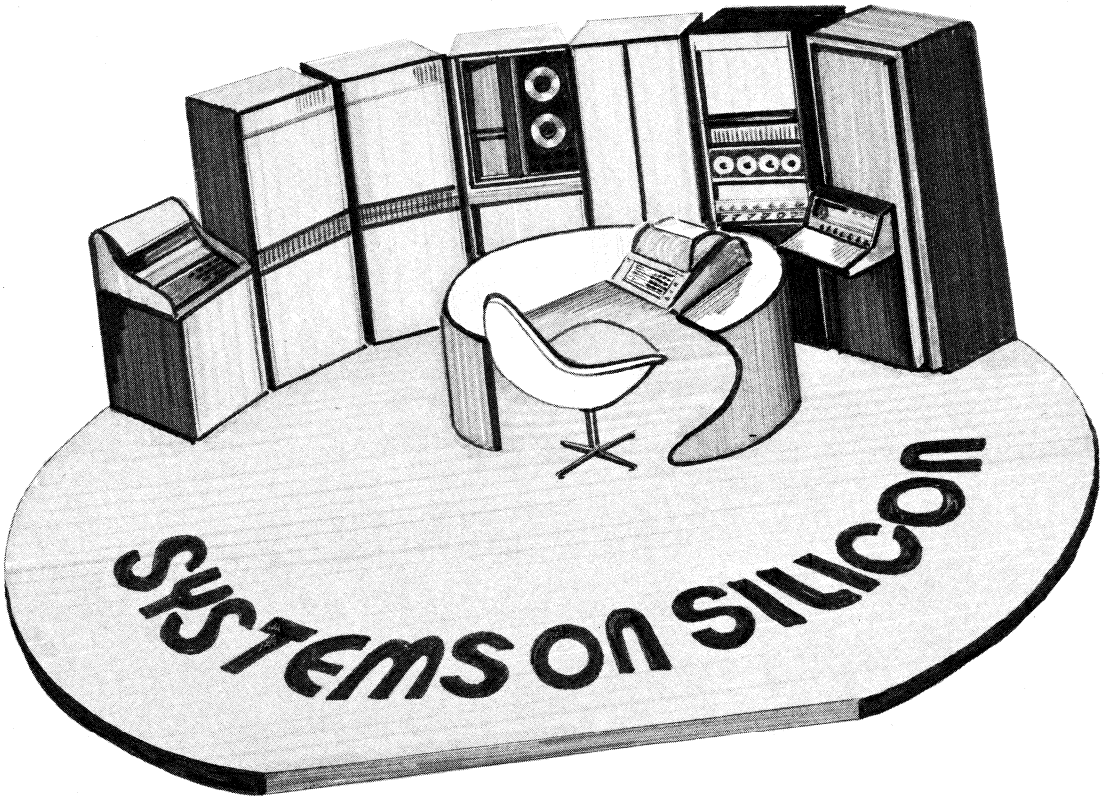
microcomputer components



MOTOROLA Semiconductors

microcomputer components

Printed in Switzerland 57—..79/25.0

EUROPEAN MOTOROLA SEMICONDUCTOR SALES OFFICES

DENMARK

Motorola A/S
Bredebovej 23
DK-2800 Lyngby
Tel. (02) 88 44 55

FRANCE

Motorola Semiconducteurs S.A.
Headquarter
15-17, avenue de Ségur
75007 Paris
Tel. 551 50 61

Sales Office
42, avenue de La Plaine-Fleurie
38240 Meylan (Grenoble)
Tel. (76) 90 22 81

WEST GERMANY

Motorola GmbH, Geschäftsbereich Halbleiter
Headquarter
Münchner Strasse 18
8043 Unterföhring
Tel. (089) 92 481

Sales Offices
Hans-Böckler-Strasse 30
3012 Langenhagen — Hannover
Tel. (0511) 77 20 37

Virnsbergerstrasse 43
8500 Nürnberg
Tel. (0911) 6 57 61

Stralsunder Strasse 1
7032 Sindelfingen
Tel. (07031) 8 30 74/8 30 75

Abraham-Lincoln-Strasse 28
6200 Wiesbaden
Tel. (06121) 76 19 21

HOLLAND

Motorola B.V.
Emmalaan 41
Utrecht
Tel. (030) 51 02 07

ITALY

Motorola S.p.A.
Headquarter
Via Ciro Menotti 11
20129 Milano
Tel. 738 61 41/2/3

Sales Office
Via Portanova 10
40123 Bologna — Tel. 26 69 05

Sales Office
Via Costantino Maes 68
00162 Roma — Tel. 831 47 46

NORWAY

Motorola A/B (Service Office)
Brugt. 1
Oslo 1 — Tel. (02) 41 91 40

SOUTH AFRICA

Motorola South Africa (Pty) Ltd.
P.O. Box 39586
Bramley 2018
Tel. 786 11 84

SWEDEN

Motorola AB.
Virebergsvaegen 19
17140 Solna — Tel. (08) 82 02 95

SWITZERLAND

Motorola Semiconductor Products S.A.
Alte Landstrasse 101
8702 Zollikon — Tel. (01) 65 56 56

UNITED KINGDOM

Motorola Ltd.
Headquarter
York House, Empire Way
Wembley Middlesex
Tel. (01) 902 88 36

Sales Office
10-12, Mount Street, Television House
Manchester M2 5WS, Lancs
Tel. (61) 833 07 31/833 07 34

Sales Office
Colvilles Road, Kelvin Estate
East Kilbride, Scotland
Tel. (3552) 3 91 01

HEADQUARTERS EUROPEAN OPERATIONS

SWITZERLAND

Motorola Inc.
Semiconductor Products Group
16, chemin de la Voie-Creuse, P.O. Box 8
1211 Genève 20
Tel. (022) 33 56 07

FRANCHISED MOTOROLA SEMICONDUCTOR DISTRIBUTORS

AUSTRIA

Elbatex GmbH
Endresstrasse 54 — 1238 Wien
Tel. (222) 88 55 11

BELGIUM

Diode Belgium
Rue Picard 202-204 — 1020 Bruxelles
Tel. (02) 428 51 05

DENMARK

Distributøren Interelko Aps
Hovedgaden 16 — 4622 Havdrup
Tel. (03) 38 57 16

FINLAND

Field Oy
Veneentekijantie 18 — 00210 Helsinki 21
Tel. (80) 692 25 77

FRANCE

Bellion Electronique
Zone Industrielle de Kerscao/Brest
29219 Le Relecq-Kerhuon — B.P. 16
Tel. (98) 28 03 03

Celdis S.A.
53, rue Charles-Frérot — 94250 Gentilly
Tel. (01) 581 00 20

Ets. F. Feutrier S.A. (Main Office)
Rue des Trois-Glorieuses
42270 St-Priest-en-Jarez (St-Etienne)
Tel. (77) 74 67 33

Ets. F. Feutrier S.A.
Avenue Laplace — Zone industrielle
13470 Carnoux
Tel. (42) 82 16 41

Feutrier Ile de France
29, rue Ledru-Rollin — 92150 Suresnes
Tel. (01) 772 46 46

Ets. Gros S.A. (Main Office)
13, rue Victor-Hugo — BP 63
59350 Saint-André-lez-Lille
Tel. (20) 51 21 33

Ets. Gros S.A.
14, avenue du Général-Leclerc — 54000 Nancy
Tel. (83) 35 17 35

Ets. Gros S.A.
5, rue Pascal — 94800 Villejuif
Tel. (01) 678 27 27

S.C.A.I.B. S.A.
80, rue d'Arcueil — Silic 137
94523 Rungis Cedex
Tel. (01) 687 23 13

Sté Commerciale Toutélectric (Main Office)
15-17, Boulevard Bonrepos — 31008 Toulouse
Tel. (61) 62 11 33

Sté. Commerciale Toutélectric
80-83, quai des Queyries — 33100 Bordeaux
Tel. (56) 86 50 31

GERMANY

Alfred Neye Enatechnik GmbH (Ab 1.1.1979)
Schillerstrasse 14 — D-2085 Quickborn/Hamburg
Tel. (04) 106 61 21

Distron oHG
Beheimstrasse 3
1000 Berlin 10
Tel. (030) 342 10 41/45

EBV Elektronik Vertriebs-GmbH (Main Office)
Gabriel-Max-Strasse 72
8000 München 90
Tel. (089) 64 40 55

EBV Elektronik Vertriebs GmbH
In der Meineworth 9
3006 Burgwebel 1 / Hannover
Tel. (051) 39 45 70

EBV Elektronik Vertriebs-GmbH
Oststrasse 129
4000 Düsseldorf
Tel. (0211) 8 48 46

EBV Elektronik Vertriebs-GmbH
Mylíusstrasse 54
6000 Frankfurt 1
Tel. (0611) 72 04 16

EBV Elektronik Vertriebs-GmbH
Alexanderstrasse 63
7000 Stuttgart 1
Tel. (0711) 24 74 81

Jermyn GmbH
Postfach 1180
6277 Camberg
Tel. (06434) 60 05

Mütron Müller & Co. KG
Bornstrasse 22
2800 Bremen
Tel. (0421) 31 04 85

RTG, E. Springorum GmbH+ Co. (Main Office)
Postfach 426
46 Dortmund 1
Tel. (0231) 5 49 51

RTG, E. Springorum GmbH+ Co.
Friedrich-Ebert Damm 112
2000 Hamburg 70
Tel. (040) 693 70 61/62

RTG, E. Springorum GmbH+ Co.
Ungererstrasse 43
8000 München 40
Tel. (089) 36 65 00

RTG, E. Springorum GmbH + Co.
Reutlingerstrasse 87
7000 Stuttgart-Degerloch
Tel. (0711) 76 64 28

RTG, E. Springorum GmbH+ Co.
Mendelssohn-Bartholdy-Strasse 16
6200 Wiesbaden
Tel. (06121) 52 73 09

SASCO Vertrieb von elektronischen Bauelementen GmbH (Main Office)
Hermann-Oberth-Strasse 16
8001 Putzbrunn b. München
Tel. (089) 46 40 61/69

SASCO Vertrieb von elektronischen Bauelementen GmbH
Postfach 3066
4005 Düsseldorf/Meerbusch 3
Tel. (02150) 14 33

SASCO Vertrieb von elektronischen Bauelementen GmbH
Postfach 890214
3000 Hannover
Tel. (0511) 86 25 86

SASCO Vertrieb von elektronischen Bauelementen GmbH
Lorenzer Strasse 15
8500 Nürnberg
Tel. (0911) 20 41 52

SASCO Vertrieb von elektronischen Bauelementen GmbH
Stafflenbergstrasse 24 — 7000 Stuttgart 1
Tel. (0711) 24 45 21

SPOERLE Electronic (Ab 1.1.1979)
Otto-Hahn-Strasse 13 — 6072 Dreieich b. Frankfurt
Tel. (06103) * 3 04-1

Technoprojekt (Main Office)
Heinrich-Ebner-Strasse 13
7000 Stuttgart — Bad Cannstatt
Tel. (0711) 56 17 12

Technoprojekt
Ostring 150 — 6231 Schwalbach/Ts
Tel. (06196) 8 21 00

GREECE

Macedonian Electronics Ltd.
Charilaou — P.O. Box 240 — Thessaloniki
Tel. 30 68 00

Macedonian Electronics Ltd.
Lloyd George 10 — Athens
Tel. (21) 360 95 71

HOLLAND

B.V. Diode
Hollantlaan 22 — Utrecht
Tel. (030) 88 42 14

Manudax Nederland B.V.
Meerstraat 7
5473 ZG Heeswijk (N.B.) — P.O. Box 25
Tel. (41) 39 12 52

HUNGARY

Interag Co., Ltd.
XIII Rajk Laszlo u. 11 — P.O. Box 184
1330 Budapest
Tel. (1) 32 93 40

IRAN

Milcom LTD, Motorola Building
Niloo Street, Vanak Square — Teheran
Tel. 66 12 14/15

ITALY

Celdis Italiana S.p.A. (Main Office)
Via F.lli Gracchi 36 — 20092 Cinisello Balsamo (MI)
Tel. (02) 612 00 41-2-3-4-5

Celdis Italiana S.p.A.
Via Lorenzo il Magnifico 109 — 00162 Roma
Tel. (06) 42 38 55

Celdis Italiana S.p.A.
Via Turati, 33 — 40055 Castenaso (Bologna)
Tel. (051) 78 80 78

Celdis Italiana S.p.A.
Via Mombarcaro 96 — 10136 Torino
Tel. (11) 35 93 12

Celdis Italiana S.p.A.
Via Ognissanti, 83 — 35100 Padova
Tel. (049) 2 68 02

Cramer Italia S.p.A. (Main Office)
Via Cristoforo Colombo 134 — 00147 Roma
Tel. (06) 51 79 81-2-3-4-5-6-7

Cramer Italia S.p.A.
Via S. Simpliciano 2 — 20121 Milano
Tel. (02) 80 93 26

Cramer Italia S.p.A.
Via Umberto 1° 59 — 35100 Padova
Tel. (049) 2 53 74

Cramer Italia S.p.A.
Via Malta 5 — 40135 Bologna
Tel. (051) 42 28 90

Cramer Italia S.p.A.
Corso Traiano 28/15 — 10135 Torino
Tel. (011) 619 20 62

Silverstar Ltd. S.p.A.
Via dei Gracchi 20 — 20146 Milano
Tel. (02) 49 96

Silverstar Ltd. S.p.A.
Via Paisiello 30 — 00198 Roma
Tel. (06) 844 88 41

Silverstar Ltd. S.p.A.
Piazza Adriano 9 — 10139 Torino
Tel. (011) 44 32 75/6 — 44 23 21

NIGERIA

Beckron International
14, Alhaji Bashorun Street
SW Ikoyi — P.O. Box 1896 — Lagos
Tel. 5 66 29

NORWAY

Ola Tandberg Elektro A/S
Skedsmogatan 25 — Oslo 6
Tel. (02) 19 70 30

POLAND

PHZ Transpol S.A. (Intraco Building)
Ul. Stawki 2 — 00-950 Warsaw 1
Tel. (004822) 39 50 79

PORTUGAL

Equipamentos de Laboratorio LDA
Rua Pedro Nunes 47 — Lisbon 1
Tel. 97 02 51

SOUTH AFRICA

L'Electron
704 Main Pretoria Road, Wynberg Tul.
P.O. Box 10544, Johannesburg 2000
Tel. 40 62 96

SPAIN

***Hispano Electronica* S.A. (Main Office)**
Polígono Industrial "Urtinsa"
Apartado de Correos 48 — Alcorcón (Madrid)
Tel. (01) 619 41 08

Hispano Electronica S.A.
Figols, 27-29
Barcelona 14 — Tel. 259 05 22/23

SWEDEN

Interelko AB.
Sandsborgsvägen 50
12233 Enskede — Tel. (08) 49 25 05

AB Gösta Bäckström
Alströmergatan 22 — Box 12009
10221 Stockholm
Tel. (08) 54 10 80

SWITZERLAND

Elbatex AG
Alb. Zwyssig-Strasse 28 — 5430 Wettingen
Tel. (056) 26 56 41

Omni Ray AG
Dufourstrasse 56 — 8008 Zürich
Tel. (01) 34 07 66

TURKEY

ERA Elektronik Sanyi Ve Ticaret A.S.
Ankara Irtibat Bürosü
Gazi Mustapha Kemal Bul. 12
KAT4D79 Yenisehir/Ankara — Tel. 25 49 33

UNITED KINGDOM

A.M. Lock & Co. Ltd
Neville Street, Middleton Road
Oldham, Lancs OL96LF
Tel. (061) 652 04 31

Celdis Ltd.
37-39 Loverock Road
Reading, Berks, RG3, 1ED
Tel. (0734) 585 171

Cramer Components Ltd
Hawke House Green Street
Sunbury on Thames, Middlesex, England
Tel. (9327) 8 55 77

Crellon Electronics Ltd.
380, Bath Road
Slough, Berks SL1 6JE
Tel. (06286) 6 36 11

ITT Electronic Services
Edinburgh Way
Harlow, Essex CM20 (2DF)
Tel. Harlow (0279) 26 777

Jermyn Industries
Vestry Estate — Sevenoaks, Kent
Tel. (732) 5 11 74

Macro-Marketing Ltd.
396, Bath Road
Slough, Berks SL1 6JD
Tel. (06286) 630 11

YUGOSLAVIA

Elektrotehna Ljubljana
Export-Import
Titova 51 — P.O. Box 34-1
61000 Ljubljana
Tel. (61) 32 02 41

Elektrotehna Ljubljana
Filiala Beograd
Marsala Tita 6/1
11000 Beograd
Tel. (011) 69 69 24

EUROPEAN SEMICONDUCTOR FACTORIES

FRANCE

Motorola Semiconducteurs S.A.
Canto Laouzetto — Le Mirail
31023 Toulouse CEDEX
Tel. (61) 40 11 88

GERMANY

Motorola GmbH
Münchner Strasse 18
8043 Unterföhring
Tel. (089) 92 481

UNITED KINGDOM

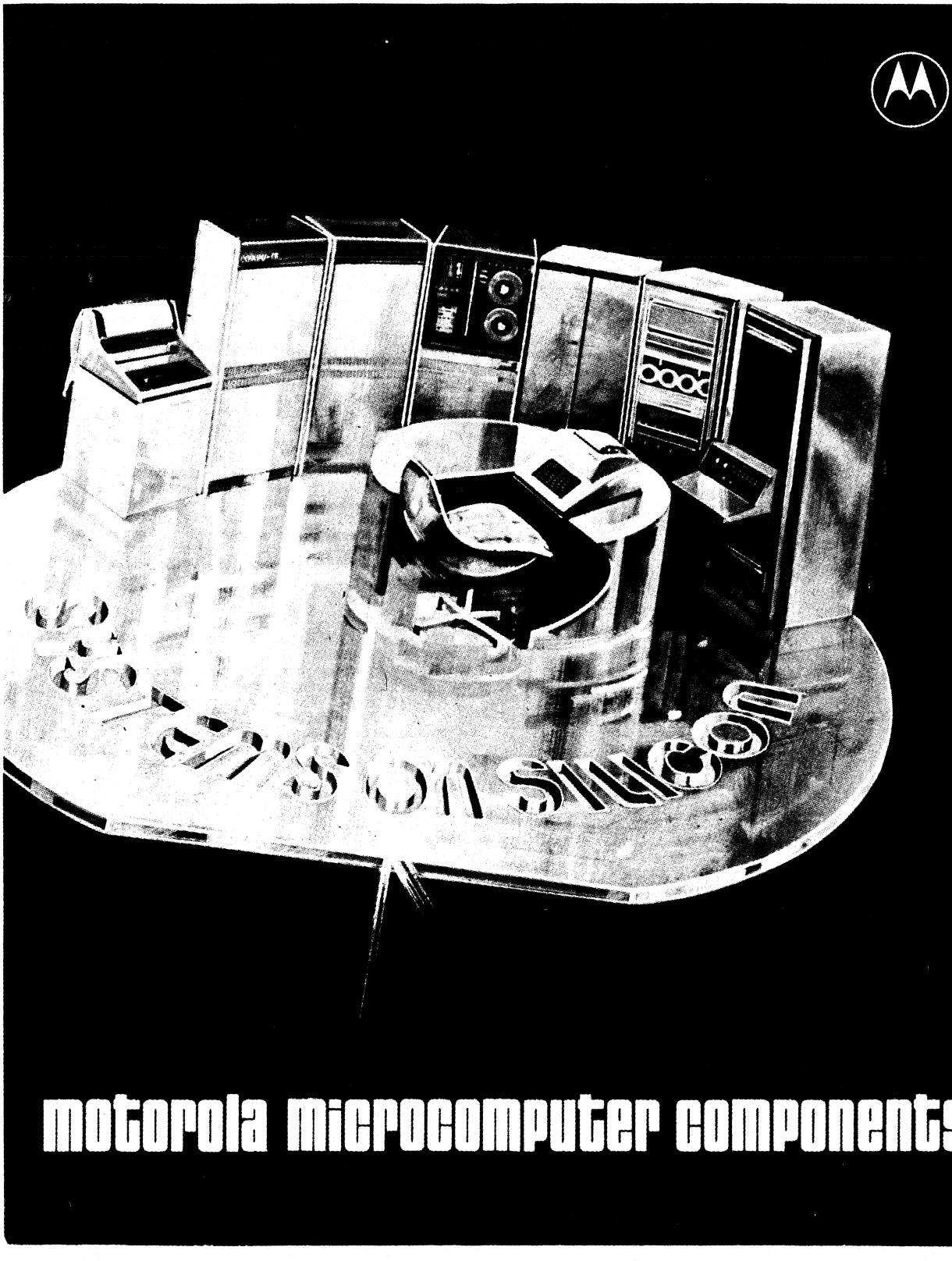
Motorola Semiconductors Ltd.
Colvilles Road, Kelvin Estate
East Kilbride/Glasgow (Scotland)
Tel. (3552) 39 101

contents

Page

Chapter 5 — Bipolar 4-Bit Slice Microprocessor Families

ALPHANUMERIC LISTING OF DEVICES


motorola microcomputer components

What you should consider before you rush into —

microcomputer design

What's so important about the microprocessor that it should have been catapulted into one of the most significant developments in the electronics field over the last decade? Well, certainly not that microprocessors, themselves, represent anything new in the way of circuit implementation. Computer designers had been using them as integral parts of large computers long before they became the buzz-word of the industry. Rather, it's the fact that technological advances into large-scale integration have made the microprocessor the most effective, most reliable, simplest, and *least expensive* way of accomplishing complex electronic functions today. It's not surprising, therefore, that the microprocessor, or in a larger sense, the microcomputer, is dominating the design thinking for most equipment currently in the planning stage.

The microcomputer is such a powerful performer that the vast majority of all possible applications probably could be served by any one of the dozens of different models available. Yet, for each application there's probably one microcomputer system that serves the purpose *best*. And so, before entrusting your design to any specific processor (or supplier) we offer the following four considerations for your investigation:

Choice

Despite occasional claims to the contrary, there's no such thing as a truly *universal* circuit—not from the standpoint of cost-effectiveness. While our M6800 microcomputer system comes as close as any other to serving the bulk of all potential applications, it's "overqualified" for some, and "underqualified" for others. That's why Motorola manufactures a variety of microcomputer systems:

- The M6800 Family—the most pervasive of the general-purpose MPU systems,
- The MC3870—a low-cost, single-chip microcomputer for dedicated applications,
- The MC141000—a CMOS alternative to the above, for lowest possible power dissipation,
- The bipolar M2900 and M10800 systems—for highest speed.

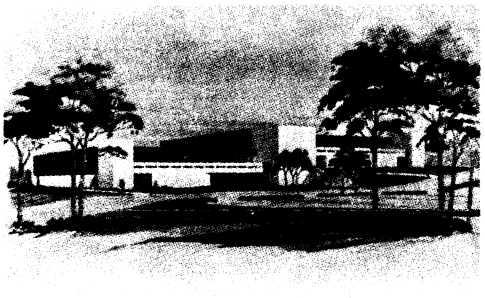
Moreover, each of these has at least one viable alternate source, so that your manufacturing requirements can not easily be compromised.

In addition to components for microcomputer systems, Motorola supplies an extensive line of micromodules—assembled subsystems—for those manufacturers who wish to begin their equipment designs at a higher level.

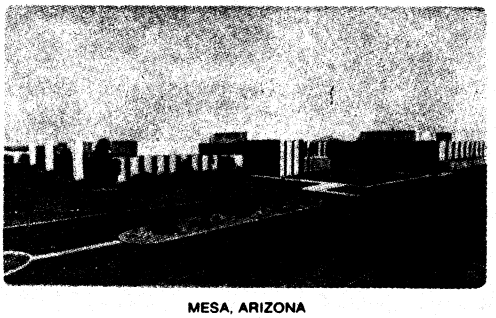
Support

The key to the successful development of a dedicated MPU system and, ultimately, to manufacture and service the system, is an umbrella of support equipment. The Motorola microcomputer product families are complemented with one of the industry's most pervasive arrays of user-oriented development aids, test equipment and support literature. The support hardware system is modularized to permit purchase of just those components required for the complexity of the system to be designed. An extensive software library, including a proven library of user-developed

MOTOROLA MICROPROCESSING MANUFACTURING FACILITIES



AUSTIN, TEXAS
Manufacturing facility for all Motorola MOS products.



MESA, ARIZONA
Bipolar processors and other MPU-related bipolar integrated circuits.

programs, is also available. And Motorola maintains a nationwide network of Field Applications Engineers to assist customers with microcomputer design problems.

Commitment

Chances are your first MPU purchase will not be your last. That's why a primary consideration in the choice of an MPU line on which to hang your designs should be the manufacturer's commitment to the expansion of that specific line—expansion of hardware to keep up with the state of the art while maintaining software compatibility with the existing system.

Motorola's commitment to the M6800 line is manifest not only by the introduction of a second-generation MPU (the MC6802), but by a literal explosion of scheduled peripheral products encompassing over 100 new type numbers in 1977 alone. All are bus-compatible with the original system and utilize the original instruction set. This dedication assures the safety of your software investment even as you switch to increasingly cost-effective hardware for future design and production.

Diversification

Even the most complex microprocessor system (or single-chip microcomputer) isn't a complete system. All need additional components of one kind or another to perform an equipment function. More memory, perhaps, for additional storage capacity; interface circuits to match various peripherals; power devices to drive external equipment. Motorola's product line extends far beyond MPU device lines. As a world leader in solid-state products, we supply devices in almost every semiconductor category—from ICs to discretes; from digital to linear; from MOS to bipolar. This proven solid-state capability is your further insurance of total-product support—today, tomorrow, and in the years ahead.

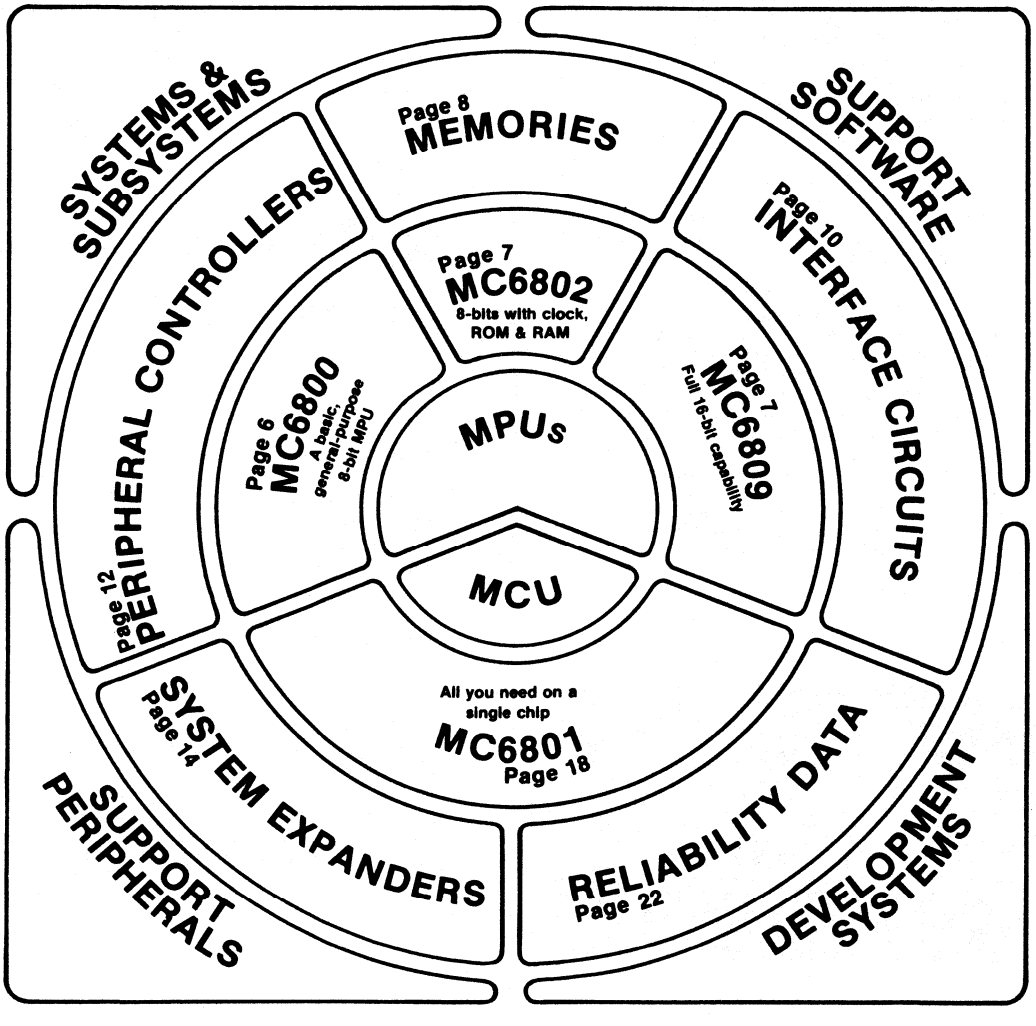
May we help you?

PHOENIX, ARIZONA
Microsystems and support products (56th Street Facility).

table of contents

For a maximum versatility —

the M6800 microcomputer family . . .



at a glance . . .

Inherent in the formula for successful MPU-based designs is the selection of the most cost-effective processor family from among the many systems available today. The M6800 family ranks high in meeting the requirements.

Its NMOS LSI architecture offers bus transfer rates up to 2 MHz, and 8- and 16-bit processing capability.

Its powerful instruction set minimizes memory requirements and enhances system throughout.

The progressive complexity of its basic MPU/MCU building blocks permits system design flexibility that yields cost-effectiveness for any potential applications.

But there are more considerations to the selection of the best microcomputer system than just technical capability. When your production is as heavily dependent on the availability of a set of components as dictated by a commitment to a specific MPU family, a constant and reliable source of supply is of paramount importance. So is the continuing flow of new and related products that guards against system obsolescence.

Motorola is dedicated to the continual expansion of the M6800 system with new products, new designs and expanded peripherals—all tailored to increase the scope and value of your investment.

M6800 LINE FEATURES

For System Design Ease:

- Powerful, variable-length instructions reduce programming complexity and development time.
- 65K memory address capability encompasses the largest program requirements likely to be encountered in microcomputers.
- Single 5-volt power supply operation and bus organization simplifies system design.

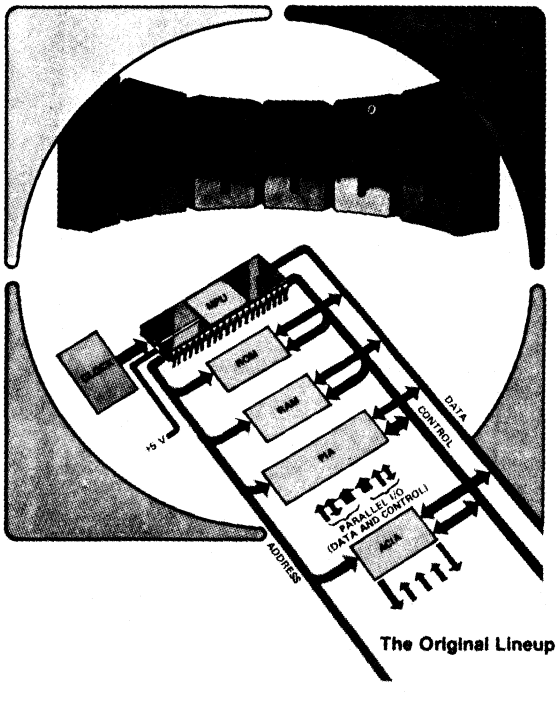
For Maximum Throughput:

- Bus transfer rates to 2 MHz provide high-speed operation.
- Automatic data stacking during interrupts reduces programming complexity.
- 3-state output.
- Vectored restart.

M6800
the MPU

Whether a microcomputer consists of a totally integrated single chip, or is composed of a number of interactive LSI chips, *the microprocessing unit (MPU) is the central control system that determines the eventual application for which the system is best suited.* Its architecture contains the complex routines that permit the system to respond correctly to each of the different "instructions" associated with a particular system. It controls the flow of signals into and out of the computer, routing each to its proper destination in the required sequence to perform an end function.

The M6800 Family currently includes two standard 8-bit MPUs, with a third, a 16-bit unit, scheduled to join the lineup during 1978. And for maximum on-chip power, a complete single-chip microcomputer will join the Family soon (see Page 18).



The Original Lineup

THE MC6800

This microprocessor was the first of the M6800 MPU Family and still remains a highly cost-effective processor for a great many process-control and data-communications applications. Seventy-two powerful instructions and six different addressing modes give it unexcelled capability and a full range of compatible peripheral chips offer the widest possible latitude in system implementation. After years of field experience, the MC6800 has earned an enviable reputation as one of the easiest to use processors available because:

Its bus organized architecture reduces component count and simplifies interconnection;

Its single 5-V supply requirement reduces system complexity and cost;

Its 16-bit address system permits selective addressing of more than 65,000 memory locations;

Its inherent design treats each peripheral as a memory location, thereby reducing programming complexity.

