

Its program flow is shown in Fig. 3. First the appropriate contents of the H-L register are D-A converted, and the V_{ref} is compared with the input $V_{K(Y)}$.

If

$|V_{ref}| > |V_{K(Y)}|$ then (CY) is set to "1"

and if

$|V_{ref}| < |V_{K(Y)}|$ then (CY) is reset to "0"

The H-L register is decremented when (CY) is 1 and decreases $|V_{ref}|$ by $V_{REF}/256$. Otherwise, the H-L register is incremented, when (CY) is 0, and increases $|V_{ref}|$ by $V_{REF}/256$. The comparison will come to an end when the magnitudes of $|V_{ref}|$ and $|V_{K(Y)}|$ are exchanged.

The contents of the H and L registers are stored in the A register, and the contents of the A register are either incremented or decremented. First, the low-order 4 bits (L register) are incremented or decremented, followed by of the high-order 4 bits (H register), and then the L register again.

To increment the A register, 1 is added to that register. Testing whether the (A) is 15 or not is performed by the A instruction and by checking if the carry is 1. To decrement the A register, 15 is added to the A register. Testing whether (A) is 0 or not is performed by the A instruction and by checking if the carry is 0.

It will test (H) = 0, when $V_{ref} = \frac{0}{256}V_{REF}$ is met, and will test (H) = 15, when $V_{ref} = \frac{256}{256}V_{REF}$ is met.

Subroutine Call

The subroutine call is executed after designating the terminal of the analog input port K and the bit position of the J register with the Y register. A-D conversion is performed for the port K_0 in the following example. However, it will reduce conversion time if the subroutine is called after setting an expected value in the H and L registers, in cases where the digital value can be anticipated.

```
LXY  0, 0
(H) ← expected value
(L) ← expected value
BM   ADC2
```

Fig. 3 A-D conversion subroutine flow chart for the sequential comparison method.

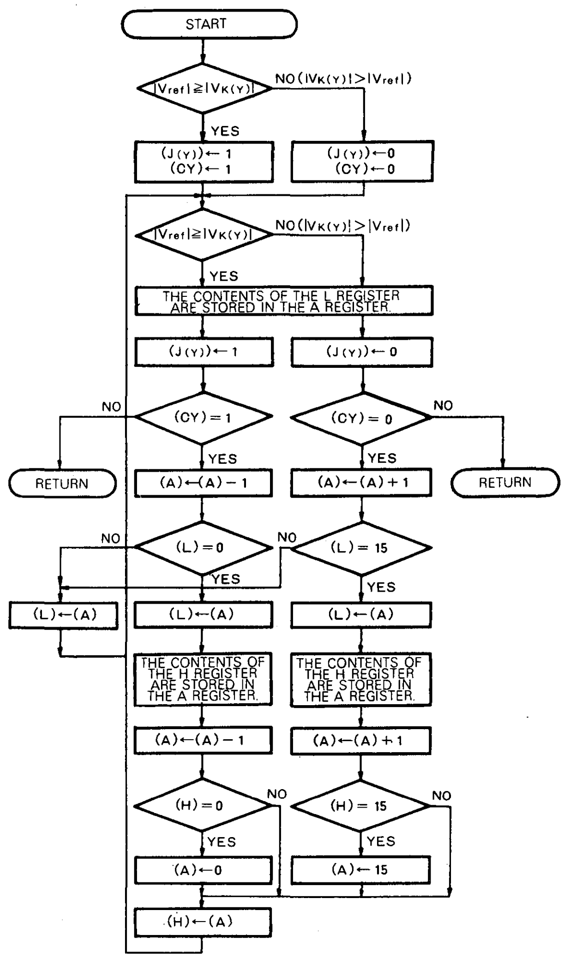


Fig. 4 ADC1 program list

```
                     ORG   E,0
              *
              *
              MAX    EQU   7
              MIN    EQU   12
              J      EQU   0
              SIGN   EQU   12
              *
              *
              *##########  ( MELPS 4 LIBLARY NO.1 )  ##########*
              *
              ***********************************************************
              *SUBR: ADC1  8-BIT A-D CONVERSION, BY SUCCESSIVE APPROXIMA.*
              ***********************************************************
     00  0B0  ADC1   LA    0      CLEAR A, (A)=0
     01  019         TLA          CLEAR L, (L)=0
     02  059         THA          CLEAR H, (H)=0
     03  057         LC7          (C)=7
     04  042  ADC10  SHL          SET H-L, BIT IS ASSIGNED BY (C)
     05  008         CPA          COMPARE PORT K & VREF
     06  000         NOP          SET J IF ABS VREF.GT.ABS VK(Y)
     07  029         SZJ          SKIP IF (J(Y))=0
     08  052         RHL          RESET H-L
     09  009         DEC          (C)=(C)-1,SKIP IF (C)=0
 *W0*0A  104         BM    ADC10  REPEAT 8 TIMES
     0B  044         RT           END OF ADC1
              *
```

Fig. 5 ADC2 program list

```
              ***********************************************************
              *SUBR: ADC2  8-BIT A-D CONVERSION, BY SEQUENTIAL COMPARISON*
              ***********************************************************
     0C  008  ADC2   CPA          COMPARE PORT K & VREF
     0D  048         RC           (CY)=0
     0E  029         SZJ          SKIP IF (J(Y))=0
     0F  049         SC           (CY)=1
     10  008  ADC21  CPA          COMPARE PORT K & VREF
     11  018         XAL          (A) EX (L)
     12  029         SZJ          SKIP IF (J(Y))=0
 *W0*13  11A         BM    ADC23  ACTS AS INSTRUCTION B ON PAGE 14
     14  02F         SZC          SKIP IF (CY)=0
     15  044         RT           RETURN,CONVERSION FINISHED
              *
     16  0A1         A     1      (A)=(A)+1,SKIP IF CARRY=0
 *W0*17  126         BM    ADC26  ACTS AS INSTRUCTION B ON PAGE 14
     18  019  ADC22  TLA          (L)=(A)
 *W0*19  110         BM    ADC21  ACTS AS INSTRUCTION B ON PAGE 14
     1A  02F  ADC23  SZC          SKIP IF (CY)=0
 *W0*1B  11D         BM    ADC24  ACTS AS INSTRUCTION B ON PAGE 14
     1C  044         RT           RETURN,CONVERSION FINISHED
     1D  0AF  ADC24  A     15     (A)=(A)+15,SKIP IF CARRY=0,(A)=(A)-1
 *W0*1E  118         BM    ADC22  ACTS AS INSTRUCTION B ON PAGE 14
     1F  019         TLA          (L)=(A)
     20  058         XAH          (A) EX (H)
     21  0AF         A     15     (A)=(A)+15,SKIP IF CARRY=0, (A)=(A)-1
 *W0*22  124         BM    ADC25  ACTS AS INSTRUCTION B ON PAGE 14
     23  0B0         LA    0      (A)=0
     24  059  ADC25  THA          (H)=(A)
 *W0*25  110         BM    ADC21  ACTS AS INSTRUCTION B ON PAGE 14
     26  019  ADC26  TLA          (L)=(A)
     27  058         XAH          (A) EX (H)
     28  0A1         A     1      (A)=(A)+1, SKIP IF CARRY=0
     29  0BF         LA    15     (A)=15
 *W0*2A  124         BM    ADC25  ACTS AS INSTRUCTION B ON PAGE 14
              *                   END OF ADC2
              *
```

3. Clear File

Program Operation

These are subroutines that are used in clearing files F0~F7, which are formed in the RAM area and are organized as up to 16 words each. The file organization is shown in Fig. 6. These are subroutines, selected by the Z register, that clear the addresses 0~MAX (MAX = 0~15) or that clear the addresses MIN~15 (MIN = 0~15). After MAX and MIN have been initialized and then an LXY instruction that designates the file number is branched, only the first LXY instruction will be executed, and the successive ones are skipped.

To use CFM to make a subroutine that clears the addresses MIN~MAX designated by the Y register of each file, the instruction set SEY max is inserted after the XAMI 0 instruction.

Subroutine Call

An example of subroutine call is shown in Fig. 7. The constants MAX and MIN first have to be equated by a pseudo instruction. A file group is then selected by the Z register as shown below:

When (Z) = "0": F0, F1, F2, F3

When (Z) = "1": F4, F5, F6, F7

then the BM instruction calls a subroutine of each file unit.

Fig. 6 Function of clear-file subroutine

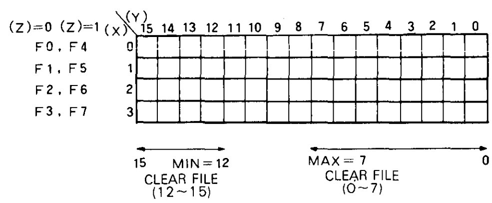


Fig. 7 How to call the clear-file subroutines

```
MAIN
MAX   EQU   7          ........ MAX is intialized to 7 during assembly.
MIN   EQU   12         ........ MIN is initialized to 12 during assembly.
        ⋮
      LZ    0          ........ File groups F0, F1, F2 and F3 are selected.
      BM    CF0        ........ Clears F0 (0~7).
      BM    CF1        ........ Clears F1 (0~7).
      BM    CF2M       ........ Clears F2 (12~15).
      BM    CF3M       ........ Clears F3 (12~15).
      LZ    1          ........ The file groups F4, F5, F6, and F7 are selected.
      BM    CF0        ........ Clears F4 (0~7).
      BM    CF2M       ........ Clears F6 (12~15).

SUBROUTINES (ON PAGE 14)
CF0   LXY   0, MAX
CF1   LXY   1, MAX
CF2   LXY   2, MAX
CF3   LXY   3, MAX
SUBROUTINE COMMON PROCESSING

      RT
```

4. Right-Shift File

Program Operation

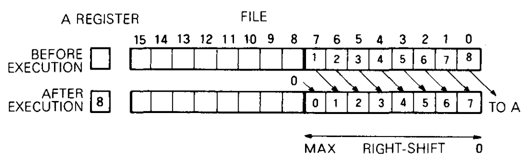
This is a subroutine that is used to right-shift the files F0~F7, as shown in Fig. 8. The contents of address 0 ~MAX (MAX = 0~15) in a file designated by the Y register are shifted right one digit. The most significant digit (MSD) is filled with 0 and the contents of the least significant digit (LSD) are stored in the A register.

Subroutine Call

The constant MAX has to be equated by using a pseudo instruction. Then the appropriate file group is selected by the X register before calling the subroutine. An example is shown below, in which the digit numbers 0~7 of the file F are shifted right 2 digits

```
MAX  EQU  7
      :
     LZ   1
     BM   RSF1
     BM   RSF1
```

Fig. 8 Example of right-shift file execution



5. Left-Shift File

Program Operation

This is a subroutine that is used to left-shift the files F0~F7, as shown in Fig. 9. The contents of address MIN ~15 (MIN = 0~15) in a file designated by the Y register are shifted left one digit. The least significant digit (LSD) is filled with 0 and the contents of the most significant digit (MSD) are stored 17 the A register.

A subroutine that is to left-shift MIN~MAX can be made by inserting

SEY max

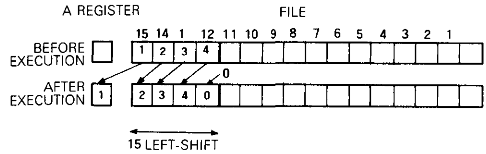
following the XAMI 0 instruction. When MIN = 0 is equated, it performs the same digits as the right-shift file subroutine. The instruction SEY, however, may be omitted when the skip condition is altered by the optional XAMI instruction.

Subroutine Call

The constant MIN has to be equated by using a pseudo instruction. Then the appropriate file group is selected by the Z register before calling the subroutine. An example is shown below, in which the digit numbers 12~15 of the file F7 are shifted left one digit.

```
MIN  EQU  12
      :
     LZ   1
     BM   LSF3
```

Fig. 9 Example of left-shift file execution



6. Transfer of File

Program Operation

This is a subroutine that is used for transferring the contents of the files F0~F7. The data (MAX + 1 words) in the addresses 0~MAX (MAX = 0~15) of the file designated by the Y register is transferred.

As already discussed in section 1.3, changing file designation in the RAM is automatically performed by the TAM j and XAMD j instructions. An example is shown in Fig. 10, in which the contents of the file F0 are transferred to the file F1. Each time the TAM 1 and XAMD 1 instructions are executed, the designated file changes to F0→ F1→F0 . . . and so on.

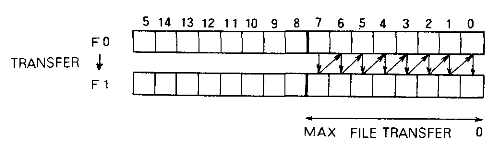
Data-transfer subroutines of the address MIN~15 can be made by changing MAX to MIN and the XAMD j instruction to XAMI j.

