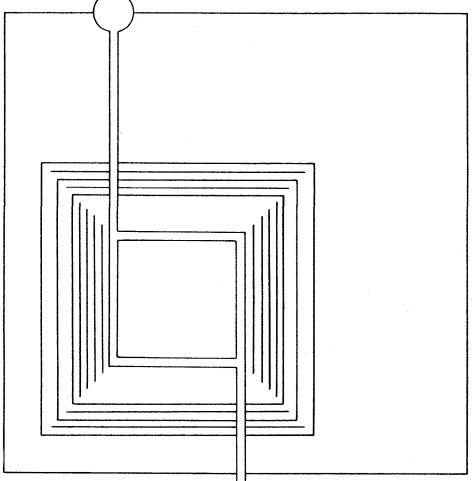
1980

MITSUBISHI LSI DATA BOOK



MITSUBISHI ELECTRIC

All values shown in this catalogue are subject to change for product improvement.

The information, diagrams and all other data included herein are believed to be correct and reliable. However, no responsibility is assumed by Mitsubishi Electric Corporation for their use, nor for any infringements of patents or other rights belonging to third parties which may result from their use.

MELPS and MELCS are registered trademarks of Mitsubishi Electric Corporation.

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.

The editors of the 1979 Mitsubishi LSI Data Book have taken the utmost care to assure that this guide will serve as a useful reference. We have, for example, adopted a new system of type designation that will aid in identifying interchangeability with the devices of other manufacturers.

We have also added a number of new products, including peripheral LSIs for the M5L8085AP and S MELPS 85 microprocessor, a single-chip, 4-bit microcomputer, IC memories, baseboard computers and their development and support systems, along with additional original 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, 1980

Kimio Sato, General Manager
Semiconductors Division
Mitsubishi Electric Corporation

CONTENTS

CONTENTS

CONTENTS

13 MICROCOMPUTER SOFTWARE

14 APPLICATIONS

INDEXES 1

INDEX BY FUNCTION

Type (Note 1)	Former designation	Circuit function and organization	Application notes	Structure (Note 2)	Ambient operating temp Ta (°C)
Static RAMs					
M5L 2101AP,S-2 **M5L 2101AP,S** **M5L 2101AP,S-4**	— — M58721P, S	1024-Bit (256×4) Static RAM	Separate data input/output, OD terminal, 2 chip-selects	N, Si, ED	0~70
M5L 2111AP,S-2 **M5L 2111AP,S** **M5L 2111AP,S-4**	— — M58722P, S	1024-Bit (256×4) Static RAM	Common data input/output, OD terminal, 2 chip-selects	N, Si, ED	0~70
M5L 2112AP,S-2 **M5L 2112AP,S** **M5L 2112AP,S-4**	— — M58723P, S	1024-Bit (256×4) Static RAM	Common data input/output	Ni, Si, ED	0~70
M5L 2102AP,S-4	M58751P, S	1024-Bit (1024×1) Static RAM		N, Si, ED	0~70
M5L 2114LP,S-2 **M5L 2114LP,S-3** **M5L 2114LP,S**	M58724P, S-2 M58724P, S-3 M58724P, S	4096-Bit (1024×4) Static RAM	Common data input/output	N, Si, ED	0~70
M5T 4044P,S-20 **M5T 4044P,S-30** **M5T 4044P,S-45**	M58754P, S-2 M58754P, S-3 M58754P, S	4096-Bit (4096×1) Static RAM	Separate data input/output	N, Si, ED	0~70
Dynamic RAMs					
M5L 2107BP,S	M58755P, S-1	4096-Bit (4096×1) Dynamic RAM		N, Si	0~70
M5K 4116P,S-2 **M5K 4116P,S-3** **M5K 4116P,S-4**	M58759P, S-15 M58759P, S-20 M58759P, S-25	16,384-Bit (16384×1) Dynamic RAM		N, Si	0~70
CMOS Static RAMs					
M5L 5101LP-1	M58980P	1024-Bit (256×4) CMOS Static RAM	Common data input/output	C, Si	0~70
M58981S-45	—	4096-Bit (1024×4) CMOS Static RAM	Common data input/output, same pin configuration as M5L2114LP, S	C, Si	0~70
Non-Volatile Static RAM					
M58656S	—	1024-Bit (256×4), Non-Volatile Static RAM	Common data input/output	P, Al	0~70
Mask ROMs					
M58730-XXXS	—	8192-Bit(1024×8) Mask Programmable ROM	Manufactured to order	N, Si	0~70
M58730-001S	—	8192-Bit(1024×8) Mask Programmed ROM	Subroutine 1: integer arithmetic operations	N, Si	0~70
M58731-XXXP,S	—	16384-Bit(2048×8) Mask Programmable ROM	Manufactured to order	N, Si, ED	0~70
M58731-001S	—	16384-Bit(2048×8) Mask Programmed ROM	With MELPS 8 BOM-B Basic operating monitor	N, Si, ED	0~70
M58333-XXXP *	—	32 768-Bit(4096×8) Mask Programmable ROM	Manufactured to order	N, Al	0~70
M58334-XXXP *	—	65 536-Bit(8192×8) Mask Programmable ROM	Manufactured to order	N, Al, ED	0~70
Field-Programmable ROMs (EEPROMs, EPROMs, PROMs)					
M5G 1400P *	M58654P	1400-Bit (100×14) Electrically Alterable ROM	Electrical programming and erasing	P, Al	0~70
M5L 2708K,S **M5L 2708K,S-65**	M58732S M58732S-1	8192-Bit (1024×8) Erasable and Electrically Reprogrammable ROM	Electrical programming, ultraviolet erasing	N, Si, FA	0~70
M5L 2716K * **M5L 2716K-65** *	— —	16384-Bit (2048×8) Erasable and Electrically Reprogrammable ROM	Electrical programming, ultraviolet erasing	N, Si, FA	0~70
M54700K,P,S	—	1024-Bit (256×4) Field-Programmable ROM with Open-Collector Outputs	Ni-Cr fuse programming	B	0~75
M54730K,P,S	—	256 Bit (32×8) Field Programmable ROM with Open-Collector Outputs	Ni-Cr fuse programming	B	0~75

Note 1 : * =New product; ** =Under development
2 : Al=Aluminum gate, B =Bipolar, C=CMOS, ED=Enhancement depletion mode, FA=FAMOS,
N =N-channel, P =P-channel, S=Schottkey, Si =Silicon gate

MITSUBISHI ELECTRIC

INDEX BY FUNCTION

Supply voltage				Clock voltage	Electrical characteristics					Package (Note 3)	Interchangeable products		Page
V_{DD}	V_{CC}	V_{SS} GND	V_{BB}	$V_1(\phi)$	Typ pwr dissipation (mW)	Max access time (ns) (Note 4)	Max cycle time (ns)	Max frequency (MHz)	TTL compatibility		Mfr.	Type	
—	5V±5%	0V	—	—	200	250	250	—	Yes	22P1 22S1	INTEL	P. C2101A-2	4—25
					175	350	350					P. C2101A	
					150	450	450					P. C2101A-4	
—	5V±5%	0V	—	—	200	250	350	—	Yes	18P1 18S1	INTEL	P. C2111A-2	4—39
					175	350	350					P. C2111A	
					150	450	450					P. C2111A-4	
—	5V±5%	0V	—	—	200	250	250	—	Yes	16P1 16S1	INTEL	P. C2112A-2	4—43
					175	350	350					P. C2112A	
					150	450	450					P. C2112A-4	
—	5V±5%	0V	—	—	100	450	450	—	Yes	16P1 16S1	INTEL	P. C2102A-4	4—29
—	5V±10%	0V	—	—	300	200	200	—	Yes	18P1 18S1	INTEL TI	P. C2114L-2 TMS4045-20JL, NL	4—49
					250	300	300					P. C2114L-3 TMS4045-30JL, NL	
					200	450	450					P. C2114L TMS4045-45JL, NL	
—	5V±10%	0V	—	—	300	200	200	—	Yes	18P1 18S1	TI	TMS4044-20JL, NL	4—57
					250	300	300					TMS4044-30JL, NL	
					200	450	450					TMS4044-45JL, NL	
12V±10%	5V±10%	0V	−5V±10%	V_{DD}±1V	300	200	400	—	Yes	22P1 22S1	INTEL TI	C2107B TMS4060-2JL	4—33
12V±10%	5V±10%	0V	−4.5 ~ −5.7V	2.7V	280	150	375	—	Yes	16P1 16S1	MOSTEK	MK4116-2	4—13
					280	200	375					MK4116-3	
					260	250	410					MK4116-4	
—	5V±10%	0V	—	—	75	450	450	—	Yes	22P1	INTEL	P5101L-1	4—53
—	5V±10%	0V	—	—	75	450	450	—	Yes	18S1	—	—	4—9
−15V±5%	—	5V±5%	—	—	500	1500	1750	—	Yes	18S1	TOSHIBA	TMM142C	4—3
12V±5%	5V±5%	0V	−5V±5%	—	250	850	—	—	Yes	24S1	INTEL	C8308	5—19
12V±5%	5V±5%	0V	−5V±5%	—	250	850	—	—	Yes	24S1	—	—	5—22
—	5V±5%	0V	—	—	200	850	—	—	Yes	24P1 24S1	INTEL	P8316A C8316A	5—23
—	5V±5%	0V	—	—	200	850	—	—	Yes	24S1	—	—	5—26
—	5V±5%	0V	—	—	300	650	—	—	Yes	24P1	—	—	5—15
—	5V±5%	0V	—	—	300	650	—	—	Yes	24P1	—	—	5—17
—	5V±5%	0V	—	—	200	20μs	—	16.8KHz	Yes	14P4	GI	1400	5—27
12V±5%	5V±5%	0V	−5V±5%	—	600	450	—	—	Yes	24K10 24S10	INTEL	D. C2708	5—31
					600	650	—				—	—	
—	5V±5%	0V	—	—	300	450	—	—	Yes	24K10	INTEL	D2716	5—35
					300	650	—				—	—	
—	5V±5%	0V	—	—	450	60	60	—	Yes	16K1 16P1 16S1	MMI	6300J 6300 6300D	5—6
—	5V±5%	0V	—	—	450	50	50	—	Yes	16K1 16P1 16S1	MMI	6330J 6330 6330D	5—11

Note 3 : 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

INDEX BY FUNCTION

Type (Note 1)	Former designation	Circuit function and organization	Application notes	Structure (Note 2)	Ambient operating temp. Ta (°C)
Single-Chip Microcomputers					
M58840-XXXP	—	Single-Chip 4-Bit Microcomputer with 8-Bit A/D Converter	68 instructions, 2K-word by 9-bit mask-prog. ROM, 128-word by 4-bit RAM	P, Al, ED	−10~70
M58842S	—	MELPS 4 System Evaluation Device	68 instructions, 128-word by 4-bit RAM	P, Al, ED	−10~70
M58494-XXXP **	—	Single-Chip 4-Bit CMOS Microcomputer	93 instructions, 4K-word by 10-bit mask-prog. ROM, 32-word by 4-bit RAM	C, Al	−10~70
Microprocessors					
M5L 8080AP,S	M58710P, S	8-Bit Parallel CPU	78 instructions	N, Si	0~70
M5L 8085AP,S * **M5L 8085AP,S-20** *	M58712P, S M58712P, S-20	Single-Chip 8-Bit N-Channel Microprocessor	80 instructions	N, Si, ED	0~70
LSIs for Peripheral Circuits					
M58609-04P,S	–	Keyboard Encoder	JIS code standard product	P, Al	−20~75
M58609-09P,S	–	Keyboard Encoder	USASCII code standard product	P, Al	−20~75
M58620-001S	–	Keyboard Encoder	JIS code standard product	P, Al	−20~75
M58741P	–	TV Interface	64×64 color-segment display	N, Si, ED	0~70
M5L 8041A-XXXP *	–	Universal Peripheral Interface	90 instructions, 1K-word by 8-bit mask-prog. ROM, 64-word by 8-bit RAM	N, Si, ED	−20~70
M5L 8212P	M54552P	8-Bit Input/Output Port	With three-state outputs	B, S	0~75
M5L 8216P	M54553P	4-Bit Parallel Bidirectional Bus Driver	With three-state non-invert outputs	B, S	0~75
M5L 8224P	M54550P	Clock Generator and Driver for M5L 8080AP, S CPU		B, S	0~75
M5L 8226P	M54554P	4-Bit Parallel Bidirectional Bus Driver	With three-state invert outputs	B, S	0~75
M5L 8228P	M54551P	System Controller and Bus Driver for M5L 8080AP, S CPU	Bidirectional, for data bus isolation	B, S	0~75
M5L 8243P *	–	Input/Output Expander	For 4-bit I/O ports	N, Si, ED	−20~70
M5L 8251AP *	–	Programmable Communication Interface	Synchronous/asynchronous operation	N, Si, ED	0~70
M5L 8253P * **M5L 8253P-5** *	–	Programmable Interval Timer	3 independent 16-bit counters	N, Si, ED	0~70
M5L 8255AP * **M5L 8255AP-5** *	– –	Programmable Peripheral Interface	24 programmable I/O pins	N, Si, ED	0~70
M5L 8257P * **M5L 8257P-5** *	– –	Programmable DMA Controller	Priority DMA request logic	N, Si, ED	0~70
M5L 8279P **M5L 8279P-5**	M58743P –	Programmable Keyboard/Display Interface	For 64- or 128-key contact-switch keyboards and dual 8- or 16-character alphanumeric displays	N, Si, ED	0~70
General Purpose MOS LSIs					
M58412P	—	CMOS LCD Digital Alarm Clock Circuit	4.2MHz oscillator/divider	C, Al	−20~65
M58413P	—	CMOS LCD Digital Alarm Clock Circuit	32kHz oscillator/divider	C, Al	−20~65
M58434P	—	CMOS Analog Clock Circuit	4.2MHz oscillator/divider	C, Si	−20~70
M58435P	—	CMOS Analog Clock Circuit	4.2MHz oscillator/divider	C, Si	−20~70
M58436-001P *	—	CMOS Analog Clock Circuit	4.2MHz oscillator/divider	C, Al	−20~70
M58437-001P *	—	CMOS Analog Clock Circuit	32kHz oscillator/divider	C, Al	−20~70
M58478P	—	17-Stage Oscillator/Divider	17-stage binary divider	C, Al	−30~70
M58479P	—	CMOS Counter/Timer	50ms~4800h timer	C, Al	−30~70
M58482P	—	CMOS Counter/Timer	50ms~4800h timer	C, Al	−30~70
M58480P *	—	30-Function Remote Control Transmitter		C, Al	−30~70
M58484P *	—	30-Function Remote Control Transmitter		C, Al	−30~70
M58481P *	—	30-Function Remote Control Receiver		C, Al	−30~70
M58485P *	—	29-Function Remote Control Receiver		C, Al	−30~70
M58487P *		22 Function Remote Control Receiver		C, Al	−30~70
M58872P *	—	Single-Chip Printing Calculator		P, Al, ED	0~50

Note 1 : * = New product; ** = Under development
2 : Al = Aluminum gate, B = Bipolar, C = CMOS, ED = Enhancement depletion mode, FA = FAMOS,
N = N-channel, P = P-channel, S = Schottkey, Si = Silicon gate

1

Supply voltage				Clock voltage $V_1(\phi)$	Electrical characteristics					Package (Note 3)	Interchangeable products		Page
V_{DD}	V_{CC}	V_{SS} GND	V_{BB}		Typ pwr dissipation (mW)	Max access time (ns) (Note 4)	Max cycle time (ns)	Max frequency (MHz)	TTL compatibility		Mfr.	Type	
−15V±10%	—	0V	—	—	250	—	10000	0.6	Yes	42P1	—	—	6−3
−15V±10%	—	0V	—	—	250	—	10000	0.6	Yes	64S1	—	—	6−19
—	5V±5%	0V	—	—	5	—	6600	0.455	Yes	68P2	—	—	7−3
12V±1V	5V±5%	0V	−5V±5%	V_{DD}+1V ~9V	780	—	—	2	Yes	40P1 40S1	INTEL	P. C8080A	8−3
—	5V±5%	0V	—	—	600	—	—	3 / 2	Yes	40P1 40S1	INTEL / —	P. C8085A / —	8−29
−12V±1V	0V	5V±10%	—	—	70	—	—	0.1	Yes	40P1 40S1	—	—	9−3
−12V±1V	0V	5V±10%	—	—	70	—	—	0.1	Yes	40P1 40S1	—	—	9−9
−12V±10%	0V	5V±10%	—	—	350	—	—	—	Yes	40S1	—	—	9−11
—	5V±5%	0V	—	—	300	—	—	—	Yes	40P1	—	—	9−19
—	5V±10%	0V	—	—	300	—	—	6	Yes	40P1	INTEL	P8041A	9−23
—	5V±5%	0V	—	—	450	35☆	—	—	Yes	24P1	INTEL	P8212	9−27
—	5V±5%	0V	—	—	475	25☆	—	—	Yes	16P4	INTEL	P8216	9−31
—	5V±5%	0V	—	—	450	—	—	—	Yes	16P4	INTEL	P8224	8−17
—	5V±5%	0V	—	—	425	25☆	—	—	Yes	16P4	INTEL	P8226	9−31
—	5V±5%	0V	—	—	550	—	—	—	Yes	28P1	INTEL	P8228	8−23
—	5V±10%	0V	—	—	50	—	—	—	Yes	24P1	INTEL	P8243	9−35
—	5V±5%	0V	—	—	300	—	—	3	Yes	28P1	INTEL	P8251A	9−39
—	5V±5%	0V	—	—	300	—	—	2	Yes	24P1	INTEL	P8253 P8253-5	9−55
—	5V±5%	0V	—	—	250	—	—	—	Yes	40P1	INTEL	P8255A P8255A-5	9−63
—	5V±5%	0V	—	—	300	—	—	3	Yes	40P1	INTEL	P8257 P8257-5	9−79
—	5V±10%	0V	—	—	650	—	—	2 / 3	Yes	40P1	INTEL	P8279 / P8279-5	9−87
−2.4~−3.8V	−1.2~−1.9V	0V	—	—	0.045	—	—	—	Yes	60P2	—	—	10−3
−2.2~−4V	−1.1~−2V	0V	—	—	0.003	—	—	—	Yes	60P2	—	—	
1.2~1.9V	—	0V	—	—	0.045	—	—	—	Yes	8P1	—	—	10−11
1.2~1.9V	—	0V	—	—	0.045	—	—	—	Yes	8P1	—	—	
1.1~1.9V	—	0V	—	—	0.053	—	—	—	Yes	8P1	—	—	
1.1~1.9V	—	0V	—	—	0.003	—	—	—	Yes	8P1	—	—	
4.75~8.5V	—	0V	—	—	16	—	—	—	Yes	8P1	—	—	10−15
7.4~9V	—	0V	—	—	2	—	—	—	Yes	14P4	—	—	10−19
3~9V	—	0V	—	—	0.2	—	—	—	Yes	14P4	—	—	
2.2~8V	—	0V	—	—	—	—	—	—	Yes	16P4	—	—	10−23
2.2~8V	—	0V	—	—	—	—	—	—	Yes	16P4	—	—	
4.5~8V	—	0V	—	—	—	—	—	—	Yes	28P1	—	—	10−27
8~14V	—	0V	—	—	—	—	—	—	Yes	28P1	—	—	10−31
4.5~8V	—	0V	—	—	—	—	—	—	Yes	28P1	—	—	10−35
−12~−14V	—	0V	—	—	50	—	—	—	Yes	40P1	—	—	10−39

Note 3 : 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

Note 4 : ☆: Indicates propagation time.

INDEX BY FUNCTION

Type	Function	Application notes	Memory capacity		I/O port (bits)	Ambient operating temp. Ta (°C)	Supply voltage (V)	Dimensions (l×w×h) (mm)	Page
			RAM (bytes)	ROM (bytes)					
Microcomputer Systems									
PCA0801	MELCS 8/2 Single-Board Computer	Using M5L8080AP	256	2K (Note 1)	48	0~55	5	125×145×17	11—3
PCA0802	MELCS 8/2 Memory and I/O Expansion Board	For PCA0801	1K	4K (Note 1)	24	0~55	12,5,−5	125×145×17	11—7
PCA0803	MELCS 8/2 Program Checker	For PCA0801,PCA0802	—	—	—	0~55	5	170×200×27	11—11
PCA0804G01* **PCA0804G02***	MELCS 8/2 Color TV Display Single-Board Computer	Using M5L8080AP,S and M58741P	1K	2K (Note 2)	24	5~40	12,5,−5	125×145×30	11—13
PCA8501G01* **PCA8501G02***	MELCS 85/2 Single-Board Computer	Using M5L8085AP	1K	4K (Note 3)	48	0~55	5	125×145×17	11—19
PC8500 *	MELCS 85/1 Portable Microcomputer Console	For microcomputer system	—	—	—	10~40	AC100	350×370×140	11—23

Note 1 : The standard product contains one M5L2708K 1K-byte EPROM
2 : The PCA0804G01 does not contain the M5L2708K EPROM
3 : The standard product contains one M5L2716K 2K-byte EPROM

Microcomputer Support Systems

PCA0401	MELCS 4 Single-Board System-Evaluation Computer	Using single-chip 4 bit microcomputer	128	2K	34	0~55	7,−15	180×190×20	12—3
PCA0402	MELCS 4 Capacitive Touch Keyboard	For PCA0401	—	—	—	0~55	−15 0~−120	180×180×20	12—7
PCA0403	MELCS 4 Program Checker	For PCA0401, PCA0402	—	—	—	0~55	7,−5 −10,−15	200×270×27	12—9

Note 1 : * : New Product

INDEX BY TYPE DESIGNATION

Type	Structure	Function	Circuit function	Page
M54700K M54700P M54700S	B	PROM	1024-bit (256-word × 4-bit) field-programmable ROM with open-collector outputs	5—6
M54730K M54730P M54730S	B	PROM	256-bit (32-word × 8-bit) field-programmable ROM with open-collector outputs	5—11
M58333-XXXP	N, Al	ROM	32768-bit (4096-word × 8-bit) mask-programmable ROM	5—15
M58334-XXXP	N, Al, ED	ROM	65536-bit (8192-word × 8-bit) mask-programmable ROM	5—17
M58412P	C, Al	Clock	CMOS LCD digital alarm clock circuit	10—3
M58413P	C, Al	Clock	CMOS LCD digital alarm clock circuit	10—3
M58434P	C, Si	Clock	CMOS analog clock circuit	10—11
M58435P	C, Si	Clock	CMOS analog clock circuit	10—11
M58436-001P	C, Al	Clock	CMOS analog clock circuit	10 11
M58437-001P	C, Al	Clock	CMOS analog clock circuit	10—11
M58478P	C, Al	Counter	17-stage oscillator/divider	10—15
M58479P	C, Al	Counter	CMOS counter/timer	10—19
M58480P	C, Al	Remo-con	30-function remote-control transmitter	10—23
M58481P	C, Al	Remo-con	30-function remote-control receiver	10—27
M58482P	C, Al	Counter	CMOS counter/timer	10—19
M58484P	C, Al	Remo-con	30-function remote-control transmitter	10—23
M58485P	C, Al	Remo-con	29-function remote-control receiver	10—31
M58487P	C, Al	Remo-con	22-function remote-control receiver	10—35
M58494-XXXP	C, Al	CPU	Single-chip 4-bit CMOS microcomputer	7—3
M58609-04P M58609-04S	P, Al	I/O	Keyboard encoder (JIS code standard product)	9—3
M58609-09P M58609-09S	P, Al	I/O	Keyboard encoder (USASCII code standard product)	9—9
M58620-001S	P, Al	I/O	Keyboard encoder (JIS code standard product)	9—11
M58656S	P, Al	RAM	1024-bit (256-word × 4-bit) non-volatile static RAM	4—3
M58730-001S	N, Si	ROM	8192-bit (1024-word × 8-bit) mask-programmed ROM subroutine 1 integer arithmetic operations	5—22
M58730-XXXS	N, Si	ROM	8192-bit (1024-word × 8-bit) mask-programmable ROM	5—19
M58731-001S	N, ED	ROM	16384-bit (2048-word × 8-bit) mask-programmed ROM with MELPS 8 ROM-B basic operating monitor	5—26
M58731-XXXP	N, ED	ROM	16384-bit (2048-word × 8-bit) mask-programmable ROM	5—23
M58731-XXXS	N, ED	ROM	16384-bit (2048-word × 8-bit) mask-programmable ROM	5—23
M58741P	N, Si, ED	I/O	TV interface	9—19
M58840-XXXP	P, Al, ED	CPU	Single-chip 4-bit microcomputer with 8-bit A/D converter	6—3
M58842S	P, Al, ED	CPU	MELPS 4-system evaluation device	6—19
M58872P	P, Al, ED	Calculator	Single-chip printing calculator	10—39
M58981S-45	C, Si	RAM	4096-bit (1024-word × 4-bit) CMOS static RAM	4—9
M5G1400P	P, Al	ROM	1400-bit (100-word × 14-bit) electrically alterable ROM	5—27
M5K4116P-2 M5K4116S-2 M5K4116P-3 M5K4116S-3 M5K4116P-4 M5K4116S-4	N, Si	RAM	16384-bit (16384-word × 1-bit) dynamic RAM	4—13

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.

INDEX BY TYPE DESIGNATION

Type	Structure	Function	Circuit function	Page
M5L 2101A P	N, Si, ED	RAM	1024-bit (256-word × 4-bit) static RAM	4—25
M5L 2101A S				
M5L 2101A P-2				
M5L 2101A S-2				
M5L 2101A P-4				
M5L 2101A S-4				
M5L 2102A P-4	N, Si, ED	RAM	1024-bit (1024-word × 1-bit) static RAM	4—29
M5L 2102A S-4				
M5L 2107B P	N, Si	RAM	4096-bit (4096-word × 1-bit) dynamic RAM	4—33
M5L 2107B S				
M5L 2111A P	N, Si, ED	RAM	1024-bit (256-word × 4-bit) static RAM	4—39
M5L 2111A S				
M5L 2111A P-2				
M5L 2111A S-2				
M5L 2111A P-4				
M5L 2111A S-4				
M5L 2112A P	N, Si, ED	RAM	1024-bit (256-word × 4-bit) static RAM	4—43
M5L 2112A S				
M5L 2112A P-2				
M5L 2112A S-2				
M5L 2112A P-4				
M5L 2112A S-4				
M5L 2114L P	N, Si, ED	RAM	4096-bit (1024-word × 4-bit) static RAM	4—49
M5L 2114L S				
M5L 2114L P-2				
M5L 2114L S-2				
M5L 2114L P-3				
M5L 2114L S-3				
M5L 2708 K	N, Si, FA	ROM	8192-bit (1024-word × 8-bit) erasable and electrically reprogrammable ROM	5—31
M5L 2708 S				
M5L 2708 K-65				
M5L 2708 S-65				
M5L 2716 K	N, Si, FA	ROM	16384-bit (2048-word × 8-bit) erasable and electrically reprogrammable ROM	5—35
M5L 2716 K-65				
M5L 5101L P-1	C, Si	RAM	1024-bit (256-word × 4-bit) CMOS static RAM	4—53
M5L 8041A-XXX P	N, Si, ED	I/O	Universal peripheral interface	9—23
M5L 8080A P	N, Si	CPU	8-bit parallel CPU	8—3
M5L 8080A S				
M5L 8085A P	N, Si, ED	CPU	Single-chip 8-bit N-channel microprocessor	8—29
M5L 8085A S-20				
M5L 8085A P-20				
M5L 8085 AS-20				
M5L 8212 P	B, S	I/O	8-bit input/output port	9—27
M5L 8216 P	B, S	I/O	4-bit parallel bidirectional bus driver (with non-invert outputs)	9—31
M5L 8224 P	B, S	I/O	Clock generator and driver for M5L8080AP, S CPU	8—17
M5L 8226 P	B, S	I/O	4-bit parallel bidirectional bus driver (with invert outputs)	9—31
M5L 8228 P	B, S	I/O	System controller and bus driver for M5L8080AP, S CPU	8—23
M5L 8243 A P	N, Si, ED	I/O	Input/output expander	9—35
M5L 8251A P	N, Si, ED	I/O	Programmable communication interface	9—39

INDEX BY TYPE DESIGNATION

Type	Structure	Function	Circuit function	Page
M5L 8253P M5L 8253P-5	N, Si, ED	I/O	Programmable interval timer	9—55
M5L 8255AP M5L 8255AP-5	N, Si, ED	I/O	Programmable peripheral interface	9—63
M5L 8257P M5L 8257P-5	N, Si, ED	I/O	Programmable DMA controller	9—79
M5L 8279P M5L 8279P-5	N, Si, ED	I/O	Programmable keyboard/display interface	9—87
M5T 4044P-20 M5T 4044S-20 M5T 4044P-30 M5T 4044S-30 M5T 4044P-45 M5T 4044S-45	N, Si, ED	RAM	4096-bit (4096-word × 1-bit) static RAM	4—57

Type	Function	Page
PC8500	MELCS 85/1 Portable Microcomputer Console	11—23
PCA0401	MELCS 4 Single-Board System-Evaluation Computer	12—3
PCA0402	MELCS 4 Capacitive Touch Keyboard	12—7
PCA0403	MELCS 4 Program Checker	12—9
PCA0801	MELCS 8/2 Single-Board Computer	11—3
PCA0802	MELCS 8/2 Memory and I/O Expansion Board	11—7
PCA0803	MELCS 8/2 Program Checker	11—11
PCA0804G01 PCA0804G02	MELCS 8/2 Color TV Display Single-Board Computer	11—13
PCA8501G01 PCA8501G02	MELCS 85/2 Single-Board Computer	11—19

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

GUIDE TO INTERCHANGEABLILITY

Function	Mitsubishi Electric	Circuit organization	Advanced Micro Devices	American Microsystems	Electronic Arrays
CPUs	M58840-XXXP	Single-chip 4-bit			
	M58842S	MELPS 4 system evaluation device			
	M58494-XXXP	Single-chip 4-bit CMOS			
	M5L 8080AP	8-bit parallel			
	M5L 8080AS	8-bit parallel	AM9080A		
	M5L 8085AP	Single-chip 8-bit N-channel			
	M5L 8085AS	Single-chip 8-bit N-channel	AM8085A		
	M5L 8085AP-20	Single-chip 8-bit N-channel			
	M5L 8085AS-20	Single-chip 8-bit N-channel			
Static RAMs	M5L 2101AP-2	256× 4-bit			
	M5L 2101AS-2	256× 4-bit			
	M5L 2101AP	256× 4-bit			
	M5L 2101AS	256× 4-bit	AM2101		
	M5L 2101AP-4	256× 4-bit			
	M5L 2101AS-4	256× 4-bit			
	M5L 2111AP-2	256× 4-bit			
	M5L 2111AS-2	256× 4-bit			
	M5L 2111AP	256× 4-bit			
	M5L 2111AS	256× 4-bit			
	M5L 2111AP-4	256× 4-bit			
	M5L 2111AS-4	256× 4-bit			
	M5L 2112AP-2	256× 4-bit			
	M5L 2112AS-2	256× 4-bit			
	M5L 2112AP	256× 4-bit			
	M5L 2112AS	256× 4-bit			
	M5L 2112AP-4	256× 4-bit			
	M5L 2112AS-4	256× 4-bit			
	M5L 2102AP-4	1024× 1-bit			
	M5L 2102AS-4	1024× 1-bit		S3102	
	M5L 2114LP-2	1024× 4-bit			
	M5L 2114LS-2	1024× 4-bit	AM91L14E	S2114L-2	
	M5L 2114LP-3	1024× 4-bit			
	M5L 2114LS-3	1024× 4-bit	AM91L14C	S2114L-3	
	M5L 2114LP	1024× 4-bit			
	M5L 2114LS	1024× 4-bit	AM91L14B		
	M5T 4044P-20	4096× 1-bit			
	M5T 4044S-20	4096× 1-bit	AM4044-20		
	M5T 4044P-30	4096× 1-bit			
	M5T 4044S-30	4096× 1-bit	AM4044-30		
	M5T 4044P-45	4096× 1-bit			
	M5T 4044S-45	4096× 1-bit	AM4044-45		

1

Fairchild Semiconductor	Fujitsu	Hitachi	Intel	Intersil	Monolithic Memories	Mostek
			P8080A			
			C8080A			
			P8085A			
			C8085A			
			P2101A-2			
			C2101A-2			
			P2101A			
			C2101A			
		HM45102	P2101A-4			
	MB8101		C2101A-4			
			P2111A-2			
			C2111A-2			
			P2111A			
			C2111A			
			P2111A-4			
	MB8111		C2111A-4			
			P2112A-2			
			C2112A-2			
			P2112A			
			C2112A			
			P2112A-4			
	MB8112		C2112A-4			
21021	MB8102		P2102A-4	IM7552-1CPE		MK4102P-1
			C2102A-4	IM7552-1CDE		
			P2114L-2			
			C2114L-2	2114L-2		
			P2114L-3			
2114L-3			C2114L-3	2114L-3		
			P2114L			
2114L			C2114L	2114L		
				IM7141-2		
				IM7141-3		
				IM7141		

GUIDE TO INTERCHANGEABLILITY

Function	Mitsubishi Electric	Circuit organization	Motorola Semiconductor Products	National Semiconductor	Nippon Electric
CPUs	M58840-XXXP	Single-chip 4-bit			
	M58842S	MELPS 4 system evaluation device			
	M58494-XXXP	Single-chip 4-bit CMOS			
	M5L 8080AP	8-bit parallel			
	M5L 8080AS	8-bit parallel		INS8080A	
	M5L 8085AP	Single-chip 8-bit N-channel			
	M5L 8085AS	Single-chip 8-bit N-channel			
	M5L 8085AP-20	Single-chip 8-bit N-channel			
	M5L 8085AS-20	Single-chip 8-bit N-channel			
Static RAMs	M5L 2101AP-2	256× 4-bit			
	M5L 2101AS-2	256× 4-bit			
	M5L 2101AP	256× 4-bit			
	M5L 2101AS	256× 4-bit			
	M5L 2101AP-4	256× 4-bit		MM2101AN-4	
	M5L 2101AS-4	256× 4-bit		MM2101AJ-4	μPD2101A
	M5L 2111AP-2	256× 4-bit			
	M5L 2111AS-2	256× 4-bit			
	M5L 2111AP	256× 4-bit			
	M5L 2111AS	256× 4-bit			
	M5L 2111AP-4	256× 4-bit		MM2111AN-4	
	M5L 2111AS-4	256× 4-bit		MM2111AJ-4	μPD2111A
	M5L 2112AP-2	256× 4-bit			
	M5L 2112AS-2	256× 4-bit			
	M5L 2112AP	256× 4-bit			
	M5L 2112AS	256× 4-bit			
	M5L 2112AP-4	256× 4-bit		MM2112AN-4	
	M5L 2112AS-4	256× 4-bit		MM2112AJ-4	
	M5L 2102AP-4	1024× 1-bit		MM2102AN-4	
	M5L 2102AS-4	1024× 1-bit		MM2102AJ-4	
	M5L 2114LP-2	1024× 4-bit	MCM21L14-20P	MM2114N-2	
	M5L 2114LS-2	1024× 4-bit	MCM21L14-20S	MM2114J-2	
	M5L 2114LP-3	1024× 4-bit	MCM21L14-30P	MM2114N-3	
	M5L 2114LS-3	1024× 4-bit	MCM21L14-30S	MM2114J-3	
	M5L 2114LP	1024× 4-bit	MCM21L14-45P	MM2114N	
	M5L 2114LS	1024× 4-bit	MCM21L14-45S	MM2114J	
	M5T 4044P-20	4096× 1-bit	MCM66L41-20P	MM5257N-2	
	M5T 4044S-20	4096× 1-bit	MCM66L41-20S	MM5257J-2	
	M5T 4044P-30	4096× 1-bit	MCM66L41-30P	MM5257N-3	
	M5T 4044S-30	4096× 1-bit	MCM66L41-30S	MM5257J-3	
	M5T 4044P-45	4096× 1-bit	MCM66L41-45P	MM5257N	
	M5T 4044S-45	4096× 1-bit	MCM66L41-45S	MM5257J	

1

Texas Instruments	Toshiba	Signetics	General Instrument
TMS8080			
TMS8080		MP8080A	
	T3374		RA9-1101A
TMS4039		2101A-4	
	TMM311P		
TMS4042	TMM311C	2111A-4	
	TMM312P		
TMS4043	TMM312C	2112A-4	
	TMM313P		
TMS4033	TMM313C	2602	
TMS4045-20NL			
TMS4045-20JL		2614-20	
TMS4045-30NL			
TMS4045-30JL		2614-30	
TMS4045-45NL			
TMS4045-45JL		2614-45	
TMS4044-20NL		2613-20N	
TMS4044-20JL		2613-20I	
TMS4044-30NL			
TMS4044-30JL			
TMS4044-45NL		2613-45N	
TMS4044-45JL		2613-45I	

GUIDE TO INTERCHANGEABLILITY

Function	Mitsubishi Electric	Circuit organization	Advanced Micro Devices	American Microsystems	Electronic Arrays
Dynamic RAMs	M5L 2107BP	4096× 1-bit			
	M5L 2107BS	4096× 1-bit	AM9060E	S4021-1	
	M5K 4116P-2	16384× 1-bit			
	M5K 4116S-2	16384× 1-bit			
	M5K 4116P-3	16384× 1-bit			
	M5K 4116S-3	16384× 1-bit	AM9016E		
	M5K 4116P-4	16384× 1-bit			
	M5K 4116S-4	16384× 1-bit	AM9016D		
CMOS Static RAMs	M5L 5101LP-1	256× 4-bit		S5101L-1	
	M58981S-45	1024× 4-bit			
Non-Volatile RAMs	M58656S	256× 4-bit			
Mask ROMs	M58730-XXXS	1024× 8-bit			
	M58730-001S	1024× 8-bit (programmed)			
	M58731-XXXP	2048× 8-bit			
	M58731-XXXS	2048× 8-bit			
	M58731-001S	2048× 8-bit (programmed)			
	M58333-XXXP	4096× 8-bit			
	M58334-XXXP	8192× 8-bit			
EEPROM	M5G 1400P	100× 14-bit FAMOS			
EPROMs	M5L 2708K	1024× 8-bit FAMOS			
	M5L 2708S	1024× 8-bit FAMOS	AM2708		EA2708C
	M5L 2708K-65	1024× 8-bit FAMOS			
	M5L 2708S-65	1024× 8-bit FAMOS			
	M5L 2716K	2048× 8-bit FAMOS			
	M5L 2716K-65	2048× 8-bit FAMOS			
FAMOS PROMS	M54700K	256× 4-bit	AM9760		
	M54700P	256× 4-bit			
	M54700S	256× 4-bit			
	M54730K	32× 8-bit			
	M54730P	32× 8-bit			
	M54730S	32× 8-bit			

Note: EEPROM = Electrically erasable and programmable ROM
EPROM = Electrically programmable ROM

Fairchild Semiconductor	Fujitsu	Hitachi	Intel	Intersil	Monolithic Memories	Mostek
			P2107B			MK4006-6P
3524-5	MB8103	HM3503	C2107B			
						MK4116-2
						MK4116-2
						MK4116-3
						MK4116-3
						MK4116-4
						MK4116-4
			P5101L-1	IM6551		
			C8308			
			P8316A			
			C8316A			
			D2708			
			C2708			
			D2716			
					6300J	
93417P					6300	
93417D					6300D	
					6330J	
					6330	
					6330D	

GUIDE TO INTERCHANGEABLILITY

Function	Mitsubishi Electric	Circuit organization	Motorola Semiconductor Products	National Semiconductor	Nippon Electric
Dynamic RAMs	M5L 2107BP	4096× 1-bit		MM5280N	μPD411D
	M5L 2107BS	4096× 1-bit			
	M5K 4116P-2	16384× 1-bit			
	M5K 4116S-2	16384× 1-bit	MCM4116L-15	MM5290J-2	
	M5K 4116P-3	16384× 1-bit			
	M5K 4116S-3	16384× 1-bit	MCM4116L-20	MM5290J-3	
	M5K 4116P-4	16384× 1-bit			
	M5K 4116S-4	16384× 1-bit	MCM4116L-25	MM5290J-4	
CMOS Static RAMs	M5L 5101LP-1	256× 4-bit	MCM145101-1P		
	M58981S-45	1024× 4-bit			
Non-Volatile RAMs	M58656S	256× 4-bit			
Mask ROMs	M58730-XXXS	1024× 8-bit			
	M58730-001S	1024× 8-bit (programmed)			
	M58731-XXXP	2048× 8-bit			
	M58731-XXXS	2048× 8-bit			
	M58731-001S	2048× 8-bit (programmed)			
	M58333-XXXP	4096× 8-bit			
	M58334-XXXP	8192× 8-bit			
EEPROM	M5G 1400P	100× 14-bit FAMOS			
EPROMs	M5L 2708K	1024× 8-bit FAMOS			
	M5L 2708S	1024× 8-bit FAMOS	MCM2708L	MM2708	
	M5L 2708K-65	1024× 8-bit FAMOS			
	M5L 2708S-65	1024× 8-bit FAMOS			
	M5L 2716K	2048× 8-bit FAMOS			
	M5L 2716K-65	2048× 8-bit FAMOS			
FAMOS PROMS	M54700K	256× 4-bit			
	M54700P	256× 4-bit		DM74S387N	
	M54700S	256× 4-bit		DM74S387J	
	M54730K	32× 8-bit			
	M54730P	32× 8-bit		DM74S188N	
	M54730S	32× 8-bit		DM74S188J	

Note: EEPROM = Electrically erasable and programmable ROM
EPROM = Electrically programmable ROMs

1

Texas Instruments	Toshiba	Signetics	General Instrument
TMS4060-2JL		2680	
		2690-2-N	
TMS4116-15		2690-2-I	
		2690-3-N	
TMS4116-20		2690-3-I	
		2690-4-N	
TMS4116-25		2690-4-I	
	TMM142C		
TMS4700			
	TMM331C		R03-9316A
TMS27L08		2708	
		N82S32	

GUIDE TO INTERCHANGEABLILITY

Function	Mitsubishi Electric	Circuit organization	Advanced Micro Devices	American Microsystems	Electronic Arrays
Peripheral circuits	M58609-04P	Keyboard encoder			
	M58609-04S	Keyboard encoder			
	M58609-09P	Keyboard encoder			
	M58609-09S	Keyboard encoder			
	M58620-001S	Keyboard encoder			
	M58741P	TV interface			
	M5L 8041A-XXXP	Universal peripheral interface			
	M5L 8212P	8-bit I/O port	AM8212		
	M5L 8216P	4-bit parallel bidirectional bus driver(non-invert outputs)	AM8216		
	M5L 8224P	Clock generator and driver	AM8224		
	M5L 8226P	4-bit bidirectional bus driver (invert outputs)	AM8226C		
	M5L 8228P	System controller and bus driver			
	M5L 8243P	Input/output expander			
	M5L 8251AP	Programmable communication interface			
	M5L 8253P	Programmable interval timer			
	M5L 8253P-5	Programmable interval timer			
	M5L 8255AP	Programmable peripheral interface	AM9555C		
	M5L 8255AP-5	Programmable peripheral interface			
	M5L 8257P	Programmable DMA controller			
	M5L 8257P-5	Programmable DMA controller			
	M5L 8279P	Programmable keyboard/display interface	AM8279		
	M5L 8279P-5	Programmable keyboard/display interface			
General-purpose MOS LSI	M58412P	CMOS LCD digital alarm clock circuit			
	M58413P	CMOS LCD digital alarm clock circuit			
	M58434P	CMOS analog clock circuit			
	M58435P	CMOS analog clock circuit			
	M58436-001P	CMOS analog clock circuit			
	M58437-001P	CMOS analog clock circuit			
	M58478P	17-stage oscillator/divider			
	M58479P	CMOS counter/timer			
	M58480P	30-function remote-control transmitter			
	M58481P	30-function remote-control receiver			
	M58482P	CMOS counter/timer			
	M58484P	30-function remote-control transmitter			
	M58485P	29-function remote-control receiver			
	M58487P	22-function remote-control receiver			
	M58872P	Single-chip printing calculator			

1

Fairchild Semiconductor	Fujitsu	Hitachi	Intel	Intersil	Monolithic Memories	Mostek
			P8041A			
	MB471		P8212			
			P8216			
			P8224			
			P8226			
			P8228			
			P8243			
			P8251A			
			P8253			
			P8253-5			
			P8255A			
			P8255A-5			
			P8257			
			P8257-5			
			P8279			
			P8279-5			

GUIDE TO INTERCHANGEABLILITY

Function	Mitsubishi Electric	Circuit organization	Motorola Semiconductor Products	National Semiconductor	Nippon Electric
Peripheral circuits	M58609-04P	Keyboard encoder			
	M58609-04S	Keyboard encoder			
	M58609-09P	Keyboard encoder			
	M58609-09S	Keyboard encoder			
	M58620-001S	Keyboard encoder			
	M58741P	TV interface			
	M5L 8041A-XXXP	Universal peripheral interface			
	M5L 8212P	8-bit I/O port		DP8212	μPB8212
	M5L 8216P	4-bit parallel bidirectional bus driver(non-invert outputs)		DP8216	
	M5L 8224P	Clock generator and driver		DP8224	μPB8224
	M5L 8226P	4-bit bidirectional bus driver (invert outputs)		DP8226	
	M5L 8228P	System controller and bus driver		DP8228	μPB8228
	M5L 8243P	Input/output expander			
	M5L 8251AP	Programmable communication interface			
	M5L 8253P	Programmable interval timer			
	M5L 8253P-5	Programmable interval timer			
	M5L 8255AP	Programmable peripheral interface		INS8255A	
	M5L 8255AP-5	Programmable peripheral interface			
	M5L 8257P	Programmable DMA controller			
	M5L 8257P-5	Programmable DMA controller			
	M5L 8279P	Programmable keyboard/display interface			
	M5L 8279P-5	Programmable keyboard/display interface			
General-purpose MOS LSI	M58412P	CMOS LCD digital alarm clock circuit			
	M58413P	CMOS LCD digital alarm clock circuit			
	M58434P	CMOS analog clock circuit			
	M58435P	CMOS analog clock circuit			
	M58436-001P	CMOS analog clock circuit			
	M58437-001P	CMOS analog clock circuit			
	M58478P	17-stage oscillator/divider			
	M58479P	CMOS counter/timer			
	M58480P	30-function remote-control transmitter			
	M58481P	30-function remote-control receiver			
	M58482P	CMOS counter/timer			
	M58484P	30-function remote-control transmitter			
	M58485P	29-function remote-control receiver			
	M58487P	22-function remote-control receiver			
	M58872P	Single-chip printing calculator			

1

Texas Instruments	Toshiba	Signetics	General Instrument
		8255A	

GUIDE TO SELECTION OF RAMs, PROMs AND ROMs

Words	Bits/word		
	1	4	8
32			PROM M54730K, P, S
256		RAMs M58656S M5L 2101AP, S-4 M5L 2111AP, S-4 M5L 2112AP, S-4 M5L 5101LP-4 PROMs M54700K, P, S	
1024	RAMs M5L 2102AP, S-4	RAMs M58981S-45 M5L 2114LP, S M5L 2114LP, S-2 M5L 2114LP, S-3	ROMs M58730-XXXS M58730-001S PROMs M5L 2708K, S M5L 2708K, S-65
2048			ROMs M58731-XXXP, S M58731-001P, S PROMs M5L 2716K M5L 2716K-65
4096	RAMs M5L 2107BP, S M5T 4044P, S M5T 4044P, S-2 M5T 4044P, S-3		ROMs M58333-XXXP
8192			ROMs M58334-XXXP
16384	RAMs M5K 4116P, S-2 M5K 4116P, S-3 M5K 4116P, S-4		

ORDERING INFORMATION AND PACKAGE OUTLINES

2

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 (military)

Series designation using 1 or 2 alpha-numeric 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.

ORDERING INFORMATION

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 (military)

Series designation of original source using 1 or 2 alphabetical characters.
- **K**: Mostek's MK series
- **L**: Intel's series
- **T**: Texas Instruments' TMS series
- **G**: General Instrument

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**

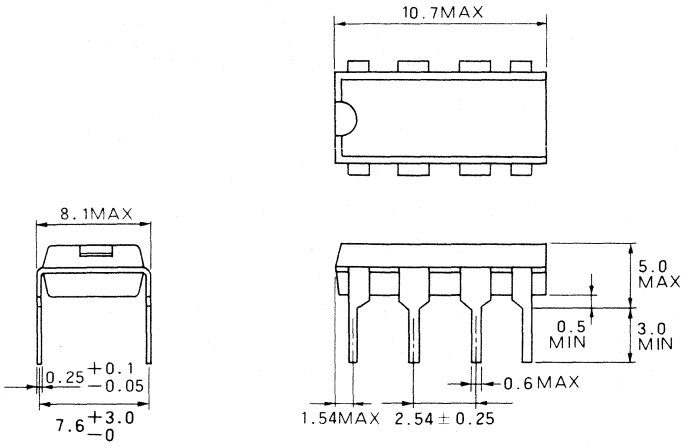
Number of pins

Package structure
- **K**: Glass-sealed ceramic
- **P**: Molded plastic
- **S**: Metal-sealed ceramic

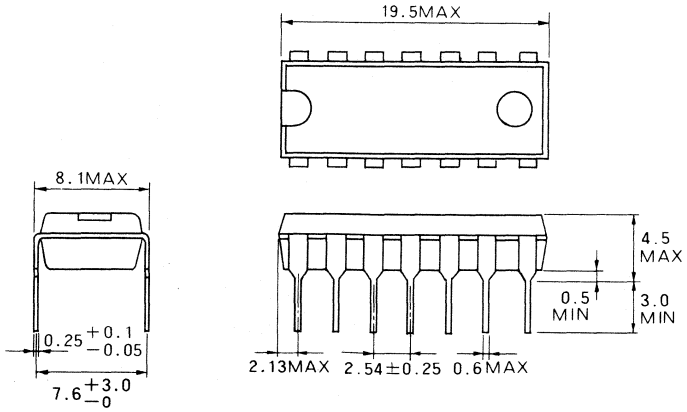
Package outline
- **1**: DIL without fin
- **2**: Flat without fin
- **4**: DIL without fin (improved)
- **10**: DIL without fin and with quartz lid

2

TYPE 8P1 8-PIN MOLDED PLASTIC DIL

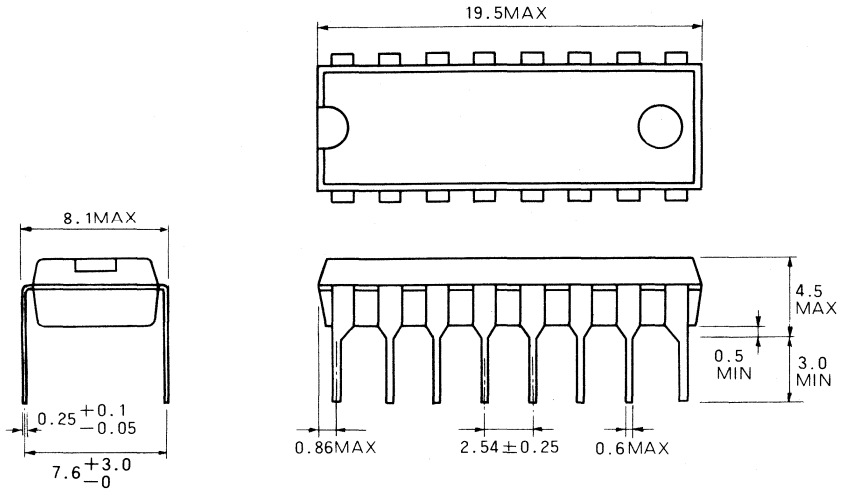


TYPE 14P1 14-PIN MOLDED PLASTIC DIL

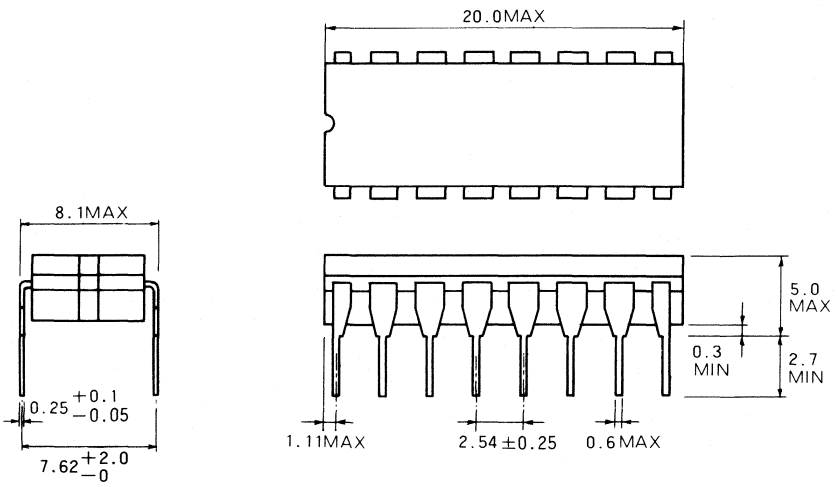


PACKAGE OUTLINES

TYPE 16P4 16-PIN MOLDED PLASTIC DIL



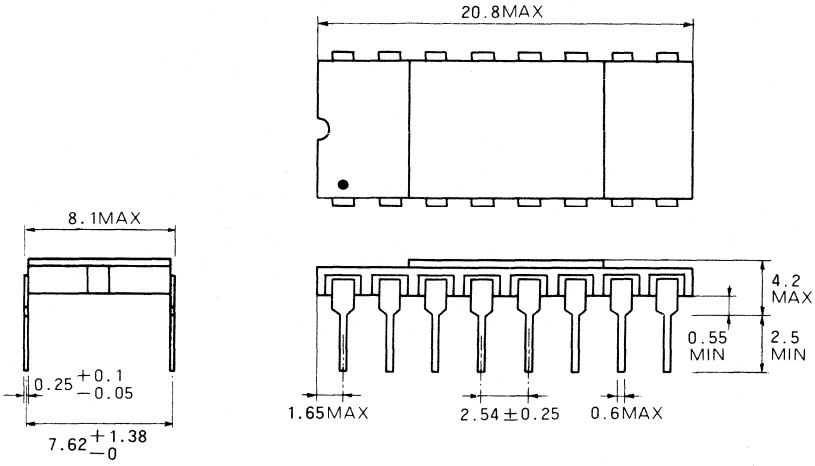
TYPE 16K1 16-PIN GLASS-SEALED CERAMIC DIL



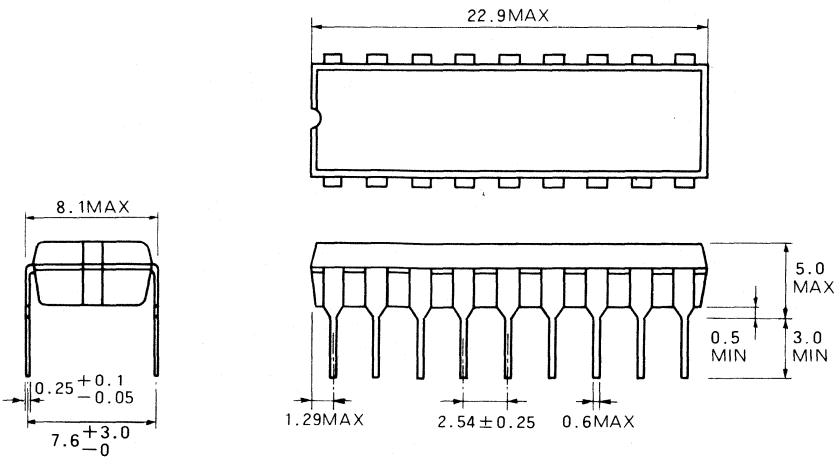
MITSUBISHI ELECTRIC

2

TYPE 16S1 16-PIN METAL-SEALED CERAMIC DIL

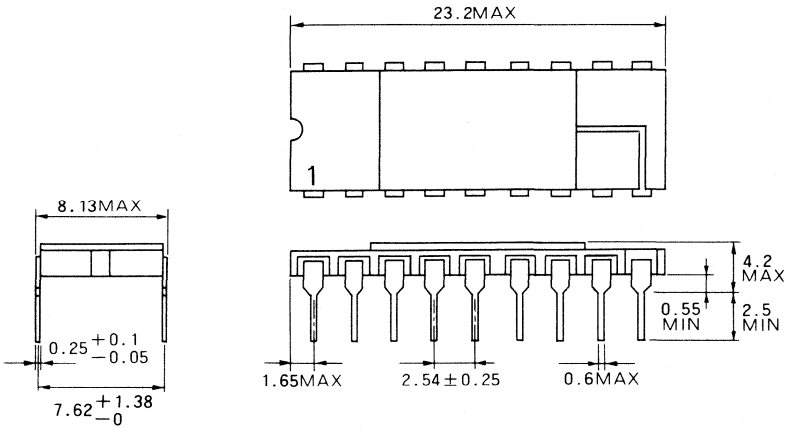


TYPE 18P1 18-PIN MOLDED PLASTIC DIL

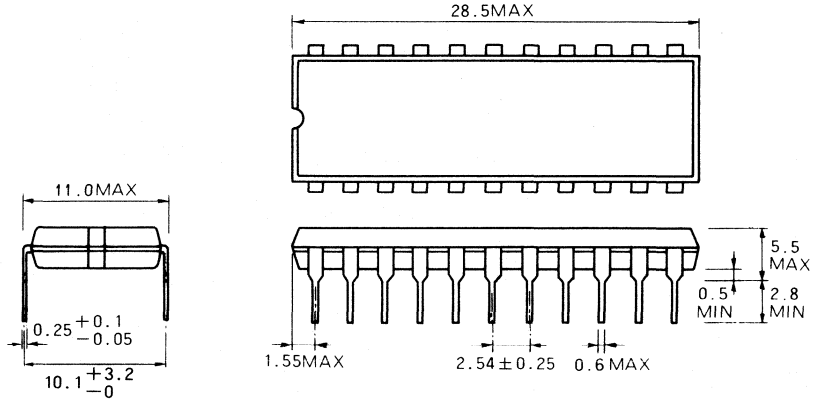


PACKAGE OUTLINES

TYPE 18S1 18-PIN METAL-SEALED CERAMIC DIL



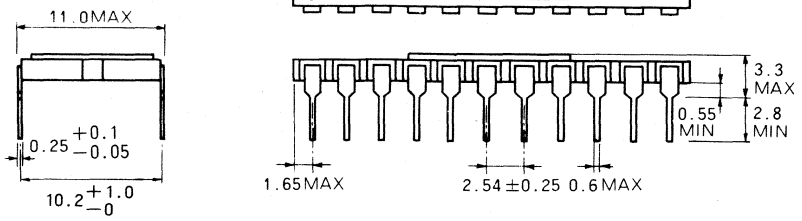
TYPE 22P1 22-PIN MOLDED PLASTIC DIL



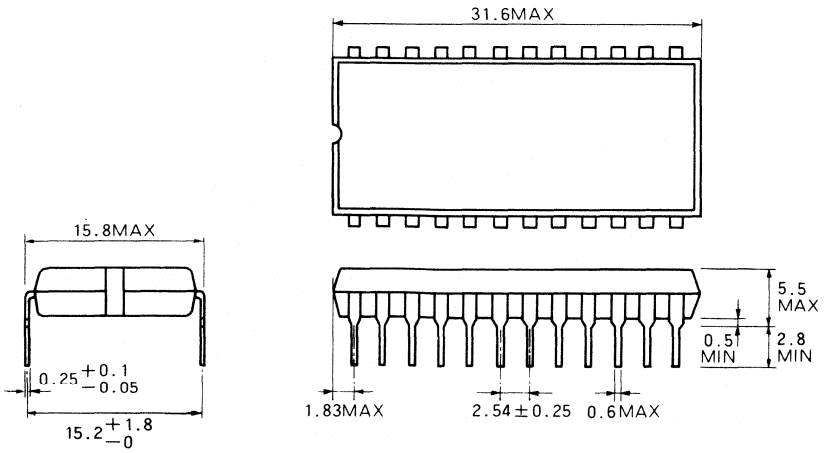
MITSUBISHI ELECTRIC

2

TYPE 22S1 22-PIN METAL-SEALED CERAMIC DIL

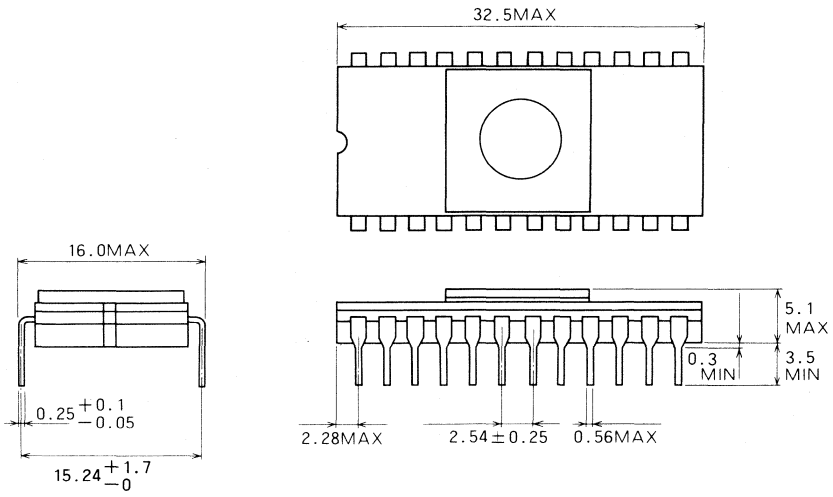


TYPE 24P1 24-PIN MOLDED PLASTIC DIL

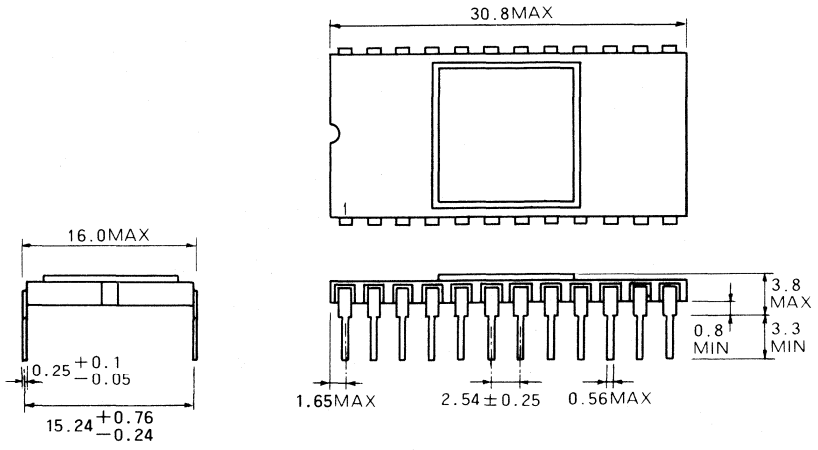


PACKAGE OUTLINES

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



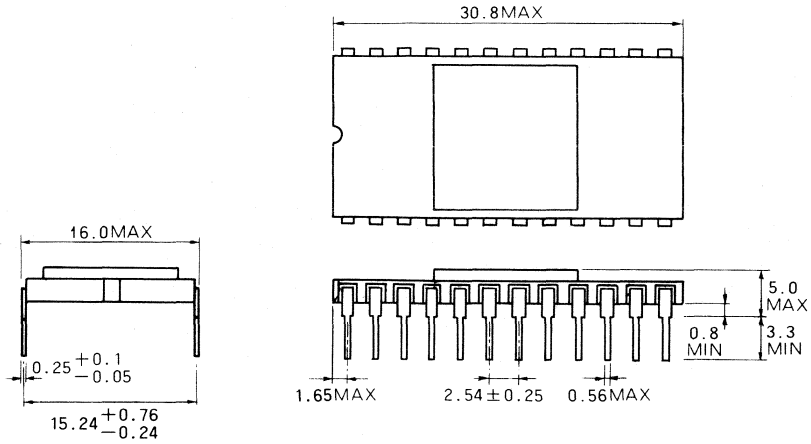
TYPE 24S1 24-PIN METAL-SEALED CERAMIC DIL



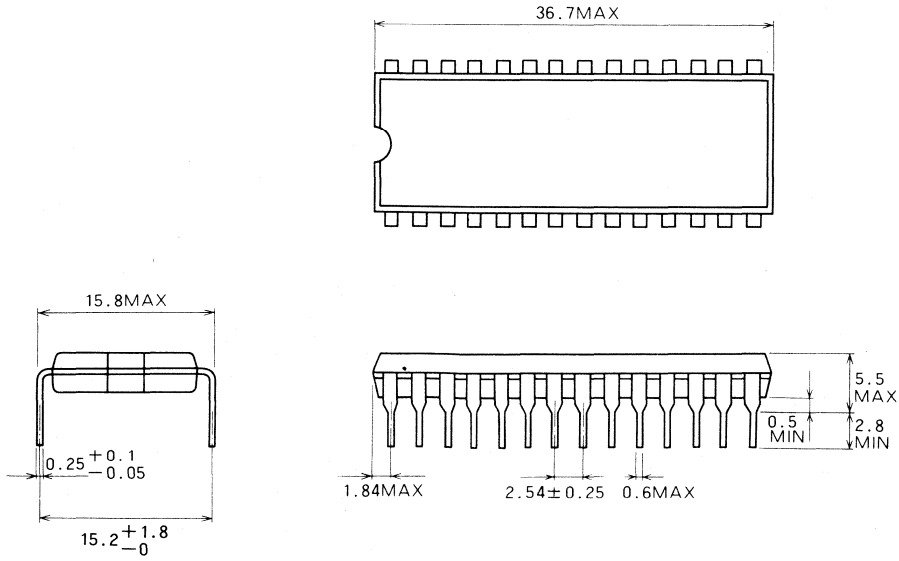
MITSUBISHI ELECTRIC

2

TYPE 24S10 24-PIN METAL-SEALED CERAMIC DIL WITH QUARTZ LID

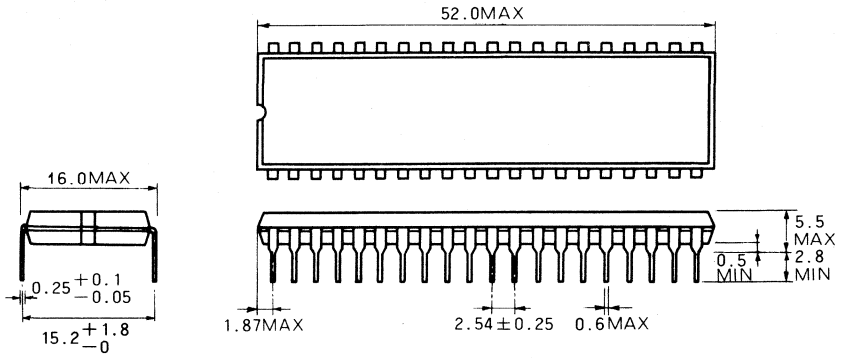


TYPE 28P1 28-PIN MOLDED PLASTIC DIL

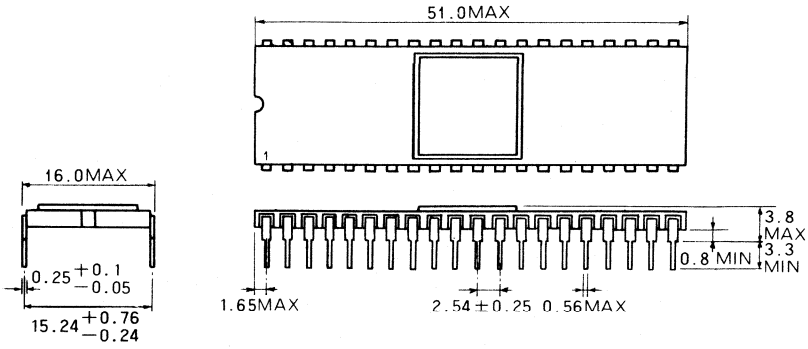


PACKAGE OUTLINES

TYPE 40P1 40-PIN MOLDED PLASTIC DIL



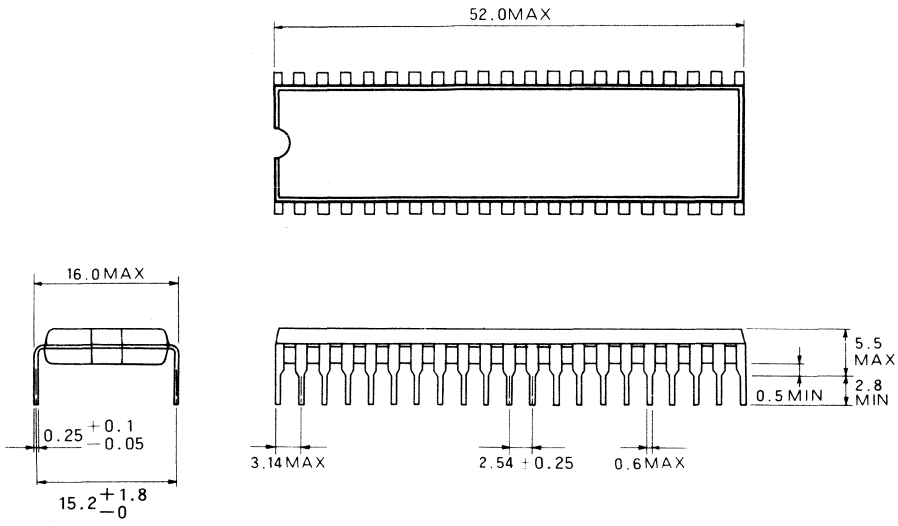
TYPE 40S1 40-PIN METAL-SEALED CERAMIC DIL



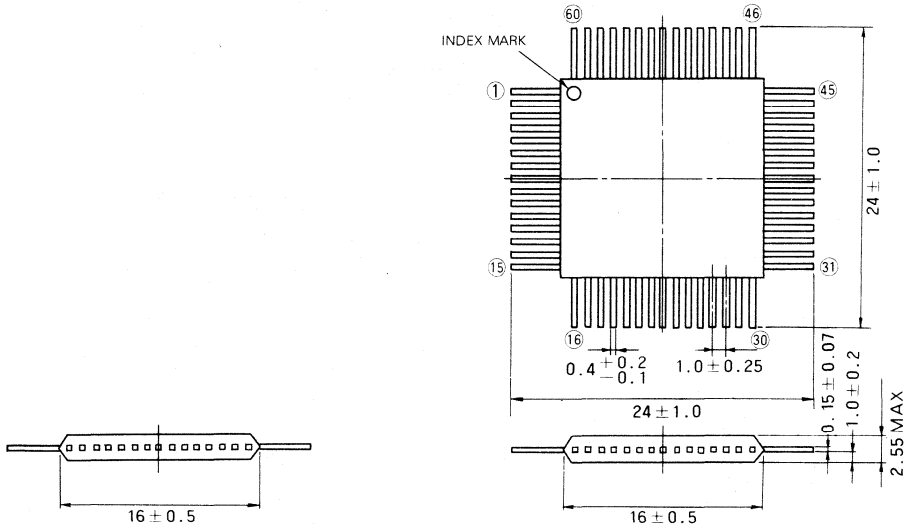
MITSUBISHI ELECTRIC

2

TYPE 42P1 42-PIN MOLDED PLASTIC DIL

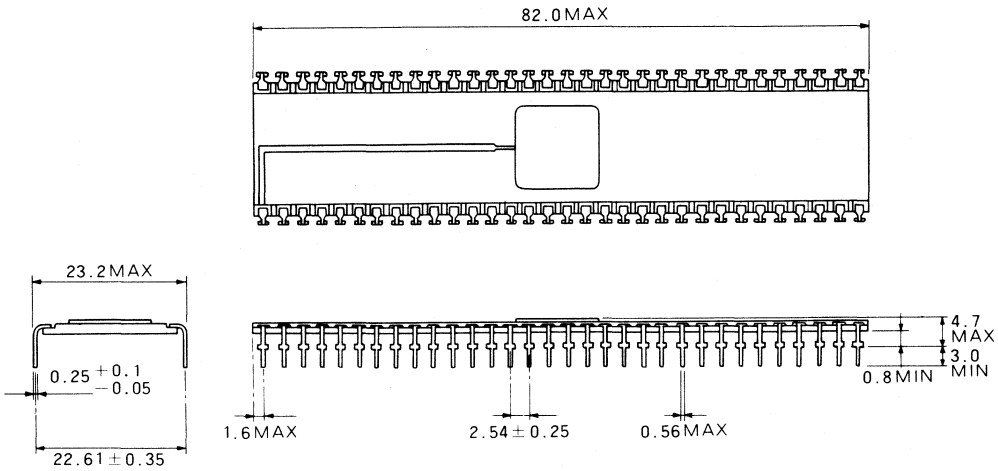


TYPE 60P2 60-PIN MOLDED PLASTIC FLAT

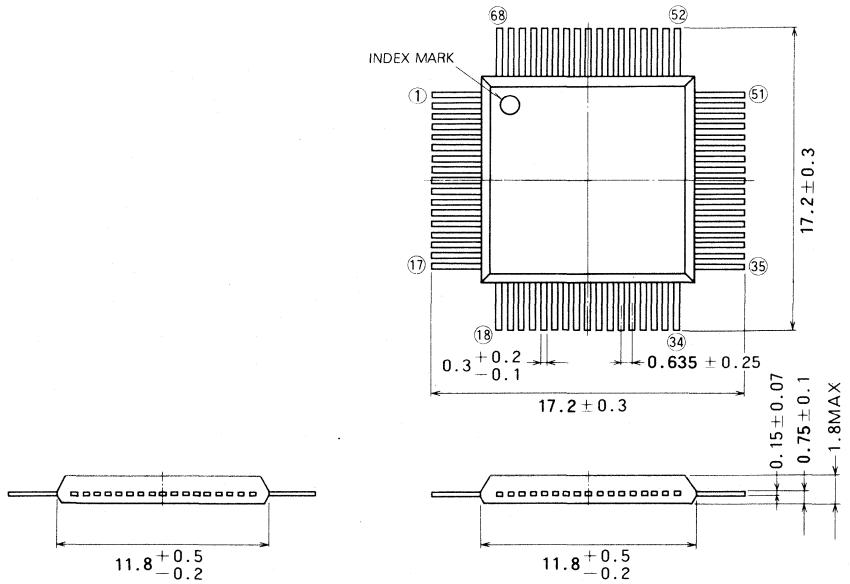


PACKAGE OUTLINES

TYPE 64S1 64-PIN METAL-SEALED CERAMIC DIL



TYPE 68P2 68-PIN MOLDED PLASTIC FLAT



MITSUBISHI ELECTRIC

GENERAL INFORMATION

3

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.

TERMINOLOGY

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 astable 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 combinatorial 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 current 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.

3

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.

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.

SYMBOLOGY

FOR DIGITAL INTEGRATED CIRCUITS

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 V_{CC}
$I_{CC(AV)}$	Average supply current from V_{CC}
$I_{CC(PD)}$	Power-down supply current from V_{CC}
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 conditions 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}	V_{SS} supply current
P_d	Power dissipation
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 signals at an output
$t_{a(AD)}$	Address access time—the time interval between the application of an address input pulse and the availability of valid data signals at an output
$t_{a(CE)}$	Chip enable access time
$t_{a(CS)}$	Chip select access time
t_c	Cycle time
$t_{c(REF)}$	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
$t_{c(RD)}$	Read cycle time—the time interval between the start of a read cycle and the start of the next cycle
$t_{c(RMW)}$	Read-modify-write cycle time—the time interval between the start of a cycle in which the memory is read and new data is entered, and the start of the next cycle
$t_{c(WR)}$	Write cycle time—the time interval between the start of a write cycle and the start of the next cycle
$t_{dv(AD)}$	Data valid time with respect to address—the time interval following an initial change of address during which data stored at the initial address continues to be valid at the output
$t_{dv(CE)}$	Data valid time with respect to chip enable—the time interval following chip enable during which output data continues to be valid
$t_{dv(CS)}$	Data valid time with respect to chip select—the time interval following chip select during which output data continues to be valid
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_{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_f	Fall time
t_h	Hold time—the time interval during which a signal is retained at a specified input terminal after an active transition occurs at another specified input terminal
$t_{h(AD)}$	Address hold time
$t_{h(CE)}$	Chip enable hold time
$t_{h(CS)}$	Chip select hold time
$t_{h(DA)}$	Data hold time

Symbol	Parameter—definition
$t_{h(RD)}$	Read hold time
$t_{h(WR)}$	Write hold time
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_{su}	Setup time—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
$t_{su(AD)}$	Address setup time
$t_{su(AD\text{-}WR)}$	Address setup time with respect to write
$t_{su(CE\text{-}P)}$	Chip enable setup time with respect to precharge
$t_{su(CS)}$	Chip select setup time
$t_{su(CS\text{-}WR)}$	Chip select setup time with respect to write
$t_{su(DA)}$	Data setup time
$t_{su(P\text{-}CE)}$	Precharge setup time with respect to chip enable
$t_{su(RD)}$	Read setup time
$t_{su(WR)}$	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_w	Pulse width—the time interval between specified reference points on the leading and trailing edges of the waveforms
$t_{w(CE)}$	Chip enable pulse width
$t_{w(CEH)}$	Chip enable high pulse width
$t_{w(CEL)}$	Chip enable low pulse width
$t_{w(CS)}$	Chip select pulse width
$t_{w(RD)}$	Read pulse width
$t_{w(WR)}$	Write pulse width
$t_{w(\phi)}$	Clock pulse width
t_{wr}	Write recovery time—the time interval between the termination of a write pulse and the initiation of a new cycle
T_a	Ambient temperature
T_{opr}	Operating temperature
T_{stg}	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 range 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

Note: The symbols shown here are, with some exceptions, extracted from IEC publication 148.

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. Pictures of some of the test equipment used are shown in Figs. 2 ~ 5.

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 EIAJ-IC-121 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
	Dropping	75cm, 3 times, wood plate, Y_1 direction
	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	○ Inspection of leads, plating, soldering and welding ○ Inspection of materials, sealing, package and marking ○ Visual inspection of other items of the specifications ○ Use of stereo microscopes, metallurgical microscopes, X-ray photographic equipment, fine leakage and gross leakage testers in the examination
2. Electrical tests	○ Checking for open circuits, short circuits and parametric degradation by electrical parameter measurement ○ Observation of characteristics by a synchroscope or a curve tracer and checking of important physical characteristics by electrical characteristics ○ Stress tests such as environmental or life tests, if required
3. Internal examination	○ Removal of the cover of the device, the optical inspection of the internal structure of the device ○ Checking of the silicon chip surface ○ Measurement of electrical characteristics by probes, if applicable ○ Use of SEM, XMA and infrared microscanner if required
4. Chip analysis	○ Use of metallurgical analysis techniques to supplement analysis of the internal examination ○ Slicing for cross-sectional inspection ○ Analysis of oxide film defects ○ Analysis of diffusion defects

QUALITY ASSURANCE AND RELIABILITY TESTING

Fig. 1 Quality assurance system

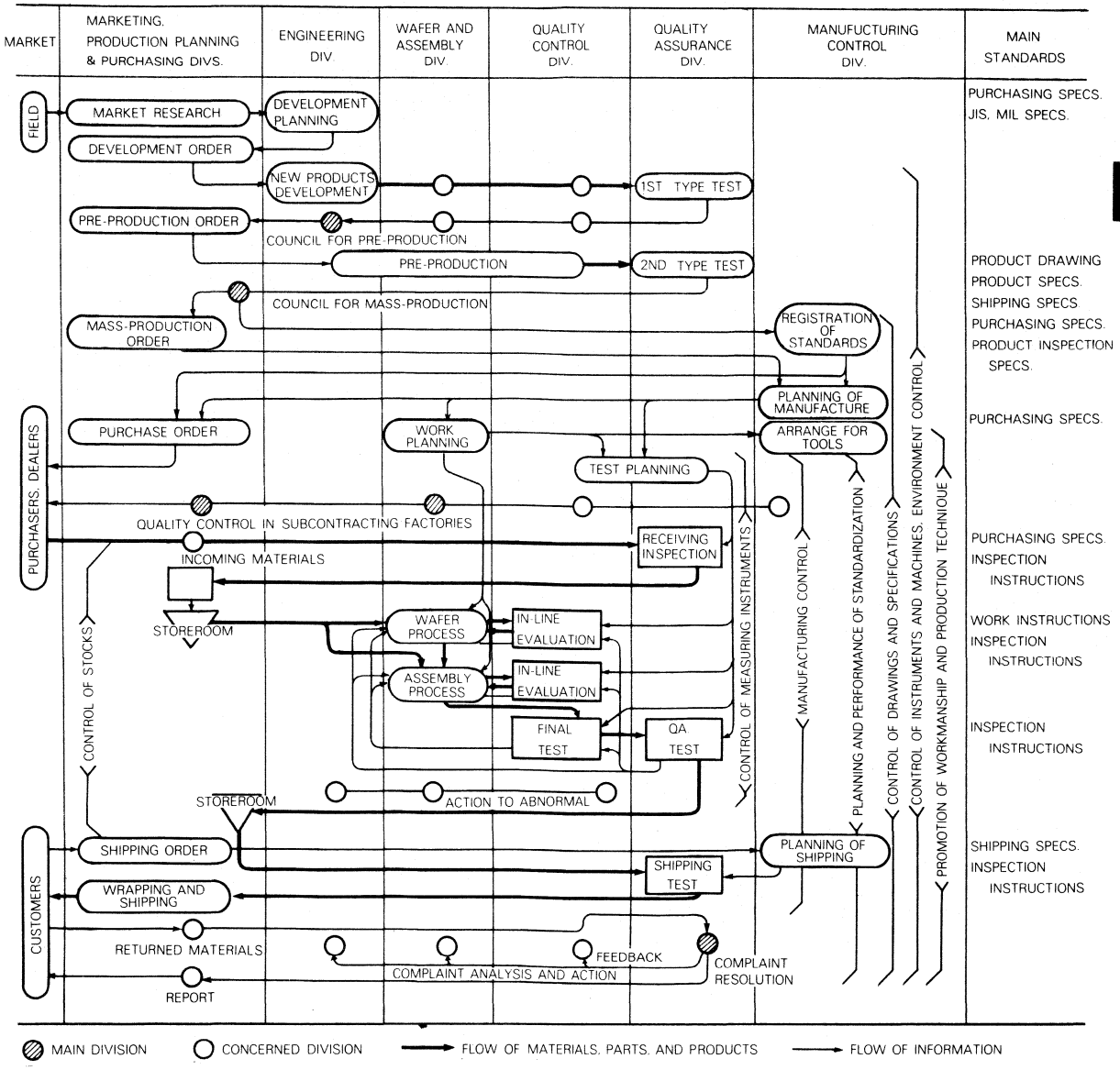


Fig. 2 Large-scale test system for LSIs

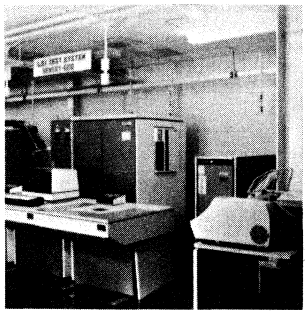


Fig. 3. Monitored temperature cycling tester

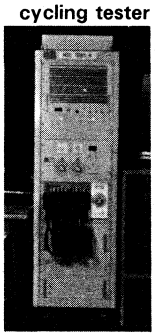


Fig. 4 Helium leakage tester

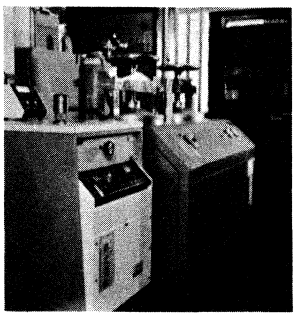
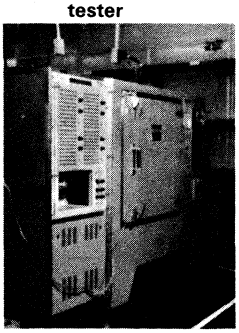


Fig. 5 Operating life tester



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. 6.
2. DC biased test: Durability is tested at high temperature biasing DC voltage, as shown in Fig. 7.
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.

Fig. 6 Operating life test procedure (for M5L2107BP, S 4K-bit dynamic RAM)

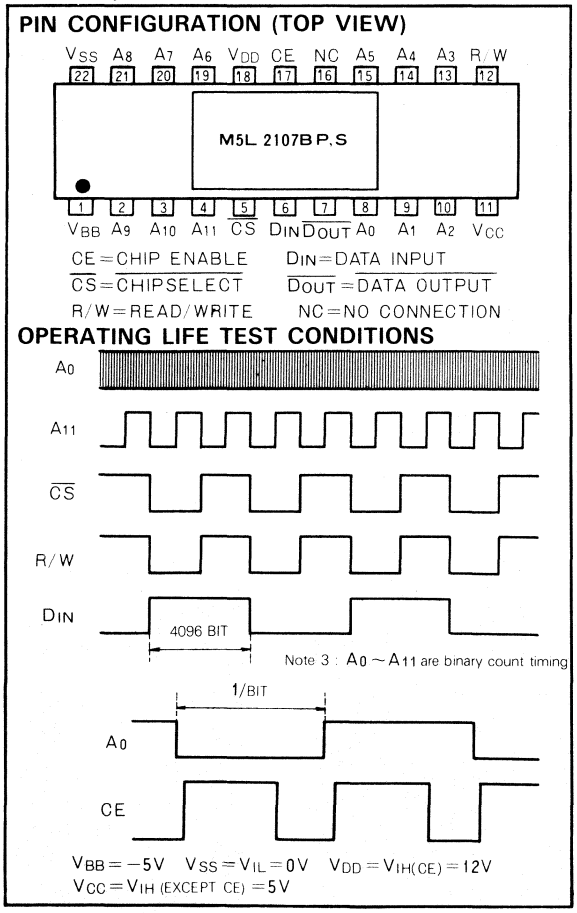


Fig. 7 DC biased test procedure (for M5L2102AP 1K-bit static RAM)

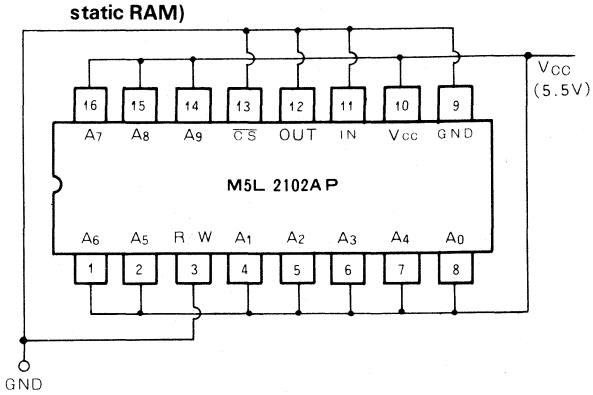


Table 3 Typical results of memory MOS LSI life tests

Type number	Package	Test	Temp °C	No. of samples	Component hours	No. of failures	Remarks
M5L 2102AP	16-pin Plastic-molded DIL	Operating life	80°C	40	80,000	0	
			125°C	173	213,000	0	
		DC biased	125°C	40	80,000	0	
		Hi-temp. stg.	150°C	5	5,000	0	
			200°C	5	5,000	0	
M5L 2111AP	18-pin plastic-molded DIL	Operating life	80°C	22	44,000	0	
			125°C	22	44,000	0	
		DC biased	125°C	22	44,000	0	
		Hi-temp. stg.	125°C	22	22,000	0	
			150°C	22	22,000	0	
M5L 2107BS	22-pin metal-sealed ceramic DIL	Operating life	80°C	39	88,000	0	Functional failure (at 240h)
			125°C	149	271,000	1	
		DC biased	125°C	66	137,000	0	
M5L 2114LS	18-pin metal-sealed cermic DIL	Operating life	125°C	44	88,000	0	
		DC biased	125°C	22	66,000	0	
		Hi-temp. stg.	150°C	22	22,000	0	
M5K 4116S	16-pin metal-sealed ceramic DIL	Operating life	125°C	172	234,000	0	
		Hi-temp. stg.	150°C	44	44,000	0	
			200°C	22	22,000	0	
M5K 4116P	16-pin plastic-molded DIL	Operating life	125°C	152	152,000	0	
		Hi temp. stg.	150°C	38	38,000	0	
M5L 5101LP	22-pin plastic-molded DIL	Operating life	125°C	88	110,000	0	
		Hi-temp. stg.	150°C	44	44,000	0	

4.2 Typical Results of Failure Analyses

Accelerated testing under conditions more severe than normal operating conditions is used to observe failures of moisture resistance, of wire bonding, of surge voltage destruction and of vapor-deposited aluminum interconnection. Typical results are shown below.

4.2.1. Failure in Moisture Resistance

An example of the results of steam pressure testing, performed to evaluate the moisture resistance of a plastic molded package, is shown in Fig. 8. The vapor-deposited aluminum interconnection was corroded due to moisture penetration.

4.2.2. Failure of Wire Bonding

An example of a failure during the monitored temperature cycling test for evaluating the reliability of the wire bonding of the inner leads of the IC is shown in Fig. 9. The cause of this failure may have been the opening of the inner lead bonding because of a difference in thermal expansion coefficients of metal and resin producing a stress on the inner lead.

4.2.3. Failure Due to Surge Voltage

Many integrated circuits fail in the field due to a surge voltage. Surge voltage marginal tests have been performed to reproduce this failure for analysis of the destruction.

Examples of failures during this test are shown in Figs. 10 ~ 13. Figs. 10 and 11 indicate the existence of a bridge that was confirmed by an X-ray microanalyzer. Figs. 12 and 13 indicate the existence of a hot spot that was confirmed by an infrared microscanner.

4.2.4. Failure of Vapor-Deposited Interconnections

Fig. 14 shows an open-circuit vapor-deposited aluminum interconnection, at a high current density region, caused by the operating life test. This test is performed as a step stress test to investigate IC degradation and failure by temperature and voltage. This phenomenon is due to aluminum electromigration, which is observed when high-current loads are applied to a vapor-deposited aluminum interconnection.

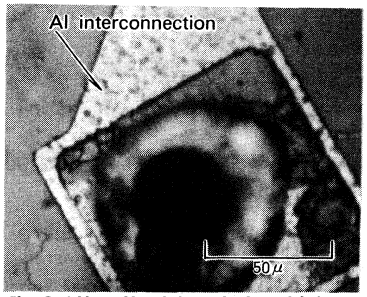


Fig. 8 Corrosion of vapor-deposited aluminum interconnection, analyzed by metallurgical microscope

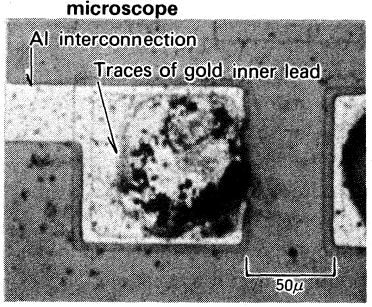


Fig. 9 Lift off of bonded gold inner lead, analyzed by metallurgical microscope

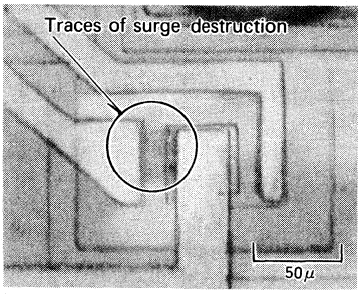


Fig. 10 Surge destruction, analyzed by metallurgical microscope

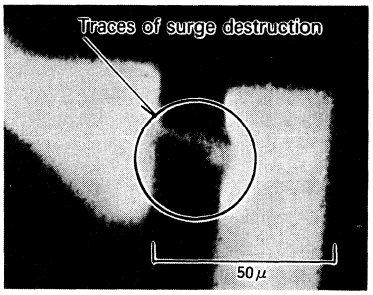


Fig. 11 Enlargement of aluminum bridge in Fig. 10, analyzed by XMA-Al kα

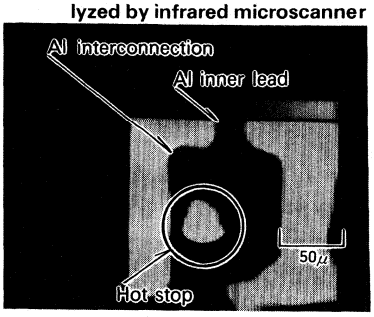


Fig. 12 Hot spot at bonding head, analyzed by infrared microscanner

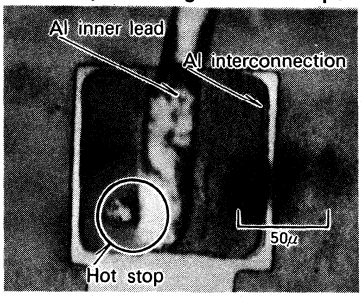


Fig. 13 Junction in Fig. 12 after removal of aluminum, analyzed by metallurgical microscope

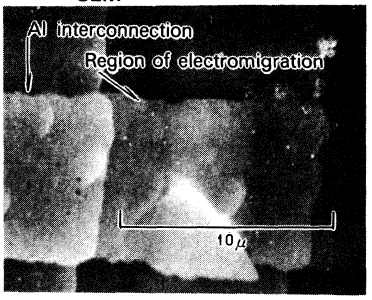


Fig. 14 Electromigration of aluminum interconnection, analyzed by SEM

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

4

1024-BIT(256-WORD BY 4-BIT) NON-VOLATILE STATIC RAM

DESCRIPTION

The M58656S is a 256-word by 4-bit non-volatile static RAM, fabricated with the P-channel MNOS process. The basic circuit of each memory cell consists of an ordinary flip-flop and a pair of electrically rewritable MNOS memory transistors for non-volatile information storage.

The non-volatile operation is effected by having MG (memory gate) signals applied when the power supply is turned on or off. Inputs and outputs are TTL-compatible through attachment of a pull-up resistor to the V_{SS} terminal. The data terminals are common for both inputs and outputs.

FEATURES

- Non-volatile operation: No backup power supply required against power-supply interruption
- Static operation: No clock required
- Access time: 1.5μs (max)
- The chip-enable signal facilitates the expansion of memory capacity
- Outputs are three-state, with OR-tie capability
- RAM/ROM use is allowed by means of $\overline{NR}$ signal
- Interchangeable with Toshiba TMM142C in pin configuration and electrical characteristics

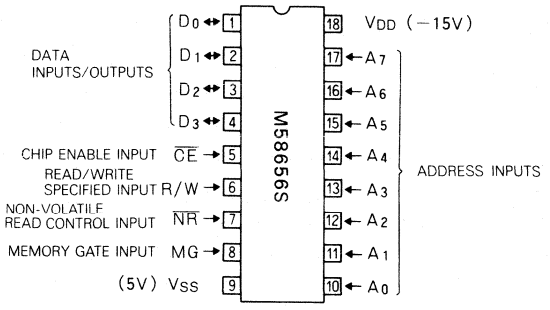
APPLICATION

- Small-capacity non-volatile memory systems

FUNCTION

The M58656S, adopting the memory cells mentioned above, permits high-speed operations as an ordinary static RAM while the power supply is on.

PIN CONFIGURATION (TOP VIEW)

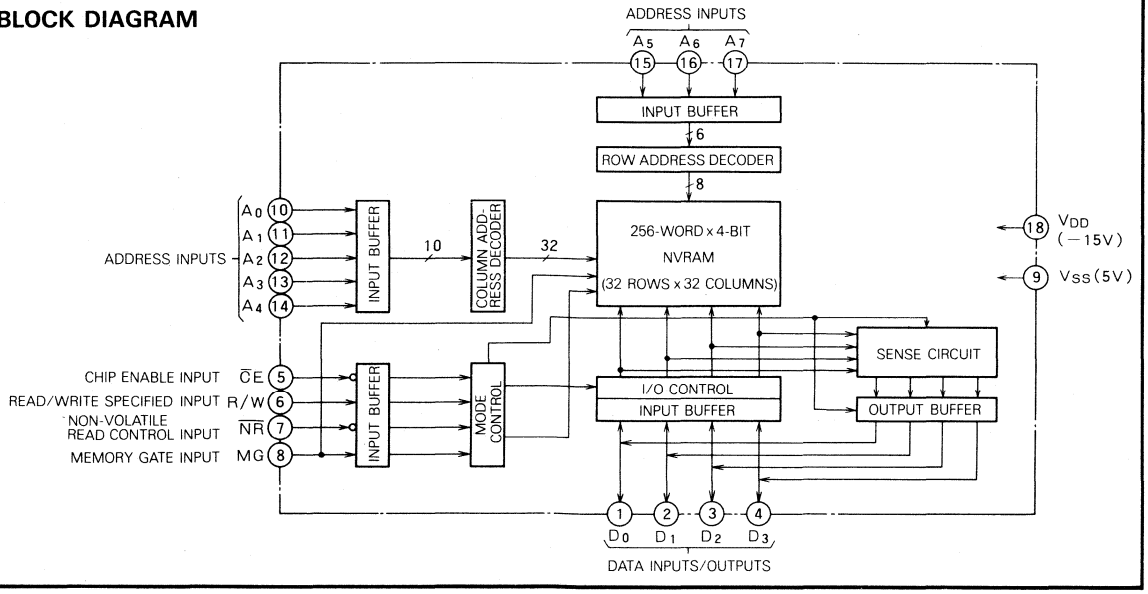


Outline 18S1

At an interruption of power a non-volatile write signal is applied to the MG terminal causing the content of the flip-flop to transfer into the MNOS memory transistor for retention as non-volatile information for upwards of one year.

On the other hand, when the power supply is turned on, a non-volatile read signal is applied to the MG terminal and thus, the non-volatile information in each MNOS memory transistor reappears in each flip-flop.

BLOCK DIAGRAM



1024-BIT (256-WORD BY 4-BIT) NON-VOLATILE STATIC RAM

EXPLANATION OF FUNCTIONS

The following five operational modes are available:

(1) Ordinary RAM operations (read/write memory mode)
(2) MG (memory gate) erase mode
(3) MG (memory gate) write mode
(4) Non-volatile read mode 1
(5) Non-volatile read mode 2

Read/Write Memory Mode

Holding the MG and $\overline{NR}$ inputs at the V_{SS} level allows the flip-flop of each memory cell to operate independently of its associated MNOS memory transistor. Thus, in this mode, this memory permits high-speed operation as an ordinary static read/write memory.

MG (Memory Gate) Erase Mode

All the bits of non-volatile information in the MNOS memory transistors may be simultaneously erased by applying positive pulses of V_{MGE} and $t_{w\,(MGE1)}$ to the MG input. This mode is available at all times other than the period of non-volatile read mode 1 or 2.

MG (Memory Gate) Write Mode

In this mode, the read/write memory information at power interruptions or specific information during the operation of read/write memory is made non-volatile. When a negative pulse of V_{MGW} and $t_{w\,(MGW1)}$ is applied as the MG input with V_{DD} more negatively biased than $V_{WT'}$ the information in the flip-flop is caused to transfer into a pair of MNOS memory transistors, so becoming non-volatile information.

Non-Volatile Read Mode 1

This mode is for reading the non-volatile information held in the MNOS memory transistors into the memory cell. The power supply V_{DD} and the memory gate input MG are caused to rise gradually against V_{SS} and the latter is then allowed to reach the specified value V_{MGR} and return to the V_{SS} level, resulting in the reproduction of the information that was stored immediately before the power interruption.

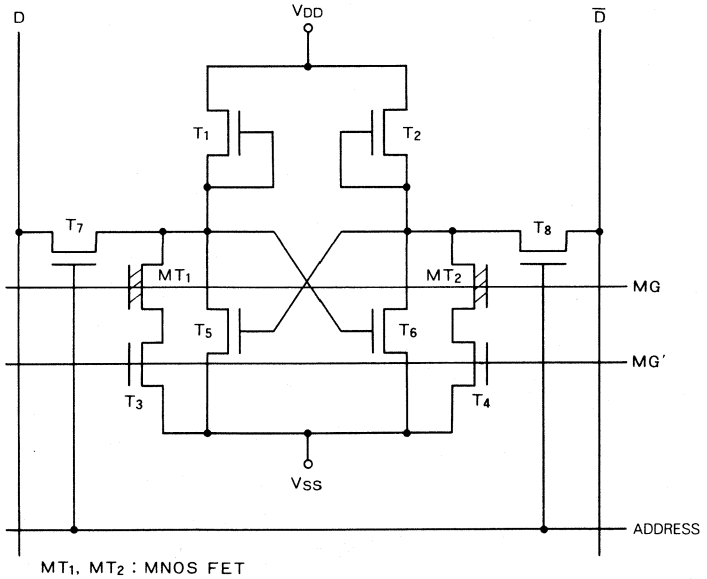
Non-Volatile Read Mode 2

In this mode, the $\overline{NR}$ input is utilized in the course of read/write memory operation to allow a reading of the non-volatile information stored in the MNOS memory transistor to interrupt. The non-volatile information is read into the memory cell by holding the $\overline{NR}$ input at the low level and applying an MG input of the same wave-form as in the non-volatile read mode 1. That is, it is possible to allow the information in the MNOS memory transistor to be read as a ROM at any points during the course of RAM operation.

Non-Volatile Memory Cell

The memory cell equivalent circuit used in the M58656S is shown below.

DIAGRAM OF EQUIVALENT CIRCUIT FOR NON-VOLATILE MEMORY CELL



MT_1, MT_2 : MNOS FET

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	0.3 ~ −30	V
$V_{I(MG)}$	Memory gate input voltage		40 ~ −40	V
V_I	Input voltage		0.3 ~ −30	V
V_O	Output voltage		0.3 ~ −30	V
t_{MG}	Memory gate input pulse width		1	s
Pd	Maximum power dissipation	Ta = 25℃	875	mW
Topr	Operating free-air ambient temperature range		0 ~ 70	℃
Tstg	Storage temperature range		−55 ~ 150	℃

4

RECOMMENDED (For Read/Write Memory) (Ta = 0 ~ 70℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	−14.25	−15	−15.75	V
V_{SS}	Supply voltage	4.75	5	5.25	V
V_{IH}	High-level input voltage	V_{SS}−1.5		V_{SS}+0.3	V
V_{IL}	Low-level input voltage	−3		0.8	V

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 70℃, V_{DD} = −15V ±5%, V_{SS} = 5V ±5%, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{IH}	High-level input voltage		V_{SS}−1.5		V_{SS}+0.3	V
V_{IL}	Low-level input voltage		−3		0.8	V
I_{IH}	High-level input current	$V_I = V_{SS}$			1	μA
I_{IL}	Low-level input current	V_I = −3V			±1	μA
I_O	Output leakage current	$V_{I(\overline{CE})} = V_{IH}$ or $V_{I(R/W)} = V_{IL}$ V_O = 0V ~ V_{SS}−1V			±10	μA
V_{OH}	High-level output voltage	I_{OH} = −0.4mA	V_{SS}−1			V
I_{OH}	High-level output current	V_{OH} = 4V	−0.4			mA
V_{OL}	Low-level output voltage	I_{OL} = 1.6mA			0.4	V
I_{OL}	Low-level output current	V_{OL} = −0.6V, Ta = 70℃			5	mA
I_{DD}	Supply current from V_{DD}	I_O = 0mA, Typical values are at Ta = 25℃		−25	−40	mA
$V_{I(MG)}$	MG input voltage		V_{SS}−1	V_{SS}	V_{SS}+1	V
$I_{I(MG)}$	MG input current	$V_{I(MG)} = V_{SS}$ ± 1V			0.1	mA
C_i	Input capacitance	V_I = 0V, f = 1MHz, Ta = 25℃			8	pF
C_O	Output capacitance	V_O = 0V, f = 1MHz, Ta = 25℃			10	pF

Note 1 : Current flowing into an IC is positive (no sign).

1024-BIT (256-WORD BY 4-BIT) NON-VOLATILE STATIC RAM

TIMING REQUIREMENTS (Ta = 0 ~ 70℃, V_{DD} = −15V ± 5%, V_{SS} = 5V ± 5%, unless otherwise noted)

Read Cycle

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{c(RD)}$	Read cycle time	Input pulse $V_{IH} = V_{SS} - 1.5V$, $V_{IL} = 0.8V$ $t_r = t_f \leqq 25ns$	1750			ns
$t_{su(AD-\overline{CE})}$	Address setup time with respect to $\overline{CE}$		50			ns
$t_{h(AD-\overline{CE})}$	Address hold time with respect to $\overline{CE}$		250			ns

Write Cycle

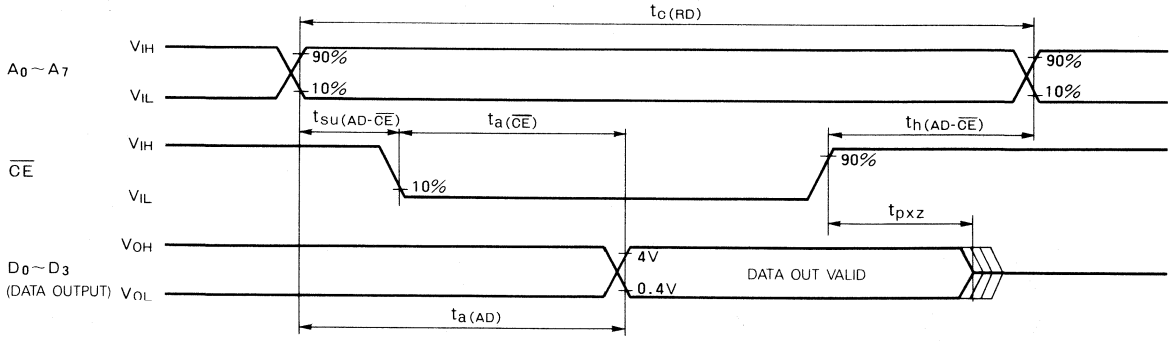
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{c(WR)}$	Write cycle time	Input pulse $V_{IH} = V_{SS} - 1.5V$ $V_{IL} = 0.8V$ $t_r = t_f \leqq 25ns$	1000			ns
$t_{su(AD-\overline{CE})}$	Address setup time with respect to $\overline{CE}$		50			ns
$t_{su(AD-R/W)}$	Address setup time with respect to R/W		50			ns
$t_{h(AD-\overline{CE})}$	Address hold time with respect to $\overline{CE}$		250			ns
$t_{w(WR)}$	Write pulse width		500			ns
$t_{su(DA)}$	Data setup time		400			ns
$t_{h(DA)}$	Data hold time		200			ns

SWITCHING CHARACTERISTICS (For Read Cycle) (Ta = 0 ~ 70℃, V_{DD} = −15V ± 5%, V_{SS} = 5V ± 5%, unless otherwise noted)

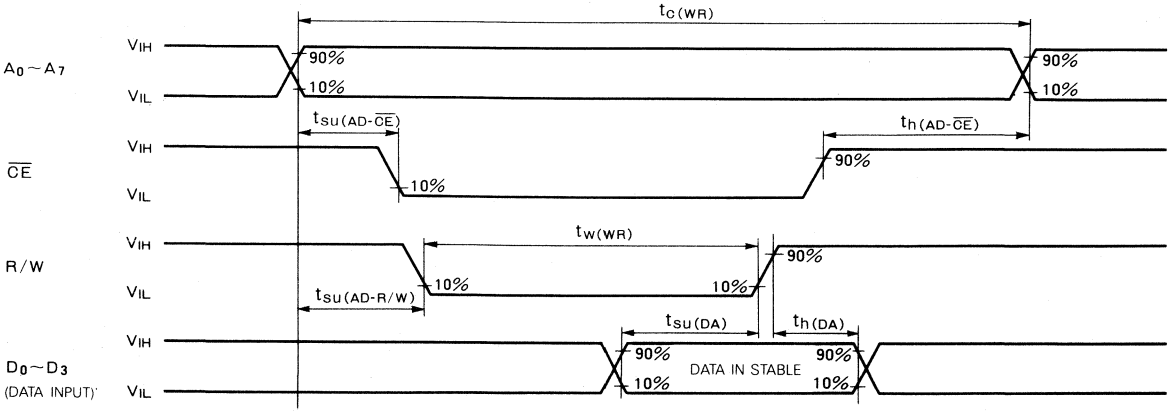
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{a(AD)}$	Address access time	Load = 1TTL, C_L = 100pF			1500	ns
$t_{a(\overline{CE})}$	Chip enable access time				1450	ns
t_{pxz}	Output disable time				600	ns

TIMING DIAGRAMS

Read Cycle ($MG = V_{I(MG)}$, $\overline{NR} = V_{IH}$)



Write Cycle ($MG = V_{I(MG)}$, $\overline{NR} = V_{IH}$)



MITSUBISHI ELECTRIC

NON-VOLATILE OPERATIONS

Electrical Characteristics (Operations 1 and 2) (T_a=0～70℃, V_{DD}=−15V±5%, V_{SS}=5V±5%, unless otherwise noted)

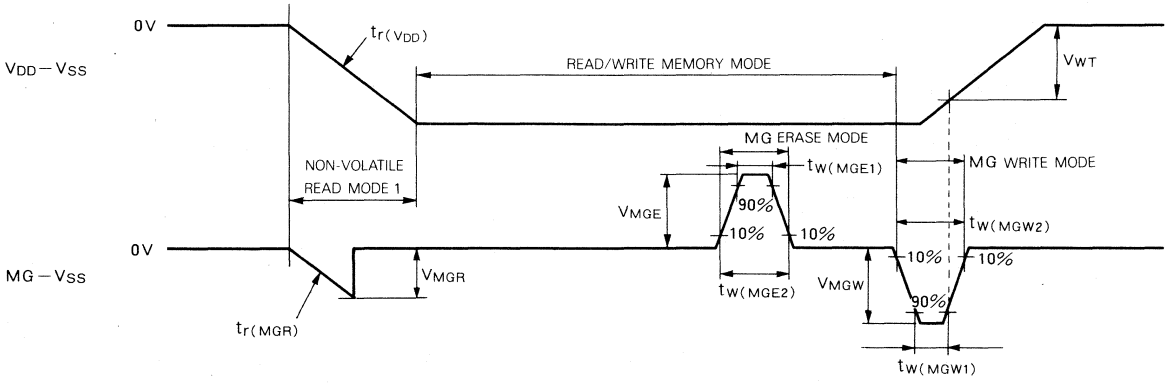
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{MGE}	MG erase voltage with respect to V_{SS}	$t_{W(MGE1)}=t_{W(MGW1)}$=0.75～1.25ms	28	29	31	V
V_{MGW}	MG write voltage with respect to V_{SS}	$V_{MGE}\geqq\lvert V_{MGW}\rvert$	−26	−28	−30	V
V_{MGR}	MG read voltage with respect to V_{SS}		−10		−15	V
V_{WT}	Supply voltage with respect to V_{SS} necessary to MG write		−15			V
$t_{W(MGE/W,1)}$	MG pulse width (1)		0.75	1	1.25	ms
$t_{W(MGE/W,2)}$	MG pulse width (2)				10	ms
$t_{r(MGR)}$	MG read shape rise time	$V_{MGR}=V_{SS}-10V$			20	V/ms
$t_{r(V_{DD})}$	Supply voltage shape rise time				20	V/ms
$t_{r(MGR)}/t_{r(V_{DD})}$	MG read, supply voltage shape rise time ratio	$V_{DD}=0V\sim V_{MGR}$	0.9	1	1.1	—
t_S	Unpowered nonvolatile data retention time	V_{MGE}=28V, V_{MGW}=−28V, $t_{W(MGE/W,1)}$=1ms	1			year
N_W	Number of erase write cycle		10^5			times
$I_{I(MG)}$	MG input current	V_{MGE}=30V, V_{MGW}=−30V			±0.4	mA

Timing Requirements (Operation 2) (T_a=0～70℃, V_{DD}=−15V±5%, V_{SS}=5V±5%, unless otherwise noted)

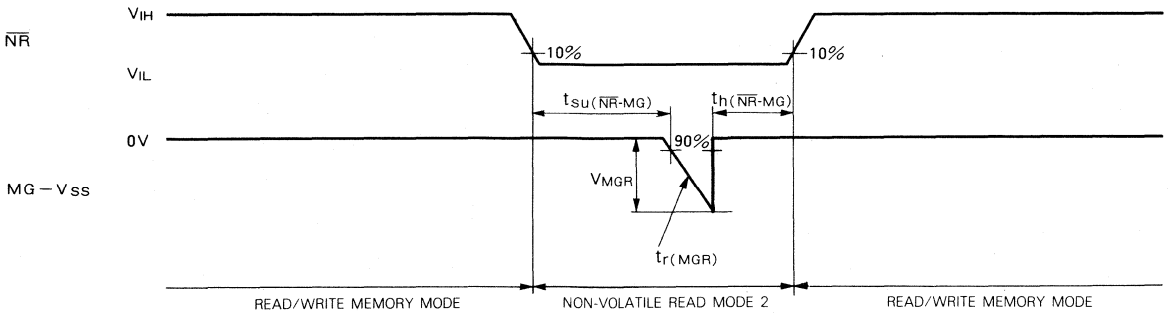
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{su(\overline{NR}\text{-}MG)}$	$\overline{NR}$ setup time with respect to MG		1.45			μs
$t_{h(\overline{NR}\text{-}MG)}$	$\overline{NR}$ hold time with respect to MG		0			μs

Timing Diagrams

Operation 1 ($\overline{NR}=V_{IH}$)



Operation 2 ($\overline{CE}=V_{IH}$)



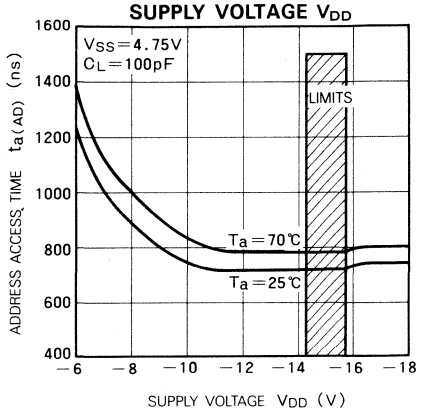
Note 2 : Both the MG erase mode and MG write modes in non-volatile operation 2 are the same as in non-volatile operation 1.

3 : Non-volatile read mode 1 is warranted only in the case of custom specification.

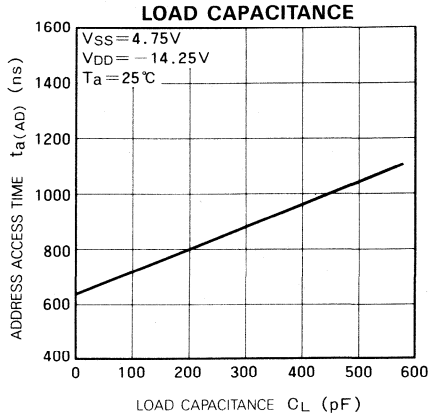
1024-BIT (256-WORD BY 4-BIT) NON-VOLATILE STATIC RAM

TYPICAL CHARACTERISTICS

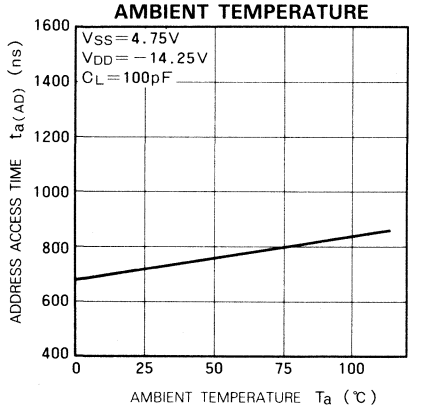
ADDRESS ACCESS TIME VS. SUPPLY VOLTAGE V_{DD}



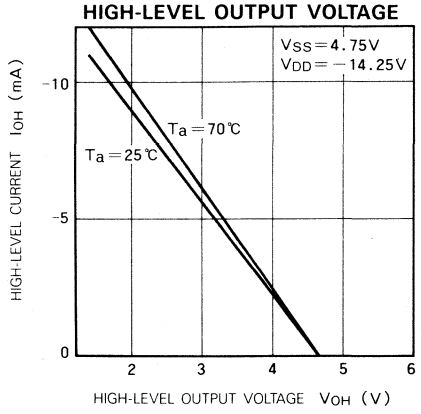
ADDRESS ACCESS TIME VS. LOAD CAPACITANCE



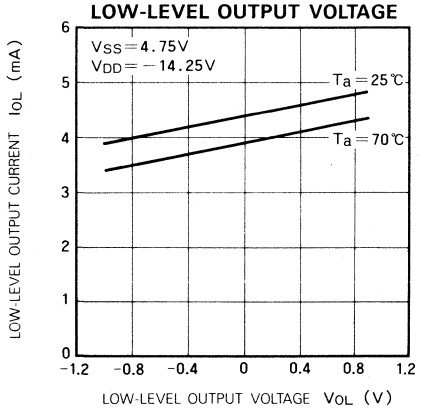
ADDRESS ACCESS TIME VS. AMBIENT TEMPERATURE



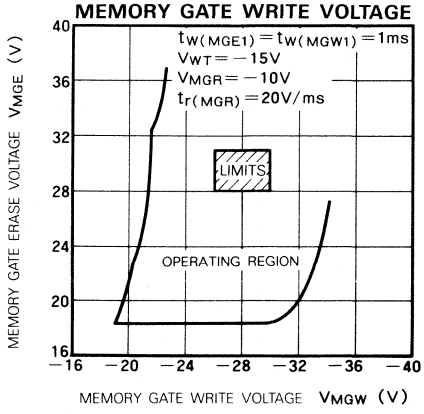
HIGH-LEVEL OUTPUT CURRENT VS. HIGH-LEVEL OUTPUT VOLTAGE



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



MEMORY GATE ERASE VOLTAGE VS. MEMORY GATE WRITE VOLTAGE



MITSUBISHI ELECTRIC

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

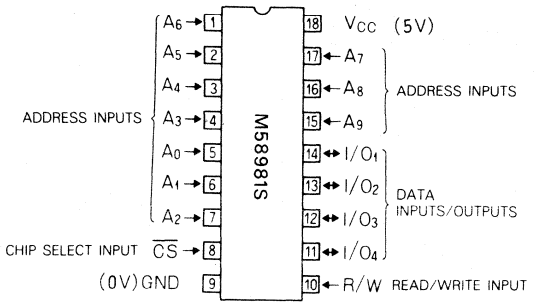
DESCRIPTION

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.

PIN CONFIGURATION (TOP VIEW)



Outline 18S1

FEATURES

- Access time: 450ns (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.

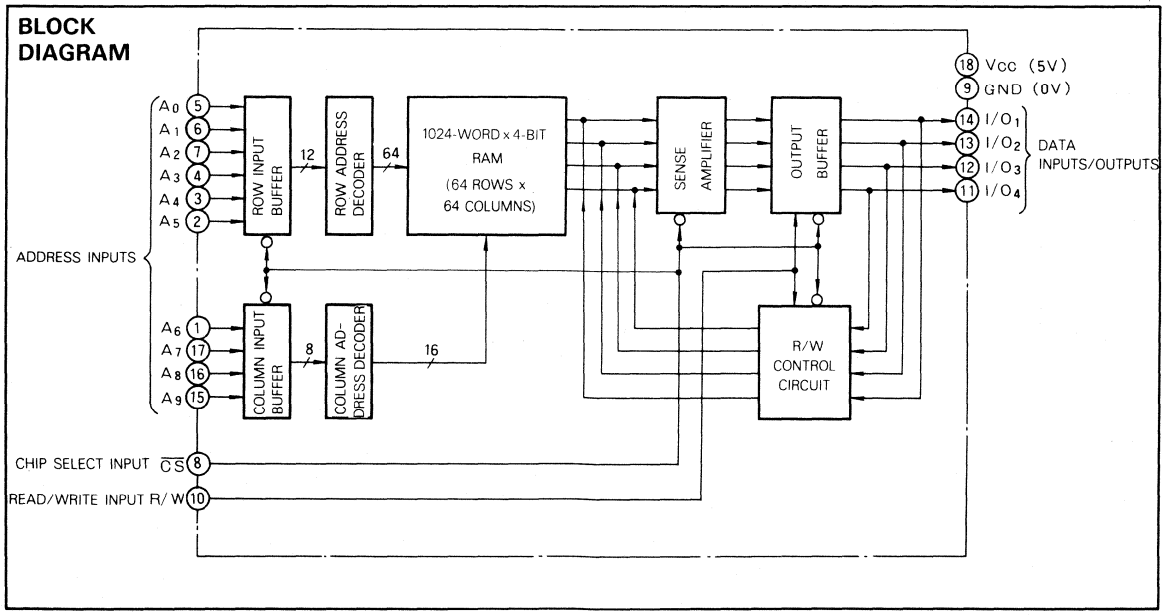
During a write cycle, when a location is designated by address signals $A_0 \sim A_9$ and signal R/W goes low, the data of the I/O at that time is written.

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.

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	T_a = 25°C	1000	mW
T_{opr}	Operating free-air ambient temperature range		0 ~ 70	°C
T_{stg}	Storage temperature range		−65 ~ 150	°C

RECOMMENDED OPERATING CONDITIONS (T_a = 0 ~ 70°C, 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	−0.3		0.65	V
V_{IH}	High-level input voltage	2.2		V_{CC}	V

ELECTRICAL CHARACTERISTICS (T_a = 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		2.2		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} = 2 mA			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}	$\overline{CS}$ ≦ 0.01V, other inputs = V_{CC}, Output open		9	25	mA
I_{CC2}	Supply current from V_{CC}	$\overline{CS}$ ≦ 0.01V, other inputs = 2.2V, Output open		13	30	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) (T_a = 0 ~ 70°C, V_{CC} = 5V ± 10%, unless otherwise noted)

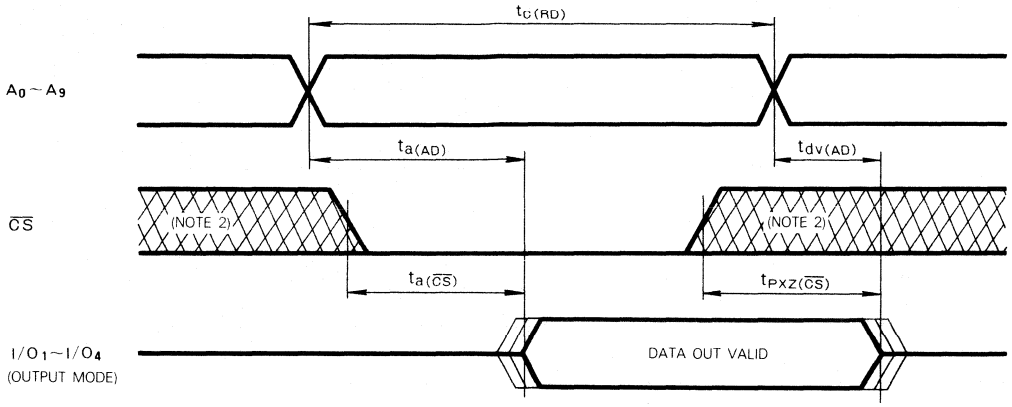
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{c(WR)}$	Write cycle time	Input pulse V_{IH} = 2.2V, V_{OH} = 0.65V, $t_r = t_f$ = 20ns, Reference level = 1.5V, Load = 1TTL, C_L = 100pF	450			ns
$t_{su(AD)}$	Address setup time with respect to write pulse		130			ns
$t_{w(WR)}$	Write pulse width		250			ns
t_{wr}	Write recovery time		50			ns
$t_{su(DA)}$	Data setup time		250			ns
$t_{h(DA)}$	Data hold time		0			ns
$t_{su(\overline{CS})}$	Chip select setup time		350			ns
$t_{pxz(WR)}$	Output disable time with respect to write pulse				100	ns

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

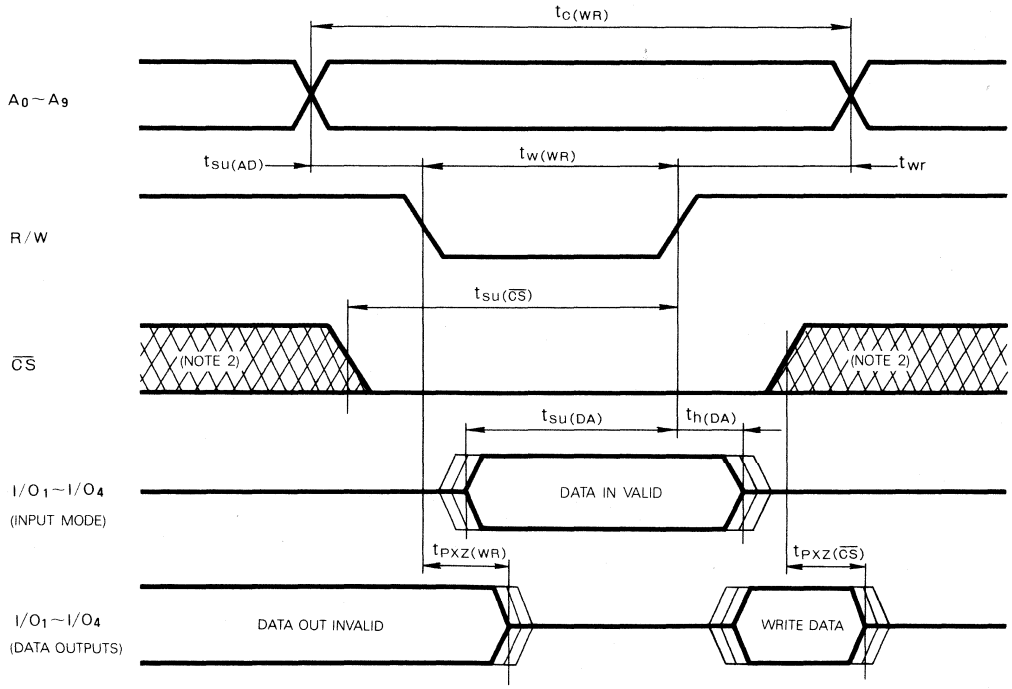
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{c(RD)}$	Read cycle time	Input pulse V_{IH} = 2.2V, V_{OH} = 0.6V, $t_r = t_f$ = 20ns, Reference level = 1.5V, Load = 1TTL, C_L = 100pF	450			ns
$t_{a(AD)}$	Address access time				450	ns
$t_{a(\overline{CS})}$	Chip select access time				450	ns
$t_{PXZ(\overline{CS})}$	Output disable time with respect to chip select				130	ns
$t_{dv(AD)}$	Data valid time with respect to address		0			ns

TIMING DIAGRAMS

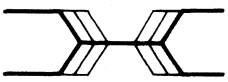
Read Cycle



Write Cycle



Note 2: Hatching indicates the state is unknown.



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

POWER-DOWN OPERATION

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

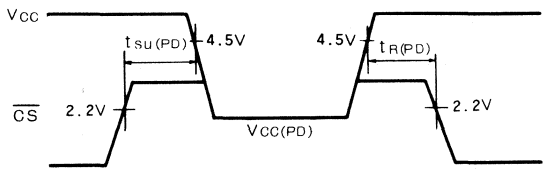
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$V_{CC(PD)}$	Power-down supply voltage		2			V
$V_I(\overline{CS})$	Power-down chip select input voltage	2.2V ≦ $V_{CC(PD)}$ ≦ V_{CC}	2.2			V
		2V ≦ $V_{CC(PD)}$ ≦ 2.2V	$V_{CC(PD)}$			V
$I_{CC(PD)}$	Power-down supply current from V_{CC}	V_{CC}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

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

Timing Requirements (T_a = 0 ~ 70°C, V_{CC} = 5V ± 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



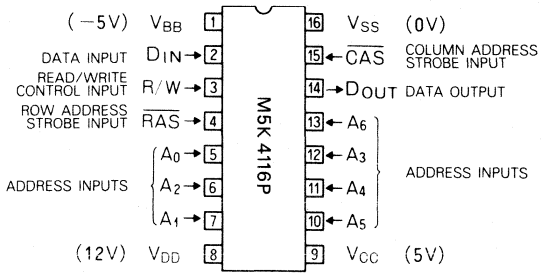
M5K 4116 P-2, S-2; P-3, S-3; P-4, S-4

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.

PIN CONFIGURATION (TOP VIEW)



Outline 16P1 (M5K4116P)
16S1 (M5K4116S)

FEATURES

Type name	Access time (max) (ns)	Cycle time (min) (ns)	Power dissipation (typ) (mW)
M5K 4116 P-2, S-2	150	320	330
M5K 4116 P-3, S-3	200	375	280
M5K 4116 P-4, S-4	250	410	260

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

APPLICATION

- Main memory unit for computers

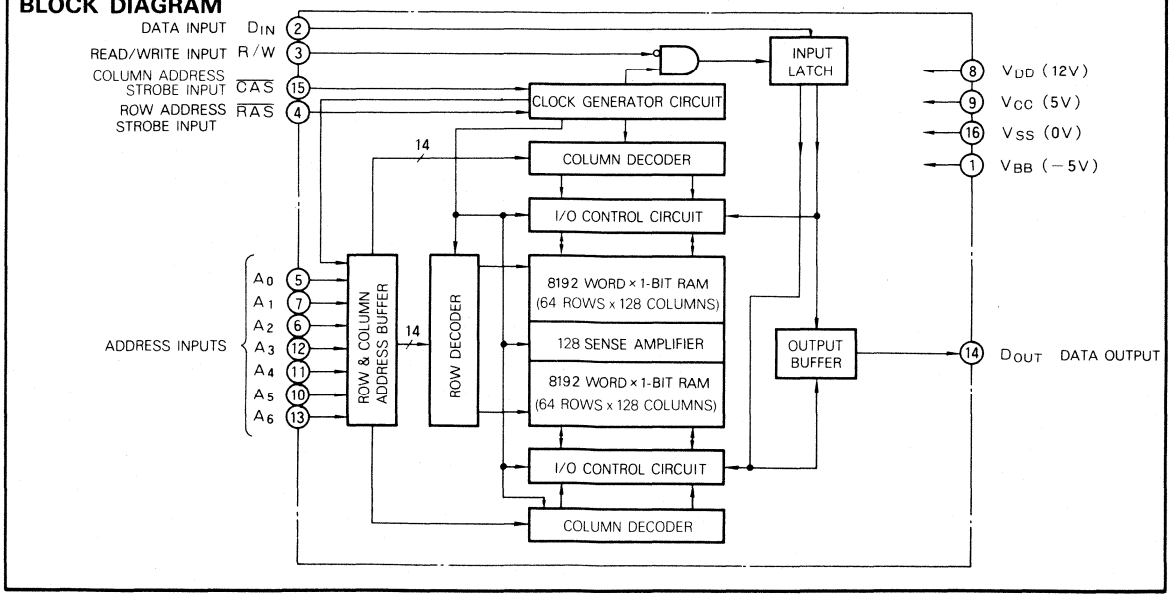
FUNCTION

The M5K4116P and S 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	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 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 and 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:

1. The delay time from $\overline{RAS}$ to $\overline{CAS}$ $t_{d(\overline{RAS}\text{-}\overline{CAS})}$ is set between the minimum and maximum values of the limits. In this case, the internal $\overline{CAS}$ control signals are inhibited almost until $t_{d(\overline{RAS}\text{-}\overline{CAS})max}$ ('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 $t_{d(\overline{RAS}\text{-}\overline{CAS})}$ 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 4116 P and S 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 4116 P and 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, 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 4116 P and S 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 4116 P and S 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 4116 P and 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 supply is applied to V_{DD}.

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

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter		Conditions	Limits	Unit
V_{DD}	Supply voltage		With respect to V_{BB}	−0.5 ~ 20	V
V_{CC}	Supply voltage			−0.5 ~ 20	V
V_{SS}	Supply voltage			−0.5 ~ 20	V
V_I	Input voltage			−0.5 ~ 20	V
V_O	Output voltage			−0.5 ~ 20	V
V_{DD}	Supply voltage		With respect to V_{SS}	−1 ~ 15	V
V_{CC}	Supply voltage			−1 ~ 15	V
$V_{BB}-V_{SS}$	Supply voltage		$V_{DD}-V_{SS}>0$	0	V
I_O	Output current			50	mA
Pd	Power dissipation	M5K4116S	Ta = 25 ℃	1000	mW
		M5K4116P	Ta = 25 ℃	700	mW
Topr	Operating free-air temperature range			0 ~ 70	℃
Tstg	Storage temperature range	M5K4116S		−65 ~ 150	℃
		M5K4116P		−40 ~ 125	℃

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

(Ta = 0 ~ 70 ℃, V_{DD} = 12V ± 10%, V_{CC} = 5V ± 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 V_{CC}, 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	$\overline{RAS} = V_{IL}$, $\overline{CAS}$ cycling $t_{c(PG)}$ = 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.

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

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

($Ta = 0 \sim 70°C$, $V_{DD} = 12V \pm 10\%$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$, $-5.7V \leq V_{BB} \leq -4.5V$, unless otherwise noted. See notes 5, 6, and 7.)

Symbol	Parameter	Alternative Symbol	M5K 4116 P-2, S-2 Limits		M5K 4116 P-3, S-3 Limits		M5K 4116 P-4, S-4 Limits		Unit
			Min	Max	Min	Max	Min	Max	
$t_{C(REF)}$	Refresh cycle time	t_{REF}		2		2		2	ms
$t_{W(\overline{RAS}H)}$	$\overline{RAS}$ high pulse width	t_{RP}	100		120		150		ns
$t_{W(\overline{RAS}L)}$	$\overline{RAS}$ low pulse width	t_{RAS}	150	10000	200	10000	250	10000	ns
$t_{W(\overline{CAS}L)}$	$\overline{CAS}$ low pulse width (Note 8)	t_{CAS}	100		135		165		ns
$t_{h(\overline{RAS}\text{-}\overline{CAS})}$	$\overline{CAS}$ hold time with respect to $\overline{RAS}$	t_{CSH}	150		200		250		ns
$t_{h(\overline{CAS}\text{-}\overline{RAS})}$	$\overline{RAS}$ hold time with respect to $\overline{CAS}$	t_{RSH}	100		135		165		ns
$t_{d(\overline{RAS}\text{-}\overline{CAS})}$	Delay time, $\overline{RAS}$ to $\overline{CAS}$ (Note 9)	t_{RCD}	20	50	25	65	35	85	ns
$t_{d(\overline{CAS}\text{-}\overline{RAS})}$	Delay time, $\overline{CAS}$ to $\overline{RAS}$	t_{CRP}	−20		−20		−20		ns
$t_{su(RA\text{-}\overline{RAS})}$	Row address setup time with respect to $\overline{RAS}$	t_{ASR}	0		0		0		ns
$t_{su(CA\text{-}\overline{CAS})}$	Column address setup time with respect to $\overline{CAS}$	t_{ASC}	−10		−10		−10		ns
$t_{h(\overline{RAS}\text{-}RA)}$	Row address hold time with respect to $\overline{RAS}$	t_{RAH}	20		25		35		ns
$t_{h(\overline{CAS}\text{-}CA)}$	Column address hold time with respect to $\overline{CAS}$	t_{CAH}	45		55		75		ns
$t_{h(\overline{RAS}\text{-}CA)}$	Column address hold time with respect to $\overline{RAS}$	t_{AR}	95		120		160		ns
t_{THL} t_{TLH}	Transition time	t_T	3	35	3	50	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 $3\mu s$ before the dummy cycles, or to keep the $\overline{RAS}$ high pulse width $t_{W(\overline{RAS}H)}$ more than $3\mu s$ 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 $t_{d(\overline{RAS}\text{-}\overline{CAS})} \geq t_{d(\overline{RAS}\text{-}\overline{CAS})}$ max. If $t_{d(\overline{RAS}\text{-}\overline{CAS})} < t_{d(\overline{RAS}\text{-}\overline{CAS})}$ max, $t_{W(\overline{CAS}L)}$ will be increased by the amount that $t_{d(\overline{RAS}\text{-}\overline{CAS})}$ has decreased.

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

SWITCHING CHARACTERISTICS ($Ta = 0 \sim 70°C$, $V_{DD} = 12V \pm 10\%$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$, $-5.7V \leq V_{BB} \leq -4.5V$, unless otherwise no

Read Cycle

Symbol	Parameter	Alternative Symbol	M5K 4116 P-2, S-2 Limits		M5K 4116 P-3, S-3 Limits		M5K 4116 P-4, S-4 Limits		Unit
			Min	Max	Min	Max	Min	Max	
$t_{C(RD)}$	Read cycle time	t_{RC}	320		375		410		ns
$t_{su(RD\text{-}\overline{CAS})}$	Read set-up time with respect to $\overline{CAS}$	t_{RCS}	0		0		0		ns
$t_{h(\overline{CAS}\text{-}RD)}$	Read hold time with respect to $\overline{CAS}$	t_{RCH}	0		0		0		ns
$t_{h(\overline{CAS}\text{-}OUT)}$	Data-out hold time	t_{OFF}	0	40	0	50	0	60	ns
$t_{a(\overline{CAS})}$	$\overline{CAS}$ access time (Note 10)	t_{CAC}		100		135		165	ns
$t_{a(\overline{RAS})}$	$\overline{RAS}$ access time (Note 11)	t_{RAC}		150		200		250	ns

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

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

Write Cycle

Symbol	Parameter	Alternative Symbol	M5K 4116 P-2, S-2 Limits		M5K 4116 P-3, S-3 Limits		M5K 4116 P-4, S-4 Limits		Unit
			Min	Max	Min	Max	Min	Max	
$t_{C(WR)}$	Write cycle time	t_{RC}	320		375		410		ns
$t_{su(WR\text{-}\overline{CAS})}$	Write set-up time with respect to $\overline{CAS}$ (Note 12)	t_{WCS}	−20		−20		−20		ns
$t_{h(\overline{CAS}\text{-}WR)}$	Write hold time with respect to $\overline{CAS}$	t_{WCH}	45		55		75		ns
$t_{h(\overline{RAS}\text{-}WR)}$	Write hold time with respect to $\overline{RAS}$	t_{WCR}	95		120		160		ns
$t_{h(WR\text{-}\overline{RAS})}$	$\overline{RAS}$ hold time with respect to write	t_{RWL}	50		70		85		ns
$t_{h(WR\text{-}\overline{CAS})}$	$\overline{CAS}$ hold time with respect to write	t_{CWL}	50		70		85		ns
$t_{W(WR)}$	Write pulse width	t_{WP}	45		55		75		ns
$t_{su(DA\text{-}\overline{CAS})}$	Data-in setup time with respect to $\overline{CAS}$	t_{DS}	0		0		0		ns
$t_{h(\overline{CAS}\text{-}DA)}$	Data-in hold time with respect to $\overline{CAS}$	t_{DH}	45		55		75		ns
$t_{h(\overline{RAS}\text{-}DA)}$	Data-in hold time with respect to $\overline{RAS}$	t_{DHR}	95		120		160		ns

Read-Write and Read-Modify-Write Cycles

Symbol	Parameter	Alternative symbol	M5K 4116 P-2, S-2 Limits		M5K 4116 P-3, S-3 Limits		M5K 4116 P-4, S-4 Limits		Unit
			Min	Max	Min	Max	Min	Max	
$t_{c(RMW)}$	Read-modify-write cycle time	t_{RWC}	320		405		500		ns
$t_{c(RW)}$	Read-write cycle time	t_{RWC}	320		375		425		ns
$t_{h(WR-\overline{RAS})}$	$\overline{RAS}$ hold time with respect to write	t_{RWL}	50		70		85		ns
$t_{h(WR-\overline{CAS})}$	$\overline{CAS}$ hold time with respect to write	t_{CWL}	50		70		85		ns
$t_{w(WR)}$	Write pulse width	t_{WP}	45		55		75		ns
$t_{su(RD-\overline{CAS})}$	Read setup time with respect to $\overline{CAS}$	t_{RCS}	0		0		0		ns
$t_{d(\overline{RAS}-WR)}$	Delay time, $\overline{RAS}$ to write (Note 12)	t_{RWD}	110		145		175		ns
$t_{d(\overline{CAS}-WR)}$	Delay time, $\overline{CAS}$ to write (Note 12)	t_{CWD}	60		80		90		ns
$t_{su(DA-WR)}$	Data-in set-up time with respect to write	t_{DS}	0		0		0		ns
$t_{h(WR-DA)}$	Data-in hold time with respect to write	t_{DH}	45		55		75		ns
$t_{h(\overline{CAS}-OUT)}$	Data-out hold time with respect to $\overline{CAS}$	t_{OFF}	0	40	0	50	0	60	ns
$t_{a(\overline{CAS})}$	$\overline{CAS}$ access time (Note 10)	t_{CAC}		100		135		165	ns
$t_{a(\overline{RAS})}$	$\overline{RAS}$ access time (Note 11)	t_{RAC}		150		200		250	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.

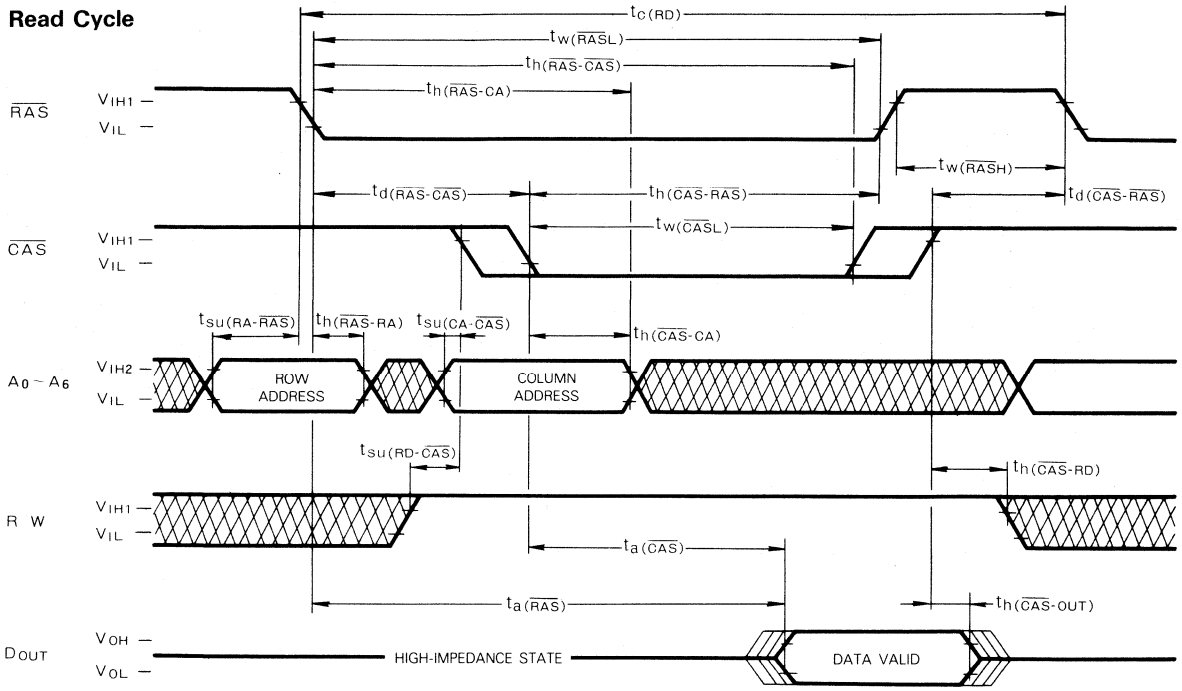
Page-Mode Cycle

Symbol	Parameter	Alternative symbol	M5K 4116 P-2, S-2 Limits		M5K 4116 P-3, S-3 Limits		M5K 4116 P-4, S-4 Limits		Unit
			Min	Max	Min	Max	Min	Max	
$t_{c(PG)}$	Page-mode cycle time	t_{PC}	170		225		275		ns
$t_{w(\overline{CASH})}$	$\overline{CAS}$ high pulse width	t_{CP}	60		80		100		ns

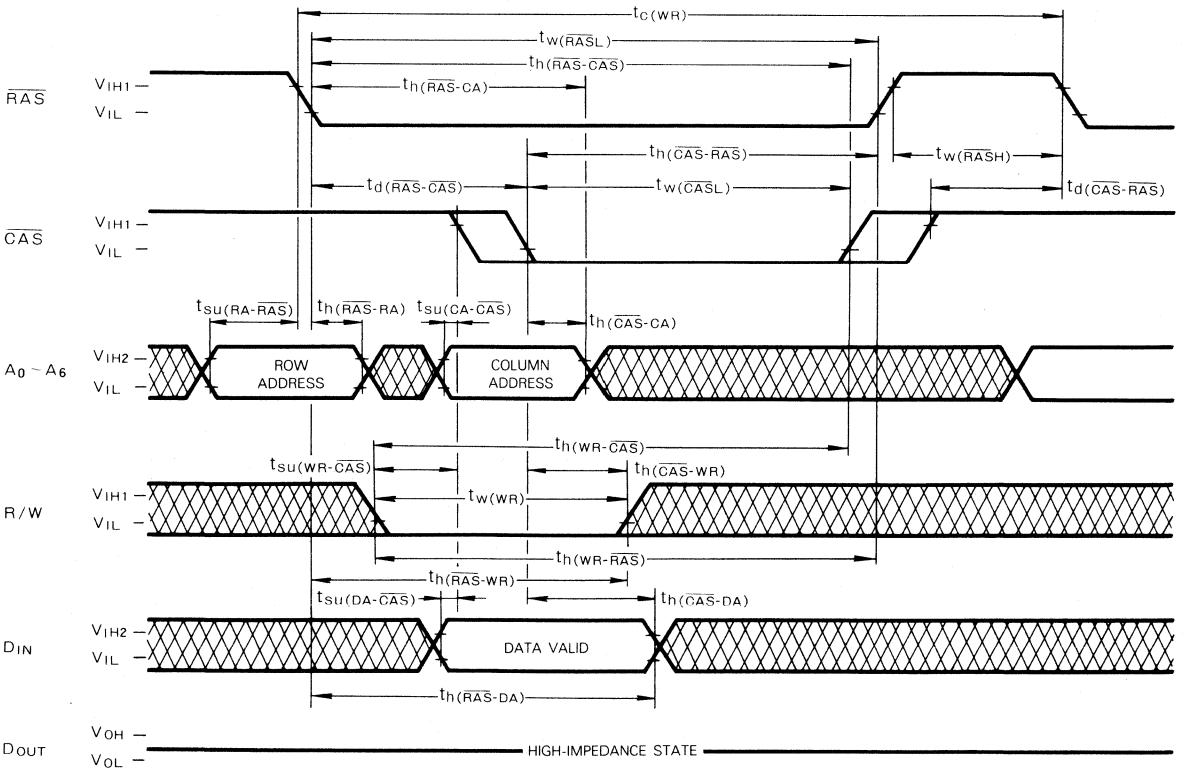
16 384-BIT (16 384-WORD BY 1-BIT) DYNAMIC RAM

TIMING DIAGRAMS

Read Cycle

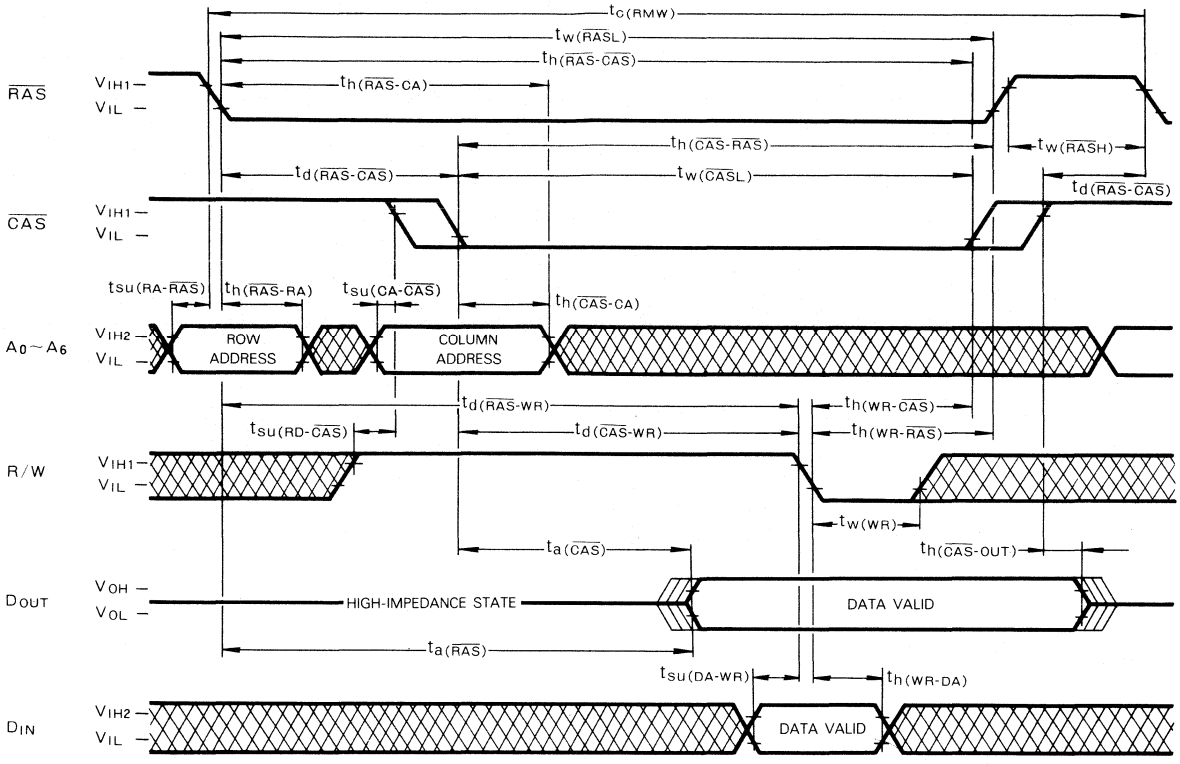


Write and Early Write Cycles

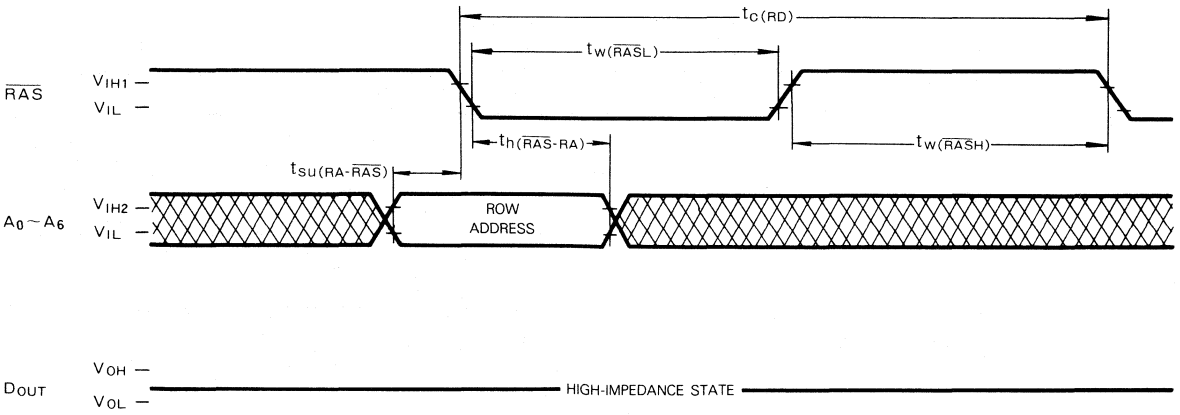


MITSUBISHI ELECTRIC

Read-Write and Read-Modify-Write Cycles



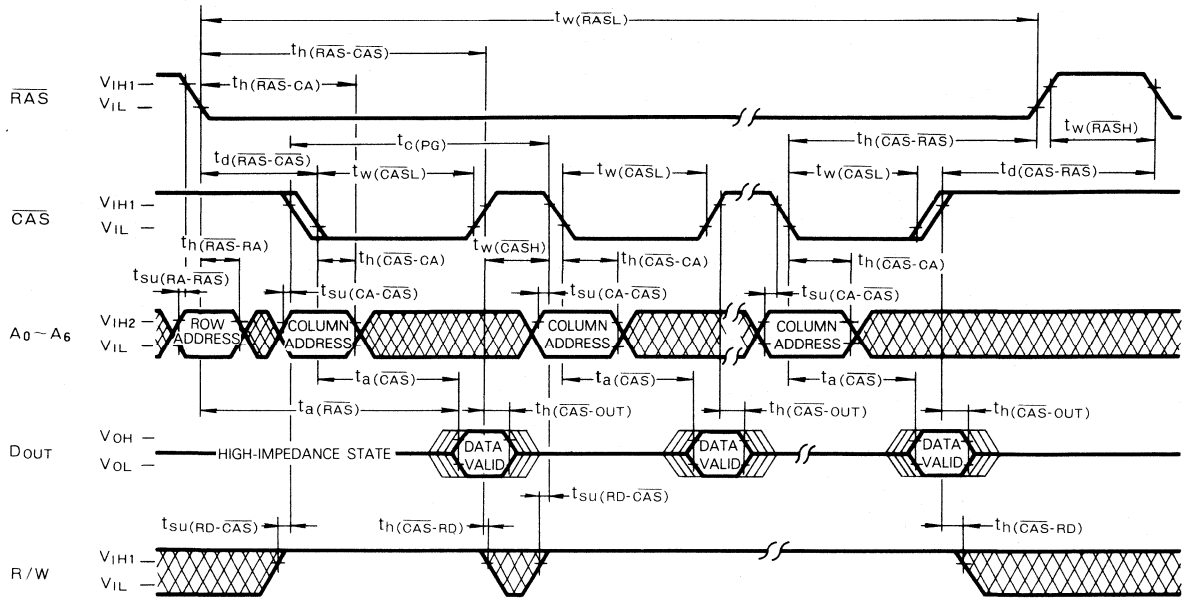
RAS-Only Refresh Cycle



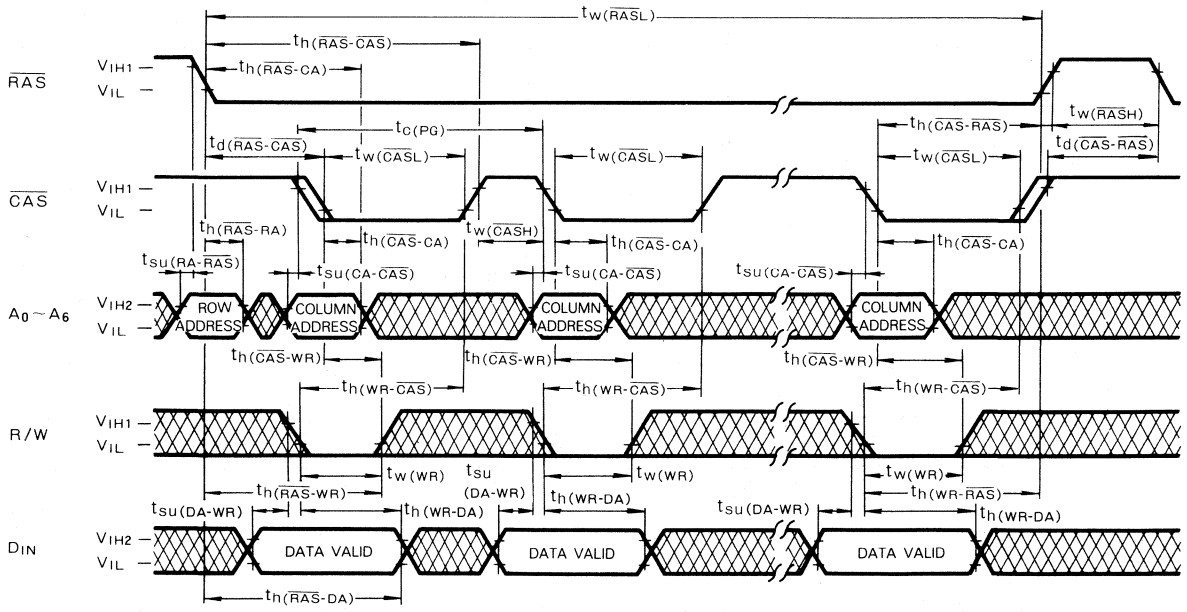
Note 13 : $\overline{CAS} = V_{IH1}$, R/W = don't care.

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

Page-Mode Read Cycle



Page-Mode Write Cycle



Note 14 ;

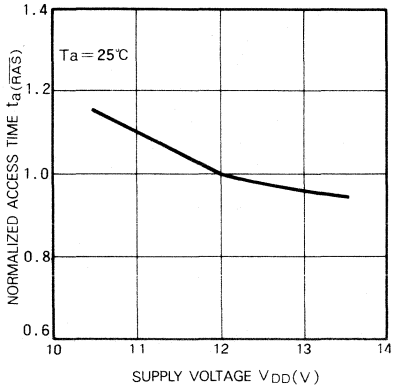
Indicates the don't care input.

The center-line indicates the high-impedance state.