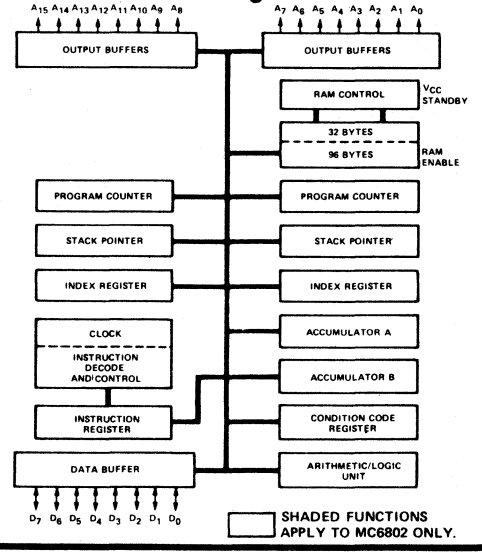
Moreover, to tailor the system to your specific needs at the lowest cost, the MC6800 (and its peripherals) is available in two different packages, three different temperature ranges and three speed ranges, as follows:

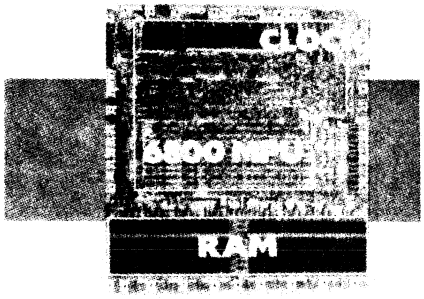
Selection

Temperature Range	Frequency Limit		
	1 MHz	1.5 MHz	2 MHz
0 to 70°C -40 to 85°C -55 to 125°C	MC6800P/L MC6800CL MC6800CQCS	MC68A00P/L	MC68B00P/L

P suffix = Plastic Package L suffix = Ceramic Package

MC6800/6802 — Programmer's Model





MC6802

Take the basic MC6800 MPU, add an on-chip clock and 128 bytes of RAM, and you essentially have the second generation M6800 MPU chip—the MC6802. This versatile processor has all the attributes of the basic unit:

—It is fully compatible with all the peripherals, features the same MPU architecture and capabilities, and works with the same instruction set—

But it reduces the component count of a minimum microcomputer system to only *two*, compared with a minimum of four with the earlier MPU.

The built-in clock operates at a maximum frequency of 1 MHz but, thoughtfully, the chip designers have added an on-chip divide-by-four circuit to permit the use of an external 4-MHz crystal in lieu of a far more expensive 1-MHz crystal. In addition the first 32 bytes of the built-in RAM may be operated in a low-power mode, from an external power source, to prevent the loss of information during a power-down situation.

Utilizing this MPU, a minimum microcomputer system consists of:

1 MC6802 MPU
1 MC6846 ROM-I/O-Timer Unit (Page 10).

Of course, the system is expandable to any requirement with the adapters, expanders and other peripheral chips that are a part of the M6800 Family.

The MC6802 is available in both ceramic (suffix L) and plastic (suffix P) package.

INTRODUCING THE MC6809 MICROPROCESSOR

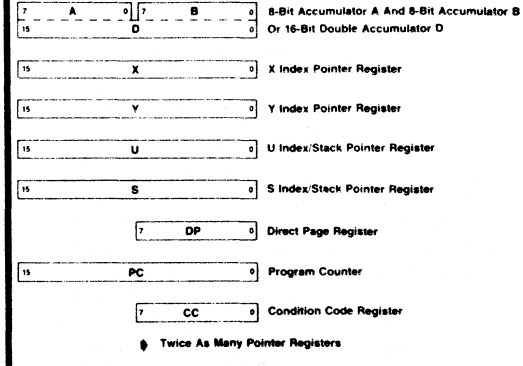
(Coming Soon)

Today, there's a lot of controversy about where a microcomputer turns into a "mini". While a number of benchmarks have been suggested, it is generally conceded that 16-bit processing capability constitutes a minimum "mini" requirement.

Motorola is a microcomputer manufacturer, but the soon-to-be-announced MC6809 Microprocessor at least borders on minicomputer capabilities.

It has 16-bit capability with 50-percent more throughput than the MC6800. It operates at 2 MHz, adds 16 new addressing modes, utilizes an expanded instruction set with high-level language capability, and features a host of other refinements that add functional expansion to, while maintaining compatibility with, the M6800 Microcomputer Components Family.

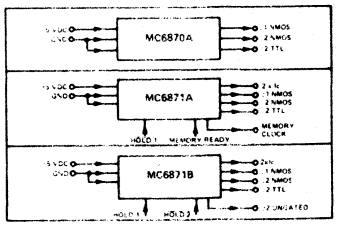
MC6809 Programming Model



M6800 CLOCKS

All M6800-based systems operate with two non-overlapping clock phases, $\phi 1$ and $\phi 2$. A variety of clock modules is available for use with the M6800 MPU (other MPUs have built-in clock). Variations include one monolithic and three hybrid versions offering a number of system design options.

THE HYBRID

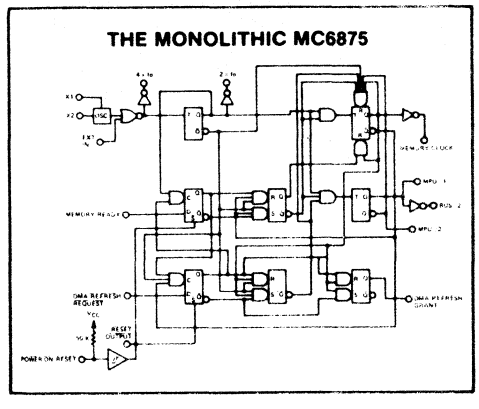


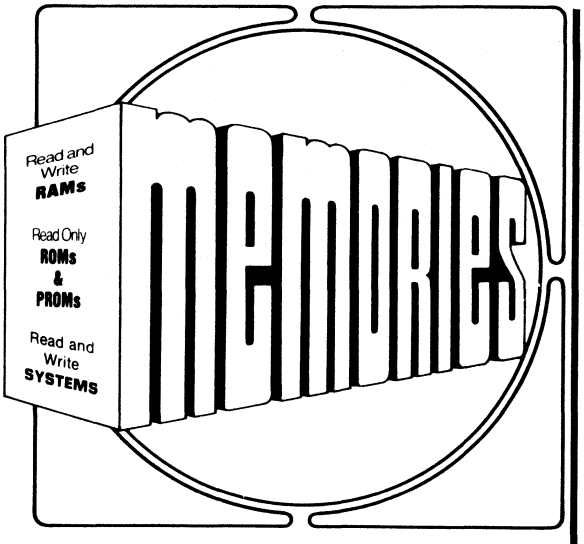
LIMITED FUNCTION
250 kHz to 2.5 MHz

FULL FUNCTION
850 kHz to 2.5 MHz

ALTERNATE FUNCTION
250 kHz to 2.5 MHz

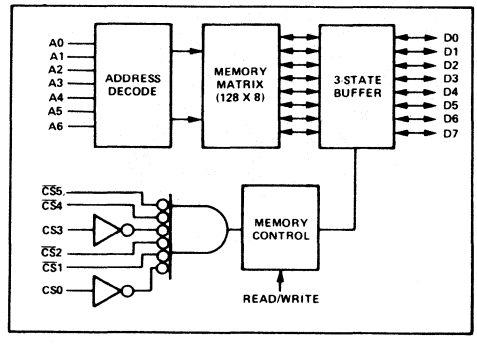
THE MONOLITHIC MC6875





Some microprocessor chips are replete with on-chip memory capacity (see Pages 16, 17, and 18); some aren't. In the interest of system flexibility, the M6800 Family of processors was designed to use external memory. Even the MC6802 processor (128 bytes of on-chip RAM) still requires external ROM and, for very complex applications, may require additional RAM as well. To meet this need, Motorola has developed a series of memories specifically structured for M6800 systems. The memories are bus organized to operate with the MPUs without any additional interface components. Most of the memories described here operate on a single 5-V power supply to obviate additional power supply complexity. However, some devices with additional voltage requirements have been included for their special attributes. These can be adapted to M6800 systems simply by supplying the additional voltages.

RAMs



128 x 8-Bit — MCM6810

This 1024-bit byte-organized memory circuit is designed to take optimum advantage of the M6800 unique features. Fabricated with high-density, N-channel, silicon-gate technology, it provides high-speed access time even at the 5-volt operating levels of the M6800 Family. Three-state, bidirectional input-output buffers interface directly with the bidirectional 8-bit data bus of the overall system. Six individual chip-select inputs, when interconnected with the systems address bus, permit extensive memory expandability and selectivity without the need of separate decoder circuits. Static operation reduces overall system complexity by eliminating refresh requirements.

AVAILABLE OPTIONS

Temperature Range	Frequency Compatibility		
	1 MHz	1.5 MHz	2 MHz
0 to 70°C	MCM6810P/L	MCM68A10P/L	MCM68B10P/L
-40 to +85°C	MCM6810CP/L	MCM68A10CP/L	
-55 to +125°C	MCM6810CJCS		
MIL883B	MCM6810BJCS		

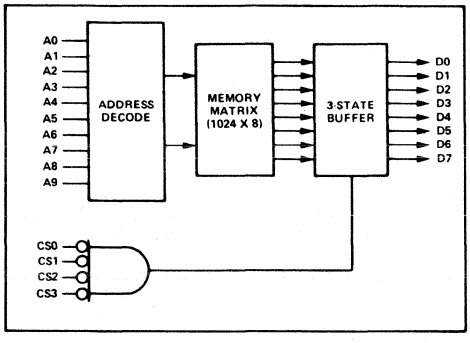
P suffix = Plastic Package L suffix = Ceramic Package

Additional Memory Building Blocks

The above RAM is uniquely tailored to the M6800 system by virtue of a single 5-volt supply requirement and 8-bit word organization. Motorola offers additional RAMs that can easily be adapted to expand M6800 memory capability.

Organization	Operation	Voltage Requirements	Type Number
16,384 x 1	Dynamic	+5, -5, +12	MCM4116
1,024 x 4	Static	+5	MCM2114
4,096 x 1	Dynamic	+5, -5, +12	MCM4027A
4,096 x 1	Static	+5	MCM66L41

ROMs — Mask Programmable



1024 x 8-Bit — MCM6830AL MCM68308L

This series of 8K-bit Read-Only memories houses the user program of a dedicated microprocessors. The customer defines the memory content and Motorola supplies the preprogrammed packages per the given instructions. As all other M6800 components, these devices are byte organized and operate from a single 5-volt supply. Memory expansion is provided through multiple chip-select inputs, with the user specifying the active levels of these inputs.

2048 x 8-Bit — MCM68316

Similar to above, but provides 16K-bit storage and 3 programmable chip-select inputs.

4096 x 8-Bit — MCM68332

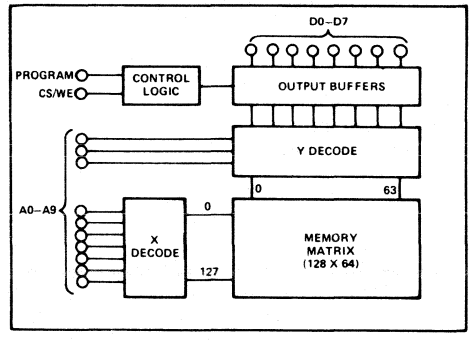
Similar to above, but provides 32K-bit storage and 2 programmable chip-select inputs.

AVAILABLE OPTIONS

Temperature Range	Frequency Compatibility		
	1 MHz	1.5 MHz	2 MHz
8K-Bit			
0 to 70°C	MCM68A30AP MCM68A308P/C	MCM68A30AP/C MCM68A308P/C	MCM68B30AP/C
-40 to +85°C	MCM68A30AC MCM68A308C	—	—
16K-Bit			
0 to 70°C	MCM68A316EP/C	MCM68A316EC	—
32K-Bit			
0 to 70°C	MCM68A332P/C	MCM68A332P/C	—

P suffix = Plastic Package C suffix = Ceramic Package

PROMs — Ultraviolet Erasable



1024 x 8-Bit — MCM68708

For system development, a user-programmable Read-Only Memory is often desirable. This 8K-bit PROM is erasable by exposure to high intensity, ultraviolet light introduced through the transparent lid of the package. Using a commercially available UV-Eraser (Turner Designs), erase time is 30 minutes. For subsequent large-volume production runs, Motorola can supply pin-for-pin compatible mask-programmable ROMs. The device is M6800 compatible, but requires additional power-supply voltage (+12 V, +5 V and -5 V).

AVAILABLE OPTIONS

Temperature Range	Frequency Compatibility		
	1 MHz	1.5 MHz	2 MHz
0 to 70°C	MCM68708L	MCM68A708L	MCM68A708L
-55 to +125°C	MCM68708CJCS	—	—

2048 x 8-Bit — TMS2716L

Similar to above, but provides 16K-bit storage.

2048 x 8-Bit — MCM2716L

Similar to above, but operates from single 5-V supply.

RAM — Memory Systems

To extend the storage capacity of monolithic memory blocks, Motorola has developed a series of memory modules for M6800 microcomputer systems. All of these are compatible with the EXORciser as system development tools, but can be used also with functional microcomputer system where large RAM capacity is required.

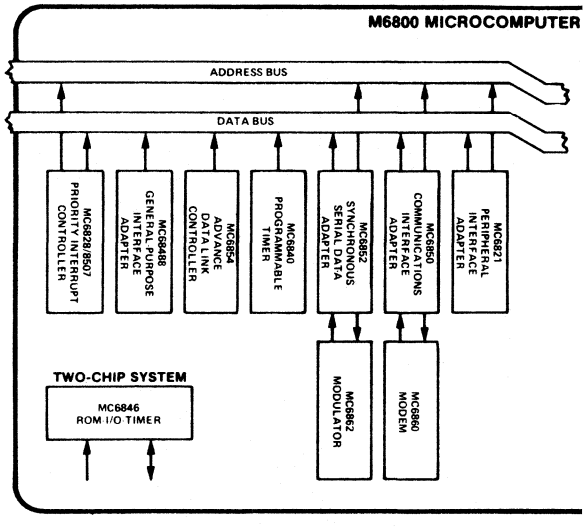
MEX6812-1 — 2048 x 8-Bit Static RAM*
MEX6815-3 — 8192 x 8-Bit Dynamic RAM**
MEX6816-1 — 16384 x 8-Bit Dynamic RAM**
MMS68102 — 16384 x 8-Bit Dynamic RAM+

*Requires 5 V power supply only
**Requires +5 V, +12 V and -12 V power supply
+Requires +5 V, -12 V power supply plus battery pack

interface circuits for **peripherals**

Simple Microcomputers, or those dedicated to one specific application, conceivably could have all the required circuitry on a single chip. General-purpose microcomputers, and those intended for complex system design, do not. The reason—the design of a chip with the memory and interface circuitry compatible with *all* possible end-use applications would make it cost-effective for *none*.

The M6800 system was conceived and designed to encompass an array of LSI components which, in various combinations, provide low-cost solutions to most eventual microcomputer applications. The choice of microprocessor chips, Page 6, together with the selection of the right memory, Page 8, and the most suitable peripheral interface circuits described here, often results in the most effective and least expensive system that can be configured for most applications.



MC6846* — ROM-I/O-Timer

Highly efficient interface chip contains 2048 bytes of ROM, together with a 16-bit programmable timer-counter and an 8-bit bidirectional data port for peripheral interface. In conjunction with MC6802 MPU, it constitutes a versatile 2-chip microcomputer system. Compatibility with other M6800 interface and peripheral circuits permits system expansion to any required additional complexity at low cost.

The built-in ROM provides read-only storage for a minimum microcomputer system and is mask-programmable to the user's specifications. The timer may be programmed to count events, measure frequencies and time intervals, generate square waves, etc. The I/O port is under software control and includes two "handshake" control lines for asynchronous interface with peripherals.

MC6821* — Peripheral Interface Adapter (PIA)

This parallel oriented periperal interface circuit is one of the most important interface circuits available. Contains two I/O circuit blocks, each capable of controlling an independent 8-bit peripheral data bus. Multiple PIAs can be used with a single system and selectively addressed by means of Chip-Select inputs. Each peripheral data line can be programmed to act as an input or output and each of four control/interrupt lines can operate in one of several control modes.

*Available in package configurations and temperature and frequency ranges to match those of the MC6800 MPU described on Page 6.

MC6828/8507 — Priority Interrupt Controller (PIC)

This bipolar device is used to add prioritized responses to inputs of microprocessor systems. The performance has been optimized for the M6800 system, but will serve to eliminate input polling routines from any processor system.

With the PIC, each interrupting device is assigned a unique ROM location which contains the starting address of the appropriate service routine. After the MPU detects and responds to an interrupt, the PIC directs the MPU to the proper memory location.

MC6840* — Programmable Timer

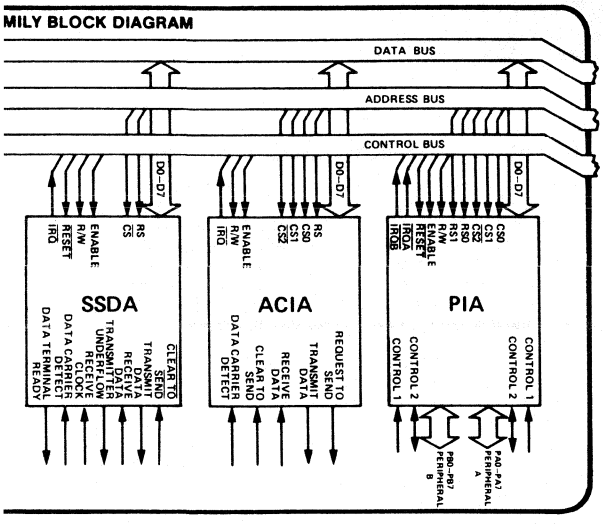
This component is designed to provide variable system time intervals. It has three 16-bit binary counters, three corresponding control registers and a status register. The counters are under software control and may be used to cause system interrupts and/or generate out-put signals. The MC6840 may be utilized for frequency measurements, event counting, interval measuring and similar tasks.

MC68488* — General-Purpose Interface Adapter

The MC68488 GPIA interfaces between the IEEE488 standard instrument bus and the M6800 System. With it, many instruments may be interconnected and remotely and automatically controlled or programmed. Data may be taken from, sent to, or transferred between instruments.

The MC68488 will automatically handle all handshake protocol needed on the instrument bus.

communications and instrumentation



In many applications, input data to a computer comes from program sources that are wired directly to the computer inputs; in others the data is derived from remotely located sources and transmitted to the computer by means of telephone lines. Remote data communications requires additional peripheral equipment—to establish contact; to convert digital signal levels into corresponding transmittable data; to assemble serially transmitted data pulses into byte-sized parallel inputs to the computer (or vice-versa).

The M6800 Family contains a number of compatible LSI components that make the development of communications interface equipment quick, easy and relatively inexpensive.

MC6860* — 0-600 bps Digital Modem

The MC6860 is a MOS subsystem designed to be integrated into a wide range of equipment utilizing serial data communications, including stand-alone modems, data-storage devices, remote-data communications terminals and I/O interfaces for minicomputers.

The modem provides the necessary modulation, demodulation and supervisory control functions to implement a serial data communications link, over a voice grade channel, utilizing frequency shift keying (FSK) at bit rates up to 600 bps.

The modem is compatible with the M6800 Microcomputer Family, interfacing directly with the Asynchronous Communications Interface Adapter to provide low-speed data communications capability.

MC6850* — Asynchronous Communications Interface Adapter (ACIA)

This circuit provides the data formatting and control to interface serial asynchronous data communications information to bus-organized systems.

The parallel data of the bus system is serially transmitted and received by the asynchronous data interface, with proper formatting and error checking. The functional configuration of the ACIA is programmed via the data bus during system initialization. A programmable Control Register provides variable word lengths, clock division ratios, transmit control, receive control, and interrupt control. Three control lines allow the ACIA to interface directly with the MC6860 Digital Modem.

MC6854* — Advanced Data Link Controller (ADLC)

The MC6854 ADLC performs the complex MPU/data communication link function for the "Advanced Data Communication Control Procedure" (ADCCP), High-Level Data Link Control (HLDC) and Synchronous Data Link Control (SDLC) standards.

In a bit-oriented data communication system the data is transmitted and received in a synchronous serial form.

The serial data stream must be converted into parallel, analyzed, and stored (for use by the MPU) in order for data link management to be accomplished. Similarly, parallel data from the MPU system must be serialized with the appropriate frame control information in order to conform to the bit-oriented protocol standards. The Advanced Data Link Controller (ADLC) provides these functions.

MC6852* — Synchronous Serial Data Adapter (SSDA)

Provides interface between the M6800 MPU system and synchronous data terminals such as floppy disk equipment, cassette or cartridge tape controllers, numerical control systems and other systems requiring movement of data blocks. Operates at speeds up to 600 kbps.

MC6862 — Digital Modulator

Offers the necessary modulation and control functions to implement a serial data communications link over voice-grade channels at bit rates of 1200 and 2400 bps.

*Available in package configurations and temperature and frequency ranges to match those of the MC6800 MPU described on Page 6.

New M6800

LSI peripheral controllers

On the one hand there are the microcomputers, and on the other there are the peripherals. Each peripheral has different needs, both functionally and electrically, and, therefore, demands a different interface circuit to adapt it to a specific microcomputer design.

The MC6821 Peripheral Interface Adapter on Page 10 permits first-order peripheral selection and I/O control, but it doesn't provide the complex functional control required by each unique computer peripheral.

Normally, functional peripheral control requires a board-full of SSI/MSI circuits. The LSI circuits described here reduce this requirement to a simple, convenient and relatively inexpensive single package.

MC6843 — Floppy Disk Controller

This 40-pin LSI circuit performs the complex MPU/Floppy Disk interface function. It contains twelve accessible and three nonaccessible internal registers which, together with a Micro-Controller/ROM network, form the communications link between the M6800 MPUs and a wide range of disk drives. Multiple disk drives can be controlled with the addition of external multiplexing rather than additional controllers.

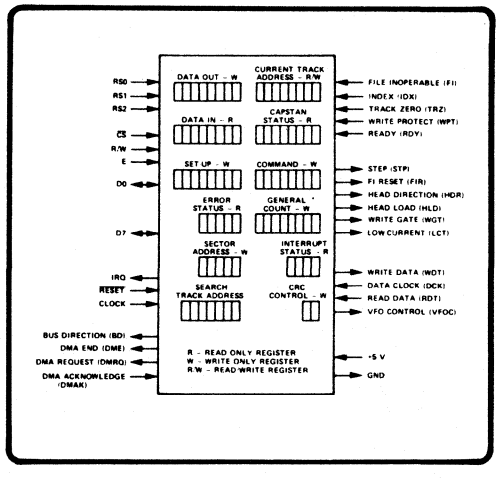
General Description

The three nonaccessible registers provide serial-to-parallel and parallel-to-serial conversions as well as Data-Clock pattern generation and detection. The disk operation is monitored by the MPU via the three status registers. Separate registers provide for Track and Section Address Information.

The Setup Register serves two purposes. One section allows generation of a programmable delay corresponding to the Seek Time of the drive in use. The remaining section provides a programmable settling time delay.

The General Count Register provides the new track number for SEK and STZ commands, and a second count for multi-sector read/write.

One bit in the Command Register selects either Program Control or Direct Memory Access. The remaining bits in the Command Register direct the internal Micro-Control Unit to perform either a micro or macro command. A set of 10 macro commands govern program operation.



Programming Model of MC6843 Floppy Disk Controller

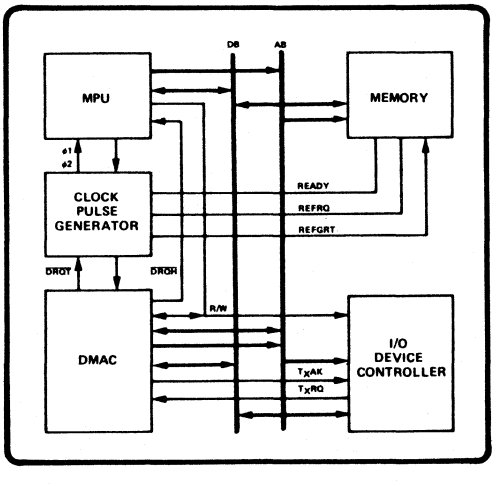
MC6844 — Direct Memory Access Controller

This DMAC works with an M6800 MPU Clock Pulse Generator and an I/O Peripheral Controller, such as the units described here, to facilitate direct access to the computer memory by the peripheral, thus bypassing MPU interactive time delay.

General Description

The MC6844 is operable in three modes: HALT Burst, Cycle Steal and TSC Steal. In the Burst Mode, the MPU is halted by the first transfer request (TxRQ) input and is restarted when the Byte Count Register (BCR) is zero. Each data transfer is synchronized by a pulse input of TxRQ. In the Cycle Steal Mode, the MPU is halted by each TxRQ and is restarted after each one byte of data transferred. In the TSC Steal Mode, DMAC uses the three-state control function of the MPU to control the system bus. One byte of data is transferred during each DMA cycle.

The DMAC has four channels. A Priority Control Register determines which of the channels is enabled. While data is being transferred on one channel, the other channels are inhibited. When one channel completes transferring, the next will become valid for DMA transfer. The PCR also utilizes a Rotate Control bit. Priority of DMA transfer is normally fixed in sequential order. The highest priority is in #0 Channel and the lowest is in #3. When this bit is in high level, channel priority is rotated such that the just-serviced channel has the lowest priority in the next DMA transfer.



Typical Direct Memory Access Diagram

MC6845 — CRT Controller

This single-chip Controller provides the complex interface between a cathode-ray terminal and a microprocessor of the M6800 Family. It is designed to simplify the development and production of equipment such as intelligent terminals, word processing and information display devices.

General Description

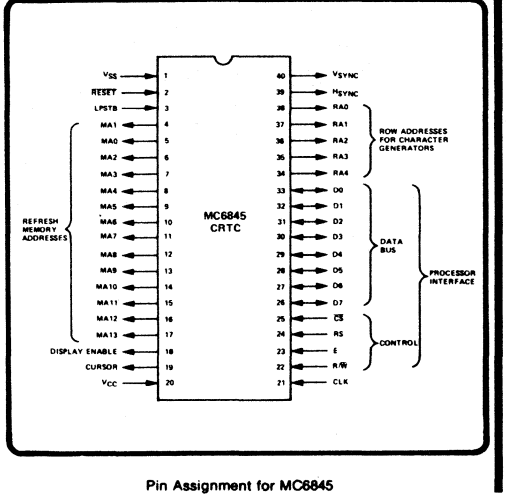
The CRTC consists of the horizontal and vertical counting circuits, a display address generator, a cursor register and comparator, and a light-pen register.

The horizontal and vertical counting circuits generate the signals: Blank, HSYNC, VSYNC, and R0–R4. R0–R4 are row count signals to the external character generator ROM. The numbers of horizontal characters per raster, rasters per character line, character lines per screen and horizontal and vertical SYNC position are programmable by the MPU.

With 14 address lines from the CRTC to the display memory, over 16K of memory may be randomly addressed for display. The CRT may be scrolled or paged through the entire display memory under MPU control.

A light pen strobe input signal allows capture of refresh address in an internal light-pen register.

The cursor control register determines the cursor location on the screen. The cursor format can be programmed for fast-blink, slow-blink, or non-blink appearance, with programmable size.

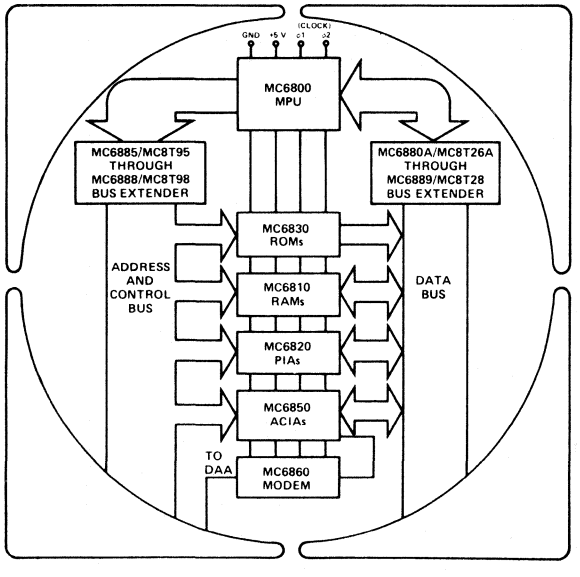


Pin Assignment for MC6845

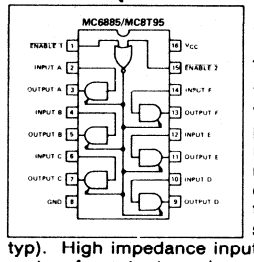
Auxiliary M6800 bipolar circuits

system expanders

The M6800 microprocessor units are capable of directly driving up to eight peripheral components (memories, interface modules, etc.) and one TTL load. A great number of systems can be configured within the framework of these limits. It is conceivable, however, that some very complex systems may require more ancillary circuits than the MPUs can safely accommodate. For systems requiring greater load capacity, and to increase the versatility of the system in general, the monolithic circuits described here have been designed. These circuits are manufactured with bipolar technology, but their operating characteristics are designed for compatibility with other (NMOS) M6800 components.

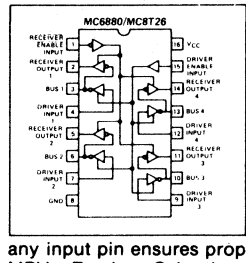


Hex 3-State Buffer/Inverters
MC6885 to MC6888 (MC8T95 to MC8T98)



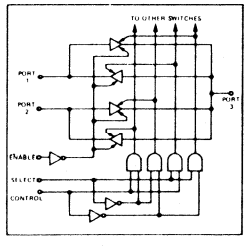
Designed as bus extenders for unidirectional bus systems such as the M6800 Address and Control bus functions. The various devices in this product sequence differ in output (inverting and noninverting) and enable configurations. Schottky technology assures high-speed operation (8 ns typ). High impedance inputs prevent loading of bus system for output requirements up to 40 mA.

Quad Bus Transceiver
MC6880A (MC8T26A) — Inverting
MC6889 (MC8T28) — Noninverting



This component consists of four separate receiver-transmitter combinations, designed for use with a *bidirectional* bus system such as the M6800 data bus. Driver and receiver output currents are 48 mA and 10 mA, respectively. Maximum input current requirement of 200μA at any input pin ensures proper operation with MC6800 MPU. Employs Schottky technology for high-speed operation.

Three-Channel Bidirectional Bus Switch
MC6881



Permits bidirectional exchange of TTL-level signals between multiple microprocessors and the data bus, or the multiplexing of signals to a single processor. Greatly expands versatility of microcomputer systems. Fully compatible with M6800 Family components.

all-on-one-chip

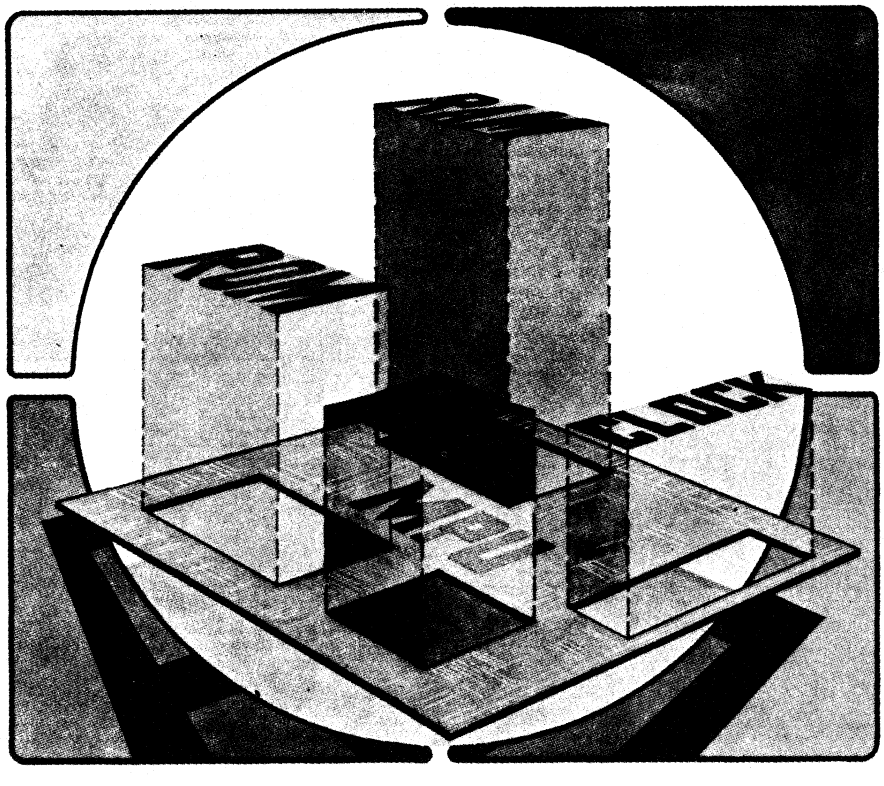
MICROCOMPUTERS

Utilization of large-scale integrated circuits always involves a series of compromises. On the one hand, the more circuitry that is incorporated on a single chip, the more limited is the system designer's control over the exact behavior of the ultimate system. On the other hand, the more complex the chip, the fewer the components needed for system implementation and, concurrently, the simpler the design and the lower the cost.

Applied to microcomputers, the previously described M6800 Family of LSI components gives the designer control over system architecture and functional operation while, at the same time, offering large enough building blocks for simplified system designs at low cost. Yet, there are numerous microcomputer applications where the end requirement permits processing of all necessary functions—MPU, Memory, I/O—into a single chip of managable size. For such applications the single-chip microcomputer results in the most cost-effective final system.

You don't just order a single-chip microcomputer; you have it built for you. Since your dedicated program must be incorporated in the read-only memory on the chip, your order constitutes, in effect, a custom order. Yet, because custom programming can be done in the final metallization stage of chip processing, such processing is relatively inexpensive.

Motorola now supplies two single-chip microcomputers of varying design with still another in the design cycle. These components, described on the following pages, merit serious consideration.