Subroutine Call

The constant MAX has to be equated by using a pseudo instruction. Then the appropriate file group is selected by the Z register before calling the subroutine. An example is shown below, in which file F0 is transferred to file F1 and F5 to F7. Digits transferred in each file are 0~7.

```
MAX  EQU  7
      :
     LZ   0
     BM   TF10
     LZ   1
     BM   TF31
```

Fig. 10 File transfer example of (F1) ← (F0)



Note 2 : The arrows show how the file is changed.

Fig. 11 CF, CFM, RST and LSF program lists

```
                ************************************************************
                *SUBR: CF     CLEAR FILE FX(0-MAX)=0                         *
                ************************************************************
     28   0C7   CF0     LXY    0,MAX   F0(0-MAX)=0 OR F4(0-MAX)=0
     2C   0D7   CF1     LXY    1,MAX   F1(0-MAX)=0 OR F5(0-MAX)=0
     2D   0E7   CF2     LXY    2,MAX   F2(0-MAX)=0 OR F6(0-MAX)=0
     2E   0F7   CF3     LXY    3,MAX   F3(0-MAX)=0 OR F7(0-MAX)=0
     2F   0B0   CF01    LA     0       (A)=0
     30   068           XAMD   0       (M) EX (A), (Y)=(Y)-1,SKIP IF (Y)=0
*WO*31   12F            BM     CF01    ACTS AS INSTRUCTION B ON PAGE 14
     32   044           RT             END OF CF
                *
                ************************************************************
                *SUBR: CFM    CLEAR FILE FX(MIN-15)=0                        *
                ************************************************************
     33   0CC   CF0M    LXY    0,MIN   F0(MIN-15)=0 OR F4(MIN-15)=0
     34   0DC   CF1M    LXY    1,MIN   F1(MIN-15)=0 OR F5(MIN-15)=0
     35   0EC   CF2M    LXY    2,MIN   F2(MIN-15)=0 OR F6(MIN-15)=0
     36   0FC   CF3M    LXY    3,MIN   F3(MIN-15)=0 OR F7(MIN-15)=0
     37   0B0   CF0M1   LA     0       (A)=0
     38   06C           XAMI   0       (M) EX (A), (Y)=(Y)+1,SKIP IF (Y)=15
*WO*39   137            BM     CF0M1   ACTS AS INSTRUCTION B ON PAGE 14
     3A   044           RT             END OF CFM
                *
                ************************************************************
                *SUBR: RSF    RIGHT-SHIFT FILE  FX(0-MAX),FX(MAX)=0,(A)=FX(0)*
                ************************************************************
     3B   0C7   RSF0    LXY    0,MAX   F0(0-MAX) R-S, F0(MAX)=0, (A)=F0(0)
     3C   0D7   RSF1    LXY    1,MAX   F1(0-MAX) R-S, F1(MAX)=0, (A)=F1(0)
     3D   0E7   RSF2    LXY    2,MAX   F2(0-MAX) R-S, F2(MAX)=0, (A)=F2(0)
     3E   0F7   RSF3    LXY    3,MAX   F3(0-MAX) R-S, F3(MAX)=0, (A)=F3(0)
     3F   0B0           LA     0       (A)=0
     40   068   RSF01   XAMD   0       (M) EX (A), (Y)=(Y)-1, SKIP IF (Y)=0
*WO*41   140            BM     RSF01   ACTS AS INSTRUCTION B ON PAGE 14
     42   044           RT             END OF RSF
                *
                ************************************************************
                *SUBR: LSF    LEFT-SHIFT FILE FX(MIN-15),FX(MIN)=0,(A)=FX(15)*
                ************************************************************
     43   0CC   LSF0    LXY    0,MIN   F0(MIN-15) L-S, F0(MIN)=0, (A)=F0(15)
     44   0DC   LSF1    LXY    1,MIN   F1(MIN-15) L-S, F1(MIN)=0, (A)=F1(15)
     45   0EC   LSF2    LXY    2,MIN   F2(MIN-15) L-S, F2(MIN)=0, (A)=F2(15)
     46   0FC   LSF3    LXY    3,MIN   F3(MIN-15) L-S, F3(MIN)=0, (A)=F3(15)
     47   0B0           LA     0       (A)=0
     48   06C   LSF01   XAMI   0       (M) EX (A), (Y)=(Y)+1, SKIP IF (Y)=15
*WO*49   148            BM     LSF01   ACTS AS INSTRUCTION B ON PAGE 14
     4A   044           RT             END OF LSF
                *
```

Fig. 12 TF program list

```
                ************************************************************
                *SUBR: TF     TRANSFER OF FILE FX1(0-MAX)=FX2(0-MAX)         *
                ************************************************************
     4B   0C7   TF10    LXY    0,MAX   F1(0-MAX)=F0(0-MAX)
     4C   0D7   TF01    LXY    1,MAX   F0(0-MAX)=F1(0-MAX)
     4D   0E7   TF32    LXY    2,MAX   F3(0-MAX)=F2(0-MAX)
     4E   0F7   TF23    LXY    3,MAX   F2(0-MAX)=F3(0-MAX)
     4F   065   TF101   TAM    1       (A)=(M(DP))
     50   069           XAMD   1       (A)=(M(DP)),(Y)=(Y)-1,SKIP IF (Y)=0
*WO*51   14F            BM     TF101   ACTS AS INSTRUCTION B ON PAGE 14
     52   044           RT             END OF TF10
                *
     53   0C7   TF20    LXY    0,MAX   F2(0-MAX)=F0(0-MAX)
     54   0D7   TF31    LXY    1,MAX   F3(0-MAX)=F1(0-MAX)
     55   0E7   TF02    LXY    2,MAX   F0(0-MAX)=F2(0-MAX)
     56   0F7   TF13    LXY    3,MAX   F1(0-MAX)=F3(0-MAX)
     57   066   TF201   TAM    2       (A)=(M(DP))
     58   06A           XAMD   2       (A)=(M(DP)),(Y)=(Y)-1,SKIP IF (Y)=0
*WO*59   157            BM     TF201   ACTS AS INSTRUCTION B ON PAGE 14
     5A   044           RT             END OF TF20
                *
     5B   0C7   TF30    LXY    0,MAX   F3(0-MAX)=F0(0-MAX)
     5C   0D7   TF21    LXY    1,MAX   F2(0-MAX)=F1(0-MAX)
     5D   0E7   TF12    LXY    2,MAX   F1(0-MAX)=F2(0-MAX)
     5E   0F7   TF03    LXY    3,MAX   F0(0-MAX)=F3(0-MAX)
     5F   067   TF301   TAM    3       (A)=(M(DP))
     60   06B           XAMD   3       (A)=(M(DP)),(Y)=(Y)-1,SKIP IF (Y)=0
*WO*61   15F            BM     TF301   ACTS AS INSTRUCTION B ON PAGE 14
     62   044           RT             END OF TF30
                *
```

7. File Exchange

Program Operation

This is a subroutine that is used for exchanging the contents of the files F0~F7. The data (MAX + 1 words) in the addresses of 0~MAX (MAX = 0~15) of the designated files is exchanged.

Exchanging of in-RAM files is performed by the TAM j and XAM j instructions.

Data-exchange subroutines of the address MIN~15 (MIN = 0~15) can be made by changing MAX to MIN and the XAMD 0 instruction to XAMI 0.

Subroutine Call

The constant MAX has to be equated by using a pseudo instruction. Then the appropriate file group is selected by the Z register before calling the subroutine. An example is shown below, in which file F0 is exchanged with file F1 and F4 with F7. Digits exchanged in all files are 0~7.

```
MAX  EQU  7
      ⋮
     LZ   0
     BM   EXF01
     LZ   1
     BM   EXF03
```

8. Increment/Decrement Memory

Program Operation

This is a subroutine that is used to increment or decrement the contents of a specific word in the RAM. The specific addresses that can be incremented or decremented are shown below.

M (z, 0, 0) ($F0_0$ or $F4_0$)
M (z, 1, 11) ($F1_{11}$ or $F5_{11}$)
M (z, 2, 13) ($F2_{13}$ or $F6_{13}$)
M (z, 3, max) ($F3_{max}$ or $F7_{max}$)

Other addresses can be programmed by changing the LXY x, y instruction.

Subroutine Call

The appropriate file group is selected by the Z register before calling the subroutine. An example is shown below, in which M (0, 0, 0) is incremented and M (1, 2, 13) is decremented:

```
LZ   0
BM   INM000
LZ   1
BM   DEM213
```

9. Skip Non-Zero Memory

Program Operation

This is a subroutine that is used in test if the contents of specific words in the RAM are 0. The specific addresses that can be tested are shown below.

M (z, 0, 0) ($F0_0$ or $F4_0$)
M (z, 1, 11) ($F1_{11}$ or $F5_{11}$)
M (z, 2, 13) ($F2_{13}$ or $F6_{13}$)
M (z, 3, max) ($F3_{max}$ or $F7_{max}$)

If the contents of the specified address of the RAM are 0, the program execution returns to the instruction following the one that called the subroutine. If the contents are not 0, this instruction is skipped, and the return is to the second instruction following the call.

Other addresses can be tested by changing the LXY x, y instruction.

Subroutine Call

The appropriate file group is selected by the Z register before calling the subroutine. When the contents of the RAM are 0, execution returns to the following instruction. When the contents of the RAM are not 0, the execution returns to the second instruction.

The following is an example in which the contents of M (1, 1, 11) are tested:

```
LZ   1
BM   SNM111
INST 1 ← Return if "0"
INST 2 ← Return if "1"
```

10. Skip Non-Zero File

Program Operation

This is a subroutine that is used to test if all the words of files F0~F7 are 0. There are two subroutines applicable: one for testing the addresses 0~MAX (MAX = 0~15) and the other for testing the addresses MIN~15 (MIN = 0~15).

If the contents of the specified file are 0, the program execution returns to the instruction following the one that called the subroutine. If the contents are not 0, this instruction is skipped, and the return is to the second instruction following the call.

In case of digit MIN~MAX, a program can be made by inserting SEY MAX next to the instruction XAMI 0 of the subroutine SNFMI.

Subroutine Call

The appropriate file group is selected by the Z register before calling the subroutine. In case the contents of the file are 0, the program execution returns to the instruction following the one that called the subroutine. If the contents are not 0, the program execution skips this instruction. An example is shown below, in which the contents of the file FO_0~FO_7 are tested.