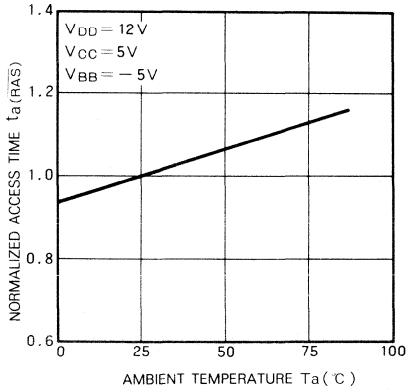
MITSUBISHI ELECTRIC

TYPICAL CHARACTERISTICS

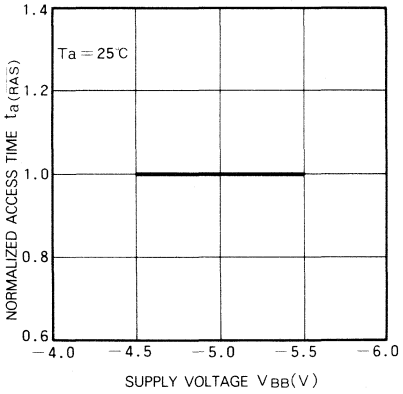
NORMALIZED ACCESS TIME VS. SUPPLY VOLTAGE V_{DD}



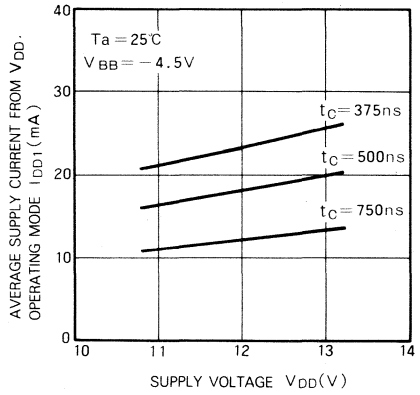
NORMALIZED ACCESS TIME VS. AMBIENT TEMPERATURE



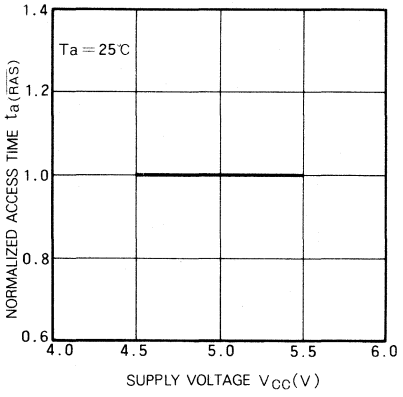
NORMALIZED ACCESS TIME VS. SUPPLY VOLTAGE V_{BB}



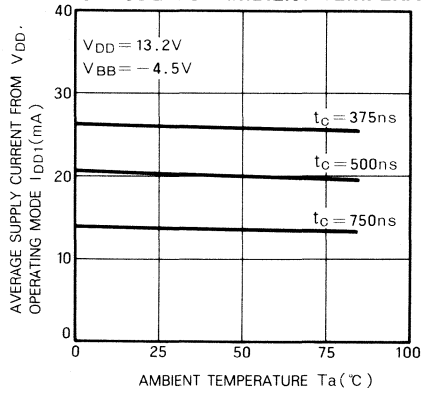
AVERAGE SUPPLY CURRENT FROM V_{DD}, OPERATING MODE VS. SUPPLY VOLTAGE



NORMALIZED ACCESS TIME VS. SUPPLY VOLTAGE V_{CC}

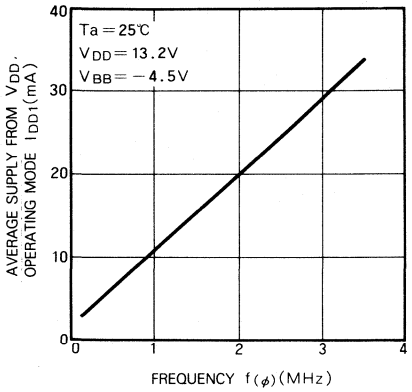


AVERAGE SUPPLY CURRENT FROM V_{DD}, OPERATING MODE VS. AMBIENT TEMPERATURE

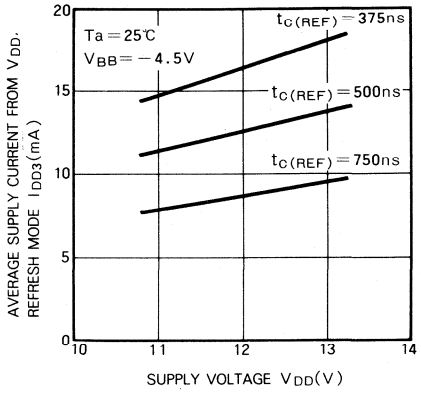


16 384-BIT (16 384-WORD BY 1-BIT) DYNAMIC RAM

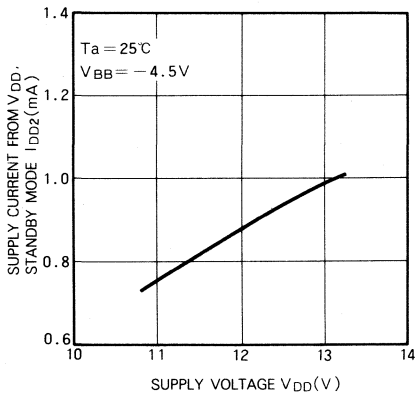
AVERAGE SUPPLY CURRENT FROM V_{DD}, OPERATING MODE VS. FREQUENCY



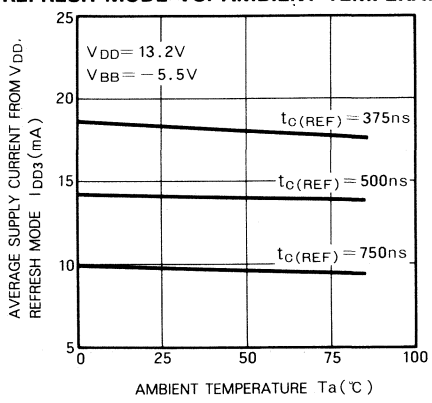
AVERAGE SUPPLY CURRENT FROM V_{DD}, REFRESH MODE VS. SUPPLY VOLTAGE



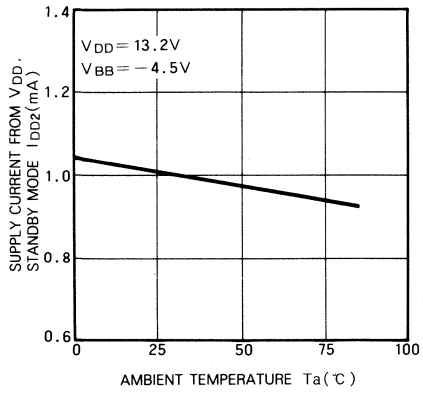
SUPPLY CURRENT FROM V_{DD}, STANDBY MODE VS. SUPPLY VOLTAGE



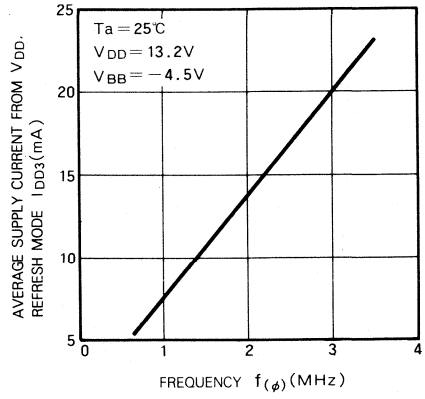
AVERAGE SUPPLY CURRENT FROM V_{DD}, REFRESH MODE VS. AMBIENT TEMPERATURE



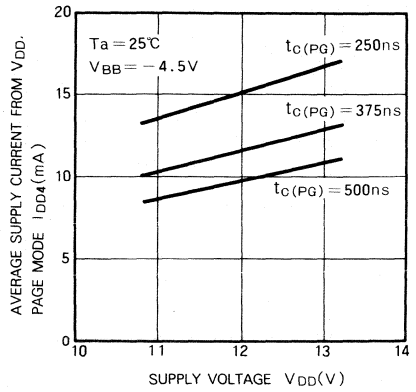
SUPPLY CURRENT FROM V_{DD}, STANDBY MODE VS. AMBIENT TEMPERATURE



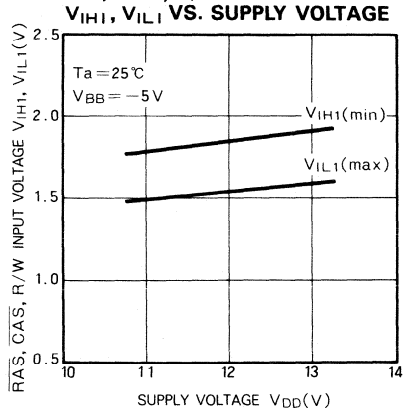
AVERAGE SUPPLY CURRENT FROM V_{DD}, REFRESH MODE VS. FREQUENCY



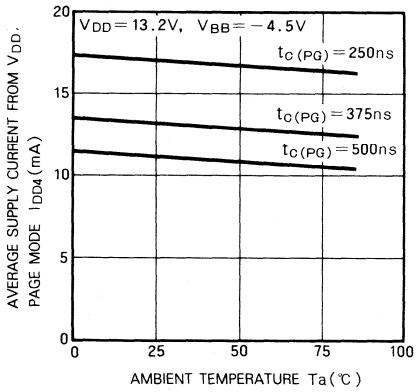
AVERAGE SUPPLY CURRENT FROM V_{DD}, PAGE MODE VS. SUPPLY VOLTAGE



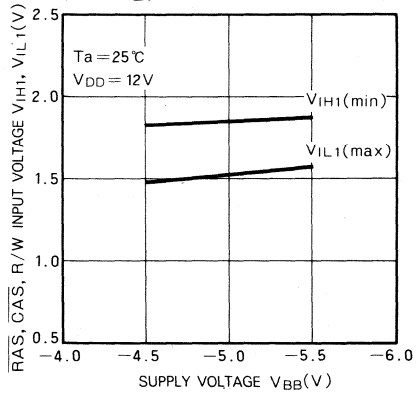
$\overline{RAS}$, $\overline{CAS}$, R/W INPUT VOLTAGE V_{IH1}, V_{IL1} VS. SUPPLY VOLTAGE



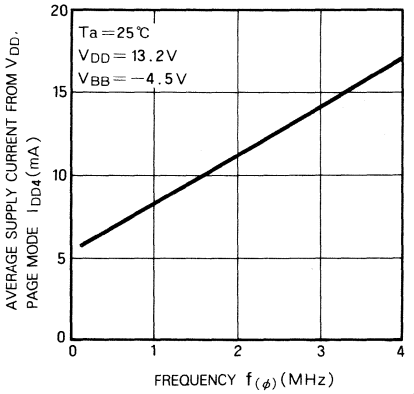
AVERAGE SUPPLY CURRENT FROM V_{DD}, PAGE MODE VS. AMBIENT TEMPERATURE



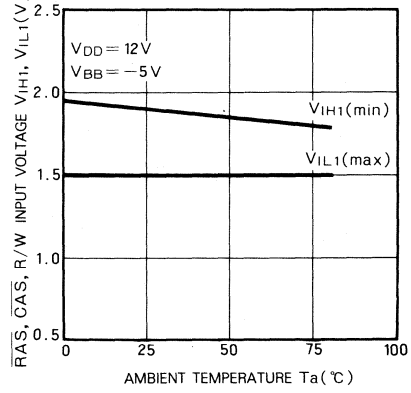
$\overline{RAS}$, $\overline{CAS}$, R/W INPUT VOLTAGE V_{IH1}, V_{IL1} VS. SUPPLY VOLTAGE



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



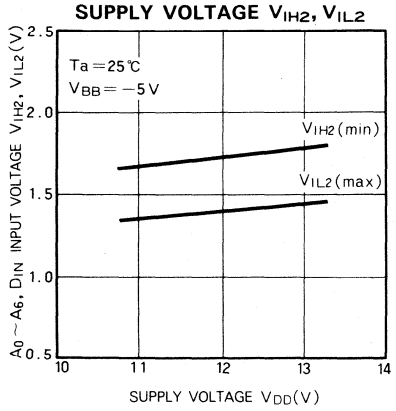
$\overline{RAS}$, $\overline{CAS}$, R/W INPUT VOLTAGE V_{IH1}, V_{IL1} VS. AMBIENT TEMPERATURE



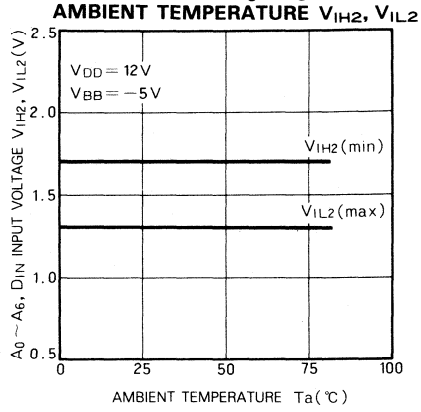
4

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

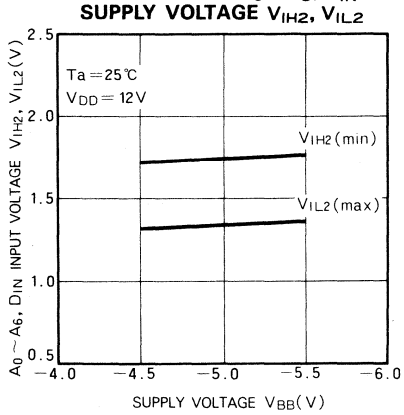
INPUT VOLTAGE $A_0 \sim A_6$, D_{IN} VS. SUPPLY VOLTAGE V_{IH2}, V_{IL2}



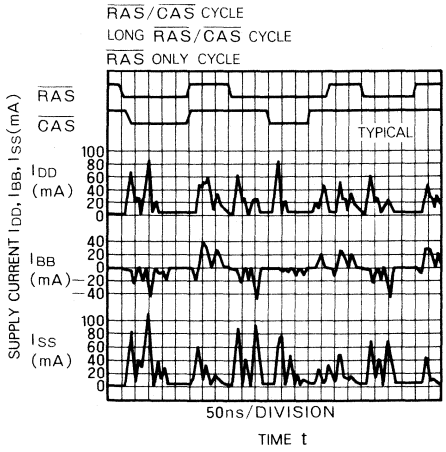
INPUT VOLTAGE $A_0 \sim A_6$, D_{IN} VS. AMBIENT TEMPERATURE V_{IH2}, V_{IL2}



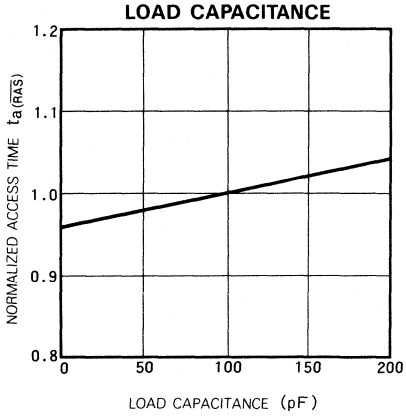
INPUT VOLTAGE $A_0 \sim A_6$, D_{IN} VS. SUPPLY VOLTAGE V_{IH2}, V_{IL2}



SUPPLY CURRENT VS. TIME



NORMALIZED ACCESS TIME VS. LOAD CAPACITANCE



M5L 2101A P, S; P-2, S-2; P-4, S-4

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

DESCRIPTION

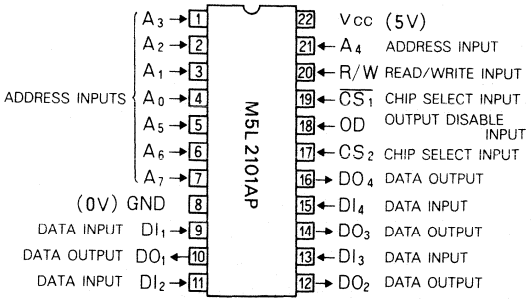
This is a family of 256-word by 4-bit static RAMs, fabricated with the N-channel silicon-gate MOS process and designed for simple interfacing. These devices operate by a single 5V supply, as does TTL, and are directly TTL-compatible.

FEATURES

Parameter	M5L2101AP,S-2	M5L 2101AP,S	M5L2101AP,S-4
Access time (max)	250ns	350ns	450ns
Cycle time (min)	250ns	350ns	450ns

- Low power dissipation: 150μW/bit (typ)
- Single 5V supply voltage
- Data holding at 1.5V supply voltage (optional)
- No clocks or refreshing required
- All inputs and outputs are directly TTL-compatible
- All outputs are three-state, with OR-tie capability
- Simple memory expansion by chip select input
- Separate data inputs and outputs
- Interchangeable with Intel's 2101A series in pin configuration and electrical characteristics

PIN CONFIGURATION (TOP VIEW)



Outline 22P1 (M5L2101AP)
22S1 (M5L2101AS)

APPLICATION

- Small-capacity memory units

FUNCTION

These devices provide 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 IN signal at that time is written.

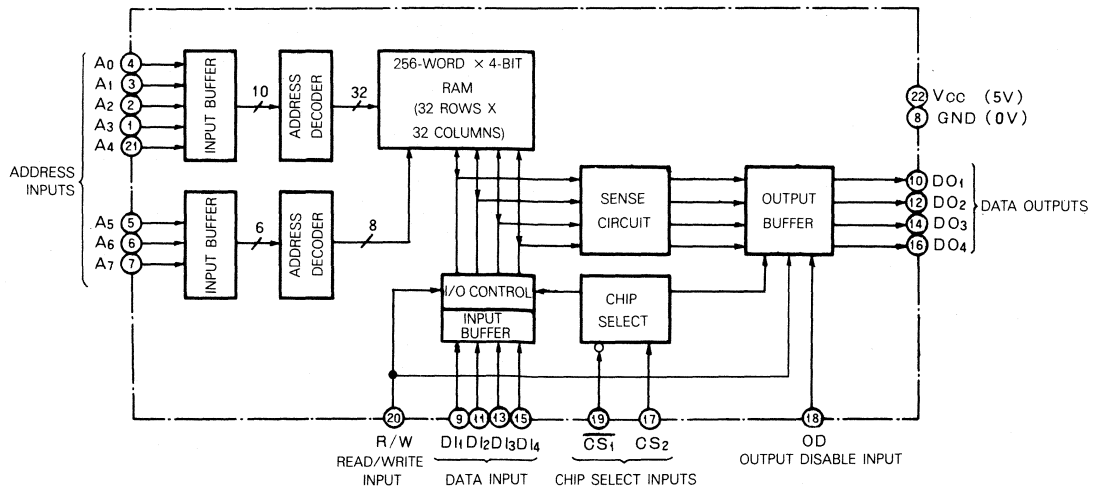
During a read cycle, when a location is designated by address signals $A_0 \sim A_7$ and R/W goes high, data of the designated address is available at the DO terminal.

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 other output terminals.

When 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 1.5V, enabling battery back-up operation during power failure and power-down operation in the standby mode.

BLOCK DIAGRAM



M5L 2101A P, S; P-2, S-2; P-4, S-4

1024-BIT (256-WORD BY 4-BIT) 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			−0.3～7	V
V_O	Output voltage			−0.3～7	V
Pd	Maximum power dissipation	M5L 2101AP	Ta＝25°C	700	mW
		M5L 2101AS		1000	mW
Topr	Operating free-air ambient temperature range			0～70	°C
Tstg	Storage temperature range	M5L 2101AP		−40～125	°C
		M5L 2101AS		−65～150	°C

RECOMMENDED OPERATING CONDITIONS (Ta＝0～10°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{IL}	Low-level input voltage	0		0.8	V
V_{IH}	High-level input voltage	2.2		V_{CC}	V

ELECTRICAL CHARACTERISTICS (Ta＝0～70°C, V_{CC}＝5V±5%, 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		0.8	V
V_{OH}	High-level output voltage	I_{OH}＝−200μA	2.4			V
V_{OL}	Low-level output voltage	I_{OL}＝3.5mA			0.45	V
I_I	Input current	V_I＝0～5.25V			10	μA
I_{OZH}	Off-state high-level output current	$V_{I(\overline{CS_1})}$＝2.2V, V_O＝2.4V～V_{CC}			10	μA
I_{OZL}	Off-state low-level output current	$V_{I(\overline{CS_1})}$＝2.2V, V_O＝0.4V			−10	μA
I_{CC}	Supply current from V_{CC}	V_I＝5.25V (all inputs), output open, Ta＝25°C		30	60	mA
C_i	Input capacitance, all inputs	V_I＝GND, f＝1MHz, 25mVrms		3	5	pF
C_o	Output capacitance	V_O＝GND, f＝1MHz, 25mVrms		8	12	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±5%, unless otherwise noted) (Note 2)

Symbol	Parameter	M5L 2101AP, S-2			M5L 2101AP, S			M5L 2101AP, S-4			Unit
		Limits			Limits			Limits			
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
$t_{C(RD)}$	Read cycle time	250			350			450			ns
$t_{a(AD)}$	Address access time			250			350			450	ns
$t_{a(CS)}$	Chip select access time			180			180			180	ns
$t_{a(OD)}$	Output disable access time			130			150			150	ns
t_{PXZ}	Output disable time (Note 3)			100			100			100	ns
$t_{dv(AD)}$	Data valid time with respect to address	40			40			40			ns

Note 2 : Test conditions : input pulse V_{IH}＝2.2V, V_{IL}＝0.8V t_r＝t_f＝20ns ; reference level ＝1.5V; load＝2TTL, C_L＝100pF

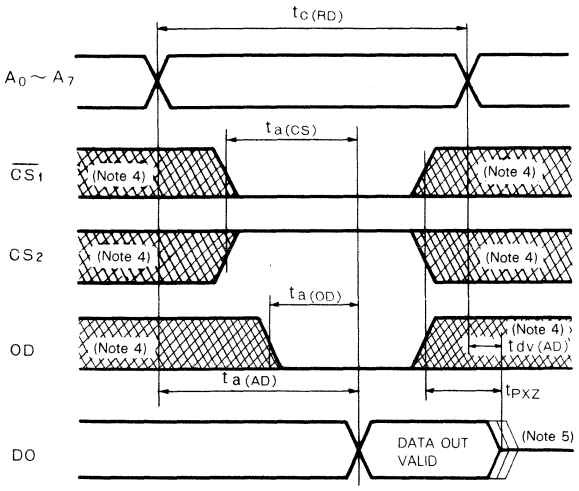
Note 3 : t_{PXZ} is with respect to $\overline{CS_1}$, CS_2, or OD, whichever occurs first.

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

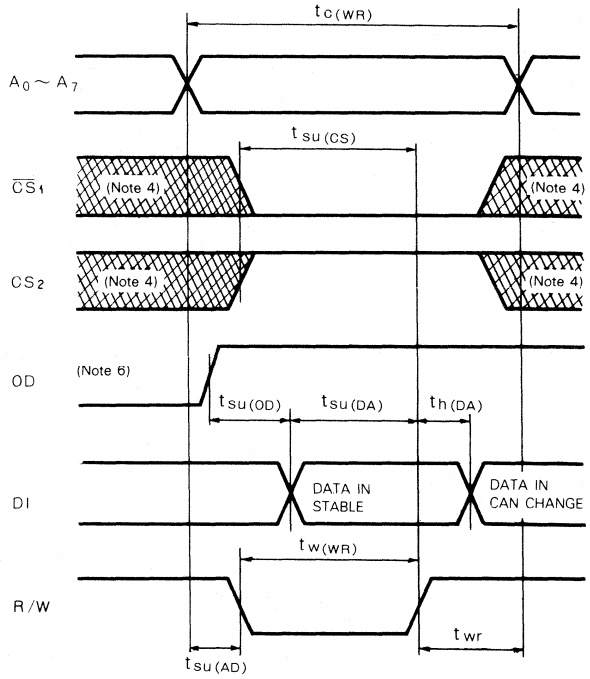
Symbol	Parameter	M5L 2101AP, S-2			M5L 2101AP, S			M5L 2101AP, S-4			Unit
		Limits			Limits			Limits			
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
$t_{C(WR)}$	Write cycle time	170			220			270			ns
$t_{W(WR)}$	Write pulse width	150			200			250			ns
$t_{su(AD)}$	Address setup time with respect to write	20			20			20			ns
t_{wr}	Write recovery time	0			0			0			ns
$t_{su(OD)}$	Output disable setup time with respect to data in	20			20			20			ns
$t_{su(DA)}$	Data setup time	100			150			170			ns
$t_{h(DA)}$	Data hold time	0			0			0			ns
$t_{su(CS)}$	Chip select setup time	150			200			250			ns

TIMING DIAGRAMS

Read Cycle



Write Cycle



Note 4 : Hatching indicates the state is unknown.

5 : 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}.

6 : OD may be kept low for the full cycle except during common input/output operation.

POWER-DOWN OPERATION (OPTIONAL) **These characteristics are guaranteed only under custom specifications.**

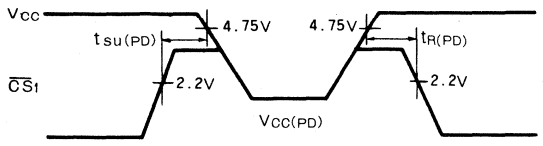
Electrical Characteristics (T_a=0~70°C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$V_{CC(PD)}$	Power-down supply voltage		1.5			V
$V_{I(\overline{CS1})}$	Power-down chip select input voltage	2.2V≦$V_{CC(PD)}$≦V_{CC}	2.2			V
		1.5V≦$V_{CC(PD)}$≦2.2V	$V_{CC(PD)}$			V
$I_{CC(PD1)}$	Power-down supply current from V_{CC}	V_{CC}=1.5V, all inputs =1.5V		15	30	mA
$I_{CC(PD2)}$	Power-down supply current from V_{CC}	V_{CC}=2.0V, all inputs =2.0V		20	40	mA

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

Symbol	Parameter	Test conditions	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

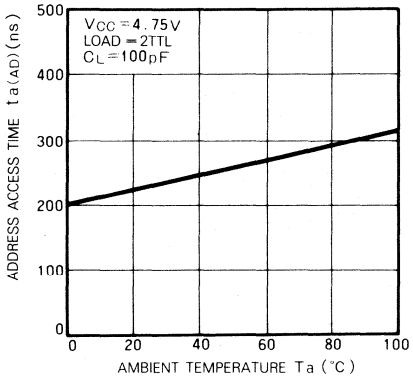
Timing Diagram



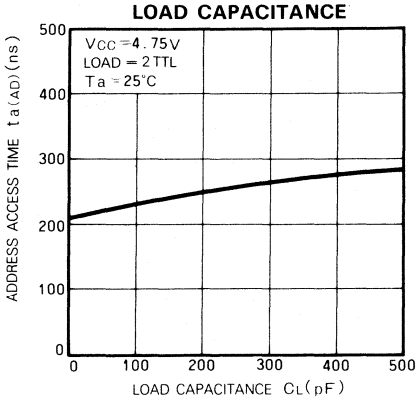
1024-BIT (256-WORD BY 4-BIT) STATIC RAM

TYPICAL CHARACTERISTICS

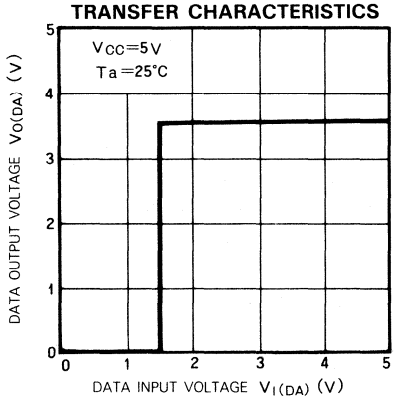
ADDRESS ACCESS TIME VS. AMBIENT TEMPERATURE



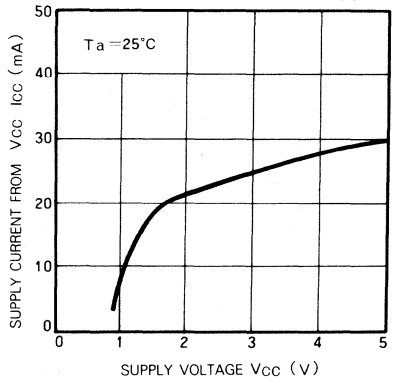
ADDRESS ACCESS TIME VS. LOAD CAPACITANCE



DATA INPUT/OUTPUT TRANSFER CHARACTERISTICS



SUPPLY CURRENT FROM Vcc VS. SUPPLY VOLTAGE Vcc



M5L 2102A P-4, S-4

1024-BIT (1024-WORD BY 1-BIT) STATIC RAM

DESCRIPTION

This is a family of 1024-word by 1-bit N-channel silicon-gate MOS static RAMs, designed for applications where ease of use is the important design object. The devices operate by a single 5V power supply, as does TTL, and all inputs and output are directly compatible with TTL.

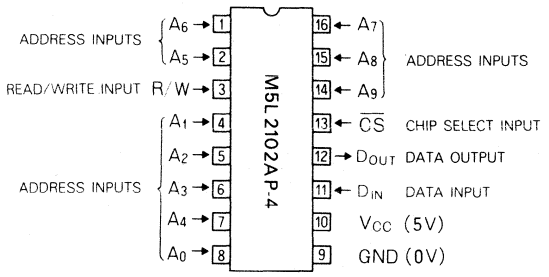
FEATURES

- Fast time: 450ns (max)
- Low power dissipation: 100μW/bit (typ)
- Single 5V power supply
- Data holding at 1.5V supply voltage is possible
- Requires no external clock or refreshing
- All inputs and output are directly compatible with TTL
- Three-state output and OR-tie capability
- Easy memory expansion by chip select input
- Interchangeable with Intel's 2102A-4 in pin configuration and electrical characteristics

APPLICATION

- Small-capacity memory systems

PIN CONFIGURATION (TOP VIEW)



Outline 16P1 (M5L 2102A P-4)
16S1 (M5L 2102A S-4)

FUNCTION

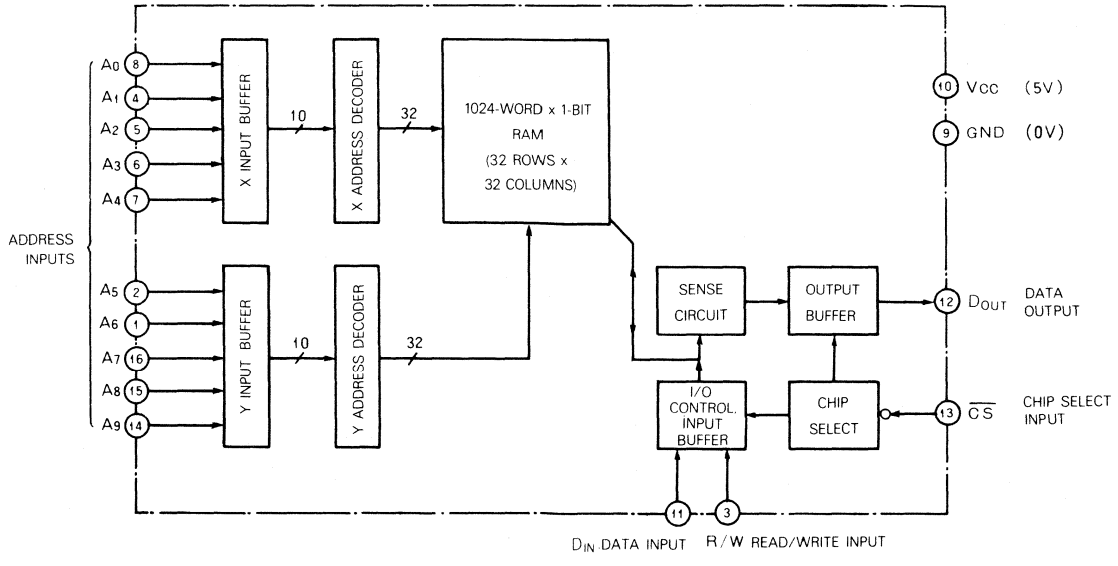
Static design makes these devices convenient to use as they require no external clocks or refreshing, and all inputs and output are directly compatible with TTL.

During writing operation, when a location is designated by address signals $A_0 \sim A_9$ and R/W goes low, D_{IN} at that time is written; during reading operation, when a location is designated by address signals $A_0 \sim A_9$ and R/W goes high, data of the designated address is available at the D_{OUT} terminal.

When $\overline{CS}$ is high, the chip is in the non-selectable state, disabling both reading and writing operations of the device. In this case the output is in the floating (high-impedance) state enabling OR-tie to other outputs.

The memory data is held when supply voltage drops to 1.5V, enabling battery back-up operation during power stoppages and low-power operation during standby.

BLOCK DIAGRAM



1024-BIT (1024-WORD BY 1-BIT) 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− 7	V
V_O	Output voltage			−0.3− 7	V
P_d	Power dissipation	M5L 2102A P-4	Ta = 25°C	700	mW
		M5L 2102A S-4		1000	mW
T_{opr}	Operating free-air temperature range			0 − 70	°C
T_{stg}	Storage temperature range	M5L 2102A P-4		−40− 125	°C
		M5L 2102A S-4		−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	5.0	5.25	V
V_{IL}	Low-level input voltage	0		0.65	V
V_{IH}	High-level input voltage	2.2		V_{CC}	V

ELECTRICAL CHARACTERISTICS (Ta = 0 − 70°C, V_{CC} = 5V ± 5%, 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		0.65	V
V_{OH}	High-level output voltage	I_{OH} = −200μA	2.4			V
V_{OL}	Low-level output voltage	I_{OL} = 2.1mA			0.4	V
I_I	Input current	V_I = 0~5.25V			10	μA
I_{OZH}	Off-state high-level output current	$V_{I(\overline{CS})}$ = 2.2V, V_O = 2.4V ~ V_{CC}			10	μA
I_{OZL}	Off-state low-level output current	$V_{I(\overline{CS})}$ = 2.2V, V_O = 0.4V			−10	μA
I_{CC}	Supply current from V_{CC}	V_I = 5.25 (all inputs), output open, Ta = 25°C		20	40	mA
C_i	Input capacitance, all inputs	V_I = GND, V_i = 25mV_{rms}, f = 1MHz		3	5	pF
C_o	Output capacitance	V_O = GND, V_o = 25mV_{rms}, f = 1MHz		7	10	pF

SWITCHING CHARACTERISTICS (Ta = 0 − 70°C, V_{CC} = 5V ± 5%, unless otherwise noted)

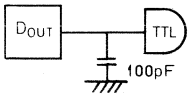
Read Cycle

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{c(RD)}$	Read cycle time	Input pulse V_{IH} = 2.2V	450			ns
$t_{a(AD)}$	Address access time	V_{IL} = 0.65V			450	ns
$t_{a(\overline{CS})}$	Chip select access time	$t_r = t_f$ = 20ns			230	ns
$t_{dv(AD)}$	Data valid time with respect to address	Reference level 1.5V	40			ns
$t_{dv(\overline{CS})}$	Data valid time with respect to chip select	Load = 1TTL, C_L = 100pF	0			ns

TIMING REQUIREMENTS (Ta=0~70°C, Vcc=5V±5%, unless otherwise noted)

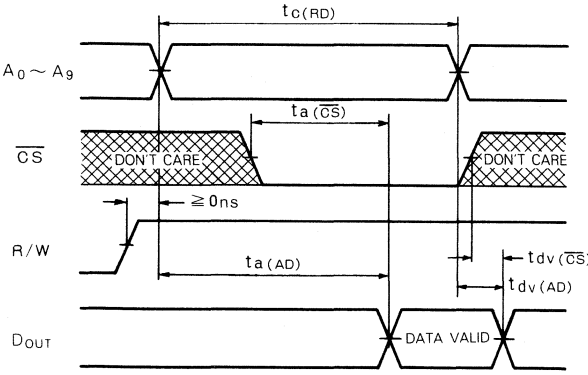
Write Cycle

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{c(WR)}$	Write cycle time	Input pulse $V_{IH}=2.2V$	450			ns
$t_{su(AD)}$	Address setup time	$V_{IL}=0.65V$	20			ns
$t_{w(WR)}$	Write pulse width	$t_r=t_f=20ns$	300			ns
$t_{h(DA)}$	Data hold time	Reference level $=1.5V$	0			ns
$t_{su(DA)}$	Data setup time	Load=1TTL, $C_L=100pF$	300			ns
t_{WR}	Write recovery time		0			ns
$t_{su(\overline{CS})}$	Chip select setup time		300			ns

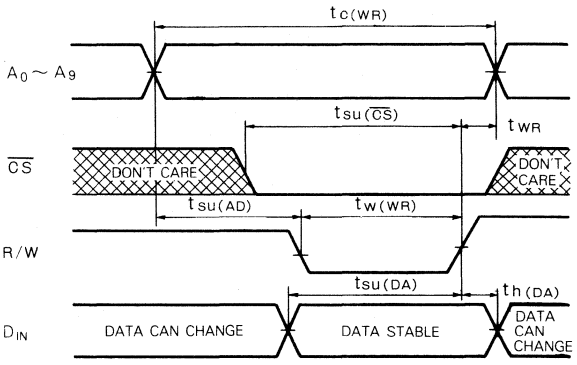


TIMING DIAGRAMS

Read Cycle



Write Cycle



POWER-DOWN OPERATION

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

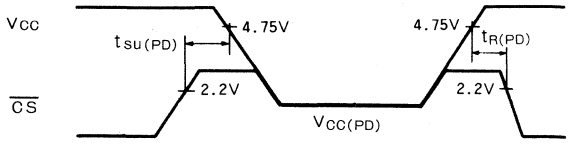
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$V_{CC(PD)}$	Power-down supply voltage		1.5			V
$V_{I(\overline{CS})}$	Power-down chip select voltage	$2.2V \leqq V_{CC(PD)} \leqq V_{CC}$	2.2			V
		$1.5V \leqq V_{CC(PD)} \leqq 2.2V$	$V_{CC(PD)}$			V
$I_{CC(PD1)}$	Power-down supply current	$V_{CC}=1.5V$, all inputs $=1.5V$		13	25	mA
$I_{CC(PD2)}$	Power-down supply current	$V_{CC}=2.0V$, all inputs $=2.0V$		15	30	mA

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

Timing Requirements (Ta=0~70°C, Vcc=5V±5%, unless otherwise noted)

Symbol	Parameter	Test conditions	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

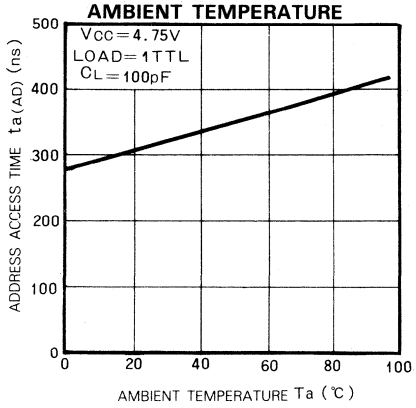
Timing Diagram



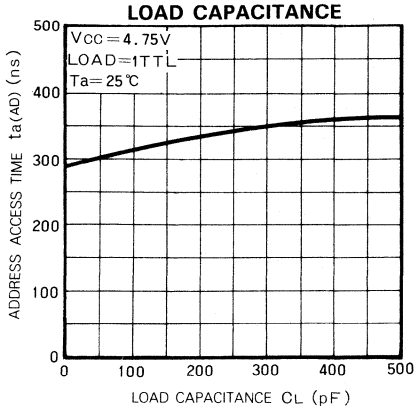
1024-BIT (1024-WORD BY 1-BIT) STATIC RAM

TYPICAL CHARACTERISTICS

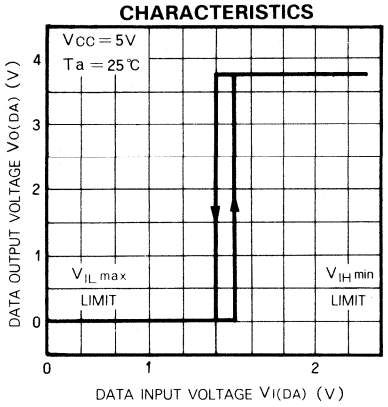
ADDRESS ACCESS TIME VS. AMBIENT TEMPERATURE



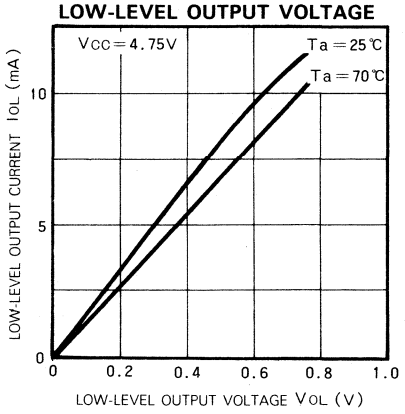
ADDRESS ACCESS TIME VS. LOAD CAPACITANCE



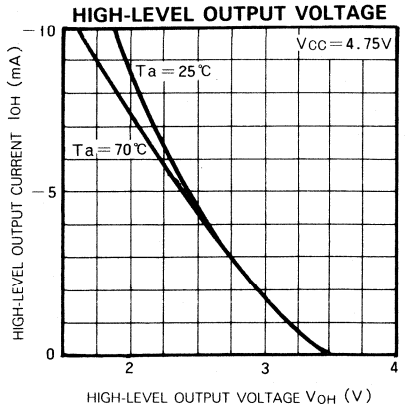
DATA INPUT-OUTPUT TRANSFER CHARACTERISTICS



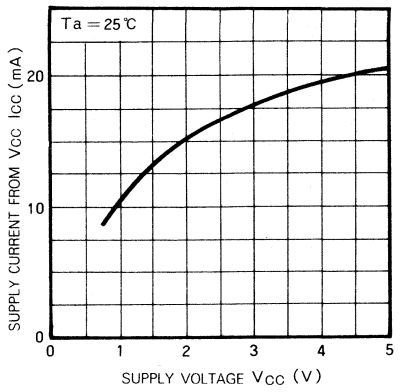
LOW-LEVEL OUTPUT CURRENT VS. LOW-LEVEL OUTPUT VOLTAGE



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



SUPPLY CURRENT FROM Vcc VS. SUPPLY VOLTAGE Vcc



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

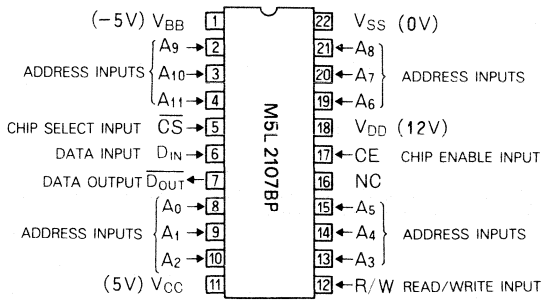
DESCRIPTION

These devices are 4096-word by 1-bit dynamic RAMs, fabricated with the N-channel silicon-gate MOS process. These RAMs are designed for large-capacity memory systems where high speed, low power dissipation and low cost are important design objects.

FEATURES

- Fast access time: 200ns (max)
- Fast cycle time: 400ns (min)
- Low active power: 300mW (typ)
- Low standby power: 0.03μW/bit (typ)
- Voltage range for all power supplies (V_{DD}, V_{CC}, V_{BB}): ±10%
- Refresh interval: 2ms (Ta = 0~70°C)
- Refresh addresses: A_0, A_1, A_2, A_3, A_4, A_5
- All inputs except CE terminal are directly TTL compatible
- Memory expansion is enabled by chip select input
- Output can be in the floating (high-impedance) state when $\overline{CS}$ is high or CE is low.
- Interchangeable with Intel's 2107B and TI's TMS4060

PIN CONFIGURATION (TOP VIEW)



NC = NO CONNECTION

Outline 22P1 (M5L2107BP)
22S1 (M5L2107BS)

APPLICATION

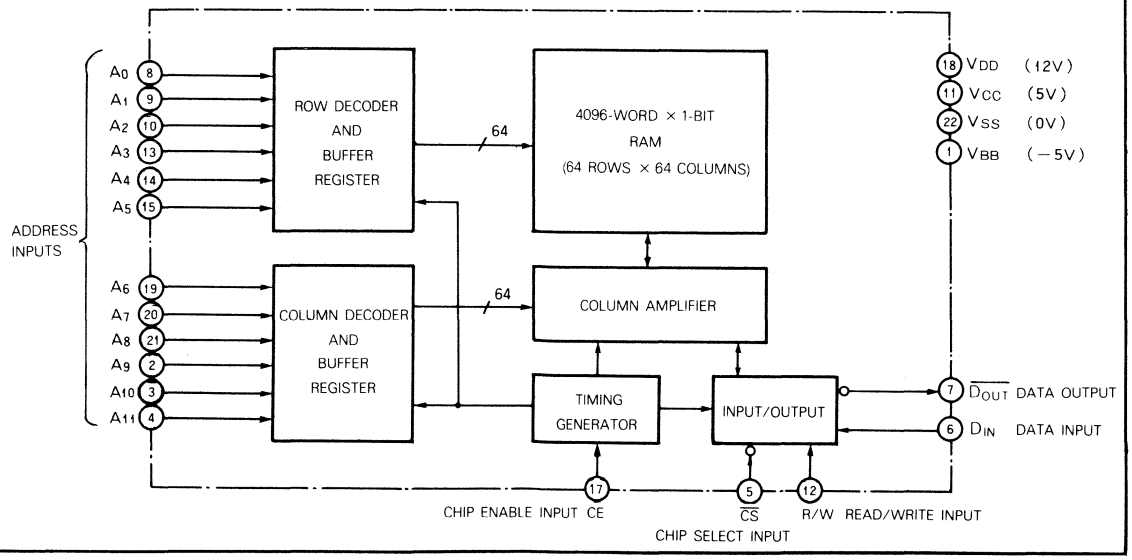
- Main memory unit for computers

FUNCTION

A location is designated by address signals A_0~A_{11}, and reading from and writing to that location is controlled by R/W. When $\overline{CS}$ is high, the chip is in the non-selectable state, disabling both read and write operations.

The devices are dynamic RAMs, and must be refreshed every 2ms to hold data stored in the memory cells. Refreshing is performed by reading sequentially the 64 locations designated by the 6 address signals A_0~A_5.

BLOCK DIAGRAM



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

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter		Conditions	Limits	Unit
V_{DD}	Supply voltage		With respect to V_{BB} (substrate)	−0.3～20	V
V_{CC}	Supply voltage			−0.3～20	V
V_{SS}	Supply voltage			−0.3～20	V
V_I	Input voltage			−0.3～20	V
V_O	Output voltage			−0.3～20	V
P_d	Power dissipation	M5L 2107BP	Ta=25℃	700	mW
		M5L 2107BS	Ta=25℃	1000	mW
T_{opr}	Operating free-air temperature range			0～70	℃
T_{stg}	Storage temperature range	M5L 2107BP		−40～125	℃
		M5L 2107BS		−65～150	℃

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

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	10.8	12	13.2	V
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage		0		V
V_{BB}	Supply voltage	−4.5	−5	−5.5	V
$V_{IH(CE)}$	High-level chip enable input voltage	$V_{DD}-1$		$V_{DD}+1$	V
V_{IH}	High-level input voltage, all inputs except chip enable	2.4		$V_{CC}+1$	V
$V_{IL(CE)}$	Low-level chip enable input voltage	−1		1	V
V_{IL}	Low-level input voltage, all inputs except chip enable	−1		0.6	V

ELECTRICAL CHARACTERISTICS

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

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$V_{IH(CE)}$	High-level chip enable input voltage		$V_{DD}-1$		$V_{DD}+1$	V
V_{IH}	High-level input voltage, all inputs except chip enable		2.4		$V_{CC}+1$	V
$V_{IL(CE)}$	Low-level chip enable input voltage		−1		1	V
V_{IL}	Low-level input voltage, all inputs except chip enable		−1		0.6	V
$I_{I(CE)}$	Input current, chip enable input	$V_I=V_{DD}+1V$		0.01	2	μA
I_I	Input current, all inputs except chip enable	V_I=6.5V		0.01	10	
V_{OH}	High-level output voltage	I_{OH}=−2mA	2.4		V_{CC}	V
V_{OL}	Low-level output voltage	I_{OL}=2mA	0		0.45	V
I_{OZ}	Off-state output current	$V_{OZ}=0～V_{CC}$			±10	μA
I_{DD1}	Supply current from V_{DD}	$V_{IL(CE)}$=−1V～0.6V		10	200	μA
I_{DD2}	Supply current from V_{DD}	$V_{IH(CE)}=V_{IH}$, $V_{IL(\overline{CS})}=V_{IL}$		10	25	mA
I_{CC}	Supply current from V_{CC}	$V_{IL(CE)}=V_{IL}$ or $V_{IH(\overline{CS})}=V_{IH}$		0.01	10	μA
I_{BB}	Supply current from V_{BB}			0.01	100	μA
$I_{DD(AV)}$	Average supply current from V_{DD}	$t_{w(CE)}$=230ns, t_c=400ns		25	40	mA
$C_{i(CE)}$	Input capacitance, chip enable input	$V_{IL}=V_{SS}$, V_{BB}=−5V, f=1MHz		17	25	pF
C_i	Input capacitance, all inputs except chip enable	$V_{IL}=V_{SS}$, V_{BB}=−5V, f=1MHz		5	7	pF
C_o	Output capacitance	$V_{OL}=V_{SS}$, V_{BB}=−5V, f=1MHz		5	7	pF

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

TIMING REQUIREMENTS ($Ta=0\sim70°C$, $V_{DD}=12V\pm10\%$, $V_{CC}=5V\pm10\%$, $V_{SS}=0V$, $V_{BB}=-5V\pm10\%$, unless otherwise noted)

Read, Write or Read-Modify-Write Cycle

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{c(REF)}$	Refresh cycle time				2	ms
$t_{w(CEL)}$	Chip enable low pulse width		130			ns
$t_{r(CE)}$	Chip enable pulse rise time				40	ns
$t_{f(CE)}$	Chip enable pulse fall time				40	ns
$t_{su(AD)}$	Address setup time		0			ns
$t_{su(\overline{CS})}$	Chip select setup time		0			ns
$t_{h(AD)}$	Address hold time		100			ns
$t_{h(\overline{CS})}$	Chip select hold time		100			ns

4

Read Cycle

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{c(RD)}$	Read cycle time	$t_r=t_f=20ns$	400			ns
$t_{w(CEH)}$	Chip enable high pulse width		230		4000	ns
$t_{su(RD)}$	Read setup time		−10			ns
$t_{h(RD)}$	Read hold time		0			ns

Write or Read-Modify-Write Cycle

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{c(WR)}$	Write cycle time	$t_r=t_f=20ns$	400			ns
$t_{c(RMW)}$	Read-modify-write cycle time		520			ns
$t_{w(CEH)}$	Chip enable high pulse width, write cycle		230		4000	ns
$t_{w(CEH)}$	Chip enable high pulse width, read-modify-write cycle		350		4000	ns
$t_{su(RD)}$	Read setup time		−10			ns
$t_{h(RD)}$	Read hold time		180			ns
$t_{su(WR)}$	Write setup time		150			ns
$t_{w(WR)}$	Write pulse width		50			ns
$t_{d(WR)}$	Write delay time		150			ns
$t_{su(DA)}$	Data setup time		0			ns
$t_{h(DA)}$	Data hold time		0			ns

SWITCHING CHARACTERISTICS ($Ta=0\sim70°C$, $V_{DD}=12V\pm10\%$, $V_{CC}=5V\pm10\%$, $V_{SS}=0V$, $V_{BB}=-5V\pm10\%$, unless otherwise noted)

Read Cycle

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{a(CE)}$	Chip enable access time	$C_L=50pF$, Load=1 TTL, $V_{REF}=2.0V$			180	ns
$t_{a(AD)}$	Address access time	$t_{su(AD)}=0ns$, $t_r=t_f=20ns$			200	ns
$t_{dv(CE)}$	Data valid time with respect to chip enable		0			ns

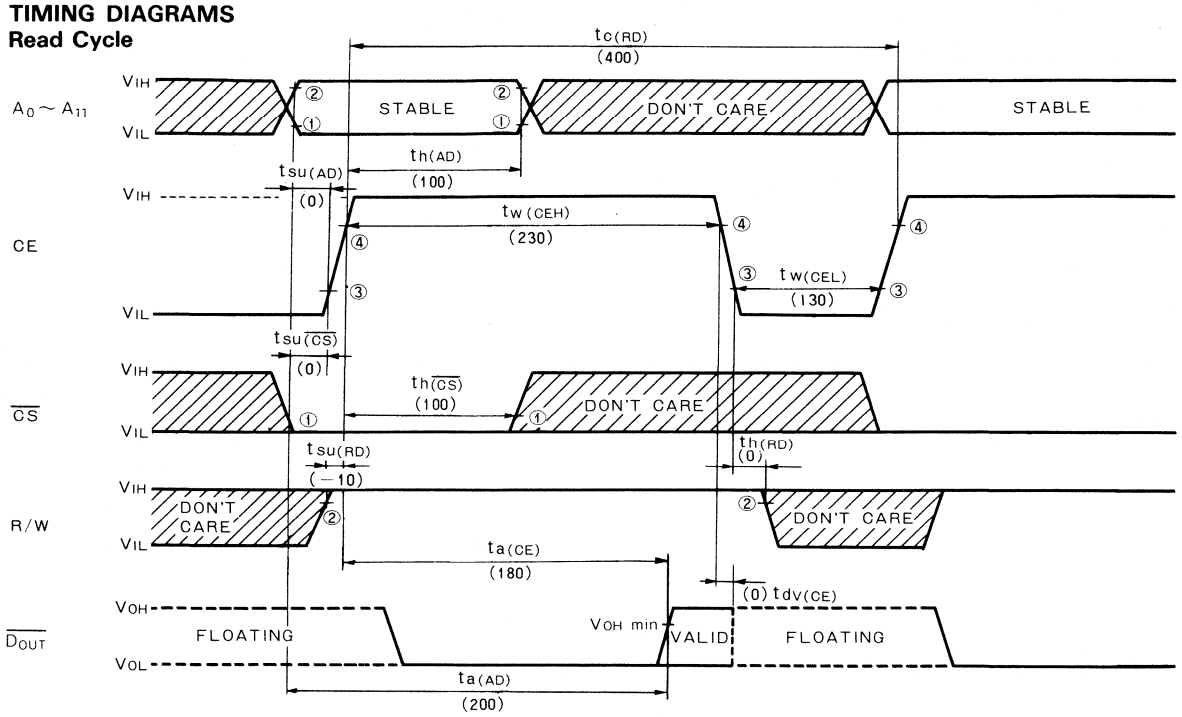
Read-Modify-Write Cycle

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{a(CE)}$	Chip enable access time	$C_L=50pF$, Load=1 TTL, $V_{REF}=2.0V$			180	ns
$t_{a(AD)}$	Address access time	$t_{su(AD)}=0ns$, $t_r=t_f=20ns$			200	ns
$t_{dv(CE)}$	Data valid time with respect to chip enable		0			ns

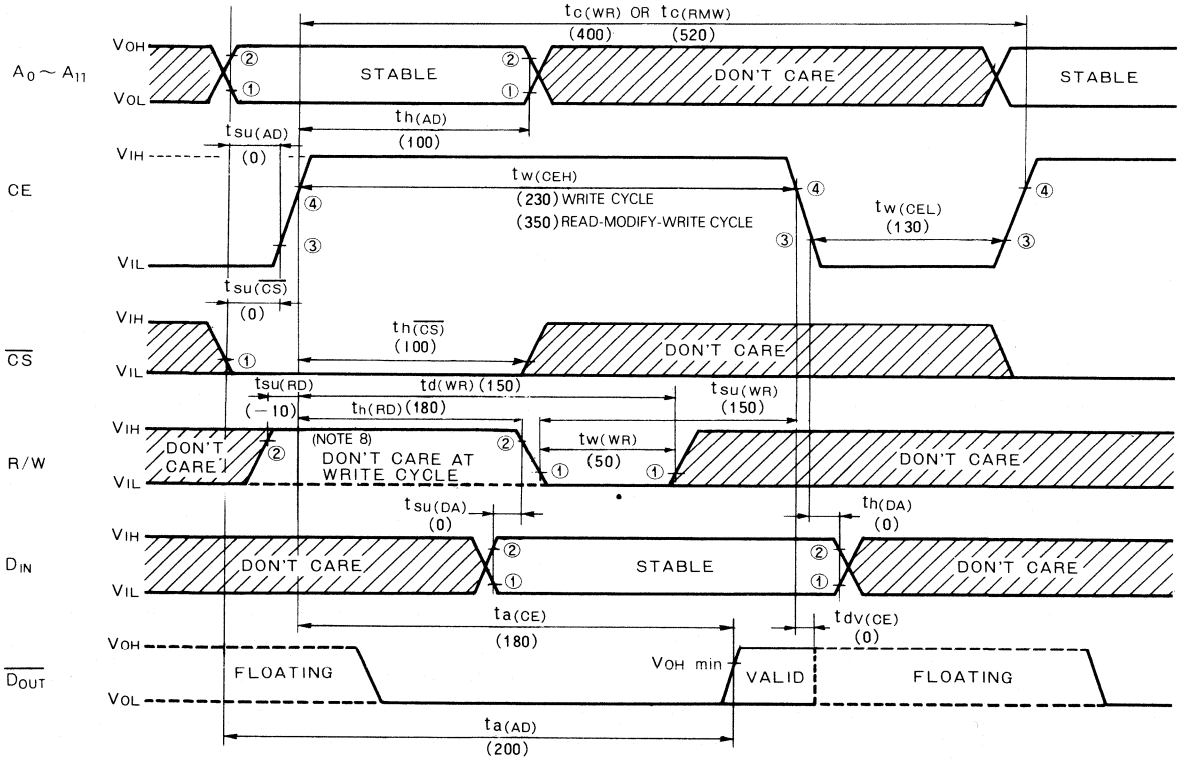
4096-BIT (4096-WORD BY 1-BIT) DYNAMIC RAM

TIMING DIAGRAMS

Read Cycle



Write or Read-Modify-Write Cycle



Note 2 : Hatching indicates the state is unknown or changing.

3 : Vss+0.6V is the reference level for point ①, and Vss+2.4V for point ②.

4 : Vss+2.0V is the reference level for point ③, and VDD −2.0V for point ④.

5 : The transition time (t_T) of the CE Pulse is defined as the transition time from ③ to ④ and from ④ to ③.

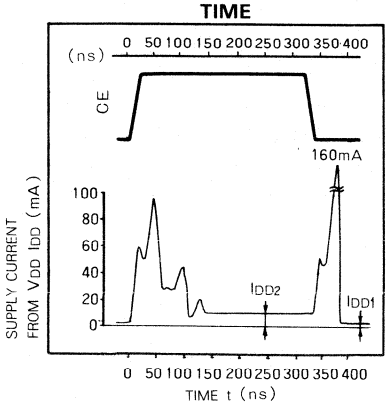
6 : The level of the dotted line should be kept high during read-modify-write cycle.

7 : Numbers in parentheses ( ) indicate the minimum timing value in ns.

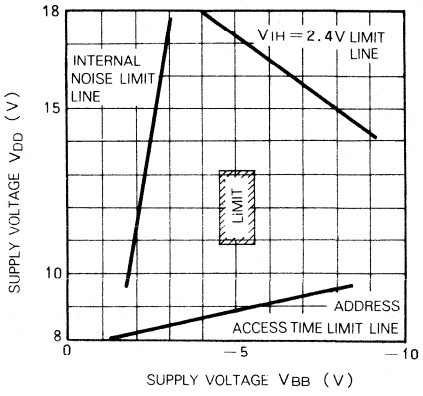
MITSUBISHI ELECTRIC

TYPICAL CHARACTERISTICS

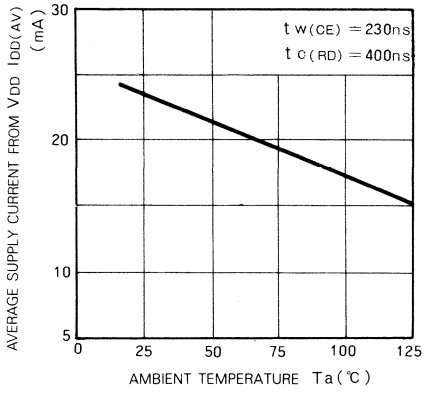
SUPPLY CURRENT FROM VDD VS. TIME



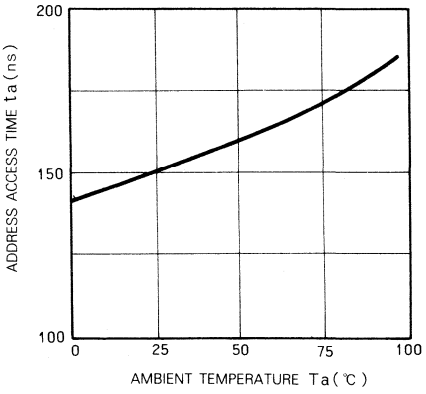
VDD VS. VBB OPERATING REGION



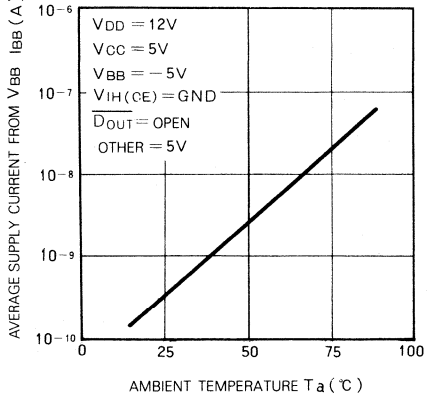
AVERAGE SUPPLY CURRENT FROM VDD VS. AMBIENT TEMPERATURE



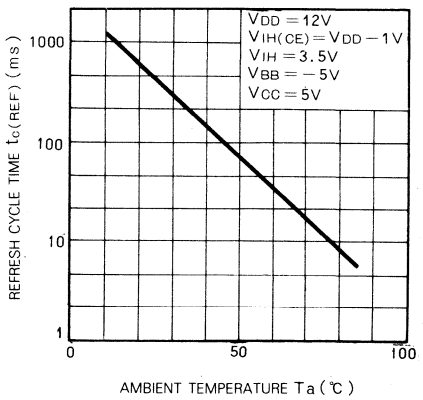
ADDRESS ACCESS TIME VS. AMBIENT TEMPERATURE



AVERAGE SUPPLY CURRENT FROM VBB VS. AMBIENT TEMPERATURE



REFRESH CYCLE TIME VS. AMBIENT TEMPERATURE



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

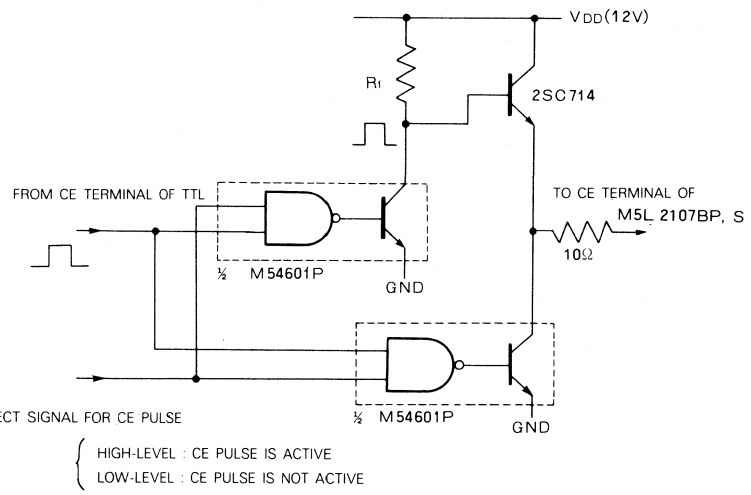
APPLICATION

Method of Refreshing

Since 64 memory cells designated by the X address can be refreshed in 1 cycle, (either read, write or read-modify-write), a read operation for all 64 addresses selected by the 6 address signals $A_0 \sim A_5$ must be performed within 2ms to refresh all 4096 memory cells. If the chip is refreshed during a write cycle or a read-modify-write cycle, then signal $\overline{CS}$ must be kept low; during a read cycle, $\overline{CS}$ can be either high or low. If a read operation is executed when the chip is in the non-designated state with $\overline{CS}$ high, refreshing can be performed with the output terminal $\overline{D_{OUT}}$ in the floating (high-impedance) state. Thus all the M5L 2107BP, S used in the memory system can be refreshed in only 64 cycles.

Recommended Driver Circuit for Chip Enable Pulse

COMPONENTS : M54601P · · · · · · · 1
2SC714 · · · · · · · · 1
RESISTORS · · · · · 2



HIGH-LEVEL : CE PULSE IS ACTIVE
LOW-LEVEL : CE PULSE IS NOT ACTIVE

Note 1 : R_1 is determined according to a required rise time of the CE pulse. For example, when C_L capacitance load for the CE pulse is 300pF, R_1 is set at 300Ω ($\frac{1}{2}$W) ; then, rise time $tr = 30ns$; fall time $tf = 30ns$; $t_{PHL} = 20ns$; $t_{PLH} = 20ns$.

2 : One M54601P dual peripheral positive AND driver circuit should be used for each CE driver circuit.

M5L 2111A P, S; P-2, S-2; P-4, S-4

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

DESCRIPTION

This is a family of 256-word by 4-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.

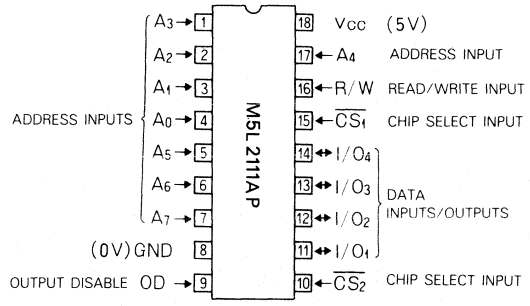
The input and output terminals are common, and an OD terminal is provided.

FEATURES

Parameter	M5L 2111AP,S-2	M5L 2111AP,S	M5L 2111AP,S-4
Access time (max)	250ns	350ns	450ns
Cycle time (min)	250ns	350ns	450ns

- Low power dissipation: 150μW/bit (typ)
- Single 5V power supply
- Data holding at 1.5V supply voltage (optional)
- No clocks or refreshing required
- All inputs and outputs are directly TTL-compatible
- All outputs are three-state, with OR-tie capability
- Simple memory expansion by chip select input
- Common data inputs and outputs
- Interchangeable with Intel's 2111A series in pin configuration and electrical characteristics

PIN CONFIGURATION (TOP VIEW)



Outline 18P1 (M5L 2111A P)
18S1 (M5L 2111A S)

APPLICATION

- Small-capacity memory units

FUNCTION

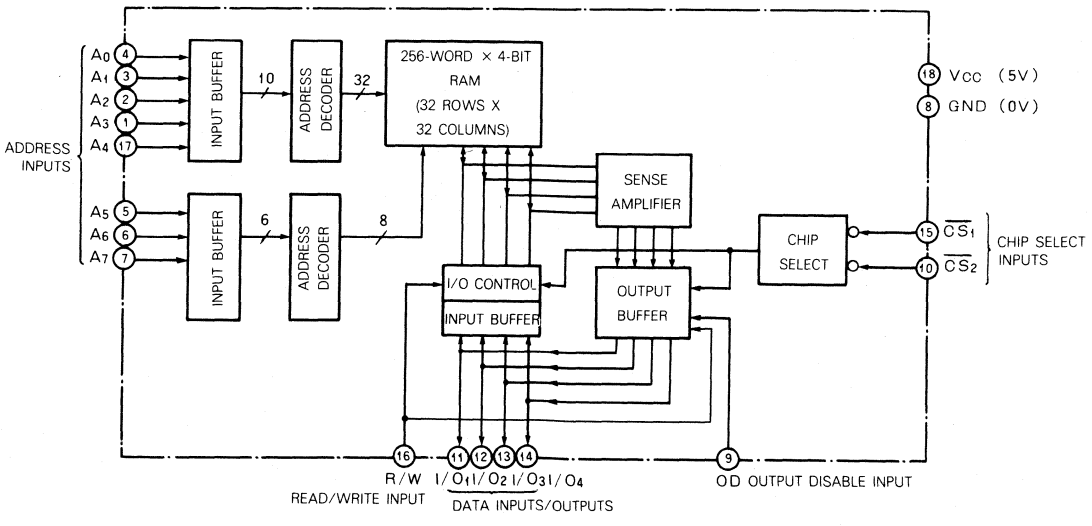
These devices provide common data input and output terminals. During a write cycle, when a location is designated by address signals A_0~A_7, the OD signal is kept high to keep the I/O terminals in the input mode, signal R/W goes low, and the data of the IN signal at that time is written.

During a read cycle, when a location is designated by address signals A_0~A_7, the OD signal is kept low to keep the I/O terminals in the output mode, signal R/W goes high, and the data of the designated address is available at the I/O terminals.

When signal $\overline{CS}_1$ or $\overline{CS}_2$ 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.

The memory data can be held at a supply voltage of 1.5V, enabling battery back-up operation during power failure and power-down operation in the standby mode.

BLOCK DIAGRAM



M5L 2111A P, S; P-2, S-2; P-4, S-4

1024-BIT (256-WORD BY 4-BIT) 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			−0.3～7	V
V_O	Output voltage			−0.3～7	V
Pd	Maximum power dissipation	M5L 2111AP	Ta＝25°C	700	mW
		M5L 2111AS		1000	mW
Topr	Operating free-air ambient temperature range			0～70	°C
Tstg	Storage temperature range	M5L 2111AP		−40～125	°C
		M5L 2111AS		−65～150	°C

RECOMMENDED OPERATING CONDITIONS (Ta＝0～10°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{IL}	Low-level input voltage	0		0.8	V
V_{IH}	High-level input voltage	2.2		V_{CC}	V

ELECTRICAL CHARACTERISTICS (Ta＝0～70°C, V_{CC}＝5V ±5%, 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		0.8	V
V_{OH}	High-level output voltage	I_{OH}＝−200μA	2.4			V
V_{OL}	Low-level output voltage	I_{OL}＝3.5mA			0.45	V
I_I	Input current	V_I＝0～5.25V			10	μA
I_{OZH}	Off-state high-level output current	$V_{I(\overline{CS_1})}$＝2.2V, V_O＝2.4V～V_{CC}			10	μA
I_{OZL}	Off-state low-level output current	$V_{I(\overline{CS_1})}$＝2.2V, V_O＝0.4V			−10	μA
I_{CC}	Supply current from V_{CC}	V_I＝5.25V (all inputs), output open, Ta＝25°C		30	60	mA
C_i	Input capacitance, all inputs	V_I＝GND, f＝1MHz, 25mVrms		3	5	pF
C_o	Output capacitance	V_O＝GND, f＝1MHz, 25mVrms		8	12	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 ±5%, unless otherwise noted) (Note 2)

Symbol	Parameter	M5L 2111A P,S-2			M5L 2111A P,S			M5L 2111A P,S-4			Unit
		Limits			Limits			Limits			
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
$t_{C(RD)}$	Read cycle time	250			350			450			ns
$t_{a(AD)}$	Address access time			250			350			450	ns
$t_{a(\overline{CS})}$	Chip select access time			180			180			180	ns
$t_{a(OD)}$	Output disable access time			130			150			150	ns
t_{PXZ}	Output disable time (Note 3)			100			100			100	ns
$t_{dv(AD)}$	Data valid time with respect to address	40			40			40			ns

Note 2 : Test conditions : Input pulse V_{IH}＝2.2V V_{IL}＝0.8V t_r＝t_f＝20ns, reference level ＝1.5V, load＝2TTL, C_L＝100pF.

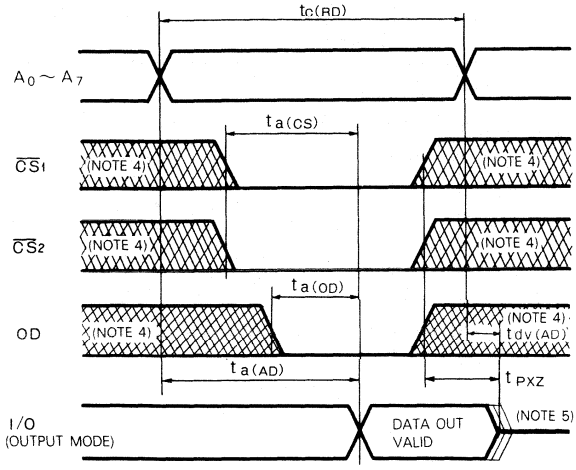
Note 3 : t_{PXZ} is with respect to $\overline{CS_1}$, $\overline{CS_2}$, or OD, whichever occurs first.

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

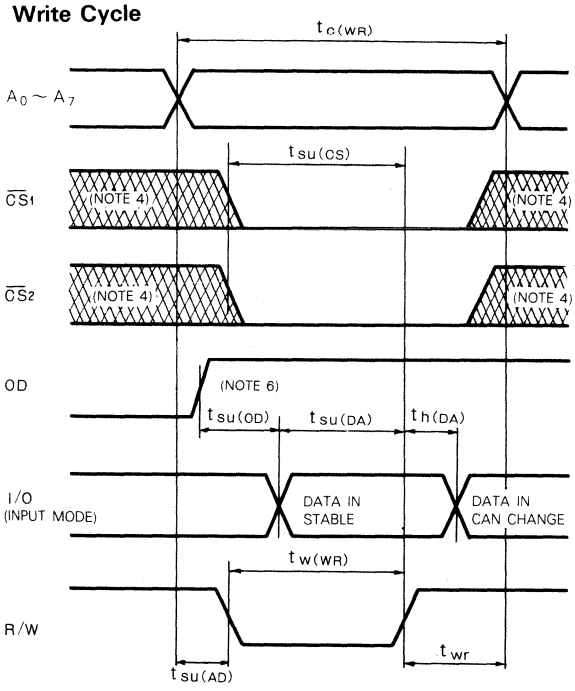
Symbol	Parameter	M5L 2111A P, S 2			M5L 2111A P, S			M5L 2111A P, S-4			Unit
		Limits			Limits			Limits			
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
$t_{C(WR)}$	Write cycle time	170			220			270			ns
$t_{W(WR)}$	Write pulse width	150			200			250			ns
$t_{su(AD)}$	Address setup time with respect to write	20			20			20			ns
t_{wr}	Write recovery time	0			0			0			ns
$t_{su(OD)}$	Output disable setup time with respect to data in	20			20			20			ns
$t_{su(DA)}$	Data setup time	100			150			170			ns
$t_{h(DA)}$	Data hold time	0			0			0			ns
$t_{su(CS)}$	Chip select setup time	150			200			250			ns

TIMING DIAGRAM

Read Cycle



Write Cycle



Note 4 : Hatching indicates the state is unknown.

5 : 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}.

6 : The input signals from the external circuits should not be applied to the I/O terminals, for during this period they are in output mode.

POWER-DOWN OPERATION (OPTIONAL) These characteristics are guaranted only under custom specifications.

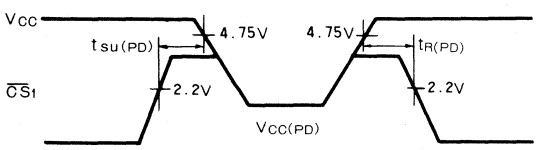
Electrical Characteristics (T_a = 0 ~ 70°C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$V_{CC(PD)}$	Power-down supply voltage		1.5			V
$V_{I(\overline{CS1})}$	Power-down chip select input voltage	2.2V ≦ $V_{CC(PD)}$ ≦ V_{CC}	2.2			V
		1.5V ≦ $V_{CC(PD)}$ ≦ 2.2V	$V_{CC(PD)}$			V
$I_{CC(PD1)}$	Power-down supply current from V_{CC}	V_{CC} = 1.5V, all inputs = 1.5V		15	30	mA
$I_{CC(PD2)}$	Power-down supply current from V_{CC}	V_{CC} = 2.0V, all inputs = 2.0V		20	40	mA

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

Symbol	Parameter	Test conditions	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

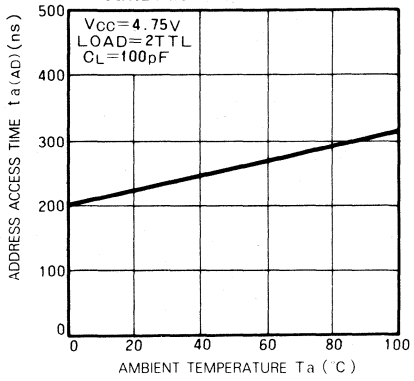
Timing Diagram



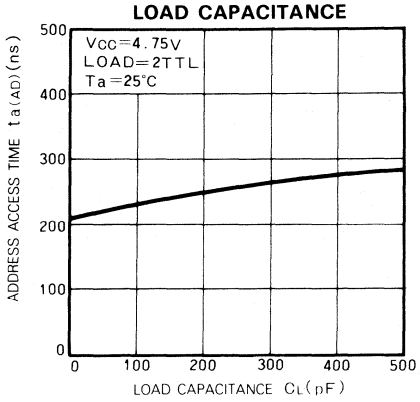
1024-BIT (256-WORD BY 4-BIT) STATIC RAM

TYPICAL CHARACTERISTICS

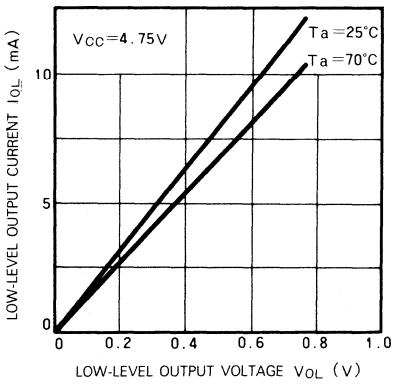
ADDRESS ACCESS TIME VS. AMBIENT TEMPERATURE



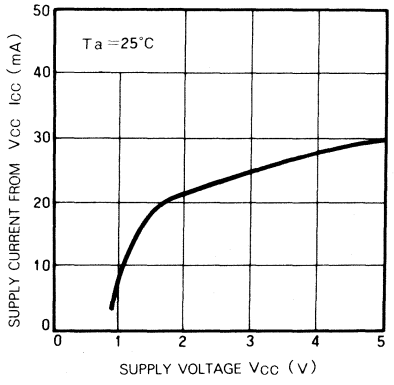
ADDRESS ACCESS TIME VS. LOAD CAPACITANCE



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



SUPPLY CURRENT FROM Vcc VS. SUPPLY VOLTAGE Vcc



M5L 2112A P, S; P-2, S-2; P-4, S-4

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

DESCRIPTION

This is a family of 256-word by 4-bit static RAMs fabricated with the N-channel silicon-gate MOS process and designed for simple interfacing. They operate from a single 5V supply, as does TTL, and are directly TTL-compatible.

The input and output terminals are common.

FEATURES

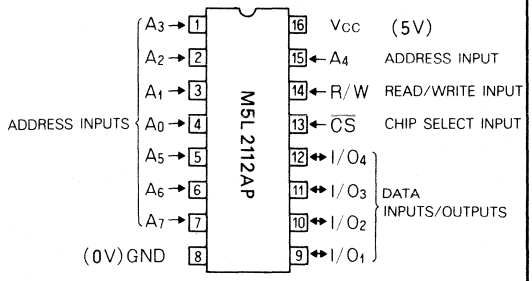
Parameter	M5L 2112A P, S-2	M5L 2112A P, S	M5L 2112A P, S-4
Access time (max)	250ns	350ns	450ns
Cycle time (min)	250ns	350ns	450ns

- Low power dissipation: 150μW/bit (typ)
- Single 5V supply voltage
- Data holding at 1.5V supply voltage (optional)
- Requires no clocks or refreshing
- All inputs and outputs are directly TTL-compatible
- All outputs are three-state, with OR-tie capability
- Simple memory expansion by chip select input
- Common data inputs and outputs
- Interchangeable with Intel's 2112A series in pin configuration and electrical characteristics

APPLICATION

- Small-capacity memory units

PIN CONFIGURATION (TOP VIEW)



Outline 16P1 (M5L 2112A P)
16S1 (M5L 2112A S)

FUNCTION

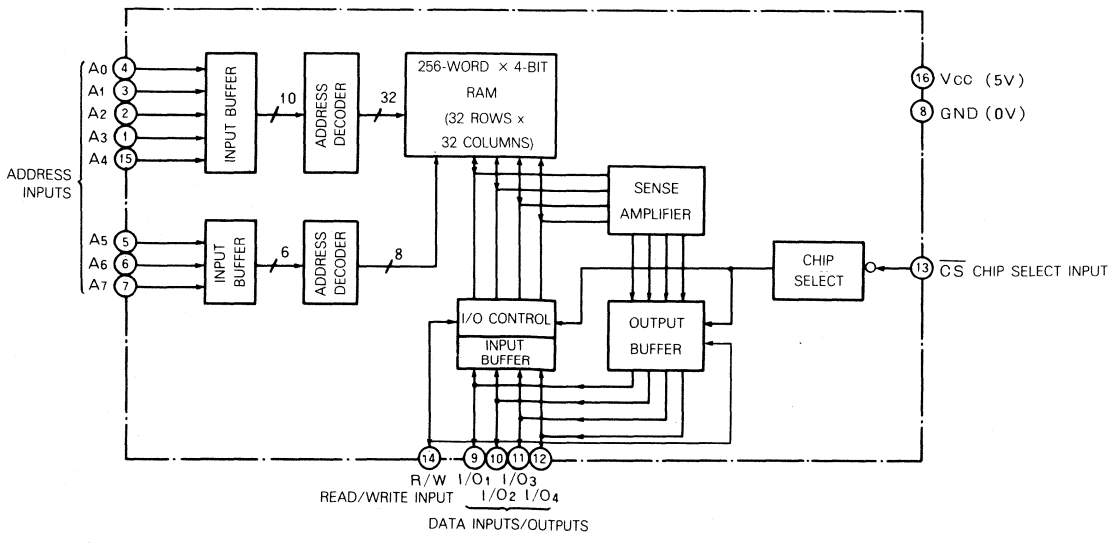
These devices provide common data input and output terminals. During a write cycle, when a location is designated by address signals A_0~A_7 and signal R/W goes low, the data of the I/O signal at that time is written.

During a read cycle, when a location is designated by address signal A_0~A_7 and R/W goes high, 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 other output terminals.

The memory data can be held at a supply voltage of 1.5V, enabling battery back-up operation during power failure and power-down operation in the standby mode.

BLOCK DIAGRAM



1024-BIT (256-WORD BY 4-BIT) 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			−0.3～7	V
V_O	Output voltage			−0.3～7	V
Pd	Maximum power dissipation	M5L 2112A P	Ta＝25°C	700	mW
		M5L 2112A S		1000	mW
Topr	Operating free-air ambient temperature range			0～70	°C
Tstg	Storage temperature range	M5L 2112A P		−40～125	°C
		M5L 2112A S		−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_{IL}	Low-level input voltage	0		0.8	V
V_{IH}	High-level input voltage	2.2		V_{CC}	V

ELECTRICAL CHARACTERISTICS (Ta＝0～70°C, V_{CC}＝5V ±5%, 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		0.8	V
V_{OH}	High-level output voltage	I_{OH}＝−200μA	2.4			V
V_{OL}	Low-level output voltage	I_{OL}＝3.5mA			0.45	V
I_I	Input current	V_I＝0～5.25V			10	μA
I_{OZH}	Off-state high-level output current	$V_{I(\overline{CS})}$＝2.2V, V_O＝2.4V～V_{CC}			10	μA
I_{OZL}	Off-state low-level output current	$V_{I(\overline{CS})}$＝2.2V, V_O＝0.4V			−10	μA
I_{CC}	Supply current from V_{CC}	V_I＝5.25V (all inputs), output open, Ta＝25°C		30	60	mA
C_I	Input capacitance, all inputs	V_I＝GND, f＝1MHz, 25mVrms		3	5	pF
C_O	Output capacitance	V_O＝GND, f＝1MHz, 25mVrms		8	12	pF

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

SWITCHING CHARACTERISTICS (For Read Cycle) ($Ta = 0 \sim 70°C$, $V_{CC} = 5V \pm 5\%$, unless otherwise noted) (Note 2)

Symbol	Parameter	M5L 2112A P, S-2			M5L 2112A P, S			M5L 2112A P, S-4			Unit
		Limits			Limits			Limits			
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
$t_{c(RD)}$	Read cycle time	250			350			450			ns
$t_{a(AD)}$	Address access time			250			350			450	ns
$t_{a(\overline{CS})}$	Chip select access time			180			180			180	ns
$t_{pxz(\overline{CS})}$	Output disable time with respect to chip select	40			40			40			ns

4

TIMING REQUIREMENTS ($Ta = 0 \sim 70°C$, $Vcc = 5V \pm 5\%$, unless otherwise noted) (Note 2)

Write Cycle 1

Symbol	Parameter	M5L 2112A P, S-2			M5L 2112A P, S			M5L 2112A P, S-4			Unit
		Limits			Limits			Limits			
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
$t_{c(WR)1}$	Write cycle time	170			220			270			ns
$t_{su(AD)1}$	Address setup time with respect to write pulse	20			20			20			ns
$t_{w(WR)1}$	Write pulse width	150			200			250			ns
t_{wr1}	Write recovery time	0			0			0			ns
$t_{su(DA)1}$	Data setup time	100			150			170			ns
$t_{h(DA)1}$	Data hold time	0			0			0			ns
$t_{h(\overline{CS})1}$	Chip select hold time	0			0			0			ns
$t_{su(WR)1}$	Write pulse setup time with respect to chip select	0			0			0			ns
$t_{su(\overline{CS})1}$	Chip select setup time	100			150			170			ns

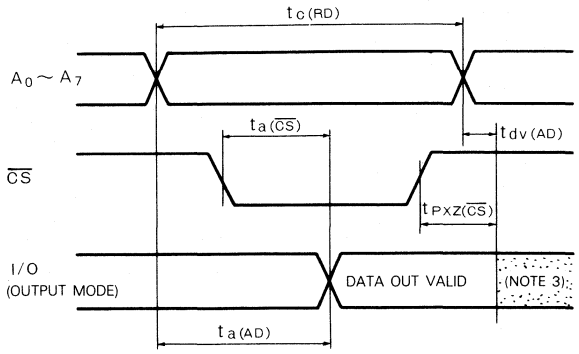
Note 2 : Test conditions : Input pulse $V_{IH} = 2.2V$ $V_{IL} = 0.8V$ $t_r = t_f = 20ns$, reference level=1.5V, load=2TTL, $C_L = 100pF$.

Write Cycle 2 (Note 2)

Symbol	Parameter	M5L 2112A P, S-2			M5L 2112A P, S			M5L 2112A P, S-4			Unit
		Limits			Limits			Limits			
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
$t_{c(WR)2}$	Write cycle time	170			220			270			ns
$t_{su(AD)2}$	Address setup time with respect to write pulse	20			20			20			ns
$t_{w(WR)2}$	Write pulse width	150			200			250			ns
t_{wr2}	Write recovery time	0			0			0			ns
$t_{su(DA)2}$	Data setup time	100			150			170			ns
$t_{h(DA)2}$	Data hold time	0			0			0			ns
$t_{h(\overline{CS})2}$	Chip select hold time	0			0			0			ns
$t_{su(\overline{CS})2}$	Chip select setup time	0			0			0			ns
$t_{pxz(WR)2}$	Output disable time with respect to write pulse			50			50			80	ns

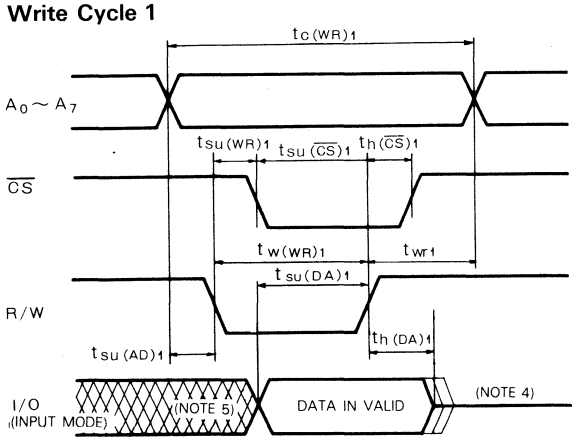
1024-BIT (256-WORD BY 4-BIT) STATIC RAM

TIMING DIAGRAMS
Read Cycle

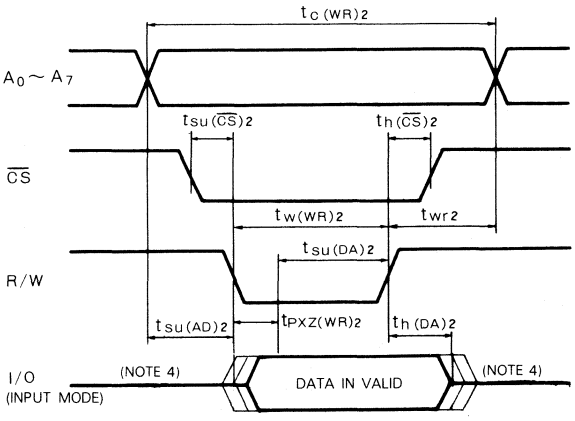


Note 3 : In this period, the data out is valid for a definition of $t_{dv(AD)}$ and is in the floating state for a definition of $t_{PXZ(\overline{CS})}$.

Write Cycle 1



Write Cycle 2



Note 4 : The input signals from the external circuits should not be applied to the I/O terminals (keeping them three-state), for during this period the I/O terminals are in the output mode.

5 : The input signals from the external circuits can be applied to the I/O terminals since the signal $\overline{CS}$ is delayed in relation to signal R/W.

POWER-DOWN OPERATION (OPTIONAL) These characteristics are guaranteed only under custom specifications.

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

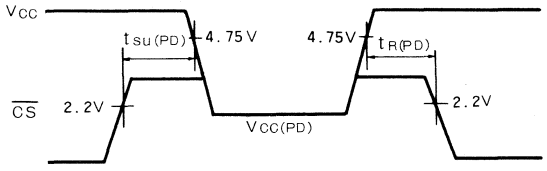
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$V_{CC(PD)}$	Power-down supply voltage		1.5			V
$V_I(\overline{CS})$	Power-down chip select input voltage	2.2V ≦ $V_{CC(PD)}$ ≦ V_{CC}	2.2			V
		1.5V ≦ $V_{CC(PD)}$ ≦ 2.2V	$V_{CC(PD)}$			V
$I_{CC(PD1)}$	Power-down supply current from V_{CC}	V_{CC} = 1.5V, all inputs = 1.5V		15	30	mA
$I_{CC(PD2)}$	Power-down supply current from V_{CC}	V_{CC} = 2.0V, all inputs = 2.0V		20	40	mA

Timing Requirements (T_a = 0 ~ 70°C, V_{CC} = 5V ± 5%, 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

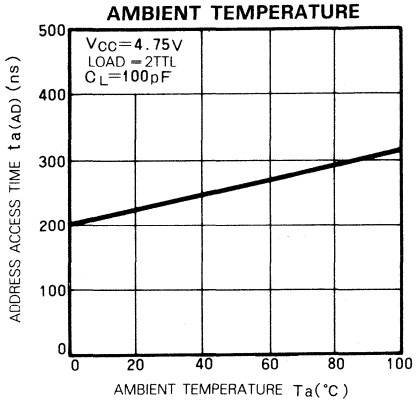
Power-Down (Optional)



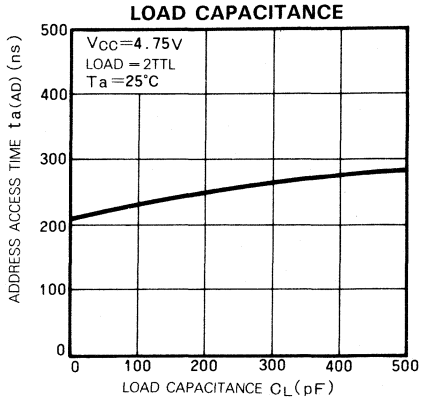
1024-BIT (256-WORD BY 4-BIT) STATIC RAM

TYPICAL CHARACTERISTICS

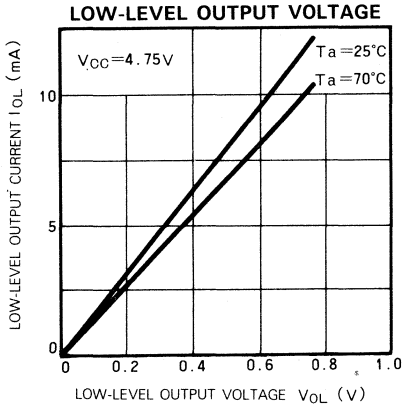
ADDRESS ACCESS TIME VS. AMBIENT TEMPERATURE



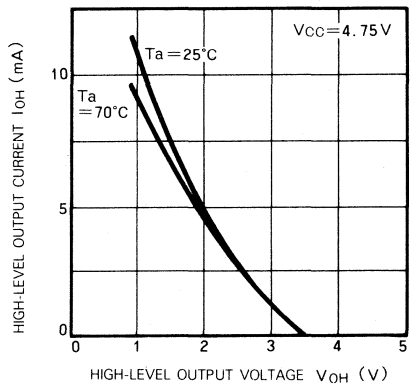
ADDRESS ACCESS TIME VS. LOAD CAPACITANCE



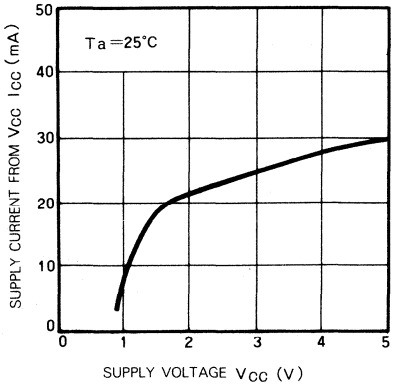
LOW-LEVEL OUTPUT CURRENT VS. LOW-LEVEL OUTPUT VOLTAGE



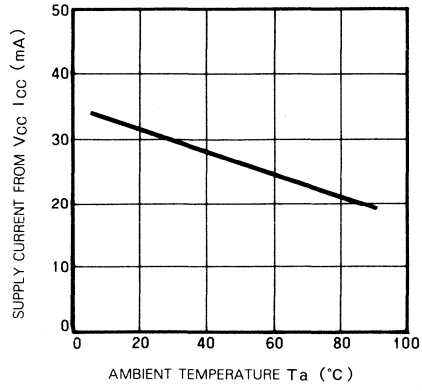
HIGH-LEVEL OUTPUT CURRENT VS. HIGH-LEVEL OUTPUT VOLTAGE



SUPPLY CURRENT FROM VCC VS. SUPPLY VOLTAGE VCC



SUPPLY CURRENT FROM VCC VS. AMBIENT TEMPERATURE



MITSUBISHI ELECTRIC

M5L 2114L P, S; P-2, S-2; P-3, S-3

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

DESCRIPTION

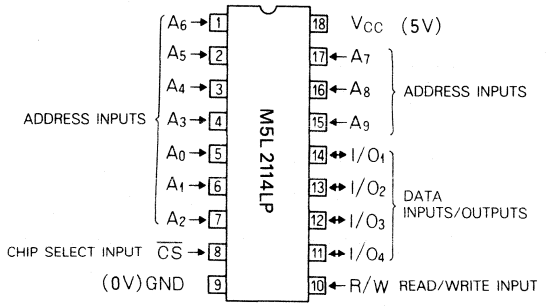
This is a family of 4096-bit static RAMs organized as 1024 words of 4 bits and designed for simple interfacing. They are fabricated using N-channel silicon-gate MOS technology. They operate with a single 5V supply, as does TTL, and the inputs and outputs are directly TTL compatible. I/O terminals are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

FEATURES

Parameter	M5L 2114LP, S-2	M5L 2114LP, S-3	M5L 2114LP, S
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

PIN CONFIGURATION (TOP VIEW)



Outline 18P1 (M5L 2114LP)
18S1 (M5L 2114LS)

APPLICATION

- Small-capacity memory units

FUNCTION

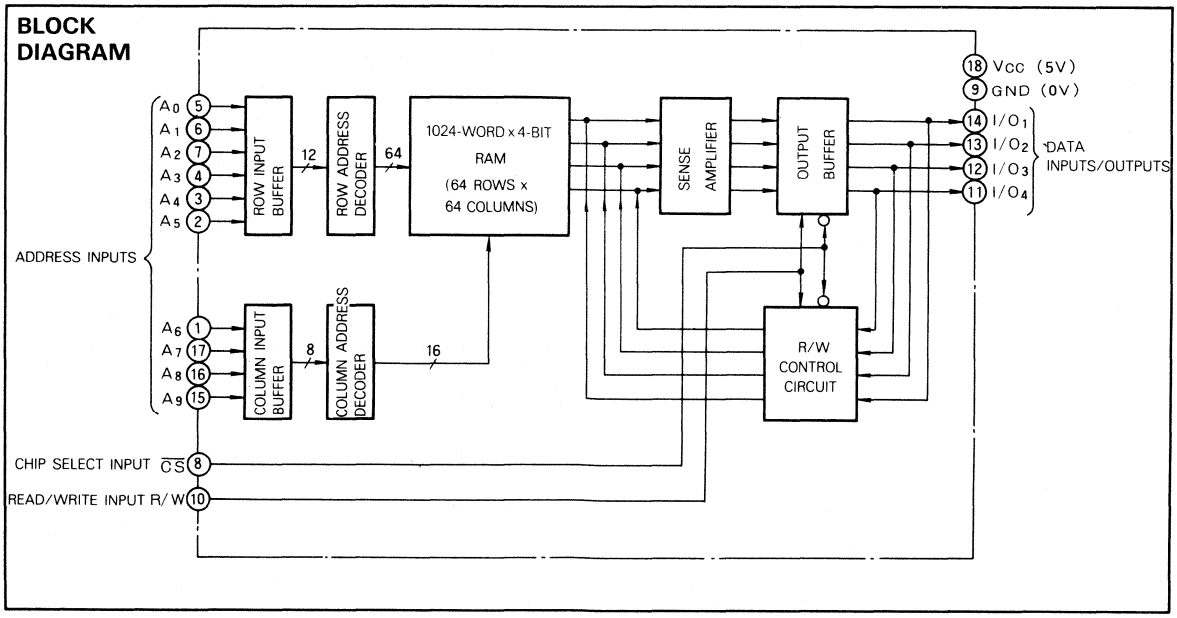
These devices operate with a single 5V power supply, and the inputs and outputs are directly compatible with TTL. All circuits are completely static, rendering external clock and refresh operations unnecessary, and making the members of the series extremely easy to use. Common data input and output terminals 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	M5L 2114L P	Ta=25℃	700	mW
		M5L 2114L S	Ta=25℃	1000	mW
T_{opr}	Operating free-air ambient temperature range			0∼70	℃
T_{stg}	Storage temperature range	M5L 2114L P		−40∼125	℃
		M5L 2114L S		−65∼150	℃

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}=−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)

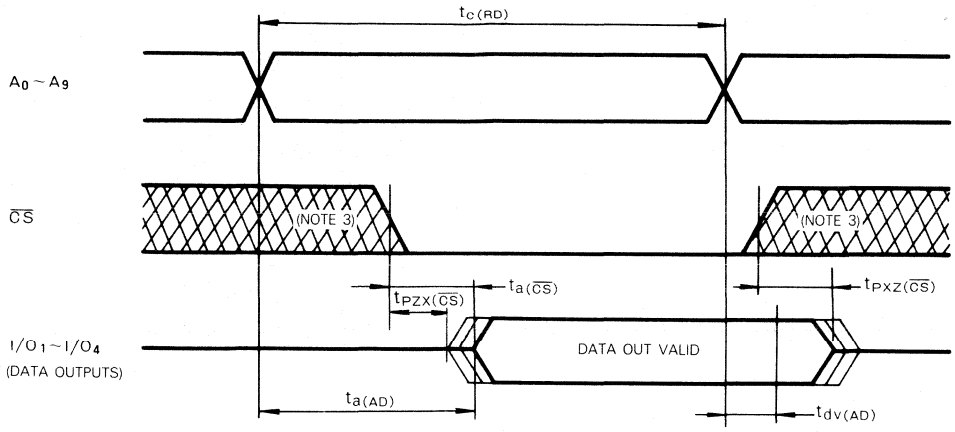
Symbol	Parameter	Alt. symbol	M5L 2114L P-2, S-2			M5L 2114L P-3, S-3			M5L 2114L P, S			Unit
			Limits			Limits			Limits			
			Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
$t_{C(WR)}$	Write cycle time	t_{WC}	200			300			450			ns
$t_{su(AD)}$	Address setup time with respect to write pulse		0			0			0			ns
$t_{w(WR)}$	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)

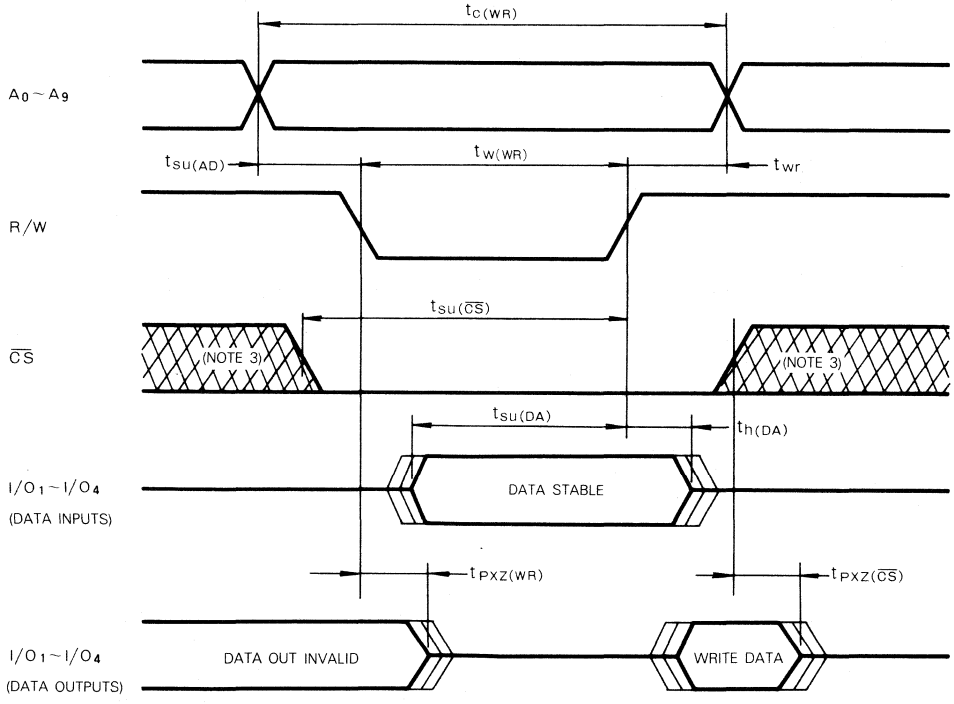
Symbol	Parameter	Alt. symbol	M5L 2114L P, S-2			M5L 2114L P, S-3			M5L 2114L P, S			Unit
			Limits			Limits			Limits			
			Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
$t_{C(RD)}$	Read cycle time	t_{RC}	200			300			450			ns
$t_{a(AD)}$	Address access time	t_A			200			300			450	ns
$t_{a(\overline{CS})}$	Chip 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

TIMING DIAGRAMS

Read Cycle



Write Cycle

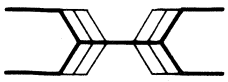


Note : 2 Test conditions

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.

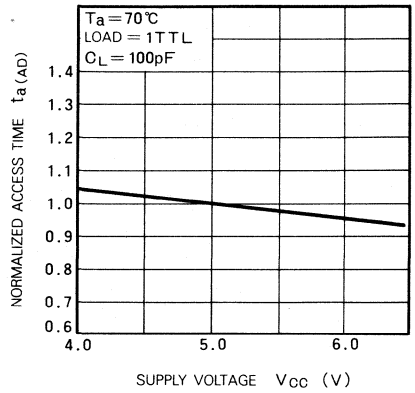


The center line indicates a floating (high-impedance) state.

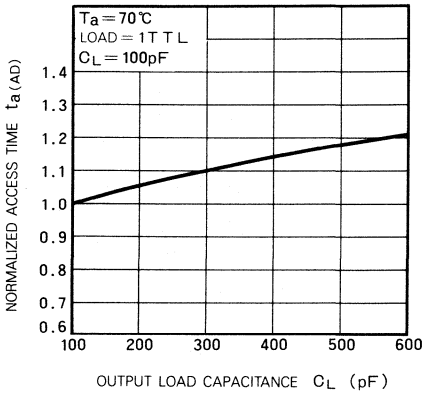
4096-BIT (1024-WORD BY 4-BIT) STATIC RAM

TYPICAL CHARACTERISTICS

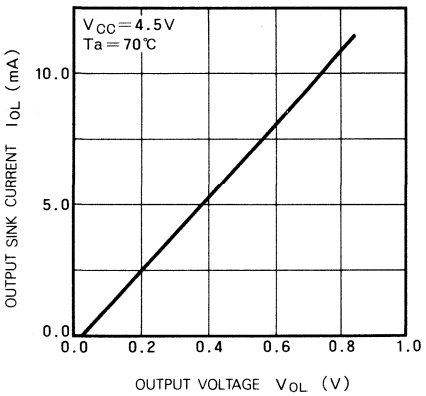
NORMALIZED ACCESS TIME VS. SUPPLY VOLTAGE



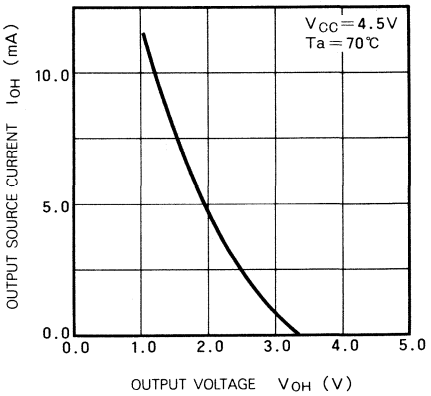
NORMALIZED ACCESS TIME VS. OUTPUT LOAD CAPACITANCE



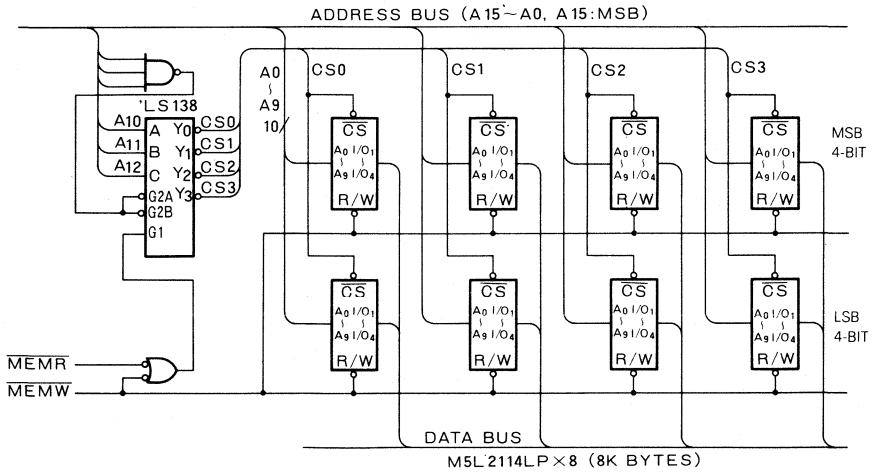
OUTPUT SINK CURRENT VS. OUTPUT VOLTAGE



OUTPUT SOURCE CURRENT VS. OUTPUT VOLTAGE



TYPICAL APPLICATION (for an M5L 8080A P CPU)



MITSUBISHI ELECTRIC

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 1μA (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: 5nW/bit (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

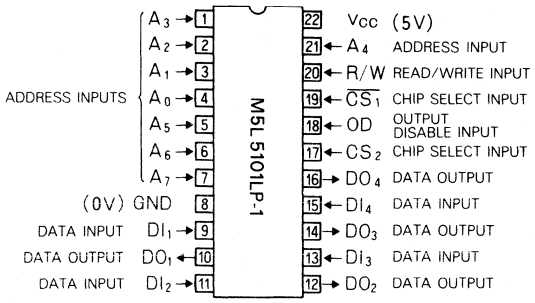
APPLICATION

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

FUNCTION

The device provides separate data input and output terminals.

PIN CONFIGURATION (TOP VIEW)



Outline 22P1

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

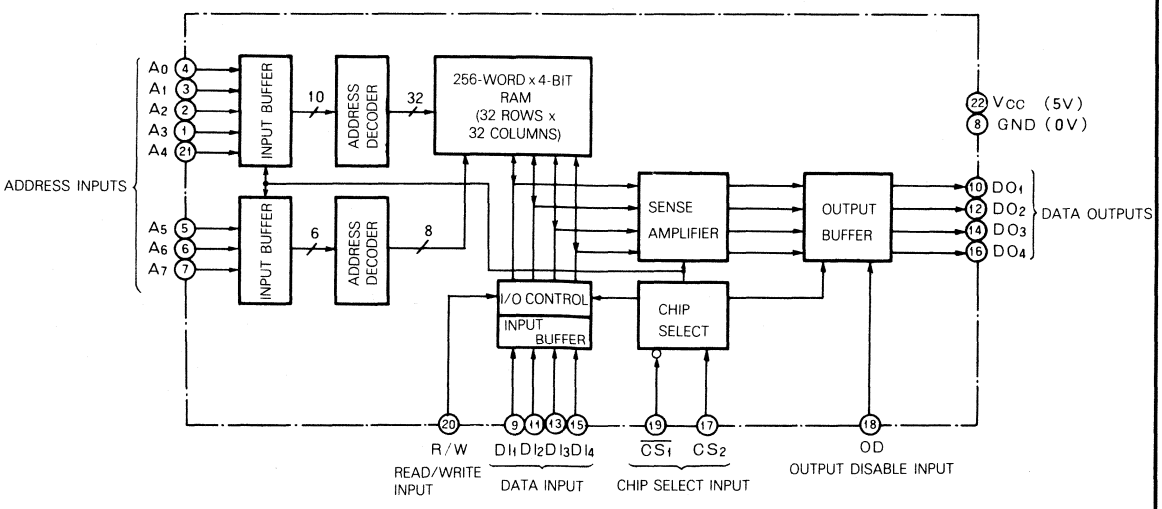
During a read cycle, when a location is designated by address signals A_0~A_7, and signal R/W goes high, the data of the designated address is available at the DO terminals.

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.

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	With respect to GND	−0.3～V_{CC}+0.3	V
V_O	Output voltage	With respect to GND	0～V_{CC}	V
Pd	Maximum power dissipation	Ta＝25°C	700	mW
Topr	Operating free-air ambient temperature range		0～70	°C
Tstg	Storage temperature range		−40～125	°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	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°C, 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＝0～5.5V			±1	μA
I_{OZH}	Off-state high-level output current	$V_{I(\overline{CS1})}$＝2.2V, V_O＝2.4V～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}	$\overline{CS}_1 \leqq 0.01$V, other inputs＝$V_{CC}$, Output open		9	22	mA
I_{CC2}	Supply current from V_{CC}	$\overline{CS}_1 \leqq 0.01$V, other inputs＝2.2V, Output open		13	27	mA
I_{CC3}	Supply current from V_{CC}	$CS_2 \leqq 0.2$V			1	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°C, 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
$t_{su(\overline{CS1})}$	Chip select setup time	t_{CW1}		350			ns
$t_{su(CS2)}$	Chip select setup time	t_{CW2}		350			ns

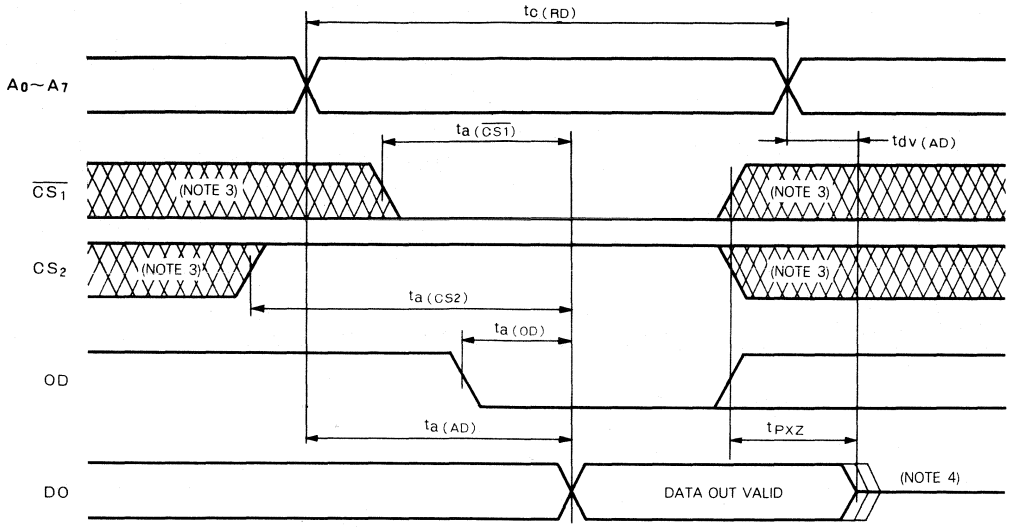
SWITCHING CHARACTERISTICS (For Read Cycle) (Ta＝0～70°C, V_{CC}＝5V±10%, unless without noted)

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
$t_{a(\overline{CS1})}$	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}				130	ns
$t_{dv(AD)}$	Data valid time with respect to address	t_{OH1}		0			ns

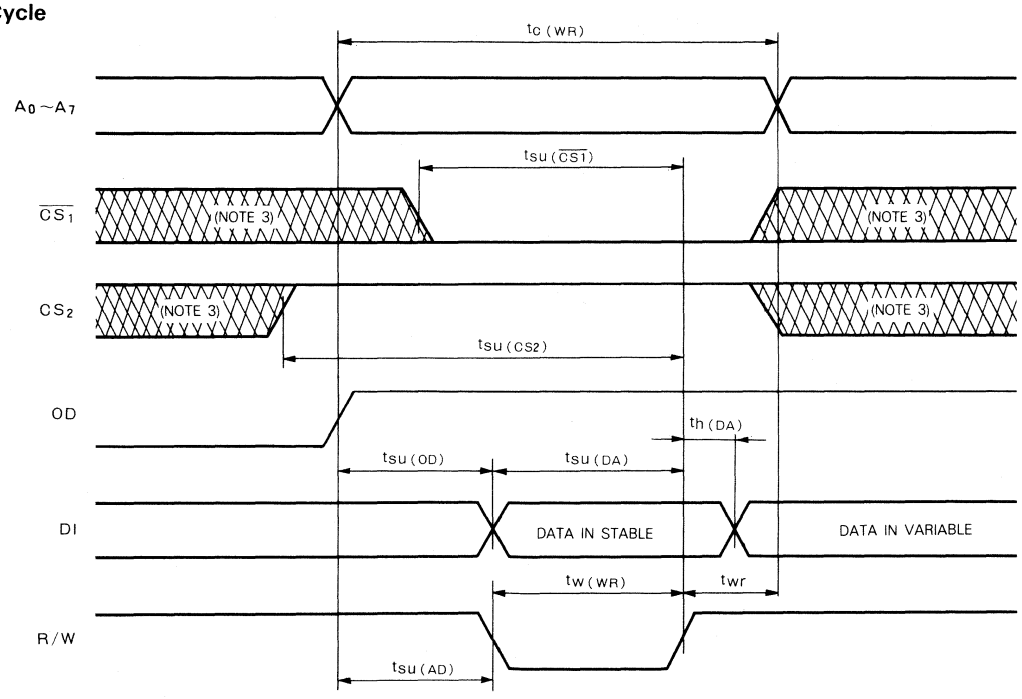
Note 2 : t_{PXZ} is from $\overline{CS_1}$, CS_2, or OD, whichever occurs first.

TIMING DIAGRAMS

Read Cycle

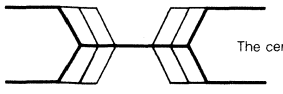


Write Cycle



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

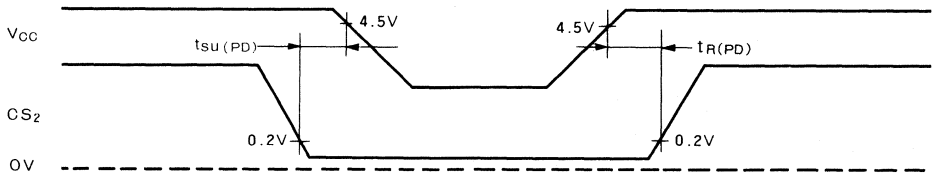
Electrical Characteristics ($T_a = 0 \sim 70°C$, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$V_{CC(PD)}$	Power-down supply voltage		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 $=2V$			1	μA

Timing Requirements ($T_a = 0 \sim 70°C$, $V_{CC}=5V \pm 10\%$, 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



M5T 4044 P-20, S-20; P-30, S-30; P-45, S-45

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

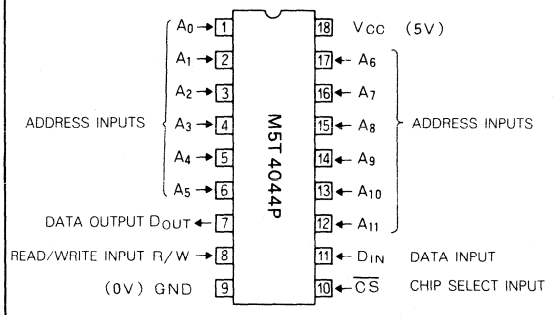
Parameter	M5T 4044 P,S-20	M5T 4044 P,S-30	M5T 4044 P,S-45
Access time (max)	200ns	300ns	450ns
Cycle time (min)	200ns	300ns	450ns

- 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 18P1 (M5T4044P)
18S1 (M5T4044S)

FUNCTION

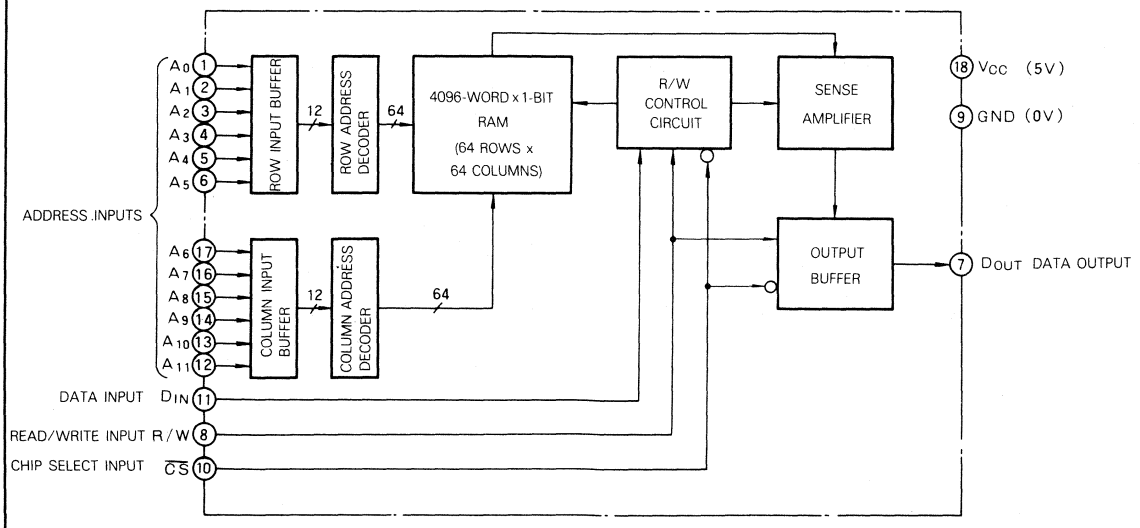
These devices are very convenient to use, as they feature static circuits which require neither external clocks nor refreshing, and all inputs and outputs are directly compatible with TTL.

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



M5T 4044 P-20, S-20; P-30, S-30; P-45, S-45

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	M5T 4044P	Ta=25℃	700	mW
		M5T 4044S	Ta=25℃	1000	mW
T_{opr}	Operating free-air ambient temperature range			0～70	℃
T_{stg}	Storage temperature range	M5T 4044P		−40～125	℃
		M5T 4044S		−65～150	℃

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℃		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,S-20			M5T 4044P-30,S-30			M5T 4044P-45,S-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

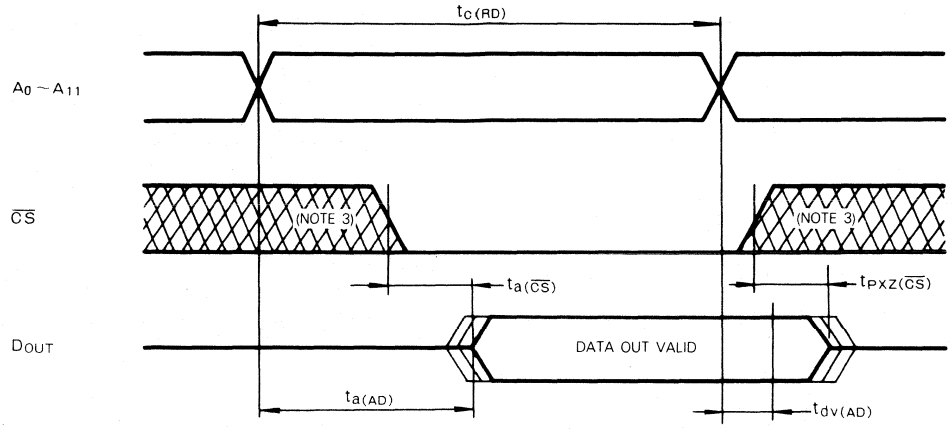
SWITCHING CHARACTERISTICS (For Read Cycle) (Ta = 0 ~ 70℃, V_{CC} = 5V ± 10%, unless otherwise noted) (Note 2)

Symbol	Parameter	M5T 4044P-20,S-20			M5T 4044P-30,S-30			M5T 4044P-45,S-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

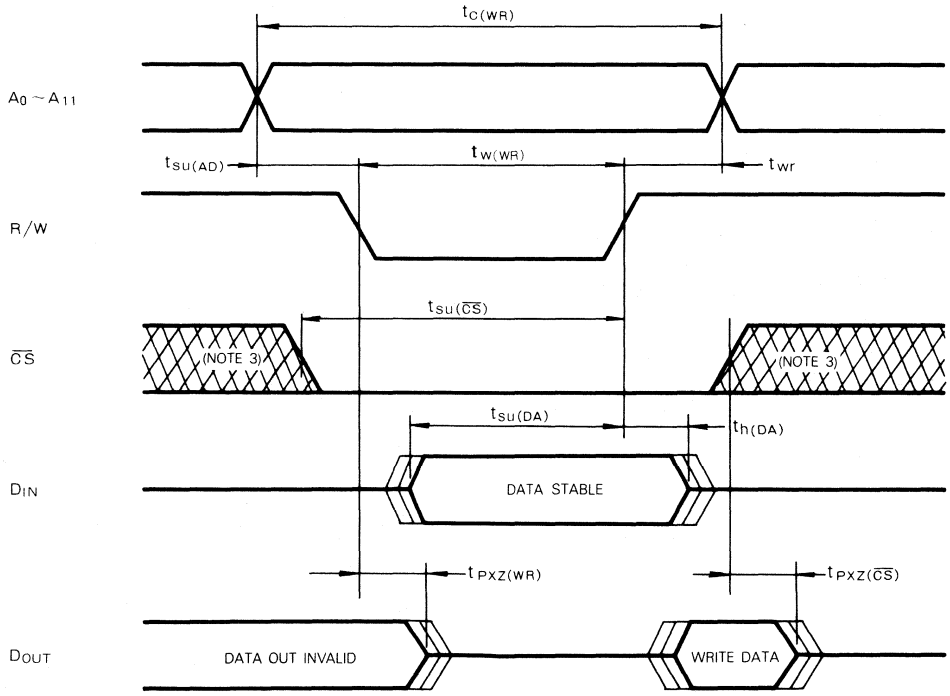
4

TIMING DIAGRAMS

Read Cycle



Write Cycle

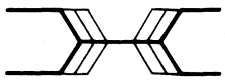


Note : 2 Test conditions

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.

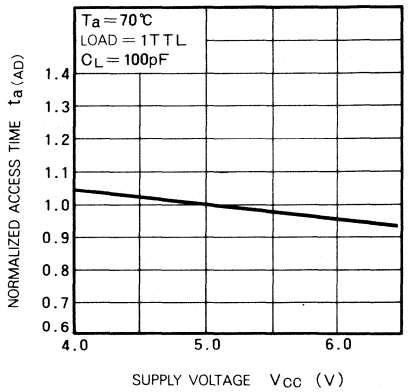


The center line indicates a floating (high-impedance) state.

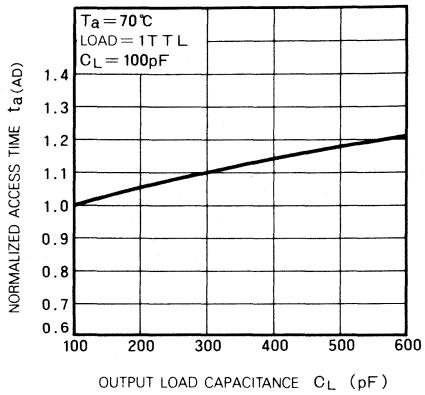
4096-BIT (4096-WORD BY 1-BIT) STATIC RAM

TYPICAL CHARACTERISTICS

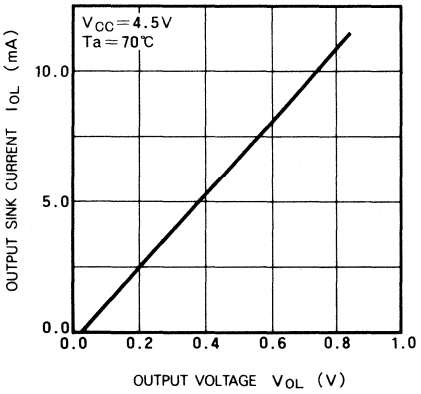
NORMALIZED ACCESS TIME VS. SUPPLY VOLTAGE



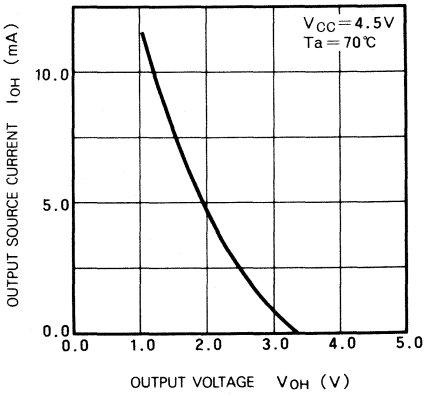
NORMALIZED ACCESS TIME VS. OUTPUT LOAD CAPACITANCE



OUTPUT SINK CURRENT VS. OUTPUT VOLTAGE

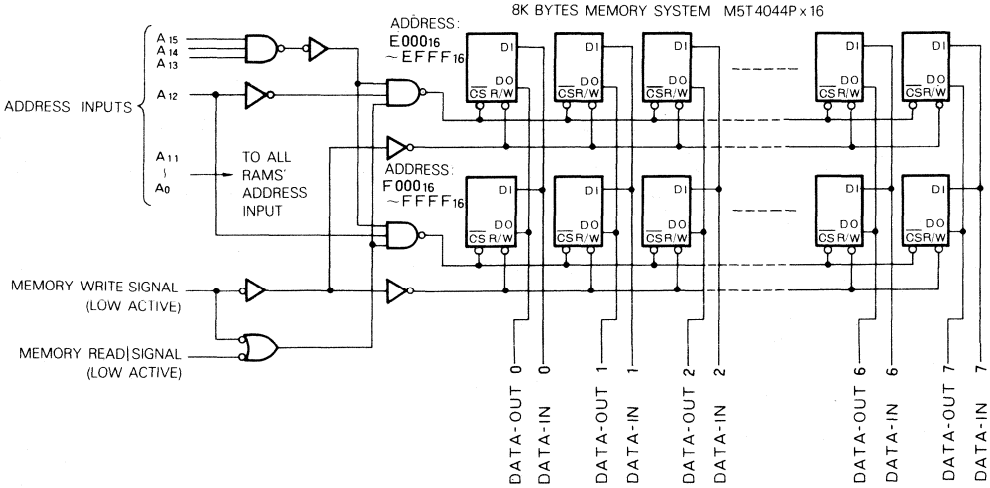


OUTPUT SOURCE CURRENT VS. OUTPUT VOLTAGE



TYPICAL APPLICATION (for 8K-Byte Memory System)

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



MITSUBISHI ELECTRIC

READ-ONLY MEMORIES

5

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.

The customer's object program specifications for the automatic design system for a mask ROM may be supplied in MELPS 8/85 binary, hexadecimal, BNPF, or Minato-hexadecimal form.

The form of the data is the same as the output from a MELPS 8/85 cross assembler or a PL/1μ cross compiler. It accepts either paper tape or magnetic tape as the input medium.

An EPROM in which a program is stored can also be used for a customer's specifications. A separate tape or a separate (set of) EPROM(s) should be produced for each object program.

Two copies of the tape or two EPROMs/sets of EPROMs should be supplied.

OBJECT PROGRAM FORMAT

- Object program addresses are absolute.
- The data can be in either MELPS 8/85 binary, hexadecimal, BNPF, or Minato-hexadecimal form.
- The output tape from a MELPS 8/85 cross assembler or a PL/1μ cross compiler can be used.
- The hexadecimal and BNPF forms are Intel-compatible.
- The character code can be ASCII or ISO, with or without parity.

OBJECT PROGRAM MEDIA

- Paper tape: 8-level, 25.4 mm (1 inch) wide
- Magnetic tape: 9-track, 800BPI, odd parity
- EPROM: The M5L 2708K, S, or M5L 2716K, are standard, but equivalent devices may be used.

ITEMS TO CONFIRM FOR ORDERING

- The form and medium of the object program
- Type M58333-XXXP or M58334-XXXP (the 3-digit number XXX, will be assigned by Mitsubishi)
- Designation of active levels for chip selects

SPECIFYING OBJECT PROGRAM FORMAT

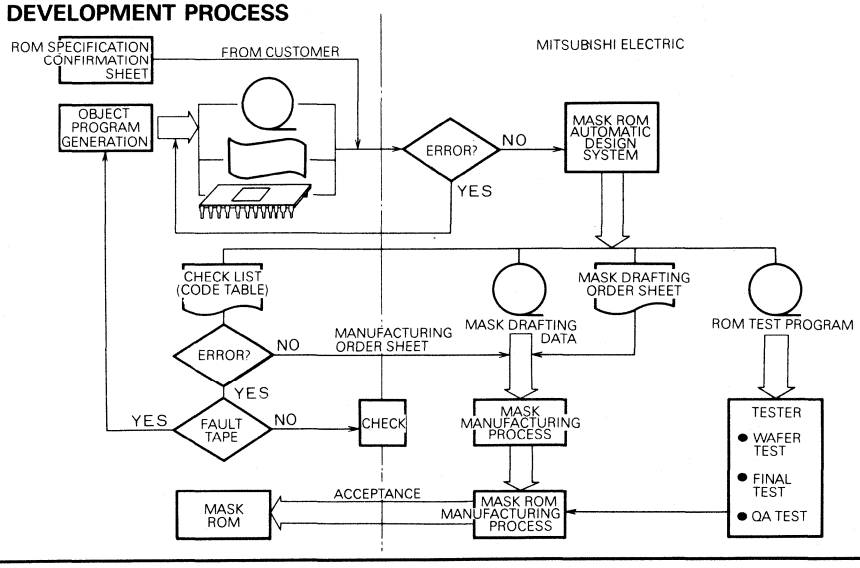
1. MELPS 8/85 Binary

The form of MELPS 8/85 binary is the same as the output from a MELPS 8/85 cross assembler or a PL/1μ cross compiler.

See a MELPS 8/85 cross assembler reference manual (GAM-SR00-02A) or a MELPS 8/85 PL/1μ cross compiler reference manual (GAM-SR00-09A).

- A separate tape should be prepared for each ROM of the object program, and two copies of the tape should be supplied.
- The final part code should be inserted at the end of each tape.
- The first address of the ROM should be specified.
- The region outside the range from the specified first address to the first address + maximum address of the ROM is ignored.

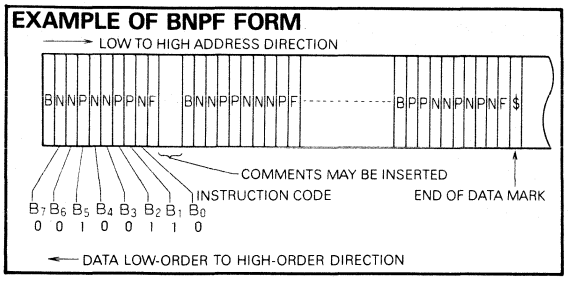
MASK ROM DEVELOPMENT PROCESS



MITSUBISHI ELECTRIC

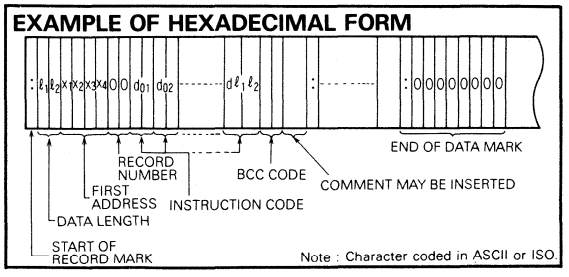
- It should be indicated whether the area to be programmed is ROM only, RAM only, or both.
- The level of bit code '1' should be specified as low or high.
- At least 100 frames of only sprocket holes should be punched at the start and the end of the tape.

2. BNPF

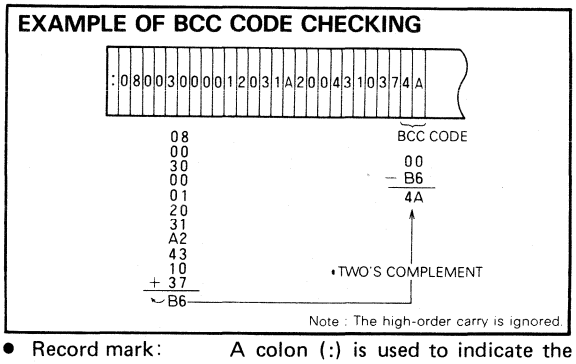


- This form is Intel-compatible.
- A separate tape should be prepared for each object program, and two copies of the tapes should be supplied. At the end of the data a '$' character should be created.
- When the data to be programmed is less than the maximum memory capacity, the unused area should be filled with appropriate codes, or the code to be filled should be specified in the confirmation sheet. If code is not specified, the unused area will be programmed low-level.
- Comments not containing 'B' or '$' characters may be inserted between the 'F' and 'B'.
- The character code is ASCII or ISO, with or without parity.
- The address is incremented in sequence by the data string.
- The magnitude of the bits between the 'B' and 'F' is defined as from high-order to low-order.
- At least 100 frames of only sprocket holes should be punched at the start and the end of the tape.
- The level of 'P' should be specified as either low or high, and as positive or negative logic.

3. Hexadecimal



- This form is Intel-compatible.
- A separate tape should be prepared for each ROM of the object program, and two copies of each tape should be supplied.
- A record of data length zero is considered as the end of one ROM's data.
- The first address of each ROM should be specified on its tapes.
- The region outside the range from the specified first address to the first address + maximum address of each ROM is ignored.
- The first address of records in the object program may be non-sequential.



- Record mark: A colon (:) is used to indicate the start of a record.
- Data length: The number of data bytes stored in the record. A two-digit hexadecimal number with the high-order digit ℓ_1 and the low-order digit ℓ_2.
- First address: Four hexadecimal digits give the first address (address of d_{01}) at which the data is to be loaded.
- Record type: Two hexadecimal digits specify the record type. It is normally '00'.
- Instruction code: Each instruction is represented by two hexadecimal digits. The number of instructions and data is limited to ℓ_1 ℓ_2.
- BCC code: The BCC code contains the hexadecimal two's complement 8-bit sum (the carry from the 8th bit is ignored) of the 8-bit data between the start of record mark and the BCC code.

- When the data to be programmed is less than the maximum memory capacity, the unused area should be filled with appropriate codes, or the code to be filled should be specified on the confirmation sheet. If the level is not specified, the unused area will be programmed low-level.
- The characters are coded in ASCII or ISO, with or without parity.
- The level of bit code '1' should be specified as either low or high and positive or negative logic.
- At least 100 frames of only sprocket holes should be punched at the start and the end of the tape.

4. Minato-Hexadecimal Form

- The paper tape should be supplied according to the form shown in reference manual GAM-SR00-32A.

5

5. CONFIRMATION MATERIAL

Note : Fill in all spaces except shaded areas marked with an ☆

Customer												Signature		
			DATE:					TEL:						
Type number								☆				☆ Mitsubishi		
M								—						

OBJECT PROGRAM FORMAT

Format	1. MELPS 8/85 binary.	2. Hexadecimal	3. BNPF
Medium	4. Paper tape (2 sets)	5. Magnetic tape (2 sets)	
Character code		6. ASCII	7. ISO
Parity		8. Odd	9. Even 10. None
ROM 1st address			
Region to be masked	11. All regions 12. ROM region 13. RAM region		
Instruction code	14. Punched bit programmed as high-level. 15. Punched bit programmed as low-level.	16. I's bit programmed as high-level. 17. I's bit programmed as low-level.	18. P's bit programmed as high-level. 19. P's bit programmed as low-level.
Address	20. Punched bit programmed as high-level. 21. Punched bit programmed as low-level.	22. I's bit programmed as high-level. 23. I's bit programmed as low-level.	

DATA OUTPUT 1. Positive Logic 2. Negative Logic

ADDRESS INPUT 1. Positive Logic 2. Negative Logic

ACTIVE CHIP-SELECT INPUT LEVEL

Type number	CS_1	CS_2	CS_3

CUSTOMER'S IDENTIFICATION MARK

If you require a special identification mark, please specify in the following format.

Mitsubishi IC type number

Note 1 : A mark field should start with the box at the extreme right.
Note 2 : The identification mark should be no more than 12 characters consisting of alphanumeric characters (except J, I and O) or dashes.

LIST TO BE PRESENTED 1. Yes (sets) 2. No

SOURCE LIST PRESENTATION 1. Necessary (sets) 2. Not Necessary

COMMENTS

DO NOT WRITE BELOW THIS LINE

M54700K, P, S

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

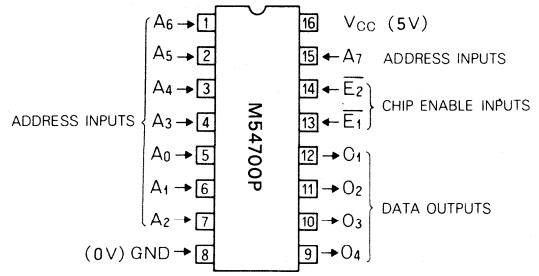
DESCRIPTION

The memory cells of the M54700K,P,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 16K1 (M54700K)
16P1 (M54700P)
16S1 (M54700S)

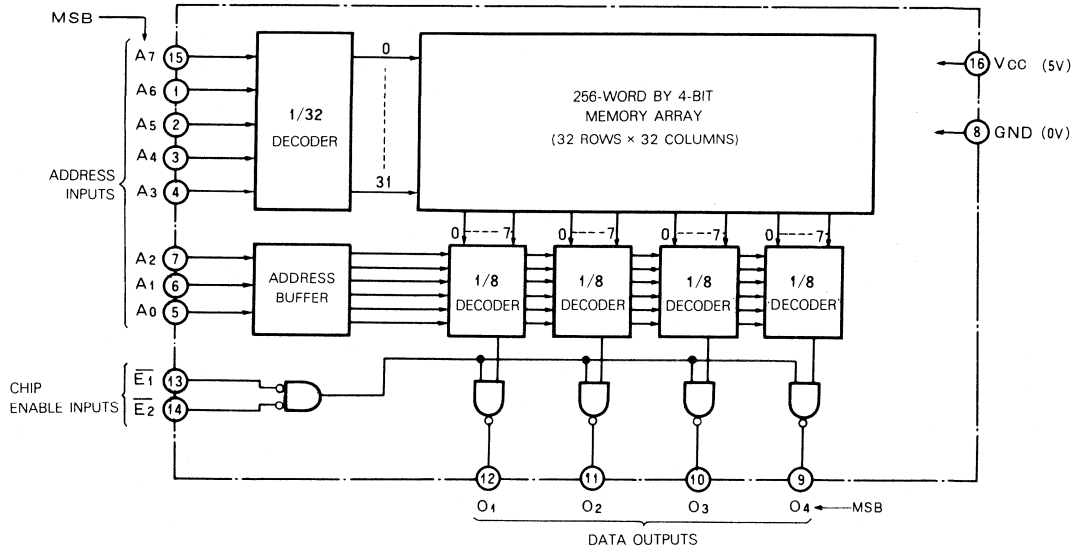
APPLICATION

- 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



MITSUBISHI ELECTRIC

1024-BIT (256-WORD BY 4-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		−55~125	°C
V_O	Output apply voltage	In case of programming	27	V
V_E	Chip enable apply voltage		35	V
$t_{w(P)}/t_{c(P)}$	Duty cycle		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

5

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

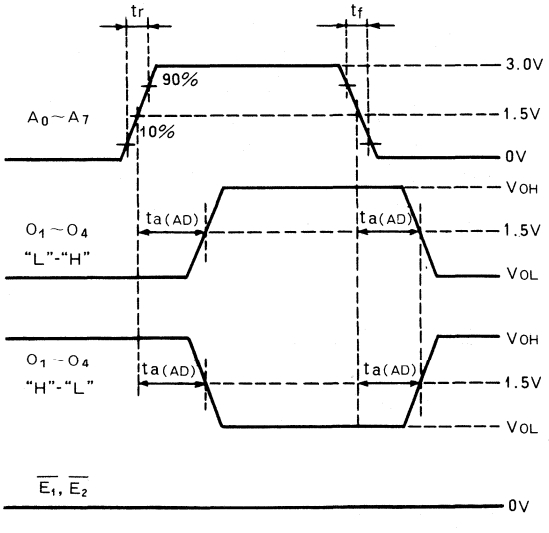
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ(Note 1)	Max	
V_{OL}	Low-level output voltage	I_{OL} = 16mA		0.3	0.45	V
I_{OH}	High-level output current	V_{OH} = 5.25V			100	μA
I_{IL}	Low-level input current	V_I = 0.4V			−1.6	mA
I_{IH}	High-level input current	V_I = 2.4V			40	μA
		V_I = 4.5V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

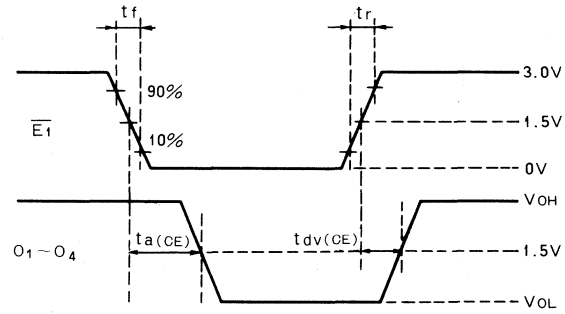
Note 1 : Typical values are at V_{CC} = 5V, Ta = 25°C

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			60	ns
$t_{a(CE)}$	Chip enable access time				35	ns
$t_{dv(CE)}$	Data valid time with respect to chip enable				35	ns

Timing Diagrams

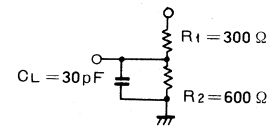




Note 2 : Rise time $t_r \leqq 5ns$; fall time $t_f \leqq 5ns$

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.



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

PROGRAMMING OPERATION

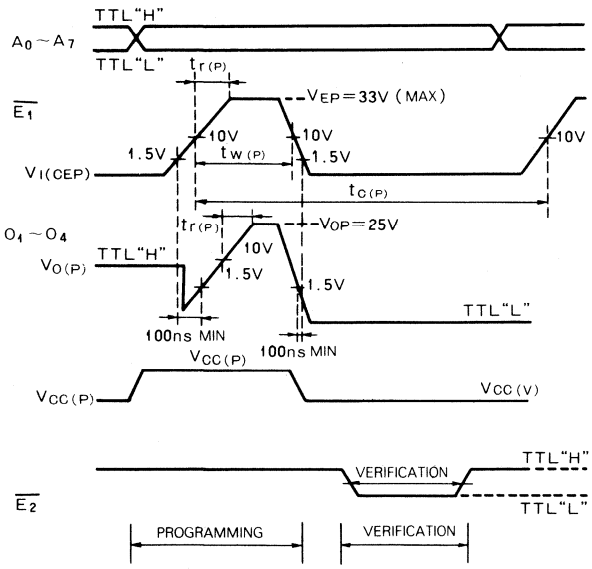
Recommended Operating Conditions

Symbol	Test conditions	Limits			Unit
		Min	Nom	Max	
$V_{I(CEP)}$	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

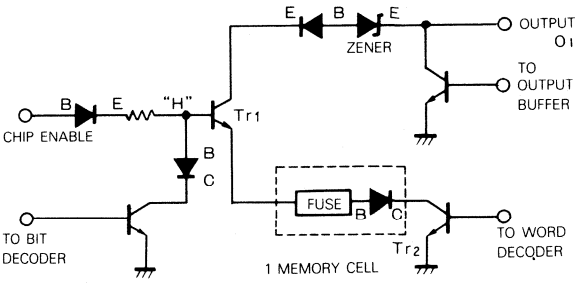
Timing Requirements

Symbol	Test conditions	Limits			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 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

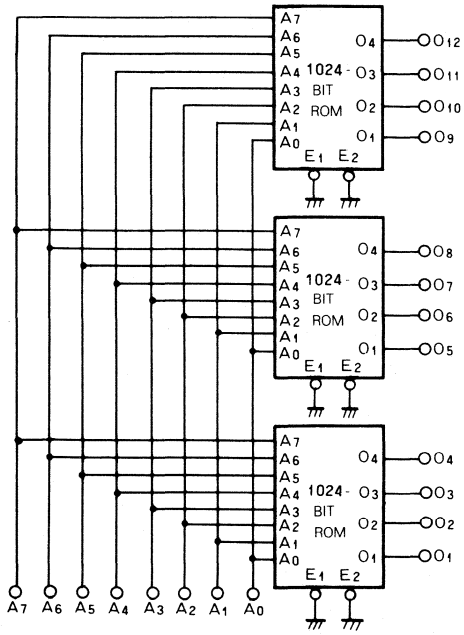
The chip enable inputs $\overline{E_1}$ and $\overline{E_2}$ are used for activating or inhibiting output O_1~O_4. $\overline{E_1}$ and $\overline{E_2}$ are NORed. Output is inhibited when any of the inputs are high-logic-level. Chip enable inputs $\overline{E_1}$ and $\overline{E_2}$ allow easy memory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

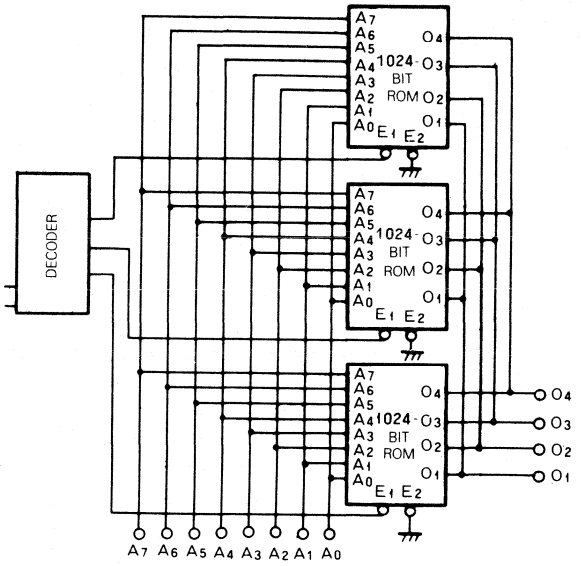


2. Expanding the Number of Words in Memory

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

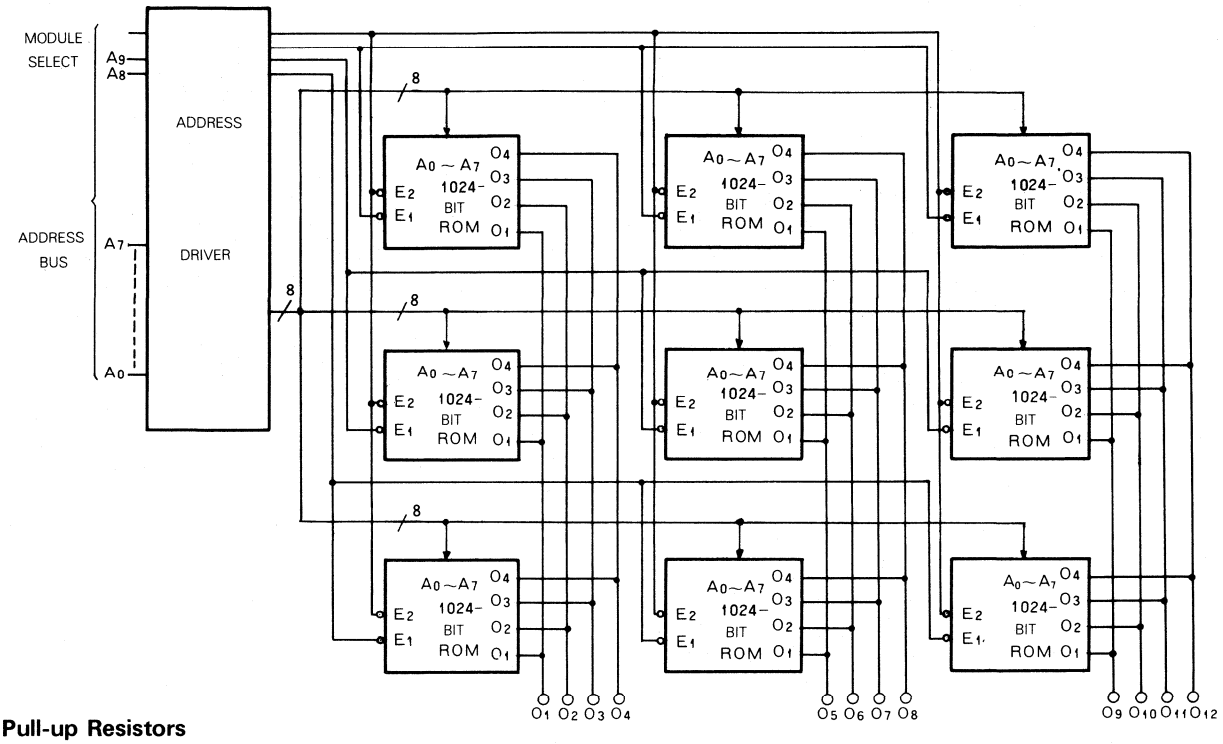
For example, using nine 1024-bit ROMs, each organized as 256 words of 4 bits, and by combining procedures 1 and 2 above, the number of words in memory along with the number of bits in a word, can be expanded as described below:

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 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:

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

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

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

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,

$$R_{L(min)} < R_L < R_{L(max)} \quad \cdots\cdots (3)$$

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

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

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

MITSUBISHI ELECTRIC

M54730K, P, S

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

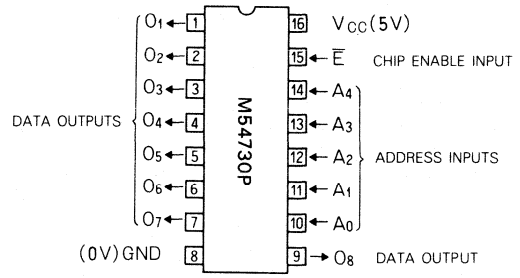
DESCRIPTION

The memory cells of the M54730K, P, 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 16K1 (M54730K)
16P1 (M54730P)
16S1 (M54730S)

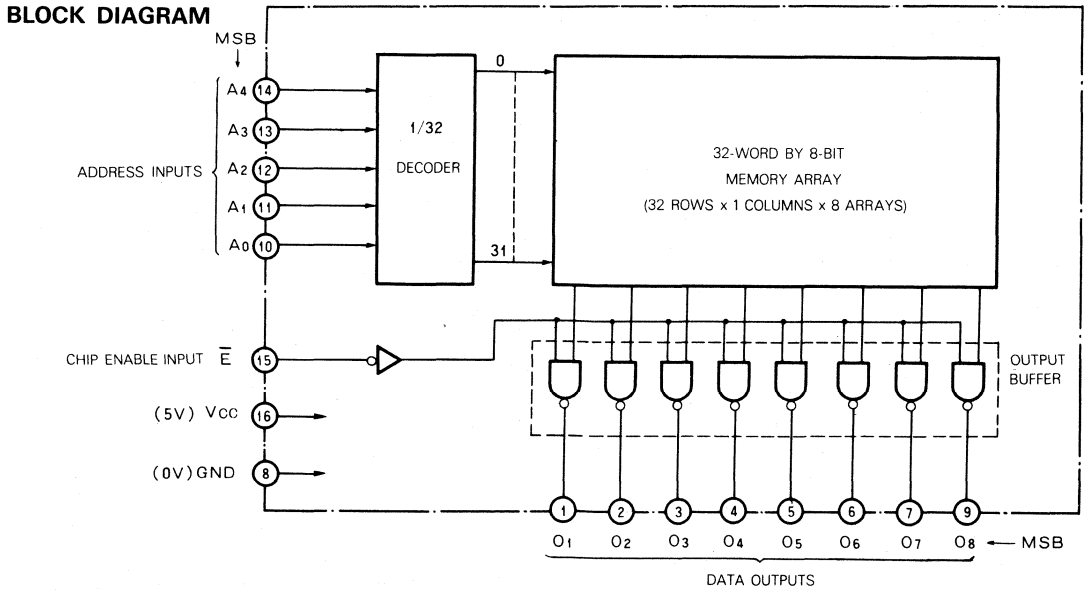
APPLICATION

- Programmable memory for the M5L8080A 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 $0_1 \sim 0_8$. 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



M54730K, P, S

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		−55~125	°C
V_O	Output apply voltage	In case of programming	27	V
$t_{w(P)}/t_{c(P)}$	Duty cycle		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)

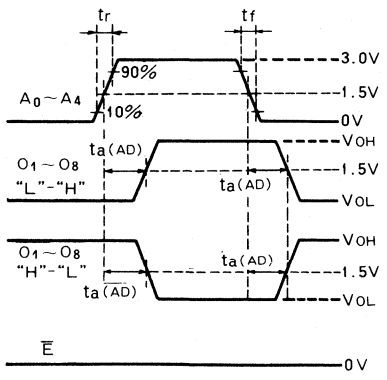
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ(Note 1)	Max	
V_{OL}	Low-level output voltage	I_{OL} = 16mA		0.3	0.45	V
I_{OH}	High-level output current	V_{OH} = 5.25V			100	μA
I_{IL}	Low-level input current	V_I = 0.4V			−1.6	mA
I_{IH}	High-level input current	V_I = 2.4V			40	μA
		V_I = 4.5V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

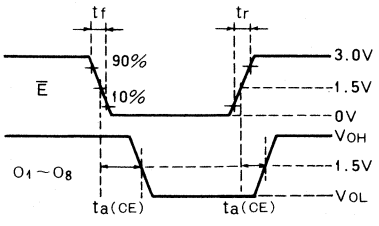
Note 1 : Typical values are at V_{CC} = 5V, Ta = 25°C.

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			50	ns
$t_{a(CE)}$	Chip enable access time				30	ns
$t_{dv(CE)}$	Data valid time with respect to chip enable				30	ns

Timing Diagrams

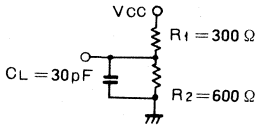




Note 2 : Rise time $t_r \leqq 5ns$; fall time $t_f \leqq 5ns$

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.



PROGRAMMING OPERATION

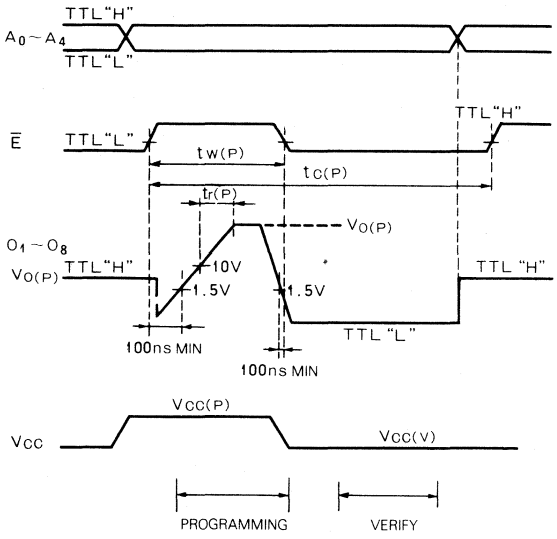
Recommended Operating Conditions

Symbol	Test conditions	Limits			Unit
		Min	Nom	Max	
$V_{I(CEP)}$	Chip enable program input voltage	29		33	V
$V_{O(P)}$	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

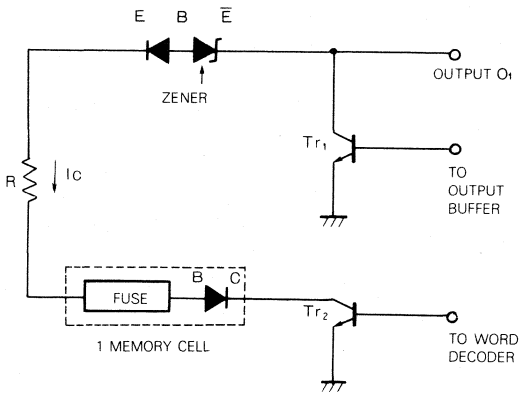
Timing Requirements

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
$t_{w(P)/to(P)}$	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

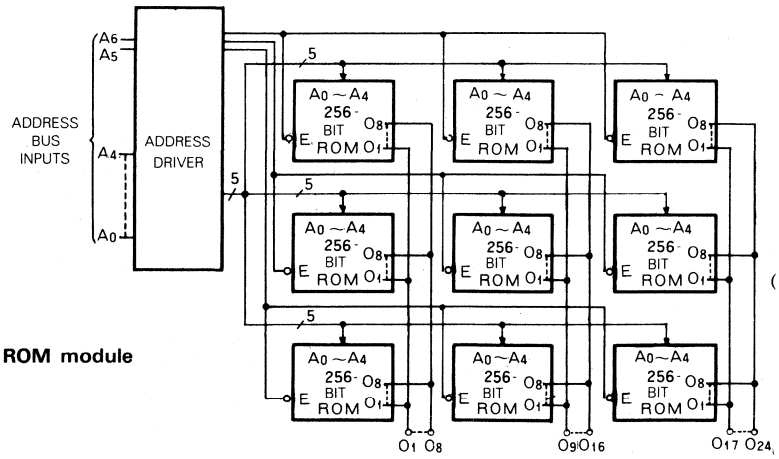
For example, using three 256-bit ROMs, each organized as 32 words of 8 bits, the number of words in memory can be expanded as described below:

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:

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

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

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

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

$$R_{L(min)} < R_L < R_{L(max)} \quad \cdots\cdots (3)$$

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

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

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

32 768-BIT (4096-WORD BY 8-BIT) MASK-PROGRAMMABLE ROM

DESCRIPTION

The M58333-XXXP is a 32768-bit static MOS mask-programmable read-only memory organized as 4096 words of 8 bits. It is fabricated in a 24-pin DIL package using N-channel aluminum-gate MOS technology. The inputs and outputs are TTL-compatible.

The XXX in the type code is a 3-digit decimal number which is assigned by Mitsubishi to identify the customer's specifications to which the ROM has been programmed.

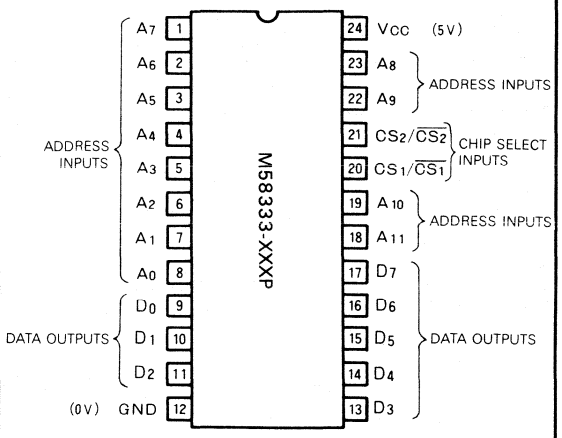
FEATURES

- Address access time: 650ns (max)
- 8-bit parallel output
- Easy memory expansion using two chip select inputs (CS_1 and CS_2)
- OR-tie capability by holding the output in the floating (high-impedance) state when not active
- Active logic level of CS_1 and CS_2 can be programmed at the time of ROM masking
- All inputs and outputs are TTL-compatible
- Provides an internal input protection circuit for all inputs

APPLICATION

- Microcomputer memories

PIN CONFIGURATION (TOP VIEW)

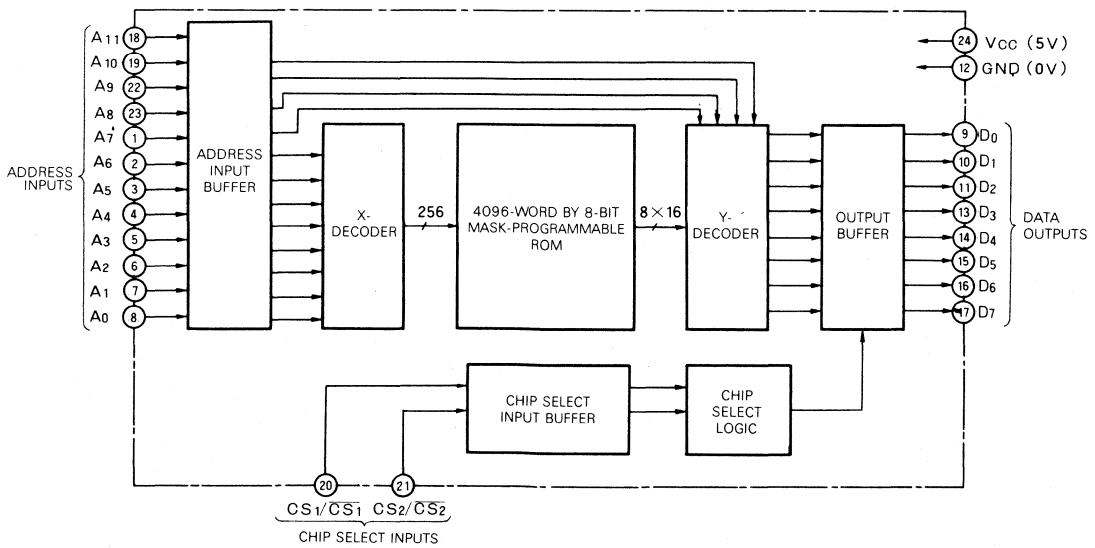


Outline 24P1

FUNCTION

Address inputs $A_0 \sim A_{11}$ are decoded to select one of the 4096 words, and the contents of that address are read out to data outputs $D_0 \sim D_7$. Chip selects (CS_1 and CS_2) are used to expand memory using two or more M58333-XXXP ROMs. The contents of the ROM can be read only when CS_1 and CS_2 are at specific input levels. Otherwise, data outputs $D_0 \sim D_7$ are held in the floating (high-impedance) state. The active logic level of CS_1 and CS_2 can be programmed at the time of manufacturing by the ROM mask.

BLOCK DIAGRAM



32 768-BIT(4096-WORD BY 8-BIT)MASK-PROGRAMMABLE ROM

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND (at V_{CC}=5V for V_I and V_O)	−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		−40 ~ 125	°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			0		V
V_{IH}	High-level input voltage	2		V_{CC}+1	V
V_{IL}	Low-level input voltage	−0.5		0.8	V

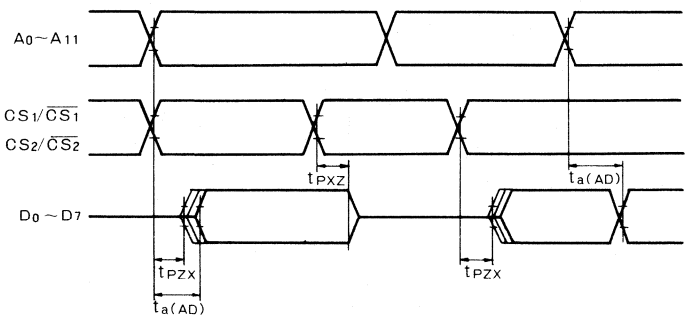
ELECTRICAL CHARACTERISTICS (Ta=0 ~ 70°C, V_{CC}=5V ± 5%, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	I_{OH} = −400 μA	2.2			V
V_{OL}	Low-level output voltage	I_{OL} = 2 mA			0.45	V
I_I	Input current	V_I=0 ~ V_{CC}	−10		10	μA
I_{OZ}	Off-state output current	V_I=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, 25mV$_{rms}$, Ta=25°C			10	pF
Co	Output capacitance				15	pF

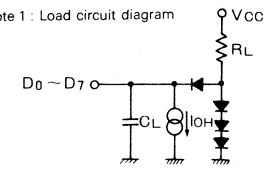
SWITCHING CHAPACTERISTICS (Ta=0 ~ 70°C, V_{CC}=5V ± 5%, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{a(AD)}$	Address access time	C_L=100pF, R_L=2.1kΩ (Note 1)		400	650	ns
t_{PZX}	Chip select propagation time				150	ns
t_{PXZ}	Chip non select propagation time		0		150	ns

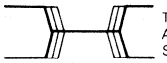
TIMING DIAGRAM



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 rise time t_r: ····· 20ns
Input pulse fall time t_f: ····· 20ns
Reference voltage for switchig characteristic measurement:

Input	V_{IH}:	2V
	V_{IL}:	0.8V
Output	V_{OH}:	2V
	V_{OL}:	0.8V

MITSUBISHI ELECTRIC

65 536-BIT(8192-WORD BY 8-BIT) MASK-PROGRAMMABLE ROM

DESCRIPTION

The M58334-XXXP is a 65 536-bit static MOS mask-programmable read-only memory organized as 8192 words of 8 bits. It is fabricated using N-channel aluminum-gate ED-MOS technology. It is operated with a single 5V power supply, and all inputs and outputs are TTL-compatible. It is designed for program storage with an M5L 8085A 8-bit parallel CPU. Programming is performed by Mitsubishi in accordance with the customer's specifications by changing a single mask during the manufacturing cycle.

The XXX in the type code is a 3-digit decimal number which is assigned by Mitsubishi to identify the customer's specifications to which the ROM has been programmed.

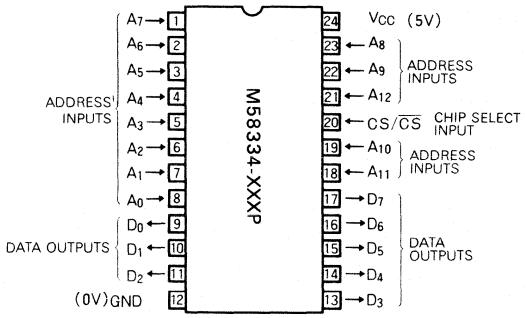
FEATURES

- Single 5V power supply
- Address access time: 650ns (max)
- Easy memory expansion using programmable chip select input
- Capable of OR-tie by holding the outputs in the floating (high-impedance) state
- All inputs and outputs are TTL-compatible
- Provides an internal input protection circuit for all inputs

APPLICATION

- Large-capacity memories for microcomputers

PIN CONFIGURATION (TOP VIEW)



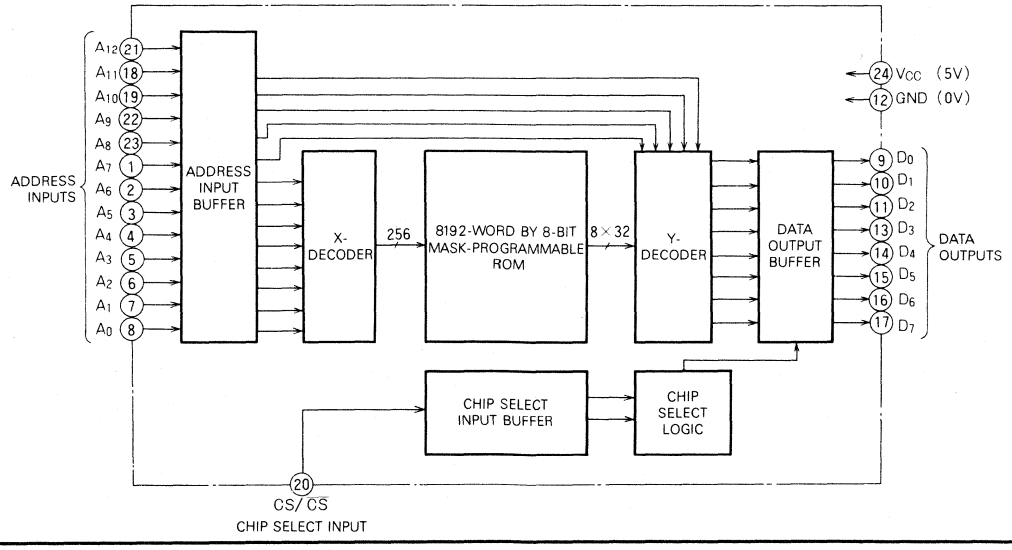
Outline 24P1

FUNCTION

Address inputs $A_0 \sim A_{12}$ are decoded to select one of the 8192 words, and the contents of that address are read out to data outputs $D_0 \sim D_7$.

The address input A_0 is the low-order bit, and A_{12} is the high-order bit. The chip select input function, whether to be active in high or low, is set by programming during the masking process. The contents of the ROM can be read out only while the chip select input is active. Otherwise, data outputs $D_0 \sim D_7$ are held in the floating (high-impedance) state.