For dedicated applications . . .

single chip microcomputers

Any application that can be formatted within the capacity of an on-chip ROM is a likely candidate for one of the single-chip microcomputers described here. The customer develops and tests his proprietary source program and sends it to Motorola for proper processing of the final chip. From receipt of the source program to delivery of prototype product takes approximately 8 weeks.

The MC3870 8-Bit MOS

Take a powerful Arithmetic Logic Unit (ALU);
Plus 2048 bytes of Read-Only Memory (ROM) and 64 bytes of scratchpad RAM;
Add four ports of TTL-compatible Input/Output; a programmable binary timer capable of operating in the Interval Timer mode, the Pulse-Width Measurement mode and the Event Counter Mode; a built-in clock with internal or external timing capability;
Make it completely compatible with the extensive software library of the popular F8 microcomputer . . .
And *you've got as versatile a single-chip microcomputer as modern technology permits.*

The Motorola MC3870 is a complete 8-bit MOS microcomputer utilizing ion-implanted, N-channel, silicon-gate technology and offers maximum cost-effectiveness for a wide range of control and logic-replacement applications. It is simple to implement (requiring only a single +5-volt ±10% power supply) and power saving in operation (requires only 275 mW, typical). Seventy-six instructions, the entire instruction set of the F8 multi-chip family, impart to the MC3870 a high degree of functional versatility.

Functional Pin Description

P0-0 to P0-7, P1-0 to P1-7, P4-0 to P4-7, and P5-0 to P5-7 are 32 lines which can be individually used as either TTL-compatible inputs or as latched outputs.

STROBE is a ready strobe associated with I/O Port 4. This pin, which is normally high, provides a single low pulse after valid data is presented on the P4-0 to P4-7 pins during an output instruction.

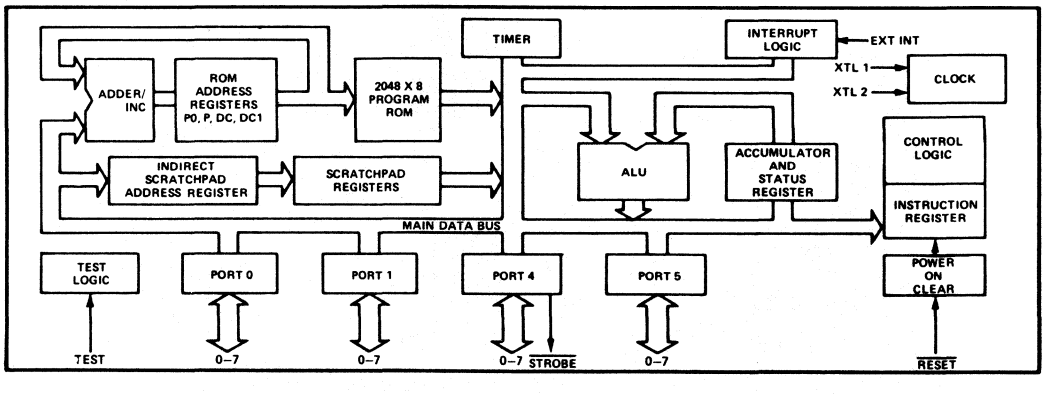
RESET may be used to externally reset the 3870. When pulled low, the 3870 will reset. When then allowed to go high, the 3870 will begin program execution at program location H "000".

EXT INT is the external interrupt input. Its active state is software programmable. This input is also used in conjunction with the timer for pulse-width measurement and event counting.

XTL1 and XTL2 are the time base inputs to which a crystal (1 to 4 MHz), LC network, RC network, or an external single-phase clock may be connected. If timing is not critical, the 3870 will operate from its internal oscillator with no external components.

TEST is an input, used only in testing the 3870. For normal circuit functionality this pin is left unconnected or may be grounded.

V_{CC} is the power supply input (+5 ± 10%).



with Mask- Programmable Read-Only Memory (ROM)

MC141000/1200 Family

Somewhat less sophisticated than the MC3870, this *4-bit* single-chip microcomputer, nevertheless, is more than adequate for a wide range of applications . . . and, it offers some unique advantages. It features CMOS circuitry, providing the lowest possible power consumption, and making it suitable for battery powered, battery back-up, or conventional 5 V operation.

Forty-three basic instructions handle I/O, constant data from ROM, bit control, internal data transfer, arithmetic processing, logic comparison, conditional and nonconditional branching, and subroutines. A 1024 x 8-bit ROM and a 64 x 4-bit RAM handle the on-chip memory requirements.

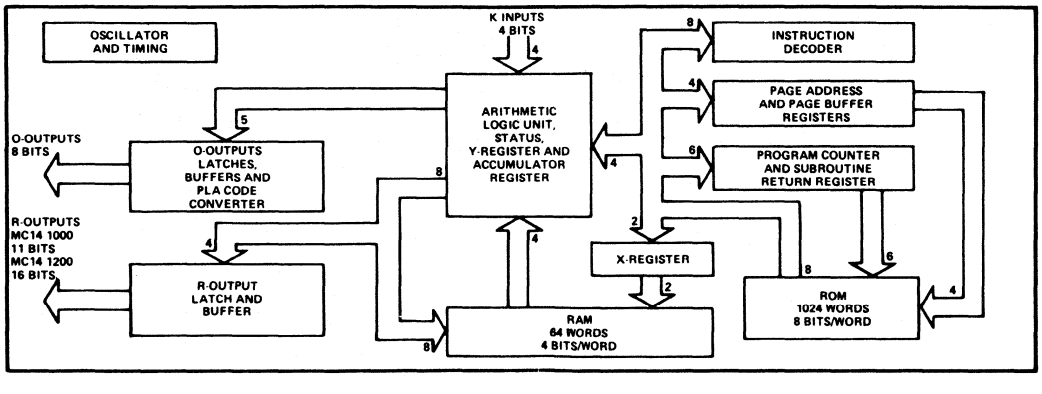
The MC141000/1200 Family is source-program compatible, pin-out compatible and architecturally similar to the well-known PMOS TMS1000 Family, but with the following additional features:

- Power Consumption — only 2.5 mW at 5 V
  only 500 μW at 3 V
- Fully Static Operation
- TTL-Compatible — Drives One TTL Load or Four LSTTL Loads
- Clock Frequency to 700 kHz at VDD = 5 V
- 16 "R" Outputs (MC141200)

Applications

- Appliance Controllers
- Calculators
- Toys
- Radio Controllers
- Communications Controllers
- Data Terminals
- Cash Registers
- Heating/Air-Conditioning Controllers
- Remote Sensing System
- Printing Controllers
- Security Systems
- Power Systems Control
- Automotive Control

The above applications of the MC141000 Family demonstrate its wide potential. Motorola will accept customer programs or will contract complete program development, given the specifications for the application. Customer hardware and software support is already available for developing programs and debugging systems. This consists of one board and a software package using the M6800 EXORciser. Contact your local sales office for status and availability of support equipment.



Single-chip microcomputers . . .

looking ahead- the MC6801

The first M6800 Microcomputer on a single chip

With expected availability before the end of '78, the MC6801 single-chip microcomputer merits serious attention for the next generation of equipment now being designed.

Why?

Well, for starters, here are a few of the more important reasons:

It is characterized with all the circuitry basic to the M6800 MPU . . .

plus

The on-chip clock oscillator and 128 x 8-bit RAM of the MC6802 . . .

plus

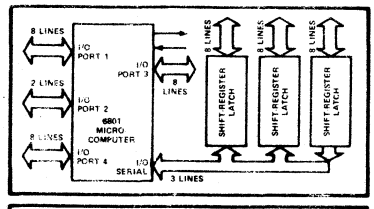
A 2K x 8-bit ROM . . .

A 16-bit timer . . .

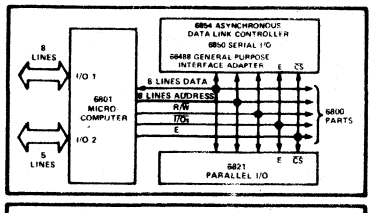
A vast expansion of I/O capacity.

There's not much left to chance in the architecture of the MC6801. There are 31 programmable parallel I/O lines for managing external peripherals, and a serial I/O port for controlling communications equipment. A powerful interrupt capability, with 8 interrupt levels, cuts design costs and boosts performance.

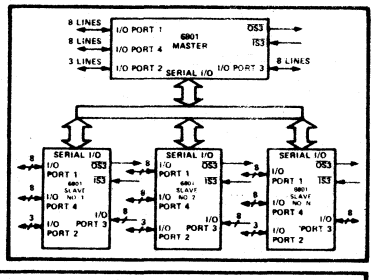
And, just in case the built-in capacity of the MC6801 is insufficient, for a specific application, it retains complete compatibility with the rest of the M6800 line. This means memory expandability up to 65K bytes and a wide variety of other functional options that makes an MC6801 system one of the most powerful in the industry.



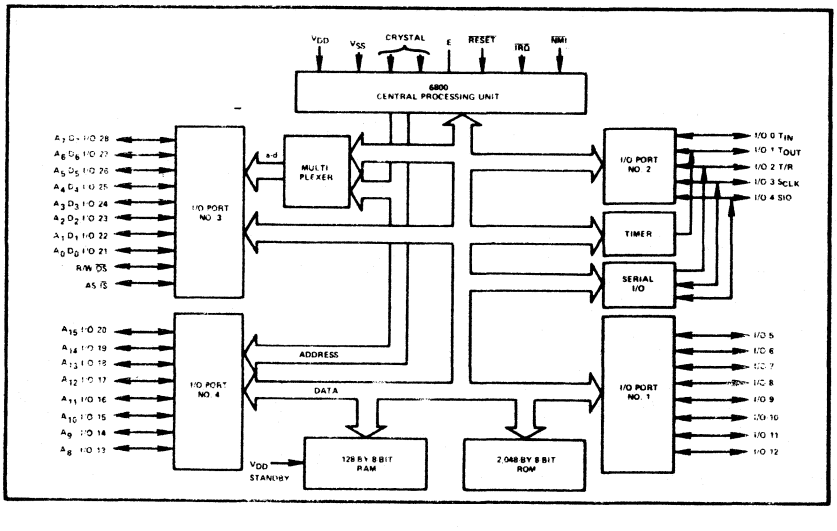
Low-cost Data Communications with simple serial I/O peripherals, such as shift registers, is facilitated by an on-chip Serial I/O port.



Expanding parallel I/O capacity is simplified by use of I/O ports to expand the chips data bus and address bus to external M6800-oriented peripherals.



Using the Serial I/O port, a number of MC6801s can be arranged in a master-slave multiprocessor setup.



Profile of an advanced single-chip microcomputer—The Motorola MC6801

Where high speed counts . . .

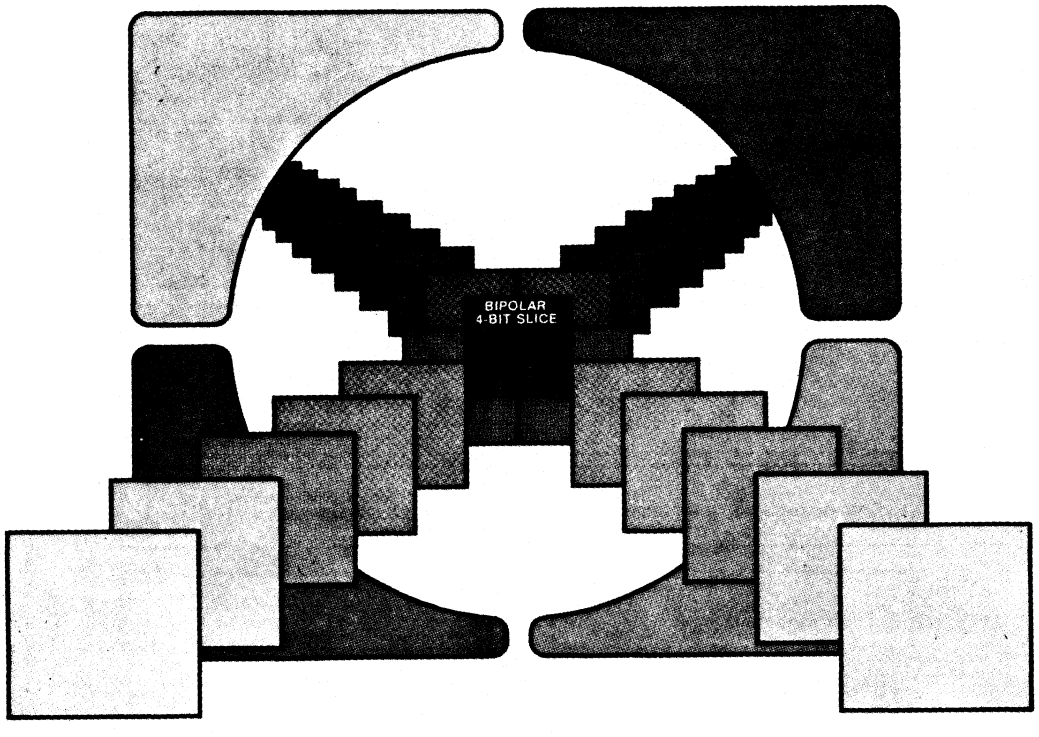
bipolar 4 bit-slice processors

Long before MOS technology became the unofficial standard for microcomputers, computers were being built with bipolar building blocks . . . and they still are. While suffering in comparison with MOS circuits in terms of processing simplicity and, therefore, cost (for LSI configurations), they have an overriding advantage in terms of speed. In many applications requiring real-time responses and complex calculations, bipolar processing represents the only alternative.

But even here, Motorola offers the system designer a choice—a choice between two bipolar circuit configurations: TTL (Schottky, of course) and ECL (Pages 20 and 21).

The bipolar approach to microcomputers differs considerably from the MOS approach. Rather than providing complete microcomputers, or even microprocessors, bipolar designers have evolved a "bit slice" concept. This consists of a series of LSI components representing various major functions of a general-purpose computer. These components are cascadable to form systems of any desired complexity. Thus, the basic 4-bit slice building blocks offered by the two Motorola bipolar families can be expanded into 8-bit, 16-bit and even 32-bit machines. Special circuit architecture makes these building blocks easily microprogrammable.

Unlike MOS MPUs, all of which are accompanied by proprietary software, the bipolar families can be designed to utilize software from any existing computer line.



Where high speed counts . . .

bipolar 4 bit slice processors

Speed and versatility are the key attributes of the bipolar 4-bit slice processor families when compared with the MOS components. Speed is the biproduct of bipolar processing; versatility results from the "slice" concept that permits virtually unlimited expansion of the computer system. Specifically, both families described here consist of 4-bit-wide components that are structured or "sliced" parallel to the data flow. This permits system expansion to larger word lengths simply by connecting several parts (of each type) in parallel.

The M2900 (LSTTL) Family
System Clock Frequency 8.3 to 9.5 MHz

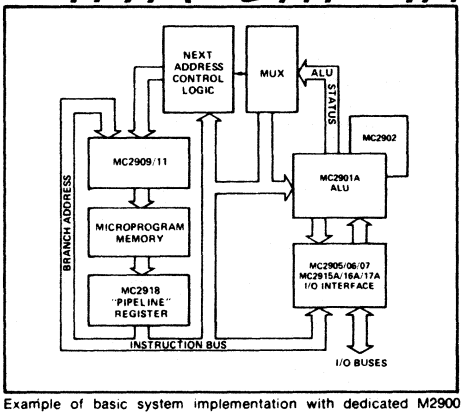
This family of TTL LSI components is microprogrammable for efficient emulation of almost any computing machine.

The heart of the system is the MC2901, a fully expandable 4-bit Arithmetic and Logic Unit (ALU). This device consists of a 16-word by 4-bit two-port RAM, a high-speed ALU, and the associated shifting, decoding and multiplexing circuitry. The ALU function has look-ahead or ripple carry, three-state outputs, and various status-flag outputs. The look-ahead carry function is performed with an MC2902 Look-Ahead Carry Generator in conjunction with the ALU.

The MC2909/2911 are four-bit wide address controllers intended for sequencing through a series of micro instructions contained in the microprogram memory. These controllers have a 4 x 4 stack that allows nesting of subroutines. The system speed can be improved by "pipelining" the contents of the microprogram into MC2918 four-bit registers. Also, the MC2918 register can be used as an address register, condition code register, or for various other register applications.

The I/O interface can be achieved with several different bus transceiver devices. The MC2905/06/07 have high-current sinking, open-collector bus outputs. The driver side has four D-type edge-triggered flip-flops and the receiver side has four D-type latches. The MC2915A/16A/17A are three-state bus output options. These bus transceivers can be used to transfer information from the ALU to the main memory or other bus applications.

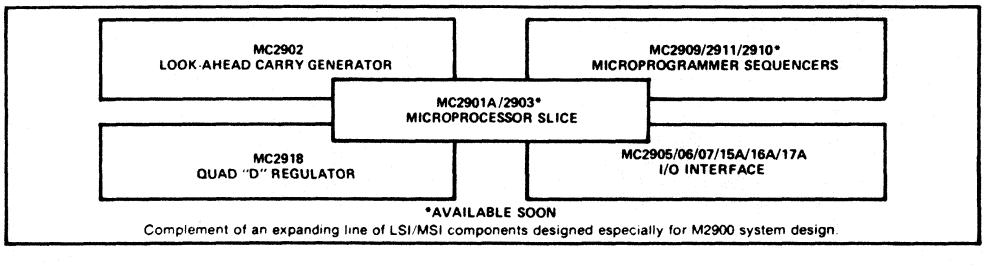
FASTER



Example of basic system implementation with dedicated M2900 components.

COMPATIBLE TTL MEMORIES

	Size (Organization)	Device No.	Access Time (ns max)	P_D (mW typ/pkg)	Temperature (°C)
RAMs	1K Bits (1024 x 1)	MCM93415 MCM93425	35	550	0 to +70
ROMs	512 Bits (64 x 8)	MCM5003 MCM5303	75	500	0 to +70 -55 to +125
		MCM5004 MCM5304	75	600	0 to +70 -55 to +125
	4K Bits (512 x 8)	MCM7640 MCM7641	40 typ	500	0 to +70
	4K Bits (1024 x 4)	MCM7642 MCM7643			



*AVAILABLE SOON

Complement of an expanding line of LSI/MSI components designed especially for M2900 system design.

The M10800 (ECL) Family

System Clock Frequency 10 to 15 MHz

Offering the fastest cycle times of any available bit-slice processor family, the M10800 series of ECL 4-bit processor slices permits the design of high-speed computer systems.

The core of any M10800-based system is the Arithmetic and Logic Unit (ALU). It operates at system-clock frequencies of 10 to 15 MHz, which represent cycle times of 60 to 100 ns. System word size starts at the ALU width of 4 bits, but can be expanded to n × 4 bits by cascading ALU sections. To support the ALU, Motorola has developed several ECL circuits that take care of most of the housekeeping without restricting the processor design.

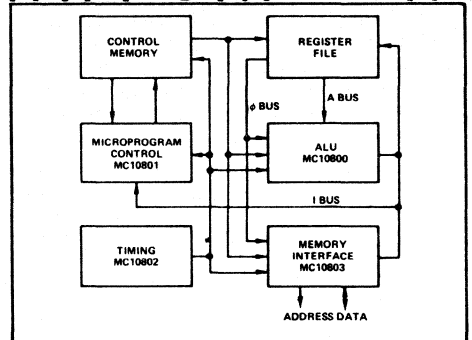
Intended to address the instructions stored in the microprogram memory, the MC10801 Microprogram Controller provides a 4-bit address that can be expanded to any size by cascading controllers. A memory interface unit, the MC10803, also has a cascadable 4-bit output bus, but it connects to the address bus of the main memory and supplies all the read and write addresses.

Acting as a register file, stack or I/O buffer, the MC10806 dual-port memory provides 32 words × 9 bits of temporary storage and can be accessed through either of its ports. For high-speed mathematical operations, the MC10808 Multibit Shifter can handle up to 16 bits and, under software control, can do left-shift, right-shift or rotate operations. Additional MC10808s can be cascaded for larger word lengths.

Other support circuits include the MC10802 Timing Generator and Clock Controller, the MC10804 and MC10805 Bidirectional Bus Translators (ECL-to-TTL and vice-versa) and all of the MECL 10,000 series of logic circuits.

MECL is a trademark of Motorola Inc.

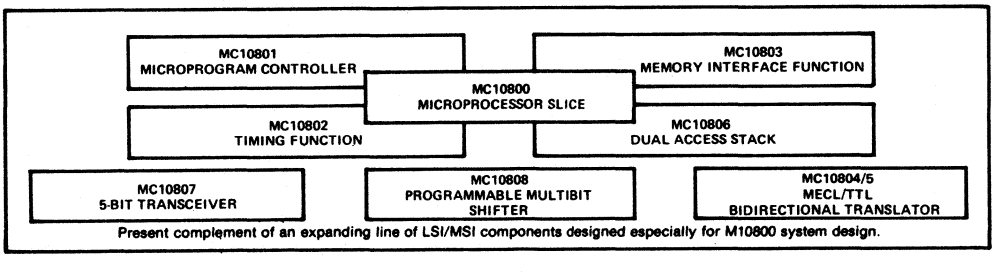
FASTEST



One example of high performance system implementation using basic M10800 Building Blocks.

COMPATIBLE ECL MEMORIES

	Size (Organization)	Device No.	Access Time (ns max)	PD (mW typ/pkg)	Temperature (°C)
RAMs	1K Bits (1024 x 1)	MCM10146	29	500	0 to +75
	256 Bits (256 x 1)	MCM10144	26	420	
		MCM10152	15	500	
	128 Bits (128 x 1)	MCM10147	12	420	-30 to +85
	64 Bits (16 x 4)	MCM10145	15	625	0 to +75
	16 Bits (8 x 2)	MCM10143	14.5	610	-30 to +85
ROMs	256 Bits (32 x 8)	MCM10139A	10 typ	500	0 to +70
	1K Bits (256 x 4)	MCM10149	25	540	-30 to +85



Present complement of an expanding line of LSI/MSI components designed especially for M10800 system design.

Commitment to . . .

component reliability

Reliability data is necessary for the component designer to evaluate and correct failure mechanisms toward the continual improvement of component reliability. It is also required by the system designer to relate component life to system requirements.

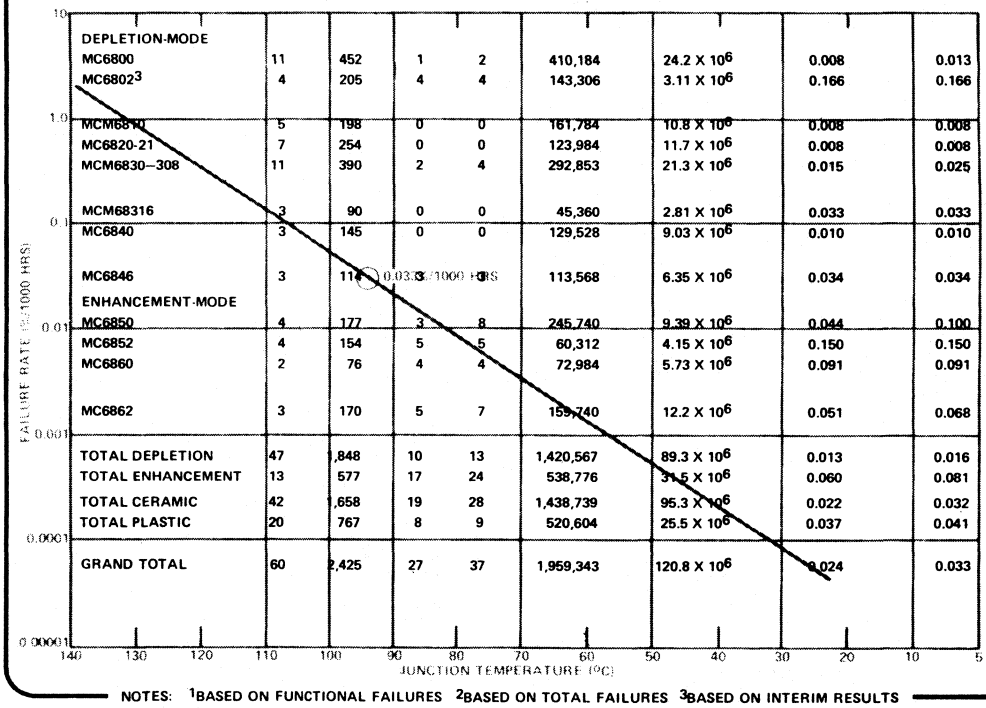
Motorola Reliability Report Number 7750 describes two tests applied to the Motorola M6800 Microprocessor Family—a long-term operation life test, and an environmental temperature-humidity-bias test. The report projects the results of 120 million equivalent device hours, on 12 different device types, as an impressively low *worst-case failure rate of 0.033% per 1000 hours, at the maximum rated temperature of 70°* C. The worst-case estimate was obtained by considering as failures even those devices that ran slightly out of spec" during the test, but which would not normally cause malfunctioning of the associated equipment.

Environmental (THB) tests are being conducted principally on plastic-packaged devices which are much more subject to failures of this nature than the corresponding ceramic-package devices. Tests are conducted at an ambient temperature of 85°C, at a relative humidity of 85%, with a 5-volt bias to accelerate corrosion type failure mechanisms. Available data under these severe operating conditions indicates a (worst-case) median life of 18000 hours.

M6800 FAMILY OPERATING LIFE TEST DATA SUMMARY

FAILURE RATES AS A FUNCTION OF AMBIENT TEMPERATURE

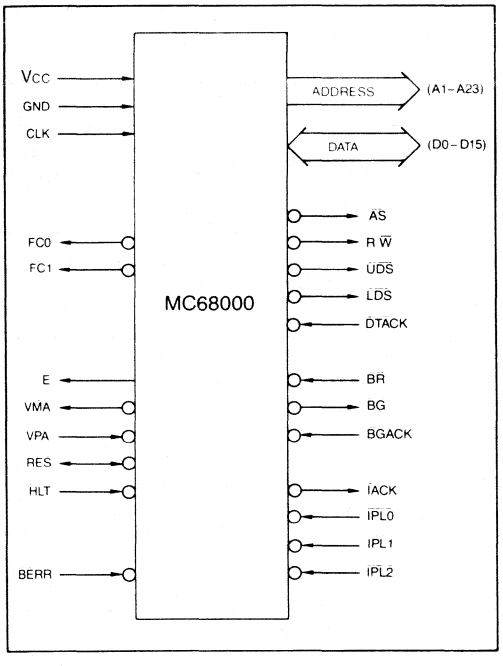
DEVICE TYPE	WAFER LOTS	DEVICES	FAILURES FUNCT	FAILURES TOTAL	125°C ACTUAL DEVICE-HOURS	70°C EQUIV DEVICE-HOURS	70°C SYSTEM[1] FAILURE RATE	70°C TOTAL[2] FAILURE RATE
DEPLETION-MODE								
MC6800	11	452	1	2	410,184	24.2×10^6	0.008	0.013
MC6802[3]	4	205	4	4	143,306	3.11×10^6	0.166	0.166
MCM6810	5	198	0	0	161,784	10.8×10^6	0.008	0.008
MC6820-21	7	254	0	0	123,984	11.7×10^6	0.008	0.008
MCM6830–308	11	390	2	4	292,853	21.3×10^6	0.015	0.025
MCM68316	3	90	0	0	45,360	2.81×10^6	0.033	0.033
MC6840	3	145	0	0	129,528	9.03×10^6	0.010	0.010
MC6846	3	114	0	0	113,568	6.35×10^6	0.034	0.034
ENHANCEMENT-MODE								
MC6850	4	177	3	8	245,740	9.39×10^6	0.044	0.100
MC6852	4	154	5	5	60,312	4.15×10^6	0.150	0.150
MC6860	2	76	4	4	72,984	5.73×10^6	0.091	0.091
MC6862	3	170	5	7	159,740	12.2×10^6	0.051	0.068
TOTAL DEPLETION	47	1,848	10	13	1,420,567	89.3×10^6	0.013	0.016
TOTAL ENHANCEMENT	13	577	17	24	538,776	31.5×10^6	0.060	0.081
TOTAL CERAMIC	42	1,658	19	28	1,438,739	95.3×10^6	0.022	0.032
TOTAL PLASTIC	20	767	8	9	520,604	25.5×10^6	0.037	0.041
GRAND TOTAL	60	2,425	27	37	1,959,343	120.8×10^6	0.024	0.033



NOTES: [1]BASED ON FUNCTIONAL FAILURES [2]BASED ON TOTAL FAILURES [3]BASED ON INTERIM RESULTS

INTRODUCING THE MC68000 ...

MOTOROLA'S ADVANCED COMPUTER SYSTEM ON SILICON



The MC68,000 microprocessor is housed in a 64-pin package that allows the use of separate (non-multiplexed) address and data buses. This large package provides optimum flexibility while at the same time maximizing bus through-put.

PIN IDENTIFICATION & DEFINITIONS

A1–A23	Address Leads	23-bit address bus; capable of addressing 16,777,216 bytes in conjunction with UDS and LDS.
D0–D15	Data Leads	16-bit data bus; transfers 8 or 16 bits of information.
AS	Address Strobe	Indicates valid address & provides a bus lock for indivisible operations.
R/W	Read/Write	Defines bus operation as Read or Write and controls external bus buffers.
UDS, LDS	Data Strobes	Identifies the byte(s) to be operated on according to R/W and AS.
DTACK	Data Transfer Acknowledge	Allows the bus cycle to synchronize with slow devices or memories.
BR	Bus Request	Input to the Processor from a device requesting the bus.
BG	Bus Grant	Output from the processor granting bus arbitration.
BGACK	Bus Grant Acknowledge	Confirmation signal from BG indicating a valid selection from the arbitration process.
IACK	Interrupt Acknowledge	Identifies that the bus is performing an interrupt service cycle.
IPL0, IPL1, IPL2	Interrupt Priority Level	Provides the priority level of the interrupting function to the processor.

FC0, FC1	Function Code	**Provides external devices with information about the current bus cycle.**
CLK	Clock	Master TTL input clock to the processor.
RES	Reset	Provides reset (initialization) signal to the processor and peripheral devices.
HLT	Halt	Stops the processor and allows single stepping.
BERR	Bus Error	Provides termination of a bus cycle if no response or an invalid response is received.
E	Enable	Enable clock for M6800 systems. Identifies addressed area as a 6800 compatible area.
VPA	Valid Peripheral Address	
VMA	Valid Memory Address	Indicates to 6800 family devices that a valid address is on the bus.
V_{cc}	+5 Volts	—
GND	Ground (2 pins)	—

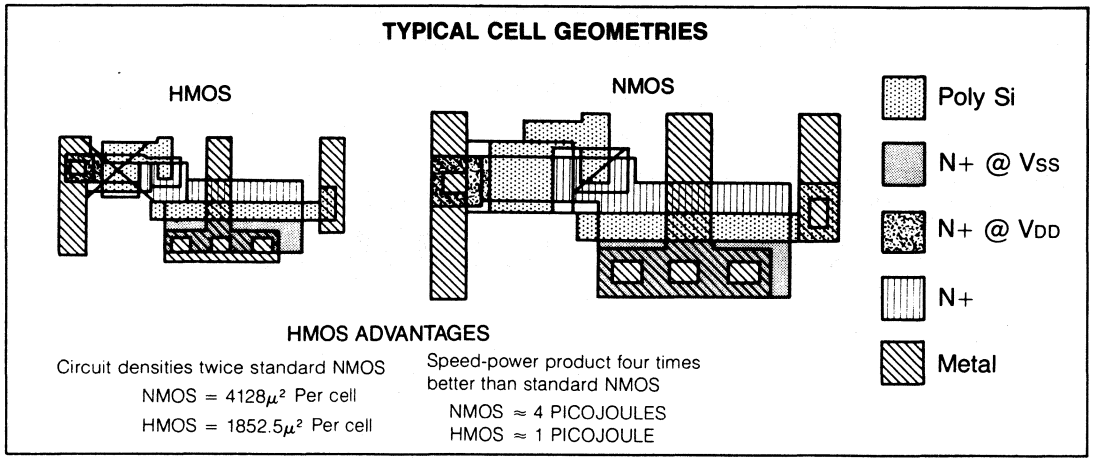


Figure 1: Comparison of HMOS and NMOS Technologies
HMOS Technology used for the MC68000 results in significant improvements to Circuit Densities and Speed-Power Products

Advances in semiconductor technology have provided the capability to put on a single silicon chip, a microprocessor at least an order of magnitude higher in performance and circuit complexity than has been previously available. The MC68000 is the first of a family of such VLSI microprocessors from Motorola. It combines state-of-the-art technology and advanced circuit design techniques with computer sciences to achieve an architecturally advanced 16-bit microprocessor containing over 68000 active devices on a silicon chip. This high density of active elements coupled with an order of magnitude increase in performance over the original MC6800 is the direct result of significant advances in semiconductor technology. Advances such as dry PLASMA etching, projection printing, and HMOS (High density short channel MOS) circuit design techniques (Figure 1) have provided a sound technological base that has allowed Motorola's system engineers, computer scientists and marketing engineers a large degree of innovative freedom. The goals of applying this innovative freedom to microprocessors are to make the microprocessor easy to use, more reliable and more flexible for applications, while maximizing performance.

The resources available to the MC68000 user consist of the following:

- 32-bit data and address registers
- 16 mega-byte direct addressing range
- 61 powerful instruction types
- operations on six main data types
- memory mapped I/O
- 14 addressing modes

Particular emphasis has been given to the architecture to make it orthogonal (regular) with respect to the registers, instructions (including all addressing modes), and data types. Orthogonality makes the architecture easy to learn and program, and, in the process, reduces both the time required to write programs and the space required to store programs. The net result is a great reduction in the cost and risk of developing software.