Fig. 13 EXT, IMN, DEM and SNM program lists

```
                 ************************************************************
                 *SUBR: EXF    EXCHANGE OF FILE FX1(O-MAX) EX FX2(O-MAX)      *
                 ************************************************************
  63      0C7    EXF01  LXY    0,MAX   F0(0-MAX) EX F1(0-MAX)
  64      0E7    EXF23  LXY    2,MAX   F2(0-MAX) EX F3(0-MAX)
  65      065    EXF001 TAM    1       (A)=(M(DP))
  66      061           XAM    1       (A) EX (M(DP))
  67      068           XAMD   0       (A) EX (M(DP)),(Y)=(Y)-1,SKIP IF (Y)=0
*WO*68    165           BM     EXF001  ACTS AS INSTRUCTION B ON PAGE 14
  69      044           RT             END OF EXF01
                 *
  6A      0C7    EXF02  LXY    0,MAX   F0(0-MAX) EX F2(0-MAX)
  6B      0D7    EXF13  LXY    1,MAX   F1(0-MAX) EX F3(0-MAX)
  6C      066    EXF002 TAM    2       (A)=(M(DP))
  6D      062           XAM    2       (A) EX (M(DP))
  6E      068           XAMD   0       (A) EX (M(DP)),(Y)=(Y)-1,SKIP IF (Y)=0
*WO*6F    16C           BM     EXF002  ACTS AS INSTRUCTION B ON PAGE 14
  70      044           RT             END OF EXF02
                 *
  71      0C7    EXF03  LXY    0,MAX   F0(0-MAX) EX F3(0-MAX)
  72      0D7    EXF12  LXY    1,MAX   F1(0-MAX) EX F2(0-MAX)
  73      067    EXF003 TAM    3       (A)=(M(DP))
  74      063           XAM    3       (A) EX (M(DP))
  75      068           XAMD   0       (A) EX (M(DP)),(Y)=(Y)-1,SKIP IF (Y)=0
*WO*76    173           BM     EXF003  ASTS AS INSTRUCTION B ON PAGE 14
  77      044           RT             END OF EXF03
                 *
                 ###########   ( MELPS 4 LIBRARY END )   ##########*
                        END
                 ###########   ( MELPS 4 LIBRARY NO.2 )   ##########*
                 ************************************************************
                 *SUBR: INM    INCREMENT MEMORY FX(Y)=FX(Y)+1                 *
                 *SUBR: DEM    DECREMENT MEMORY FX(Y)=FX(Y)-1                 *
                 ************************************************************
  00      0C0    INM000 LXY    0,0     F0(0)   =F0(0)   +1 OR F4(0)   =F4(0)   +1
  01      0DB    INM111 LXY    1,11    F1(11)  =F1(11)  +1 OR F5(11)  =F5(11)  +1
  02      0ED    INM213 LXY    2,13    F2(13)  =F2(13)  +1 OR F6(13)  =F6(13)  +1
  03      0F7    INM3MA LXY    3,MAX   F3(MAX)=F3(MAX)+1 OR F7(MAX)=F7(MAX)+1
  04      0B1           LA     1       (A)=1
  05      00A    INM    AM             (A)=(A)+(M(DP))
  06      060           XAM    0       (A) EX (M(DP))
  07      0B0           LA     0       (A)=0
  08      044           RT
  09      0C0    DEM000 LXY    0,0     F0(0)   =F0(0)   -1 OR F4(0)   =F4(0)   -1
  0A      0DB    DEM111 LXY    1,11    F1(11)  =F1(11)  -1 OR F5(11)  =F5(11)  -1
  0B      0ED    DEM213 LXY    2,13    F2(13)  =F2(13)  -1 OR F6(13)  =F6(13)  -1
  0C      0F7    DEM3MA LXY    3,MAX   F3(MAX)=F3(MAX)-1 OR F7(MAX)=F7(MAX)-1
  0D      0BF           LA     15      (A)=15
*WO*0E    105           BM     INM     END OF INM AND DEM
                 *
                 ************************************************************
                 *SUBR: SNM    SKIP NON-ZERO MEMORY  FX(Y).NE.0 ?             *
                 ************************************************************
  0F      0C0    SNM000 LXY    0,0     F0(0)   .NE.0 ? OR F4(0)   .NE.0 ?
  10      0DB    SNM111 LXY    1,11    F1(11)  .NE.0 ? OR F5(11)  .NE.0 ?
  11      0ED    SNM213 LXY    2,13    F2(13)  .NE.0 ? OR F6(13)  .NE.0 ?
  12      0F7    SNM3MA LXY    3,MAX   F3(MAX).NE.0 ? OR F7(MAX).NE.0 ?
  13      064           TAM    0       (A)=(M(DP))
  14      0AF           A      15      (A)=(A)+15,SKIP IF CARRY=0
  15      045           RTS            RETURN IF FX(Y).NE.0
  16      044           RT             RETURN IF FX(Y).EQ.0
                 *                     END OF SNM
                 *
```

```
MAX  EQU  7
      :
     LZ   0
     BM   SNFOMA
```

Instruction 1 ← Return if ($F0_0 \sim F0_7 = 0$)
Instruction 2 ← Return if not ($F0_0 \sim F0_7 = 0$)

11. BCD Addition of Files

Program Operation

This is a subroutine that is used to perform addition in the BCD mode among the files F0~F7. It performs BCD addition of 16–MIN digits in the addresses MIN~15 (MIN = 0~15). The flowchart is shown in Fig. 14, and an example is shown in Fig. 15.

First, the carry CY has to be cleared. The file FX1 is BCD compensated by adding 6 to its contents. Then the contents of the FX2 are added to the contents of the FX1 and the carry is checked. When the carry is off, 10 is added to its contents, which is the same as subtracting 6, and there is no need to BCD adjust. The files FX1 and the FX2 can be alternated by the TAM j and XAMI j instructions. When the BCD addition of the most significant digit is completed, the contents of the carry CY are checked. If (CY) = 0, the program execution returns to the main program after skipping the instruction following the call. When (CY) = 1, indicating an overflow, the program execution will return to the instruction following the call. It is possible to test for overflow state by testing the CY. In this case, the instruction following the instruction SZC is replaced with the instruction RT.

Selection of the files FX1 and the FX2 is made by changing x of the LXY x, y instruction and j of the TAM j and XAMI j instructions.

SUB-ROUTINE	x	j	(FX1)←(FX1)+(FX2)
ADF10	0	1	(F1)←(F1)+(F0) or (F5)←(F5)+(F4)
	0	2	(F2)←(F2)+(F0) or (F6)←(F6)+(F4)
	0	3	(F3)←(F3)+(F0) or (F7)←(F7)+(F4)
ADF01	1	1	(F0)←(F0)+(F1) or (F4)←(F4)+(F5)
	1	2	(F3)←(F3)+(F1) or (F7)←(F7)+(F5)
	1	3	(F2)←(F2)+(F1) or (F6)←(F6)+(F5)
ADF32	2	1	(F3)←(F3)+(F2) or (F7)←(F7)+(F6)
	2	2	(F0)←(F0)+(F2) or (F4)←(F4)+(F6)
	2	3	(F1)←(F1)+(F2) or (F5)←(F5)+(F6)
ADF23	3	1	(F2)←(F2)+(F3) or (F6)←(F6)+(F7)
	3	2	(F1)←(F1)+(F3) or (F5)←(F5)+(F7)
	3	3	(F0)←(F0)+(F3) or (F4)←(F4)+(F7)

Subroutine Call

The value j has to be equated by using a pseudo instruction. The appropriate file group is selected by the Z register before calling the subroutine. The program execution will skip the instruction following the subroutine call when the result of the BCD addition is correct, and return to this instruction when there is an overflow. An example of ($F0_{15} \sim F0_{12}$) ← ($F0_{15} \sim F0_{12}$) + ($F1_{15} \sim F1_{12}$) is shown below:

```
MIN  EQU  12
J    EQU  1
      :
     LZ   0
     BM   ADF01
```

Instruction 1 ← Return if overflow
Instruction 2 ← Return if no overflow

Fig. 14 BCD file addition subroutine flowchart

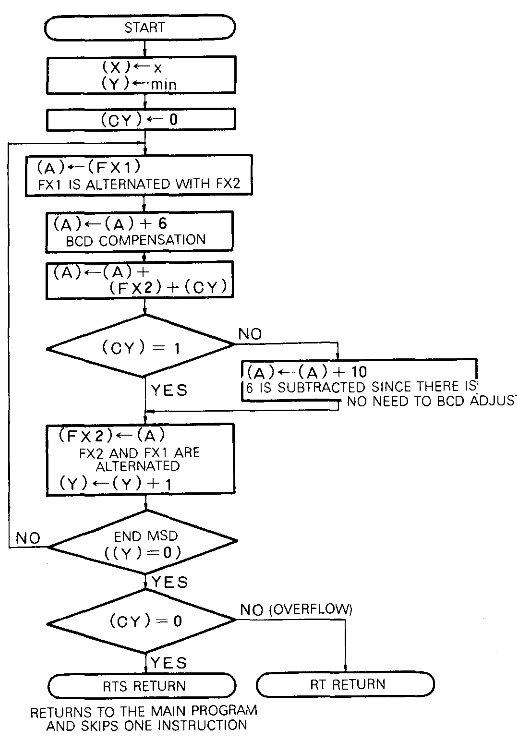
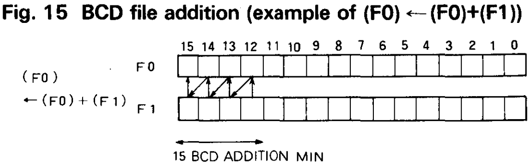


Fig. 15 BCD file addition (example of (F0) ← (F0)+(F1))



Note 3 : The arrows show how the file is changed.

12. BCD Subtraction of Files

Program Operation

This is a subroutine that is used to perform subtraction in the BCD mode among the files F0~F7. It performs BCD subtraction of 16–MIN digits of the address MIN~15 (MIN = 0~15).

It has the same program procedure as BCD addition, performing subtraction by adding the 1's complement. When the borrow is 1, BCD adjustment is performed by adding 10.

File selection of the files FX1 and FX2 is made by changing x of the LXY x, y instruction and j of the TAM j and XAMI j instructions, as in BCD addition. Please refer to the procedure given in the section for BCD addition.

Subroutine Call

The value j has to be equated by using a pseudo instruction. An appropriate file group is selected by the Z register before calling the subroutine. The program execution will skip the instruction following the subroutine call when the result of the BCD subtraction is correct, and return to the next instruction when subtraction results in a carry. An example of $(F7_{15}\sim F7_{12}) \leftarrow (F7_{15}\sim F7_{12}) - (F5_{15}\sim F5_{12})$ is shown at right:

```
MIN  EQU  12
J    EQU  2
      :
     LZ   1
     BM   SBF32
```

Instruction 1 ← Return if overflow
Instruction 2 ← Return if no overflow

13. Sign Change of file

Program Operation

This is a subroutine that is used to invert the sign in the sign digit, SIGN (SIGN = 0~15), of the files F0~F7. The positive state is indicated when the 8 bit is 0, and the negative state when the 8 bit is 1. Thus inversion is attained by adding 8 to memory.

Subroutine Calling Method

Psuedo instructions are used to fix the code digit. Register Z specifies the file group and the subroutine is called. Next, the 12th digit of file F0 is inverted as shown.

```
SIGN  EQU 12
        :
        :
      LZ 0
      BM SCFO
```

Fig. 16 SNFMA and SNFMI program lists

```
                   ************************************************************
                   *SUBR: SNFMA SKIP NON-ZERO FILE    FX(0-MAX).NE.0 ?        *
                   ************************************************************
      17     0C7   SNF0MA LXY   0,MAX  F0(0-MAX).NE.0 ? OR F4(0-MAX).NE.0 ?
      18     0D7   SNF1MA LXY   1,MAX  F1(0-MAX).NE.0 ? OR F5(0-MAX).NE.0 ?
      19     0E7   SNF2MA LXY   2,MAX  F2(0-MAX).NE.0 ? OR F6(0-MAX).NE.0 ?
      1A     0F7   SNF3MA LXY   3,MAX  F3(0-MAX).NE.0 ? OR F7(0-MAX).NE.0 ?
      1B     0B0   SNF4   LA    0      (A)=0
      1C     026          SEAM         SKIP IF (A).EQ.(M(DP))
      1D     045          RTS          RETURN IF FX(0-MAX).NE.0
      1E     068          XAMD  0      (A)=(M(DP)),(Y)=(Y)-1,SKIP IF (Y)=0
  *WO*1F     11B          BM    SNF4   ACTS AS INSTRUCTION B ON PAGE 14
      20     044          RT           RETURN IF FX(0-MAX).EQ.0
                   *                   END OF SNFMA
                   *
                   ************************************************************
                   *SUBR: SNFMI SKIP NON-ZERO FILE    FX(MIN-15).NE.0 ?       *
                   ************************************************************
      21     0CC   SNF0MI LXY   0,MIN  F0(MIN-15).NE.0 ? OR F4(MIN-15).NE.0 ?
      22     0DC   SNF1MI LXY   1,MIN  F1(MIN-15).NE.0 ? OR F5(MIN-15).NE.0 ?
      23     0EC   SNF2MI LXY   2,MIN  F2(MIN-15).NE.0 ? OR F6(MIN-15).NE.0 ?
      24     0FC   SNF3MI LXY   3,MIN  F3(MIN-15).NE.0 ? OR F7(MIN-15).NE.0 ?
      25     0B0   SNF5   LA    0      (A)=0
      26     026          SEAM         SKIP IF (A).EQ.(M(DP))
      27     045          RTS          RETURN IF FX(MIN-15).NE.0
      28     06C          XAMI  0      (A)=(M(DP)),(Y)=(Y)+1,SKIP IF (Y)=15
  *WO*29     125          BM    SNF5   ACTS AS INSTRUCTION B ON PAGE 14
      2A     044          RT           RETURN IF FX(MIN-15).EQ.0
                   *                   END OF SNFMI
                   *
```