BLOCK DIAGRAM



65 536-BIT (8192-WORD BY 8-BIT) MASK-PROGRAMMABLE ROM

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
Pd	Power dissipation		1000	mW
Topr	Operating free-air temperature range		0～70	℃
Tstg	Storage temperature range		−40～125	℃

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			0		V
V_{IH}	High-level input voltage	2		$V_{CC}+1$	V
V_{IL}	Low-level input voltage	−0.5		0.8	V

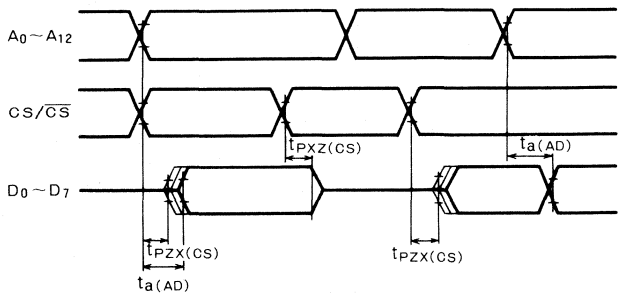
ELECTRICAL CHARACTERISTICS (Ta = 0 ～70℃, V_{CC} = 5 V ± 5 %, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	I_{OH} = −400μA	2.2			V
V_{OL}	Low-level output voltage	I_{OL} = 2 mA			0.45	V
I_{CC}	Supply current from V_{CC}	Output open		70	120	mA
I_I	Input leakage current	V_I = 0 V～V_{CC}	−10		10	μA
$I_{\overline{OZ}}$	Off-state output current	Floating state, V_I = 0.45V～V_{CC}	−20		10	μA
C_I	Input capacitance	0V except test terminal		4	10	pF
$C_{\overline{O}}$	Output capacitance	1MHz, Ta = 25℃		8	15	pF

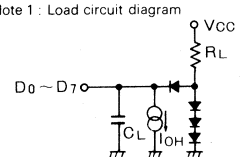
SWITCHING CHARACTERISTICS (Ta = 0 ～70℃, V_{CC} = 5 V ± 5 %, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
ta(AD)	Address access time	C_L = 100pF, R_L = 2.1KΩ (Note 1)		450	650	ns
$t_{PXZ(CS)}$	Chip non select propagation time				150	ns
$t_{PZX(CS)}$	Chip select propagation time		0		150	ns

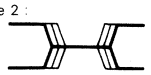
TIMING DIAGRAM



Note 1 : Load circuit diagram



Note 2 :



THE CENTER LINE INDICATES A FLOATING (HIGH-IMPEDANCE) STATE.

Input pulse level ..... 0.45～2.4 V
Input pulse rise time t_r 20ns
Input pulse fall time t_f 20ns
Reference voltage for switching characteristics measurement:
Input V_{IH} : 2V
V_{IL} : 0.8V
Output V_{OH} : 2V
V_{OL} : 0.8V

M58730-XXXS

Alternative Designation 8308

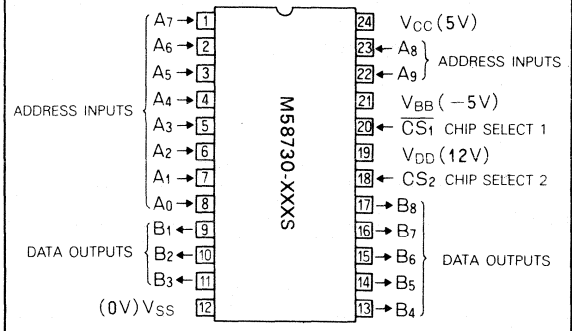
8192-BIT (1024-WORD BY 8-BIT) MASK-PROGRAMMABLE ROM

DESCRIPTION

The M58730-XXXS is an 8192-bit static MOS mask-programmable read-only memory organized as 1024 words of 8 bits. It is fabricated using N-channel silicon-gate MOS technology, and is designed for fixed-memory applications such as program storage with an M5L8080A 8-bit parallel CPU. The inputs and outputs are TTL-compatible. Programming is performed by Mitsubishi in accordance with the customer's specifications by changing a single mask during the manufacturing cycle.

The XXX in the type code stands for a 3-digit decimal number which is assigned by Mitsubishi to identify the customer's specifications to which the ROM has been programmed.

PIN CONFIGURATION (TOP VIEW)



Outline 24S1

FEATURES

- Fast access time: 850ns (max.)
- Two chip select inputs ($\overline{CS_1}$, CS_2) for easy memory expansion
- Three-state output; OR-tie capability
- Inputs and outputs are TTL-compatible
- Input protection circuits for all inputs
- Pins compatible with Intel's 8308

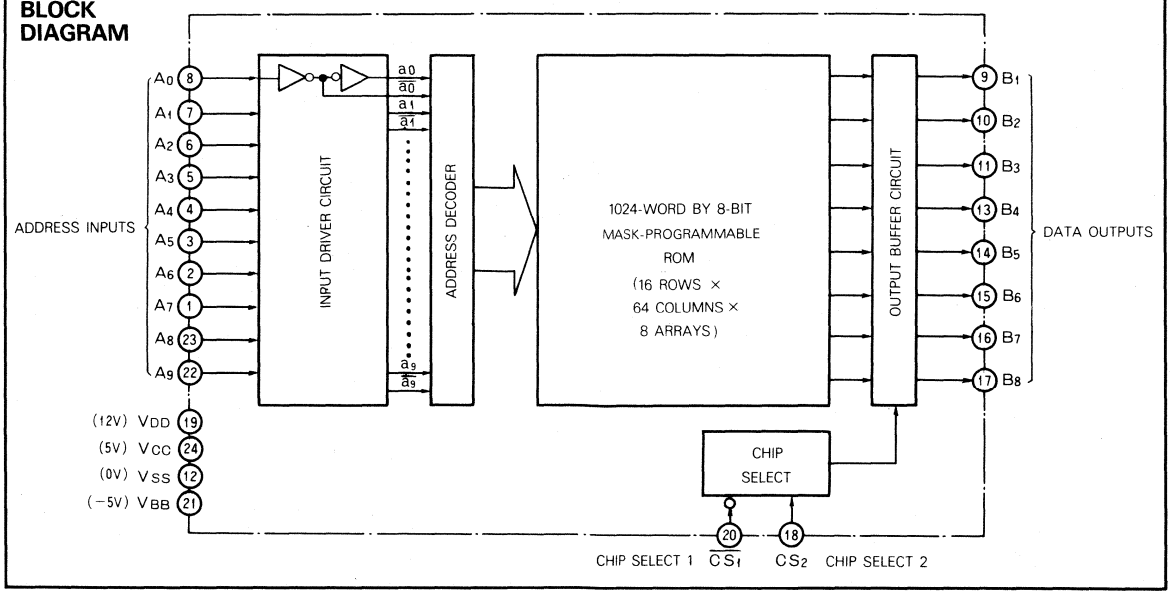
APPLICATION

- Microcomputer memories

FUNCTION

Address inputs A_0~A_9 are decoded to select one of the 1024 words, and the contents of that address are read out to data outputs B_1~B_8. Chip select 1 ($\overline{CS_1}$) and chip select 2 (CS_2) are used to connect two or more M58730-XXXS ROMs. When $\overline{CS_1}$ is high or CS_2 is low, all outputs are disabled and will assume a floating (high-impedance) state.

BLOCK DIAGRAM



5

8192-BIT (1024-WORD BY 8-BIT) MASK-PROGRAMMABLE ROM

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{BB}	−0.3~20	V
V_{CC}	Supply voltage		−0.3~20	V
V_{SS}	Supply voltage		−0.3~20	V
V_I	Input voltage		−0.3~20	V
Pd	Power dissipation	Ta=25°C	1.0	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_{DD}	Supply voltage	11.4	12	12.6	V
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{SS}	Supply voltage		0		V
V_{BB}	Supply voltage	−4.75	−5	−5.25	V
V_{IH}	High-level input voltage	3.3		$V_{CC}+1$	V
V_{IL}	Low-level input voltage	$V_{SS}-1$		0.8	V

ELECTRICAL CHARACTERISTICS (Ta=0~70°C, V_{DD}=12V±5%, V_{CC}=5V±5%, V_{SS}=0V, V_{BB}=−5V±5%, unless otherwise noted).

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	I_{OH}=−100μA	$V_{CC}-1$			V
V_{OL}	Low-level output voltage	I_{OL}=1.9mA			0.45	V
I_I	Input current	V_I=0V~V_{CC}			±10	μA
I_{OZ}	Off-state output current	V_O=0V~V_{CC} ($\overline{CS_1}$ and CS_2 are in a floating condition. See Timing Diagram).			10 −100	μA
I_{DD}	V_{DD} supply current				60	mA
I_{CC}	V_{CC} supply current	Output open			100	μA
I_{BB}	V_{BB} supply current			−0.01	−1	mA
C_i	Input capacitance	Ta=25°C, V_I=0V, 1MHz, 25mVrms V_{DD}=V_{CC}=V_{SS}=0V (Note 2)			10	pF
C_o	Output capacitance	Ta=25°C, V_I=0V, 1MHz, 25mVrms V_{DD}=V_{CC}=V_{SS}=0V (Note 2)			10	pF

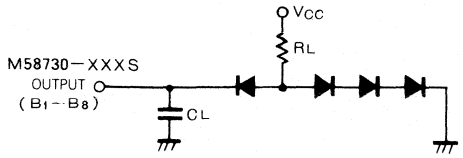
Note 1 : The current flowing into an IC is positive; out is negative. The maximum and minimum are defined by absolute values.

2 : All terminals other than the test terminal are connected to V_{SS} during measurement of input and output capacitance.

SWITCHING CHARACTERISTICS (T_a = 0 ~ 70°C, V_{DD} = 12V ± 5%, V_{CC} = 5V ± 5%, V_{SS} = 0V, V_{BB} = −5V ± 5%, unless otherwise noted)

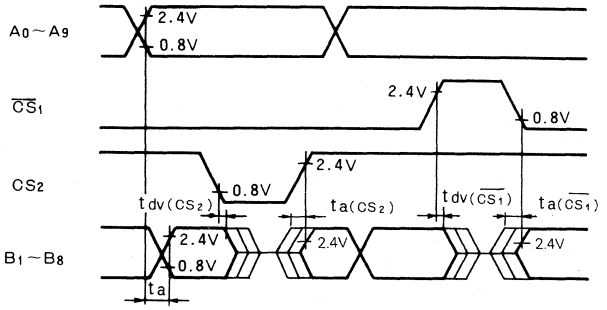
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
t_a	Access time	C_L = 100pF, R_L = 2.1kΩ (Note 3)		400	850	ns
$t_{a(\overline{CS_1})}$	Chip select access time			100	300	ns
$t_{a(CS_2)}$	Chip select access time			100	300	ns
$t_{dv(\overline{CS_1})}$	Data valid time with respect to $\overline{CS_1}$			100	300	ns
$t_{dv(CS_2)}$	Data valid time with respect to CS_2			100	300	ns

Note 3 : Load circuit diagram:

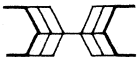


5

TIMING DIAGRAM



Chip select 1 $\overline{CS_1}$	Chip select 2 CS_2	Data output $B_1 \sim B_8$
L	L	Z
H	L	Z
L	H	O
H	H	Z

Note 1 :  THE CENTER LINE INDICATES A FLOATING (HIGH-IMPEDANCE) STATE

2 : H indicates high-level inputs; L indicates low-level inputs.

3 : Z indicates floating (off) state.

4 : O indicates that outputs are enabled.

5 : Rise time $t_r \leqq 20$ns,
Fall time $t_f \leqq 20$ns.

M58730-001S

8192-BIT (1024-WORD BY 8-BIT) MASK-PROGRAMMED ROM SUBROUTINE 1 INTEGER ARITHMETIC OPERATIONS

DESCRIPTION

- The M58730-001S is an M58730-XXXS that has been developed for use with an M5L8080A CPU.
- It includes 18 subroutines for an M5L8080A 8-bit parallel CPU.
- It can perform integer arithmetic operations, logical operations and shift operations with 16-bit or 32-bit data.

UNIT OF INFORMATION

The basic unit of an M5L8080A is 8 bits, but with subroutines it has two operand lengths.

- Single word length:

An operand consisting of 2 bytes (16 bits). In binary form it is capable of expressing numbers from -2^{15} to $2^{15}-1$.

- Double word length:

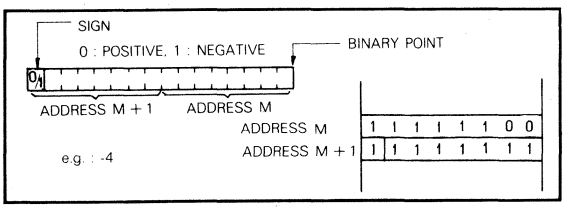
An operand consisting of 4 bytes (32 bits). In decimal form it is equivalent to 7 decimal digits. In binary form it is capable of expressing numbers from -2^{31} to $2^{31}-1$.

NUMERICAL EXPRESSIONS

1. Binary Numbers

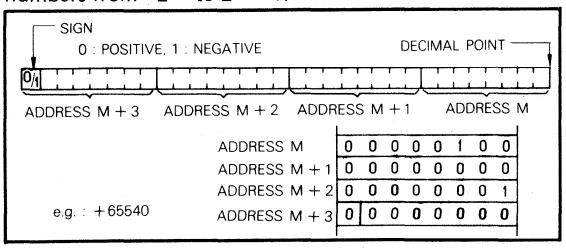
1. Single Word Length (2 Bytes)

This binary number consists of 16 bits. Negative numbers are in 2's complement form. It is capable of expressing numbers from -2^{15} to $2^{15}-1$.



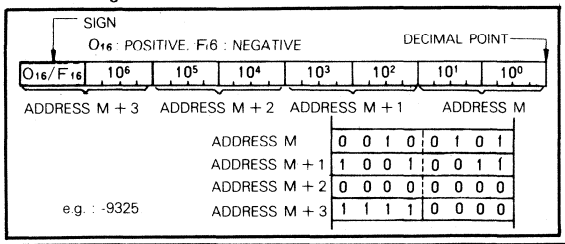
2. Double Word Length (4 bytes)

This binary number consists of 32 bits. Negative numbers are in 2's complement form. It is capable of expressing numbers from -2^{31} to $2^{31}-1$.

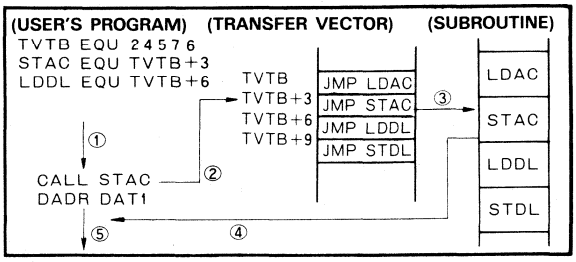


2. Double Word Length Decimal Numbers

This decimal number consists of 32 bits. The numerical portion is seven digits and the sign is the most significant digit. It has a range of -10^7+1 to 10^7-1.



SUBROUTINE REFERENCE



Note : The processing order is ①, ②, ③, ④, ⑤. A transfer vector is used to set the entry address of each subroutine.

SUBROUTINE FUNCTIONS

- Load pseudo accumulator

The pseudo accumulator is loaded with the specified single word (2 bytes) or double word (4 bytes) data.

- Store pseudo accumulator

The contents of the pseudo accumulator, single word (2 bytes) or double word (4 bytes) data, are stored in the address location specified.

- Shift pseudo accumulator

The contents of the pseudo accumulator, 32 bits (2 words) of data, are shifted right or left n positions.

- Arithmetic right shift of pseudo accumulator

The contents of the pseudo accumulator, 32 bits (2 words) of data, are arithmetically shifted right n positions.

- Logical operations

The specified single word (2 bytes) data is logically inclusive ORed, ANDed or exclusive ORed to the contents of the pseudo accumulator, and the result retained in the pseudo accumulator.

- Binary integer add or subtract

The specified single word (2 bytes) or double word (4 bytes) binary data is binarily added to or subtracted from the contents of the pseudo accumulator, and the result is retained in the pseudo accumulator.

- Decimal integer add or subtract

The specified double word (4 bytes) decimal data is decimally added to or subtracted from the contents of the pseudo accumulator, and the result is retained in the pseudo accumulator.

- Binary integer multiply

The single word (2 bytes) data in the pseudo accumulator is multiplied by a specified single word (2 bytes) data, and the result is retained in the pseudo accumulator.

- Binary integer divide

The double word (4 bytes) data in the pseudo accumulator is divided by a specified single word (2 bytes) data, and the result is retained in the pseudo accumulator.

RESERVED MEMORY LOCATIONS

Memory locations 6000_{16} to $63FF_{16}$ are reserved by ROM. In addition, a 50-byte RAM region, locations $3FCE_{16}$ to $3FFF_{16}$, is reserved for executing the ROM programs.

M58731-XXXP, S

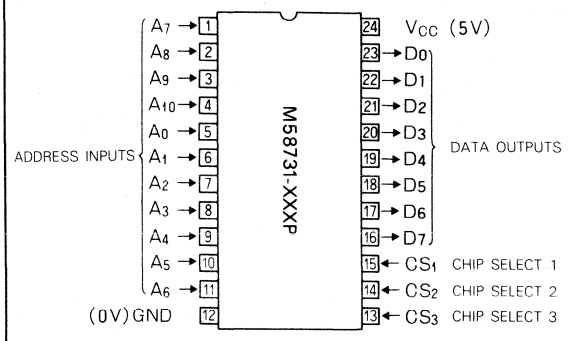
Alternative Designation 8316A

16 384-BIT (2048-WORD BY 8-BIT) MASK-PROGRAMMABLE ROM

DESCRIPTION

The M58731-XXXP, S are 16 384-bit parallel output, static read-only memories organized as 2048 words of 8 bits. They are fabricated using N-channel silicon-gate ED-MOS technology. They have a single supply voltage. The inputs and outputs interface with TTL circuits without additional circuits. The M58731-XXXP, S are designed for high-density fixed-memory applications such as program storage for an M5L 8080A 8-bit parallel CPU. Programming is performed by Mitsubishi in accordance with the customer's specifications by changing a single mask during the manufacturing cycle.

PIN CONFIGURATION (TOP VIEW)



Outline 24P1 (M58731-XXXP)
24S1 (M58731-XXXS)

FEATURES

- 2048-word by 8-bit organization
- Single 5V power supply
- Low power dissipation: 31.4μW/bit (max.)
- Read access time: 850ns (max.)
- Three programmable chip select inputs (CS_1, CS_2, CS_3) for easy memory expansion
- Three-state output for OR-ties
- All inputs and outputs are TTL-compatible
- Input protection circuits at all inputs
- Electrical characteristics and pins are compatible with Intel's 8316A

APPLICATION

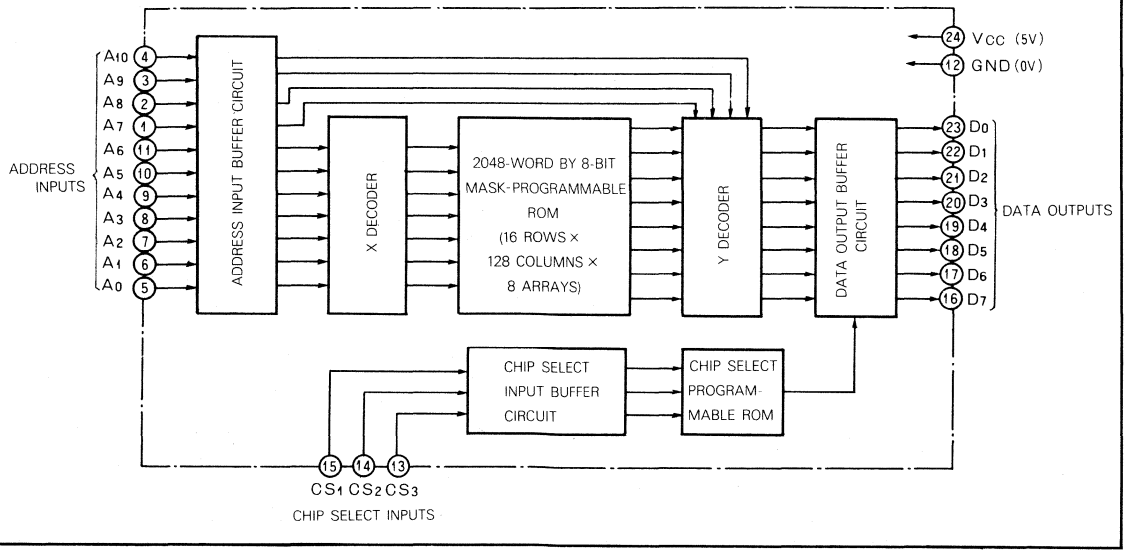
- High-density microcomputer memories

FUNCTION

When any of the 2048 addresses are selected by positive-logic input signals (A_0~A_{10}), the contents of that address in the ROM are read out to the data outputs (D_0~D_7). A_0 is the least-significant bit and A_{10} is the most-significant bit of the address. The three chip select inputs are programmable during the masking process, and any combination of active high-level and active low-level may be used for chip selection. When a chip is selected, the contents of the ROM are read out; and under other conditions, the data outputs (D_0~D_7) are in the floating (high-impedance) state.

The XXX in the type code stands for a 3-digit decimal number which is assigned by Mitsubishi to identify the customer's specifications to which the ROM has been programmed.

BLOCK DIAGRAM



5

M58731-XXXP, S

Alternative Designation 8316A

16 384-BIT (2048-WORD BY 8-BIT) MASK-PROGRAMMABLE ROM

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	M58731-XXXP	Ta=25°C	1000	mW
		M58731-XXXS		1500	mW
Topr	Operating free-air temperature range			0～70	°C
Tstg	Storage temperature range	M58731-XXXP		−40～125	°C
		M58731-XXXS		−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			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±5%, unless otherwise noted)

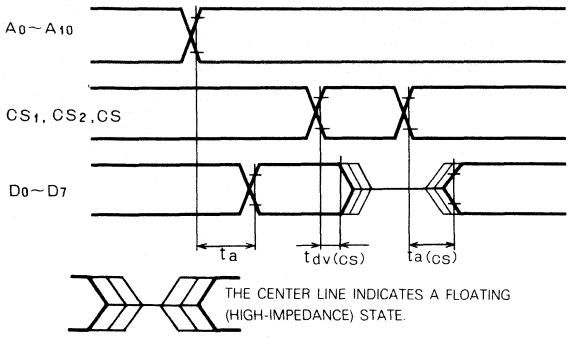
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	I_{OH}=−100μA	2.2			V
V_{OL}	Low-level output voltage	I_{OL}=2mA			0.45	V
I_{CC}	Supply current from V_{CC}	All inputs=5.25V, output open		40	98	mA
I_I	Input current	V_I=0V～V_{CC}			10	μA
I_{OZ}	Off-state input current	Floating state, V_I=0.45V～V_{CC}	−20		10	μA
C_i	Input capacitance	0V except test terminal, 1MHz, Ta=25°C		4	10	pF
C_O	Output capacitance			8	15	pF

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

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

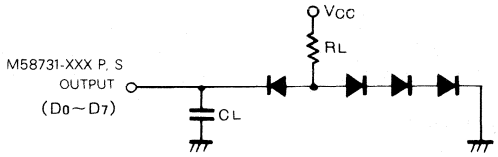
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
t_a	Access time	$C_L = 100pF$ $R_L = 2.1k\Omega$ (Note 2)		400	850	ns
$t_{a(CS)}$	Chip select access time				300	ns
$t_{dv(CS)}$	Data valid time with respect to chip select		0		300	ns

TIMING DIAGRAM



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
Referance voltage at timing measurement	
Input	0.8~2.00V
Output	0.8~2V

Note 2 : Load circuit diagram :



5

M58731-001S

16 384-BIT (2048-WORD BY 8-BIT) MASK-PROGRAMMED ROM MELPS 8 BOM-B BASIC OPERATING MONITOR

DESCRIPTION

The M58731-001S is an M58731-XXXS that has been developed for use with an M5L8080A CPU. It contains the BOM-B basic operating monitor for an M5L8080A CPU. BOM-B is a monitor program that controls the execution and debugging of user's programs and is contained in 2K bytes of memory.

FEATURES

- A standard mask ROM useful for microcomputer control and program debugging
- Three macroinstructions and nine monitor commands
- User's monitor commands are easily added
- The BOM-B program cannot be destroyed by a user's program

FUNCTION

The BOM-B has 9 monitor commands and 3 macroinstructions as shown in Table 1. They are used for the following functions:

1. Controlling program execution
2. Loading programs
3. Punching memory
4. Debugging programs
5. Controlling input and output

Start of Execution of BOM-B Program

The execution is started at address 6800_{16}. The following message is printed out and then a monitor command can be typed in: **MELPS 8 BOM-B A01**

//

Conditions for Hardware

1. Reserved Memory Locations
   Memory locations 6800_{16} to $6FFF_{16}$ are reserved. In addition a 78-byte RAM region, locations $3FCE_{16}$ to $3FFF_{16}$, is reserved for executing the ROM programs.
2. Input/Output Device Number
   PTR, for keyboard input $7B_{16}$ (IN 7B#)
   PTP, for print output $7B_{16}$ (OUT 7B#)
   Status input $3B_{16}$ (IN 3B#)
   Where the status bits are defined as follows:

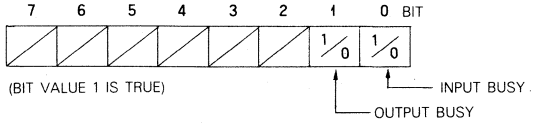


Table 1 A list of the 9 monitor commands and the 3 macroinstructions for BOM-B

Names of monitor commands or macroinstructions		Function	Monitor command input format or calling sequence	Parameter
Command	G	Go to program execution	//G para 1(4) ␣ [para 2(4)] CR LF	para 1(4): Start address para 2(4): Change start address
	R	Program restart	//R CR LF	—
	L	MELPS 8 binary loader	//L CR LF	—
	H	MELPS 8 hexadecimal loader	//H CR LF	—
	T	MELPS 8 binary punch text block of memory data	//T para 1(4), para 2(4) CR LF	para 1(4): First address para 2(4): End address
	E	MELPS 8 binary punch end block	//E [para 1(4)] CR LF	para 1(4): Start address
	P	Print hexadecimal text block of memory data	//P para 1(4), para 2(4) CR LF	para 1(4): First address para 2(4): End address
	S	Substitute memory	//S para 1(4) CR LF	para 1(4): Change address
	M	Print and modify register data	//M CR LF	—
Macroinstruction	EXIT	Exit the end of a program	CALL 6806#	
	PAUSE	Pause program execution	CALL 6803#	
	EXIO	Execution input/output control	CALL 6EBC# ---------- Execute EXIO macroinstruction. DADR DCB1 ---------- First address of data control block (DCB). DCB1 DEF IOD --- IOD for PTR is 52#, for PTP is 50#, for KEYBOARD is 4B#, for PRINT is 44#. DADR DA --- Set the first address of input/output data area. DADR DL --- Set the data length of input/output data. DA: Input/output data store memory (DL bytes)	

Note 1: Para n(m) : A hexadecimal number (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F) of the nth parameter in one command (of an operator's input or a monitor's print-out), which has a valid length of 1 to m. If the length exceeds m, the least-significant digits are valid.
2 : _____ (underline) : Indicates an input by an operator.
3 : [ ] : The parameter may be omitted.
4 : # : Indicates a hexadecimal number in assembler language.

MITSUBISHI ELECTRIC

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.

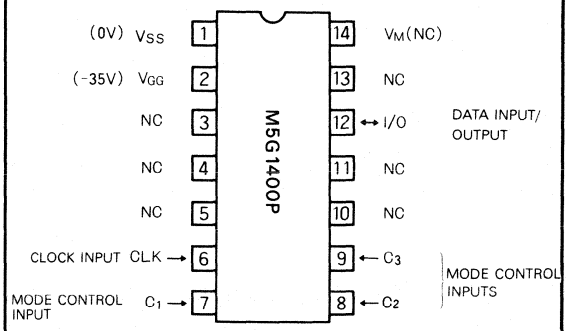
FEATURES

- 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: 10^9 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

PIN CONFIGURATION (TOP VIEW)



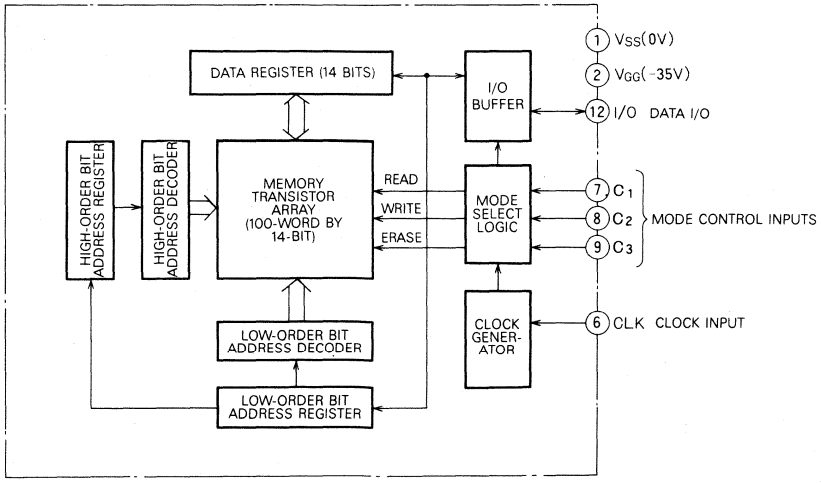
Outline 14P4

NC: NO CONNECTION

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.

BLOCK DIAGRAM



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
T_{stg}	Storage temperature range		−40～125	℃
T_{opr}	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}	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

ELECTRICAL CHARACTERISTICS (Ta=−10～70℃, V_{GG}=−35V±8%, 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		$V_{SS}-15$		$V_{SS}-8$	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 1 : 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			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	Rise time, fall time	tr, tf				1	μs
tsu(c−φ)	Control setup time before the fall of the clock pulse	tCS		0			ns
th(φ−c)	Control hold time after the rise of the clock pulse	tCH		0			ns

SWITCHING CHARACTERISTICS (Ta=−10～70℃, V_{GG}=−35V±8%, unless otherwise noted.)

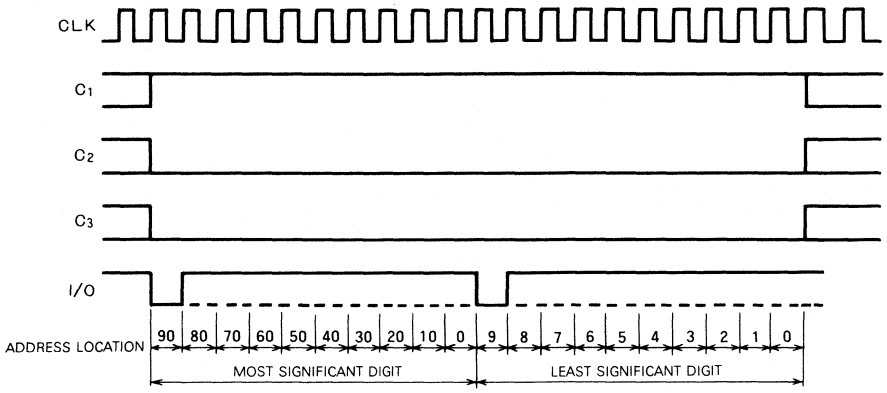
Symbol	Parameter	Alternative symbols	Test conditions	Limits			Unit
				Min	Typ	Max	
ta(c)	Read access time	tPW	C_L=100pF, $V_{OH}=V_{SS}-2V$, $V_{OL}=V_{SS}-8V$			20	μs
ts	Unpowered nonvolatile data retention time	TS	$N_{EW}=10^4$, tw(W)=20ms, tw(E)=20ms	10			Year
		TS	$N_{EW}=10^5$, tw(W)=20ms, tw(E)=20ms	1			Year
NEW	Number of erase/write cycles	NW		10^5			Times
NRA	Number of read access unrefreshed	NRA		10^9			Times
tdv	Data valid time	tPW				20	μs

5

1400-BIT (100-WORD BY 14-BIT) ELECTRICALLY ALTERABLE ROM

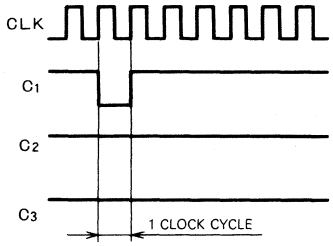
TIMING DIAGRAM

Accept Data Mode

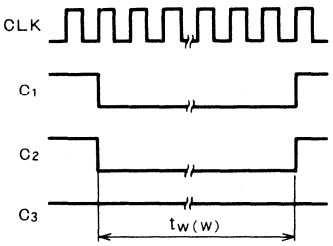


Note 2 : The address is designated by two one-of-ten-coded digits. The figure shows designation of the address 99.

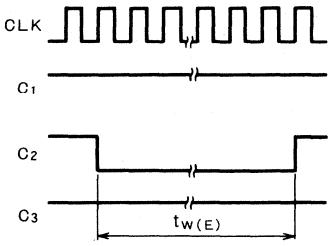
Read Mode



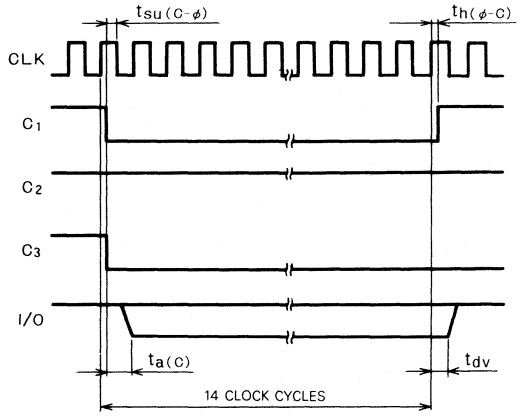
Write Mode



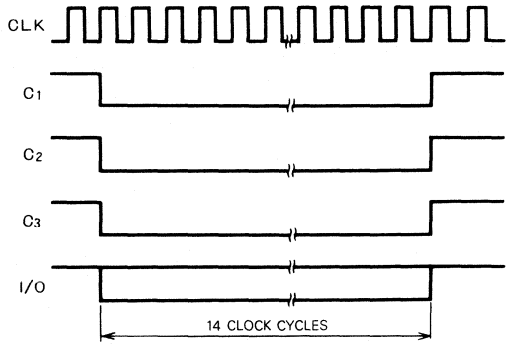
Erase Mode



Shift Data Output Mode



Accept Data Mode



MITSUBISHI ELECTRIC

M5L 2708K, S; K-65, S-65

8192-BIT (1024-WORD BY 8-BIT) ERASABLE AND ELECTRICALLY REPROGRAMMABLE ROM

DESCRIPTION

This is a family of FAMOS (floating-gate avalanche-injection MOS) ultraviolet-light erasable and electrically reprogrammable 8192-bit (1024-word by 8-bit) EPROMs. They incorporate N-channel silicon-gate MOS technology, are designed for microcomputer system applications, and have direct TTL-compatibility for all inputs and outputs, without extra interface circuits.

FEATURES

- Fast programming: 100s/8192 bits (typ)
- Access time:
  M5L 2708K, S: 450ns (max)
  M5L 270K-65, S-65: 650ns (max)
- Low power dissipation during programming
- No clocks required; the circuitry is entirely static
- Data inputs and outputs TTL-compatible during read and program modes
- Easy memory expansion by chip-select/write-enable ($\overline{CS}$/WE) input
- Typical power supply voltages: 12V, 5V, –5V
- For large volume production; pin compatible with the Mitsubishi M58730-XXXS mask-programmable ROM
- Interchangeable with Intel's 2708 in pin configuration and electrical characteristics

APPLICATION

- Computers and peripheral equipment

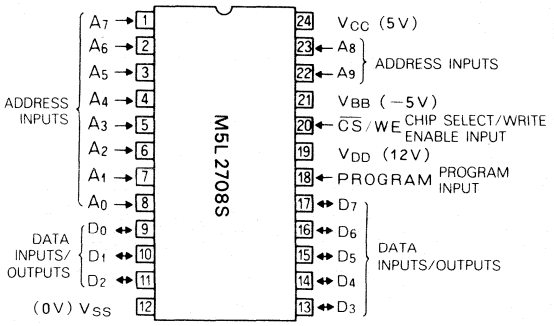
FUNCTION

Read—Set the $\overline{CS}$/WE terminal to the read mode (0~5V). Low-level input to $\overline{CS}$/WE and address signals to the address input ($A_0 \sim A_9$) make the data contents of the designated address location available at the data inputs/outputs ($D_0 \sim D_7$). When the $\overline{CS}$/WE signal is high, data inputs/outputs ($D_0 \sim D_7$) are in a floating state.

Write—Set the $\overline{CS}$/WE terminal to the write mode (12V). A program pulse will effect the write operation for the data at each address loaded via data inputs/outputs ($D_0 \sim D_7$). For details refer to the description of the programming mode.

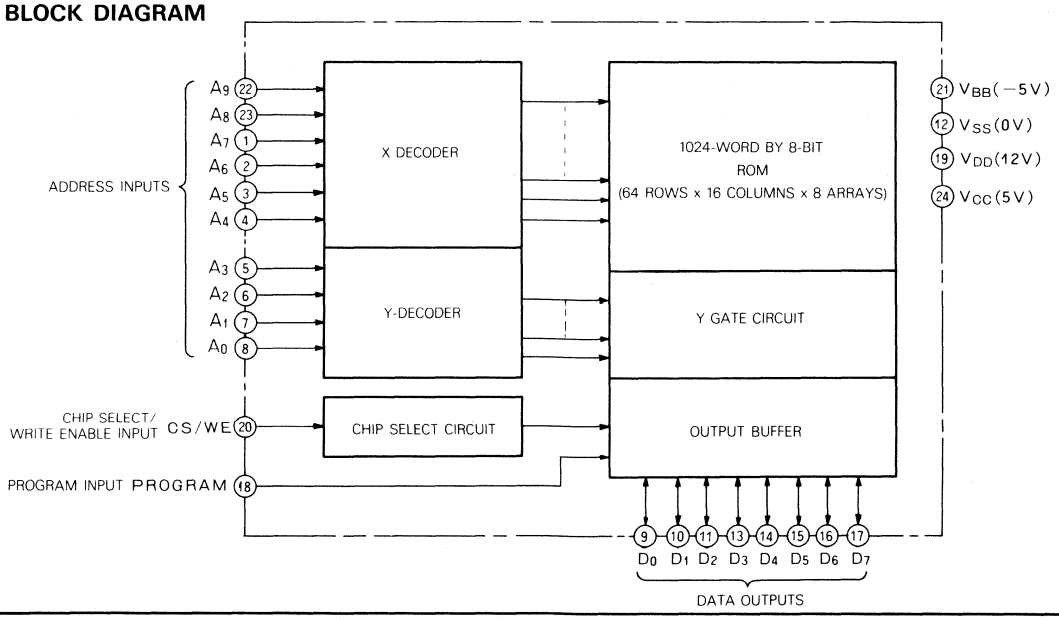
Erase—Erase is effected by exposure to ultraviolet light through the transparent window.

PIN CONFIGURATION (TOP VIEW)



Outline 24K10 (M5L 2708K)
24S10 (M5L 2708S)

BLOCK DIAGRAM



5

8192-BIT (1024-WORD BY 8-BIT) ERASABLE AND ELECTRICALLY REPROGRAMMABLE ROM

FUNCTIONAL OPERATIONS

Programming Procedure

These devices are in the '1' state (with high-level output) after erase, and go into the '0' state (with low-level output) after programming. All bits of the M5L 2708S, S-65 are initially in the '1' state, and must be programmed according to the following procedures.

The chip enters the program mode when 12V is supplied to the $\overline{CS}$/WE input (pin 20). Data to be programmed are presented, 8 bits in parallel, to the data inputs/outputs ($D_0 \sim D_7$) and the addresses are set up by the address inputs. After address and data set-up, one program pulse is applied to the program input (pin 18) for each address from 0 to 1023. This pass through all addresses, known as a program loop, must be repeated a number of times, N, which depends upon the width of the program pulse and must satisfy the condition $N \cdot t_{W(P)} \geqq 100ms$.

Erase Procedure

These devices can be erased by exposure to high-intensity short-wave ultraviolet light at a wavelength of 2537Å through the transparent lid provided. The required exposure is approximately $15Ws/cm^2$. If the energy of the lamp used is unknown, find the total time (t_E) required to erase all bits and use a short-wave ultraviolet-light exposure time of 3 to 5 times this value.

HANDLING PRECAUTIONS FOR FAMOS DEVICES

In addition to general handling precautions for MOS devices, the following points apply to FAMOS devices.

1. 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.
2. Before erasing, clean the surface of the transparent lid to remove completely oily impurities, which may impede irradiation and affect the erasing characteristics.
3 This ultraviolet-light erasable PROM is erasable by ultraviolet-light with wavelengths under 4000Å. For use involving long exposure to direct sunlight or to lamps radiating at these wavelengths, the transparent window should be covered with opaque tape.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{I1}	Input voltage, V_{DD} and $\overline{CS}/WE$ write mode	With respect to V_{BB}	−0.3～20	V
V_{I2}	Input voltage, V_{CC}, V_{SS}, address and data signal		−0.3～15	V
V_{I3}	Input voltage, program mode		−0.3～35	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
V_{DD}	Supply voltage	11.4	12	12.6	V
V_{BB}	Supply voltage	−4.75	−5	−5.25	V
V_{SS}	Supply voltage (GND)		0		V
V_{IL}	Low-level input voltage	V_{SS}		0.65	V
V_{IH}	High-level input voltage	3		$V_{CC}+1$	V

Electrical Characteristics

Ta=0～70℃, V_{CC}=5V±5%, V_{DD}=12V±5%, V_{BB}=−5V±5%, V_{SS}=0V, unless otherwise noted. Note 1)

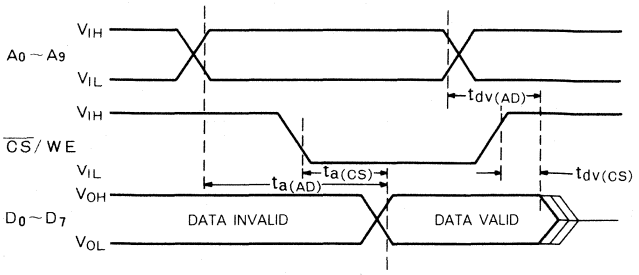
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
I_{IL}	Low-level input current, address, chip select input	V_I=5.25V			10	μA
I_{OZ}	Off-state (high-impedance-state) output current	V_O=5.25V, $V_{O(\overline{CS}/WE)}$=5V			10	μA
I_{DD}	Supply current from V_{DD}	Worst case,		50	65	mA
I_{CC}	Supply current from V_{CC}	all inputs high.		6	10	mA
I_{BB}	Supply current from V_{BB}	$V_{O(\overline{CS}/WE)}$=5V, Ta=0℃		30	45	mA
V_{OL}	Low-level output voltage	I_{OL}=1.6mA			0.45	V
V_{OH1}	High-level output voltage	I_{OH}=−100μA	3.7			V
V_{OH2}	High-level output voltage	I_{OH}=−1mA	2.4			V
Pd	Power dissipation	Ta=70℃			800	mW
C_i	Input capacitance	V_I=0V, f=1MHz		4	6	pF
C_o	Output capacitance			8	12	pF

Note 1 : Typical values are at Ta=25℃ and nominal supply voltage.

Switching Characteristics (Ta=0～70℃, V_{CC}=5V±5%, V_{DD}=12V±5%, V_{BB}=−5V±5%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{a(AD)}$	Address access time	M5L 2708K, S			280	450	ns
		M5L 2708K-65, S-65				650	ns
$t_{a(CS)}$	Chip select access time				60	120	ns
$t_{dv(CSLH)}$	Data valid time with respect to chip select low-to-high			0		120	ns
$t_{dv(AD)}$	Data valid time with respect to address			0			ns

Timing Diagram



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

Test Conditions for Switching Characteristics

Input voltage : V_{IL}=0.65V, V_{IH}=3V
Reference voltage at timing measurement : Input 0.8～2.8V
output 0.8～2.4V

8192-BIT (1024-WORD BY 8-BIT) ERASABLE AND ELECTRICALLY REPROGRAMMABLE ROM

PROGRAM OPERATION

Recommended Operating Conditions (T_a=25℃, V_{CC}=5V±5%, V_{DD}=12V±5%, V_{BB}=−5V±5%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
$V_{IL1(P)}$	Low-level input voltage, except program input	V_{SS}		0.65	V
$V_{IH1(P)}$	High-level input voltage, address, data input	3		V_{CC}+1	V
$V_{IH2(P)}$	High-level input voltage, $\overline{CS}$/WE	11.4		12.6	V
$V_{IH3(P)}$	High-level input voltage, program mode 2	25		27	V
$V_{IL2(P)}$	Low-level input voltage, program mode 3	V_{SS}		1	V

Note 2 : With respect to V_{SS}
3 : Where $V_{IH3(P)} - V_{IL2(P)}$ = 25V (min)

Electrical Characteristics (T_a=25℃, V_{CC}=5V±5%, V_{DD}=12V±5%, V_{BB}=−5V±5%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$I_{IL1(P)}$	Low-level input current, address, chip select input	V_I = 5.25V			10	μA
$I_{IL2(P)}$	Low-level input current, program input				3	mA
$I_{IH1(P)}$	High-level current, program input				20	mA
I_{DD}	V_{DD} supply current	Worst case,		50	65	mA
I_{CC}	V_{CC} supply current	all inputs high 4		6	10	mA
I_{BB}	V_{BB} supply current	$\overline{CS}$/WE=5V, T_a=0℃		30	45	mA

Note 4 : Typical values are at T_a=25℃ and nominal supply voltage.

Timing Requirements (T_a=25℃, V_{CC}=5V±5%, V_{DD}=12V±5%, V_{BB}=−5V±5%, V_{SS}=0V, unless otherwise noted)

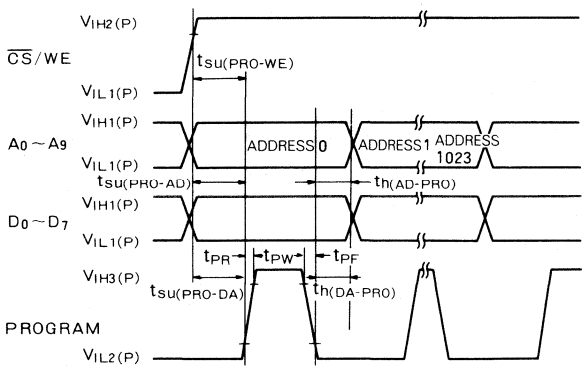
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{su(PRO-AD)}$	Program setup time with respect to address		10			μs
$t_{su(PRO-WE)}$	Program setup time with respect to WE low-to-high		10			μs
$t_{su(PRO-DA)}$	Program setup time with respect to data		10			μs
$t_{h(AD-PRO)}$	Address hold time with respect to program		1			μs
$t_{h(WE-PRO)}$	WE hold time with respect to program		0.5			μs
$t_{h(DA-PRO)}$	Data hold time with respect to program		1			μs
$t_{w(P)}$	Program pulse width		0.1		1	ms
$t_{r(P)}$	Program rise time		0.5		2	μs
$t_{f(P)}$	Program fall time		0.5		2	μs

Switching Characteristics (T_a=25℃, V_{CC}=5V±5%, V_{DD}=12V±5%, V_{BB}=−5V±5%, V_{SS}=0V, unless otherwise noted)

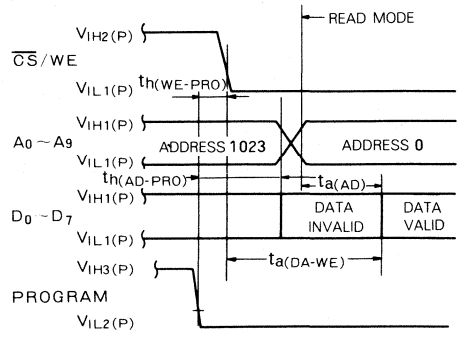
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{a(DA-WE)}$	Access time with respect to WE high-to-low				10	μs
$t_{dv(DA-\overline{CS})}$	Data valid time with respect to $\overline{CS}$ low-to-high		0		120	ns

Timing Diagram

Program Mode



From Program Mode to Read Mode



MITSUBISHI ELECTRIC

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 re-programmable 16 384-bit (2048-word by 8-bit) EPROMs. They incorporate N-channel silicon-gate MOS technology, and are designed for microprocessor programming applications.

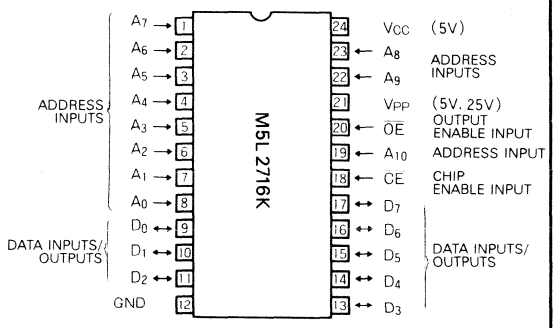
FEATURES

- Fast programming : 100s/16 384 bits (typ)
- Access time M5L 2716K : 450ns (max)
  M5L 2716K-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

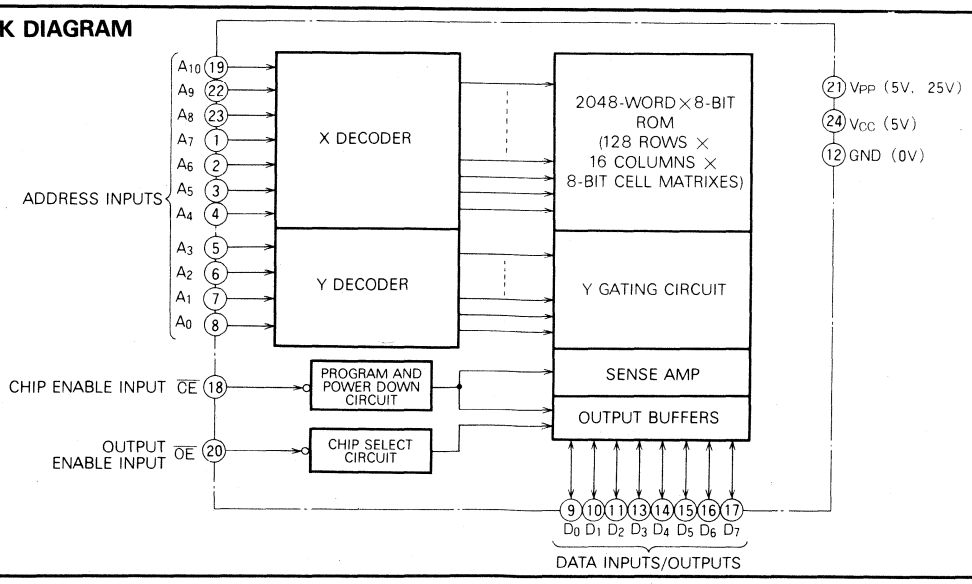
PIN CONFIGURATION (TOP VIEW)



Outline 24K10

5

BLOCK DIAGRAM



M5L 2716 K, K-65

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 $45ms \leqq t_{w(CE)} \leqq 55ms$.

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}	
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

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.

16 384-BIT (2048-WORD BY 8-BIT) ERASABLE AND ELECTRICALLY REPROGRAMMABLE ROM

ABSOLUTE MAXIMUM RATING

Symbol	Parameter	Conditions	Limits	Unit
V_{I1}	Input voltage, V_{PP}	With respect to GND	−0.3∼26.5	V
V_{I2}	Input voltage, V_{CC}, address, $\overline{OE}$, $\overline{CE}$, data		−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
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

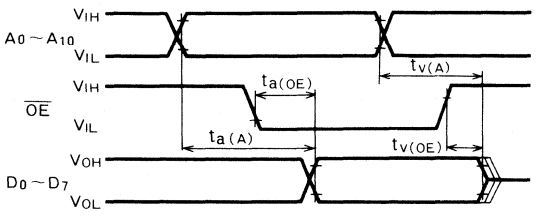
Switching Characteristics (Ta= 0 ∼70℃, V_{CC}= 5 V± 5 %, $V_{PP}=V_{CC}$, unless otherwise noted)

Symbol	Parameter		Test conditions		Limits			Unit
					Min	Typ (Note 1)	Max	
$t_{a(A)}$	Address access time	M5L 2716K	$\overline{OE}=\overline{CE}=V_{IL}$	$t_r \leqq 20$ns, $t_f \leqq 20$ns, 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		$\overline{OE}=V_{IL}$		0		100	ns
$t_{v(CE)}$	Data valid time after chip select		$\overline{CE}=V_{IL}$		0		100	ns
$t_{v(A)}$	Data valid time after address		$\overline{OE}=\overline{CE}=V_{IL}$		0			ns

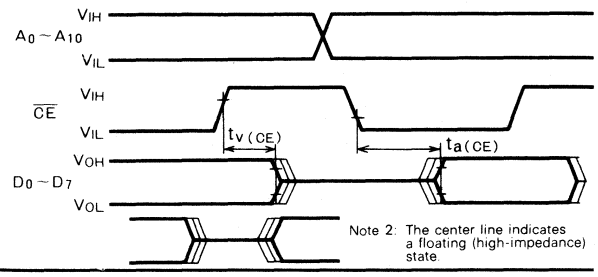
Note 1: at Ta=25℃ and normal supply voltage.

Timing Diagrams (Read Operation)

When Power-Down Mode Not Used



Power-Down Mode



Note 2: The center line indicates a floating (high-impedance) state.

M5L 2716 K, K-65

16 384-BIT (2048-WORD BY 8-BIT) ERASABLE AND ELECTRICALLY REPROGRAMMABLE ROM

PROGRAM MODE

Recommended Operating Conditions ($T_a = 25 \pm 5$ °C, 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 ($T_a = 25 \pm 5$ °C, $V_{CC} = 5\ V \pm 5\%$, $V_{PP} = 25 \pm 1$ 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

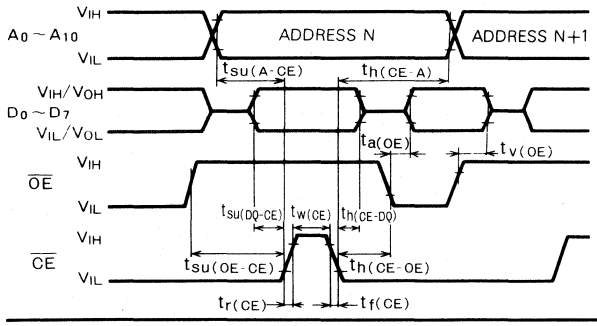
Timing Requirements ($T_a = 25 \pm 5$ °C, $V_{CC} = 5\ V \pm 5\%$, $V_{PP} = 25 \pm 1$ V, unless otherwise noted)

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 ($T_a = 25 \pm 5$ °C, $V_{CC} = 5\ V \pm 5\%$, $V_{PP} = 25 \pm 1$ 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)



MITSUBISHI ELECTRIC

MELPS 4 MICROCOMPUTERS

6

M58840-XXXP

SINGLE-CHIP 4-BIT MICROCOMPUTER WITH 8-BIT A/D CONVERTER

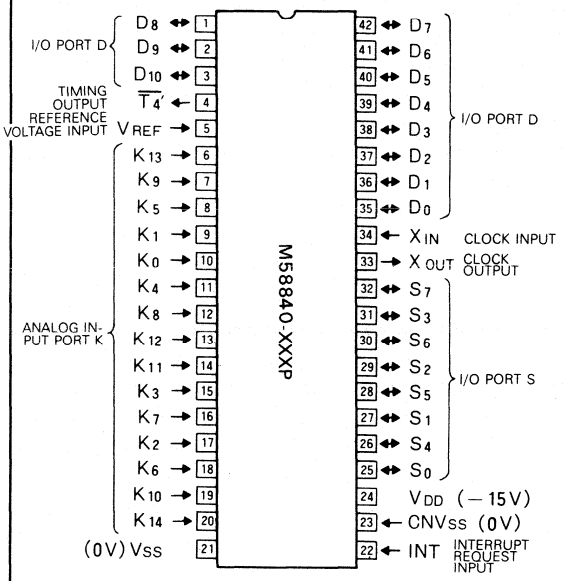
DESCRIPTION

The M58840-XXXP is a single-chip 4-bit microcomputer developed using P-channel aluminum-gate ED-MOS technology encased in a 42-pin plastic DIL package. It contains an 8-bit A/D converter and an analog input port. It is ideal for applications using a capacitive touch panel, because it is designed for systems that require the upgraded reliability that is assured by the elimination of mechanical switches from the control panels, and for systems requiring control based on analog signals received from sensors for temperature, humidity, light intensity, pressure velocity, etc.

FEATURES

- Basic machine instructions: 68
- Basic instruction execution time (at 600kHz frequency): 10 μs
- Large memory: ROM: 2048 words × 9 bits
  RAM: 128 words × 4 bits
- Single −15V power supply
- Internal A/D converter with ±1.2% accuracy
- Two data pointers for stack operations
- Subroutine nesting: 3 levels
- Analog input port (K): 15 bits
  (Can be used for a capacitive touch panel or analog inputs.)
- I/O port (S): Output: 8-bit × 1
  Input: 4-bit × 2
- I/O port (D): Output: 1 bit × 11
  Sense input: 1 bit × 11
- I/O port output voltage V_O: −33V (max)
- I/O port output current: I_{OH}(port S): −8mA (max)
  I_{OH}(port D): −15mA (max)
  (Direct drive for large fluorescent display tubes is possible.)

PIN CONFIGURATION (TOP VIEW)



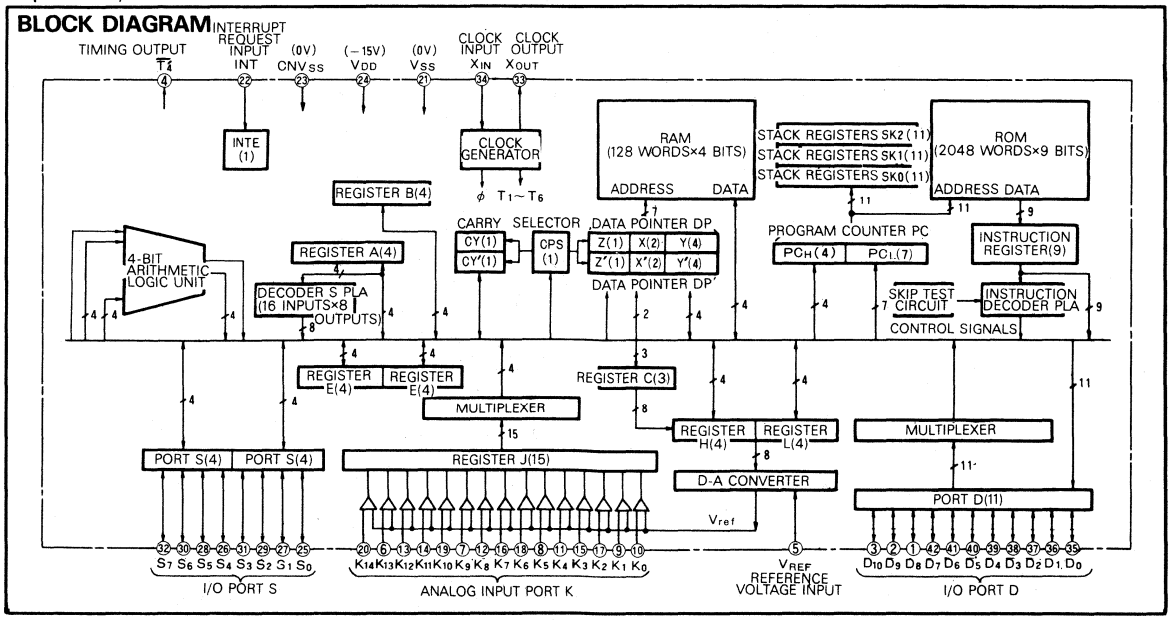
Outline 42P1

- Interrupt function: 1-level
- Internal (programmable logic array) for the output decoder: 16 inputs × 8 outputs
- On-chip clock generator

APPLICATIONS

- Microwave ovens, air conditioners, washing machines, home sewing machines
- Office equipment and copying machines

BLOCK DIAGRAM



M58840-XXXP

SINGLE-CHIP 4-BIT MICROCOMPUTER WITH 8-BIT A/D CONVERTER

FUNCTIONS

The device is composed of a 2048-word × 9-bit mask-programmable ROM, a 128-word × 4-bit RAM, a 4-bit arithmetic logic unit, a clock generator, I/O ports, an A/D converter, and an interrupt circuit.

The ROM stores 16 128-word pages of program, addressed by the program counter. The return addresses for subroutines and interrupts are saved in the 3 11-bit stack registers.

The RAM stores 8 16-digit files of data which are addressed by one of two data pointers.

Instructions (RAM addressing, register-to-register transfer, PLA output, RAM-to-accumulator transfer, arithmetic, etc. are executed through the 4-bit register A (accumulator). Any 4-bit data code from register A can be converted to a predetermined 8-bit code through the output S programmable logic array (PLA), because the output code of the PLA can be programmed during manufacture of the ROM mask.

The analog input port K is composed of 15 inputs. Analog voltage applied through these inputs is compared with the internal reference voltage V_{ref}, which is generated by the D-A converter from the value in register H-L, and the result is stored in register J.

Once the analog input signal is converted to 8-bit digital form, it can be freely handled by the processor.

The I/O port S is composed of 8 bits, and an 8-bit latch circuit is provided on its output side to latch the 8-bit data transferred from register A via the output S PLA, or data transferred directly from registers A and B, or data transferred from register E. An 8-bit signal applied to port S is transferred to register A in 4-bit units.

Eleven I/O lines are provided for the I/O port D, each of which can be operated independently as an input or output line. Latches are provided for all output bits. After designating a particular bit position with the contents of Y in the data pointer, any one of the lines can be output or sensed.

When port S or port D is used as input, its output has to be cleared and changed to a low-level state before sensing any input data.

PERFORMANCE SPECIFICATIONS

Parameter			Performance
Basic machine instruction			68
Basic instruction execution time			10μs (at 600kHz frequency)
Clock frequency			300~600kHz
Memory capacity	ROM		2048 words × 9 bits
	RAM		128 words × 4 bits
I/O port	K	Input	1-bit × 15
	S	Output	8-bit × 1
		Input	4-bit × 2
	D	Output	1-bit × 11
		Sense input	1-bit × 11
A/D conversion circuit			Internal (±1.2% accuracy)
Touch panel interface			Internal
Subroutine nesting			3 levels (including 1 level of interrupt)
Clock generation circuit			Internal (external CR or IF ceramic filter is provided)
I/O characteristics of ports	I/O port output voltage		−33V (max)
	Port S output current		−8mA (max)
	Port D output current		−15mA (max)
Supply voltage	V_{DD}		−15V (nom)
	V_{SS}		0V
Device structure			P-channel aluminum-gate ED-MOS
Package			42-pin plastic molded DIL
Power dissipation			700mW (max)

OPERATION OF BASIC FUNCTION BLOCKS

Function	Operation
Program counter PC	Used for designating ROM address and determining readout sequence of the instructions stored in the ROM. The PC is a pure binary counter consisting of 11 bits, of which the high-order four (PC_H) designate the ROM page and the low-order seven (PC_L) the address on each page. Each time an instruction is executed, PC_L is incremented by one step. Its value is set to the designated address when a branch, subroutine call or return instruction is executed.
Stack registers SK0, SK1, SK2	Temporarily stores the contents of the PC, while executing subroutine or interrupt programs, until the program returns to its original routine. The (SKs) are organized in 3 words of 11 bits each, enabling up to 3 levels of subroutine nesting. If one word is used for an interrupt routine, the remaining two levels can be used for subroutine calls.
Program storage memory ROM	This 2048-word × 9-bit mask-programmable ROM can be programmed with any machine instruction code in accordance with the customer's specification. It consists of 16 pages, each containing 128 words of instructions.
Instruction register	Stores the 9-bit instruction code fetched from the ROM. Control signals are then transferred to the logic circuit through the PLA instruction decoder. The skip flag circuit determines the skip condition of the XAMI instruction, and this should be specified when the ROM is ordered.
Data pointers DP, DP'	Assigned to designate RAM address, 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 the DP) designates the RAM file group; register X (the succeeding two bits) designates a RAM file, and register Y (the least significant four 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.
Data memory RAM	This 512-bit (128 word × 4-bit) RAM stores both processing and control data. Bit manipulation is possible over the entire area of storage. All of 128 words can be treated as an organization of 2 file groups × 4 files × 16 digits × 4 bits. When any instruction related to the RAM is to be executed, it is essential that the desired selector CPS and data pointer DP are selected.
4-bit arithmetic logic unit	This unit executes 4-bit arithmetic and logic operations through the 4-bit adder and its related logic circuits. The arithmetic logic unit performs subtraction, addition, logical comparison, arithmetic comparison, and bit manipulation.
Register A and carry flag CY	Register A is an 4-bit accumulator that constitutes the center of the microcomputer. Data processing procedures such as arthmetic and logical operations, data transfer, exchange, conversion, and data input/output are executed through this register. Overflow of register A is stored in the carry flags CY and CY1 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 by 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 data storage, but also as a temporary storage register for I/O port S.
Output S PLA	4-bit data from register A is translated into a one-of-16 code by the decoder and applied to the PLA to generate an 8-bit code output. The customer can specify the output code of the PLA when ordering the mask ROM. However, the code of output S PLA of the M58842S system evaluation device has already been programmed with the standard code.
A/D conversion circuit	This consists of register C, register H-L, D-A converter, register J and 15 comparators. The D-A converter generates the analog output signal V_{ref} for the value in register H-L. All analog inputs to port K are compared with this value, and the results set/reset register J. Register C is used for bit designation of register H-L.
Interrupt flag INTE	INTE is a 1-bit flip-flop and controls interrupt operation. When INTE="1", the CPU is ready to accept an interrupt, but it inhibits interrupts when INTE="0". When the instruction EI is executed INTE is set to "1", and reset to "0" when the instruction DI is executed.

PIN DESCRIPTIONS

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 independent 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.
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 filter 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 filter 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 is used for requests is selected by means of the program. When the instruction INTE 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.
$\overline{T4}$	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.
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.
CNV_{SS}	CNV_{SS} input	In	This input terminal should be conected with the V_{SS} and have a high-level input (0V) applied.

OPERATION

ROM Address Map

One word of the ROM is composed of 9 bits, one page is composed of 128 words (address 0~127), and it has 16 pages (page 0~15). Total memory capacity is composed of 2048 words (128 words × 16 pages) × 9 bits. The ROM address map is shown in Fig. 1.

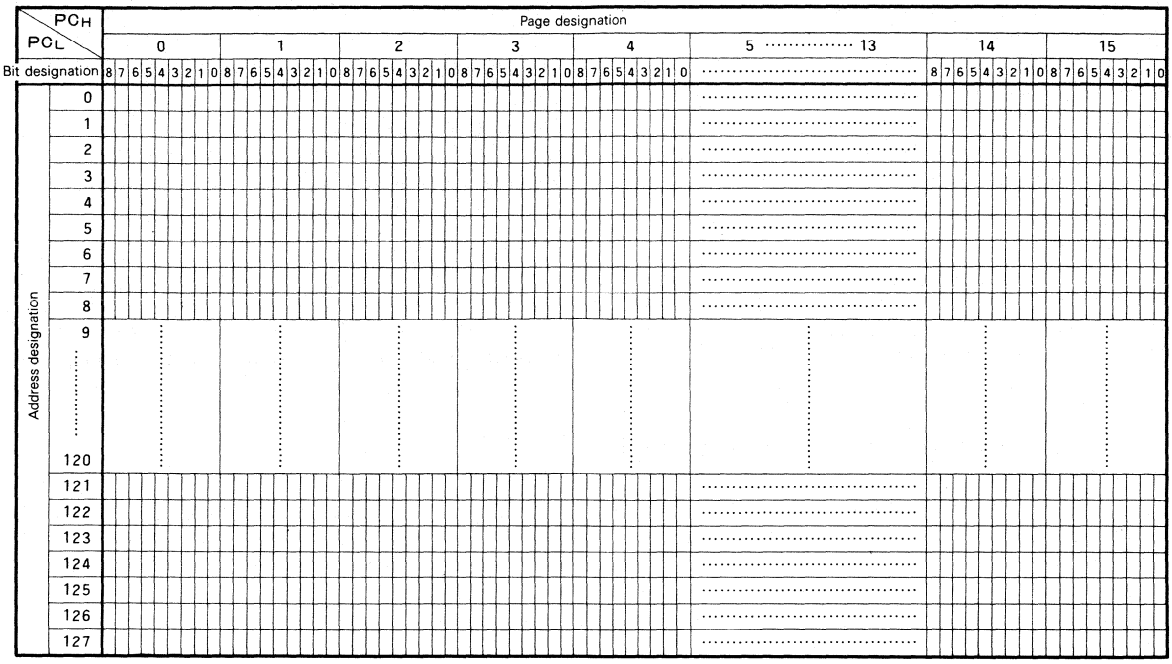
The page is designated by the high-order four bits of the program counter PC_H, while the address is designated by the low-order 7 bits of the program counter PC_L. After the execution of an instruction stored in address 127 of a page, the program returns to the address 0 of the same page. To change the page, the following branch instructions are used:

BL, BML, BLA, and BMA.

Pages 14 and 15, however, constitute special pages for subroutine call. Page 14 is specially designed for storage of subroutines. When a subroutine call instruction BM or BMA is executed in any page other than page 14, the page is automatically changed to 14. In other words, instructions BM and BMA can call a subroutines program on page 14 with just one instruction.

However, when instruction BM or BMA is executed on page 14, it performs similar branch instruction B or BA. If the instruction B or BA is executed on page 14, the program will jump to the specified address on page 15.

Fig. 1 ROM address map



RAM Address Map

One word of the RAM is 4 bits, and the total capacity of the RAM is 128 words × 4 bits. The address is designated by the data pointer; the most significant bit of the data pointer is designated register Z, while the next two bits are designated register X and the least significant four bits register Y. The address map of the RAM is shown in Fig. 2.

It is considered that there are 8 files (F0~F7) in the RAM, and each file is composed of 16 words × 4 bits. It would be convenient to use a file as a register of 16 digits. These files are designated by the registers Z and X.

Instructions

TAM, XAM, XAMD, and XAMI,

can serve to change the file assignment by the value of the j-modifier after the execution of the instruction, so that these instructions are useful to program for shifting and transferring the data between files.

Fig. 2 RAM address map

File designation	Register Z	0				1			
	Register X	0	1	2	3	0	1	2	3
File name		F0	F1	F2	F3	F4	F5	F6	F7
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	3 2 1 0	3 2 1 0
Address designation (register Y)	0								
	1								
	2								
	3								
	4								
	5								
	6								
	7								
	8								
	9								
	10								
	11								
	12								
	13								
	14								
	15								

A-D CONVERSION CIRCUIT

The following functional blocks are implemented in the A/D conversion circuit. Its diagram is shown in Fig. 3.

Comparators

Each comparator is composed of a single-channel MOS circuit employing chopper-type amplification. All input signals $V_{K(Y)}$ (where, (Y)=0~14) from port K are compared with the V_{ref} from the D-A converter by these comparators.

Register J

Register J is composed of fifteen 1-bit registers, and the comparison results of each comparator are stored simultaneously in all the 1-bit registers.

The value of comparison is;

"1", when $|V_{ref}| > |V_{K(Y)}|$, and

"0", when $|V_{ref}| < |V_{K(Y)}|$,

where (Y) represents bit position in register J which is designated by register Y.

Each bit of these comparison results can be checked by the instruction SZJ.

Register A

This register is a 4-bit accumulator which is designed to be the center of major data-processing functions of the microcomputer, performing functions such as arithmetic, control and I/O operations.

Register H-L

The two 4-bit registers H and L are capable of transferring and exchanging data to/from register A. The 8-bit digital data for the D-A converter is transferred from these, the high-order four bits from H and the low-order four bits from L.

Register C

This 3-bit register is a counter which is used to designate bit positions in the register H-L.

D-A Converter

The D-A converter converts the digital value stored in register H-L, referencing with the external reference voltage V_{REF} applied from the terminal V_{REF}, to the analog value of the internal reference voltage V_{ref}.

The theoretical value of the internal reference voltage V_{ref} is expressed with the following equations;

$$V_{ref} = \frac{n-0.5}{256} \times V_{REF}, \text{ where, } n=1,2,\cdots\cdots 255$$

$$V_{ref} = 0\,V, \text{ where, } n=0$$

The value n is the contents of the register H-L.

A-D CONVERSION ALGORITHMS

The A-D conversion on the M58840-XXXP chip, uses one of two conversion algorithms, successive approximation or sequential comparison. The choice is made by program.

Successive Approximation

In this algorithm register H-L is first cleared, and in most significant bit (MBS) is set to "1". Then the analog input signal $V_{K(Y)}$ is compared with the internal reference voltage V_{ref}, and if $|V_{K(Y)}|$ is smaller than $|V_{ref}|$, the data is unchanged. If $|V_{K(Y)}|$ is greater than $|V_{ref}|$, the compared bit is reset to "0", and the next bit is set to "1". The same procedure is repeated until the least significant bit (LSB) is compared. The 8-bit digital value which is converted from the analog value of the input signal is stored in register H-L.

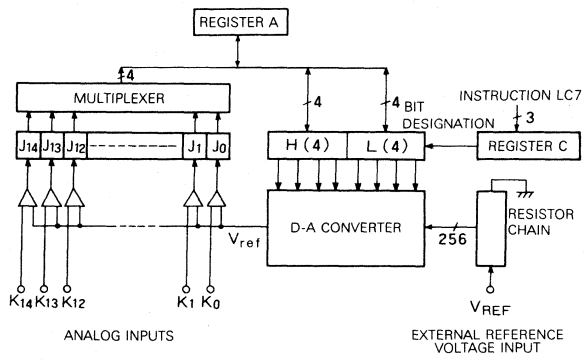
With this method, the conversion time is constant regardless of the signal level of the analog input, and is therefore suitable for detecting rapid variable analog signals or detecting different analog values of multi-channels. Time required for 8-bit A/D conversion is about 0.6ms at 600kHz.

Sequential Comparison

In this algorithm, the analog input signal $V_{K(Y)}$ is compared with the internal reference voltage V_{ref}. As a result, if $|V_{K(Y)}|$ is greater than $|V_{ref}|$, the contents of register H-L is incremented by one, and if $|V_{K(Y)}|$ is smaller than $|V_{K(Y)}|$, it is decremented by one. The same procedure is repeated until such a time that the data is increased and decreased alternately. Then the 8-bit digital value which is converted from the analog value of the input signal is stored in register H-L.

This method is suitable for applications where variation of analog value is already known to be small and conversion speed is faster than the Successive Approximation Algorithm. However, considerable conversion time is required to find the initial value.

Fig. 3 A/D conversion circuit block diagram



MITSUBISHI ELECTRIC

CLOCK GENERATION CIRCUIT

The clock pulse is easily generated by connecting an external CR circuit or IF ceramic filter between the pins X_{IN} and X_{OUT}, because a clock generation circuit is contained on the chip. In case the clock signal is to be supplied from an external source, the clock oscillation source should be connected to pin X_{IN}, leaving pin X_{OUT} open. Examples of such circuits are shown in Fig. 4~6.

Fig. 4 Externally provided CR circuit

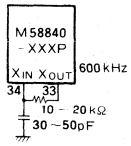


Fig. 5 Ceramic filter circuit

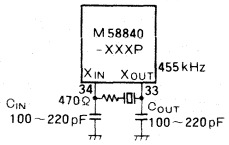
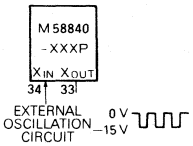


Fig. 6 External synchronization circuit



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 is reset to disable further interrupt, after which the program jumps from the main program to address 0 of page 12, where the interrupt program is stored. When the interrupt routine is executed, one of the three stack registers is used for the interrupt, leaving the other two stack registers available for subroutines.

After the interrupt routine is completed the program will return to the main program using the instruction RTI, but it is necessary to be sure to save the contents of the registers—such as the data pointer DP, register A, carry flag CY and other registers that might be used in the interrupt program—by means of a program that is executed at the start of the interrupt program, and also be sure to restore those data to the respective flags and registers at the end of the interrupt program before the RTI instruction is executed.

When an interrupt request signal is applied, the internal state of the microcomputer is changed as follows:

(1) Program counter
Current address of the main program is stored in a stack register, and address 0 of page 12 is set in the program counter.

(2) Interrupt flag INTE
The flag INTE is reset to disable further interrupts. The disable state will continue even after the program has returned from the interrupt routine to the main program by the instruction RTI. It can be released only when the flag INTE is set to "1". When the instruction INTH has been executed, interrupt is enabled with the high-level INT signal input. The interrupt is not enabled as long as the interrupt request signal INT remains high-level. But interrupt will be enabled as soon as the interrupt request signal INT is turned to high-level after it has once been changed to low-level.

(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 condition is retained.

POWER-ON RESET FUNCTION

When the power is turned on, and the power-on signal applied to the power supply terminal V_{DD} meets with the requirement described in Fig. 7, the microcomputer is reset by the internally provided automatic reset circuit. The reset operation is automatic, and the program counter is set to address 0 of page 0, where the program will start. After the power-on reset function is activated, the following functions are initialized:

(1) The program counter is set to address 0 of page 0 (PC)←0.

(2) Interrupt mode is in the interrupt disable state (INTE)←0. This is the same state as when the instruction DI is executed.

(3) Turning the interrupt request signal INT to high-level produces the interrupt enable state, the same condition as when instruction INTH is executed.

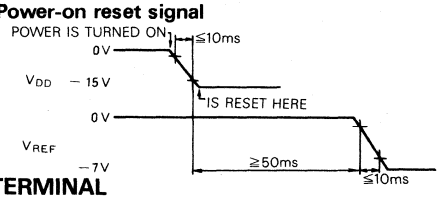
(4) All outputs of the port S are cleared to low-level. (S)←0

(5) All outputs of the port D are cleared to low-level. (D)←0

(6) The carry and data pointer selector CPS is reset to 0 to designate the DP and CY side. (CPS)←0

In the event that the power-on reset function cannot be operated satisfactorily due to the inadequate rising characteristics of the power supply, the same perfect power-on reset function can be obtained by applying such a waveform as shown in Fig. 7 to the external reference voltage input V_{REF}. If V_{REF} is kept high for more than 50ms after V_{DD} has turned low, the microcomputer will be reset. The program is then ready to start from address 0 as soon as V_{REF} is turned low (−7V).

Fig. 7 Power-on reset signal



TEST TERMINAL

Even though the pin $\overline{T_4'}$ is provided for timing output from the internal logic of the LSI, it is not required for the operation of the microcomputer, so that it should be connected with the pin Vss (0V). The pin CNVss should also be connected with the pin Vss (0V).

SINGLE-CHIP 4-BIT MICROCOMPUTER WITH 8-BIT A/D CONVERTER

METHOD OF MASKING DESIGNATION

The following items can be specified by a customer for programming and masking the microcomputer M58840-XXXP:

(1) ROM data: 2048 words × 9 bits
(2) Output S PLA data: 16 outputs × 8 bits
(3) Skip conditions for the instruction XAMI: 4 bits
(4) Load resistors and discharging transistors for the port K: 15 inputs

When the above specifications have been supplied, an automatic mask design program for the single chip microcomputer generates the following in order to meet the customer's specifications accurately and promptly:

(1) The plotter instructions for automatic mask specification.
(2) A check list for verifying that the customer's specifications have been met correctly.
(3) A test program to assure that the production microcomputers meet specifications.

ROM Data

Data to be stored in the ROM is a program of 2048 words × 9 bits. This program should be supplied in one of the following three formats:

(1) MELPS 4 source program.
(2) MELPS 4 absolute object program in Takeda format.
(3) MELPS 4 absolute object program in Minato format.

The source program should be prepared in the assembly language provided for the MELPS 4 cross assembler. The object program should be prepared in hexadecimal format applicable for the PROM writer, and be separated in blocks of 1024 bytes each. In other words, the object program should be separated into two sections, each having 1024 words of the object program containing the low-order 8 bits of the program, and another two sections of the program, each having 1024 words of the object program containing the high-order 1 bit of the program. This is the same procedure as in evaluating the program with the M5L2708S EPROM using the PCA0401 MELCS 4 system evaluation computer. It has format interchangeability with Takeda Riken and Minato Electronics PROM writer tapes. These object programs can be prepared automatically by using the MELPS 4 PROM writer tape generation program.

Source Program Format and Medium

The source program should be prepared in MELPS 4 assembly language.

Punched card: 80 columns/line (equivalent to the IBM punched card). Character codes should be in Hollerith code and the cards compatible with the IBM 029 key punch.

Paper tape: 8-channel, 25.4mm width. Character codes should be in ASCII code with even parity.

Object Program Format and Medium

The object program should be prepared in absolute format and be separately prepared in blocks of 1024 bytes in hexadecimal format.

MELPS 4 Takeda format: Interchangeable with Takeda Riken's PROM writer tape format.

MELPS 4 Minato format: Interchangeable with Minato Electronics' PROM writer tape format.

Paper tape: 8-channel, 25.4mm width. Character codes should be in ASCII code with even parity.

Output S PLA Data

The S PLA is a mask-programmable logic array which, on the basis of 4-bit data from register A, generates an 8-bit output for the I/O port S or register E. The desired output levels of the outputs $S_0 \sim S_7$ of port S should be specified by using "H" (high-level: 0V) or "L" (low-level: −33V) corresponding to the 16 possible inputs (contents of register A 0~15). For the code specification form, refer to the output S PLA code list in the data sheet prepared for the MELPS 4 system evaluator device M58842S.

Skip Condition for the Instruction XAMI

Standard products (the single-chip 4-bit microcomputer M58840-XXXP and the MELPS 4 system evaluator device M58842S) are designed to skip the next instruction when (Y)=15 before execution,

(A) ⇆ (M(DP)), and
(Y) ← (Y) + 1.

Finally the next instruction is executed or skipped depending on the initial contents of (Y). Because of this function, the test instruction SEY_n, which tests (Y)=n, may be eliminated to reduce program steps when (Y)=15.

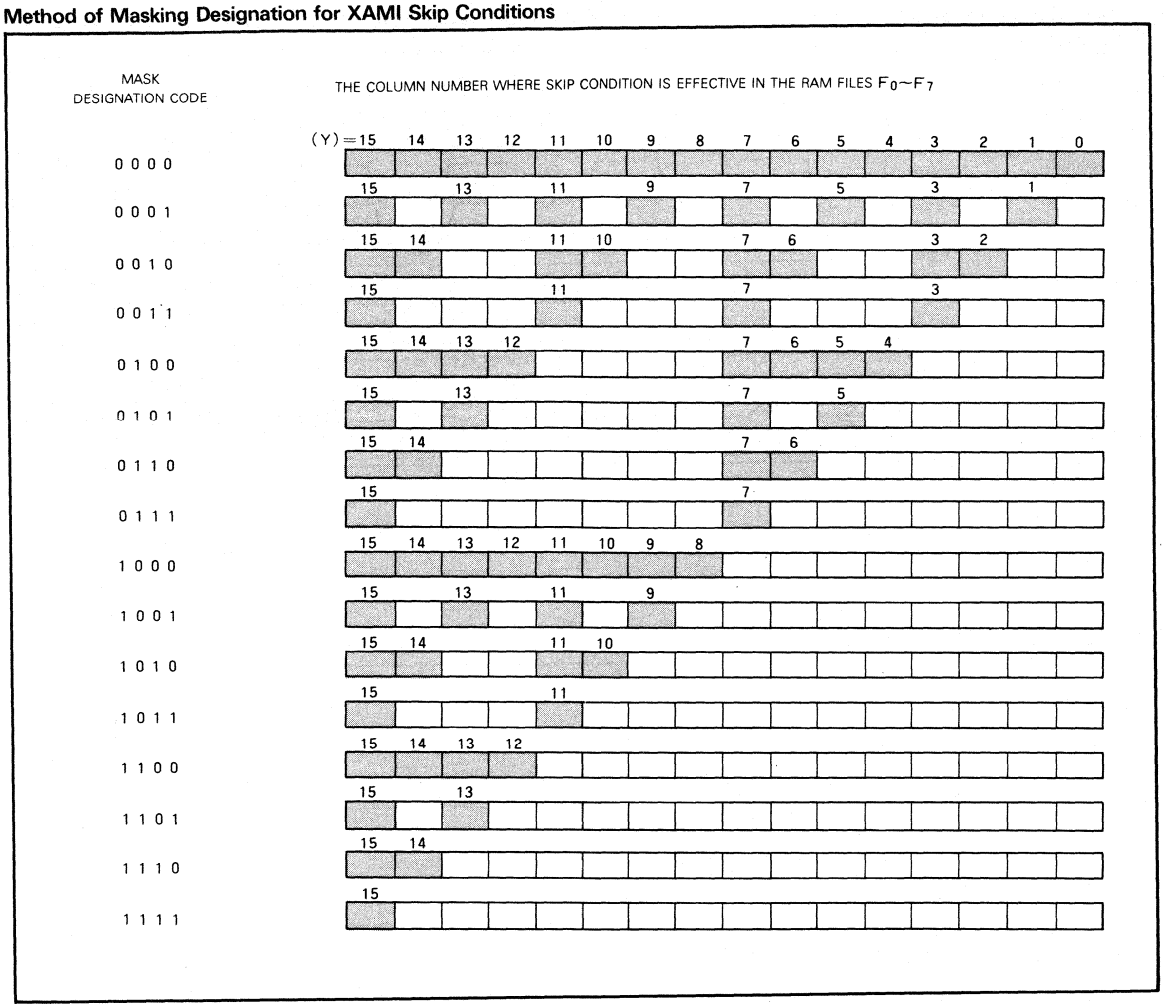
As an optional feature any of the following skip conditions may be substituted for the condition (Y)=15. This feature is useful for handling data of various lengths by providing a number of small capacity registers within the files.

Mask designation codes for the optional skip condition should be specified with one of the following 4-bit binary numbers.

Mask designation code	Y values that will cause a skip of the next instruction
0000	(Y)=0, 1, 2·····15
0001	(Y)=1, 3, 5, 7, 9, 11, 13, 15
0010	(Y)=2, 3, 6, 7, 10, 11, 14, 15
0011	(Y)=3, 7, 11, 15
0100	(Y)=4, 5, 6, 7, 12, 13, 14, 15
0101	(Y)=5, 7, 13, 15
0110	(Y)=6, 7, 14, 15
0111	(Y)=7, 15
1000	(Y)=8, 9, 10, 11, 12, 13, 14, 15
1001	(Y)=9, 11, 13, 15
1010	(Y)=10, 11, 14, 15
1011	(Y)=11, 15
1100	(Y)=12, 13, 14, 15
1101	(Y)=13, 15
1110	(Y)=14, 15
1111	(Y)=15

The standard mask designation code is 1111.

Method of Masking Designation for XAMI Skip Conditions



Load Resistors for the Input Port K

Load resistors for input lines of the port K can be provided on the M58840-XXXP chip as an optional feature. An enhancement-type MOS transistor, whose resistance is 100~200kΩ, is used for the load resistors. But this load resistor is not provided on the M58842S MELPS 4 system evaluation device. Mask designation format details for this option can be found in the masking confirmation sheet.

Discharging Transistors for the Input Port K

Discharging transistors for input lines of the port K can be provided on the M58840-XXXP as an optional feature. These discharging transistors are necessary when the capacitive touch panel is used. They are internally contained, and all of the input lines of the port K on the M58842S system evaluation device have been provided with these transistors.

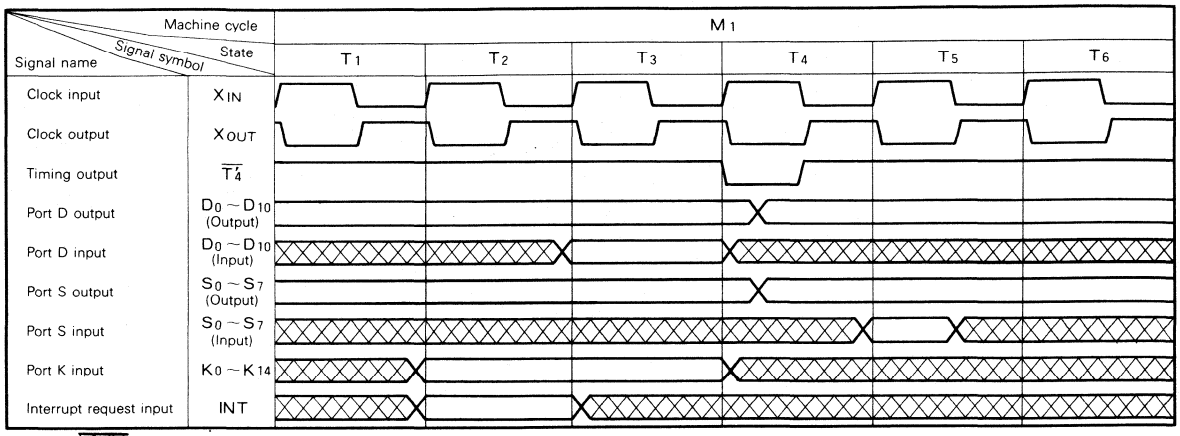
Mask designation format details for this option can be found in the masking confirmation sheet.

Materials Required with Order

The following information should be given when the masking is ordered:

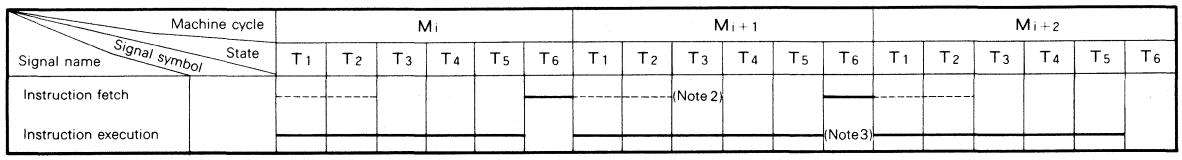
(1) MELPS 4 masking confirmation sheet.
(2) ROM data: one set of the program should be prepared when punched cards are to be used, but two copies are required if punched paper tape is to be used.
(3) Output S PLA data: punched cards or listed on the confirmation sheet.
(4) Skip condition for the instruction XAMI: punched cards or listed on the confirmation sheet.
(5) Load resistors for input port K: punched cards or listed on the confirmation sheet.
(6) Discharging transistors for input port K: punched cards or listed on the confirmation sheet.

BASIC TIMING CHART



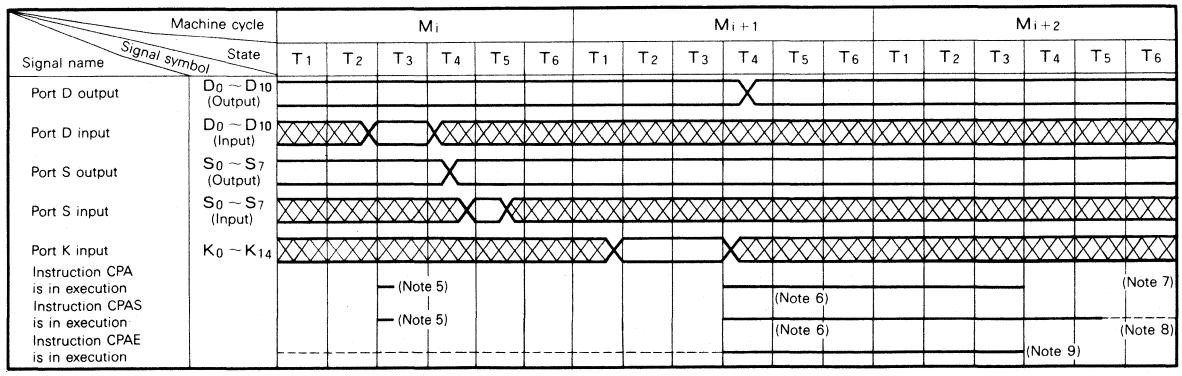
Note 1: XXX indicates invalid signal input.

INSTRUCTION FETCH TIMING



Note 2 : Instruction fetch time can differ depending on the types of the instructions.
3 : The instruction which was fetched in the preceding cycle is executed.
4 : The execution of the instruction and addressing of ROM and RAM are performed simultaneously.

I/O INSTRUCTION EXECUTION TIMING



Note 5 : By short-circuiting port K inputs with the V_{SS} (0V), capacitance connected to the port K input is discharged.
6 : Analog value applied to port K inputs is maintained to be compared with the reference voltage V_{ref}.
7 : Analog value applied to port K inputs is read until the next CPA or CPAS instruction is executed.
8 : The state of Note 6 is maintained until an instruction CPAE is executed, during which time the analog value applied to port K is not read. This time should be less than 100μs to assure the accuracy of the A/D conversion.
9 : The condition of Note 8 is released.

BRANCH AND SUBROUTINE CALL INSTRUCTION EXECUTION TIMING

Machine cycle / Operation / State	M_i						M_{i+1}						M_{i+2}					
	T_1	T_2	T_3	T_4	T_5	T_6	T_1	T_2	T_3	T_4	T_5	T_6	T_1	T_2	T_3	T_4	T_5	T_6
Instruction B_{xy} (to be operated as the branch instruction, when the instruction BM or BMA was not executed before).																		
Program counter	$(\overline{PC_L}) \leftarrow xy$		$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$						$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$									
ROM address	$(\overline{\text{ROM address}}) \leftarrow (PC)$						$(\overline{\text{ROM address}}) \leftarrow (PC)$											
Execution of program	Execution of the branch instruction						Execution of the instruction stored in the branched address											
Instruction B_{xy} (to be operated as the branch instruction to page 15, when the instruction BM or BMA was not executed before).																		
Program conter	$(PC_H) \leftarrow 15$ $(PC_L) \leftarrow xy$		$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$						$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$									
ROM address	$(\overline{\text{ROM address}}) \leftarrow (PC)$						$(\overline{\text{ROM address}}) \leftarrow (PC)$											
Execution of program	Execution of the branch instruction						Execution of the instruction stored in the branched address on page 15											
Instruction BM_{xy} (subroutine call instruction).																		
Program counter	$(PC_H) \leftarrow 14$ $(PC_L) \leftarrow xy$		$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$						$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$									
ROM address	$(\overline{\text{ROM address}}) \leftarrow (PC)$						$(\overline{\text{ROM address}}) \leftarrow (PC)$											
Stack register	$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$																	
Execution of program	Execution of the subroutine call instruction						Execution of the instruction stored in the subroutine called address											
Instruction BL p, xy (branch instruction)																		
Program counter	$\overline{\text{Temporary register}} \leftarrow p$		$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$				$(PC_H) \leftarrow \overline{\text{Temporary register}}$ $(PC_L) \leftarrow xy$		$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$				$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$					
ROM address							$(\overline{\text{ROM address}}) \leftarrow (PC)$						$(\overline{\text{ROM address}}) \leftarrow (PC)$					
Execution of program	Page number is stored temporarily						Execution of branch instruction						Execution of the instruction stored in the branched address					
Instruction BML p, xy (subroutine call instruction)																		
Program counter	$\overline{\text{Temporary register}} \leftarrow p$		$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$				$(PC_H) \leftarrow \overline{\text{Temporary register}}$ $(PC_L) \leftarrow xy$		$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$				$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$					
ROM address							$(\overline{\text{ROM address}}) \leftarrow (PC)$						$(\overline{\text{ROM address}}) \leftarrow (PC)$					
Stack register			$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$															
Execution of program	Page number is stored temporarily						Execution of the subroutine call instruction						Execution of the instruction stored in the subroutine called address					

Note 10 : For instructions BA, BMA, BLA and BMLA in the preceding cycle of the execution of the instruction B, BM, BL and BML, respectively, extra period is added, in which the flag is set to replace the low-order 4 bits of the program counter with the contents of the register A.

INTERRUPT EXECUTION TIMING

Machine cycle / State		M_i						M_{i+1}						M_{i+2}					
		T_1	T_2	T_3	T_4	T_5	T_6	T_1	T_2	T_3	T_4	T_5	T_6	T_1	T_2	T_3	T_4	T_5	T_6
Interrupt request input	INT																		
Program counter	(PC)			$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$						$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$		(注11)		$(PC_H) \leftarrow 12$ $(PC_L) \leftarrow 0$		$(\overline{PC_L}) \leftarrow (\overline{PC_L}) + 1$			
ROM address								$(\overline{\text{ROM address}}) \leftarrow (PC)$						$(\overline{\text{ROM address}}) \leftarrow (PC)$					
Stack register												$(SK2) \leftarrow (SK1) \leftarrow (SK0) \leftarrow (PC)$							
Execution of program														Jump to address 0 of page 12 executed					

Note 11 : When the instruction executed in the machine cycle M_{i+1} is not a BA, BMA, BL, BML, BLA or BMLA instruction, the interrupt takes effect at the next state T_1 without executing $(PC_L) \leftarrow (PC_L) + 1$.

SINGLE-CHIP 4-BIT MICROCOMPUTER WITH 8-BIT A/D CONVERTER

MACHINE INSTRUCTIONS

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
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 4 *i*	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	(Y_1Y_0)=0 when : (A)←($J_3J_2J_1J_0$) (Y_1Y_0)=1 when : (A)←($J_7J_6J_5J_4$) (Y_1Y_0)=2 when : (A)←($J_{11}J_{10}J_9J_8$) (Y_1Y_0)=3 when : (A)←(0 $J_{14}J_{13}J_{12}$)	–	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 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)=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	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 (2)	(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 is not produced and 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)←$\overline{(A)}$	–	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		Skips next instruction when the contents of register Y are equal to the value y 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
	SHL	0 0100 0010	0 42	1	1	(C_2)=1 when : ($H_{(C1-C0)}$)←1 (C_2)=0 when : ($L_{(C1-C0)}$)←1	–	X	Sets the bit in register L or H designated by register C. The box shows the relationship between register C and bit position.
	RHL	0 0101 0010	0 52	1	1	(C_2)=1 when : ($H_{(C1-C0)}$)←0 (C_2)=0 when : ($L_{(C1-C0)}$)←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)	$\|V_{ref}\| > \|V_{K(i)}\|$ when : ($J_{(i)}$)←1 $\|V_{ref}\| < \|V_{K(i)}\|$ when : ($J_{(i)}$)←0 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, wherever $\|V_{ref}\| > \|V_{K(i)}\|$ is true, or resets it, wherever $\|V_{ref}\| < \|V_{K(i)}\|$ is true.
	CPAS	0 0101 0001	0 51	1	1	$\|V_{ref}\| > \|V_{K(i)}\|$ when : ($J_{(i)}$)←1 $\|V_{ref}\| < \|V_{K(i)}\|$ when : ($J_{(i)}$)←0 i=0~14	–	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 change
	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.

(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

MITSUBISHI ELECTRIC

Tape of instruction	Mnemonic	Instruction code D8 D7D6D5D4 D3D2D1D0	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	(PCL)←16x+y	–	X	Jumps to address xy of the current page
						(PCH)←15, (PCL)←16x+y			Jumps to address xy on page 15 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	(PCH)←p (PCL)←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	(PCL)←16x+(A)	–	X	Jumps to address x(A) of the current page.
						(PCH)←15, (PCL)←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	(PCH)←p (PCL)←16x+(A)	–	X	Jumps to the address x(A) of page p.
Subroutine calls	BM xy	1 0xxx yyyy	1 xy	1	1	(SK2)←(SK1)←(SK0)←(PC) (PCH)←14, (PCL)←16x+y	–	X	Calls for the subroutine starting at address xX on page 14.
						(PCH)←14, (PCL)←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)←(SK1)←(SK0)←(PC) (PCH)←p, (PCL)←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)←(SK1)←(SK0)←(PC) (PCH)←14, (PCL)←16x+(A)	–	X	Calls for the subroutine starting at address x(A) of page 14.
						(PCH)←14, (PCL)←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)←(SK1)←(SK0)←(PC) (PCH)←p, (PCL)←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)←(SK0)←(SK1)←(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)←(SK0)←(SK1)←(SK2)	–	X	Returns to the main routine from the subroutine.
	RTS	0 0100 0101	0 45	1	2	(PC)←(SK0)←(SK1)←(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))←1, where, (Z)=1, (Y)=0~10	–	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	2	(D(Y))←0, where, (Z)=1, (Y)=0~10	–	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, (Y)=0~10	(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	(S7~S4)←(B), (S3~S0)←(A)		X	Outputs contents of registers A and B to port S.
	OSPA	0 0001 0111	0 17	1	1	(S7~S0)← through PLA←(A)		X	Decodes contents of register A by PLA and the result is output to port S.
	OSE	0 0000 1011	0 0B	1	1	(S)←(E)		X	Outputs contents of register E to port S.
	IAS i	0 0101 010i	0 54 + i	1	1	i=0 : (A)←(S7~S4) i=1 : (A)←(S3~S0)		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)←0		X	Clears port D.
	CLS	0 0001 0000	0 10	1	1	(S)←0		X	Clears port S.
	CLDS	0 0001 0001	0 11	1	1	(D)←0 (S)←0		X	Clears ports S and D.
Interrupts	EI	0 0000 0101	0 05	1	1	(INTE)←1		X	Sets interrupt flag INTE to enable interrupts.
	DI	0 0000 0100	0 04	1	1	(INTE)←0		X	Resets interrupt flag INTE to disable interrupts.
	INTH	0 0000 0110	0 06	1	1	(INTP)←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)←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)←(PC)+1		X	No operation

Symbol	Contents	Symbol	Contens	Symbol	Contents
A	4-bit register (accumulator)	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.
PCH	The high-order four bits of the program counter.	D	11-bit port	C + x	Hexadecimal number C + binary number x.
PCL	The low-order seven bits of the program counter.	K	15-bit port		
PC	11-bit program counter, combination of PCH and PCL.	S	8-bit port		

6

SINGLE-CHIP 4-BIT MICROCOMPUTER WITH 8-BIT A/D CONVERTER

LIST OF INSTRUCTION CODES

D_8-D_4 / D_3-D_0 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 12 : 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 asterisk (*) and bar (-) must not be of used.

Note 13 : 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

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	700	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			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	−16.5	−15	−13.5	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 and S inputs	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	−7		−5	V
V_{OL}	Low-level output voltage; port D and S outputs	−33		0	V
$f(\phi)$	Internal clock oscillation frequency	300		600	kHz

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 70°C, V_{DD} = −15V ± 10%, V_{SS} = 0V, $f(\phi)$ = 300 ~ 600kHz, unless otherwise noted)

Symbol	Parameter	Conditions	Limits			Unit
			Min	Typ	Max	
V_{IH}	High-level input voltage, port D and S inputs		−1.5		0	V
V_{IL}	Low-level input voltage, port D and S inputs		−33		−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
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 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			±3	LSB
	A-D conversion zero error	V_{REF} = −7V			±3	LSB
	A-D conversion full-scale error	V_{REF} = −7V			±3	LSB

6

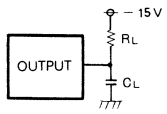
TIMING REQUIREMENTS ($Ta = 0 \sim 70°C$, $V_{DD} = -15V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted)

Symbol	Parameter	Conditions	Limits			Unit
			Min	Typ	Max	
$t_{su(D-X_{IN})}$	Data setup time before clock input, port D input	$f(\phi) = 600kHz$	0			μs
$t_{su(S-X_{IN})}$	Data setup time before clock input, port S input		0			μs
$t_{su(K-X_{IN})}$	Data setup time before clock input, port K input		0			μs
$t_{su(INT-X_{IN})}$	Data setup time before clock input, INT input		0			μs
$t_{h(D-X_{IN})}$	Data hold time after clock input, port D input		0.4			μs
$t_{h(S-X_{IN})}$	Data hold time after clock input, port S input		0.4			μs
$t_{h(K-X_{IN})}$	Data hold time after clock input, port K input		0.4			μs
$t_{h(INT-X_{IN})}$	Data hold time after clock input, INT input		0.4			μs
$t_{r(V_{DDL})}$	Supply voltage V_{DD} rise time, at the time of power-on reset				10	ms
$t_{r(V_{REFL})}$	Reference voltage V_{REF} rise time, at the time of power-on reset				10	ms
$t_{h(V_{REFH})}$	High-level reference voltage V_{REF} hold time, at the time of power-on reset		50			ms

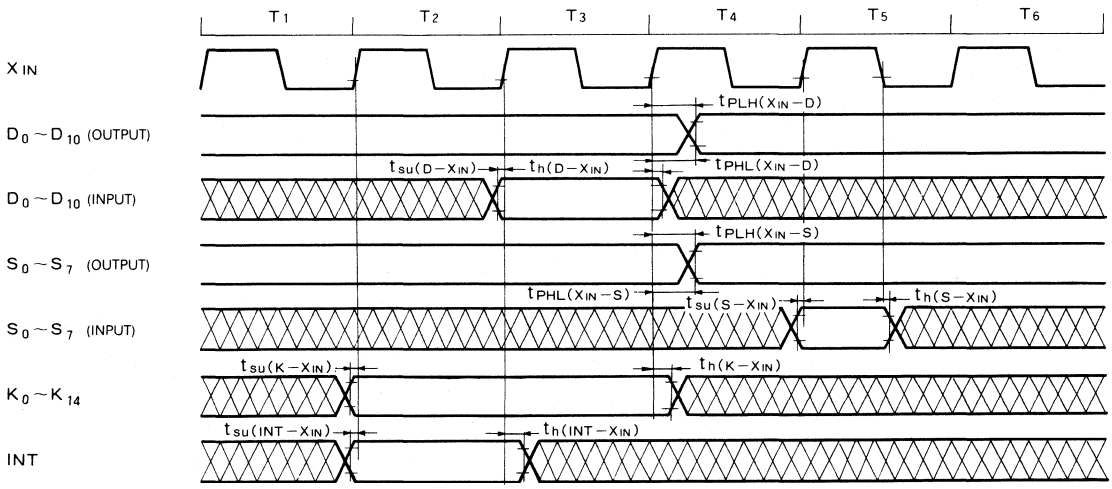
SWITCHING CHARACTERISTICS ($Ta = 0 \sim 70°C$, $V_{DD} = -15V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted)

Symbol	Parameter	Conditions	Limits			Unit
			Min	Typ	Max	
$t_{PLH(X_{IN}-D)}$	Low to high-level data output propagation time between clock input and port, port D output	$f(\phi) = 600kHz$, $C_L = 100pF$ (Note 14), $R_L = 6.8k\Omega$		0.6	1	μs
$t_{PLH(X_{IN}-S)}$	Low to high-level data output propagation time between clock input and port, port S output			0.9	1.5	μs
$t_{PHL(X_{IN}-D)}$	High to low-level data output propagation time between clock input and port, port D output				2.6	μs
$t_{PHL(X_{IN}-S)}$	High to low-level data output propagation time between clock input and port, port S output				2.6	μs

Note 14 : Measuring circuit diagram



TIMING DIAGRAM



M58842S

MELPS 4 SYSTEM EVALUATION DEVICE

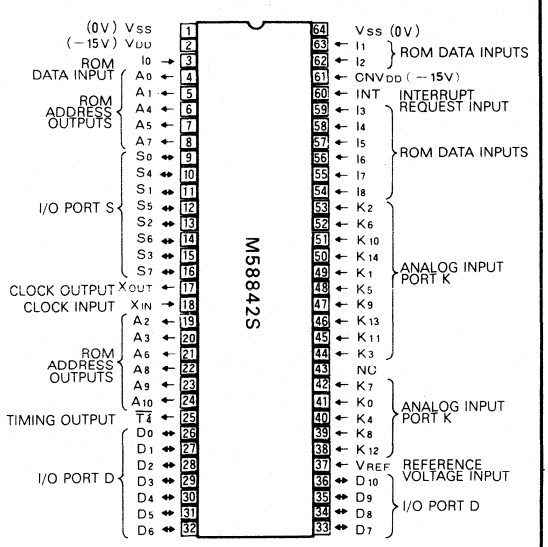
DESCRIPTION

The M58842S MELPS 4 system evaluation device is designed to emulate the M58840-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. By taking the mask-programmable ROM out of the M58840-XXXP, the M58842S facilitates fast development of new systems for the customer.

FEATURES

- Except for the mask ROM, all functions are equivalent to the M58840-XXXP.
- Large capacity: (128-word x 4-bit) RAM
- Single −15V power supply
- Built-in A/D converter with ±1.2% accuracy
- Two data pointers
- Subroutine nesting: 3 levels
- Interrupt: 1 level
- Internal clock generator
- Internal PLA (programmable logic array) for the decoder of port S: 16 inputs x 8 outputs
- Analog input port (port K): 15 bits (Convenient to accept signal from a capacitive touch panel or any analog devices.)
- ROM data input: 9 bits
- ROM address output: 11 bits
- I/O port (port S):
  - Output: 8-bits x 1
  - Input: 4-bits x 2
- I/O port (port D):
  - Output: 1-bit x 11
  - Sense input: 1-bit x 11
- I/O port output voltage (V_O): −33V (max)

PIN CONFIGURATION (TOP VIEW)



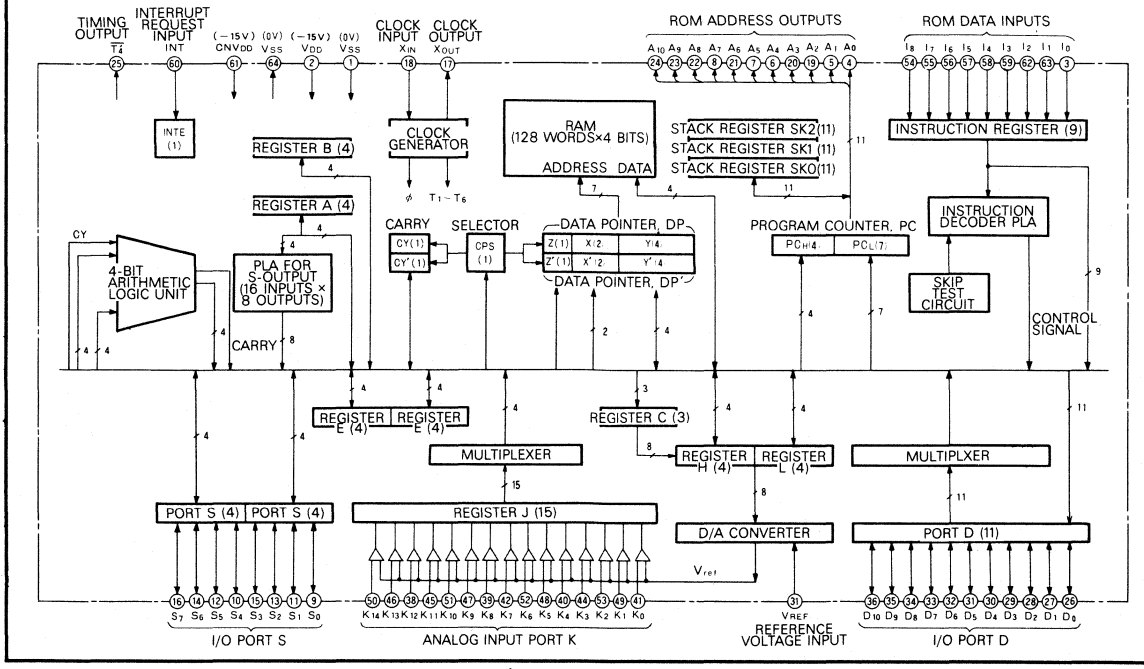
Outline 64S1

- I/O port output current:
  - I_{OH} (port S): −8mA (max)
  - I_{OH} (port D): −15mA (max)
- (Capable of direct drive of large size fluorescent display tubes)

APPLICATIONS

- System development and prototyping of equipment using the M58840-XXXP single-chip 4-bit microcomputer.

BLOCK DIAGRAM



6

MELPS 4 SYSTEM EVALUATION DEVICE

FUNCTIONS

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.

SUMMARY OF OPERATIONS

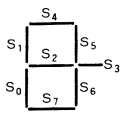
Programmable Logic Array (PLA) for the S-Output

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	0
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

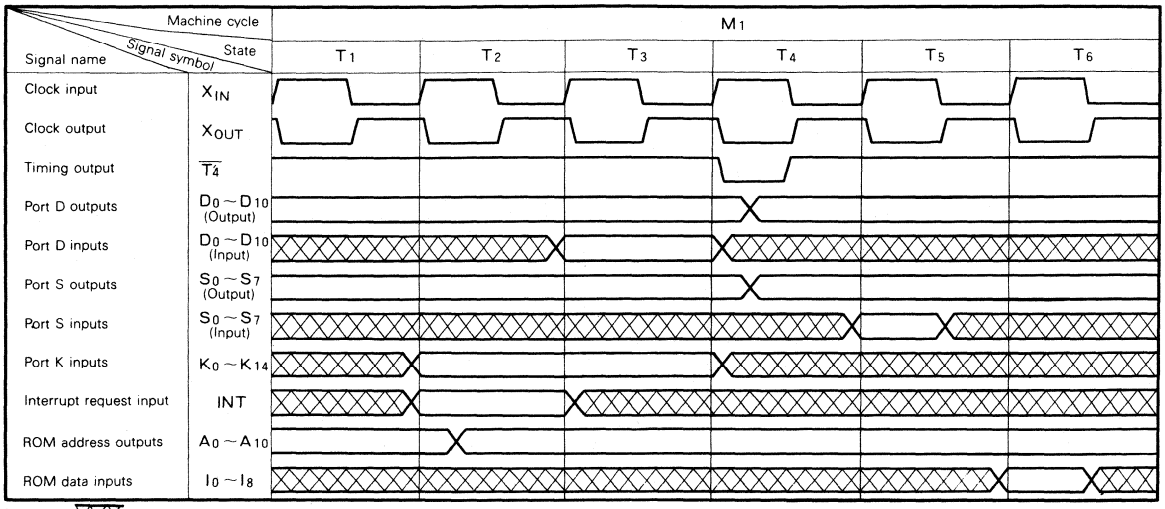
PIN DESCRIPTIONS

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 independent 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 filter 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 filter 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.
$\overline{T4}$	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.

6

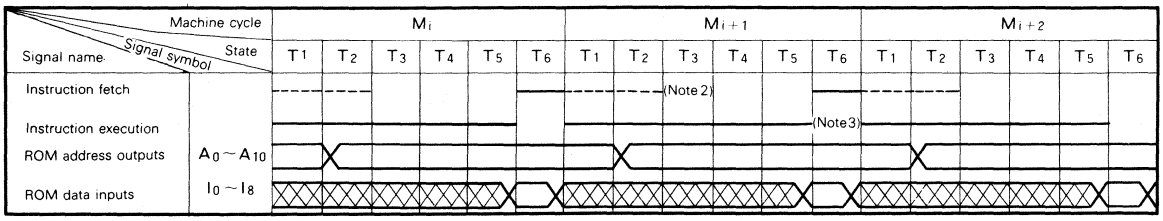
MELPS 4 SYSTEM EVALUATION DEVICE

BASIC TIMING CHART



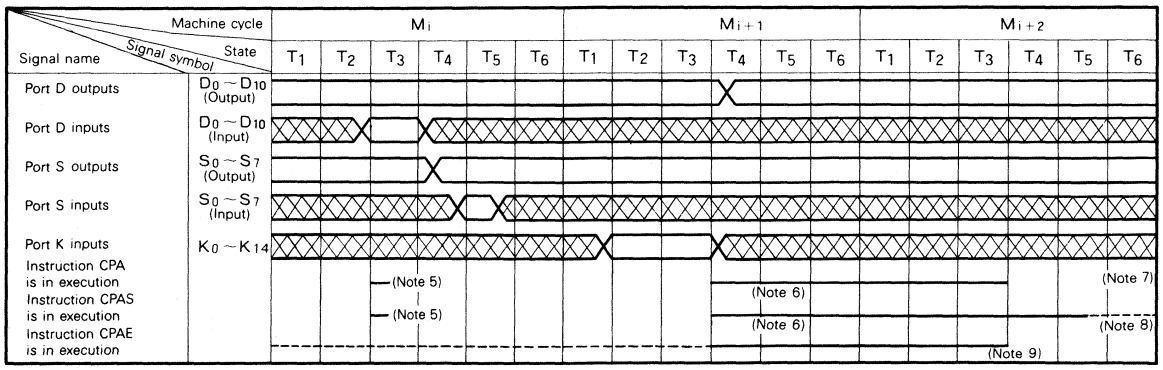
Note 1: ▨ indicates invalid signal input.

INSTRUCTION FETCH TIMING



Note 2 : Instruction fetch time can differ depending on the types of the instructions.
3 : The instruction which was fetched in the preceding cycle is executed.
4 : The execution of the instruction and addressing of ROM and RAM are performed simultaneously.

I/O INSTRUCTION EXECUTION TIMING



Note 5 : By short-circuiting port K inputs with the V_{SS} (0V), capacitance connected to the port K input is discharged.
6 : Analog value applied to port K inputs is maintained to be compared with the reference voltage V_{ref}
7 : Analog value applied to port K inputs is read until the next CPA or CPAS instruction is executed.
8 : The state of Note 6 is maintained until an instruction CPAE is executed, during which time the analog value applied to port K is not read. This time should be less than 100μs to assure the accuracy of the A/D conversion.
9 : The condition of Note 8 is released.

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	700	mW
Topr	Operating temperature		0 ~ 70	°C
Tstg	Storage temperature		−40 ~ 125	°C

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

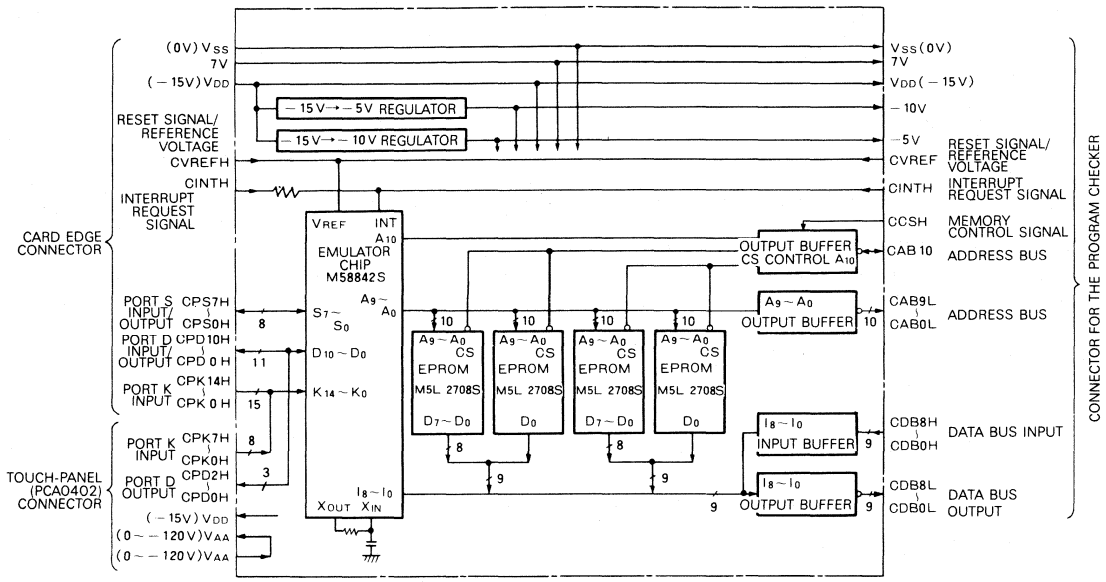
Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	−16.5	−15	−13.5	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage	−1.5		0	V
$V_{IH(\phi)}$	High-level clock input voltage	−1.5		0	V
V_{IL}	Low-level input voltage, other than port D, port S and INT inputs	V_{DD}		−4.2	V
V_{IL}	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	−7		−5	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

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 70°C, V_{DD} = −15V ± 10%, V_{SS} = 0V, $f(\phi)$ = 300 ~ 600kHz, unless otherwise noted)

Symbol	Parameter	Conditions	Limits			Unit
			Min	Typ	Max	
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			±3	LSB
	A-D conversion zero error	V_{REF} = −7V			±3	LSB
	A-D conversion full-scale error	V_{REF} = −7V			+3	LSB

MELPS 4 SYSTEM EVALUATION DEVICE

EXAMPLE OF APPLICATION



MITSUBISHI ELECTRIC

MELPS 41 MICROCOMPUTERS

7

SINGLE-CHIP 4-BIT CMOS MICROCOMPUTER

DESCRIPTION

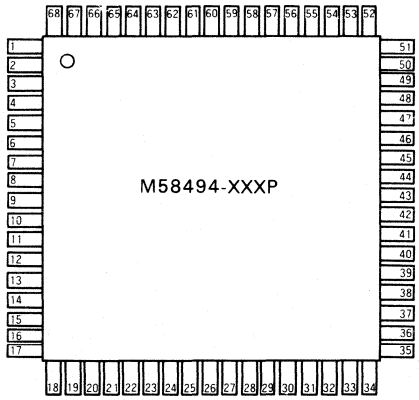
The M58494-XXXP is a single-chip 4-bit microcomputer fabricated using CMOS technology in a 68-pin plastic flat package. It has a 4096-word by 10-bit mask-programmable ROM and a 32-word by 4-bit RAM. RAM capacity can be expanded to as much as 4096 words by 4 bits by directly connecting generally available CMOS RAMs.

This device is designed for applications where the low power dissipation of CMOS is essential.

FEATURES

- Single 5V power supply
- Basic machine instructions: 92
- Basic instruction execution time (at 455 kHz clock frequency): 6.6μs
- Large memory capacity:
  - ROM: 4096-word x 10-bit
  - Internal RAM: 32-word x 4-bit
  - External RAM: 4096-word x 4-bit (max)
- Saving of last data pointer: 4-level
- Subroutine nesting: 12-level
- Internal timer:
  - Prescaled: 14-bit
  - Timer: 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, 0D): 2-bit
  - Data I/O (port D): 4-bit
- General-purpose registers: 4 x 8-bit
- I/O port (port Q): 8-bit
- I/O port (port R): 2 x 4-bit

PIN CONFIGURATION (TOP VIEW)



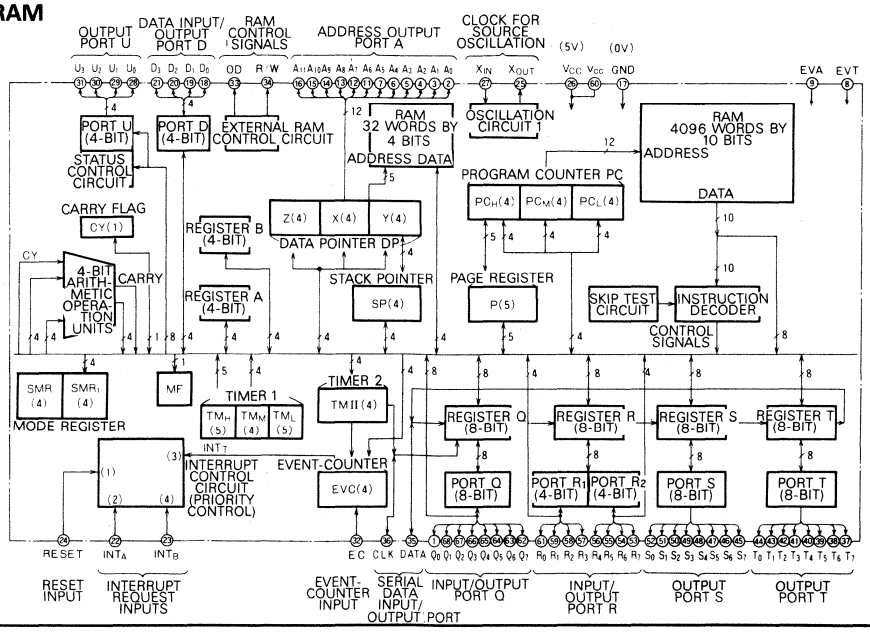
Outline 68P2

- I/O port (serial data port): 2-bit
- Output ports (port S, port T): 2 x 8-bit
- 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



SINGLE-CHIP 4-BIT CMOS MICROCOMPUTER

FUNCTION

The M58494-XXXP consists of a mask ROM and RAM, a 4-bit arithmetic logical unit, a clock generator, input/output ports, interface for a multiprocessor system, timers, an event counter and interrupt circuit. RAM capacity can be expanded by connecting CMOS RAMs externally.

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. After the execution of the instruction at address 127 of each page is completed, the page designation counter is automatically incremented, and the 7-bit binary counter is reset to zero (goes to address 0 of the next page). The return addresses from a subroutine or interrupt are saved in the 12-bit by 12-level stack registers which are reserved in the fixed area of the external RAMs. When an interrupt occurs, the jump address is fixed as follows: in the case of a reset signal, the address is reset to page 0 address 0; for the INTA signal, to page 0 address 2. For the carry signal of either the timer or the event counter it is reset to page 0 address 8, and for the INT_B signal, to page 0 address 4.

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 $\times$ 4-bit words, thus incrementing the basic external minimum RAM organization of 256 x 4-bit words.

The RAM addressing instructions, RAM-to-accumulator transfer instructions, arithmetic instructions, register-to-register transfer instructions, input and output instructions, input and output control instructions, and timer instructions are executed mainly with register A (accumulator).

RAM contains general registers of 32 bits for use by the arithmetic processing unit, which consists of the accumulator etc., and input/output ports. They are four 8-bit shift registers basically, and control the functional combinations of the serial input, the parallel input, the serial output and the parallel output by means of instructions. They execute the data transfer between output or input/output ports, loading the 8-bit value of the DATA field in the ROM, sending out internal serial data and receiving external serial data.

The input/output port Q consists of 8 bits. It has an 8-bit output latch and is connected to the 8-bit general-purpose register Q. Register Q is connected in parallel with registers A and B, and also with port Q for parallel data transfer, and is connected in serial with the external serial input for the serial data transfer. It can load the 8-bit data of the data field in the ROM. Thus, register Q stores the data transferred from registers A and B, the internal or external serial data, and data from the 8-bit data field in the ROM. The 8-bit input data to port Q can also be transferred to register A or B. The 8-bit data can be transferred between port Q and register Q at the same time by the input/output instructions.

The input/output port R consists of 8 bits, has an 8-bit output latch and is connected to the 8-bit general-purpose register. Register R has the same function as the previously described register Q except that the serial data is read from the least significant bit of register Q. It stores the data transferred from registers A and B, serial data, and the 8-bit value of the immediate field in the ROM. An 8-bit signal applied to port R can be transferred to register B 4 bits at a time. The 8-bit data can also be transferred between port R and register R at the same time by input/output instructions.

The output port S consists of 8 bits, has an 8-bit output latch and is connected with the 8-bit general-purpose register S. Except that the serial data is read starting from the least significant bit of register R and that port S is not used for input, register S has the same configuration as that of register Q described above. It stores data transferred from registers A and B, and also serial data and the 8-bit value of the data field in the ROM. By use of an input/output instructions, port S and register S can transfer 8-bit data in parallel.

The output port T consists of 8 bits, has an 8-bit output latch and is connected with the 8-bit general-purpose register T. Except that the serial data is read from the least significant bit of register S for transfer of serial output from the port data, register T has the same configuration as that of register S described above. It stores the data transferred from registers A and B, serial data, and also the 8-bit value of the data field in the ROM. By suitable input/output instructions, port T and register T can transfer the 8-bit data in parallel.

By input/output instructions, the 8-bit data of port Q, port R, port S and port T can be transferred to each general-purpose register.

When port Q or port R is used for input or output, it is necessary to set the input or output mode by SMR instructions.

When the general-purpose register is used as a serial input or serial output shift register, the value of the data field stored in the register is shifted in the register, or '0'/'1' control data is entered as the input from the most significant

bit of the general-purpose register Q for shift control, and data is also transferred from the least significant bit of the general-purpose register T through the serial input/output port, DATA. External serial data are received at the serial input/output port DATA, and read into the most significant bit of the general-purpose register Q. Though the input/output port CLK is normally in floating status, it generates A shift clock pulse synchronized with transmitting data in the output mode, and reads the external shift clock pulse synchronized with receiving data.

Timer 1, the basic source oscillation frequency (1/3 of one machine cycle) or an external reference oscillation frequency, divided by 14, is used as a pre-scaler.

Timer 2 and the event counter consist of 4 bits each and are used as a discrete unit or in combination for multiple applications.

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.
INTA	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.
INTB	Interrupt request signal B	Input	Disable	
EC	Event counter input	Input	—	The input signal for the event counter, which programs $2^0 \sim 2^4$ 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.
$A_0 \sim A_{11}$	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 ($A_0 \sim A_7$) are not affected by this mode, since data pointers X and Z are not related to latch operation.
$D_0 \sim D_3$	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 is generated at the data input cycle of 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.
$U_0 \sim U_3$	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.
$Q_0 \sim Q_7$	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.
$R_0 \sim R_7$	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.
$S_0 \sim S_7$	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.
$T_0 \sim T_7$	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 ROM to 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 Q).
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.

7

BASIC FUNCTIONAL BLOCKS AND THEIR OPERATIONS

Program Counter PC

The program counter consists of 12 bits, the upper 5 bits of which compose the page register, and the lower 7 bits the binary counter. It designates the address of the 4096 words of 10 bits mask-programmable ROM, and controls the read-out sequence of instructions stored in the ROM. 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. In the page register of the program counter, the contents of register P are loaded by instructions BL, BA, BM, and BMA. The binary counter is incremented as every single instruction is executed, but in the execution of instructions B, BL and BM, the value of the data field of the ROM is loaded, becoming the value of the specified address. In the execution of instructions BA and BMA, the contents of register A are loaded in the upper 4 bits of the binary counter, and the value of the data field is loaded in the lower 3 bits. Thereby a multi-branch instruction modified by the contents of the A register can be carried out.

There are three instructions relating to register P: instruction LP loads the value (5 bits) of the data field of the ROM, and instructions TPAC and TACP transfer data between register A and carry flag CY.

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. 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

INT_T is the interrupt request signal from timer II and the event counter.

Instruction BMAB is provided for easy handling of data conversion or using ROM as data tables, and usually by application of this instruction in combination with the instruction OPI. In two machine cycles it can load the 8-bit value of the data field of the ROM addressed by the contents of registers A and B into an arbitrary general-purpose register (Q, R, S and T).

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.

Instruction OPI loads one of the four general-purpose registers selected by the input/output address r with the value (8 bits) of the data field. The input/output address r 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. To enable independent use of instruction OPI, the input/output address r, along with the carry flag CY and the mode flag MF, gives and takes data to and from the register by instructions TACM and TCMA. Thereby data can be saved and returned at interrupt time.

Table 1 Relationship between input/output address r and general-purpose registers

Input/output address	Immediate data: r 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

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 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 addresses 16 files, each of which comprises 16 words. By making an exclusive OR with 2 bits of the data field in the execution of instructions TAM, XAM, XAMD, and XAMI, the lower 2 bits can modify the next file designation. Register Y addresses data of 16 files (a file comprises 16 words), being incremented and decremented by the arithmetic unit in the execution of the instructions INY, DEY, XAMI, XAMD, TSMI, and TMSI. When the contents become an O or F which is the boundary of a file, control skips execution of the next instruction. 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. Address signals $A_0 \sim A_{11}$ for the external main memory are output from the contents of DP in the MM mode, except in case of interruption or the execution of instructions BM, BMA, RT, RTS, RTI, SDP, and LDP. In the SM mode the address signals, except for $A_0 \sim A_7$, are output from DP after latching the contents of the data pointer to the auxiliary output latch, but prior to changing the mode. During an interrupt or execution of instructions BM, BMA, RT, RTS, RTI, SDP or LDP, the partial address is secured independent of the mode. This is designated by Z=0 (external basic main memory), X=D, E or F where Y=the contents of the stack pointer SP of the program counter PC, or the value of the data field (indicated save level of the data pointer), as shown in the following figure. When the data pointer stack is not used, all of the stack region can be used as program counter stack for a total of 16 levels.

Fig. 1 External basic main memory (Z=0) and RAM map

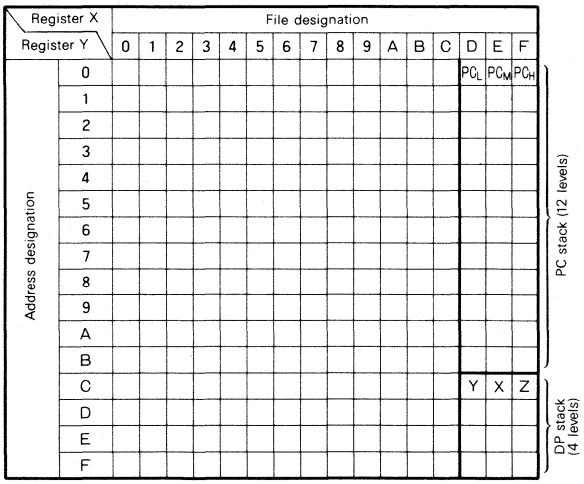


Table 2 Address designation of data pointer stack

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

When the internal scratch-pad memory is addressed in the SM mode, only five bits (four bits of register Y plus the least significant bit of register X) are employed.

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.

7

Timers and Event Counter

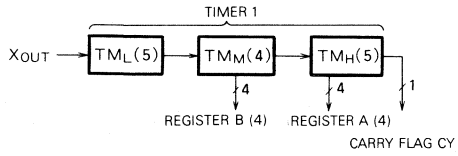
This block is composed of a 14-bit timer 1, a 4-bit timer 2 and a 4-bit event counter.

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:

(Fundamental output frequency X_{IN}) x 2^5 (TM_L) x 2^4 (TM_M) x 2^5 (TM_H) = 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 RTM clears the contents of timer 1 and resets it to 0.

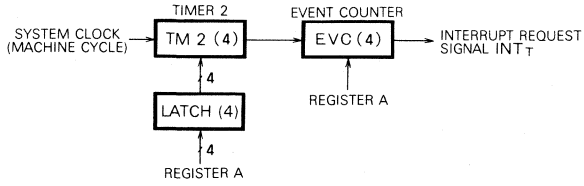
Fig. 2 Outline of timer 1 configuration



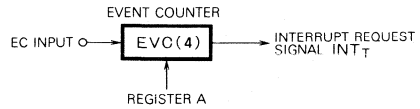
Timer 2 is composed of a 4-bit counter and a 4-bit latch. The contents of register A are stored as the starting value in the latch and the counter by an STM instruction, whereupon counting down starts in synchronization with each machine cycle. When the contents of the counter become F during countdown, the pre-programmed starting value is restored in the counter from the latch.

Fig. 3 Outline configuration of timer 2 and event counter

(1) Timer mode: When TMM=1 is set by the instruction SMR1



(2) Event mode: When TMM=0 is set by the instruction SMR1



The cycle period of timer 2 is given by the following formula:

Machine cycle x [1 + (2^0~2^4)]

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 + (2^0~2^4) x (2^0~2^4)]

Event mode

EC input period x (2^0~2^4)

The recurrence period of the shift clock pulse CLK which is generated in synchronization with data transmission for series data transmission is given by the period programmed by timer 2. The CLK output is set by the mode SDM=1, RVM=0 of instruction SMR1. It is generated with a period determined by the contents of the latch of timer 2 and by the execution of the instructions SST and RST.

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.

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. The shift instructions comprise instruction SST, which shifts 32 bits of data by setting the shift register input to '1'; instruction RST, which shifts 32 bits of data and resets the input to '0'; and instruction IST, which shifts the data by reading the data of the serial data input data. Instruction IST, except in the mode where SDM=0 and RVM=1, reads the serial data of the DATA terminal (which is used as the input or the output terminal). In the mode where SDM=0 and RVM=1, it reads the data at the rising edge of the clock input CLK and shifts the data. Note that it cannot respond to data input that has a transfer rate faster than the machine cycle rate, since detection of the rising edge of CLK is carried out by using the internal clock pulse. In the mode where SDM=1 and RVM=0, data is output synchronized with the clock pulse output CLK generated by the period programmed in timer 2 (from the least significant bits of the general-purpose register). In the mode where SDM=1 and RVM=1, data is output synchronized with the shift instruction and is transmitted from the least significant bit of the general-purpose register.

Instruction TNAB stores 8-bit data from registers A and B in one of the general-purpose registers designated by the input/output address r, which is defined by 2 bits of the data field.

Instruction TABN transfers 8-bit data from one of the general-purpose registers designated by the input/output address r, which is defined by 2 bits of the data field into A and B registers.

Instruction OPI stores the 8-bit value of the data field in one of the general-purpose registers designated by the input/output address r by using it in combination with instruction BMAB or by using instruction OPI alone, as described in the explanation of the program counter function.

Instruction TPRN stores the contents of the register designated by the input/output address r in the latch of the output port corresponding to the input/output address r, which is then usable.

Instruction TRPA stores the contents of the 32-bit register in corresponding latches of all the output ports. These are then valid at the outputs.

Instruction TRPN can restore from the contents of the output latch port designated by the input/output address r into the register corresponding to the input/output address r.

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 order of priority is determined as follows:

RESET > INT_A > INT_T > INT_B

The interrupt enable instructions comprise EIA, EIB, EIAB and EIT and the interrupt disable instructions comprise DIA, DIB, DIAB and DIT. 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 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 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 terminal		Input	Input	Output	Output
CLK terminal		Floating	Input (rising edge trigger)	Output (generated by timer 2)	Output (generated by shift instruction)
Shift data input	SST, RST	Immediate field data	DATA terminal input	Immediate field data	Immediate field data
	IST	DATA terminal outpt		DATA terminal output	DATA terminal 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)	Receiving	Transmission	Transmission

Mode Register

The mode register is composed of 10 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 5.

The mode flag IMQ will select the use of the input/output port Q for input or output.

The mode flags IMR_1 and IMR_2 select the use of the input/output port R for input or for output in 4-bit unit.

The mode flag LCD will select the instruction that activates output port U. In case of status '0', only instruction SU is valid. In case of status '1', port U can be set by instruction SU and by instructions TPRN and TPRA, which move the contents of the general-purpose register to the output port. Every bit of port U, as shown in Table 4, can be programmed for three states as determined by the contents of registers A and B. For example, by using instruction TPRA in LCD=1, we can drive a 1/2-duty liquid crystal display panel by the 1/2 voltage equalization method, where port U is used for the common output and the ports Q, R, S and T for the segment outputs.

The mode setting by instruction SMR1 is shown in Table 6.

The mode flag TMM determines whether the event counter is in the independent event mode or in the timer mode by connecting with timer 2.

The mode flag determines whether all the three-state signals for the external main memory (RAM), $A_0 \sim D_{11}$, $D_0 \sim D_3$, 0D and R/W, are in floating or activated state.

The mode flags RVM and SDM select the functions of the terminals DATA and CLK, which are the transmission/receiving and input/output ports, when the 32-bit general-purpose register is used as a shift register. For further details, refer to the explanation of the general-purpose register.

Table 4 Three-state-condition setting of output port U

Contents of register B	Contents of register A	State of port U (when executing SU, TPRN or TRRA)
0	0	Floating
0	1	Floating
1	0	0
1	1	1

Note 1 : Registers A, B, and port U correspond to one another in regard to ther bit order.

Table 5 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	Port Q is used as an 8-bit output port.	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 6 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 ($A_{11} \sim A_0$, $D_3 \sim D_0$, OD and R/W) for external main memory (RAM) are put in floating.	0
		1	All signals ($A_{11} \sim A_0$, $D_3 \sim D_0$, 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		

Outline Specifications of M58494-XXXP

Item			Performance
Number of basic instructions			92
Execution time of basic instructions			6.6μs (at V_{CC}=5V, f=455kHz)
Clock frequency			200~455kHz
Memory capacity	ROM		4096 words by 10 bits
	RAM (built-in)		32 words by 4 bits
	RAM (external)		4096 words by 4 bits (max.)
Input/output port for external RAM	Address (port A)		12 bits × 1 (3 states)
	Control signal (port OD and R/W)		2 bits (3 states)
	Data bus (port D)		4 bits × 1 (3 states)
Input/output port	Q	Input	8 bits × 1
		Output	8 bits × 1
	R	Input	4 bits × 2
		Output	8 bits × 1
	S	Output	8 bits × 1
	T	Output	8 bits × 1
	DATA	Serial data	1 bit (input/output port)
	CLK	Synchronizing pulse	1 bit (input/output port)
	U	Output	4 bit × 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 characteristics		Interchangeable with CMOS logic series
Power supply voltage	V_{CC}		5V (nominal)
	V_{SS}		0V
Element structure			CMOS
Package			68-pin plastic molded flat package
Power dissipation			5 mW (at V_{CC}=5V, f=455kHz)

Note 2 : Ports to be used will be determined in accordance with user's specifications.

7

SINGLE-CHIP 4-BIT CMOS MICROCOMPUTER

MACHINE INSTRUCTIONS

Classification	Symbol	Code I9 I8 I7 I6 I5 I4 I3 I2 I1 I0	16mal notation	No. of words	No. of cycle	Function	Skip conditions	Flag CY
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 = 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)←($\overline{A}$)	—	—

MITSUBISHI ELECTRIC

Item / Classification	Symbol	Code I_9I_8 $I_7I_6I_5I_4$ $I_3I_2I_1I_0$	16mal notation	No. of words	No. of cycle	Function	Skip conditions	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	$(PC_L)\leftarrow y$, $(PC_M)\leftarrow x$ where 16x+y=0~127	—	—
	BL xy	11 0xxx yyyy	3xy	1	1	$(PC_L)\leftarrow y$, $(PC_M)\leftarrow(P_0, x)$, $(PC_H)\leftarrow(P_4, P_3, P_2, P_1)$ where 16x+y= 0 ~127	—	—
	BA i	00 1101 0iii	0Di	1	1	$(PC_L)\leftarrow(A_0, i)$ where, i= 0 ~ 7; $(PC_M)\leftarrow(P_0, A_3, A_2, A_1)$; $(PC_H)\leftarrow(P_4, P_3, P_2, P_1)$	—	—
	BMAB r	00 1100 10rr	0C8 + r	1	1	$(PC_L)\leftarrow(A)$; $(PC_M)\leftarrow(B)$; $(PC_H)\leftarrow(P_4, P_3, P_2, P_1)$ but returns unconditionally after one machine cycle. Input/output address r = 0~3 designates general-purpose register.	—	—
	LP p	01 110p pppp	1C0 ++ pp	1	1	(P)←p where memory page number (p = 0, 1, 2, 3, 4, 5.....30, 31) corresponds to decimal notation 0, 16, 1, 17, 3, 18.....15, 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	$(PC_L)\leftarrow y$; $(PC_M)\leftarrow(P_0, x)$, where 16x + y = 0~127; $(PC_H)\leftarrow(P_4, P_3, P_2, P_1)$; (M(SP))←(PC); (SP)←(SP)+ 1	—	—
	BMA i	00 1101 1iii	0D8 + j	1	3	$(PC_L)\leftarrow(A_0, i)$, where, j=0 ~ 7; $(PC_M)\leftarrow(P_0, A_3, A_2, A_1)$; $(PC_H)\leftarrow(P_4, P_3, P_2, P_1)$; (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; Resets the interrupt control.	—	—

SINGLE-CHIP 4-BIT CMOS MICROCOMPUTER

Item / Classification	Symbol	Code $I_9 I_8 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0$	16mal notation	No. of words	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	$(TMII) \leftarrow (A)$	—	—
	SEC	00 1100 0110	0C6	1	1	$(EVC) \leftarrow (A)$	—	—
Input/output	**ID**	00 0010 1110	02E	1	1	$(B) \leftarrow (D)$, $(OD) \leftarrow$ low level	—	—
	OD	00 0100 1100	04C	1	1	$(D) \leftarrow (B)$, $(R/W) \leftarrow$ low level	—	—
	OPI s	10 ssss ssss	2ss	1	1	$(R(r)) \leftarrow s$ Where the general-purpose register is designated with r = 0~3.	—	—
	TNAB r	00 0100 10rr	048 + r	1	1	$(R(r)) \leftarrow (A, B)$ Where the general-purpose register is designated with r = 0~3.	—	—
	TABN r	00 0010 10rr	028 + r	1	1	$(A, B) \leftarrow (R(r))$ Where the general-purpose register is designated with r = 0~3.	—	—
	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	$(U) \leftarrow (A)$, when $(B)_i = 1$, $(U) \leftarrow$ floating, when $(B)_i = 0$	—	—
	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	—	—

MITSUBISHI ELECTRIC

Symbol	Details
A	4-bit register (accumlator)
Ai	Indicates the bits of register A. Where i = 0~3.
B	4-bit auxiliary register
B(j)	The bit of register B. addressed when j = 0~3.
CY	1-bit carry flag
D	4-bit input/output port (3-state)
DATA	1-bit input/output port for serial data
DP	12-bit data pointer composed of registers X, Y and Z
EVC	4-bit event counter
M(DP)	4-bit data memory addressed by the data pointer DP
M_I	12-bit data from the scratch-pad memory addressed by i = 0~3 (data pointer number in the fixed area)
M_i(DP)	4-bit data from external memory addressed by the contents data pointer DP, where i = 0~3.
MF	1-bit flag for selection of internal scratch-pad memory (MF ← 0 at instruction SM) or external main memory (MF ← 1 at instruction MM)
MM(DP)	4-bit external main memory data addressed by the data pointer DP
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, IMRI, IMR2)
MR1	4-bit mode flag (TMM, BF, RVM, SDM)
OD	1-bit output port used for the read signal for external main memory.
P	5-bit page register
Pi	Indicates the bits of register P, where i = 0~4.
PC	12-bit program counter composed of counters PC_L, PC_M and PC_H
PC_H	4-bit counter
PC_L	4-bit counter
PC_M	4-bit counter
P(All)	Indicates all the 8-bit ports, Q, R, S, T (32-bit)
P(r)	The port selected by r (corresponds with ports Q, R, S, and T at r = 0~3)

Symbol	Details
P(Q)	8-bit port Q
$P(R_1)$	4-bit port R_1
$P(R_2)$	4-bit port R_2
R(All)	Indicates all the 8-bit registers, Q, R, S, T (32-bit)
$R(Q_0)$	1st bit of register Q
R(r)	The register selected by r (r corresponds with registers Q, R, S, and T where r = 0~3)
R/W	1-bit output port which is used for the write signal of the external main memory
SM(DP)	The 4-bit internal scratch-pad memory addressed by the data pointer DP
SP	4-bit stack pointer
TM1	14-bit counter composed of TM_L, TM_M and TM_H counters
TM2	4-bit counter
TM_H	5-bit counter
TM_{Hi}	Indicates the bit of TM_H counter, where i = 0~4
TML	5-bit counter
TM_M	4-bit counter
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.
Y	4-bit register where Y = 0~15, which addresses the word unit of 16 words by 4 bits.
Z	4-bit register where Z = 0~15, which addresses 16 files × 16 words × 4 bits
iii	3-bit binary variable
jj	2-bit binary constant
nnnn	4-bit binary constant
ppppp	5-bit binary constant
r	Input/output address to select one of the general-purpose registers Q, R, S and T. (r = 0~3)
rr	2-bit binary constant
ssss ssss	8-bit binary constant
xxxx	4-bit binary variable
yyyy	4-bit binary variable
zzzz	4-bit binary variable

7

INSTRUCTION CODE LIST

I_9~I_4 / 16mal notation / I_3~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 3 : I_3~I_0 indicate the low-order 4 bits of the machine code and I_9~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.

MELPS 8/85 MICROPROCESSORS

8

M5L 8080A P, S

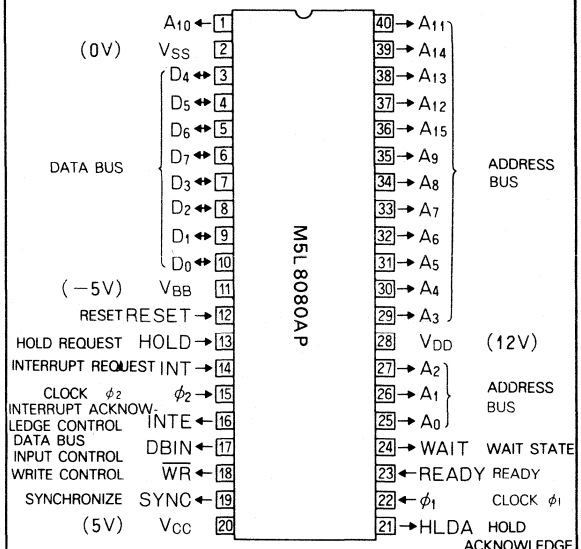
8-BIT PARALLEL CPU

The M5L 8080A P, S is an 8-bit parallel central processing unit (CPU) fabricated on a single chip using a high-speed N-channel silicon-gate MOS process,in a ceramic DIL package.

FEATURES

- Basic machine instructions: 78
- Execution time (at clock frequency 2MHz): 2μs
- Directly accessible memory capacity: 65 536 bytes
- Number of input/output ports: 256 each
- Multi-level interruption
- Direct memory access (DMA) operation
- All outputs are fully TTL-compatible; I_{OL} = 1.9mA
- Unlimited subroutine nesting
- Interchangeable with the Intel's 8080A in pin-to-pin connections and machine instructions

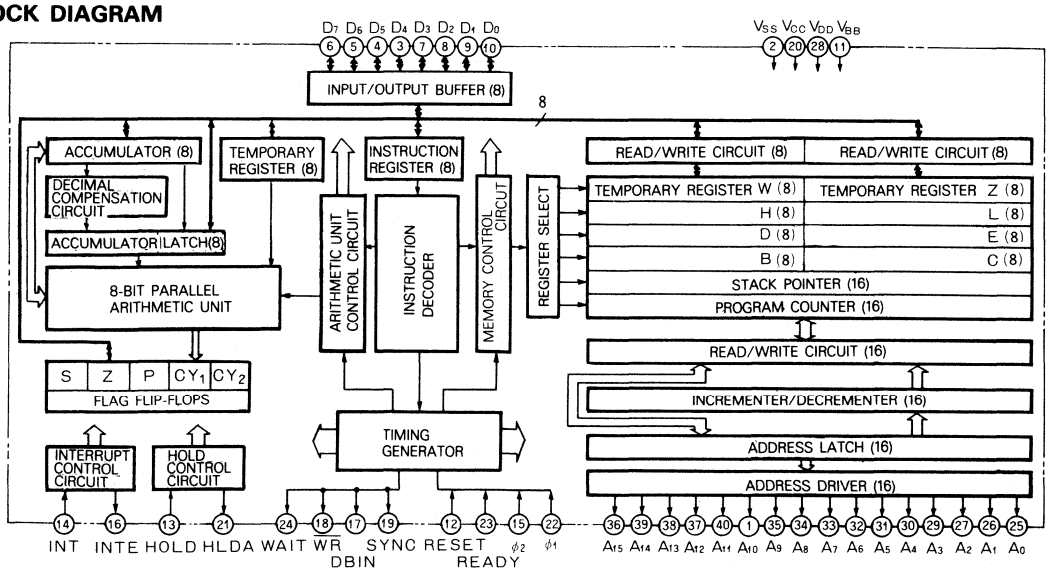
PIN CONFIGURATION (TOP VIEW)



Outline 40P1 (M5L 8080AP)
40S1 (M5L 8080AS)

8

BLOCK DIAGRAM



8-BIT PARALLEL CPU

PIN DESCRIPTIONS

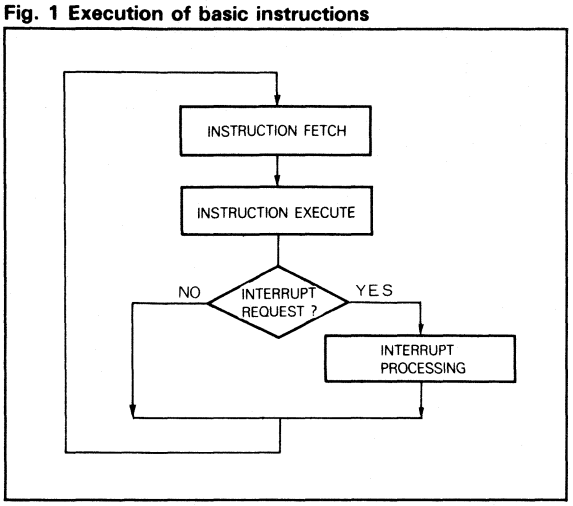
Pin	Name	Input or output	Function significance
A_0 ⟩ A_{15}	Address bus	Out	Provides the address signal to external memory up to 65 536 bytes or denotes the I/O device number for up to 256 input and 256 output devices. Address terminals are three-state, and remain in the floating state during the HLT instruction execute cycle T_{WH} or in the hold state.
D_0 ⟩ D_7	Data bus	In/Out	Provides bidirectional transfer of instructions and data between CPU and the external memory or the I/O ports. Status signals are also transferred. When $\overline{WR}$ is low, data goes to memory or output ports. When DBIN is high, the data is received by the CPU. The status signals are sent on the data bus, synchronizing with SYNC. Like the address bus, the data bus maintains the floating state during the HLT instruction execute cycle (T_{WH}) and in the hold state.
SYNC	Synchronizing signal	Out	Indicates the beginning of machine cycles M_1 through M_5. The status signals for each cycle are sent out on the data bus while SYNC is active, and are latched in the external registers during SYNC.
DBIN	Data bus input control signal	Out	Indicates to the external circuits that the data bus is in the input mode, in which the CPU receives instructions or data from memory or input ports. The CPU receives instructions or data on the data bus when DBIN is high.
READY	Ready signal	In	Indicates to the CPU that data from memory or input/output ports is valid on the data bus. When the READY signal is not high in the T_2 state, the CPU will enter a waiting state (T_W) and the WAIT signal goes high. When READY is high, its state advances from T_2 or T_W to T_3. This READY signal is used to synchronize the CPU with slower memory or input/output ports.
WAIT	Wait state signal	Out	Indicates that the CPU has entered a waiting state. When the WAIT signal is high, the CPU is in a waiting state (T_W) and the output on the address bus and the data bus is kept stable.
$\overline{WR}$	Write control signal	Out	Indicates timing of a data write-in operation to memory or output ports. When $\overline{WR}$ is low, data on the data bus is valid; when the WAIT signal is high, it is kept low.
HOLD	Hold request signal	In	When READY is high, the CPU enters the hold state provided that: · the CPU is in the HLD instruction execute state (T_{WH}), · the CPU is in the T_2 or T_W state and the READY signal is high. When the CPU is in the hold state, the data bus and the address bus will be in the floating state, and will be used with the memory or input/output ports regardless of CPU operation.
HLDA	Hold acknowledge signal	Out	When high, indicates that the CPU is in the hold state and the address bus and the data bus will be in the floating state.
INTE	Interrupt enable control signal	Out	When high, indicates that an interruption will be accepted by the CPU. It is set to high by instruction EI and is reset to low by instruction DI. It is automatically reset to low at state T_1 of machine cycle M_1 when an interrupt is accepted, and is also reset by the RESET signal.
INT	Interrupt request signal	In	Indicates to the CPU M5L 8080A P that an interrupt is being requested. When the INT is high, the interrupt request will be accepted by the CPU unless HLDA is high or INTE is low. If INT is accepted, INTE will go low and status information INTA will be transferred to the data bus as an interrupt request signal.
RESET	Reset signal	In	When high, the program counter is reset to '0' and instruction NOP is set to the instruction register. INTE is reset to low, and the CPU will not accept interrupts. While RESET is high, the address bus and the data bus remain in the floating state; when RESET goes low, the program will start at location 0. The data registers, accumulator, stack pointer and flag flip-flops are not reset by this signal. No synchronization is necessary for the RESET signal, but the high level must be kept for a minimum of 3 clock cycles.

BASIC TIMING

Execution of instructions proceeds in two stages: 1) fetch, and 2) analyze and execute.

Fig. 1 shows the consecutive relationship between stages 1 and 2, after which it is determined whether or not there has been an interrupt request. If there has not, the next instruction is fetched immediately; if there has, it is fetched after completing the necessary interrupt processing. One cycle of this loop completes the execution of one instruction.

Fig. 1 Execution of basic instructions



There are five machine cycles (M_1, M_2, M_3, M_4 and M_5) and the fetching, analysis, and execution of a single instruction requires from 1 to 5 machine cycles.

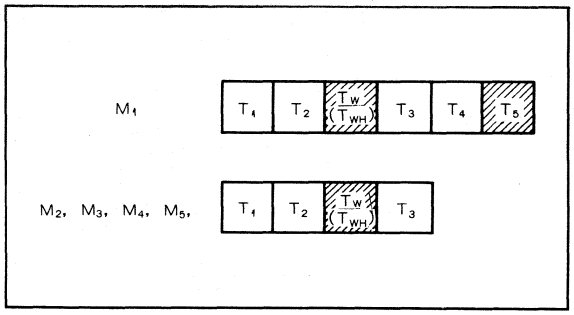
Each cycle consists of from three to five states (T_1, T_2, T_3, T_4 and T_5), the actual number depending upon the instruction being executed. The duration of one state is defined by successive low-to-high transitions of the ϕ_1 clock. (500ns at a clock frequency of 2MHz).

There is also another state, T_W, situated between T_2 and T_3 (see Fig. 2) and controlled by the external signals READY and HOLD, and instruction HLT. The duration of T_W is an integral multiple of the clock cycle.

The first machine cycle (M_1) in every instruction cycle is a fetch cycle, and the address for memory read is sent on the address bus. M_1 is composed of states $T_1 \sim T_4$ or $T_1 \sim T_5$, as shown in Fig. 2. Machine cycles M_2, M_3, M_4 and M_5 are usually composed of three states ($T_1 \sim T_3$), with the exception of the instruction XTHL, which requires five states: $T_1 \sim T_5$.

When the clock period is 500ns and there is no T_W, M_1 requires 2μs or 2.5μs, and the other machine cycles require 1.5μs to execute an instruction. When T_W exists, the execution time increases accordingly. Since the minimum instruction cycle requires four states ($T_1 \sim T_4$) of machine cycle M_1, the minimum instruction execution time is 2μs.

Fig. 2 Machine cycle states (hatched blocks indicate a state that may not be required)



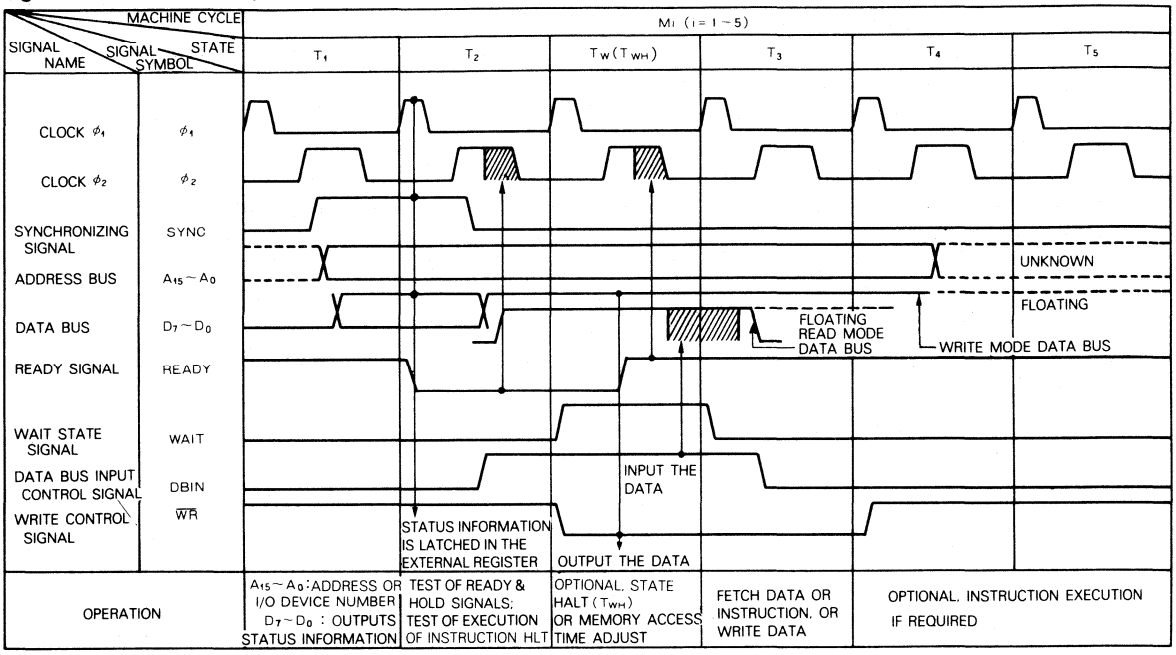
INTERRUPT

When an interruption is requested, the decision whether to accept it or not is taken after the instruction in progress is completed; that is, during the last state of the last machine cycle.

When interrupt is requested and the CPU is in the interrupt-enable state (signal INTE is high), the CPU accepts the interrupt and begins a special interrupt machine cycle M_1 in which the program counter is not incremented and the CPU sends out status information INTA (the interrupt acknowledge signal). During state T_3 of special interrupt machine cycle M_1, the external interrupt control circuit sends the interrupt instruction corresponding to interrupt factors on the data bus, and the CPU fetches and executes this instruction. This instruction is a special one-byte call (instruction RST) or a special three-byte call (instruction CALL) which facilitates the processing of interrupts.

8

Fig. 3 Basic instruction cycle



- Besides the states shown in Fig.3, there is a state T_H, in which the CPU stays in the hold state after the machine cycle.
- States T_W, T_4 and T_5 are optional.
- Hatched portions indicate periods during which input data should be kept stable.
- The address data is valid during the period designated by solid lines.
- The period of T_W depends on the condition of the READY signal.

Table 1 Status information

Data bus	Signal symbol	Status information designation	Function
D_0	INTA	Interrupt acknowledge	Goes high when the CPU accepts the interrupt request signal from the INT terminal.
D_1	$\overline{WO}$	Write mode	Goes high when the current machine cycle is in a read mode, and falls when in a write (output) mode.
D_2	STACK	Stack	Goes high when the address bus holds the pushdown stack address from the stack pointer.
D_3	HLTA	HLT instruction acknowledge	Goes high when the CPU executes the HLT instruction and maintains the halt state.
D_4	OUT	Output instruction acknowledge	Goes high when the address bus contains the address of an output device and the data bus contains the output data. (The address of an output device is contained simultaneously in the upper 8 bits and the lower 8 bits of the address bus.)
D_5	M_1	M_1 status	Goes high when the CPU is in the fetch cycle for the first byte of an instruction.
D_6	INP	Input instruction acknowledge	Goes high when the address bus contains the address of an input device and the data bus receives the input data. (The address of an output device is contained simultaneously in the upper 8 bits and the lower 8 bits of the address bus).
D_7	MEMR	Memory read	Goes high when the data bus is used for memory read data.

STATUS INFORMATION

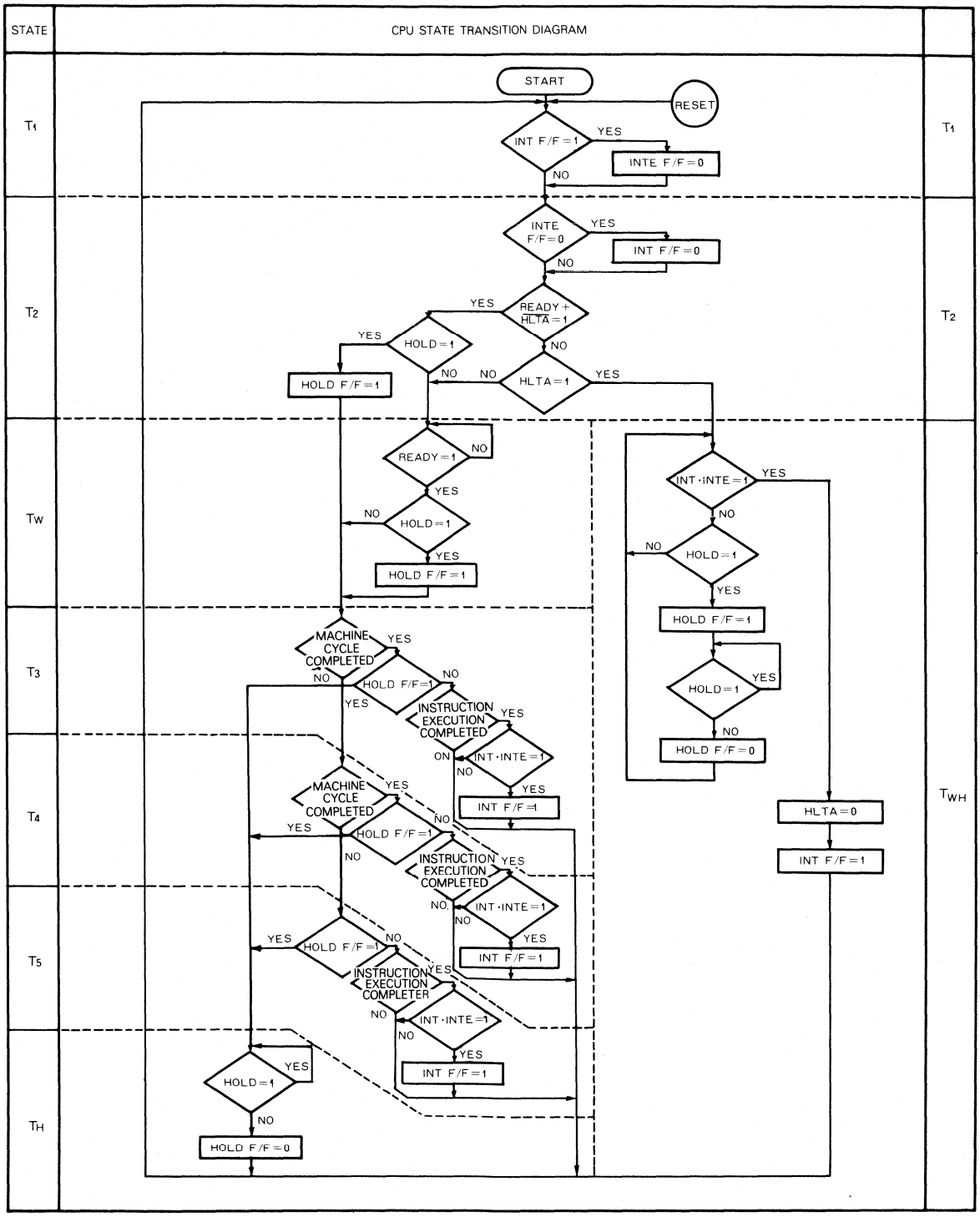
The M58710S sends out 8 bits of status information on data bus (D_7~D_0) at the first state of each machine cycle ($M_i \cdot T_1$) synchronizing with signal SYNC·ϕ_1. Table 1 gives the functions of the status information that will be sent out on the data bus.