High systems throughput (up to an aggregate of two million instruction and data word transfers per second) is achieved even with readily available standard product memories with comparatively slow access times. The design flexibility of the data bus allows the mixing of slow and fast memories or peripherals with the processor, automatically optimizing the transfer rate on every access to keep the system operating at peak efficiency.

The hardware design of the CPU was heavily influenced by advances made in software technology. High level language compilers as well as code produced from high level languages must run efficiently on the new generation 16-bit and 32-bit microprocessors. The MC68000 supports high level languages with its consistent architecture, multiple registers and stacks, large addressing range and high level language oriented instructions (LINK, UNLINK, CHK, etc.). Also, operating systems for controlling the software operating environment of the MC68000 MPU are supported by privileged instructions, memory management, a powerful vectored multi-level interrupt and trap structure, and specific instructions (EXG, LDM, STM, TRAP, etc.).

The processor also provides both hardware and software interlocks for multiprocessor systems. The CPU chip contains bus arbitration logic for a shared bus and shared memory environment (shared with other MC68000 processors, DMA devices, etc.). Multiprocessor systems are also supported with software instructions (TEST and SET, TEST and RESET, etc.). The MC68000 offers the maximum flexibility for microprocessor based multiprocessor systems.

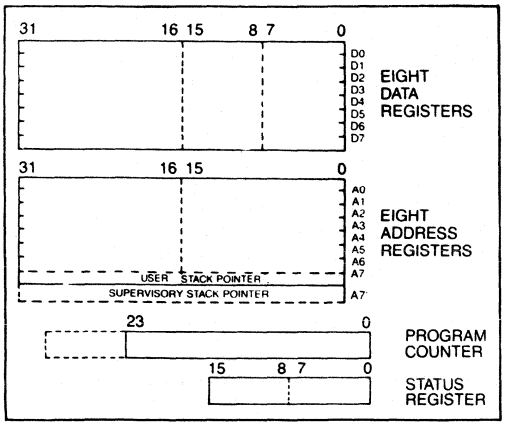


Figure 2: MC68000 Programming Model

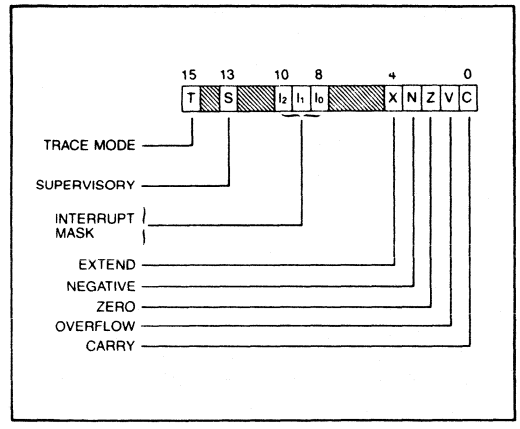


Figure 3: MC68000 Status Register

THE MC68000 CPU

Advanced architecture processors must not only offer efficient solutions to large complex problems but must be able to handle the small, simple problems with proportional efficiency. The CPU has been designed to offer the maximum in performance and versatility to solve simple and complex problems efficiently.

The MC68000 offers sixteen 32-bit registers in addition to the 24-bit program counter and 16-bit status register (Figure 2). The first eight registers (D0–D7) are used as data registers for byte (8-bit), word (16-bit) and long word (32-bit) operations. The second set of eight registers (A0–A7) may be used as software Stack Pointers and Base Address Registers. In addition, the second set of eight registers may be used for word and long word data operations. All of the sixteen registers may be used as Index Registers.

The 24-bit Program Counter provides a memory addressing range of more than 16 mega-bytes (actually 16,777,216 bytes). This large range of addressing capability, coupled with a Memory Management Unit, allows large, modular programs to be developed and operated without resorting to cumbersome and time consuming software bookkeeping and paging techniques.

The Status Register (Figure 3) contains the Interrupt Level Mask (8 levels available) as well as the Condition Code; Overflow (V), Zero (Z), Negative (N), Carry (C), and Extend (X). Additional status bits indicate that the processor is in a TRACE (T) mode or in a SUPERVISORY (S) state . Ample space remains in the Status Register for future extensions of the M68000 family.

Six basic data types are supported. These data types are:

- Bits
- BCD digits
- ASCII characters
- Bytes (8-bits)
- Words (16-bits)
- Long words (32-bits)

In addition operations on other data types such as memory addresses, status word data, etc. are provided for in the instruction set.

DEFINITIONS:

EA = Effective Address
Ax = Address Register
Dx = Data Register
Rx = Address or Data Register used as Index Register
SR = Status Register
PC = Program Counter
D_8 = Eight-Bit Offset
D_{16} = Sixteen-Bit Offset
N = 1 for Byte, 2 for Word and 4 for Long Word
( ) = Contents of
← = Replaces

TABLE I: MC68000 DATA ADDRESSING MODES

REGISTER DIRECT ADDRESSING	
Data Register Direct	EA = D_x
Address Register Direct	EA = A_x
Status Register Direct	EA = SR
ABSOLUTE DATA ADDRESSING	
A. Absolute Short	EA = (Next Word)
B. Absolute Long	EA = (Next two Words)
PROGRAM COUNTER RELATIVE ADDRESSING	
Relative with Offset	EA = (PC) + D_{16}
Relative with Index & Offset	EA = (PC) + (Rx) + D_8
REGISTER INDIRECT ADDRESSING	
Register Indirect	EA = (Ax)
Post-increment Register Indirect	EA = (Ax), Ax ← Ax + N
Pre-decrement Register Indirect	Ax ← Ax – N, EA = (Ax)
Register Indirect with Offset	EA = (Ax) + D_{16}
Indexed Register Indirect with Offset	EA = (Ax) + (Rx) + D_8
IMMEDIATE DATA ADDRESSING	
Immediate	DATA = Next Word(s)
Quick Immediate	INHERENT DATA

The 14 flexible addressing modes, shown in Table I, include five basic types:

- Register Direct
- Immediate
- Register Indirect
- Absolute
- Program Counter Relative

Included in the addressing modes is the capability to do Post-incrementing, Pre-decrementing, Offsetting and Indexing.

THE INSTRUCTION SET

The MC68000 instruction set is rich and full as evidenced by the 61 distinct types shown in Table II. Special emphasis during the design has been given to the instruction set's support of structured high level languages that facilitate ease of programming. Each instruction, with few exceptions, operates on bytes, words, and long words and most instructions can use any of the 14 addressing modes. Combining instruction types, data types, and addressing modes, over 1000 useful instructions are provided. These instructions include signed and unsigned multiply and divide, "quick" arithmetic operations, BCD arithmetic and extended operations (through traps). The processor offers the most comprehensive and flexible instruction set of any microprocessor of any class, available today. Additionally, it's highly orthogonal, proprietary microcoded structure provides a sound flexible base for the future.

REDUCED SOFTWARE COST AND RISK

Advances in VLSI semiconductor technology have resulted in a significant reduction in the cost of computer hardware in recent years. The MC68000 microprocessor, for example, provides in a single integrated circuit package computing power that just a decade ago would have been three or four orders of magnitude more expensive. Software costs during this same period of time have, as a percentage of total system cost, increased significantly. This has been due primarily to inflation and the labor intensive nature of programming. Without significant architectural advances in computers, this trend can do nothing but continue. One of Motorola s major goals in developing this new microprocessor has been to reduce the costs of software. Many innovative features have been incorporated to make programming easier, faster and more reliable.

An Orthogonal 16-BIT MPU — The highly orthogonal or regular structure of the MC68000 microprocessor greatly simplifies the effort required to write programs in Assembly Language as well as in High Level Languages. Operations on integer data in registers and memory are independent of the data itself. Separate special instructions that operate on byte (8-bit), word (16-bit) and long-word (32-bit) integers are not necessary. The programmer merely has to remember one mnemonic for each type of operation and then specify data size, source addressing mode and destination addressing mode. This has helped keep the total number of instruction mnemonics for the M68000 to an easily remembered, yet complete, 61 types, eleven fewer than on Motorola's MC6800.

The dual operand nature of many of the instructions significantly increases the flexibility and power of this new Motorola microprocessor. Consistency again is maintained since *all* data registers and memory locations may be either a source or destination **for most operations on integer data.**

TABLE II: MC68000 INSTRUCTION SET SUMMARY

MNEMONIC	DESCRIPTION
ABCD	Add Decimal with Extend
ADD	Add
ADDX	Add with Extend
AND	Logical And
ASL	Arithmetic Shift Left
ASR	Arithmetic Shift Right
BCC	Branch Conditionally
BCHG	Bit Test and Change
BCLR	Bit Test and Clear
BRA	Branch Always
BSET	Bit Test and Set
BSR	Branch to Subroutine
BTST	Bit Test
CHK	Check Register Against Bounds
CLR	Clear Operand
CMP	Arithmetic Compare
DCNT	Decrement and Branch Non-Zero
DIVS	Signed Divide
DIVU	Unsigned Divide
EOR	Exclusive Or
EXG	Exchange Registers
EXT	Sign Extend
JMP	Jump
JSR	Jump to Subroutine
LDM	Load Multiple Registers
LDQ	Load Register Quick
LEA	Load Effective Address
LINK	Link Stack
LSL	Logical Shift Left
LSR	Logical Shift Right
MOVE	Move
MULS	Signed Multiply
MULU	Unsigned Multiply
NBCD	Negate Decimal with Extend
NEG	Two's Complement
NEGX	Two's Complement with Extend
NOP	No Operation
NOT	One's Complement
OR	Logical Or
PACK	Pack ASCII to BCD
PEA	Push Effective Address
RESET	Reset External Devices
ROTL	Rotate Left without Extend
ROTR	Rotate Right without Extend
ROTXL	Rotate Left with Extend
ROTXR	Rotate Right with Extend
RTR	Return and Restore
RTS	Return from Subroutine
SBCD	Subtract Decimal with Extend
SCC	Set Conditional
STM	Store Multiple Registers
STOP	Stop
SUB	Subtract
SUBX	Subtract with Extend
SWAP	Swap Data Register Halves
TAS	Test and Set Operand
TRAP	Trap
TRAPV	Trap on Overflow
TST	Test
UNLK	Unlink Stack
UNPK	Unpack BCD to ASCII

The addressing modes have been kept simple without sacrificing efficiency. All fourteen addressing modes operate consistently and are independent of the instruction operation itself. Additionally, all address registers may be used for the Direct, Register Indirect and Indexed addressing modes. (Immediate, Program Counter Relative and Absolute addressing by definition do not use address registers). For increased flexibility, any data register — as well as any address register — may be used as an Index Register. Address register consistency is maintained for stacking operations since any of the eight address registers may be utilized as User Program Stack pointers with the Register Indirect Post-increment/Predecrement addressing modes. Register A7, however, is a special register that, in addition to its normal addressing capability, functions as the System Stack Pointer when stacking the Program Counter and Status Register for subroutine calls, traps and interrupts; while in the supervisory mode.

Structured Modular Programming — The art of programming microprocessors has evolved rapidly in the past few years. Numerous advanced techniques have been developed to allow easier, more consistent and reliable generation of software. In general, these techniques require that the programmer be more disciplined in observing a defined programming structure such as modular programming. Modular programming allows a required function or process to be broken down in short modules or subroutines that are concisely defined and easily programmed and tested. Such a technique is greatly simplified by the availability of advanced macro assemblers and block structured High Level Languages such as PASCAL. Such concepts are virtually useless, however, unless parameters are easily transferred between and within software modules that operate on a reentrant and recursive basis. (To be reentrant a routine must be usable by interrupt and non-interrupt driven programs without the loss of data. A recursive routine is one that may call or use itself). The MC68000 microprocessor provides the necessary architectural features to allow efficient reentrant modular programming. The "LINK" and "UNLINK" instructions reduce subroutine call overhead in two complementary instructions by allowing the manipulation of linked lists of data areas on the stack. The "STM" (Store Multiple Registers) and "LDM" (Load Multiple Registers) instructions also reduce subroutine call programming overhead. These allow the loading or storing, via an effective address, multiple registers that are specified by the programmer. Sixteen software trap vectors are provided with the "TRAP" instruction and are useful in operating system call routines or user generated "macro routines." Other instructions that support modern structured programming techniques are PEA (Push Effective Address), LEA (Load Effective Address), RTR (Return to Restore) as well as the normal JSR, BSR and RTS.

Of course, the powerful vectored priority interrupt structure of the microprocessor allows straightforward generation of reentrant modular Input/Output routines. Eight maskable levels of priority with 192 vector locations provide maximum flexibility for I/O control. (A total of 256 vector locations are available for interrupts, hardware traps, and software traps.)

Improved Software Testability — One of the major tasks the system programmer encounters when writing software for microcomputers is the detection and correction of errors, or "debugging." The time taken to "debug" software nearly always exceeds the time it takes to write the software. In practice, the old 20/80 rule often applies: "The last 20% of the job requires 80% of the effort." The microprocessor incorporates several features that reduce the chance for errors. These features, such as Orthogonality and the Structured Modular Programming capability, have already been discussed.

Of major importance to the systems programmer are features that have been incorporated specifically to detect the occurrence of programming errors or "bugs." Several hardware traps, provided to indicate abnormal internal conditions of the MC68000 processor, detect the following error conditions:

Word access with an odd address
Illegal instructions
Unimplemented instructions
Illegal addressing mode
Illegal Memory access (bus error)
Overflow on divide (divide by zero)
Overflow condition code (separate instruction TRAPV)

Additionally, the sixteen software TRAP instructions may be utilized by the programmer to provide applications oriented error detection or correction routines.

An additional error detection tool is the CHK (Check Register Against Bounds) instruction used for array bound checking by verifying that $0 \leq (REG) < LIMIT$. A trap occurs if the register contents are negative or greater than the limit.

Finally, the MC68000 includes a facility that allows instruction-by-instruction tracing of a program being debugged. This TRACE MODE results in a trap being made to a tracing routine after each instruction execution. The TRACE MODE is available to the programmer when the microprocessor is in the SUPERVISORY state as well as the USER state, but may only be entered while in the supervisory state. The SUPERVISORY/USER states provide an additional degree of error protection for the microprocessor by providing memory protection of selected areas of memory when an external memory management device is used.

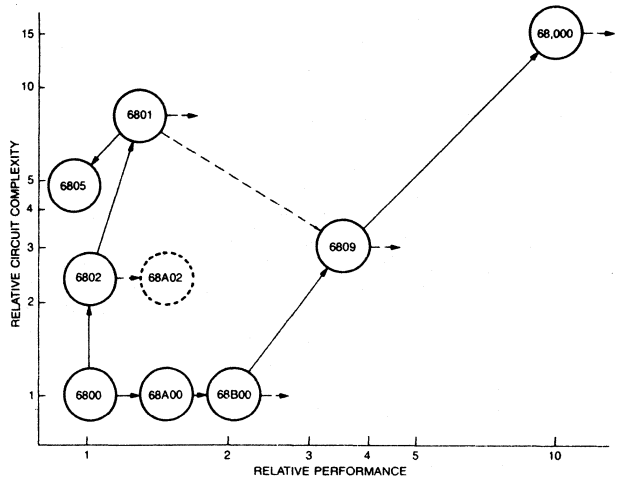


Figure 4:
Motorola's
Microprocessor Evolution

FUTURE FLEXIBILITY

Microprocessor VLSI circuit technology is advancing at an ever increasing rate. For example, the Motorola MC6800 — originally introduced in 1974 — has evolved into a number of more advanced products. This evolution has been along two paths: increased functionality, with the MC6802 and MC6801 microcomputers, and increased performance with the MC68A00, MC68B00 and MC6809 microprocessors. (Figure 4). The sound, well planned, architectural base provided by the original MC6800 made it possible to develop these improved products while taking full advantage of the major speed and density enhancements to NMOS VLSI. This was accomplished while maintaining an unprecedented degree of compatibility and consistency with the original MC6800 MPU.

Similarly, a major consideration in the development of the MC68000 microprocessor has been to provide a good, solid, but flexible, base for future extensibility. Several architectural concepts have been incorporated that will allow this advanced product to be enhanced as semiconductor technological advances are made. For example, the highly orthogonal structure of the CPU allows operations on 8-bit, 16-bit and 32-bit integers without the need for concatenation of registers or multiplexing of internal data buses. This regular structure of the CPU lends itself to a more consistent, reliable design that can be easily expanded.

The MC68000 incorporates a proprietary multilevel micro-programmed structure that allows significant versatility in the implementation of instructions. In fact, more than one-eighth of the instruction op-code map has been set aside specifically for implementation of future instructions. In the interim, **user implementation of instructions not currently in the instruction set is possible through the use of the** TRAP instruction, as well as the hardware trap structure.

MEMORY MANAGEMENT OF LARGE ADDRESSING SPACE

The ever-decreasing costs of semiconductor memories in combination with the use of high level languages and sophisticated disc operating systems allow Motorola's new generation of high performance microprocessors to be used in complex, memory intensive applications. In order to meet the needs of such applications, the MC68000 is capable of directly addressing more than 16 mega-bytes of memory. This large address space is directly accessed and managed very efficiently on a word or byte basis since operand size is specified by the instruction. The use of Upper Data Strobe (UDS) and Lower Data Strobe (LDS) signals allows easy access to high order bytes, low order bytes, or words.

Several additional useful features are provided that allow the programmer to efficiently manage memory usage. Powerful memory addressing modes such as Register Indirect, Indexed, Short and Long Absolute, and Program Counter Relative allow well-ordered access to specific memory locations. These addressing modes allow easy address calculations (Register Indirect and Indexed), direct access to memory location (Short and Long Absolute) and position independent or relocatable coding (Program Counter Relative). Of course, the Pre-decrement/ Post-increment Register Indirect Addressing modes also allow efficient management of data in memory

by permitting the programmer to generate as many as eight concurrent stacks or queues. Another feature that allows the programmer to manage the use of memory is the CHK (Check Register Against Bounds) Instruction. This instruction permits the software implementation of a basic memory protection/ management structure.

Still another significant feature provided in the MC68000 microprocessor is the distinction between a USER and a SUPERVISOR mode. The SUPERVISOR mode permits certain protected operations within the processor system. Of particular interest is that an external Memory Management Controller may be used when the processor is in the USER mode to manage the large address space for the programmer. The controller's memory management operations are transparent to the programmer when in the USER mode and can be changed or updated only in the SUPERVISOR mode. The Memory Management Controller provides both management of a variable number of variable size segments (Memory Segmentation) and dynamic management of multi-task memory relocation and protection. The Memory Management Controller regulates access to storage segments that are dedicated to read only data, read/write data, program code and protected data/ code.

REDUCED CODE DENSITY AND IMPROVED SPEED

With the advent of low cost, very high density VLSI RAMS and ROMS, it might incorrectly be assumed that the number of bytes of code needed to execute a given program is no longer important. Code density, however, is very critical, since microprocessor speed is highly dependent upon the number of executed instruction words. During the early development of Motorola's MC68000 microprocessor, extensive studies were made of the use of instructions and sequences of instructions in many microprocessor applications. These studies identified not only statically frequent instructions but also dynamically frequent instructions. (The dynamic frequency of instructions is a measure of how often an instruction is executed while static frequency is a measure of how often it occurs in a program listing or is encountered by an assembler). The major contributer to the increased efficiency, as a result of the studies, is the highly regular or orthogonal structure of the architecture. The consistency of the architecture, instruction set, and addressing modes significantly reduces the number of instructions needed to accomplish a given task. Additionally, many instructions have been included to specifically improve code density and speed. For example, single word Add and Subtract instructions using Quick Immediate addressing allow fast, small value arithmetic operations on data registers and memory. A Load Quick Immediate (LDQ) provides the ability to load a small (8-bit) signed word into any register in a single word operation. In order to improve the speed of loop operations, a single word instruction for Decrement Count by One and Branch if non-zero (DCNT) is included. Of course, the TRAP, Store Multiple Registers (STM), Load Multiple Registers (LDM), Link Stack (LINK), Unlink Stack (UNLK) and Check Limit (CHK) instructions significantly reduce code requirements for subroutines, operating system calls and stacking operations.

Other instructions that help reduce coding requirements and improve performance of arithmetic operations are Signed and Unsigned Multiply (MULS and MULU), Signed and Unsigned Divide (DIVS and DIVU), BCD Arithmetic (ABCD, SBCD, PACK and UNPK) as well as the standard binary integer operations. In order to improve the efficiency of moving or transferring data, a powerful MOVE data instruction has been incorporated that allows the transfer of bytes, words and long words and operates in all data addressing modes. Thus; register-to-register, register-to-memory, memory-to-register and memory-to-memory transfers are permitted.

In addition to the powerful instructions that provide a substantial improvement in processor through-put, numerous architectural features significantly reduce the execution times for all instructions. The separate (non-multiplexed) address and data buses, instruction pre-fetch pipeline and 32-bit internal registers are major contributors to the processor's unequaled performance. As an example of the performance capability of the MC68000 Table III and the accompanying graphs in figures 5 and 6 summarize the execution times for a number of common instructions. For comparison purposes, similar information is provided for Zilog's Z-8000 microprocessor. It is interesting to note that the MC68000 has significantly faster execution times.

TABLE III — EXECUTION TIMES FOR *MOVB R, SRC* INSTRUCTION FOR VARIOUS ADDRESSING MODES

Source Addressing	Motorola MC68000	Zilog Z-8000
Register	0.5us	0.75us
Indirect Register	1.0	1.75
Absolute Addressing (Direct)	1.5	2.25
Indexed Addressing	1.5	2.50
Immediate	1.0	1.00

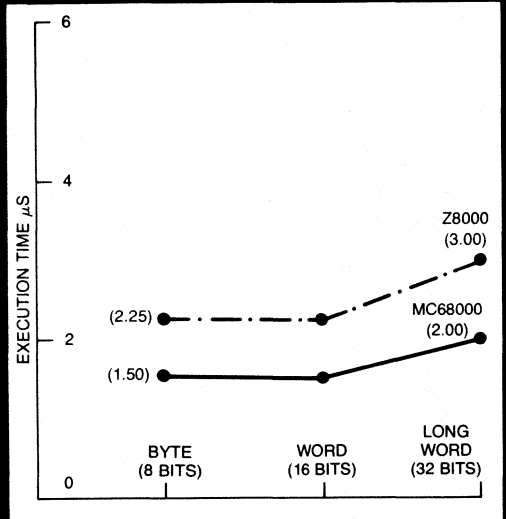


FIGURE 5: Execution Time for the *Add Data Element to a register from a short Absolute Address* Instruction.

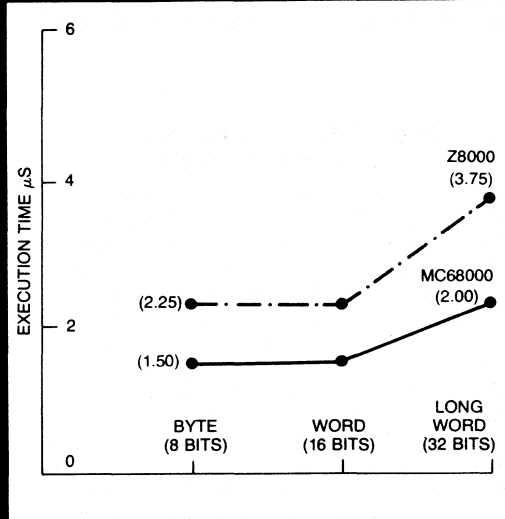


FIGURE 6: Execution Time for the *move a data element from memory to a register from short Absolute Address* Instruction.

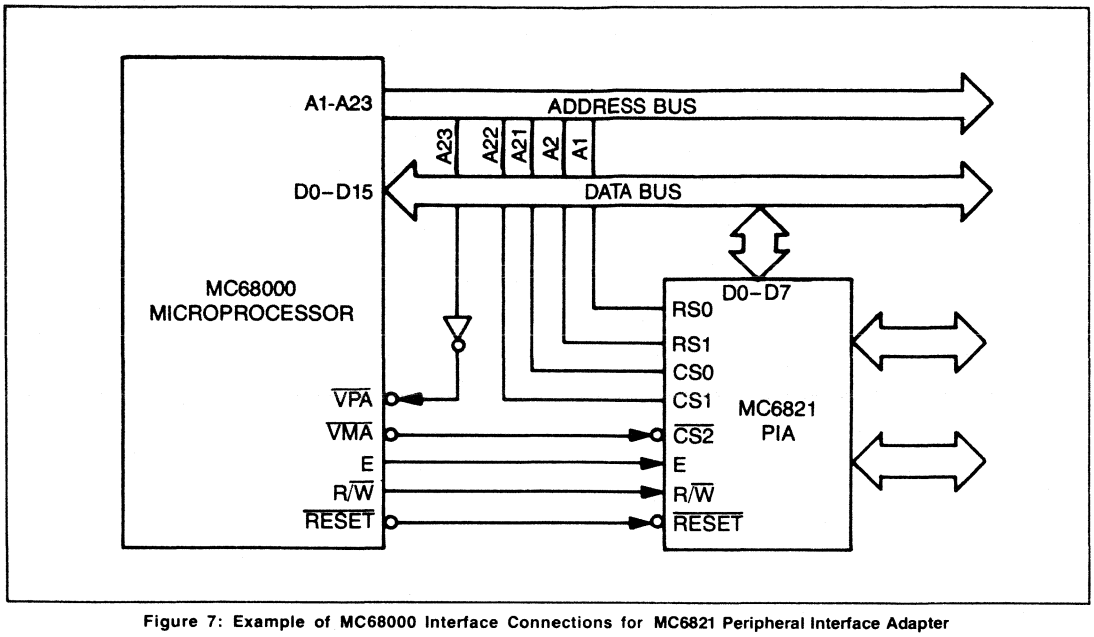


Figure 7: Example of MC68000 Interface Connections for MC6821 Peripheral Interface Adapter

SOFTWARE SUPPORT AND MC6800 COMPATIBILITY

The system designers and programmers using the MC68000 in an application have available a complete, compatible system of hardware and software. The microprocessor is supported by a full range of software development tools including disc operating systems, debug aids, assemblers, and high level languages. In addition, a translator will allow the present M6800 Family user to convert existing programs to run on the MC68000 with a minimum of programmer intervention.

The careful planning of this new microprocessor provides a superset of the MC6800 instruction set enhanced by the addition of more and larger registers, powerful orthogonal structure and many flexible addressing modes. This allows efficient translation of existing MC6800 programs, which can then be further optimized by taking full advantage of the versatile and powerful features of the MC68000.

This careful planning of similarities between the MC68000 and the MC6800 does not stop at software compatibility (by translation) but also extends to peripheral controller interfacing. Motorola's extensive line of intelligent M6800 family peripherals (including the MC6854 Advanced Data Link Controller and the MC68488 General Purpose Interface Adapter) can be directly and easily interfaced to the MC68000. Three signal lines; Enable (E), Valid Memory Address (VMA), and Valid Peripheral Address (VPA) are provided to simplify the interface to Motorola's standard MC6800 peripherals as shown in Figure 7. Interface to the new MC6801E (Single Chip Programmable Controller) is also possible, allowing user implementation of specialized input/output functions. In addition, the MC68000 is supported by unique peripheral controllers expected of an advanced architecture microprocessor, including a DMA Controller and a Memory Management Unit.

The MC68000 is not just a component. By a unique blend of VLSI design, software engineering and careful planning, the MC68000 *it is* Motorola's Advanced Computer System on Silicon.

chapter 1

the M6800 NMOS family

MOTOROLA

MC6800
(1.0 MHz)
MC68A00
(1.5 MHz)
MC68B00
(2.0 MHz)

MOS
(N-CHANNEL, SILICON-GATE, DEPLETION LOAD)

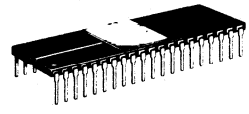
MICROPROCESSOR

8-BIT MICROPROCESSING UNIT (MPU)

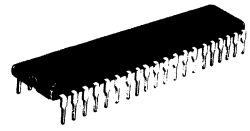
The MC6800 is a monolithic 8-bit microprocessor forming the central control function for Motorola's M6800 family. Compatible with TTL, the MC6800, as with all M6800 system parts, requires only one +5.0-volt power supply, and no external TTL devices for bus interface.

The MC6800 is capable of addressing 65K bytes of memory with its 16-bit address lines. The 8-bit data bus is bidirectional as well as 3-state, making direct memory addressing and multiprocessing applications realizable.

- Eight-Bit Parallel Processing
- Bidirectional Data Bus
- Sixteen-Bit Address Bus – 65K Bytes of Addressing
- 72 Instructions – Variable Length
- Seven Addressing Modes – Direct, Relative, Immediate, Indexed, Extended, Implied and Accumulator
- Variable Length Stack
- Vectored Restart
- Maskable Interrupt Vector
- Separate Non-Maskable Interrupt – Internal Registers Saved in Stack
- Six Internal Registers – Two Accumulators, Index Register, Program Counter, Stack Pointer and Condition Code Register
- Direct Memory Addressing (DMA) and Multiple Processor Capability
- Simplified Clocking Characteristics
- Clock Rates as High as 2.0 MHz
- Simple Bus Interface Without TTL
- Halt and Single Instruction Execution Capability

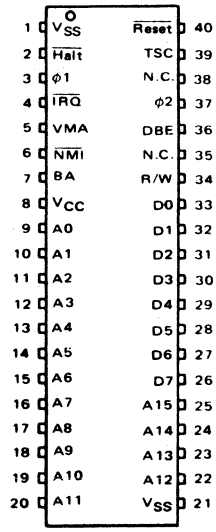


L SUFFIX
CERAMIC PACKAGE
CASE 715



P SUFFIX
PLASTIC PACKAGE
CASE 711

PIN ASSIGNMENT



ORDERING INFORMATION

Speed	Device	Temperature Range
1.0 MHz	MC6800P,L	0 to 70°C
	MC6800CP,CL	-40 to +85°C
MIL-STD-883B MIL-STD-883C	MC6800BQCS MC6800CQCS	-55 to +125°C
1.5 MHz	MC68A00P,L	0 to +70°C
	MC68A00CP,CL	-40 to +85°C
2.0 MHz	MC68B00P,L	0 to +70°C

TABLE 1 – MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	-0.3 to +7.0	Vdc
Input Voltage	V_{in}	-0.3 to +7.0	Vdc
Operating Temperature Range–T_L to T_H MC6800, MC68A00, MC68B00 MC6800C, MC68A00C MC6800BQCS, MC6800CQCS	T_A	 0 to +70 -40 to +85 -55 to +125	°C
Storage Temperature Range	T_{stg}	-55 to +150	°C
Thermal Resistance Plastic Package Ceramic Package	θ_{JA}	 70 50	°C/W

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit.