Fig. 17 ADF, SBF and SCF program lists

```
                   *********************************************************************
                   *SUBR: ADF   BCD ADDITION OF FILE FX1(MIN-15)=FX1(MIN-15)+FX2(MIN-15)*
                   *********************************************************************
      23    0CC    ADF10   LXY    0,MIN   J=1: F1(MIN-15)=F1(MIN-15)+F0(MIN-15)
                   *                      J=2: F2(MIN-15)=F2(MIN-15)+F0(MIN-15)
                   *                      J=3: F3(MIN-15)=F3(MIN-15)+F0(MIN-15)
      2C    0DC    ADF01   LXY    1,MIN   J=1: F0(MIN-15)=F0(MIN-15)+F1(MIN-15)
                   *                      J=2: F3(MIN-15)=F3(MIN-15)+F1(MIN-15)
                   *                      J=3: F2(MIN-15)=F2(MIN-15)+F1(MIN-15)
      2D    0EC    ADF32   LXY    2,MIN   J=1: F3(MIN-15)=F3(MIN-15)+F2(MIN-15)
                   *                      J=2: F0(MIN-15)=F0(MIN-15)+F2(MIN-15)
                   *                      J=3: F1(MIN-15)=F1(MIN-15)+F2(MIN-15)
      2E    0FC    ADF23   LXY    3,MIN   J=1: F2(MIN-15)=F2(MIN-15)+F3(MIN-15)
                   *                      J=2: F1(MIN-15)=F1(MIN-15)+F3(MIN-15)
                   *                      J=3: F0(MIN-15)=F0(MIN-15)+F3(MIN-15)
      2F    048            RC             (CY)=0
      30    064    ADF011  TAM    J       (A)=(M(DP))
      31    0A6            A      6       (A)=(A)+6
      32    00F            AMCS           (A)=(A)+(M(DP))+(CY),(CY)=CARRY
      33    0AA            A      10      (A)=(A)+10,SKIP IF CARRY=0,BCD ADJUST
      34    000            NOP            (A)=(A)-6
      35    06C            XAMI   J       (A) EX (M(DP)),(Y)=(Y)+1,SKIP IF (Y)=15
*WO*36      130            BM     ADF011  ACTS AS INSTRUCTION B ON PAGE 14
      37    02F            SZC            SKIP IF (CY)=0
      38    044            RT             RETURN IF OVERFLOW
      39    045            RTS            END OF ADF01
                   *
                   ***************************************************************
                   *SUBR: SBF   BCD SUBTRACTION  OF FILE                         *
                   *                                =FX1(MIN-15)-FX2(MIN-15)     *
                   ***************************************************************
      3A    0CC    SBF10   LXY    0,MIN   J=1: F1(MIN-15)=F1(MIN-15)-F0(MIN-15)
                   *                      J=2: F2(MIN-15)=F2(MIN-15)-F0(MIN-15)
                   *                      J=3: F3(MIN-15)=F3(MIN-15)-F0(MIN-15)
      3B    0DC    SBF01   LXY    1,MIN   J=1: F0(MIN-15)=F0(MIN-15)-F1(MIN-15)
                   *                      J=2: F3(MIN-15)=F3(MIN-15)-F1(MIN-15)
                   *                      J=3: F2(MIN-15)=F2(MIN-15)-F1(MIN-15)
      3C    0EC    SBF32   LXY    2,MIN   J=1: F3(MIN-15)=F3(MIN-15)-F2(MIN-15)
                   *                      J=2: F0(MIN-15)=F0(MIN-15)-F2(MIN-15)
                   *                      J=3: F1(MIN-15)=F1(MIN-15)-F2(MIN-15)
      3D    0FC    SBF23   LXY    3,MIN   J=1: F2(MIN-15)=F2(MIN-15)-F3(MIN-15)
                   *                      J=2: F1(MIN-15)=F1(MIN-15)-F3(MIN-15)
                   *                      J=3: F0(MIN-15)=F0(MIN-15)-F3(MIN-15)
      3E    049            SC             (CY)=1
      3F    064    SBF011  TAM    J       (A)=(M(DP))
      40    08F            CMA            COMPLEMENT (A)
      41    00F            AMCS           (A)=(A)+(M(DP))+(CY),(CY)=CARRY
      42    0AA            A      10      (A)=(A)+10,SKIP IF CARRY=0,BCD ADJUST
      43    06C            XAMI   J       (A) EX (M(DP)),(Y)=(Y)+1,SKIP IF (Y)=15
*WO*44      13F            BM     SBF011  ACTS AS INSTRUCTION B ON PAGE 14
      45    02F            SZC            SKIP IF (CY)=0
      46    045            RTS            END OF SBF01
      47    044            RT             RETURN IF OVERFLOW
                   *
                   ***********************************************************
                   *SUBR: SCF   SIGN CHANGE OF FILE FX(SIGN) EX              *
                   ***********************************************************
      48    0CC    SCF0    LXY    0,SIGN F0(SIGN) EX
      49    0DC    SCF1    LXY    1,SIGN F1(SIGN) EX
      4A    0EC    SCF2    LXY    2,SIGN F2(SIGN) EX
      4B    0FC    SCF3    LXY    3,SIGN F3(SIGN) EX
      4C    0B8            LA     8       (A)=8
      4D    00A            AM             (A)=(A)+(M(DP))
      4E    060            XAM    0       (A) EX (M(DP))
      4F    044            RT             END OF SCF0
                   *
                   ##########  ( MELPS 4 LIBRARY END )  ##########*
                           END
```

APPLICATION OF MELPS 4 SINGLE-CHIP 4-BIT MICROCOMPUTER (M58840-XXXP) IN A MICROWAVE OVEN

DESCRIPTION

A typical example of an application in which a Mitsubishi MELPS 4 single-chip 4-bit microcomputer is used in the microwave oven.

The system is designed to control the magnetron, fan and buzzer of the microwave oven by the touch-keyboard input, and to display the time and temperature, along with the power, on the large fluorescent display tube, as well as displaying the MODE on the LEDs (8 pieces). Its features include controls for designating the start-up time and controlling the defrosting process (time and power), the cooking process #1 (time, temperature and power) and the cooking process #2 (time, temperature and power). In addition, the clock can be used as an independent timer.

The program for the microwave oven application is stored in the M58840-001P.

FEATURES

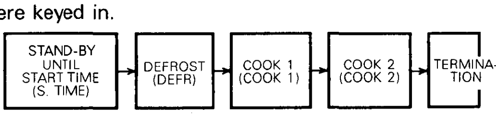
- Programmed operation for DEFROST, COOK 1 and COOK 2 processes
- Time, temperature and power controls
- Clock and timer
- Display of the time, temperature and power on the large fluorescent display tube
- The simplification in circuit design facilitates cost reduction and miniaturization of the oven.

FUNCTION

1. Microwave Oven Function

(1) Outline of operation

When the start key is depressed after setting up the cooking conditions (time, temperature and power) through the touch key, the oven starts operating in the following sequence regardless of the order the conditions were keyed in.



As soon as one process is completed, the next process is started, skipping those processes that are not designated, until finished. In addition, the clock can be used as an independent timer.

(2) Clock

The clock has a 12-hour dial and indicates hours and minutes.

(3) Timer

The timer actuates the buzzer at the specific time designated in minutes and seconds.

(4) Start time

It designates the start time and starts the cooking when that specific time is reached.

(5) Defrosting

Power and time can be selected for defrosting , but when no power setting is made, the oven automatically uses a 50% setting. During the set time, the system controls the magnetron, on and off, to maintain the power specified, and turns the magnetron off as soon as the specific period is over. The oven is kept in this halt condition for the duration.

(6) COOK 1

The operating power, temperature and time can be selected for this process. If no specific power is designated, the oven automatically uses a 100% setting. The operating temperature can be selected in the range of 35°C~95°C. The magnetron is operated, on and off, at the power setting after the cooking has started until the selected temperature is reached. Although the magnetron is turned off after reaching the selected temperature, it is turned on again when the temperature in the oven falls 3°C below the selected temperature. This procedure is repeated until the time is reached for completion of the COOK 1 process.

When no temperature setting is made, the oven operates at the power specified and completes the COOK 1 process when the set time is reached.

(7) COOK 2

The procedures for COOK 2 are the same as those for COOK 1.

(8) Clear

The clear switch is used to change key entries or to advance to the next process and discontinue the process in operation.

(9) Reset

Depressing the reset key terminates the entire cooking process and shifts to clock operation.

(10) Stop

When the stop key is depressed or the door is opened, the cooking process is interrupted. The start key has to be depressed again if the operation is to be resumed.

(11) Display

The operating time, power and temperature are displyed on the fluorescent display tube. The tube displays key-entry data during the key entry. The clock is displayed on the screen by the use of the CLOCK key. It usually indicates remaining cooking time during the cooking operation, but memory contents can be recalled for the clock, power and temperature settings. The oven temperature can also be displayed.

The cooking mode is indicated on the LED.

2. Inputs

(1) Key input: $K_0 \sim K_7$

22 keys are arranged in a matrix through the K ports and the D ports, using the touch keyboard for input. All inputs are checked 8 times in a 100ms period before being accepted as valid. This is done to prevent errors in operating the oven. Furthermore, successive key entry cannot be made until it is confirmed 8 times in a period of 100ms that there were no keys depressed.

The following 22 keys are provided: defrost (DEFR), cook 1 (COOK 1), cook 2 (COOK 2), temperature (TEMP), power (POWER), start (START), stop (STOP), clear (CLEAR), reset (RESET), timer (TIMER), clock (CLOCK), start time (S. TIME) and numbers (0~9).

(2) Time detection input: K_{13}

This input is used to count the time. Rectified AC waveform from the power source is applied.

(3) 50/60Hz switching input: K_9

This input is used to compensate for the power source, 50Hz or 60Hz.

(4) Temperature sensor input: K_{11}

Voltage appropriate to the temperature is applied from the thermistor located in the temperature probe.

(5) Temperature probe SW input: K_8

This input is used in checking whether the temperature probe is operating.

(6) Door SW input, K_{10}

This input is used to check whether the door is open.

(7) Touch keyboard comparison voltage setup input: K_{14}

This is an input with which the detection level is set up for the touch keyboard. It very useful when the specifications of the touch keyboard are altered.

3. Outputs

(1) Magnetron control output: D_4

The magnetron is activated with a high-level output, and disabled with a low-level output. Alternate on/off operations are repeated with the designated power (duty) in units of 30 seconds. For instance, the magnetron is activated for a period of 9 seconds and disabled for a period of 21 seconds, when the power setting is 30%. It also provides on/off action for controlling the temperature.

(2) Fan output: D_3

The fan is started as soon as the DEFROST, COOK 1 or COOK 2 process is begun, and is turned off as soon as the stop switch is depressed or the cooking process is completed.

(3) Buzzer output: D_5

There are three buzzer-control outputs.

0.2-second buzzer . . . This buzzer is activated each time a validated key entry is made.

0.5-second buzzer . . . This buzzer is activated each time one stage is completed.

3-second buzzer . . . This buzzer repeats 0.2-second intermittent actuation for a period of 3 seconds when the timer completes its counting or the cooking process is completed.

(4) Fluorescent display tube: S_0~S_7, D_6~D_9, D_2

A large fluorescent display tube can be driven directly with these outputs. (With maximum output voltage of 33V, and maximum of 15mA for the D ports and a maximum of 8mA for the S ports.)

The display is activated dynamically, and its duty is about 1/14, with an on duration of 0.9ms.

The following type of a display is taken into consideration. When indicating the temperature and a "C" is displayed in the least significant column, the colon in the center of the display is not displayed. Also for power display the colon is not displayed, and a "P" is displayed in the least significant column.

(5) LED display: S_0~S_7, D_{10}

Key entry number or the cooking mode is displayed on the LED, and the contents of one or more of the following are displayed: [S. TIME], [DEFR], [COOK 1], [COOK 2], [TIMER], [START], [STOP], and [TEMP].

The LED is activated dynamically, and its duty is about 70%, with an on duration of about 9ms.

(6) Capacitive panel detection outputs, D_0~D_2

Inverted D-port outputs are amplified and supplied to the touch keyboard in order to identify the key depressed in the matrix through the K ports.

Output D_2 is used for displaying the colon on the fluorecent display tube.

4. Key Entries

After depressing a function key, a number key is depressed. Then the data thus entered will be stored in the RAM, after another function key has beed depressed, if no error was detected in the data.

(1) Setting the time

Setup of hours and minutes:

Used to set the CLOCK and S. TIME. Must be set within the range of 1:00~12:59.

Setup of minutes and seconds:

Used to set the TIMER, DEFR, COOK 1 and COOK 2 periods.

Must be set within the range of 1 second~99 minutes and 59 seconds.

Error:

When key entry is made over the above upper limits or more than 6 digits are entered, an error indication (EE:EE) is displayed.

An example of setting the clock operation is shown in the following illustration:

Example of key entry (1)

	KEY	DISPLAY
1ST STEP	[CLOCK]	: 0
2ND STEP	[1]	: 1
3RD STEP	[2]	:12
4TH STEP	[3]	1:23
5TH STEP	[4]	12:34
6TH STEP	[START]	12:34

15

When a key entry error is detected in the fifth or sixth step, an error indication "EE:EE" is displayed, after the CLEAR key has been depressed. Then the data must be reentered. When there is no error in the key entry, the clock operation will start as soon as the start key is depressed.