Table 2 Status

Status information / Mode No.		1	2	3	4	5	6	7	8	9	10
Data bus bit	Status signal name	Instruction fetch	Memory read	Stack read	Input read	Interrupt acknowl.	Halt acknowledge	Interrupt acknowl. while halt	Memory write	Stack write	Output write
D_0	INTA	0	0	0	0	1	0	1	0	0	0
D_1	$\overline{WO}$	1	1	1	1	1	1	1	0	0	0
D_2	STACK	0	0	1	0	0	0	0	0	1	0
D_3	HLTA	0	0	0	0	0	1	1	0	0	0
D_4	OUT	0	0	0	0	0	0	0	0	0	1
D_5	M_1	1	0	0	0	1	0	1	0	0	0
D_6	INP	0	0	0	1	0	0	0	0	0	0
D_7	MEMR	1	1	1	0	0	1	0	0	0	0

MITSUBISHI ELECTRIC

CPU STATE TRANSITION DIAGRAM



8-BIT PARALLEL CPU

MACHINE INSTRUCTIONS

Instr. class	Mnemonic	Instruction code	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		5	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 1 1 0 <B2>		7	2	2	(r) ← n	X X X X X			<B2>	I	M4
	MVI M, n	0 0 1 1 0 1 1 0 <B2>	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 0 0 1 <B2> <B3>	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 0 0 1 <B2> <B3>	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 0 0 1 <B2> <B3>	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 0 0 1 <B2> <B3>	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	5	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 0 1 0 <B2> <B3>	3 2	13	3	4	(m) ← (A)	X X X X X	m	M4	(A)	O	M4
	LDA m	0 0 1 1 1 0 1 0 <B2> <B3>	3 A	13	3	4	(A) ← (m)	X X X X X	m	M4	(m)	I	M4
	SHLD m	0 0 1 0 0 0 1 0 <B2> <B3>	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 0 1 0 <B2> <B3>	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	18	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 1 1 0 <B2>	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 1 1 0 <B2>	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 1 1 0 <B2>	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 1 1 0 <B2>	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 O					
	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 O	M	M4	(M)	I	M4
	ANI n	1 1 1 0 0 1 1 0 <B2>	E 6	7	2	2	(A) ← (A) ∧ n	O O O 0 O			<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 1 1 0 <B2>	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 1 1 0 <B2>	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 1 1 0 <B2>	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		5	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		5	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	5	1	1	(B) (C) ← (B) (C) + 1	X X X X X					
	INX D	0 0 0 1 0 0 1 1	1 3	5	1	1	(D) (E) ← (D) (E) + 1	X X X X X					
	INX H	0 0 1 0 0 0 1 1	2 3	5	1	1	(H) (L) ← (H) (L) + 1	X X X X X					
	INX SP	0 0 1 1 0 0 1 1	3 3	5	1	1	(SP) ← (SP) + 1	X X X X X					
	DCX B	0 0 0 0 1 0 1 1	0 B	5	1	1	(B) (C) ← (B) (C) − 1	X X X X X					
	DCX D	0 0 0 1 1 0 1 1	1 B	5	1	1	(D) (E) ← (D) (E) − 1	X X X X X					
	DCX H	0 0 1 0 1 0 1 1	2 B	5	1	1	(H) (L) ← (H) (L) − 1	X X X X X					
	DCX SP	0 0 1 1 1 0 1 1	3 B	5	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) ← (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.

MITSUBISHI ELECTRIC

Instr. class	Mnemonic	Instruction code D7 D6 D5 D4 D3 D2 D1 D0	Hexadecimal notation	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 0 1 1 <B2> <B3>	C3	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	E9	5	1	1	(PC)←(H)(L)	X X X X X					
	JC m	1 1 0 1 1 0 1 0 <B2> <B3>	DA	10	3	3	(CY2)=1	X X X X X	If condition is true.				
	JNC m	1 1 0 1 0 0 1 0 <B2> <B3>	D2	10	3	3	(CY2)=0 If condition is true (PC)←m	X X X X X					
	JZ m	1 1 0 0 1 0 1 0 <B2> <B3>	CA	10	3	3	(Z)=1	X X X X X			<B2> <B3>	I I	M2 M3
	JNZ m	1 1 0 0 0 0 1 0 <B2> <B3>	C2	10	3	3	(Z)=0	X X X X X					
	JP m	1 1 1 1 0 0 1 0 <B2> <B3>	F2	10	3	3	(S)=0 If condition is false (PC)←(PC)+3	X X X X X	If condition is false.				
	JM m	1 1 1 1 1 0 1 0 <B2> <B3>	FA	10	3	3	(S)=1	X X X X X					
	JPE m	1 1 1 0 1 0 1 0 <B2> <B3>	EA	10	3	3	(P)=1	X X X X X					
	JPO m	1 1 1 0 0 0 1 0 <B2> <B3>	E2	10	3	3	(P)=0	X X X X X					
Subroutine call	CALL m	1 1 0 0 1 1 0 1 <B2> <B3>	CD	17	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⟩15-8 ⟨(PC)+3⟩7-0	I I O O	M2 M3 M4 M5
	RST n	1 1 A A A 1 1 1		11	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⟩15-8 ⟨(PC)+1⟩7-0	O O	M4 M5
	CC m	1 1 0 1 1 1 0 0 <B2> <B3>	DC	17/11	3	5/3	(CY2)=1	X X X X X	If condition is true.				
	CNC m	1 1 0 1 0 1 0 0 <B2> <B3>	D4	17/11	3	5/3	(CY2)=0	X X X X X					
	CZ m	1 1 0 0 1 1 0 0 <B2> <B3>	CC	17/11	3	5/3	(Z)=1 ((SP)−1)((SP)−2)←(PC)+3 (PC)←m	X X X X X			<B2> <B3>	I I	M2 M3
	CNZ m	1 1 0 0 0 1 0 0 <B2> <B3>	C4	17/11	3	5/3	(Z)=0 (SP)←(SP)−2	X X X X X	(SP)−1 (SP)−2	M4 M5	⟨(PC)+3⟩15-8 ⟨(PC)+3⟩7-0	O O	M4 M5
	CP m	1 1 1 1 0 1 0 0 <B2> <B3>	F4	17/11	3	5/3	(S)=0	X X X X X					
	CM m	1 1 1 1 1 1 0 0 <B2> <B3>	FC	17/11	3	5/3	(S)=1 If condition is false (PC)←(PC)+3	X X X X X					
	CPE m	1 1 1 0 1 1 0 0 <B2> <B3>	EC	17/11	3	5/3	(P)=1	X X X X X					
	CPO m	1 1 1 0 0 1 0 0 <B2> <B3>	E4	17/11	3	5/3	(P)=0	X X X X X					
Return	RET	1 1 0 0 1 0 0 1	C9	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	D8	11/5	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	D0	11/5	1	3/1	(CY2)=0	X X X X X					
	RZ	1 1 0 0 1 0 0 0	C8	11/5	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	C0	11/5	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	F0	11/5	1	3/1	(S)=0	X X X X X					
	RM	1 1 1 1 1 0 0 0	F8	11/5	1	3/1	(S)=1 If condition is false	X X X X X	If condition is false.				
	RPE	1 1 1 0 1 0 0 0	E8	11/5	1	3/1	(P)=1 (PC)←(PC)+1	X X X X X					
	RPO	1 1 1 0 0 0 0 0	E0	11/5	1	3/1	(P)=0	X X X X X					
Input/output control	IN n	1 1 0 1 1 0 1 1 <B2>	DB	10	2	3	(A)←(Input buffer)←(Input device of number n). (Input data)	X X X X X			<B2>	O	M4
	OUT n	1 1 0 1 0 0 1 1 <B2>	D3	10	2	3	(Output device of number n)←(A)	X X X X X	<B2> B2>	M5	(Input data) <B2>	I O	M5 M4
									<B2> <B2>	M5	(A)	O	M5
Interrupt control	EI	1 1 1 1 1 0 1 1	FB	4	1	1	(INTE)←1	X X X X X					
	DI	1 1 1 1 0 0 1 1	F3	4	1	1	(INTE)←0	X X X X X					
Stack control	PUSH PSW	1 1 1 1 0 1 0 1	F5	11	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	C5	11	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	D5	11	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	E5	11	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	F1	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	C1	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	D1	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	E1	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	76	7	1	1	(PC)←(PC)+1	X X X X X					
	NOP	0 0 0 0 0 0 0 0	00	4	1	1	(PC)←(PC)+1	X X X X X					

*: State is T1. **: State is T2.

Symbol	Meaning
r	Register
m	Two-byte data
n	One-byte data
<B2>	Second byte of instruction
<B3>	Third byte of instruction
AAA	Binary representation for RST instruction n
F	8-bit data from the most to the least significant bit S, Z, 0, CY1, 0, P, 1, CY2
PC	Program counter
SP	Stack pointer

Symbol	Meaning
SSS or DDD	Bit pattern designating register or memory. Where, M=(H)(L)

Register or memory	SSS or DDD
B	000
C	001
D	010
E	011
H	100
L	101
M	110
A	111

Symbol	Meaning
←	Data is transferred in direction shown
( )	Contents of register or memory location
∨	Inclusive OR
∀	Exclusive OR
∧	Logical AND
¯	1's complement
X	Content of flag is not changed after execution
O	Content of flag is set or reset after execution
I	Input mode
O	Output mode

8

8-BIT PARALLEL CPU

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	(NOP)	(NOP)	(NOP)	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	(NOP)	(NOP)	(NOP)	(NOP)	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	(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	(JMP)	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	(CALL)	(CALL)	(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 instructions. $D_3 \sim D_0$ indicate the lower 4 bits of the machine code and $D_7 \sim D_4$ indicate the upper 4 bits. Hexadecimal numbers are also used to indicate this code. The instruction may consist of one, two, or three bytes, but only the first byte is listed.

indicates a three-byte instruction.
indicates a two-byte instruction.
( ) is not a formal instruction, but if this code is accessed, the instruction in parentheses may be executed. This is not, however, guaranteed.

MITSUBISHI ELECTRIC

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{BB} (substrate)	−0.3～20	V
V_{CC}	Supply voltage		−0.3～20	V
V_{SS}	Supply voltage		−0.3～20	V
V_I	Input voltage		−0.3～20	V
Pd	Maximum power dissipation	Ta = 25°C	1500	mW
Topr	Operating free-air temperature range		0～70	°C
Tstg	Storage temperature range		−65～150	°C

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{BB}	Supply voltage	−4.75	−5	−5.25	V
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{DD}	Supply voltage	11.4	12	12.6	V
V_{SS}	Supply voltage		0		V
V_{IH}	High-level input voltage	3.3		$V_{CC}+1$	V
V_{IL}	Low-level input voltage	−1		0.8	V
$V_{IH(\phi)}$	High-level clock input voltage	9		$V_{DD}+1$	V
$V_{IL(\phi)}$	Low-level clock input voltage	−1		0.8	V
Topr	Operating free-air temperature	0		70	°C

ELECTRICAL CHARACTERISTICS ($T_a = 0 \sim 70°C$, $V_{DD} = 12V \pm 5\%$, $V_{CC} = 5V \pm 5\%$, $V_{BB} = -5V \pm 5\%$, $V_{SS} = 0V$, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OL}	Low-level output voltage	$I_{OL} = 1.9mA$, All output			0.45	V
V_{OH}	High-level output voltage	$I_{OH} = -150\mu A$	3.7			V
I_{BB}	V_{BB} supply current	Operating at $t_{c(\phi)} = 480ns$, $T_a = 25°C$ (Note 2)		−0.01	−1	mA
I_{CC}	V_{CC} supply current			60	75	mA
I_{DD}	V_{DD} supply current			40	70	mA
I_I	Input current, except clock and data bus	$0 \leqq V_I \leqq V_{CC}$			±10	μA
$I_{I(\phi)}$	Clock input current	$0 \leqq V_{I(\phi)} \leqq V_{DD}$			±10	μA
$I_{I(DB)}$	Input current, data bus (Note 3)	$0 \leqq V_{I(DB)} \leqq V_{IL}$ $V_{IL} \leqq V_{I(DB)} \leqq V_{CC}$			10 −100	μA
$I_{I(HOLD)}$	Input current during hold, address or data bus	At hold state $0.45V \leqq V_O \leqq V_{CC}$			10 −100 −2	μA μA mA
$C_{i(\phi_1)}$	Input capacitance, clock input (ϕ_1)	$V_{(\phi_1)} = 0V$, $V_{(\phi_2)} = 0V$, $V_I = 0V$, $V_O = 0V$, f = 1MHz, 25mVr.m.s		20	25	pF
$C_{i(\phi_2)}$	Input capacitance, clock input (ϕ_2)			15	20	pF
C_i	Input capacitance, any input except clock			5	10	pF
Co	Output capacitance			5	20	pF

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

2 : $t_{c(\phi)} = t_{d(\phi_{1H} \cdot \phi_2)} + t_{r(\phi_2)} + t_{w(\phi_2)} + t_{f(\phi_2)} + t_{d(\phi_2 \cdot \phi_1)} + t_{r(\phi_1)}$

3 : Active pull-up resistors will be switched on to the data bus when DBIN is high and data input voltage is more positive than V_{IH} min.

TIMING REQUIREMENTS ($Ta = 0 \sim 70°C$, $V_{DD} = 12V \pm 5\%$, $V_{CC} = 5V \pm 5\%$, $V_{BB} = -5V \pm 5\%$, $V_{SS} = 0V$, unless otherwise noted)

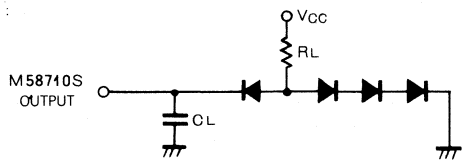
Symbol	Parameter	Limits			Unit
		Min	Typ	Max	
$t_{c(\phi)}$	Clock cycle time (Note 4)	480		2000	ns
$t_{r(\phi)}$	Clock rise time	0		50	ns
$t_{f(\phi)}$	Clock fall time	0		50	ns
$t_{w(\phi 1)}$	Clock 1 pulse width	60			ns
$t_{w(\phi 2)}$	Clock 2 pulse width	220			ns
$t_{d(\phi 1L-\phi 2)}$	Delay time, clock 1 to clock 2	0			ns
$t_{d(\phi 2-\phi 1)}$	Delay time, clock 2 to clock 1	70			ns
$t_{d(\phi 1H-\phi 2)}$	Delay time, clock 1 high to clock 2	80			ns
$t_{su(DA-\phi 1)}$	Data setup time with respect to clock 1	30			ns
$t_{su(DA-\phi 2)}$	Data setup time with respect to clock 2	150			ns
$t_{su(HOLD)}$	Hold setup time	140			ns
$t_{su(INT)}$	Interrupt setup time	120			ns
$t_{su(RDY)}$	Ready setup time	120			ns
$t_{h(DA)}$	Data hold time	$t_{PD(DBI)}$			ns
$t_{h(HOLD)}$	Hold input hold time	0			ns
$t_{h(INT)}$	Interrupt hold time	0			ns
$t_{h(RDY)}$	Ready hold time	0			ns

Note 4 : $t_{c(\phi)} = t_{d(\phi 1L-\phi 2)} + t_{r(\phi 2)} + t_{w(\phi 2)} + t_{r(\phi 2)} + t_{d(\phi 2-\phi 1)} + t_{r(\phi 1)}$

SWITCHING CHARACTERISTICS ($Ta = 0 \sim 70°C$, $V_{DD} = 12V \pm 5\%$, $V_{CC} = 5V \pm 5\%$, $V_{BB} = -5V \pm 10\%$, $V_{SS} = 0V$, unless otherwise noted)

Symbol	Parameter	Test conditions (Note 5)	Limits			Unit
			Min	Typ	Max	
$t_{PD(AD)}$	Propagation delay time, clock 2 to address outputs	$R_L = 2.1k\Omega$, $C_L = 100pF$			200	ns
$t_{PD(DA)}$	Propagation delay time, clock 2 to data bus	$R_L = 2.1k\Omega$, $C_L = 100pF$			220	ns
$t_{PD(CONT)}$	Propagation delay time, clocks to control outputs	$R_L = 2.1k\Omega$, $C_L = 50pF$			120	ns
$t_{PD(DBI)}$	Propagation delay time, clock 2 to DBIN output	$R_L = 2.1k\Omega$, $C_L = 50pF$	25		140	ns
$t_{PD(INT)}$	Propagation delay time, clock 2 to INTE output	$R_L = 2.1k\Omega$, $C_L = 50pF$			200	ns
$t_{PD(DI)}$	Time for data bus to enter input mode				$t_{PD(DBI)}$	ns
t_{PXZ}	Disable time to high-impedance state during hold address output and data bus				120	ns
$t_{d(\overline{WR}-AD)}$	Delay time, write signal to address output	$R_L = 2.1k\Omega$, $C_L = 100pF$	$t_{d(\phi 1H-\phi 2)}$			ns
$t_{d(AD-\overline{WR})}$	Delay time, address output to write signal	$R_L = 2.1k\Omega$, $C_L = 100pF$	Note 6			ns
$t_{d(\overline{WR}-DA)}$	Delay time, write signal to data output	$R_L = 2.1k\Omega$, $C_L = 100pF$	$t_{d(\phi 1H-\phi 2)}$			ns
$t_{d(DA-\overline{WR})}$	Delay time, data output to write signal	$R_L = 2.1k\Omega$, $C_L = 100pF$	Note 7			ns

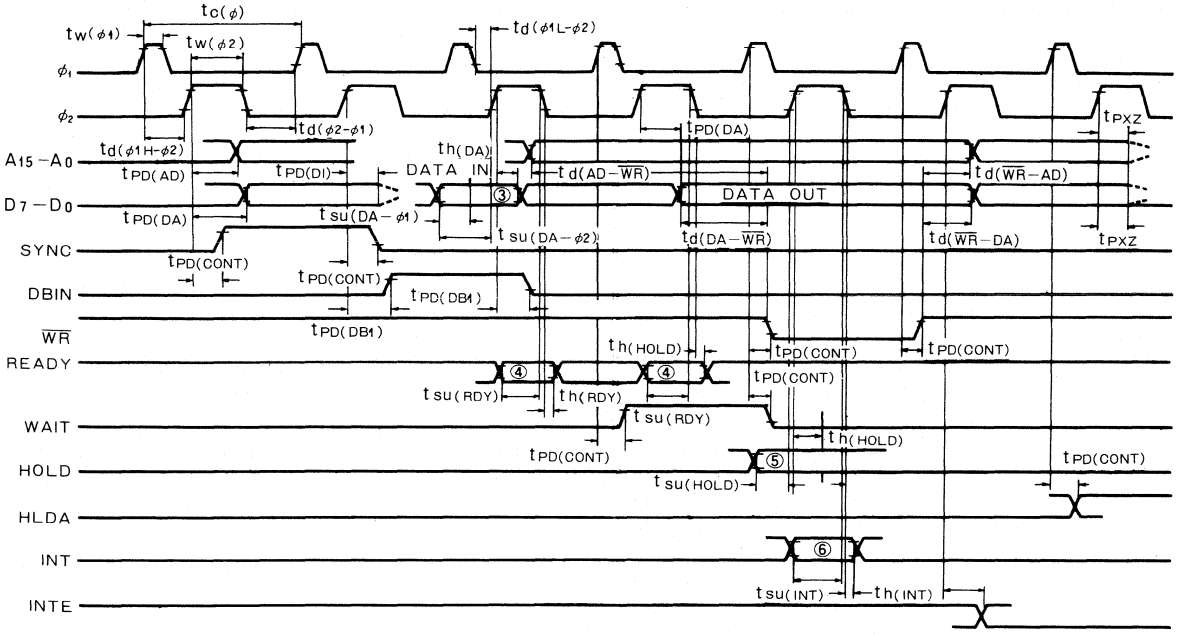
Note 5 : Load circuit :



Note 6 : $t_{d(AD-\overline{WR})} = 2t_{c(\phi)} - t_{d(\phi 1H-\phi 2)} - t_{r(\phi)} - 140ns$

7 : $t_{d(DA-\overline{WR})} = t_{c(\phi)} - t_{c(\phi 1H-\phi 2)} - t_{r(\phi)} - 170ns$

TIMING DIAGRAM



Note 8 : This timing diagram shows timing relationships only; it does not represent any specific machine cycle.

9 : Time measurements are made at the following reference voltages: Clock voltage H = 8.0V, L = 1.0V; input voltage, H = 3.3V, L = 0.8V; output voltage, H = 2.0V, L = 0.8V.

10 : Data on the data bus must be stable for this period in the input mode. Requirements $t_{su(DA-\phi1)}$, $t_{su(DA-\phi2)}$, $t_{h(DA)}$ must be satisfied.

11 : The ready signal must be stable for this period during state T_2 or T_W. External synchronization is required.

12 : The hold signal must be stable for this period during state T_2 or T_W when entering the hold mode and during states T_3, T_4, T_5, T_{WH} and T_H when in the hold mode. External synchronization is not required.

13 : The interrupt signal INT must be stable for the period immediately before the last state of any instruction in order to be recognized on the following machine cycle M_1. External synchronization is not required.

8

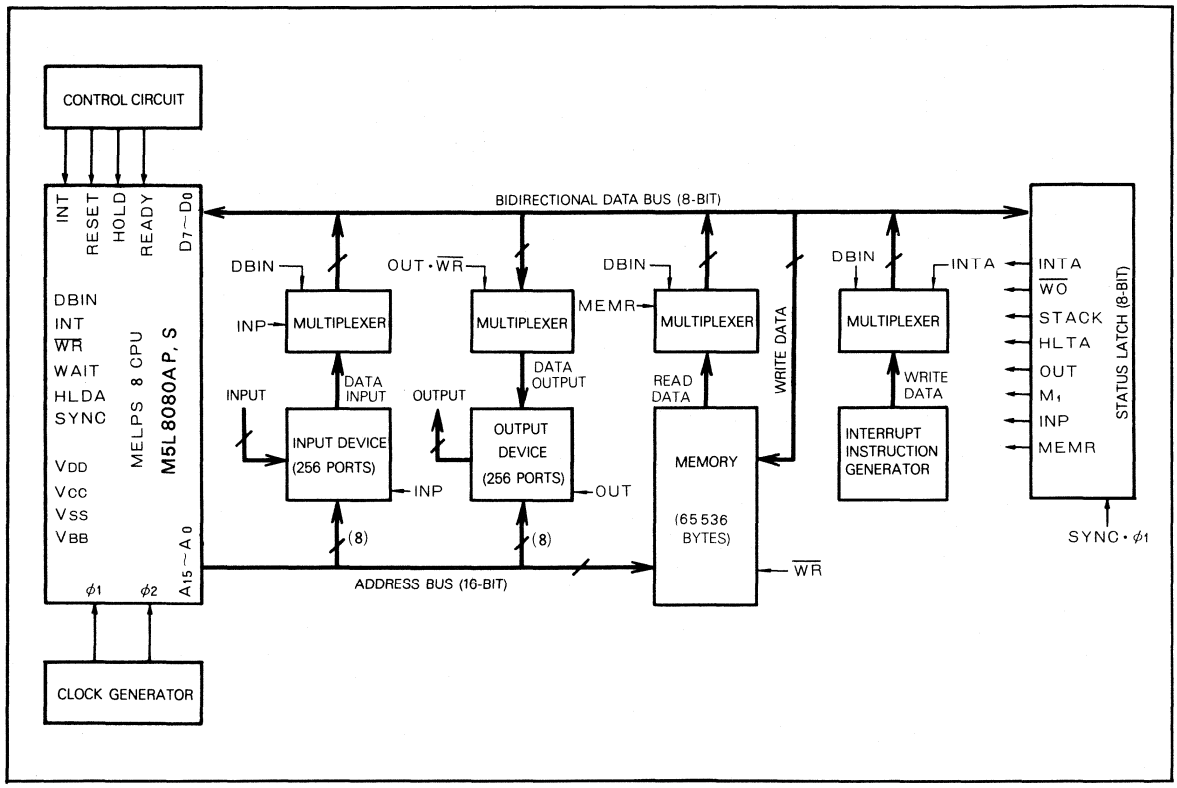
8-BIT PARALLEL CPU

APPLICATIONS

A Basic System Using the M5L8080A P, S

The configuration of a system using the M5L8080AP, S will depend on the functions of the system. A typical basic system is shown in Fig. 1, and a summary of its operation is as follows:

Fig. 1 An example of a basic system using the M5L8080A P, S



1. After the CPU receives the two phase clocks ϕ_1 and ϕ_2 from the clock generator and the external reset signal, the address bus provides the address to memory location zero.
2. At the same time the CPU sends out status signals, which are latched temporarily in the status latch (flip-flops in which status information is latched). The status signals alert external circuits as to the state of the machine cycle that the CPU is ready to execute. When the CPU calls for data or instructions to be read from memory, status signal MEMR is applied to the multiplexer, and the 8-bit data from memory is read into the CPU through the bidirectional data bus across the multiplexer.
3. The 8-bit data coming from memory is decoded as an instruction. If it is a register-reference-arithmetic instruction, it is executed in the CPU; if it is a move-to-memory instruction, the CPU outputs the memory location to the address bus and data to be written on the data bus in the next machine cycle (Note 1). The memory write in operation is executed by write control signal WR.
4. During input and output operation, the CPU outputs the I/O device number to the address bus, outputs a status signal (INP in the input mode; OUT in the output mode) and executes the read/write operation to the I/O devices using the bidirectional data bus.
5. If there is a signal from terminal INT to the CPU, the CPU is in the interrupt enable state, and it sends out status information INTA (Note 15), and an interrupt instruction is sent to the CPU from the interrupt instruction generator across the multiplexer. By executing this interrupt instruction, the CPU can jump to the interrupt processing subroutine.

Note 14 : Each instruction may have five machine cycles. For register-to-register transfer or arithmetic instruction, instruction fetching and execution are carried out by machine cycle M_1 but memory access instructions, or 2-byte or 3-byte instructions require more than one machine cycle.

15 : The interrupt acknowledge signal goes high when the CPU accepts an interrupt request (INT) signal.

Push-Down Stack Operation

The M58710S has a last-in first-out stack. This stack has a pointer that maintains the address of the next available stack location in memory and can be initialized to use any position of memory. Since the stack pointer has a 16-bit register, it can locate any stack location up to 65536 bytes according to memory capacity. An example of the interrupt request is shown in Fig. 2 and the operation of the stack pointer in Fig. 3.

Fig. 2 Processing an interrupt request

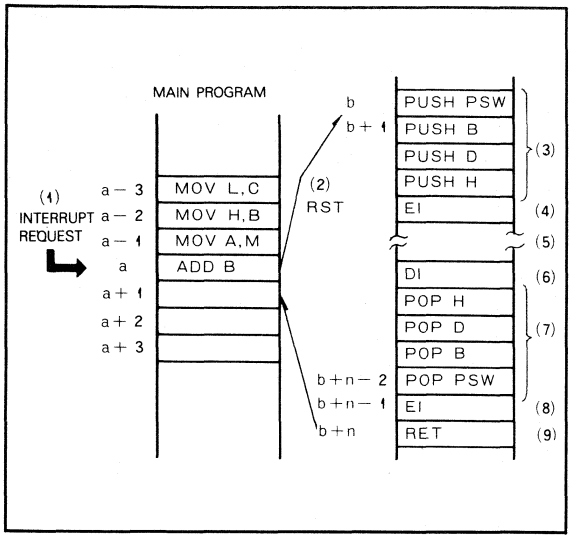


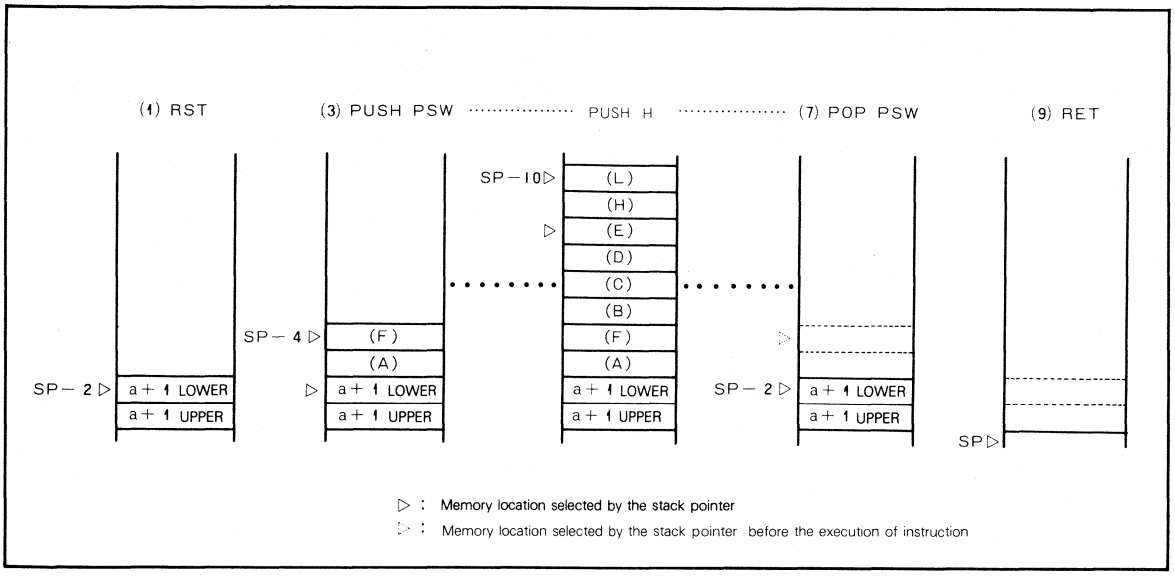
Fig. 2 is explained as follows:

1. An external interrupt request occurs when the CPU executes the instruction stored at location a in the main program.
2. Instruction RST is fetched, the content of the program counter is incremented and pushed onto the push-down stack. Then the CPU jumps to the first location b of the interrupt operation program.
3. The contents of the register are pushed onto the stack. F (in Fig. 3) indicates 8-bit data of flag flip-flops including CY_2, CY_1, Z, P and S. These are, from the most to the least significant bit, S, Z, 0, CY_1, 0, P, 1, CY_2.
4. Instruction EI is executed, enabling the CPU to accept the next interrupt request.
5. The interrupt operation is carried out.
6. The CPU enters the interrupt disable state.
7. The contents of registers are popped off the stack.
8. Instruction EI is executed, enabling the CPU to accept the interrupt request after return to the main program.
9. The content of the program counter is returned to location a+1 of the main program.

The operation of the push-down stack shown in Fig. 2 is described in Fig. 3, where SP indicates the content of the stack pointer before the interrupt is requested. Instruction LXI SP should be used to initialize the stack pointer.

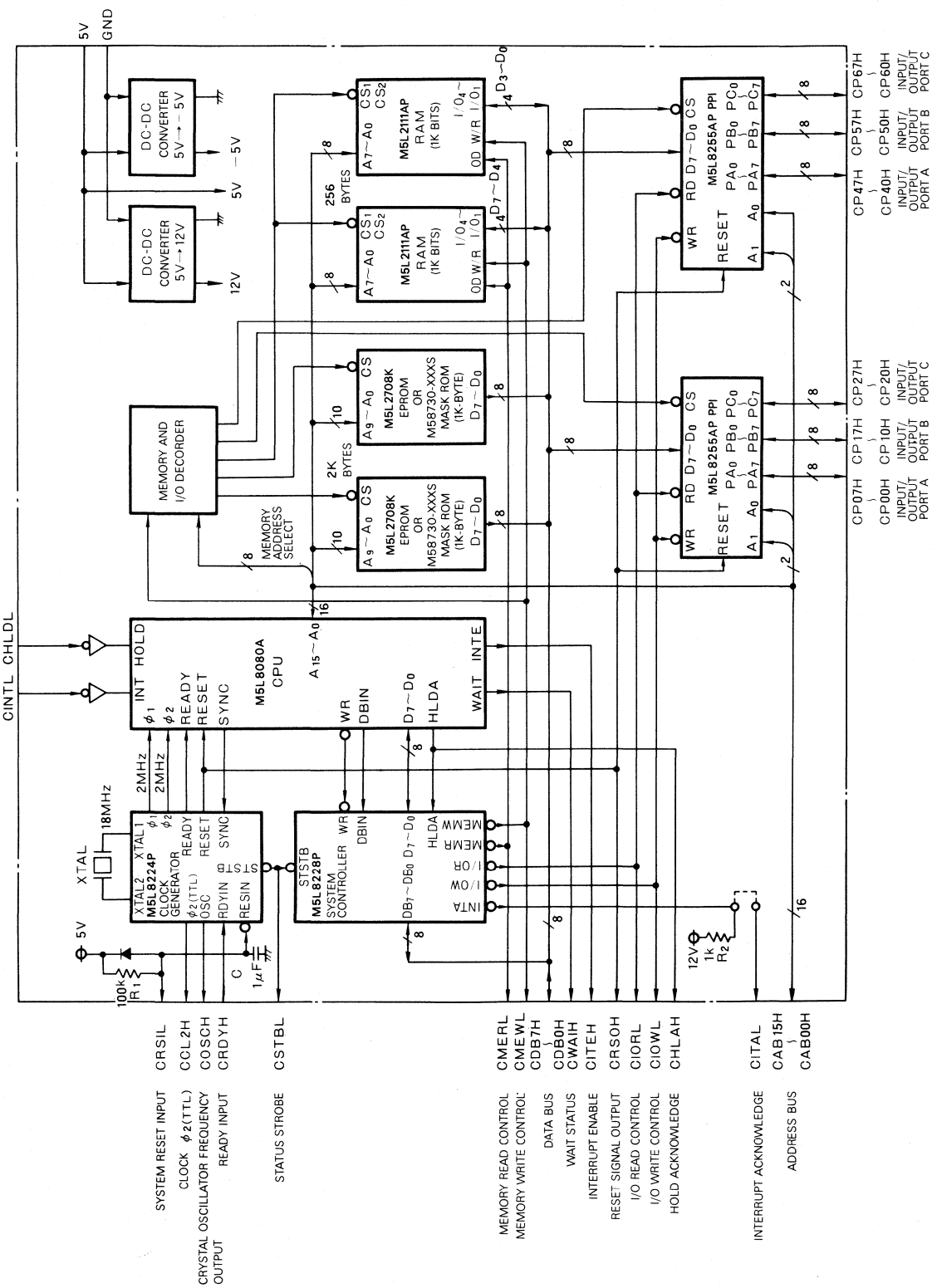
The content of the stack pointer is SP-4 at (3), but at (9), after the execution of instruction RET, it returns to the initial state, and the content of the stack point is SP.

Fig. 3 Operation of the push-down stack



8-BIT PARALLEL CPU

EXAMPLE OF APPLICATION CIRCUIT



MITSUBISHI ELECTRIC

CLOCK GENERATOR AND DRIVER FOR M5L 8080A P, S CPU

DESCRIPTION

The M5L8224P is a clock generator/driver for M5L8080A P,S CPUs. It is controlled by a crystal, selected by the user, to meet a variety of system speed requirements. It is fabricated by using Schottky TTL technology.

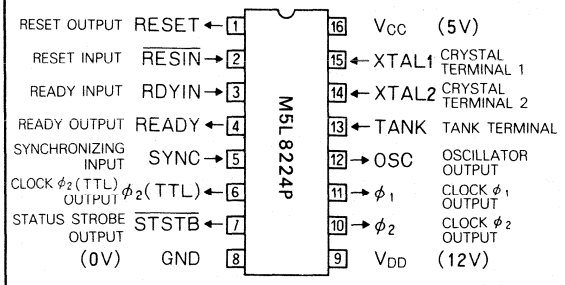
FEATURES

- Crystal controlled for stable clock frequency generation
- Clock outputs ϕ_1, ϕ_2, and ϕ_2(TTL level), and an oscillator output are brought out
- Power-up reset for CPU auto-reset
- Status latch signal
- Synchronizing ready signal output
- Interchangeable with Intel's 8224 in terms of pin configuration and electrical characteristics

APPLICATION

- Single chip clock generator/driver for M5L 8080A P, S CPUs

PIN CONFIGURATION (TOP VIEW)



Outline 16P1

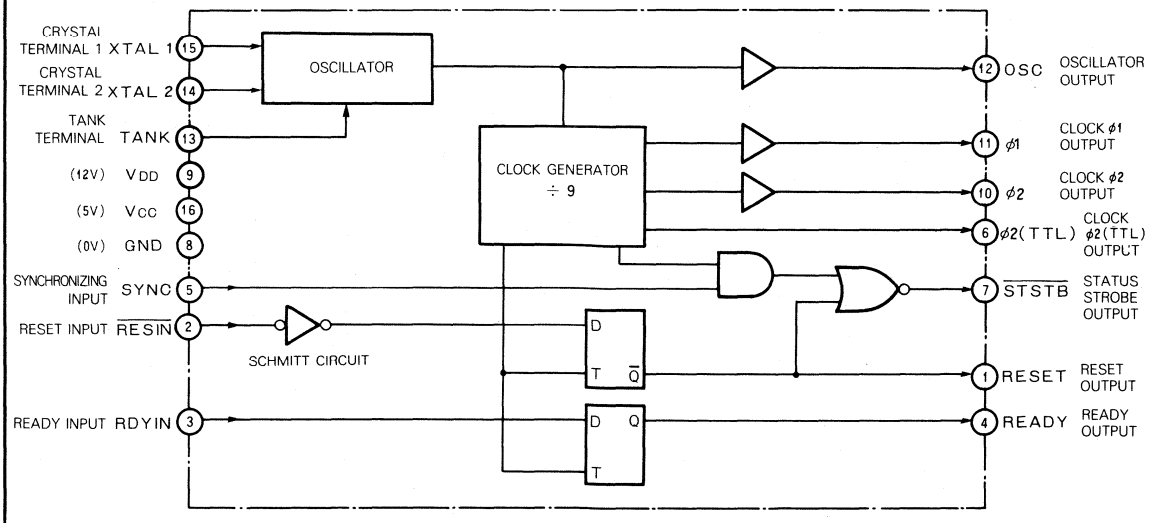
FUNCTION

When an 18MHz crystal is connected between XTAL1 and XTAL2, clock outputs ϕ_1, ϕ_2, and ϕ_2(TTL level), along with oscillator output, are brought out for a CPU with a basic cycle time of 500ns. At this time, ϕ_1 pulse width is 110ns (2X55ns), ϕ_2 pulse width is 275ns (5X55ns). When an overtone mode crystal is used, the external LC network is connected to the TANK input to provide additional gain.

If an external RC network is connected to $\overline{\text{RESIN}}$ at system power-up time, a reset signal is generated; and the system is reset automatically. When a signal from a CPU is applied to the SYNC, $\overline{\text{STSTB}}$ is generated. The RDYIN input sends a synchronous "wait request" signal to the internal D-type flip-flop, and a synchronized READY signal is generated.

8

BLOCK DIAGRAM



CLOCK GENERATOR AND DRIVER FOR M5L 8080A P, S CPU

SUMMARY OF OPERATIONS

Oscillator

This circuit delivers the basic oscillation frequency to the system. A crystal must be connected between terminals XTAL1 and XTAL2. The oscillation frequency is buffered and taken out to the terminal OSC to be utilized as a basic frequency source for other system timing signals.

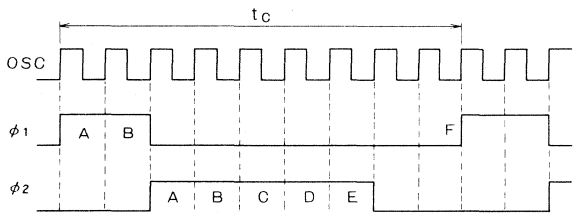
'Divide-By-Nine' Counter

The oscillation frequency is divided by this counter to create two clocks necessary for the M5L 8080A P, S CPU. A TTL level phase 2 clock signal $\phi_{2(TTL)}$ is also available for external timing purposes.

Status Strobe Output ($\overline{STSTB}$)

This signal is used in latching the status information from the CPU and is generated in synchronism with F of clock ϕ_1 (see Fig.1) when the SYNC signal from the CPU is supplied to terminal SYNC, and SYNC is in high-level. It is applied to input terminal $\overline{STSTB}$ of the M5L 8228P system controller.

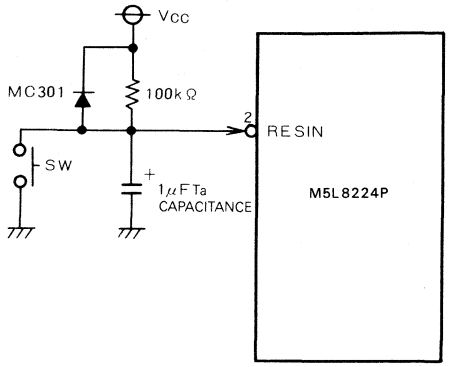
Fig. 1 Clock signal timing diagram



Reset Input ($\overline{RESIN}$)

When this signal turns low-level, output from terminal RESET goes high-level, synchronizing with the B on clock ϕ_2 (see Fig.1). This signal input is connected with the Schmitt trigger, providing a power-on-reset function as in the typical reset circuit shown in Fig.2.

Fig. 2 Typical reset circuit



Ready Input (RDYIN)

When this signal turns low-level, terminal READY also turns low-level, synchronizing with the B on clock ϕ_2. It serves to issue wait requests to the CPU (see Fig.1).

MITSUBISHI ELECTRIC

CLOCK GENERATOR AND DRIVER FOR M5L 8080A P, S CPU

USAGE

When the M5L 8224P clock generator is used with the M5L 8080A P CPU, crystal oscillator frequency (f_x) must be 9 times the desired CPU cycle time (t_c). For example if t_c is to be 500ns, f_x will be:

$$\frac{1}{500\times10^{-9}}\times 9 = 18\,(\mathrm{MHz})$$

However, t_c limits are 2μs (max) and 480ns (min) so that f_x must be selected within a range of 4.5~18.75MHz. It is recommended that the fundamental be used. If a higher harmonic (overtone) is to be used, an L/C network must be connected to terminal TANK as shown in Fig. 4. If a fundamental of 10MHz or more is used, a capacitance of about 10pF must be inserted in series with the crystal (see Fig. 5).

Crystals must satisfy following conditions:

Tolerance:	0.005% between 0~75°C
Series resonance:	Fundamental
Load capacitance:	20~35pF
Equivalent resistance:	75Ω or less

Fig. 3 Typical M5L 8224P connection

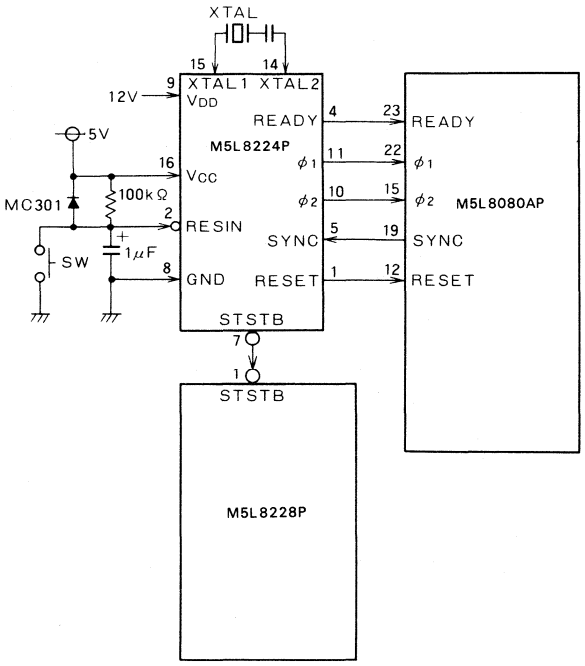


Fig. 4 Example of use of a crystal oscillator in an overtone circuit

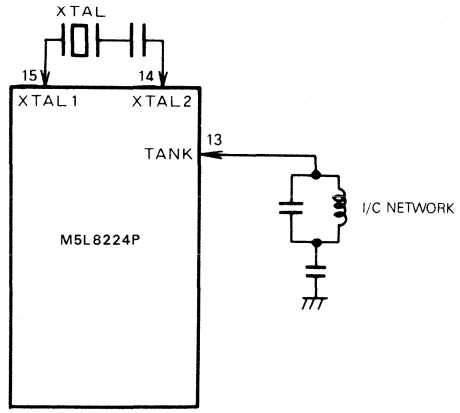
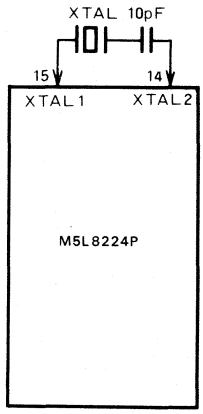


Fig. 5 Example of use of a crystal of 10MHz or above



Precautions

1. Do not short clocks ϕ_1 or ϕ_2 to ground.
2. Application of nominal V_{DD} (12V) before application of nominal V_{CC} (5V) may damage the device. Proper switching order must be observed.

8

CLOCK GENERATOR AND DRIVER FOR M5L 8080A P, S CPU

ABSOLUTE MAXIMUM RATINGS (Ta=0～75°C, unless otherwise noted)

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage		7.0	V
V_{DD}	Supply voltage		13.5	V
V_I	Input voltage		7.0	V
V_O	Output voltage, all outputs except ϕ_1 and ϕ_2		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
V_{DD}	Supply voltage	11.4	12	12.6	V
I_{OH}	High-level output current, $\phi1$, $\phi2$, READY, RESET			−100	μA
I_{OH}	High-level output current, all other outputs			−1	mA
I_{OL}	Low-level output current, $\phi1$, $\phi2$, READY, RESET, $\overline{STSTB}$			2.5	mA
I_{OL}	Low-level output current, all other outputs			16	mA
$f_{r\,max}$	Maximum repetition frequency			27	MHz

ELECTRICAL CHARACTERISTICS (Ta=0～75°C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{IH}	High-level input voltage, $\overline{RESIN}$		2.6			V
V_{IH}	High-level input voltage, all other inputs		2.0			V
V_{IL}	Low-level input voltage				0.8	V
$V_{IH}-V_{IL}$	Input hysteresis voltage, $\overline{RESIN}$	V_{CC}=5.0V, V_{DD}=12.0V	0.25			V
V_{IC}	Input clamped voltage	V_{CC}=4.75V, I_{IC}=−5mA			−1.0	V
V_{OH}	High-level output voltage, $\phi1$, $\phi2$	V_{CC}=4.75V, V_{DD}=11.4V, I_{OH}=−100μA	9.4			V
V_{OH}	High-level output voltage, READY, RESET	V_{CC}=4.75V, V_{DD}=11.4V, I_{OH}=−100μA	3.6			V
V_{OH}	High-level output voltage, other outputs	V_{CC}=4.75V, V_{DD}=11.4V, I_{OH}=−1mA	2.4			V
V_{OL}	Low-level output voltage, $\phi1$, $\phi2$, READY, RESET, $\overline{STSTB}$	V_{CC}=4.75V, V_{DD}=11.4V, I_{OL}=2.5mA			0.5	V
V_{OL}	Low-level output voltage, all other outputs	V_{CC}=4.75V, V_{DD}=11.4V, I_{OL}=16mA			0.5	V
I_{IH}	High-level input current	V_{CC}=5.25V, V_{DD}=12.6V, V_I=5.25V			10	μA
I_{IL}	Low-level input current	V_{CC}=5.25V, V_{DD}=12.6V, V_I=0.5V			−0.25	mA
I_{OS}	Short-circuit output current (Note 3)	V_{CC}=5.0V, V_{DD}=12.0V V_O=0V, V_{IH}=4.5V, V_{IL}=0V	−10		−60	mA
I_{CC}	Supply current from V_{CC}	V_{CC}=5.25V, V_{DD}=12.6V, V_{IH}=4.5V V_{IL}=0V			115	mA
I_{DD}	Supply current from V_{DD}				12	mA

Note 1 : All voltages are with respect to GND terminal. Reference voltage (pin 8) 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. Outputs $\phi1$ and $\phi2$ should not be short-circuited to GND.

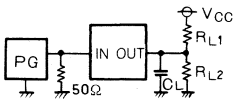
TIMING REQUIREMENTS (Ta=25°C, V_{CC}=5V, V_{DD}=12V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{su(RDYIN)}$	RDYIN setup time with respect to $\overline{STSTB}$	$\overline{STSTB}$ output terminal C_L=15pF R_{L1}=2kΩ R_{L2}=4kΩ	$50-\frac{4t_C}{9}$			ns
$t_{h(RDYIN)}$	RDYIN hold time with respect to $\overline{STSTB}$		$\frac{4t_C}{9}$			ns

SWITCHING CHARACTERISTICS (Ta=25°C, V_{CC}=5V, V_{DD}=12V, unless otherwise noted)

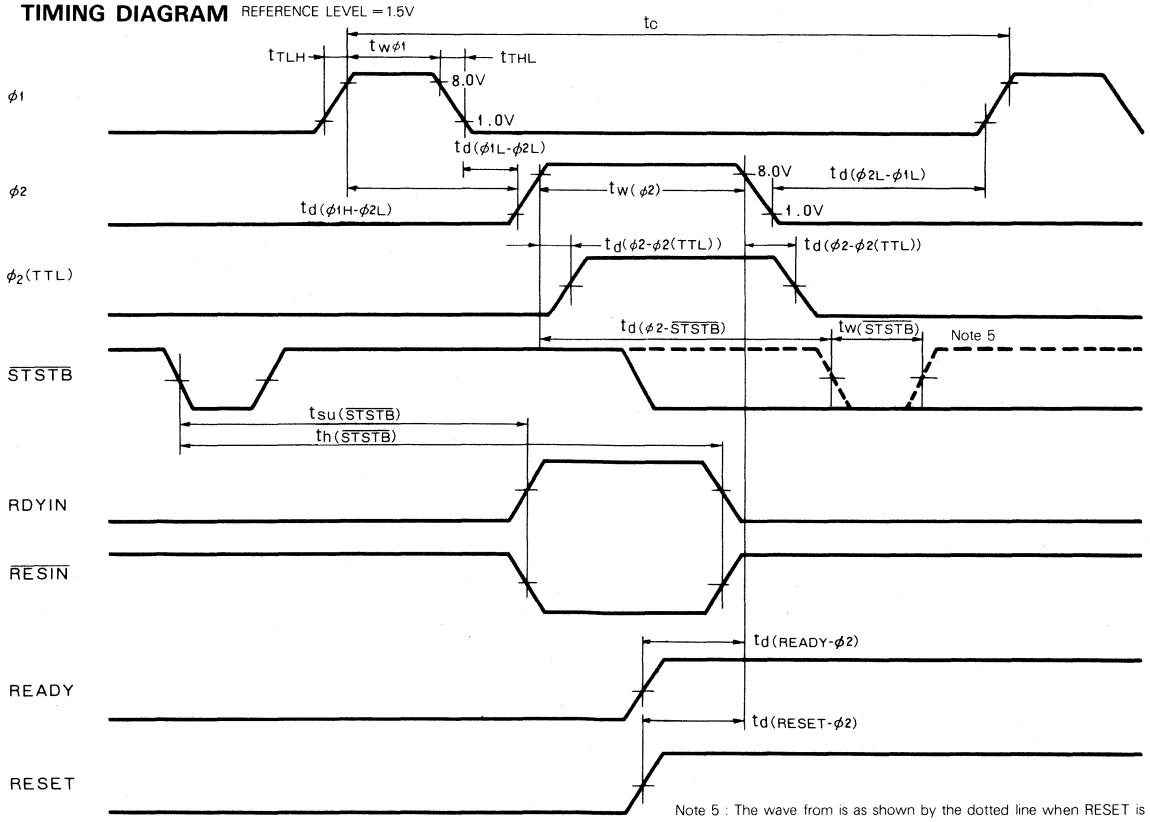
Symbol	Parameter	Test conditions (Note 4)	Limits			Unit
			Min	Typ	Max	
$t_{w(\phi 1)}$	Clock $\phi 1$ pulse width	C_L=20~50pF R_{L1}=∞Ω, R_{L2}=∞Ω	$\frac{2t_C}{9}-20$			ns
$t_{w(\phi 2)}$	Clock $\phi 2$ pulse width		$\frac{5t_C}{9}-35$			ns
$t_{d(\phi 1L-\phi 2L)}$	Delay time from $\phi 1$ low-level to $\phi 2$ low-level		0			ns
$t_{d(\phi 2L-\phi 1L)}$	Delay time from $\phi 2$ low-level to $\phi 1$ low-level		$\frac{2t_C}{9}-30$			ns
$t_{d(\phi 1H-\phi 2L)}$	Delay time from $\phi 1$ high-level to $\phi 2$ low-level		$\frac{2t_C}{9}-5$		$\frac{2t_C}{9}+25$	ns
t_{TLH}	Transition time, low-to-high-level $\phi 1$ and $\phi 2$	C_L=20~50pF R_{L1}=∞Ω, R_{L2}=∞Ω			20	ns
t_{THL}	Transition time, high-to-low-level $\phi 1$ and $\phi 2$				20	ns
$t_{d(\phi 2-\phi 2(TTL))}$	Delay time from $\phi 2$ to $\phi 2$ (TTL)	$\phi 2$(TTL) output C_L=30pF, R_{L1}=300Ω, R_{L2}=600Ω	−10		20	ns
$t_{d(\phi 2-\overline{STSTB})}$	Delay time from $\phi 2$ to $\overline{STSTB}$	$\overline{STSTB}$ output C_L=15pF, R_{L1}=2kΩ, R_{L2}=4kΩ	$\frac{6t_C}{9}-30$		$\frac{6t_C}{9}$	ns
$t_{w(\overline{STSTB})}$	$\overline{STSTB}$ pulse width		$\frac{t_C}{9}-15$			ns
$t_{d(READY-\phi 2)}$	Delay time from READY to $\phi 2$	READY, RESET output C_L=10pF, R_{L1}=2kΩ, R_{L2}=4kΩ	$\frac{4t_C}{9}-25$			ns
$t_{d(RESET-\phi 2)}$	Delay time from RESET to $\phi 2$					ns

Note 4 : Measurement circuit:



Voltage measurement points:

TIMING DIAGRAM REFERENCE LEVEL = 1.5V

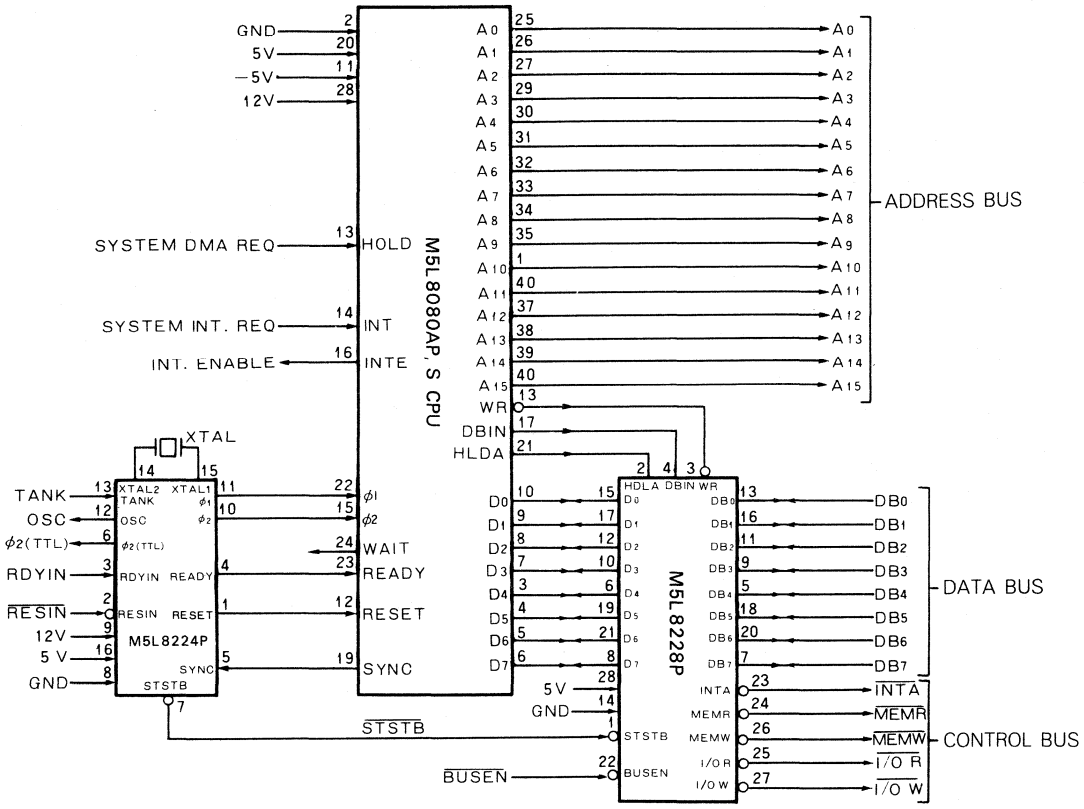


Note 5 : The wave from is as shown by the dotted line when RESET is low.

8

CLOCK GENERATOR AND DRIVER FOR M5L 8080A P, S CPU

TYPICAL APPLICATION CIRCUIT



MITSUBISHI ELECTRIC

SYSTEM CONTROLLER AND BUS DRIVER FOR M5L 8080A P, S CPU

DESCRIPTION

The M5L 8228P is a system controller and bus driver for M5L 8080A P, S CPUs. It generates all signals required to directly interface the MELPS 8 series RAMs, ROMs and input/output devices. A bidirectional bus driver, along with system control signals, provides for high system TTL fan-out. It is fabricated using Schottky TTL technology.

FEATURES

- Built-in bidirectional bus driver for data bus isolation
- Built-in status signal
- High system TTL fan-out
- User selected single level interrupt vector (RST 7)
- Interchangeable with Intel's 8228 in terms of pin configuration and electrical characteristics

APPLICATION

- Data bus driver and status signal generation for M5L 8080A P, S CPU

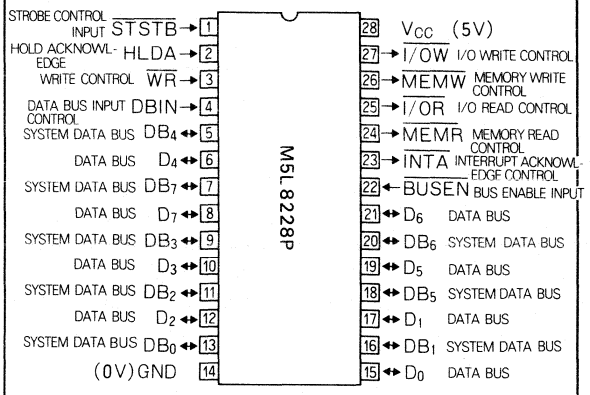
FUNCTION

The bidirectional bus driver provides high system TTL fanout, as well as isolation for an M5L 8080A P, S CPU data bus from memory and I/O devices.

Status signals from a CPU are latched in the internal status latch when the status strobe signal $\overline{STSTB}$ goes low. The gating array generates control signals (memory read $\overline{MEMR}$, memory write $\overline{MEMW}$, input/output read $\overline{I/OR}$, input/output write $\overline{I/OW}$, and interrupt acknowledge $\overline{INTA}$) by gating the output of the status latch with the control signals DBIN, $\overline{WR}$ and HLDA from a CPU. The bus enable input $\overline{BUSEN}$ forces the data bus output buffers and control signal buffers to high-impedance state if they are in the high-state.

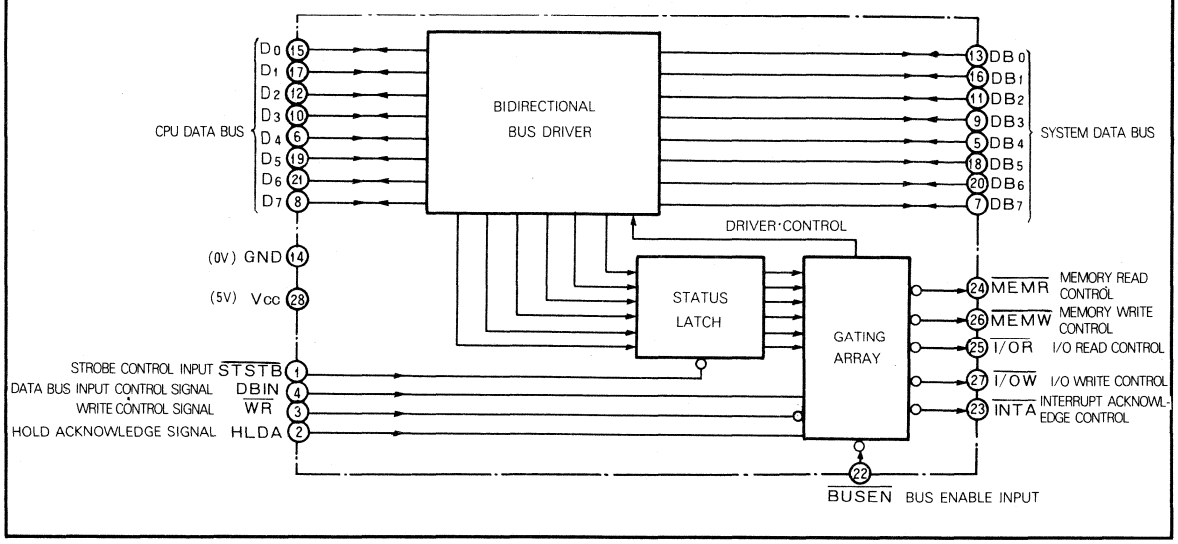
An RST 7 instruction gated to the bus as an interrupt is acknowledged when the DBIN input is active and a 12V supply in series with a 1kΩ resistor is connected to the acknowledge output $\overline{INTA}$.

PIN CONFIGURATION (TOP VIEW)



Outline 28P1

BLOCK DIAGRAM



8

SYSTEM CONTROLLER AND BUS DRIVER FOR M5L 8080A P, S CPU

SUMMARY OF OPERATIONS

Bidirectional Bus Driver

An 8-bit bidirectional bus driver is provided to buffer the data bus of the M5L 8080A CPU from the memory and I/O devices. Its flow is controlled by the gating array. The system data bus output is provided with ample load-driving capacity (I_{OL}=10mA). It can be turned to the high-impedance state by the bus enable ($\overline{BUSEN}$) input in order to separate the memory and I/O devices from the CPU.

Status Latch

Latches status information from the CPU and is used in deriving the memory and I/O control signals. Status information from CPU terminals D_0 – D_7 is latched at the rising edge of $\overline{STSTB}$. Terminal $\overline{STSTB}$ is usually connected to output terminal $\overline{STSTB}$ of the M5L 8224P clock generator.

FUNCTION OF THE STATUS SIGNALS

Data bus	Signal name	Status information	Functions
D_0	INTA	Interrupt acknowledge	Turns high when CPU acknowledges interrupt request by INT.
D_1	$\overline{WO}$	Write mode discriminating	Turns high when CPU is in read mode, and turns low when in write mode.
D_2	STACK	Stack	Turns high during that part of the machine cycle when the value in the stack pointer, that is, the address of the push down stack is output on the address bus.
D_3	HLTA	HLT instruction acknowledge	Turns high during that part of the machine cycle when the CPU halts on executing the HLT instruction.
D_4	OUT	Output instruction acknowledge	Turns high during that part of the machine cycle when the output port number is carried to the address bus and the data is carried to the data bus. The output port number is output on both the upper and lower eight bits of the address bus at the same time.
D_5	M_1	M_1	Turns high during that part of the machine cycle when the CPU fetches the first byte to the instruction.
D_6	INP	Input instruction acknowledge	Turns high during that part of the machine cycle when the input port number is carried to the address bus and the data bus becomes the input mode. The input port number is output on both the upper and lower eight bits of the address bus at the same time.
D_7	MEMR	Memory read	Turns high during that part of the machine cycle when the data bus is utilized to fetch the memory contents.

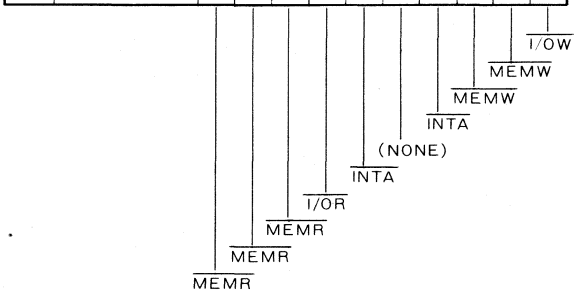
Gating Array

Memory and I/O control signals are generated from this circuit after logically combining the contents of the status latch and signals DBIN, $\overline{WR}$ and HLDA from the CPU.

The relationship between the CPU status information and the M5L 8228P control signal is tabulated below.

STATUS INFROMATION AND THE TYPES OF THE MACHINE CYCLES

Status information \ Mode number	1	2	3	4	5	6	7	8	9	10
Data bus bit / Name of status signal	Instruction fetch	Memory read	Stack read	Input read	Interrupt acknowledge	HALT acknowledge	Interrupt acknowledge while HALT	Memory write	Stack write	Output write
D_0 INTA	0	0	0	0	1	0	1	0	0	0
D_1 $\overline{WO}$	1	1	1	1	1	1	1	0	0	0
D_2 STACK	0	0	1	0	0	0	0	0	1	0
D_3 HLTA	0	0	0	0	0	1	1	0	0	0
D_4 OUT	0	0	0	0	0	0	0	0	0	1
D_5 M_1	1	0	0	0	1	0	1	0	0	0
D_6 INP	0	0	0	1	0	0	0	0	0	0
D_7 MEMR	1	1	1	0	0	1	0	0	0	0



MITSUBISHI ELECTRIC

Use Of Terminal $\overline{INTA}$

1. When Interrupt Instruction is Applied Externally

Fig. 1 Typical external interrupt instruction

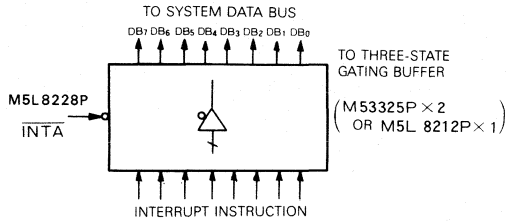
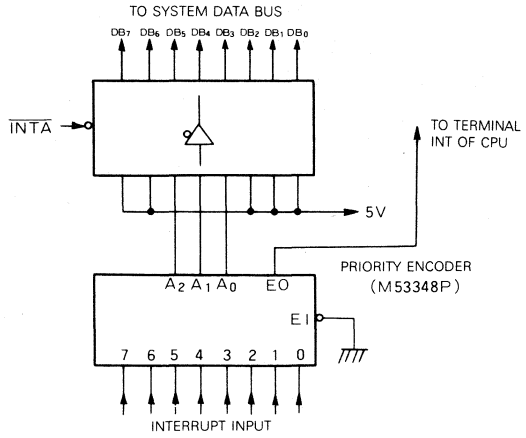


Fig. 2 In case priority is given to an interrupt instruction



2. When Interrupt Instruction is Generated from the M5L 8228P

When terminal $\overline{INTA}$ is connected to the 12V line through a 1kΩ resistance, an instruction coded "FF_{16}" (RST 7) is automatically generated on the CPU data bus at the next DBIN which follows the cycle when the CPU issued the $\overline{INTA}$ status (interrupt acknowledge).

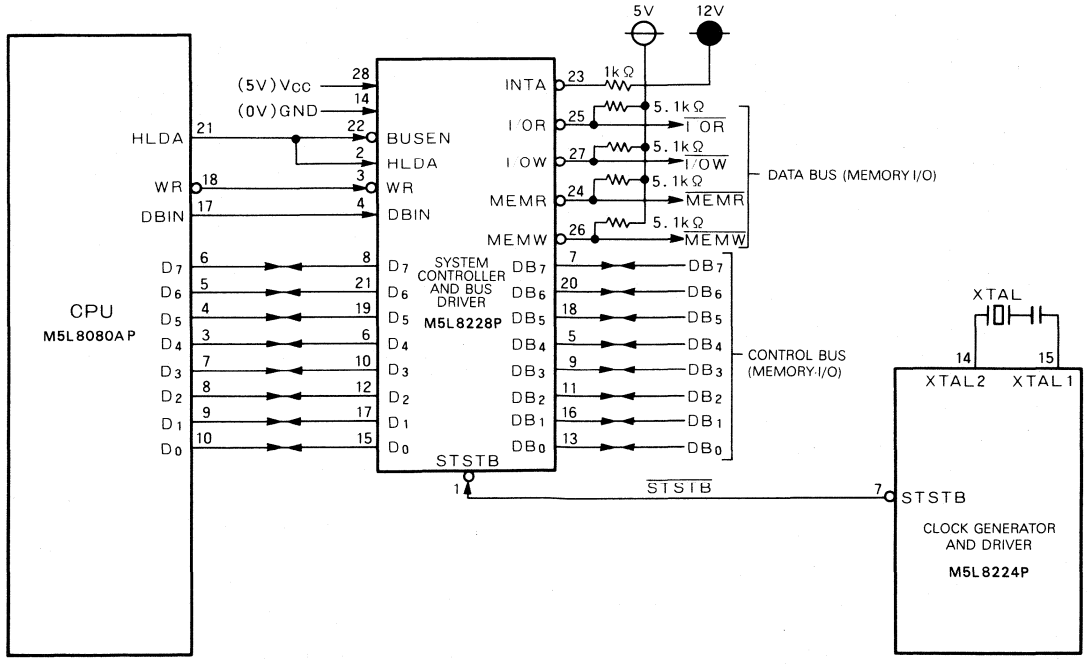
State After Initial Power-On Time

State of the status latch within the M5L 8228P is unstable immediately after the initial power-on, however, the $\overline{STSTB}$ signal is sent to the M5L 8228P when the M5L 8224P clock generator sends the reset signal to the CPU. Even if the CPU data bus is in the high-impedance state at this time, $D_2 = D_8 =$ "1" is latched, as the pull-up resistance is connected with D_2 and D_8 in the M5L 8228P. As the internal flip-flop is reset this way, there will not be any unrequired control signals being issued during the power-on time.

Use of Terminal $\overline{BUSEN}$

Fig.3 shows typical M5L 8228P connection. When terminal $\overline{BUSEN}$ turns high-level, all the data bus buffers and control output buffers of the M5L 8228P turn to the high-impedance state. Therefore, the data and control buses of the system can be controlled externally when signal HLDA (hold acknowledge) is issued from the CPU as the HOLD request was applied to the CPU, if the terminal HLDA of the CPU is connected with the terminal $\overline{BUSEN}$ of the M5L 8228P. This feature is very useful in direct memory accessing DMA.

Fig. 3 Typical M5L 8228P connection



8

SYSTEM CONTROLLER AND BUS DRIVER FOR M5L 8080A P, S CPU

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, $D_0 \sim D_7$ and $\overline{STSTB}$ input		V_{CC}	V
V_I	Input voltage, all other inputs		7.0	V
V_O	Output voltage		V_{CC}	V
P_d	Power dissipation		1.0	W
Topr	Operating free-air temperature		0 ~ 75	°C
Tstg	Storage temperature		−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, $D_0 \sim D_7$ outputs			−10	μA
I_{OH}	High-level output current, all other outputs			−1	mA
I_{OL}	Low-level output current, $D_0 \sim D_7$ outputs			2	mA
I_{OL}	Low-level output current, all other outputs			10	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		2.0			V
V_{IL}	Low-level input voltage				0.8	V
V_{IC}	Input clamp voltage	V_{CC} = 4.75V, I_{IC} = −5mA			−1.0	V
V_{OH}	High-level output voltage, $D_0 \sim D_7$ outputs	V_{CC} = 4.75V, V_{IH} = 2.0V, V_{IL} = 0.8V, I_{OH} = −10μA	3.6			V
	High-level output voltage, all other outputs	V_{CC} = 4.75V, V_{IH} = 2.0V, V_{IL} = 0.8V, I_{OH} = −1mA	2.4			
V_{OL}	Low-level output voltage, $D_0 \sim D_7$ outputs	V_{CC} = 4.75V, V_{IH} = 2.0V, V_{IL} = 0.8V, I_{OL} = 2mA			0.5	V
	Low-level output voltage, all other outputs	V_{CC} = 4.75V, V_{IH} = 2.0V, V_{IL} = 0.8V, I_{OL} = 10mA			0.5	
I_{OZ}	Three-state output current	V_{CC} = 5.25V, V_{IH} = 2.0V, V_{IL} = 0.8V, V_O = 5.25V			20	μA
	Three-state output current	V_{CC} = 5.25V, V_{IH} = 2.0V, V_{IL} = 0.8V, V_O = 0.5V			−20	
I_{IH}	High-level input current, $\overline{STSTB}$ input	V_{CC} = 5.25V, V_{IH} = 4.5V, V_{IL} = 0V, V_I = 5.25V			100	μA
	High-level input current, $DB_0 \sim DB_7$ inputs				20	
	High-level input current, all other inputs				100	
I_{IL}	Low-level input current, $\overline{STSTB}$ input	V_{CC} = 5.25V, V_{IH} = 4.5V, V_{IL} = 0V, V_I = 0.5V			−0.5	mA
	Low-level input current, D_2, D_6 inputs				−0.75	
	Low-level input current, D_0, D_1, D_4, D_5, D_7 inputs				−0.25	
	Low-level input current, all other inputs				−0.25	
I_{OS}	Short-circuit output current (Note 3)	V_{CC} = 5.0V, V_{IH} = 4.5V, V_{IL} = 0V	−15		−90	mA
$I_{I(\overline{INTA})}$	$\overline{INTA}$ terminal current	V_{DD} = 12V, R_L = 1kΩ ±10%			5	mA
I_{CC}	Supply current from V_{CC}	V_{CC} = 5.25V, V_{IH} = 4.5V, V_{IL} = 0V			190	mA

Note 1 : All voltages are with respect to GND terminal. Reference voltage (pin 14) 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 = 0 ~ 75°C, unless otherwise noted)

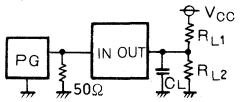
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{w(\overline{STSTB})}$	$\overline{STSTB}$ pulse width		22			ns
$t_{su(DA)}$	$D_0 \sim D_7$ setup time with respect to $\overline{STSTB}$		8			ns
$t_{su(DB)}$	$DB_0 \sim DB_7$ setup time with respect to HLDA		10			ns
$t_{h(DA)}$	$D_0 \sim D_7$ hold time with respect to $\overline{STSTB}$		5			ns
$t_{h(DB)}$	$DB_0 \sim DB_7$ hold time with respect to HLDA		20			ns

MITSUBISHI ELECTRIC

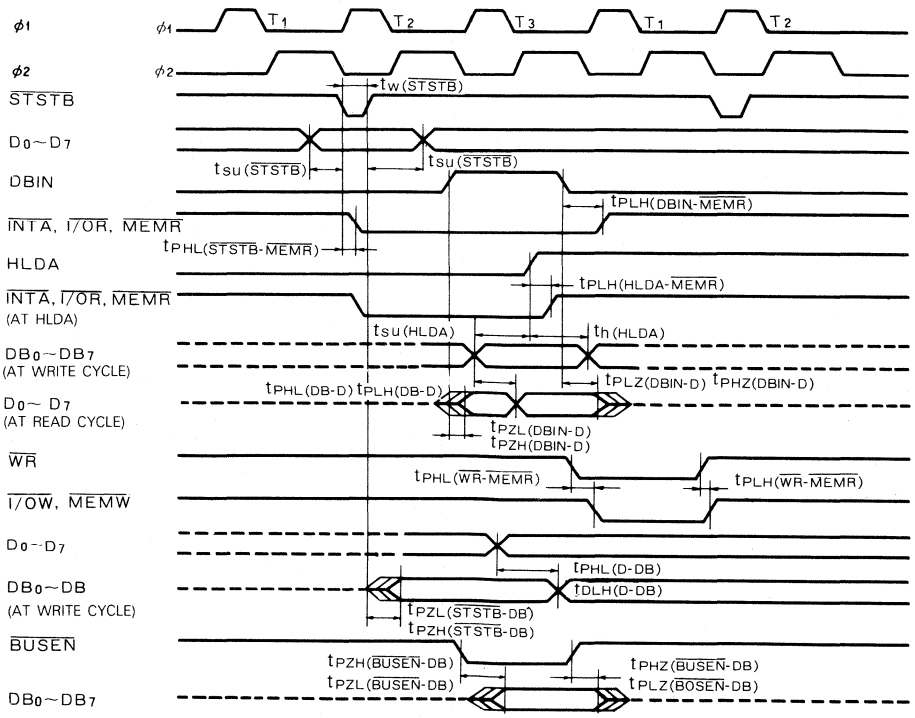
SWITCHING CHARACTERISTICS (Ta = 25°C, V_{CC} = 5V, unless otherwise noted)

	Parameter	Test conditions (Note 4)	Limits			
			Min	Typ	Max	
$t_{PHL(\overline{STSTB}\text{-}\overline{MEMR})}$	High-to-low-level output propagation time, from input $\overline{STSTB}$ to output $\overline{MEMR}$, $\overline{I/OR}$ and $\overline{INTA}$	V_{IH} = 4.5V, V_{IL} = 0V, C_L = 100pF, R_{L1} = 500Ω, R_{L2} = 1kΩ	20		70	ns
$t_{PLH(DBIN\text{-}\overline{MEMR})}$	Low-to-high-level output propagation time, from input DBIN to output $\overline{MEMR}$, $\overline{I/OR}$ and				40	ns
$t_{PZL(DBIN\text{-}D)}$ $t_{PZH(DBIN\text{-}D)}$ $t_{PHZ(DBIN\text{-}D)}$ $t_{PLZ(DBIN\text{-}D)}$	Z-to-low-level, Z-to-high-level, high-to-Z-level and low-to-Z-level output propagation time, from input DBIN to outputs $D_0 \sim D_7$	C_L = 25pF, R_{L1} = 4kΩ, R_{L2} = ∞Ω			55	ns
$t_{PHL(DB\text{-}D)}$ $t_{PLH(DB\text{-}D)}$	High-to-low-level and low-to-high-level output propagation time, from inputs $DB_0 \sim DB_7$ to outputs $D_0 \sim D_7$				40	ns
$t_{PHL(\overline{WR}\text{-}\overline{MEMW})}$ $t_{PLH(\overline{WR}\text{-}\overline{MEMW})}$	High-to-low-level and low-to-high-level output propagation time, from input $\overline{WR}$ to outputs $\overline{MEMW}$ and $\overline{I/OW}$	C_L = 100pF, R_{L1} = 500Ω, R_{L2} = 1kΩ	5		55	ns
$t_{PZL(\overline{STSTB}\text{-}DB)}$ $t_{PZH(\overline{STSTB}\text{-}DB)}$	Z-to-low-level and Z-to-high-level output propagation time, from input $\overline{STSTB}$ to outputs $DB_0 \sim DB_7$				40	ns
$t_{PHL(D\text{-}DB)}$ $t_{PLH(D\text{-}DB)}$	High-to-low-level and low-to-high-level output propagation time, from inputs $D_0 \sim D_7$ to outputs $DB_0 \sim DB_7$		5		50	ns
$t_{PZL(\overline{BUSEN}\text{-}DB)}$ $t_{PZH(\overline{BUSEN}\text{-}DB)}$ $t_{PHZ(\overline{BUSEN}\text{-}DB)}$ $t_{PLZ(\overline{BUSEN}\text{-}DB)}$	Z-to-low-level, Z-to-high-level, high-to-Z-level and low-to-Z-level output propagation time, from input $\overline{BUSEN}$ to outputs $DB_0 \sim DB_7$				40	ns
$t_{PLH(HLDA\text{-}\overline{MEMR})}$	Low-to-high-level output propagation time, from input HLDA to outputs $\overline{MEMR}$, $\overline{I/OR}$ and $\overline{INTA}$				35	ns

Note 4 : Measurement circuit:



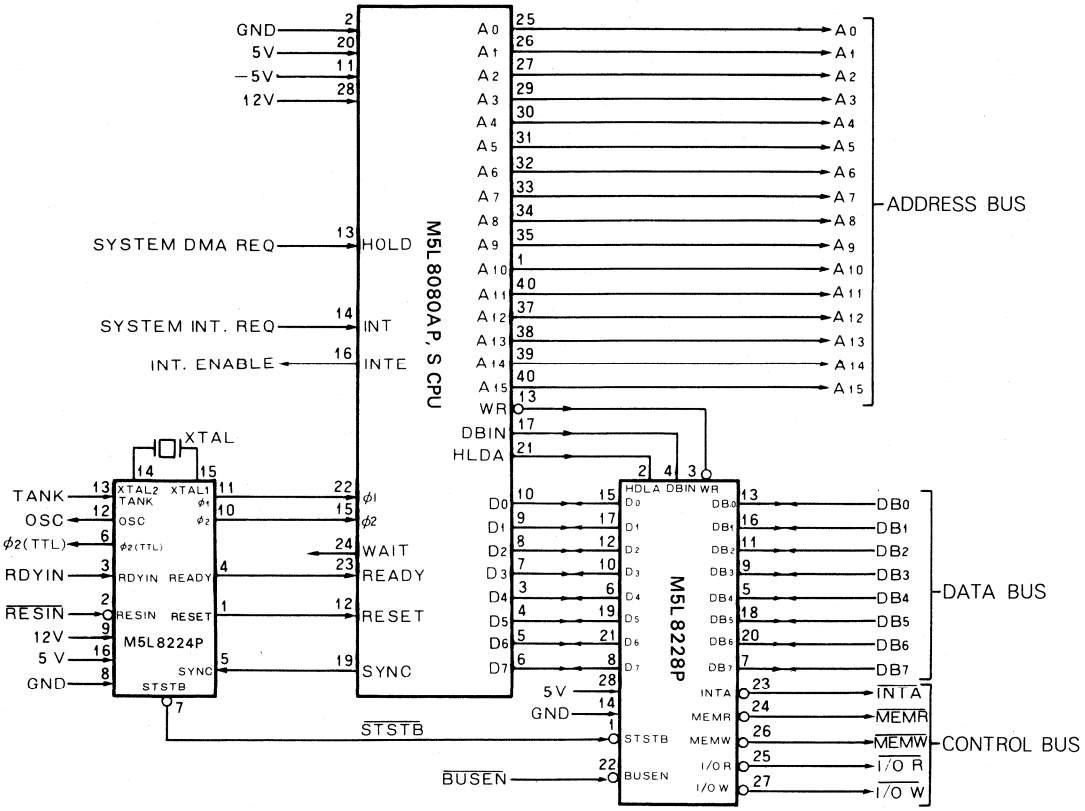
TIMING DIAGRAM REFERENCE LEVEL = 1.5V



8

SYSTEM CONTROLLER AND BUS DRIVER FOR M5L 8080A P, S CPU

TYPICAL APPLICATION CIRCUIT



MITSUBISHI ELECTRIC

M5L 8085AP, S; P-20, S-20

SINGLE-CHIP 8-BIT N-CHANNEL MICROPROCESSOR

GENERAL 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 M5L 8080A, this device is designed to improve on the M5L 8080A with higher system speed.

FEATURES

- Single 5V power supply
- Software compatibility with the M5L 8080A (with two additional instructions)
- Instruction cycle:
  M5L 8085AP, S: 1.3μs (min)
  M5L 8085AP-20, S-20: 2.0μs (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

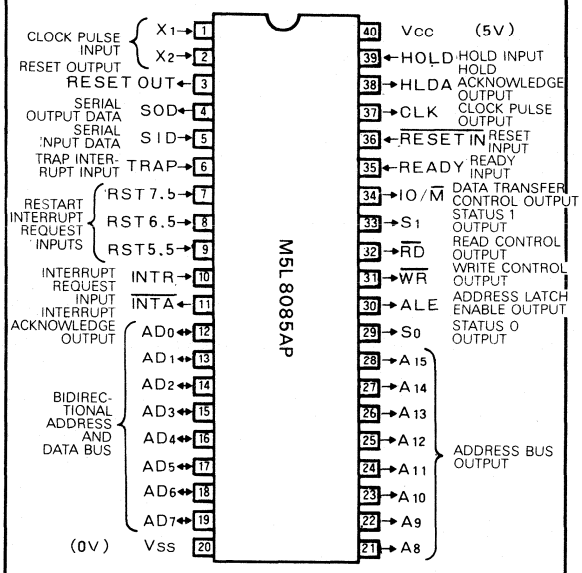
APPLICATION

- Central processing unit for a microcomputer

FUNCTION

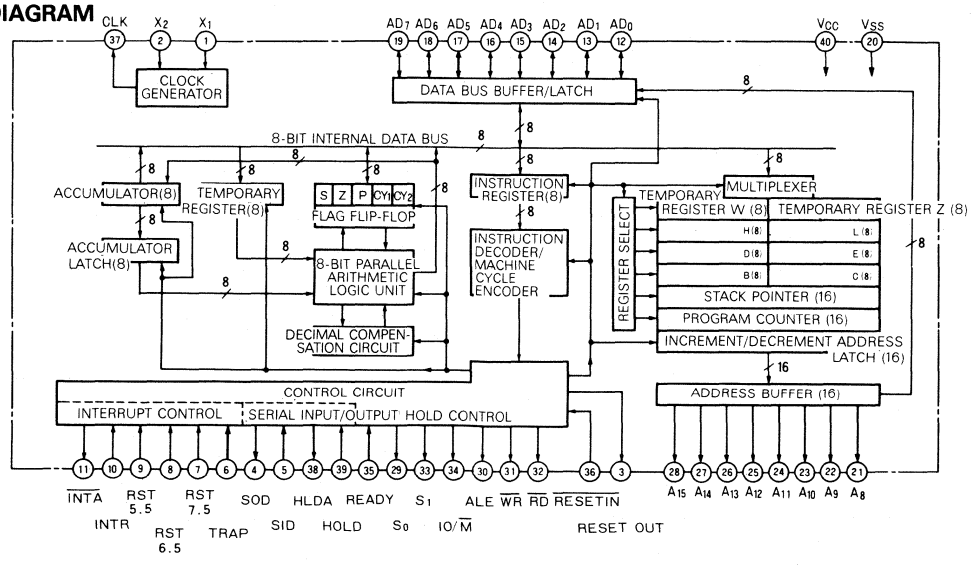
Under the multiplexed data bus concept adopted, the high-order 8 bits of the address are used only as an address bus and the low-order 8 bits are used as an address/data bus. During the first clock cycle of an instruction cycle, the address is transferred. The low-order 8 bits of the address are stored 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 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 (M5L 8085AP), 40S1 (M5L 8085AS)

BLOCK DIAGRAM



8

SINGLE-CHIP 8-BIT N-CHANNEL MICROPROCESSOR

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.

STATUS INFORMATION

Status information can be obtained directly from the M5L 8085A. 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 M5L 8085A 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 M5L 8085A 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 M5L 8080A. 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 M5L 8080A, 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.

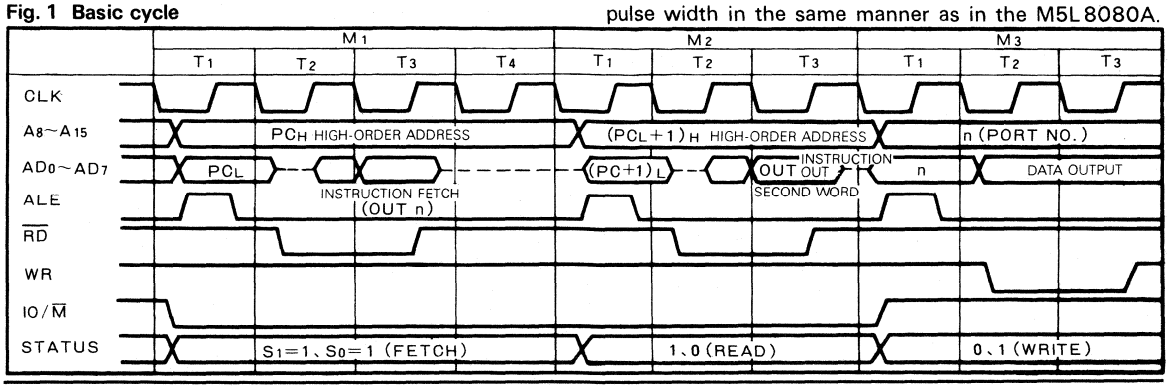
The serial I/O system is also considered to be an interrupt as it is controlled by instructions RIM and SIM. The SID is read by instruction RIM and the SOD data is set by instruction SIM.

8

BASIC TIMING

The M5L 8085A 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 M5L 8085A 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 M5L 8080A.

Fig. 1 Basic cycle



M5L 8085AP, S; P-20, S-20

SINGLE-CHIP 8-BIT N-CHANNEL MICROPROCESSOR

MACHINE INSTRUCTIONS

Instr. class	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 1 1 0 <B2>		7	2	2	(r) ← n	X X X X X			<B2>	I	M4
	MVI M, n	0 0 1 1 0 1 1 0 <B2>	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 0 0 1 <B2> <B3>	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 0 0 1 <B2> <B3>	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 0 0 1 <B2> <B3>	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 0 0 1 <B2> <B3>	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 0 1 0 <B2> <B3>	3 2	13	3	4	(m) ← (A)	X X X X X	m	M4	(A)	O	M4
	LDA m	0 0 1 1 1 0 1 0 <B2> <B3>	3 A	13	3	4	(A) ← (m)	X X X X X	m	M4	(m)	I	M4
	SHLD m	0 0 1 0 0 0 1 0 <B2> <B3>	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 0 1 0 <B2> <B3>	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 1 1 0 <B2>	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 1 1 0 <B2>	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 1 1 0 <B2>	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 1 1 0 <B2>	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 1 1 0 <B2>	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 1 1 0 <B2>	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 1 1 0 <B2>	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 1 1 0 <B2>	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 T1 **: State is T2

MITSUBISHI ELECTRIC

SINGLE-CHIP 8-BIT N-CHANNEL MICROPROCESSOR

Instr. class	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	If condition is true.				
	JNC	m	1 1	0 1 0 <B2> <B3>	0 1 0	D 2	10/7	3	3/2	(CY2) = 0	X X X X X					
	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	X X X X X	If condition is false. (PC)←(PC) + 3				
	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	If condition is true.		<B2> <B3>	I I	M2 M3
	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) + 3 (PC) + 3	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	If condition is true.				
	CNC	m	1 1	0 1 0 <B2> <B3>	1 0 0	D 4	18/9	3	5/2	(CY2) = 0	X X X X X					
	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 (SP) − 2	M4 M5	(PC) + 3	O	M4 M5
	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				
	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	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	EI		1 1	1 1 1	0 1 1	F B	4	1	1	(INTE) ← 1	X X X X X					
	DI		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 accumulator.	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 T_{1_0} **: State is T_{2_0}

Symbol	Meaning
r	Register
m	Two-byte data
n	One-byte data
<B2>	Second byte of instruction
<B3>	Third byte of instruction
AAA	Binary representation for RST instruction n
F	8-bit data from the most to the least significant bit S, Z, X, CY1, X, P, X, CY2
PC	Program counter
SP	Stack pointer

Symbol	Meaning
SSS or DDD	Bit pattern designating register or memory. Where M = (H) (L)

Register or memory	SSS or DDD
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

Symbol	Meaning
←	Data is transferred in direction shown
( )	Contents of register or memoy location
V	Inclusive OR
⊻	Exclusive OR
∧	Logical AND
‾	1's complement
X	Content of flag is not changed after execution
O	Content offlag is set or reset after execution
I	Input mode
O	Output mode

8

SINGLE-CHIP 8-BIT N-CHANNEL MICROPROCESSOR

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.

indicates a three-byte instruction.

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	M5L 8085AS		−65 ~ 150	°C
		M5L 8085AP		−40 ~ 125	°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	2.2		V_{CC}+0.5	V
V_{IL}	Low-level input voltage	−0.3		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	−0.3		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	$0.45V \leqq V_O \leqq V_{CC}$	−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 V_{CC} with the resistor 3kΩ (typ) when $V_I \geqq V_{IH}$ ($\overline{RESIN}$)

8

TIMING REQUIREMENTS (Ta = 0 ~ 70°C, V_{CC} = 5V ± 5%, V_{SS} = 0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	M5L 8085AP, S			M5L 8085AP-20, S-20			Unit
			Limits			Limits			
			Min	Typ	Max	Min	Typ	Max	
$t_{C(CLK)}$	Clock cycle time	T_{CYC}	320		2000	500		2000	ns
$t_{SU(DA-AD)}$	DA input setup time	$-t_{AD}$	−575			−1040			ns
$t_{SU(DA-\overline{RD})}$	DA input setup time	$-t_{RD}$	−300			−590			ns
$t_{h(DA-\overline{RD})}$	DA input hold time	t_{RDH}	0			0			ns
$t_{SU(RDY-AD)}$	READY input setup time	$-t_{ARY}$	−220			−510			ns
$t_{SU(RDY-CLK)}$	READY input setup time	$-t_{RYS}$	110			150			ns
$t_{h(RDY-CLK)}$	READY input hold time	t_{RYH}	0			0			ns
$t_{SU(DA-ALE)}$	DA input setup time	$-t_{LDR}$	−460			−720			ns
$t_{SU(HLD-CLK)}$	HOLD input setup time	t_{HDS}	170			250			ns
$t_{h(HLD-CLK)}$	HOLD input hold time	t_{HDH}	0			0			ns
$t_{SU(INT-CLK)}$	Interrupt setup time	t_{INS}	160			250			ns
$t_{h(INT-CLK)}$	Interrupt hold time	t_{INH}	0			0			ns

Note 2: The input voltage level of the input voltage level is V_{IL} = 0.45V and V_{IH} = 2.4V

SINGLE-CHIP 8-BIT N-CHANNEL MICROPROCESSOR

SWITCHING CHARACTERISTICS (T_a = 0 ~ 70℃, V_{CC} = 5 V ± 5 %, V_{SS} = 0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	M5L 8085AP, S			M5L 8085AP-20, S-20			Unit
			Limits			Limits			
			Min	Typ	Max	Min	Typ	Max	
$t_{W(\overline{CLK})}$	CLK output low-level pulse width	t_1	80			170			ns
$t_{W(CLK)}$	CLK output high-level pulse width	t_2	120			210			ns
$t_{r(CLK)}$	CLK output rise time	t_r			30			30	ns
$t_{f(CLK)}$	CLK output fall time	t_f			30			30	ns
$t_{d(AD-ALE)}$	Delay time, address output to ALE signal	t_{AL}	45			135			ns
$t_{d(ALE-AD)}$	Delay time, ALE signal to address output	t_{LA}	100			180			ns
$t_{W(ALE)}$	ALE pulse width	t_{LL}	140			230			ns
$t_{d(ALE-CLK)}$	Delay time, ALE to CLK	t_{LCK}	100			180			ns
$t_{d(ALE-CONT)}$	Delay time, ALE to control signal	t_{LC}	130			200			ns
$t_{DXZ(RD-AD)}$	Address disable time from read	t_{AFR}			0			0	ns
$t_{DZX(RD-AD)}$	Address enable time from read	t_{RAE}	150			210			ns
$t_{d(CONT-AD)}$	ADdress valid time after control signal	t_{CA}	120			190			ns
$t_{d(DA-WR)}$	Delay time, data output to $\overline{WR}$ signal	t_{DW}	420			670			ns
$t_{d(WR-DA)}$	Delay time, $\overline{WR}$ signal to data output	t_{WD}	100			170			ns
$t_{W(CONT)}$	Control signal pulse width	t_{CC}	400			670			ns
$t_{d(CLK-ALE)}$	Delay time, CLK to ALE signal	t_{CL}	50			120			ns
$t_{d(CLK-HLDA)}$	Delay time, CLK to HLDA signal	t_{HACK}	110			180			ns
$t_{DXZ(HLDA-BUS)}$	Bus disable time from HLDA	t_{HABF}			240			330	ns
$t_{DZX(HLDA-BUS)}$	Bus enable time from HLDA	t_{HABE}			240			330	ns
$t_{d(\overline{CONT}-CONT)}$	Control signal disable time	t_{RV}	400			650			ns
$t_{d(AD-CONT)}$	Delay time, address output to control signal	t_{AC}	240			420			ns

Note 3: at $A_8 \sim A_{15}$; 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, $C_L = 150$pF
M5L 8085AP-20, S-20 $t_{C(CLK)} \geqq 500$ns, $C_L = 150$pF

5: Reference evel for the input/output voltage is V_{OL} = 0.8V, V_{OH} = 2V.

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

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

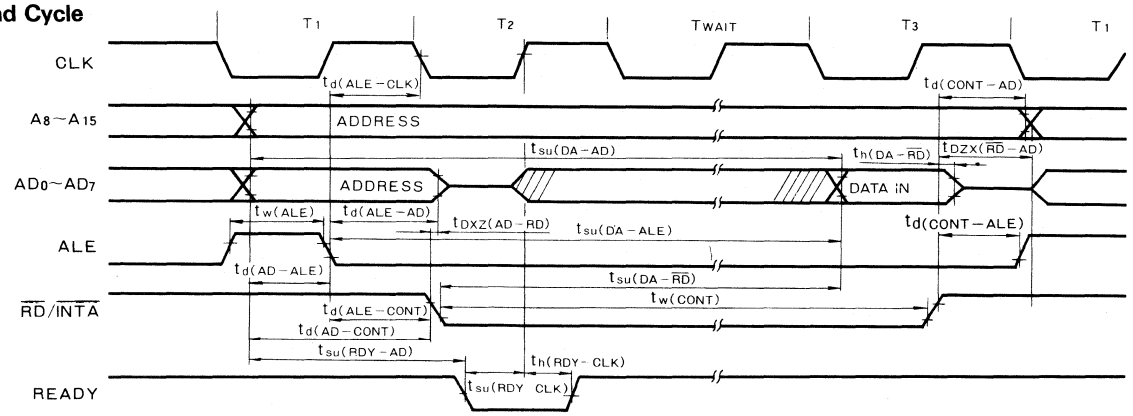
Note 6: N indicates the total number of wait cycles.
$T = t_{C(CLK)}$

Table 1 (continued)

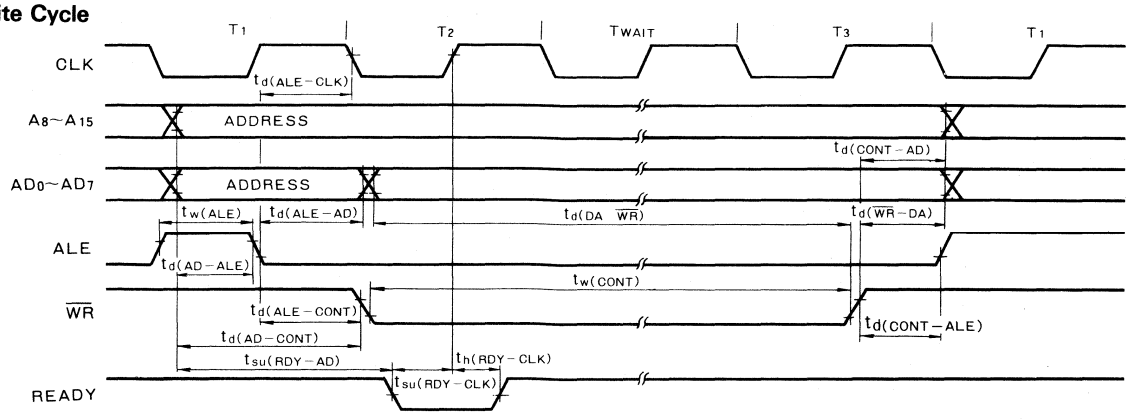
Symbol	Parameter	Alternative symbol	Test conditions	Relational expression (Note 6)	Limit
$t_{W(\overline{CLK})}$	CLK output low-level pulse width	t_1	$C_L = 150pF$	$(1/2)T-80$	Min
$t_{W(CLK)}$	CLK output high-level pulse width	t_2		$(1/2)T-40$	Min
$t_{d(AD-ALE)}$	Delay time, address output to ALE signal	t_{AL}		$(1/2)T-115$	Min
$t_{d(ALE-AD)}$	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
$t_{d(ALE-CLK)}$	Delay time, ALE to CLK	t_{LCK}		$(1/2)T-60$	Min
$t_{d(ALE-CONT)}$	Delay time, ALE to control signal	t_{LC}		$(1/2)T-30$	Min
$t_{DZX(RD-AD)}$	Address enable time from read	t_{RAE}		$(1/2)T-10$	Min
$t_{d(CONT-AD)}$	Address valid time after control signal	t_{CA}		$(1/2)T-40$	Min
$t_{d(DA-WR)}$	Delay time, data output to $\overline{WR}$ signal	t_{DW}		$(3/2+N)T-60$	Min
$t_{d(WR-DA)}$	Delay time, $\overline{WR}$ signal to data output	t_{WD}		$(1/2)T-60$	Min
$t_{W(CONT)}$	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
$t_{d(CLK-HLDA)}$	Delay time, CLK to HLDA signal	t_{HACK}		$(1/2)T-50$	Min
$t_{DXZ(HLDA-BUS)}$	Bus disable time from HLDA	t_{HABF}		$(1/2)T+80$	Max
$t_{DZX(HLDA-BUS)}$	Bus enable time from HLDA	t_{HABE}		$(1/2)T+80$	Max
$t_{d(\overline{CONT}-CONT)}$	Control signal disable time	t_{RV}		$(3/2)T-80$	Min
$t_{d(AD-CONT)}$	Delay time, address output to control signal	t_{AC}		$T-80$	Min

TIMING DIAGRAM

Read Cycle

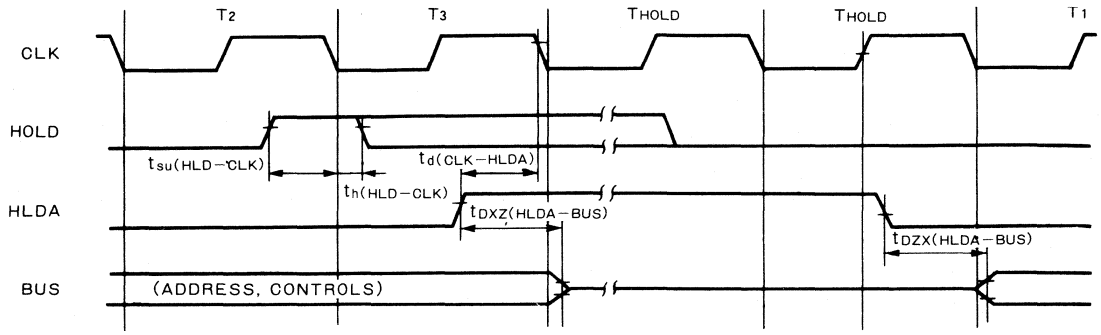


Write Cycle

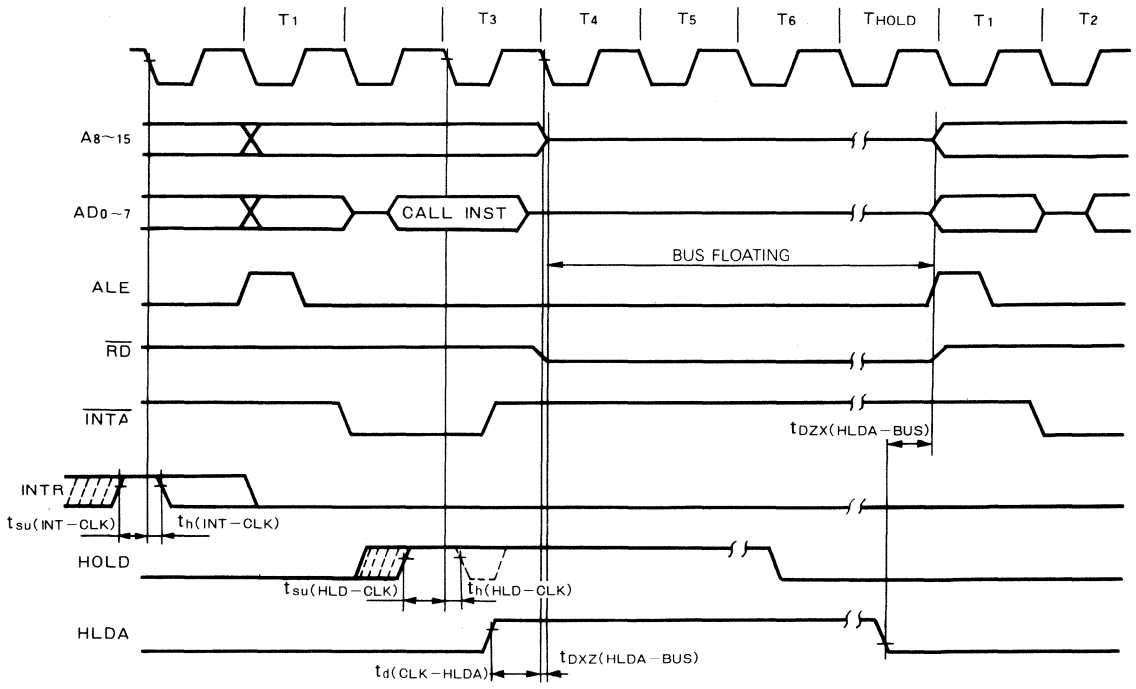


SINGLE-CHIP 8-BIT N-CHANNEL MICROPROCESSOR

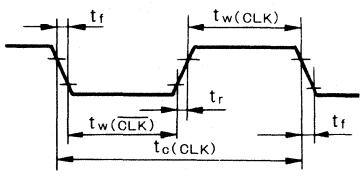
Hold Cycle



Interrupt and Hold Cycle



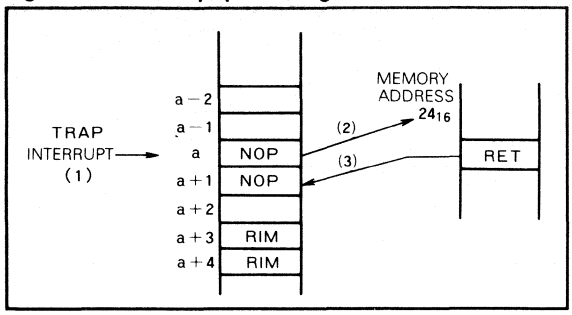
Clock Output Timing Waveform



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 2 and 3.

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 2 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.

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

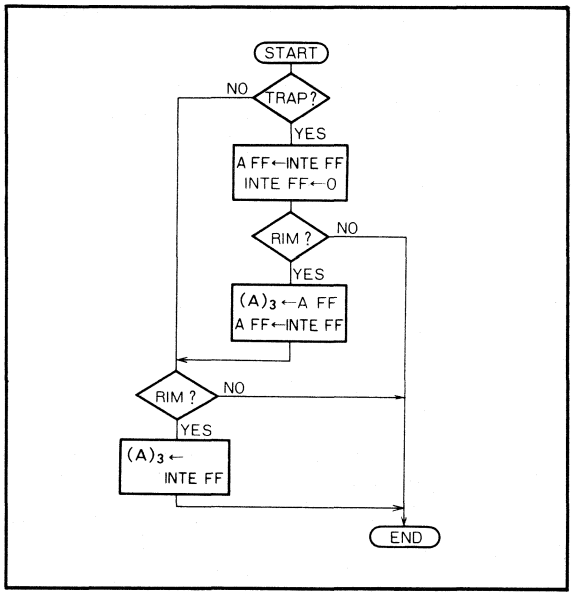
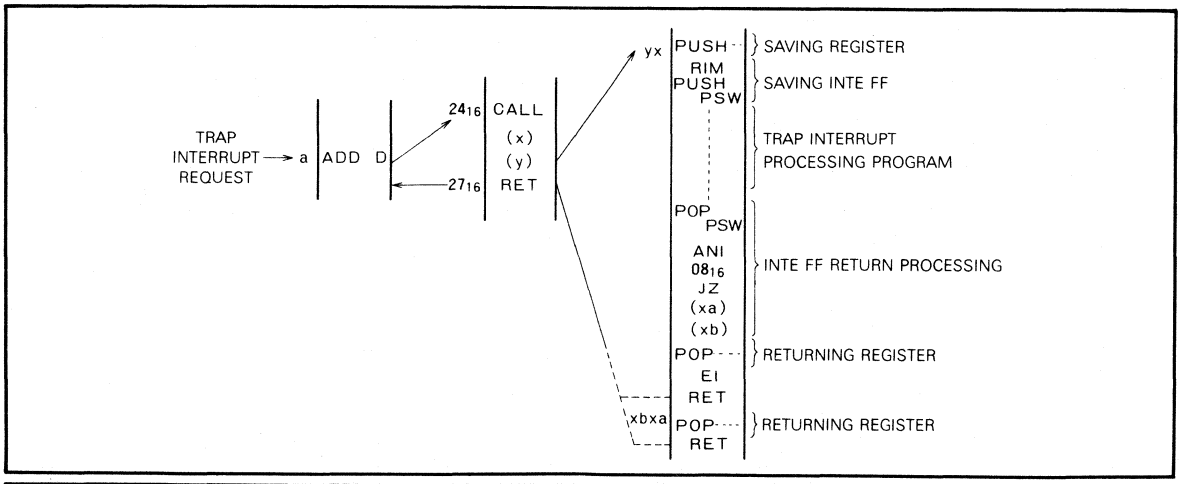


Fig. 3 TRAP interrupt processing routine



SINGLE-CHIP 8-BIT N-CHANNEL MICROPROCESSOR

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

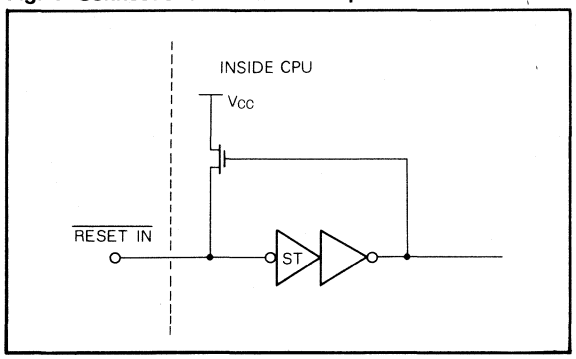
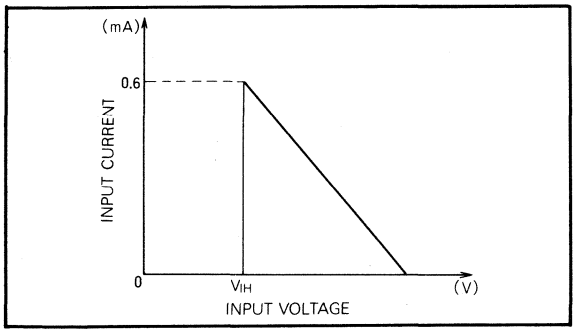


Fig. 5 $\overline{\text{RESET IN}}$ input current vs input voltage



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

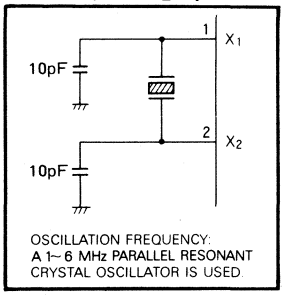
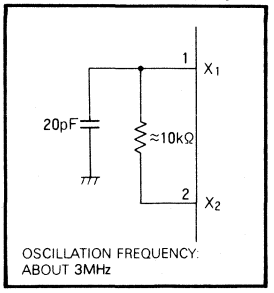


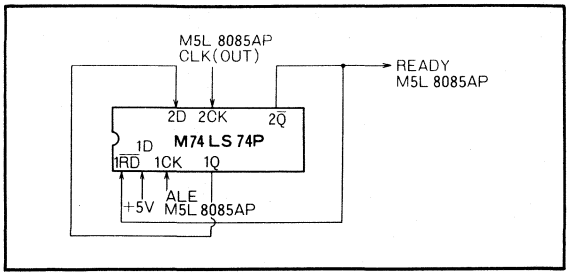
Fig. 7 Connections when RC network is used for X_1 and X_2 inputs



WAIT STATE GENERATOR

Fig. 8 shows a typical 1-wait state generator for low speed RAM and ROM applications.

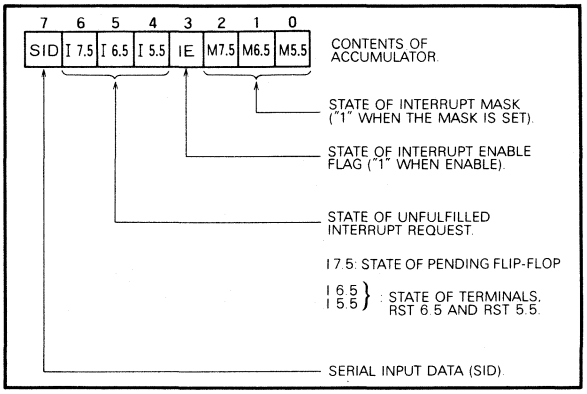
Fig. 8 1-wait state generator



RELATION OF RIM AND SIM INSTRUCTIONS WITH THE ACCUMULATOR (SUPPLEMENTARY DESCRIPTION).

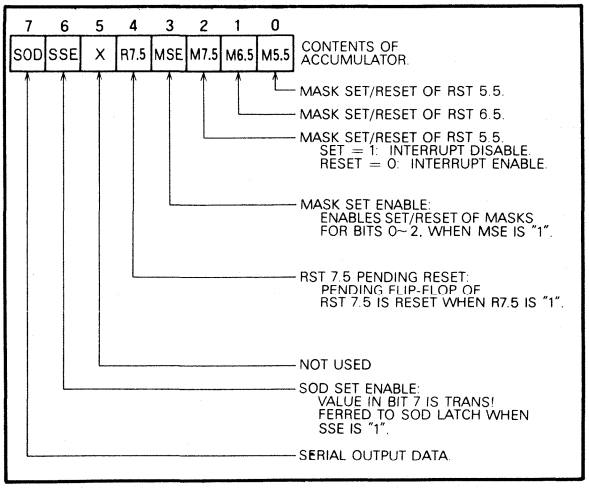
The contents of the accumulator after the execution of a RIM instruction is shown in Table 4.

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



LSIs FOR PERIPHERAL CIRCUITS

9

M58609-04P, S

KEYBOARD ENCODER (JIS CODE STANDARD PRODUCT)

DESCRIPTION

The M58609-04P, S is a keybord encoder for reed switches of terminal equipment. It is fabricated using P-channel aluminum-gate MOS technology and is packaged in a 40-pin DIL package. It contains a 3168-bit mask-programmable read-only memory, and the 8-bit codes specified in JIS C-6220-1969 "Codes for Information Interchange" are stored in the ROM. The output consists of an 8-bit code and a parity bit. The address is selected by the 8-bit and 11-bit ring counters.

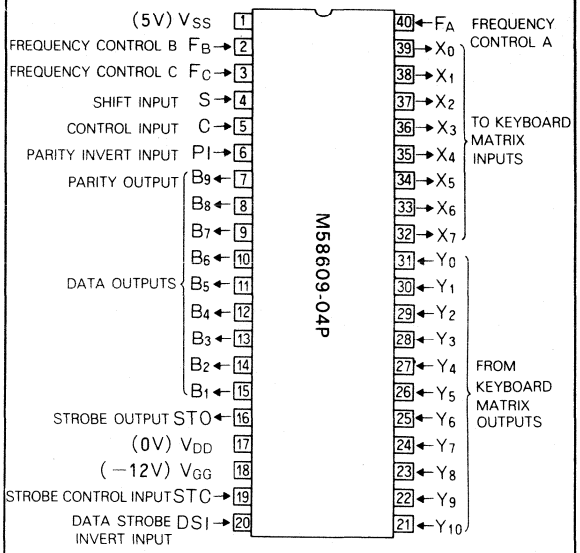
FEATURES

- TTL/DTL-compatible (except X, Y terminals)
- Two-key rollover operation
- Self-contained clock generator circuit
- Strobe delay circuit for eliminating key contact bounce
- External control for output polarity (positive or negative logic)
- External control for selecting odd or even parity

APPLICATION

- Encoder for full-keyboard terminal equipment

PIN CONFIGURATION (TOP VIEW)



Outline 40P1 (M58609-04P)
40S1 (M58609-04S)

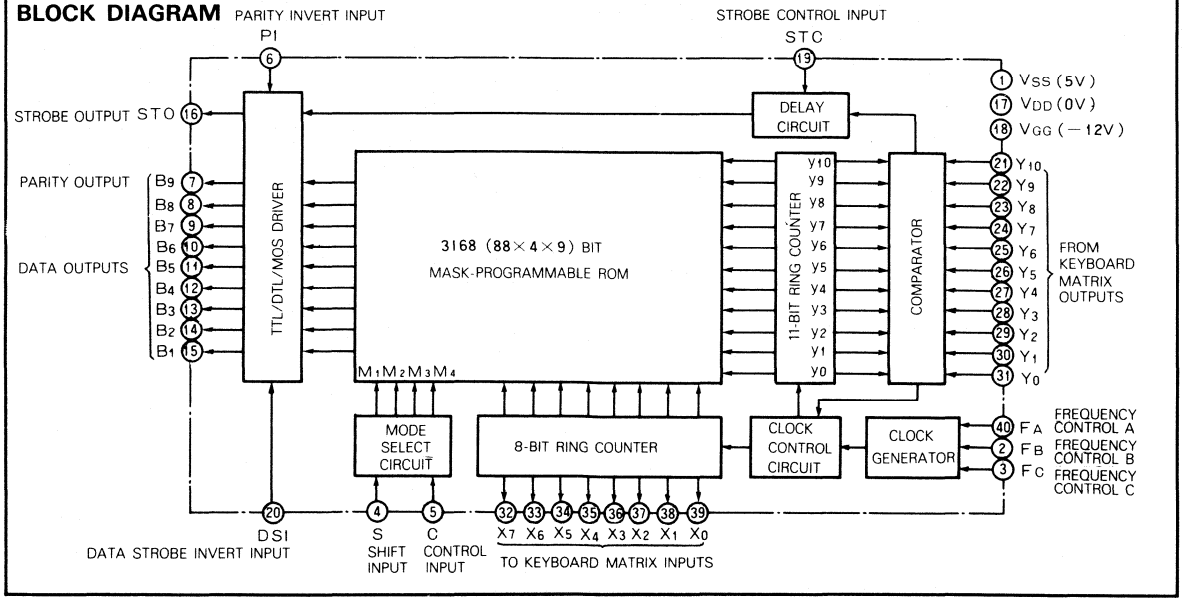
FUNCTION

Outputs (X_0~X_7) of the 8-bit ring counter and inputs (Y_0~Y_{10}) of the 11-bit comparator are wired to the keyboard to form an 8×11 (88-cross points) switch matrix.

When the key connected with X_i and Y_j is depressed, a path is formed between them. When the level of Y_j matches that of X_i, which comes from the 8-bit ring counter, the comparator generates a coincidence signal for clock control and delay circuit. This clock control stops the clock signals to the ring counter and data outputs (B_1~B_9) stabilizing the selected 9-bit code. The stabilization is indicated by a valid signal on the strobe output. A strobe output signal is generated at the time set by the externally controlled delay circuit which receives the coincidence signal. Data outputs and strobe output remain stable until the key is released.

9

BLOCK DIAGRAM



KEYBOARD ENCODER (JIS CODE STANDARD PRODUCT)

FUNCTION (Data Output and Parity Output)

The relationships between B_1~B_8 in the code table and B_1 ~B_8 in data outputs are shown in Table 1, and those between the parity output B_9 and the parity bit, in Table 2. The parity bit in the table is defined as a '0' when the number of '1's in the code B_1~B_8 is odd and a '1' when it is even.

Mode selection is shown in Table 3.

Table 1 Relationship between code table and data outputs

B_1 ~ B_8 Code table	Data strobe invert input DSI	Data output B_1 ~ B_8	Logic
1	L	H	Positive logic
1	H	L	Negative logic
0	L	L	Positive logic
0	H	H	Negative logic

Table 2 Parity output

Parity bit	Parity invert input PI	Parity output B_9
1	L	H
1	H	L
0	L	L
0	H	H

Table 3 Mode selection

Shift input S	Control input C	Selected mode
L	L	1
H	L	2
L	H	3
H	H	4

CODE TABLE (JIS C-6220-1969)

NUMBER OF BITS

B_9*	B_8	B_7	B_6	B_5	B_4	B_3	B_2	B_1	ROW \ COL	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
PARITY BIT										0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
										0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
										0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
										0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
					0	0	0	0	0	NUL	DLE	SP	0	@	P	、					—	タ	ミ		
					0	0	0	1	1	SOH		!	1	A	Q					。	ア	チ	ム		
					0	0	1	0	2	STX		"	2	B	R					「	イ	ツ	メ		
					0	0	1	1	3	ETX		#	3	C	S					」	ウ	テ	モ		
					0	1	0	0	4	EOT		$	4	D	T					、	エ	ト	ヤ		
					0	1	0	1	5	ENQ	NAK	%	5	E	U					・	オ	ナ	ユ		
					0	1	1	0	6	ACK	SYN	&	6	F	V					ヲ	カ	ニ	ヨ		
					0	1	1	1	7	BEL	ETB	'	7	G	W					ァ	キ	ヌ	ラ		
					1	0	0	0	8	BS	CAN	(	8	H	X					ィ	ク	ネ	リ		
					1	0	0	1	9	HT	EM	)	9	I	Y					ゥ	ケ	ノ	ル		
					1	0	1	0	10	LF	SUB	*	:	J	Z					ェ	コ	ハ	レ		
					1	0	1	1	11	VT	ESC	+	;	K	[		{			ォ	サ	ヒ	ロ		
					1	1	0	0	12	FF		,	<	L	¥		|			ャ	シ	フ	ワ		
					1	1	0	1	13	CR		-	=	M	]		}			ュ	ス	ヘ	ン		
					1	1	1	0	14	SO		.	>	N	^		~			ョ	セ	ホ	゛		
					1	1	1	1	15	SI		/	?	O	_		DEL			ッ	ソ	マ	゜		

* B_9 is an odd parity bit for the 8-bit code (B_1 ~ B_8)

Note 1 : A '1' or '0' in the code table indicates that the output level goes high for '1' and low for '0' when input DSI and PI are low-level.

KEYBOARD ENCODER (JIS CODE STANDARD PRODUCT)

CODE ARRANGEMENT TABLE

Yi \ Xi	Mode	X0	X1	X2	X3	X4	X5	X6	X7
Y0	1	NUL	DLE	^	0	;	L	O	9
	2	NUL	DLE	―	NUL	+	NUL	NUL	)
	3	NUL	DLE	ヘ	ワ	レ	リ	ラ	ヨ
	4	NUL	DLE	NUL	ヲ	NUL	NUL	NUL	ョ
Y1	1	SOH	6	0	―	／	K	I	8
	2	SOH	6	0	＝	?	NUL	NUL	(
	3	SOH	6	0	ホ	メ	ノ	ニ	ユ
	4	SOH	6	0	NUL	・	NUL	NUL	ュ
Y2	1	STX	7	1	P	．	J	U	7
	2	STX	7	1	NUL	＞	NUL	NUL	゛
	3	STX	7	1	セ	ル	マ	ナ	ヤ
	4	STX	7	1	NUL	。	NUL	NUL	ャ
Y3	1	ETX	8	2	｛	，	H	Y	6
	2	ETX	8	2	｜	＜	NUL	NUL	&
	3	ETX	8	2	°	ネ	ク	ン	オ
	4	ETX	8	2	「	、	NUL	NUL	ォ
Y4	1	EOT	9	3	¥	M	G	T	5
	2	EOT	9	3	\	NUL	NUL	NUL	%
	3	EOT	9	3	―	モ	キ	カ	エ
	4	EOT	9	3	NUL	NUL	NUL	NUL	ェ
Y5	1	ENQ	NAK	4	BS	N	F	R	4
	2	ENQ	NAK	4	BS	NUL	NUL	NUL	$
	3	ENQ	NAK	4	BS	ミ	ハ	ス	ウ
	4	ENQ	NAK	4	BS	NUL	NUL	NUL	ゥ
Y6	1	ACK	SYN	5	NUL	B	D	E	3
	2	ACK	SYN	5	_	NUL	NUL	NUL	#
	3	ACK	SYN	5	ロ	コ	シ	イ	ア
	4	ACK	SYN	5	NUL	NUL	NUL	ィ	ァ
Y7	1	BEL	ETB	+	〕	V	S	W	2
	2	BEL	ETB	+	｝	NUL	NUL	NUL	"
	3	BEL	ETB	+	ム	ヒ	ト	テ	フ
	4	BEL	ETB	+	」	NUL	NUL	NUL	NUL
Y8	1	=	CAN	SP	CR	C	A	Q	!
	2	=	CAN	SP	CR	NUL	NUL	NUL	!
	3	=	CAN	SP	CR	ソ	チ	タ	ヌ
	4	=	CAN	SP	CR	NUL	NUL	NUL	NUL
Y9	1	SO	EM	・	LF	X	FF	HT	@
	2	SO	EM	・	LF	NUL	FF	HT	`
	3	SO	EM	・	LF	サ	FF	HT	｀
	4	SO	EM	・	LF	NUL	FF	HT	NUL
Y10	1	SI	SUB	―	DEL	Z	ESC	VT	:
	2	SI	SUB	―	DEL	NUL	ESC	VT	*
	3	SI	SUB	―	DEL	ツ	ESC	VT	ケ
	4	SI	SUB	―	DEL	ッ	ESC	VT	NUL

9

SYMBOLOGY

Symbol	Code name	Col/Row in code table	X/Y/Mode in code arrangement table
SP	Space	2 / 0	2/8/1～4
!	Exclamation mark	2 / 1	7/8/2
"	Quotation mark, umlaut	2 / 2	7/7/2
#	Number sign	2 / 3	7/6/2
$	Dollar sign	2 / 4	7/5/2
%	Percentage	2 / 5	7/4/2
&	Ampersand	2 / 6	7/3/2
'	Apostrophe, acute accent	2 / 7	7/2/2
(	Left parenthesis	2 / 8	7/1/2
)	Right parenthesis	2 / 9	7/0/2
*	Asterisk, multiplication sign	2 / 10	7/10/2
+	Positive sign, plus sign	2 / 11	2/7/1～4,4/0/2
,	Comma	2 / 12	4/3/1
―	Negative sign, subtraction sign	2 / 13	2/10/1～4,3/1/1
.	Period	2 / 14	2/9/1～4,4/2/1
/	Slash, virgule, division sign, per	2 / 15	4/1/1
:	Colon	3 / 10	7/10/1
;	Semicolon	3 / 11	4/0/1
<	Less than sign	3 / 12	4/3/2
=	Equal sign	3 / 13	0/8/1～4,3/1/2
>	Greater than sign	3 / 14	4/2/2

Symbol	Coue name	Col/Row in code table	X/Y/Mode in code arrangement table
?	Question mark	3 / 15	4/1/2
@	At mark	4 / 0	7/9/1
[	Left bracket	5 / 11	3/3/1
¥	Yen sign	5 / 12	3/4/1
]	Right bracket	5 / 13	3/7/1
^	Circumflex accent	5 / 14	2/0/1
_	Underline	5 / 15	3/6/2
`	Grave accent	6 / 0	7/9/2
{	Left brace	7 / 11	3/3/2
|	Separate sign, logical add sign	7 / 12	3/4/2
}	Right brace	7 / 13	3/7/2
―	Overline, logical not sign	7 / 14	2/0/2
。	Japanese period	10 / 1	4/2/4
「	Japanese initial quotation mark	10 / 2	3/3/4
」	Japanese final quotation mark	10 / 3	3/7/4
、	Japanese comma	10 / 4	4/3/4
・	Middle dot	10 / 5	4/1/4
―	Long vowel mark	11 / 0	3/4/3
゛	Voiced consonant mark	13 / 14	7/9/3
°	Semi-voiced consonant mark	13 / 15	3/3/3

KEYBOARD ENCODER (JIS CODE STANDARD PRODUCT)

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{GG}	Supply voltage	With respect to V_{SS}	0.3~−20	V
V_{DD}	Supply voltage		0.3~−20	V
V_I	Input voltage		0.3~−20	V
Topr	Operating free-air temperature range		−20~75	°C
Tstg	Storage temperature range		−40~125	°C

RECOMMENDED OPERATING CONDITIONS (Ta = −20~75°C, unless otherwise noted)

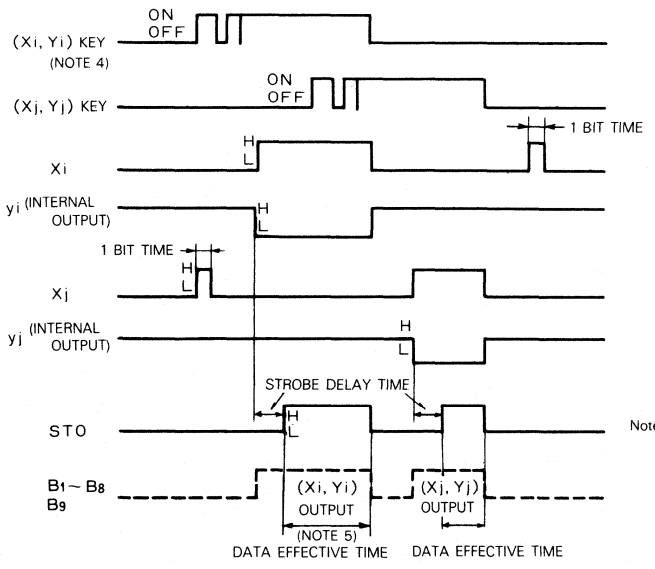
	Parameter	Limits			Unit
		Min	Nom	Max	
V_{GG}	Supply voltage	−11	−12	−13	V
V_{DD}	Supply voltage		0		V
V_{SS}	Supply voltage	4.5	5	5.5	V
V_{IH}	High-level input voltage	$V_{SS}-1$			V
V_{IL}	Low-level input voltage			0.8	V
$f_{(\phi)}$	Clock frequency	20	50	100	kHz
$t_{D(STO)}$	Strobe delay time		1.5		ms
R_{OFF}	Switch off resistance	10			MΩ
R_{ON}	Switch on resistance			300	Ω

ELECTRICAL CHARACTERISTICS (Ta = −20~75°C, V_{GG} = −12±1V, V_{SS} = 5±0.5V, V_{DD} = 0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$V_{OH(B_1,STO)}$	High-level output voltage, B_1~B_9 and STO	I_{OH} = −100μA	$V_{SS}-1$			V
$V_{OH(X_i)}$	High-level output voltage, X_0~X_7	I_{OH} = −100μA	$V_{SS}-1.3$			V
$V_{OL(B_i,STO)}$	Low-level output voltage, B_1~B_9 and STO	I_{OL} = 1.6mA			0.4	V
$V_{OL(X_i)}$	Low-level output voltage, X_0~X_7	I_{OL} = 1μA			−3	V
R_I	Input resistance, S, C, DSI and PI	V_I = −12V	1			MΩ
Pd	Power dissipation	Ta = 25°C		70	200	mW
C_I	Input capacitance	V_I = 0V, f = 1MHz, Ta = 25°C			15	pF

Note 2 : Current flowing into an IC is positive; out is negative.

TIMING DIAGRAM



Note 3 : DSI = "L"

4 : (Xi, Yi) KEY indicates the key switch that is located at the cross point of Xi and Yi of the keyboard matrix.

5 : (Xi, Yi) OUTPUT indicates the code output of the key that is selected by the (Xi, Yi) KEY.

FUNCTION TABLES

Data (B_1~B_9) Invert

DSI (Pin ⑳)	Code table (B_1~B_9)	Data output (B_1~B_9)
H	1	L
L	1	H
H	0	H
L	0	L

Strobe (STO) Invert

DSI (Pin⑳)	Internal strobe (Note 3)	STO (Pin⑮)
H	H	L
L	H	H
H	L	H
L	L	L

Parity (B_9) Invert

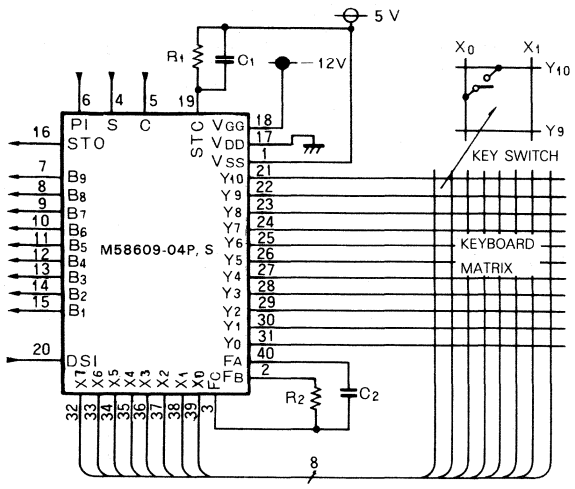
PI (Pin⑥)	Code table (B_9)	B_9 (Pin⑦)
H	1	L
L	1	H
H	0	H
L	0	L

Mode Select

S (Pin④)	C (Pin⑤)	Mode
L	L	M_1
H	L	M_2
L	H	M_3
H	H	M_4

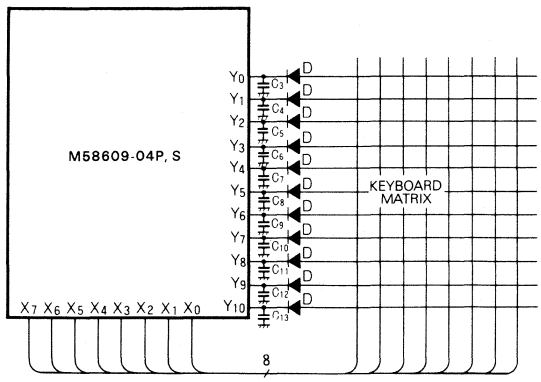
Note 6 : The internal signal of the strobe output (STO) becomes high-level when the strobe signal is generated.

TYPICAL APPLICATION CIRCUIT



Note 7 : $R_1 = 1.5M\Omega$, $C_1 = 0.001\mu F$ provides approximately 1.5ms delay time.
8 : $R_2 = 75k\Omega$, $C_2 = 50pF$ provides approximately 50kHz clock frequency.

Fig. 1 Measures against coupling



Insert diodes and capacitors (C_3~C_{13} should be approx. 50~100pF) between the keyboard matrix and the Y inputs.

Cautions in Use

1. Coupling at the Keyboard Matrix

Depending on the capacitance of the keyboard matrix wiring, depressing one key while another is depressed gives rise to capacitance coupling, which may result in repetition of the strobe output, the same condition that would occur if a single key were depressed twice. In this case, proceed as shown in Fig. 1.

2. N-Key Rollover

This device is for 2-key rollover; when 3 or more keys are depressed simultaneously, code output is indeterminate.

3. Maximum Chatter Times

1. With Key Off (t_{KOFF})

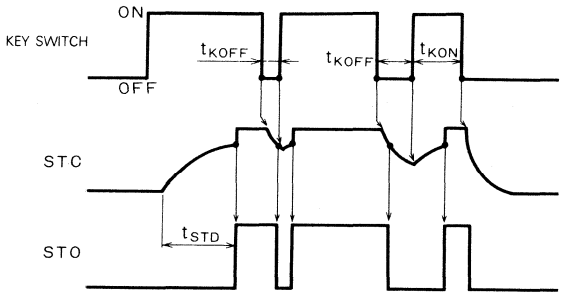
t_{KOFF} is defined as the maximum time that, when a key is turned on but because of faulty contact, etc. the key input signal chatters, the chattering can be disregarded. This may vary according to ambient temperature, power supply conditions, etc., but is approximately 100ns.

2. With Key On (t_{KON})

Conversely, t_{KON} is the maximum time that, when a key is turned off, but the input signal chatters, the chattering condition can be disregarded.

when $t_{KOFF} \leqq 1/f_\phi$ $t_{KON} \leqq 100ns$
when $t_{KOFF} > 1/f_\phi$ $t_{KON} \leqq t_{STD} + t_{SC}$
where: f_ϕ = clock frequency
t_{STD} = strobe delay time
t_{SC} = scanning time

Fig. 2 Timing diagram for maximum chatter time

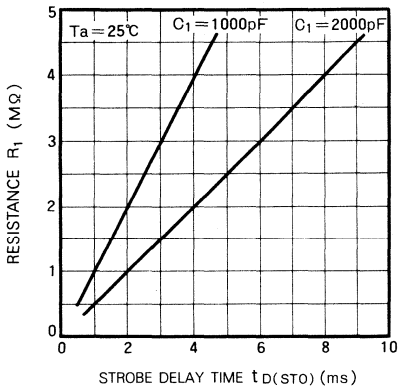


9

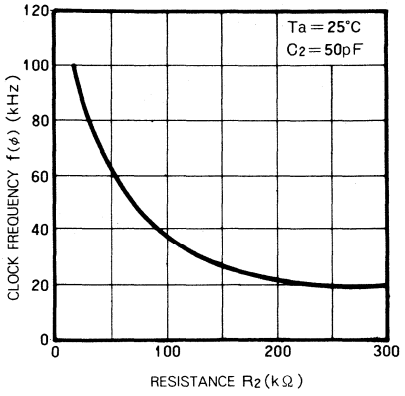
KEYBOARD ENCODER (JIS CODE STANDARD PRODUCT)

TYPICAL CHARACTERISTICS (V_{GG} = −12V, V_{DD} = 0 V, V_{SS} = 5 V)

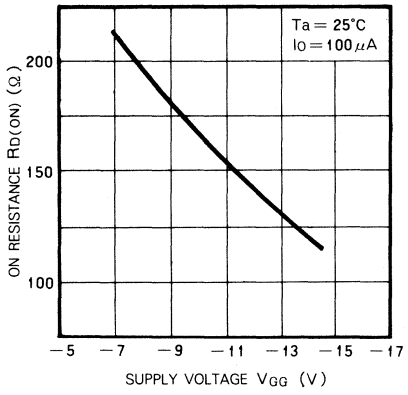
RESISTANCE VS. STROBE DELAY TIME



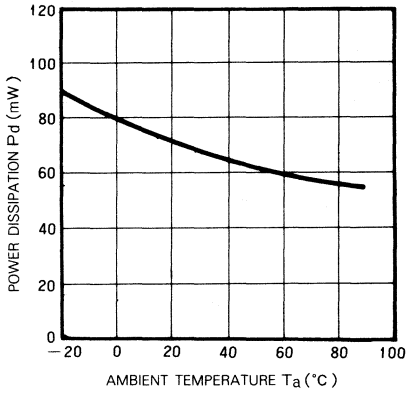
CLOCK FREQUENCY VS. RESISTANCE



ON RESISTANCE OF OUTPUT DRIVER VS. POWER SUPPLY VOLTAGE



POWER DISSIPATION VS. AMBIENT TEMPERATURE



KEYBOARD ENCODER (USASCII CODE STANDARD PRODUCT)

DESCRIPTION

The M58609-09P, S has the 7-bit codes specified in US ASCII "Codes for Information Interchange". The codes can be odd or even parity. The function, pin configuration and electrical characteristics are the same as those of an M58609-04P, S.

FUNCTION (Data Output and Parity Output)

The relationships between B_1~B_8 in the code table and B_1 ~B_8 in data outputs are shown in Table 1, and those between the parity output B_9 and the parity bit, in Table 2. The parity bit in the table is defined as a '0' when the number of '1's in the code B_1~B_8 is odd and a '1' when it is even.

Mode selection is shown in Table 3.

Table 1 Relationship between code table and data outputs

B_1 ~ B_8 Code table	Data strobe invert input DSI	Data output B_1 ~ B_8	Logic
1	L	H	Positive logic
1	H	L	Negative logic
0	L	L	Positive logic
0	H	H	Negative logic

Table 2 Parity output

Parity bit	Parity invert input PI	Parity output B_9
1	L	H
1	H	L
0	L	L
0	H	H

Table 3 Mode selection

Shift input S	Control input C	Selected mode
L	L	1
H	L	2
L	H	3
H	H	4

CODE TABLE

NUMBER OF BITS									0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
B_9*	B_8	B_7	B_6	B_5	B_4	B_3	B_2	B_1	ROW \ COL	0	1	2	3	4	5	6	7
					0	0	0	0	0	NUL	DLE	SP	0	@	P		p
					0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
					0	0	1	0	2	STX	DC2	"	2	B	R	b	r
					0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
					0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
					0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
					0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
					0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
					1	0	0	0	8	BS	CAN	(	8	H	X	h	x
					1	0	0	1	9	HT	EM	)	9	I	Y	i	y
					1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
					1	0	1	1	11	VT	ESC	+	;	K	[	k	{
					1	1	0	0	12	FF	FS	,	<	L	\	l	¦
					1	1	0	1	13	CR	GS	−	=	M	]	m	}
					1	1	1	0	14	SO	RS	.	>	N	^	n	~
					1	1	1	1	15	SI	US	/	?	O	—	o	DEL

* B_8 is an even parity bit for the 8-bit code (B_0~B_7). B_9 is an odd parity bit.

Note 1 : A '1' or '0' in the code table indicates that the output level goes high for '1' and low for '0' when input DSI and PI are low-level.

KEYBOARD ENCODER (USASCII CODE STANDARD PRODUCT)

CODE ARRANGEMENT TABLE

Y_i \ X_i	Mode	X_0	X_1	X_2	X_3	X_4	X_5	X_6	X_7
Y_0	1	/	;	P	0	·	l	o	9
	2	?	+	P	0	>	L	O	)
	3	/	;	DLE	0	·	FF	SI	9
	4	?	+	DLE	0	>	FF	SI	)
Y_1	1	·	:	@	—	'	k	i	8
	2	·	*	\	=	<	K	I	(
	3	·	:	@	—	'	VT	HT	8
	4	·	*	\	=	<	VT	HT	(
Y_2	1	'	\	[	]	m	j	u	7
	2	·	|	¦	¦	M	J	U	,
	3	'	FS	ESC	GS	CR	LF	NAK	7
	4	'	FS	ESC	GS	CR	LF	NAK	,
Y_3	1	GS	LF	—	^	n	h	y	6
	2	GS	LF	—	~	N	H	Y	&
	3	GS	LF	US	RS	SO	BS	EM	6
	4	GS	LF	US	RS	SO	BS	EM	&
Y_4	1	FS	DEL	CR	BS	b	g	t	5
	2	FS	DEL	CR	BS	B	G	T	%
	3	FS	DEL	CR	BS	STX	BEL	DC4	5
	4	FS	DEL	CR	BS	STX	BEL	DC4	%
Y_5	1	RS	US	BEL	NUL	v	f	r	4
	2	RS	US	BEL	NUL	V	F	R	$
	3	RS	US	BEL	NUL	SYN	ACK	DC2	4
	4	RS	US	BEL	NUL	SYN	ACK	DC2	$
Y_6	1	0	1	4	7	c	d	e	3
	2	0	1	4	7	C	D	E	#
	3	0	1	4	7	ETX	EOT	ENQ	3
	4	0	1	4	7	ETX	EOT	ENQ	#
Y_7	1	·	2	5	8	x	s	w	2
	2	·	2	5	8	X	S	W	"
	3	·	2	5	8	CAN	DC3	ETB	2
	4	·	2	5	8	CAN	DC3	ETB	"
Y_8	1	=	3	6	9	z	a	q	1
	2	=	3	6	9	Z	A	Q	!
	3	=	3	6	9	SUB	SOH	DC1	1
	4	=	3	6	9	SUB	SOH	DC1	!
Y_9	1	+	—	*	/	SP	LF	ESC	HT
	2	+	—	*	/	SP	LF	ESC	HT
	3	+	—	*	/	SP	LF	ESC	HT
	4	+	—	*	/	SP	LF	ESC	HT
Y_{10}	1	ENQ	ACK	SUB	EM	CR	DEL	FF	VT
	2	ENQ	ACK	SUB	EM	CR	DEL	FF	VT
	3	ENQ	ACK	SUB	EM	CR	DEL	FF	VT
	4	ENQ	ACK	SUB	EM	CR	DEL	FF	VT

SYMBOLOGY

Symbol	Code name	Col/Row in code table	X/Y/Mode in code arrangement table
SP	Space	2/0	4/9/1~4
!	Exclamation mark	2/1	7/8/2.4
"	Quotation mark, umlaut	2/2	7/7/2.4
#	Number sign	2/3	7/6/2.4
$	Dollar sign	2/4	7/5/2.4
%	Percentage	2/5	7/4/2.4
&	Ampersand	2/6	7/3/2.4
'	Apostrophe, acute accent	2/7	7/2/2.4
(	Left parenthesis	2/8	7/1/2.4
)	Right parenthesis	2/9	7/0/2.4
*	Asterisk, multiplication sign	2/10	2/9/1~4, 1/1/2.4
+	Positive sign, plus sign	2/11	0/9/1~4, 1/0/2.4
,	Comma	2/12	0/2/1~4, 4/1/1.3
—	Negative sign, subtraction sign	2/13	1/9/1~4, 3/1/1.3
.	Period	2/14	0/1/1~4, 0/7/1~4, 4/0/1.3
/	Slash, virgule, division sign, per	2/15	3/9/1~4, 0/0/1.3

Symbol	Code name	Col/Row in code table	X/Y/Mode in code arrangement table
:	Colon	3/10	1/1/1, 3
;	Semicolon	3/11	1/0/1.3
<	Less than sign	3/12	4/1/2.4
=	Equal sign	3/13	0/8/1~4, 3/1/2.4
>	Greater than sign	3/14	4/0/2.4
?	Question mark	3/15	0/0/2.4
@	At mark	4/0	2/1/2.4
[	Left bracket	5/11	2/2/1
]	Right bracket	5/13	3/2/1
^	Circumflex accent	5/14	3/3/1
_	Underline	5/15	2/3/1.2
¦	Left brace	7/11	2/2/2
|	Separate sign, logical add sign	7/12	1/2/2
¦	Right brace	7/13	3/2/2
\	Reverse slant	5/12	2/1/2.4, 1/2/1
~	Swung dash	7/14	3/3/2

MITSUBISHI ELECTRIC

KEYBOARD ENCODER (JIS CODE STANDARD PRODUCT)

DESCRIPTION

The M58620-001S is a keyboard encoder for solid-state switches and is fabricated with P-channel aluminum-gate MOS technology.

It contains a 3640-bit mask-programmable read-only memory, and the 7-bit and 8-bit codes specified in JIS publication C-6220-1969 "Codes for Information interchange" are stored in the ROM. The mode shift is selected by the combination of shift input, control input and shift control input. The output consists of a 9-bit plus parity bit code. All inputs and outputs are TTL-compatible.

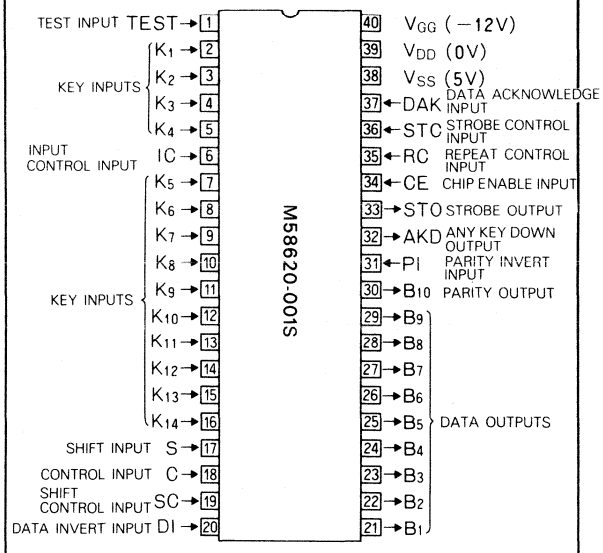
FEATURES

- All inputs and outputs are TTL-compatible
- Output buffer register
- Strobe inhibit circuit for unused codes
- One shot output (the pulse width is variable) or static output for strobed output
- Chip enable terminal
- 2-key rollover capability (N-key rollover is also available, if the logic output of the switches is pulsive)

APPLICATION

- Encoder for full-keyboard terminal equipment

PIN CONFIGURATION (TOP VIEW)



Outline 40S1

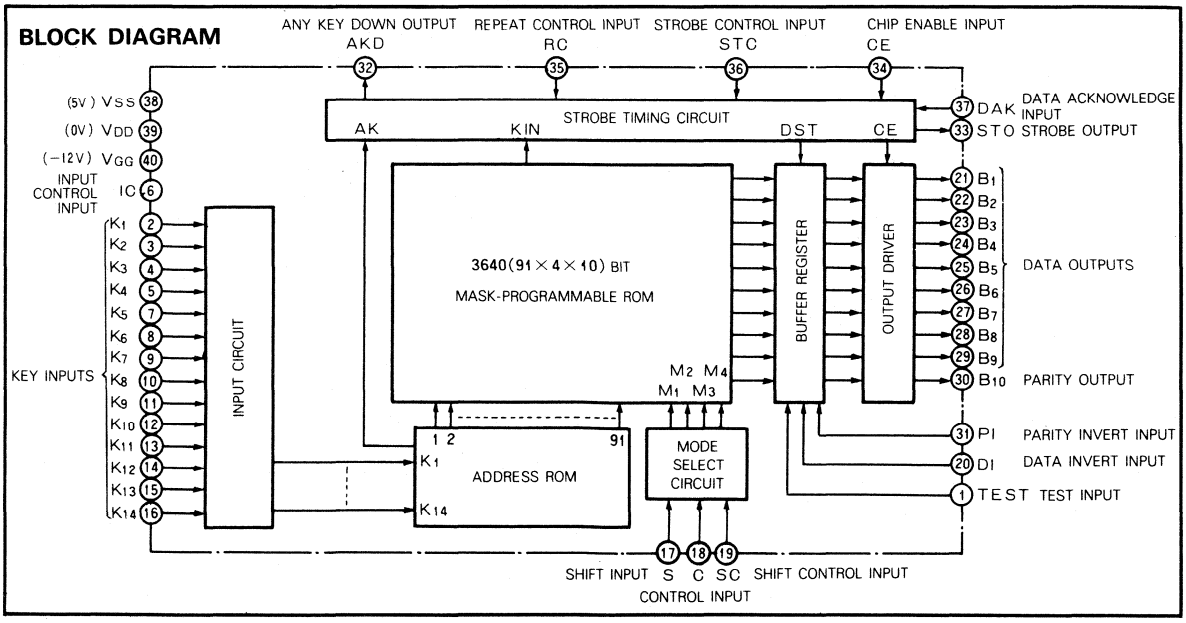
FUNCTION

The output of each keyboard switch is connected to 2-key inputs selected from K_1~K_{14} (2 of 14) to form 91 addresses. Therefore, the character code for output is selected by 2 of 14 key inputs, shift input, control input and shift control input.

When a key is depressed, the output of that keyboard switch is applied to two key inputs selected from K_1~K_{14}; the address ROM generates an address that is used for input to the 3640-bit ROM. After the encoded data from the ROM is transferred to the buffer register, a strobed output is generated, validating the encoded data.

9

BLOCK DIAGRAM



KEYBOARD ENCODER (JIS CODE STANDARD PRODUCT)

FUNCTION (Data Output and Parity Output)

The relationships between B_1~B_8 in the code table and B_1 ~B_8 in data outputs are shown in Table 1, and those between the parity output (B_{10} or B_9) and the parity bit, in Tables 2 and 3. The parity bit in the tables is defined as a '0' when the number of '1's in the code (B_1~B_8 or B_1~B_7) is odd and a '1' when it is even.

Mode selection is shown in Table 4.

Table 1 Relation between code table and outputs

B_1 ~ B_8 in code table	Data invert input DI	Data output B_1 ~ B_8	Logic
1	L	H	Positive logic
1	H	L	Negative logic
0	L	L	Positive logic
0	H	H	Negative logic

Table 2 Parity output of 8-bit code

Parity bit	Parity invert output PI	Parity output B_{10}
1	L	H
1	H	L
0	L	L
0	H	H

Table 3 Parity output of 7-bit code

Parity bit	Data invert input DI	Data output B_9
1	L	H
1	H	L
0	L	L
0	H	H

Table 4 Mode selection

Shift input S	Control input C	Shift control input SC	Selected mode
L	L	L	1
H	L	L	2
L	H	L	3
H	H	L	4
L	L	H	4
H	L	H	—
L	H	H	—
H	H	H	—

CODE TABLE (JIS C-6220-1969)

NUMBER OF BITS																					
									0	0	0	0	0	0	0	0	1	1	1	1	1
									0	0	0	0	1	1	1	1	0	0	0	0	1
									0	0	1	1	0	0	1	1	0	0	1	1	0
									0	1	0	1	0	1	0	1	0	1	0	1	0
B_{10}†	B_9*	B_8	B_7	B_6	B_5	B_4	B_3	B_2	B_1	ROW / COL	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
PARITY BIT						0	0	0	0	0	NUL	DLE	SP	0	@	P	`					—	タ	ミ		
						0	0	0	1	1	SOH	DC1	!	1	A	Q					。	ア	チ	ム		
						0	0	1	0	2	STX	DC2	"	2	B	R					「	イ	ツ	メ		
						0	0	1	1	3	ETX	DC3	#	3	C	S					」	ウ	テ	モ		
						0	1	0	0	4	EOT		$	4	D	T					、	エ	ト	ヤ		
						0	1	0	1	5	ENQ	NAK	%	5	E	U					・	オ	ナ	ユ		
						0	1	1	0	6	ACK	SYN	&	6	F	V					ヲ	カ	ニ	ヨ		
						0	1	1	1	7	BEL	ETB	'	7	G	W					ァ	キ	ヌ	ラ		
						1	0	0	0	8	BS	CAN	(	8	H	X					ィ	ク	ネ	リ		
						1	0	0	1	9	HT	EM	)	9	I	Y					ゥ	ケ	ノ	ル		
						1	0	1	0	10	LF	SUB	*	:	J	Z					ェ	コ	ハ	レ		
						1	0	1	1	11	VT	ESC	+	;	K	[		{			ォ	サ	ヒ	ロ		
						1	1	0	0	12	FF		,	<	L	¥		|			ャ	シ	フ	ワ		
						1	1	0	1	13	CR		−	=	M	]		}			ュ	ス	ヘ	ン		
						1	1	1	0	14	SO		.	>	N	^		~			ョ	セ	ホ	゛		
						1	1	1	1	15	SI		/	?	O	_		DEL			ッ	ソ	マ	゜		

* B_9 is an odd parity bit for the 7-bit code (B_1~B_7).

† B_{10} is an odd parity bit for the 8-bit code (B_1~B_8).

Note 1 : When inputs DI and PI are low-level, a '1' in the code table indicates that the output level goes high, a '0' that it goes low.

KEYBOARD ENCODER (JIS CODE STANDARD PRODUCT)

CODE ARRANGEMENT TABLE

Kn \ Km	Mode	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12	K13	K14
K1	1	Z	X	C	V	B	N	M	,	.	／		A	S
	2								<	>	?	_		
	3	ツ	サ	ソ	ヒ	コ	ミ	モ	ネ	ル	メ	ロ	チ	ト
	4	ッ							､	｡	･			
K2	1		D	F	G	H	J	K	L	;	:	)	Q	W
	2					・				+	*	｝		
	3		シ	ハ	キ	ク	マ	ノ	リ	レ	ケ	ム	タ	テ
	4											｣		
K3	1			E	R	T	Y	U	I	O	P	@	〔	1
	2										｡	､	｛	!
	3			イ	ス	カ	ン	ナ	ニ	ラ	セ	゛	｡	ヌ
	4			ｨ									｢	
K4	1				2	3	4	5	6	7	8	9	0	—
	2				"	#	$	%	&	'	(	)		=
	3				フ	ア	ウ	エ	オ	ヤ	ユ	ヨ	ワ	ホ
	4					ｱ	ｳ	ｴ	ｵ	ｬ	ｭ	ｮ	ﾜ	
K5	1					^	¥	DEL	SP	SOH	STX	ETX	EOT	ENQ
	2					―	|	DEL	SP	SOH	STX	ETX	EOT	ENQ
	3					へ	―	DEL	SP	SOH	STX	ETX	EOT	ENQ
	4							DEL	SP	SOH	STX	ETX	EOT	ENQ
K6	1						ACK	BEL	BS	HT	LF	VT	FF	CR
	2						ACK	BEL	BS	HT	LF	VT	FF	CR
	3						ACK	BEL	BS	HT	LF	VT	FF	CR
	4						ACK	BEL	BS	HT	LF	VT	FF	CR
K7	1							SO	SI	DLE	DC1	DC2	DC3	NAK
	2							SO	SI	DLE	DC1	DC2	DC3	NAK
	3							SO	SI	DLE	DC1	DC2	DC3	NAK
	4							SO	SI	DLE	DC1	DC2	DC3	NAK
K8	1								SYN	ETB	CAN	EM	SUB	ESC
	2								SYN	ETB	CAN	EM	SUB	ESC
	3								SYN	ETB	CAN	EM	SUB	ESC
	4								SYN	ETB	CAN	EM	SUB	ESC
K9	1									NUL	+	—	=	・
	2									NUL	+	—	=	・
	3									NUL	+	—	=	・
	4									NUL	+	—	=	・
K10	1										1	2	3	4
	2										1	2	3	4
	3										1	2	3	4
	4										1	2	3	4
K11	1											5	6	7
	2											5	6	7
	3											5	6	7
	4											5	6	7
K12	1												8	9
	2												8	9
	3												8	9
	4												8	9
K13	1													0
	2													0
	3													0
	4													0

SYMBOLOGY

Symbol	Code name	Col/Row in code table	km/kn/Mode in code arrangement table
SP	Space	2 ／ 0	K9 / K5 / 1～4
!	Exclamation mark	2 ／ 1	K14 / K3 / 2
"	Quotation mark, umlaut	2 ／ 2	K5 / K4 / 2
#	Number sign	2 ／ 3	K6 / K4 / 2
$	Dollar sign	2 ／ 4	K7 / K4 / 2
%	Percentage	2 ／ 5	K8 / K4 / 2
&	Ampersand	2 ／ 6	K9 / K4 / 2
'	Apostrophe, acute accent	2 ／ 7	K10 / K4 / 2
(	Left parenthesis	2 ／ 8	K11 / K4 / 2
)	Right parenthesis	2 ／ 9	K12 / K4 / 2
*	Asterisk, multiplication sign	2 ／10	K11 / K2 / 2
+	Positive sign, plus sign	2 ／11	K10 / K2 / 2 *
,	Comma	2 ／12	K9 / K1 / 1
—	Negative sign, subtraction sign	2 ／13	K14 / K4 / 1 *
.	Period	2 ／14	K10 / K1 / 1 *
／	Slash, virgule division sign, per	2 ／15	K11 / K1 / 1
:	Colon	3 ／10	K11 / K2 / 1
;	Semicolon	3 ／11	K10 / K2 / 1
<	Less than sign	3 ／12	K9 / K1 / 2
=	Equal sign	3 ／13	K14 / K4 / 2 *
>	Greater than sign	3 ／14	K10 / K4 / 2

*See K11～K14/K9/1～4

Symbol	Code name	Col/Row in code table	km/kn/Mode in code arrangement table
?	Question mark,	3 ／15	K11 / K1 / 2
@	At mark	4 ／ 0	K12 / K3 / 1
[	Left bracket	5 ／11	K13 / K3 / 1
¥	Yen sign	5 ／12	K7 / K5 / 1
]	Right bracket	5 ／13	K12 / K2 / 1
^	Circumflex accent	5 ／14	K6 / K5 / 1
_	Underline	5 ／15	K12 / K1 / 2
`	Grave accent	6 ／ 0	K12 / K3 / 2
{	Left brace	7 ／11	K13 / K3 / 2
|	Separate sign, logical add sign	7 ／12	K7 / K5 / 2
}	Right brace	7 ／13	K12 / K2 / 2
―	Overline, logical not sign	7 ／14	K6 / K5 / 2
｡	Japanese period	10／ 1	K10 / K1 / 4
｢	Japanese initial quotation mark	10／ 2	K13 / K3 / 4
｣	Japanese final quotation mark	10／ 3	K12 / K2 / 4
､	Japanese comma	10／ 4	K9 / K1 / 4
･	Middle dot	10／ 5	K11 / K1 / 4
ｰ	Long vowel mark	11／ 0	K7 / K5 / 3
ﾞ	Voiced consonant mark	13／14	K12 / K3 / 3
ﾟ	Semi-voiced consonant mark	13／15	K13 / K3 / 3

9

KEYBOARD ENCODER (JIS CODE STANDARD PRODUCT)

OPERATION

1. 2-Key Rollover (N-Key Lockout)

When more than 2 keyboard switches are depressed at the same time, all outputs 1~91 of the address ROM go high-level, and the 3640-bit ROM is not addressed. The internal key input signal also is not applied to the timing circuit; as a result, a strobe signal is not generated. Also, the coded outputs hold the preceding state. Then, if any one key (key 1) is not released while the other keys are, key 1 becomes valid.

2. N-Key Rollover

If the key input signals are pulsive, the primary depressed key (key 1) is read; after the coded output of key 1 is transferred to the buffer register, a strobe signal is generated and the coded output becomes valid. Then, if a second key is depressed while key 1 is in the depressed state, the second key (key 2) is read; and the coded output of key 2 is transferred to the buffer register succeeding the coded output of key 1 described above. A strobe signal is generated, and the coded output becomes valid. Then if a third, fourth . . . Nth key is depressed while preceding keys are still in the depressed state, its code will become valid as described above.

3. Any-Key-Down Output

When any one or more of the 91 keys are depressed, an internal any-key signal is transferred from the address ROM to the timing circuit where an any-key-down signal (AKD) is generated.

4. Strobe Inhibit When an Unused Code Is Addressed

If either an unused mode of the 4 modes or an unused key is selected (its ROM code is 0000000000), the strobe output is inhibited and it makes the key invalid. The data output still holds the preceding state.

5. Repeat Function

When a repeat signal is applied to the repeat control input (RC), a strobe signal is repeatedly generated so that any character can be repeated. The strobe signal is inhibited when the RC terminal is high.

6. Data Acknowledge Input

The strobe output is reset by applying a data acknowledge input. The pulse width of the strobe signal output can be adjusted with a resistor and a capacitor connected between the strobe output terminal (STO) and the data acknowledge input terminal (DAK).