TABLE 2 – ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V, ± 5%, V_{SS} = 0, T_A = T_L to T_H unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage	Logic	V_{IH}	V_{SS} + 2.0	–	V_{CC}	Vdc
	$\phi1,\phi2$	V_{IHC}	V_{CC} - 0.6	–	V_{CC} + 0.3	
Input Low Voltage	Logic	V_{IL}	V_{SS} - 0.3	–	V_{SS} + 0.8	Vdc
	$\phi1,\phi2$	V_{ILC}	V_{SS} - 0.3	–	V_{SS} + 0.4	
Input Leakage Current		I_{in}				μAdc
(V_{in} = 0 to 5.25 V, V_{CC} = max)	Logic*		–	1.0	2.5	
(V_{in} = 0 to 5.25 V, V_{CC} = 0.0 V)	$\phi1,\phi2$		–	–	100	
Three-State (Off State) Input Current	D0-D7	I_{TSI}	–	2.0	10	μAdc
(V_{in} = 0.4 to 2.4 V, V_{CC} = max)	A0-A15,R/$\overline{W}$		–	–	100	
Output High Voltage		V_{OH}				Vdc
(I_{Load} = -205 μAdc, V_{CC} = min)	D0-D7		V_{SS} + 2.4	–	–	
(I_{Load} = -145 μAdc, V_{CC} = min)	A0-A15,R/$\overline{W}$,VMA		V_{SS} + 2.4	–	–	
(I_{Load} = -100 μAdc, V_{CC} = min)	BA		V_{SS} + 2.4	–	–	
Output Low Voltage (I_{Load} = 1.6 mAdc, V_{CC} = min)		V_{OL}	–	–	V_{SS} + 0.4	Vdc
Power Dissipation		P_D	–	0.5	1.0	W
Capacitance		C_{in}				pF
(V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	$\phi1$		–	25	35	
	$\phi2$		–	45	70	
	D0-D7		–	10	12.5	
	Logic Inputs		–	6.5	10	
	A0-A15,R/$\overline{W}$,VMA	C_{out}	–	–	12	pF

TABLE 3 – CLOCK TIMING (V_{CC} = 5.0 V, ± 5%, V_{SS} = 0, T_A = T_L to T_H unless otherwise noted)

Characteristics		Symbol	Min	Typ	Max	Unit
Frequency of Operation	MC6800	f	0.1	–	1.0	MHz
	MC68A00		0.1	–	1.5	
	MC68B00		0.1	–	2.0	
Cycle Time (Figure 1)	MC6800	t_{cyc}	1.000	–	10	μs
	MC68A00		0.666	–	10	
	MC68B00		0.500	–	10	
Clock Pulse Width	$\phi1,\phi2$ – MC6800	$PW_{\phi H}$	400	–	9500	ns
(Measured at V_{CC} - 0.6 V)	$\phi1,\phi2$ – MC68A00		230	–	9500	
	$\phi1,\phi2$ – MC68B00		180	–	9500	
Total $\phi1$ and $\phi2$ Up Time	MC6800	t_{ut}	900	–	–	ns
	MC68A00		600	–	–	
	MC68B00		440	–	–	
Rise and Fall Times		$t_{\phi r}$, $t_{\phi f}$	–	–	100	ns
(Measured between V_{SS} + 0.4 and V_{CC} - 0.6)						
Delay Time or Clock Separation (Figure 1)		t_d				ns
(Measured at V_{OV} = V_{SS} + 0.6 V @ t_r = $t_f \leq$ 100 ns)			0	–	9100	
(Measured at V_{OV} = V_{SS} + 1.0 V @ t_r = $t_f \leq$ 35 ns)			0	–	9100	

TABLE 4 – READ/WRITE TIMING (Reference Figures 2 through 8)

Characteristic	Symbol	MC6800			MC68A00			MC68B00			Unit
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Address Delay	t_{AD}										ns
C = 90 pF		–	–	270	–	–	180	–	–	150	
C = 30 pF		–	–	250	–	–	165	–	–	135	
Peripheral Read Access Time $t_{ac} = t_{ut} - (t_{AD} + t_{DSR})$	t_{acc}	–	–	530	–	–	360	–	–	250	ns
Data Setup Time (Read)	t_{DSR}	100	–	–	60	–	–	40	–	–	ns
Input Data Hold Time	t_H	10	–	–	10	–	–	10	–	–	ns
Output Data Hold Time	t_H	10	25	–	10	25	–	10	25	–	ns
Address Hold Time (Address, R/$\overline{W}$, VMA)	t_{AH}	30	50	–	30	50	–	30	50	–	ns
Enable High Time for DBE Input	t_{EH}	450	–	–	280	–	–	220	–	–	ns
Data Delay Time (Write)	t_{DDW}	–	–	225	–	–	200	–	–	160	ns
Processor Controls											
Processor Control Setup Time	t_{PCS}	200	–	–	140	–	–	110	–	–	ns
Processor Control Rise and Fall Time	t_{PCr}, t_{PCf}	–	–	100	–	–	100	–	–	100	ns
Bus Available Delay	t_{BA}	–	–	250	–	–	165	–	–	135	ns
Three-State Delay	t_{TSD}	–	–	270	–	–	270	–	–	220	ns
Data Bus Enable Down Time During ϕ1 Up Time	$t_{\overline{DBE}}$	150	–	–	120	–	–	75	–	–	ns
Data Bus Enable Rise and Fall Times	t_{DBEr}, t_{DBEf}	–	–	25	–	–	25	–	–	25	ns

FIGURE 1 – CLOCK TIMING WAVEFORM

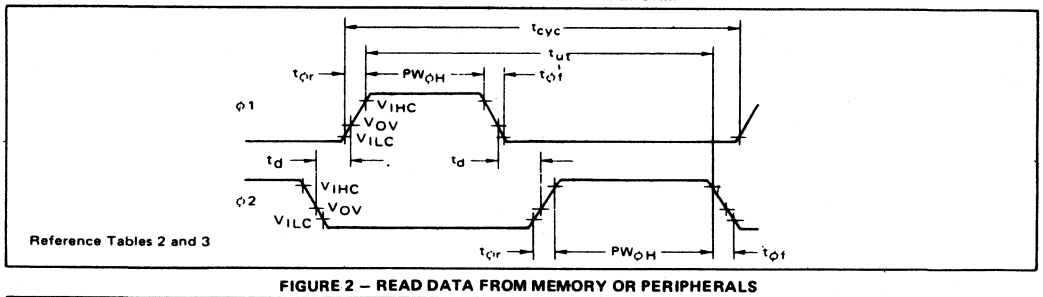


FIGURE 2 – READ DATA FROM MEMORY OR PERIPHERALS

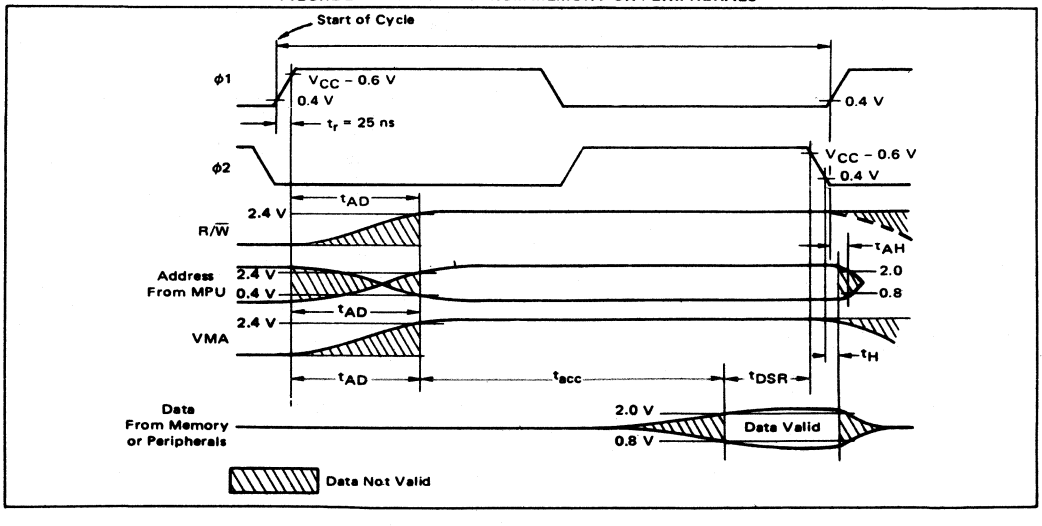


FIGURE 3 – WRITE IN MEMORY OR PERIPHERALS

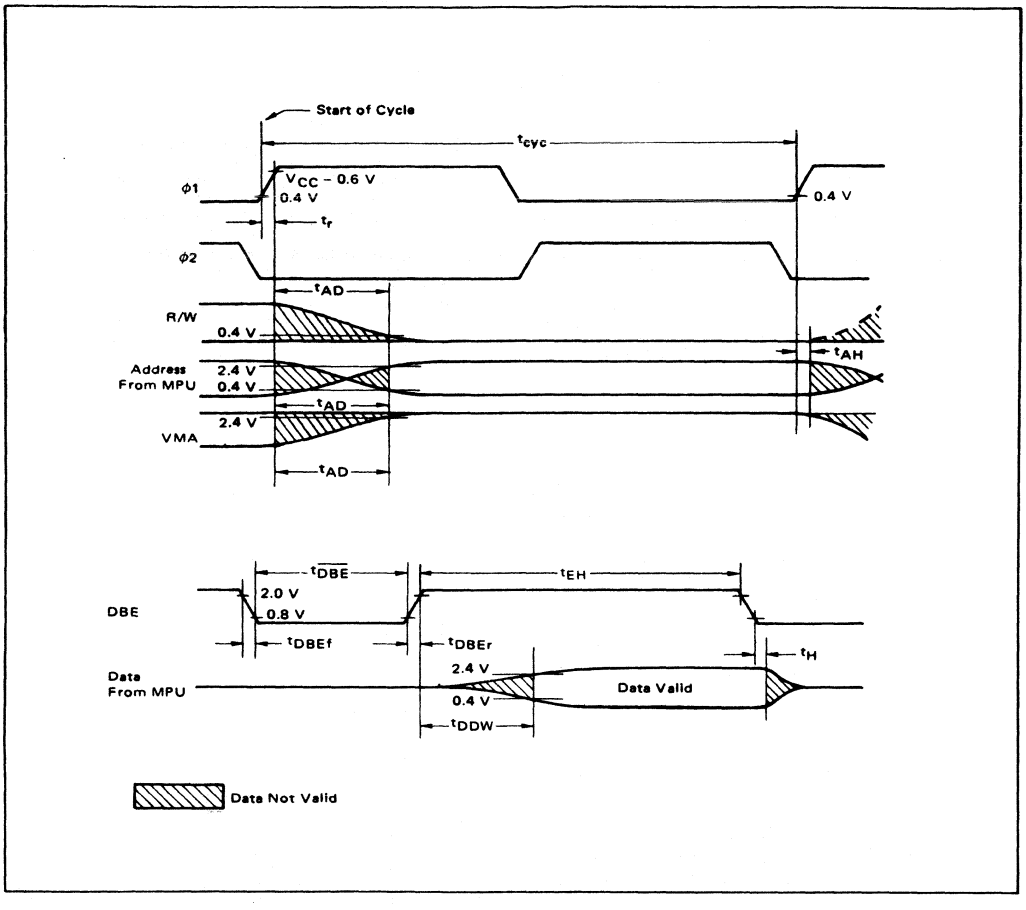


FIGURE 4 – TYPICAL DATA BUS OUTPUT DELAY versus CAPACITIVE LOADING (T_{DDW})

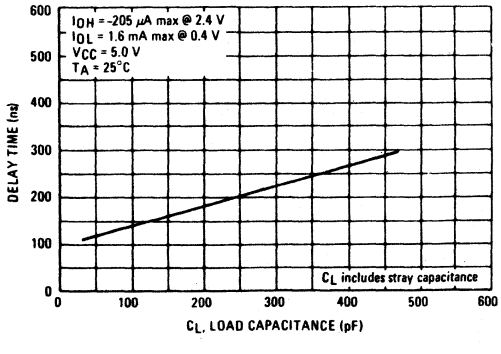


FIGURE 5 – TYPICAL READ/WRITE, VMA, AND ADDRESS OUTPUT DELAY versus CAPACITIVE LOADING (T_{AD})

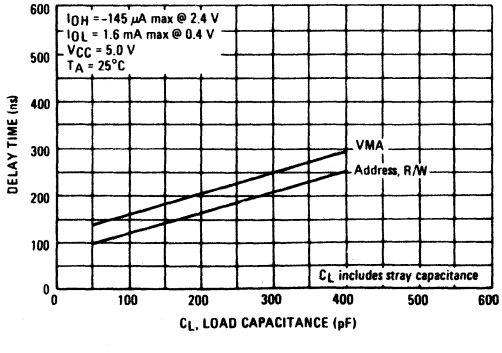
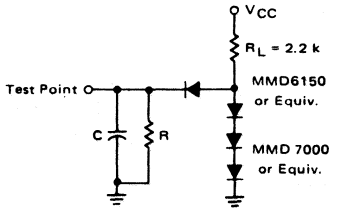


FIGURE 6 – BUS TIMING TEST LOADS



C = 130 pF for D0-D7, E
= 90 pF for A0-A15, R/W, and VMA
(Except t_{AD2})
= 30 pF for A0-A15, R/W, and VMA
(t_{AD2} only)
= 30 pF for BA

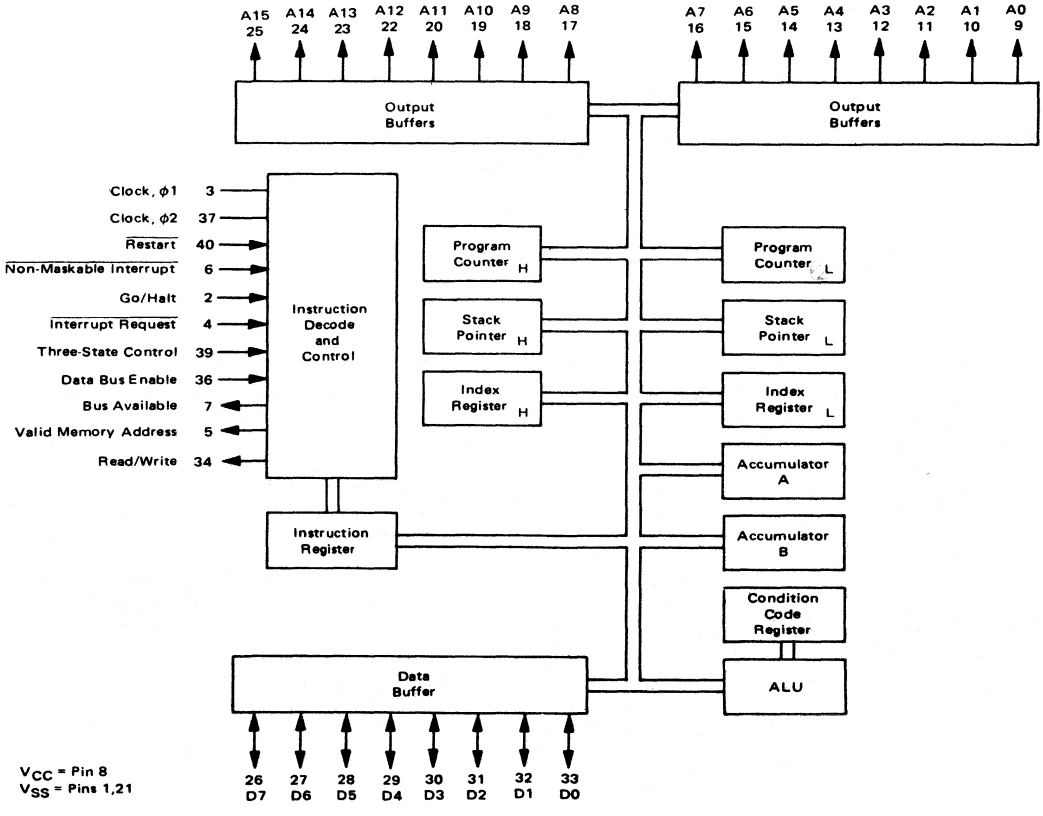
R = 11.7 kΩ for D0-D7
= 16.5 kΩ for A0-A15, R/W, and VMA
= 24 kΩ for BA

TEST CONDITIONS

The dynamic test load for the Data Bus is 130 pF and one standard TTL load as shown. The Address, R/W, and VMA outputs are tested under two conditions to allow optimum operation in both buffered and unbuffered systems. The resistor (R) is chosen to insure specified load currents during V_{OH} measurement.

Notice that the Data Bus lines, the Address lines, the Interrupt Request line, and the DBE line are all specified and tested to guarantee 0.4 V of dynamic noise immunity at both "1" and "0" logic levels.

FIGURE 7 – EXPANDED BLOCK DIAGRAM



MPU SIGNAL DESCRIPTION

Proper operation of the MPU requires that certain control and timing signals be provided to accomplish specific functions and that other signal lines be monitored to determine the state of the processor.

Clocks Phase One and Phase Two ($\phi1, \phi2$) – Two pins are used for a two-phase non-overlapping clock that runs at the V_{CC} voltage level.

Figure 1 shows the microprocessor clocks, and Table 3 shows the static and dynamic clock specifications. The high level is specified at V_{IHC} and the low level is specified at V_{ILC}. The allowable clock frequency is specified by f (frequency). The minimum $\phi1$ and $\phi2$ high level pulse widths are specified by $PW_{\phi H}$ (pulse width high time). To guarantee the required access time for the peripherals, the clock up time, t_{ut}, is specified. Clock separation, t_d, is measured at a maximum voltage of V_{OV} (overlap voltage). This allows for a multitude of clock variations at the system frequency rate.

Address Bus (A0-A15) – Sixteen pins are used for the address bus. The outputs are three-state bus drivers capable of driving one standard TTL load and 90 pF. When the output is turned off, it is essentially an open circuit. This permits the MPU to be used in DMA applications. Putting TSC in its high state forces the Address bus to go into the three-state mode.

Data Bus (D0-D7) – Eight pins are used for the data bus. It is bidirectional, transferring data to and from the memory and peripheral devices. It also has three-state output buffers capable of driving one standard TTL load and 130 pF. Data Bus is placed in the three-state mode when DBE is low.

Data Bus Enable (DBE) – This input is the three-state control signal for the MPU data bus and will enable the bus drivers when in the high state. This input is TTL compatible; however in normal operation, it would be driven by the phase two clock. During an MPU read cycle, the data bus drivers will be disabled internally. When it is desired that another device control the data bus such as in Direct Memory Access (DMA) applications, DBE should be held low.

If additional data setup or hold time is required on an MPU write, the DBE down time can be decreased as shown in Figure 3 (DBE $\neq \phi2$). The minimum down time for DBE is $t_{\overline{DBE}}$ as shown and must occur within $\phi1$ up time. The minimum delay from the trailing edge of DBE to the trailing edge of $\phi1$ is t_{DBED}. By skewing DBE with respect to E in this manner, data setup or hold time can be increased.

Bus Available (BA) – The Bus Available signal will normally be in the low state; when activated, it will go to the high state indicating that the microprocessor has stopped and that the address bus is available. This will occur if the $\overline{Halt}$ line is in the low state or the processor is in the WAIT state as a result of the execution of a WAIT instruction. At such time, all three-state output drivers will go to their off state and other outputs to their normally inactive level. The processor is removed from the WAIT state by the occurrence of a maskable (mask bit I = 0) or nonmaskable interrupt. This output is capable of driving one standard TTL load and 30 pF. If TSC is in the high state, Bus Available will be low.

Read/$\overline{Write}$ (R/$\overline{W}$) – This TTL compatible output signals the peripherals and memory devices whether the MPU is in a Read (high) or Write (low) state. The normal standby state of this signal is Read (high). Three-State Control going high will turn Read/$\overline{Write}$ to the off (high impedance) state. Also, when the processor is halted, it will be in the off state. This output is capable of driving one standard TTL load and 90 pF.

$\overline{Reset}$ – The $\overline{Reset}$ input is used to reset and start the MPU from a power down condition resulting from a power failure or initial start-up of the processor. This input can also be used to reinitialize the machine at any time after start-up.

If a high level is detected in this input, this will signal the MPU to begin the reset sequence. During the reset sequence, the contents of the last two locations (FFFE, FFFF) in memory will be loaded into the Program Counter to point to the beginning of the reset routine. During the reset routine, the interrupt mask bit is set and must be cleared under program control before the MPU can be interrupted by $\overline{IRQ}$. While $\overline{Reset}$ is low (assuming a minimum of 8 clock cycles have occurred) the MPU output signals will be in the following states: VMA = low, BA = low, Data Bus = high impedance, R/$\overline{W}$ = high (read state), and the Address Bus will contain the reset address FFFE. Figure 8 illustrates a power up sequence using the $\overline{Reset}$ control line. After the power supply reaches 4.75 V a minimum of eight clock cycles are required for the processor to stabilize in preparation for restarting. During these eight cycles, VMA will be in an indeterminate state so any devices that are enabled by VMA which could accept a false write during this time (such as a battery-backed RAM) must be disabled until VMA is forced low after eight cycles. $\overline{Reset}$ can go high asynchronously with the system clock any time after the eighth cycle.

$\overline{Reset}$ timing is shown in Figure 8 and Table 4. The maximum rise and fall transition times are specified by t_{PCr} and t_{PCf}. If $\overline{Reset}$ is high at t_{PCS} (processor control setup time) as shown in Figure 8 in any given cycle, then the restart sequence will begin on the next cycle as shown. The $\overline{Reset}$ control line may also be used to reinitialize the MPU system at any time during its operation. This is accomplished by pulsing $\overline{Reset}$ low for the duration of a minimum of three complete $\phi2$ cycles. The $\overline{Reset}$ pulse can be completely asynchronous with the MPU system clock and will be recognized during $\phi2$ if setup time t_{PCS} is met.

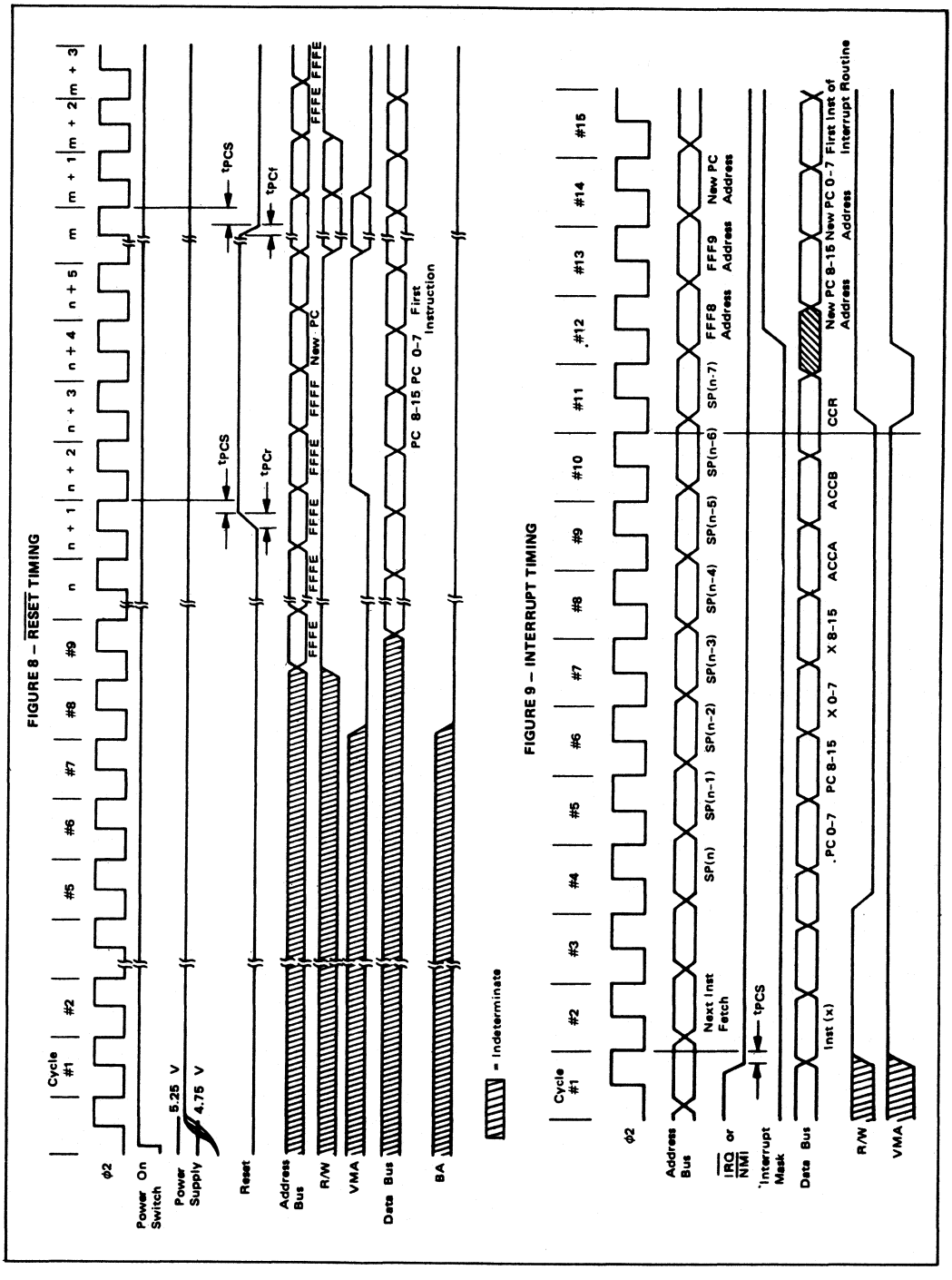


FIGURE 8 – RESET TIMING
Cycle #1
#2
#5
#6
#7
#8
#9
n
n + 1
n + 2
n + 3
n + 4
n + 5
m
m + 1
m + 2
m + 3
φ2
Power On
Power Switch
Power Supply
5.25 V
4.75 V
Reset
tPCS
tPCr
Address Bus
R/W
VMA
Data Bus
BA
FFFE
FFFF
New PC
PC 8-15
PC 0-7
First Instruction
= Indeterminate
FIGURE 9 – INTERRUPT TIMING
Cycle #1
#2
#3
#4
#5
#6
#7
#8
#9
#10
#11
#12
#13
#14
#15
φ2
Address Bus
Next Inst Fetch
SP(n)
SP(n-1)
SP(n-2)
SP(n-3)
SP(n-4)
SP(n-5)
SP(n-6)
SP(n-7)
FFF8 Address
FFF9 Address
New PC Address
IRQ or NMI
tPCS
Interrupt Mask
Data Bus
Inst (x)
PC 0-7
PC 8-15
X 0-7
X 8-15
ACCA
ACCB
CCR
New PC 8-15 Address
New PC 0-7 Address
First Inst of Interrupt Routine
R/W
VMA

Interrupt Request ($\overline{IRQ}$) – This level sensitive input requests that an interrupt sequence be generated within the machine. The processor will wait until it completes the current instruction that is being executed before it recognizes the request. At that time, if the interrupt mask bit in the Condition Code Register is not set, the machine will begin an interrupt sequence. The Index Register, Program Counter, Accumulators, and Condition Code Register are stored away on the stack. Next the MPU will respond to the interrupt request by setting the interrupt mask bit high so that no further interrupts may occur. At the end of the cycle, a 16-bit address will be loaded that points to a vectoring address which is located in memory locations FFF8 and FFF9. An address loaded at these locations causes the MPU to branch to an interrupt routine in memory. Interrupt timing is shown in Figure 9.

The $\overline{Halt}$ line must be in the high state for interrupts to be serviced. Interrupts will be latched internally while $\overline{Halt}$ is low.

The $\overline{IRQ}$ has a high impedance pullup device internal to the chip; however a 3 kΩ external resistor to V_{CC} should be used for wire-OR and optimum control of interrupts.

Non-Maskable Interrupt ($\overline{NMI}$) and Wait for Interrupt (WAI) – The MC6800 is capable of handling two types of interrupts: maskable ($\overline{IRQ}$) as described earlier, and non-maskable ($\overline{NMI}$). $\overline{IRQ}$ is maskable by the interrupt mask in the condition code register while $\overline{NMI}$ is not maskable. The handling of these interrupts by the MPU is the same except that each has its own vector address. The behavior of the MPU when interrupted is shown in Figure 9 which details the MPU response to an interrupt while the MPU is executing the control program. The interrupt shown could be either $\overline{IRQ}$ or $\overline{NMI}$ and can be asynchronous with respect to $\phi 2$. The interrupt is shown going low at time t_{PCS} in cycle #1 which precedes the first cycle of an instruction (OP code fetch). This instruction is not executed but instead the Program Counter (PC), Index Register (IX), Accumulators (ACCX), and the Condition Code Register (CCR) are pushed onto the stack.

The Interrupt Mask bit is set to prevent further interrupts. The address of the interrupt service routine is then fetched from FFFC, FFFD for an $\overline{NMI}$ interrupt and from FFF8, FFF9 for an $\overline{IRQ}$ interrupt. Upon completion of the interrupt service routine, the execution of RTI will pull the PC, IX, ACCX, and CCR off of the stack; the Interrupt Mask bit is restored to its condition prior to Interrupts.

Figure 11 is a similar interrupt sequence, except in this case, a WAIT instruction has been executed in preparation for the interrupt. This technique speeds up the MPU's response to the interrupt because the stacking of the PC, IX, ACCX, and the CCR is already done. While the MPU is waiting for the interrupt, Bus Available will go high indicating the following states of the control lines: VMA is low, and the Address Bus, R/$\overline{W}$ and Data Bus are all in the high impedance state. After the interrupt occurs, it is serviced as previously described.

TABLE 5 – MEMORY MAP FOR INTERRUPT VECTORS

Vector		
MS	LS	Description
FFFE	FFFF	Restart
FFFC	FFFD	Non-maskable Interrupt
FFFA	FFFB	Software Interrupt
FFF8	FFF9	Interrupt Request

Refer to Figure 11 for program flow for Interrupts.

Three State Control (TSC) – When the Three-State Control (TSC) line is a logic "1", the Address Bus and the R/$\overline{W}$ line are placed in a high impedance state. VMA and BA are forced low when TSC = "1" to prevent false reads or writes on any device enabled by VMA. It Is necessary to delay program execution while TSC is held high. This is done by insuring that no transitions of $\phi 1$ (or $\phi 2$) occur during this period. (Logic levels of the clocks are irrelevant so long as they do not change.) Since the MPU is a dynamic device, the $\phi 1$ clock can be stopped for a maximum time $PW_{\phi H}$ without destroying data within the MPU. TSC then can be used in a short Direct Memory Access (DMA) application.

Figure 12 shows the effect of TSC on the MPU. TSC must have its transitions at t_{TSE} (three-state enable) while holding $\phi 1$ high and $\phi 2$ low as shown. The Address Bus and R/$\overline{W}$ line will reach the high impedance state at t_{TSD} (three-state delay), with VMA being forced low. In this example, the Data Bus is also in the high impedance state while $\phi 2$ is being held low since DBE = $\phi 2$. At this point in time, a DMA transfer could occur on cycles #3 and #4. When TSC is returned low, the MPU Address and R/$\overline{W}$ lines return to the bus. Because it is too late in cycle #5 to access memory, this cycle is dead and used for synchronization. Program execution resumes in cycle #6.

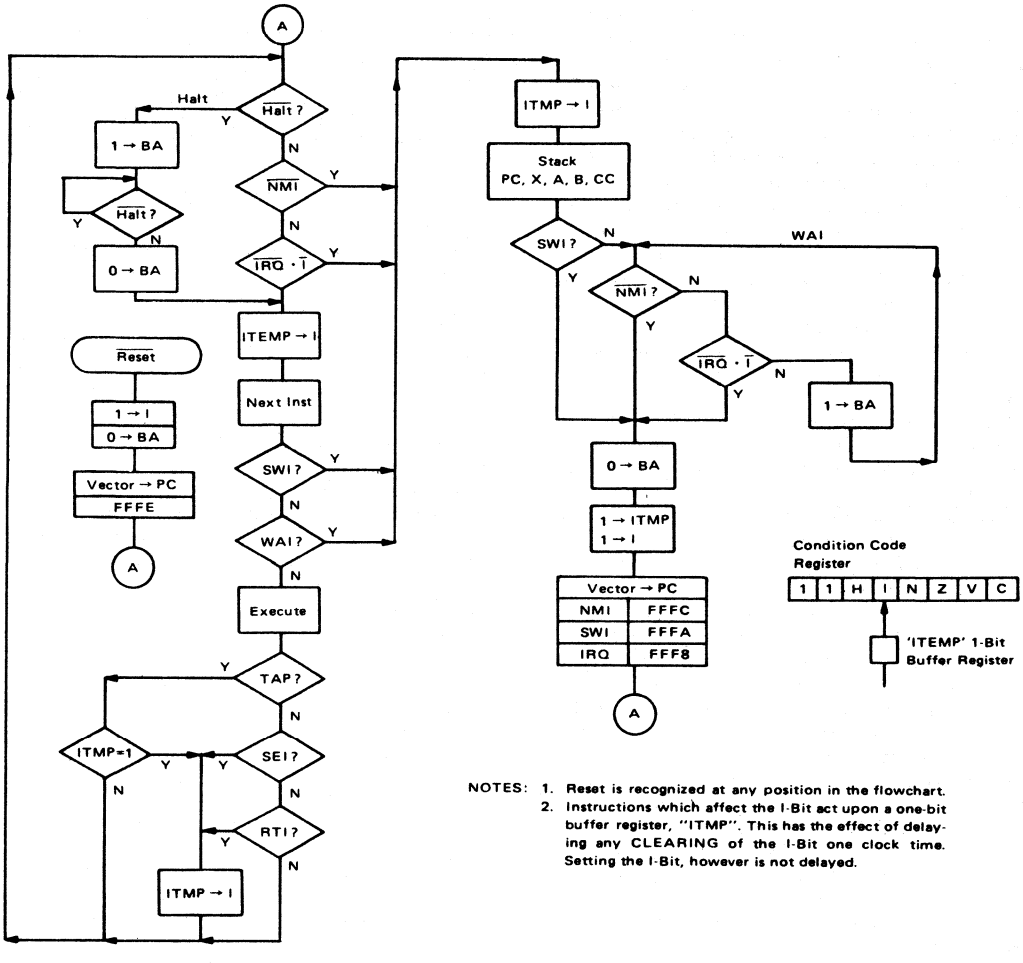
Valid Memory Address (VMA) – This output indicates to peripheral devices that there is a valid address on the address bus. In normal operation, this signal should be utilized for enabling peripheral interfaces such as the PIA and ACIA. This signal is not three-state. One standard TTL load and 90 pF may be directly driven by this active high signal.

$\overline{Halt}$ – When this input is in the low state, all activity in the machine will be halted. This input is level sensitive.

The $\overline{Halt}$ line provides an input to the MPU to allow control of program execution by an outside source. If $\overline{Halt}$ is high, the MPU will execute the instructions; if it is low, the MPU will go to a halted or idle mode. A response signal, Bus Available (BA) provides an indication of the current MPU status. When BA is low, the MPU is in the process of executing the control program; if BA is high, the MPU has halted and all internal activity has stopped.

When BA is high, the Address Bus, Data Bus, and R/$\overline{W}$ line will be in a high impedance state, effectively removing the MPU from the system bus. VMA is forced low so that the floating system bus will not activate any device on the bus that is enabled by VMA.

FIGURE 10 – MPU FLOW CHART



NOTES: 1. Reset is recognized at any position in the flowchart.
2. Instructions which affect the I-Bit act upon a one-bit buffer register, "ITMP". This has the effect of delaying any CLEARING of the I-Bit one clock time. Setting the I-Bit, however is not delayed.

While the MPU is halted, all program activity is stopped, and if either an $\overline{NMI}$ or $\overline{IRQ}$ interrupt occurs, it will be latched into the MPU and acted on as soon as the MPU is taken out of the halted mode. If a $\overline{Reset}$ command occurs while the MPU is halted, the following states occur: VMA = low, BA = low, Data Bus = high impedance, R/$\overline{W}$ = high (read state), and the Address Bus will contain address FFFE as long as $\overline{Reset}$ is low. As soon as the Halt line goes high, the MPU will go to locations FFFE and FFFF for the address of the reset routine.