(2) Setup of duty for the magnetron

The operating power must be set in the following sequence: [POWER] → [DEFR, COOK 1, or COOK 2] → [NUMBERS]. Power duty in the range of 0 ~100% can be used for COOK 1 and COOK 2 operations, but for the DEFR operation the range is 0~50%. Even though the rate is set over 50% for DEFR, a rate of only 50% will be used because of the limit.

Entry of power duty settings 0~90% is made by depressing one number key that is the desired setting to the closest 10%. An entry of 100% is made by depressing the 1 followed by a 0. Deviating from this will cause an error.

Example of key entry (2)

	KEY	DISPLAY
1ST STEP	[POWER]	P
2ND STEP	[COOK 1]	100P
3RD STEP	[2]	20P
4TH STEP	[COOK 2]	: 0

Automatically 100% of the duty is recalled from the memory in the second step, 20% is displayed in the third step, 20% is stored in the RAM in the fourth step, and then the time of the COOK 2 is recalled from the memory. (But only [0] is displayed in this case, because the data for COOK 2 has not yet been entered.

(3) Setup of temperature

The operating temperature must be set in the sequence of [TEMP] → [COOK 1 or COOK 2] → [NUMBERS]. The temperature must be within the range of 35°C ~95°C. Exceeding this range will cause an error.

5. Data Display

(1) Before the start

Data during key entry is displayed in the manner mentioned previously, but this data can be recalled from memory by depressing the appropriate function key when needed for reference.

Example of key entry (3)

	KEY	DISPLAY
1ST STEP	[CLOCK]	11:56
2ND STEP	[TEMP]	c
3RD STEP	[COOK 2]	62c
4TH STEP	[COOK 1]	5:30
5TH STEP	[POWER]	P
6TH STEP	[COOK 2]	60P

In the first step the present time is displayed from the clock. Then the temperature setting for COOK 2 is recalled from memory in the second and third steps. The time setting for COOK 1 is recalled in the fourth step. Then the power setting for COOK 2 is recalled from memory in the fifth and sixth steps.

(2) After the start

After the start key is depressed, the remaining cooking time is displayed, but the following data can be recalled and displayed for 3 seconds.

Power: Depression of the [POWER] key displays the current power setting.

Clock: Depression of the [CLOCK] key displays the time.

Operating temperature: Depression of the [TEMP] key once displays the current operating temperature setting.

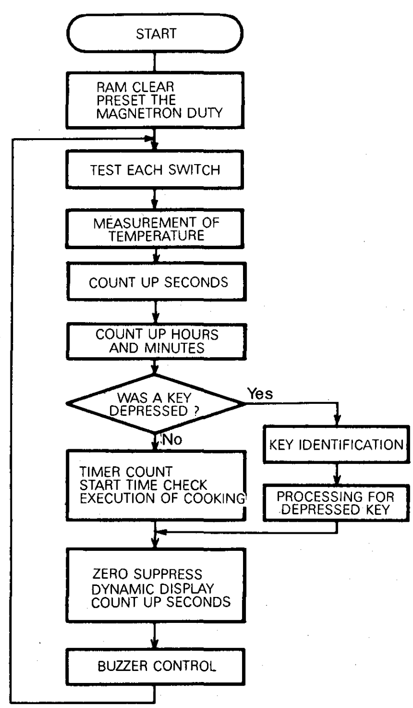
Measured temperature: Depression of the [TEMP] key twice displays of the measured temperature at the present stage.

6. Correction of Data

As the function keys are depressed to recall data, correction of the data can be made by entering the new corrected data after the key operation in the usual manner. To correct the data while in operation, the stop key must first be depressed to stop the operation.

7. General flowchart

A flowchart of the M58840-001P is shown in the following illustration.



8. Routines for Other Applications

Program routines of the M58840-001P that may be suitable for other applications are shown below.

(1) Temperature measurement
After measurement of the temperature, the data, output as H and L signals, is converted to BCD.

(2) Counting seconds
Up to 60 seconds can be counted by supplying the power-supply waveform to the K_{13} port.

(3) Counting hours and minutes
Up to 12 hours can be counted.

(4) Use of touch keyboards
Depression of a touch-keyboard key can be detected.

(5) Key identification
Up to 22 keys can be identified.

(6) Displaying
A fluorescent display tube and LEDs can be displayed dynamically.

(7) Temperature comparison
Temperature comparison can be made to detect a 2°C fall in temperature for temperature control.

(8) 0.5-second flickering
Display "C" or the LED can be flickered in units of 0.5 seconds.

(9) Count of time
The time settings can be decremented each second, and used to terminate or start operations when the count reaches 0.

(10) Buzzer control
Buzzer actuation can be controlled for a duration of 0.2, 0.5 or 3 seconds. The 3-second actuation is on-off at 0.2-second intervals.

(11) Time monitoring
Time can be monitored and used to terminate or start operations when the time setting is reached.

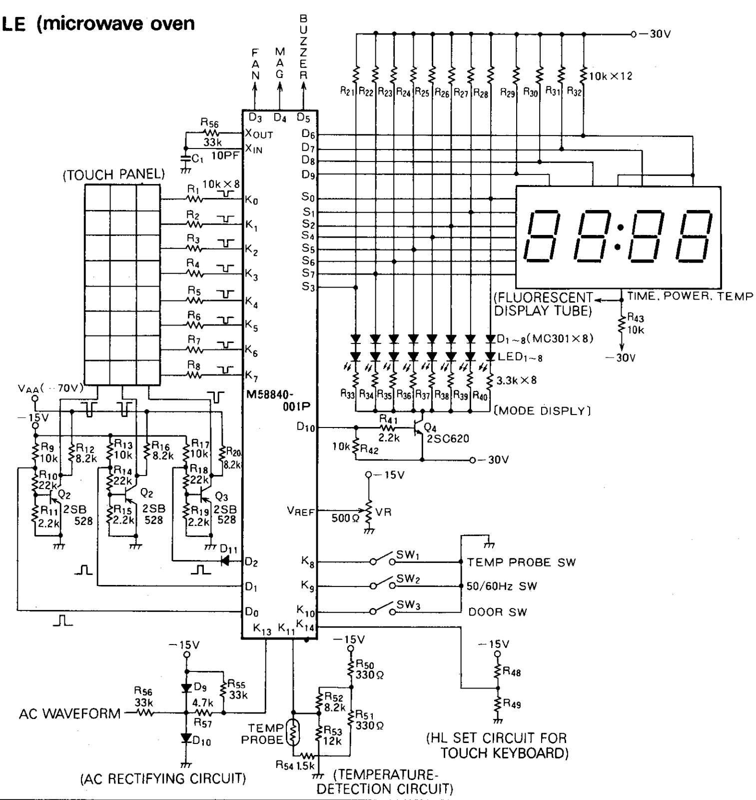
9. Typical control circuit of a microwave oven

A typical example of a microwave oven circuit is shown.

Details of input and output performance are as previously described. Please refer to the information provided for the PCA0402 in regard to capacitive touch-keyboard operation. The diode D_{11} is provided to prevent counterflow because D_2 is also used for the colon output and display. The temperature-detection circuit K_{11} compensates for the nonlinear output of the temperature probe and facilitates easy temperature conversion.

The touch-keyboard interface and the A/D conversion circuit are contained in the M58840-XXXP. The wide range of S ports and high maximum output voltage of the S and D ports simplify circuit design. This results in cost reduction, improved performance and improved reliability because fewer parts are required. The use of fewer parts also helps miniaturization.

APPLICATIONS EXAMPLE (microwave oven M58840-001P)



1. CODE-CONVERSION PROGRAMS

There are 4 code-conversion programs for conversions between hexadecimal numbers and their corresponding ASCII code in binary notation. Details of these programs are given below.

Table 1 Correspondence of number formats

Hexadecimal symbols	Machine language binary number	ASCII code in binary notation for hexadecimal symbols
0	0000	00110000
1	0001	00110001
2	0010	00110010
3	0011	00110011
4	0100	00110100
5	0101	00110101
6	0110	00110110
7	0111	00110111
8	1000	00111000
9	1001	00111001
A	1010	01000001
B	1011	01000010
C	1100	01000011
D	1101	01000100
E	1110	01000101
F	1111	01000110

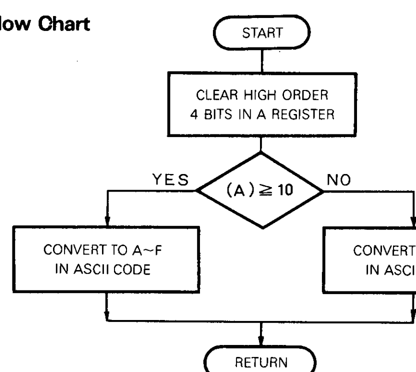
1.1 Binary (4 Bits) to ASCII (1 Character) Conversion (BTA)

This program converts the low-order 4 bits in the A register (a hexadecimal number 0~F) to the corresponding 8-bit ASCII-coded hexadecimal symbol '0'~'F'. The result is retained in the A register. Registers B, C, D, H and L are not affected.

Register Status

Register	Contents at start	Contents at return
A	Binary number to be converted in the low-order 4 bits	8-bit ASCII code
B, C, D, E, H and L		Contents at start

Flow Chart



Program Listing

```
*
*  ***  SUB(BTA)  */BINARY TO ASCII
*
BTA     ANI    0F#
        CPI    10
        JNC    B1
        ADI    48
        RET
B1      ADI    55
        RET
```

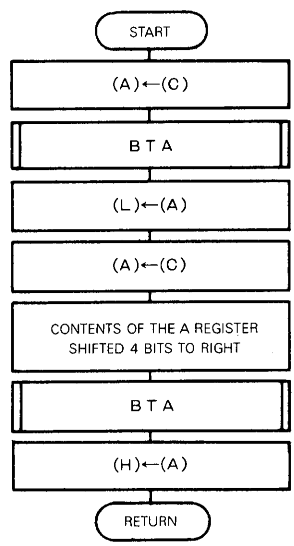
1.2 Binary (8 Bits) to ASCII (2 Characters) Conversion (BTA 2)

This program converts the 8 bits in the C register (a 2-digit hexadecimal number 00~FF) to the 2 corresponding 8-bit ASCII-coded hexadecimal symbols '0'~'F'. The results are retained in registers H (high order) and L (low order). The B, D and E registers are not affected.

Register Status

Register	Contents at start	Contents at return
A		8-bit ASCII code for the high-order hexadecimal symbol
C	Binary number to be converted	Binary number to be converted
H		8-bit ASCII code for the high-order hexadecimal symbol
L		8-bit ASCII code for the low-order hexadecimal symbol
B, D and E		Contents at start

Flow Chart



Program Listing

```
*
*  ***  SUB(BTA2)  *
*  *BINARY TO TWO ASCII CHARACTERS
*
BTA2    MOV    A,C
        CALL   BTA
        MOV    L,A
        MOV    A,C
        RRC
        RRC
        RRC
        RRC
        CALL   BTA
        MOV    H,A
        RET
```


MITSUBISHI ELECTRIC

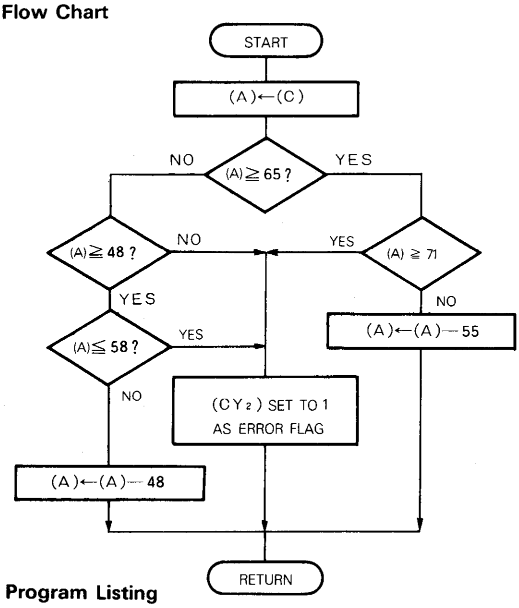
1.3 ASCII (1 Character) to Binary (4 Bits) Conversion (ATB)

This program converts the 8-bit ASCII code in the C register (a hexadecimal symbol '0'~'F') to a 4-bit binary number 0000~1111. The result is retained in the low-order 4 bits of the A register. If the C register contains a code for a character other than a hexadecimal symbol 0~F, it is recognized as an error; the carry flip-flop is set, and the program is exited. Registers B, D, E, H and L are not affected.