7. Data Invert and Parity Invert Inputs

The level of each output B_1~B_9 and B_{10} can be inverted when data invert input (DI) and parity invert input (PI) are high-level.

8. Chip Enable Input

Data outputs B_1~B_{10}, strobe output and any-key-down output are in the floating state when chip enable input (CE) is high.

This floating state means a high-impedance state and is equivalent to an open-circuit output.

9. Input Control Input

When input control input (IC) is high, key inputs (K_1~K_{14}) can be operated with high-level signals.

10. Strobe Control Input

The strobe delay time can be set by the strobe control input STC terminal. The delay time is set to $t_{d(ST-B)}$, which depends on the internal delay circuit when the strobe control input terminal is connected to V_{SS}.

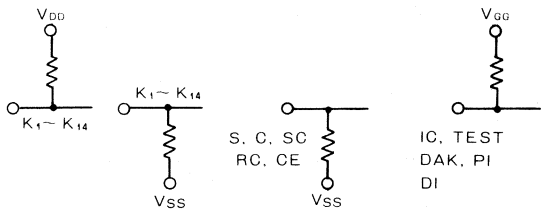
11. Test Input

Data outputs (B_1~B_{10}) can be independently set either high or low irrespective of the 3640-bit ROM outputs. When test input (TEST) is high, B_1~B_{10} goes high if both DI and PI are low, and B_1~B_{10} goes low if both DI and PI are high.

12. Pull-up Resistors

External resistors are not required because pull-up resistors are built-in at all input terminals. But if the strobe control input terminal is not used, it should be connected to V_{SS}. To determine the value of the resistor required, see Electrical Characteristics.

Pull-up resistors



(a) IC = "H" (b) IC = "L" K_1~K_{14} INPUT

S, C, SC, RC, CE INPUT

IC, TEST DAK, PI, DI INPUT

Table 1 Data-output level in relation to data invert (DI), parity invert (PI) and chip enable (CE)

ROM CODE	DI, PI	CE	B_1~B_{10}
1	H	L	L
	L	L	H
0	H	L	H
	L	L	L
1	H	H	Z
	L	H	Z
0	H	H	Z
	L	H	Z

Note 2 : Z indicates a floating state.

3 : The code table is described in positive logic, for outputs B_1 ~B_{10}, when DI and PI are low.

Table 2 Function table of the mode select circuit

S	C	SC	MODE
H	H	H	—
L	H	H	—
H	L	H	—
L	L	H	M_4
H	H	L	M_4
L	H	L	M_3
H	L	L	M_2
L	L	L	M_1

MITSUBISHI ELECTRIC

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{GG}	Supply voltage	With respect to V_{SS}	0.3～−20	V
V_{DD}	Supply voltage		0.3～−20	V
V_I	Input voltage		0.3～−20	V
Pd	Power dissipation	Ta＝25°C	1.0	W
Topr	Operating free-air temperature range		−20～75	°C
Tstg	Storage temperature range		−40～125	°C

RECOMMENDED OPERATING CONDITIONS (Ta＝−20～75°C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{GG}	Supply voltage	−10.8	−12	−13.2	V
V_{DD}	Supply voltage		0		V
V_{SS}	Supply voltage	4.5	5	5.5	V
V_{IH}	High-level input voltage, all inputs except STC	V_{SS}−1.5		V_{SS}	V
V_{IL}	Low-level input voltage	V_{DD}		V_{SS}−3.5	V
t_r	Rise time (10～90%), all inputs except DAK			1	μs
t_f	Fall time (10～90%)			1	μs
$t_{r(DAK)}$	Rise time (10～90%), DAK			100	μs
$t_{f(DAK)}$	Fall time (10～90%), DAK			100	μs

ELECTRICAL CHARACTERISTICS (Ta＝−20～75°C, V_{GG}＝−12V±10%, V_{DD}＝0V, V_{SS}＝5V±10%, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage	I_{OH}＝−100μA	V_{SS}−1			V
V_{OL}	Low-level input voltage	I_{OL}＝1.6mA, (Note 2)			0.4	V
$I_{I(1)}$	Input current, TEST, IC, DI, PI, and DAK	V_I＝V_{GG}		−0.01	−10	μA
$I_{I(2)}$	Input current, K_1～K_{14}	V_I＝V_{DD}, $V_{I(IC)}$＝V_{IH}		−0.02	−20	μA
$R_{I(1)}$	Input resistance, IC, PI, DI, DAK, and TEST	V_I＝V_{SS}, Ta＝25°C	100	180	300	kΩ
$R_{I(2)}$	Input resistance, S, C, SC, CE, and RC	V_I＝V_{DD}, Ta＝25°C	5		30	kΩ
$R_{I(3)}$	Input resistance, K_1～K_{14}	V_I＝V_{SS}, $V_{I(IC)}$＝V_{IH}, Ta＝25°C	10	20	40	kΩ
$R_{I(4)}$	Input resistance, K_1～K_{14}	V_I＝V_{DD}, $V_{I(IC)}$＝V_{IL}, Ta＝25°C	2	5	15	kΩ
Pd	Power dissipation	Ta＝25°C		350	500	mW
C_I	Input capacitance	All terminals except the tested terminal are 0V. V_I＝0V, Vrms＝25mV, f＝1MHz			15	pF

Note 4 : Current flowing into an IC is positive; out is negative.

5 : When all outputs are at I_{OL}＝1.6mA, $V_{OL}max$＝0.6V

9

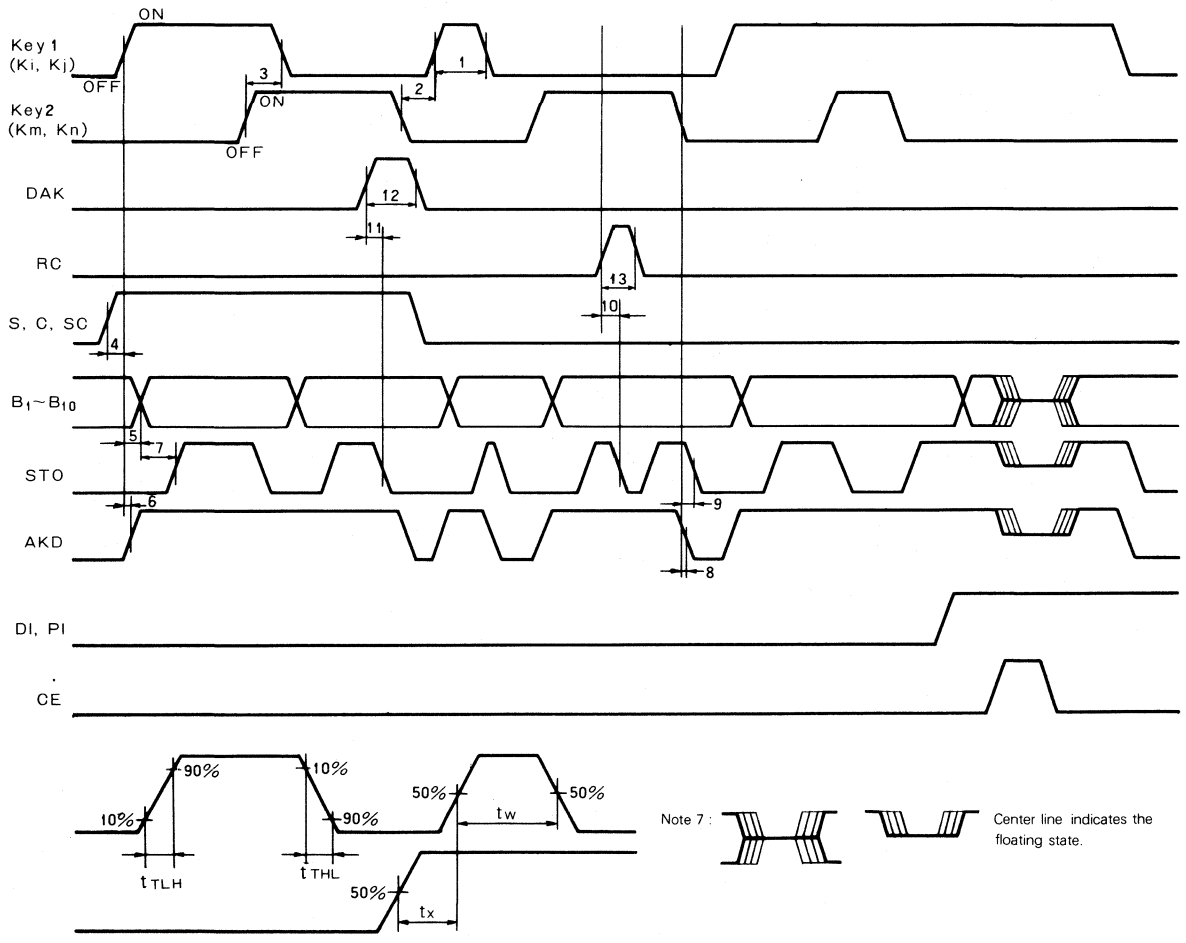
KEYBOARD ENCODER (JIS CODE STANDARD PRODUCT)

SWITCHING CHARACTERISTICS (T_a=25°C, V_{GG}=−12V±10%, V_{DD}=0V, V_{SS}=5V±10%, unless otherwise noted)

Symbol	Parameter	Test conditions (Note 6)	Limits			Unit
			Min	Typ	Max	
t_{TLH}	Low-to-high-level output transition time	C_L=50pF, I_{OH}=−0.1mA		0.7	2	μs
t_{THL}	High-to-low-level output transition time	C_L=50pF, I_{OL}=1.6mA		0.5	1.5	μs
$t_{w(KI)}$	Key input pulse width	*1, tw	30			μs
$t_{d(KILH-K2HL)}$	Delay time from key 1 low-to-high-level	*2, tx	10			μs
$t_{h(K_1-K_2)}$	Key 1 hold time with respect to key 2	*3, tx	10			μs
$t_{su(M-KON)}$	S, C, SC setup time with respect to key input (ON)	*4, tx			1.5	μs
$t_{D(B-KON)}$	Delay time from key input (ON) to B_1~B_{10}	*5, tx	2	7	15	μs
$t_{D(AK-KON)}$	Delay time from key input (ON) to AKD	*6, tx		0.5	2	μs
$t_{d(ST-B)}$	Delay time from B_1~B_{10} to STO	*7, tx, C_L=50pF, STC-Vss shorted	1	5	12	μs
$t_{D(AK-KOF)}$	Delay time from key input (OFF) to AKD	*8, tx, C_L=50pF		0.5	2	μs
$t_{D(ST-KOF)}$	Delay time from key input (OFF) to STO	*9, tx, C_L=50pF		4	10	μs
$t_{(ST-RC)}$	Delay time from RC to STO	*10, tx, C_L=50pF		3.5	20	μs
$t_{D(ST-DAK)}$	Delay time from DAK to STO	*11, tx, C_L=50pF		4	10	μs
$t_{w(DAK)}$	DAK pulse width	*12, tw	10			μs
$t_{w(RC)}$	RC pulse width	*13, tw	15			μs
$t_{w(STO)}$	STO pulse width	tw, C_L=50pF, STO-DAK shorted	1	4	10	μs

Note 6 : See the Timing Diagram for 'tw' and 'tx'. Numbers 1 through 13 in the diagram correspond to *1 through *13 above.

TIMING DIAGRAM

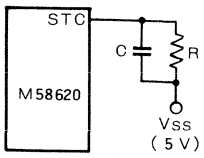


Note 7 : Center line indicates the floating state.

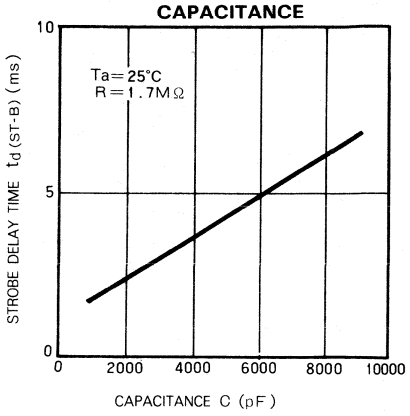
MITSUBISHI ELECTRIC

TYPICAL CHARACTERISTICS ($V_{GG}=-12V$, $V_{DD}=0V$, $V_{SS}=5V$)

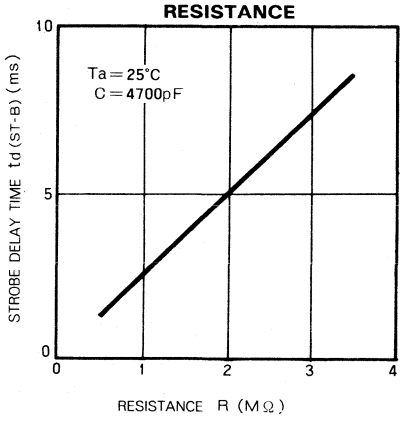
MEASUREMENT CIRCUIT



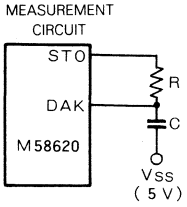
STROBE DELAY TIME VS. CAPACITANCE



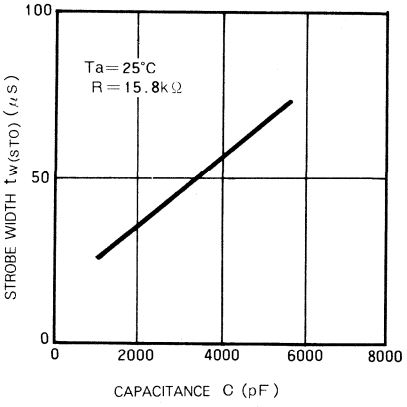
STROBE DELAY TIME VS. RESISTANCE



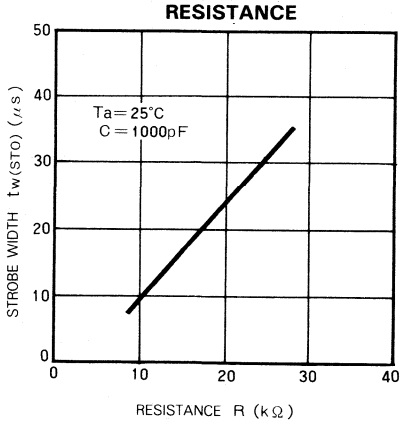
MEASUREMENT CIRCUIT



STROBE WIDTH VS. CAPACITANCE



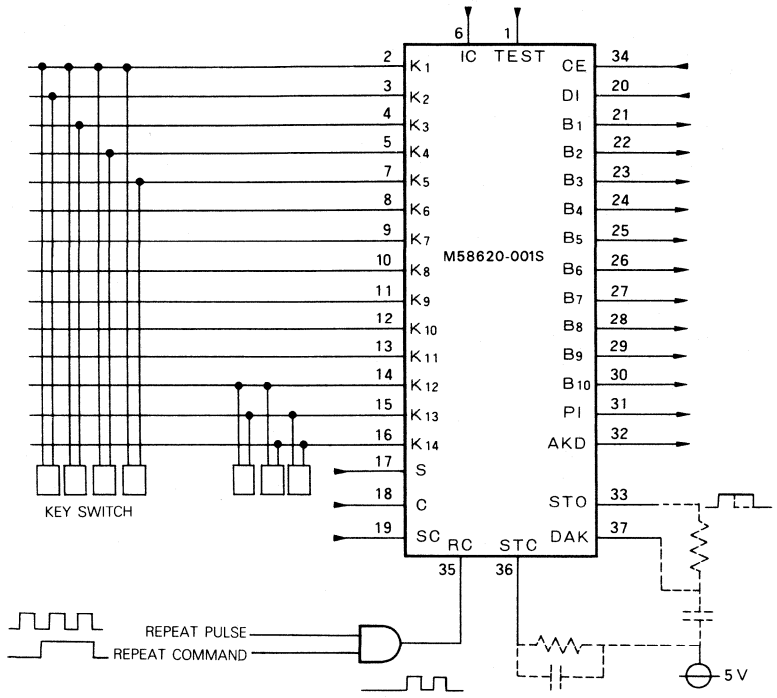
STROBE WIDTH VS. RESISTANCE



9

KEYBOARD ENCODER (JIS CODE STANDARD PRODUCT)

TYPICAL APPLICATION CIRCUIT



Note 8 : Use a key switch having outputs that are open-collector, or mutually separated by diodes.

MITSUBISHI ELECTRIC

DESCRIPTION

The M58741P is a color TV interface device designed for use with an 8-bit parallel M5L8080AP CPU as a graphic display controller using an NTSC home color television set. This device is fabricated using N-channel silicon gate ED-MOS technology for a single supply voltage.

FEATURES

- Graphic display interface for NTSC color TV sets
- 64 by 64 color elements
- Every element can be displayed in 8 colors (including black and white)
- Every element can be blinked by external hardware
- Compatible with M5L8080AP, S and M5L8085AP, S CPUs
- Single 5V power supply

APPLICATION

- Home computers, TV games

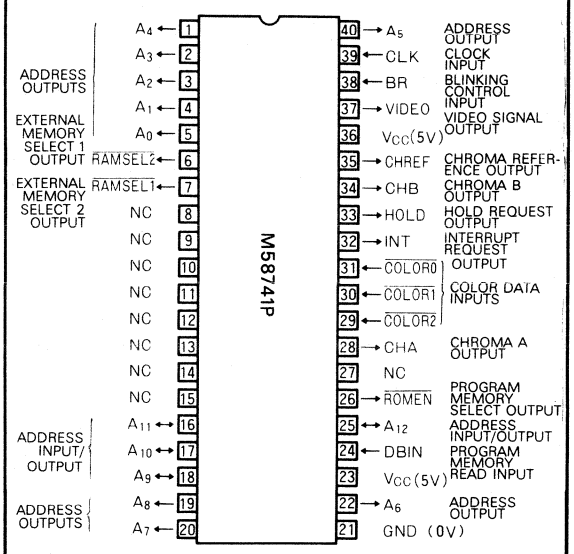
FUNCTION

The M58741P and a MELPS 8 or similar 8-bit microcomputer can display the 64 by 64 color elements on the CRT screen of the television set.

The M58741P reads out the color data from RAMs and it is very easy to change or move the image on the screen. It is effective to reduce the software programming time.

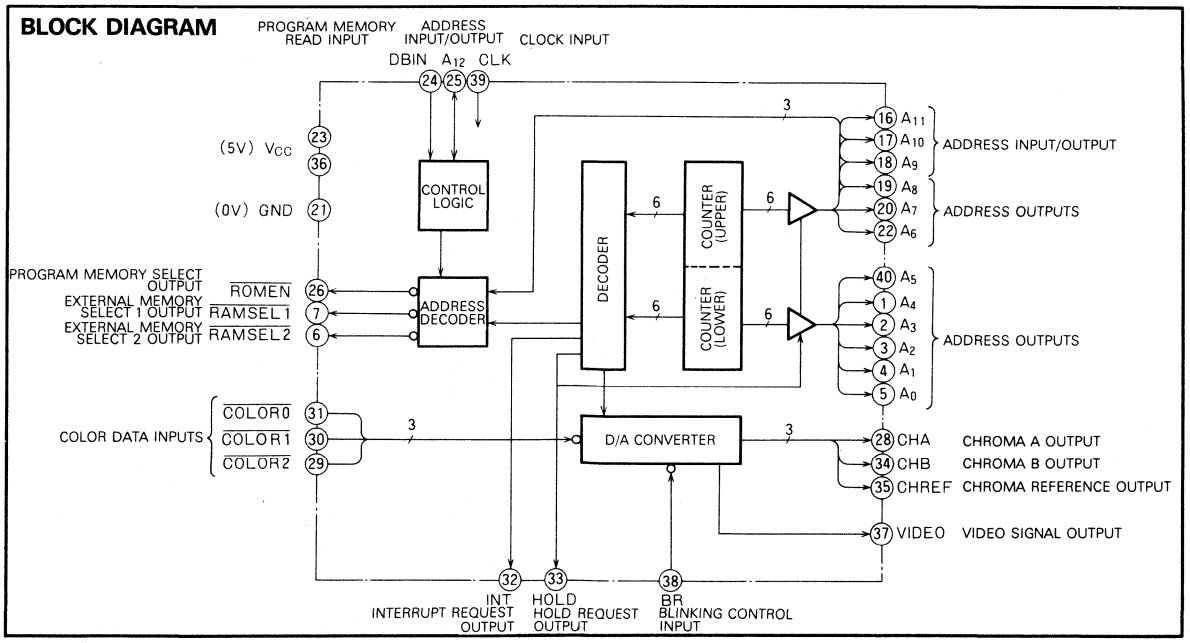
The smallest color graphic display system using the M58741P, requires only 10 chips including the M51342P modulator IC, CPU, ROM and RAMs.

PIN CONFIGURATION (TOP VIEW)



Outline 40P1 NC : MUST BE OPEN

BLOCK DIAGRAM



TV INTERFACE

PIN DESCRIPTION

Symbol	Name	Input or output	Function
$A_0 \sim A_5$	Address output	Out	When hold request output is high, a row location of the refresh memory is designated.
$A_6 \sim A_8$	Address output	Out	When hold request output is high, a column location of the refresh memory is designated.
$A_9 \sim A_{11}$	Address input/output	In/out	When hold request output is high, a column location of the refresh memory is designated. When hold request output is low, $\overline{\text{RAMSEL 1}}$, $\overline{\text{RAMSEL 2}}$, $\overline{\text{ROMEN}}$ become active according to the data from these inputs. A_{12} | A_{11} | A_{10} | A_9 | PIN 0 | 0 | X | X | $\overline{\text{ROMEN}}$ (Note 1) 0 | 1 | 0 | 0 | $\overline{\text{RAMSEL 1}}$ 0 | 1 | 0 | 1 | $\overline{\text{RAMSEL 2}}$ 0 | 1 | 1 | X | Inhibit Note 1 : $\overline{\text{ROMEN}} = \overline{\text{DBIN} \cdot \overline{A_{11}} \cdot \overline{A_{12}}}$
A_{12}	Address input	In	When high-level is applied to this pin, internal address decoder goes inactive.
$\overline{\text{RAMSEL 1}}$	External memory select 1	Out	When $\overline{A_{12}} \cdot A_{11} \cdot \overline{A_{10}} \cdot \overline{A_9}$ is high, $\overline{\text{RAMSEL 1}}$ goes low.
$\overline{\text{RAMSEL 2}}$	External memory select 2	Out	When $\overline{A_{12}} \cdot A_{11} \cdot \overline{A_{10}} \cdot A_9$ is high, $\overline{\text{RAMSEL 2}}$ goes low.
$\overline{\text{ROMEN}}$	Program memory select	Out	When DBIN $\cdot A_{11} \cdot A_{12}$ is high, $\overline{\text{ROMEN}}$ goes low.
HOLD	Hold request output	Out	High-level indicates the M58741P occupies the address bus lines.
INT	Interrupt request output	Out	After one flame is swept, this output keeps high about 65μs (both HOLD and INT are high in 30μs).
DBIN	Program memory read input	In	When HOLD is low, high-level input makes $\overline{\text{ROMEN}}$ active.
$\overline{\text{COLOR 0}}$ \~ $\overline{\text{COLOR 2}}$	Color data inputs	In	Color data inputs from the refresh memory designates the color of displayed element. $\overline{\text{COLOR 2}}$ | $\overline{\text{COLOR 1}}$ | $\overline{\text{COLOR 0}}$ | COLOR 0 | 0 | 0 | magenta 0 | 0 | 1 | red 0 | 1 | 0 | green 0 | 1 | 1 | white 1 | 0 | 0 | orange 1 | 0 | 1 | syan 1 | 1 | 0 | blue syan 1 | 1 | 1 | black
BR	Blinking control input	In	High-level inhibits display, turning the CRT screen black. Thus high-low switching results in blinking of the screen display. This input is sampled one clock cycle earlier than color data inputs. Blinking frequency can be selected.
VIDEO	Video signal output	Out	A 4-level baseband video signal including the sync signal can be derived. A color subcarrier and a color signal are not superimposed.
CH A CH B CH REF	Color signal outputs	Out	Two-3-level outputs and its reference bias output are generated for the M51342P : NTSC TV game modulator.
CLK	Clock input	In	1.79MHz, 25% duty clock input signal is required.

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℃	600	mW
T_{opr}	Operating free-air ambient temperature range		0～70	℃
T_{stg}	Storage temperature range		−40～125	℃

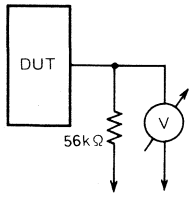
RECOMMENDED OPERATING CONDITIONS (T_a＝0～70℃, unless otherwise noted)

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
$t_{C(CLK)}$	Clock cycle time		558.7		ns

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

Symbol	Parameter	Conditions	Limits			Unit
			Min	Typ	Max	
I_{CC}	Supply current from V_{CC}				100	mA
V_{OH}	High-level output voltage	I_{OH}＝−100μA	2.4			V
V_{OL}	Low-level output voltage	I_{OL}＝1mA			0.4	V
V_{SYNC}	Video sync level voltage	(Note 2)		0.8•V_{CC}		V
V_{PED}	Video pedestal level voltage			0.7•V_{CC}		V
V_{DARK}	Video dark level voltage			0.6•V_{CC}		V
V_{BRIGHT}	Video bright level voltage			0.5•V_{CC}		V
V_{CH0}	Color signal 0 level voltage			0.6•V_{CC}		V
V_{CH1}	Color signal 1 level voltage			0.68•V_{CC}		V
V_{CH2}	Color signal 2 level voltage			0.76•V_{CC}		V
V_{CHREF}	Color reference signal level voltage			0.68•V_{CC}		V

Note 2 : Measurement circuit 1.



9

TV INTERFACE

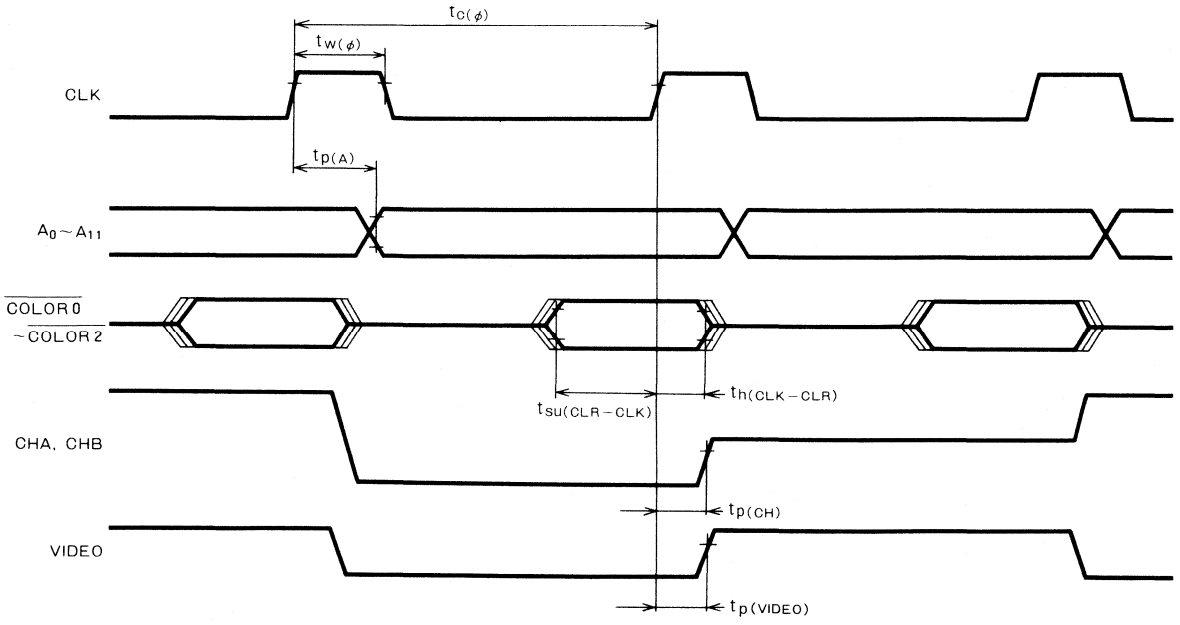
TIMING REQUIREMENTS (T_a=0~70℃, V_{CC}=5V±5%, unless otherwise noted)

Symbol	Parameter	Conditions	Limits			Unit
			Min	Typ	Max	
$t_{C(\phi)}$	Clock cycle time			558.7		ns
$t_{W(\phi)}$	Clock pulse width		120		300	ns
$t_{su(CLR-CLK)}$	Color signal setup time before clock		50			ns
$t_{h(CLK-CLR)}$	Color signal hold time after clock		100			ns
$t_{su(BRIG-CLK)}$	Blinking control setup time before clock		50			ns
$t_{h(CLK-BRIG)}$	Blinking control hold time after clock		100			ns

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

Symbol	Parameter	Conditions	Limits			Unit
			Min	Typ	Max	
$t_{p(A)}$	Propagation time from clock to address	C_L=50pF, Load 1LSTTL			200	ns
$t_{p(CH)}$	Propagation time from clock to color	C_L=10pF, R_L=50kΩ			150	ns
$t_{p(VIDEO)}$	Propagation time from clock to video	C_L=10pF, R_L=50kΩ			150	ns
$t_{p(DBIN-ROM)}$	Propagation time from DBIN to program memory select	C_L=50pF, Load 1LSTTL			250	ns
$t_{p(A-SEL)}$	Propagation time from address to external memory select	C_L=50pF, Load 1LSTTL			200	ns

TIMING DIAGRAM



M5L 8041A-XXXP

UNIVERSAL PERIPHERAL INTERFACE

DESCRIPTION

The M5L 8041A-XXXP is a general-purpose, programmable interface device designed 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

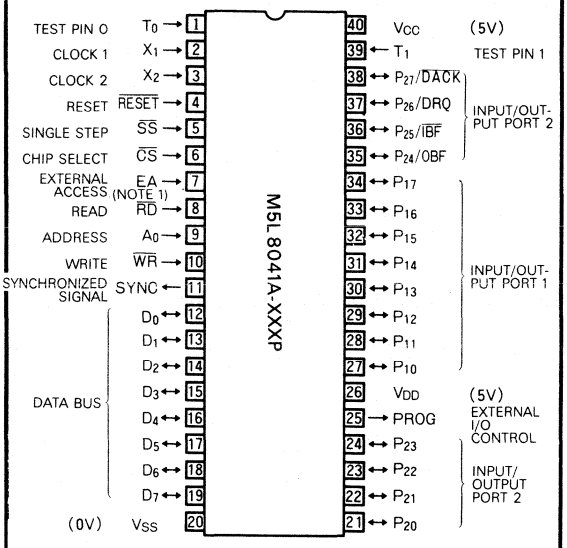
- Alternative to custom LSI for peripheral interface

FUNCTION

The M5L 8041A-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 M5L 8041A-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 M5L 8041A-XXXP is a complete microcomputer, it is easy to develop a user-oriented mask-programmed peripheral LSI only by changing control software.

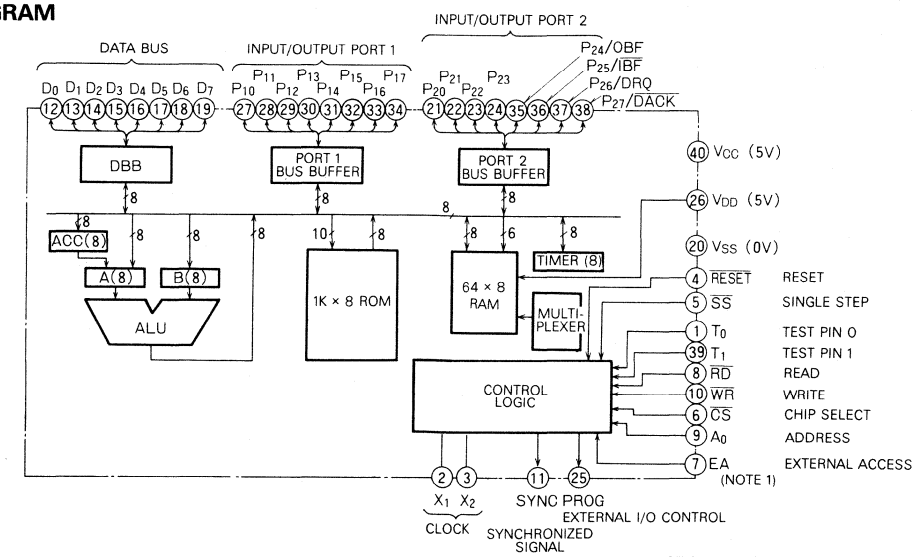
PIN CONFIGURATION (TOP VIEW)



Outline 40P

Note 1: Connect to VSS in the operating condition.

BLOCK DIAGRAM



UNIVERSAL PERIPHERAL INTERFACE

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	Power dissipation	Ta＝25℃	1000	mW
T_{opr}	Operating free-air temperature range		−20～70	℃
T_{stg}	Storage temperature range		−40～125	℃

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 (T_a＝−20～70℃, 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		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}				20	mA
$I_{CC}+I_{DD}$	Total supply current				135	mA

Note 2 : 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

TIMING REQUIREMENTS ($Ta = -20 \sim 70℃$, $V_{CC} = 5\ V \pm 10\%$, 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 width	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
$t_{rec(RW)}$	Recovery time between read and/or write	t_{RV}		300			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}		1400			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}		140			ns
$t_{su(Q-PR)}$	Output data setup time before PROG	t_{DP}		220			ns
$t_{su(D-PR)}$	Input data hold time before PROG	t_{PS}		700			ns
$t_{h(PR-D)}$	Input data hold time after PROG	t_{PF}		110			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 ($Ta = -20 \sim 70℃$, $V_{CC} = 5\ V \pm 10\%$, 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_{RDF}				100	ns

DMA

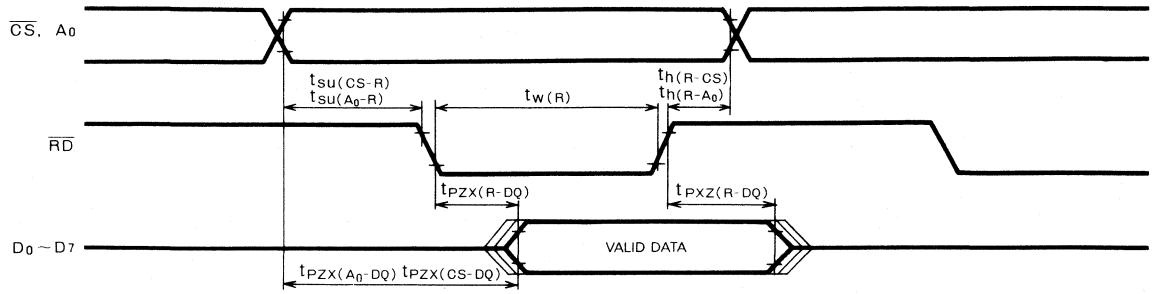
Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{PZX(DACK-DQ)}$	Data enable time after DACK	t_{ACD}	100pF Load			225	ns
$t_{PHL(R-DRQ)}$	DRQ disable time after read	t_{CRQ}	100pF Load			200	ns
$t_{PHL(W-DRQ)}$	DRQ disable time after write	t_{CRQ}	100pF Load			200	ns

9

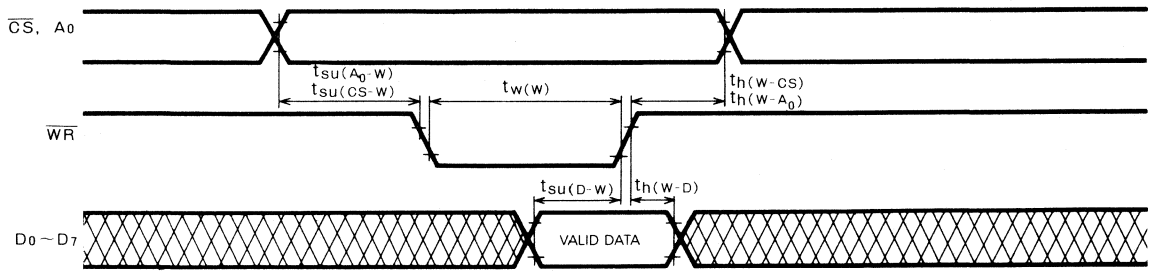
UNIVERSAL PERIPHERAL INTERFACE

TIMING DIAGRAMS

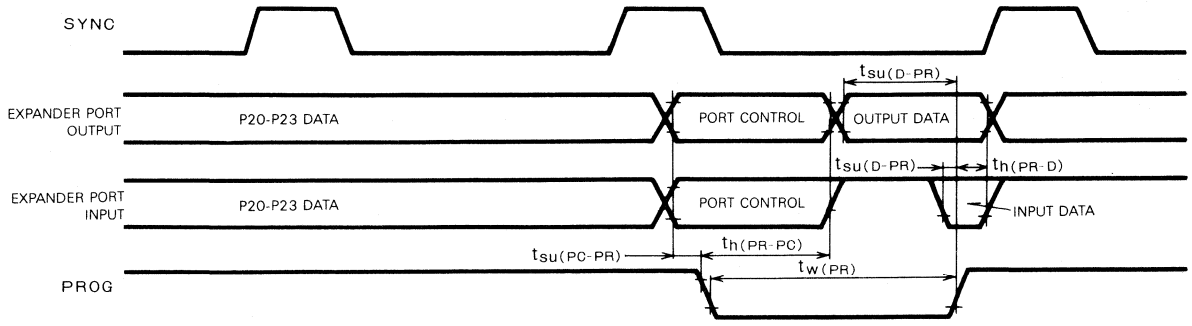
Read



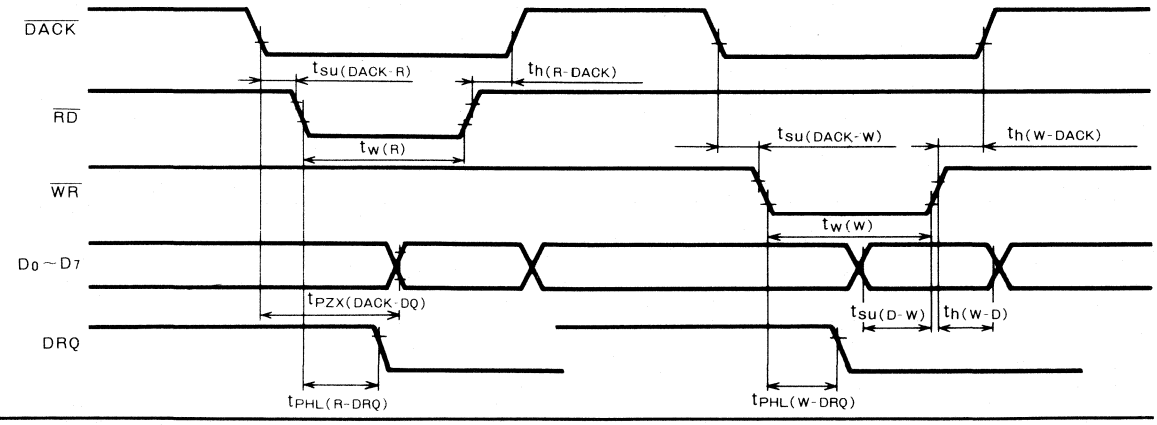
Write



Port 2



DMA



MITSUBISHI ELECTRIC

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.

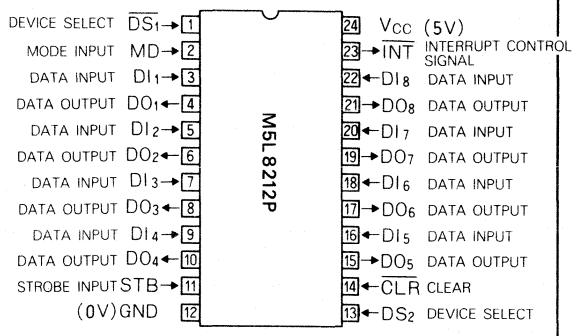
FEATURES

- 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

APPLICATION

- Input/output port for a M5L8080AP, S
- Latches, gate buffers or multiplexers
- Peripheral and input/output functions for microcomputer systems

PIN CONFIGURATION (TOP VIEW)



Outline 24P1

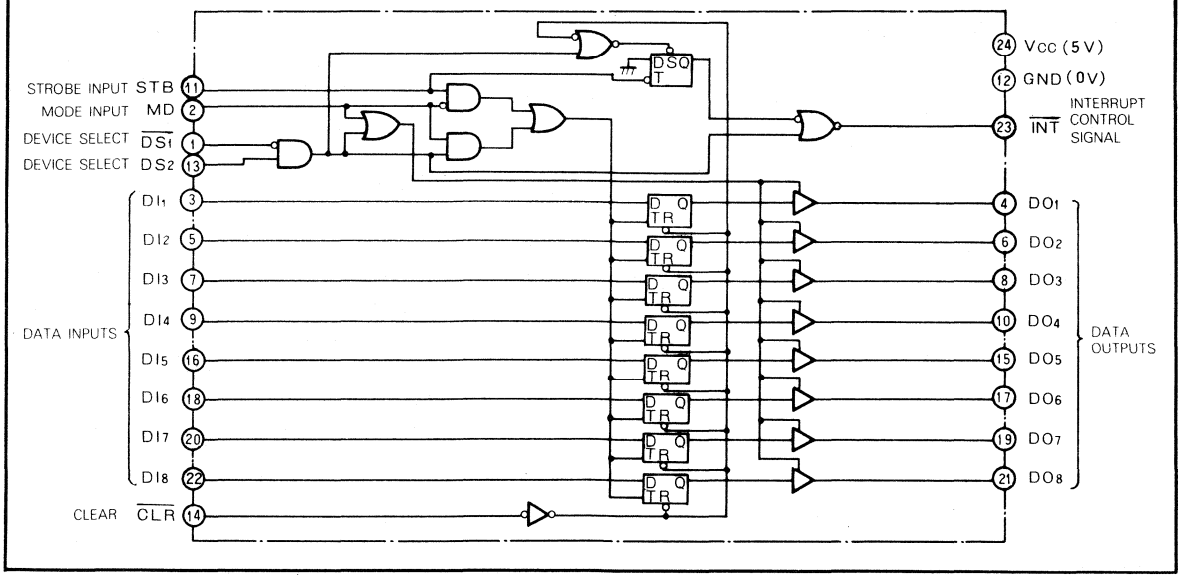
FUNCTION

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~DO_8; and the service request flip-flop SR is set. Also, the strobed input STB is active, the data inputs DI_1~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.

9

BLOCK DIAGRAM



M5L 8212P

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, $\overline{DS1}$, MD inputs		V_{CC}	V
V_I	Input voltage, all other inputs except $\overline{DS1}$, 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_O = 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 ~ 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 ~ 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)

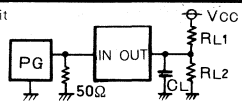
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{w(DS2)}$	Input pulse width, $\overline{DS1}$, DS2 and STB		30			ns
$t_{w(CLR)}$	Input pulse width $\overline{CLR}$		45			ns
$t_{su(DA)}$	Data setup time with respect to $\overline{DS1}$, DS2 and STB		15			ns
$t_{h(DA)}$	Data hold time with respect to $\overline{DS1}$, DS2 and STB		20			ns

8-BIT INPUT/OUTPUT PORT WITH 3-STATE OUTPUT

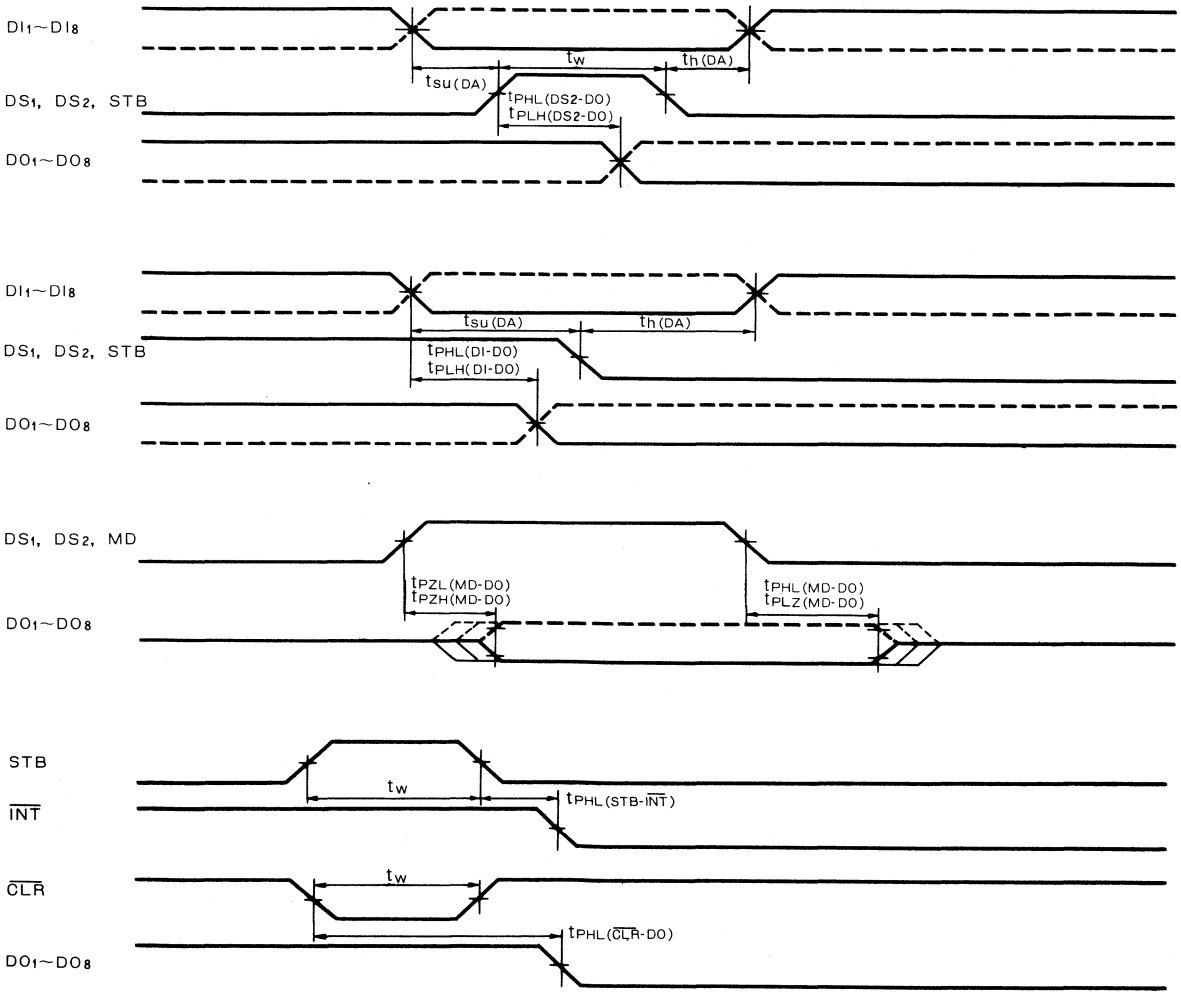
SWITCHING CHARACTERISTICS (Ta = 25°C, Vcc = 5V, unless otherwise noted)

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 inputs $\overline{DS1}$, 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

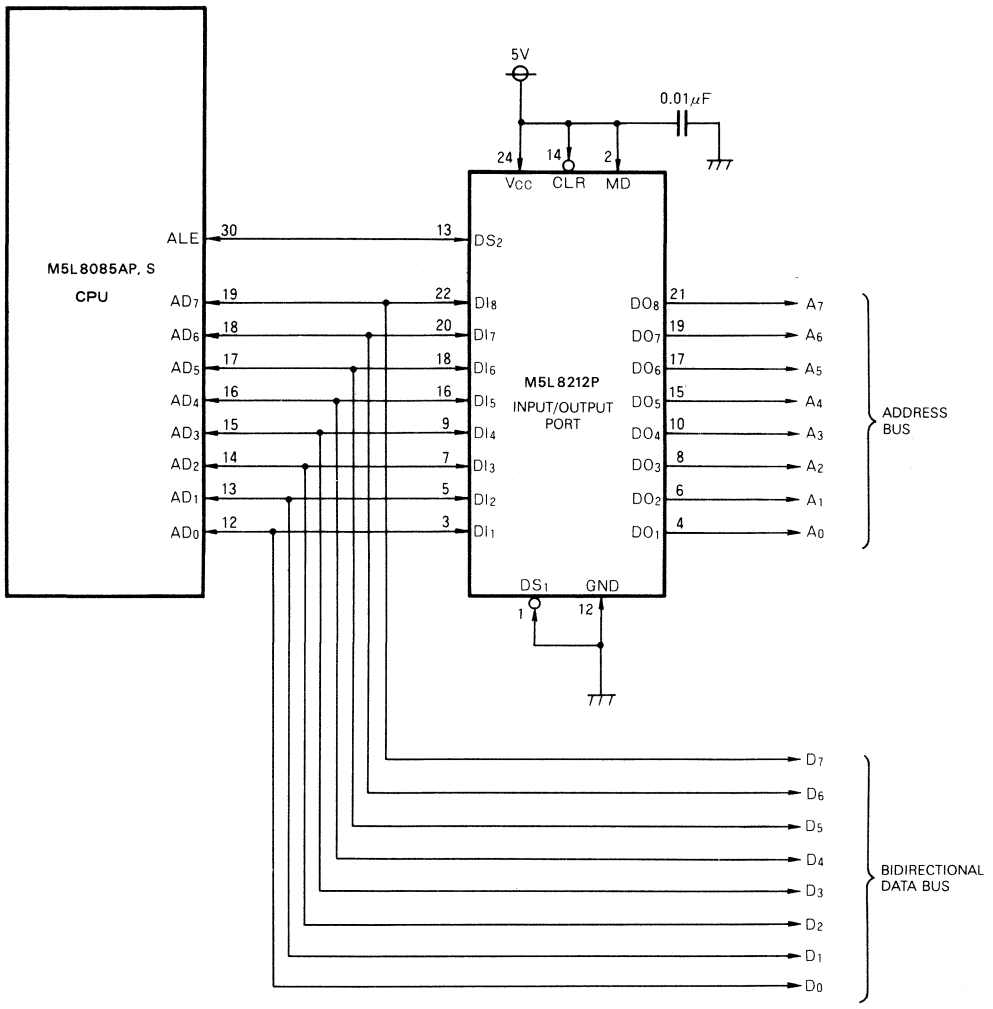


9

M5L 8212P

8-BIT INPUT/OUTPUT PORT WITH 3-STATE OUTPUT

APPLICATION CIRCUIT



M5L 8216P, M5L 8226P

4-BIT PARALLEL BIDIRECTIONAL BUS DRIVERS

DESCRIPTION

The M5L 8216P and M5L 8226P are 4-bit bidirectional bus drivers and suitable for the 8-bit parallel CPU M5L 8080AP, 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 M5L 8216P
  DB: $I_{OL} = 55mA$(max)
  $I_{OH} = -10mA$(max)
  DO: $I_{OH} = -1mA$(max)
  M5L 8226P
  DB: $I_{OL} = 50mA$(max)
  $I_{OH} = -10mA$(max)
  DO: $I_{OH} = -1mA$(max)
- Outputs can be connected with the CPU M5L 8080AP, S: $V_{OH} = 3.65V$(min)
- Three-state output
- The M5L 8216P has interchangeability with Intel's 8216 in pin configuration and electrical characteristics, and the M5L 8226P with Intel's 8226.

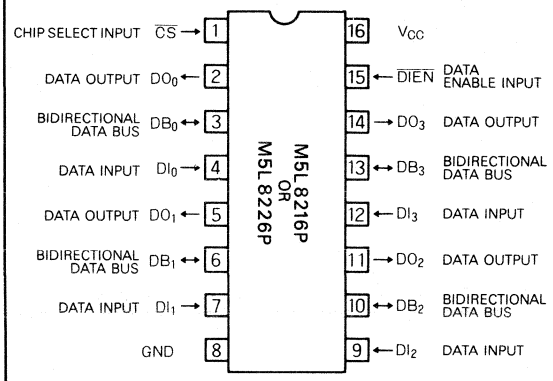
APPLICATION

Bidirectional bus driver/receiver for various types of microcomputer systems.

FUNCTION

The M5L 8216P is a noninverting and the M5L 8226P is an inverting 4-bit bidirectional bus driver.

PIN CONFIGURATION (TOP VIEW)



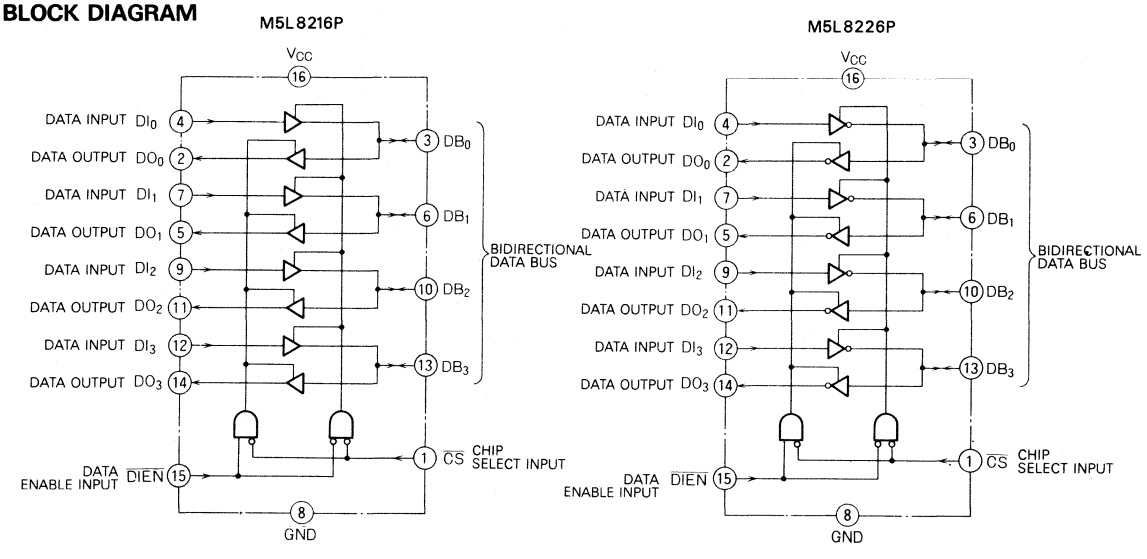
Outline 16P4

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.

9

BLOCK DIAGRAM



M5L 8216P, M5L 8226P

4-BIT PARALLEL BIDIRECTIONAL BUS DRIVERS

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		−55～125	℃

RECOMMENDED OPERATING CONDITIONS (T_a=0～75℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
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)

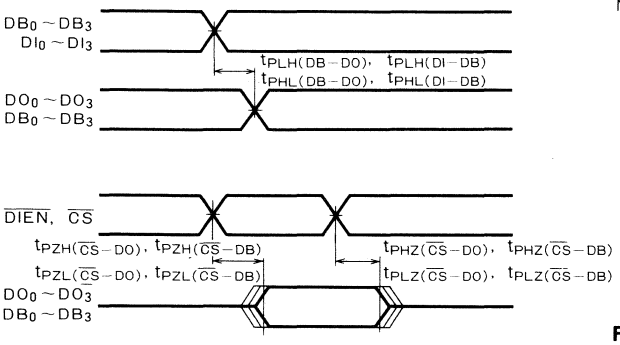
Symbol	Parameter		Conditions		Limits			Unit
					Min	Typ	Max	
V_{IH}	High-level input voltage				2			V
V_{IL}	Low-level input voltage						0.95	V
V_{IC}	Input clamp voltage		V_{CC}=4.75V, I_{IC}=−5mA				−1	V
V_{OH}	High-level output voltage, DO output		V_{CC}=4.75V, V_{IH}=2V, V_{IL}=0.95V	I_{OH}=−1mA	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, $\overline{DB}$ output (Note 2)				−30		−120	mA
I_{CC}	Supply current		V_{CC}=5.25V				130	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.

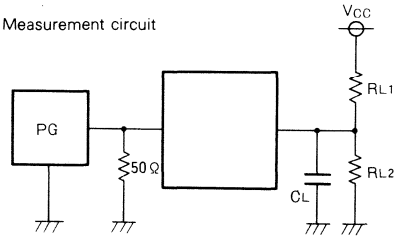
SWITCHING CHARACTERISTICS (V_{CC}=5V, Ta=25℃, unless otherwise noted)

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		C_L=300pF, R_{L1}=90Ω, R_{L2}=180Ω			30	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			35	ns
$t_{PZH(\overline{CS}-DO)}$	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			35	ns
$t_{PZH(\overline{CS}-DB)}$	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)



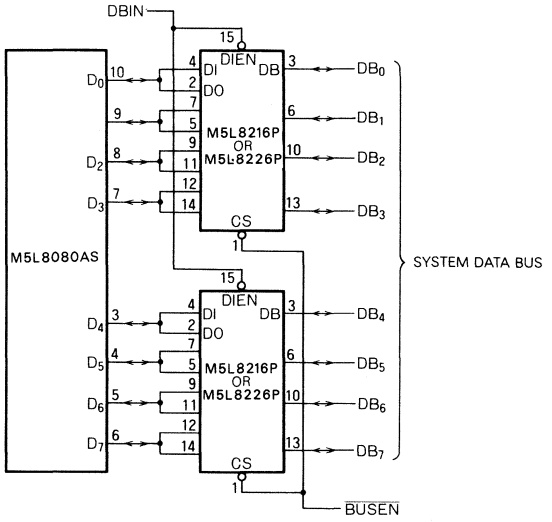
Note 3 : Measurement circuit



TYPICAL APPLICATIONS

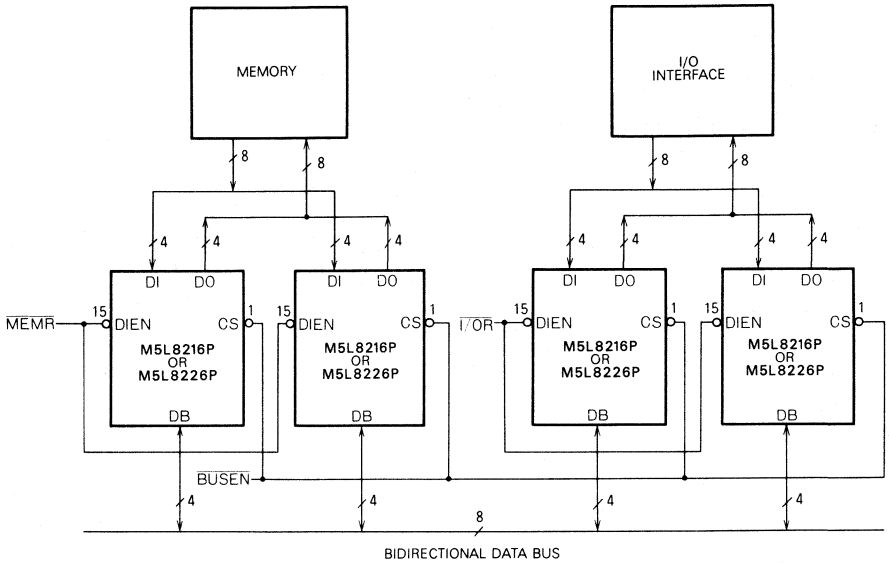
Fig. 1 shows a pair of M5L 8216PS or M5L 8226PS which are directly connected with the M5L 8080A CPU data bus, and their control signal. Fig. 2 shows an example circuit in which the M5L 8216P or M5L 8226P is used as an interface for memory and I/O to a bidirectional bus.

Fig. 1 Data bus buffer



4-BIT PARALLEL BIDIRECTIONAL BUS DRIVERS

Fig. 2 Memory and I/O interface to bidirectional data bus



INPUT/OUTPUT EXPANDER

DESCRIPTION

The M5L 8243P 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 M5L 8041A-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

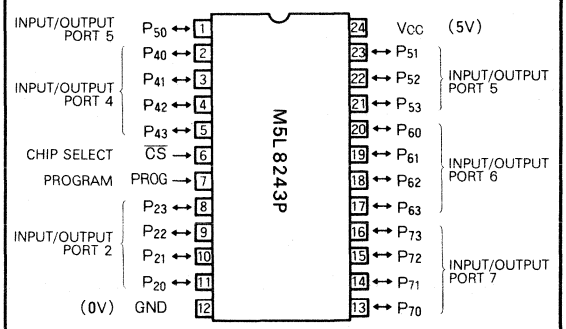
- I/O expansion for the MELPS 8-48 single-chip microcomputers.

FUNCTION

The M5L 8243P is designed to provide a low-cost means of I/O expansion for the M5L 8041A-XXXP universal peripheral interface and the M5L 8048 and M5L 8049 single-chip microcomputers. The M5L 8243P consists of four 4-bit bidirectional static I/O ports and one 4-bit port which serves as an interface to the M5L 8041A-XXXP and M5L 8048/9. Thus multiple M5L 8243Ps can be added to a single master.

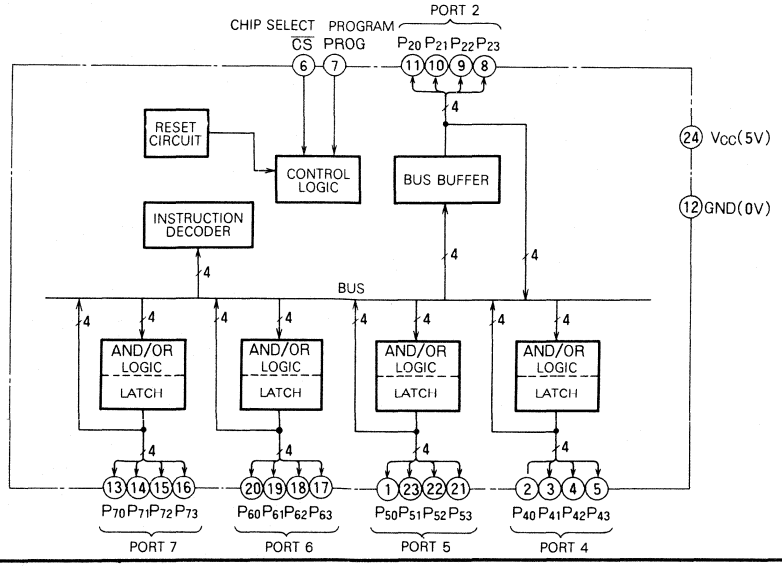
Using the original instruction set of the master, the M5L 8243P serves as the in resident I/O facility. Its I/O ports are accessed by instructions MOV, ANL and ORL.

PIN CONFIGURATION (TOP VIEW)



Outline 24P1

BLOCK DIAGRAM



9

INPUT/OUTPUT EXPANDER

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
P_d	Maximum power dissipation	Ta＝25℃	600	mW
T_{opr}	Operating free-air temperature range		−20～70	℃
T_{stg}	Storage temperature range		−40～125	℃

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			0.8	V

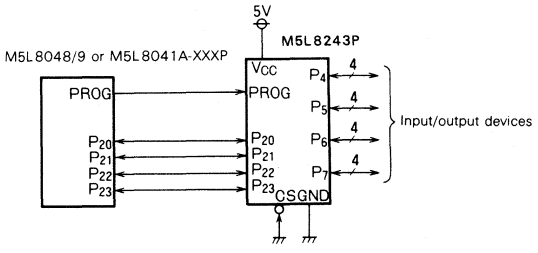
ELECTRICAL CHARACTERISTICS (T_a＝−20～70℃, V_{CC}＝5V±10%, unless otherwise noted)

Symbol	Parameter	Conditions	Limits			Unit
			Min	Typ	Max	
V_{IL}	Low-level input voltage		−0.5		0.8	V
V_{IH}	High-level input voltage		2		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				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}＝200μA	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≦in≦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 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



9

INPUT/OUTPUT EXPANDER

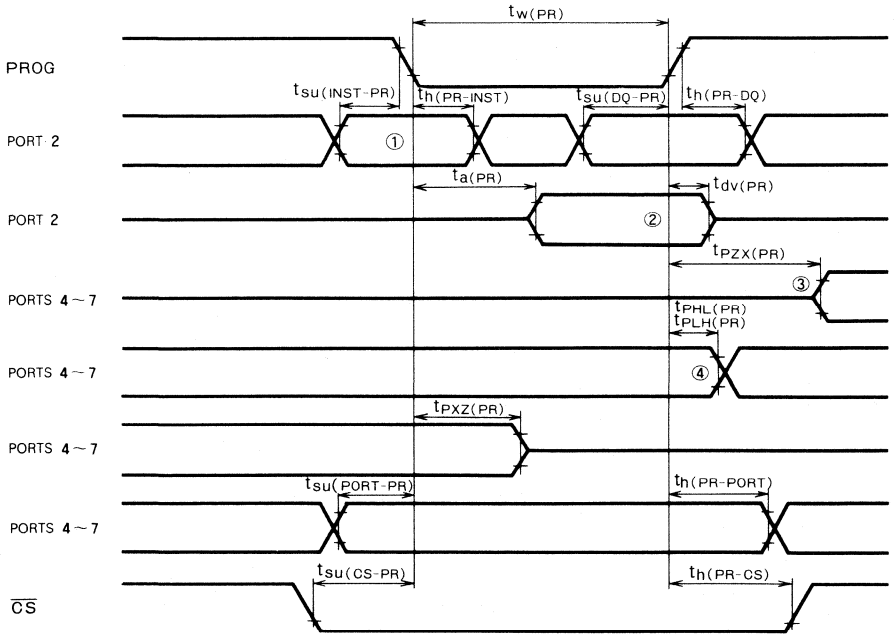
TIMING REQUIREMENTS (T_a = −20~70℃, V_{CC} = 5V ± 10%, V_{SS} = 0V, unless otherwise noted)

Symbol	Parameter	Alternative symbol	Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{su(INST-PR)}$	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			Unit
				Min	typ	Max	
$t_{a(PR)}$	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

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

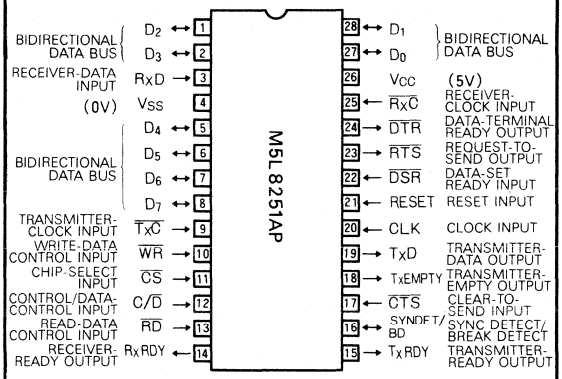
APPLICATIONS

- Modem control of data communications using microcomputers
- Control of CRT, TTY and other terminal equipment

FUNCTION

The M5L 8251AP 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 28P1

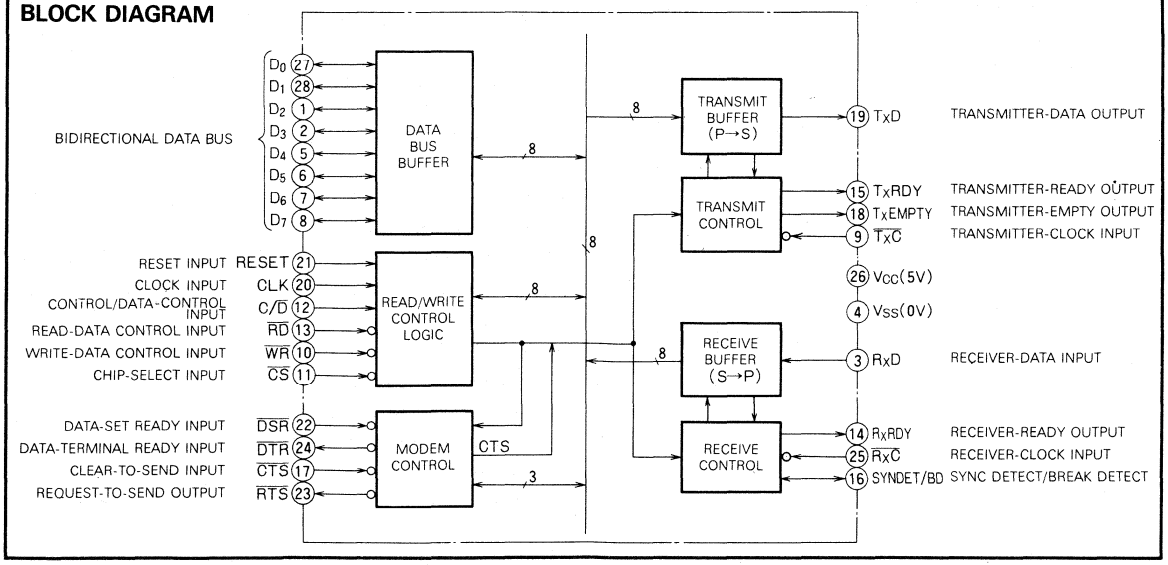
The M5L 8251AP receives parallel-format data from the CPU, converts it into a serial format, and then transmits via the TxD pin. It also receives data sent in via the RxD pin from the external circuit, and converts it into a parallel format for sending to the CPU.

On receipt of parallel-format data for transmission from the CPU or serial data for the CPU from external devices, the M5L 8251AP informs the CPU using the TxRDY or RxRDY pin. In addition, the CPU can read the M5L 8251AP status at any time.

The M5L 8251AP 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.

9

BLOCK DIAGRAM

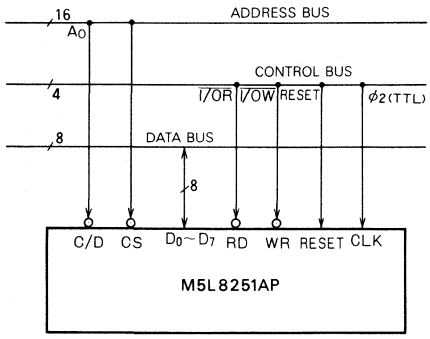


PROGRAMMABLE COMMUNICATION INTERFACE

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.

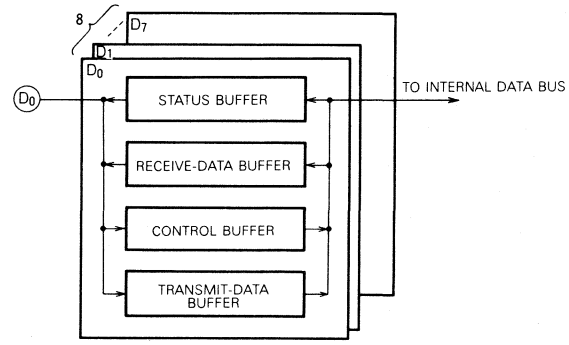
Table 1 M5L8251AP Access Methods

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_3 bit of the command instruction. When D_3 is equal to '1', $\overline{RTS}$ = L, and when D_3 is 0, $\overline{RTS}$ = H.

Command register D_3 = 1 → $\overline{RTS}$ = L

Command register D_3 = 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

9

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)·(T_XEN = 1)·($\overline{CTS}$ = 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 M5L 8251AP 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 M5L 8251AP'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 M5L 8251AP. The USART command instruction contains an internal-reset IR instruction (D_6 bit) that makes it possible to return the M5L 8251AP 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

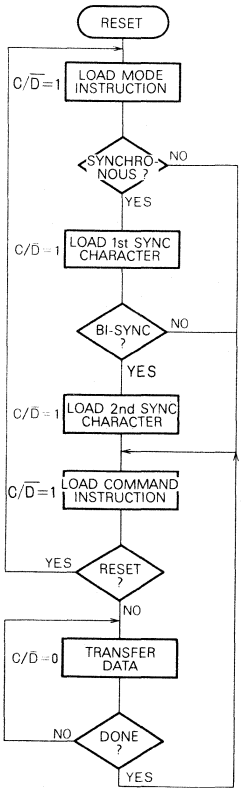


Fig. 4 Mode-instruction format

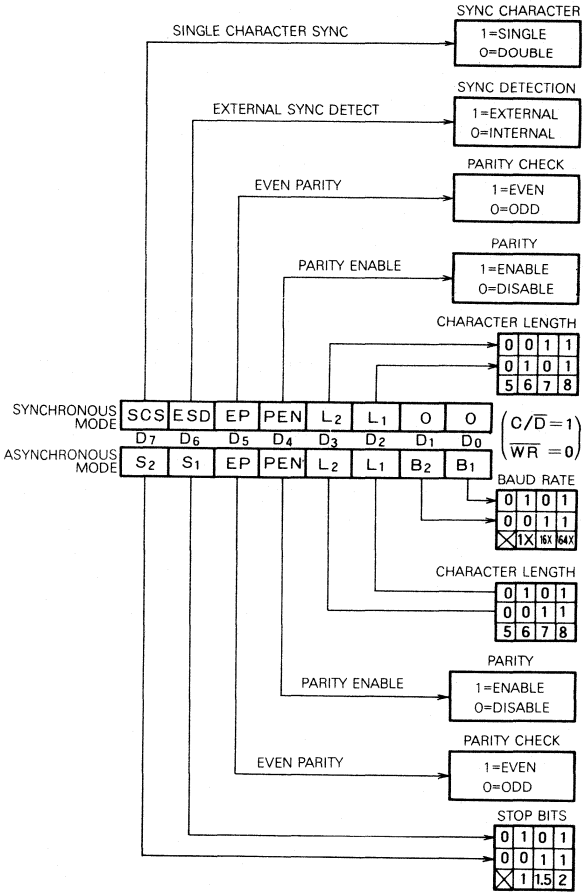
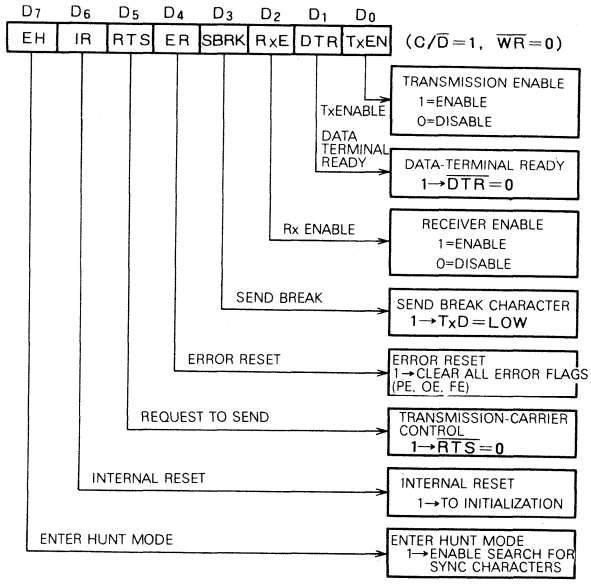


Fig. 5 Command-instruction format



9

PROGRAMMABLE COMMUNICATION INTERFACE

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 M5L 8251AP 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)

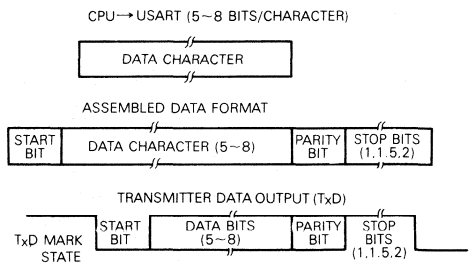
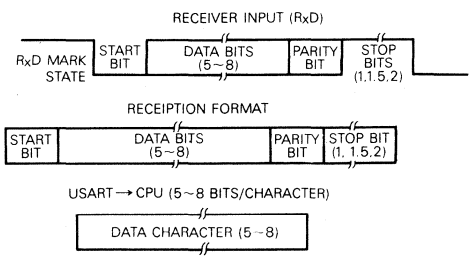


Fig. 7 Asynchronous transmission format II (reception)



Note : When the data character is 5, 6, or 7 bits/character length, the unused bits (for USART→ CPU) are set to zero.

MITSUBISHI ELECTRIC

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 $\overline{RxC}$ 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 $\overline{RxC}$ 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 TxD 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 RxE = 0 does not mean that data reception via the RxD pin is inhibited; it means that the RxRDY is masked and error flags are inhibited.

Fig. 8 Synchronous transmission format I (transmission)

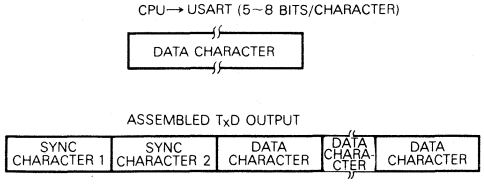
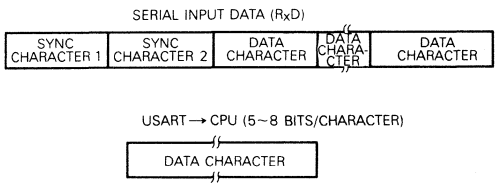


Fig. 9 Synchronous transmission format II (reception)



Note : When the data character is 5, 6, or 7 bits/character length, the unused bits (for USART→ CPU) are set to zero.

PROGRAMMABLE COMMUNICATION INTERFACE

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'.

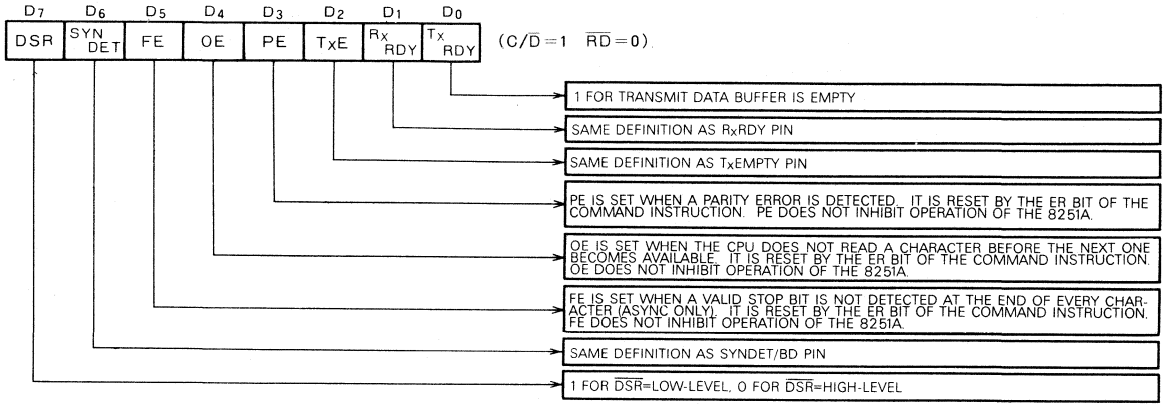
FE: The occurrence of a frame error in the receiver section makes the status information FE '1'.

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#   2D16 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

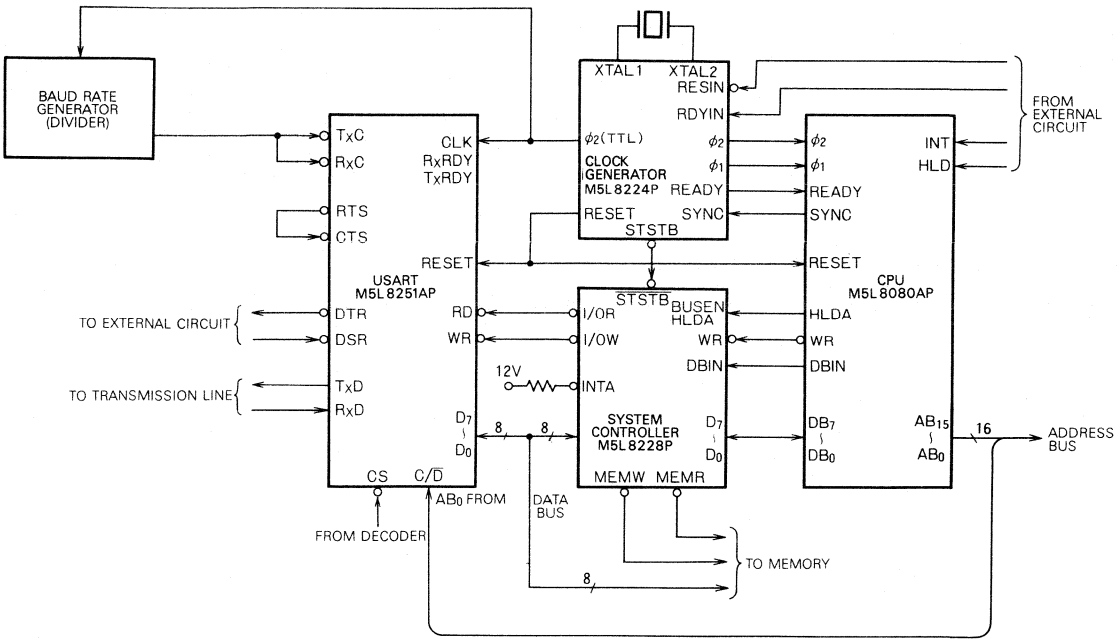
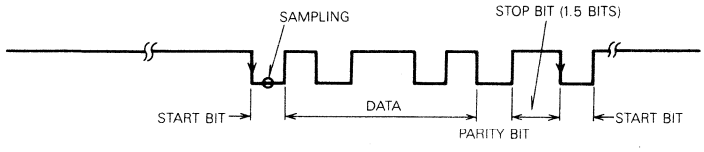


Fig. 12 Example of data transmission



9

PROGRAMMABLE COMMUNICATION INTERFACE

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		−40～125	℃

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}= −400μA	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 ($Ta = 0 \sim 70℃$, $V_{CC} = 5\,V \pm 5\,\%$, $V_{SS} = 0\,V$, 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}		150		$t_{C(\phi)}-90$	ns
$t_{W(\overline{\phi})}$	Clock low pulse width		$t_{\overline{\phi}}$		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 × baud rate	f_{TX}		DC		64	kHz
		16 × baud rate	f_{TX}		DC		310	kHz
		64 × 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}		80			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}, $f_{RX} \leqq 1/(30\,t_{C(\phi)})$. For 16X, 64X baud rate f_{TX}, $f_{RX} \leqq 1/(4.5\,t_{C(\phi)})$

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 $16\,t_{C(\phi)}$.

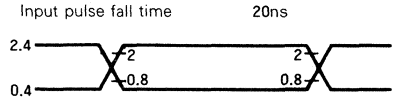
SWITCHING CHARACTERISTICS ($Ta = 0 \sim 70℃$, $V_{CC} = 5\,V \pm 5\,\%$, $V_{SS} = 0\,V$, unless otherwise noted.)

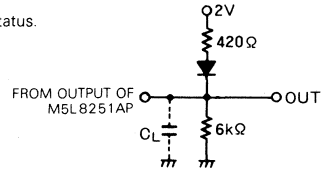
Symbol	Parameter	Alternative symbol	Test conditions (Note 7)	Limits Min	Limits Typ	Limits Max	Unit
$t_{PZV(R-DQ)}$	Output data enable time after read (Note 5)	t_{RD}	$C_L = 150pF$			200	ns
$t_{PVZ(R-DQ)}$	Output data disable time afer read	t_{DF}		10		100	ns
$t_{PZV(TxC-TxD)}$	TxD enable time after falling 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)	$t_{TxRDY\ CLEAR}$				180	ns
$t_{PLH(CLB-RxR)}$	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)	$t_{RxRDY\ CLEAR}$				150	ns
$t_{PLH(RxC-SYD)}$	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 $V_{OH} = 2\,V$, $V_{OL} = 0.8\,V$



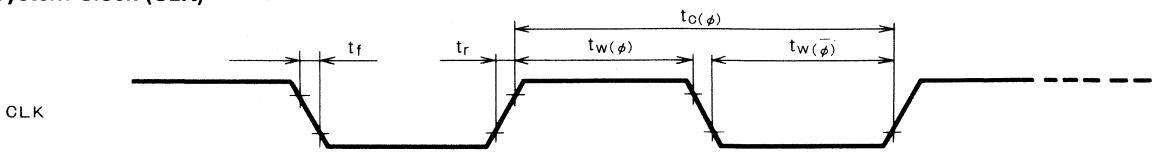


9

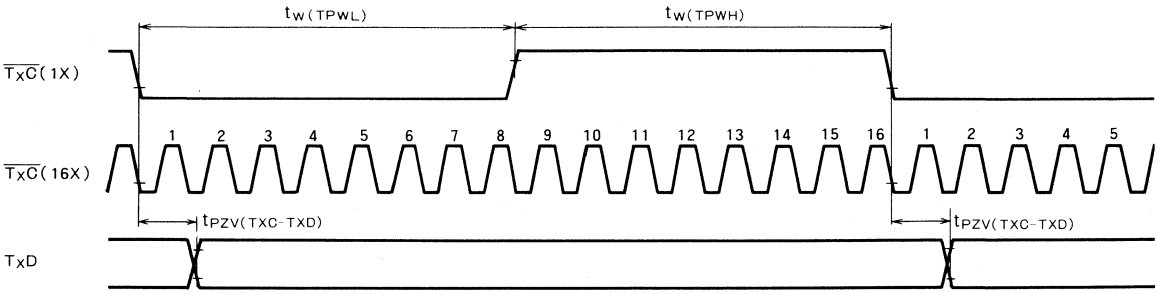
PROGRAMMABLE COMMUNICATION INTERFACE

TIMING DIAGRAMS

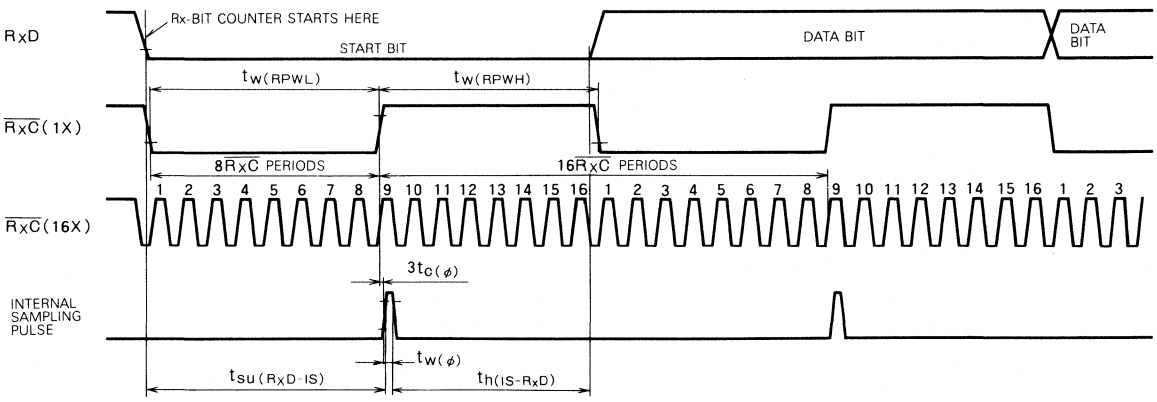
System Clock (CLK)



Transmitter Clock & Data

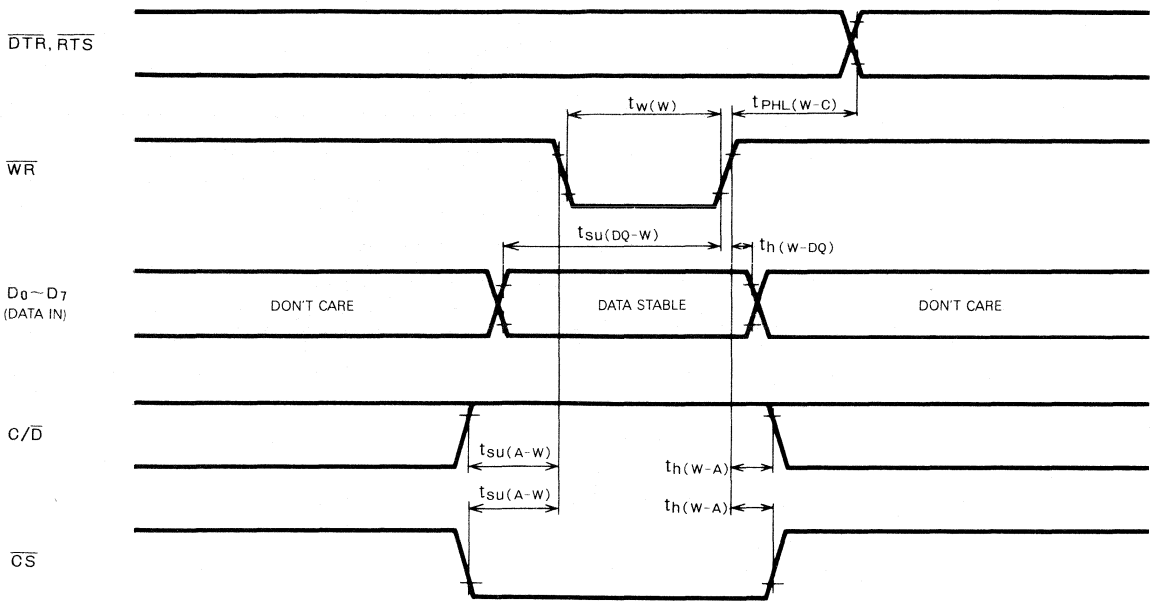


Receiver Clock & Data

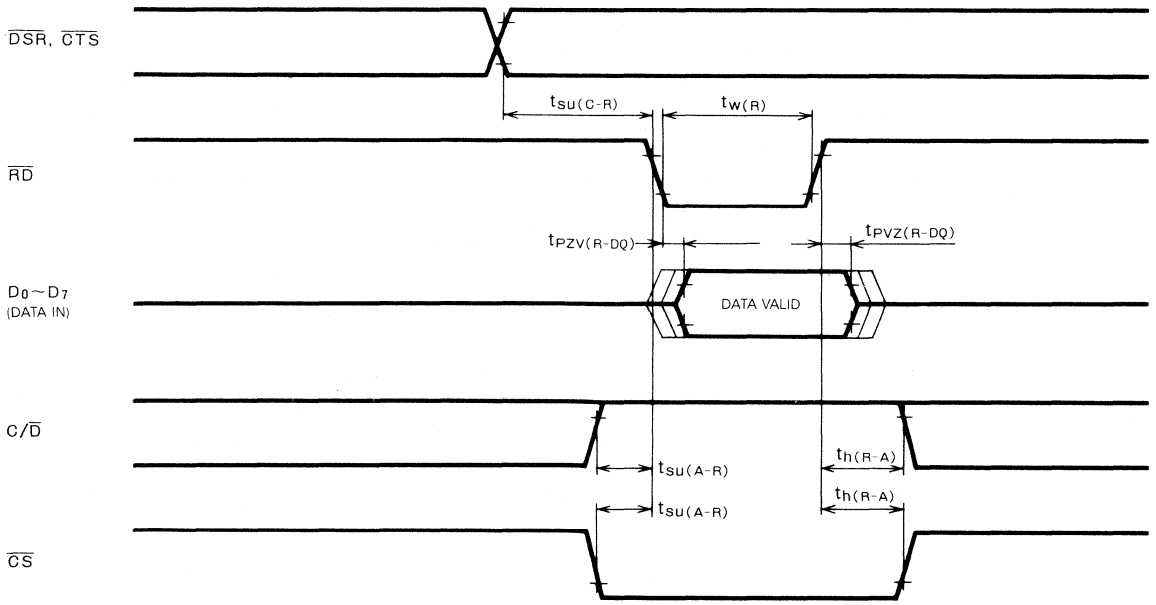


MITSUBISHI ELECTRIC

Write Control Cycle (CPU → USART)



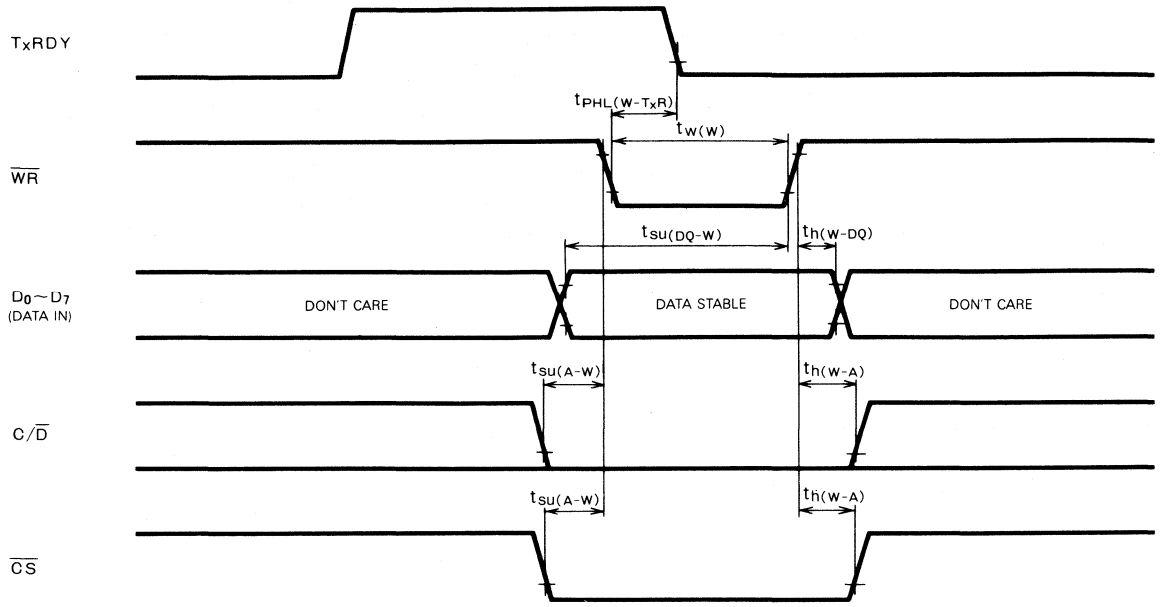
Read Control Cycle



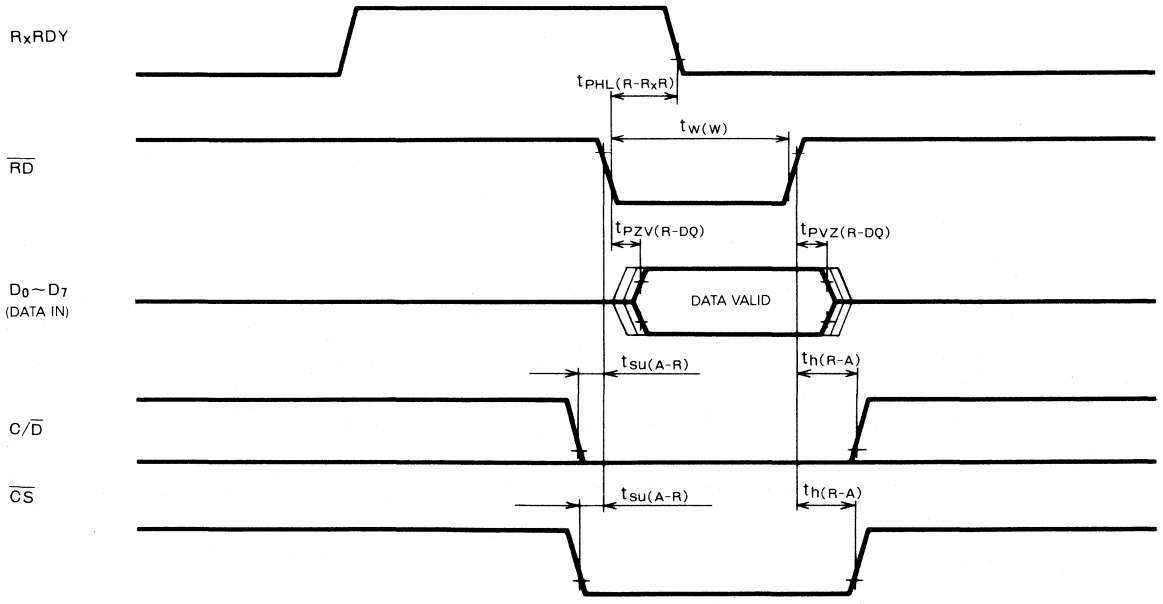
9

PROGRAMMABLE COMMUNICATION INTERFACE

Write Data Cycle

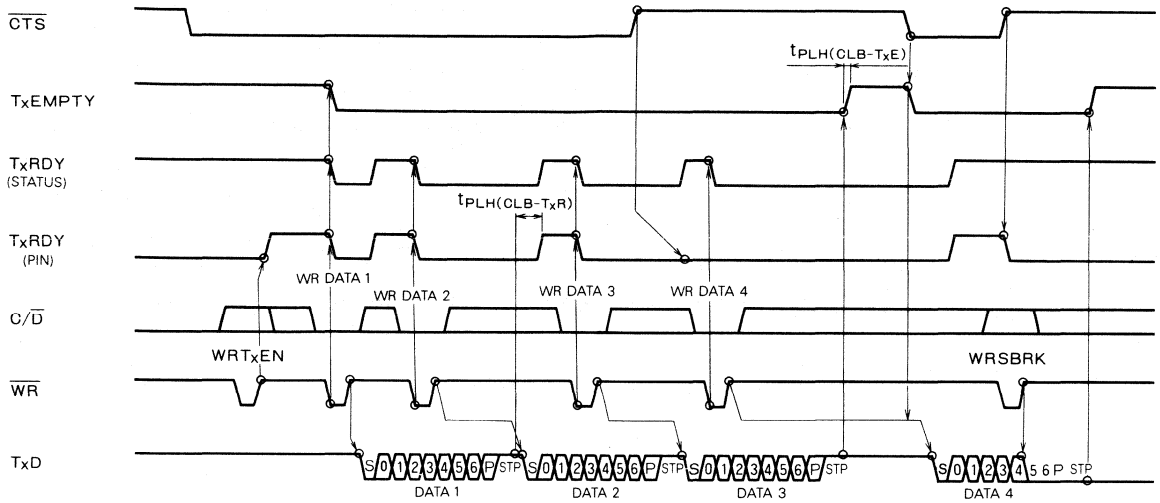


Read Data Cycle



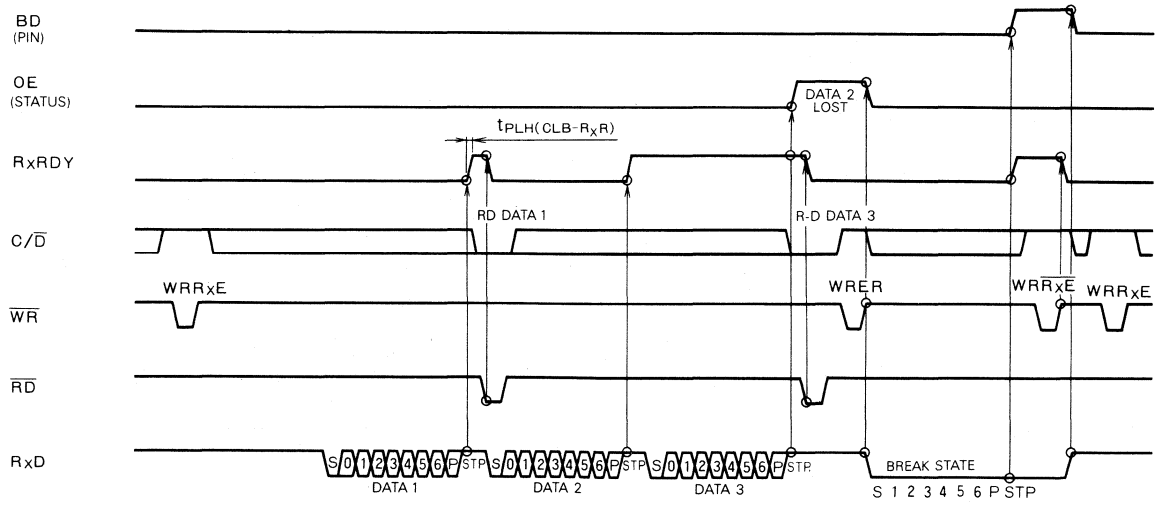
MITSUBISHI ELECTRIC

Transmitter Control & Flag Timing (Async Mode)



Note 8 : Example format = 7 bits/character with parity & 2 stop bits.
9 : TxRDY (pin) = 1 ← (Transmit-data buffer is empty) · (TxEN = 1) · (CTS =0) =1
10 : TxRDY (status) = 1 ← (Transmit-data buffer is empty) = 1

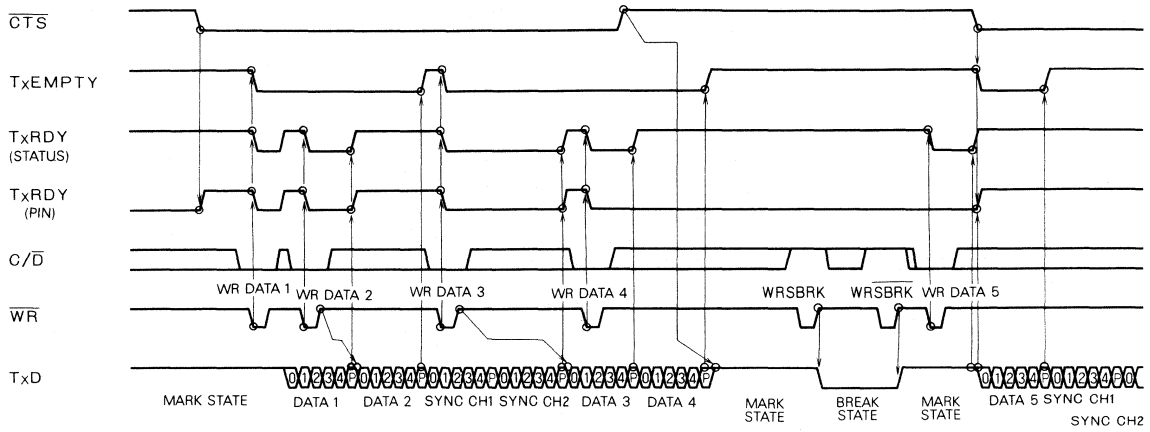
Receiver Control & Flag Timing (Async Mode)



Note 11 : Example format = 7 bits/character with parity & 2 stop bits

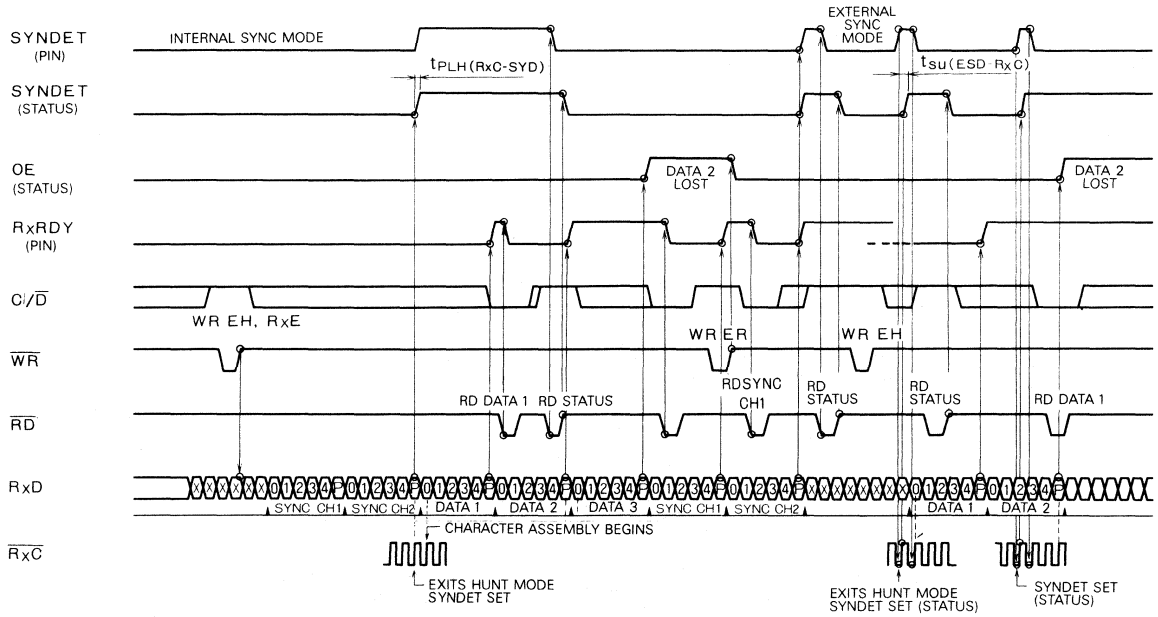
PROGRAMMABLE COMMUNICATION INTERFACE

Transmitter Control & Flat Timing (Sync Mode)



Note 12 : Example format = 5 bits/character with parity, bi-sync characters.

Receiver Control & Flag Timing (Sync Mode)



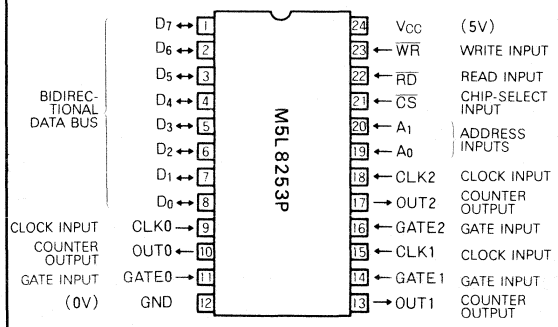
Note 13 : Example format = 5 bits/character with parity, bi-sync characters.

MITSUBISHI ELECTRIC

DESCRIPTION

The M5L 8253P 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 works on a single power supply, and both its input and output can be connected to a TTL circuit.

PIN CONFIGURATION (TOP VIEW)



Outline 24P1

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

APPLICATIONS

Delayed-time setting, pulse counting and rate generation in microcomputers.

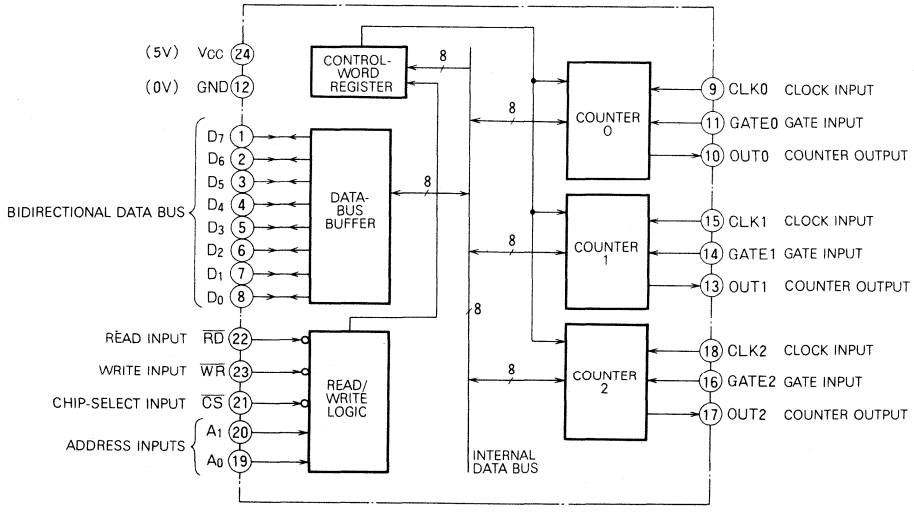
FUNCTION

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.

9

BLOCK DIAGRAM



PROGRAMMABLE INTERVAL TIMER

DESCRIPTION OF FUNCTIONS

Data-Bus Buffer

This 3-state, bidirectional, 8-bit buffer is used to interface the M5L 8253P 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 M5L 8253P. 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 M5L 8253P 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 M5L 8253P 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)

9

PROGRAMMABLE INTERVAL TIMER

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

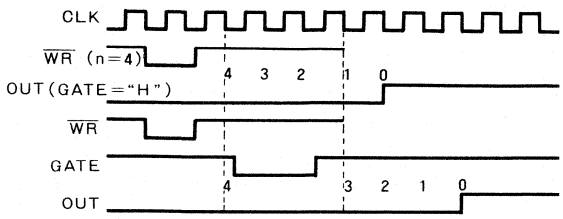


Fig. 2 Mode 1

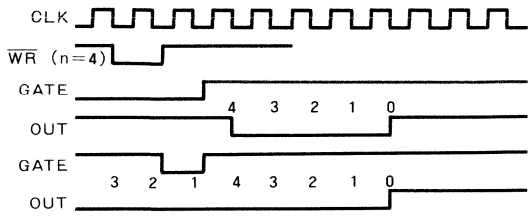


Fig. 3 Mode 2

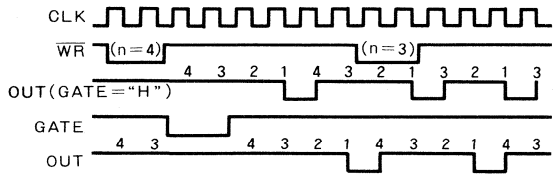


Fig. 4 Mode 3

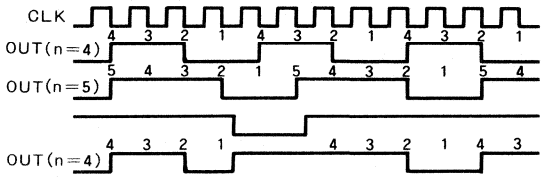


Fig. 5 Mode 4

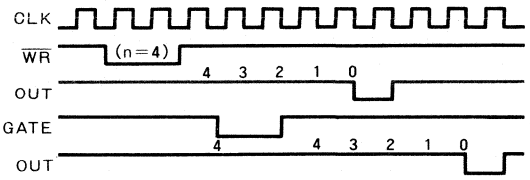
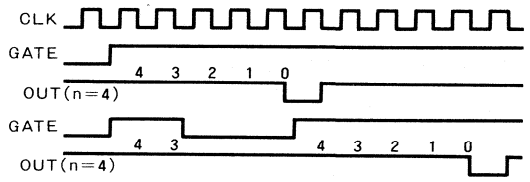


Fig. 6 Mode 5



COUNTER MONITORING

Sometimes the counter must be monitored by reading its count or using it as an event counter. The M5L 8253P 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, 1000 XXXX· · · $D_5 = D_4 = 0$ 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 M5L 8253P 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.

9

PROGRAMMABLE INTERVAL TIMER

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
P_d	Maximum power dissipation	Ta＝25℃	1000	mW
T_{opr}	Operating free-air temperature range		0～70	℃
T_{stg}	Storage temperature range		−40～125	℃

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 (T_a＝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, 25mV_{rms}, Ta＝25℃			10	pF
$C_{i/o}$	Input/output capacitance	$V_{I/OL}$＝GND, f＝1MHz, 25mV_{rms}, Ta＝25℃			20	pF

Note 1 : M5L 8253P : I_{OH}＝−150μA, M5L 8253P-5 : I_{OH}＝−400μA
2 : M5L 8253P : I_{OL}＝1.6mA, M5L 8253P-5 : I_{OL}＝2.2mA

TIMING REQUIREMENTS (T_a=0~70℃, V_{CC}=5V±5%, GND=0V, unless otherwise noted.) (Note 3)

Read Cycle

Symbol	Parameter	Alternative symbol	M5L 8253P			M5L 8253P-5			Unit
			Limits			Limits			
			Min	Typ	Max	Min	Typ	Max	
$t_{W(R)}$	Read pulse width	t_{RR}	400			300			ns
$t_{su(A-R)}$	Address setup time before read	t_{AR}	50			50			ns
$t_{h(R-A)}$	Address hold time after read	t_{RA}	5			5			ns
$t_{rec(R)}$	Read recovery time	t_{RV}	1000			1000			ns

Write Cycle

Symbol	Parameter	Alternative symbol	M5L 8253P			M5L 8253P-5			Unit
			Limits			Limits			
			Min	Typ	Max	Min	Typ	Max	
$t_{W(W)}$	Write pulse width	t_{WW}	400			300			ns
$t_{su(A-W)}$	Address setup time before write	t_{AW}	50			50			ns
$t_{h(W-A)}$	Address hold time after write	t_{WA}	30			30			ns
$t_{su(DQ-W)}$	Data setup time before write	t_{DW}	300			250			ns
$t_{h(W-DQ)}$	Data hold time after write	t_{WD}	40			30			ns
$t_{rec(W)}$	Write recovery time	t_{RV}	1000			1000			ns

Clock and Gate Timing

Symbol	Parameter	Alternative symbol	M5L 8253P			M5L 8253P-5			Unit
			Limits			Limits			
			Min	Typ	Max	Min	Typ	Max	
$t_{W(\phi H)}$	Clock high pulse width	t_{PWH}	230			230			ns
$t_{W(\phi L)}$	Clock low pulse width	t_{PWL}	150			150			ns
$t_{C(\phi)}$	Clock cycle time	t_{CLK}	380		DC	380		DC	ns
$t_{W(GH)}$	Gate high pulse width	t_{GW}	150			150			ns
$t_{W(GL)}$	Gate low pulse width	t_{GL}	100			100			ns
$t_{su(G-\phi)}$	Gate setup time before clock	t_{GS}	100			100			ns
$t_{h(\phi-G)}$	Gate hold time after clock	t_{GH}	50			50			ns

Note 3 : Test conditions: M5L 8253P : C_L = 100pF, M5L 8253P-5 : C_L = 150pF

9

PROGRAMMABLE INTERVAL TIMER

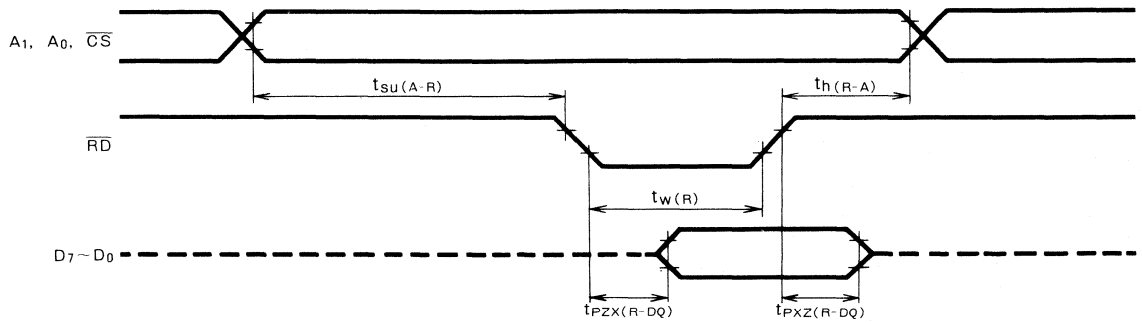
SWITCHING CHARACTERISTICS (T_a = 0~70℃, V_{CC} = 5V ± 5%, V_{SS} = 0V, unless otherwise noted.) (Note 4)

Symbol	Parameter	Alternative symbol	M5L 8253P Limits			M5L 8253P-5 Limits			Unit
			Min	Typ	Max	Min	Typ	Max	
$t_{PZX(R-DQ)}$	Propagation time from read to output	t_{RD}			300			250	ns
$t_{PXZ(R-DQ)}$	Propagation time from read to output floating	t_{DF}	25		125	25		100	ns
$t_{PZX(G-DQ)}$	Propagation time from gate to output	t_{ODG}			300			300	ns
$t_{PZX(\phi-DQ)}$	Propagation time from clock to output	t_{OD}			400			400	ns

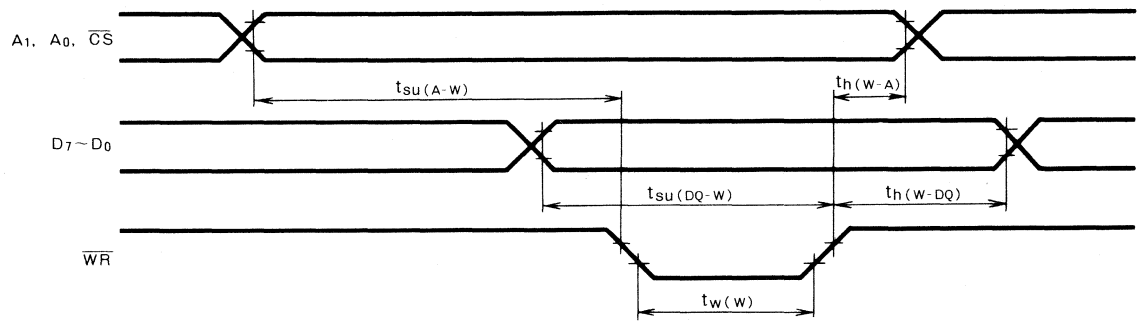
Note 4 : Test conditions: M5L 8253P : C_L = 100pF, M5L 8253P-5 : C_L = 150pF

TIMING DIAGRAMS (Reference Voltage : High = 2.2V, Low = 0.8V)

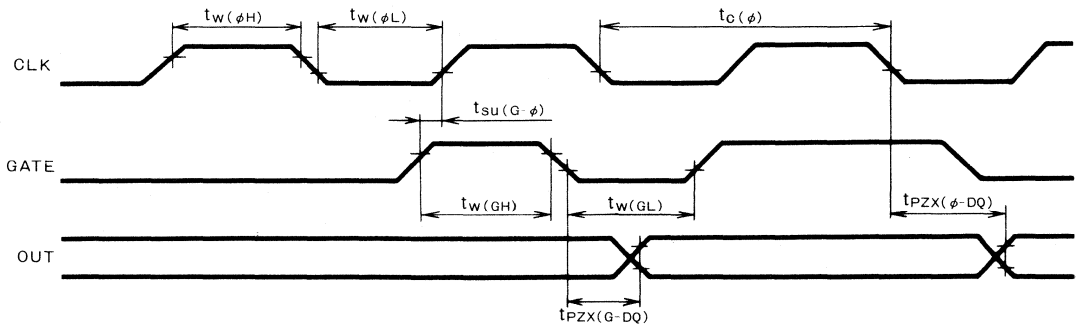
Read Cycle



Write Cycle



Clock and Gate Cycle



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 85 microprocessor

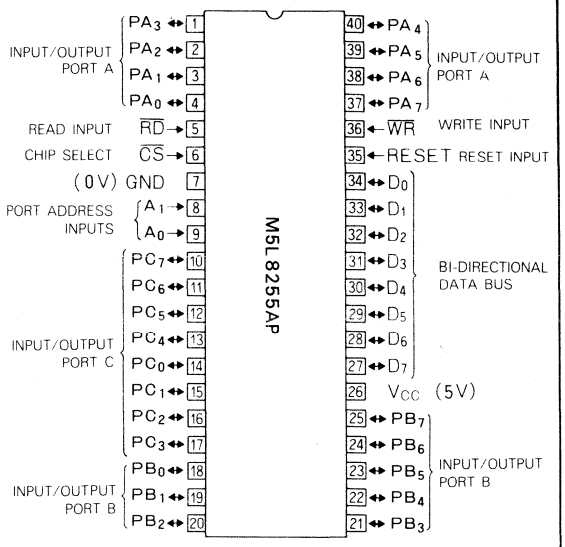
FUNCTION

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.

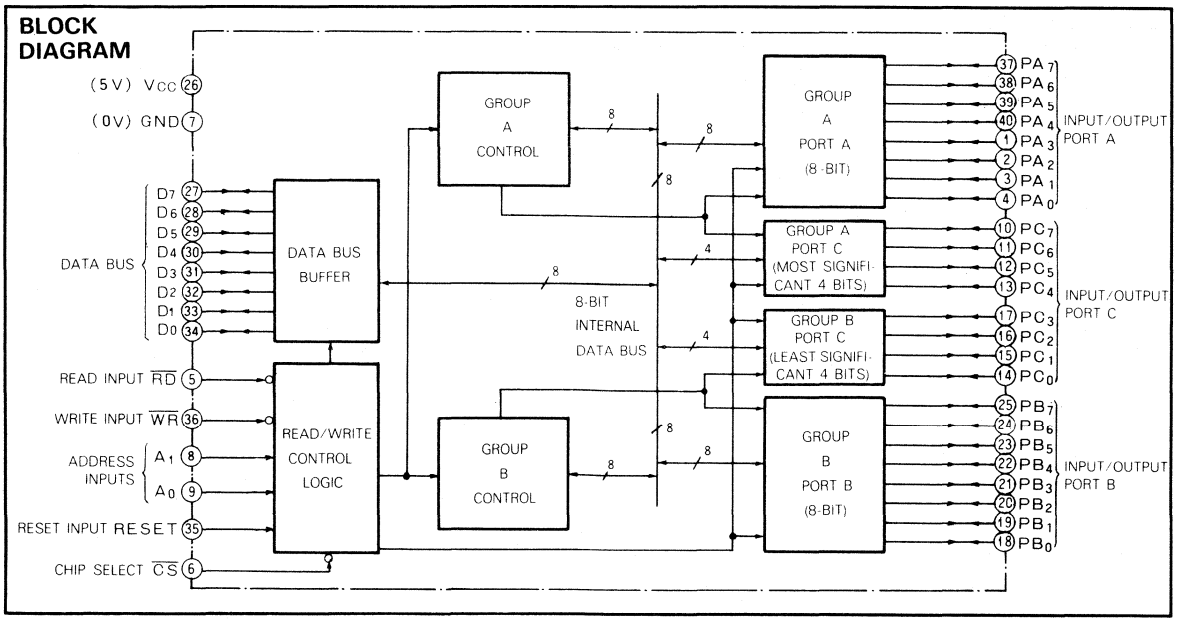
Bit set/reset is controlled by CPU. A high-level reset input (RESET) clears all internal registers, and all ports are set to the input mode (high-impedance state).

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

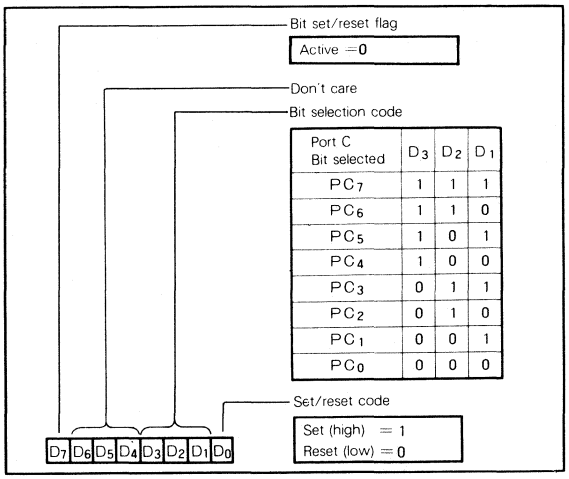
A_1	A_0	$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	Operation
0	0	0	0	1	Data bus ← Port A
0	1	0	0	1	Data bus ← Port B
1	0	0	0	1	Data bus ← Port C
0	0	0	1	0	Port A ← Data bus
0	1	0	1	0	Port B ← Data bus
1	0	0	1	0	Port C ← Data bus
1	1	0	1	0	Control register ← Data bus
X	X	1	X	X	Data bus is in high-impedance 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



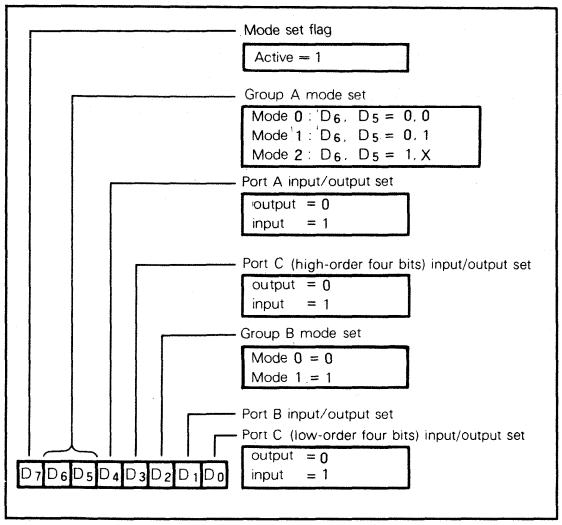
BASIC OPERATING MODES

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)

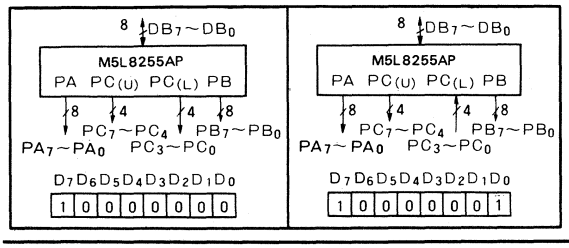
The mode of both group A and group B can be selected independently. The control word format for mode set is shown in Fig. 2.

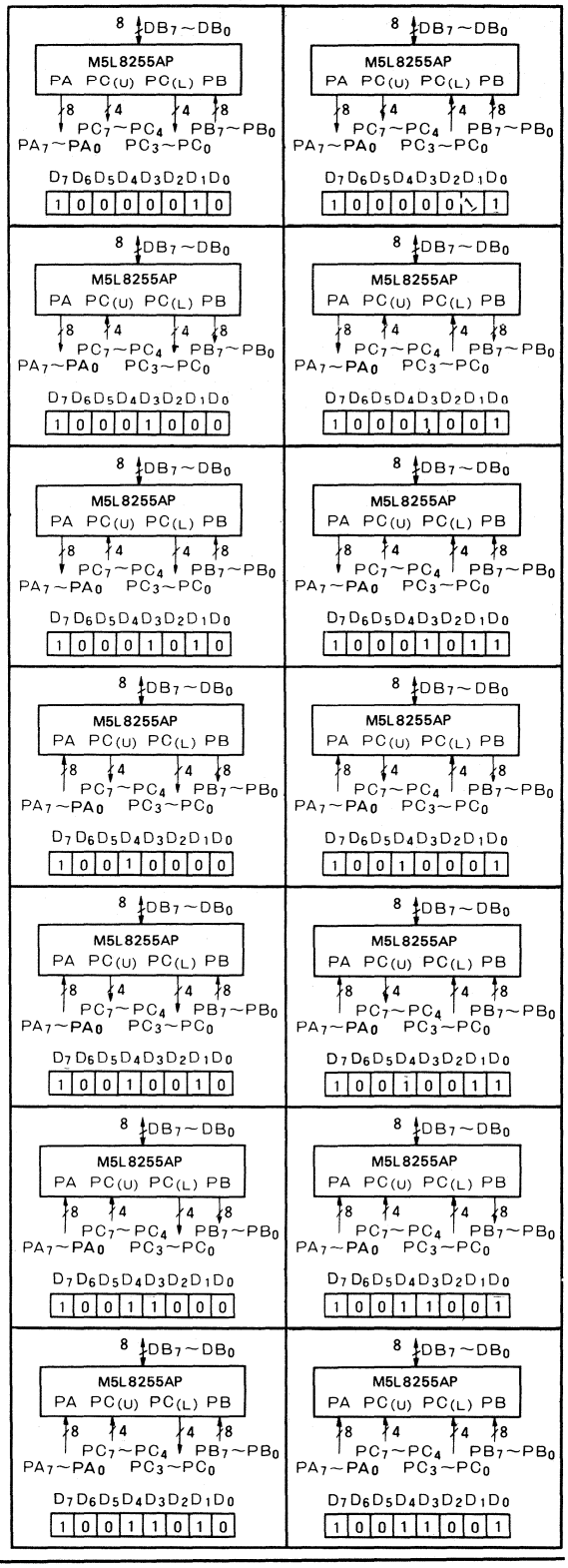
Fig. 2 Control word format for mode set.



1. Mode 0 (Basic Input/Output)

This functional configuration provides simple input and output operations for each of the three ports. No "handshaking" 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.





9

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

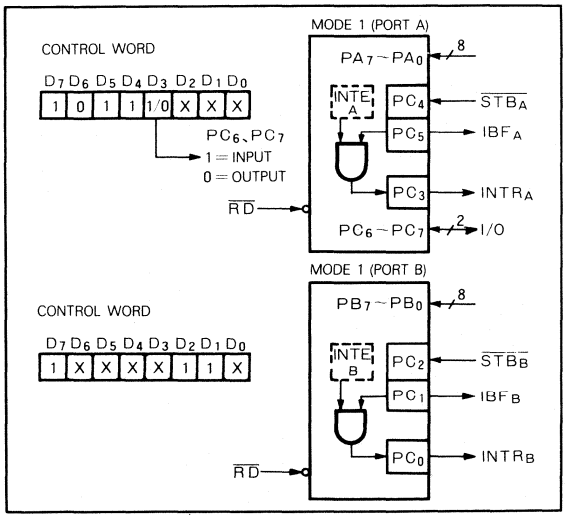
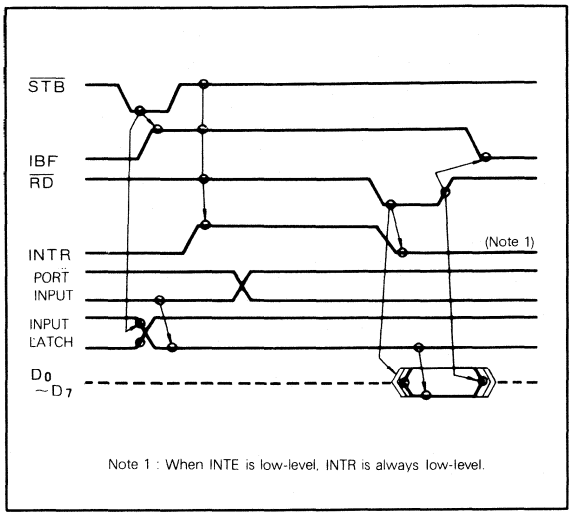


Fig. 4 Timing chart



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

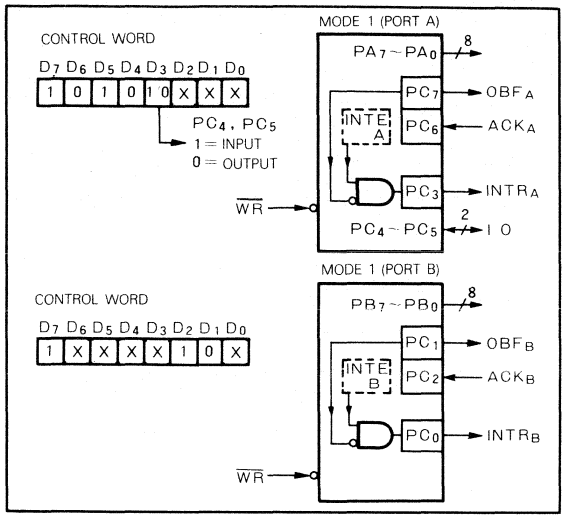


Fig. 6 Timing diagram

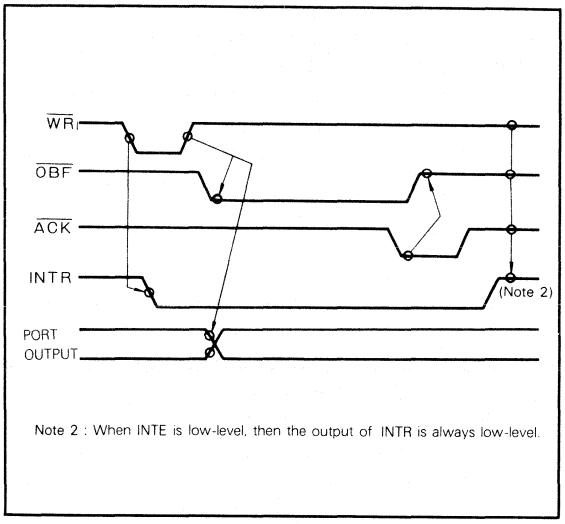


Fig. 7 Mode 1 port A and port B I/O example

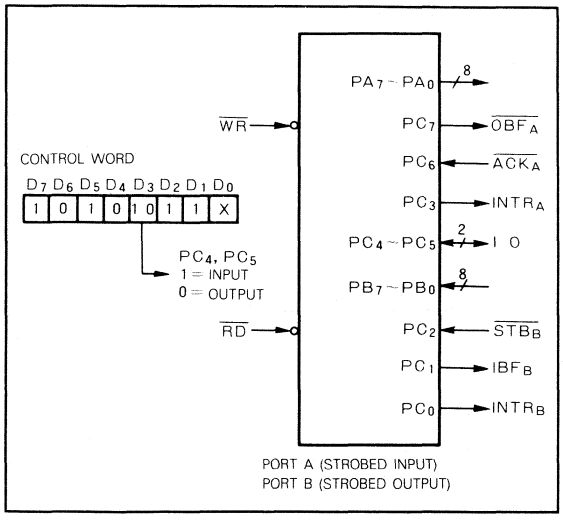
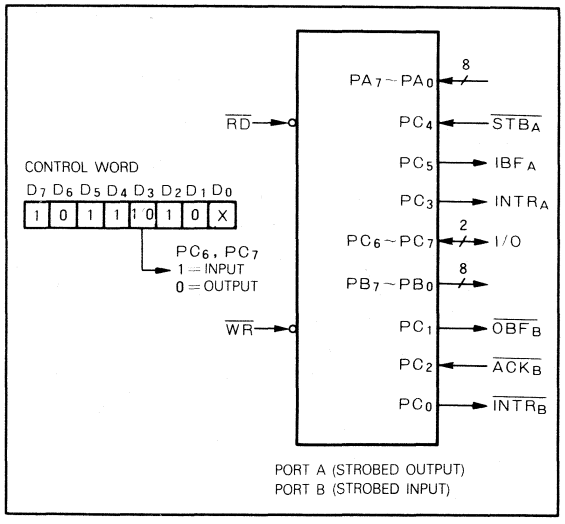


Fig. 8 Mode 1 port A and port B I/O example



9

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)

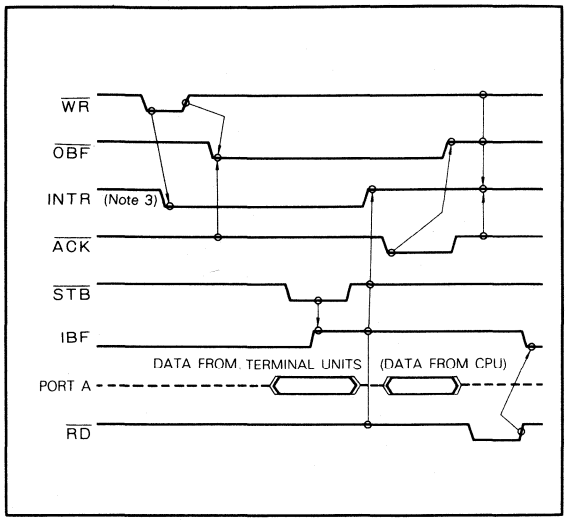
This output is used to interrupt the CPU and its operations the same as in mode 1. There are two interrupt enable flags that correspond to $INTE_A$ for mode 1 output and mode 1 input.

$INTE_1$ 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.

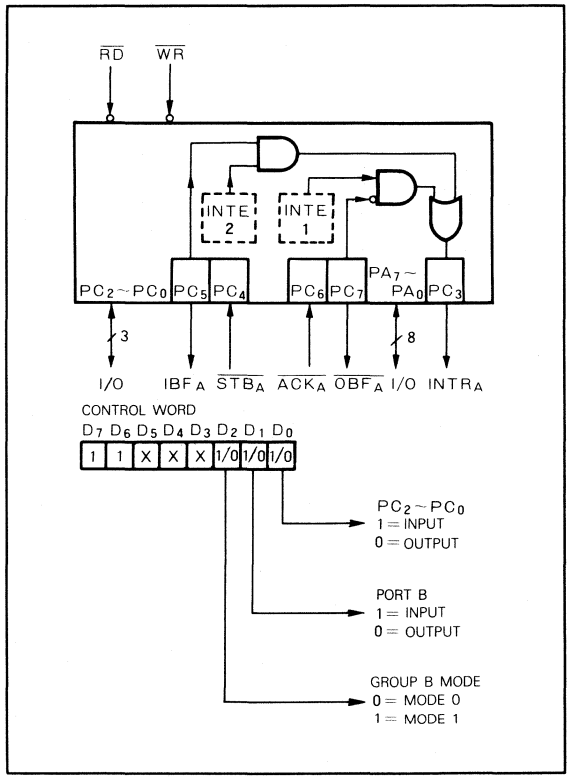
Fig. 9 shows the timing diagram of mode 2, and Fig. 10 is an example of mode 2 operation.

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	$\overline{OBF_A}$	INTE1	IBF_A	INTE2	$INTR_A$	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			
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexa-decimal	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	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.

9

PROGRAMMABLE PERIPHERAL INTERFACE

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
Topr	Operating free-air temperature range		0 ～70	°C
Tstg	Storage temperature range		−40～125	°C

RECOMMENDED OPERATING CONDITIONS (T_a=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 (T_a=0～70°C, V_{CC}=5V±5%, GND=0V, unless otherwise noted)

Symbol	Parameter		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 T_a=25°C				10	pF
$C_{i/o}$	Input/output terminal capacitance		$V_{I/OL}$=GND, f=1MHz, 25mVrms T_a=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.

9

TIMING REQUIREMENTS (T_a=0～70°C, V_{CC}=5V±5%, GND=0V, unless otherwise noted)

Symbol	Prameter	Alternative symbol	Limits						Unit
			M5L 8255AP			M5L 8255AP-5			
			Min	Typ	Max	Min	Typ	Max	
$t_{W(R)}$	Read pulse width	t_{RR}	300			300			ns
$t_{SU(PER)}$	Peripheral setup time before read	t_{IR}	0			0			ns
$t_{h(R-PE)}$	Peripheral hold time after read	t_{HR}	0			0			ns
$t_{SU(A-R)}$	Address setup time before read	t_{AR}	0			0			ns
$t_{h(R-A)}$	Address hold time after read	t_{RA}	0			0			0 ns
$t_{W(W)}$	Write pulse width	t_{WW}	400			300			ns
$t_{SU(DQ-W)}$	Data setup time before write	t_{DW}	100			100			ns
$t_{h(W-DQ)}$	Data hold time after write	t_{WD}	50			50			ns
$t_{SU(A-W)}$	Address setup time before write	t_{AW}	0			0			ns
$t_{h(W-A)}$	Address hold time after write	t_{WA}	40			40			ns
$t_{W(ACK)}$	Acknowledge pulse width	t_{AK}	300			300			ns
$t_{W(STB)}$	Strobe pulse width	t_{ST}	500			500			ns
$t_{SU(PE-STB)}$	Peripheral setup time before strobe	t_{PS}	0			0			ns
$t_{h(STB-PE)}$	Peripheral hold time after strobe	t_{PH}	180			180			ns
$t_{C(RW)}$	Read/write cycle time	t_{RV}	850			850			ns

PROGRAMMABLE PERIPHERAL INTERFACE

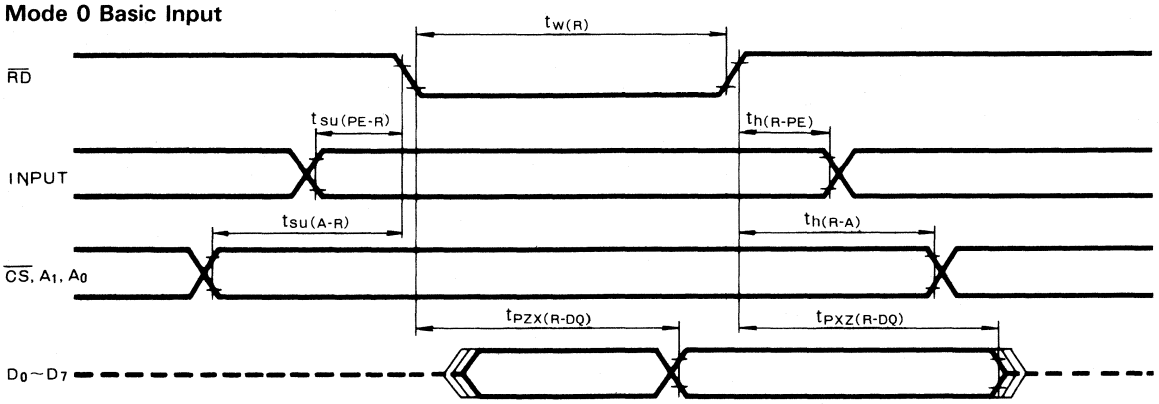
SWITCHING CHARACTERISTICS (T_a=0~70°C, V_{CC}=5V±5%, unless otherwise noted)

Symbol	Parameter	Alternative symbol	M5L 8255AP Limits			M5L 8255AP-5 Limits			Unit
			Min	Typ	Max	Min	Typ	Max	
$t_{PZX(R-DQ)}$	Propagation time from read to data output	t_{RD}			250			250	ns
$t_{PXZ(R-DQ)}$	Propagation time from read to data floating (Note 12)	t_{DF}			150			100	ns
$t_{PHL(W-PE)}$ $t_{PLH(W-PE)}$	Propagation time from write to output	t_{WB}			350			350	ns
$t_{PLH(STB-IBF)}$	Propagation time from strobe to IBF flag	t_{SIB}			300			300	ns
$t_{PLH(STB-INTR)}$	Propagation time from strobe to interrupt	t_{SIT}			300			300	ns
$t_{PHL(R-INTR)}$	Propagation time from read to interrupt	t_{RIT}			400			400	ns
$t_{PHL(R-IBF)}$	Propagation time from read to IBF flag	t_{RIB}			300			300	ns
$t_{PHL(W-INTR)}$	Propagation time from write to interrupt	t_{WIT}			850			850	ns
$t_{PHL(W-OBF)}$	Propagation time from write to $\overline{OBF}$ flag	t_{WOB}			700			700	ns
$t_{PLH(ACK-OBF)}$	Propagation time from acknowledge to $\overline{OBF}$ flag	t_{AOB}			350			350	ns
$t_{PLH(ACK-INTR)}$	Propagation time from acknowledge to interrupt	t_{AIT}			350			350	ns
$t_{PZX(ACK-PE)}$	Propagation time from acknowledge to data output	t_{AD}			300			300	ns
$t_{PXZ(ACK-PE)}$	Propagation time from acknowledge to data output (Note 11)	t_{KD}			250			250	ns

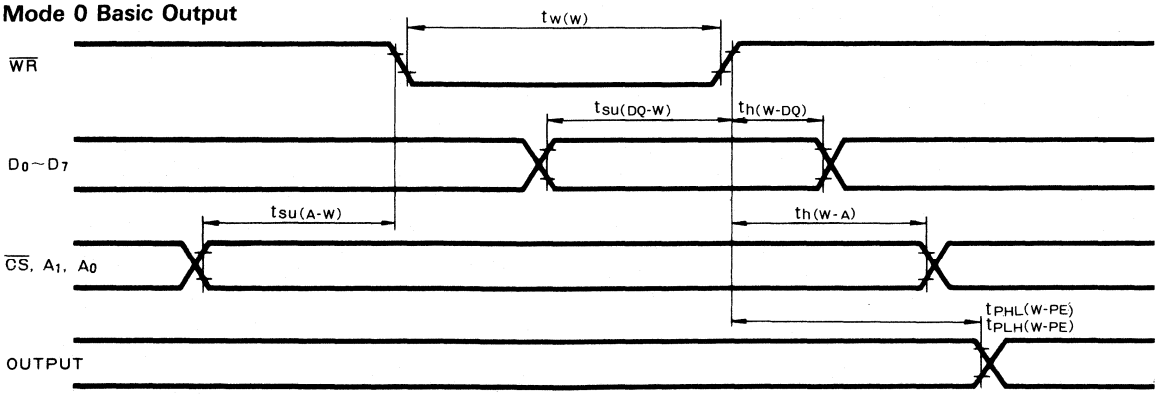
Note 11 : Measurement conditions:
C_L = 100pF for M5L8255AP, S
C_L = 150pF for M5L8255AP-5, S-5
12 : Measurement conditions of note 11 are not applied.

TIMING DIAGRAMS REFERENCE LEVEL = "H"=2V, "L"=0.8V

Mode 0 Basic Input

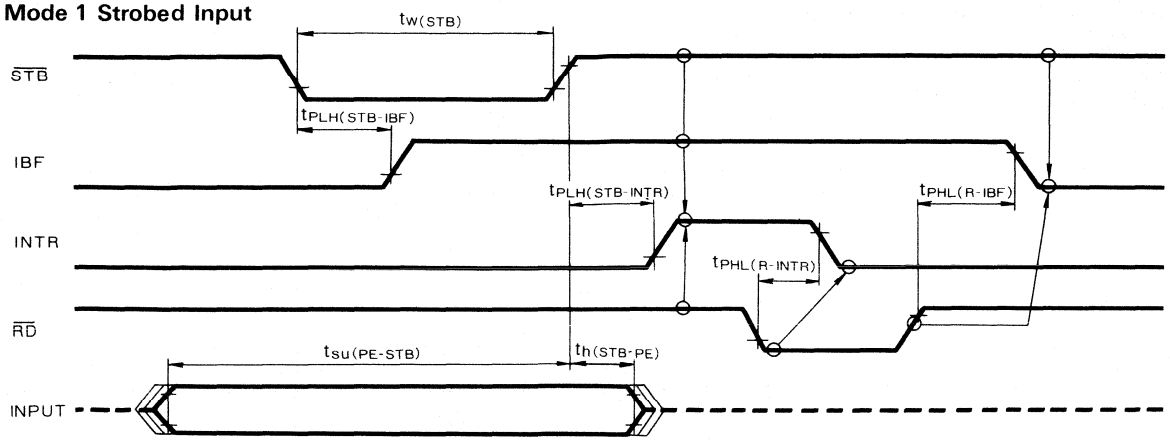


Mode 0 Basic Output

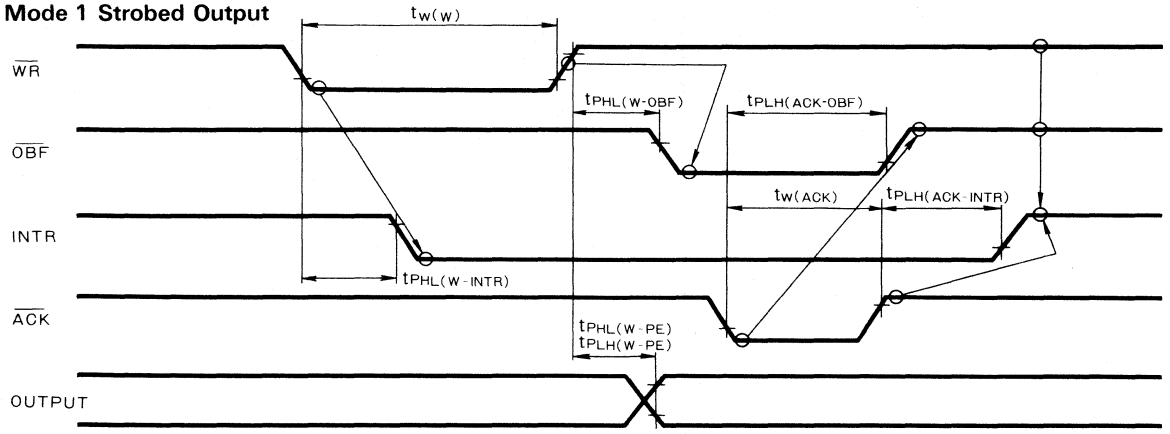


MITSUBISHI ELECTRIC

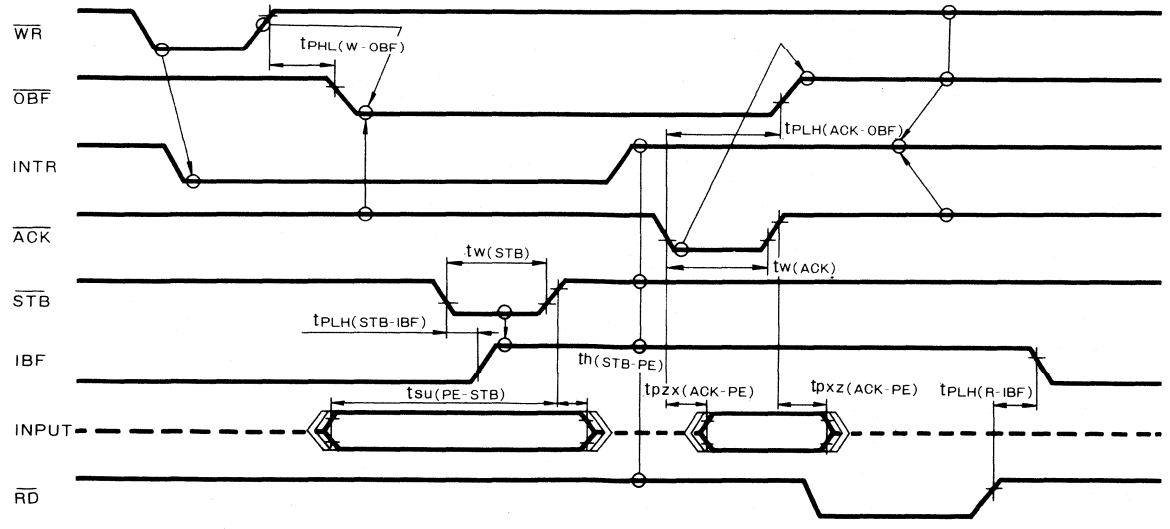
Mode 1 Strobed Input



Mode 1 Strobed Output



Mode 2 Bidirectional



Note 13 : $INTR = IBF \cdot \overline{MASK} \cdot \overline{STB} \cdot \overline{RD} + \overline{OBF} \cdot \overline{MASK} \cdot \overline{ACK} \cdot \overline{WR}$

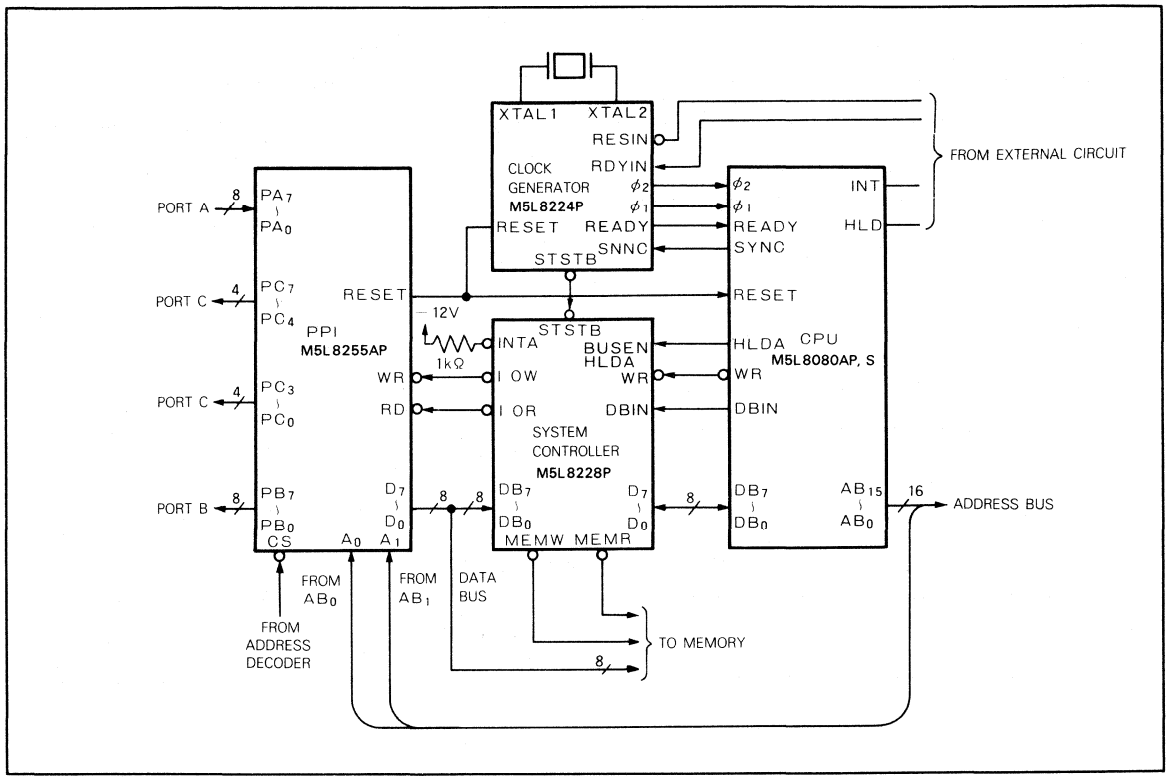
9

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.

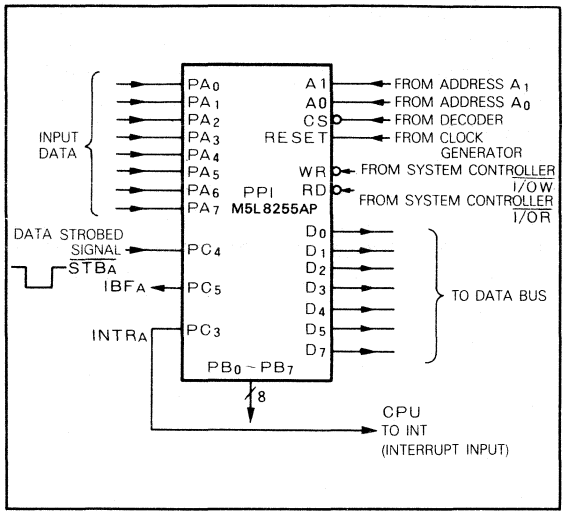
```
IN     00#    CPU A register ← Port A
OUT    01#    Port B ← A register
MVI    A, 01# Bit-setting control word for PC0
OUT    03#    Outputting to control address
              (CS = "0", A1 = A0 = "1")
```

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#   PC4 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:

```
0038₁₆  DI
        IN    00#  CPU register A ← Port A
                   PC3 interrupt signal becomes
                   low-level
        RET
```

9

PROGRAMMABLE PERIPHERAL INTERFACE

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:

IN 01# (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:

IN 00# (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:

OUT 00# (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.

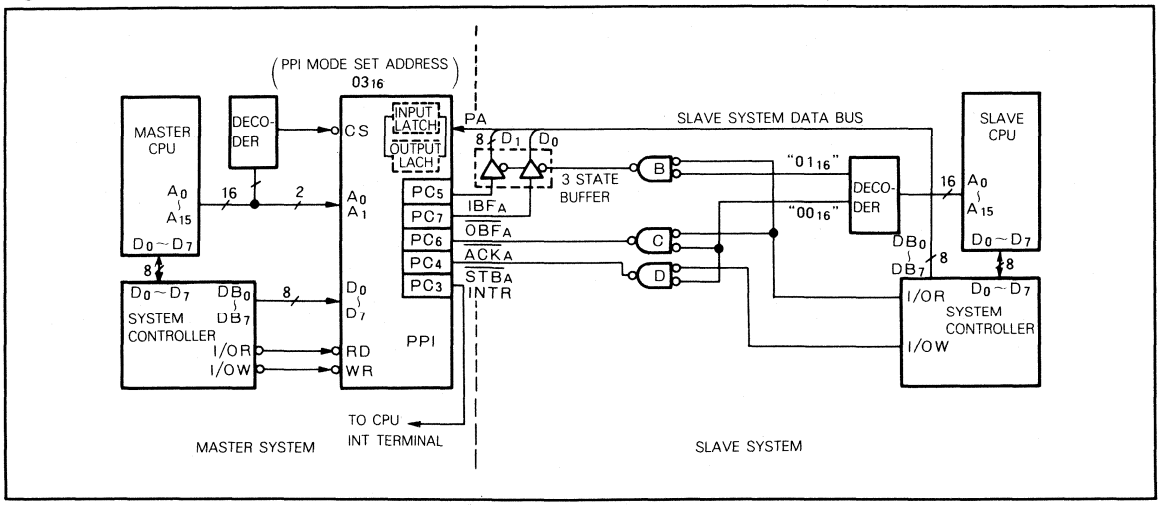
Actual operations are as follows:

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.

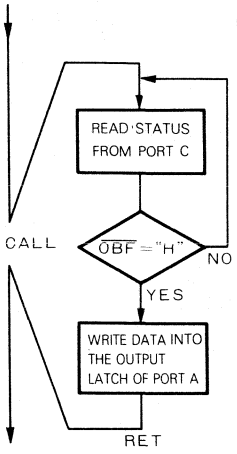
The operation, in mode 2, for group A of the PPI is considered here.

Fig. 13 A circuit for an application using mode 2



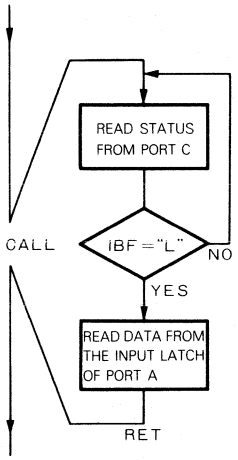
MITSUBISHI ELECTRIC

1. Master CPU subroutine for transmitting data to the slave CPU.



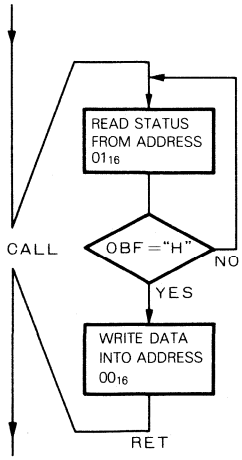

```
MOUT   PUSH   PSW
OBF    IN     02#
       ANI    80#
       JNZ    OBF
       POP    PSW
       OUT    00#
       RET
```

2. Subroutine for receiving data from the slave CPU.



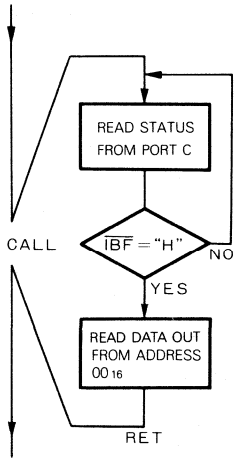

```
MIN    IN     02#
       ANI    20#
       JZ     MIN
       IN     00#
       RET
```

3. Slave CPU subroutine for transmitting data to the master CPU.




```
SOUT   PUSH   PSW
IBF    IN     02#
       ANI    02#
       JNZ    IBF
       POP    PSW
       OUT    00#
       RET
```

4. Subroutine for receiving data from the master CPU.




```
SIN    IN     02#
       ANI    01#
       JZ     SIN
       IN     00#
       RET
```

9

PROGRAMMABLE PERIPHERAL INTERFACE

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)

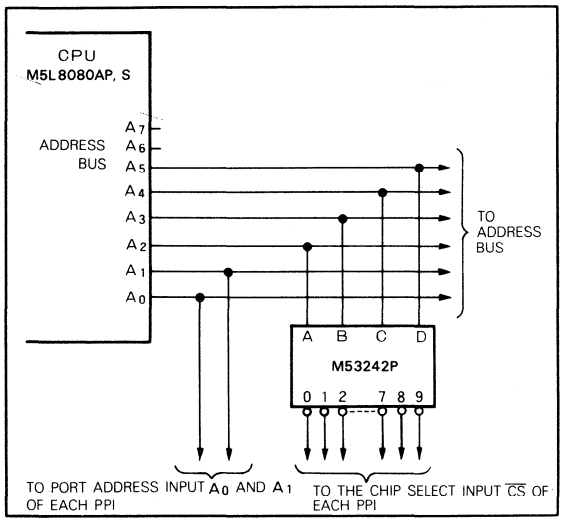
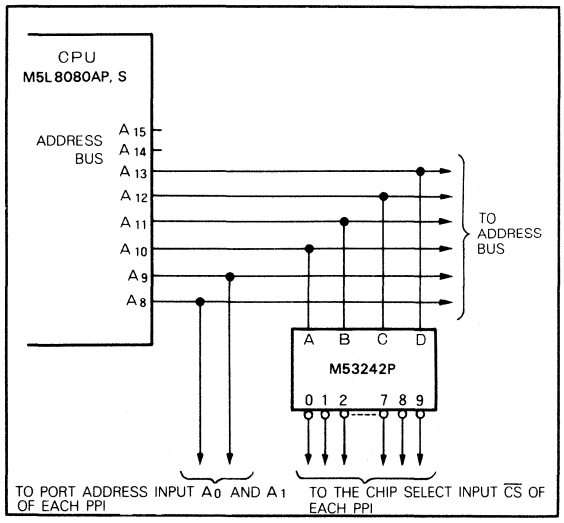


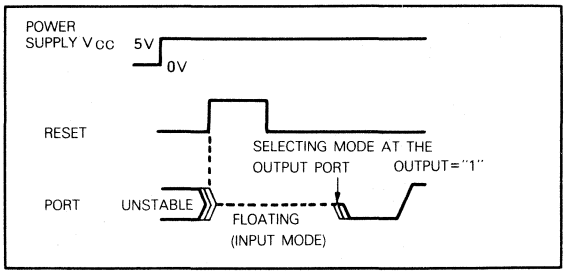
Fig. 15 PPI address decoding (case 2)



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.

DESCRIPTION

The M5L 8257P 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.

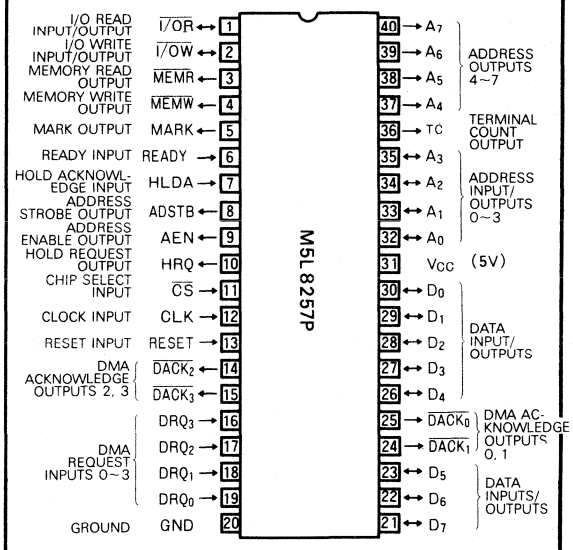
FEATURES

- 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 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 M5L 8257P controller is used in combination with the M5L 8212P 8-bit input/output port in 8-bit microcomputer systems.

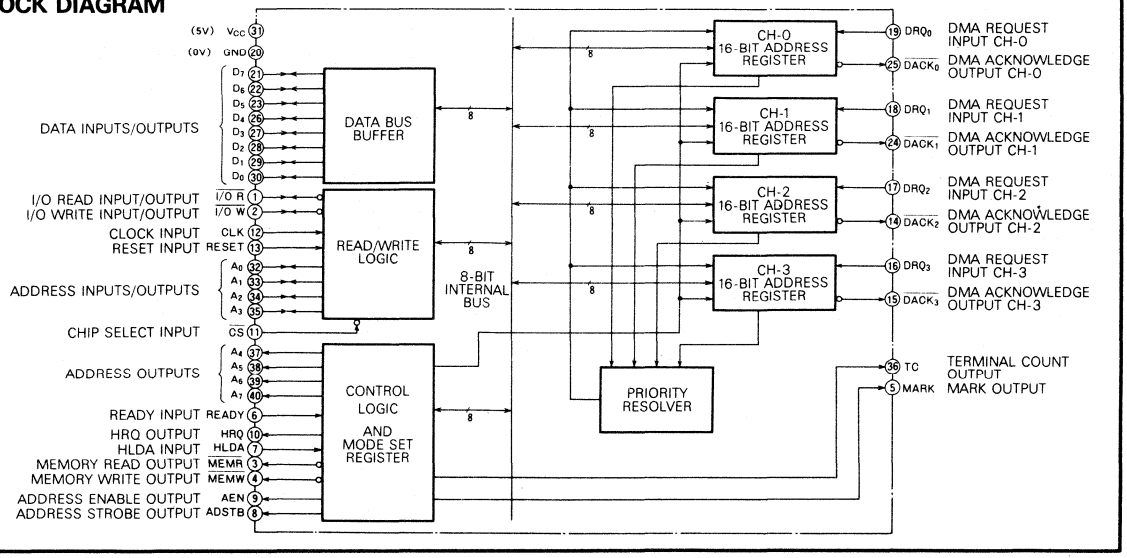
It consists of a channel section to acknowledge DMA requests, control logic to exchange commands and data with the CPU, read/write logic, and registers to hold transfer addresses and count the number of bytes to be transferred.

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 M5L 8257P 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~D_7. The contents of the low-order 8 bits are transmitted through pins A_0~A_7. After address transmission, DMA transfer can be started by dispatching read and write signals to the memories and peripherals.

9

BLOCK DIAGRAM



PROGRAMMABLE DMA CONTROLLER

OPERATION

Data-Bus Buffer

This three-state, bidirectional, 8-bit buffer interfaces the M5L 8257P 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 is in slave-mode operation, this three-state, 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 is in slave-mode operation, is 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 and is connected to the $\phi_{2(TTL)}$ output of the M5L 8224P clock generator.