Figure13 shows the timing relationships involved when halting the MPU. The instruction illustrated is a one byte, 2 cycle instruction such as CLRA. When $\overline{Halt}$ goes low, the MPU will halt after completing execution of the current instruction. The transition of $\overline{Halt}$ must occur t_{PCS} before the trailing edge of $\phi 1$ of the last cycle of an instruction (point A of Figure 13). $\overline{Halt}$ must not go low any time later than the minimum t_{PCS} specified.

The fetch of the OP code by the MPU is the first cycle of the instruction. If $\overline{Halt}$ had not been low at Point A but went low during $\phi 2$ of that cycle, the MPU would have halted after completion of the following instruction. BA will go high by time t_{BA} (bus available delay time) after the last instruction cycle. At this point in time, VMA is low and R/$\overline{W}$, Address Bus, and the Data Bus are in the high impedance state.

To debug programs it is advantageous to step through programs instruction by instruction. To do this, $\overline{Halt}$ must be brought high for one MPU cycle and then returned low as shown at point B of Figure 13. Again, the transitions of $\overline{Halt}$ must occur t_{PCS} before the trailing edge of $\phi 1$. BA will go low at t_{BA} after the leading edge of the next $\phi 1$, indicating that the Address Bus, Data Bus, VMA and R/$\overline{W}$ lines are back on the bus. A single byte, 2 cycle instruction such as LSR is used for this example also. During the first cycle, the instruction Y is fetched from address M + 1. BA returns high at t_{BA} on the last cycle of the instruction indicating the MPU is off the bus. If instruction Y had been three cycles, the width of the BA low time would have been increased by one cycle.

FIGURE 11 – WAIT INSTRUCTION TIMING

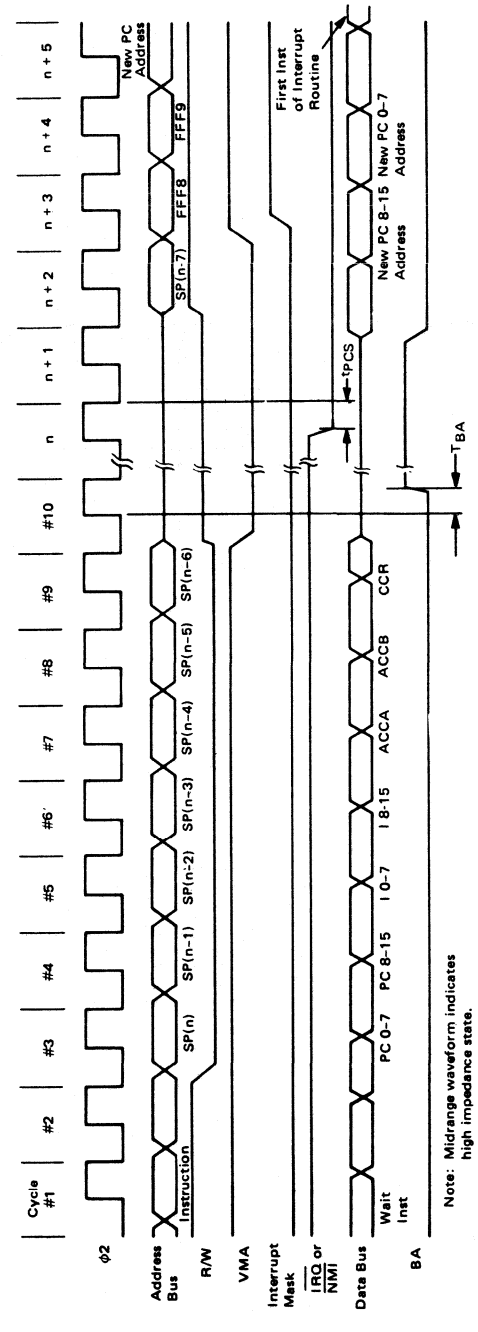


FIGURE 12 – THREE STATE CONTROL TIMING

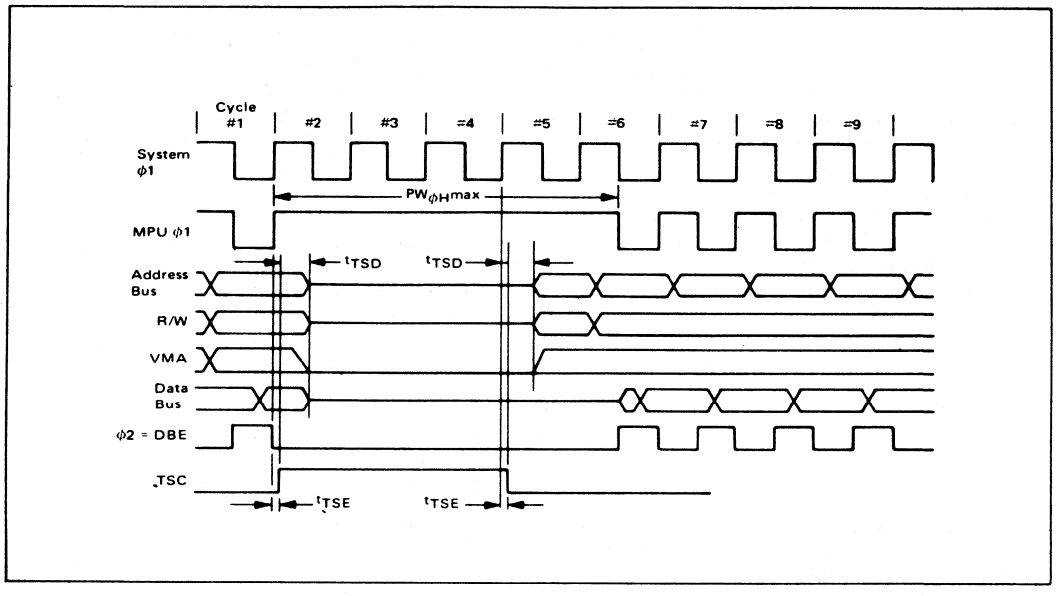
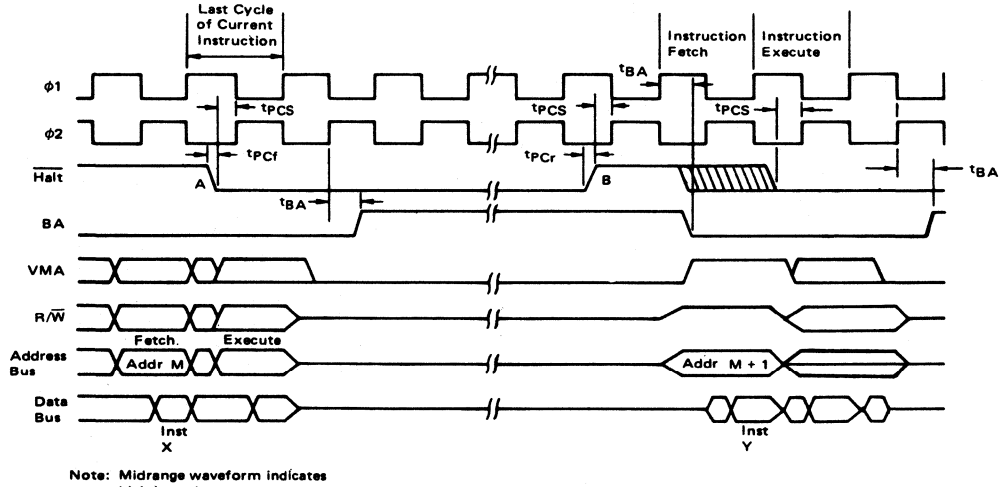


FIGURE 13 – $\overline{\text{HALT}}$ AND SINGLE INSTRUCTION EXECUTION FOR SYSTEM DEBUG



Note: Midrange waveform indicates high impedance state.

MPU REGISTERS

The MPU has three 16-bit registers and three 8-bit registers available for use by the programmer (Figure 14).

Program Counter – The program counter is a two byte (16 bits) register that points to the current program address.

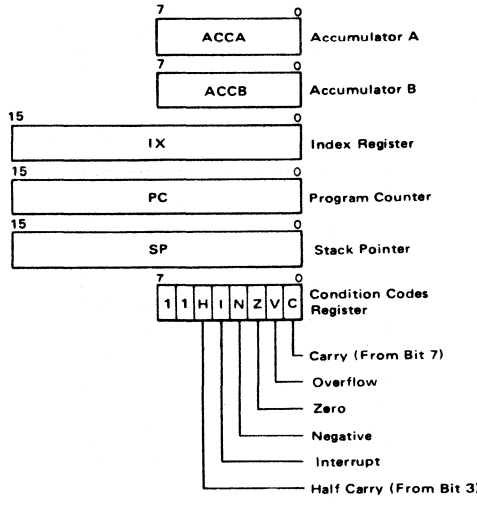
Stack Pointer – The stack pointer is a two byte register that contains the address of the next available location in an external push-down/pop-up stack. This stack is normally a random access Read/Write memory that may have any location (address) that is convenient. In those applications that require storage of information in the stack when power is lost, the stack must be nonvolatile.

Index Register – The index register is a two byte register that is used to store data or a sixteen bit memory address for the Indexed mode of memory addressing.

Accumulators – The MPU contains two 8-bit accumulators that are used to hold operands and results from an arithmetic logic unit (ALU).

Condition Code Register – The condition code register indicates the results of an Arithmetic Logic Unit operation: Negative (N), Zero (Z), Overflow (V), Carry from bit 7 (C), and half carry from bit 3 (H). These bits of the Condition Code Register are used as testable conditions for the conditional branch instructions. Bit 4 is the interrupt mask bit (I). The unused bits of the Condition Code Register (b6 and b7) are ones.

FIGURE 14 – PROGRAMMING MODEL OF THE MICROPROCESSING UNIT



MPU INSTRUCTION SET

The MC6800 instructions are described in detail in the M6800 Programming Manual. This Section will provide a brief introduction and discuss their use in developing MC6800 control programs. The MC6800 has a set of 72 different executable source instructions. Included are binary and decimal arithmetic, logical, shift, rotate, load, store, conditional or unconditional branch, interrupt and stack manipulation instructions.

Each of the 72 executable instructions of the source language assembles into 1 to 3 bytes of machine code. The number of bytes depends on the particular instruction and on the addressing mode. (The addressing modes which are available for use with the various executive instructions are discussed later.)

The coding of the first (or only) byte corresponding to an executable instruction is sufficient to identify the instruction and the addressing mode. The hexadecimal equivalents of the binary codes, which result from the translation of the 72 instructions in all valid modes of addressing, are shown in Table 6. There are 197 valid machine codes, 59 of the 256 possible codes being unassigned.

When an instruction translates into two or three bytes of code, the second byte, or the second and third bytes contain(s) an operand, an address, or information from which an address is obtained during execution.

Microprocessor instructions are often divided into three general classifications: (1) memory reference, so called because they operate on specific memory locations; (2) operating instructions that function without needing a memory reference; (3) I/O instructions for transferring data between the microprocessor and peripheral devices.

In many instances, the MC6800 performs the same operation on both its internal accumulators and the external memory locations. In additon, the MC6800 interfaces adapters (PIA and ACIA) allow the MPU to treat peripheral devices exactly like other memory locations, hence, no I/O instructions as such are required. Because of these features, other classifications are more suitable for introducing the MC6800's instruction set: (1) Accumulator and memory operations; (2) Program control operations; (3) Condition Code Register operations.

TABLE 6 – HEXADECIMAL VALUES OF MACHINE CODES

00	*		40	NEG	A		80	SUB	A	IMM	C0	SUB	B	IMM
01	NOP		41	*			81	CMP	A	IMM	C1	CMP	B	IMM
02	*		42	*			82	SBC	A	IMM	C2	SBC	B	IMM
03	*		43	COM	A		83	*			C3	*		
04	*		44	LSR	A		84	AND	A	IMM	C4	AND	B	IMM
05	*		45	*			85	BIT	A	IMM	C5	BIT	B	IMM
06	TAP		46	ROR	A		86	LDA	A	IMM	C6	LDA	B	IMM
07	TPA		47	ASR	A		87	*			C7	*		
08	INX		48	ASL	A		88	EOR	A	IMM	C8	EOR	B	IMM
09	DEX		49	ROL	A		89	ADC	A	IMM	C9	ADC	B	IMM
0A	CLV		4A	DEC	A		8A	ORA	A	IMM	CA	ORA	B	IMM
0B	SEV		4B	*			8B	ADD	A	IMM	CB	ADD	B	IMM
0C	CLC		4C	INC	A		8C	CPX	A	IMM	CC	*		
0D	SEC		4D	TST	A		8D	BSR		REL	CD	*		
0E	CLI		4E	*			8E	LDS		IMM	CE	LDX		IMM
0F	SEI		4F	CLR	A		8F	*			CF	*		
10	SBA		50	NEG	B		90	SUB	A	DIR	D0	SUB	B	DIR
11	CBA		51	*			91	CMP	A	DIR	D1	CMP	B	DIR
12	*		52	*			92	SBC	A	DIR	D2	SBC	B	DIR
13	*		53	COM	B		93	*			D3	*		
14	*		54	LSR	B		94	AND	A	DIR	D4	AND	B	DIR
15	*		55	*			95	BIT	A	DIR	D5	BIT	B	DIR
16	TAB		56	ROR	B		96	LDA	A	DIR	D6	LDA	B	DIR
17	TBA		57	ASR	B		97	STA	A	DIR	D7	STA	B	DIR
18	*		58	ASL	B		98	EOR	A	DIR	D8	EOR	B	DIR
19	DAA		59	ROL	B		99	ADC	A	DIR	D9	ADC	B	DIR
1A	*		5A	DEC	B		9A	ORA	A	DIR	DA	ORA	B	DIR
1B	ABA		5B	*			9B	ADD	A	DIR	DB	ADD	B	DIR
1C	*		5C	INC	B		9C	CPX		DIR	DC	*		
1D	*		5D	TST	B		9D	*			DD	*		
1E	*		5E	*			9E	LDS		DIR	DE	LDX		DIR
1F	*		5F	CLR	B		9F	STS		DIR	DF	STX		DIR
20	BRA	REL	60	NEG		IND	A0	SUB	A	IND	E0	SUB	B	IND
21	*		61	*			A1	CMP	A	IND	E1	CMP	B	IND
22	BHI	REL	62	*			A2	SBC	A	IND	E2	SBC	B	IND
23	BLS	REL	63	COM		IND	A3	*			E3	*		
24	BCC	REL	64	LSR		IND	A4	AND	A	IND	E4	AND	B	IND
25	BCS	REL	65	*			A5	BIT	A	IND	E5	BIT	B	IND
26	BNE	REL	66	ROR		IND	A6	LDA	A	IND	E6	LDA	B	IND
27	BEQ	REL	67	ASR		IND	A7	STA	A	IND	E7	STA	B	IND
28	BVC	REL	68	ASL		IND	A8	EOR	A	IND	E8	EOR	B	IND
29	BVS	REL	69	ROL		IND	A9	ADC	A	IND	E9	ADC	B	IND
2A	BPL	REL	6A	DEC		IND	AA	ORA	A	IND	EA	ORA	B	IND
2B	BMI	REL	6B	*			AB	ADD	A	IND	EB	ADD	B	IND
2C	BGE	REL	6C	INC		IND	AC	CPX		IND	EC	*		
2D	BLT	REL	6D	TST		IND	AD	JSR		IND	ED	*		
2E	BGT	REL	6E	JMP		IND	AE	LDS		IND	EE	LDX		IND
2F	BLE	REL	6F	CLR		IND	AF	STS		IND	EF	STX		IND
30	TSX		70	NEG		EXT	B0	SUB	A	EXT	F0	SUB	B	EXT
31	INS		71	*			B1	CMP	A	EXT	F1	CMP	B	EXT
32	PUL	A	72	*			B2	SBC	A	EXT	F2	SBC	B	EXT
33	PUL	B	73	COM		EXT	B3	*			F3	*		
34	DES		74	LSR		EXT	B4	AND	A	EXT	F4	AND	B	EXT
35	TXS		75	*			B5	BIT	A	EXT	F5	BIT	B	EXT
36	PSH	A	76	ROR		EXT	B6	LDA	A	EXT	F6	LDA	B	EXT
37	PSH	B	77	ASR		EXT	B7	STA	A	EXT	F7	STA	B	EXT
38	*		78	ASL		EXT	B8	EOR	A	EXT	F8	ADC	B	EXT
39	RTS		79	ROL		EXT	B9	ADC	A	EXT	F9	ADC	B	EXT
3A	*		7A	DEC		EXT	BA	ORA	A	EXT	FA	ORA	B	EXT
3B	RTI		7B	*			BB	ADD	A	EXT	FB	ADD	B	EXT
3C	*		7C	INC		EXT	BC	CPX		EXT	FC	*		
3D	*		7D	TST		EXT	BD	JSR		EXT	FD	*		
3E	WAI		7E	JMP		EXT	BE	LDS		EXT	FE	LDX		EXT
3F	SWI		7F	CLR		EXT	BF	STS		EXT	FF	STX		EXT

Notes: 1. Addressing Modes: A = Accumulator A, B = Accumulator B, REL = Relative, IND = Indexed, IMM = Immediate, DIR = Direct

2. Unassigned code indicated by "*"

TABLE 7 – ACCUMULATOR AND MEMORY OPERATIONS

The accumulator and memory operations and their effect on the CCR are shown in Table 7.
Included are Arithmetic Logic, Data Test and Data Handling instructions.

OPERATIONS	MNEMONIC	IMMED OP	IMMED ~	IMMED #	DIRECT OP	DIRECT ~	DIRECT #	INDEX OP	INDEX ~	INDEX #	EXTND OP	EXTND ~	EXTND #	IMPLIED OP	IMPLIED ~	IMPLIED #	BOOLEAN/ARITHMETIC OPERATION (All register labels refer to contents)	5 H	4 I	3 N	2 Z	1 V	0 C
Add	ADDA	8B	2	2	9B	3	2	AB	5	2	BB	4	3				A + M → A	↕	•	↕	↕	↕	↕
	ADDB	CB	2	2	DB	3	2	EB	5	2	FB	4	3				B + M → B	↕	•	↕	↕	↕	↕
Add Acmltrs	ABA													1B	2	1	A + B → A	↕	•	↕	↕	↕	↕
Add with Carry	ADCA	89	2	2	99	3	2	A9	5	2	B9	4	3				A + M + C → A	↕	•	↕	↕	↕	↕
	ADCB	C9	2	2	D9	3	2	E9	5	2	F9	4	3				B + M + C → B	↕	•	↕	↕	↕	↕
And	ANDA	84	2	2	94	3	2	A4	5	2	B4	4	3				A · M → A	•	•	↕	↕	R	•
	ANDB	C4	2	2	D4	3	2	E4	5	2	F4	4	3				B · M → B	•	•	↕	↕	R	•
Bit Test	BITA	85	2	2	95	3	2	A5	5	2	B5	4	3				A · M	•	•	↕	↕	R	•
	BITB	C5	2	2	D5	3	2	E5	5	2	F5	4	3				B · M	•	•	↕	↕	R	•
Clear	CLR							6F	7	2	7F	6	3				00 → M	•	•	R	S	R	R
	CLRA													4F	2	1	00 → A	•	•	R	S	R	R
	CLRB													5F	2	1	00 → B	•	•	R	S	R	R
Compare	CMPA	81	2	2	91	3	2	A1	5	2	B1	4	3				A − M	•	•	↕	↕	↕	↕
	CMPB	C1	2	2	D1	3	2	E1	5	2	F1	4	3				B − M	•	•	↕	↕	↕	↕
Compare Acmltrs	CBA													11	2	1	A − B	•	•	↕	↕	↕	↕
Complement, 1's	COM							63	7	2	73	6	3				M → M	•	•	↕	↕	R	S
	COMA													43	2	1	A → A	•	•	↕	↕	R	S
	COMB													53	2	1	B → B	•	•	↕	↕	R	S
Complement, 2's (Negate)	NEG							60	7	2	70	6	3				00 − M → M	•	•	↕	↕	①	②
	NEGA													40	2	1	00 − A → A	•	•	↕	↕	①	②
	NEGB													50	2	1	00 − B → B	•	•	↕	↕	①	②
Decimal Adjust, A	DAA													19	2	1	Converts Binary Add. of BCD Characters into BCD Format	•	•	↕	↕	↕	③
Decrement	DEC							6A	7	2	7A	6	3				M − 1 → M	•	•	↕	↕	4	•
	DECA													4A	2	1	A − 1 → A	•	•	↕	↕	4	•
	DECB													5A	2	1	B − 1 → B	•	•	↕	↕	4	•
Exclusive OR	EORA	88	2	2	98	3	2	A8	5	2	B8	4	3				A⊕M → A	•	•	↕	↕	R	•
	EORB	C8	2	2	D8	3	2	E8	5	2	F8	4	3				B⊕M → B	•	•	↕	↕	R	•
Increment	INC							6C	7	2	7C	6	3				M + 1 → M	•	•	↕	↕	⑤	•
	INCA													4C	2	1	A + 1 → A	•	•	↕	↕	⑤	•
	INCB													5C	2	1	B + 1 → B	•	•	↕	↕	⑤	•
Load Acmltr	LDAA	86	2	2	96	3	2	A6	5	2	B6	4	3				M → A	•	•	↕	↕	R	•
	LDAB	C6	2	2	D6	3	2	E6	5	2	F6	4	3				M → B	•	•	↕	↕	R	•
Or, Inclusive	ORAA	8A	2	2	9A	3	2	AA	5	2	BA	4	3				A + M → A	•	•	↕	↕	R	•
	ORAB	CA	2	2	DA	3	2	EA	5	2	FA	4	3				B + M → B	•	•	↕	↕	R	•
Push Data	PSHA													36	4	1	A → M_{SP}, SP − 1 → SP	•	•	•	•	•	•
	PSHB													37	4	1	B → M_{SP}, SP − 1 → SP	•	•	•	•	•	•
Pull Data	PULA													32	4	1	SP + 1 → SP, M_{SP} → A	•	•	•	•	•	•
	PULB													33	4	1	SP + 1 → SP, M_{SP} → B	•	•	•	•	•	•
Rotate Left	ROL							69	7	2	79	6	3				M, A, B ← C b7 b0	•	•	↕	↕	⑥	↕
	ROLA													49	2	1		•	•	↕	↕	⑥	↕
	ROLB													59	2	1		•	•	↕	↕	⑥	↕
Rotate Right	ROR							66	7	2	76	6	3				M, A, B → C b7 b0	•	•	↕	↕	⑥	↕
	RORA													46	2	1		•	•	↕	↕	⑥	↕
	RORB													56	2	1		•	•	↕	↕	⑥	↕
Shift Left, Arithmetic	ASL							68	7	2	78	6	3				M, A, B ← 0 C b7 b0	•	•	↕	↕	⑥	↕
	ASLA													48	2	1		•	•	↕	↕	⑥	↕
	ASLB													58	2	1		•	•	↕	↕	⑥	↕
Shift Right, Arithmetic	ASR							67	7	2	77	6	3				M, A, B → b7 b0 C	•	•	↕	↕	⑥	↕
	ASRA													47	2	1		•	•	↕	↕	⑥	↕
	ASRB													57	2	1		•	•	↕	↕	⑥	↕
Shift Right, Logic	LSR							64	7	2	74	6	3				M, A, B → 0 b7 b0 C	•	•	R	↕	⑥	↕
	LSRA													44	2	1		•	•	R	↕	⑥	↕
	LSRB													54	2	1		•	•	R	↕	⑥	↕
Store Acmltr.	STAA				97	4	2	A7	6	2	B7	5	3				A → M	•	•	↕	↕	R	•
	STAB				D7	4	2	E7	6	2	F7	5	3				B → M	•	•	↕	↕	R	•
Subtract	SUBA	80	2	2	90	3	2	A0	5	2	B0	4	3				A − M → A	•	•	↕	↕	↕	↕
	SUBB	C0	2	2	D0	3	2	E0	5	2	F0	4	3				B − M → B	•	•	↕	↕	↕	↕
Subtract Acmltrs.	SBA													10	2	1	A − B → A	•	•	↕	↕	↕	↕
Subtr. with Carry	SBCA	82	2	2	92	3	2	A2	5	2	B2	4	3				A − M − C → A	•	•	↕	↕	↕	↕
	SBCB	C2	2	2	D2	3	2	E2	5	2	F2	4	3				B − M − C → B	•	•	↕	↕	↕	↕
Transfer Acmltrs	TAB													16	2	1	A → B	•	•	↕	↕	R	•
	TBA													17	2	1	B → A	•	•	↕	↕	R	•
Test, Zero or Minus	TST							6D	7	2	7D	6	3				M − 00	•	•	↕	↕	R	R
	TSTA													4D	2	1	A − 00	•	•	↕	↕	R	R
	TSTB													5D	2	1	B − 00	•	•	↕	↕	R	R
																		H	I	N	Z	V	C

LEGEND:

- OP Operation Code (Hexadecimal);
- ~ Number of MPU Cycles;
- # Number of Program Bytes;
- + Arithmetic Plus;
- − Arithmetic Minus;
- • Boolean AND;
- M_{SP} Contents of memory location pointed to be Stack Pointer;
- + Boolean Inclusive OR;
- ⊕ Boolean Exclusive OR;
- M Complement of M;
- → Transfer Into;
- 0 Bit = Zero;
- 00 Byte = Zero;

Note – Accumulator addressing mode instructions are included in the column for IMPLIED addressing

CONDITION CODE SYMBOLS:

- H Half-carry from bit 3;
- I Interrupt mask
- N Negative (sign bit)
- Z Zero (byte)
- V Overflow, 2's complement
- C Carry from bit 7
- R Reset Always
- S Set Always
- ↕ Test and set if true, cleared otherwise
- • Not Affected

TABLE 7 – CONTINUED

CONDITION CODE REGISTER NOTES: (Bit set if test is true and cleared otherwise)

1 (Bit V) Test: Result = 10000000?
2 (Bit C) Test: Result = 00000000?
3 (Bit C) Test: Decimal value of most significant BCD Character greater than nine? (Not cleared if previously set.)
4 (Bit V) Test: Operand = 10000000 prior to execution?
5 (Bit V) Test: Operand = 01111111 prior to execution?
6 (Bit V) Test: Set equal to result of N⊕C after shift has occurred.

PROGRAM CONTROL OPERATIONS

Program Control operation can be subdivided into two categories; (1) Index Register/Stack Pointer instructions; (2) Jump and Branch operations.

Index Register/Stack Pointer Operations

The instructions for direct operation on the MPU's Index Register and Stack Pointer are summarized in Table 8. Decrement (DEX, DES), increment (INX, INS), load (LDX, LDS), and store (STX, STS) instructions are provided for both. The Compare instruction, CPX, can be used to compare the Index Register to a 16-bit value and update the Condition Code Register accordingly.

The TSX instruction causes the Index Register to be loaded with the address of the last data byte put onto the "stack". The TXS instruction loads the Stack Pointer with a value equal to one less than the current contents of the Index Register. This causes the next byte to be pulled from the "stack" to come from the location indicated by the Index Register. The utility of these two instructions can be clarified by describing the "stack" concept relative to the M6800 system.

The "stack" can be thought of as a sequential list of data stored in the MPU's read/write memory. The Stack Pointer contains a 16-bit memory address that is used to access the list from one end on a last-in-first-out (LIFO) basis in contrast to the random access mode used by the MPU's other addressing modes.

The M6800 instruction set and interrupt structure allow extensive use of the stack concept for efficient handling of data movement, subroutines and interrupts. The instructions can be used to establish one or more "stacks" anywhere in read/write memory. Stack length is limited only by the amount of memory that is made available.

TABLE 8 – INDEX REGISTER AND STACK POINTER INSTRUCTIONS

		IMMED			DIRECT			INDEX			EXTND			IMPLIED				COND. CODE REG.					
																		5	4	3	2	1	0
POINTER OPERATIONS	MNEMONIC	OP	~	=	OP	~	=	OP	~	=	OP	~	=	OP	~	=	BOOLEAN/ARITHMETIC OPERATION	H	I	N	Z	V	C
Compare Index Reg	CPX	8C	3	3	9C	4	2	AC	6	2	BC	5	3				$X_H - M, X_L - (M+1)$	•	•	①	↕	②	•
Decrement Index Reg	DEX													09	4	1	$X - 1 \rightarrow X$	•	•	•	↕	•	•
Decrement Stack Pntr	DES													34	4	1	$SP - 1 \rightarrow SP$	•	•	•	•	•	•
Increment Index Reg	INX													08	4	1	$X + 1 \rightarrow X$	•	•	•	↕	•	•
Increment Stack Pntr	INS													31	4	1	$SP + 1 \rightarrow SP$	•	•	•	•	•	•
Load Index Reg	LDX	CE	3	3	DE	4	2	EE	6	2	FE	5	3				$M \rightarrow X_H, (M+1) \rightarrow X_L$	•	•	③	↕	R	•
Load Stack Pntr	LDS	8E	3	3	9E	4	2	AE	6	2	BE	5	3				$M \rightarrow SP_H, (M+1) \rightarrow SP_L$	•	•	③	↕	R	•
Store Index Reg	STX				DF	5	2	EF	7	2	FF	6	3				$X_H \rightarrow M, X_L \rightarrow (M+1)$	•	•	③	↕	R	•
Store Stack Pntr	STS				9F	5	2	AF	7	2	BF	6	3				$SP_H \rightarrow M, SP_L \rightarrow (M+1)$	•	•	③	↕	R	•
Indx Reg → Stack Pntr	TXS													35	4	1	$X - 1 \rightarrow SP$	•	•	•	•	•	•
Stack Pntr → Indx Reg	TSX													30	4	1	$SP + 1 \rightarrow X$	•	•	•	•	•	•

① (Bit N) Test: Sign bit of most significant (MS) byte of result = 1?
② (Bit V) Test: 2's complement overflow from subtraction of ms bytes?
③ (Bit N) Test: Result less than zero? (Bit 15 = 1)

FIGURE 15 – STACK OPERATION, PUSH INSTRUCTION

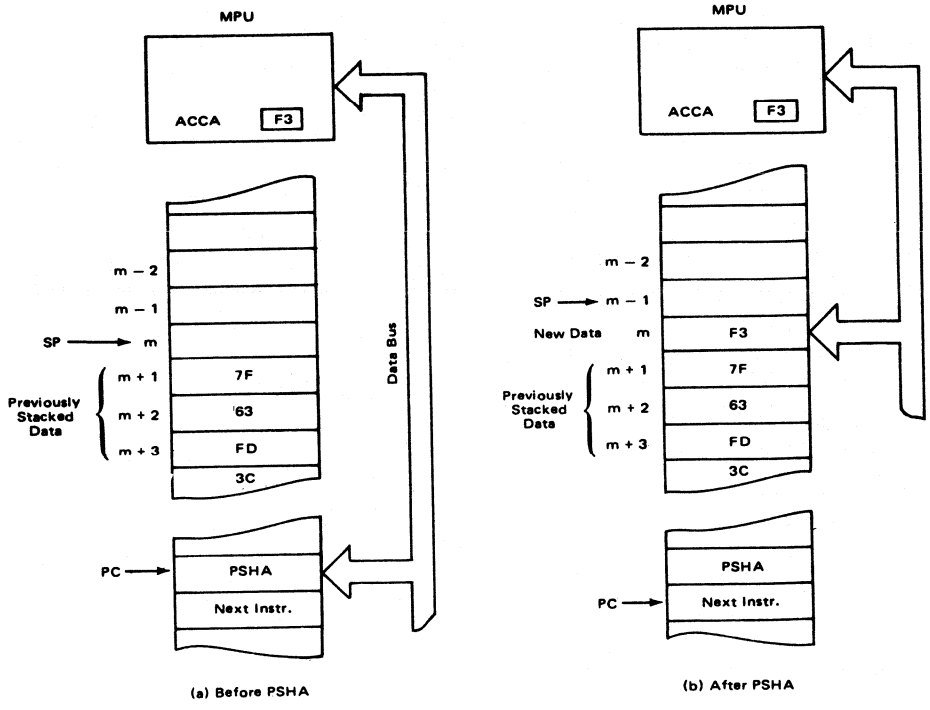
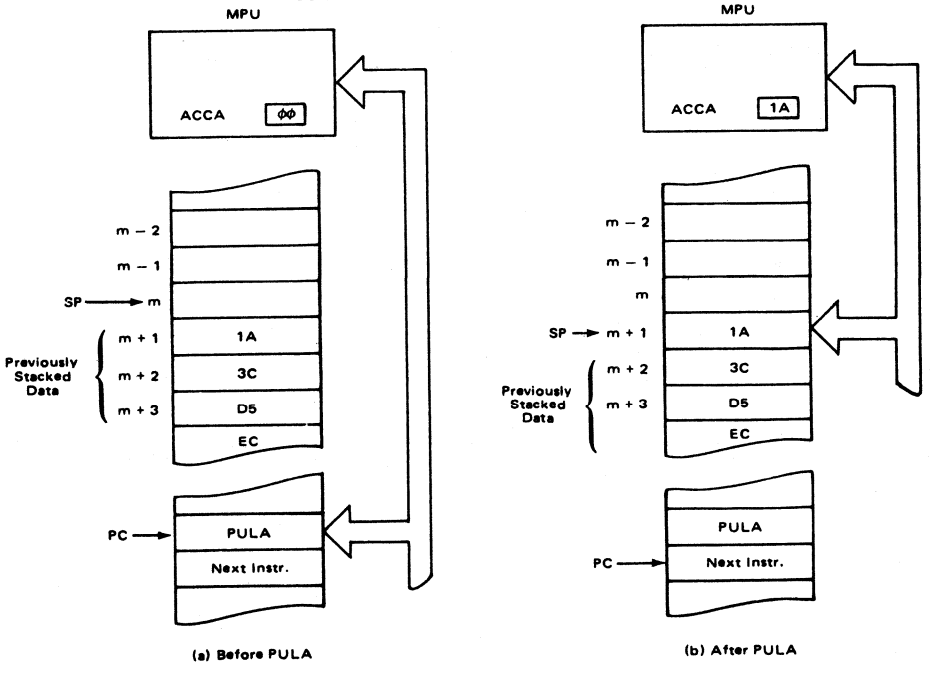


FIGURE 16 – STACK OPERATION, PULL INSTRUCTION



Operation of the Stack Pointer with the Push and Pull instructions is illustrated in Figures 15 and 16. The Push instruction (PSHA) causes the contents of the indicated accumulator (A in this example) to be stored in memory at the location indicated by the Stack Pointer. The Stack Pointer is automatically decremented by one following the storage operation and is "pointing" to the next empty stack location. The Pull instruction (PULA or PULB) causes the last byte stacked to be loaded into the appropriate accumulator. The Stack Pointer is automatically incremented by one just prior to the data transfer so that it will point to the last byte stacked rather than the next empty location. Note that the PULL instruction does not "remove" the data from memory; in the example, 1A is still in location (m + 1) following execution of PULA. A subsequent PUSH instruction would overwrite that location with the new "pushed" data.