Register Status

Register	Contents at start	Contents at return
A		Hexadecimal number in binary form in the low order 4 bits
D	ASCII coded hexadecimal symbol to be converted	ASCII coded hexadecimal symbol to be converted
B, D, E, H and L		Contents at start

Flow Chart



Program Listing

```
*
*  ***  SUB(ATB)    */ASCII TO BINARY
*
ATB     MOV   A,C
        CPI   65
        JC    A1
        CPI   71
        JNC   A3
        SUI   55
        ANA   A
        RET
A1      CPI   48
        JC    A3
        CPI   58
        JNC   A3
A2      SUI   48
        ANA   A
        RET
A3      STC
        RET
```

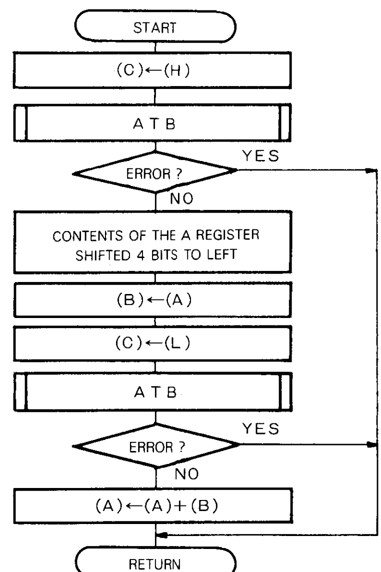
1.4 ASCII (2 Characters) to Binary (8 Bits) Conversion (ATB 2)

This program converts the two 8-bit ASCII codes in the H and L registers (2 hexadecimal symbols '0'~'F', high order in the H register and low order in the L register) to an 8-bit binary number ($0\sim255_{10}$). The result is retained in the A register. If the H or L register contains a code for a character other than a hexadecimal symbol '0'~'F', it is recognized as an error; the carry flip-flop is set, and the program is exited. The D and E registers are not affected.

Register Status

Register	Contents at start	Contents at return
A		8-bit binary number (2 hexadecimal digits)
B		4-bit binary number in the high-order 4-bits conversion of high-order hexadecimal symbol
C		Low-order ASCII coded hexadecimal symbol to be converted
H	High-order ASCII coded hexadecimal symbol to be converted	High-order ASCII coded hexadecimal symbol to be converted
L	Low-order ASCII coded hexadecimal symbol to be converted	Low-order ASCII coded hexadecimal symbol to be converted
D and E		Contents at start

Flow Chart



Program Listing

```
*
*  ***  SUB(ATB2)  *
*  /TWO  ASCII  CHARACTERS  TO  BINARY
*
ATB2    MOV   C,H
        CALL  ATB
        RC
        RLC
        RLC
        RLC
        RLC
        MOV   B,A
        MOV   C,L
        CALL  ATB
        RC
        ADD   B
        RET
```

2. SORTING PROGRAM (SORT)

This program sorts records (1 byte in length) in descending order. Up to 65 535 records can be sorted. The binary number 255_{10} cannot be used as data because it is reserved for the end-of-data mark. This data is stored in descending order according to its rank.

The program sorts by comparing a data item with all other data items, thus determining its rank. The data is then stored in descending order according to that rank.

This program can also recall the data associated with any rank. If the rank k ($1 \leq k \leq 65\ 535$) is stored in memory locations ORD and ORD+1, the 1-byte data associated with that rank is stored in the A register, and then control is returned to the user's program. If k is specified as zero, the A register is reset to zero and control is returned to the user's program.

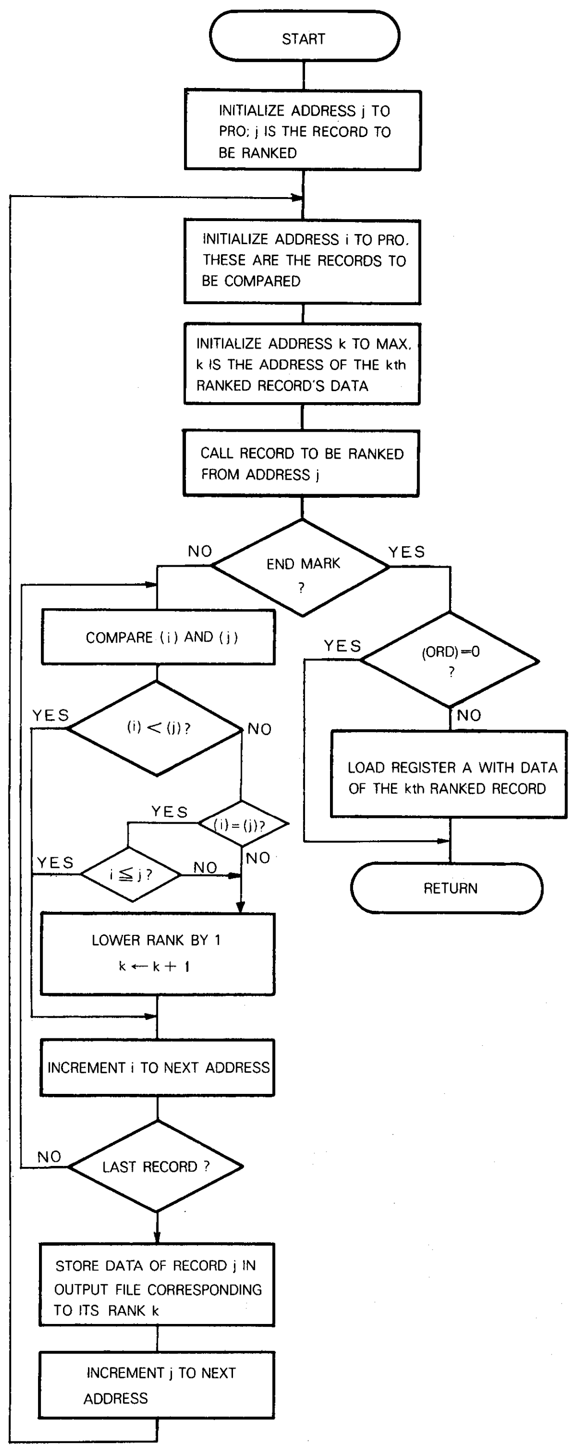
Register Status

Register	Use during execution	Contents changed at return
A	Calculates and recalls data of rank k	yes
B	Storage for data being compared	yes
C	Not used	no
D	Memory address for storing data after ranking	yes
E		yes
H	Memory address of data to be ranked	yes
L		yes

Symbolic Memory Address

	Symbolic address	Use during execution	No. of bytes	Contents changed at return
User's area	ORD	k (the rank of data to be recalled)	2	no
	PRO	Storage area for records to be sorted (PRO is the first address)	n+1	no
	MAX	Storage area for sorted data (MAX is the first address)	n+1	yes
Control area	DADD	Address in PRO of record being sorted	2	no
	RADD	Address in MAX for storing result	2	no
	M1	Address of record to be ranked	2	yes
	M2	Address of record being compared	2	yes
	COUNT	Counter for number of records	2	yes

Flow Chart



Program Listing

```
*
*  ***  SUBROUTINE (SORT)  ***
*
*  /SORTING PROGRAM
*
          NAM     SORT .......................①
*  <USERS DATA AREA>  * ......................②
ORD       DADR    0
PRO       DEF     1
          DEF     55
          DEF     100 ........................③
          DEF     15
           :

          DEF     FF# ........................④
MAX       BSS     MAX-PRO ....................⑤
*
*  <CONTROL DATA AREA>
DADD      DADR    PRO
RADD      DADR    MAX
M1        DADR    0
M2        DADR    0
COUNT     DADR    0
*
*  ***  PROGRAM START  ***  *
*
          ROM     ............................⑥
SORT      LHLD    DADD
          XCHG
          LXI     H,0
R1        SHLD    COUNT
          LHLD    RADD
          XCHG
*
R2        MOV     B,M
          CPI     FF#
          JZ      R8
          SHLD    M1
          LHLD    DADD
R3        MOV     A,M
          SHLD    M2
          CPI     FF#
          JZ      R7
          CMP     B
          JC      R6
          JNZ     R5
          PUSH    PSW
          LHLD    M1
          LDA     M2
          SUB     L
          LDA     M2+1
          JC      R4
          POP     PSW
          JMP     R6
R4        POP     PSW
R5        INX     D
R6        INX     H
          JMP     R3
*
R7        LHLD    M1
          INX     H
          MOV     A,B
          STAX    D
          XCHG
          LHLD    COUNT
          INX     H
          JMP     R1
*
R8        LHLD    COUNT
          XCHG
          LHLD    RADD
          DAD     D
          MOV     M,A
*
          LHLD    ORD
          MOV     A,L
          ORA     H
          JZ      R9
          DCX     H
          XCHG
          LHLD    RADD
          DAD     D
          MOV     A,M
R9        RET
```

Explanation Keyed to Program Listing

① The program name is defined as 'SORT'.

② If column 1 of a statement is '*', it is considered a comment.

③ Defines the value of data.

④ The '#' in FF# indicates that FF is a hexadecimal number.

⑤ Reserves a region to store the results.

⑥ The above program is defined as a RAM region because its contents are variable at time of execution, and this is a ROM region because its contents are fixed.

APPLICATION OF MELCS 8/2 SINGLE-BOARD COMPUTER

(PCA 0801) IN DATA TRANSMISSION THROUGH A MASTER-SLAVE MULTICOMPUTER SYSTEM

DESCRIPTION

Three PCA0801 single-board computers are connected to form a master-slave microcomputer data-transmission system. Such a system contributes significantly to reducing the load on the host computer and to improving the operational efficiency and functions of the system. This is an example of a mode 2 application of the M5L8255AP programmable peripheral interface (PPI).

FUNCTION

One of the three PCA0801s serves as the master computer, and the other two as the slave computers that complete the system. When the No. 1 PPI (C.W.$=03_{16}$) is set to mode 2, data is transmitted between the master and either of the slaves using the I/O port PA as a bidirectional data bus.

OPERATION

The master computer, storing 200 bytes of the transmission data within its No. 2 EPROM (M5L2708K), starts to transmit that data to the No. 1 slave computer via the I/O port PA (PA_0~PA_7). After receiving the data, the slave computer inverts the data and stores it in its RAM (M5L2111AP). This inverted data is then sent back to the master computer, after which it is stored in the master computer's RAM.

The master computer now starts to transmit 200 bytes of the RAM data to the No. 2 slave computer, where the data is inverted and stored in the RAM to be sent back to the master computer.

The master computer, having completed storage of the data in its RAM, executes an inspection routine for the stored data, and compares the 200-byte contents of the EPROM and the RAM for discrepancies.

If all the data is correct, LED 1, which acts as an indicator, is turned on. If not, LED 2 is lit, and execution is terminated.

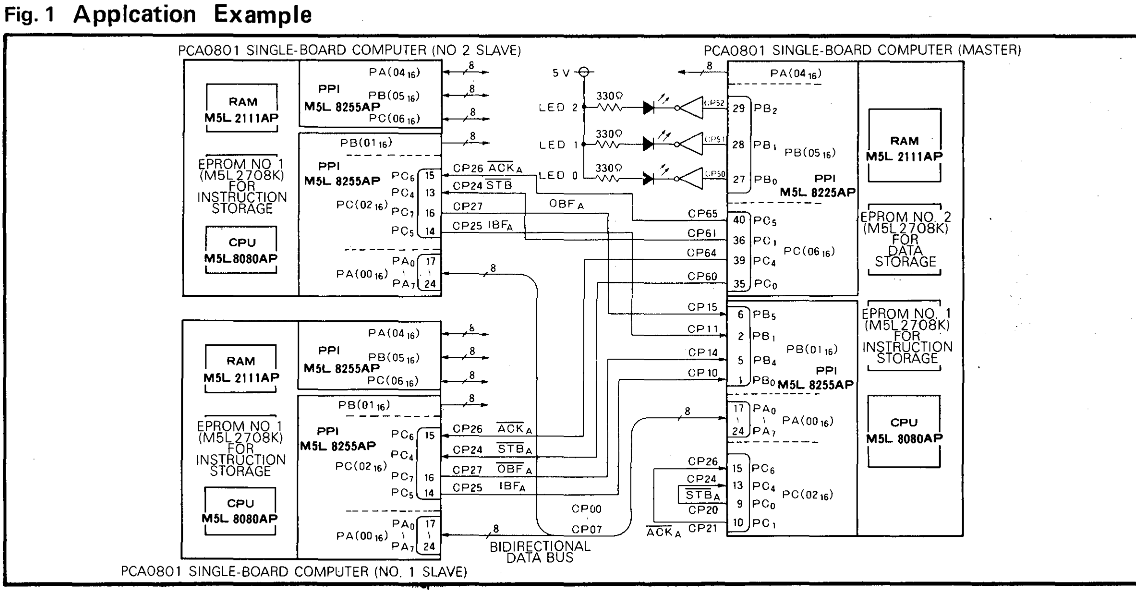
The operational status of the LEDs (on or off), and their significance, are shown in Table 1. These status indications are shown in the sequence of CPU progress, so that the operating status of the master computer may be readily recognized from the combination of LED 0~LED 2 indicators.