Reset Input (RESET)

This asynchronous input clears all registers and control lines inside the M5L 8257P.

Address Inputs/Outputs (A_0~A_3)

The four bits of these input/output pins are bidirectional. When the M5L 8257P 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 M5L 8257P 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 during all DMA cycles.

Ready Input (READY)

This asynchronous input is used to extend the memory read and write cycles in the M5L 8257P 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.

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 M5L 8212P 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 M5L 8228P 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.

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~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~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.
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.

PROGRAMMABLE DMA CONTROLLER

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	Power dissipation (max.)	Ta＝25℃	1000	mW
Topr	Operating free-air temperature range		0～70	℃
Tstg	Storage temperature range		−65～150	℃

RECOMMENDED OPERATING CONDITIONS (Ta＝ 0 ～75℃, unless otherwise noted.)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
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
V_{IL}	Low-level input voltage			0.8	V

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

Symbol	Parameter	Conditions	Limits			Unit
			Min	Typ	Max	
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		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				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	M5L 8257P Limits			M5L 8257P-5 Limits			Unit
			Min	Typ	Max	Min	Typ	Max	
$t_{w(R)}$	Read pulse width	T_{RR}	250			250			ns
$t_{su(A-R)}$ $t_{su(\overline{CS}-R)}$	Address or $\overline{CS}$ setup time before read	T_{AR}	0			0			ns
$t_{h(R-A)}$ $t_{h(R-\overline{CS})}$	Address or $\overline{CS}$ hold time after read	T_{RA}	0			0			ns
$t_{su(R-DQ)}$	Data setup time before read	T_{RD}	0		300	0		200	ns
$t_{h(R-DQ)}$	Data hold time after read	T_{DF}	20		150	20		100	ns
$t_{w(W)}$	Write pulse width	T_{WW}	200			200			ns
$t_{su(A-W)}$	Address setup time before write	T_{AW}	20			20			ns
$t_{h(W-A)}$	Address hold time after write	T_{WA}	0			0			ns
$t_{su(DQ-W)}$	Data setup time before write	T_{DW}	200			200			ns
$t_{h(W-DQ)}$	Data hold time after write	T_{WD}	0			0			ns
$t_{w(RST)}$	Reset pulse width	T_{RSTW}	300			300			ns
$t_{su(V_{CC}-RST)}$	Supply voltage setup time before reset	T_{RSTD}	500			500			μs
tr	Input signal rise time	Tr			20			20	ns
tf	Input signal fall time	Tf			20			20	ns
$t_{su(RST-W)}$	Reset setup time before write	T_{RSTS}	2			2			$t_{c(\phi)}$
$t_{c(\phi)}$	Clock cycle time	T_{CY}	0.32		4	0.32		4	μs
$t_{w(\phi)}$	Clock pulse width	T_Q	120		$0.8t_{c(\phi)}$	80		$0.8t_{c(\phi)}$	ns
$t_{su(DRQ-\phi)}$	DRQ setup time before clock	T_{QS}	120			120			ns
$t_{h(HLDA-DRQ)}$	DRQ hold time after HLDA	T_{QH}	0			0			ns
$t_{su(HLDA-\phi)}$	HLDA setup time before clock	T_{HS}	100			100			ns
$t_{su(RDY-\phi)}$	Ready setup time before clock	T_{RS}	30			30			ns
$t_{h(\phi-RDY)}$	Ready hold time after clock	T_{RH}	20			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	M5L 8257P Limits			M5L 8257P-5 Limits			Unit
			Min	Typ	Max	Min	Typ	Max	
$t_{PLH(\phi-HRQ)}$ $t_{PHL(\phi-HRQ)}$	Propagation time from clock to HRQ (Note 3)	T_{DQ}			160			160	ns
$t_{PLH(\phi-HRQ)}$ $t_{PHL(\phi-HRQ)}$	Propagation time from clock to HRQ (Note 5)	T_{DQI}			250			250	ns
$t_{PLH(\phi-AEN)}$	Propagation time from clock to AEN (Note 3)	T_{AEL}			300			300	ns
$t_{PHL(\phi-AEN)}$	Propagation time from clock to AEN (Note 3)	T_{AET}			200			200	ns
$t_{PZV(AEN-A)}$	Propagation time from AEN to address active (Note 6)	T_{AEA}	20			20			ns
$t_{PZV(\phi-A)}$	Propagation time from clock to address active (Note 4)	T_{FAAB}			250			250	ns
$t_{PVZ(\phi-A)}$	Propagation time from clock to address floating (Note 4)	T_{AFAB}			150			150	ns
$t_{su(\phi-A)}$	Address setup time after clock (Note 4)	T_{ASM}			250			250	ns
$t_{h(\phi-A)}$	Address hold time after clock (Note 4)	T_{AH}	$t_{su(\phi-A)}-50$			$t_{su(\phi-A)}-50$			ns
$t_{h(R-A)}$	Address hold time after read (Note 6)	T_{AHR}	60			60			ns
$t_{h(W-A)}$	Address hold time after write (Note 6)	T_{AHW}	300			300			ns
$t_{PZV(\phi-DQ)}$	Propagation time from clock to data active	T_{FADB}			300			300	ns
$t_{PVZ(\phi-DQ)}$	Propagation time from clock to data floating (Note 4)	T_{AFDB}	$t_{PHL(\phi-ASTB)}+20$		250	$t_{PHL(\phi-ASTB)}+20$		170	ns
$t_{PHL(A-ASTB)}$	Propagation time from address to address strobe (Note 4)	T_{ASS}	100			100			ns
$t_{h(ASTB-A)}$	Propagation time from address strobe to address hold (Note 6)	T_{AHS}	50			50			ns
$t_{PLH(\phi-ASTB)}$	Propagation time from clock to address strobe (Note 3)	T_{STL}			200			200	ns
$t_{PHL(\phi-ASTB)}$	Propagation time from clock to address strobe (Note 3)	T_{STT}			140			140	ns
$t_{w(ASTB)}$	Address strobe pulse width (Note 6)	T_{STW}	$t_{c(\phi)}-100$			$t_{c(\phi)}-100$			ns
$t_{PHL(AS-R)}$ $t_{PHL(AS-WE)}$	Propagation time from address strobe to read or extended write (Note 6)	T_{ASC}	70			70			ns
$t_{h(DQ-R)}$ $t_{h(DQ-WE)}$	Read or extended write hold time after data (Note 6)	T_{DBC}	20			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			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			200	ns
$t_{PLH(\phi-R)}$ $t_{PLH(\phi-W)}$	Propagation time from clock to read or write (Notes 4, 9)	T_{DCT}			200			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			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			150	ns
$t_{w(R)}$	Read pulse width (Note 6)	T_{RAM}	$2t_{c(\phi)}+t_{w(\phi)}-50$			$2t_{c(\phi)}+t_{w(\phi)}-50$			ns ns
$t_{w(W)}$	Write pulse width (Note 6)	T_{WRM}	$t_{c(\phi)}-50$			$t_{c(\phi)}-50$			ns
$t_{w(WE)}$	Extended write pulse width	T_{WWME}	$2t_{c(\phi)}-50$			$2t_{c(\phi)}-50$			ns

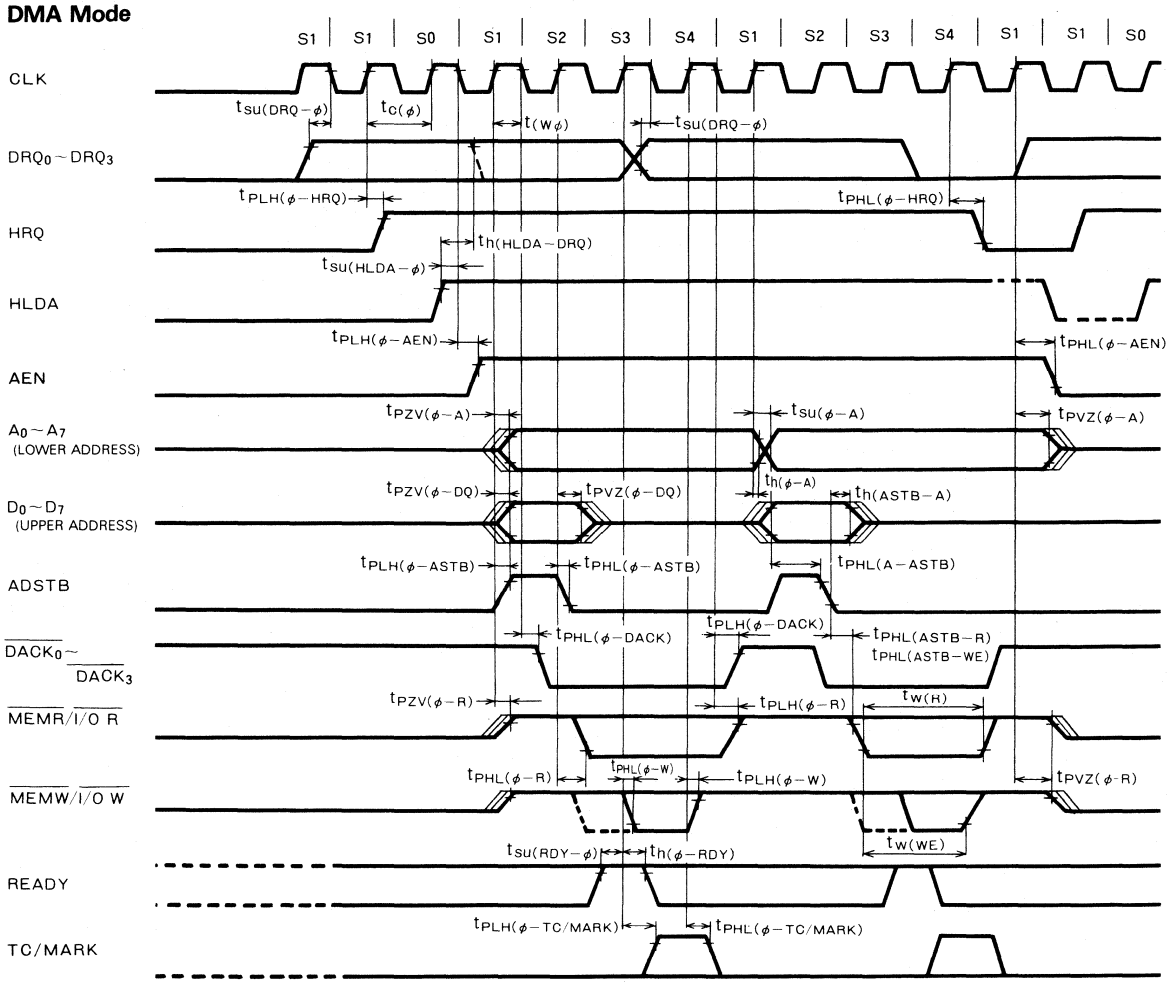
Note 2 : Reference level is V_{OH} = 3.3V.
3 : Load = 1 TTL.
4 : Load = 1 TTL + 50pF.
5 : Load = 1 TTL + (R_L = 3.3k Ω), V_{OH} = 3.3V.
Note 6 : Tracking specification
7 : $\triangle t_{PLH(\phi-DACK)} < 50ns$, $\triangle t_{PHL(\phi-TC/MARK)} < 50ns$, $\triangle t_{PLH(\phi-TC/MARK)} < 50ns$.
8 : $\triangle t_{PHL(\phi-R)} < 50ns$, $\triangle t_{PHL(\phi-W)} < 50ns$, $\triangle t_{PHL(\phi-WE)} < 50ns$.
9 : $\triangle t_{PLH(\phi-R)} < 50ns$, $\triangle t_{PLH(\phi-W)} < 50ns$.

9

PROGRAMMABLE DMA CONTROLLER

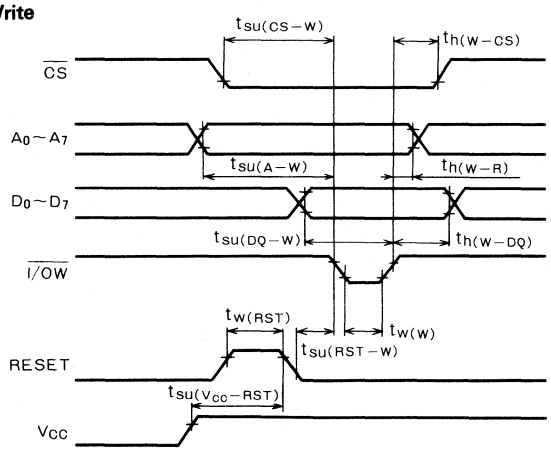
TIMING DIAGRAMS

DMA Mode

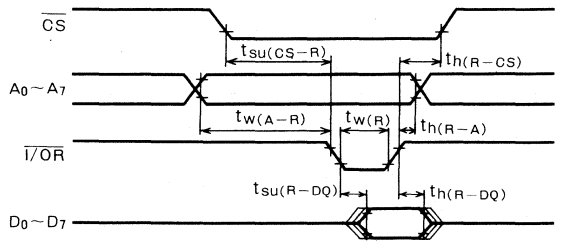


Slave Mode (Reference voltage: "H" = 2V "L" = 0.8V)

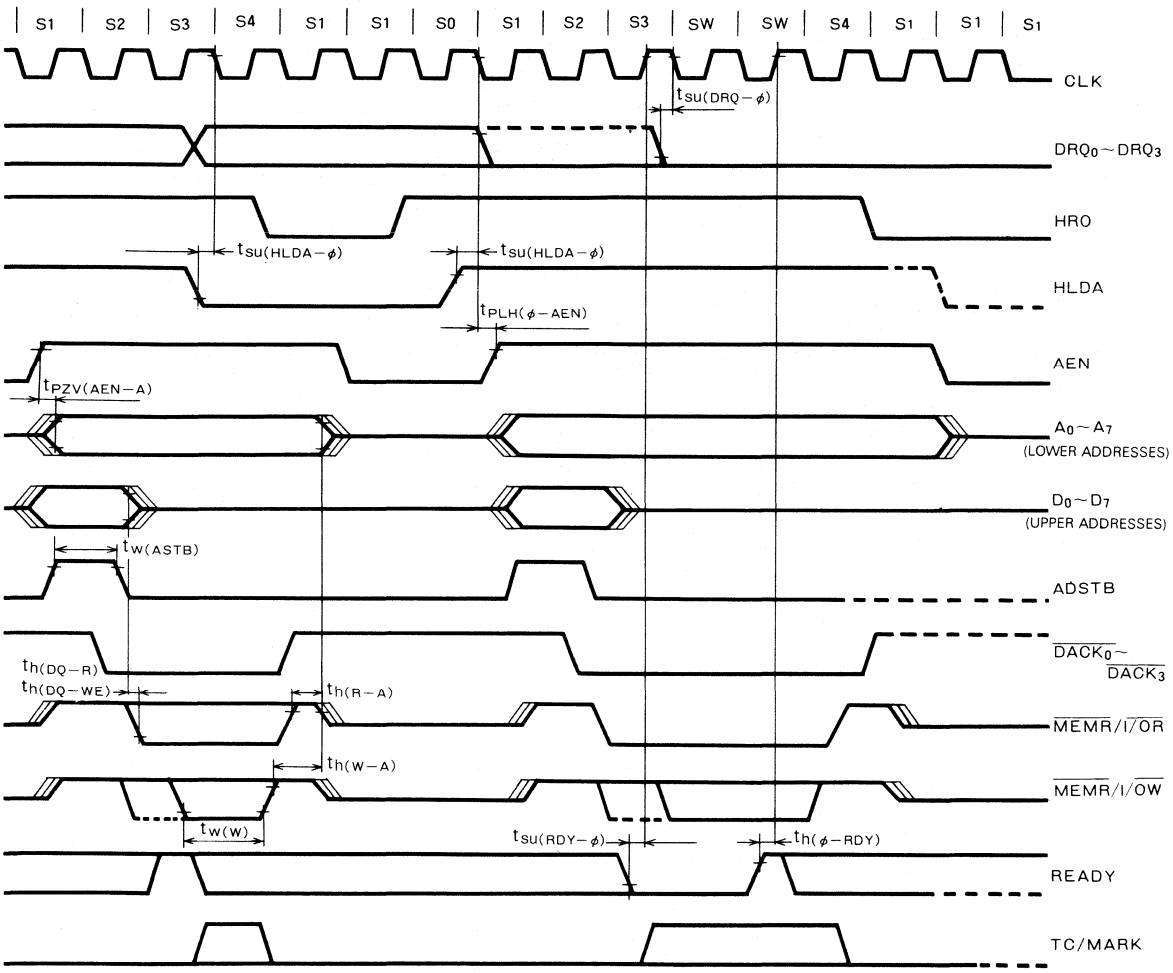
Write



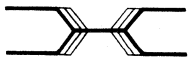
Read



MITSUBISHI ELECTRIC



Note 10 :

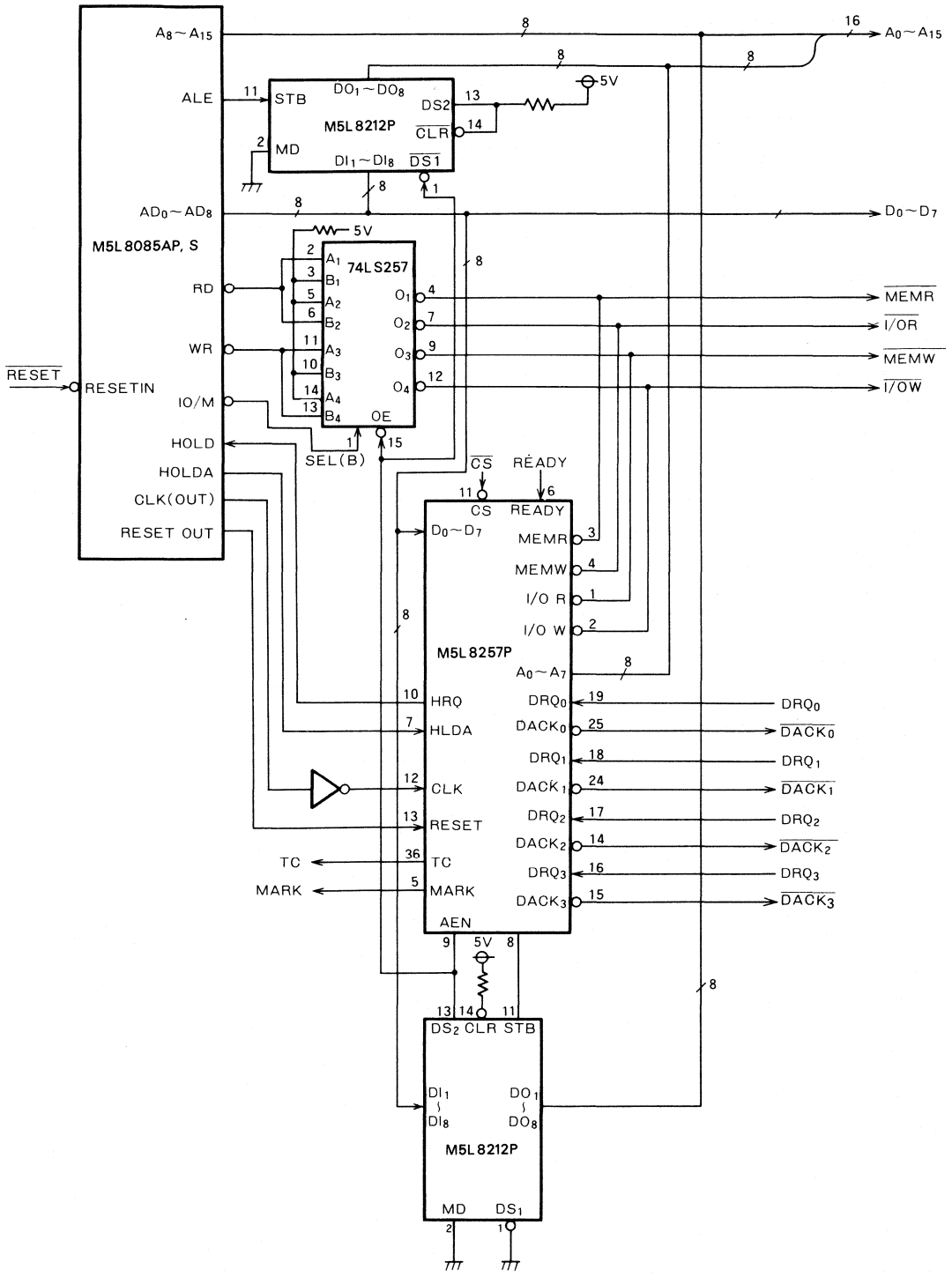


The center line indicates a floating (high-impedance) state.

9

PROGRAMMABLE DMA CONTROLLER

TYPICAL APPLICATION CIRCUIT



MITSUBISHI ELECTRIC

M5L 8279P, M5L 8279P-5

PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE

DESCRIPTION

The M5L8279P 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

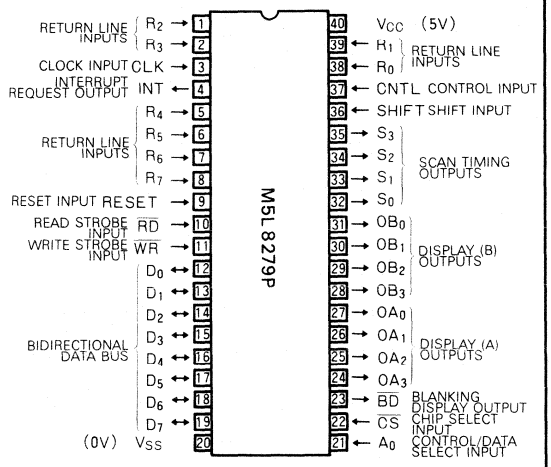
Parameter	M5L8279P	M5L 8279P-5
Output enable time after read (max)	300ns	250ns
Output enable time after address (max)	450ns	250ns
Clock cycle time (min)	500ns	320ns

- 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- or 128-contact key input device for such items as electronic cash registers
- Dual 8- or single 16-alphanumeric display

PIN CONFIGURATION (TOP VIEW)



Outline 40P1

FUNCTIONS

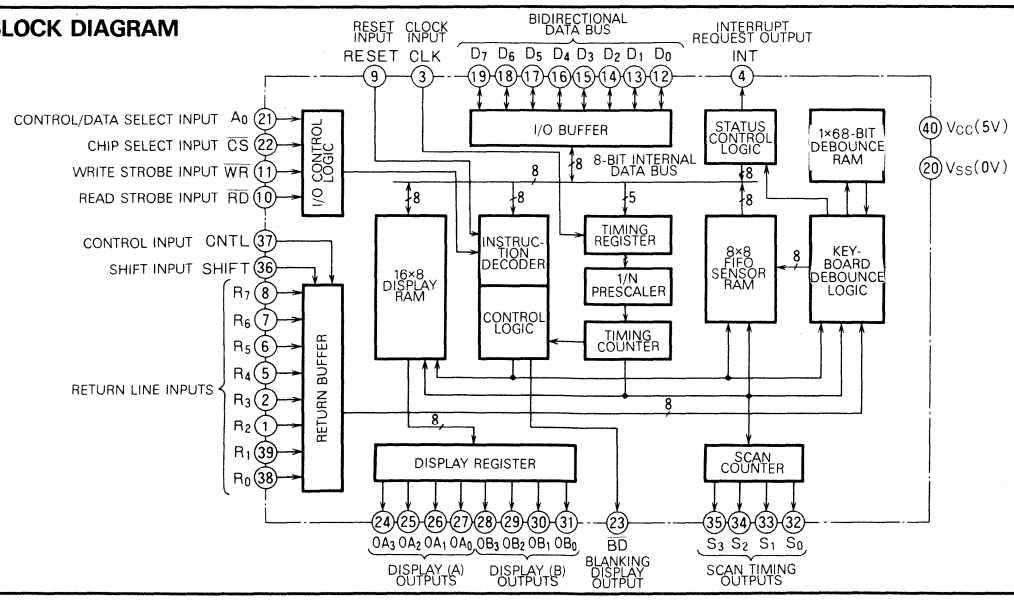
The total chip, consisting of a keyboard interface and a display interface, can be programmed by eight 8-bit commands.

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.

9

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 low-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 rollover 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

	MSB						LSB
Code :	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 rollover[1]
0 0 1 Decoded display keyboard mode — 2-key rollover
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

	MSB						LSB
Code :	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

	MSB							LSB	
Code :	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

	MSB							LSB
Code :	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

	MSB							LSB
Code :	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

	MSB							LSB	
Code :	1	0	1	X	IW	IW	BL	BL	X=Don't care
					A	B	A	B	

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

	MBS							LSB
Code :	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".

9

PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE

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

MSB LSB

Code:

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.

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.

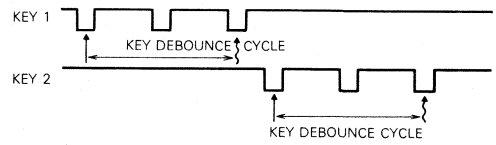
The internal key debounce logic works for a 64-key matrix that is obtained by combining the return line inputs with the scan timing.

For the keyboard interface, M5L8279P 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 Rollover (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.

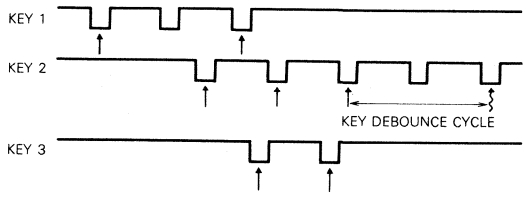
Example 1: Accepting two successive key depressions



Note 2 : ↑ : Debounce counter reset

{ : Key input

Example 2: Overlapped depression of three keys



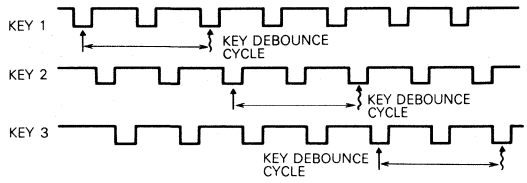
Note 3 : Only key 2 is acceptable.

9

PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE

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:

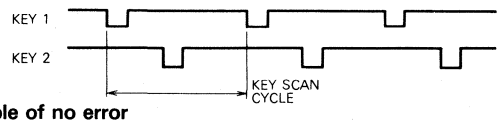


The scanned key input signal does not always reflect the actual key depressing action, as the key matrix is scanned by the timing signal.

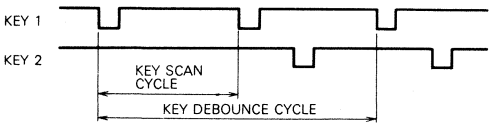
With N-key rollover, there is a mode provided with which error is caused when there are more than two key inputs in a key scan cycle, which can be programmed by using the end interrupt/error mode set command. In this mode (special error mode), recognition of the above error sets the IRQ signal to "1" and sets the bit S/E in the status word.

In case two key entries are made separately in more than a scan cycle, there would be no problem, as key depression is clearly identified. And no problem exists for 2-key rollover, as the both keys are recognized invalid.

Example of error



Example of no 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.

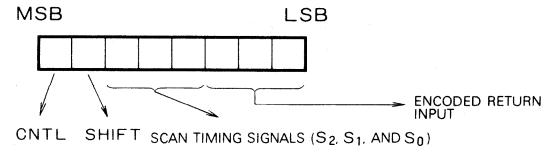
In the sensor matrix mode with the bit E = 0 of the end interrupt/error mode set command, the second most significant bit of the status word (S/E bit) is set to "1" when any sensor switch is depressed.

4. Strobe Mode

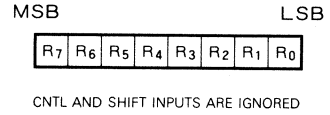
The data is entered into the FIFO from the return lines (R_0~R_7) at the rising edge of a CNTL pulse. The INT goes high while any data exists in the FIFO, in the same manner as in the keyboard mode. The key debounce circuit will not operate.

Formats of data entered into the FIFO in each of the above modes are described in the following:

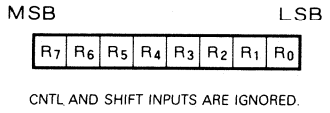
Keyboard matrix



Sensor matrix mode



Strobe mode



DISPLAY INTERFACE

The display interface is done by eight display outputs (OA_0~OA_3, OB_0~OB_3), a blanking signal ($\overline{BD}$), and scan timing outputs (S_0~S_3).

The relation between the data bus and the display outputs is as shown below:

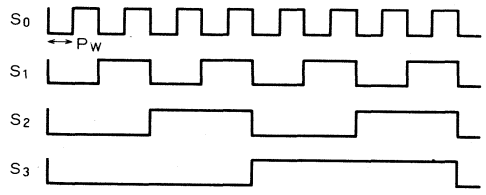
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
↓	↓	↓	↓	↓	↓	↓	↓
OA_3	OA_2	OA_1	OA_0	OB_3	OB_2	OB_1	OB_0

Clearing the display RAM is achieved by the reset signal (9-pin) but requires the execution of the clear command.

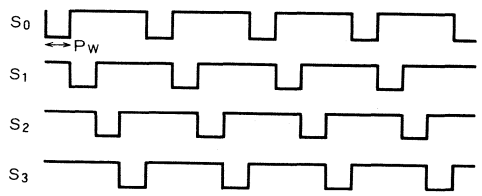
The timing diagrams for both the encoded and decoded modes are shown below.

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

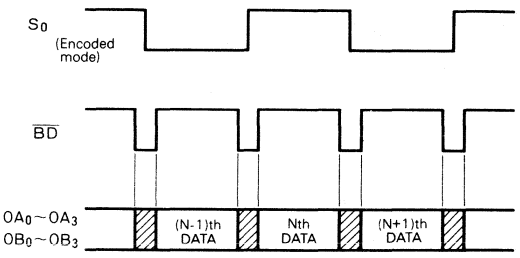


(2) Decoded mode



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~OA_3, OB_0~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~OA_3, OB_0~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.

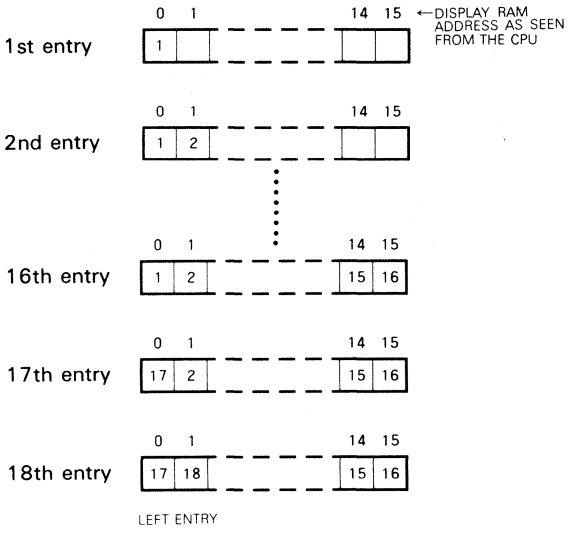
9

PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE

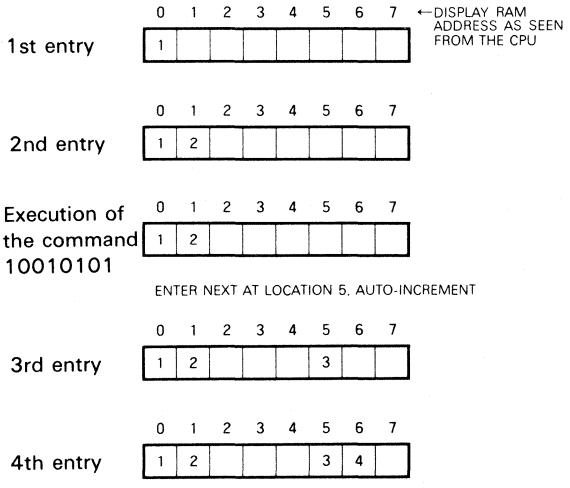
KEY ENTRY METHODS

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.

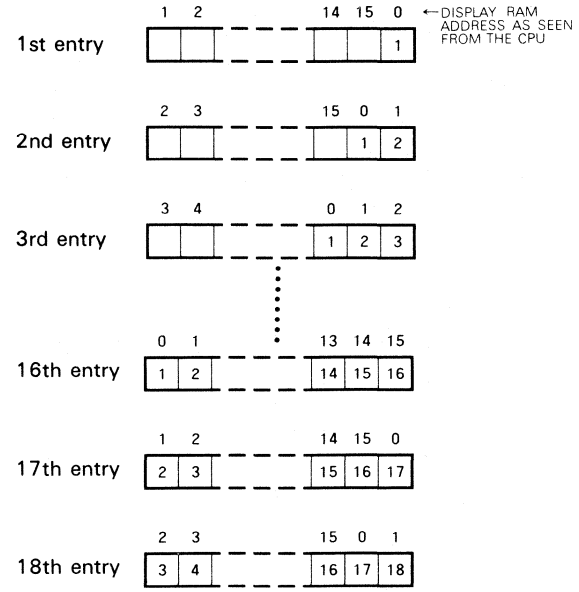


Auto-increment mode

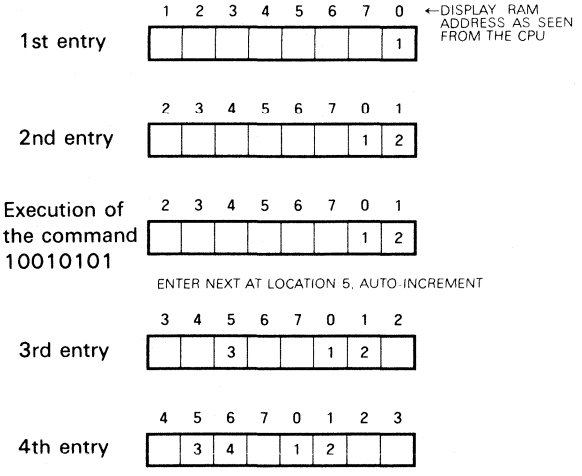


2. Right Entry

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



MITSUBISHI ELECTRIC

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		−40∼125	℃

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

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage		(Note 6)		V
V_{SS}	Supply voltage		0		V
$V_{IH(RL)}$	High-level input voltage, return line inputs, shift input and control input	2.2			V
V_{IH}	High-level input voltage, all others	2			V
$V_{IL(RL)}$	Low-level input voltage, return line inputs, shift input and control input	V_{SS}-0.5		1.4	V
V_{IL}	Low-level input voltage, all others	V_{SS}-0.5		0.8	V

Note 6 : M5L8279P, V_{CC}=5V±5% ; M5L8279P-5, V_{CC}=5V±10%

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	M5L 8279P	I_{OH}=−100μA	2.4			V
		M5L 8279P-5	I_{OH}=−400μA				
$V_{OH(INT)}$	Low-level output voltage, interrupt request output	M5L 8279P	I_{OH}=−100μA	3.5			V
		M5L 8279P-5	I_{OH}=−300μA				
V_{OL}	Low-level output voltage	M5L 8279P	I_{OL}=1.6mA			0.45	V
		M5L 8279P-5	I_{OL}=2.2mA				
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

9

PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE

TIMING REQUIREMENTS (T_a=0~70℃, V_{CC}=(Note 6), V_{SS}=0V, unless otherwise noted.)

Read Cycle

Symbol	Parameter	Alternative symbol	Test conditions	M5L8279P Limits			M5L8279P-5 Limits			Unit
				Min	Typ	Max	Min	Typ	Max	
$t_{C(R)}$	Read cycle time	t_{RCY}	(Note 7)	1000			1000			ns
$t_{W(R)}$	Read pulse width	t_{RR}		420			250			ns
$t_{SU(A-R)}$	Address setup time before RD	t_{AR}		50			0			ns
$t_{h(R-A)}$	Address setup time after RD	t_{RA}		5			0			ns

Write Cycle

Symbol	Parameter	Alternative symbol	Test conditions	M5L8279P Limits			M5L8279P-5 Limits			Unit
				Min	Typ	Max	Min	Typ	Max	
$t_{W(W)}$	Write pulse width	t_{WW}	(Note 7)	400			250			ns
$t_{SU(A-W)}$	Address setup time before WR	t_{AW}		50			0			ns
$t_{h(W-A)}$	Address hold time after WR	t_{WA}		20			0			ns
$t_{SU(DQ-W)}$	Data input setup time before WR	t_{DW}		300			150			ns
$t_{h(W-DQ)}$	Data input hold time after WR	t_{WD}		40			0			ns

Other Timings

Symbol	Parameter	Alternative symbol	Test conditions	M5L8279P Limits			M5L8279P-5 Limits			Unit
				Min	Typ	Max	Min	Typ	Max	
$t_{C(\phi)}$	Clock cycle time	t_{CY}	(Note 7)	500			320			ns
$t_{W(\phi)}$	Clock pulse width	$t_{\phi W}$		230			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 7 : 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
M5L8279P, C_L=100pF; M5L8279P-5, C_L=150pF

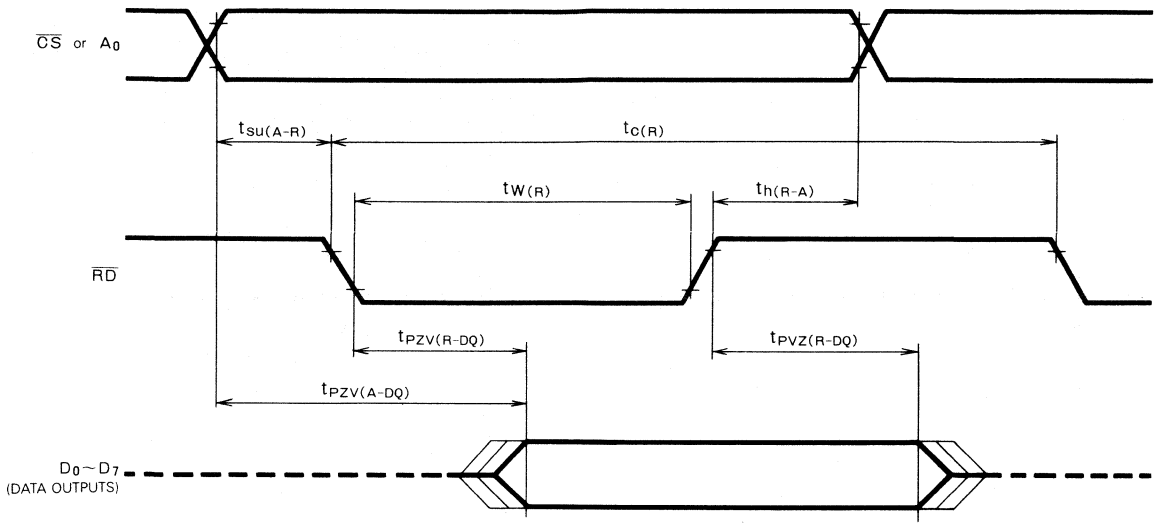
SWITCHING CHARACTERISTICS (T_a=0~70℃, V_{CC}=(Note 1), V_{SS}=0V, unless otherwise noted.)

Symbol	Parameter	Alternative symbol	Test conditions	M5L8279P Limits			M5L8279P-5 Limits			Unit
				Min	Typ	Max	Min	Typ	Max	
$t_{PZV(R-DQ)}$	Output enable time after read	t_{RD}	(Note 3)			300			250	ns
$t_{PZV(A-DQ)}$	Output enable time after address	t_{AD}				450			250	ns
$t_{PVZ(R-DQ)}$	Output disable time after read	t_{DF}		10		100	10		100	ns

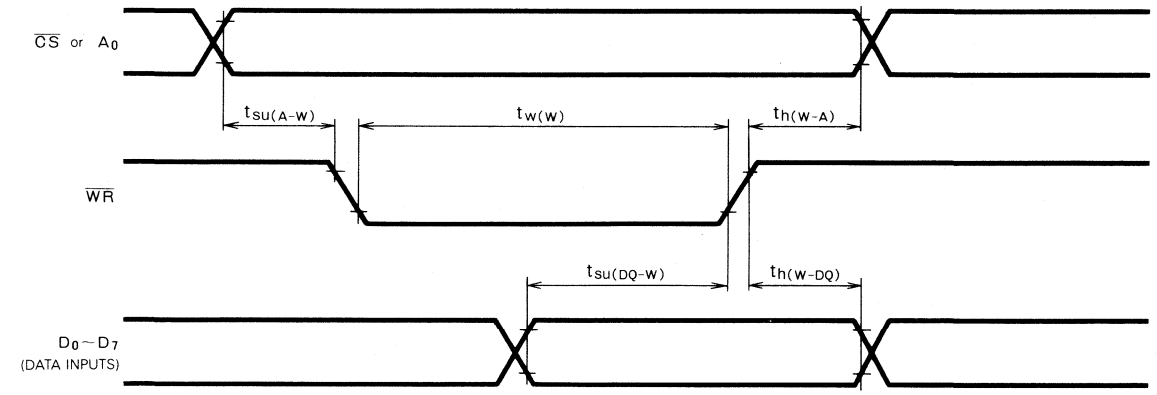
Note 8 : Test conditions
Input pulse level: 0.45~2.4V
Input pulse rise time: 20ns
Input pulse fall time: 20ns
High-level input reference voltage: 2V
Low-level input reference voltage: 0.8V
High-level output reference voltage: 2V
M5L8279P, C_L=100pF ; M5L8279P-5, C_L=150pF

TIMING DIAGRAM

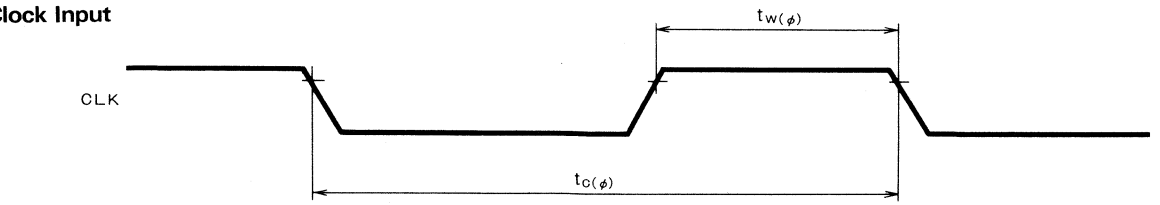
Read Mode



Write Mode



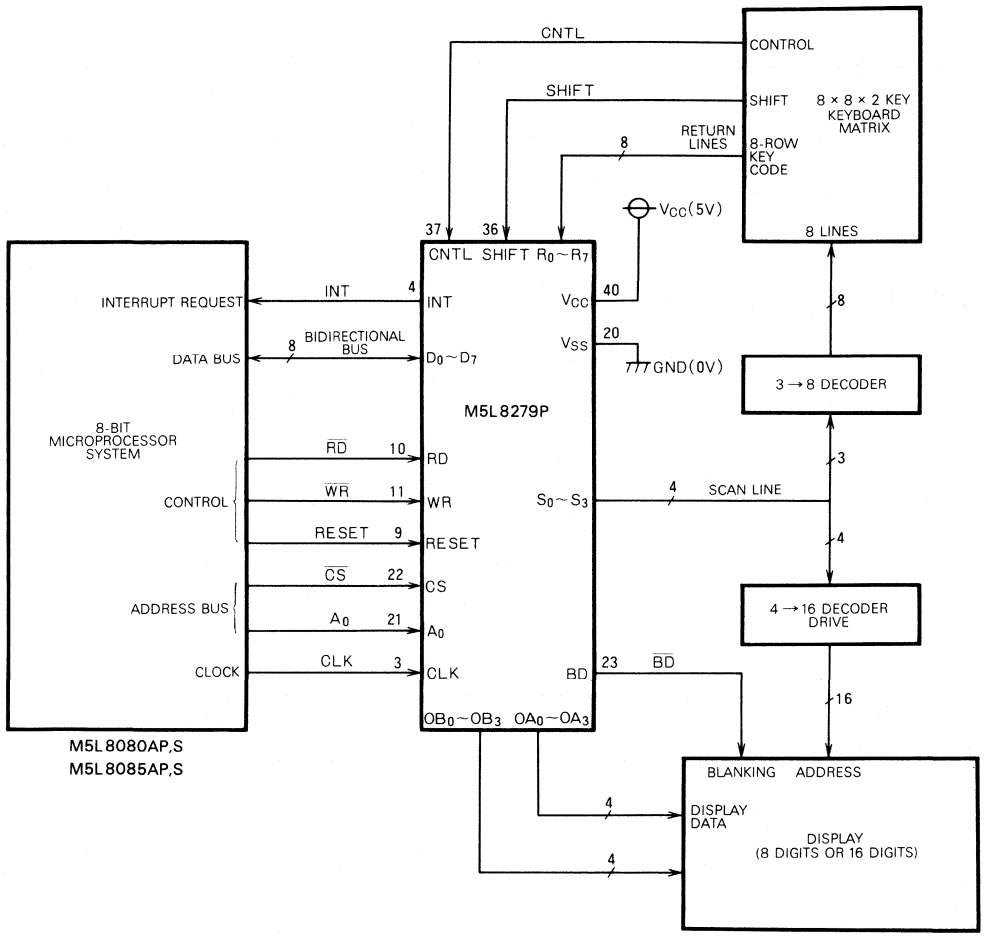
Clock Input



9

PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE

TYPICAL APPLICATION CIRCUIT



GENERAL-PURPOSE MOS LSIs

10

M58412P, M58413P

CMOS LCD DIGITAL ALARM CLOCK CIRCUITS

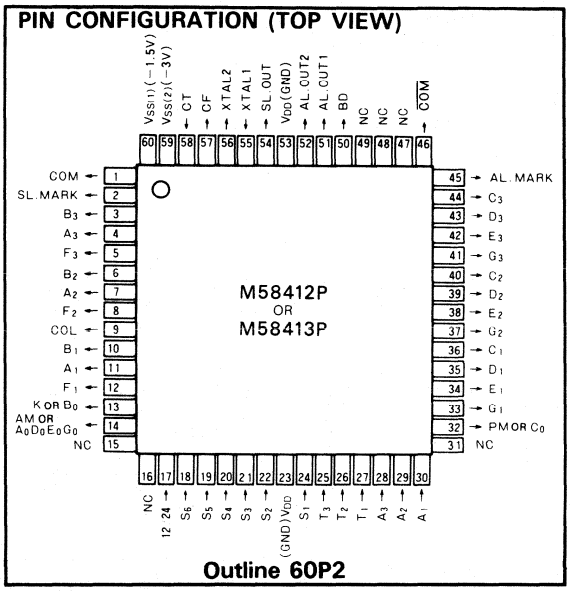
DESCRIPTION

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

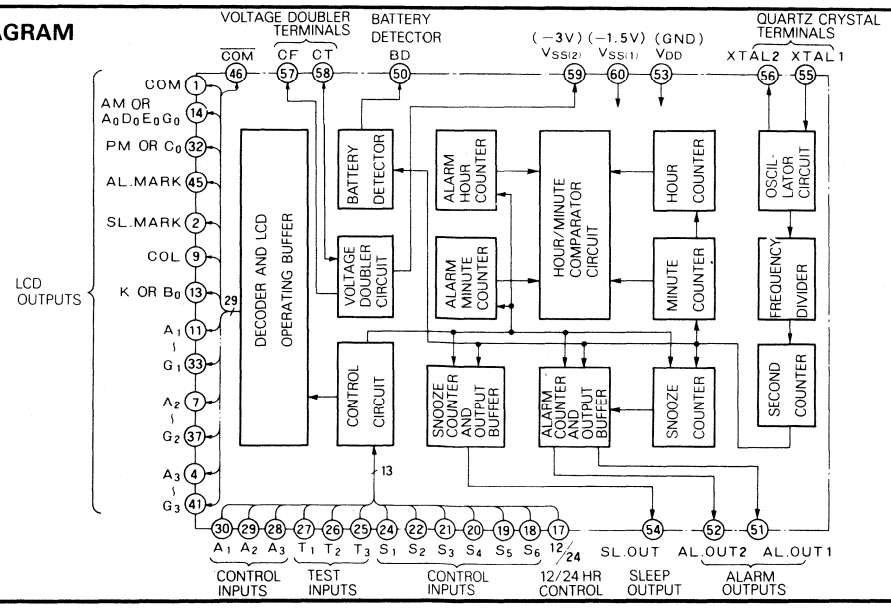
APPLICATIONS

- Alarm clocks with a 'snooze' function
- Sleep timers
- Travel watches
- Switching timers for electronic apparatus
- Auto-recording timers for audio equipment

FUNCTIONS

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



CMOS LCD DIGITAL ALARM CLOCK CIRCUITS

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)

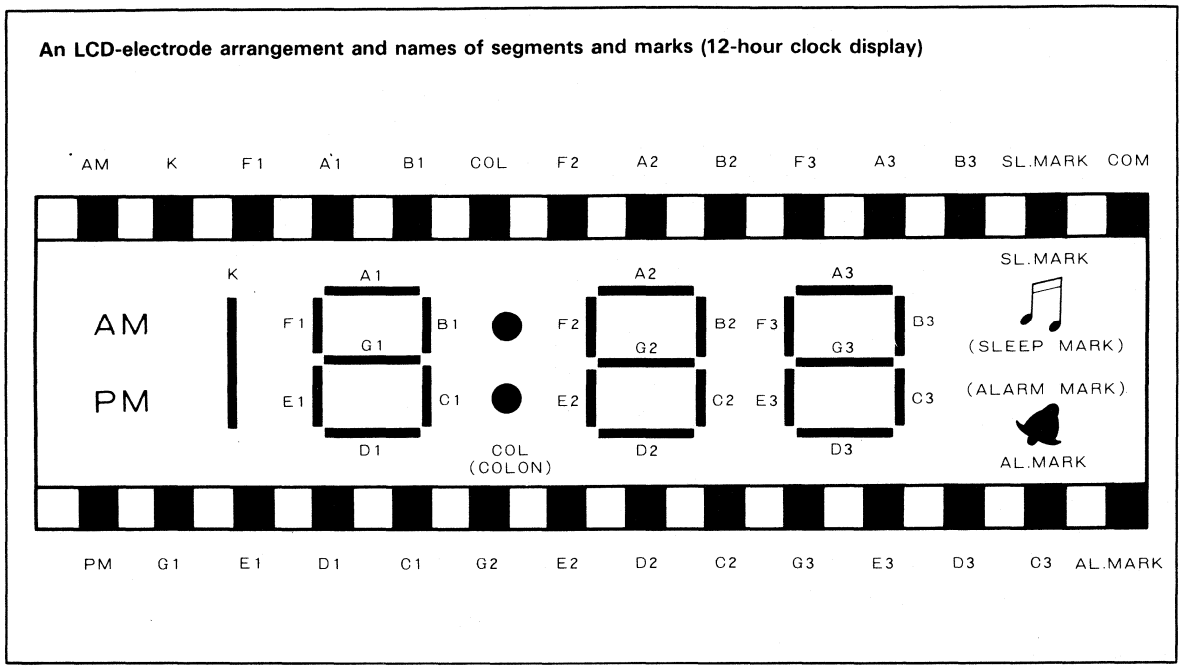


Table 1 12-Hour Clock Display

Mode	Display	Meaning of mark display
Normal clock ordinary display	PM 12:56	Sleep timer is in operation. Alarm timer is being set.
Alarm	AM 7:00	Alarm time is displayed; adjustment possible.
Sleep-time	30	Sleep time is displayed; adjustment possible. Alarm timer is being set.

Note 1. The symbol ✳ indicates a 2sec on-off flash.

An LCD-electrode arrangement and names of segments and marks (24-hour clock display)

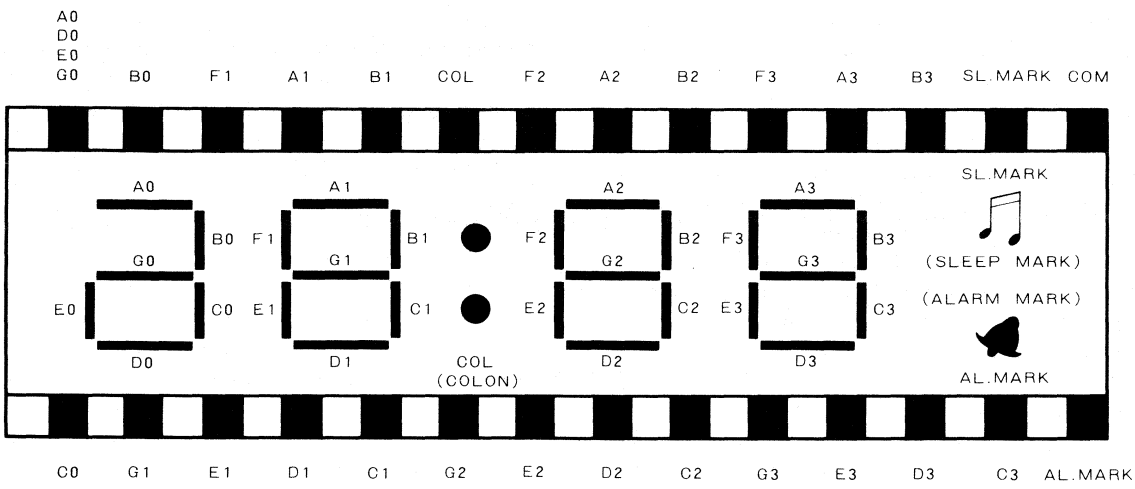


Table 2 24-Hour Clock Display

Mode	Display	Meaning of mark display
Normal display	20:56	Sleep timer is in operation. Alarm timer is being set.
Alarm display	7:00	Alarm time is displayed; adjustment possible.
Sleep-time display	30	Sleep time is displayed; adjustment possible. Alarm timer is being set.

Note 2 : The symbol ⁜ indicates a 2sec on-off flash.

10

CMOS LCD DIGITAL ALARM CLOCK CIRCUITS

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

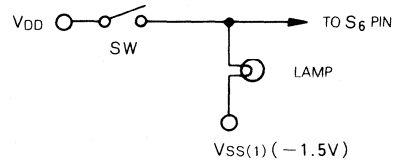
This pin is used to provide alarm-timer set input. Maintaining this input pin at the V_{DD} level causes the alarm mark to stay on. When the normal clock time coincides with the alarm time, two types of alarm output, AL. OUT1 and AL. OUT2, generate alarm signals. (The alarm signal with a pulse width 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	Operation of bells, buzzers and other sound sources without oscillator circuits (intermittent sound)
V_{DD}	N.C.	V_{DD} / $V_{SS(1)}$: 1s, 1s	Operation of sound sources with oscillator circuits (intermittent sound)
N.C.	V_{DD}	V_{DD} / $V_{SS(1)}$: 1024Hz	Operation of sound sources without oscillation circuits (continuous sound)
V_{DD}	V_{DD}	V_{DD} / $V_{SS(1)}$: 250ms	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 'AM 12:00' (0:00 for 24-hour cycle) is displayed in the normal-clock ordinary-display mode. Here, the alarm time is AM 12:00 (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, $\overline{\text{COM}}$ 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 $\overline{\text{COM}}$ output is used for permanently-displayed segments or marks.

AL. OUT1 (Alarm output 1) Pin

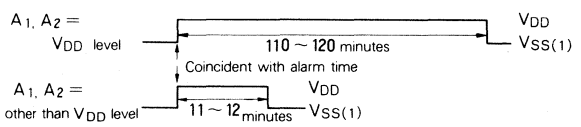
When the normal-clock ordinary-display time coincides with the alarm time, alarm signals with the waveforms shown in Table 3 are generated at the AL. OUT1 pin for 12 minutes. The 250 ms pulse-width alarm output, however, is given only once after the coincidence.

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



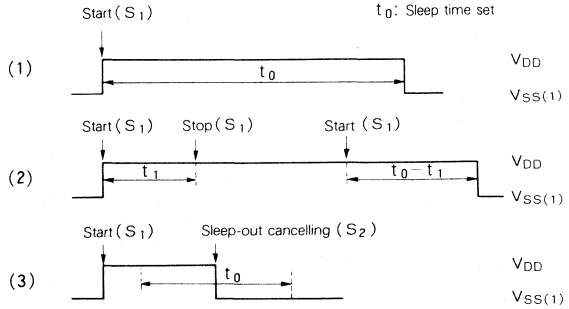
10

CMOS LCD DIGITAL ALARM CLOCK CIRCUITS

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

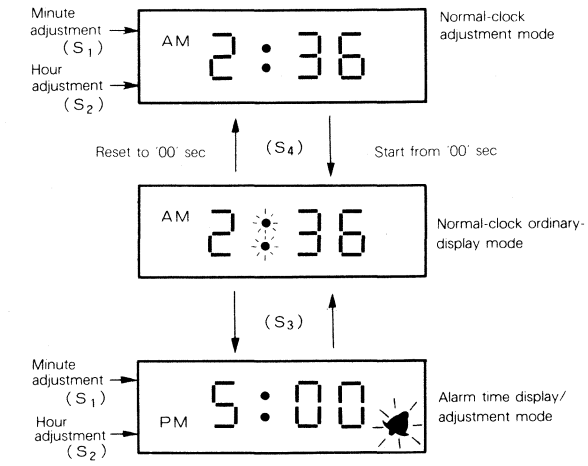
The electrical power supply is a 1.5V battery ($=V_{DD}-V_{SS(1)}$). Use of 0.1μF condensers between the CF and CT pins and between the $V_{SS(2)}$ and $V_{DD(GND)}$ pins gives voltage about double the power voltage, making possible direct operation of the LCD.

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 15kΩ 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

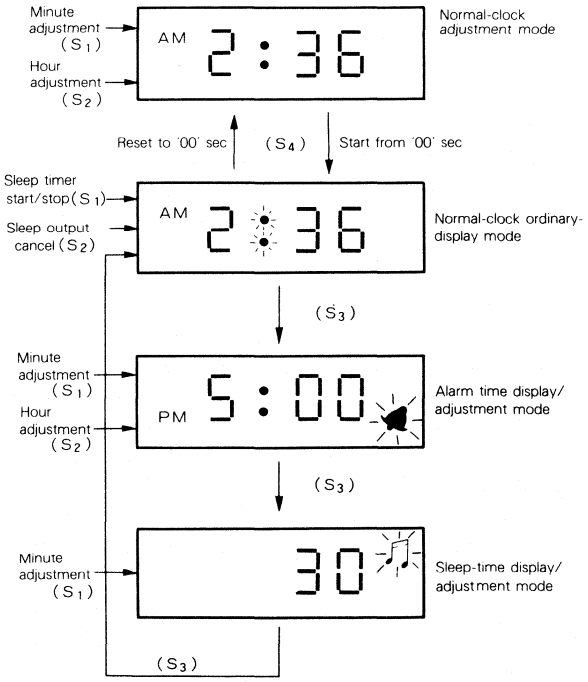
Fig. 4 Operation when the A_3 Pin is N.C.



Note 3 : The symbol ☼ shows a 2sec-period on-off flash.

4 : 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 5 : The symbol ☼ shows a 2sec-period on-off flash.

6 : 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, except where otherwise specified)

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 = 30kΩ	−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 = 30kΩ	−2.2	−3	−4	V

ELECTRICAL CHARACTERISTICS

(Ta = 25°C, V_{DD} = GND, M58412P : f = 4.1943MHz, M58413P : f = 32.768kHz, except where otherwise specified)

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 = 30kΩ		2	5	μA
$V_{I(OSC)}$	Oscillator input voltage	M58412P	C_{IN} = 15pF, C_{OUT} = 10pF, R_O = 20Ω within 1sec of oscillation			−1.2	V
		M58413P	C_{IN} = 15pF, C_{OUT} = 30pF, R_O = 30kΩ 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		15kΩ ≦ R_{BD} ≦ 750kΩ	−1.2		−1.5	V

Note 7 : R_O refers to a crystal impedance.

10

CMOS LCD DIGITAL ALARM CLOCK CIRCUITS

TYPICAL APPLICATION CIRCUITS

Fig. 6 An alarm clock with 'snooze' and sleep functions

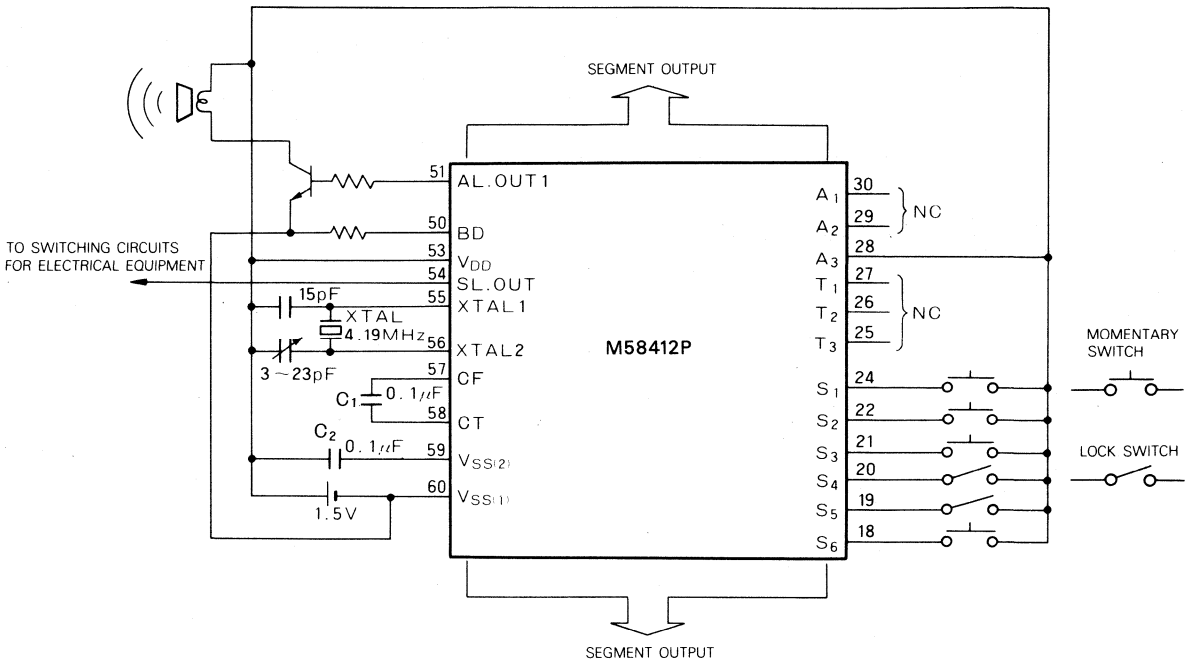
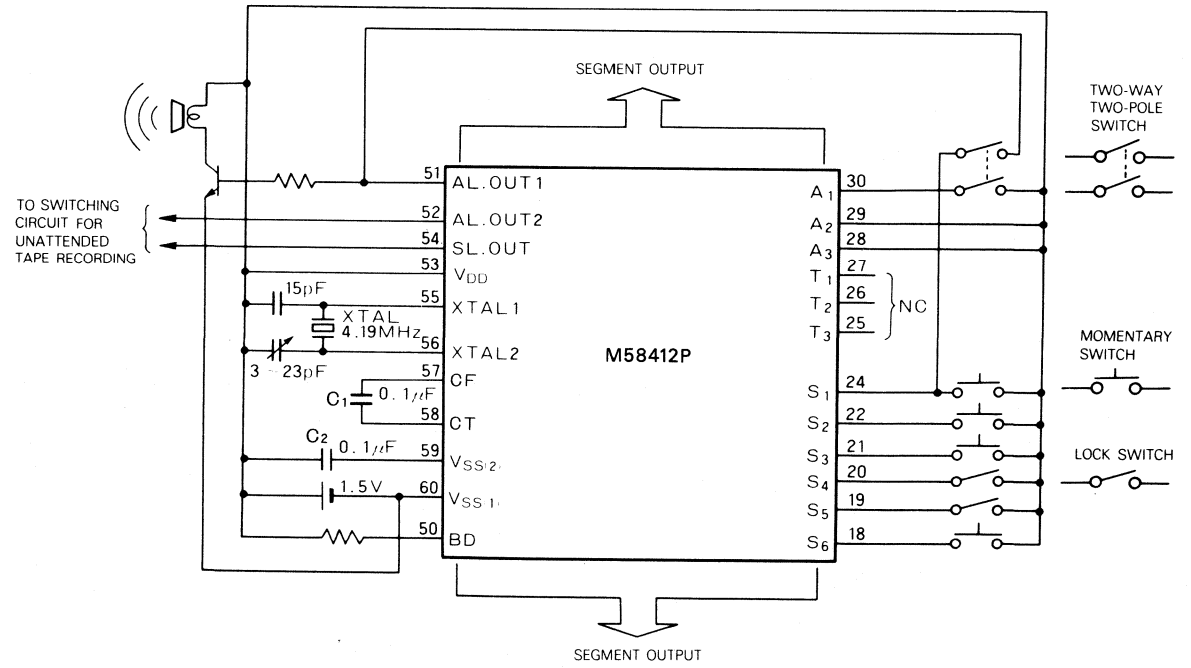


Fig. 7 An alarm clock with 'snooze' and auto-recording functions



Note 8 : The circuit of Fig. 6 gives intermittent alarm-bell tones.

9 : The circuit of Fig. 7 gives continuous alarm-bell tones.

10 : 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 11 : 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

MITSUBISHI ELECTRIC

M58434P, M58435P
M58436-001P, 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
M58434P	Silicon-gate CMOS	4.1943MHz	Induction motor	1024Hz
M58435P	Silicon-gate CMOS	4.1943MHz	Stepping motor	1024Hz
M58436 −001P	Aluminum-gate CMOS	4.1943MHz	Stepping motor	4096×8×1Hz
M58437 −001P	Aluminum-gate CMOS	32.768KHz	Stepping motor	4096×8×1Hz

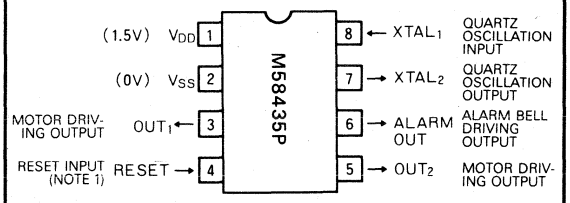
FEATURES

- Low power dissipation:
  - M58434P, M58435P: 30μA (typ)
  - M58436-001P: 35μA (typ)
  - M58437-001P: 2μA (typ)
- Low voltage operation:
  - M58434P, M58435P: 1.2V (min)
  - M58436-001P: 1.1V (min)
- Direct drive of ceramic resonator (M58436-001P and M58437-001P only)

APPLICATIONS

- Crystal-controlled alarm clock
- Precision timepiece for electronic apparatus
- Frequency divider for electronic apparatus

PIN CONFIGURATION (TOP VIEW)



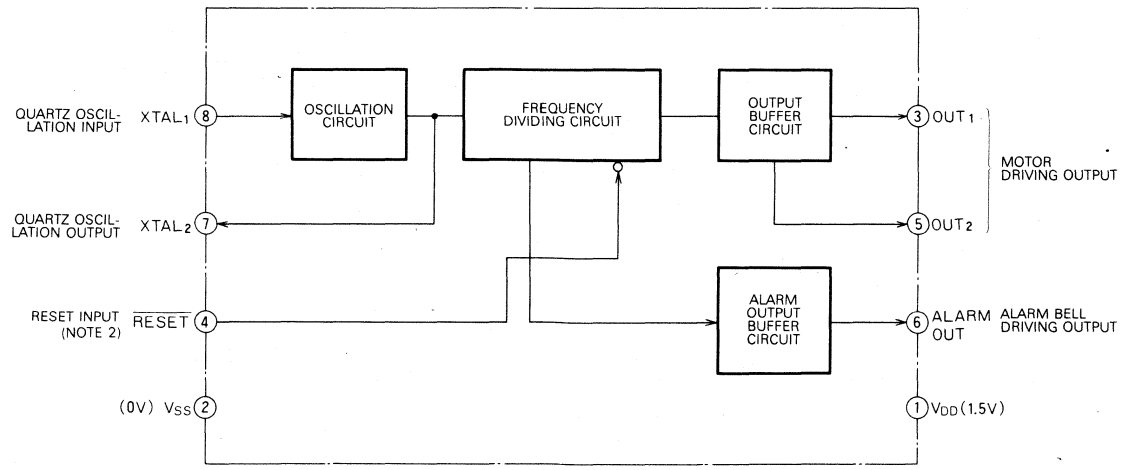
Note 1 : This pin is non-connected for M58434P.

Outline 8P1 (M58434P)
(M58435P)
(M58436-001P)
(M58437-001P)

FUNCTION

Circuitry consists of an oscillator, frequency divider, bridge-type driver circuit for an induction motor (M58434P) or a stepping motor (M58435P, M58436-001P, M58437-001P), and an alarm bell driver circuit. The oscillator frequency is 32.768 kHz for the M58437-001P and 4.1943 MHz for the other types.

BLOCK DIAGRAM



Note 2 : M58434P has no reset input pin.

10

M58434P, M58435P
M58436-001P, M58437-001P

CMOS ANALOG CLOCK CIRCUITS

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 M58434P and M58435P, Outputs OUT_1 and OUT_2 are always in a mutually reversed phase, while in the M58436-001P and 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 (M58436-001P and 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.

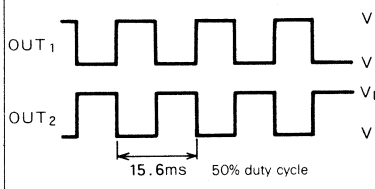
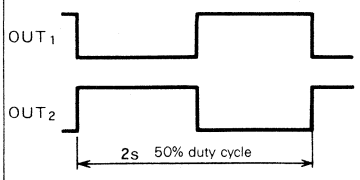
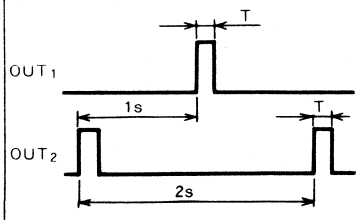
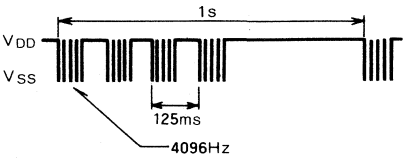
Reset Input ($\overline{\text{RESET}}$)

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 M58436-001P and M58437-001P. Direct drive of the ceramic resonator by M58436-001P and 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
M58434P		———	
M58435P		———	
M58436-001P 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 (T_a=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
f_{OSC}	Crystal oscilation frequency	M58434P M58435P M58436-001P		4.1943		-MHz
		M58437-001P		32.768		kHz
R_O	Crystal impedance of crystal oscillator	M58434P M58435P M58436-001P		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}= 0 V, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
V_{DD}	Supply voltage	M58434P M58435P	$C_{IN}=C_{OUT}$=20pF, R_O=30Ω	1.2	1.5	1.9	V
		M58436-001P	$C_{IN}=C_{OUT}$=20pF, R_O=30Ω	1.1	1.5	1.9	V
		M58437-001P	$C_{IN}=C_{OUT}$=20pF, R_O=20kΩ				
I_{DD}	Supply current	M58434P M58435P	V_{DD}=1.5V, $C_{IN}=C_{OUT}$=20pF, R_O=30Ω		30	50	μA
		M58436-001P	V_{DD}=1.5V, $C_{IN}=C_{OUT}$=20pF, R_O=30Ω		35	70	μA
		M58437-001P	V_{DD}=1.5V, $C_{IN}=C_{OUT}$=20pF, R_O=20kΩ		2	5	μA
$R_{ON(P+N)}$	Motor driving output saturation resistance (P-channel + N-channel)	M58434P M58435P	V_{DD}=1.5V, I_{OUT}=± 3 mA		150	300	Ω
		M58436-001P M58437-001P	V_{DD}=1.5V, I_{OUT}=± 3 mA		100	200	Ω
$R_{ON(AL)}$	Alarm bell driving output saturation resistance (N-channel)	M58434P M58435P	V_{DD}=1.5V, I_{OUT}= 3 mA		0.5	1	kΩ
		M58436-001P M58437-001P	V_{DD}=1.5V, I_{OUT}= 3 mA		100	200	Ω
I_{SW}	Reset input current	M58435P M58436-001P M58437-001P	V_{DD}=1.5V,			1	μA

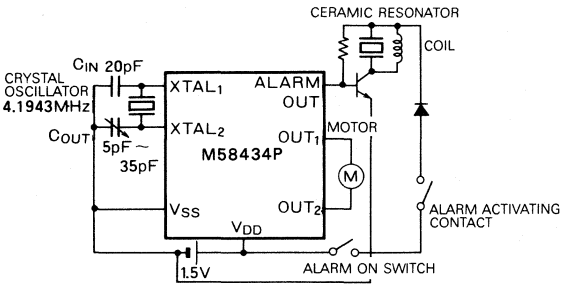
10

M58434P, M58435P
M58436-001P, M58437-001P

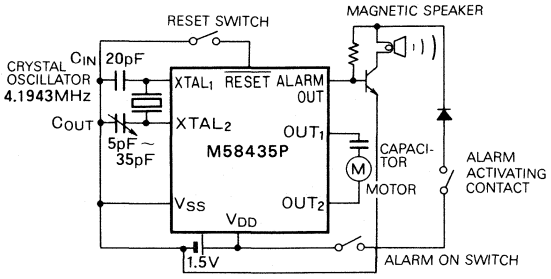
CMOS ANALOG CLOCK CIRCUITS

TYPICAL APPLICATION CIRCUITS

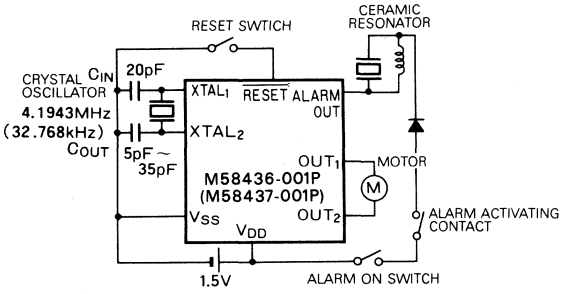
1. Ceramic resonator with M58434P



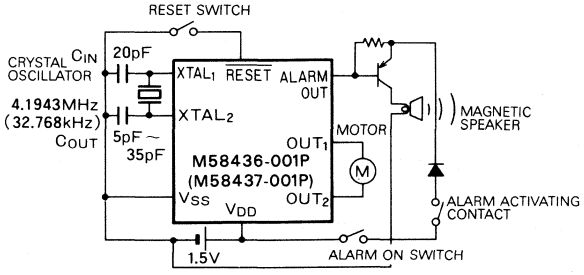
2. Magnetic speaker with M58435P



3. Ceramic buzzer with M58436-001P or M58437-001P



4. Magnetic speaker with M58436-001P or M58437-001P



17-STAGE OSCILLATOR/DIVIDER

GENERAL DESCRIPTION

The M58478P is a frequency divider manufactured by aluminum CMOS technology. It produces a frequency of 1/59719 or 1/88672 of the input frequency (3.58MHz to 60Hz or 4.43MHz to 50Hz).

FEATURES

- Makes possible a crystal oscillator circuit
- Capable of handling small-amplitude input signals as low as $0.3V_{PP}$
- Frequency-dividing ratio selected through terminal 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

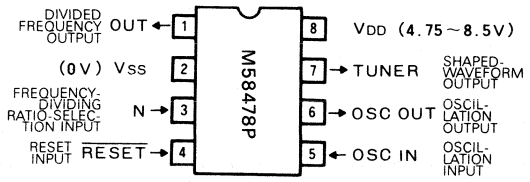
APPLICATIONS

- Frequency divider for VTR

SUMMARY OF FUNCTIONS

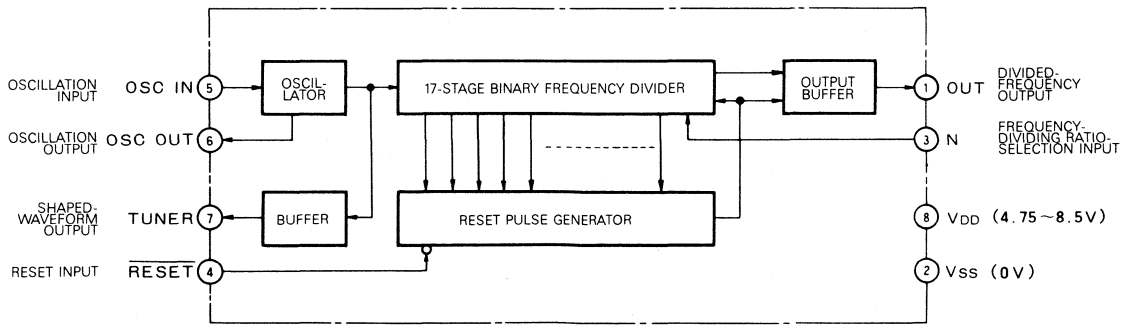
The M58478P has a programmable counter consisting of a 17-stage binary frequency divider, which delivers 60Hz or 50Hz output from an input of 3.58MHz or 4.43MHz, depending on the state of terminal N.

PIN CONFIGURATION (TOP VIEW)



Outline 8P1

BLOCK DIAGRAM



10

17-STAGE OSCILLATOR/DIVIDER

FUNCTIONAL DESCRIPTION

Crystal Oscillator

A crystal oscillator is obtained by connecting a quartz resonator element between terminals OSC IN and OSC OUT, and capacitances C_{LI} and C_{LO} between the two terminals and V_{SS}. (A feedback resistor is contained on-chip.) An amplifier is contained in terminal OSC IN, allowing even a small amplitude signal to be input via the coupling capacitor C_C.

Output Frequency

When the input N is open (high), the frequency-dividing ratio is set at 59719, producing a 60Hz output from a 3.58MHz input. When N is low, the ratio is 88672, producing 50Hz from 4.43MHz input. Figure 1 shows the waveform of the divided frequency output.

Shaped-waveform output of the same frequency as the input signal or oscillation frequency is produced on the TUNER output.

Reset Function

When the RESET input is changed from high to low (edge trigger reset of the active low), the output OUT turns low.

Pull-up Resistance

There is a pull-up resistor in both terminals N and $\overline{\text{RESET}}$, eliminating the need for any external resistor. Resistance of the standard pull-up resistor is 20kΩ.

Frequency Dividing Ratio

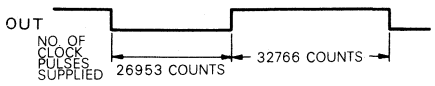
The frequency-dividing ratio is determined by the data contents of the programmable counter.

Change of the Frequency-Dividing Ratio

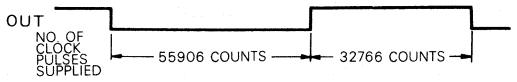
Different frequency-dividing ratios are available. Any frequency-dividing ratio from 5 to 131071 ($=2^{17}-1$) can be provided by changing the data input to the programmable counter.

Fig. 1 Waveform of divided-frequency output

When input N is open (high):



When input N is low:



Note 1 : The frequency-dividing ratio in the following cycle is decided by the state input N just before the output OUT turns from high to low.

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	Maximum 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 (Ta = −30 ~ 70°C, unless otherwise noted.)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
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 the terminal N in high-level		3.58	5.5	MHz
	Input frequency, with the terminal N in low-level		4.43	5.5	MHz

ELECTRICAL CHARACTERISTICS (Ta = 25°C, V_{DD} = 6.5V, V_{SS} = 0V, f_{IN} = 4.5MHz, unless otherwise noted.)

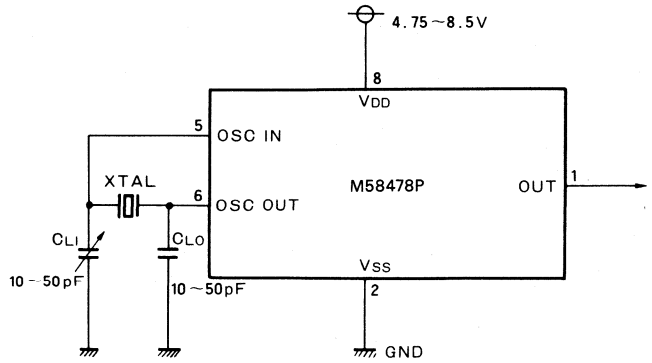
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{DD}	Operational supply voltage	Ta = −30 ~ 70°C	4.75		8.5	V
I_{DD}	Supply current	N, $\overline{RESET}$ Input/output open			5	mA
V_{IH}	High-level input voltage		V_{DD}-0.5			V
V_{IL}	Low-level input voltage				0.5	V
V_{OH}	High-level output voltage		V_{DD}-0.5			V
V_{OL}	Low-level output voltage				0.5	V
I_{OH}	High-level output current	$V_D = V_{SS}$	−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			Vpp
f_{MAX}	Maximum operating frequency	V_{DD} = 4.75V	5.5			MHz

10

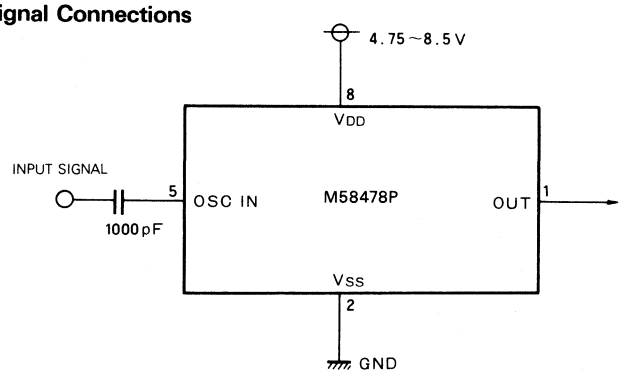
17-STAGE OSCILLATOR/DIVIDER

TYPICAL APPLICATION CIRCUIT

Crystal Oscillator (with Built-In Feedback Resistance)



External Input Signal Connections



GENERAL 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.

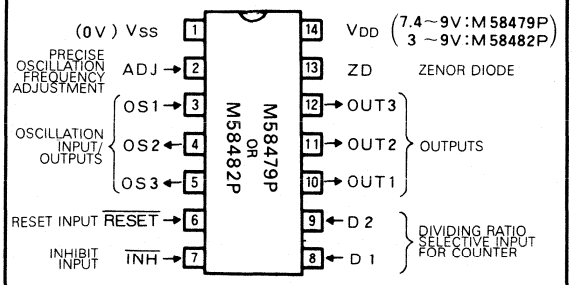
FEATURES

- 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)



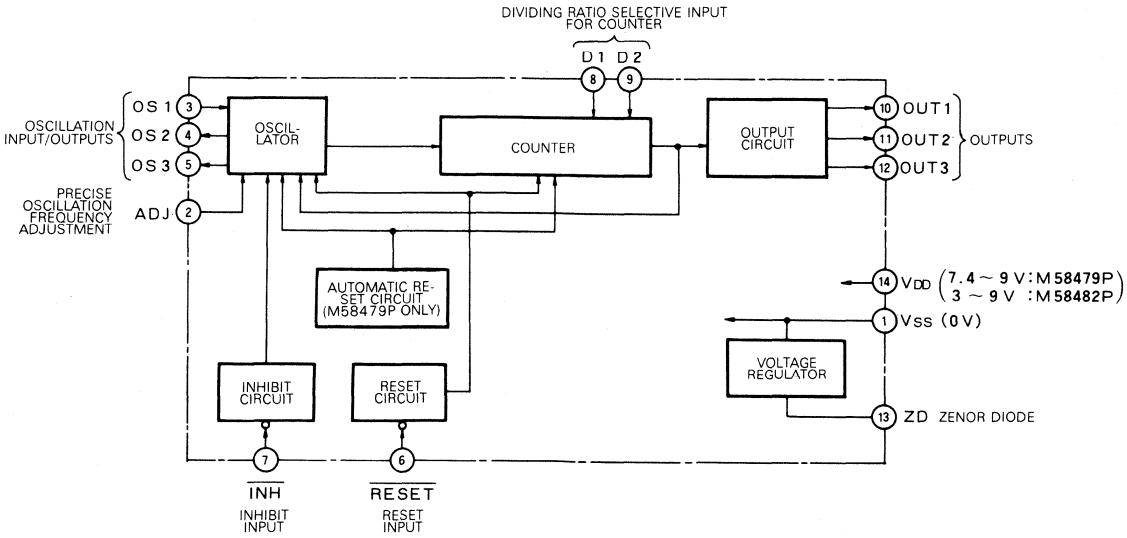
Outline 14P4

SUMMARY OF FUNCTIONS

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



10

CMOS COUNTER/TIMERS

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 OS2 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:

$$T_0 = -R_{FC} \cdot C_{FC} \left\{ \left| n \frac{V_{TR}}{V_{DD}+V_{BE}} + \left| n \frac{V_{DD}-V_{TR}}{V_{DD}+V_{BE}} \right\} \cdots (1) \right.$$

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 $\overline{RESET}$ and V_{SS}.

Reset Function

When the $\overline{RESET}$ input turns low (V_{SS}), oscillation of the oscillator can be stopped and the counter reset.

Inhibit Function

When terminal $\overline{INH}$ turns low (V_{SS}) while the timer is in action, the oscillation halts. When input $\overline{INH}$ 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	$1024 \times 10 \times 6$	$T_1 \times 10 \times 6$	1 h
L	L	$1024 \times 10 \times 6 \times 10$	$T_1 \times 10 \times 6 \times 10$	10 h

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		$V_{SS} \leqq V_I \leqq V_{DD}$	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_{ADJ}	Resistance for fine-adjustment of oscillation frequency		0		100	kΩ

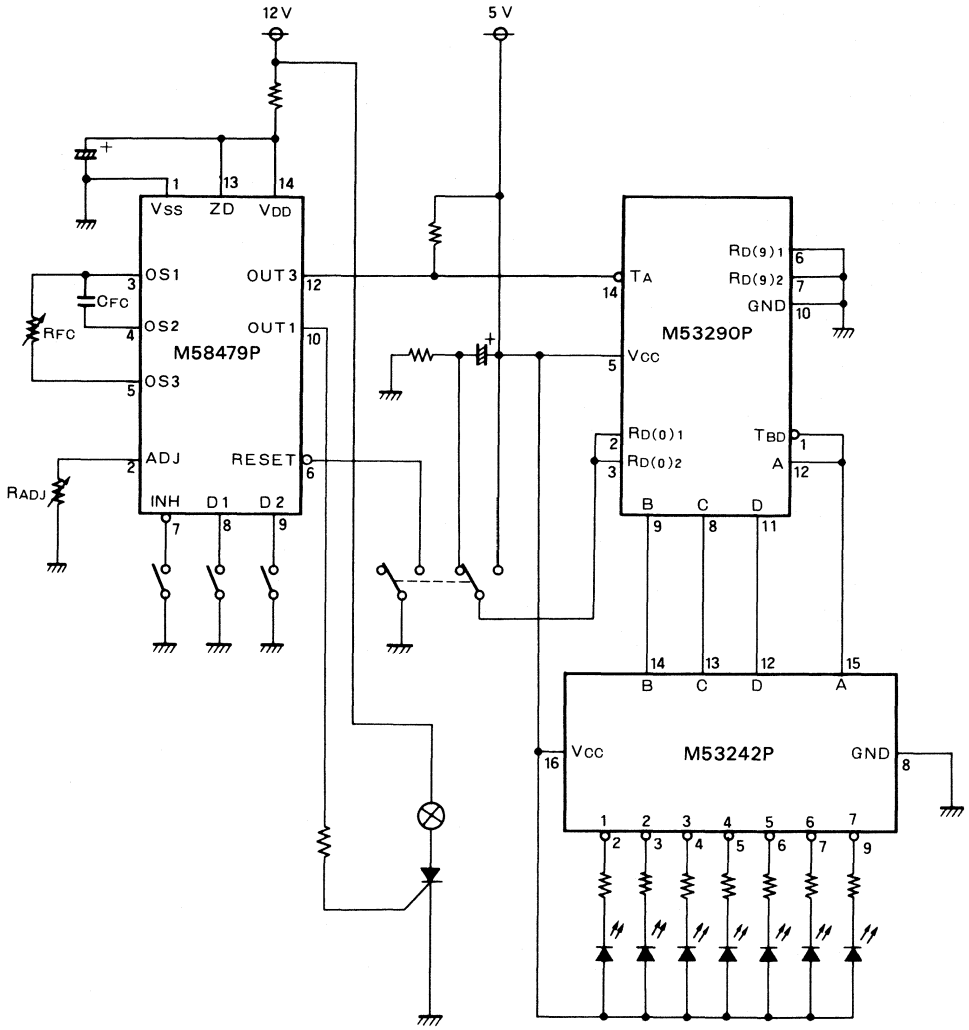
ELECTRICAL CHARACTERISTICS (Ta = 25°C, unless otherwise noted.)

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, V_O = 0V	5	10		mA
I_{OL}	Low-level output current, OUT1, OUT2, and OUT3 outputs		V_{DD} = 7.5V, V_O = 7.5V	10	20		mA
I_{OZH}	Off-state output current, OUT3 output		V_{DD} = 7.5V, V_O = 7.5V			1	μA
I_{OL}	Low-level output current; OUT1, OUT2, and OUT3 outputs		V_{DD} = 7.5V, V_O = 0.4V	1.6			mA
I_{OL}	Low-level output current; OUT1, OUT2, and OUT3 outputs	M58482P	V_{DD} = 4.5V, V_O = 0.4V	1.6			mA
V_{OL}	Low-level output voltage, OUT1, OUT2, and OUT3 outputs		V_{DD} = 7.5V			0.1	V

10

CMOS COUNTER/TIMERS

TYPICAL APPLICATION CIRCUIT



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.

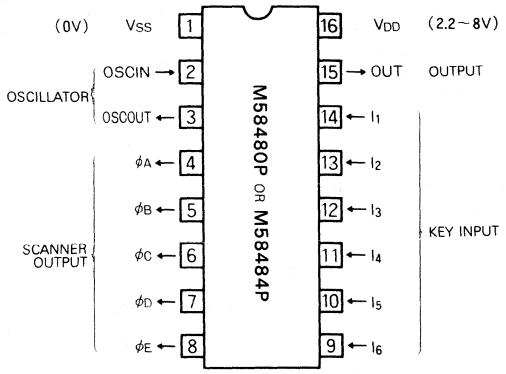
FEATURES

- 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)

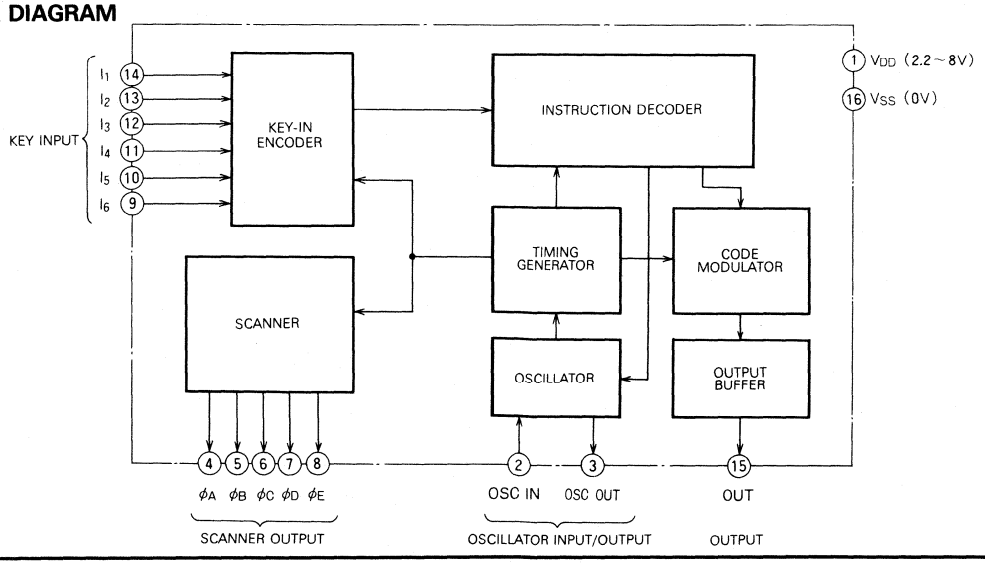


Outline 16P4

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



10

30-FUNCTION REMOTE-CONTROL TRANSMITTERS

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)

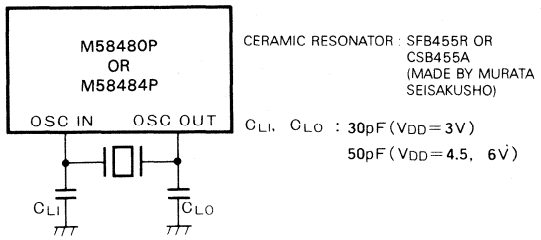
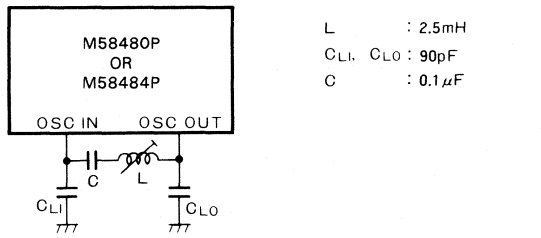


Fig. 2 An example of an oscillator (using an LC network)



Setting the oscillation frequency to 480 kHz (or 455 kHz) 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 $\phi A \sim \phi E$.

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	ϕE	ϕD	ϕC	ϕB	ϕA
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

Transmission code						Function	Remarks
D_1	D_2	D_3	D_4	D_5	D_6		
1	0	0	0	0	0	CH UP	Analog control
0	1	0	0	0	0	CH 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	
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)	Normalization of analog
1	1	0	1	0	0	BR(1/2)	
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	Channels selected directly
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	

Transmission Coding

When oscillation frequency f_{OSC} is 480kHz, transmission of data code is executed as follows: when f_{OSC} is other than 480 kHz, period is multiplied by 480 kHz/f_{OSC} and its frequency by f_{OSC}/480 kHz.

A single pulse is amplitude-modulated by a carrier of 40 kHz, 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.

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)

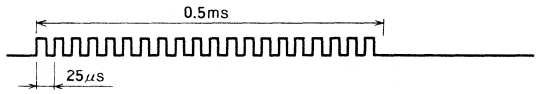


Fig. 4 Distinction between the bits "1" and "0"

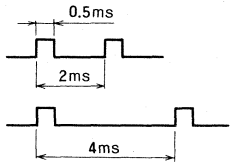
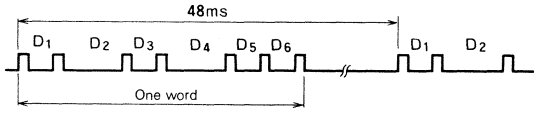
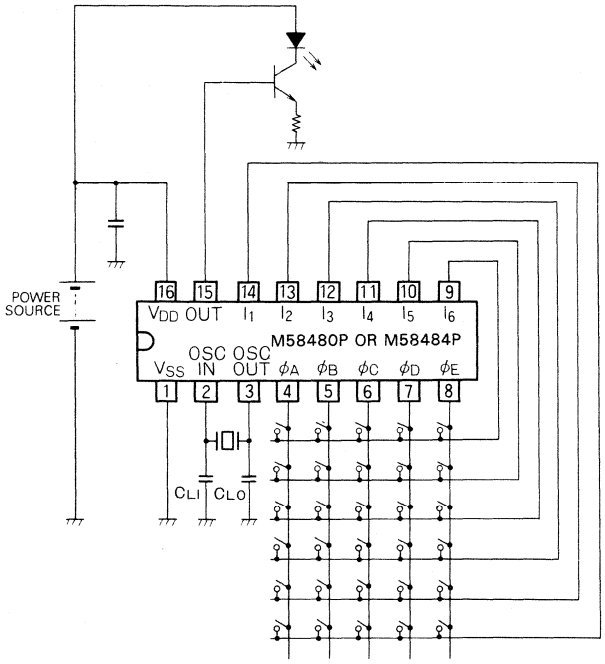


Fig. 5 Synthesis of one word (the code below shows 010100)



APPLICATION EXAMPLE



10

30-FUNCTION REMOTE-CONTROL TRANSMITTERS

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
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	2.2		8	V
f_{OSC}	Oscillation frequency		455		kHz
			480		kHz

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} = 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_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

30-FUNCTION REMOTE-CONTROL RECEIVER

DESCRIPTION

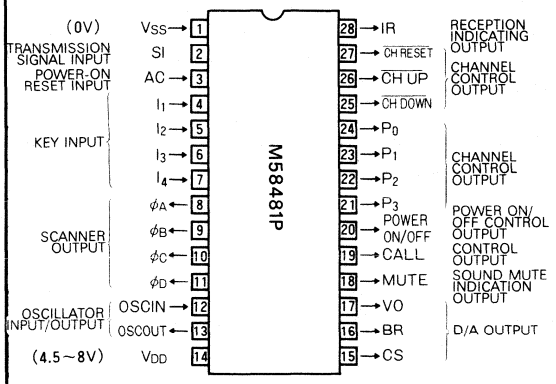
The M58481P is a 30-function remote-control receiver circuit manufactured by aluminum-gate CMOS technology for use in television receivers, audio equipment, and the like using infrared for transmission. It enables direct control of 16 functions at the receiver.

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 28P1

APPLICATIONS

- Remote-control receiver for TV or other applications

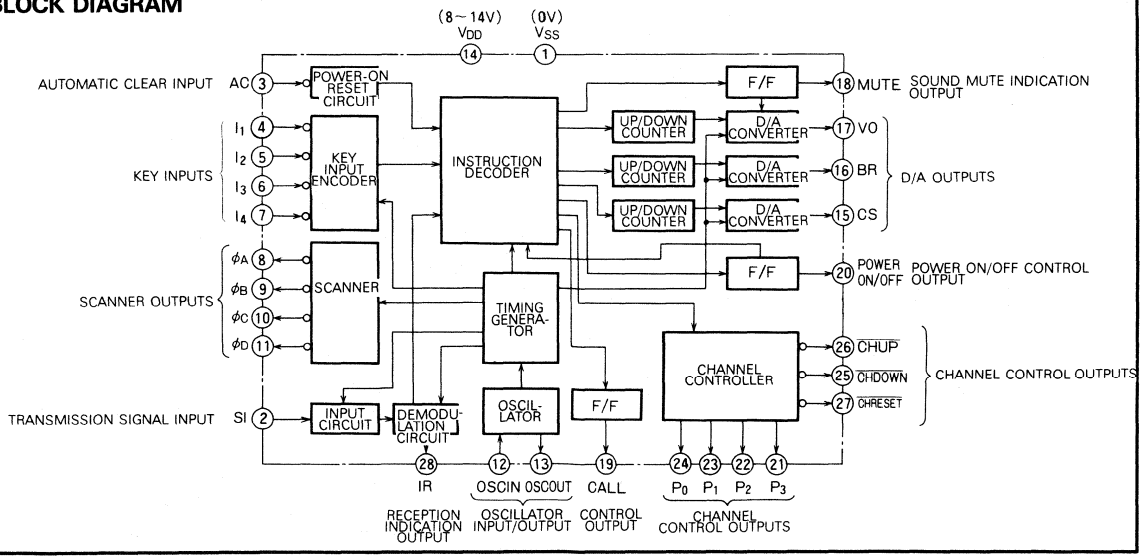
FUNCTION

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



10

30-FUNCTION REMOTE-CONTROL RECEIVER

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)

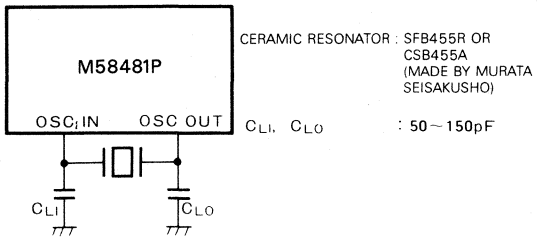
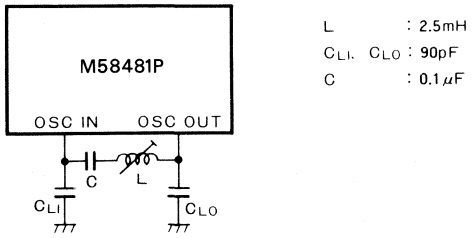


Fig. 2 An example of an oscillator (using LC network)



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)

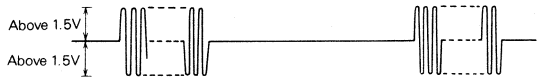


Fig. 4 SI input waveform (when applied directly)

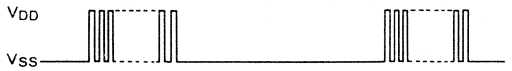
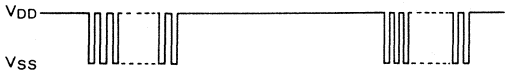


Fig. 5 SI input waveform (when applied directly)



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	Analog control
0	0	1	0	0	0	VO DOWN	
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	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)	
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	Channels selected directly
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

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	ϕ_D	ϕ_C	ϕ_B	ϕ_A
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 1.25 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/0.1sec 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}=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.

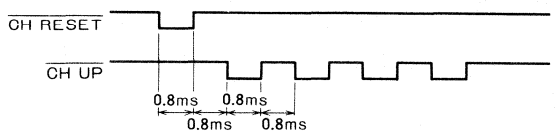
Channel Control

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 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.

10

30-FUNCTION REMOTE-CONTROL RECEIVER

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_O	Output voltage		$V_{SS} \leqq V_O \leqq V_{DD}$	—
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	℃

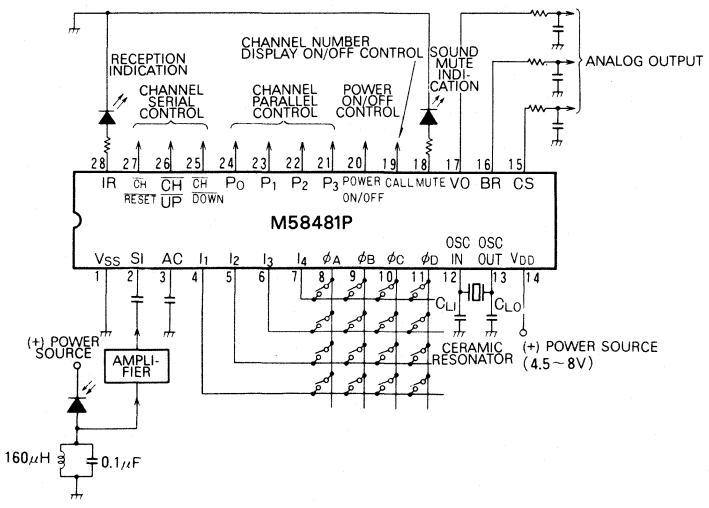
RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	4.5		8	V
f_{OSC}	Oscillation frequency		455		kHz
			480		kHz
V_I	Input voltage, SI	3			V_{P-P}

ELECTRICAL CHARACTERISTICS (T_a = 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, $I_1 \sim I_4$			20		kΩ
I_{OL}	Low-level output currents, $\phi_A \sim \phi_D$	V_{DD} = 8V, V_O = 8V	3			mA
I_{OL}	Low-level output currents, $\overline{\text{CH UP}}$, $\overline{\text{CH DOWN}}$, $\overline{\text{CH RESET}}$	V_{DD} = 8V, V_O = 8V	15			mA
I_{OZH}	Off-state output currents, $\overline{\text{CH UP}}$, $\overline{\text{CH DOWN}}$, $\overline{\text{CH RESET}}$	V_{DD} = 8V, V_O = 8V			1	μA
I_{OH}	High-level output currents, $P_0 \sim P_3$	V_{DD} = 8V, V_O = 0V	−0.5			mA
I_{OL}	Low-level output currents, $P_0 \sim P_3$	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

An example of an application circuit



MITSUBISHI ELECTRIC

29-FUNCTION REMOTE-CONTROL RECEIVER

DESCRIPTION

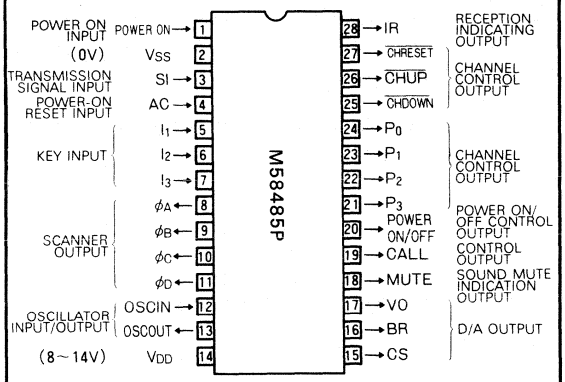
The M58485P is a 29-function remote-control receiver circuit manufactured by aluminum-gate CMOS technology for use in television receivers, audio equipment, and the like using infrared for transmission. It enables direct control of 12 functions at the receiver.

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 28P1

APPLICATIONS

- Remote-control receiver for TV or other applications

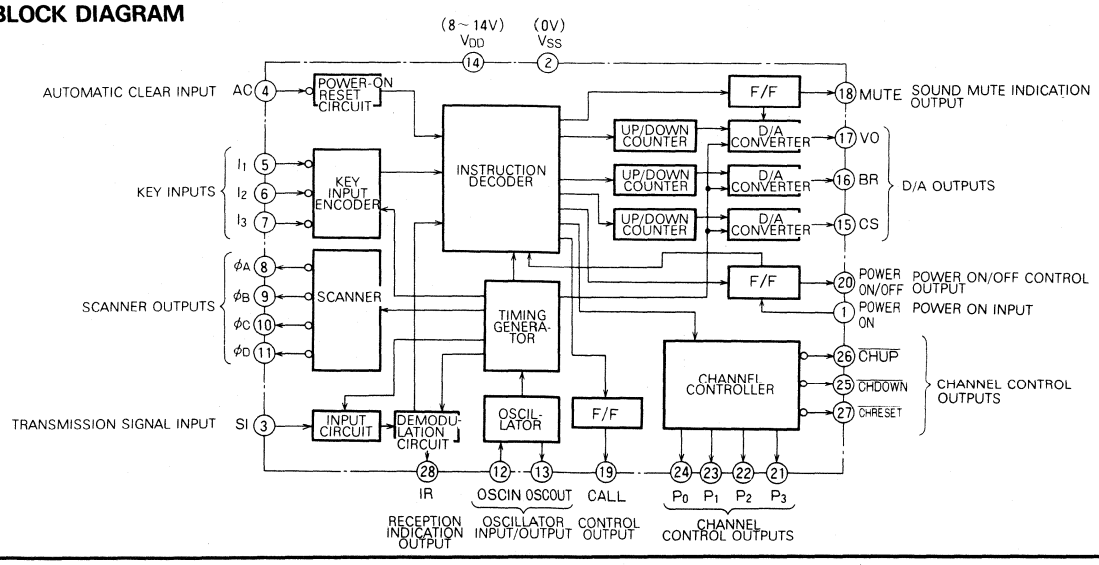
FUNCTION

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



10

29-FUNCTION REMOTE-CONTROL RECEIVER

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)

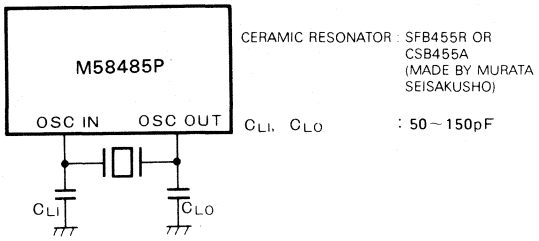
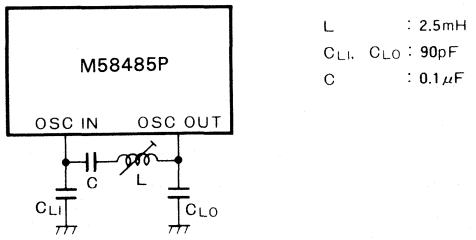


Fig. 2 An example of an oscillator (using LC network)



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)

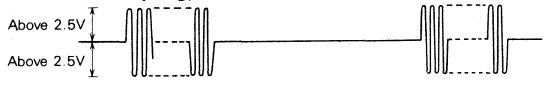


Fig. 4 SI input waveform (when applied directly)

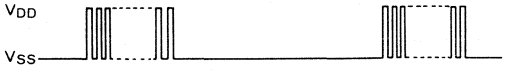
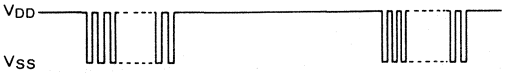


Fig. 5 SI input waveform (when applied directly)



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	Analog control
0	0	1	0	0	0	VO DOWN	
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	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	Channels selected directly
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

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	ϕD	ϕC	ϕB	ϕA
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 $\overline{\text{CH RESET}}$.

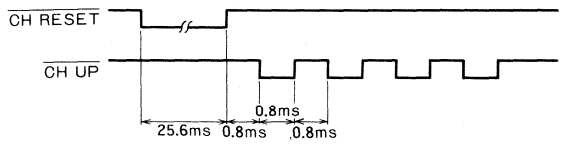
In parallel control, a 4-bit address corresponding to a selected channel number appears at output $P_0 \sim P_3$. Table 3 shows the relations between channel numbers and outputs $P_0 \sim P_3$.

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.

10

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		$V_{SS} \leqq V_I \leqq V_{DD}$	—
V_O	Output voltage		$V_{SS} \leqq V_O \leqq V_{DD}$	—
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	℃

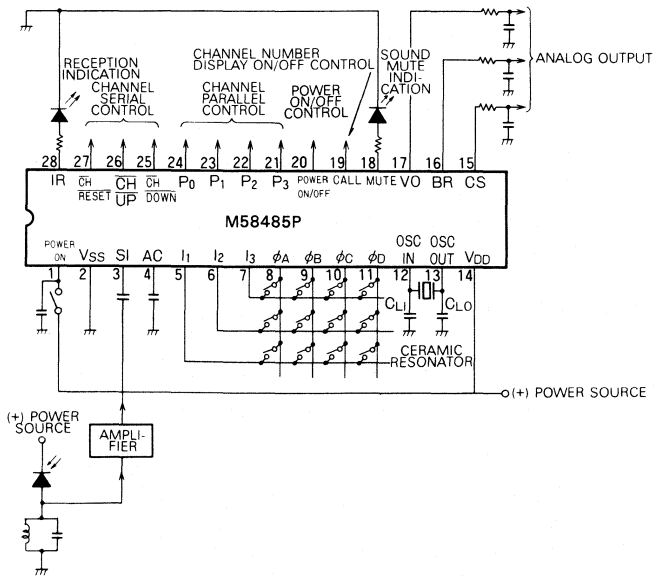
RECOMMENDED OPERATING CONDITIONS

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, SI	5			V_{P-P}

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, $I_1 \sim I_3$			20		kΩ
I_{OL}	Low-level output currents, $\phi_A \sim \phi_D$	V_O = 12V	5			mA
I_{OL}	Low-level output currents, $\overline{CH\ UP}$, $\overline{CH\ DOWN}$, $\overline{CH\ RESET}$	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, $P_0 \sim P_3$	V_O = 12V	20			mA
I_{OZH}	Off-state output currents, $P_0 \sim P_3$	V_O = 12V			1	μA
I_{OH}	High-level output currents, VO, BR, CS	V_O = 0 V	−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 = 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

An example of an application circuit



MITSUBISHI ELECTRIC

22-FUNCTION REMOTE CONTROL RECEIVER

DESCRIPTION

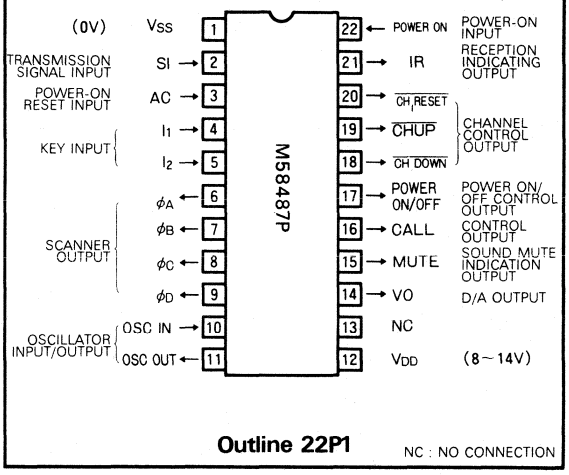
The M58487P is a 22-function remote-control receiver circuit manufactured by aluminum-gate CMOS technology for use in television receivers, audio equipment, and the like, using infrared for transmission. It enables direct control of 8 functions at the receiver.

The M58487P 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 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 saturation—are independently controlled to 64 stages by three 6-bit D/A converters
- 8 commands are controlled at the M58487P 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

PIN CONFIGURATION (TOP VIEW)



Outline 22P1

NC : NO CONNECTION

APPLICATIONS

- Remote-control receiver for TV or other applications

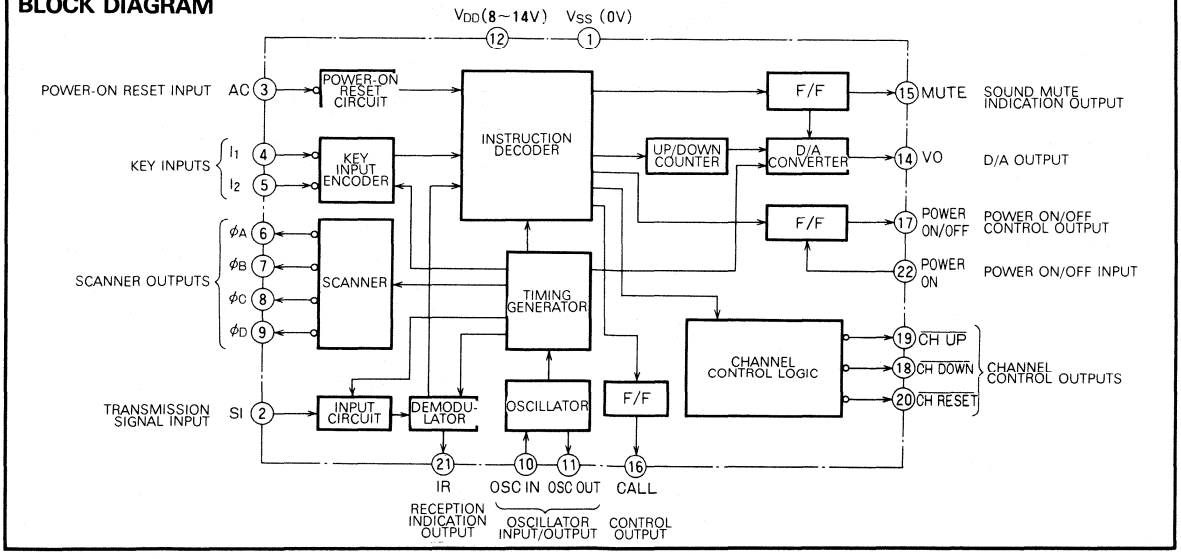
FUNCTIONS

The M58487P 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



10

22-FUNCTION REMOTE CONTROL RECEIVER

FUNCTION 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)

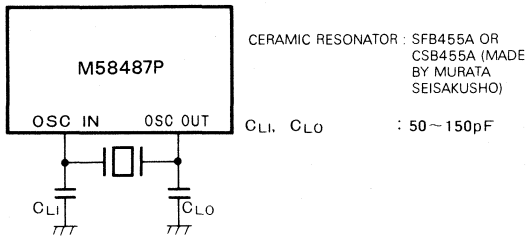
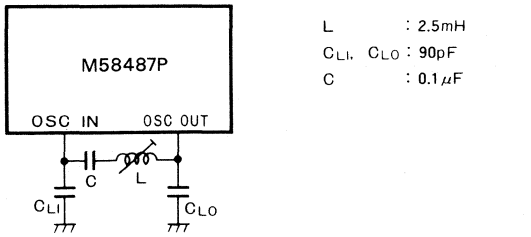


Fig. 2 An example of an oscillator (when a LC network is used)



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)

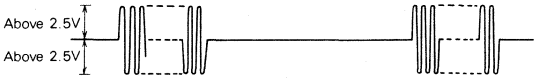


Fig. 4 SI input waveform (when applied directly)

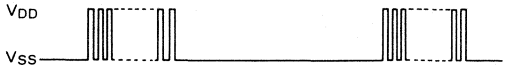
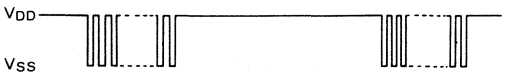


Fig. 5 SI input waveform (when applied directly)



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	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 2X4 keyboard matrix consisting of inputs I_1~I_2 and scanner outputs ϕA~ϕD. 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	ϕD	ϕC	ϕB	ϕA
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

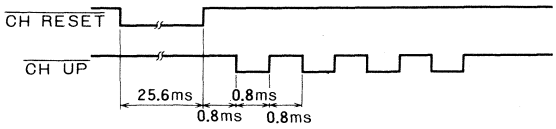
Sound mute on/off is controlled through the remote control or the key input. When sound mute is on, output VO goes low, and output MUTE goes high.

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 fosc = 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

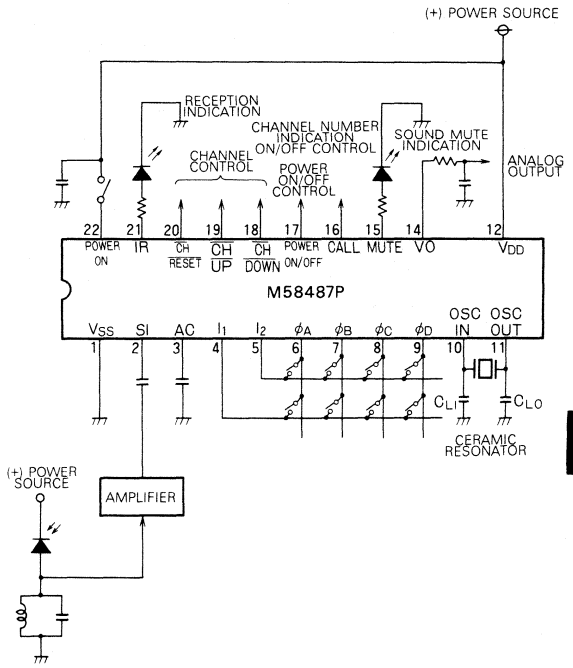
The output CALL is turned high or low by remote control or the key input. This output effects on/off control of channel number indication or change of receiving modes of multi-channel broadcasting.

Power-on Reset

Attaching a capacitor to terminal AC activates the power-on reset when power is on to the M58487P.

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.

An Example of an Application Circuit



22-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		$V_{SS} \leqq V_I \leqq V_{DD}$	V
V_O	Output voltage		$V_{SS} \leqq V_O \leqq V_{DD}$	V
Pd	Maximum power dissipation	Ta=25℃	300	mW
Topr	Operating free-air temperature range		−30∼70	℃
Tstg	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, SI	5			V_{P-P}

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, ϕ_A∼ϕ_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

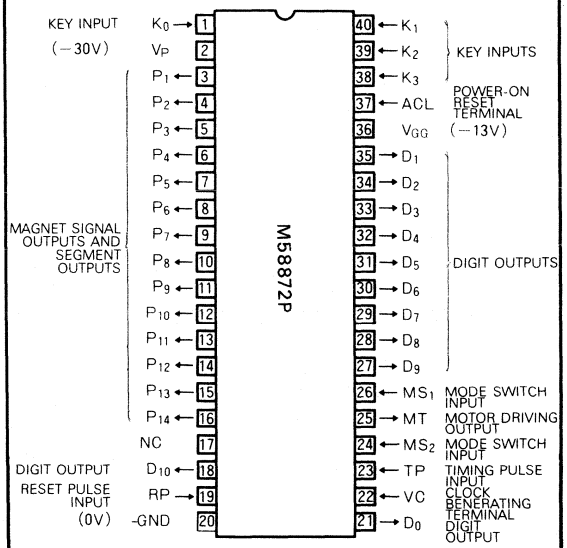
SINGLE-CHIP PRINTING CALCULATOR

DESCRIPTION

The M58872P is a single-chip printing calculator with 10-digit display and one memory using a P-channel aluminum ED-MOS process, and it is packed in a 40-pin DIL package.

Load resistors for display are built-in on the chip, enabling the direct driving of a small fluorescent display tube. Power dissipation as low as 50mW (typ), including that of load resistors for display, suits this device for compact printing calculators that use Shin Seiki's M710, M722, M723 or other microprinter. The M355 printer can also be driven by adding an external circuit on the reset pulse input terminal (terminal RP).

PIN CONFIGURATION (TOP VIEW)

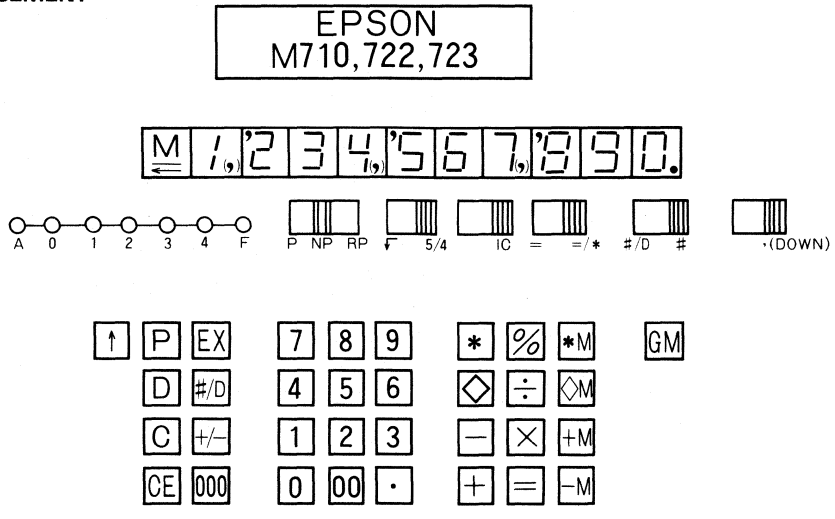


Outline 40P1

FEATURES

- Drives microprinters such as Shin Seiki's M710, M722, M723 (item 1-3-1), and M355.
- Printout format: 13 digits (10 numerical + 1 decimal point + 2 symbols) with floating-minus, leading zero suppression, and two color printout (M355 only).
- Display format: 11 digits (10 numerical + 1 sign + 3 punctuation marks) with floating-minus and leading zero suppression features.
- Adding machine operating method: 5-key method
- Independent memory: 1 register
- Arithmetic functions:
  Four basic arithmetic operations, continuous calculation, percent, surcharge and discount, add mode calculation, gross margin, constant-factor calculations (X, ÷, %, gross margin), repetitive calculation (all four arithmetic operations), power calculation, memory calculation, rounding (truncation, round-off), item counting (000~999), non-print/result only printout feature, date memorizing feature, key chattering and bouncing protecting function, 2-key rollover function, and key input buffer (6 keys).
- Choice of floating decimal point (F) and fixed decimal point (0, 1, 2, 3, 4, A).
- Direct fluorescent display tube driving capability (built-in load resistors).
- Low power dissipation (including display resistors): 50mW (typ)

KEY ARRANGEMENT



10

SINGLE-CHIP PRINTING CALCULATOR

KEY FUNCTIONS

Symbol	Key name	Function
0 ~ 9	Numerical	Entry of numerals 0~9.
·	Decimal point	Setting of decimal point is set with this key. When key is depressed more than once during entry of a number, the first depressed decimal point is effective.
+	Add	Adds to the accumulator. Continuous depression of this key performs repetitive addition. Depressing immediately after the % key effects surgharge calculation.
−	Subtract	Subtracts from the accumulator. Continuous depression of this key performs repetitive subtraction. Depressing immediately after the % key effects discount calculation.
×	Multiply	Sets up multiplication which is executed upon depression of the next function key. Continuous depression of this key performs repetitive multiplication.
÷	Divide	Sets up division which is executed upon depression of the next function key. Continuous depression of this key performs repetitive division.
=	Equals key	When multiplication or division is preset, depression of this key executes that instruction. If the %+ mode is on and when multiplication or division is not set, depression of this key reads out the contents of the accumulator.
%	Percent key	If the condition was set for multiplication or division previously, depression of this key executes percentage multiplication or division.
GM	Gross margin key	If the condition was set for division previously, depression of this key executes gross margin calculation. The following key procedure is required for the gross margin calculation: A ÷ B GM $\longrightarrow A/(1-\frac{B}{100})$ A ÷ B +/− GM $\longrightarrow A/(1+\frac{B}{100})$ However, in the case of B>100, resultant symbol will coincide with the symbol of A.
◇	Subtotal key	Depression of this key reads out the contents of the accumulator. But the contents of the accumulator are not affected.
*	Total key	Depression of this key reads out the contents of the accumulator, and then the contents of the accumulator are cleared.
+M	Memory plus key	If the × or ÷ key was depressed before, depression of this key executes multiplication or division with its result being added to the memory register. If neither the × nor the ÷ key was depressed before, the contents of the display register are added to the memory register.
−M	Memory minus key	If the × or ÷ key was depressed before, depression of this key executes multiplication or division with its result being subtracted from the memory register. If neither the × nor the ÷ key was depressed before, the contents of the display register are subtracted from the memory register.
◇M	Memory sub-total key	The contents of the memory register are read out, but remain unchanged.
*M	Memory total key	The contents of the memory register are read out, and cleared.
#/D	Non-add/date key	When the decimal point key has been depressed twice or more, it operates as a date printout key ("D" is printed in the symbol column). If the decimal point key has been depressed less than twice, it operates as a non-add key ("#" is printed out on most significant digit). Meanwhile, if the # mode is on, the key is operated as a non-add key only. Up to two decimal points are available.
D	Date key	The date is stored in the date register by the depression of this key, with the date being printed out on the tally roll ("D" is printed out on the symbol column). Up to two decimal points are available. Depression of this key without previous numerical key entry will read out the contents of the date register.
P	Print key	The contents of the display register are printed out.
EX	Exchange key	Depression of this key causes the contents of the display register to be exchanged with the contents of the multiplication/division constant register.
+/−	Change sign key	Depression of this key changes the sign of the display register.
CE	Clear entry key	Depression of this key clears the numerical entry. Also, this key is used in releasing overflow condition of the key input buffer.
C	Clear key	Depression of this key clears all registers and operating states except the memory register.
⇧	Paper feed key	Depression of this key advances tally roll.

MODE SWITCH FUNCTIONS

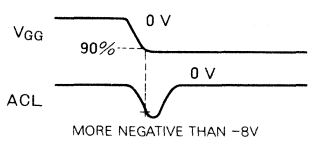
Symbol	Switch name	Function
TAB	Floating/fixed decimal point selection switch	With this switch, function of floating (F) or fixed decimal point (0, 1, 2, 3, 4, A) is designated.
5/4	Round (5/4)	When off, rounding (retain 5, strike 4) (5.4) is performed.
IC	Item print	When on, contents of the item counter are printed.
NP	Non-print	When at the NP, all printouts except by the P key are disabled. When at RP, results by =, %, and GM key, and add-on/discount calculation are printed out, and depression of ◇, *, RM, *M, C, %, D and P keys also print out. When OFF, all printouts are performed.
,	Punctuation selection	When at OFF position, punctuation is indicated at the upper-left corner of the relevant columns; If ON, it is indicated at the lower-right corner of the relevant columns.
=/*	Equal total mode	If this switch is located at the position OFF, the = key functions as an equal key during multiplication/division operation. If this switch is located at the position ON, the = key functions as an equal key during multipication/division operation, otherwise, as a total key.
#	Non-add mode switch	If this key is located at the position ON, depression of the %/0 performs only the non-add function.

PIN FUNCTIONS

Symbol	Pin name	Input or output	Functional significance
$K_0 \sim K_3$	Key input	Input	Consists of four independent lines and is used to identify key engagement in conjunction with the matrix composed with the digit outputs $D_1 \sim D_9$.
MS 1 MS 2	Mode switch inputs	Input	Mode switch input consists of two independent lines and is used to identify the mode in conjunction with the matrix composed with the digit outputs $D_1 \sim D_9$.
$D_0 \sim D_{10}$	Digit outputs	Output	This terminal is used for a digit designation signal which is supplied for dynamic displaying of a fluorescent display tube. $D_1 \sim D_9$ constitutes the matrix with the key and mode switch inputs.
$P_1 \sim P_{14}$	Magnet signal outputs	Output	The output terminals $P_2 \sim P_{14}$ are used to supply signals with which the printer hammer triggering magnets are controlled. The signal P_1 is used in driving ink-ribbon switching mechanisms for the type M355 printer. Signals $P_2 \sim P_6$, P_8, P_9, P_{11} and P_{12} are used in common with the fluorescent tube segment driving outputs S', S•, Sg ~ Sa.
TP	Timing pulse input	Input	Input terminal for receiving the printer timing pulse.
RP	Reset pulse input	Input	Input terminal for receiving the printer reset pulse.
MT	Motor drive output	Output	Output terminal for motor driving output.
VC	Clock oscillation terminal	Input	An external CR network is connected through this terminal for setting the oscillation frequency in the internal clock oscillation circuit.
ACL	Power-on reset terminal	Input	An external CR network is connected through this terminal for attaining power-on reset during power-on time.

CONDITIONS FOR ESTABLISHING POWER-ON RESET FUNCTION

(1) In order to satisfy the following waveform against the supply voltage (V_{GG}), it is necessary to decide the constant factor for the C_a and R_a of the external circuit.



(2) In such a case the power switch has to be operated repeatedly, and the residual voltage in the supply voltage (V_{GG}) during switch opening time must be dropped to the GND level completely.

(3) Connection of the terminal ACL with the external circuit should be as short as possible so as to avoid noise in the ACL terminal.

10

SINGLE-CHIP PRINTING CALCULATOR

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{GG}	Supply voltage	With respect to GND	0.3～−22	V
V_I	Input voltage		0.3～−22	V
V_O	Output voltage		0.3～−33	V
Pd	Power dissipation	Ta=25℃	250	mW
Topr	Operating free-air temperature range		0～50	℃
Tstg	Storage temperature range		−40～125	℃

RECOMMENDED OPERATING CONDITIONS (Ta= 0 ～50℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{GG}	Supply voltage	−12	−13	−14	V
V_{IH}	High-level input voltage	0		−3	V
V_{IL}	Low-level input voltage	−8		−14	V
I_{OH}	High-level output current			−4	mA
V_{OL}	Low-level output voltage	−28		−30	V
V_P	Supply voltage for display section			−30	V
Rf	Frequency setup resistance	31.4	33	34.6	kΩ
Cf	Frequency setup capacitance	64.6	68	71.4	pF
tcon	Key contact duration	60			ms
toff	Key off duration	80			ms
C_{KB}	Keyboard capacitance			30	pF

Note 1 : The Rf should be connected between the terminals VC and V_{GG}, and Cf between VC and GND, in the shortest distance possible.

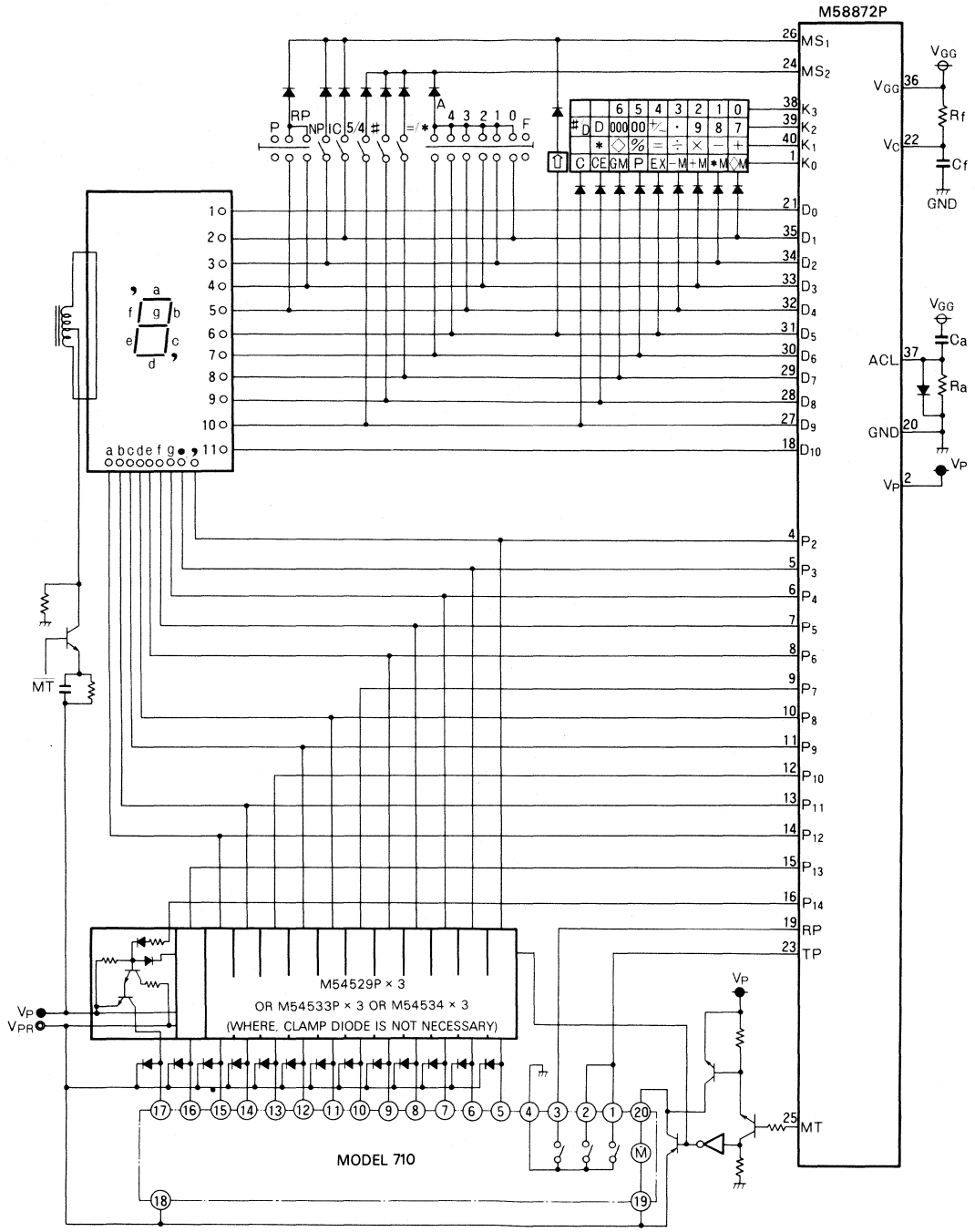
ELECTRICAL CHARACTERISTICS (Ta= 0 ～50℃, V_{GG}=−13± 1V, V_P=−30V, Rf=33kΩ, Cf=68pF, unless otherwise noted.)

Symbol	Parameter	Test condtitions	Limits			Unit
			Min	Typ	Max	
V_{IH}	High-level input voltage		0		−3	V
V_{IL}	Low-level input voltage		−7	−13	−14	V
V_{OH}	High-level output voltage	I_{OH}=− 4 mA			−2	V
V_{OL}	Low-level output voltage		−28		−30	V
I_{IL}	Low-level input current	V_I=−14V			−14	μA
C_I	Input capacitance	V_I= 0 V, f= 1MHz, 25mVrms Ta=25℃			7	pF
I_{GG}	Supply current	Ta=25℃		−2.5	−4	mA
I_P	Supply current for display section	Ta=25℃		−0.6	−2	mA
Pd	Power dissipation	$V_{GG}\times I_{GG}+V_P\times I_P$ Ta=25℃		49	116	mW

SWITCHING CHARACTERISTICS (Ta= 0 ～50℃, V_{GG}=−13± 1V, V_P=−30V, Rf=33kΩ, Cf=68pF, unless otherwise noted.)

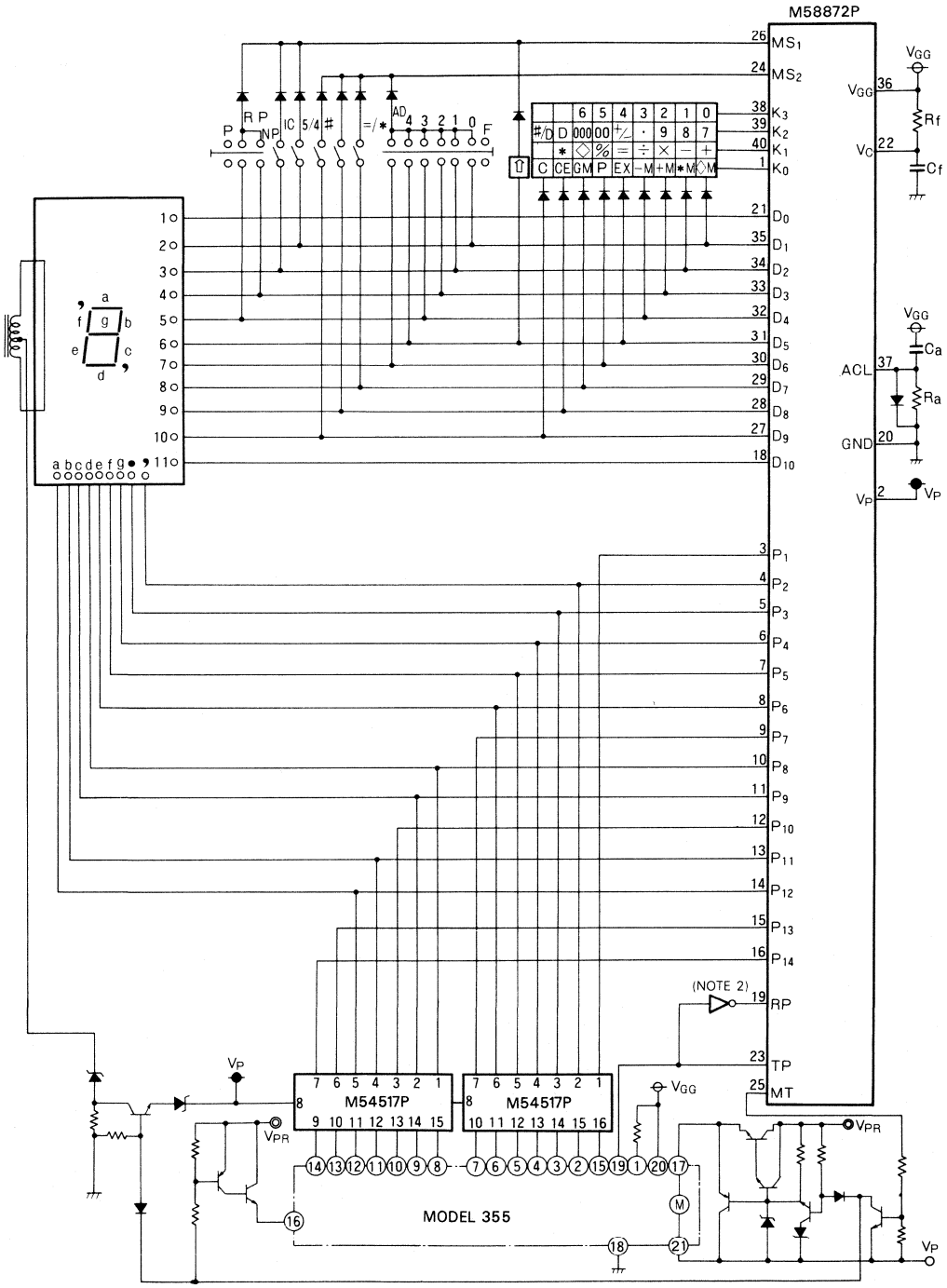
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
fr	Repetitive frequency	Ta=25℃, 1/6 of the original oscillation frequency	54	70	90	kHz
t_B	Interval of scanning pulse	Ta=25℃	22		37	μs
t_C	Comparison time of print data	Ta=25℃			0.2	ms

SCHEMATIC DIAGRAM FOR M710, M722, AND M723



SINGLE-CHIP PRINTING CALCULATOR

SCHEMATIC DIAGRAM FOR M355



Note 2 : Delay of the RP from the TP should be within 1 clock cycle (about 10μs).

PRINT WHEEL ARRANGEMENTS
M710, M722, AND M723/1-3-1

13	12	11	10	9	8	7	6	5	4	3	2	1	
0	0	0	0	0	0	0	0	0	0	0	D	+	0
1	1	1	1	1	1	1	1	1	1	1	↕	−	1
2	2	2	2	2	2	2	2	2	2	2	G	×	2
3	3	3	3	3	3	3	3	3	3	3	G	÷	3
4	4	4	4	4	4	4	4	4	4	4	K	=	4
5	5	5	5	5	5	5	5	5	5	5	K	%	5
6	6	6	6	6	6	6	6	6	6	6	M	◇	6
7	7	7	7	7	7	7	7	7	7	7	P	*	7
8	8	8	8	8	8	8	8	8	8	8	√	S	8
9	9	9	9	9	9	9	9	9	9	9	C	T	9
#	·	·	·	·	·	·	·	·	·	·	E	M+	10
—	—	—	—	—	—	—	—	—	—	—	—	M−	11

FOR M355

13	12	11	10	9	8	7	6	5	4	3	2	1	
0	0	0	0	0	0	0	0	0	0	0	D	+	0
1	1	1	1	1	1	1	1	1	1	1	↕	−	1
2	2	2	2	2	2	2	2	2	2	2	G	×	2
3	3	3	3	3	3	3	3	3	3	3	G	÷	3
4	4	4	4	4	4	4	4	4	4	4	Σ	=	4
5	5	5	5	5	5	5	5	5	5	5	Σ	%	5
6	6	6	6	6	6	6	6	6	6	6	M	◇	6
7	7	7	7	7	7	7	7	7	7	7	P	*	7
8	8	8	8	8	8	8	8	8	8	8	K	S	8
9	9	9	9	9	9	9	9	9	9	9	C	T	9
#	·	·	·	·	·	·	·	·	·	·	E	M+	10
—	—	—	—	—	—	—	—	—	—	—	—	M−	11

EXAMPLE OF SYMBOL PRINTOUT

Key	Column 13	3	2	1	
[+]				+	
				+	
				*	(SP)
[−]				−	
				−	
				*	(SP)
[×]				×	
[÷]				÷	
[%]				%	
				*	(SP)
[=]				=	
				*	(SP)
[◇]				◇	
[*]				*	(SP)
[#G]	#				
			D		
[D]			D		
[C]			C		(SP)
[GM]			G		
				*	(SP)
[EX]			↕		
[+M]				M+	
				=	
				M−	(SP)
[−M]				M−	
				=	
				M−	(SP)
[◇M]				S	
[*M]				T	(SP)
(OVF)				*	
			E		(SP)
				T	
			E		(SP)
(POC)			C		(SP) (SP)

Note 3 : For the type M355 printer, printout is carried out in red when the results are negative or when the [−] or [−M] key (in the case of memory subtraction) is depressed.

10

SINGLE-CHIP PRINTING CALCULATOR

OPERATIONAL EXAMPLES

The notation "Red" in the printout column shows printout in red when the type M355 printer is used.

1. Addition and Subtraction

(1) 10+11+12+12+13−14=

(2) 1 + 2 + (1 +2) + 5 = (T_1)

(3) −(T_1)−(T_1)+(T_1)+ 3 =

(4) 1.23+12.1234+ 4 +0.05=

2. Multiplication and Division

(1) 123× 3 =

(2) 123× 3 ÷ 2 =

(3) 2 ÷ 4 =

(4) 2 ÷ 3 =

(5) 12× 12=

(6) 12.12× 12.12=

(7) 12.456× 12×0.346÷ 1.4567=

(8) 12.3× 12.3÷ 12.3× 1 =

Mode switch							Key operation		Display	Printout		
TAB	5/4	IC	NP	#	,	⁷⁄₄						
2	ON	OFF	OFF	OFF	OFF	OFF						
							10	+	10.00		10.00	+
							11	+	21.00		11.00	+
							12	+	33.00		12.00	+
								+	45.00		12.00	+
								◇	45.00		45.00	◇
							13	+	58.00		13.00	+
							14	−	44.00	Red	14.00	−
								*			44.00	*
									44.00		(1SP)	
↓												
F							1	+	1.		1.	+
							2	+	3.		2.	+
							(4)	◇	3.		3.	◇
								+	6.		3.	+
							5	+	11.		5.	+
							(6)	*			11.	*
									11.		(1SP)	
								−	−11.	Red	11.	−
								−	−22.	Red	11.	−
								+	−11.		11.	+
								#D	−11.		#−11.	
							3.0	#D	3.0		#3.0	
								#D	3.0		#3.0	
								+	−8.		3.	+
								*		Red	8.	*
									8.		(1SP)	
↓												
A		ON					123	+	1.23		1.23	+
							12.1290	+	13.35		12.12	+
							4.	+	17.35		4.00	+
							5	+	17.40		0.05	+
								◇	17.40	004	17.40	◇
								*		004		
											17.40	*
									17.40		(1SP)	
								*		000		
											0.	*
									0.		(1SP)	

Mode switch							Key operation		Display	Printout		
TAB	5/4	IC	NP	#	,	⁷⁄₄						
2	OFF	OFF	OFF	OFF	OFF	OFF						
							123	×	123.		123.	×
							3	=			3.	=
											369.00	*
									369.00		(1SP)	
↓												
A							123	×	123.		123.	×
							3	÷	369.		3.	÷
							2	=			2.	=
											184.50	*
									184.50		(1SP)	
							2	÷	2.		2.	÷
							4	=			4.	=
											0.50	*
									0.50		(1SP)	
							2	÷	2.		2.	÷
							3	=			3.	=
											0.67	*
									0.67		(1SP)	
↓												
3							12	×	12.		12.	×
								=			12.	=
											144.000	*
									144.000		(1SP)	
							1.212	×	12.12		12.12	×
								=			12.12	=
											146.894	*
									146.894		(1SP)	
							12.456	×	12.456		12.456	×
							12	×	149.472		12.	×
							0.346	÷	51.717312		0.346	÷
							1.4567	=			1.4567	=
											35.503	*
									35.503		(1SP)	
↓												
1							12.3	×	12.3		12.3	×
							12.3	÷	151.29		12.3	÷
								EX	12.3		151.29	↕
							12.3	×	12.3		12.3	×
							1	=			1.	=
											12.3	*
									12.3		(1SP)	

MITSUBISHI ELECTRIC

3. Multiplication/Division Using Constant Factor and Power Calculation

(1) 1.05 × 100 =　　1.05 × 200 =　　1.05 × 300 =

(2) 150 ÷ 3 =　　6 ÷ 3 =　　2 ÷ 3 =

(3) 2^4 =　　2 × 2 × 2 × 2 =

Mode switch							Key operation		Display	Printout	
TAB	5/4	IC	NP	#	,	⁄.					
F	OFF	OFF	OFF	OFF	OFF	OFF	1.05	×	1.05	1.05	×
							100	=		100.	=
										105.	*
									105.	(1SP)	
							200	=		200.	=
										210.	*
									210.	(1SP)	
							300	=		300.	=
										315.	*
									315.	(1SP)	
							150	÷	150.	150.	÷
							3	=		3.	=
										50.	*
									50.	(1SP)	
							6	=		6.	=
										2.	*
									2.	(1SP)	
							2	=		2.	=
										0.666666666	*
									0.666666666	(1SP)	
							2	×	2.	2.	×
								=		2.	=
										4.	*
									4.	(1SP)	
								=		4.	=
										8.	*
									8.	(1SP)	
								=		8.	=
										16.	*
									16.	(1SP)	
							2	×	2.	2.	×
								×	4.	2.	×
								×	8.	2.	×
								P	8.	8.	
								=		2.	=
										16.	*
									16.	(1SP)	

4. Percentage Calculation

(1) 200 × 3 %　　200 × 4 % = α　　200 × α%

(2) 123 ÷ 200%　　6 ÷ 200%

(3) 200 × 5 %　　200 × (100 − 5)% = β　　β × (100 + 10)%

Mode switch							Key operation		Display	Printout	
TAB	5/4	IC	NP	#	,	⁄.					
F	OFF	OFF	OFF	OFF	OFF	OFF	200	×	200.	200.	×
							3	%		3.	%
									6.	6.	*
										(1SP)	
							4	%		4.	%
									8.	8.	*
										(1SP)	
								%		8.	%
									16.	16.	*
										(1SP)	
							123	÷	123.	123.	÷
							200	%		200.	%
									61.5	61.5	*
										(1SP)	
							6	%		6.	%
									3.	3.	*
										(1SP)	
							200	×	200.	200.	×
							5	%		5.	%
									10.	10.	*
										(1SP)	
								−		Red	−
										190.	*
									190.	(1SP)	
								×	190.	190.	×
							10	%		10.	%
									19.	19.	*
										(1SP)	
								+			+
										209.	*
									209.	(1SP)	
								%		10.	%
									19.	19.	*
										(1SP)	
								C		0.	C
									0.	(1SP)	
							10	%	10.	NO-OP	
							123		10' 123.		

10

SINGLE-CHIP PRINTING CALCULATOR

5. Combined Calculation

(1) 15 + 16 × 2 − 17 × 3 − 15 ÷ 2 =

(2) (1 + 2 + 3) × (4 + 5) =

(3) (1 + 2 + 3) × (6 − 7) =

6. Memory Calculation

(1) 11.111111 − 22.22 + 55.55555 =

(2) − (123 × 11) + 34 = α α − α =

(3) (2 × 3) + 0.04 − (5 × 6) − 0.07 =

Mode switch	Key operation	Display	Printout
TAB 5/4 IC NP # , F/*			
2 OFF OFF OFF OFF OFF OFF	15 [+]	15.00	15.00 +
	16 [×]	16.	16. ×
	2 [=]		2. =
			32.00 *
		32.00	(1SP)
	[+]	47.00	32.00 +
	17 [×]	17.	17. ×
	3 [=]		3. =
			51.00 *
		51.00	(1SP)
	[−]	−4.00	Red 51.00 −
	15 [÷]	15.	15. ÷
	2 [=]		2. =
			7.50. *
		7.50	(1SP)
	[−]	−11.50	Red 7.50 −
	[*]		Red −11.50 *
		−11.50	(1SP)
	1 [+]	1.00	1.00 +
	2 [+]	3.00	2.00 +
	3 [+]	6.00	3.00 +
	[*]		6.00 *
		6.00	(1SP)
	[×]	6.00	6.00 ×
	4 [+]	4.00	4.00 +
	5 [+]	9.00	5.00 +
	[*]		9.00 *
		9.00	(1SP)
	[=]		9.00 =
			54.00 *
		54.00	(1SP)
	6 [+]	6.00	6.00 +
	7 [−]	−1.00	Red 7.00 −
	[*]		Red −1.00 *
		−1.00	(1SP)
	[=]		Red −1.00 =
			Red −6.00 *
		−6.00	(1SP)

Mode switch	Key operation	Display	Printout
TAB 5/4 IC NP # , F/*			
4 OFF OFF OFF OFF OFF OFF	11.111111 [+M]	· 11.1111	11.1111 M+
	. 22,22 [−M]	· 22.2200	22.2200 M−
	55.55555 [+M]	· 55.5555	55.5555 M+
	[◇M]	· 44.4466	4.44466 S
↓2F	[◇M]	· 44.4466	44.4466 S
	[*M]		44.4466 T
		44.4466	(1SP)
	[◇M]	0.	0. S
	123 [×]	123.	123. ×
	11 [−M]		11. =
			1353. M−
		· 1'353.	(1SP) -
	[×]	· 1'353.	1353. ×
	22 [=]		22. =
			29766. *
		· 29'766.	(1SP)
	34 [+M]	· 34.	34. M+
	[◇M]	· −1'319.	Red −1319. S
	[−M]	−1'319.	Red −1319. M−
	[*M]		Red −0. T
		−0.	(1SP)
↓A	2 [×]	2.	2. ×
	3 [+M]		3. =
			6.00 M+
		· 6.00	(1SP)
	4 [+M]	· 0.04	0.04 M+
	5 [×]	· 5.	5. ×
	6 [−M]		6. =
			30.00 M−
		· 30.00	(1SP)
	7 [−M]	· 0.07	Red 0.07 M−
	[*M]		Red −24.03 T
		−24.03	(1SP)

MITSUBISHI ELECTRIC

7. Gross Margin Calculation

(1) 200 ÷ 10 GM

(2) 200 ÷ 10 +/- GM

(3) 600 ÷ 20 GM

(4) 6000 ÷ 20 +/- GM

Mode switch TAB	5/4	IC	NP	#	,	%	Key operation		Display		Printout		
2	ON	OFF	OFF	OFF	OFF	OFF	200	÷	200.		200.		÷
							10	GM			10.	G	
											222.22		*
									222.22		(1SP)		
							200	÷	200.		200.		÷
							10	+/-	−10.				
								GM		Red	−10.	G	
											181.18		*
									181.81		(1SP)		
							600	÷	600.		600.		÷
							20	GM			20.	G	
											750.00		*
									750.00		(1SP)		
							6000	÷	6000.		6000.		÷
							20	+/-	−20.				
								GM		Red	−20.	G	
											5000.00		*
									5000.00		(1SP)		

8. Overflow Error

(1) 9999999999 + 1000 =

(2) −9999999999 − 1000 =

(3) 1111111111 × 1000 =

(4) 1111111111 × 10 ×

(5) 1111111111 ÷ 0.000000001 =

(6) 1111111111 ÷ 0.000000001 %

(7) 1 ÷ 0 =

(8) 9999999999 +M 1.5 +M

Mode switch TAB	5/4	IC	NP	#	,	%	Key operation		Display		Printout		
F	OFF	ON	OFF	OFF	OFF	OFF	9999999999	+	9'999'999'999.		9999999999.		+
							1000	+			1000.		+
											002		
											10000000099		*
												E	
									E 1.000000099		(1SP)		
								C			0.	C	
									0.		(1SP)		
							9999999999	−	−9'999'999'999.	Red	9999999999.		−

Mode switch TAB	5/4	IC	NP	#	,	/	Key operation		Display		Printout		
F	OFF	ON	OFF	OFF	OFF	OFF	1000	−		Red	1000.		−
											002		
										Red	1000000999	−	*
												E	
									E 1.000000099		(1SP)		
								C			0.	C	
									0.		(1SP)		
							1111111111	×	1'111'111'111.		1111111111.		×
							1000	=			1000.		=
											111.1111111		*
												E	
									E 111.1111111		(1SP)		
								C			0.	C	
									0.		(1SP)		
							11111111111	×	1'111'111'111.		1111111111.		×
							10	×			10.		×
											1.111111111		*
												E	
									E 1.111111111		(1SP)		
								C			0.	C	
									0.		(1SP)		
							1111111111	÷	1'111'111'111.		1111111111		÷
							0.000000001	=			0.000000001		=
											111111111.1		*
												E	
									E 111111111.1		(1SP)		
								C			0.	C	
											(1SP)		
		OFF					1111111111	÷	1'111'111'111.		1111111111		÷
							0.000000001	%			0.000000001		%
											1.111111111		*
												E	
									E 1.111111111		(1SP)		
								C			0.	C	
									0.		(1SP)		
							1	÷	1.		1.		÷
							0	=			0.		=
											0.		*
												E	
									E 0.		(1SP)		
								C			0.	C	
									0.		(1SP)		
							9999999999	+M	.9'999'999'999.		9999999999.		M+
							1.5	+M			1.5		M+
											10000000000		T
												E	
									E 1.000000000		(1SP)		
								C			0.	C	
									0.		(1SP)		

10

SINGLE-CHIP PRINTING CALCULATOR

9. Other Calculations

Mode switch								Key operation		Display	Printout	
TAB	5/4	IC	NP	#	,	⁻/₊						
2	OFF	ON	RP	OFF	ON	ON		(POC)			(1SP)	
											0.	C
										0.	(1SP)	
								78.10.2	D	78.102	78.10.2	D
								56789	+	56,789.00		
								45632	−	11,157.00		
								38426	−	−27,269.00		
									=		003	
											Red −27269.00	*
										−2,726,900	(1SP)	
									−	27,269.00		
									−	54,538.00		
									◇	54,538.00	002	
											54538.00	◇
									=		002	
										54,538.00	54538.00	*
											(1SP)	
								200	×	200.		
								25	=			
											5000.00	*
										5,000.00	(1SP)	
								2	=			
											400.00	*
										400.00	(1SP)	
									+	400.00		
									+	800.00		
								3	=		002	
											800.00	*
										800.00	(1SP)	
									=		000	
										0.	0.	*
											(1SP)	
								12.34	#/D	1.234	#1.234	
								12.3.4	#/D	12.34	12.3.4	D
				ON				53.4.6	#/D	53.46	#53.4.6	
									D	53.46	78.10.2	D
									C		0.	C
										0.	(1SP)	
		OFF						2	×	2.		
								3	+M			
										6.00		
								4	×	4.		
								5	−M			
										20.00		
									RM	−14.00	Red −14.00	S
								9999999999	−M			

Mode switch								Key operation		Display	Printout		
TAB	5/4	IC	NP	#	,	⁻/₊							
2	OFF	OFF	RP	ON	ON	ON							
											Red 10000000013	−	T
												E	
										E 1.000000001	(1SP)		
									C		0.	C	
										0.	(1SP)		
									*M		0.		T
										0.	(1SP)		

MITSUBISHI ELECTRIC

MICROCOMPUTER SYSTEMS

11

MELCS 8/2 SINGLE-BOARD COMPUTER

DESCRIPTION

The PCA0801 is a single-board computer that is fabricated on a single 125 x 145 mm printed circuit board using the MELPS 8 CPU (Mitsubishi Electric LSI Processor: M5L8080AP). It is designed for applications where ease of use as a built-in component in a user's cabinet and high performance reliability are required.

FEATURES

- Capacity of RAM: 256 bytes
- Capacity of EPROM or mask ROM: 2K bytes (max)
- Programmable I/O ports: 8-bit x 6 (48 bits)
- Interrupt: 1-level (externally expandable up to 8-level)
- Single 5V power supply
- Easy memory and I/O port expansion by using the PCA0802 memory and I/O expansion board
- 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

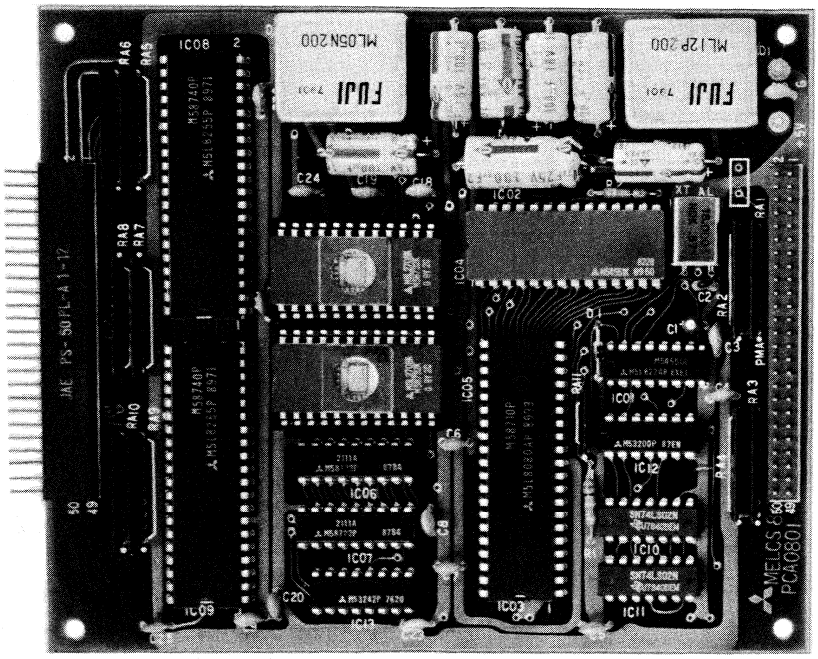
The PCA0801 is a highly reliable single-board computer designed around Mitsubishi's M5L8080AP CPU and its LSI family. The 8-bit parallel CPU is fabricated by the N-channel silicon-gate MOS process.

The standard version of the PCA0801 comes with 1K-byte of electrically programmable read-only memory (EPROM) in the form of one M5L2708K, but an auxiliary socket on the board enables a second 1K-byte M5L2708K EPROM to be plugged in to provide a 2K-byte total. The standard board also comes with 256 bytes of static random-access memory (RAM) in the form of two M5L2111APs.

For its I/O ports, the PCA0801 contains two programmable peripheral interfaces (PPI: M5L8255AP×2), providing 48 input/output pins that comprise six 8-bit programmable input/output ports. DC-DC converters are provided for 12V and −5V power supplies, so that only a 5V external power supply is needed.

Optional features of the PCA0801 include the PCA0802 memory and I/O expansion board and the PCA0803 program checker.

Mitsubishi PCA0801 microcomputer



11

MELCS 8/2 SINGLE-BOARD COMPUTER

OPERATIONS

When an 18MHz crystal oscillator is connected externally between pins XTAL1 and XTAL2, clocks ϕ_1 and ϕ_2 for the CPU with a basic cycle of 500ns and $\phi_{2\,(TTL)}$ for oscillator source are generated by the M5L8224P clock generator. By connecting an RC circuit to the reset input terminal RESIN, the reset signal RESET is generated when the system power supply is turned on. When the SYNC signal from the CPU is applied to the SYNC terminal, the STSTB signal is generated for latching the status. When the CPU receives the RESET signal, the CPU is then reset at the rising-edge of ϕ_1 so as to clear the contents of the program counter and instruction register to 0, and flip-flops INTA and HLDA are also reset. Interrupt is being inhibited, and both the address bus and the data bus are in the floating state at this moment. All the control signals generated from the CPU are also reset. As the content of the program counter is at 0 upon completion of the RESET signal, program execution is started from the address 0.

The system controller functions to separate the data bus of the CPU M5L8080AP from the memory and I/O ports (PPI); it generates all the necessary signals for directly controlling the memory and the I/O ports.

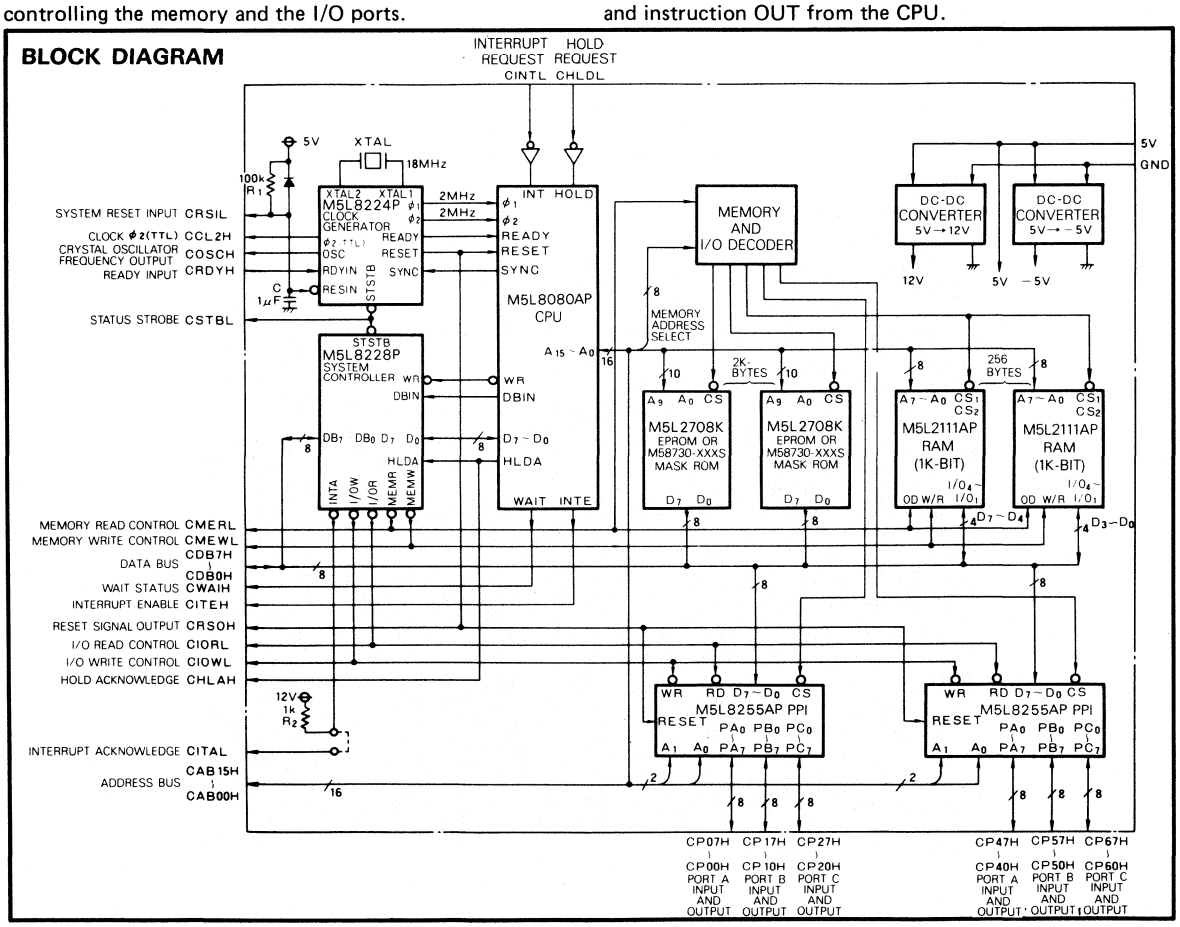
The strobe signal STSTB latches the status information from the CPU into the internal status latch. Depending on the control signals from the CPU (DBIN, WR, and HLDA) and the contents of the internal status latch, the following five control signals are generated: memory read (MEMR), memory write (MEMW), input/output read (I/OR), input/output write (I/OW) and interrupt acknowledge (INTA). These control signals control read/write data transfer between the ROM, RAM and I/O ports.