Execution of the Branch to Subroutine (BSR) and Jump to Subroutine (JSR) instructions cause a return address to be saved on the stack as shown in Figures 18 through 20. The stack is decremented after each byte of the return address is pushed onto the stack. For both of these instructions, the return address is the memory location following the bytes of code that correspond to the BSR and JSR instruction. The code required for BSR or JSR may be either two or three bytes, depending on whether the JSR is in the indexed (two bytes) or the extended (three bytes) addressing mode. Before it is stacked, the Program Counter is automatically incremented the correct number of times to be pointing at the location of the next instruction. The Return from Subroutine instruction, RTS, causes the return address to be retrieved and loaded into the Program Counter as shown in Figure 21.

There are several operations that cause the status of the MPU to be saved on the stack. The Software Interrupt (SWI) and Wait for Interrupt (WAI) instructions as well as the maskable ($\overline{IRQ}$) and non-maskable ($\overline{NMI}$) hardware interrupts all cause the MPU's internal registers (except for the Stack Pointer itself) to be stacked as shown in Figure 23. MPU status is restored by the Return from Interrupt, RTI, as shown in Figure 22.

Jump and Branch Operation

The Jump and Branch instructions are summarized in Table 9. These instructions are used to control the transfer of operation from one point to another in the control program.

The No Operation instruction, NOP, while included here, is a jump operation in a very limited sense. Its only effect is to increment the Program Counter by one. It is useful during program development as a "stand-in" for some other instruction that is to be determined during debug. It is also used for equalizing the execution time through alternate paths in a control program.

TABLE 9 – JUMP AND BRANCH INSTRUCTIONS

		RELATIVE			INDEX			EXTND			IMPLIED				COND. CODE REG.					
OPERATIONS	MNEMONIC	OP	~	#	OP	~	#	OP	~	#	OP	~	#	BRANCH TEST	5 H	4 I	3 N	2 Z	1 V	0 C
Branch Always	BRA	20	4	2										None	•	•	•	•	•	•
Branch If Carry Clear	BCC	24	4	2										C = 0	•	•	•	•	•	•
Branch If Carry Set	BCS	25	4	2										C = 1	•	•	•	•	•	•
Branch If = Zero	BEQ	27	4	2										Z = 1	•	•	•	•	•	•
Branch If ≥ Zero	BGE	2C	4	2										N ⊕ V = 0	•	•	•	•	•	•
Branch If > Zero	BGT	2E	4	2										Z + (N ⊕ V) = 0	•	•	•	•	•	•
Branch If Higher	BHI	22	4	2										C + Z = 0	•	•	•	•	•	•
Branch If ≤ Zero	BLE	2F	4	2										Z + (N ⊕ V) = 1	•	•	•	•	•	•
Branch If Lower Or Same	BLS	23	4	2										C + Z = 1	•	•	•	•	•	•
Branch If < Zero	BLT	2D	4	2										N ⊕ V = 1	•	•	•	•	•	•
Branch If Minus	BMI	2B	4	2										N = 1	•	•	•	•	•	•
Branch If Not Equal Zero	BNE	26	4	2										Z = 0	•	•	•	•	•	•
Branch If Overflow Clear	BVC	28	4	2										V = 0	•	•	•	•	•	•
Branch If Overflow Set	BVS	29	4	2										V = 1	•	•	•	•	•	•
Branch If Plus	BPL	2A	4	2										N = 0	•	•	•	•	•	•
Branch To Subroutine	BSR	8D	8	2											•	•	•	•	•	•
Jump	JMP				6E	4	2	7E	3	3				See Special Operations	•	•	•	•	•	•
Jump To Subroutine	JSR				AD	8	2	BD	9	3					•	•	•	•	•	•
No Operation	NOP										01	2	1	Advances Prog. Cntr. Only	•	•	•	•	•	•
Return From Interrupt	RTI										3B	10	1		———			①	———	
Return From Subroutine	RTS										39	5	1		•	•	•	•	•	•
Software Interrupt	SWI										3F	12	1	See Special Operations	•	•	•	•	•	•
Wait for Interrupt*	WAI										3E	9	1		•	②	•	•	•	•

*WAI puts Address Bus, R/W, and Data Bus in the three-state mode while VMA is held low.

① (All) Load Condition Code Register from Stack. (See Special Operations)
② (Bit 1) Set when interrupt occurs. If previously set, a Non-Maskable Interrupt is required to exit the wait state.

Execution of the Jump Instruction, JMP, and Branch Always, BRA, affects program flow as shown in Figure 17. When the MPU encounters the Jump (Indexed) instruction, it adds the offset to the value in the Index Register and uses the result as the address of the next instruction to be executed. In the extended addressing mode, the address of the next instruction to be executed is fetched from the two locations immediately following the JMP instruction. The Branch Always (BRA) instruction is similar to the JMP (extended) instruction except that the relative addressing mode applies and the branch is limited to the range within −125 or +127 bytes of the branch instruction itself. The opcode for the BRA instruction requires one less byte than JMP (extended) but takes one more cycle to execute.

The effect on program flow for the Jump to Subroutine (JSR) and Branch to Subroutine (BSR) is shown in Figures 18 through 20. Note that the Program Counter is properly incremented to be pointing at the correct return address before it is stacked. Operation of the Branch to Subroutine and Jump to Subroutine (extended) instruction is similar except for the range. The BSR instruction requires less opcode than JSR (2 bytes versus 3 bytes) and also executes one cycle faster than JSR. The Return from Subroutine, RTS, is used at the end of a subroutine to return to the main program as indicated in Figure 21.

The effect of executing the Software Interrupt, SWI, and the Wait for Interrupt, WAI, and their relationship to the hardware interrupts is shown in Figure 22. SWI causes the MPU contents to be stacked and then fetches the starting address of the interrupt routine from the memory locations that respond to the addresses FFFA and FFFB. Note that as in the case of the subroutine instructions, the Program Counter is incremented to point at the correct return address before being stacked. The Return from Interrupt instruction, RTI, (Figure 22) is used at the end of an interrupt routine to restore control to the main program. The SWI instruction is useful for inserting break points in the control program, that is, it can be used to stop operation and put the MPU registers in memory where they can be examined. The WAI instruction is used to decrease the time required to service a hardware interrupt; it stacks the MPU contents and then waits for the interrupt to occur, effectively removing the stacking time from a hardware interrupt sequence.

FIGURE 17 – PROGRAM FLOW FOR JUMP AND BRANCH INSTRUCTIONS

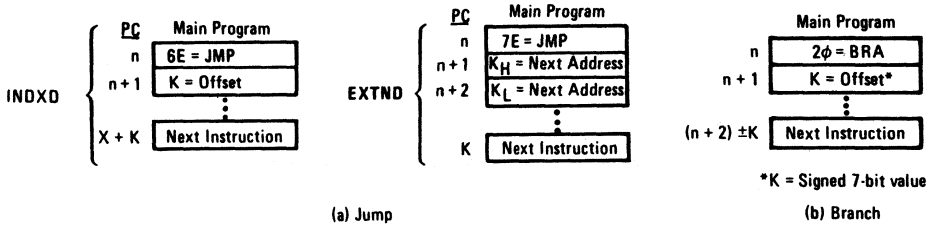


FIGURE 18 – PROGRAM FLOW FOR BSR

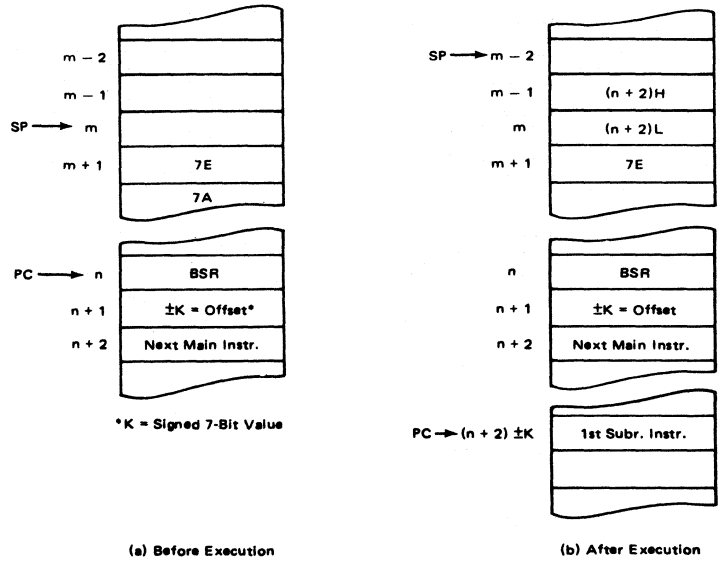


FIGURE 19 – PROGRAM FLOW FOR JSR (EXTENDED)

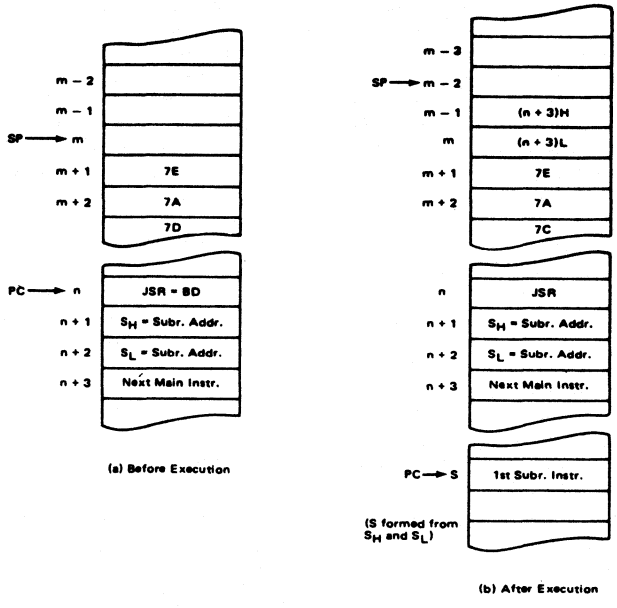


FIGURE 20 – PROGRAM FLOW FOR JSR (INDEXED)

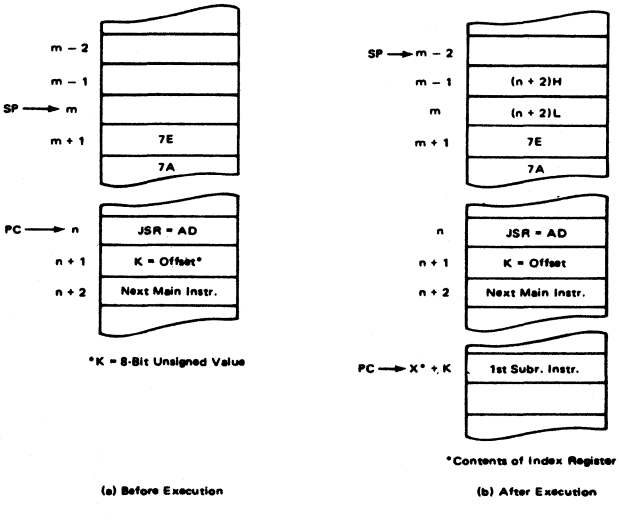
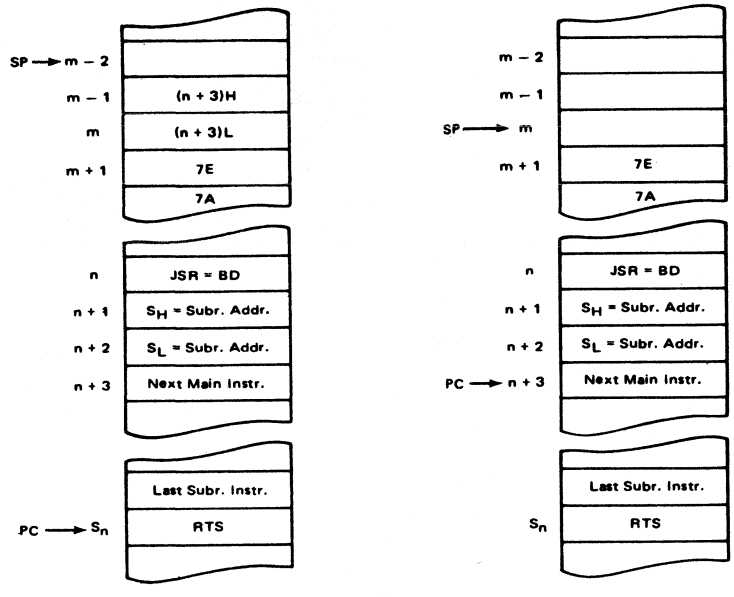


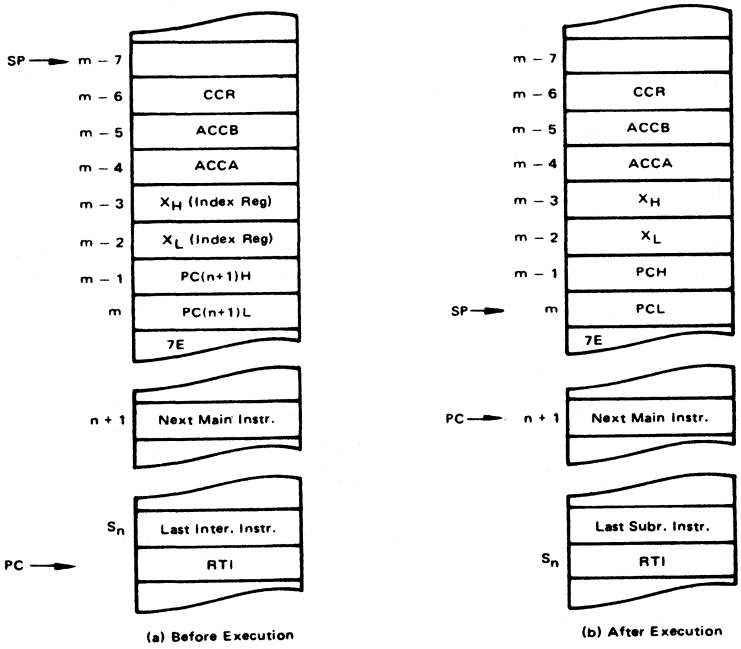
FIGURE 21 – PROGRAM FLOW FOR RTS



(a) Before Execution

(b) After Execution

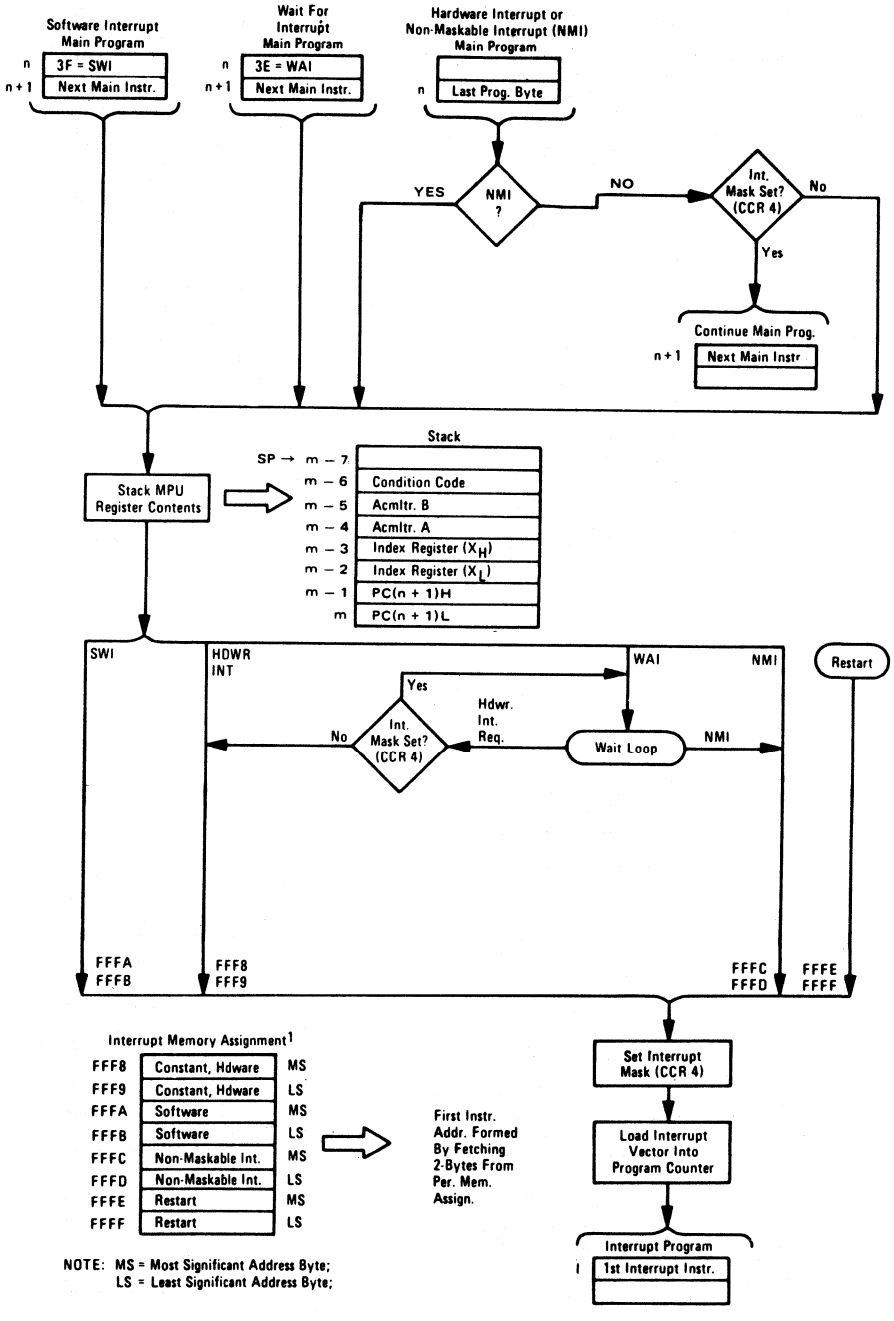
FIGURE 22 – PROGRAM FLOW FOR RTI



(a) Before Execution

(b) After Execution

FIGURE 23 – PROGRAM FLOW FOR INTERRUPTS



NOTE: MS = Most Significant Address Byte;
LS = Least Significant Address Byte;

FIGURE 24 – CONDITIONAL BRANCH INSTRUCTIONS

BMI :	N = 1 ;	BEQ :	Z = 1 ;
BPL :	N = ϕ ;	BNE :	Z = ϕ ;
BVC :	V = ϕ ;	BCC :	C = ϕ ;
BVS :	V = 1 ;	BCS :	C = 1 ;
BHI :	C + Z = ϕ ;	BLT :	N⊕V = 1 ;
BLS :	C + Z = 1 ;	BGE :	N⊕V = ϕ ;
BLE :	Z + (N⊕V) = 1 ;		
BGT :	Z + (N⊕V) = ϕ ;		

The conditional branch instructions, Figure 24, consists of seven pairs of complementary instructions. They are used to test the results of the preceding operation and either continue with the next instruction in sequence (test fails) or cause a branch to another point in the program (test succeeds).

Four of the pairs are used for simple tests of status bits N, Z, V, and C:

1. Branch on Minus (BMI) and Branch On Plus (BPL) tests the sign bit, N, to determine if the previous result was negative or positive, respectively.

2. Branch On Equal (BEQ) and Branch On Not Equal (BNE) are used to test the zero status bit, Z, to determine whether or not the result of the previous operation was equal to zero. These two instructions are useful following a Compare (CMP) instruction to test for equality between an accumulator and the operand. They are also used following the Bit Test (BIT) to determine whether or not the same bit positions are set in an accumulator and the operand.

3. Branch On Overflow Clear (BVC) and Branch On Overflow Set (BVS) tests the state of the V bit to determine if the previous operation caused an arithmetic overflow.

4. Branch On Carry Clear (BCC) and Branch On Carry Set (BCS) tests the state of the C bit to determine if the previous operation caused a carry to occur. BCC and BCS are useful for testing relative magnitude when the values being tested are regarded as unsigned binary numbers, that is, the values are in the range 00 (lowest) to FF (highest). BCC following a comparison (CMP) will cause a branch if the (unsigned) value in the accumulator is higher than or the same as the value of the operand. Conversely, BCS will cause a branch if the accumulator value is lower than the operand.

The fifth complementary pair, Branch On Higher (BHI) and Branch On Lower or Same (BLS) are in a sense complements to BCC and BCS. BHI tests for both C and Z = 0; if used following a CMP, it will cause a branch if the value in the accumulator is higher than the operand. Conversely, BLS will cause a branch if the unsigned binary value in the accumulator is lower than or the same as the operand.

The remaining two pairs are useful in testing results of operations in which the values are regarded as signed two's complement numbers. This differs from the unsigned binary case in the following sense: In unsigned, the orientation is higher or lower; in signed two's complement, the comparison is between larger or smaller where the range of values is between –128 and +127.

Branch On Less Than Zero (BLT) and Branch On Greater Than Or Equal Zero (BGE) test the status bits for $N \oplus V = 1$ and $N \oplus V = 0$, respectively. BLT will always cause a branch following an operation in which two negative numbers were added. In addition, it will cause a branch following a CMP in which the value in the accumulator was negative and the operand was positive. BLT will never cause a branch following a CMP in which the accumulator value was positive and the operand negative. BGE, the complement to BLT, will cause a branch following operations in which two positive values were added or in which the result was zero.

The last pair, Branch On Less Than Or Equal Zero (BLE) and Branch On Greater Than Zero (BGT) test the status bits for $Z \oplus (N + V) = 1$ and $Z \oplus (N + V) = 0$, respectively. The action of BLE is identical to that for BLT except that a branch will also occur if the result of the previous result was zero. Conversely, BGT is similar to BGE except that no branch will occur following a zero result.

CONDITION CODE REGISTER OPERATIONS

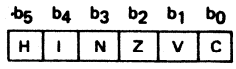
The Condition Code Register (CCR) is a 6-bit register within the MPU that is useful in controlling program flow during system operation. The bits are defined in Figure 25.

The instructions shown in Table 10 are available to the user for direct manipulation of the CCR. In addition, the MPU automatically sets or clears the appropriate status bits as many of the other instructions on the condition code register was indicated as they were introduced.

A CLI-WAI instruction sequence operated properly with early M6800 processors only if the preceding instruction was odd. (Least Significant Bit = 1.) Similarly it was advisable to precede any SEI instruction with an odd opcode—such as NOP. These precautions are not necessary for M6800 processors indicating manufacture in November, 1977 or later.

Systems which require an interrupt window to be opened under program control should use a CLI-NOP-SEI sequence rather than CLI-SEI.

FIGURE 25 – CONDITION CODE REGISTER BIT DEFINITION



H = Half-carry; set whenever a carry from b_3 to b_4 of the result is generated by ADD, ABA, ADC; cleared if no b_3 to b_4 carry; not affected by other instructions.

I = Interrupt Mask; set by hardware or software interrupt or SEI instruction; cleared by CLI instruction. (Normally not used in arithmetic operations.) Restored to a zero as a result of an RTI instruction if I_m stored on the stacked is low.

N = Negative; set if high order bit (b_7) of result is set; cleared otherwise

Z = Zero; set if result = 0; cleared otherwise.

V = Overlow; set if there was arithmetic overflow as a result of the operation; cleared otherwise.

C = Carry; set if there was a carry from the most significant bit (b_7) of the result; cleared otherwise.

TABLE 10 – CONDITION CODE REGISTER INSTRUCTIONS

		IMPLIED				COND. CODE REG.					
						5	4	3	2	1	0
OPERATIONS	MNEMONIC	OP	~	±	BOOLEAN OPERATION	H	I	N	Z	V	C
Clear Carry	CLC	0C	2	1	0→C	•	•	•	•	•	R
Clear Interrupt Mask	CLI	0E	2	1	0→I	•	R	•	•	•	•
Clear Overflow	CLV	0A	2	1	0→V	•	•	•	•	R	•
Set Carry	SEC	0D	2	1	1→C	•	•	•	•	•	S
Set Interrupt Mask	SEI	0F	2	1	1→I	•	S	•	•	•	•
Set Overflow	SEV	0B	2	1	1→V	•	•	•	•	S	•
Acmltr A→CCR	TAP	06	2	1	A→CCR	———	———	①	———	———	———
CCR→Acmltr A	TPA	07	2	1	CCR→A	•	•	•	•	•	•

R = Reset
S = Set
• = Not affected
① (ALL) Set according to the contents of Accumulator A.

ADDRESSING MODES

The MPU operates on 8-bit binary numbers presented to it via the Data Bus. A given number (byte) may represent either data or an instruction to be executed, depending on where it is encountered in the control program. The M6800 has 72 unique instructions, however, it recognizes and takes action on 197 of the 256 possibilities that can occur using an 8-bit word length. This larger number of instructions results from the fact that many of the executive instructions have more than one addressing mode.

These addressing modes refer to the manner in which the program causes the MPU to obtain its instructions and data. The programmer must have a method for addressing the MPU's internal registers and all of the external memory locations.

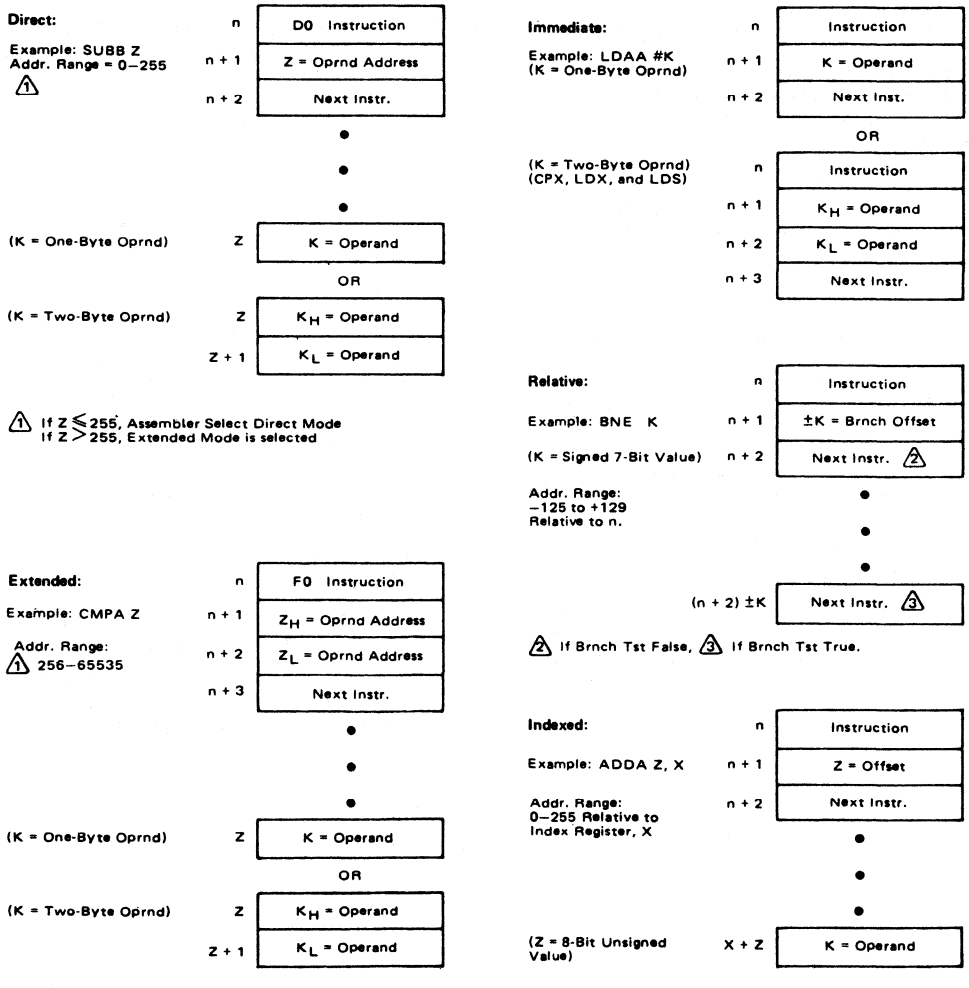
Selection of the desired addressing mode is made by the user as the source statements are written. Translation into appropriate opcode then depends on the method used. If manual translation is used, the addressing mode is inherent in the opcode. For example, the Immediate,

Direct, Indexed, and Extended modes may all be used with the ADD instruction. The proper mode is determined by selecting (hexidecimal notation) 8B, 9B, AB, or BB, respectively.

The source statement format includes adequate information for the selection if an assembler program is used to generate the opcode. For instance, the Immediate mode is selected by the Assembler whenever it encounters the "#" symbol in the operand field. Similarly, an "X" in the operand field causes the Indexed mode to be selected. Only the Relative mode applies to the branch instructions, therefore, the mnemonic instruction itself is enough for the Assembler to determine addressing mode.

For the instructions that use both Direct and Extended modes, the Assembler selects the Direct mode if the operand value is in the range 0-255 and Extended otherwise. There are a number of instructions for which the Extended mode is valid but the Direct is not. For these instructions, the Assembler automatically selects the Extended mode even if the operand is in the 0-255 range. The addressing modes are summarized in Figure 26.

FIGURE 26 – ADDRESSING MODE SUMMARY



Inherent (Includes "Accumulator Addressing" Mode)

The successive fields in a statement are normally separated by one or more spaces. An exception to this rule occurs for instructions that use dual addressing in the operand field and for instructions that must distinguish between the two accumulators. In these cases, A and B are "operands" but the space between them and the operator may be omitted. This is commonly done, resulting in apparent four character mnemonics for those instructions.

The addition instruction, ADD, provides an example of dual addressing in the operand field:

	Operator	Operand	Comment
	ADDA	MEM12	ADD CONTENTS OF MEM12 TO ACCA
or	ADDB	MEM12	ADD CONTENTS OF MEM12 TO ACCB

The example used earlier for the test instruction, TST, also applies to the accumulators and uses the "accumulator addressing mode" to designate which of the two accumulators is being tested:

	Operator	Comment
	TSTB	TEST CONTENTS OF ACCB
or	TSTA	TEST CONTENTS OF ACCA

A number of the instructions either alone or together with an accumulator operand contain all of the address information that is required, that is, "inherent" in the instruction itself. For instance, the instruction ABA causes the MPU to add the contents of accumulators A and B together and place the result in accumulator A. The instruction INCB, another example of "accumulator addressing", causes the contents of accumulator B to be increased by one. Similarly, INX, increment the Index Register, causes the contents of the Index Register to be increased by one.

Program flow for instructions fo this type is illustrated in Figures 27 and 28. In these figures, the general case is shown on the left and a specific example is shown on the right. Numerical examples are in decimal notation. Instructions of this type require only one byte of opcode. Cycle-by-cycle operation of the inehrent mode is shown in Table 11.