Table 1 Status as Indicated by the LED Display

CPU Sequence	LED 0	LED 1	LED 2	Description of the status indicated
	0	0	0	System is not transmitting data
	0	0	1	Data is being started between the master and slave computer No. 1.
	1	0	0	Data is being transmitted between the master and slave computer No. 1.
	0	0	0	Data is completed between the master and slave computer No. 1.
	0	0	1	System is in the idle condition, with no transmission between the master and slave computer No. 2.
	1	1	1	Data is being started between the master and slave computer No.2.
	0	1	0	Data transmission has been completed, having transmitted the data correctly.
	0	0	1	Data transmission has completed, but a transmission error has been found.

Note 1 : "ON" indicates where the LED turns on, and "OFF" where the LED turns off.
2 : The slave computers, No. 1 and No. 2, must be in operation prior to the engagement of the master computer.

Fig. 1 Applcation Example



MITSUBISHI ELECTRIC

Fig. 2 Master microcomputer flow chart

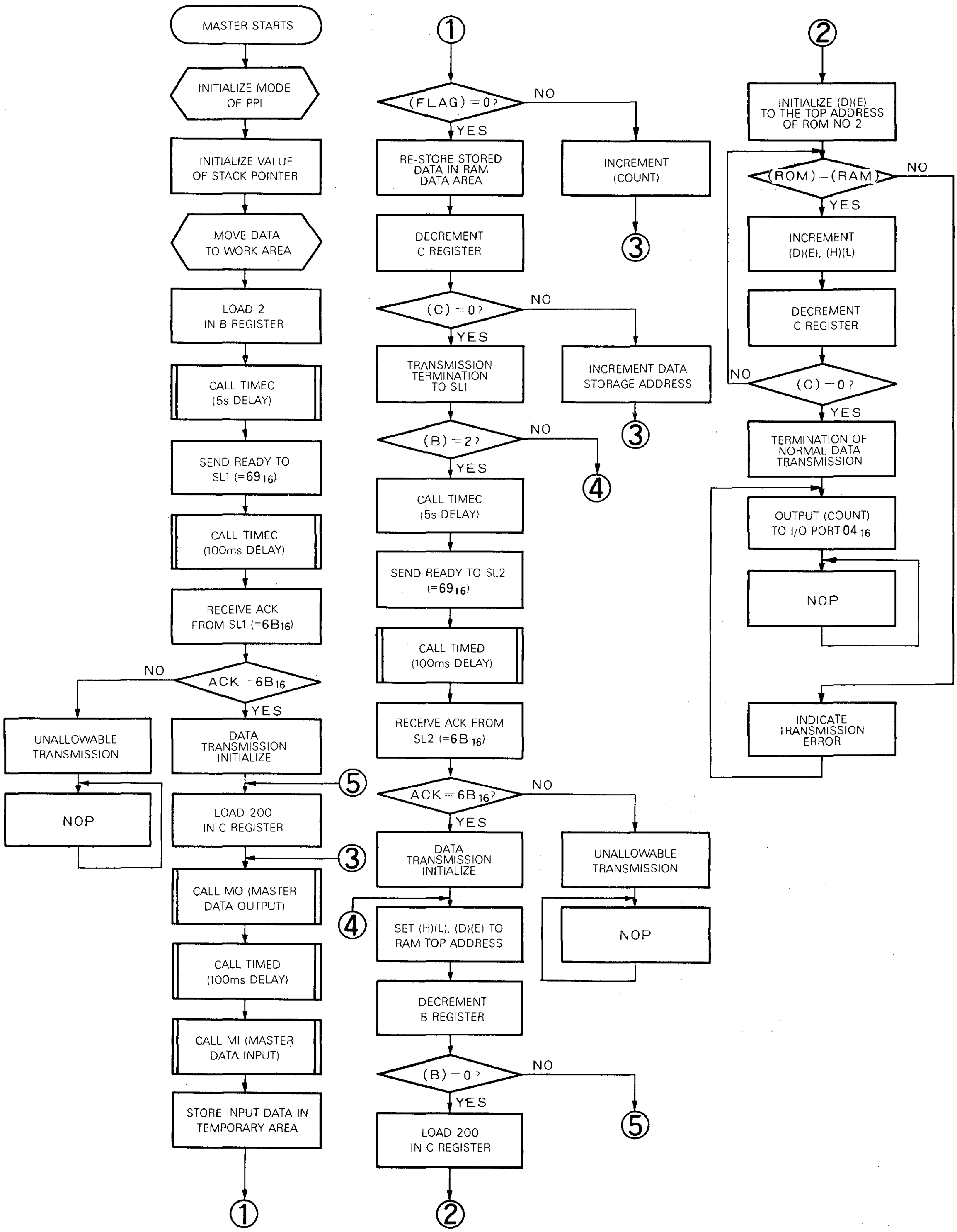
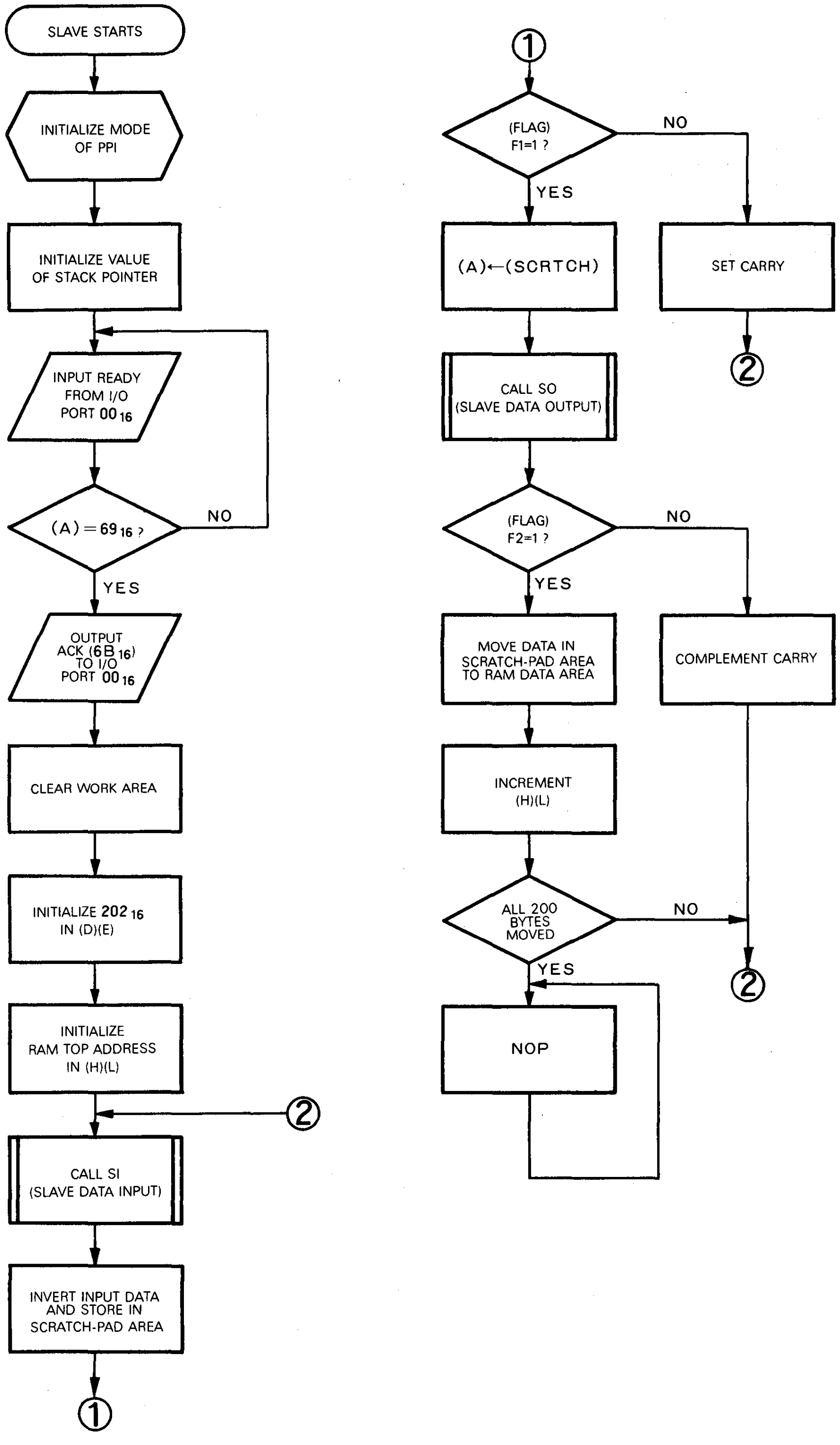