The decoder for the memory and I/O ports reads the address of the memory and I/O ports from the address bus, and sends the control signal to the corresponding memory and I/O port address.

The PCA0801 contains 1K-byte of EPROM (M5L2708K x 1) and 256 bytes of static RAM (M5L2111AP x 2). A socket for an additional EPROM is provided to give a maximum of 2K bytes.

Two PPIs (M5L8255AP) are provided on the board to be used as 48-bit programmable I/O ports (8-bit x 6). Basically, they consist of three 8-bit three-state bidirectional buffers: they transfer data according to instruction IN and instruction OUT from the CPU.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

SPECIFICATIONS

Processing Method

Method: 8-bit parallel operation

CPU: M5L8080AP

Word length:

Instruction: 8, 16, 24 bits

Data: 8 bits

Cycle time:

Basic cycle time:: 2μs

CPU clock frequency:

2MHz ±1% (T_a=0～55° C, V_{CC}=5V ±5%)

(Quartz oscillation frequency; 18MHz ±1%)

Memory Address and Memory Capacity

EPROM (M5L2708K)

Memory address:

#1: 0000_{16}～$03FF_{16}$

#2: 0400_{16}～$07FF_{16}$

Memory capacity:

#1: 1K-byte (An EPROM is fitted to the standard product)

#2: 1K-byte (Only a socket is provided on the standard product)

RAM (M5L2111AP x 2)

Memory address:

4000_{16}～$40FF_{16}$

Memory capacity:

256 bytes

Externally expandable up to a maximum of 64K-bytes

I/O Address and I/O Capacity

I/O address:

PPI (M5L8255S x 2):

I/O Port		Signal name	Address
# 1	PA	CP00H～CP07H	00_{16}
	PB	CP10H～CP17H	01_{16}
	PC	CP20H～CP27H	02_{16}
	C.W.	Control word	03_{16}
# 2	PA	CP40H～CP47H	04_{16}
	PB	CP50H～CP57H	05_{16}
	PC	CP60H～CP67H	06_{16}
	C.W.	Control word	07_{16}

As two PPIs (Programmable Peripheral Interfaces) are provided on the board, the PCA0801 has I/O ports of 48 bits (8-bit x 6).

The number of I/O devices can be expanded externally to a maximum of 256.

Interrupt

The PCA0801 incorporates a one-level interrupt function.

The instruction RST 7 is automatically generated by means of the board logic function.

When an external interrupt circuit is to be used, this can be achieved by short-circuiting the interrupt terminals on the board using a jumper wire.

Interface

Bus: all signals TTL compatible

I/O: signals TTL compatible

DMA

Accessible

Connectors

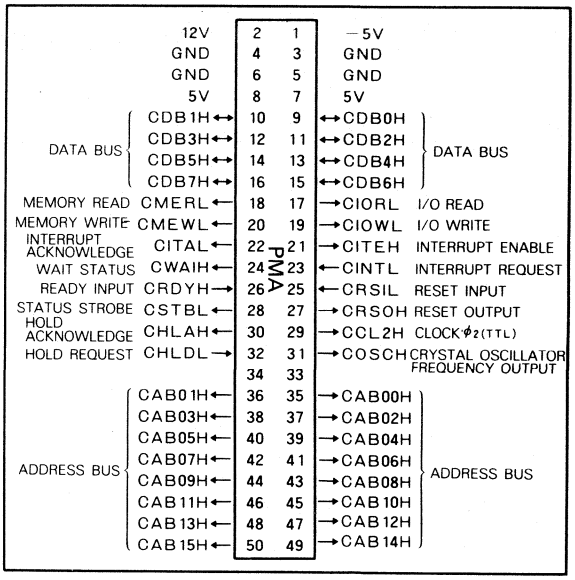
Bus extension connector (connector PMA):

Straight pin header, T type, 50 pins

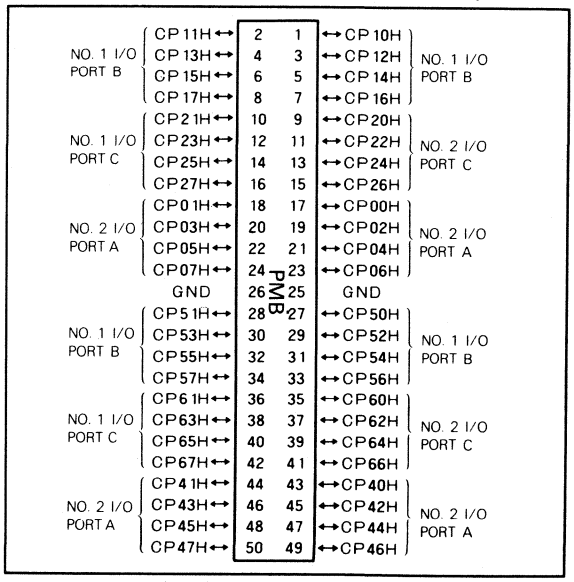
I/O port connector (connector PMB):

Angle pin header, L type, 50 pins

PIN CONFIGURATION (CONNECTOR PMA)



PIN CONFIGURATION (CONNECTOR PMB)



11

PCA0801

MELCS 8/2 SINGLE-BOARD COMPUTER

Physical Dimensions

(L x W x H) 125 x 145 x 17mm

Memory and I/O Address

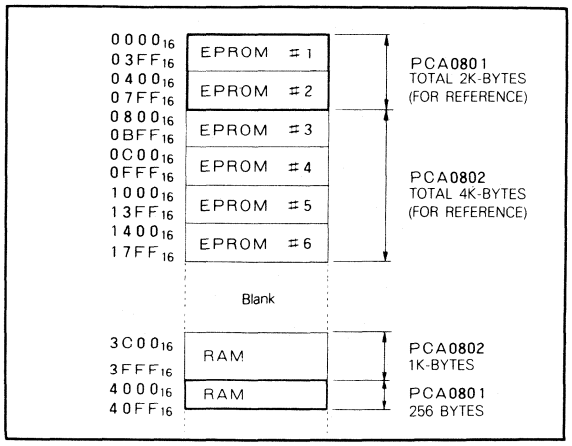
As memory and I/O addresses are fixed in this single-board computer, it is necessary to designate extra addresses besides those already assigned, if any additional external memory or I/O devices are to be employed.

I/O ADDRESS

	PPI #1				PPI #2				PPI #3			
	Port A	Port B	Port C	C.W	Port A	Port B	Port C	C.W	Port A	Port B	Port C	C.W
I/O Address	00_{16}	01_{16}	02_{16}	03_{16}	04_{16}	05_{16}	06_{16}	07_{16}	08_{16}	09_{16}	$0A_{16}$	$0B_{16}$

PCA0801 | PCA0802 (for reference)

MEMORY ADDRESS



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	7	V
V_I	Input voltage		5.5	V
V_O	Output 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

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 55℃, V_{CC} = 5V ± 5%, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
V_{OH}	High-level output voltage	CP00H \ CP07H	I_{OH} = −50μA	2.4			V
V_{OH}	High-level output voltage	CCL2H COSCH CSTBL CMERL CMEWL CIORL CIOWL CITAL CRSON	I_{OH} = −1mA	2.4			V
V_{OH}	High-level output voltage	Others	I_{OH} = −100μA	2.4			V
V_{OL}	Low-level output voltage	CRSOH CSTBL	I_{OL} = 2.5mA			0.5	V
V_{OL}	Low-level output voltage	COSCH CCL2H	I_{OL} = 16mA			0.5	V
V_{OL}	Low-level output voltage	EMERL CMEWL CIORL CIOWL CITAL	I_{OL} = 10mA			0.5	V
V_{OL}	Low level output voltage	Others	I_{OL} = 1.0mA			0.4	V
I_{CC}	Supply current from V_{CC}		With two EPROMs in use		0.8		A

EXPANSION BOARD

The PCA0802 memory and I/O expansion board is provided for expanding memory capacity and I/O ports as follows:
RAM capacity: 1K-byte,
EPROM or mask ROM capacity: 4K-bytes, and
Programmable I/O ports (3 x 8 bits = 24 bits).
Physical dimensions (L x W x H) 125 x 145 x 17mm

MITSUBISHI ELECTRIC

MELCS 8/2 MEMORY AND I/O EXPANSION BOARD

DESCRIPTION

The PCA0802 is a memory and I/O expansion board that is used with the PCA0801 single-board computer to expand the capacity of RAMs, EPROMs and I/Os. It is assembled on a 125 x 145 mm printed circuit board. The PCA0802 can be easily attached to the PCA0801 single-board computer by using the bus-extending connector.

FEATURES

- Memory capacity of RAM: 1K byte
- Memory capacity of EPROM or mask ROM: 4K bytes (max)
- Programmable I/O ports: 48 bits (8-bit x 6)
- Power supply from the PCA0801 single-board computer
- Compact dimensions (L x W x H): 125 x 145 x 17mm

APPLICATIONS

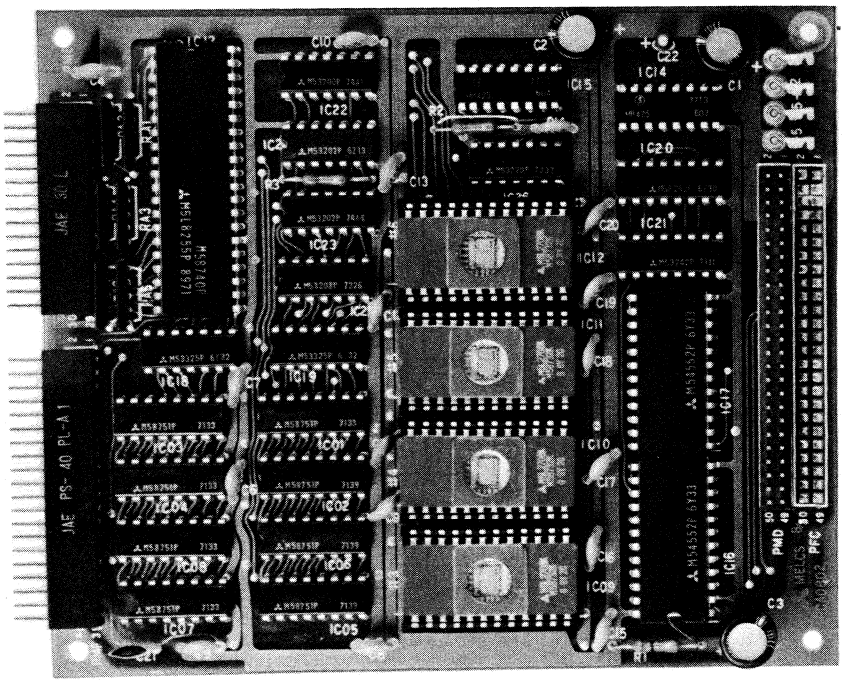
- Personal-computer expansion modules
- Small-size automatic testing or control equipment modules

FUNCTION

The basic PCA0802 memory and I/O expansion board consists of up to 4K bytes of EPROM (M5L2708K × 4; a 1K-byte EPROM is fitted to the standard product, and a further three EPROMs can be plugged into the sockets provided on the board). 1K byte of RAM (M5L2102AP × 8) and a PPI (programmable peripheral interface: M5L8255AP).

The PPI (M5L8255AP) comprises three programmable 8-bit I/O ports, so that it can be used as 24-bit input or output terminals.

PCA0802 memory and I/O expansion board



11

MELCS 8/2 MEMORY AND I/O EXPANSION BOARD

OPERATIONS

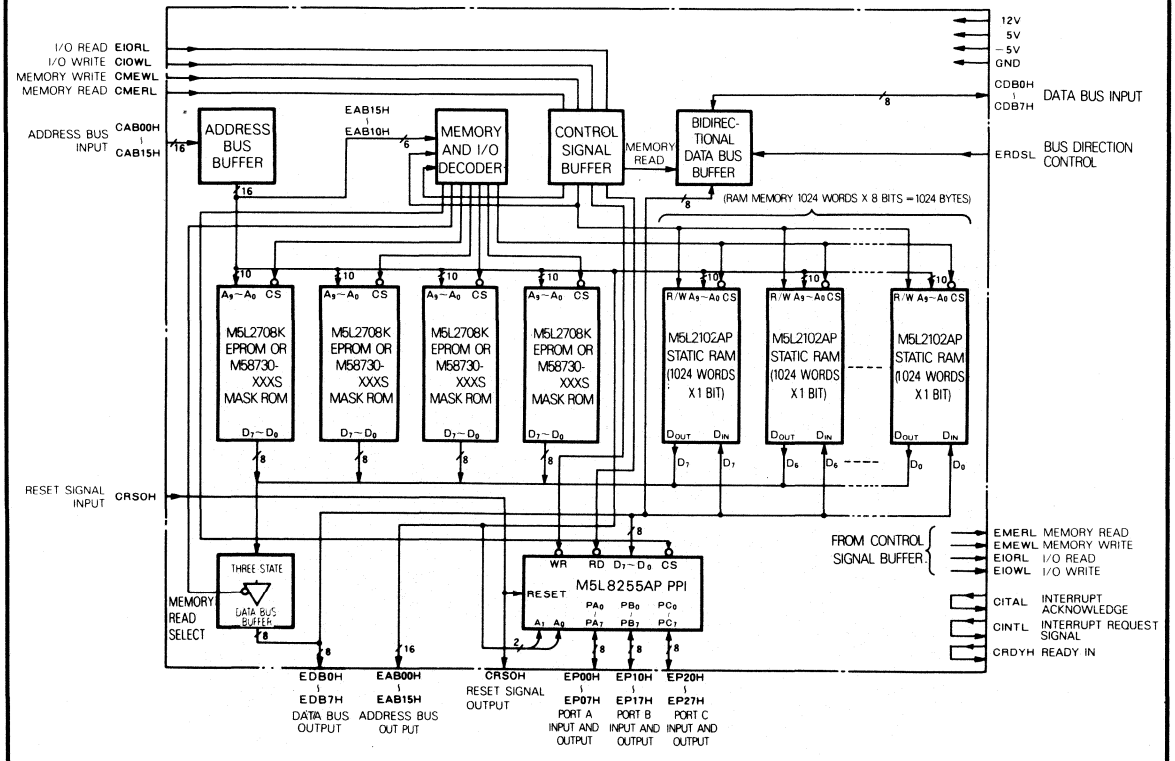
The data bus (CDB0H ~ CDB7H) is connected to each of the data input lines of the memory via the bidirectional data-bus buffer. The signal ERDSL (data-bus control signal) is normally kept at the high-level to maintain the data bus in the input mode. When the low-level signal is applied, the data bus then changes to the output mode.

Memory consists of a maximum of 4K bytes of M5L2708K EPROM (a 1K-byte EPROM is fitted to the standard product, and a further three EPROMs can be plugged into the sockets provided on the board), and 1K byte of M5L2102P static RAMs is installed on the board.

To perform memory read, memory write, I/O read and I/O write operations, the memory and I/O address from the address bus is first decoded by the memory and I/O decoder, and then the memory read/write and I/O read/write signals are sent out from the control signal buffer by CMERL, OMEWL, CIORL and CIOWL control signals generated by the PCA0801 single board computer.

One M5L8255AP PPI is mounted on the expansion board and is used as 24-bit programmable I/O ports. Fundamentally these I/O ports are three units of 8-bit three-state bidirectional buffer, and they function to transfer data in accordance with instructions IN and OUT from the CPU.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

SPECIFICATIONS

Memory Address and Memory Capacity

EPROM (M5L2708K)

Memory allocation:

3: $0800_{16} \sim 0BFF_{16}$
4: $0C00_{16} \sim 0FFF_{16}$
5: $1000_{16} \sim 13FF^{16}$
6: $1400_{16} \sim 17FF_{16}$

Memory capacity:

3: 1K byte (an EPROM fitted to the standard product)
4: 1K byte (socket provided on the standard product)
5: 1K byte (socket provided on the standard product)
6: 1K byte (socket provided on the standard product)
Total: 4K bytes

RAM (M5L2102AP x 8)

Memory allocation:

$3C00_{16}$ $3FFF_{16}$

Memory capacity:

1K byte, expandable up to the maximum 64K bytes

I/O Address and I/O Capacity

I/O address

PPI (M5L8255AP) :

I/O port	Signal name	Address
PA	EP00H~EP07H	08_{16}
PB	EP10H~EP17H	09_{16}
PC	EP20H~EP27H	$0A_{16}$
C.W.	Control word	$0B_{16}$

One PPI is mounted on the board providing three 8-bit programmable I/O ports, which permits up to 256 I/O devices to be connected externally.

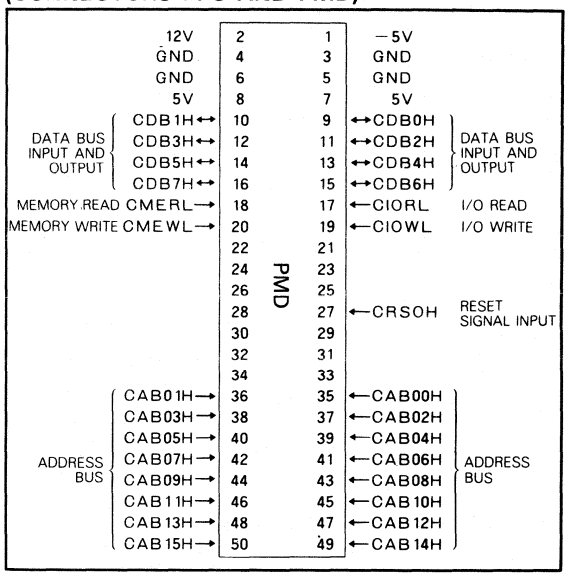
Interface

Bus: all signals TTL compatible
I/O: all signals TTL compatible

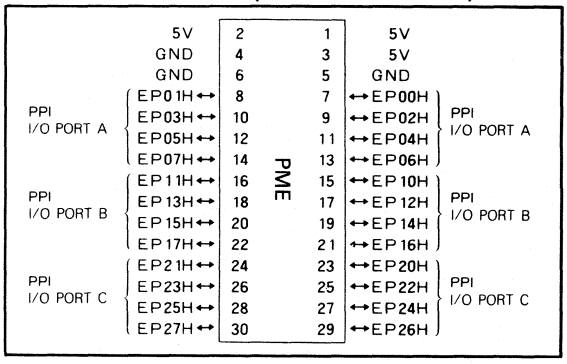
Connectors

For bus connection (connector PFC):
Straight dip type, 50 pins
For bus connection (connector PMD):
Straight pin header, T type, 50 pins
For I/O ports (connector PME):
Angle pin header, L type, 30 pins
For debugging (connector PMF):
Angle pin header, L type, 40 pins

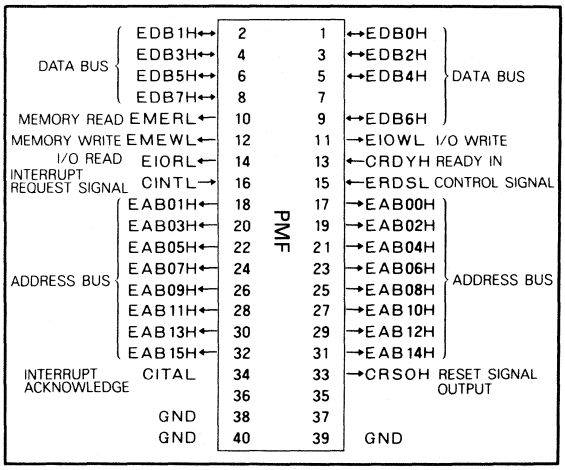
PIN CONFIGURATION (CONNECTORS PFC AND PMD)



PIN CONFIGURATION (CONNECTOR PME)



PIN CONFIGURATION (CONNECTOR PFC)



11

MELCS 8/2 MEMORY AND I/O EXPANSION BOARD

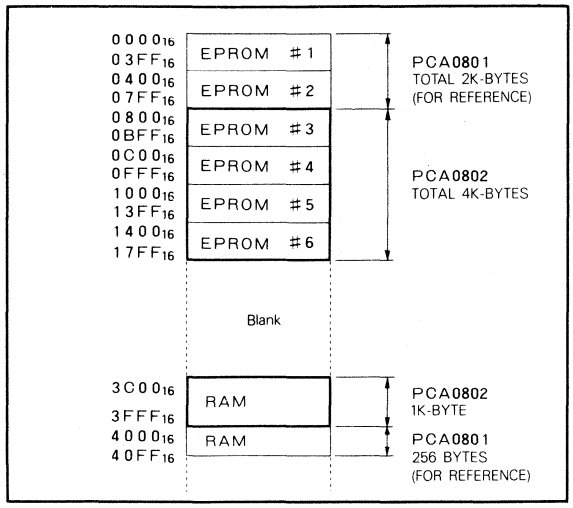
Physical Dimensions
(L x W x H): 125 x 145 x 17mm

Memory and I/O Addresses
As memory and I/O addresses are fixed in this expansion board, 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				PPI #3			
	Port A	Port B	Port C	C.W	Port A	Port B	Port C	C.W	Port A	Port B	Port C	C.W
I/O Address	00_{16}	01_{16}	02_{16}	03_{16}	04_{16}	05_{16}	06_{16}	07_{16}	08_{16}	09_{16}	$0A_{16}$	$0B_{16}$
	PCA0801 (For reference)								PCA0802			

MEMORY ADDRESS



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{CC}	Supply voltage	With respect to GND	7	V
V_{BB}	Supply voltage	With respect to GND	−15~0.3	V
V_{DD}	Supply voltage	With respect to V_{BB}	−0.3~20	V
V_I	Input voltage	With respect to GND	5.5	V
V_O	Output voltage	With high-level output	V_{CC}	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_{BB}	Supply voltage	−5.25	−5	−4.75	V
V_{DD}	Supply voltage	1.4	12	12.6	V
V_{IH}	High-level input voltage	3		V_{CC}	V
V_{IL}	Low-level input voltage	0		0.65	V

ELECTRICAL CHARACTERISTICS (Ta = 0~55℃, V_{CC} = 5V ± 5%, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
V_{OH}	High-level output voltage	CDB7H~CDB0H	I_{OH} = −10mA	2.4	3		V
V_{OH}	High-level output voltage	EMERL, EIORL	I_{OH} = −740μA				V
V_{OH}	High-level output voltage	EMEWL, EIOWL	I_{OH} = −800μA	2.4			V
V_{OH}	High-level output voltage	CRSOH	I_{OH} = −1mA	2.4			V
V_{OH}	High-level output voltage	EP27H ~ EP00H	I_{OH} = −50μA	2.4			V
V_{OH}	High-level output voltage	EDB7H~EDB0H	I_{OH} = −1mA	3.65	4		V
V_{OH}	High-level output voltage	EAB15H~EAB00H	I_{OH} = 0.94mA	3.65			V
V_{OL}	Low-level output voltage	CDB7H~CDB0H	I_{OH} = 25mA		0.3	0.45	V
V_{OL}	Low-level output voltage	EMERL, EIORL	I_{OH} = 14.4mA		0.22	0.4	V
V_{OL}	Low-level output voltage	EMEWL, EIOWL	I_{OH} = 16mA		0.22	0.4	V
V_{OL}	Low-level output voltage	CRSOH	I_{OH} = 2.5mA			0.5	V
V_{OL}	Low-level output voltage	EP27H~EP00H	I_{OH} = 1.6mA			0.4	V
V_{OL}	Low-level output voltage	EDB7H~EDB0H	I_{OH} = 15mA		0.3	0.45	V
V_{OL}	Low-level output voltage	EAB15H~EAB00H	I_{OH} = 14.4mA			0.5	V

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

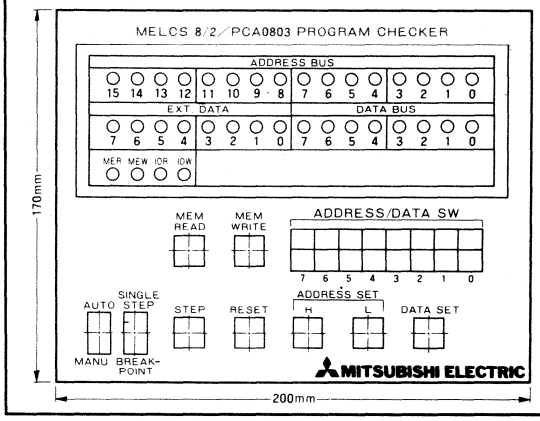
APPLICATIONS

- For design evaluation of equipment embodying the PCA0801 single-board computer and the PCA0802 memory and I/O expansion board, as well as for field system troubleshooting.

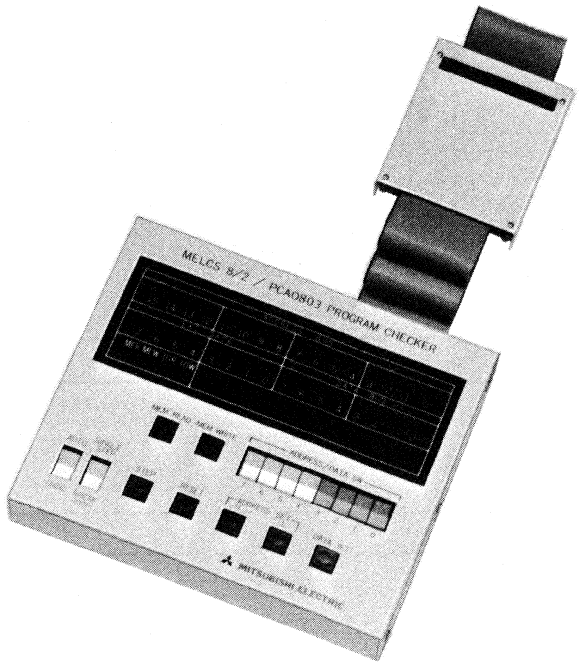
FUNCTION

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



11

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

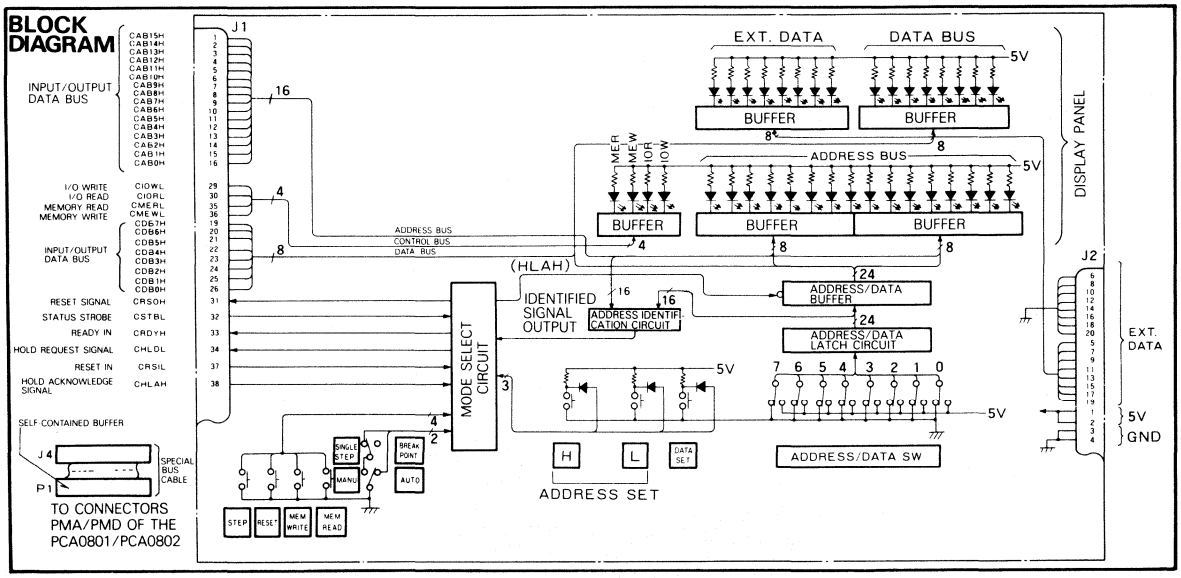
Each time this key is depressed, it executes one program step.

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. Model 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

PCA0804G01, G02

MELCS 8/2 COLOR TV DISPLAY SINGLE-BOARD COMPUTER

DESCRIPTION

The PCA0804 is a single-board computer that is composed of the MELPS 8 LSI family and TV interface circuits and fabricated on a single 125 × 145mm printed circuit board. It enables 8-color 64 × 64 dot matrix color display on a TV screen just by connecting it to the antenna terminal of a home-use color TV receiver. In addition to the board computer itself, there is a kit that contains a keyboard; one M5L 2708K EPROM, in which a monitor program and character display subroutines are stored; and one blank M5L 2708K EPROM for the user's program storage.

FEATURES

Type	Contents	
PCA0804G01	Single-board computer only	
PCA0804G02	PCA0804G01 single-board computer.	1 pc
	PCA0805 keyboard switch.	1 pc
	M5L 2708K (005) EPROM with monitor program and character or dot-line display subroutine stored.	1 pc
	User blank M5L 2708K EPROM for user's program storage:	1 pc
	Instruction manual:	1 pc

- A single-board computer complete with CPU, memory, I/O and TV interfaces.
- Enables 8-color 64 x 64 dot matrix color display on a screen of a home-use color TV receiver.
- Capacity of EPROM or mask ROM: 2K bytes (max)
- Capacity of RAM: 1K bytes
- Programmable I/O port: 24 bits (3 × 8-bit)
- Enables frame interrupt (per each frame sweep)
- Compact dimensions (L x W x H): 124 x 145 x 30mm

APPLICATIONS

- TV game machines
- Personal computers
- Simple color graphic display
- Display terminals for equipment using microcomputers
- Store-front commercial display
- Slave computer for the MELCS 8/2

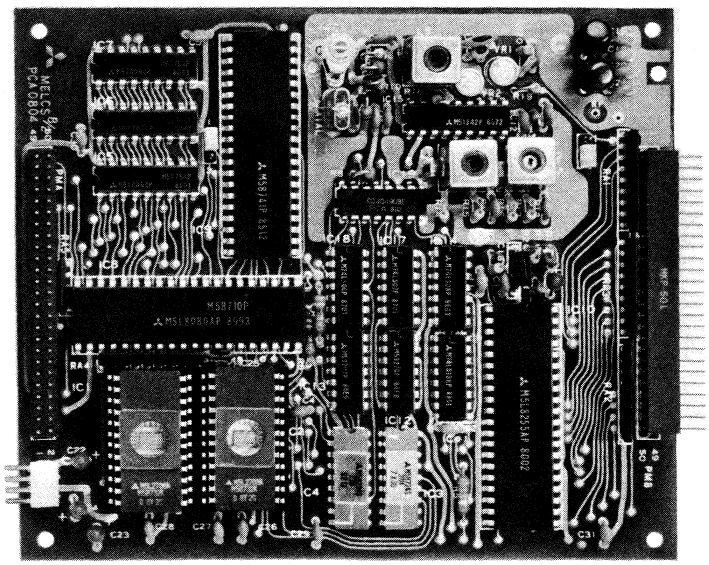
FUNCTION

The PCA0804 is a single-board computer with TV display functions designed around Mitsubishi's M5L8080AP CPU, its LSI family and color LSI M58741P TV interface. The PCA0804 comes with 2K bytes of electrically programmable read-only memory (EPROM) in the form of two M5L 2708Ks and 1K bytes of random-access memory in the form of two M5L 2114LPs.

For its I/O ports, the PCA0804 has one M5L 8255AP programmable peripheral interface (PPI), providing 8 bits × 3 = 24-bit programmable I/O ports.

The TV interface consists of the M58741 TV interface LSI and the M51342P RF modulator IC, and three M5T 4044P 4K static RAMs are employed for the screen display memory.

The 64 × 64 dot matrix color image can be displayed on a screen by feeding the RF signal from the board to the antenna terminal of a home-use color TV receiver.



11

MELCS 8/2 COLOR TV DISPLAY SINGLE-BOARD COMPUTER

OPERATIONS

The M5L8080P CPU executes programs stored in the M5L2708K ROM or M5L2114LP RAM. Data transaction with the external source is carried out through the PPI M5L8255AP.

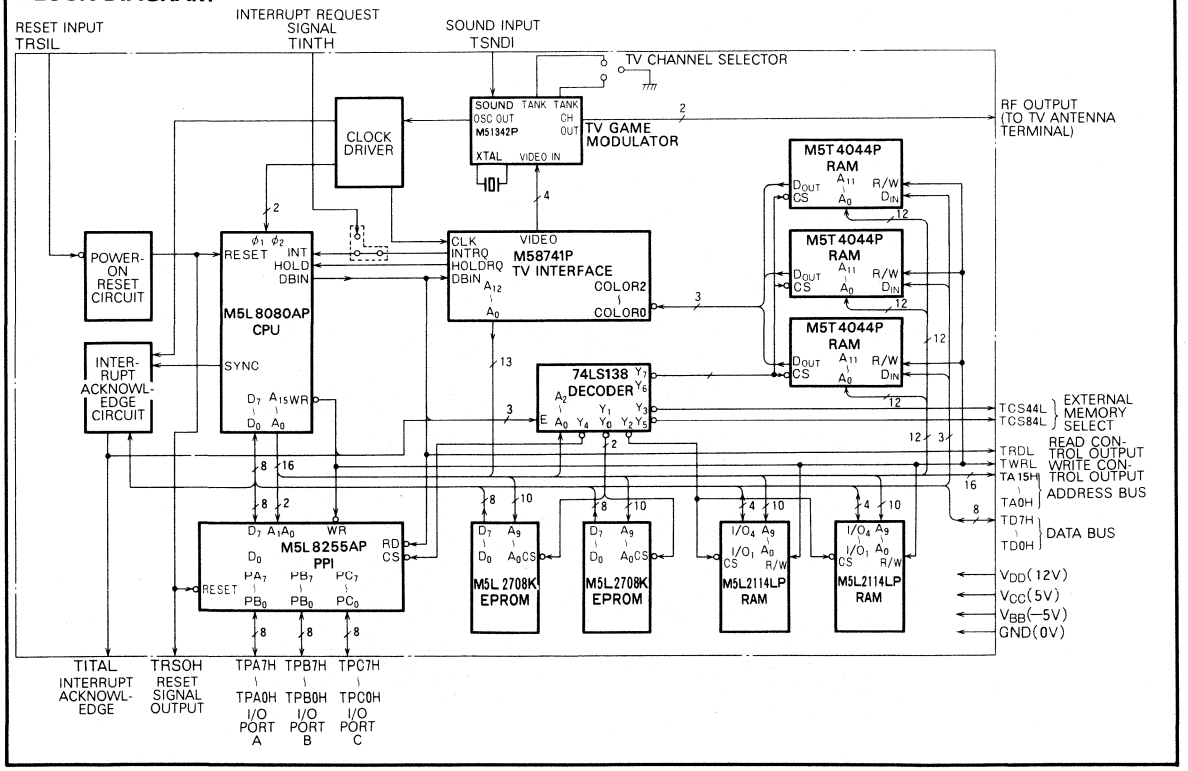
Three M5T4044P RAMs which are employed for the screen display can be used as a simple memory to store the display data, perfectly independent from the CPU operation. As the CPU comes to the hold state after the TV interface M58741P has generated a hold request signal to the CPU synchronizing with the video synchronous pulses, it reads out the contents of the screen display RAM sequentially by controlling the address bus and control bus.

The data thus fetched from the screen display RAMs is combined and converted into the image signal by the TV interface. This screen signal is applied to the RF modulator M51342P and is modulated into the TV wave of either channel 1 or channel 2.

SPECIFICATIONS

Item	Description
Method	8-bit parallel operation
CPU component	Mitsubishi's M5L8080AP (equivalent to Intel's 8080A)
Cycle time	Basic cycle time 2.23μs (at clock frequency 1.79MHz)
Memory	EPROMs Capacity: 2K bytes (M5L2708K × 2) Address: $000016 \sim 07FF16$ RAMs Capacity: 1K bytes (M5L2114LP × 2) Address: $400016 \sim 43FF16$ Screen display memory Capacity: 4K × 3 bits (M5T 4044P × 3) Address: $C00016 \sim FFFF16$
I/O	Programmable port Capacity: 8-bit × 3 (PPI, M5L8255AP) Address: $800016 \sim 800316$ (memory mapped I/O)
Video output	NTSC system, Japan channel CH 1 or CH 2
Display method	64 × 64 dot matrix, in color (black + 7 other colors)
Interrupt	Per each frame sweep or external interrupt
Power supply	12V, 5V, −5V
Applicable connector	Straight pin header, T-type, 50 pins for bus extension. Angle pin header, L-type, 50 pins for port connection.
Physical dimensions	(L × W × H) 124 × 145 × 30mm

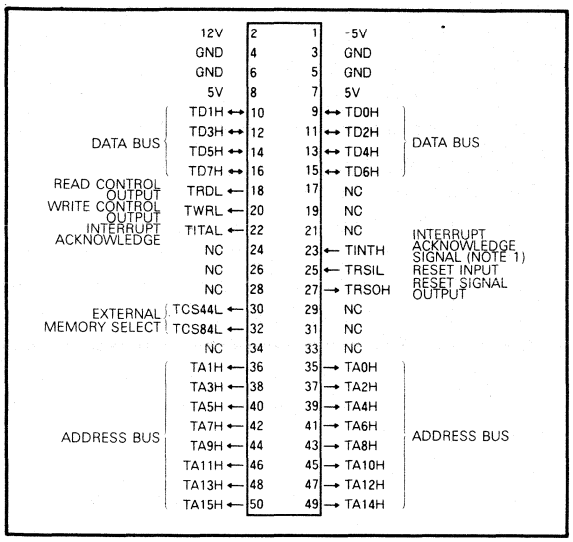
BLOCK DIAGRAM



MITSUBISHI ELECTRIC

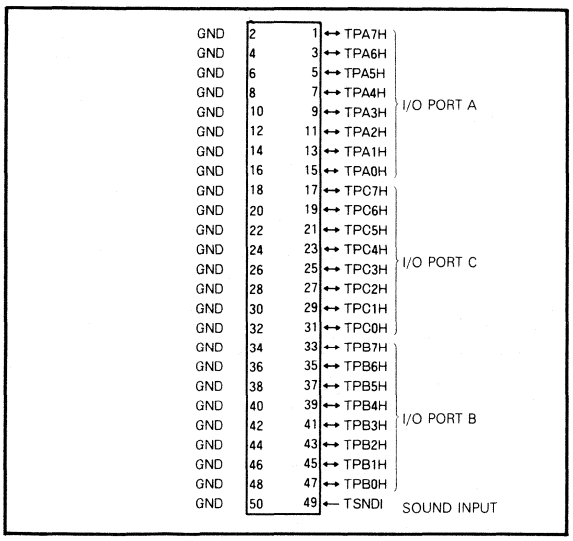
PIN CONFIGURATIONS

Connector RMA

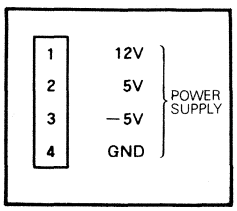


Note 1 : This signal is only effective when it is connected with pin 14 (INT) signal) of the CPU M5L 8080AP by an inline connector.
2 : NC indicates no connection

Connector PMB



Connector PMC



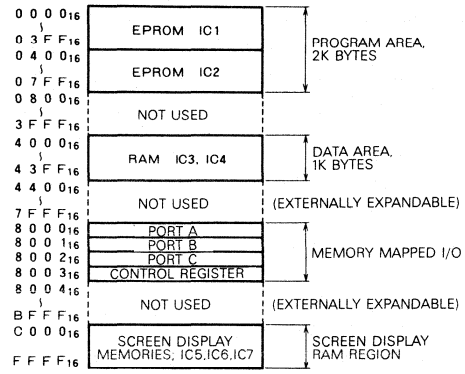
PROGRAM EXAMPLE

```
MVI  A,07#
STA  C000#
MVI  A,00#
STA  C001#
```

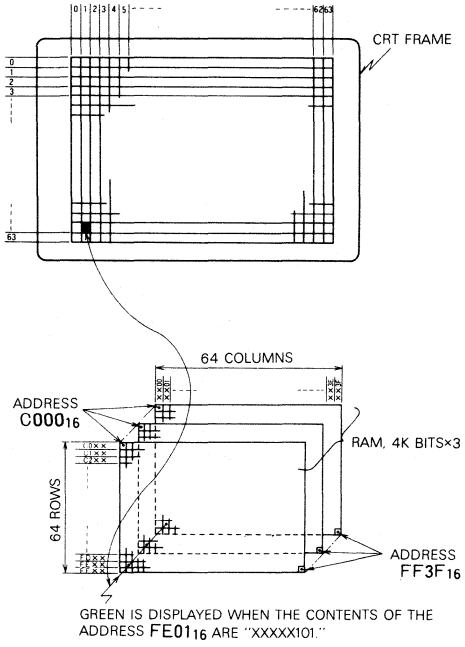
UPPER LEFT CORNER OF THE SCREEN : MAGENTA

RIGHT OF THE ABOVE : BLACK

MEMORY ADDRESS MAP



INTERRELATION OF SCREEN DISPLAY MEMORY WITH THE COORDINATE



GREEN IS DISPLAYED WHEN THE CONTENTS OF THE ADDRESS $FE01_{16}$ ARE "XXXXX101."

COLOR CODE DESIGNATION

D2	D1	D0	Color
1	1	1	Magenta
1	1	0	Red
1	0	1	Green
1	0	0	White
0	1	1	Orange
0	1	0	Cyan
0	0	1	Blue cyan
0	0	0	Black

11

MELCS 8/2 COLOR TV DISPLAY SINGLE-BOARD COMPUTER

PCA0804G02 FUNCTIONAL SUMMARY

The PCA0804G02 consists of a PCA0804G01 single-board computer, a keyboard and one M5L 2708K in which are contained a monitor program and character or dot-line display subroutines. These capabilities offer very convenient means of programming and picture generation.

1. Keyboard and Monitor Program

The keys have the following functions, as the arrangement in the block diagram shows:

(1) EXM (EXAMINE):
This key displays the set address and its designated contents.

(2) ENXT (EXAMINE NEXT):
This key displays the set address and its designated contents sequentially.

(3) DEP (DEPOSIT):
An instruction or data which is set from the keyboard is written to the address designated by the EXM key.

(4) START:
Depression of this key starts program execution from the set address.

(5) RESET:
Depression of this key sends out a reset signal to the CPU and the program returns to its initial state.

With above procedures, a color image is displayed on a TV screen. Successive depression of the EXAM, ENXT and DEP keys makes relevant five addresses and their contents are displayed on the screen.

2. Line or Block Display Subroutines

If the top address, its color and length of a display pattern are set to the register in the CPU before calling the following subroutines, it allows easy generation of specific patterns.

(1) TATE: Horizontal line at desired location, color and length.

(2) YOKO: Vertical line at desired location, color and length.

(3) MEN: Rectangle at desired location, color and dimensions.

Use of these subroutines makes generation of complicated patterns much easier.

3. Character Display Subroutine

When a specific character-displaying position, its color, number of characters, and their character codes are set in a certain RAM area, those specific characters can be displayed easily by calling this subroutine.

The number of characters applicable on a display frame is 10 characters × 6 lines with 8 colors (black + 7 other colors) selective. The following 44 characters and symbols are applicable:

0 1 2 3 4 5 6 7 8 9
A B C D E F G H I J K L M N O P Q R S T U V W X
Y Z + − = (blank) . , ? /

4. An Example of Typical PCA0804G02 Operation

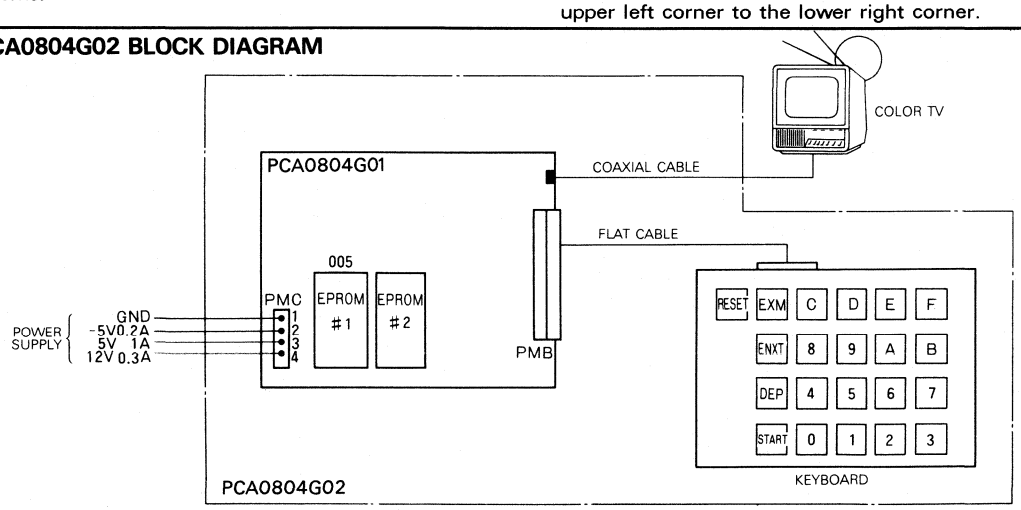
A simple program is stored in the RAM area starting from the address 4000_{16}, and then the program is executed.

(1) Example Program

Address		Mnemonic		Machine code
4000	DEMO	LXI	H, COCO#	21 CO CO
4003		XRA	A	AF
4004	DEM*	MOV	M, A	77
4005		INR	A	3C
4006		INR	H	24
4007		INR	L	2C
4008		JNZ	DEM*	C3 04 40
400B		HLT		76

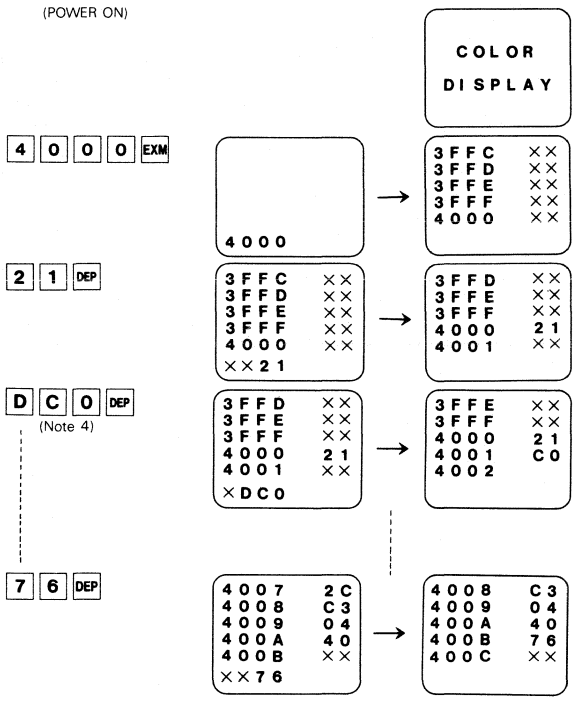
This example program displays a diagonal line from the upper left corner to the lower right corner.

PCA0804G02 BLOCK DIAGRAM



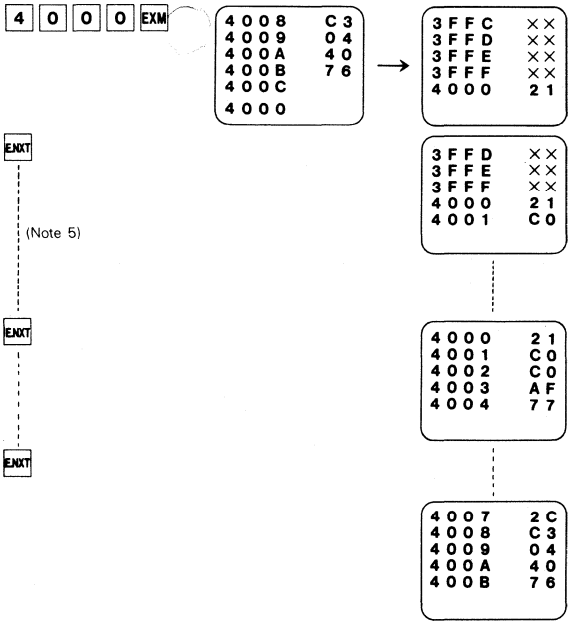
Note 3 : #1 M5L 2708K (005): Contains the monitor program, and line, block, and character display program.
#2 M5L 2708K: Blank EPROM for user's program storage.

(2) A program is written from the address 4000_{16}.



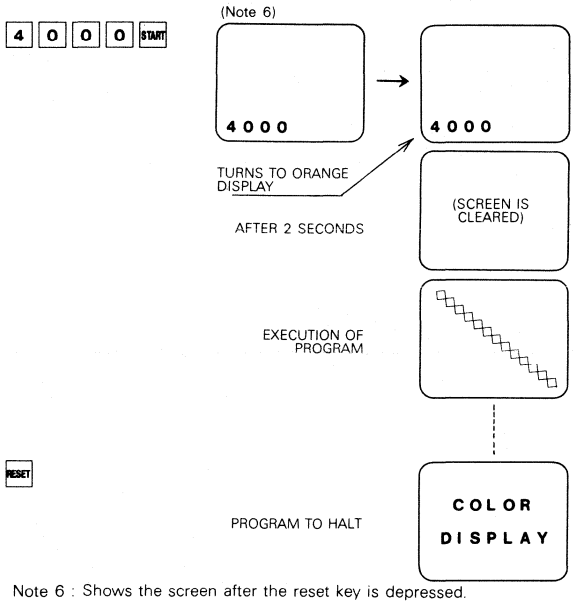
Note 4 : Ignore the error data and retain continue entry.

(3) The content of program is checked from the address



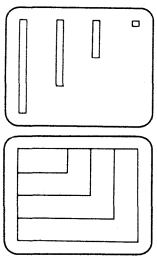
Note 5 : Check contents of program in succession by referring to the program data on the screen, and correct the error data by depressing [X] [X] [DEP]

(4) The program is started from the address 4000_{16}.

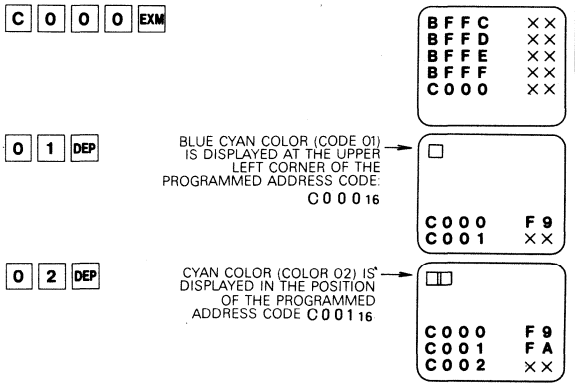


Note 6 : Shows the screen after the reset key is depressed.

(5) Example of display using the line and block display subroutine.
Various complicated patterns can be generated easily by using the subroutine stored in the M5L2708K (005).



(6) Writing data directly into the screen memory area ($C000_{16}$~$FFFF_{16}$).



11

MELCS 8/2 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_{DD}	Supply voltage		−0.3～20	V
V_I	Input voltage		5.5	V
Topr	Operational free-air ambient temperature range		5～40	℃
Tstg	Storage temperature range		−10～70	℃

Note 7 : Basically the power should be applied in the sequence of $V_{BB} \rightarrow V_{CC} \rightarrow V_{DD}$, and turned off in the reverse sequence.

RECOMMENDED OPERATING CONDITIONS (T_a = 5～40℃, 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
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 (T_a = 25℃, V_{CC} = 5V ± 5%, V_{BB} = −5V ± 5%, V_{DD} = 12V ± 5%, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage, TITAL, TCS4L, TCS8L, TRSOH TRDL, TWRL	I_{OH} = −300μA	2.4			V
V_{OH}	High-level output voltage, others	I_{OH} = −100μA	2.4			V
V_{OL}	Low-level output voltage, TITAL, TCS4L, TCS8L, TRSOH TRDL	I_{OL} = 10mA			0.4	V
V_{OL}	Low-level output voltage, others	I_{OL} = 1.6mA			0.4	V
I_{CC}	V_{CC} supply current	When two M5L 2708K EPROMs are un use			1	A
I_{BB}	V_{BB} supply current				0.2	A
I_{DD}	V_{DD} supply current				0.3	A
fr	CPU clock frequency			1.79		MHz

PCA8501G01, G02

MELPS 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

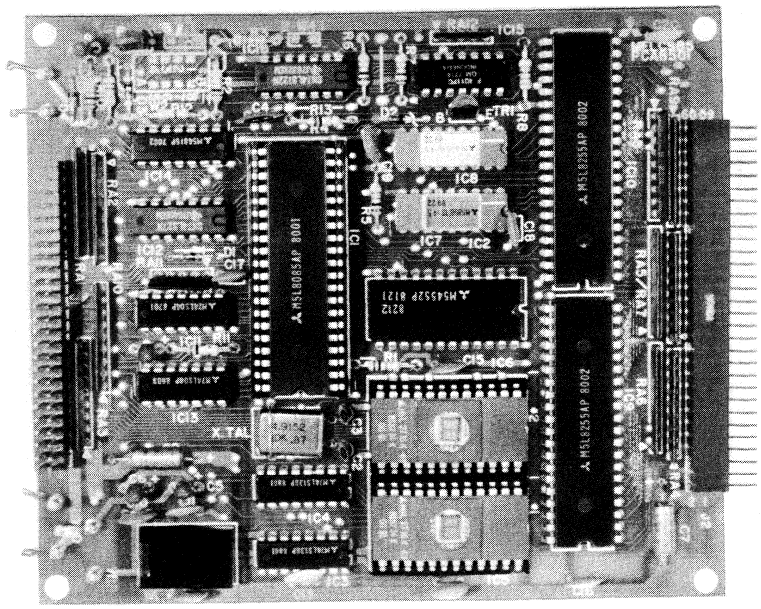
- Personal computers
- Small automatic testing or control equipment
- Data-communication terminal equipment
- Data loggers and data-collection equipment
- Process-control equipment
- Instrument monitoring controllers

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.



11

MELPS 85/2 SINGLE-BOARD COMPUTER

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.

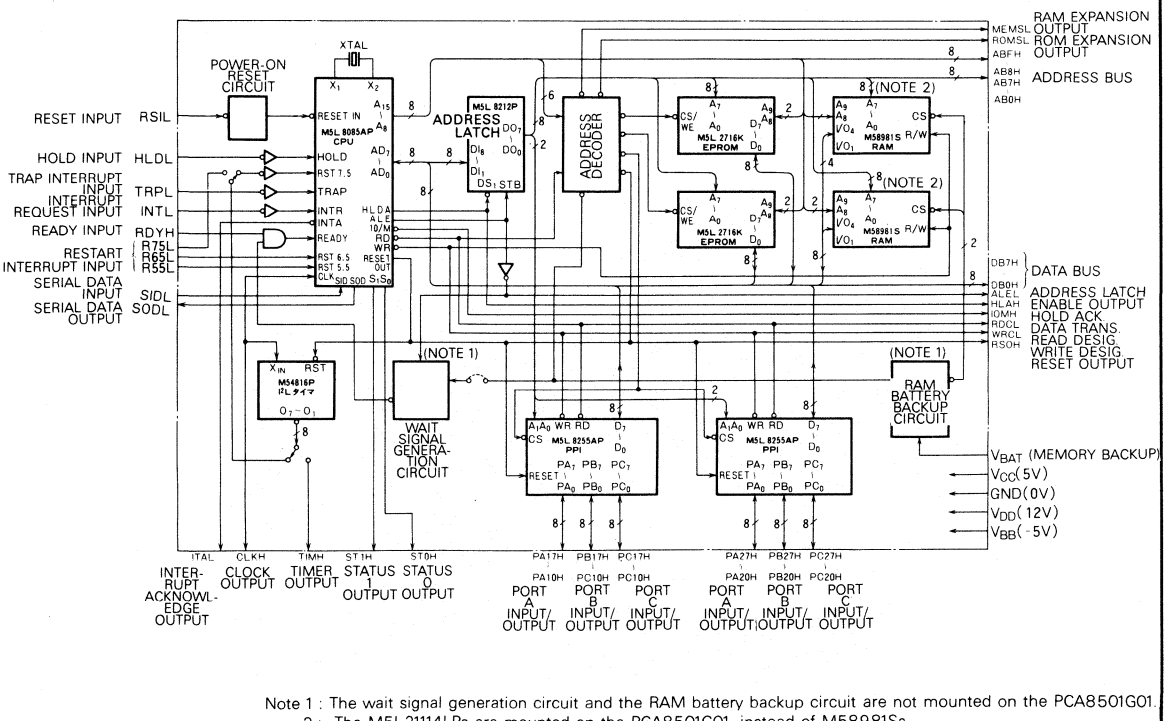
Parallel data can be read/written through the PPIs, and serial data through the SID and SOD of the M5L8085AP CPU.

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: 1.6 μs

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}~$OFFF_{16}$

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):

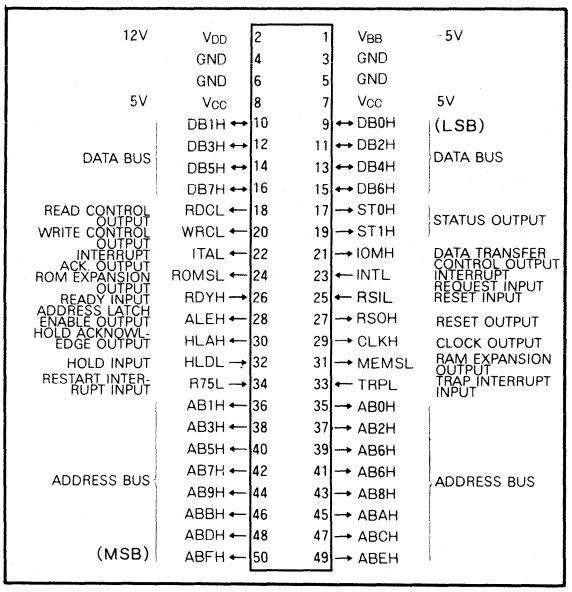
Straight pin header, T-type, 50 pins

For I/O port connection (connector J2):

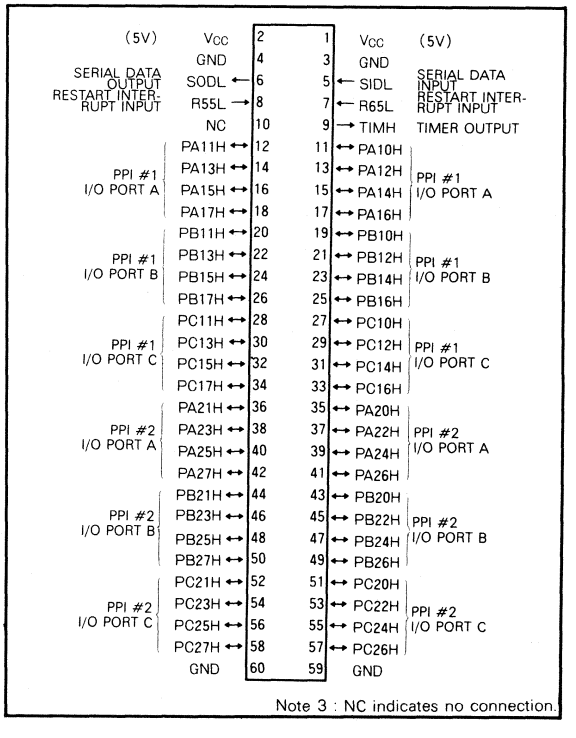
Angle pin header, L-type, 60 pins

PIN CONFIGURATIONS

Connector J1



Connector J2



Note 3 : NC indicates no connection.

11

MELPS 85/2 SINGLE-BOARD COMPUTER

Memory and I/O Addresses

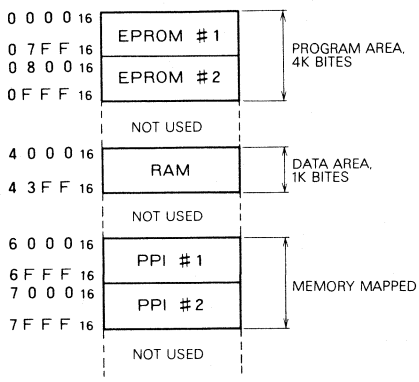
As memory and I/O addresses are fixed in this single-board computer, it is necessary to designate extra addresses besides those already assigned, if any additional external memory or I/O devices are to be employed.

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
T_{opr}	Operating free-air ambient temperature range		0~55	℃
T_{stg}	Storage temperature range		−30~70	℃

RECOMMENDED OPERATING CONDITIONS (T_a=0~55℃, 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
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 (T_a=0~55℃, V_{CC}=5V±5%, unless otherwise noted.)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{OH}	High-level output voltage, PA11H~PC26H outputs	I_{OH} = −50μA	2.4			V
V_{OH}	High-level output voltage, AB0H~AB7H outputs	I_{OH} = −900μA	3.65			V
V_{OH}	High-level output voltage, CLKH output	I_{OH} = −300μA	2.4			V
V_{OH}	High-level output voltage, MEMSL output and ROMSL output	I_{OH} = −300μA	2.4			V
V_{OH}	High-level output voltage, other outputs	I_{OH} = −400μA	2.4			V
V_{OL}	Low-level output voltage, PA11H~PC26H 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, CLKH output	I_{OL} = 1.9mA			0.45	V
V_{OL}	Low-level output voltage, MEMSL output and ROMSL 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

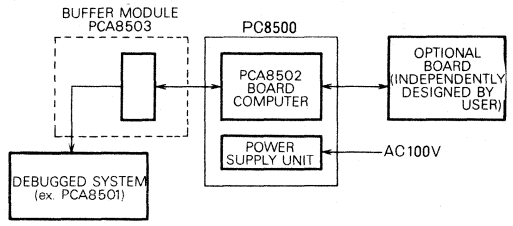
MITSUBISHI ELECTRIC

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).

11

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

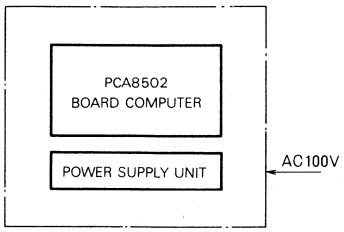
The PCA8502 offers 1K bytes of EPROM and 4K bytes of RAM and also releases the M5L8255AP PPI (8-bit × 3 programmable I/O ports) for a parallel I/O interface.

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

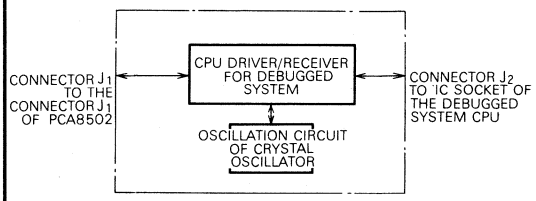
The PCA8503 is a buffer module employed in interfacing the PC8500 (PCA8502) with a user system (debugged board), as shown in the block diagram, and supplied as an optional board.

BLOCK DIAGRAMS

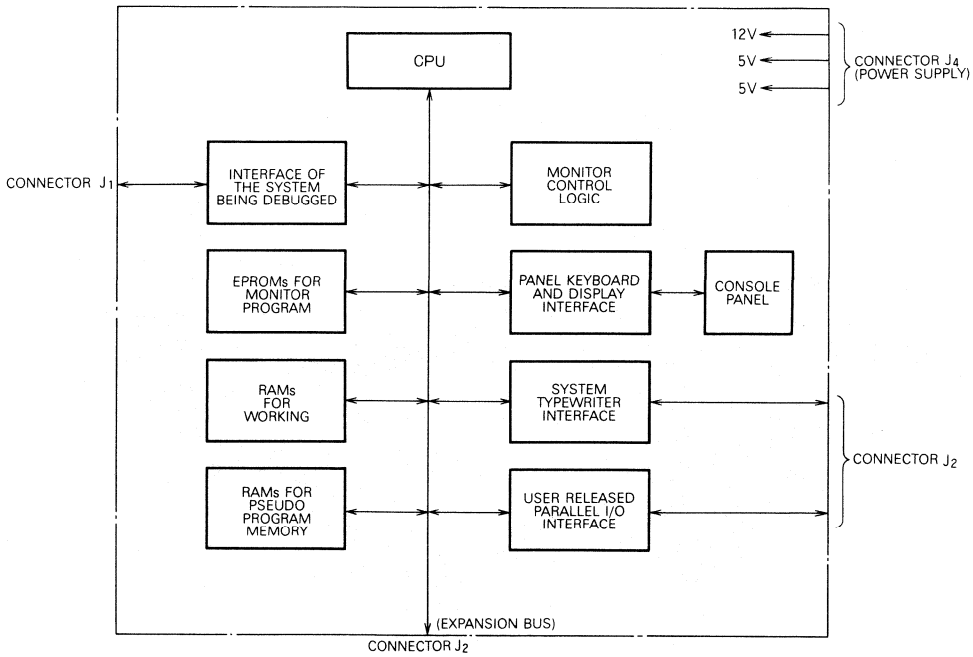
PC8500



PCA8503



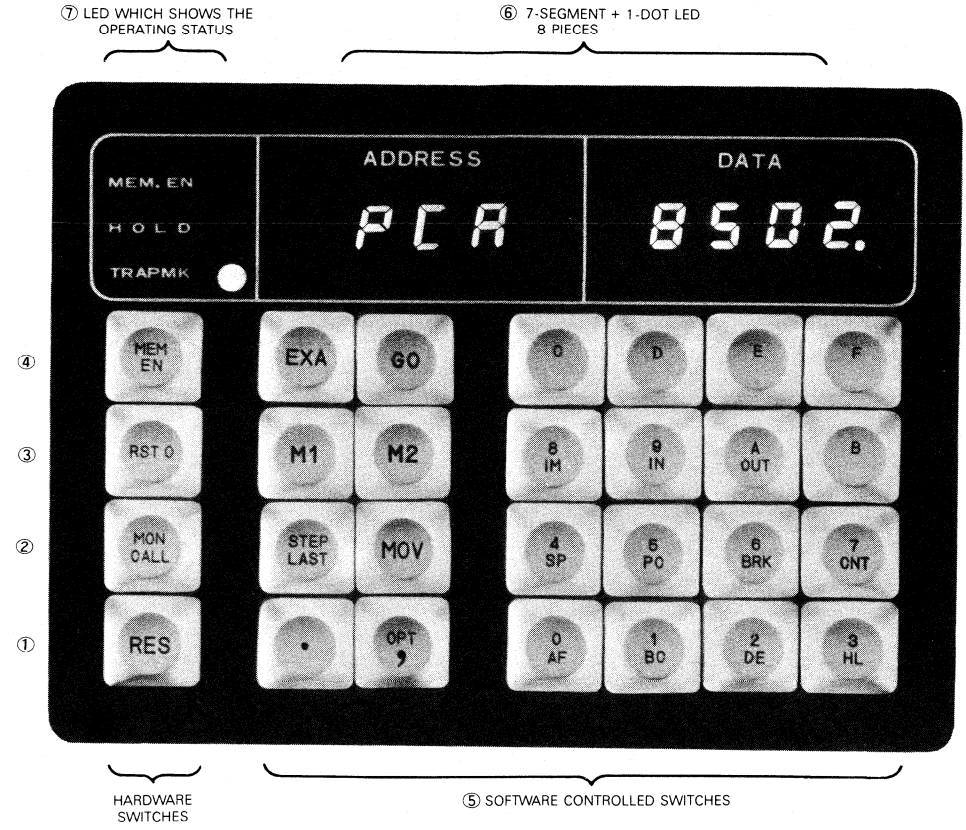
PCA8502 BLOCK DIAGRAM



MITSUBISHI ELECTRIC

MELCS 85/1 PORTABLE MICROCOMPUTER CONSOLE

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.

11

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: $F800_{16}$ ~ $FFFF_{16}$ = 2K bytes for a monitor program ROM: $F400_{16}$ ~ $F7FF_{16}$ = 1K bytes for user released area RAM: $F300_{16}$ ~ $F3FF_{16}$ = 256 bytes for monitor used area Inhibited area: $F000_{16}$ ~ $F2FF_{16}$
	User released area	RAM: $*000_{16}$ ~ $*FFF_{16}$, max 4K bytes. Where * indicates any number from 0_{16} ~ 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: $F8_{16}$ ~ FB_{16}: system control interface I/O address of the area, $F0_{16}$ ~ 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} ~ 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)
	CPU clock	Can be operated with clock from the user's system (3.125MHz max).
	Interface with user system	To be connected with the IC socket of the CPU of user's system through the buffer module (PCA8503)
	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: $F000_{16}$ ~ $FFFF_{16}$ I/O address area: $F0_{16}$ ~ FF_{16}
	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.
	Pseudo program memory	It is possible to substitute the address area 0000_{16} ~ $0FFF_{16}$ of the user's system with the RAM in the debugging system.

Item		Description
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

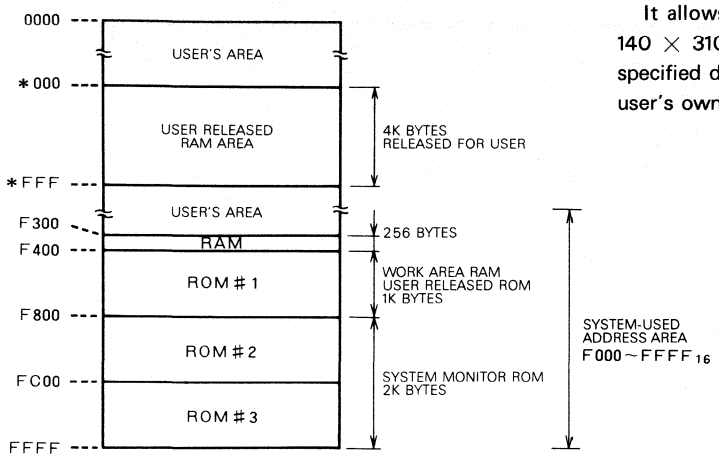
MITSUBISHI ELECTRIC

SYSTEM ADDRESS AREAS

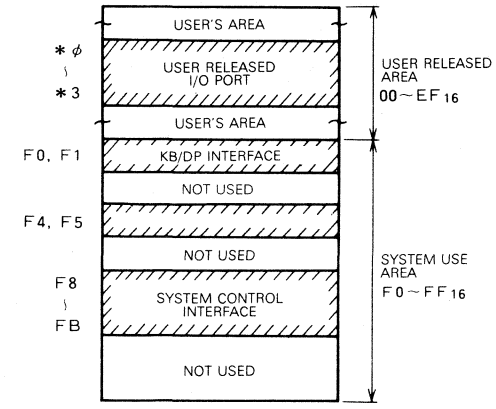
Among the address areas used by the system, the memory addresses 0000_{16}~$EFFF_{16}$ and the I/O device addresses 00_{16}~EF_{16} are all released for the user, and the rest of the areas are used by the system. So the user should stay within the prescribed areas.

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.

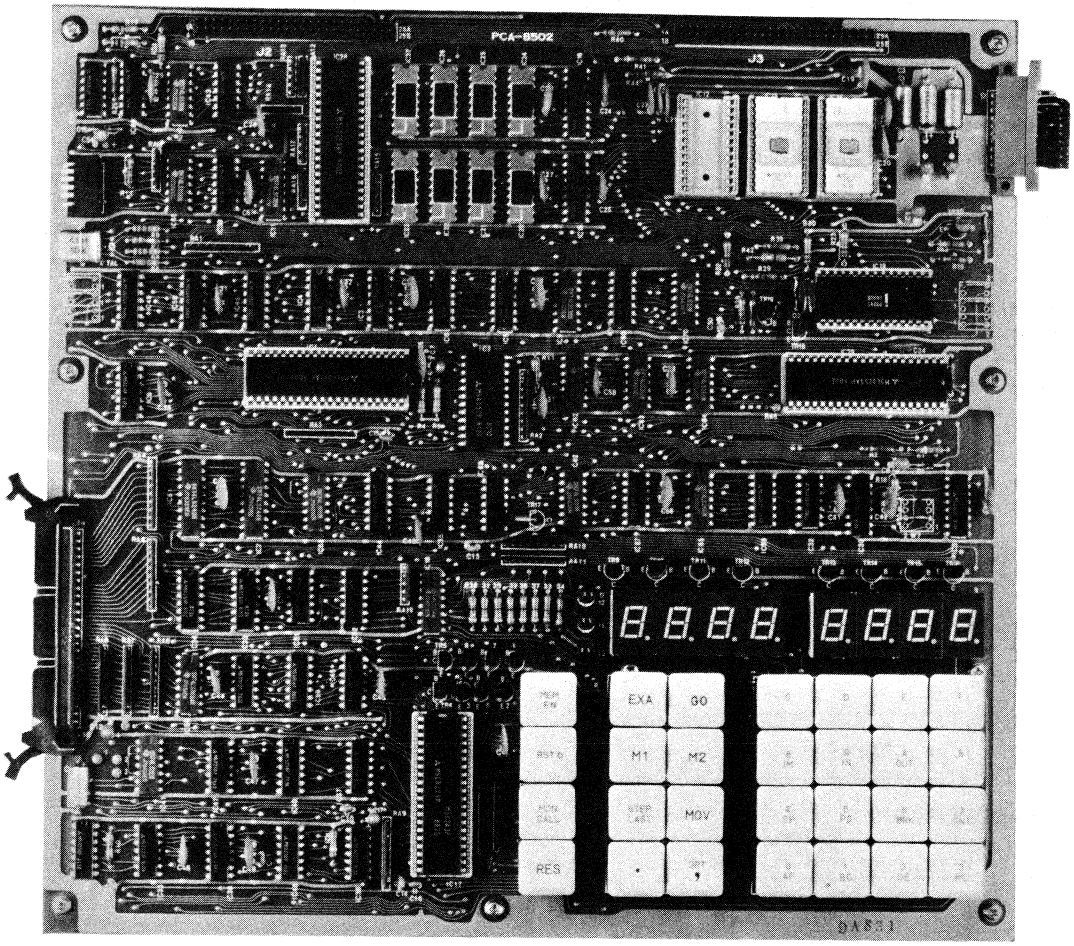
OPTIONAL BOARD

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.

MELCS 85/1 PORTABLE MICROCOMPUTER CONSOLE

PCA8502



MITSUBISHI ELECTRIC

MICROCOMPUTER SUPPORT SYSTEMS

12

PCA0401

MELCS 4 SINGLE-BOARD SYSTEM-EVALUATION COMPUTER

DESCRIPTION

The PCA0401 is a single-board system-evaluation computer for the M58840-XXXP single-chip 4-bit microcomputer. It is designed around the M58842S MELPS 4 system evaluation device and M5L2708K, electrically erasable and re-programmable ROMs and is similar in function to the M5880-XXXP,which contains a mask-programmable ROM. The PCA0401 is suitable for debugging, correcting and testing the application program in development before masking it.

FEATURES

- Single-board computer equivalent in function to the M58840-XXXP
- Easy program alteration by means of EPROMs
- Can be mounted with four M5L2708K 1K-byte EPROMs
- Can be connected with the PCA0402 touch-panel
- Programs can be checked by means of the PCA0403 program checker
- Can be connected with equipment through the 68-pin card-edge connector
- Dimensions (L x W x H): 190 x 180 x 20mm

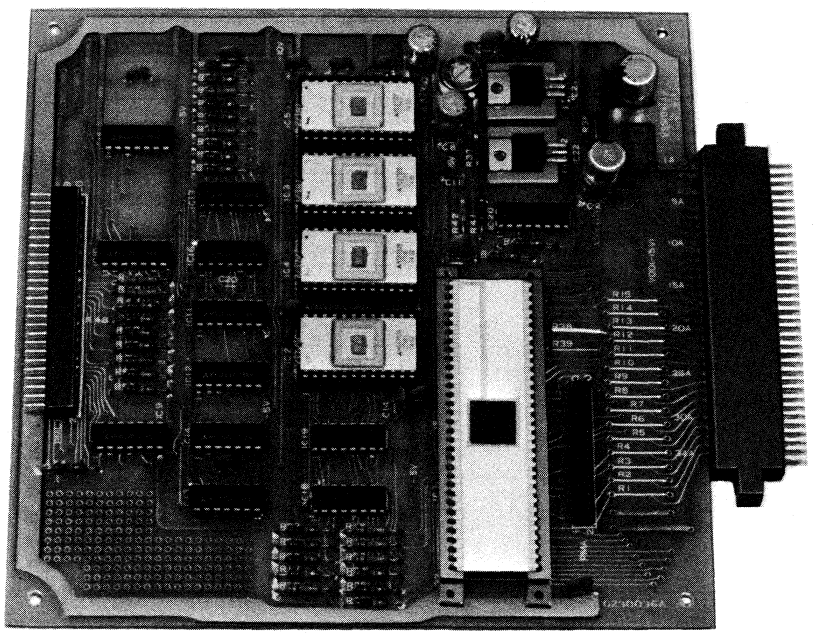
APPLICATIONS

- System development of application program for the M58840-XXXP single-chip 4-bit microcomputer, and prototyping of application equipment.

FUNCTIONS

In applying a single-chip 4-bit microcomputer for the control of equipment operation, the operating sequence of the equipment must be entered into a program. Errors can be costly and time-consuming when discovered after the mask has been prepared. With the PCA0401, however, the program is stored in erasable PROMs (EPROMs), allowing errors to be easily corrected as soon as they are detected.

Basically identical in function with the M58840-XXXP, the PCA0401 consists of an M58842S system-evaluator device and up to four M5L2708K EPROMs. The only difference is that the M58842S has address output pins (A_0~A_{10}) and data input pins (I_0~I_8) for external connection of the EPROMs.



12

MELCS 4 SINGLE-BOARD SYSTEM-EVALUATION COMPUTER

SUMMARY OF OPERATIONS

EPROM

Since the instruction word length of the M5880-XXXP is 9 bits, the PCA0401 was designed to store instructions by using a pair of EPROMs: one for the low-order 8 bits of the instruction code, and the other for the high-order bit. Thus the use of four EPROMs provides the same 2048-word × 9-bit memory as the M58840-XXXP.

Because address output of the M58840-XXXP is composed of the output lines $A_0 \sim A_{16}$, lines $A_0 \sim A_9$ can simply be connected with input lines $A_0 \sim A_9$ of the EPROM, with line A_{10} assigned to chip select. A memory control signal line allows the output of the EPROM to be kept in the floating state by means of an external control signal, for use as a program checker.

Clock Generator

Like the M58840-XXXP, the M58842S has an on-chip clock generator and oscillates in the 300 kHz~600 kHz range by connecting a CR circuit or ceramic filter between the pins X_{IN} and X_{OUT}. External synchronization may be effected by connecting an external clock source through terminal X_{IN}. The PCA0401 is designed to oscillate at 500 kHz.

Reference Voltage and Reset Signal Input

The input pin V_{REF} is designed to accept the reference voltage for A/D conversion, and it can also be used to apply a reset signal from an external source.

Interrupt

The 1-level of interrupt can be carried out by applying the interrupt request signal to the INT terminal.

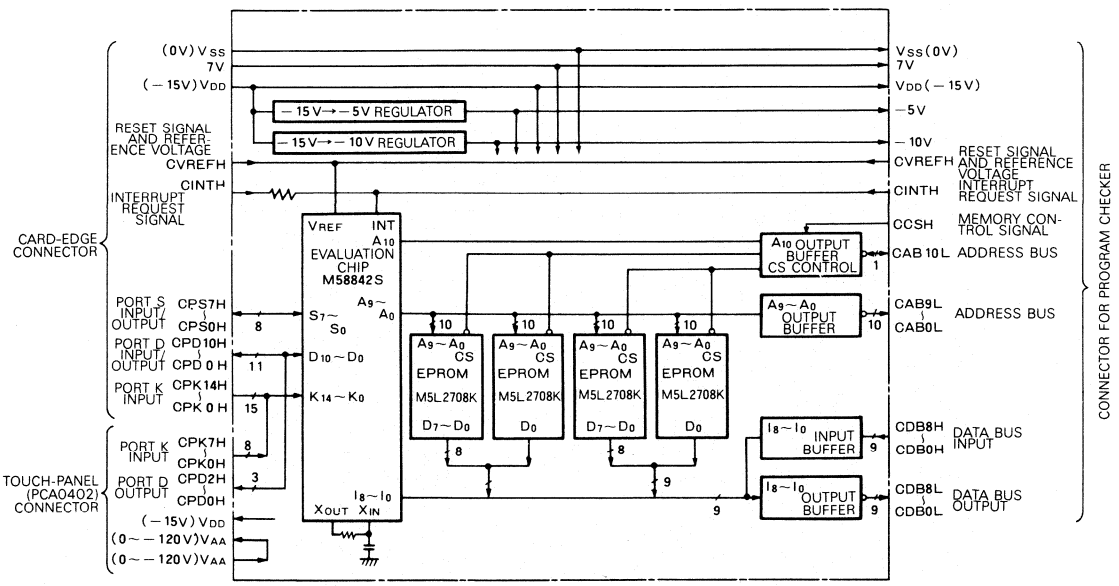
Input/Output Ports

I/O port S has 8.bit I/O lines, outputting information in 8-bit units, and receiving input in 4-bit units. Port S converts 4-bit data in register A to 8-bit information using a Programmable Logic Array (PLA). As the PLA code structure can be determined freely at the time of masking, it is highly suitable for driving segments of the numerical and character display device. The port S output is also capable of directly driving a large fluorescent display tube. The standard codes are masked in the S-port output PLA. For further details on these codes, see the code table for the S-output PLA of the M58842S.

I/O port D has 11-bit I/O lines, and each bit can be set/reset when used for output. Port D is also capable of directly driving a large fluorescent display tube, and can be used to test input signals when port D is used for input.

Port K has 15 bits of input lines. It is used not only for receiving digital data, but also in manipulating analog signals and in interfacing with a touch keyboard by means of the on-chip A/D converter.

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

Power Supply Source

The PCA0401 requires external power supplies of −15V, (from which it produces −5V and −10V internally) and 7V. It supplies −15V to the M58842S and 7V, −5V, and −10V to the EPROMs. In addition, 0~−120V power can be supplied for the PCA0402 touch keyboard through an input terminal on the PCA0401.

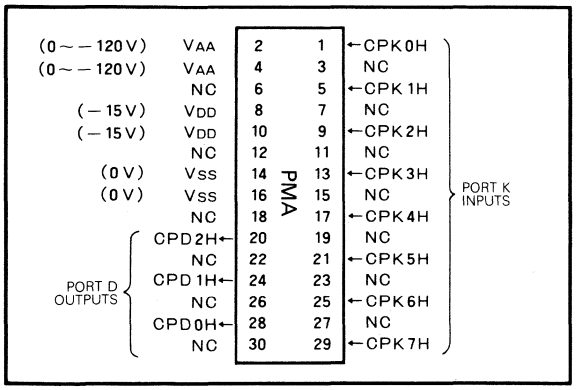
Terminals

The card-edge connector of the PCA0401 has the same I/O signal lines as the M58840-XXXP, so the equipment control can be effected by connecting to the terminals of the card-edge connector instead of to the pins of the M58840XXP.

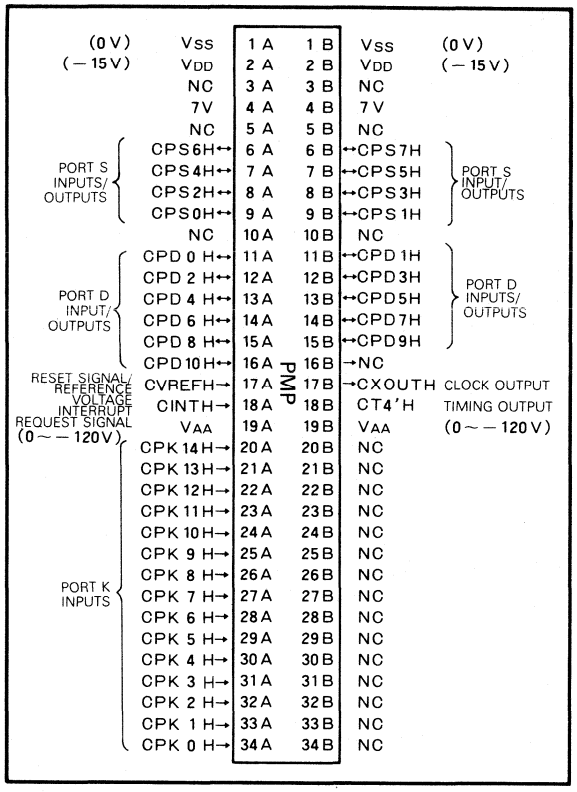
The PCA0402 touch keyboard can easily be connected to the PCA0401 via a connector—one of the major advantages of the M58840-XXXP.

For the PCA0403 program checker, another connector is provided to control program execution by fetching address and data information from the M58824S through the buffer or by applying instructions for program testing.

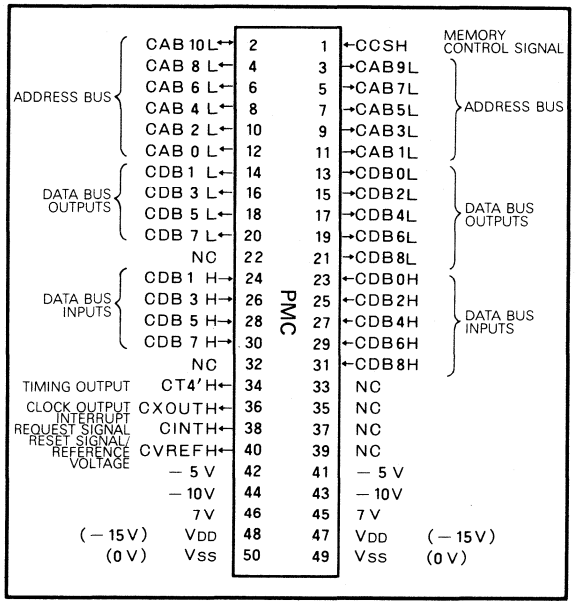
PIN CONFIGURATION OF PMA CONNECTOR



PIN CONFIGURATION OF PMP CONNECTOR



PIN CONFIGURATION OF PMC CONNECTOR



12

MELCS 4 SINGLE-BOARD SYSTEM-EVALUATION COMPUTER

SPECIFICATIONS

Memory Capacity

ROM: 2048 words x 9 bits
[With 4 x EPROM (M5L2708S) in use]
RAM: 128 words x 4 bits
(Implemented in the M58842S)

I/O Ports

Analog input port (port K): 15 bits
(with the internal touch-keyboard interface and the data-manipulation circuit:
I/O port (port S): 8 bits
Output: 8 bits
Input: 4 bits × 2
I/O port (port D):
Output: 1-bit × 11
Sense input (sense high or low-level): 1-bit × 11

Instruction Timing

Execution time: 10μs~20μs (variable)
Oscillation frequency: 300 kHz~600 kHz (variable)

Connectors

For I/O terminals (PMP connector):
card-edge type, 68-pin (34 pins on one side)
For the touch keyboard PCA0402 (PMA connector):
straight pin header, 30-pin
For the program checker PCA0403 (PMC connector):
angle pin header, 50-pin

Power Supply

V_{DD}: −15, 0.4A (typ)
7V: 7V, 0.1A (typ)
V_{AA}: Maximum 0~−120V, 0.1A
Dimensions (L x W x H): 190 x 180 x 20 mm

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Limits	Unit
V_{DD}	Supply voltage	With respect to V_{SS}	−20~0.3	V
7V	Supply voltage		−5.3~15	V
V_{AA}	Supply voltage		−130~4	V
V_I	Input, except program checker		−20~0.3	V
V_O	Output voltage, port D and S outputs, except program checker		−35~0.3	V
V_I	Input voltage, program checker, except power source, CVREFH and CINTH		−5.5~0.5	V
V_O	Output voltage, program checker, except the power source, CVREFH and CINTH		−5.5~0	V
Topr	Operating free-air temperature range		0~70	°C
Tstg	Storage temperature range		−40~125	°C

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

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	−16.5	−15	−13.5	V
7V	Supply voltage	6.5	7	7.5	V
V_{AA}	Supply voltage	−120		0	V
V_{IH}	High-level input voltage, port D and S inputs	−1.5		0	V
V_{IL}	Low-level input voltage, port D and S inputs	−33		−4.2	V
V_I	Analog input voltage, port K input	V_{REF}		0	V
V_I	Digital input voltage	V_{DD}		0	V
V_{REF}	Reference voltage	−7		−5	V

ELECTRICAL CHARACTERISTICS (Ta = 0 ~ 70°C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{IH}	High-level input voltage, port D and S inputs		−1.5		0	V
V_{IL}	Low-level input voltage, port D and S inputs		−33		−4.2	V
V_I	Analog input voltage, port K input		V_{REF}		0	V
V_I	Digital input voltage		V_{DD}		0	V
V_{OH}	High-level output voltage, port D output	V_{DD} = −15V, I_{OH} = −15mA, Ta = 25°C	−2.5			V
V_{OH}	High-level output voltage, port S output	V_{DD} = −15V, I_{OH} = −8 mA, Ta = 25°C	−2.5			V
V_{OL}	Low-level output voltage, port D and S outputs		−33		0	V
I_{OH}	High-level output current, port D output	V_{DD} = −15V, V_{OH} = −2.5V, Ta = 25°C	−15		0	mA
I_{OH}	High-level output current, port S output	V_{DD} = −15V, V_{OH} = −2.5V, Ta = 25°C	−8		0	mA
I_{OL}	Low-level output current, port D and S outputs	V_{OL} = −33V	−33			μA

MELCS 4 CAPACITIVE TOUCH KEYBOARD

DESCRIPTION

The PCA0402 is to be used with the PCA0401 single-board computer for evaluating the program of the M58840-XXXP MELPS 4 single-chip 4-bit microcomputer. Fabricated on a 180 x 180 mm printed circuit board, the PCA0402 is put into use by connecting it with the PCA0401.

FEATURES

- Programmed detection of touch contact by means of program
- Completely isolated from the equipment
- Number of keys: 24
- Dimensions (L x W x H): 180 x 180 x 20 mm

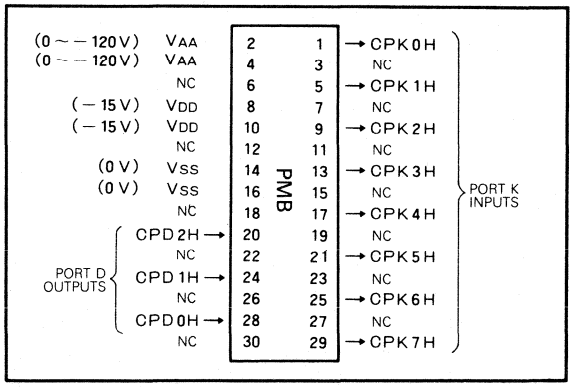
APPLICATIONS

- For the PCA0401 single-board computer designed to evaluate the program of the M58840-XXXP MELPS 4 single-chip 4-bit microcomputer

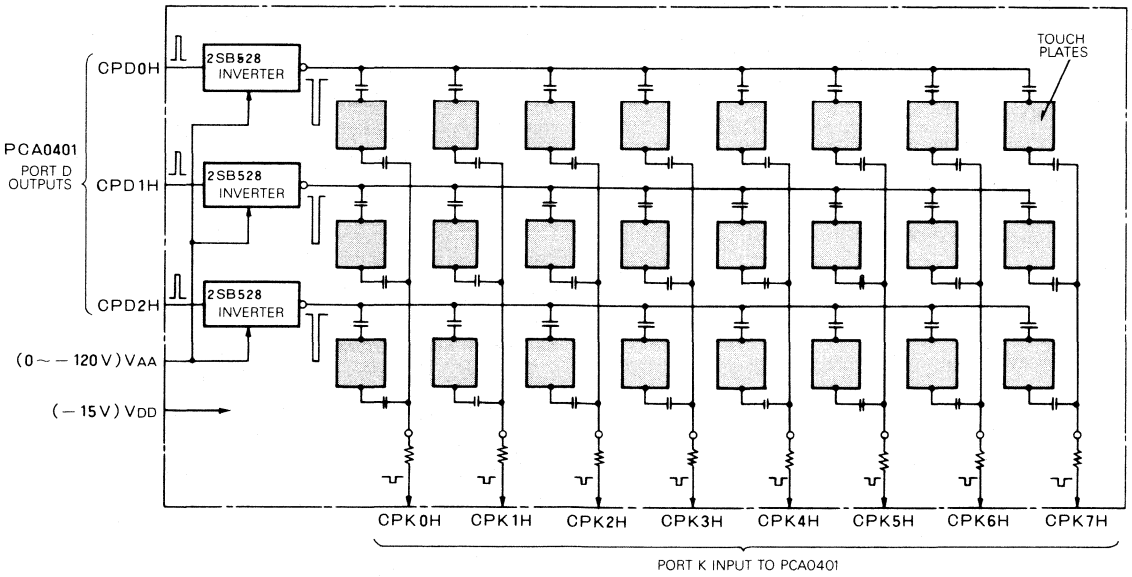
FUNCTIONS

The PCA0402 touch-keyboard comprises 3 rows × 8 columns = 24 touch plates. The plates can be assigned for independent use. Touch keyboards actually employed on the equipment will be of glass; this plate assembly permits the testing of panel operation in combination with the PCA0401 system evaluation computer.

PIN CONFIGURATION OF PMB CONNECTOR



BLOCK DIAGRAM



12

MELCS 4 CAPACITIVE TOUCH KEYBOARD

SUMMARY OF OPERATIONS

As shown in Fig. 1, each touch plate is formed of three electrodes; one is on the component side of the printed circuit board, and two on the soldered side. The electrode on the soldered side is applied with an inverted pulse from the port D output, while the others are connected to input port K. Capacitance is thus formed between the electrodes on the component and soldered sides. Fig. 2 shows the equivalent circuit. Pulses from port D are conducted to port K through that particular capacitance, but the level decreases to 1/10~1/30 of the original owing to capacitance in other plates, stray capacitance of lead wires, and input capacitance in the M58842S system evaluation device. When one touches the touch plate the effect is as though a capacitance of about 100~200pF were connected between the touch plate and ground, so the level pulse carried to input port K is significantly reduced. The pulse signal is sampled and compared with the reference voltage from the D/A converter under the control of the program.

Inverted pulses from the port D outputs $D_2 \sim D_0$ are supplied to 3 rows, each consisting of 8 electrodes. Pulses that have passed through capacitances in each touch key in this way are conducted to the respective input liners of port K after being attenuated with stray capacitance.

The pulse supplied from the port D output is simultaneously amplified and inverted, and its level can be changed freely in the 0~−120V range. Owing to the differences in capacitance of glass touch keyboards and the stray capacitance of lead wires, it is impossible to determine in advance what the precise input pulse level to port K of the M58840-XXXP would be in actual application. Therefore design provides that the level of the input pulse to the PCA0401 may be adjusted in accordance with port K input pulse level in each case by adjusting the pulse voltage. This is done by varying the inverter source voltage for port D of the PCA0402.

Be sure that the equipment is effectively grounded, because touch-key operation depends on the capacitance between the ground and the body of the operator.

The port D output, port K input and power supply can be connected to and disconnected from the PCA0401 by means of the connector.

Fig. 1 Touch-keyboard construction

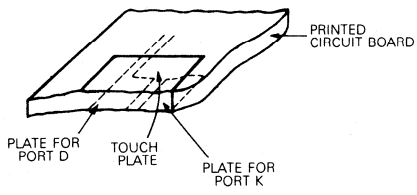
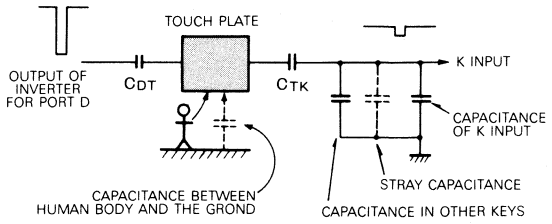


Fig. 2 Equivalent circuit of touch keyboard



C_{DT} : Capacitance between plate for port D and touch plate.
C_{TK} : Capacitance between plate for port K and touch plate.

SPECIFICATIONS

Safe and Satisfactory Operation

As the touch keyboard utilizes capacitance between the electrodes on the circuit board or glass frame, it is completely isolated from the circuit of the equipment, for operational safety. To assure satisfactory performance the equipment must be grounded.

Touch-Plate Inspection Operation by Means of Program

Because the comparison voltage for touch-plate operation is produced through the on-chip A/D converter by means of software, the comparison voltage must be pre-programmed within a range of −7V~0V. It is also possible to set up individual comparison voltage for each individual touch plate for greater freedom in system design.

Number of Touch Plates

The 24-plate (3-row x 8-column) configuration of the PCA0402 should be sufficient for most applications, but both the M58840-XXXP and the M58824S are furnished with further potential for comprising more touch plates.

Connection to PCA0401

Electrical connection of the PCA0402 to the PCA0401 computer is effected simply through 30-pin connectors provided on both sides.

Power Supply

Power at −15V is supplied automatically from the PCA0401 once connection to the PCA0402 is made, and a 0~−120V variable power supply is available through the card-edge connector of the PCA0401 to allow control of the input voltage to port K of the PCA0401 in accordance with the actual K port input voltage of the M58840-XXXP.

Connector

Receptacle: straight, 30-pin (PMB connector)
This connector can be attached to the PCA0401 through screw holes at the four corners.

Physical Dimensions

L x W x H: 180 x 180 x 20 mm

MELCS 4 PROGRAM CHECKER

DESCRIPTION

The PCA0403 is a simple program checker, not requiring a monitor program, that can test the functions of systems comprising the PCA0401 MELCS 4 single-board system-evaluation computer and the PCA0402 touch keyboard.

Use of the PCA0403 facilitates program testing and system evaluation.

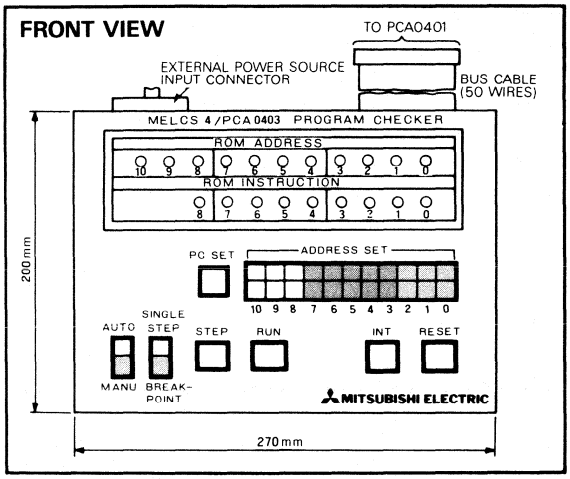
FEATURES

- Single-step function allows step-by-step program execution for each machine cycle with the CPU halted at any designated address.
- Breakpoint function causes the CPU to halt at any designated address, so that the program execution resumes from that address.
- Program execution is possible from any desired address.
- Reset function applies a reset signal to the CPU, so that the program will start from address 0 of page 0.
- Depression of the INT switch causes generation of interrupt signal.
- Uses special bus cable approx. 1000mm in length.
- Compact dimensions (L x W x H): 200 x 270 x 27 mm

APPLICATIONS

- Program test and system evaluation for systems comprising a MELPS 4 single-chip 4-bit microcomputer.

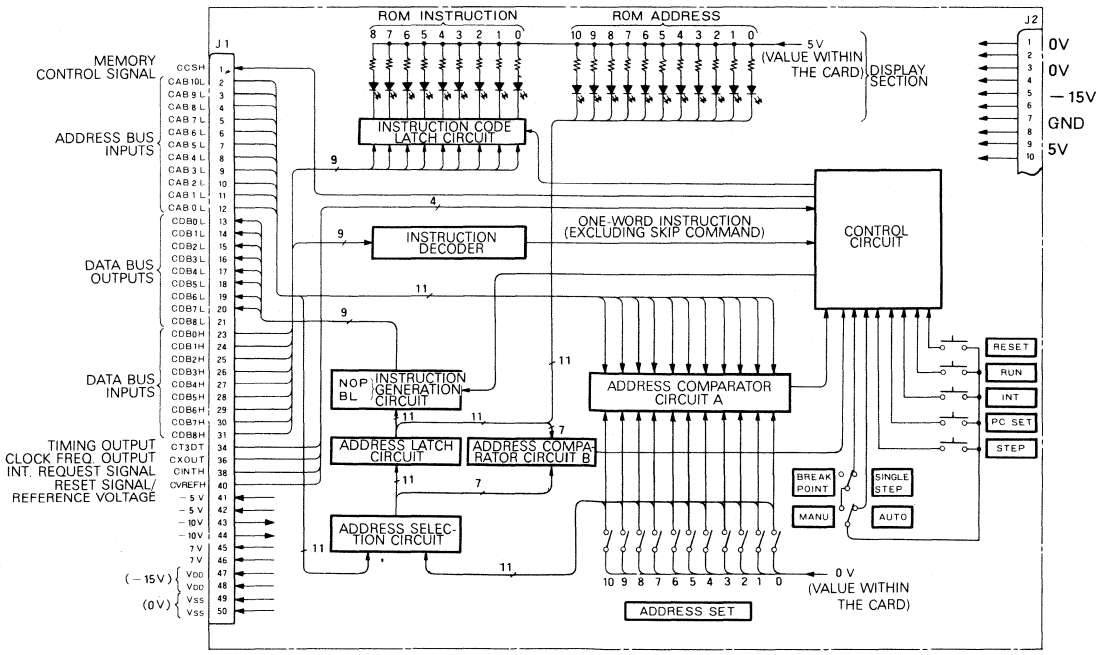
FRONT VIEW



FUNCTION

Since the PCA0403 is hardware-oriented and has no software requirement, simply connecting it to the system under test provides easy debugging of software and hardware. With the PCA0403, the program can be started from any desired address and program execution is a single-step program. It also serves for analyzing and determining interfacing requirements between the touch keyboard and the system-evaluation device.

BLOCK DIAGRAM



MELCS 4 PROGRAM CHECKER

FUNCTIONS

1. Display

When the CPU is halted, the address display indicates the address to be executed, and the instruction-display section indicates the instruction code that is in the address where the CPU halts.

2. Instruction-Code Latch

Latches and holds the instruction code.

3. Address Latch

Latches and holds the next address to be executed. Address latch timing differs depending on whether the instruction is of one, two, or three words. Thus with the PCA 0403, it is not possible to latch the address during execution of jump instructions. Instead, the address after the execution of the jump instruction is latched. Similarly for skip instructions, the address after the execution of the skip instruction is latched.

4. Instruction Decoder

Decodes instruction codes and sends out appropriate control signals to the control circuit in accordance with the number of words in the instruction. When a two- or three-word instruction, such as a jump instruction, is fetched, the decoder sends out a control signal to suppress the address latch signal and latch the address after the execution of the instruction.

5. Address-Selection Circuit

Selects the system address signal and the address signal from the ADDRESS SET switch.

6. Address Comparator Circuits A, B

When a program is to be started from the designated address or a breakpoint is set, these circuits compare the system address signal with the address signal from the ADDRESS SET switch, and output a signal to the control logic when these addresses match.

7. NOP/BL Instruction Generation Circuit

Generates an NOP instruction when the CPU is to be halted, and a BL instruction to resume the program execution from the next instruction.

8. Control Circuit

Generates various timing pulses required inside the program checker and controls execution procedures according to the instruction code given. It also generates control signals corresponding to each of the function switches to control the operation.

9. ADDRESS SET Switches and PC SET Switch

The ADDRESS SET switches are used in setting a specific starting address and breakpoint (stopping) address for the execution of a program. After setting the starting address, the PC SET switch is depressed, and the breakpoint address is set.

10. INT Switch

Generates a 100μs interrupt signal pulse when turned on.

11. RESET Switch

When this switch is on, the CPU starts to execute instruction from address 0 of page 0.

12. SINGLE STEP/BREAKPOINT Switch

If this switch is turned to the BREAKPOINT side, the desired address is entered through the ADDRESS SET switches, and the CPU is set to RUN, the CPU continues to execute instructions until that specific address is reached, at which point it stops.

If the CPU is set to RUN with the switch turned to the SINGLE STEP side, the program is executed step by step each time the RUN switch is depressed. If single-step operation is to be carried out up to and including page 14, the STEP switch must be turned on, and the AUTO/MANU switch must also be kept on the MANU side while this process is continued.

13. AUTO/MANU Switch

When the RUN switch is depressed with the AUTO/MANU switch set to the AUTO side, the CPU starts to run under the automatic operation mode and executes the instructions in order.

14. STEP and RUN Switches

Once the desired starting address and breakpoint address are set (see §9 above), the instructions from start to breakpoint can be executed by depressing the STEP switch when the start address in on the current page, and the RUN switch when it is not.

MICROCOMPUTER SOFTWARE

13

SOFTWARE CODES

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)

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
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)

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
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

Note 1 : Includes contents of all above manuals concerning MELPS 4 software.

13

MELPS 4/41 SOFTWARE

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.

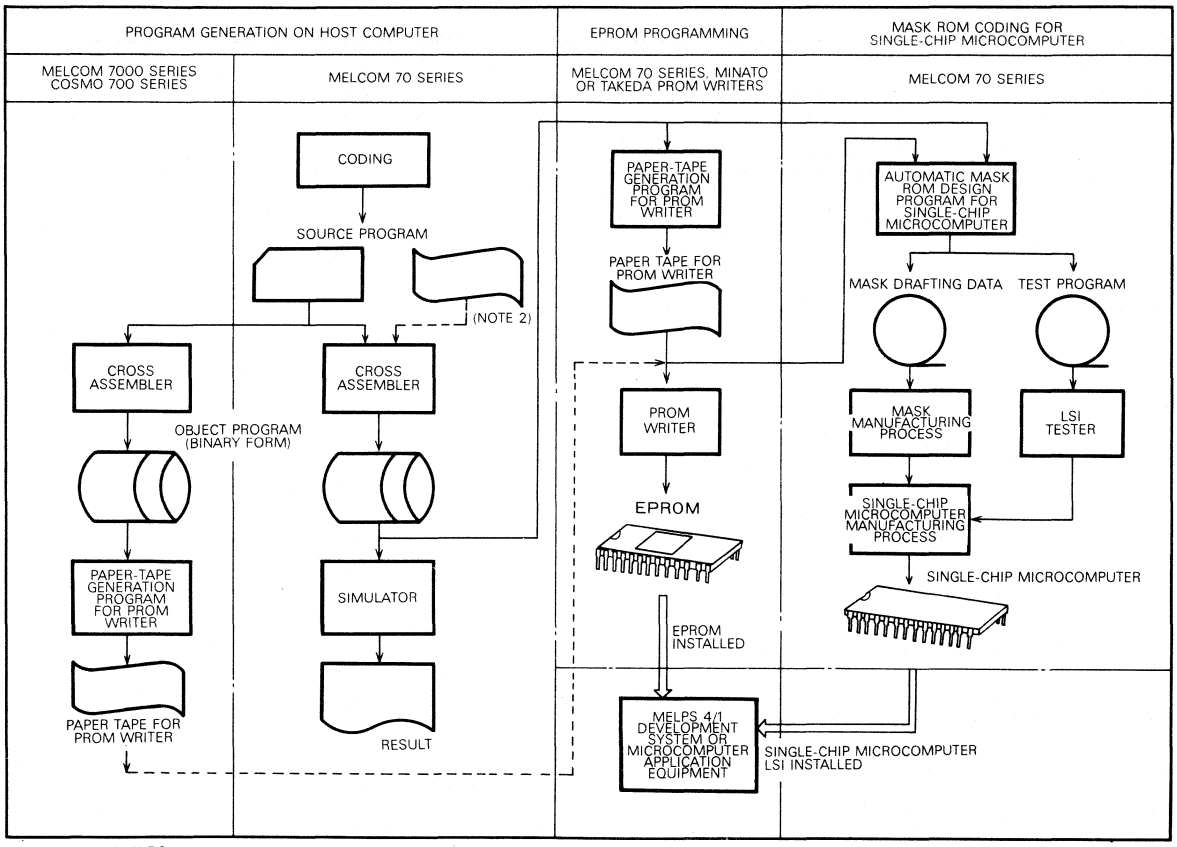
DEVELOPMENT OF APPLICATION PROGRAMS

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 2 : With MELPS 41, paper-tape can also be used for source program input.

MELPS 8/85 SOFTWARE

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)
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

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
	PL/1μ cross compiler	Simulator	Paper-tape generation program for PROM writers
Host programs	Compiles a source program written in PL/1μ language and produces as output an object program in machine language. The complete Intel PL/M language is a subset of PL/1μ. Therefore, any program written in PL/M can be compiled using a PL/1μ compiler. Additional functions have been included in PL/1μ that make it easy to use.	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 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.
	Cross assembler		Mask-programmable ROM
	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.		M58730-XXXS, M58731-XXXS and M58332-XXXS 1K~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.
	Self assembler	BOM-B and BOM-PTS Basic operating monitors	Editor
Target programs	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.	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 BOM-B is stored in the M58731-0001S mask-programmed ROM.	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. This program is stored in the M58730-001S mask-programmed ROM.

13

MELPS 8/85 SOFTWARE

DEVELOPMENT OF APPLICATION PROGRAMS

The user can develop his application programs using MELPS 8/85 software in any of three ways.

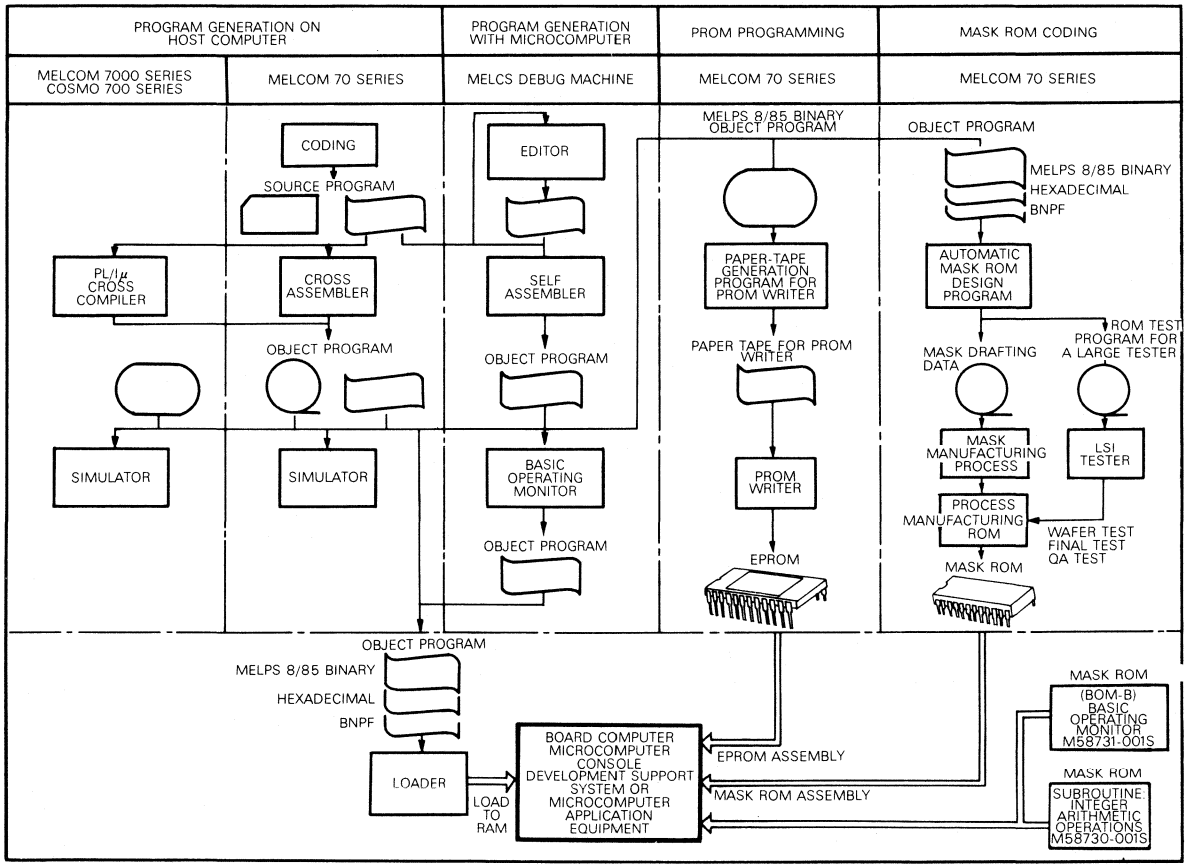
1. On a host computer: the MELPS 8/85 cross compiler or cross assembler is used for object-program generation, and the simulator is used for program debugging.
2. On a microcomputer: the MELPS 8/85 assembler is used for object-program generation, and the microcomputer is used for execution and implementation 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

DESCRIPTION

The MELPS 4 cross assembler has been prepared for the development of application programs suitable for equipment using the M58840-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

- 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

FUNCTIONS

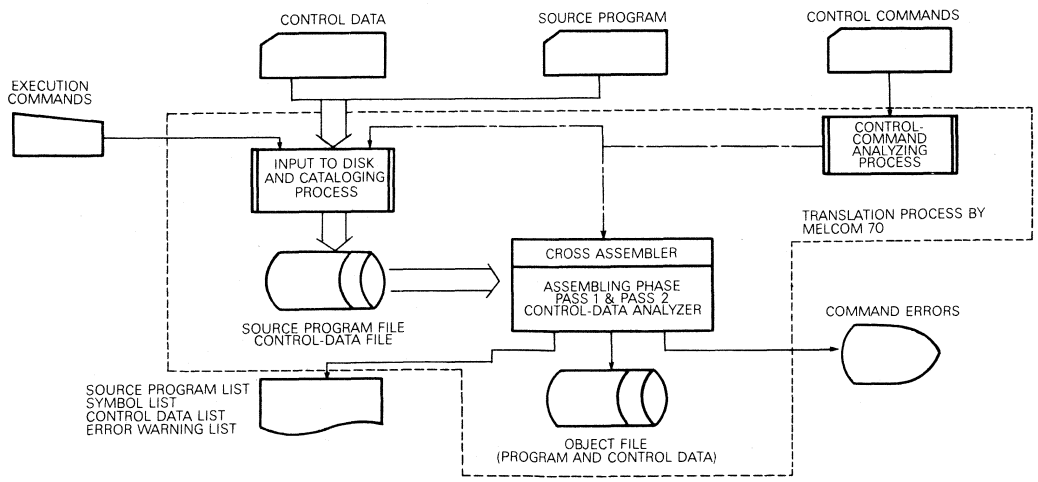
This cross assembler converts source programs written in the MELPS 4 assembly language to machine instruction codes that are filed in disk storage in the form of binary absolute object codes.

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<93AO>

13

CROSS ASSEMBLER

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 = (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) y = (C / D) **x** : Control data input **y** : Source program input **C** : Punched card input **D** : Disk input

MITSUBISHI ELECTRIC

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

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 and control data.

An asterisk (*) in the first column of a line indicates that the entire statement is used as a comment field.

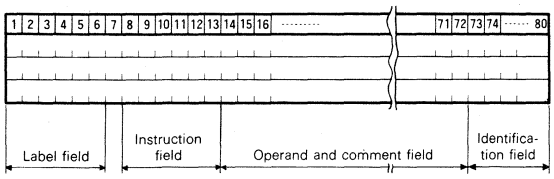
The following are valid characters for use in statements:

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.

13

CROSS ASSEMBLER

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 $

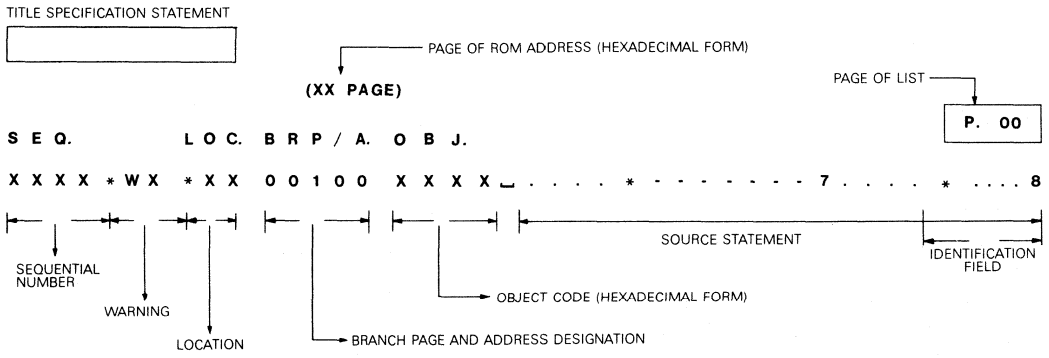
where "**x x x**" indicates the type of error by a numerical code.

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
SEQ.   LOC.BRP/A. OBJ. ....*....1....*....2...SOURCE STATEMENT....*....5....*....6....*....7....*....8
                                ( 00 PAGE )

  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   0,0                                 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.

MITSUBISHI ELECTRIC

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

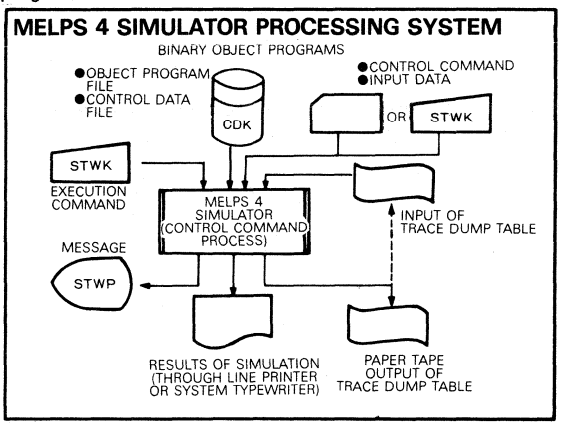
- In conjunction with the MELPS 4 cross assembler as a tool for developing application programs for 4-bit microcomputers
- Especially useful for debugging programs prepared for the M58840-XXXP

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.



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␣(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>

SIMULATOR

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.

TYPICAL APPLICATION

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
```

SIMULATOR

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  = 0000000000000000  D 0-A  = 0000000000 }
     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
        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 ---------- ⑯
///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  = 0000000000000000  D 0-A  = 0000000000
     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
        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 ---------- ⑲
///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

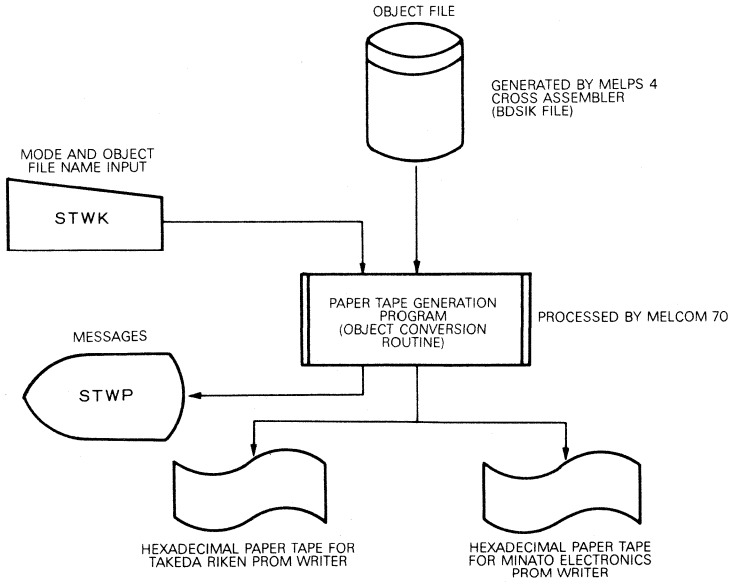
APPLICATIONS

- 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.

FUNCTIONS

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 1380). 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 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>

13

PAPER-TAPE GENERATION PROGRAM FOR PROM WRITERS

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

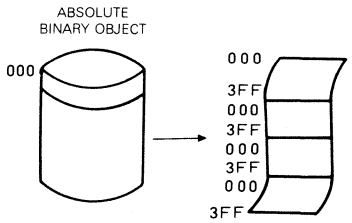


Fig. 1 Example of hexadecimal paper tape format of Takeda Riken

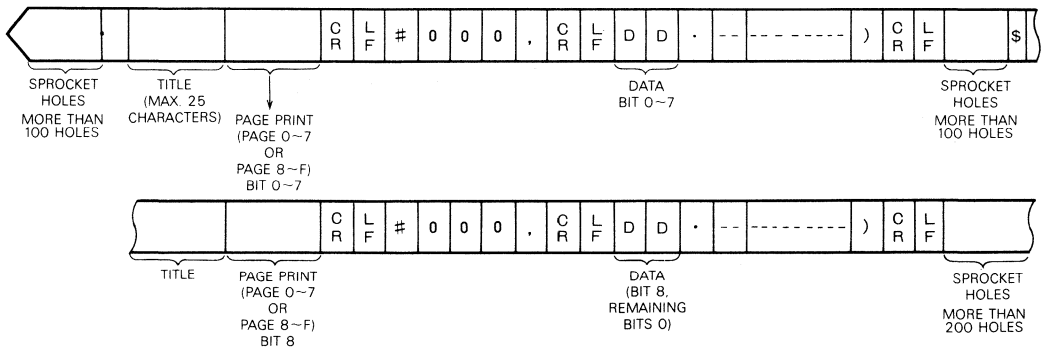
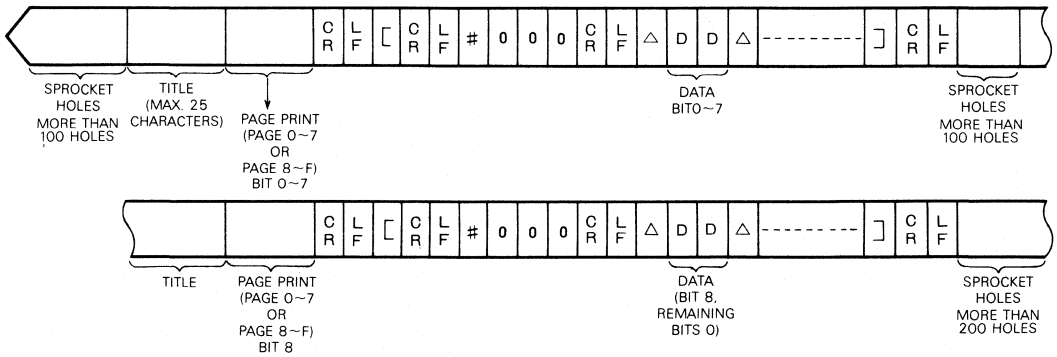


Fig. 2 Example of hexadecimal paper tape format of Minato Electronics



MITSUBISHI ELECTRIC

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 microcomputer.

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 typewriter

FUNCTIONS

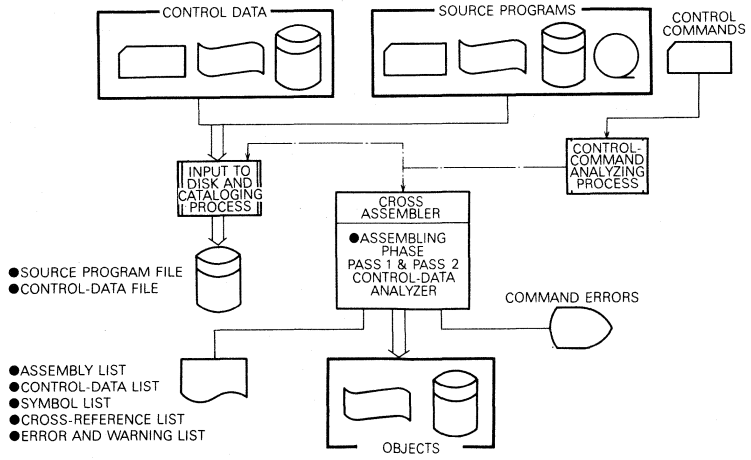
This cross assembler converts source programs written in the MELPS 41 assembly language to machine instruction codes, which are filed in disk storage in the form of binary absolute object codes.

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

13

CROSS ASSEMBLER

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 — A : ASCII code E : EBCDIC 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.

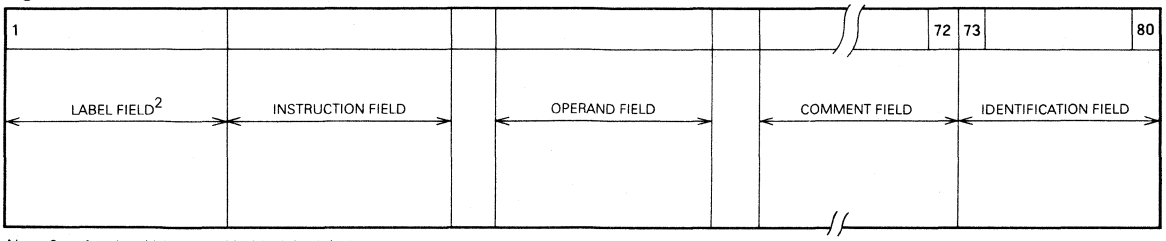
Writing an asterisk(*) in the first column of the source statement enables the whole statement to be used as a comment.

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.

13

CROSS ASSEMBLER

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     0)         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)

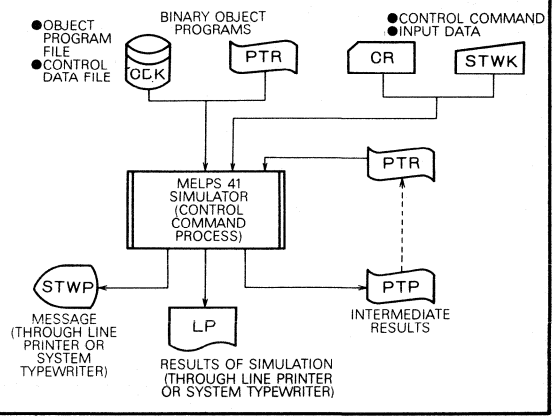
INPUT/OUTPUT MEDIA

- 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.

MELPS 41 SIMULATOR PROCESSING SYSTEM



FUNCTIONS

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

PROGRAM ORDERING INFORMATION

Program name	Ordering number	Software manuals included
MELPS 41 simulator	GB1SM0002	MELPS 41 Simulator Manual MELPS 41 Simulator Operating Manual

SIMULATOR

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).
	Program saving	Execution-program saving	SV	Outputs intermediate results of the program content, register, port, F/F, Timer, and memory in punched tape.
	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 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.
		Trace-region assignment discontinued	TD	Discontinues trace-region assignment by table-number designation.
		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 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.
		Reset 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, 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.
	Port control	Input-port control	IN	Controls the input-port data read-in device and the input port by print-mode designation.
		Export-port control	OT	Designates an output device for the data obtained from the output port.
	Interruption	Interruption-generation assignment started	IT	Sets interruption conditions such as interruption type, interruption generation, head address, and generation cycle number.
		Interruption-generation assignment discontinued	ID	Deletes the interruption-generation table.
		Printout of the interruption-generation table	PI	Prints out the interruption-generation table.
	Execution step time	Execution-timer setting and printout	TI	Sets the execution timer and prints out the number of execution steps.
	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.
		Memory-protection-region assignment discontinued	PD	Discontinues the memory-protection assignment by the memory-protection table number.
		Printout of the memory-protection region	PP	Dumps the contents of the memory-protection table.
	Execution start	Program-execution start-up	RN	Starts simulation execution. Termination by executing the halt point and the execution-limit step number.
		Program execution	GO	Starts simulation execution. Termination by halt-point execution. Trace-region assignment is invalid here.
	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.)

TYPICAL APPLICATIONS

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

Typical 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, 0:0, 0: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␣0:5, 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, 0: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, 0: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, 0:0** **0:0 0=1** **0:1 0=•**	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.

13

SIMULATOR

Fig. 1 Example of simulation results

```
*** START SIMULATOR OF MELPS 41 ***

///S.T                                                    ..... ①

///LOAD 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
 2:1:1  0=.                                              } ..... ④

///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
            2:1:0   A 0 0 0 0 0 0 0 0 0 0 0 0 B C   }  ..... ⑭

 00:17  181      LX        01

              (EXTM)
            2:0:0   1 2 0 0 0 0 0 0 0 0 0 0 0 0 3
            2:1:0   A 0 0 0 0 0 0 0 0 0 0 0 0 B C   }  ..... ⑮

 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 3
            1:0   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   }  ..... ⑯

 00:1A  000      NOP

              (INTM)
            0:0   1 2 0 0 0 0 0 0 0 0 0 0 0 0 3
            1:0   A 0 0 0 0 0 0 0 0 0 0 0 0 B C   }  ..... ⑰

///TI DIS

          ***    EXECUTION  COUNTER    ***
             # STEP #          # TIME #              }  ..... ⑱


                  75.          0 H 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.

Typical examples of the use of the MELPS 41 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

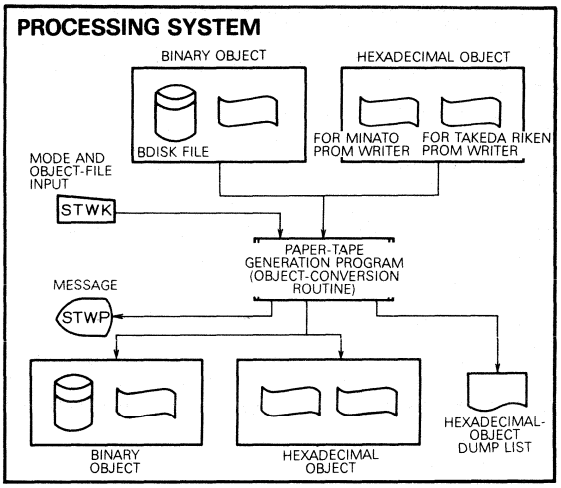
- Input: Cartridge-disk units, paper tapes (ASCII code, even parity)
- Output: Paper tapes (ASCII code, even parity), cartridge-disk units
- Control-command outputs: Through system-typewriter keyboard
- Message: System-typewriter printout

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).

FUNCTIONS

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.



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.

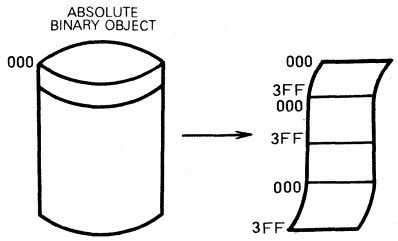
Example of Hexadecimal Tape Output

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



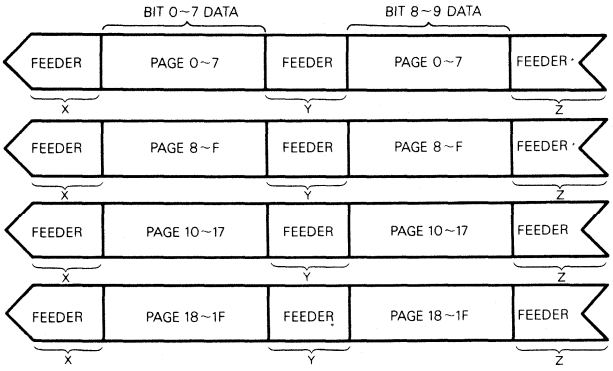
Error Processing

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.

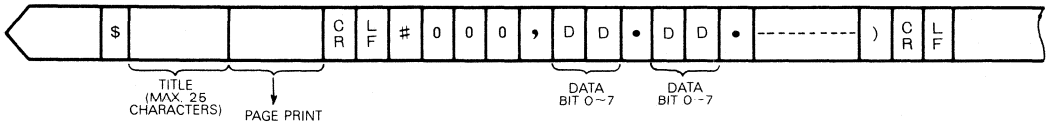
Fig. 1 Example of hexadecimal paper-tape output



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

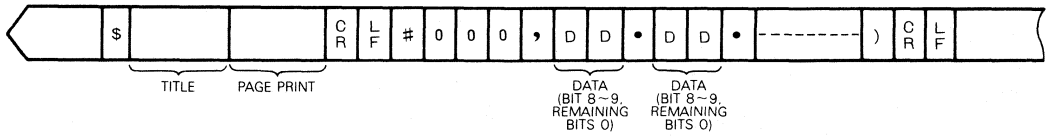
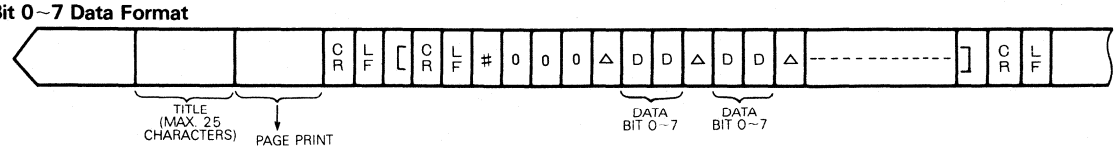
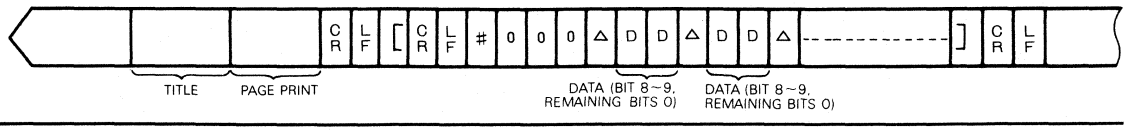


Fig. 3 Example of hexadecimal paper-tape format of Minato Electronics

●Bit 0~7 Data Format



●Bit 8~9 Data Format



MITSUBISHI ELECTRIC

DESCRIPTION

The MELPS 8-48 cross assembler has been prepared for aiding the development of application programs suitable for equipment using the M5L 8041-XXXP and M5L 8049-XXXP single-chip 8-bit microcomputers.

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.

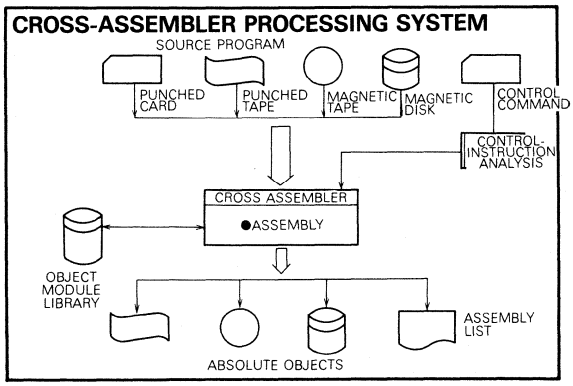
The assembler language has machine and pseudo instructions. The full equipment of pseudo instructions and control commands ensures high programming and debugging efficiency. Coding can be carried out in a free format.

FEATURES OF THE CROSS ASSEMBLER

- Flexibility in assembler-language changing
- Various input/output media available
- Object output: Punched tapes, magnetic tapes, magnetic disks

FEATURES OF THE ASSEMBLER LANGUAGE

- 13 pseudo instructions
- Numerical formula used
- Character constants and strings used
- In addition to decimal notation as the standard format, 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

FUNCTIONS

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.

- Free-format coding
- Output of the symbol table inside the object
- Execution computer: MELCOM 70 minicomputer (Memory capacity, more than 24K-words, monitor, BDOS)
- Implementation language: FORTRAN IV (parts are written in assembler language)

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

13

CROSS ASSEMBLER

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 and pseudo 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 and M5L 8049-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.

Fig. 1 Object-module configuration

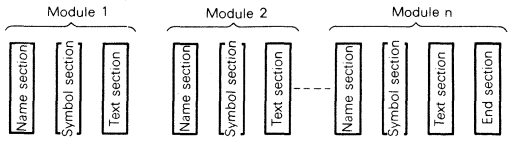


Table 1 Control commands and their functions

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.
Input/output assignment control	///ASMB48, X, Y, Z	Designates the use of source-program list and assigns the source-program input device and object output for the internal numeric-symbol table. X = (L / N) — X : Designation of assembly list; L : Listing needed; N : No listing needed Y = {C / P / M / D} — Y : Designation of source-program input device; C : Card reader; P : Punched-tape reader; M : Magnetic tape; D : Magnetic disk Z = (S / N) — Z : Designation of internal-numeric-code object output; S : Output needed; N : No output needed
File-assignment control	///SDISK, XXXXX	Designates the source-program file name (max. 5 characters)
	///ODISK, YYYYY	Outputs the object after assembly, and designates the file name (max. 5 characters)
	///BDISK, ZZZZZ	Outputs the object after linkage, and designates the file name (max. 5 characters)
Link control	///LINKG, W, Z, Y, F_1, F_2···	Assignment control for the object output at the time of linkage. W = (A / R) — W : Object designation; A : Absolute object; R : Relocatable object Z = (S / N) — Z : Designation of internal-numeric-code object output; S : Output needed; N : No output needed Y = {P / M / D} — Y : Designation of object-output media; P : Punched tape; M : Magnetic tape; O : Magnetic disk F_1, F_2 ····· File names (designation in 5 or less characters is possible up to 20 names)
Relocation-assignment control	///LKLOC, xxxxx, yyyyy	Assigns the ROM and RAM head addresses in the case of absolute-object generation. xxxxx····· ROM-area head address (hexadecimal in max. 4 digits) yyyyy····· RAM-area head address (hexadecimal in max. 4 digits)

MITSUBISHI ELECTRIC

3. Language Format

The following free format should be used in coding programs in this cross assembler.

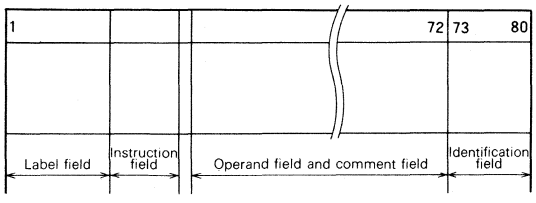
The single-line statement of the source program is composed of label, instruction, operand, comment, and identification fields. The format of the source statement is free, as shown in Fig. 2, allowing easy coding and punching without the fear of dislocated columns. The following characters can be used 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
	ROM	ROM-region declation instruction
	RAM	RAM-region declaration instruction
	END	End-declaration instruction
Numeric-symbol and memory-content definition instructions	EQU	Numeric-symbol definition instruction
	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 Expression formats for numeric values, character constants and formula

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'**
Formula	4 Arithmetic-rule operations	**+, −, *, /**
	Logic formula	—
Others	Program counter	**$**

13

CROSS ASSEMBLER

5. Identification field

The use of this field is optional. Many operators find it convenient to use it for the sequential identification card number.

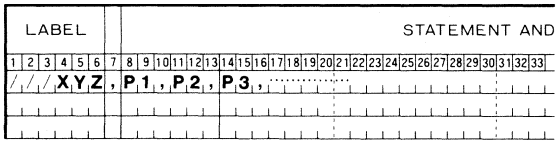
CODING FORMAT

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

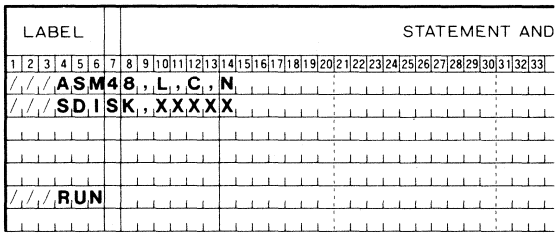
1. Control-command general format



XYZ : Assembly-control-command symbols

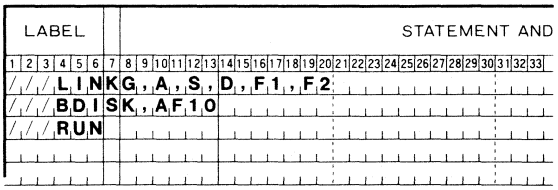
P1, P2, P3····· : Assembly-control parameters

2. Example of assembly control



The source program is input through the card reader, and an assembly list is output.

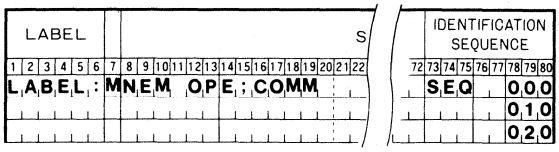
3. Example of link control



After assembly, files F_1 and F_2 of the relocatable objects stored in the disk are linked to generate an absolute object in File AF10.

2. MELPS 8/48 Assembler-Language Program

1. General format of program coding



LABEL: ····· A colon (:) is always placed behind the label name.

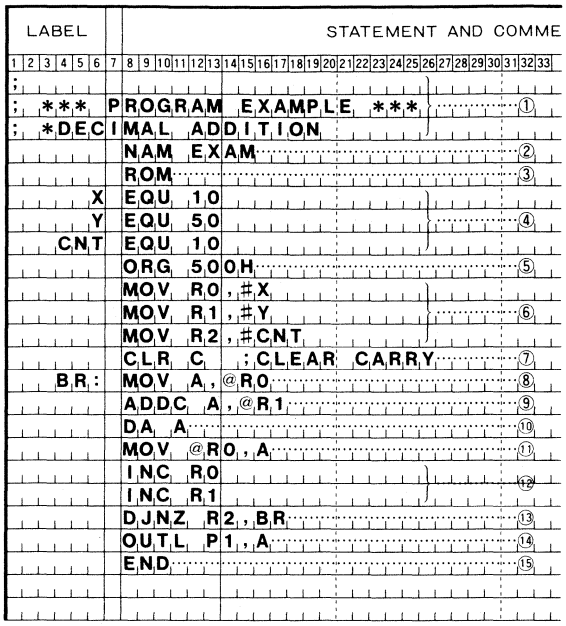
MNEM····· Instruction symbol (mnemonic)

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



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.

MITSUBISHI ELECTRIC

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

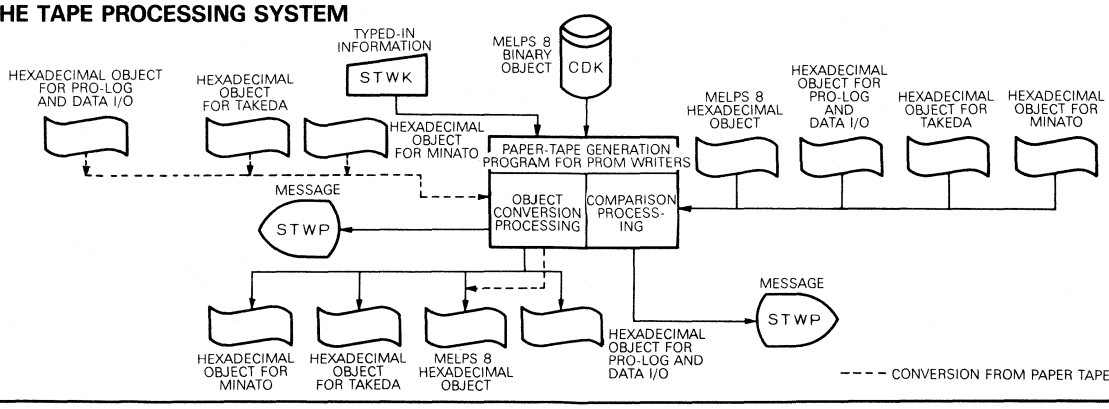
APPLICATIONS

- Programs are applicable to the M5L 2708K and -S (1K-word by 8-bit), M5L 2716K (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

13

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.

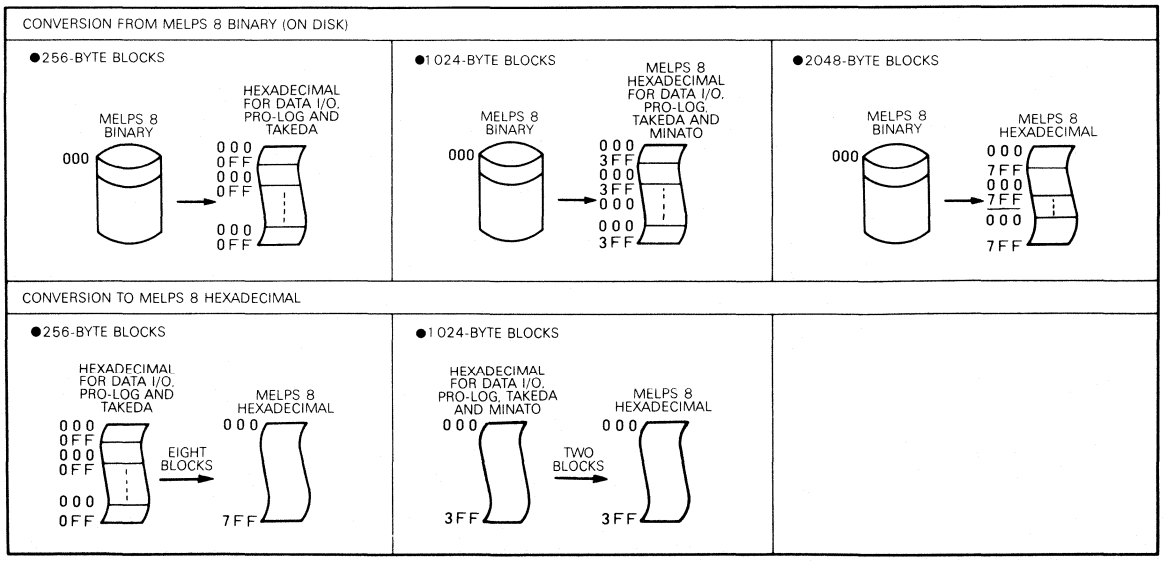
Table 1 Object conversions

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

Comparison \ Objects compared	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



MITSUBISHI ELECTRIC

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

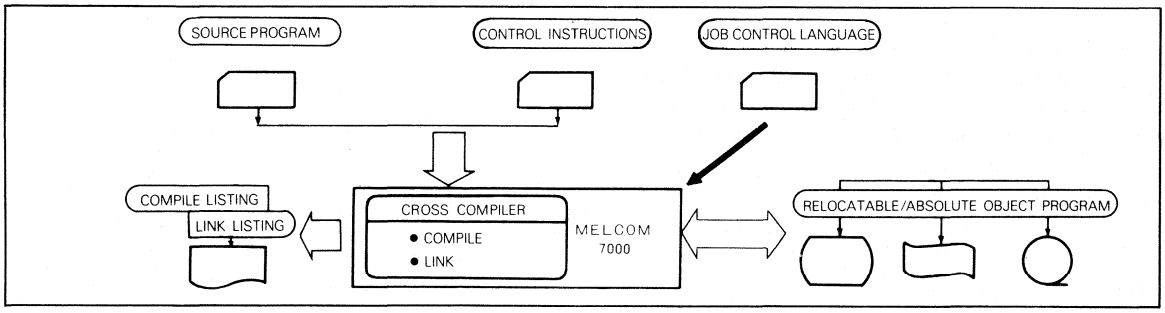
- Conditional compile with preprocessor
- Inline assembly
- Source program editing at compile time
- Assignment of programs to ROM or RAM regions
- Generates a relocatable object program
- Linking function
- Easily understood error messages
- Flexibility in input/output media
- Execution computer: MELCOM 7000 (BPM/UTS monitor)
- Implementation computer: MELPS 8/85 microcomputer
- Implementation language: FORTRAN IV

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).

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

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 (B-version)	GAM-SR00-07A
		MELPS 8/85 PL/Iμ Compiler Language Manual (B-version)	GAM-SR00-08A
		MELPS 8/85 PL/Iμ Cross Compiler Operating Manual (B-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 (B-version)	GAM-SR00-07A
MELPS 8/85 PL/Iμ Compiler Language Manual (B-version)	GAM-SR00-08A
MELPS 8/85 PL/Iμ Cross Compiler Operating Manual (B-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

PL/1μ CROSS COMPILER

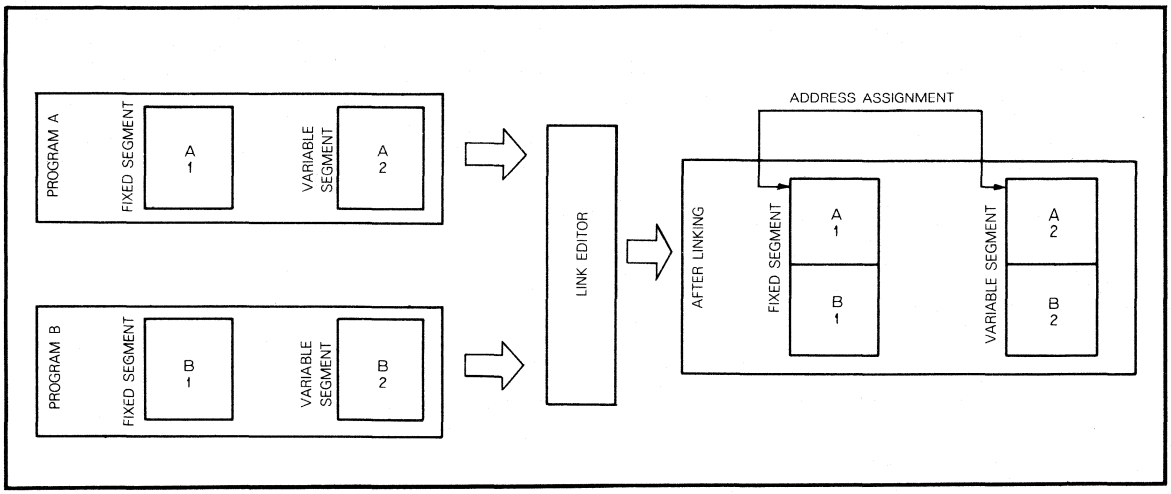
FUNCTIONS

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/Iμ 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.

MITSUBISHI ELECTRIC

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}.

PL/1μ CROSS COMPILER

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	ADDRESS ALIGNED AND BASED BINARY BY BYTE CALL CASE DATA DECLARE DISABLE DO ELSE ENABLE END ENTRY EOF EXTERNAL GENERATE GO GOTO HALT IF INITIAL INTERNAL INTERRUPT LABEL LITERALLY MAIN MINUS MOD NOT ON OPTIONS OR PLUS PROCEDURE RELOCATE RETURN THEN TO UNALIGNED WHILE XOR
Constant types	Binary, octal, decimal, hexadecimal character string
Variable declaration option	BINARY(n) 1≦n≦15、BIT(m) 1≦m≦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, CALL statement, DECLARE statement, DISABLE statement, DO group, ENABLE statement, ENTRY statement, GENERATE statement, GOTO statement, HALT statement, IF statement, NULL statement, ON statement, PROCEDURE statement, RELOCATE statement, RETURN statement,
DO group	DO WHILE, repeat DO, DO CASE
Library functions	CARRY DEC HIGH INPUT LAST LENGTH LOW MEMORY OUTPUT PARITY ROL ROR SHL SHR SIGN TIME ZERO
Preprocessor statements	% insert statement, %ACTIVATE statement, %DEACTIVATE statement, %END statement, %EXCLUDE statement, %GOTO statement, %IF statement, %INCLUDE statement, %MACRO statement, %NULL statement

MITSUBISHI ELECTRIC

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)

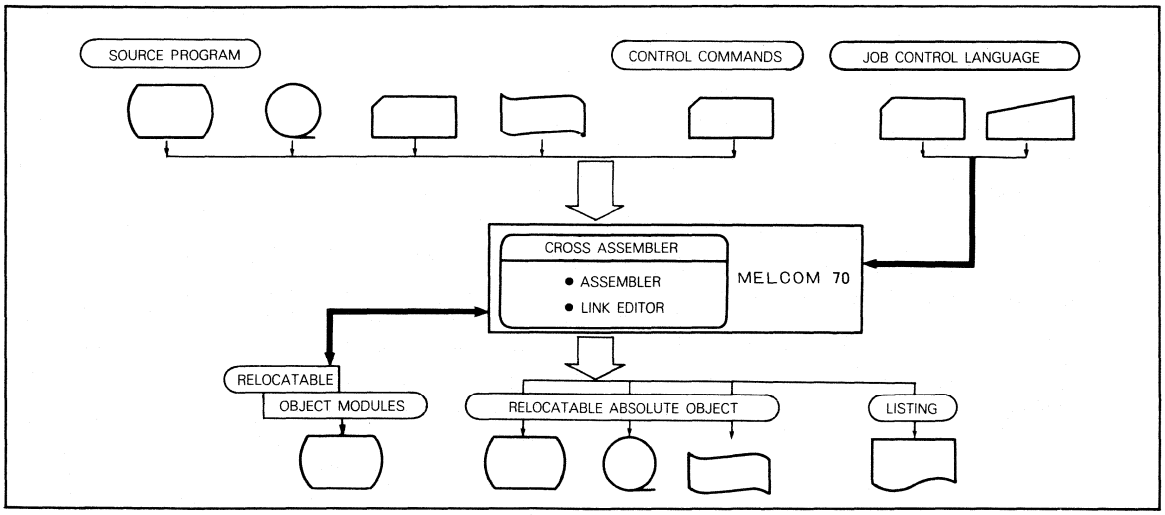
Of the Assembly Language

- 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)	GAM-SR00-34A
		MELPS 8/85 Cross Assembler Operating Manual (A-version)	GAM-SR00-02A
		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)	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

CROSS ASSEMBLER

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

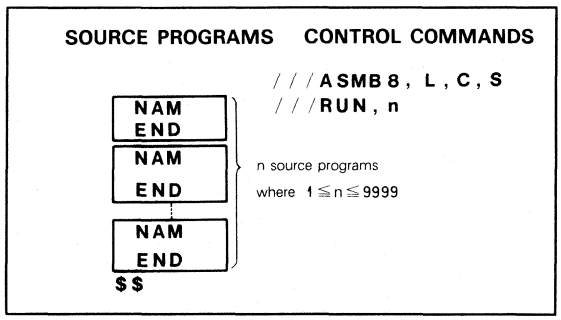
Classification		Control command name	Mnemonic
Assembler control		Execution start	RUN
		End	END
Execution control	Assembly control command	Input/output assignment	ASMB8
		Block assignment	BLOCK
		File assignment	SDISK
			ODISK
			BDISK
	Link control command	Link assignment	LINKG
		Link location assignment	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

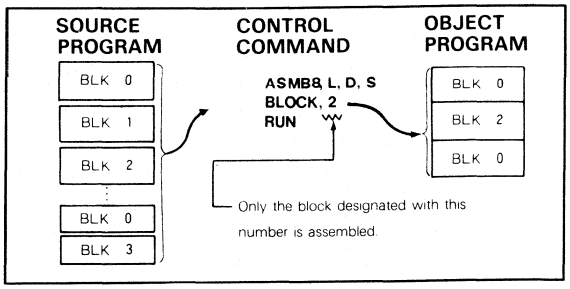
1. Multi-Assembly

Many programs can be batch-assembled in one run.



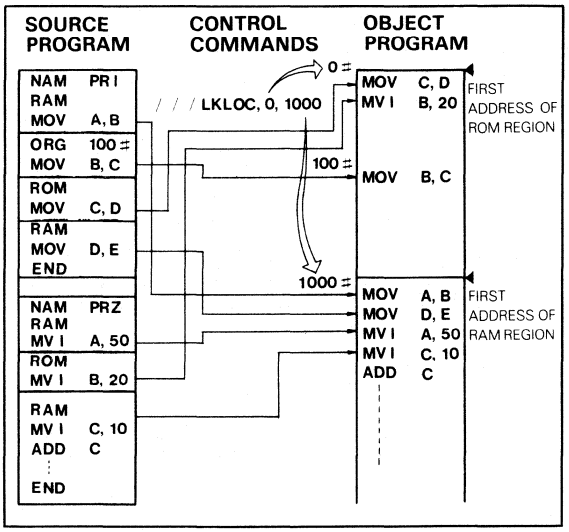
2. Conditional Assembly

Only the designated blocks of a source program are assembled.



3. Linking of ROM/RAM Regions

ROM and RAM regions are linked separately.



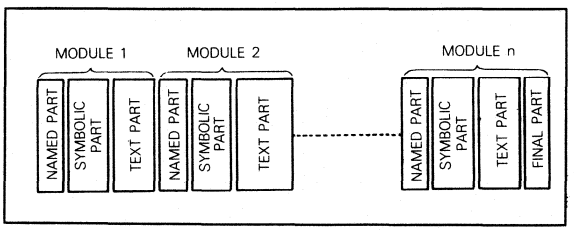
CROSS-ASSEMBLER OBJECT PROGRAM

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.

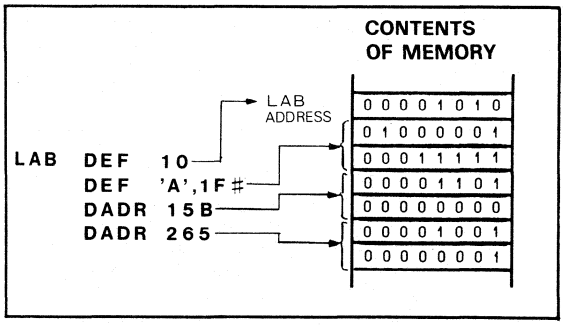
Table 4 List of pseudo instructions

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



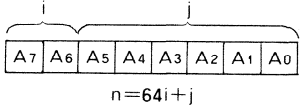
3. Macro Instructions

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



$n = 64i + j$

13

CROSS ASSEMBLER

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.

```
////ASMB8,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                    ....③
          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,SHORI                 ....⑩
*
LOOP2     MVI   H,0
          MOV   L,C
          LXI   B,SENS
          DAD   H
          DAD   B
          PCHL
*
          RAM                                     ....⑪
INDATA    DEF   45H                               ....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.

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)

FUNCTIONS

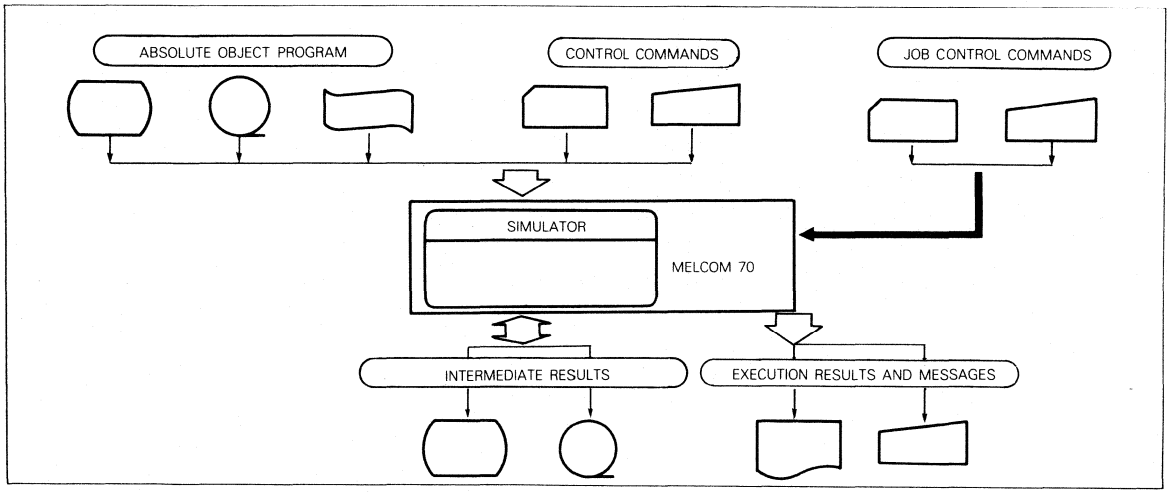
The trace command function assigns a specific trace region so that it traces only the specified program steps. Execution of the simulation can be halted by a breakpoint that can be assigned to any location. Program debugging efficiency can be expected to increase by the use of these functions.

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

SIMULATOR

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		Converted result
	A register	Carry flip-flop	
Integer 0~65,535	0	0	Correct
More than 65,535	0	1	Not correct
Character other than decimal digits	1	0	Not converted

Fig. 2 Assembly listing of the objective program "CON102"

```
**CROSS ASSEMBLER OF 8-BIT MICROPROCESSOR
0001*                 *
0002*      CON102     *
0003*                 *
0004  2328                       ORG  9000
0005  2328  219923   CON102      LXI  H',DED1
0006  232B  0605                 MVI  B,5
0007  232D  7E         CO100     MOV  A,M
0008  232E  FE3B                 CPI  48
0009  2330  DA9423               JC   ER
0010  2333  FE3B                 CPI  59
0011  2335  D29423               JNC  ER
0012  2338  23                   INX  H
0013  2339  05                   DCR  B
0014  233A  C22C23   D23         JN2  CO100
0015  233D  3A9D23   CO000       LDA  DED5
0016  2340  D630                 SUI  48
0017  2342  2600                 MVI  H,0
0018  2344  6F                   MOV  L,A
0019  2345  3A9C23   CO001       LDA  DED4
0020  2348  D630                 SUI  48
0021  234A  110A00               LXI  D,10
0022  234D  CA5523   CO101       JZ   CO002
0023  2350  19                   DAD  D
0024  2351  3D                   DCR  A
0025  2352  C34D23               JMP  CO101
0026  2355  3A9B23   CO002       LDA  DED3
0027  2358  D630                 SUI  48
0028  235A  116400               LXI  D,100
0029  235D  CA6523   CO102       JZ   CO003
0030  2360  19                   DAD  D
0031  2361  3D                   DCR  A
0032  2362  C35D23               JMP  CO102
0033  2365  3A9A23   CO003       LDA  DED2
0034  2368  D630                 SUI  48
0035  236A  11E803               LXI  D,1000
0036  236D  CA7523   CO103       JZ   CO004
0037  2370  19                   DAD  D
0038  2371  3D                   DCR  A
0039  2372  C36D23               JMP  CO103
0040  2375  3A9923   CO004       LDA  DED1
0041  2378  FE37                 CPI  37H
0042  237A  D29023               JNC  OV
0043  237D  D630                 SUI  48
0044  237F  111027               LXI  D,10000
0045  2382  CABA23   CO104       JZ   CO005
0046  2385  19                   DAD  D
0047  2386  3D                   DCR  A
0048  2387  C38223               JMP  CO104
0049  238A  229E23   CO005       SHLD BID
0050  238D  C39723               JMP  CO006
0051  2390  37         OV        STC
0052  2391  C39723               JMP  CO006
0053  2394  3E01       ER        MVI  A,11
0054  2396  A7                   ANA  A
0055  2397  00         CO006     NOP
0056  2398  76                   HLT
0057  2399  00         DED1      DEF  0
0058  239A  00         DED2      DEF  0
0059  239B  00         DED3      DEF  0
0060  239C  00         DED4      DEF  0
0061  239D  00         DED5      DEF  0
0062  239E  0000       BID       DADR
0063  2328                       END
```

SIMULATOR

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 CO002,CO003,CO004,CO005	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 CO002,CO003,CO004,CO005	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 DED1+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

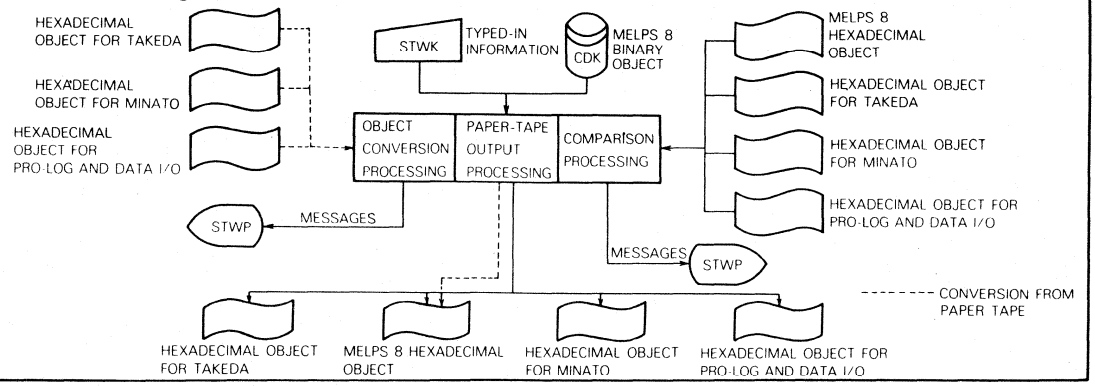
APPLICATIONS

- 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, Pro-log, Takeda or Minato Electronics.

FUNCTIONS

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

13

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.

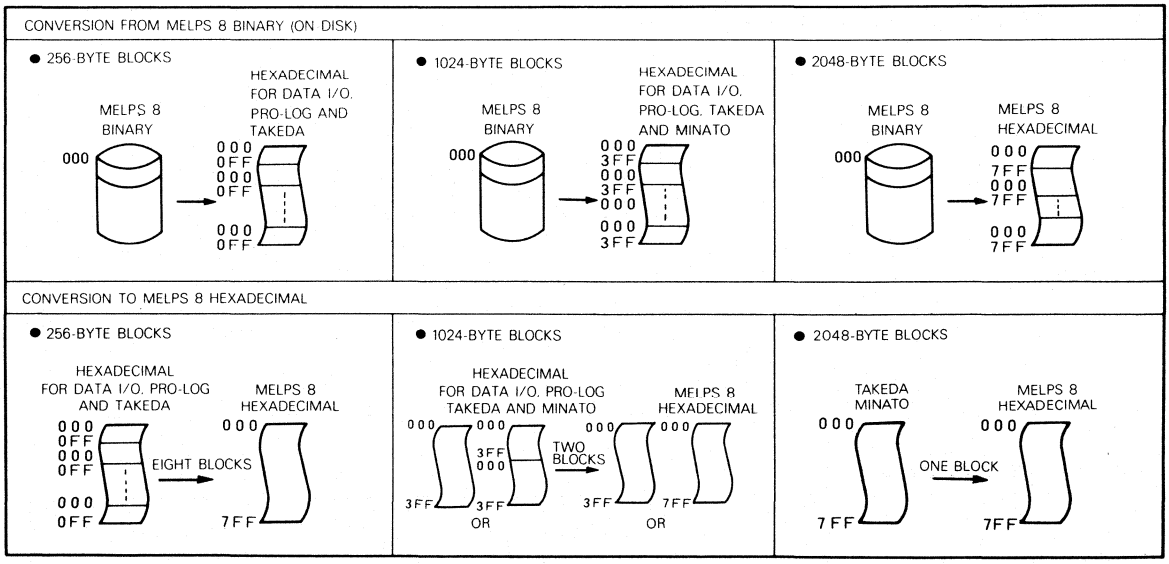
Table 1 Object conversions

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

Comparison \ Objects compared	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



MITSUBISHI ELECTRIC

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

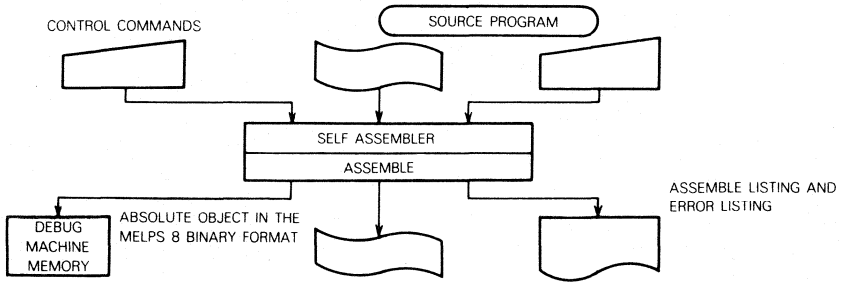
- Source input: Keyboard or paper tape
- Control command input: Keyboard
- Object output: Paper tape or debug machine memory
- Program supply media: Paper tape (object)

FUNCTIONS

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

13

SELF ASSEMBLER

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

Item / Functional classification	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	**▼A B ▼**
	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

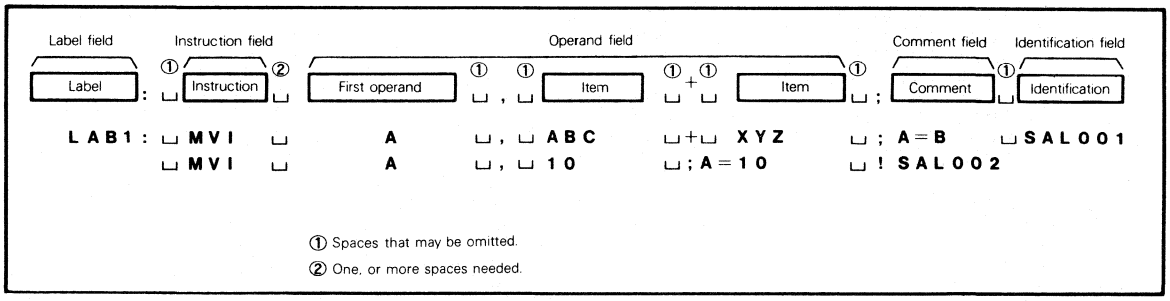
A line preceded by a semicolon (;) and a character string following a semicolon (;) placed at the end of a command or at an arbitrary position along a line are regarded as comments.

Comment examples;
```
THIS LINE IS COMMENT
;COMMENT
;
LI: MOV A, B;COMMENT;ABC
```

5. Identification field

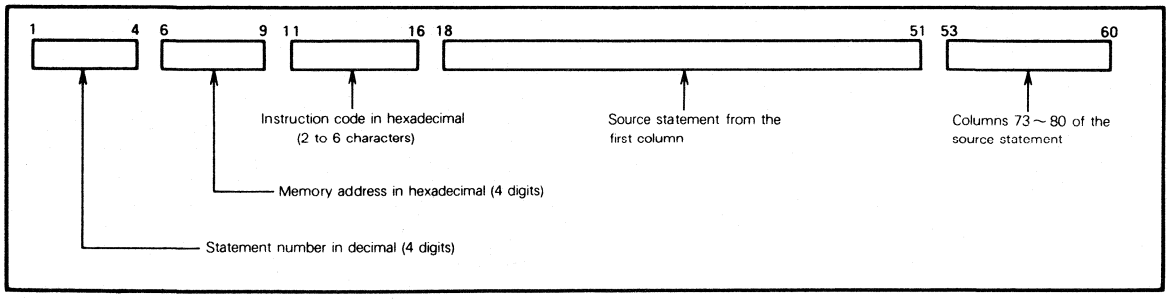
The field is composed of the characters in columns 73~80 or from 1 to 8 characters following !. This field is placed at the end of one statement and may be omitted.

Fig. 1 Source program format



Note : Mark ␣ denotes space.

Fig. 2 Assemble list format



13

SELF ASSEMBLER

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       16 18                                  51 53       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 microprocessors.

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
- 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

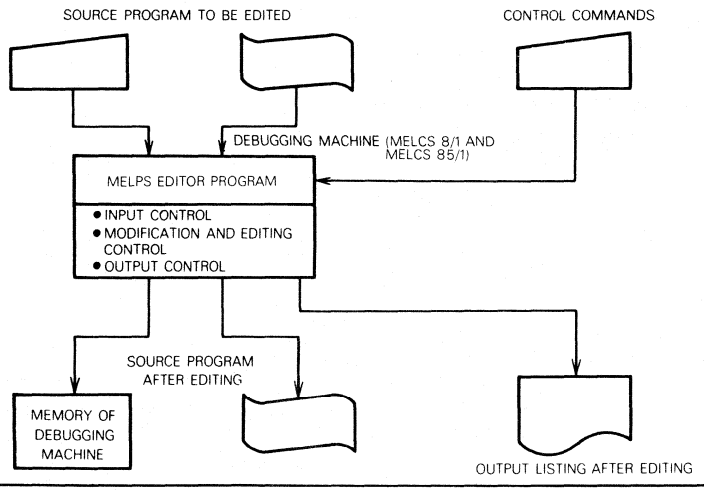
- Programs for editing: Keyboard or paper tape
- Control commands input: Keyboard
- Output after editing: Printer or paper tape

FUNCTIONS

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



ORDERING INFORMATION

Program

Program	Program code 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

Reference Manuals for Separate Ordering

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

13

EDITOR

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

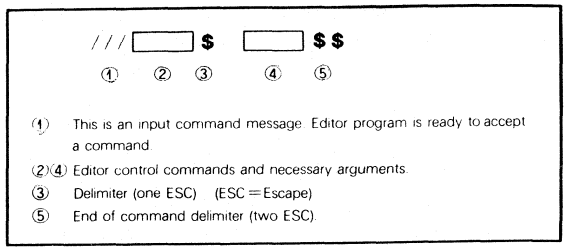
The format for repetition of a command is as follows:

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 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 ③ ⑤

Before the above command is executed and the modifications are made.

```
PUSH H     ①
MVI  D, 1  ②
MOV  E, A  ③
SHLD XY    ④
POP  D     ⑤
```

After the above command is executed and the modifications are made.

```
PUSH H     ①
MVI  D, 10 ②
SHLD XY    ④
POP  D;C   ⑦
```

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.

MITSUBISHI ELECTRIC

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

FUNCTIONS

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

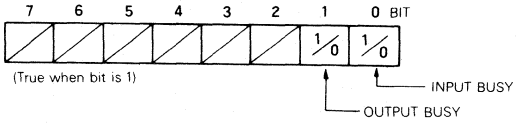
When the BOM start switch on the panel of the debugging machine MELCS 8/1 is turned on, the following message is printed out. After the printout, monitor commands can be entered.

BOM-PTS A00 'READY'
//

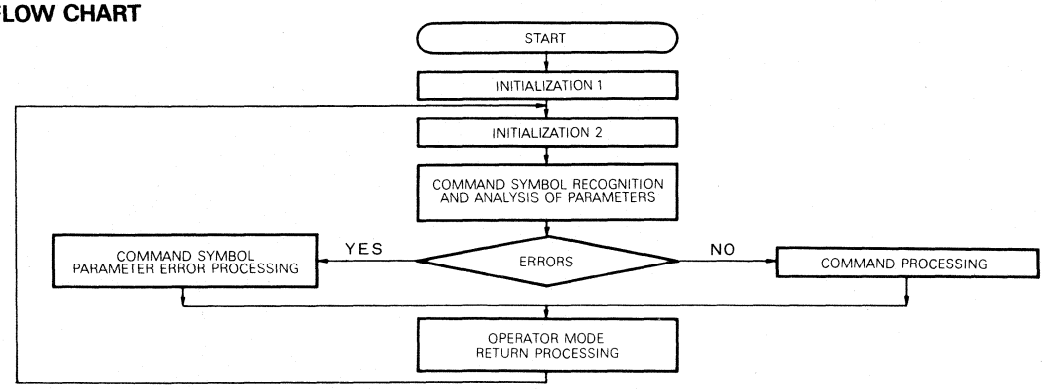
Hardware Limitations

1. Memory Configuration
   Memory locations in the ROM are:
   $E000_{16}$~$FCFF_{16}$
   In addition to the ROM, the following 78 bytes of RAM area are required:
   $F000_{16}$~$EDFF_{16}$
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)	GA20S0100	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

BASIC OPERATING MONITOR—PAPER-TAPE SYSTEM

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	//Upara1(4) CR LF	para1(4): First address of the user pseudo I/O processing routine
	LM	MELPS 8 binary loader	//LMpara1(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)	//DMpara1(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)	//DMparp1(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	//PMpara1(4), para2(4) CR LF	para1(4): Starting address para2(4): End address
	PA	Reverse assembler	//PApara1(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	//MMpara1(4) CR LF	para1(4): Starting address
	MC	Complement the memory data	//MCpara1(4), para2(4) CR LF	parp1(4): Starting address para2(4): End address
	MS	Set up constants in memory	//MSpara1(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	//MTpara1(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	//Ipara1(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	//Hpara1(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	//Hpara1(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	//Spara1(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	//Tpara1(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 instruction is in execution.
		Cancel trace and debug table	//Tpara1(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	//FPpara1(4), para2(4), para3(2) CR LF	para1(4): Starting address para2(4): End address para3(2): PROM writing address
	FT	Transfer writing address	//FTpara1(4) CR LF	para1(4): Starting address
	FC	Compare PROM data with main memory data	//FCpara1(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	CALL F00C # ···· 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

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

FUNCTIONS

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:

   $3F80_{16}$~$3FCD_{16}$

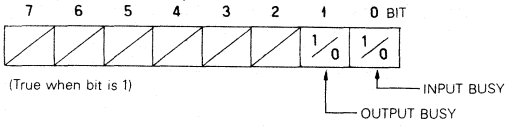
2. Input/Output Device Addresses

   PTR, for keyboard input: $7B_{16}$ **(In 7B#)**

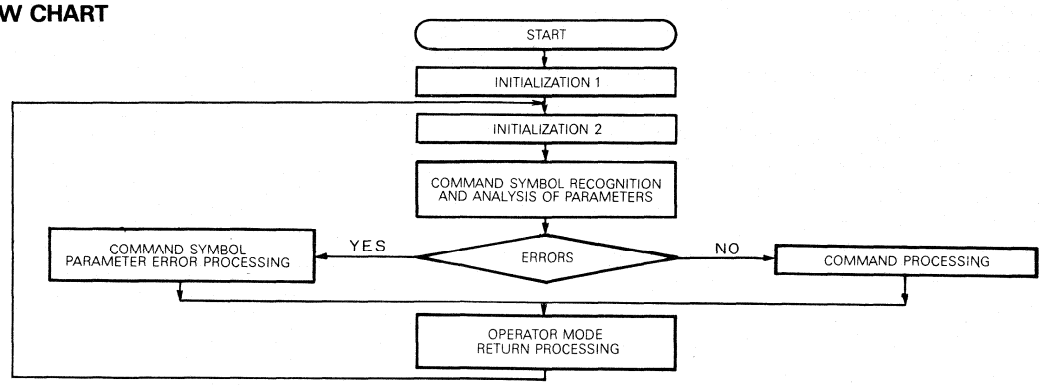
   PTP, for printout: $7B_{16}$ **(OUT 7B#)**

   Status input: $7B_{16}$ **(IN 7B#)**

The structure of the status bits is as follows:



FLOW CHART



ORDERING INFORMATION

Program

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

Reference Manuals for Separate Ordering

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

BASIC OPERATING MONITOR—BASIC SYSTEM

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	CALL 6EBC# ---------- Execute EXIO macro instruction DADR DCB1 ---------- First address of data control block (DCB) DCB1 DEF IOD --- I/O operation designation; PTR (IOD=52#), PTP (=50#), keyboard (=4B#), printout (=44#). DADR DA --- Set up the first address of I/O data area DADR DL --- Set up the data length for I/O data DA: I/O data-storing memory (DL byte)	

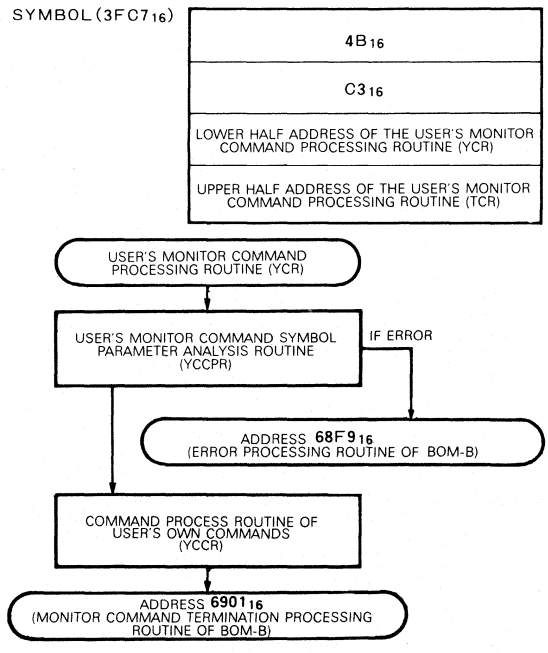
Note 1 : para n (m): This designation shows the nth parameter in a command (operator input or monitor printout), and also shows it to be a hexadecimal parameter (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F) of which the significant digits are 1~m. If the length exceeds m, the least significant digits 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

APPLICATION OF 16K-BIT DYNAMIC RAM

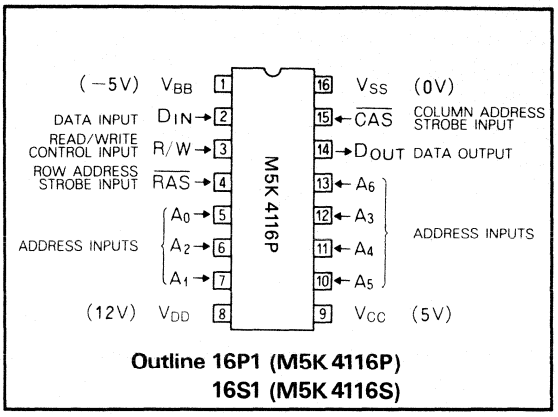
(M5K 4116P, S)

INTRODUCTION

The M5K 4116P, S are 16 384-word by 1-bit dynamic RAMs, fabricated by 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 1 compares the M5K 4116 16 384-bit dynamic RAM with a 4096-bit static RAM.

Fig. 1 Pin configuration (top view)



Outline 16P1 (M5K 4116P)
16S1 (M5K 4116S)

Table 1 Comparison of the 16 384 dynamic RAM and 4K static RAM

Device / Characteristics	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 : M5L2114S-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.

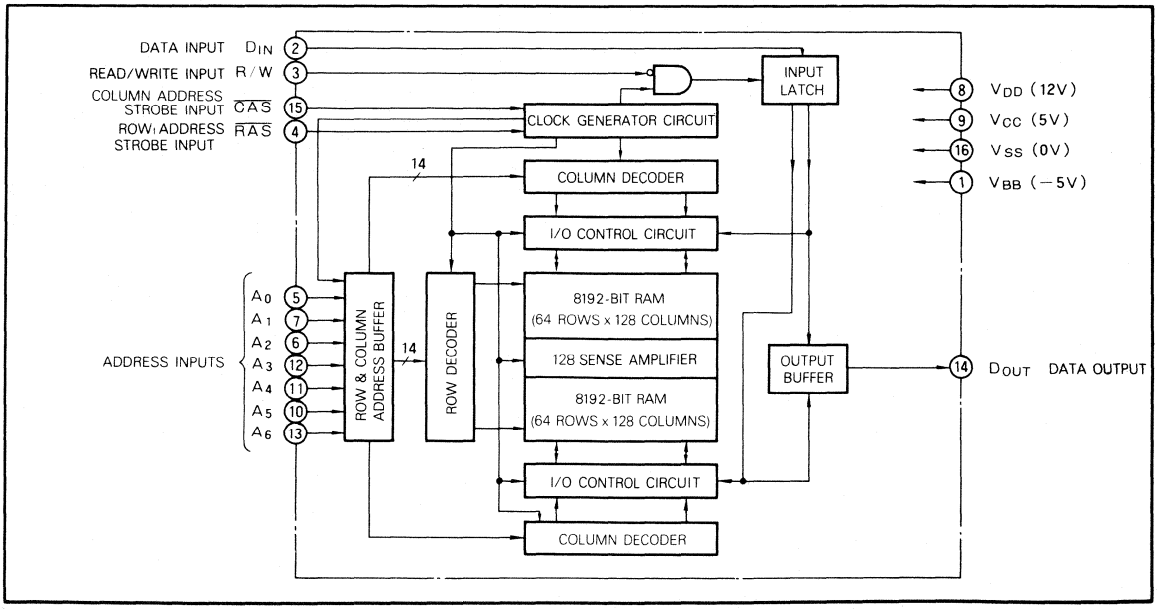
Table 2 compares that the requirements of the two RAM types when a 16K-byte memory system is constructed.

Table 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 Block diagram



14

APPLICATION OF 16K-BIT DYNAMIC RAM

(M5K 4116P, S)

FUNCTIONS

In addition to normal read, write, and read-modify-write operations, the M5K 4116P, 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 3.

If you interchange address pins as shown in Fig. 3, you can get a sequential location map for the 16,384 memory bits.

Table 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. 3 Method for converting sequential address

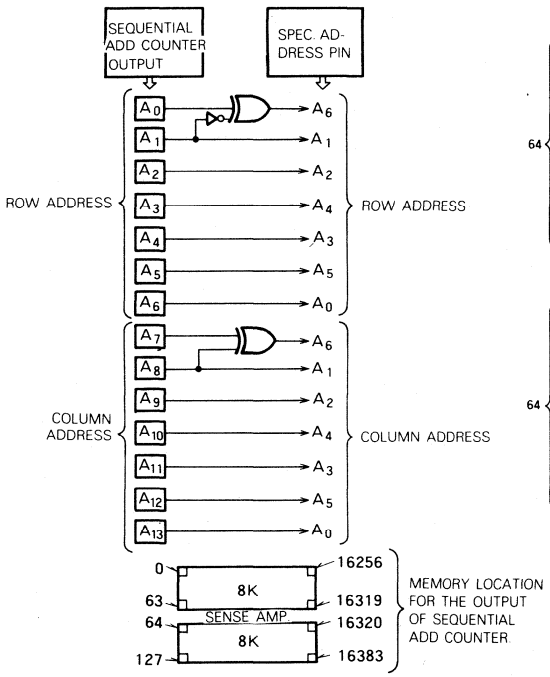
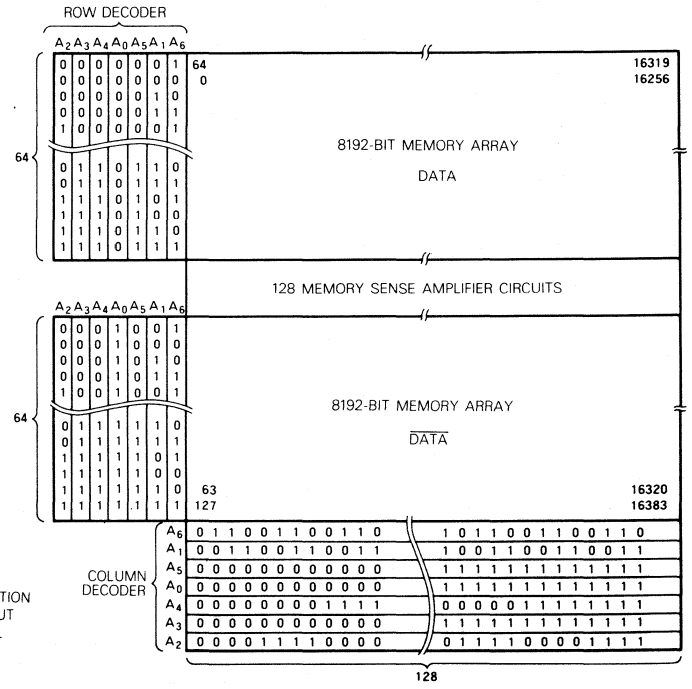


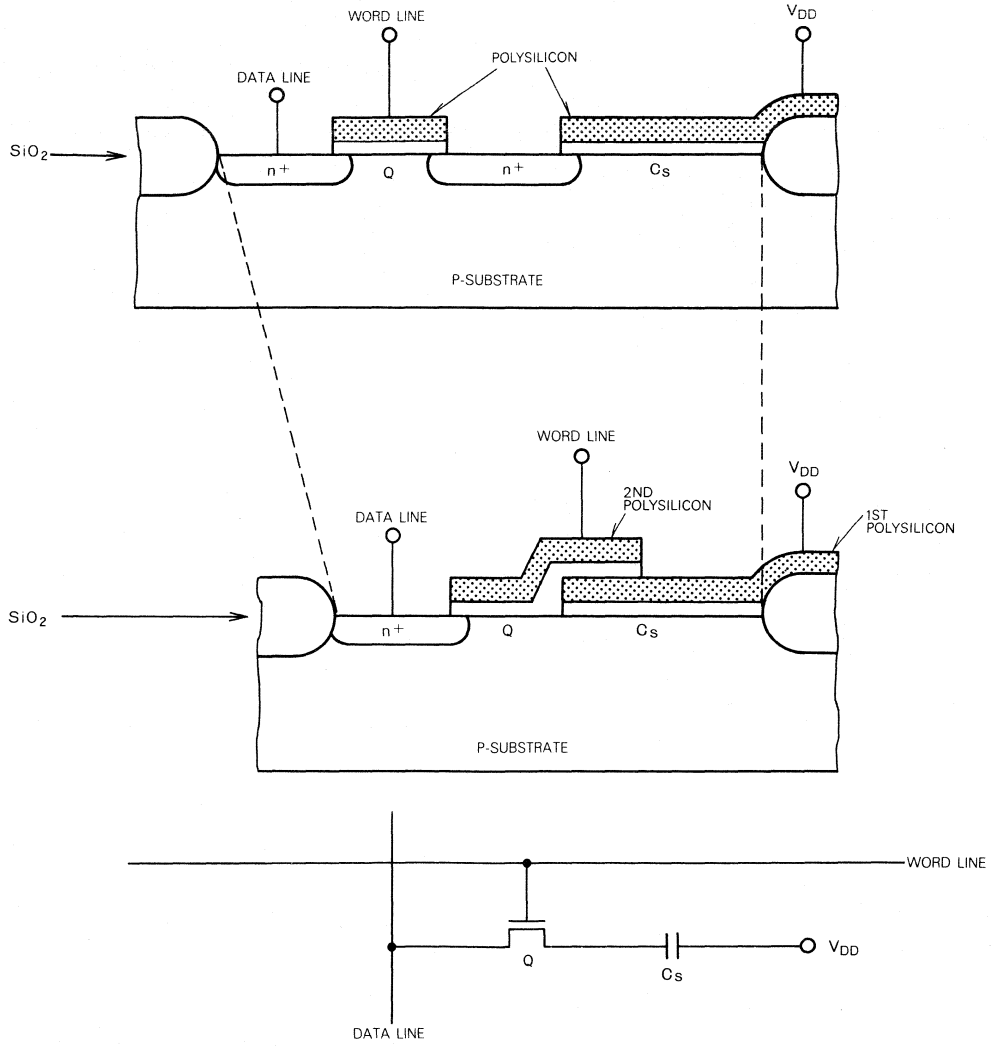
Fig. 4 M5K 4116P, S memory map



N-CHANNEL DOUBLE-LAYER POLY-SILICON GATE MOS PROCESS

In order to fabricate the M5K 4116P, S series, single transistor memory cells and the N-channel double-layer poly-silicon 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 poly-silicon gate MOS process, so that the memory cell area is reduced by 75% from that of the previous process.

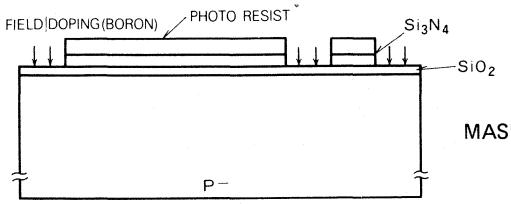
Fig. 5 Structure of memory cell



APPLICATION OF 16K-BIT DYNAMIC RAM

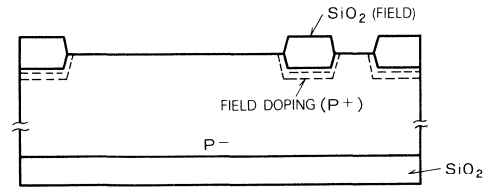
(M5K 4116P, S)

Fig. 6 Wafer manufacturing process

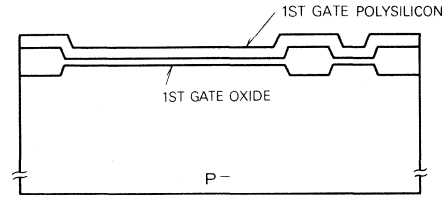


MASK 1

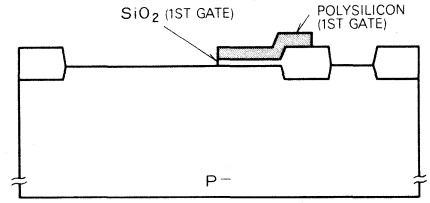
Deposit silicon nitride (Si_3N_4) on wafer and coat with photo resist.
Define active area with MASK 1 and implant fields.



Remove photo resist and grow field oxide using nitride as mask.

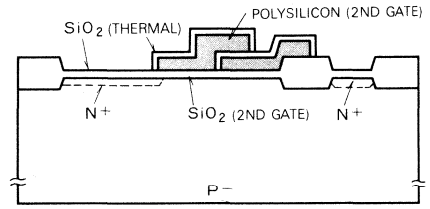


Grow 1st gate oxide and deposit 1st polysilicon.



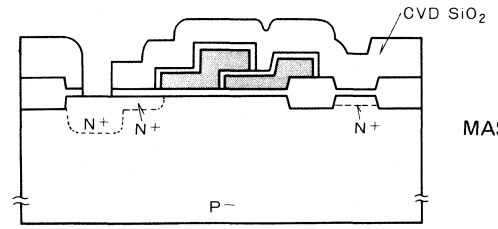
MASK 2

Define 1st polysilicon gate using MASK 2.



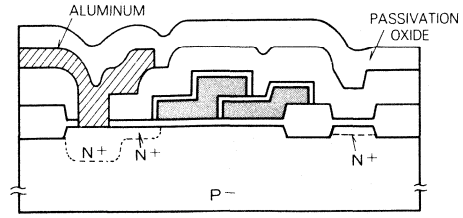
MASK 3

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.

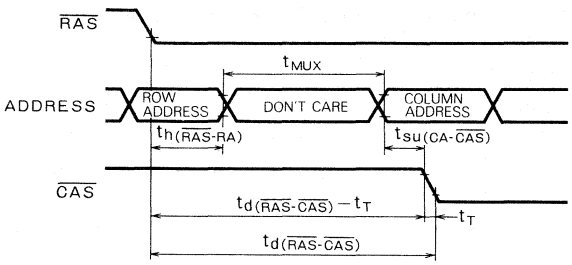
MITSUBISHI ELECTRIC

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. 7 Address multiplex



1. The delay time from $\overline{RAS}$ to $\overline{CAS}$ $t_{d(\overline{RAS}\text{-}\overline{CAS})}$ is set between the minimum and maximum values of the limits. In this case, the internal $\overline{CAS}$ control signals are inhibited until almost $t_{d(\overline{RAS}\text{-}\overline{CAS})max}$ ('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 easily be changed from row address to column address. This interval is called the 'multiplex time'. Eq. 1 gives the multiplex time.

$$t_{MUX} = t_{d(\overline{RAS}\text{-}\overline{CAS})} - t_T - t_{h(\overline{RAS}\text{-}RA)} - t_{SU(CA\text{-}\overline{CAS})} \quad \text{.... Eq. 1}$$

In the next conditions, the multiplex time (t_{MUX}) is maximized.

$$t_{d(\overline{RAS}\text{-}\overline{CAS})} = max$$
$$t_{h(\overline{RAS}\text{-}RA)} = min$$
$$t_{su(CA\text{-}\overline{CAS})} = min$$

Table 4 shows the maximum multiplex time in the case where the access time is not greater than $t_{a(\overline{RAS})MAX}$.

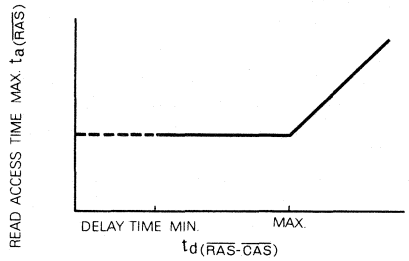
Table 4 Maximum multiplex time

Type number	t_{MUX}	$t_{d(\overline{RAS}\text{-}\overline{CAS})}$	$t_{h(\overline{RAS}\text{-}RA)}$	$t_{su(CA\text{-}\overline{CAS})}$
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 $t_{d(\overline{RAS}\text{-}\overline{CAS})}$ 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. 8 Read access time vs. delay time

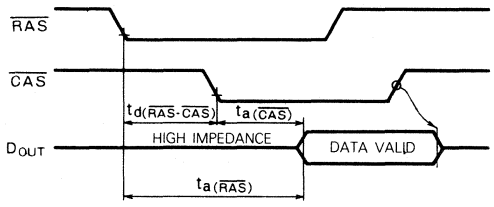


Data Input

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. 9 Read cycle



APPLICATION OF 16K-BIT DYNAMIC RAM

(M5K 4116P, S)

The output of the M5K 4116P, 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. 10 Write cycle

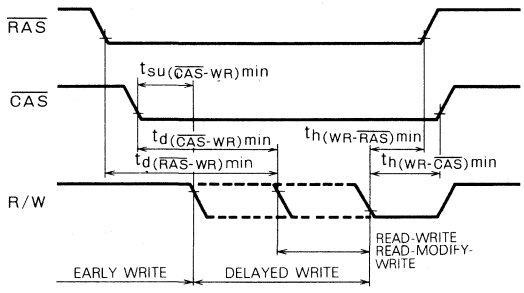


Table 5 Output state in write cycle

Operation mode	Output state
Early write	High impedance
Read-write, read-modify-write	Data valid
Others	Unspecified

These output conditions of the M5K 4116P, 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 M5K 4116P, 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 M5K 4116P, 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 I_{BBSB} (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. 11 Distribution of average I_{DD}

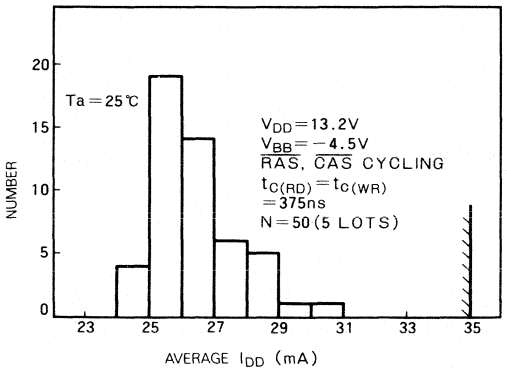
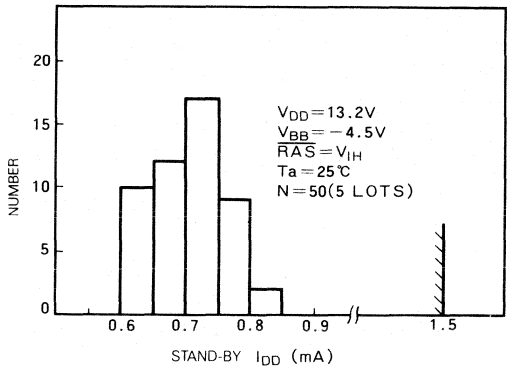


Fig. 12 Distribution of stand-by I_{DD}



2. RAS-only refresh

$$I_{DDSB}=I_{DD3(AV)} \times \left\{128 \times \frac{t_C}{t_{C(REF)}}\right\} + I_{DD2} \times \left\{1-(128 \times \frac{t_C}{t_{C(REF)}})\right\} \quad \text{.... Eq. 4}$$

$$I_{BBSB}=I_{BB3(AV)} \times \left\{128 \times \frac{t_C}{t_{C(REF)}}\right\} + I_{BB2} \times \left\{1-(128 \times \frac{t_C}{t_{C(REF)}})\right\} \quad \text{.... Eq. 5}$$

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 M5K 4116P, 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 RAS/CAS refresh cycles or RAS-only refresh cycles.

Table 6 Change of stand-by current

Device / Condition	#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	%

APPLICATION OF 16K-BIT DYNAMIC RAM

(M5K 4116P, S)

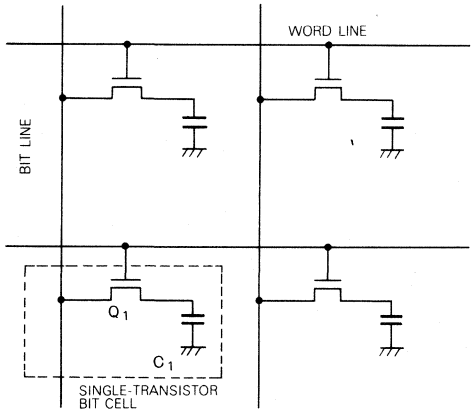
APPLICATIONS FOR DYNAMIC RAM

Dynamic RAM (Random-Access Memory) can be a very effective component in the implementation of reliable, high-performance, low-cost memory systems. However, this device has 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. 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. 13 Single-transistor memory bit cell



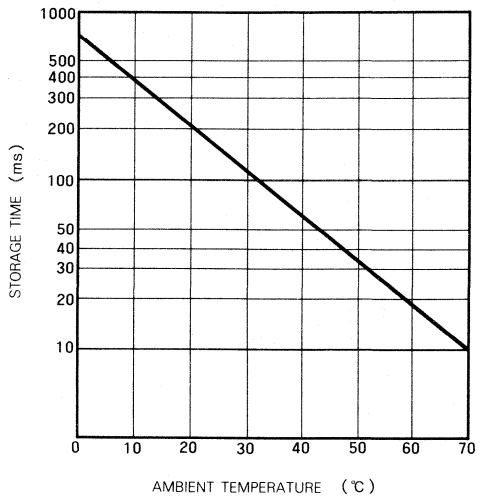
During the write operation, the selected word line is brought to an active state (high). This causes the bit cell transistor Q_1 to turn "On" and the data that is placed on the bit line is stored in the capacitor C_1. The stored data is retained even if transistor Q_1 turns "Off".

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 M5K 4116P, S is shown in Fig. 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. 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 M5K 4116P, 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 imputs 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

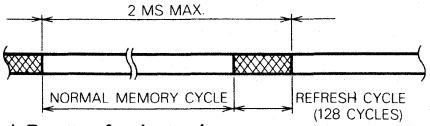
In most memory systems it is difficult to guarantee that normal memory operations will cause all the rows within a memory to be sensed within the specified refresh interval. For this reason, most dynamic memory systems have special circuitry that will cause all rows of memory cells to be sensed within 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. 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. 15 (b). The third is called "invisible or trans-

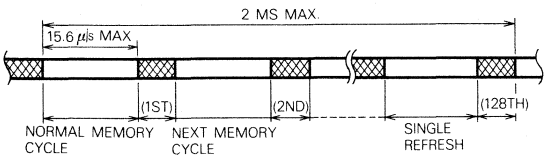
parent mode," where refresh cycles are introduced at the times when the memory is not being accessed and thus refresh is invisible to the processor. (The processor sees no delay due to the refresh function.) This mode is shown in Fig. 15 (c). The memory cycle of the invisible refresh mode is generally longer than that of the first or second method because single memory access continues after single memory access.

In designing dynamic memory systems, it is important to decide whether the memory refresh will be synchronous or asynchronous. In synchronous refresh, the designer uses a system clock to trigger the refresh function. In asynchronous refresh, the designer must provide for a local timer to trigger the refresh. With an asynchronous refresh, there will usually be cases when a system memory request and local refresh request occur simultaneously. To resolve these simultaneous asynchronous requests, some arbitration logic must be present in the memory controller. This block diagram is shown in Fig. 16.

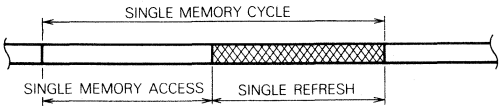
Fig. 15 Refresh techniques



(a) Burst refresh mode

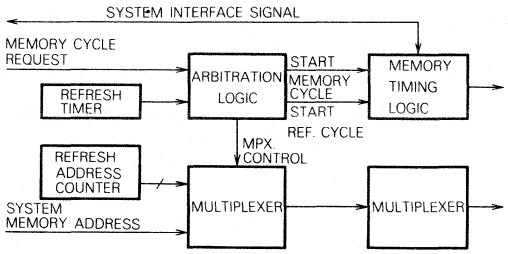


(b) Cycle steal refresh mode



(c) Invisible refresh mode

Fig. 16 Memory Controller Block Diagram



An example of address multiplexing logic is shown in Fig. 17. In the memory cycle, row address (A0 ~ A6) or column address (A7 ~ A13) are multiplexed by the signal MUX. CONTROL. In the refresh cycle, signal REFRESH goes high and the refresh address is placed on the A0 ~ A6 outputs. The $\overline{RAS}$ and $\overline{CAS}$ signals are introduced by $\overline{RAS}$ ENABLE and $\overline{CAS}$ ENABLE. Typical memory timing of this type is shown in Fig. 18. The designer must design a memory control logic which will suit the desired system bus. Fig. 19 shows M5K 4116P, S 64K x 8-bit memory array.

Fig. 17 Address multiplexing logic

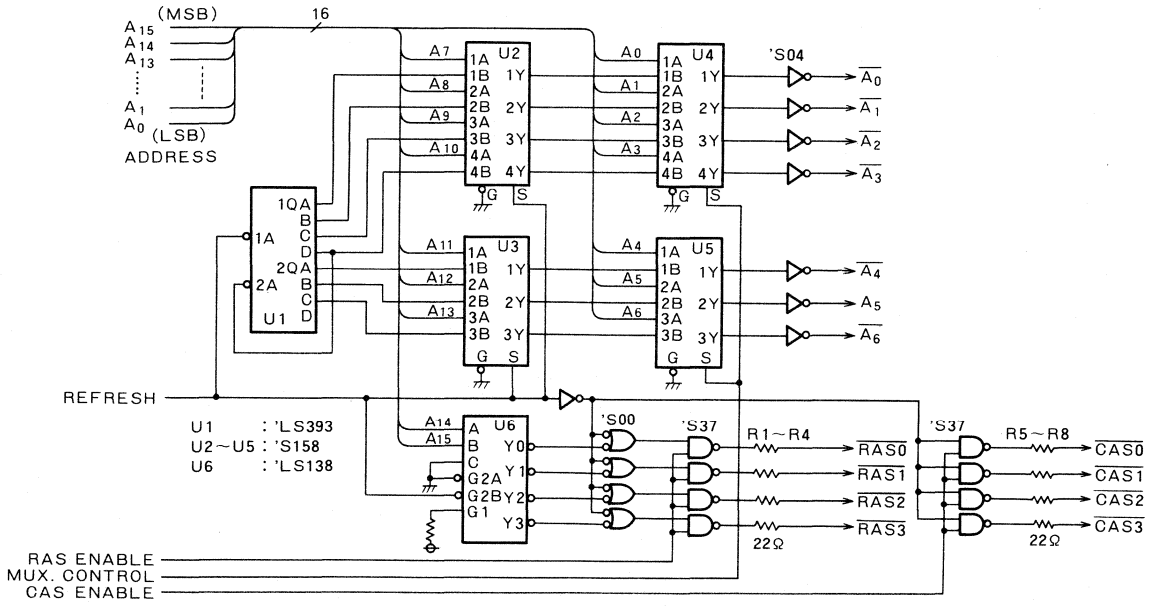
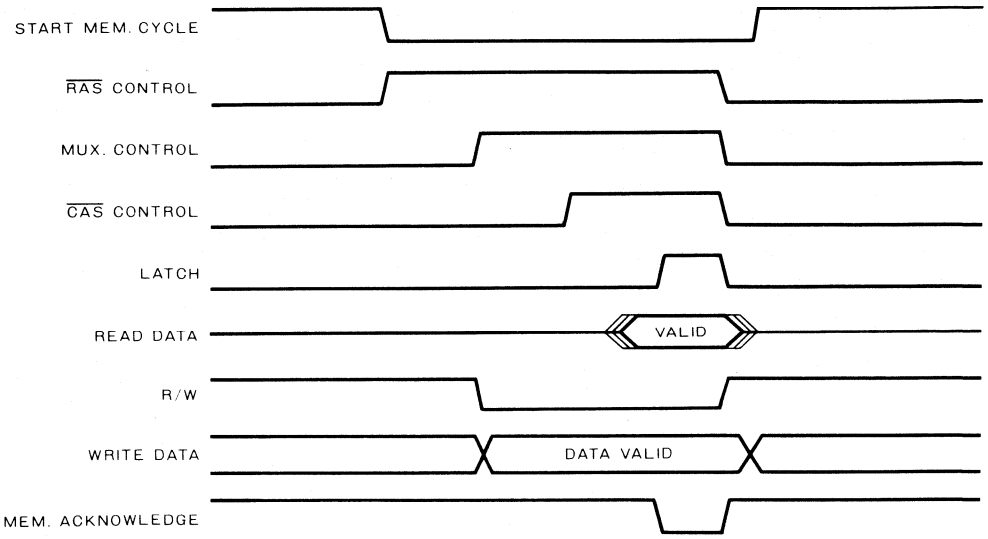
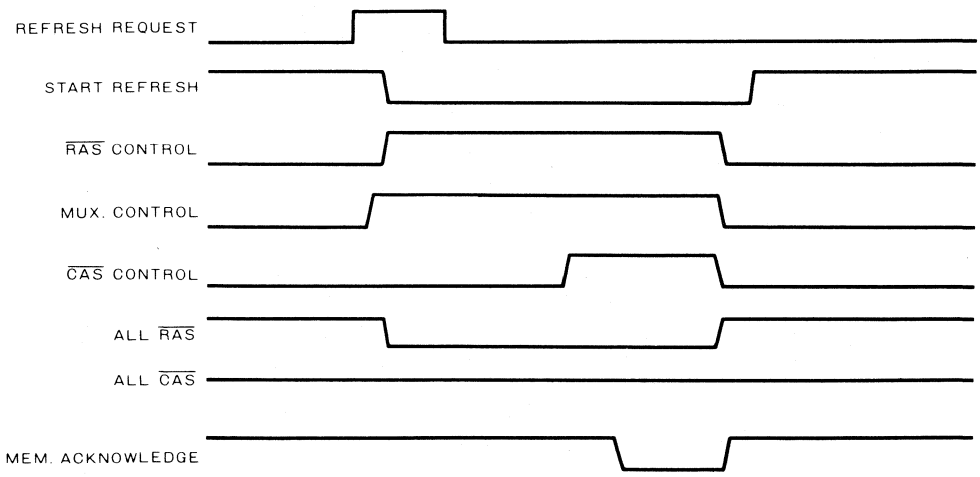


Fig. 18 Example of memory timing

(a) Read-write cycle time



(b) Refresh cycle time

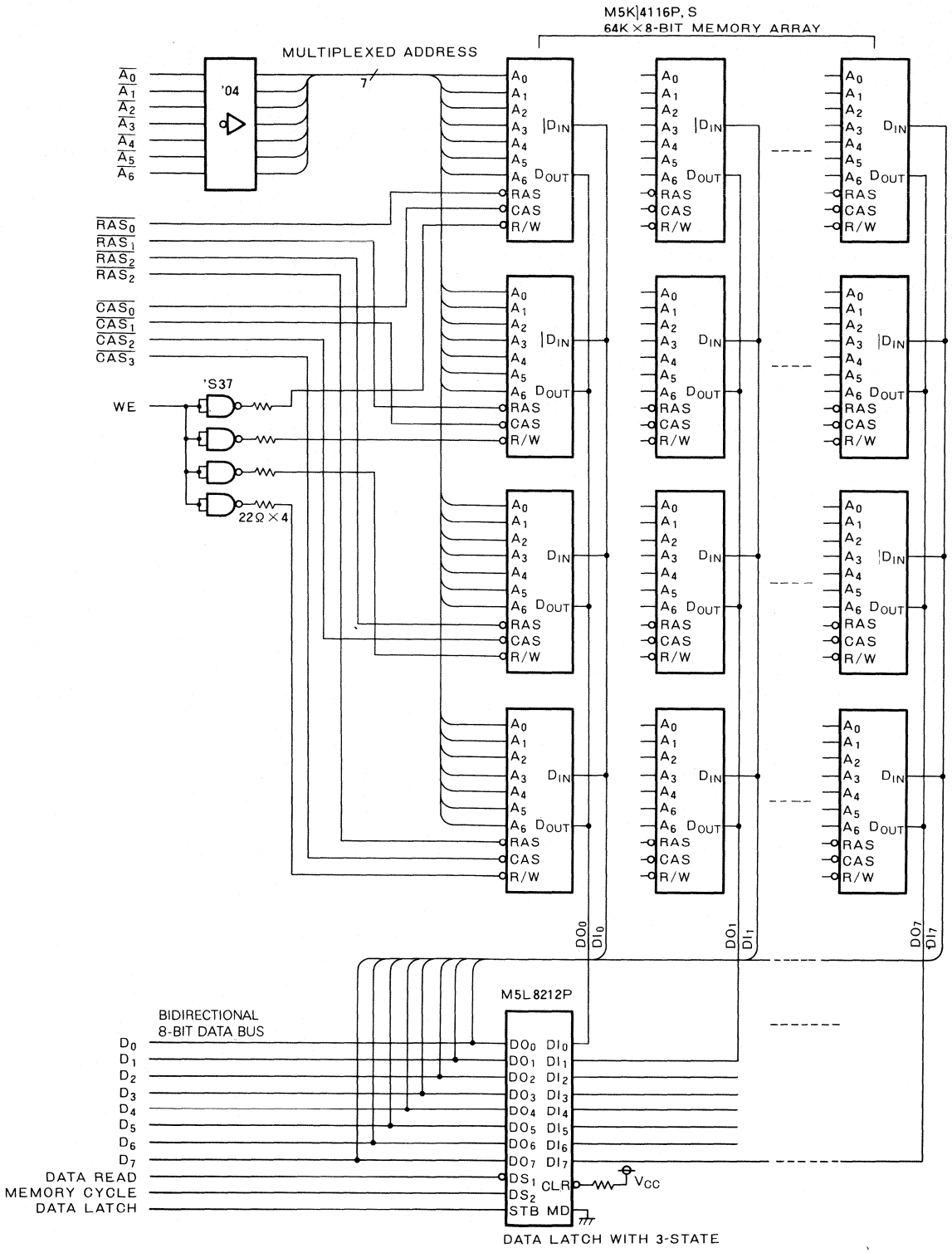


Note 4 : START MEM. CYCLE is generated by memory timing logic when the timing logic receives the request (read/write) from the system.

5 : If the memory control logic receives the request at a time when memory is under refresh cycle, MEM. ACKNOWLEDGE must go low until the refresh cycle is completed.

6 : $\overline{CAS} = V_{IH1}$

Fig. 19 M5K4116P, S 64K X 8-bit memory array



APPLICATION OF 16K-BIT DYNAMIC RAM

(M5K 4116P, S)

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 insure 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 M5K 4116P, S are shown in our data sheets. Distribution and decoupling techniques must be used to suppress these noises, which can cause data loss.

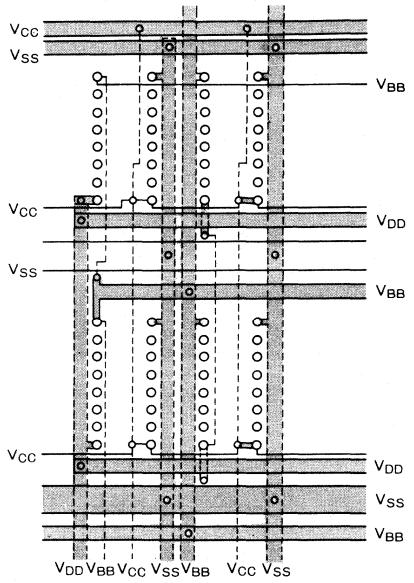
The layout should have an effectively gridded power-supply distribution network to supply adequate current and to minimize inductive effects. The distribution of circuit grounding is most important in reducing ground noise and inductive effects, and to provide a ground plane for the signal lines. An example of the power grid of the M5K 4116P, S is shown in Fig. 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 in to 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.

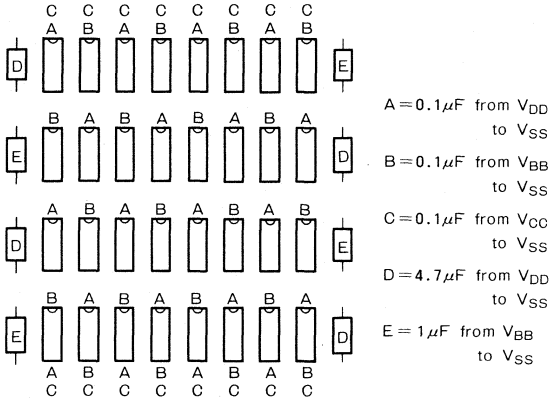
In addition to the ceramic capacitor, it is recommended that a 2$\sim$5μF tantalum or equivalent capacitor be connected between V_{DD} and V_{SS} adjacent to the array for each group of 16 memory devices. Use of a slightly smaller-value bulk capacitor is also recommended between V_{BB} and V_{SS}. An example of capacitor placement is shown in Fig. 21.

Fig. 20 Suggested power grid for M5K41116P, S



Note 7 : The dotted lines show the soldered side of the P.C. board.

Fig. 21 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 P.C. 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.

APPLICATION OF CMOS STATIC RAMs

(M5L 5101LP 1K-BIT, M58981S 4K-BIT) FOR DESIGNING NON-VOLATILE MEMORY SYSTEMS

INTRODUCTION

Mitsubishi M5L 5101L P and M58981 S are static RAMs that are fabricated with a CMOS technology. The M5L 5101L P is organized as 256 words of 4 bits, and the M58981 S 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 outline of various circuit techniques for battery-supported non-volatile memory systems. Electronic information regarding the two RAMs can be found in the previous pages.

NON-VOLATILE MEMORY SYSTEM

We can relatively design a large non-volatile memory system with a small additional interface logic by using CMOS RAMs. The block diagram of the basic computer system that uses CMOS RAMs is shown in Fig. 1, and the power supply on-off timings of the system are shown in Fig. 2. 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 down. And after the RAMs have been protected, their power source is replaced V_{CC} by V_{BAT}, as shown in Fig. 2.

Fig. 1 Non-volatile memory system

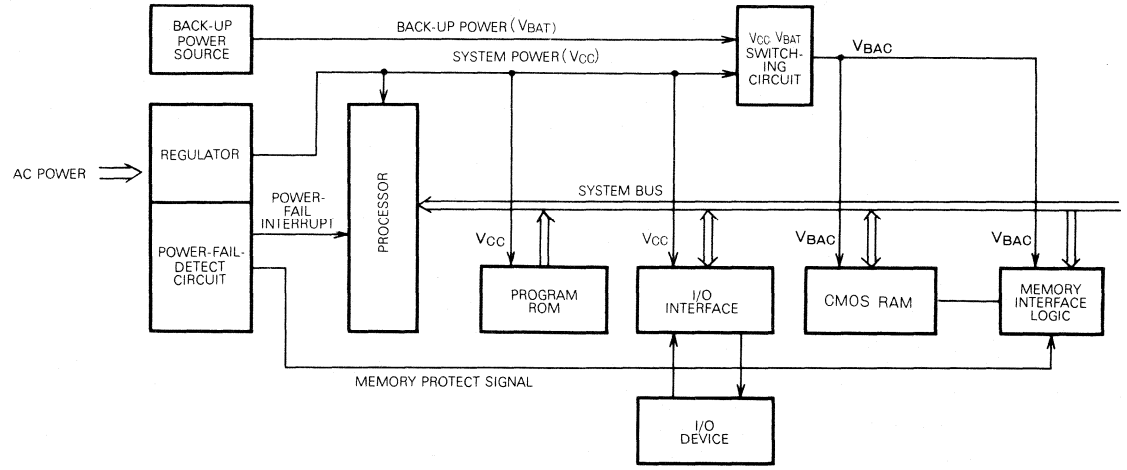
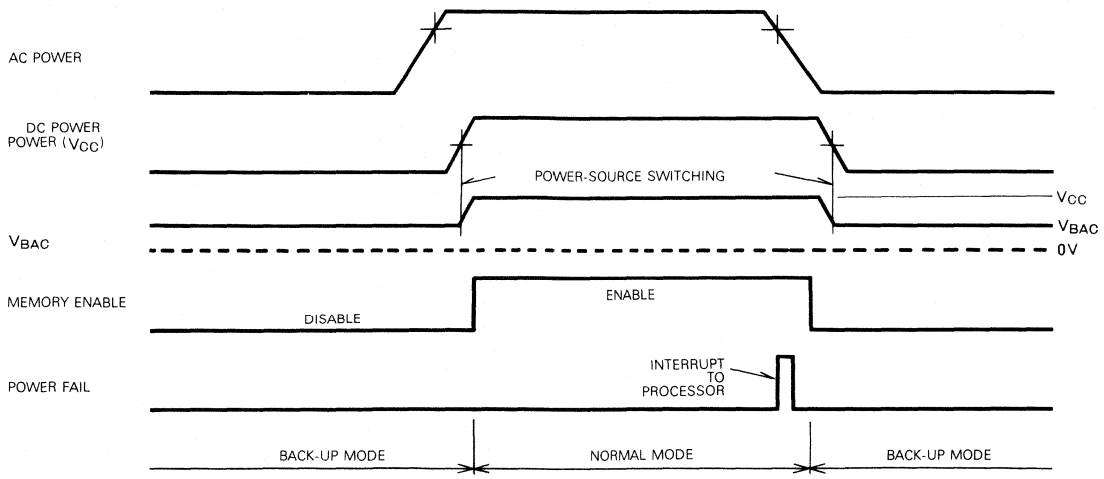


Fig. 2 Power on-off timing



APPLICATION OF CMOS STATIC RAMs

(M5L 5101LP 1K-BIT, M58981S 4K-BIT) FOR DESIGNING NON-VOLATILE MEMORY SYSTEMS

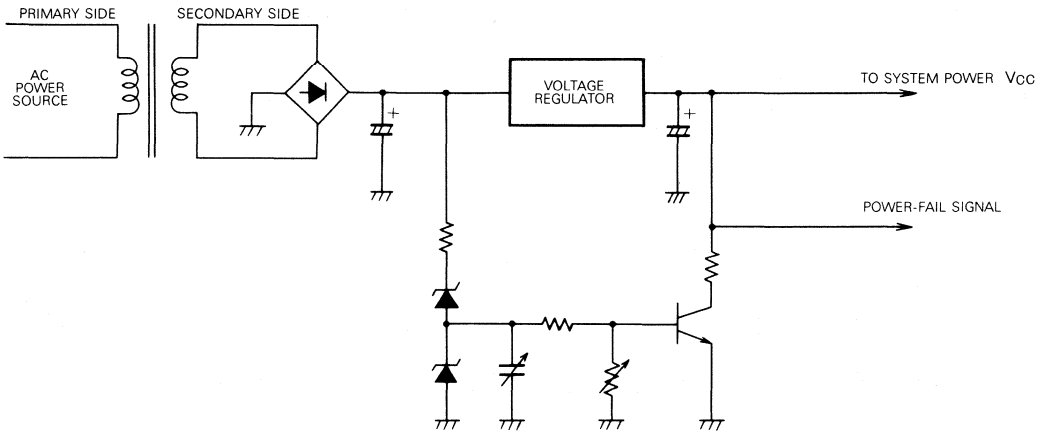
EXAMPLE OF CMOS NON-VOLATILE MEMORY SYSTEM

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. 3 shows the abstract 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. 3 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 the 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. 4 and Fig. 5. The diode-coupled circuit in Fig. 4 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 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 staturation for Q1.

Fig. 4 Diode-coupled switching circuit

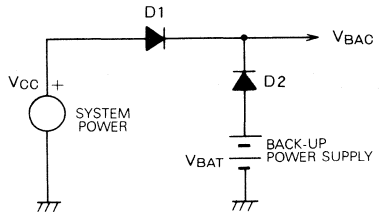
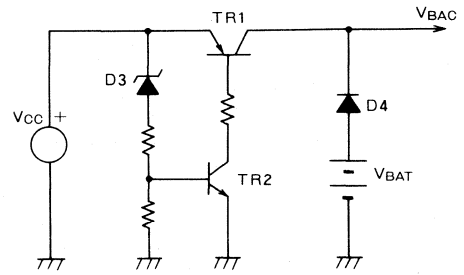


Fig. 5 Transistor-coupled switching circuit

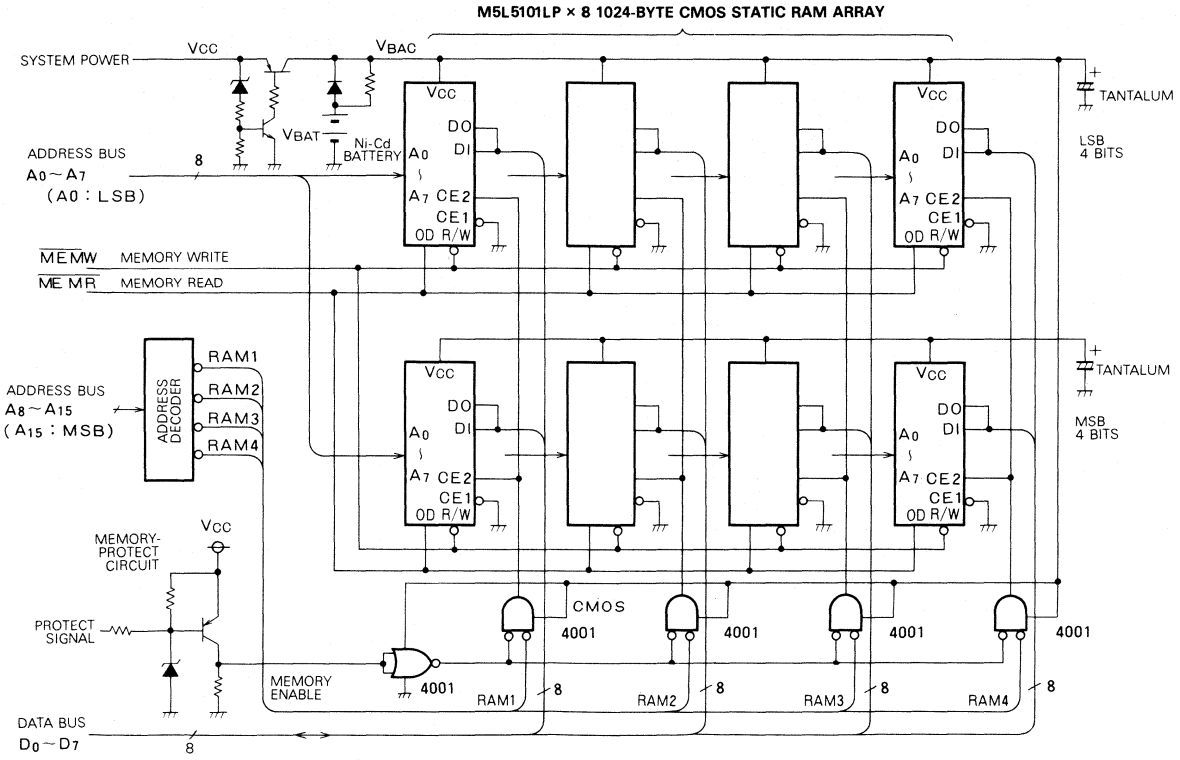


TYPICAL APPLICATION CIRCUIT

An Example of M5L 5101L P Application

An example of a 1K-byte non-volatile M5101LP memory system is shown in Fig. 6. 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. 3) to produce the memory-protect signal that protects RAMs at the time when the V_{CC} is dropping or rising as shown in Fig. 2. The CE2 terminal 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. 6 Example of M5L 5101LP



APPLICATION OF CMOS STATIC RAMs

(M5L 5101LP 1K-BIT, M58981S 4K-BIT) FOR DESIGNING NON-VOLATILE MEMORY SYSTEMS

An Example of M58981 S

The M58981 S is a CMOS RAM which is full pin compatible with M5L 2114L P, S and is organized as 1024 words of 4 bits. The M58981 S 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. 7 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 M58981 S is shown in Fig. 8.

Fig. 7 Power on-off timing (M58981 S)

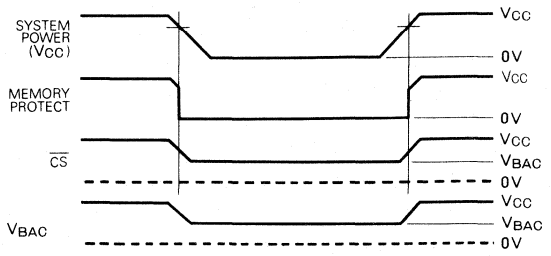
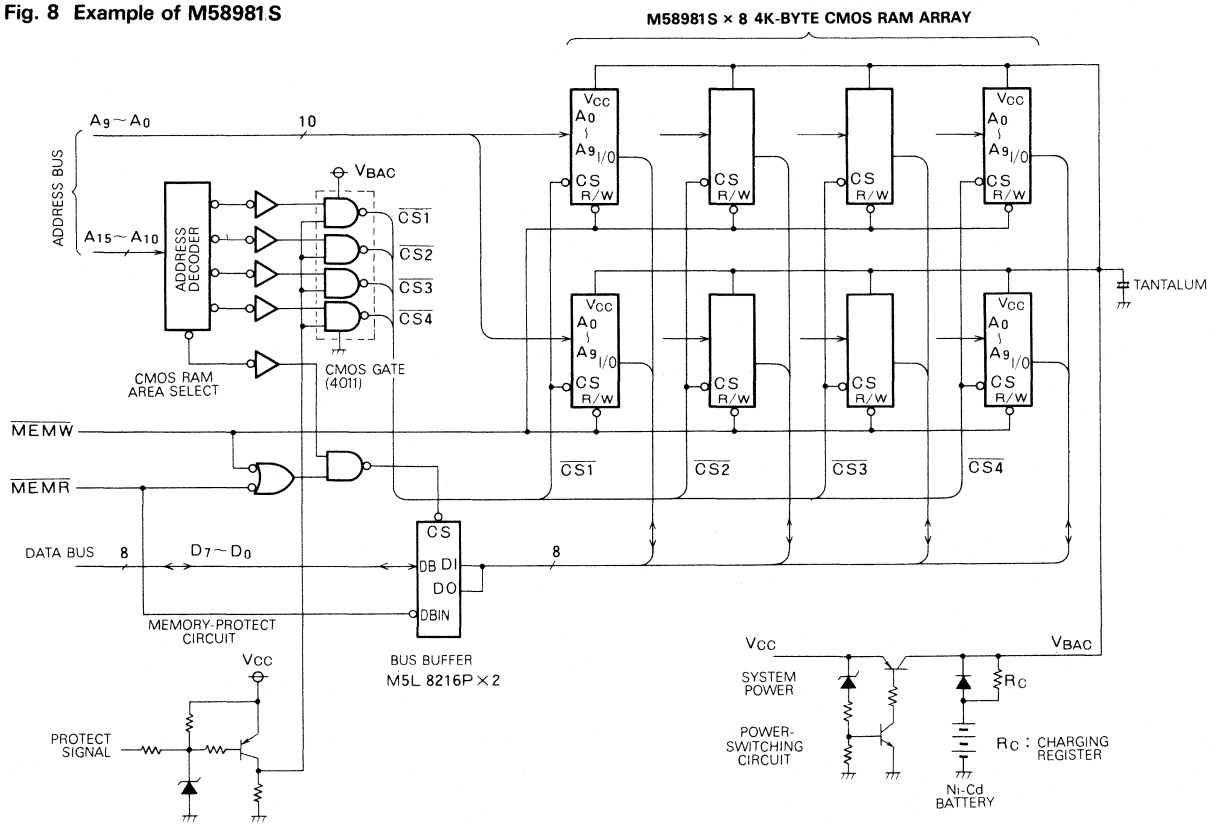


Fig. 8 Example of M58981 S



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 the details of this, see related articles.
2. In order to decrease DC power-source impedance, decoupling capacitors whose leak currents are small should be used. It is also necessary to place 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. 6 and Fig. 8, it should lbe 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.

APPLICATION CIRCUITS FOR 1K-BIT NON-VOLATILE STATIC RAM

(M58656S)

DESCRIPTION

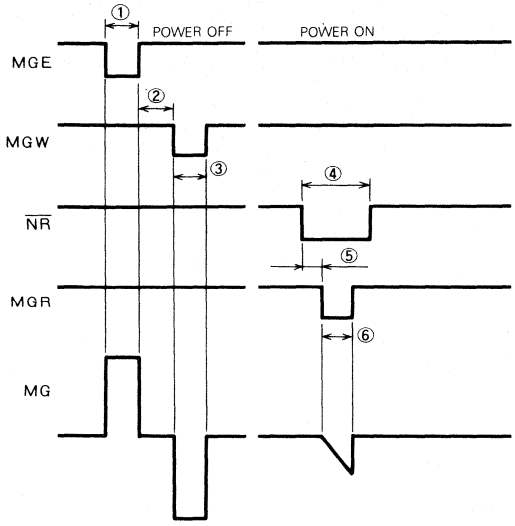
To operate the M58656 S as a non-volatile random-access memory, it is necessary to apply erase, write and read signals to the memory gate (MG) terminal.

Since the operation is executed at a transient state of power on-off, circuit configuration of power supply is critical.

For the first requirement, an MG signal driver circuit with non-volatile read mode 2 is given as an example, and for the second requirement, an example of a power cut-off detection circuit is shown.

EXAMPLE OF MG SIGNAL DRIVER CIRCUIT

Timing Diagram

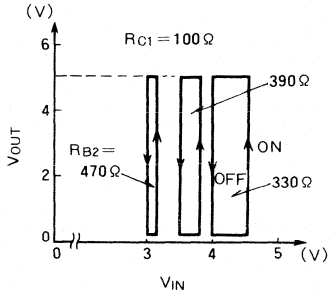


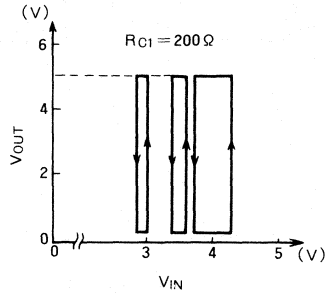
① Setting of erase pulse width $t_{W(MGE1)}$
② Pulse interval between erase and write determined by load capacitance (R_1 is variable)
③ Setting of write pulse width $t_{W(MGW1)}$
④ Setting of $\overline{NR}$ signal low-level duration
⑤ Setting of $t_{su(\overline{NR}\text{-}MG)}$
⑥ Pulse width setting for read signal
The read signal is determined by ⑥ and the values of R_2 and C_1.

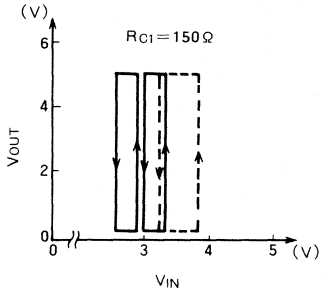
Schmitt Trigger Circuit

The on-off detection voltage of power supply V_{CC} is variable depending on the values of R_{C1}, R_{B2}.

The data for reference is as shown below. (V_{CC} = 5V, V_{DD} = −15V)







Supply Current

Power supply	Voltage condition	Current
V_{CC}	5V	150mA
V_{DD}	−15V	50mA
V_{MGE}	34V	0.3mA
V_{MGW}	−23V	0.3mA

Note 1 : The above measurements are at Ta = 25°C with four M53322Ps as TTL.

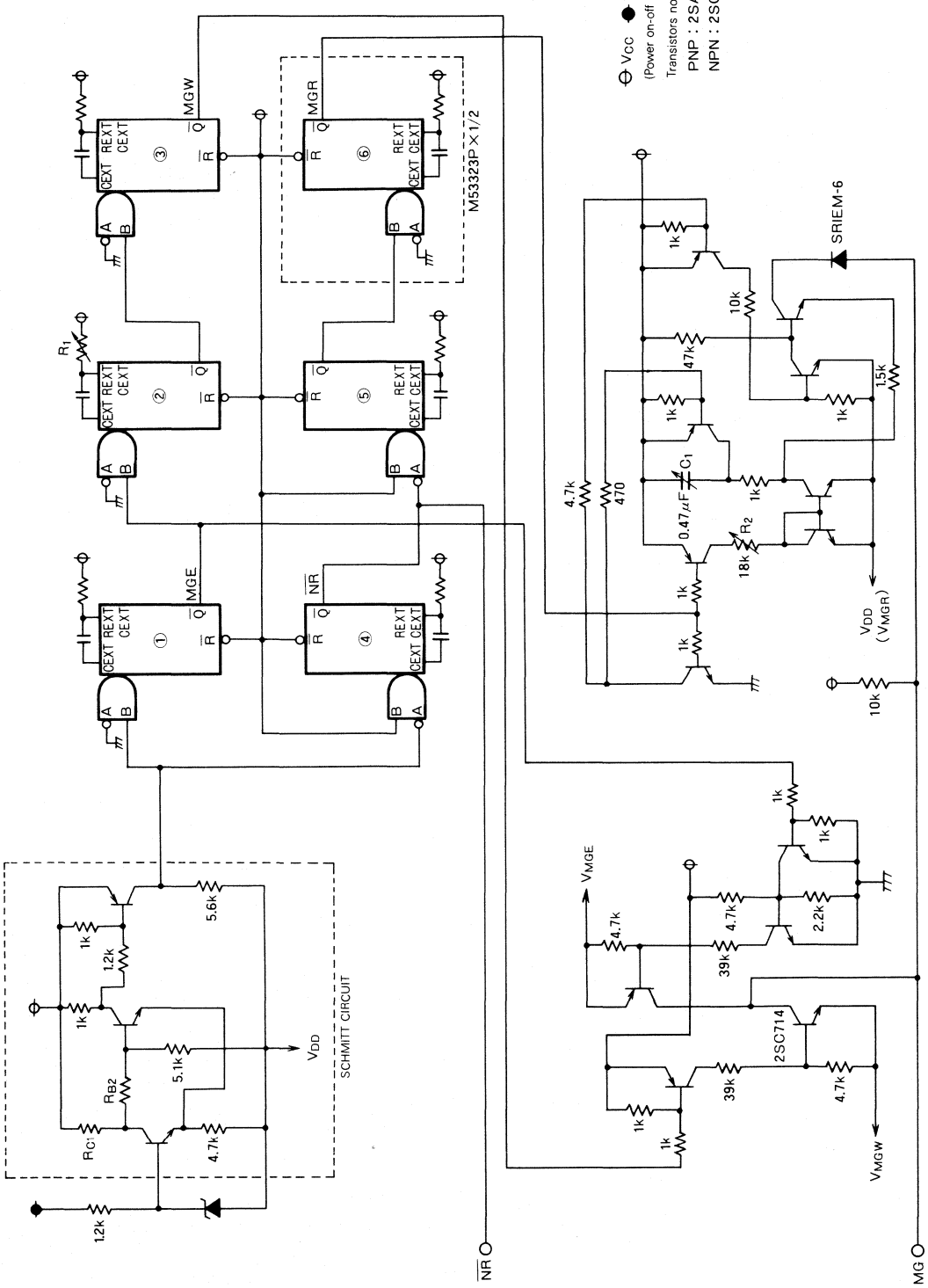
Other Recommendations

1. For gate protection, connect MG terminal to MG signal through 1 kΩ resistance in series.
2. Connect TTL output to input terminal through 1 kΩ resistance placed between V_{SS}.

APPLICATION CIRCUITS FOR 1K-BIT NON-VOLATILE STATIC RAM

(M58656S)

Fig. 1 Example of MG signal driver circuit



MITSUBISHI ELECTRIC

APPLICATION CIRCUITS FOR 1K-BIT NON-VOLATILE STATIC RAM

(M58656S)

Power Cut-Off Detection Circuit

For non-volatile configuration of the memory system, the non-volatile mode must be realized during the period from the detection of abnormal power supply to the actual drop of the power. That is, the specified voltage must be maintained to the point where erase and write modes of non-volatile operation are completed.

For the detection of abnormal power supply, therefore, an independent circuit apart from the stabilized power supply must be provided so as to report abnormal conditions to the system before the stabilized power supply (system power) starts to drop.

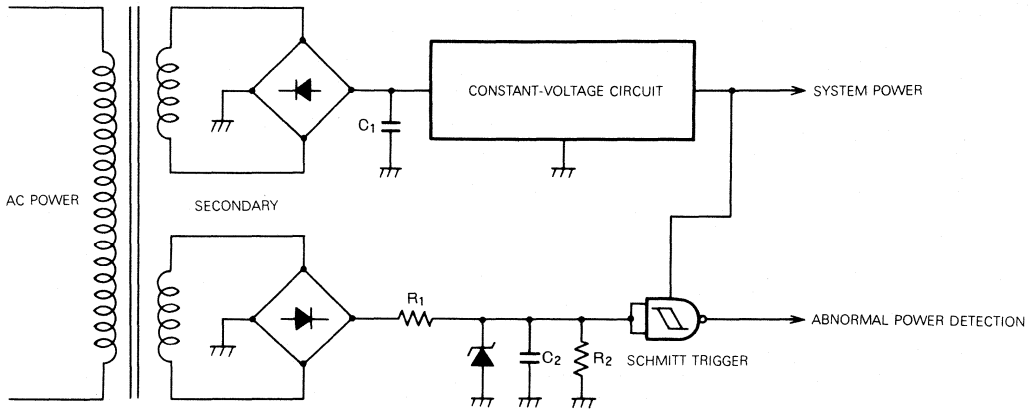
Fig. 2 shows an example of such a detection circuit.

The detection point of abnormal power is near the secondary, where the Schmitt trigger functions to warn of power cut-off when the rectified secondary voltage falls below the set voltage by the Zener diode.

The C_2 and R_2 are so adjusted as not to sense the voltage variation within the range not influential over RAM operation but to warn of abnormal variation before the system power falls.

Abnormal power is processed during the time lag between the detection and the drop of the system power due to the delay of the C_1 and a constant-voltage circuit.

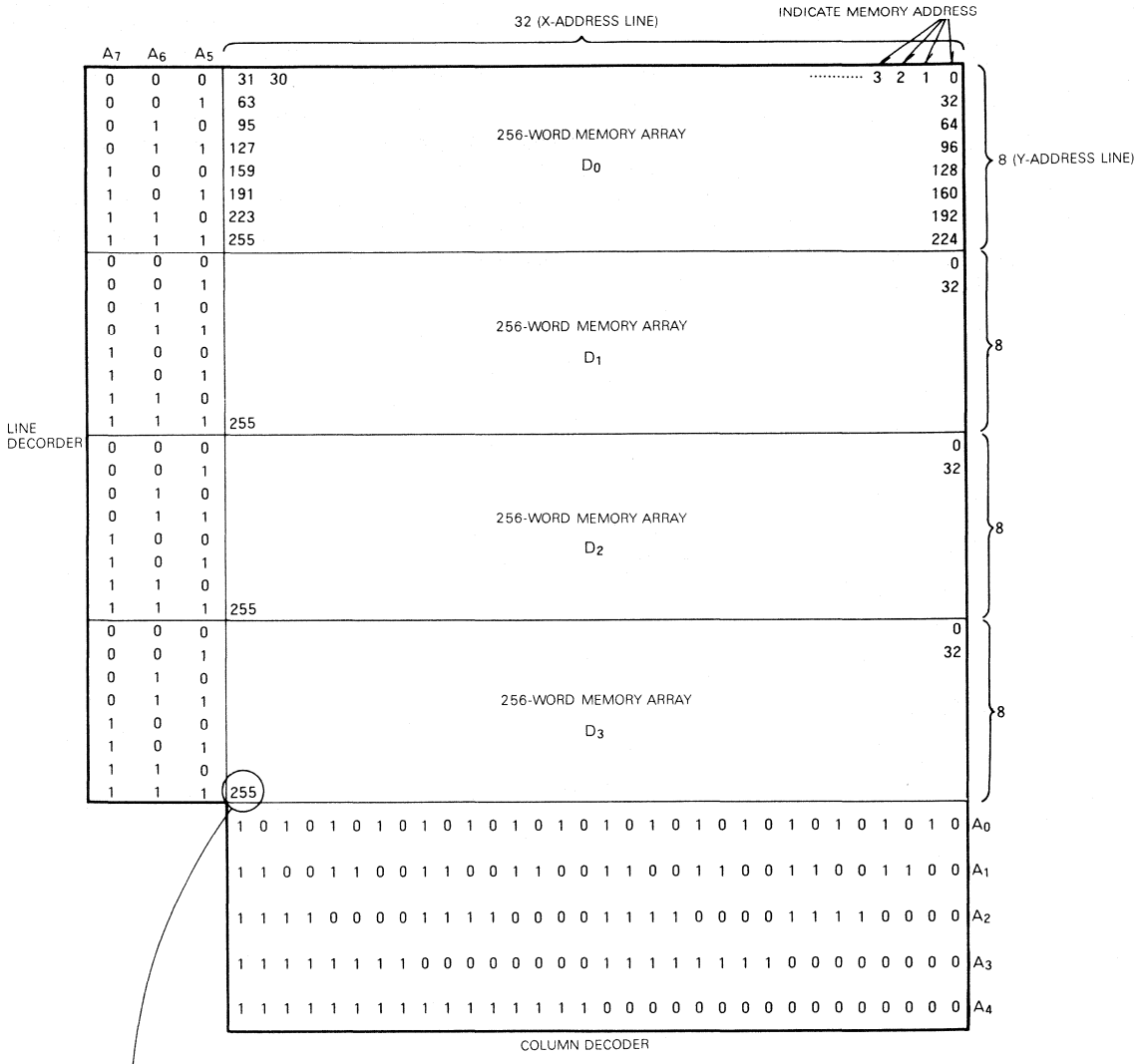
Fig. 2 Typical detection circuit



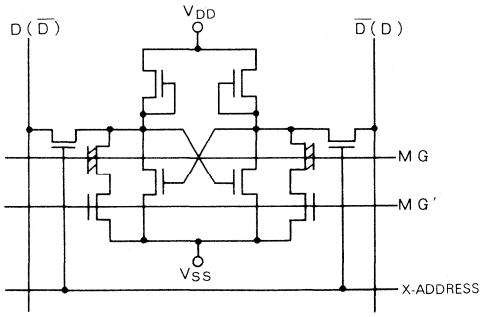
APPLICATION CIRCUITS FOR 1K-BIT NON-VOLATILE STATIC RAM

(M58656S)

M58656S Memory Cell Configuration



Memory Cell Circuit (for one word)



Note 2 : As for the address, "0" is for high-level and "1" is for low-level.
3 : In the memory cell circuit, the flip-flop node is not fixed either at "D" or at "D̄"

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:

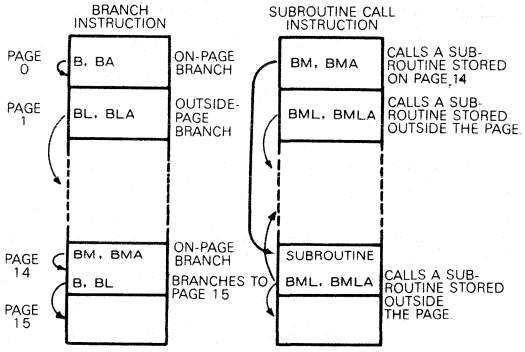
BM, BMA, BML, BMLA

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

SUBROUTINES

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

$$|V_{ref}| > |V_{K(Y)}| \quad \text{i.e. } J_{(Y)} = 1$$

the MSB of the H-L register is reset to "0".

If

$$|V_{ref}| < |V_{K(Y)}| \quad \text{i.e. } J_{(Y)} = 0$$

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).

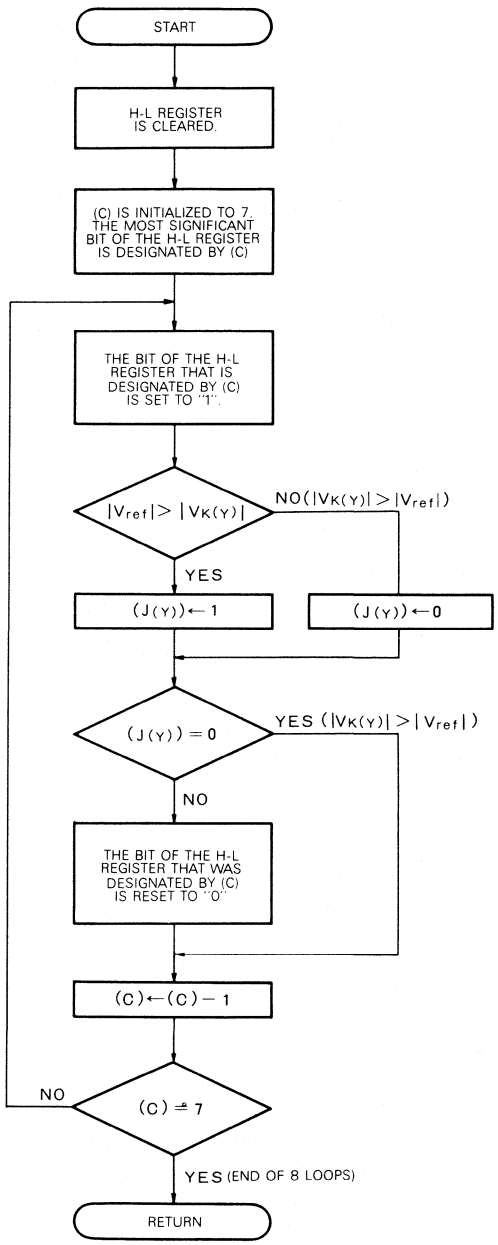
This successive approximation method has a constant conversion speed—approximately 0.6ms at 600kHz—and thus it is suitable for examining analog value with large variations and detecting different analog values from multiple channels.

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 for the port K_0 in the following example.

```
LXY  0, 0
BM   ADC1
```

Fig. 2 A-D conversion subroutine flow chart for the successive approximation method



MITSUBISHI ELECTRIC

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 (C_y) is set to "1"

and if

$|V_{ref}| < |V_{K(Y)}|$ then (C_y) 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.

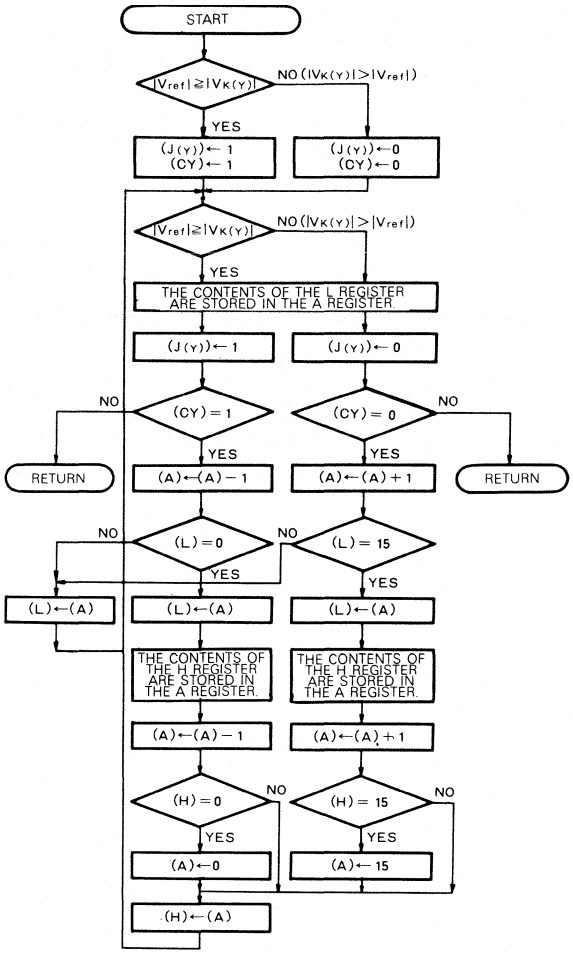
It will test (H) = 0, when $V_{ref} = \frac{0}{256}V_{REF}$ is met, and will test (H) = 15, when $V_{ref} = \frac{256}{256}V_{REF}$ is met.

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.



SUBROUTINES

Fig. 4 ADC1 program list

```
 2                          ORG   E,0
 3                    *
 4                    *
 5                    MAX   EQU   7
 6                    MIN   EQU   12
 7                    J     EQU   0
 8                    SIGN  EQU   12
 9                    *
10                    *
11                    ##########  ( MELPS 4 LIBLARY NO.1 )  ##########*
12                    *
13                    ************************************************************
14                    *SUBR: ADC1  8-BIT A-D CONVERSION, BY SUCCESSIVE APPROXIMA.*
15                    ************************************************************
16     00       0B0   ADC1  LA    0       CLEAR A, (A)=0
17     01       019         TLA           CLEAR L, (L)=0
18     02       059         THA           CLEAR H, (H)=0
19     03       057         LC7           (C)=7
20     04       042   ADC10 SHL           SET H-L, BIT IS ASSIGNED BY (C)
21     05       008         CPA           COMPARE PORT K & VREF
22     06       000         NOP           SET J IF ABS VREF.GT.ABS VK(Y)
23     07       029         SZJ           SKIP IF (J(Y))=0
24     08       052         RHL           RESET H-L
25     09       009         DEC           (C)=(C)-1,SKIP IF (C)=0
26*W0*0A        104         BM    ADC10   REPEAT 8 TIMES
27     0B       044         RT            END OF ADC1
28                    *
```

Fig. 5 ADC2 program list

```
29                    ************************************************************
30                    *SUBR: ADC2  8-BIT A-D CONVERSION, BY SEQUENTIAL COMPARISON*
31                    ************************************************************
32     0C       008   ADC2  CPA           COMPARE PORT K & VREF
33     0D       048         RC            (CY)=0
34     0E       029         SZJ           SKIP IF (J(Y))=0
35     0F       049         SC            (CY)=1
36     10       008   ADC21 CPA           COMPARE PORT K & VREF
37     11       018         XAL           (A) EX (L)
38     12       029         SZJ           SKIP IF (J(Y))=0
39*W0*13        11A         BM    ADC23   ACTS AS INSTRUCTION B ON PAGE 14
40     14       02F         SZC           SKIP IF (CY)=0
41     15       044         RT            RETURN,CONVERSION FINISHED
42                    *
43     16       0A1         A     1       (A)=(A)+1,SKIP IF CARRY=0
44*W0*17        126         BM    ADC26   ACTS AS INSTRUCTION B ON PAGE 14
45     18       019   ADC22 TLA           (L)=(A)
46*W0*19        110         BM    ADC21   ACTS AS INSTRUCTION B ON PAGE 14
47     1A       02F   ADC23 SZC           SKIP IF (CY)=0
48*W0*1B        11D         BM    ADC24   ACTS AS INSTRUCTION B ON PAGE 14
49     1C       044         RT            RETURN,CONVERSION FINISHED
50     1D       0AF   ADC24 A     15      (A)=(A)+15,SKIP IF CARRY=0,(A)=(A)-1
51*W0*1E        118         BM    ADC22   ACTS AS INSTRUCTION B ON PAGE 14
52     1F       019         TLA           (L)=(A)
53     20       058         XAH           (A) EX (H)
54     21       0AF         A     15      (A)=(A)+15,SKIP IF CARRY=0, (A)=(A)-1
55*W0*22        124         BM    ADC25   ACTS AS INSTRUCTION B ON PAGE 14
56     23       0B0         LA    0       (A)=0
57     24       059   ADC25 THA           (H)=(A)
58*W0*25        110         BM    ADC21   ACTS AS INSTRUCTION B ON PAGE 14
59     26       019   ADC26 TLA           (L)=(A)
60     27       058         XAH           (A) EX (H)
61     28       0A1         A     1       (A)=(A)+1, SKIP IF CARRY=0
62     29       0BF         LA    15      (A)=15
63*W0*2A        124         BM    ADC25   ACTS AS INSTRUCTION B ON PAGE 14
64                    *                   END OF ADC2
65                    *
```

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

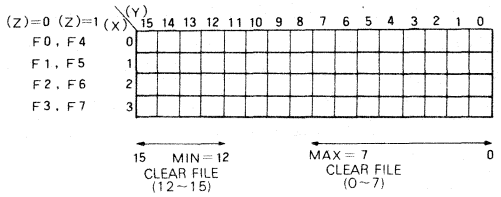


Fig. 7 How to call the clear-file subroutines

```
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
```

SUBROUTINES

4. Right-Shift File

Program Operation

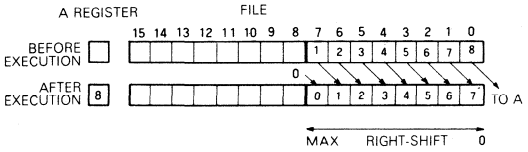
This is a subroutine that is used to right-shift the files F0~F7, as shown in Fig. 8. The contents of address 0 ~MAX (MAX = 0~15) in a file designated by the Y register are shifted right one digit. The most significant digit (MSD) is filled with 0 and the contents of the least significant digit (LSD) 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

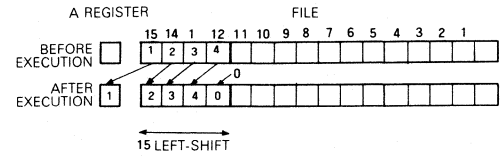
following the XAMI 0 instruction. When MIN = 0 is equated, it performs the same digits as the right-shift file subroutine. The instruction SEY, however, may be omitted when the skip condition is altered by the optional XAMI instruction.

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→ A→F1 . . . and so on.

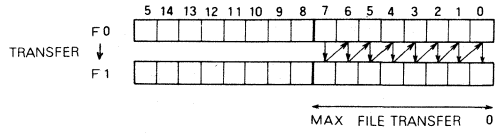
Data-transfer subroutines of the address MIN~15 can be made by changing MAX to MIN and the XAMD j instruction to XAMI j.

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
*W0*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
*W0*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
*W0*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
*W0*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
*W0*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
*W0*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
*W0*61       15F            BM    TF301  ACTS AS INSTRUCTION B ON PAGE 14
 62          044            RT           END OF TF30
                    *
```

14

SUBROUTINES

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

```
144                     ************************************************************
145                     *SUBR: EXF    EXCHANGE OF FILE FX1(0-MAX) EX FX2(0-MAX)     *
146                     ************************************************************
147     63      0C7     EXF01  LXY    0,MAX   F0(0-MAX) EX F1(0-MAX)
148     64      0E7     EXF23  LXY    2,MAX   F2(0-MAX) EX F3(0-MAX)
149     65      065     EXF001 TAM    1       (A)=(M(DP))
150     66      061            XAM    1       (A) EX (M(DP))
151     67      068            XAMD   0       (A) EX (M(DP)),(Y)=(Y)-1,SKIP IF (Y)=0
152*W0*68       165            BM     EXF001  ACTS AS INSTRUCTION B ON PAGE 14
153     69      044            RT             END OF EXF01
154                     *
155     6A      0C7     EXF02  LXY    0,MAX   F0(0-MAX) EX F2(0-MAX)
156     6B      0D7     EXF13  LXY    1,MAX   F1(0-MAX) EX F3(0-MAX)
157     6C      066     EXF002 TAM    2       (A)=(M(DP))
158     6D      062            XAM    2       (A) EX (M(DP))
159     6E      068            XAMD   0       (A) EX (M(DP)),(Y)=(Y)-1,SKIP IF (Y)=0
160*W0*6F       16C            BM     EXF002  ACTS AS INSTRUCTION B ON PAGE 14
161     70      044            RT             END OF EXF02
162                     *
163     71      0C7     EXF03  LXY    0,MAX   F0(0-MAX) EX F3(0-MAX)
164     72      0D7     EXF12  LXY    1,MAX   F1(0-MAX) EX F2(0-MAX)
165     73      067     EXF003 TAM    3       (A)=(M(DP))
166     74      063            XAM    3       (A) EX (M(DP))
167     75      068            XAMD   0       (A) EX (M(DP)),(Y)=(Y)-1,SKIP IF (Y)=0
168*W0*76       173            BM     EXF003  ASTS AS INSTRUCTION B ON PAGE 14
169     77      044            RT             END OF EXF03
170                     *
171                     ##########  ( MELPS 4 LIBRARY END )  ##########*
172                             END
 11                     ##########  ( MELPS 4 LIBRARY NO.2 )  ##########*
 12                     ************************************************************
 13                     *SUBR: INM    INCREMENT MEMORY FX(Y)=FX(Y)+1                *
 14                     *SUBR: DEM    DECREMENT MEMORY FX(Y)=FX(Y)-1                *
 15                     ************************************************************
 16     00      0C0     INM000 LXY    0,0     F0(0)  =F0(0)  +1 OR F4(0)  =F4(0)  +1
 17     01      0DB     INM111 LXY    1,11    F1(11) =F1(11) +1 OR F5(11) =F5(11) +1
 18     02      0ED     INM213 LXY    2,13    F2(13) =F2(13) +1 OR F6(13) =F6(13) +1
 19     03      0F7     INM3MA LXY    3,MAX   F3(MAX)=F3(MAX)+1 OR F7(MAX)=F7(MAX)+1
 20     04      0B1            LA     1       (A)=1
 21     05      00A     INM    AM             (A)=(A)+(M(DP))
 22     06      060            XAM    0       (A) EX (M(DP))
 23     07      0B0            LA     0       (A)=0
 24     08      044            RT
 25     09      0C0     DEM000 LXY    0,0     F0(0)  =F0(0)  -1 OR F4(0)  =F4(0)  -1
 26     0A      0DB     DEM111 LXY    1,11    F1(11) =F1(11) -1 OR F5(11) =F5(11) -1
 27     0B      0ED     DEM213 LXY    2,13    F2(13) =F2(13) -1 OR F6(13) =F6(13) -1
 28     0C      0F7     DEM3MA LXY    3,MAX   F3(MAX)=F3(MAX)-1 OR F7(MAX)=F7(MAX)-1
 29     0D      0BF            LA     15      (A)=15
 30*W0*0E        105            BM     INM     END OF INM AND DEM
 31                     *
 32                     ************************************************************
 33                     *SUBR: SNM    SKIP NON-ZERO MEMORY  FX(Y).NE.0 ?            *
 34                     ************************************************************
 35     0F      0C0     SNM000 LXY    0,0     F0(0)  .NE.0 ? OR F4(0)  .NE.0 ?
 36     10      0DB     SNM111 LXY    1,11    F1(11) .NE.0 ? OR F5(11) .NE.0 ?
 37     11      0ED     SNM213 LXY    2,13    F2(13) .NE.0 ? OR F6(13) .NE.0 ?
 38     12      0F7     SNM3MA LXY    3,MAX   F3(MAX).NE.0 ? OR F7(MAX).NE.0 ?
 39     13      064            TAM    0       (A)=(M(DP))
 40     14      0AF            A      15      (A)=(A)+15,SKIP IF CARRY=0
 41     15      045            RTS            RETURN IF FX(Y).NE.0
 42     16      044            RT             RETURN IF FX(Y).EQ.0
 43                     *                     END OF SNM
 44                     *
```

14

SUBROUTINES

```
MAX  EQU  7
      :
     LZ   0
     BM   SNF0MA
```

Instruction 1 ← Return if ($F0_0$~$F0_7$ = 0)

Instruction 2 ← Return if not ($F0_0$~$F0_7$ = 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)		
ADF 10	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)
ADF 01	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)
ADF 32	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)
ADF 23	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 ($F0_{15}$~$F0_{12}$) ← ($F0_{15}$~$F0_{12}$) + ($F1_{15}$~$F1_{12}$) 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

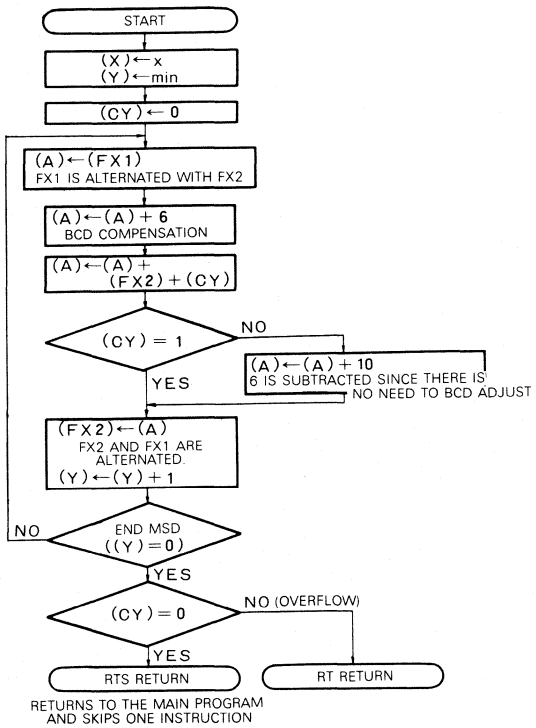
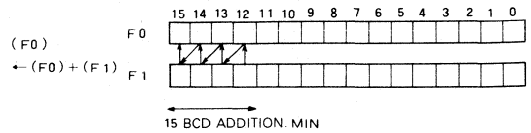


Fig. 15 BCD file addition (example of (F0) ← (F0)+(F1))



Note 3 : The arrows show how the file is changed.

MITSUBISHI ELECTRIC

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.

Fig. 16 SNFMA and SNFMI program lists

```
45                      ************************************************************
46                      *SUBR: SNFMA SKIP NON-ZERO FILE    FX(0-MAX).NE.0 ?       *
47                      ************************************************************
48     17       0C7     SNF0MA LXY    0,MAX   F0(0-MAX).NE.0 ? OR F4(0-MAX).NE.0 ?
49     18       0D7     SNF1MA LXY    1,MAX   F1(0-MAX).NE.0 ? OR F5(0-MAX).NE.0 ?
50     19       0E7     SNF2MA LXY    2,MAX   F2(0-MAX).NE.0 ? OR F6(0-MAX).NE.0 ?
51     1A       0F7     SNF3MA LXY    3,MAX   F3(0-MAX).NE.0 ? OR F7(0-MAX).NE.0 ?
52     1B       0B0     SNF4   LA     0       (A)=0
53     1C       026             SEAM           SKIP IF (A).EQ.(M(DP))
54     1D       045             RTS            RETURN IF FX(0-MAX).NE.0
55     1E       068             XAMD   0       (A)=(M(DP)),(Y)=(Y)-1,SKIP IF (Y)=0
56*W0*1F        11B             BM     SNF4    ACTS AS INSTRUCTION B ON PAGE 14
57     20       044             RT             RETURN IF FX(0-MAX).EQ.0
58                      *                      END OF SNFMA
59                      *
60                      ************************************************************
61                      *SUBR: SNFMI SKIP NON-ZERO FILE    FX(MIN-15).NE.0 ?      *
62                      ************************************************************
63     21       0CC     SNF0MI LXY    0,MIN   F0(MIN-15).NE.0 ? OR F4(MIN-15).NE.0 ?
64     22       0DC     SNF1MI LXY    1,MIN   F1(MIN-15).NE.0 ? OR F5(MIN-15).NE.0 ?
65     23       0EC     SNF2MI LXY    2,MIN   F2(MIN-15).NE.0 ? OR F6(MIN-15).NE.0 ?
66     24       0FC     SNF3MI LXY    3,MIN   F3(MIN-15).NE.0 ? OR F7(MIN-15).NE.0 ?
67     25       0B0     SNF5   LA     0       (A)=0
68     26       026             SEAM           SKIP IF (A).EQ.(M(DP))
69     27       045             RTS            RETURN IF FX(MIN-15).NE.0
70     28       06C             XAMI   0       (A)=(M(DP)),(Y)=(Y)+1,SKIP IF (Y)=15
71*W0*29        125             BM     SNF5    ACTS AS INSTRUCTION B ON PAGE 14
72     2A       044             RT             RETURN IF FX(MIN-15).EQ.0
73                      *                      END OF SNFMI
74                      *
```

SUBROUTINES

Fig. 17 ADF, SBF and SCF program lists

```
 75                     ********************************************************************
 76                     *SUBR: ADF    BCD ADDITION OF FILE FX1(MIN-15)=FX1(MIN-15)+FX2(MIN-15)
 77                     ********************************************************************
 78     23       0CC    ADF10  LXY   0,MIN   J=1: F1(MIN-15)=F1(MIN-15)+F0(MIN-15)
 79                     *                    J=2: F2(MIN-15)=F2(MIN-15)+F0(MIN-15)
 80                     *                    J=3: F3(MIN-15)=F3(MIN-15)+F0(MIN-15)
 81     2C       0DC    ADF01  LXY   1,MIN   J=1: F0(MIN-15)=F0(MIN-15)+F1(MIN-15)
 82                     *                    J=2: F3(MIN-15)=F3(MIN-15)+F1(MIN-15)
 83                     *                    J=3: F2(MIN-15)=F2(MIN-15)+F1(MIN-15)
 84     2D       0EC    ADF32  LXY   2,MIN   J=1: F3(MIN-15)=F3(MIN-15)+F2(MIN-15)
 85                     *                    J=2: F0(MIN-15)=F0(MIN-15)+F2(MIN-15)
 86                     *                    J=3: F1(MIN-15)=F1(MIN-15)+F2(MIN-15)
 87     2E       0FC    ADF23  LXY   3,MIN   J=1: F2(MIN-15)=F2(MIN-15)+F3(MIN-15)
 88                     *                    J=2: F1(MIN-15)=F1(MIN-15)+F3(MIN-15)
 89                     *                    J=3: F0(MIN-15)=F0(MIN-15)+F3(MIN-15)
 90     2F       048            RC           (CY)=0
 91     30       064    ADF011  TAM   J      (A)=(M(DP))
 92     31       0A6            A     6      (A)=(A)+6
 93     32       00F            AMCS         (A)=(A)+(M(DP))+(CY),(CY)=CARRY
 94     33       0AA            A     10     (A)=(A)+10,SKIP IF CARRY=0,BCD ADJUST
 95     34       000            NOP          (A)=(A)-6
 96     35       06C            XAMI  J      (A) EX (M(DP)),(Y)=(Y)+1,SKIP IF (Y)=15
97*W0*36        130            BM    ADF011 ACTS AS INSTRUCTION B ON PAGE 14
 98     37       02F            SZC          SKIP IF (CY)=0
 99     38       044            RT           RETURN IF OVERFLOW
100     39       045            RTS          END OF ADF01
101                     *
102                     ********************************************************************
103                     *SUBR: SBF    BCD SUBTRACTION  OF FILE                             *
104                     *                                 =FX1(MIN-15)-FX2(MIN-15)         *
105                     ********************************************************************
106     3A       0CC    SBF10  LXY   0,MIN   J=1: F1(MIN-15)=F1(MIN-15)-F0(MIN-15)
107                     *                    J=2: F2(MIN-15)=F2(MIN-15)-F0(MIN-15)
108                     *                    J=3: F3(MIN-15)=F3(MIN-15)-F0(MIN-15)
109     3B       0DC    SBF01  LXY   1,MIN   J=1: F0(MIN-15)=F0(MIN-15)-F1(MIN-15)
110                     *                    J=2: F3(MIN-15)=F3(MIN-15)-F1(MIN-15)
111                     *                    J=3: F2(MIN-15)=F2(MIN-15)-F1(MIN-15)
112     3C       0EC    SBF32  LXY   2,MIN   J=1: F3(MIN-15)=F3(MIN-15)-F2(MIN-15)
113                     *                    J=2: F0(MIN-15)=F0(MIN-15)-F2(MIN-15)
114                     *                    J=3: F1(MIN-15)=F1(MIN-15)-F2(MIN-15)
115     3D       0FC    SBF23  LXY   3,MIN   J=1: F2(MIN-15)=F2(MIN-15)-F3(MIN-15)
116                     *                    J=2: F1(MIN-15)=F1(MIN-15)-F3(MIN-15)
117                     *                    J=3: F0(MIN-15)=F0(MIN-15)-F3(MIN-15)
118     3E       049            SC           (CY)=1
119     3F       064    SBF011  TAM   J      (A)=(M(DP))
120     40       08F            CMA          COMPLEMENT (A)
121     41       00F            AMCS         (A)=(A)+(M(DP))+(CY),(CY)=CARRY
122     42       0AA            A     10     (A)=(A)+10,SKIP IF CARRY=0,BCD ADJUST
123     43       06C            XAMI  J      (A) EX (M(DP)),(Y)=(Y)+1,SKIP IF (Y)=15
124*W0*44        13F            BM    SBF011 ACTS AS INSTRUCTION B ON PAGE 14
125     45       02F            SZC          SKIP IF (CY)=0
126     46       045            RTS          END OF SBF01
127     47       044            RT           RETURN IF OVERFLOW
128                     *
129                     ************************************************************
130                     *SUBR: SCF    SIGN CHANGE OF FILE FX(SIGN) EX                *
131                     ************************************************************
132     48       0CC    SCF0   LXY   0,SIGN  F0(SIGN) EX
133     49       0DC    SCF1   LXY   1,SIGN  F1(SIGN) EX
134     4A       0EC    SCF2   LXY   2,SIGN  F2(SIGN) EX
135     4B       0FC    SCF3   LXY   3,SIGN  F3(SIGN) EX
136     4C       0B8            LA    8      (A)=8
137     4D       00A            AM           (A)=(A)+(M(DP))
138     4E       060            XAM   0      (A) EX (M(DP))
139     4F       044            RT           END OF SCF0
140                     *
141                     ##########  ( MELPS 4 LIBRARY END )  ##########*
142                     END
```


MITSUBISHI ELECTRIC

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

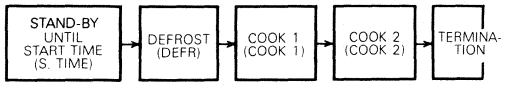
- Programmed operation for DEFROST, COOK 1 and COOK 2 processes
- Time, temperature and power controls
- Clock and timer
- Display of the time, temperature and power on the large fluorescent display tube
- The simplification in circuit design facilitates cost reduction and miniaturization of the oven.

FUNCTIONS

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 displayed 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~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.

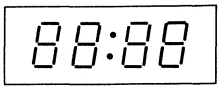
3. Outputs

(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

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.

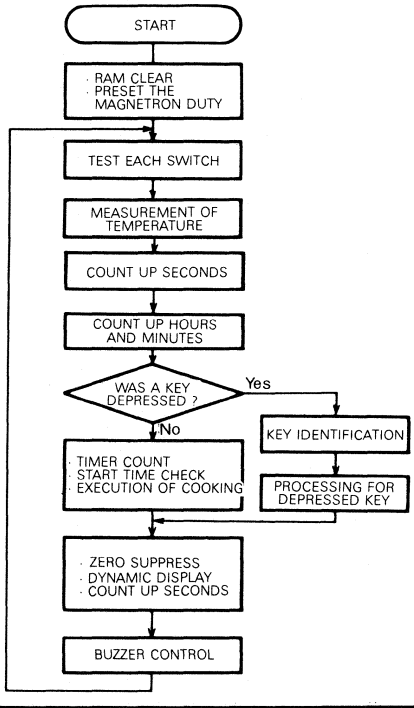
Measured temperature: Depression of the [TEMP] key twice displays of the measured temperature at the present stage.

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.



APPLICATION OF MELPS 4 SINGLE-CHIP 4-BIT MICROCOMPUTER

(M58840-XXXP) IN A MICROWAVE OVEN

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.

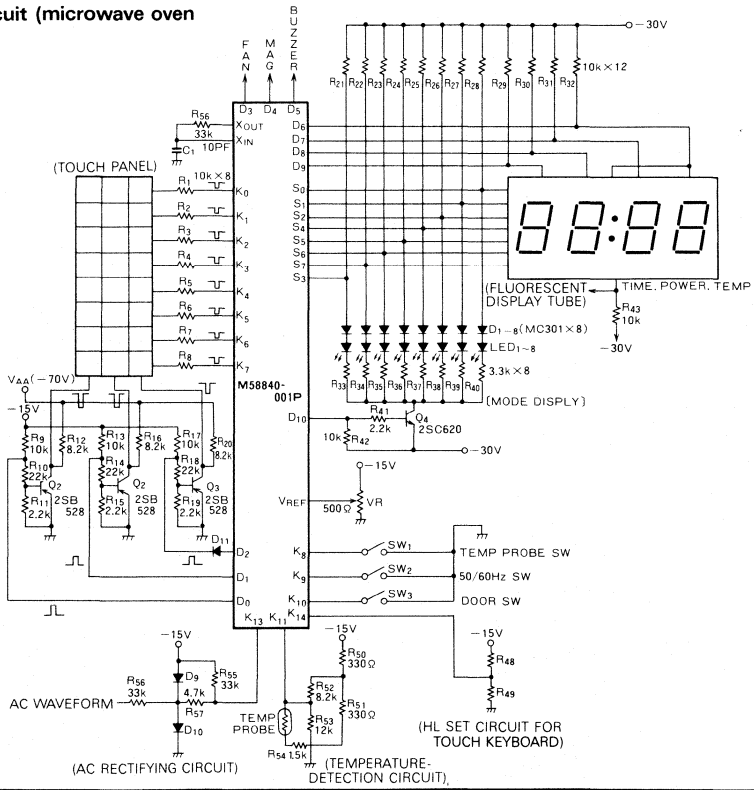
9. Typical control circuit of a microwave oven

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.

Example of application circuit (microwave oven M58840-001P)



MITSUBISHI ELECTRIC

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

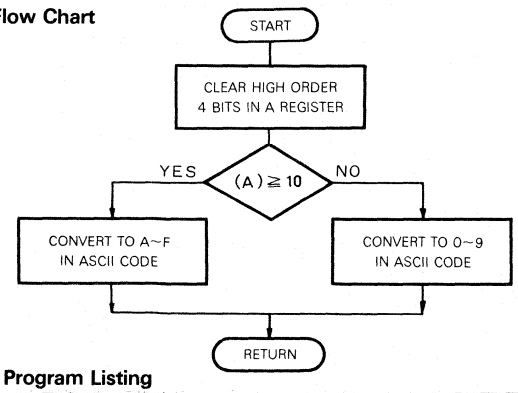
1.1 Binary (4 Bits) to ASCII (1 Character) Conversion (BTA)

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     0FH
        CPI     10
        JNC     B1
        ADI     48
        RET
B1      ADI     55
        RET
```

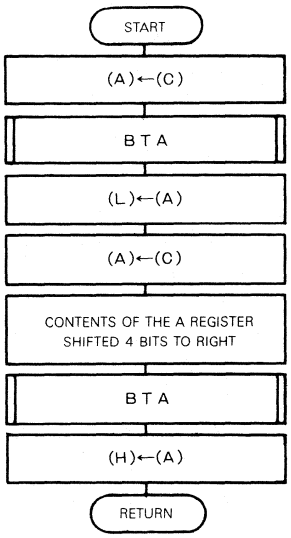
1.2 Binary (8 Bits) to ASCII (2 Characters) Conversion (BTA 2)

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
```

14

SUBROUTINES

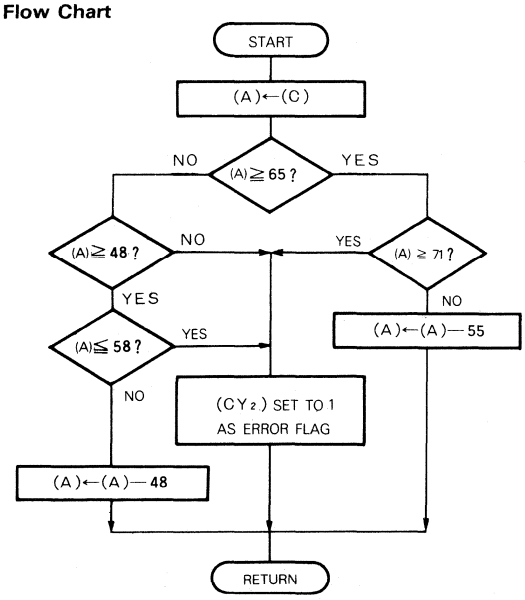
1.3 ASCII (1 Character) to Binary (4 Bits) Conversion (ATB)

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

```
1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33
*
*  ***  SUB(ATB)     */ASCII TO BINARY
*
ATB       MOV       A,C
          CPI       65
          JC        A1
          CPI       71
          JNC       A3
          SUI       55
          RET
A1        CPI       48
          JC        A3
          CPI       58
          JNC       A3
A2        SUI       48
          RET
A3        STC
          RET
```

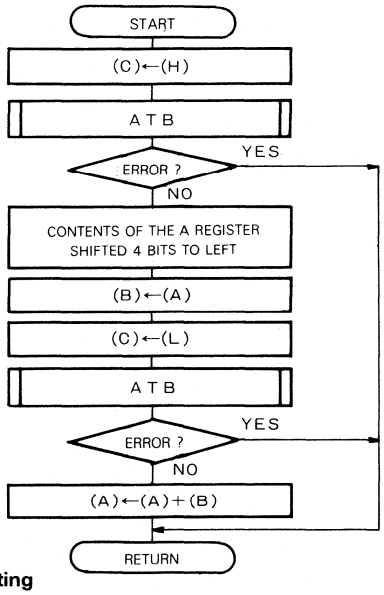
1.4 ASCII (2 Characters) to Binary (8 Bits) Conversion (ATB 2)

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 ($0 \sim 255_{10}$). 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

```
1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33
*
*  ***  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 ($1 \leqq k \leqq 65\ 535$) 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.

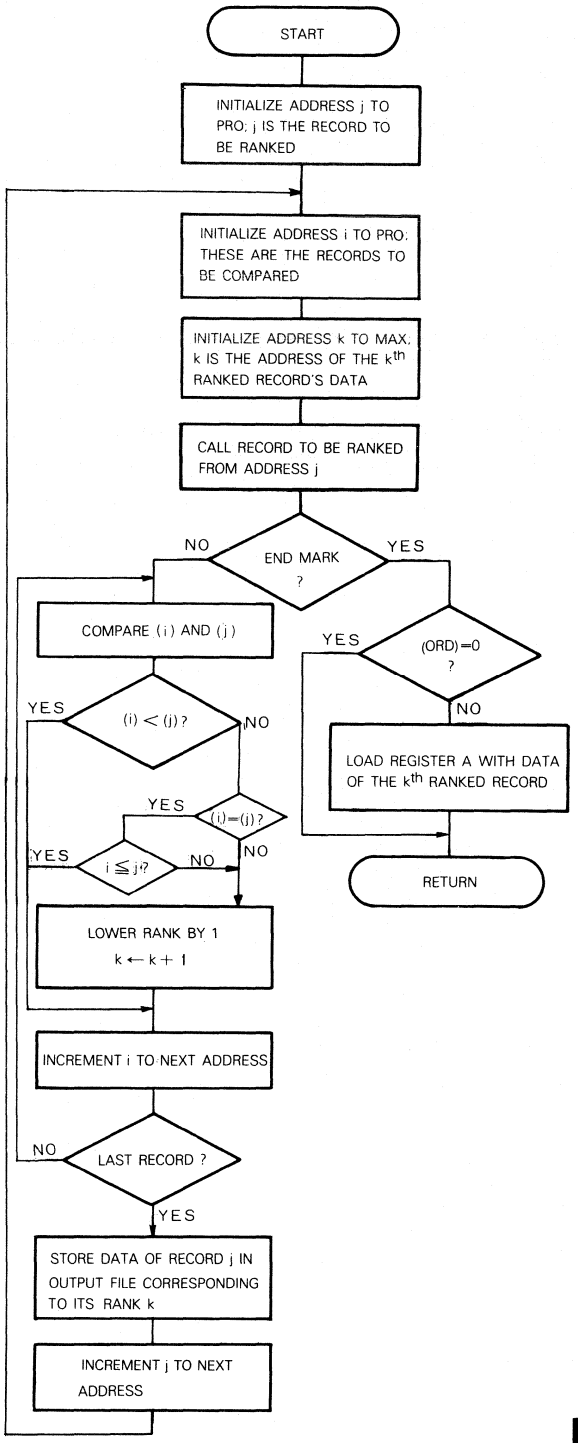
Register Status

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



SUBROUTINES

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
     JNX    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.

SUBROUTINE 1: INTEGER ARITHMETIC OPERATIONS IN 8K-BIT MASK-PROGRAMMED ROM (M58730-001S)

DESCRIPTION

The MELPS 8/85 Subroutine 1 'Integer Arithmetic Operation' is programmed on a standard M58730-001S mask ROM. It includes 18 subroutines for a MELPS 8/85 CPU. Although the basic unit of a MELPS CPU is 1 byte (8 bits), units of 2 bytes (16 bits) and 4 bytes (32 bits) can be easily processed using these subroutines.

These subroutines contain sections of common coding; therefore, when using the subroutines, the CPU must be running in the interrupt disable mode.

These subroutines can be divided into the following general classifications:

- Addition routines
- Subtraction routines
- Multiplication routines
- Division routines
- Shift-operation routines
- Logic-operation routines

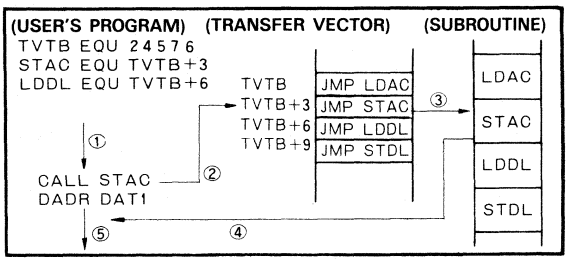
FEATURES

- All programs implemented using a pseudo accumulator in a RAM region
- Easy processing of 2-byte or 4-byte data
- Jump to subroutines via transfer vectors

1. SUBROUTINE CALL

In a user's program, the subroutine call is as follows:

Fig. 1.1 Subroutine call



Note 1 : The processing order is ①, ②, ③, ④ and ⑤. A transfer vector is used to set the entry address of each subroutine.

2 : Transfer vectors are used for subroutine calls because they are not affected by changes in program size.

3 : The absolute address of a subroutine or its transfer vector must be defined before it is called.

4 : The absolute address of a subroutine or its transfer vector refers to the table of subroutine functions.

2. RESERVED MEMORY LOCATIONS

Memory locations 6000_{16}~$63FF_{16}$ of the ROM region are reserved. In addition, a 50-byte RAM region, locations $3FCE_{16}$~$3FFF_{16}$, is reserved for executing the ROM subroutines.

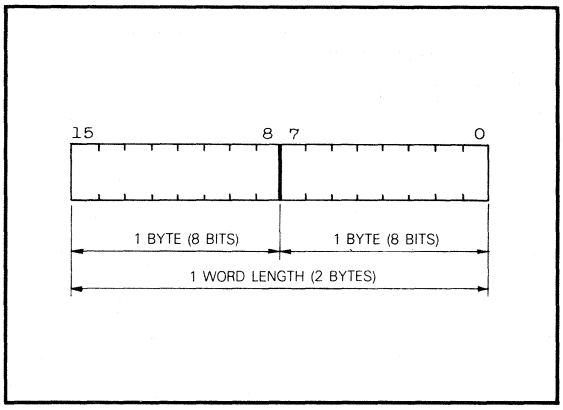
3. DATA PROCESSING UNITS OF SUBROUTINES

The MELPS 8 CPU processes data units of 8 bits (occasionally 16 bits) while these subroutines process data units of 2 bytes (16 bits) or 4 bytes (32 bits).

3.1 One Word Length (2 bytes)

A data unit of 2 bytes (16 bits) can represent three binary coded decimal digits, 16 logical elements, a binary number with a range of -2^{15}~$2^{15}-1$, or two characters. This data structure is shown in Fig. 3.1.

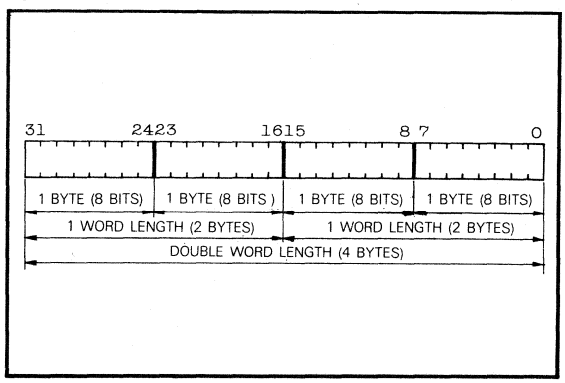
Fig. 3.1 Data structure of one word length (2 bytes)



3.2 Double Word Length (4 bytes)

A data unit of 4 bytes (32 bits) can represent seven binary coded decimal digits, a binary number with a range of -2^{31}~$2^{31}-1$, or four characters. The data structure is shown in Fig. 3.2.

Fig. 3.2 Data structure of double word length (4 bytes)



SUBROUTINE 1: INTEGER ARITHMETIC OPERATIONS IN 8K-BIT MASK-PROGRAMMED ROM (M58730-001S)

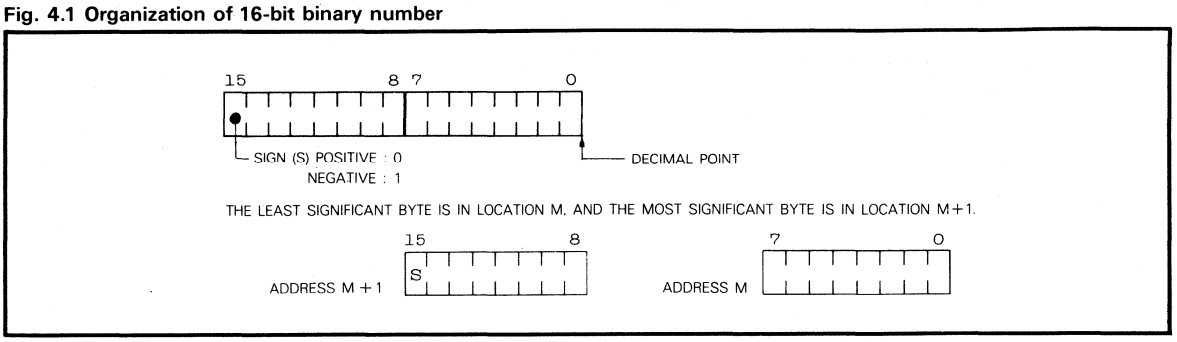
4. NUMERICAL EXPRESSIONS

Numbers can be organized in 16-bit or 32-bit units as shown below.

4.1 16-Bit Binary Number

This binary number of 16 bits is organized as one unit. Negative numbers are in 2's complement form. The number has a range of -2^{15}~$2^{15}-1$ (−32768~32767).

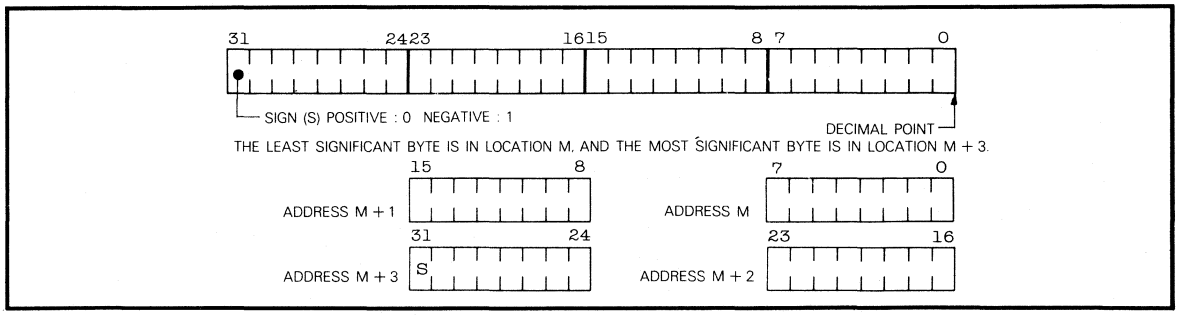
Fig. 4.1 Organization of 16-bit binary number



4.2 32-Bit Binary Number

This binary number of 32 bits is organized as one unit. Negative numbers are in 2's complement form. The number has a range of -2^{31}~$2^{31}-1$ (−2147483648~2147483647).

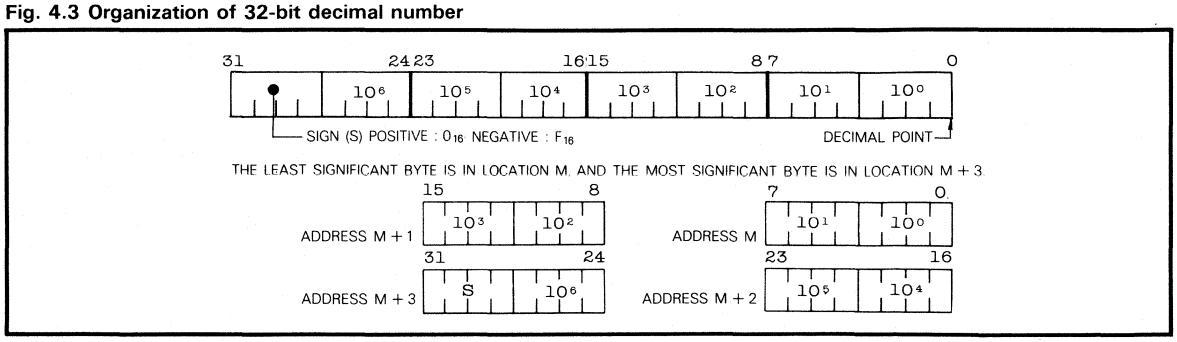
Fig. 4.2 Organization of 32-bit binary number



4.3 32-Bit Decimal Number

This decimal number of 32 bits consists of a 7-decimal digit numerical part and a 1-digit sign part. The number has a range of -10^7+1~10^7-1 (−9999999~9999999).

Fig. 4.3 Organization of 32-bit decimal number



MITSUBISHI ELECTRIC

5. SUBROUTINE FUNCTIONS

Subroutine name	Function and error condition	Number of steps	Absolute address in hexadecimal (in decimal)	Transfer vector symbolic address	Processing time (max) in ms
LDAC	Load one word (2 bytes) data into the pseudo accumulator (Note 5)	19	60B7 (24759)	TVTB (Note 2)	0.2
STAC	Store one-word (2 bytes) data of the pseudo accumulator in the location specified by the operand address.	14	60CA (24778)	TVTB+3	0.2
LDDL	Load double-length (4 bytes) data into the pseudo accumulator.	20	60D8 (24792)	TVTB+6	0.3
STDL	Store double-length (4 bytes) data of the pseudo accoumulator in the location specified by the operand address.	20	60EC (24812)	TVTB+9	0.3
SLDL	Shift left double-length (4 bytes) data in the pseudo accumulator n bits. When n does not statisfy the inequality $1 \leqq n \leqq 32$, it is considered an error conditon. Then the A register is set to 1, and the pseudo accumulator is not shifted.	39	6100 (24832)	TVTB+21	0.3
SRDL	Shift right double-length (4 bytes) data in the pseudo accumulator n bits. When n does not statisfy the inequality $1 \leqq n \leqq 32$, it is considered an error condition. Then register A is set to 1, and the pseudo accumulator is not shifted.	39	6127 (24871)	TVTB+24	0.3
ARDL	Arithmetic shift right double-length (4 bytes) data in the pseudo accumulator n bits. When n does not satisfy the inequality $1 \leqq n \leqq 31$, it is considered an error condition. Then the A register is set to 1, and the pseudo accumulator is not shifted.	64	614E (24910)	TVTB+27	0.3
XRAC	Exclusively OR the pseudo accumulator (2 bytes) data and the operand. The result is retained in the pseudo accumulator.	18	618E (24974)	TVTB+18	0.2
NDAC	Add the pseudo accumulator (2 bytes) data and the operand. The result is retained in the pseudo accumulator.	18	61A0 (24992)	TVTB+12	0.2
ORAC	Inclusive OR the pseudo accumulator (2 bytes) data and the operand. The result is retained in the pseudo accumulator.	18	61B2 (25010)	TVTB+15	0.2
ADAC	Add the contents of the pseudo accumulator (2 bytes) and the operand. The sum is retained in the pseudo accumulator. If a carry is generated by the additon, the A register is set to 1 (overflow); otherwise, it is reset to 0.	12+(20) (Note 8)	61C4 (25028)	TVTB+30	0.3
ADDL	Add the contents of the double-length pseudo accumulator (4 bytes) and the operand. The sum is retained in the pseudo accumulator. If a carry is generated by the addition, the A register is set to 1 (overflow); otherwise, it is reset to 0.	12+(22) (Note 8)	61D0 (25040)	TVTB+36	0.3
SBAC	Subtract the operand from the contents of the pseudo accumulator (2 bytes). The difference is retained in the pseudo accumulator. If a borrow is generated by the subtraction, the A register is set to 1 (overflow), otherwise, it is reset to 0.	12+(20) (Note 8)	61F0 (25072)	TVTB+33	0.3
SBDL	Subtract the operand from the double-length pseudo accumulator (4 bytes). The difference is retained in the pseudo accumulator. If a borrow is generated by the subtraction, the A register is set to 1 (overflow); otherwise, it is reset to 0.	12+(22) (Note 8)	61FC (25084)	TVTB+39	0.3
MLAC	Multiply the contents of the pseudo accumulator (2 bytes) by the operand. The product is retained in the pseudo accumulator.	67	621E (25118)	TVTB+42	12.0
DVAC	Divide the contents of the pseudo accumulator (4 bytes) by the 2-byte operand. The quotient is retained in the high-order 2 bytes, and the remainder in the low-order 2 bytes of the pseudo accumulator. If the 2-byte operand (divisor) is greater than or equal to the high-order 2 bytes of the dividend or is 0, it is considered an error condition. Then the A register is set to 1, and the contents of the pseudo accumulator are unaltered.	195	6261 (25185)	TVTB+45	15.0
DCAD	Decimally add the contents of the pseudo accumulator (4 bytes) and the operand. The sum is retained in the pseudo accumulator. If a carry is generated by the addition (overflow), it is considered an error condition; and the A register is set to 1.	12+(155) (Note 3)	6324 (25380)	TVTB+48	0.7
DCSB	Decimally subtract the operand from the contents of the pseudo accumulator (4 bytes). The difference is retained in the pseudo accumulator. If a carry is generated by the subtraction (overflow), it is considered an error condition, and the A register is set to 1.	12+(155) (Note 3)	6330 (25392)	TVTB+51	1.3

Note 5 : The pseudo accumulator is a double-length (4-byte) register reserved in the RAM.
6 : The starting address of the transfer vector table (TVTB) is 24759.
7 : The number in ( ) is the number of steps in common routines.
Note 8 : The subroutines occupy 800 bytes of memory. The transfer vector table occupies 54 bytes of memory. The save registers B, C, D, E, H and L and return routines occupy 129 bytes of memory. Total memory requirement is 983 bytes.

SUBROUTINE 1: INTEGER ARITHMETIC OPERATIONS IN 8K-BIT MASK-PROGRAMMED ROM (M58730-001S)

6. BASIC CALLING SEQUENCE

Label	Instruction	Operand	
	CALL	SUB	Describes the subroutine named 'SUB' being called.
	DADR	ABC	Symbolic address of the operand (ABC) used by the called subroutine.
ABC	BSS		Defines the operand (ABC) and reserves memory for it.

In this example using this subroutine, the program adds two 4-byte binary numbers and stores the sum in locations WORK~WORK+3.

Label	Instruction	Operand	
	ORG	3700#	Absolute address of the program's start.
TVTB	EQU	2475 9	Absolute address of the transfer vector table's start.
LDDL	EQU	TVTB+6	Absolute address of subroutine LDDL's transfer vector.
STDL	EQU	TVTB+9	Absolute address of subroutine STDL's transfer vector.
ADDL	EQU	TVTB+36	Absolute address of subroutine ADDL's transfer vector.
DATA1	DEF	9C#	Operand
	DEF	2A#	
	DEF	45#	
	DEF	03#	
ABC	DEF	09#	Operand
	DEF	23#	
	DEF	19#	
	DEF	0A#	
WORK	BSS	4	Reserves a 4-byte region to store the sum.
	CALL	LDDL	Loads operand (DATA 1) into the pseudo accumulator.
	DADR	DATA1	
	CALL	ADDL	Adds operand (ABC) to the contents of the pseudo accumulator and retains the result in the pseudo accumulator.
	DADR	ABC	
	CALL	STDL	Stores the sum in locations WORK ~ WORK + 3.
	DADR	WORK	
	END		

Note : '#' flags a hexadecimal number.

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).

FUNCTIONS

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.

OPERATIONS

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	System is not operating, because no line is connected between the master and the slave computer No. 1.
	1	0	0	Data is being transmitted between the master and slave computer No. 1.
	0	0	0	Data transmission to computer No. 1 has been completed. System is in the 5-second delay routine.
	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 transmitted 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 Example of application circuit

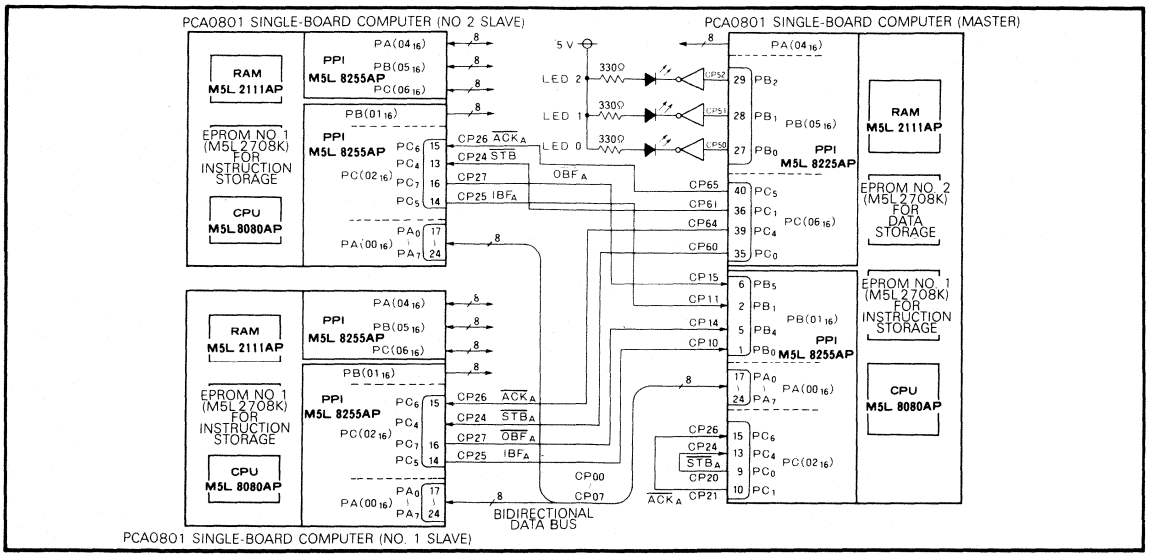


Fig. 2 Master microcomputer flow chart

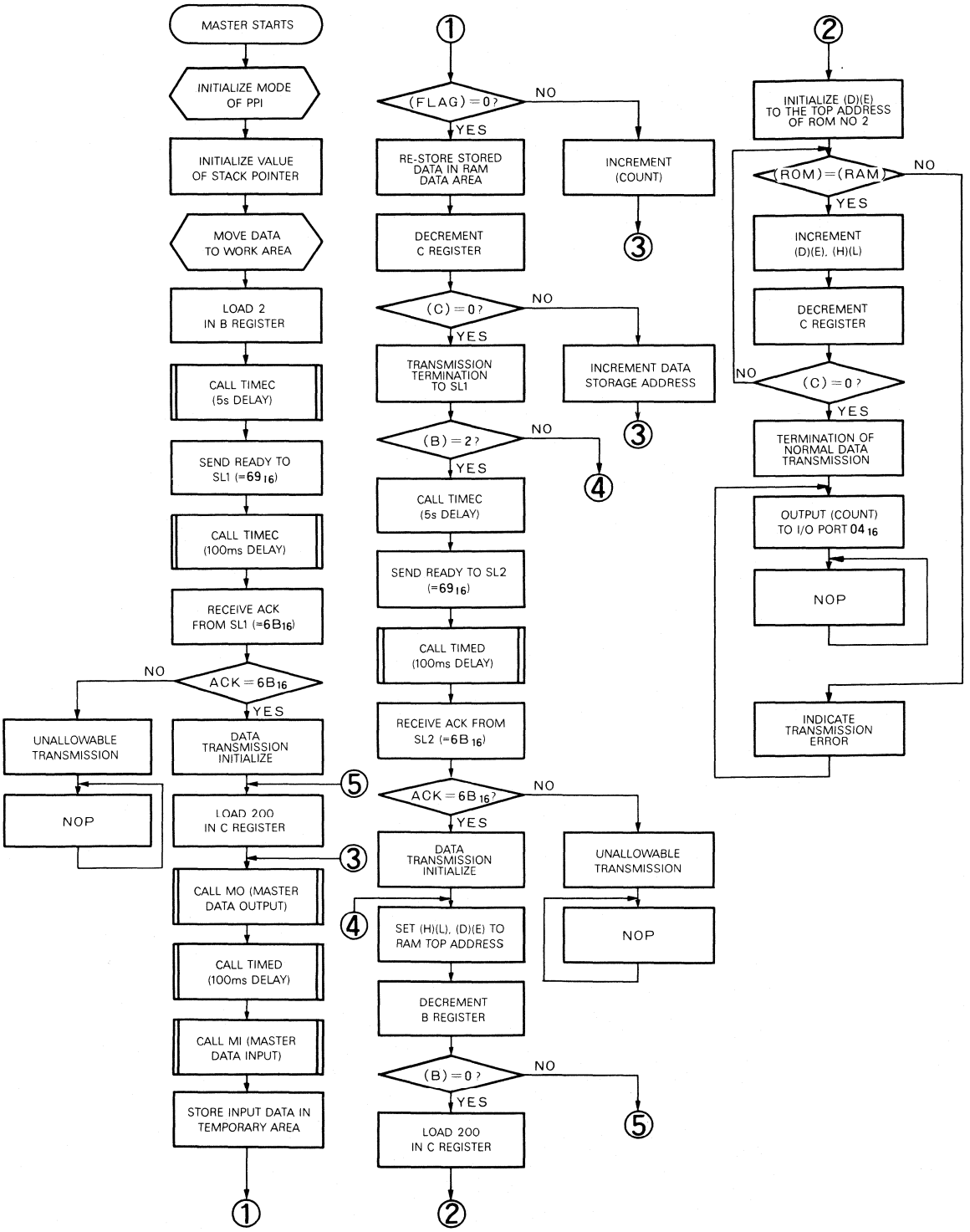
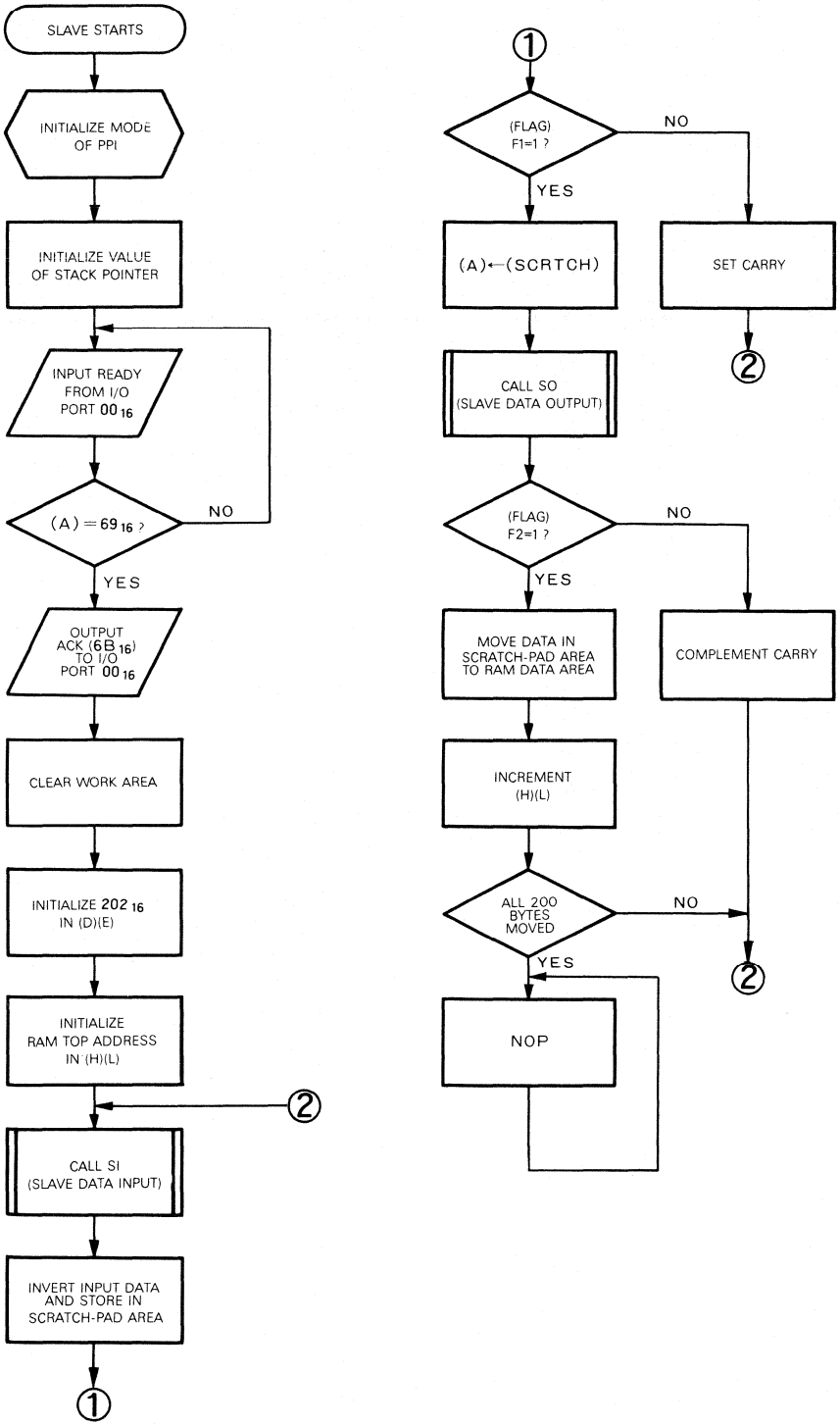


Fig. 3 Slave microcomputer flow chart



APPLICATION OF MELCS 8/2 SINGLE-BOARD COMPUTER

(PCA 0801) IN DATA TRANSMISSION THROUGH A MASTER-SLAVE MULTICOMPUTER SYSTEM

MASTER MICROCOMPUTER MAIN PROGRAM LIST

```
**CROSS ASSEMBLER OF 8-BIT MICROPROCESSOR

0001*** MASTER MICROCOMPUTER MAIN PROGRAM ***
0002 0400          ROM2ST EQU   0400#
0003 4000          RAMST  EQU   4000#
0004 40C8          DSTNT1 EQU   40C8#
0005 40CA          DSTNT2 EQU   40CA#
0006 40CC          FLAG   EQU   40CC#
0007 40CD          COUNT  EQU   40CD#
0008 40CE          TEMPRY EQU   40CE#
0009*
0010*
0011 0000                 ORG   0000#
0012 0000 3EC2    MASTER  MVI   A,C2#
0013 0002 D303            OUT   03#
0014 0004 3E01            MVI   A,01#
0015 0006 D303            OUT   03#
0016 0008 3E03            MVI   A,03#
0017 000A D303            OUT   03#
0018 000C 3E80            MVI   A,80#
0019 000E D307            OUT   07#
0020 0010 3E00            MVI   A,00#
0021 0012 D305            OUT   05#
0022 0014 3EFF            MVI   A,FF#
0023 0016 D306            OUT   06#
0024 0018 3EAA            MVI   A,AA#
0025 001A D304            OUT   04#
0026 001C 31FF40          LXI   SP,40FF#
0027 001F 0602            MVI   B,2
0028 0021 21DD01          LXI   H,X
0029 0024 22C840          SHLD  DSTNT1
0030 0027 21E501          LXI   H,Y
0031 002A 22CA40          SHLD  DSTNT2
0032 002D AF              XRA   A
0033 002E 32CE40          STA   TEMPRY
0034 0031 32CD40          STA   COUNT
0035 0034 3E59            MVI   A,59#
0036 0036 32CC40          STA   FLAG
0037 0039 CD1201          CALL  TIMEC
0038 003C 3E69            MVI   A,69#
0039 003E D300            OUT   00#
0040 0040 3E02            MVI   A,02#
0041 0042 D303            OUT   03#
0042 0044 3E00            MVI   A,00#
0043 0046 D307            OUT   07#
0044 0048 3C              INR   A
0045 0049 D307            OUT   07#
0046 004B 3E03            MVI   A,03#
0047 004D D303            OUT   03#
0048 004F CD2E01          CALL  TIMED
0049 0052 3E08            MVI   A,08#
0050 0054 D307            OUT   07#
0051 0056 3E00            MVI   A,00#
0052 0058 D303            OUT   03#
0053 005A 3C              INR   A
0054 005B D303            OUT   03#
0055 005D 3E09            MVI   A,09#
0056 005F D307            OUT   07#
0057 0061 DB00            IN    00#
0058 0063 D66B            SUI   6B#
0059 0065 C20B01          JNZ   NOCOMC
0060 0068 3E01            MVI   A,01#
0061 006A D305            OUT   05#
0062 006C 210004          LXI   H,ROM2ST
0063 006F 110040          LXI   D,RAMST
0064 0072 0EC8    RPT2    MVI   C,200
0065 0074 7E      RPT1    MOV   A,M
0066 0075 CD5501          CALL  MC
0067 0078 CD2E01          CALL  TIMED
0068 007B CD6001          CALL  MI
0069 007E 32CE40          STA   TEMPRY
0070 0081 3ACC40          LDA   FLAG
0071 0084 B7              ORA   A
0072 0085 C20101          JNZ   SUM
0073 0088 3ACE40          LDA   TEMPRY
0074 008B EB              XCHG
0075 008C 77              MOV   M,A
0076 008D 0D              DCR   C
0077 008E EB              XCHG
0078 008F C2FC00          JNZ   YET
0079 0092 3E00            MVI   A,00#
0080 0094 D305            OUT   05#
0081 0096 78              MOV   A,B
0082 0097 FE02            CPI   02#
0083 0099 C2CF00          JNZ   S2END
0084 009C CD1201          CALL  TIMEC
0085 009F 3E69            MVI   A,69#
0086 00A1 D300            OUT   00#
0087 00A3 3E02            MVI   A,02#
0088 00A5 D303            OUT   03#
0089 00A7 3E02            MVI   A,02#
0090 00A9 D307            OUT   07#
0091 00AB 3C              INR   A
0092 00AC D307            OUT   07#
0093 00AE 3E03            MVI   A,03#
0094 00B0 D303            OUT   03#
0095 00B2 CD2E01          CALL  TIMED
0096 00B5 3E0A            MVI   A,0A#
0097 00B7 D307            OUT   07#
0098 00B9 3E00            MVI   A,00#
0099 00BB D303            OUT   03#
0100 00BD 3C              INR   A
0101 00BE D303            OUT   03#
0102 00C0 3E0B            MVI   A,0B#
0103 00C2 D307            OUT   07#
0104 00C4 DB00            IN    00#
0105 00C6 D66B            SUI   6B#
0106 00C8 C20B01          JNZ   NOCOMC
0107 00CB 3E07            MVI   A,07#
0108 00CD D305            OUT   05#
0109 00CF 210040  S2END   LXI   H,RAMST
0110 00D2 54              MOV   D,H
0111 00D3 5D              MOV   E,L
0112 00D4 05              DCR   B
0113 00D5 C27200          JNZ   RPT2
0114 00D8 0EC8            MVI   C,200
0115 00DA 110004          LXI   D,ROM2ST
0116 00DD 1A      SCAN    LDAX  D
0117 00DE BE              CMP   M
0118 00DF C2F500          JNZ   TRMERR
0119 00E2 13              INX   D
0120 00E3 23              INX   H
0121 00E4 0D              DCR   C
0122 00E5 C2DD00          JNZ   SCAN
0123 00E8 3E02            MVI   A,02#
0124 00EA D305            OUT   05#
0125 00EC 3ACD40  NO2     LDA   COUNT
0126 00EF D304            OUT   04#
0127 00F1 00      NO1     NOP
0128 00F2 C3F100          JMP   NO1
0129*
0130 00F5 3E04    TRMERR  MVI   A,04#
0131 00F7 D305            OUT   05#
0132 00F9 C3EC00          JMP   NO2
0133*
0134 00FC 23      YET     INX   H
0135 00FD 13              INX   D
0136 00FE C37400          JMP   RPT1
0137*
0138*** NO-PASS SUM ***
0139*
```

```
0140 0101 3ACD40 SUM     LDA   COUNT
0141 0104 3C             INR   A
0142 0105 32CD40         STA   COUNT
0143 0108 C37400         JMP   RPT1
0144*
0145*
0146*** NOCOMMUNICATE ***
0147*
0148 010B 3E04   NOCOMC  MVI   A,04#
0149 010D D305           OUT   05#
0150 010F C3F100         JMP   NO1
0151*
0152*** SUBROUTINE TIMEC ***
0153*
0154 0112 1E32   TIMEC   MVI   E,50
0155 0114 CD2E01 TIMEC1  CALL  TIMED
0156 0117 1D             DCR   E
0157 0118 CA1E01         JZ    TIMEC2
0158 011B C31401         JMP   TIMEC1
0159 011E C9     TIMEC2  RET
0160*
0161*
0162*
0163*** SUBROUTINE TIMEF ***
0164 011F D5     TIMEF   PUSH  D
0165 0120 1E0A           MVI   E,10
0166 0122 CD2E01 TIMEF1  CALL  TIMED
0167 0125 1D             DCR   E
0168 0126 CA2C01         JZ    TIMEF2
0169 0129 C32201         JMP   TIMEF1
0170 012C D1     TIMEF2  POP   D
0171 012D C9             RET
0172*
0173*** SUBROUTINE TIMED ***
0174*
0175 012E F5     TIMED   PUSH  PSW
0176 012F C5             PUSH  B
0177 0130 D5             PUSH  D
0178 0131 E5             PUSH  H
0179 0132 1614           MVI   D,20
0180 0134 0E14           MVI   C,20
0181 0136 06C8   TIMED6  MVI   B,200
0182 0138 3EC8           MVI   A,200
0183 013A C33D01 TIMED1  JMP   TIMED2
0184 013D C34001 TIMED2  JMP   TIMED3
0185 0140 05     TIMED3  DCR   B
0186 0141 3D             DCR   A
0187 0142 CA4801         JZ    TIMED4
0188 0145 C33A01         JMP   TIMED1
0189 0148 15     TIMED4  DCR   D
0190 0149 0D             DCR   C
0191 014A CA5001         JZ    TIMED7
0192 014D C33601         JMP   TIMED6
0193 0150 E1     TIMED7  POP   H
0194 0151 D1             POP   D
0195 0152 C1             POP   B
0196 0153 F1             POP   PSW
0197 0154 C9             RET
0198*
0199*
0200*** SUBROUTINE MO ***
0201*
0202 0155 D300   MO      OUT   00#
0203 0157 D5             PUSH  D
0204 0158 110202         LXI   D,202#
0205 015B CD6B01         CALL  DECODO
0206 015E D1             POP   D
0207 015F C9             RET
0208*
0209*
0210*** SUBROUTINE MI ***
0211*
0212 0160 D5     MI      PUSH  D
0213 0161 110202         LXI   D,202#
0214 0164 CDA201         CALL  DECODI
0215 0167 DB00           IN    00#
0216 0169 D1             POP   D
0217 016A C9             RET
0218*
0219*
0220*** SUBROUTINE DECODO ***
0221*
0222 016B E5     DECODO  PUSH  H
0223 016C D5             PUSH  D
0224 016D 2AC840         LHLD  DSTNT1
0225 0170 DB01   MIBF    IN    01#
0226 0172 A6             ANA   M
0227 0173 C29B01         JNZ   MIBFP
0228 0176 3E02           MVI   A,02#
0229 0178 D303           OUT   03#
0230 017A 23             INX   H
0231 017B 7E             MOV   A,M
0232 017C D307           OUT   07#
0233 017E 3C             INR   A
0234 017F D307           OUT   07#
0235 0181 3E03           MVI   A,03#
0236 0183 D303           OUT   03#
0237 0185 79             MOV   A,C
0238 0186 D601           SUI   01#
0239 0188 CA9701         JZ    FINE1
0240 018B 2B             DCX   H
0241 018C 22C840 STORE1  SHLD  DSTNT1
0242 018F 3E24           MVI   A,24#.
0243 0191 32CC40         STA   FLAG
0244 0194 D1     NOIBF   POP   D
0245 0195 E1             POP   H
0246 0196 C9             RET
0247 0197 23     FINE1   INX   H
0248 0198 C38C01         JMP   STORE1
0249 019B 15     MIBFP   DCR   D
0250 019C C27001         JNZ   MIBF
0251 019F C39401         JMP   NOIBF
0252*
0253*
0254*** SUBROUTINE DECODI ***
0255*
0256 01A2 E5     DECODI  PUSH  H
0257 01A3 D5             PUSH  D
0258 01A4 2ACA40         LHLD  DSTNT2
0259 01A7 DB01   MOBF    IN    01#
0260 01A9 A6             ANA   M
0261 01AA C2D601         JNZ   MOBFP
0262 01AD 23             INX   H
0263 01AE 7E             MOV   A,M
0264 01AF D307           OUT   07#
0265 01B1 3E00           MVI   A,00#
0266 01B3 D303           OUT   03#
0267 01B5 3E01           MVI   A,01#
0268 01B7 D303           OUT   03#
0269 01B9 7E             MOV   A,M
0270 01BA 3C             INR   A
0271 01BB D307           OUT   07#
0272 01BD 79             MOV   A,C
0273 01BE D601           SUI   01#
0274 01C0 CAD201         JZ    FINE2
0275 01C3 2B             DCX   H
0276 01C4 22CA40 STORE2  SHLD  DSTNT2
0277 01C7 3ACC40         LDA   FLAG
0278 01CA D624           SUI   24#
0279 01CC 32CC40         STA   FLAG
0280 01CF D1     NOOBF   POP   D
0281 01D0 E1             POP   H
```

APPLICATION OF MELCS 8/2 SINGLE-BOARD COMPUTER

(PCA 0801) IN DATA TRANSMISSION THROUGH A MASTER-SLAVE MULTICOMPUTER SYSTEM

```
0282 01D1 C9              RET
0283 01D2 23      FINE2   INX   H
0284 01D3 C3C401          JMP   STORE2
0285 01D6 1D      MOBFP   DCR   E
0286 01D7 C2A701          JNZ   MOBF
0287 01DA C3CF01          JMP   NOOBF
0288*
0289*
0290*   SELECTIVE CHARACTER O TEIGI SURU *
0291 01DD 01      X       DEF   01#
0292 01DE 00              DEF   00#
0293 01DF 02              DEF   02#
0294 01E0 02              DEF   02#
0295 01E1 04              DEF   04#
0296 01E2 04              DEF   04#
0297 01E3 08              DEF   08#
0298 01E4 06              DEF   06#
0299 01E5 10      Y       DEF   10#
0300 01E6 08              DEF   08#
0301 01E7 20              DEF   20#
0302 01E8 0A              DEF   0A#
0303 01E9 40              DEF   40#
0304 01EA 0C              DEF   0C#
0305 01EB 80              DEF   80#
0306 01EC 0E              DEF   0E#
0307 0000                 END   MASTER
```

SLAVE MICROCOMPUTER MAIN PROGRAM LIST

```
**CROSS ASSEMBLER OF 8-BIT MICROPROCESSOR

0001*** SLAVE MICROCOMPUTER MAIN PROGRAM ***
0002*
0003*
0004 4000         RAMST   EQU   4000#
0005 40D0         SCRTCH  EQU   40D0#
0006 40D1         F1      EQU   40D1#
0007 40D2         F2      EQU   40D2#
0008*
0009 0000                 ORG   0000#
0010 0000 3EC0    SLAVE   MVI   A,C0#
0011 0002 D303            OUT   03#
0012 0004 3E81            MVI   A,81#
0013 0006 D307            OUT   07#
0014 0008 31FF40          LXI   SP,4CFF#
0015 000B DB00    WAIT    IN    00#
0016 000D D669            SUI   69#
0017 000F C20B00          JNZ   WAIT
0018 0012 3E6B            MVI   A,6B#
0019 0014 D300            OUT   00#
0020 0016 AF              XRA   A
0021 0017 32D140          STA   F1
0022 001A 32D240          STA   F2
0023 001D 110202          LXI   D,202#
0024 0020 210040          LXI   H,RAMST
0025 0023 CD5500  BACK1   CALL  SI
0026 0026 2F              CMA
0027 0027 32D040          STA   SCRTCH
0028 002A 3AD140          LDA   F1
0029 002D B7              ORA   A
0030 002E CA4D00          JZ    NOPAS1
0031 0031 3AD040          LDA   SCRTCH
0032 0034 CD7100          CALL  SO
0033 0037 3AD240          LDA   F2
0034 003A B7              ORA   A
0035 003B CA5100          JZ    NOPAS2
0036 003E 3AD040          LDA   SCRTCH
0037 0041 77              MOV   M,A
0038 0042 23              INX   H
0039 0043 7D              MOV   A,L
0040 0044 FEC8            CPI   C8#
0041 0046 DA2300          JC    BACK1
0042 0049 00      NO      NOP
0043 004A C34900          JMP   NO
0044*
0045*** NOPASS 1 ***
0046*
0047 004D 37      NOPAS1  STC
0048 004E C32300          JMP   BACK1
0049*
0050*** NOPASS 2 ***
0051 0051 3F      NOPAS2  CMC
0052 0052 C32300          JMP   BACK1
0053*
0054                      EJE
0055*
0056*** SUBROUTINE SI ***
0057*
0058 0055 D5      SI      PUSH  D
0059 0056 DB02    SIBF    IN    02#
0060 0058 E620            ANI   20#
0061 005A C26700          JNZ   SIN
0062 005D 15              DCR   D
0063 005E C25600          JNZ   SIBF
0064 0061 AF              XRA   A
0065 0062 32D140          STA   F1
0066 0065 D1      SIND    POP   D
0067 0066 C9              RET
0068 0067 3E01    SIN     MVI   A,01#
0069 0069 32D140          STA   F1
0070 006C DB00            IN    00#
0071 006E C36500          JMP   SIND
0072*
0073*
0074*** SUBROUTINE SO ***
0075*
0076 0071 F5      SO      PUSH  PSW
0077 0072 D5              PUSH  D
0078 0073 DB02    SOBF    IN    02#
0079 0075 E680            ANI   80#
0080 0077 C28500          JNZ   SOUT
0081 007A 1D              DCR   E
0082 007B C27300          JNZ   SOBF
0083 007E AF              XRA   A
0084 007F 32D240          STA   F2
0085 0082 D1              POP   D
0086 0083 F1              POP   PSW
0087 0084 C9              RET
0088 0085 3E01    SOUT    MVI   A,01#
0089 0087 32D240          STA   F2
0090 008A D1              POP   D
0091 008B F1              POP   PSW
0092 008C D300            OUT   00#
0093 008E C9              RET
0094 0000                 END   SLAVE
```

CONTACT ADDRESSES FOR FURTHER INFORMATION

JAPAN
Electronics Overseas 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 Rd.
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.
2200 West Artesia Blvd.
Compton CA 90220, U.S.A.
Telex: 698246 MELA CMTN
Telephone: (213) 979-6055
(213) 979-6371

Mitsubishi Electronics America, Inc.
100 Wade Ave.
South Plainfield, NJ 07080, U.S.A.
Telex: 833244 MELA SOPH
Telephone: (201) 753-1600
(800) 631-5377

Mitsubishi Electronics America, Inc.
1230 Oakmead Parkway
Suite 206 Sunnyvale CA 94086 U.S.A.
Telex: 172296 MELA SUVL
Telephone: (408) 730-5900

WEST GERMANY
Mitsubishi Electric Europe GmbH
Brandenburger Str. 40
4030 Ratingen, West Germany
Telex: 8585070 MED D
Telephone: (02102) 44089~44094

U.K.
Mitsubishi Electric (U.K.) Ltd.
Otterspool Way, Watford, Hertfordshire
WD2 8LD, U.K.
Telex: 927908
Telephone: (923) 40566~9

MITSUBISHI ELECTRIC