Fig. 3 Slave microcomputer flow chart



MITSUBISHI ELECTRIC

MASTER MICROCOMPUTER MAIN PROGRAM LIST

```
**CROSS ASSEMBLER OF 8-BIT MICROPROCESSOR

*** MASTER MICROCOMPUTER MAIN PROGRAM ***
0400           ROM2ST EQU   0400#
4000           RAMST  EQU   4000#
40C8           DSTNT1 EQU   40C8#
40CA           DSTNT2 EQU   40CA#
40CC           FLAG   EQU   40CC#
40CD           COUNT  EQU   40CD#
40CE           TEMPRY EQU   40CE#
*
*
0000                  ORG   0000#
0000 3EC2      MASTER MVI   A,C2#
0002 D303             OUT   03#
0004 3E01             MVI   A,01#
0006 D303             OUT   03#
0008 3E03             MVI   A,03#
000A D303             OUT   03#
000C 3E80             MVI   A,80#
000E D307             OUT   07#
0010 3E00             MVI   A,00#
0012 D305             OUT   05#
0014 3EFF             MVI   A,FF#
0016 D306             OUT   06#
0018 3EAA             MVI   A,AA#
001A D304             OUT   04#
001C 31FF40           LXI   SP,40FF#
001F 0602             MVI   B,2
0021 21DD01           LXI   H,X
0024 22C840           SHLD  DSTNT1
0027 21E501           LXI   H,Y
002A 22CA40           SHLD  DSTNT2
002D AF               XRA   A
002E 32CE40           STA   TEMPRY
0031 32CD40           STA   COUNT
0034 3E59             MVI   A,59#
0036 32CC40           STA   FLAG
0039 CD1201           CALL  TIMEC
003C 3E69             MVI   A,69#
003E D300             OUT   00#
0040 3E02             MVI   A,02#
0042 D303             OUT   03#
0044 3E00             MVI   A,00#
0046 D307             OUT   07#
0048 3C               INR   A
0049 D307             OUT   07#
004B 3E03             MVI   A,03#
004D D303             OUT   03#
004F CD2E01           CALL  TIMED
0052 3E08             MVI   A,08#
0054 D307             OUT   07#
0056 3E00             MVI   A,00#
0058 D303             OUT   03#
005A 3C               INR   A
005B D303             OUT   03#
005D 3E09             MVI   A,09#
005F D307             OUT   07#
0061 DB00             IN    00#
0063 D66B             SUI   6B#
0065 C20B01           JNZ   NOCOMC
0068 3E01             MVI   A,01#
006A D305             OUT   05#
006C 210004           LXI   H,ROM2ST
006F 110040           LXI   D,RAMST
0072 0EC8      RPT2   MVI   C,200
0074 7E        RPT1   MOV   A,M
0075 CD5501           CALL  MC
0078 CD2E01           CALL  TIMED
007B CD6001           CALL  MI
007E 32CE40           STA   TEMPRY
0081 3ACC40           LDA   FLAG
0084 B7               ORA   A
0085 C20101           JNZ   SUM
0088 3ACE40           LDA   TEMPRY
008B EB               XCHG
008C 77               MOV   M,A
008D 0D               DCR   C
008E EB               XCHG
008F C2FC00           JNZ   YET
0092 3E00             MVI   A,00#
0094 D305             OUT   05#
0096 78               MOV   A,B
0097 FE02             CPI   02#
0099 C2CF00           JNZ   S2END
009C CD1201           CALL  TIMEC
009F 3E69             MVI   A,69#
00A1 D300             OUT   00#
00A3 3E02             MVI   A,02#
00A5 D303             OUT   03#
00A7 3E02             MVI   A,02#
00A9 D307             OUT   07#
00AB 3C               INR   A
00AC D307             OUT   07#
00AE 3E03             MVI   A,03#
00B0 D303             OUT   03#
00B2 CD2E01           CALL  TIMED
00B5 3E0A             MVI   A,0A#
00B7 D307             OUT   07#
00B9 3E00             MVI   A,00#
00BB D303             OUT   03#
00BD 3C               INR   A
00BE D303             OUT   03#
00C0 3E0B             MVI   A,0B#
00C2 D307             OUT   07#
00C4 DB00             IN    00#
00C6 D66B             SUI   6B#
00C8 C20B01           JNZ   NOCOMC
00CB 3E07             MVI   A,07#
00CD D305             OUT   05#
00CF 210040    S2END  LXI   H,RAMST
00D2 54               MOV   D,H
00D3 5D               MOV   E,L
00D4 05               DCR   B
00D5 C27200           JNZ   RPT2
00D8 0EC8             MVI   C,200
00DA 110004           LXI   D,ROM2ST
00DD 1A        SCAN   LDAX  D
00DE BE               CMP   M
00DF C2F500           JNZ   TRMERR
00E2 13               INX   D
00E3 23               INX   H
00E4 0D               DCR   C
00E5 C2DD00           JNZ   SCAN
00E8 3E02             MVI   A,02#
00EA D305             OUT   05#
00EC 3ACD40    NO2    LDA   COUNT
00EF D304             OUT   04#
00F1 00        NO1    NOP
00F2 C3F100           JMP   NO1
*
00F5 3E04      TRMERR MVI   A,04#
00F7 D305             OUT   05#
00F9 C3EC00           JMP   NO2
*
00FC 23        YET    INX   H
00FD 13               INX   D
00FE C37400           JMP   RPT1
*
*** NO-PASS SUM ***
*
```

```
0101 3ACD40 SUM    LDA  COUNT
0104 3C            INR  A
0105 32CD40        STA  COUNT
0108 C37400        JMP  RPT1
*
*
*** NOCOMMUNICATE ***
*
010B 3E04   NOCOMC MVI  A,04#
010D D305          OUT  05#
010F C3F100        JMP  N01
*
*** SUBROUTINE TIMEC ***
*
0112 1E32   TIMEC  MVI  E,50
0114 CD2E01 TIMEC1 CALL TIMED
0117 1D            DCR  E
0118 CA1E01        JZ   TIMEC2
011B C31401        JMP  TIMEC1
011E C9     TIMEC2 RET
*
*
*
*** SUBROUTINE TIMEF ***
011F D5     TIMEF  PUSH D
0120 1E0A          MVI  E,10
0122 CD2E01 TIMEF1 CALL TIMED
0125 1D            DCR  E
0126 CA2C01        JZ   TIMEF2
0129 C32201        JMP  TIMEF1
012C D1     TIMEF2 POP  D
012D C9            RET
*
*** SUBROUTINE TIMED ***
*
012E F5     TIMED  PUSH PSW
012F C5            PUSH B
0130 D5            PUSH D
0131 E5            PUSH H
0132 1614          MVI  D,20
0134 0E14          MVI  C,20
0136 06C8   TIMED6 MVI  B,200
0138 3EC8          MVI  A,200
013A C33D01 TIMED1 JMP  TIMED2
013D C34001 TIMED2 JMP  TIMED3
0140 05     TIMED3 DCR  B
0141 3D            DCR  A
0142 CA4801        JZ   TIMED4
0145 C33A01        JMP  TIMED1
0148 15     TIMED4 DCR  D
0149 0D            DCR  C
014A CA5001        JZ   TIMED7
014D C33601        JMP  TIMED6
0150 E1     TIMED7 POP  H
0151 D1            POP  D
0152 C1            POP  B
0153 F1            POP  PSW
0154 C9            RET
*
*
*** SUBROUTINE MO ***
*
0155 D300   MO     OUT  00#
0157 D5            PUSH D
0158 110202        LXI  D,202#
015B CD6B01        CALL DECODO
015E D1            POP  D
015F C9            RET
*
*
*** SUBROUTINE MI ***
*
0160 D5     MI     PUSH D
0161 110202        LXI  D,202#
0164 CDA201        CALL DECODI
0167 DB00          IN   00#
0169 D1            POP  D
016A C9            RET
*
*
*** SUBROUTINE DECODO ***
*
016B E5     DECODO PUSH H
016C D5            PUSH D
016D 2AC840        LHLD DSTNT1
0170 DB01   MIBF   IN   01#
0172 A6            ANA  M
0173 C29B01        JNZ  MIBFP
0176 3E02          MVI  A,02#
0178 D303          OUT  03#
017A 23            INX  H
017B 7E            MOV  A,M
017C D307          OUT  07#
017E 3C            INR  A
017F D307          OUT  07#
0181 3E03          MVI  A,03#
0183 D303          OUT  03#
0185 79            MOV  A,C
0186 D601          SUI  01#
0188 CA9701        JZ   FINE1
018B 2B            DCX  H
018C 22C840 STORE1 SHLD DSTNT1
018F 3E24          MVI  A,24#
0191 32CC40        STA  FLAG
0194 D1     NOIBF  POP  D
0195 E1            POP  H
0196 C9            RET
0197 23     FINE1  INX  H
0198 C38C01        JMP  STORE1
019B 15     MIBFP  DCR  D
019C C27001        JNZ  MIBF
019F C39401        JMP  NOIBF
*
*
*** SUBROUTINE DECODI ***
*
01A2 E5     DECODI PUSH H
01A3 D5            PUSH D
01A4 2ACA40        LHLD DSTNT2
01A7 DB01   MOBF   IN   01#
01A9 A6            ANA  M
01AA C2D601        JNZ  MOBFP
01AD 23            INX  H
01AE 7E            MOV  A,M
01AF D307          OUT  07#
01B1 3E00          MVI  A,00#
01B3 D303          OUT  03#
01B5 3E01          MVI  A,01#
01B7 D303          OUT  03#
01B9 7E            MOV  A,M
01BA 3C            INR  A
01BB D307          OUT  07#
01BD 79            MOV  A,C
01BE D601          SUI  01#
01C0 CAD201        JZ   FINE2
01C3 2B            DCX  H
01C4 22CA40 STORE2 SHLD DSTNT2
01C7 3ACC40        LDA  FLAG
01CA D624          SUI  24#
01CC 32CC40        STA  FLAG
01CF D1     NOOBF  POP  D
01D0 E1            POP  H
```

```
01D1 C9                RET
01D2 23      FINE2     INX    H
01D3 C3C401            JMP    STORE2
01D6 1D      MOBFP     DCR    E
01D7 C2A701            JNZ    MOBF
01DA C3CF01            JMP    NOOBF
*
*
*   SELECTIVE CHARACTER O TEIGI SURU *
01DD 01      X         DEF    01#
01DE 00                DEF    00#
01DF 02                DEF    02#
01E0 02                DEF    02#
01E1 04                DEF    04#
01E2 04                DEF    04#
01E3 08                DEF    08#
01E4 06                DEF    06#
01E5 10      Y         DEF    10#
01E6 08                DEF    08#
01E7 20                DEF    20#
01E8 0A                DEF    0A#
01E9 40                DEF    40#
01EA 0C                DEF    0C#
01EB 80                DEF    80#
01EC 0E                DEF    0E#
0000                   END    MASTER
```

SLAVE MICROCOMPUTER MAIN PROGRAM LIST

```
**CROSS ASSEMBLER OF 8-BIT MICROPROCESSOR

*** SLAVE MICROCOMPUTER MAIN PROGRAM ***
*
*
4000         RAMST     EQU    4000#
40D0         SCRTCH    EQU    40D0#
40D1         F1        EQU    40D1#
40D2         F2        EQU    40D2#
*
0000                   ORG    0000#
0000 3EC0    SLAVE     MVI    A,C0#
0002 D303              OUT    03#
0004 3E81              MVI    A,81#
0006 D307              OUT    07#
0008 31FF40            LXI    SP,4CFF#
000B DB00    WAIT      IN     00#
000D D669              SUI    69#
000F C20B00            JNZ    WAIT
0012 3E6B              MVI    A,6B#
0014 D300              OUT    00#
0016 AF                XRA    A
0017 32D140            STA    F1
001A 32D240            STA    F2
001D 110202            LXI    D,202#
0020 210040            LXI    H,RAMST
0023 CD5500  BACK1     CALL   SI
0026 2F                CMA
0027 32D040            STA    SCRTCH
002A 3AD140            LDA    F1
002D B7                ORA    A
002E CA4D00            JZ     NOPAS1
0031 3AD040            LDA    SCRTCH
0034 CD7100            CALL   SO
0037 3AD240            LDA    F2
003A B7                ORA    A
003B CA5100            JZ     NOPAS2
003E 3AD040            LDA    SCRTCH
0041 77                MOV    M,A
0042 23                INX    H
0043 7D                MOV    A,L
0044 FEC8              CPI    C8#
0046 DA2300            JC     BACK1
0049 00      NO        NOP
004A C34900            JMP    NO
*
*** NOPASS 1 ***
*
004D 37      NOPAS1    STC
004E C32300            JMP    BACK1
*
*** NOPASS 2 ***
0051 3F      NOPAS2    CMC
0052 C32300            JMP    BACK1
*
                       EJE
*
*** SUBROUTINE SI ***
*
0055 D5      SI        PUSH   D
0056 DB02    SIBF      IN     02#
0058 E620              ANI    20#
005A C26700            JNZ    SIN
005D 15                DCR    D
005E C25600            JNZ    SIBF
0061 AF                XRA    A
0062 32D140            STA    F1
0065 D1      SIND      POP    D
0066 C9                RET
0067 3E01    SIN       MVI    A,01#
0069 32D140            STA    F1
006C DB00              IN     00#
006E C36500            JMP    SIND
*
*
*** SUBROUTINE SO ***
*
0071 F5      SO        PUSH   PSW
0072 D5                PUSH   D
0073 DB02    SOBF      IN     02#
0075 E680              ANI    80#
0077 C28500            JNZ    SOUT
007A 1D                DCR    E
007B C27300            JNZ    SOBF
007E AF                XRA    A
007F 32D240            STA    F2
0082 D1                POP    D
0083 F1                POP    PSW
0084 C9                RET
0085 3E01    SOUT      MVI    A,01#
0087 32D240            STA    F2
008A D1                POP    D
008B F1                POP    PSW
008C D300              OUT    00#
008E C9                RET
0000                   END    SLAVE
```

CONTACT ADDRESSES FOR FURTHER INFORMATION

JAPAN
Electronics Marketing Division
Mitsubishi Electric Corporation
2-3, Marunouchi 2-chome
Chiyoda-ku, Tokyo 100, Japan
Telex: 24532 MELCO J
Telephone: (03) 218-3473
(03) 218-3499

HONG KONG
Ryoden Electric Engineering Co., Ltd.
22nd fl., Leighton Centre
77, Leighton Road
Causeway Bay, Hong Kong
Telex: 73411 RYODEN HX
Telephone: (5) 790-7021

TAIWAN
Mitsubishi Electric Corporation
Taipei Representative Office
Room 1303, 13th fl., Huei Fong Bldg.
27, Sec. 3, Chung Shan N. Road
Taipei, R.O.C.
Telex: 11211 MITSUBISHI
Telephone: (597) 3111

U.S.A.
Mitsubishi Electronics America, Inc.
1230 Oakmead Parkway
Suite 206 Sunnyvale CA 94086 U.S.A.
Telex: 172296 MELA SUVL
Telephone: (408) 730-5900

Mitsubishi Electronics America, Inc.
2200 West Artesia Blvd.
Compton CA 90220, U.S.A.
Telex: 698246 MELA CMTN
Telephone: (213) 979-6055

Mitsubishi Electronics America, Inc.
200 Unicorn Park Drive
Woburn, MA 01801, U.S.A.
Telex: 951796 MELASB WOBN
Telephone: (617) 938-1220

WEST GERMANY
Mitsubishi Electric Europe GmbH
Brandenburger Str. 40
4030 Ratingen, West Germany
Telex: 8585070 MED D
Telephone: (02102) 4860

U.K.
Mitsubishi Electric (U.K.) Ltd.
Polycherome House Sandown Road
Watford, Hearts, U.K.
Telex: 927908
Telephone: (923) 37334

AUSTRALIA
Melco Australia Pty. Ltd.
33rd Level, Australia Square,
Sydney, N.S.W., 2000, Australia
P.O. Box H129, Australia Square
Telex: MESYO AA 26614
Telephone: (232) 6277

1982 MITSUBISHI DATA BOOK LSI

MITSUBISHI ELECTRIC CORPORATION
HEAD OFFICE: MITSUBISHI DENKI BLDG., MARUNOUCHI, TOKYO 100. TELEX: J24532 CABLE: MELCO TOKYO

Revised publication, effective Dec. 1981,
superseding publication H-C4024-B of Apr. 1981
Specifications subject to change without notice

H-C4024-C KI-8201 Printed in Japan (TOT)