FIGURE 27 – INHERENT ADDRESSING

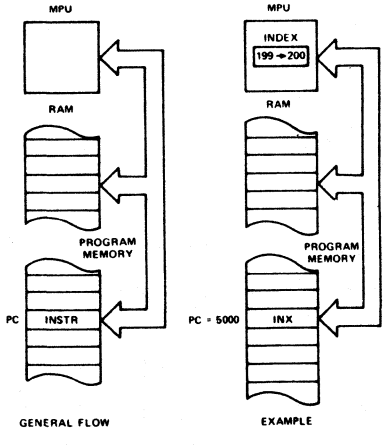


FIGURE 28 – ACCUMULATOR ADDRESSING

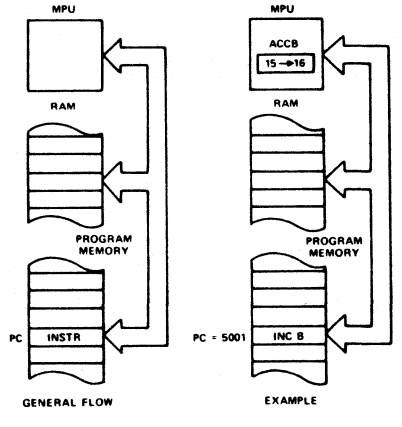


TABLE 11 – INHERENT MODE CYCLE BY CYCLE OPERATION

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
ABA DAA SEC ASL DEC SEI ASR INC SEV CBA LSR TAB CLC NEG TAP CLI NOP TBA CLR ROL TPA CLV ROR TST COM SBA	2	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
DES DEX INS INX	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	0	Previous Register Contents	1	Irrelevant Data (Note 1)
		4	0	New Register Contents	1	Irrelevant Data (Note 1)
PSH	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	1	Stack Pointer	0	Accumulator Data
		4	0	Stack Pointer – 1	1	Accumulator Data
PUL	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	0	Stack Pointer	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer + 1	1	Operand Data from Stack
TSX	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	0	Stack Pointer	1	Irrelevant Data (Note 1)
		4	0	New Index Register	1	Irrelevant Data (Note 1)
TXS	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	0	Index Register	1	Irrelevant Data
		4	0	New Stack Pointer	1	Irrelevant Data
RTS	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Irrelevant Data (Note 2)
		3	0	Stack Pointer	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer + 1	1	Address of Next Instruction (High Order Byte)
		5	1	Stack Pointer + 2	1	Address of Next Instruction (Low Order Byte)

TABLE 11 – INHERENT MODE CYCLE BY CYCLE OPERATION (Continued)

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
WAI	9	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	1	Stack Pointer	0	Return Address (Low Order Byte)
		4	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		5	1	Stack Pointer – 2	0	Index Register (Low Order Byte)
		6	1	Stack Pointer – 3	0	Index Register (High Order Byte)
		7	1	Stack Pointer – 4	0	Contents of Accumulator A
		8	1	Stack Pointer – 5	0	Contents of Accumulator B
		9	1	Stack Pointer – 6 (Note 3)	1	Contents of Cond. Code Register
RTI	10	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Irrelevant Data (Note 2)
		3	0	Stack Pointer	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer + 1	1	Contents of Cond. Code Register from Stack
		5	1	Stack Pointer + 2	1	Contents of Accumulator B from Stack
		6	1	Stack Pointer + 3	1	Contents of Accumulator A from Stack
		7	1	Stack Pointer + 4	1	Index Register from Stack (High Order Byte)
		8	1	Stack Pointer + 5	1	Index Register from Stack (Low Order Byte)
		9	1	Stack Pointer + 6	1	Next Instruction Address from Stack (High Order Byte)
		10	1	Stack Pointer + 7	1	Next Instruction Address from Stack (Low Order Byte)
SWI	12	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Irrelevant Data (Note 1)
		3	1	Stack Pointer	0	Return Address (Low Order Byte)
		4	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		5	1	Stack Pointer – 2	0	Index Register (Low Order Byte)
		6	1	Stack Pointer – 3	0	Index Register (High Order Byte)
		7	1	Stack Pointer – 4	0	Contents of Accumulator A
		8	1	Stack Pointer – 5	0	Contents of Accumulator B
		9	1	Stack Pointer – 6	0	Contents of Cond. Code Register
		10	0	Stack Pointer – 7	1	Irrelevant Data (Note 1)
		11	1	Vector Address FFFA (Hex)	1	Address of Subroutine (High Order Byte)
		12	1	Vector Address FFFB (Hex)	1	Address of Subroutine (Low Order Byte)

Note 1. If device which is addressed during this cycle uses VMA, then the Data Bus will go to the high impedance three-state condition. Depending on bus capacitance, data from the previous cycle may be retained on the Data Bus.

Note 2. Data is ignored by the MPU.

Note 3. While the MPU is waiting for the interrupt, Bus Available will go high indicating the following states of the control lines: VMA is low; Address Bus, R/$\overline{W}$, and Data Bus are all in the high impedance state.

Immediate Addressing Mode – In the Immediate addressing mode, the operand is the value that is to be operated on. For instance, the instruction

Operator	Operand	Comment
LDAA	#25	LOAD 25 INTO ACCA

causes the MPU to "immediately load accumulator A with the value 25"; no further address reference is required. The Immediate mode is selected by preceding the operand value with the "#" symbol. Program flow for this addressing mode is illustrated in Figure 29.

The operand format allows either properly defined symbols or numerical values. Except for the instructions CPX, LDX, and LDS, the operand may be any value in the range 0 to 255. Since Compare Index Register (CPX), Load Index Register (LDX), and Load Stack Pointer (LDS), require 16-bit values, the immediate mode for these three instructions require two-byte operands. In the Immediate addressing mode, the "address" of the operand is effectively the memory location immediately following the instruction itself. Table 12 shows the cycle-by-cycle operation for the immediate addressing mode.

Direct and Extended Addressing Modes – In the Direct and Extended modes of addressing, the operand field of the source statement is the *address* of the value that is to be operated on. The Direct and Extended modes differ only in the range of memory locations to which they can direct the MPU. Direct addressing generates a single 8-bit operand and, hence, can address only memory locations 0 through 255; a two byte operand is generated for Extended addressing, enabling the MPU to reach the remaining memory locations, 256 through 65535. An example of Direct addressing and its effect on program flow is illustrated in Figure 30.

FIGURE 29 – IMMEDIATE ADDRESSING MODE

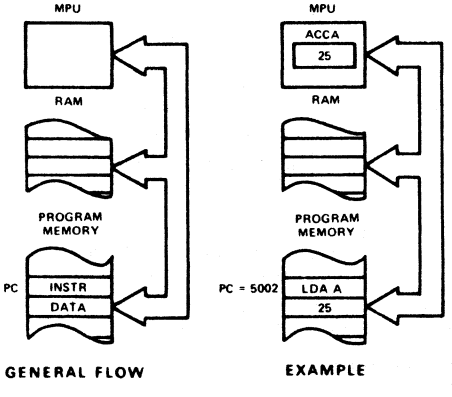


FIGURE 30 – DIRECT ADDRESSING MODE

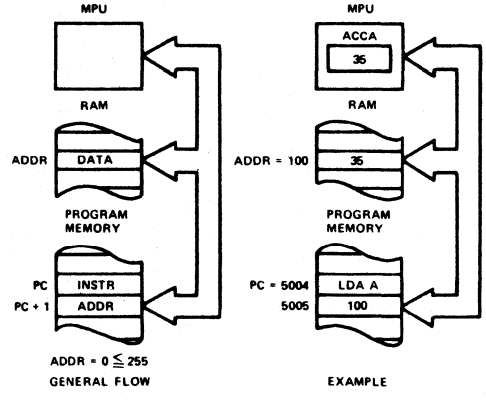


TABLE 12 – IMMEDIATE MODE CYCLE BY CYCLE OPERATION

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	2	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Operand Data
CPX LDS LDX	3	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Operand Data (High Order Byte)
		3	1	Op Code Address + 2	1	Operand Data (Low Order Byte)

The MPU, after encountering the opcode for the instruction LDAA (Direct) at memory location 5004 (Program Counter = 5004), looks in the next location, 5005, for the address of the operand. It then sets the program counter equal to the value found there (100 in the example) and fetches the operand, in this case a value to be loaded into accumulator A, from that location. For instructions requiring a two-byte operand such as LDX (load the Index Register), the operand bytes would be retrieved from locations 100 and 101. Table 13 shows the cycle-by-cycle operation for the direct mode of addressing.

Extended addressing, Figure 31, is similar except that a two-byte address is obtained from locations 5007 and 5008 after the LDAB (Extended) opcode shows up in location 5006. Extended addressing can be thought of as the "standard" addressing mode, that is, it is a method of reaching anyplace in memory. Direct addressing, since only one address byte is required, provides a faster method of processing data and generates fewer bytes of control code. In most applications, the direct addressing range, memory locations 0-255, are reserved for RAM. They are used for data buffering and temporary storage of system variables, the area in which faster addressing is of most value. Cycle-by-cycle operation is shown in Table 14 for Extended Addressing.

TABLE 13 – DIRECT MODE CYCLE BY CYCLE OPERATION

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/$\overline{W}$ Line	Data Bus
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	3	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand
		3	1	Address of Operand	1	Operand Data
CPX LDS LDX	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand
		3	1	Address of Operand	1	Operand Data (High Order Byte)
		4	1	Operand Address + 1	1	Operand Data (Low Order Byte)
STA	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Destination Address
		3	0	Destination Address	1	Irrelevant Data (Note 1)
		4	1	Destination Address	0	Data from Accumulator
STS STX	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand
		3	0	Address of Operand	1	Irrelevant Data (Note 1)
		4	1	Address of Operand	0	Register Data (High Order Byte)
		5	1	Address of Operand + 1	0	Register Data (Low Order Byte)

Note 1. If device which is address during this cycle uses VMA, then the Data Bus will go to the high impedance three-state condition. Depending on bus capacitance, data from the previous cycle may be retained on the Data Bus.

FIGURE 31 – EXTENDED ADDRESSING MODE

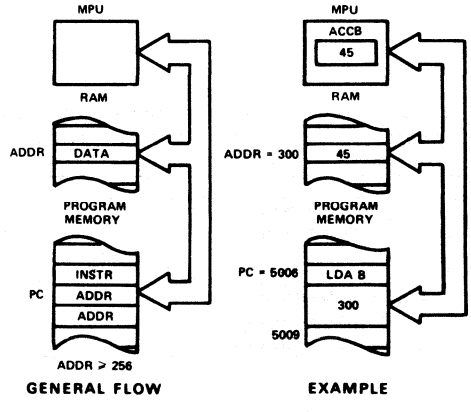


TABLE 14 – EXTENDED MODE CYCLE BY CYCLE

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
STS STX	6	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	0	Address of Operand	1	Irrelevant Data (Note 1)
		5	1	Address of Operand	0	Operand Data (High Order Byte)
		6	1	Address of Operand + 1	0	Operand Data (Low Order Byte)
JSR	9	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Subroutine (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Subroutine (Low Order Byte)
		4	1	Subroutine Starting Address	1	Op Code of Next Instruction
		5	1	Stack Pointer	0	Return Address (Low Order Byte)
		6	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		7	0	Stack Pointer – 2	1	Irrelevant Data (Note 1)
		8	0	Op Code Address + 2	1	Irrelevant Data (Note 1)
		9	1	Op Code Address + 2	1	Address of Subroutine (Low Order Byte)
JMP	3	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Jump Address (High Order Byte)
		3	1	Op Code Address + 2	1	Jump Address (Low Order Byte)
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	1	Address of Operand	1	Operand Data
CPX LDS LDX	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	1	Address of Operand	1	Operand Data (High Order Byte)
		5	1	Address of Operand + 1	1	Operand Data (Low Order Byte)
STA A STA B	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Destination Address (High Order Byte)
		3	1	Op Code Address + 2	1	Destination Address (Low Order Byte)
		4	0	Operand Destination Address	1	Irrelevant Data (Note 1)
		5	1	Operand Destination Address	0	Data from Accumulator
ASL LSR ASR NEG CLR ROL COM ROR DEC TST INC	6	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	1	Address of Operand	1	Current Operand Data
		5	0	Address of Operand	1	Irrelevant Data (Note 1)
		6	1/0 (Note 2)	Address of Operand	0	New Operand Data (Note 2)

Note 1. If device which is addressed during this cycle uses VMA, then the Data Bus will go to the high impedance three-state condition. Depending on bus capacitance, data from the previous cycle may be retained on the Data Bus.

Note 2. For TST, VMA = 0 and Operand data does not change.

Relative Address Mode – In both the Direct and Extended modes, the address obtained by the MPU is an absolute numerical address. The Relative addressing mode, implemented for the MPU's branch instructions, specifies a memory location relative to the Program Counter's current location. Branch instructions generate two bytes of machine code, one for the instruction opcode and one for the "relative" address (see Figure 32). Since it is desirable to be able to branch in either direction, the 8-bit address byte is interpreted as a signed 7-bit value; the 8th bit of the operand is treated as a sign bit, "0" = plus and "1" = minus. The remaining seven bits represent the numerical value. This results in a relative addressing range of ±127 with respect to the location of the branch instruction itself. However, the branch range is computed with respect to the next instruction that would be executed if the branch conditions are not satisfied. Since two bytes are generated, the next instruction is located at PC + 2. If D is defined as the address of the branch destination, the range is then:

$$(PC + 2) - 127 \leqslant D \leqslant (PC + 2) + 127$$

or

$$PC - 125 \leqslant D \leqslant PC + 129$$

that is, the destination of the branch instruction must be within −125 to +129 memory locations of the branch instruction itself. For transferring control beyond this range, the unconditional jump (JMP), jump to subroutine (JSR), and return from subroutine (RTS) are used.

In Figure 32, when the MPU encounters the opcode for BEQ (Branch if result of last instruction was zero), it tests the Zero bit in the Condition Code Register. If that bit is "0", indicating a non-zero result, the MPU continues execution with the next instruction (in location 5010 in Figure 32). If the previous result was zero, the branch condition is satisfied and the MPU adds the offset, 15 in this case, to PC + 2 and branches to location 5025 for the next instruction.

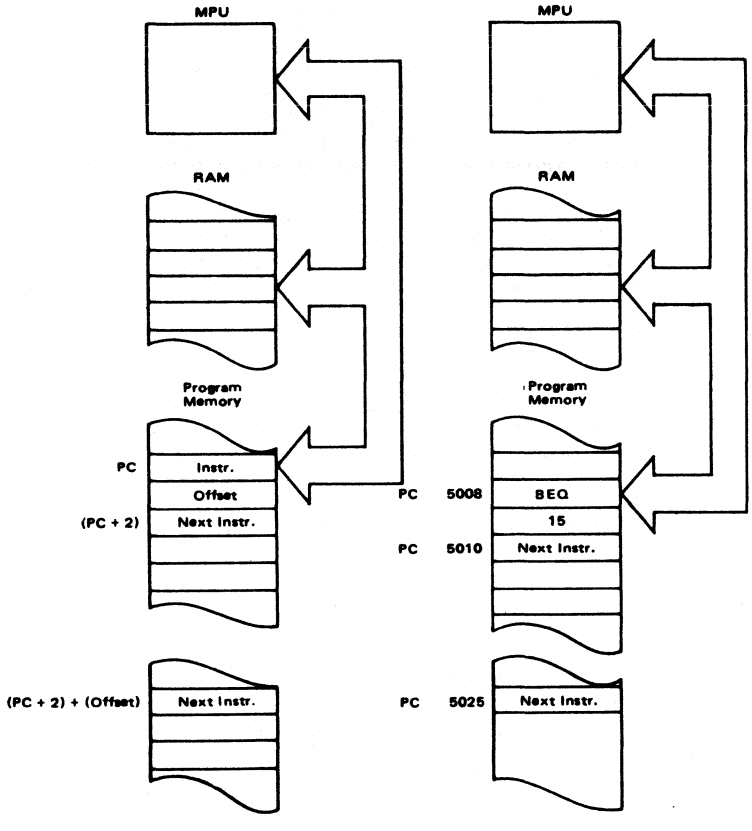
The branch instructions allow the programmer to efficiently direct the MPU to one point or another in the control program depending on the outcome of test results. Since the control program is normally in read-only memory and cannot be changed, the relative address used in execution of branch instructions is a constant numerical value. Cycle-by-cycle operation is shown in Table 15 for relative addressing.

TABLE 15 – RELATIVE MODE CYCLE-BY-CYCLE OPERATION

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
BCC BHI BNE BCS BLE BPL BEQ BLS BRA BGE BLT BVC BGT BMI BVS	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Branch Offset
		3	0	Op Code Address + 2	1	Irrelevant Data (Note 1)
		4	0	Branch Address	1	Irrelevant Data (Note 1)
BSR	8	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Branch Offset
		3	0	Return Address of Main Program	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer	0	Return Address (Low Order Byte)
		5	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		6	0	Stack Pointer – 2	1	Irrelevant Data (Note 1)
		7	0	Return Address of Main Program	1	Irrelevant Data (Note 1)
		8	0	Subroutine Address	1	Irrelevant Data (Note 1)

Note 1. If device which is addressed during this cycle uses VMA, then the Data Bus will go to the high impedance three-state condition. Depending on bus capacitance, data from the previous cycle may be retained on the Data Bus.

FIGURE 32 – RELATIVE ADDRESSING MODE



Indexed Addressing Mode – With Indexed addressing, the numerical address is variable and depend on the current contents of the Index Register. A source statement such as

Operator	Operand	Comment
STAA	X	PUT A IN INDEXED LOCATION

causes the MPU to store the contents of accumulator A in the memory location specified by the contents of the Index Register (recall that the label "X" is reserved to designate the Index Register). Since there are instructions for manipulating X during program execution (LDX, INX, DEX, etc.), the Indexed addressing mode provides a dynamic "on the fly" way to modify program activity.

The operand field can also contain a numerical value that will be automatically added to X during execution. This format is illustrated in Figure 33.

When the MPU encounters the LDAB (Indexed) opcode in location 5006, it looks in the next memory location for the value to be added to X (5 in the example) and calculates the required address by adding 5 to the present Index Register value of 400. In the operand format, the offset may be represented by a label or a numerical value in the range 0-255 as in the example. In the earlier example, STAA X, the operand is equivalent to 0,X, that is, the 0 may be omitted when the desired address is equal to X. Table 16 shows the cycle-by-cycle operation for the Indexed Mode of Addressing.

FIGURE 33 – INDEXED ADDRESSING MODE

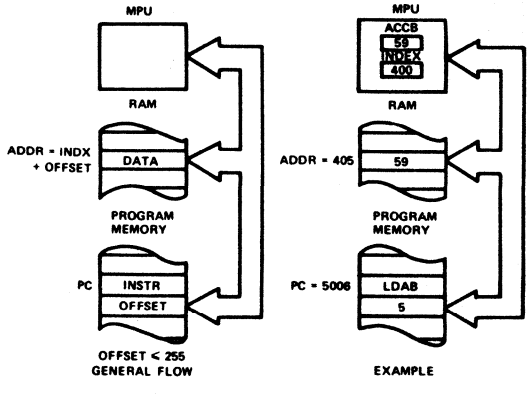


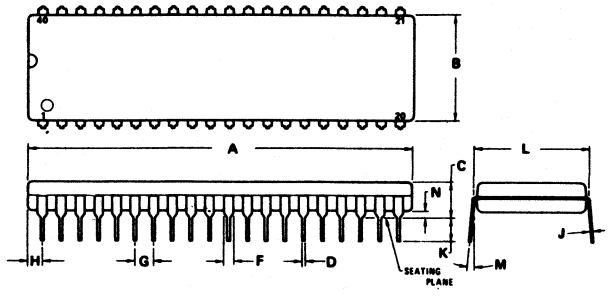
TABLE 16 – INDEXED MODE CYCLE BY CYCLE

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
INDEXED						
JMP	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	1	Index Register Plus Offset	1	Operand Data
CPX LDS LDX	6	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	1	Index Register Plus Offset	1	Operand Data (High Order Byte)
		6	1	Index Register Plus Offset + 1	1	Operand Data (Low Order Byte)
STA	6	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	0	Index Register Plus Offset	1	Irrelevant Data (Note 1)
		6	1	Index Register Plus Offset	0	Operand Data
ASL LSR ASR NEG CLR ROL COM ROR DEC TST INC	7	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	1	Index Register Plus Offset	1	Current Operand Data
		6	0	Index Register Plus Offset	1	Irrelevant Data (Note 1)
		7	1/0 (Note 2)	Index Register Plus Offset	0	New Operand Data (Note 2)
STS STX	7	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	0	Index Register Plus Offset	1	Irrelevant Data (Note 1)
		6	1	Index Register Plus Offset	0	Operand Data (High Order Byte)
		7	1	Index Register Plus Offset + 1	0	Operand Data (Low Order Byte)
JSR	8	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer	0	Return Address (Low Order Byte)
		5	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		6	0	Stack Pointer – 2	1	Irrelevant Data (Note 1)
		7	0	Index Register	1	Irrelevant Data (Note 1)
		8	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)

Note 1. If device which is addressed during this cycle uses VMA, then the Data Bus will go to the high impedance three-state condition. Depending on bus capacitance, data from the previous cycle may be retained on the Data Bus.

Note 2. For TST, VMA = 0 and Operand data does not change.

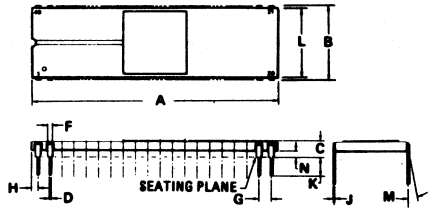
PACKAGE DIMENSIONS



NOTES:
1. LEADS TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION (DIM "D").
2. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	51.82	52.32	2.040	2.060
B	13.72	14.22	0.540	0.560
C	4.57	5.08	0.180	0.200
D	0.36	0.51	0.014	0.020
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	3.05	3.56	0.120	0.140
L	15.24 BSC		0.600 BSC	
M	0°	10°	0°	10°
N	0.51	1.02	0.020	0.040

CASE 711-02
(PLASTIC)



	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	50.29	51.31	1.980	2.020
B	14.86	15.62	0.585	0.615
C	2.54	4.19	0.100	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.33	0.008	0.013
K	2.54	4.19	0.100	0.165
L	14.60	15.37	0.575	0.605
M	0°	10°	0°	10°
N	0.51	1.52	0.020	0.060

NOTE:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA (AT SEATING PLANE), AT MAX. MAT'L CONDITION.

CASE 715-02
(CERAMIC)

MOTOROLA

MC6801

Advance Information

MOS

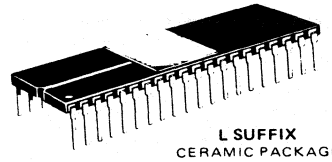
(N-CHANNEL, SILICON-GATE DEPLETION LOAD)

MICROCOMPUTER

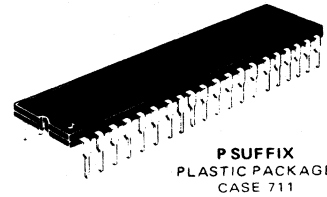
MICROCOMPUTER UNIT (MCU)

The MC6801 MCU is an 8-bit microcomputer system which is compatible with the M6800 family of parts. The MC6801 MCU is object code compatible with the MC6800 with improved execution times of key instructions plus several new 16-bit and 8-bit instructions including an 8 X 8 unsigned multiply with 16-bit result. The MC6801 MCU can operate as a single chip microcomputer or be expanded to 65K words. The MC6801 MCU is TTL compatible and requires one +5.0 volt power supply. The MC6801 MCU has 2K bytes of ROM and 128 bytes of RAM on chip, Serial Communications Interface (S.C.I.), and parallel I/O as well as a three function 16-bit timer. Block diagram is shown in Figure 1. Features of the MC6801 include the following:

- Expanded M6800 Instruction Set
- 8 X 8 Multiply
- On-Chip Serial Communications Interface (S.C.I.)
- Object Code Compatible With The MC6800 MPU
- 16-Bit Timer
- Single Chip Or Expandable To 65K Words
- 2K Bytes Of ROM
- 128 Bytes Of RAM (64 Bytes Retainable On Power Down)
- 31 Parallel I/O Lines
- Internal Clock/Divided-By-Four
- TTL Compatible Inputs And Outputs
- Interrupt Capability
- **External Clock/Divide-By-One Mask Option (MC6801E) And EPROM Versions MC68701 And MC68701E Available Soon.**



L SUFFIX
CERAMIC PACKAGE
CASE 715



P SUFFIX
PLASTIC PACKAGE
CASE 711

FIGURE 1 - SINGLE-CHIP MICROCOMPUTER BLOCK DIAGRAM

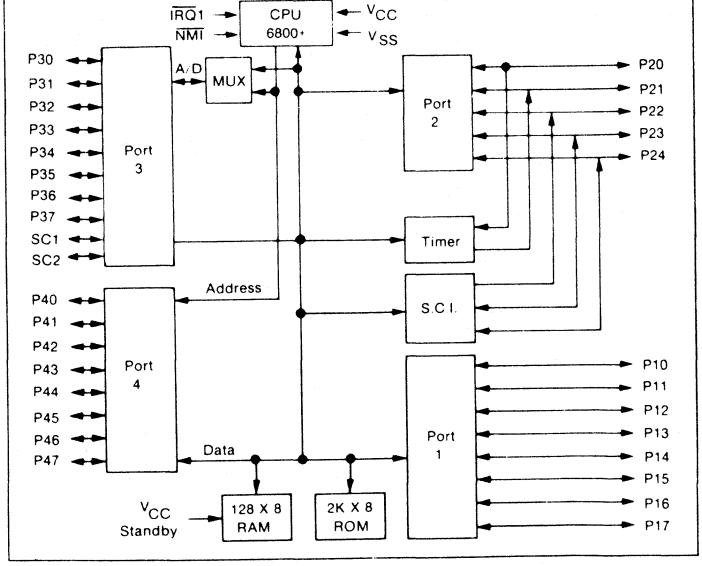
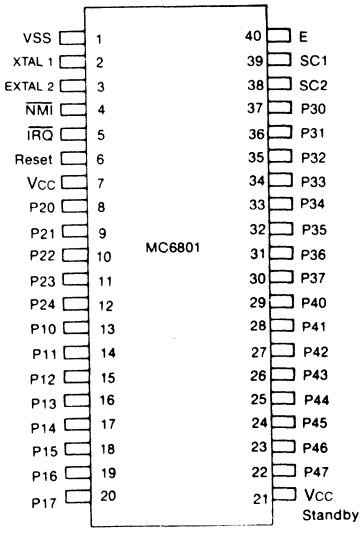


FIGURE 2 — PIN ASSIGNMENT



This is advance information and specifications are subject to change without notice.

ADI-803R1

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0V ±5%, V_{SS} = 0, T_A = T_L to T_H unless otherwise noted.)

Characteristic	Symbol	Min	Typ	Max	Unit
Input High Voltage Reset	V_{IH}	V_{SS} + 2.0 V_{SS} + 4.0	-	V_{CC} V_{CC}	Vdc
Input Low Voltage	V_{IL}	V_{SS} - 0.3	-	V_{SS} + 0.8	Vdc
Three-State (Off State) Input Current P10-P17 (V_{in} = 0.4 to 2.4 Vdc) P20-P24, P30-P37	I_{TSI} I_{TSI}	- -	2.0 2.0	10 10	μAdc μAdc
Output High Voltage All Outputs Except XTAL 1 and EXTAL 2 (I_{Load} = -200 μAdc)	V_{OH}	V_{SS} + 2.4	-	-	Vdc
Output Low Voltage All Outputs Except XTAL 1 and EXTAL 2 (I Load = 1.6 mAdc)	V_{OL}	-	-	V_{SS} +0.4	Vdc
Power Dissipation	P_D	-	-	1200	mW
Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz) P10-P17, P20-P24, P40-P47 P30-P37 Reset SC1, SC2, IRQ	C_{in}	 - - -	 - - -	 12.5 10 7.5	pF
Peripheral Data Setup Time (Figure 5)	t_{PDSU}	200	-	-	ns
Peripheral Data Hold Time (Figure 5)	t_{PDH}	0	-	-	ns
Delay Time, Enable negative transition to OS3 negative transition	t_{OSD1}	-	-	1.0	μs
Delay Time, Enable negative transition to OS3 positive transition	t_{OSD2}	-	-	1.0	μs
Delay Time, Enable negative transition to Peripheral Data Valid (Figure 6)	t_{PWD}	-	-	350	ns
Delay Time, Enable negative transition to Peripheral CMOS Data Valid (V_{CC} - 30% V_{CC}, P20-P24 (Figure 6)	t_{CMOS}	-	-	2.0	μs
Darlington Drive Current V_O = 1.5 Vdc P10-P17	I_{OH}	-1.0	-2.5	-10	mAdc
Standby Voltage (Not Operating) (Operating)	V_{SBB} V_{SB}	4.00 4.75	- -	5.25 5.25	Vdc

NOTE: The above electricals satisfy Ports 1 and 2 always, and Ports 3 and 4 in the single chip mode only.

BUS TIMING (Figure 9)

Characteristic	Symbol	Min	Typ	Max	Unit
Cycle Time	t_{CYC}	1000	-	-	ns
Address Strobe Pulse Width High	PW_{ASH}	220	-	-	ns
Address Strobe Rise Time	t_{ASR}	-	-	50	ns
Address Strobe Fall Time	t_{ASF}	-	-	50	ns
Address Strobe Delay Time	t_{ASD}	60	-	-	ns
Enable Rise Time	t_{ER}	-	-	50	ns
Enable Fall Time	t_{EF}	-	-	50	ns
Enable Pulse Width High Time	PW_{EH}	450	-	-	ns
Enable Pulse Width Low Time	PW_{EL}	450	-	-	ns
Address Strobe to Enable Delay Time	t_{ASED}	60	-	-	ns
Address Delay Time	t_{AD}	-	-	270	ns
Data Delay Write Time	t_{DDW}	-	-	225	ns
Data Set-up Time	t_{DSR}	100	-	-	ns
Hold Time Read	t_{HR}	20	-	100	ns
Hold Time Write	t_{HW}	20	-	-	ns
Address Delay Time for Latch	t_{ADL}	-	-	200	ns
Address Hold Time for Latch	t_{AHL}	20	-	-	ns
Pulse Width	PW_O	370	370	-	ns
Address Hold Time	t_{AH}	20	-	-	ns
Total Up Time	t_{UT}	750	-	-	ns

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	VCC	-0.3 to +7.0	Vdc
Input Voltage	Vin	-0.3 to +7.0	Vdc
Operating Temperature Range	TA	0 to 70	°C
Storage Temperature Range	Tstg	-55 to +150	°C
Thermal Resistance Plastic Package Ceramic Package	θ_{JA}	100 50	°C/W

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leqq (V_{in}$ or $V_{out}) \leqq V_{DD}$.

TABLE 1 — MODE AND PORT SUMMARY

MCU SIGNAL DESCRIPTION

This section gives a description of the MCU signals for the various modes. Figure 2 shows the general pin assignments for the signals. SC1 and SC2 are signals which vary with the mode that the chip is in. Table 1 gives a summary of their function.

MODE	PORT 1 Eight Lines	PORT 2 Five Lines	PORT 3 Eight Lines	PORT 4 Eight Lines	SC1	SC2
SINGLE CHIP	I/O	I/O	I/O	I/O	$\overline{IS3}$(I)	$\overline{OS3}$(O)
EXPANDED MUX	I/O	I/O	ADDRESS BUS (A0-A7) DATA BUS (D0-D7)	ADDRESS BUS* (A8-A15)	AS(O)	R/$\overline{W}$(O)
EXPANDED NON-MUX	I/O	I/O	DATA BUS (D0-D7)	ADDRESS BUS* (A0-A7)	$\overline{IOS}$(O)	R/$\overline{W}$(O)

*These lines can be substituted for I/O (Input Only) starting with the most significant address line.

I = Input
O = Output
R/$\overline{W}$ = Read/$\overline{Write}$
IS = Input Strobe
OS = Output Strobe
IOS = I/O Select
SC = Strobe Control
AS = Address Strobe

READ/WRITE TIMING FOR PORTS 3 AND 4 (Figures 3-4)

Characteristic	Symbol	Min	Typ	Max	Unit
Address Delay	t_{AD}	-	-	270	ns
Peripheral Read Access Time $t_{acc} = t_{ut} - (t_{AD} + t_{DSR})$	t_{acc}	-	-	530	ns
Data Setup Time (Read)	t_{DSR}	100	-	-	ns
Input Data Hold Time	t_{HR}	10	-	-	ns
Output Data Hold Time	t_{HW}	20	-	-	ns
Address Hold Time (Address, R/W)	t_{AH}	20	-	-	ns
Data Delay Time (Write)	t_{DDW}	-	165	225	ns
Processor Controls					
Processor Control Setup Time	t_{PCS}	200	-	-	ns
Processor Control Rise and Fall Time (Measured between 0.8V and 2.0V)	t_{PCr}, t_{PCf}	-	-	100 100	ns

PORT 3 STROBE TIMING (Figures 7-8)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Strobe Delay 1	t_{DSD1}	-	-	1.0	μs
Output Strobe Delay 2	T_{OSD2}	-	-	1.0	μs
Input Strobe Pulse Width	PW_{IS}	200	-	-	ns
Input Data Hold Time	t_{IH}	20	-	-	ns
Input Data Setup Time	t_{IS}	100	-	-	ns

FIGURE 3 — READ DATA FROM MEMORY OR PERIPHERALS EXPANDED NON-MULTIPLEXED

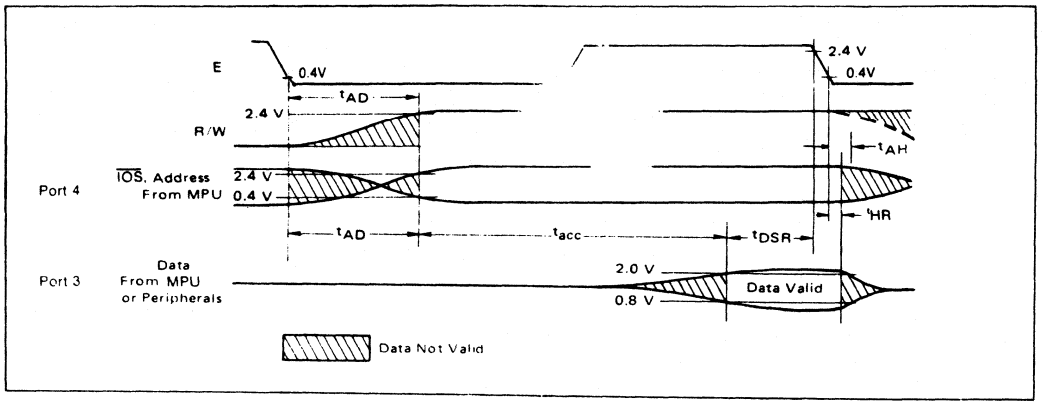
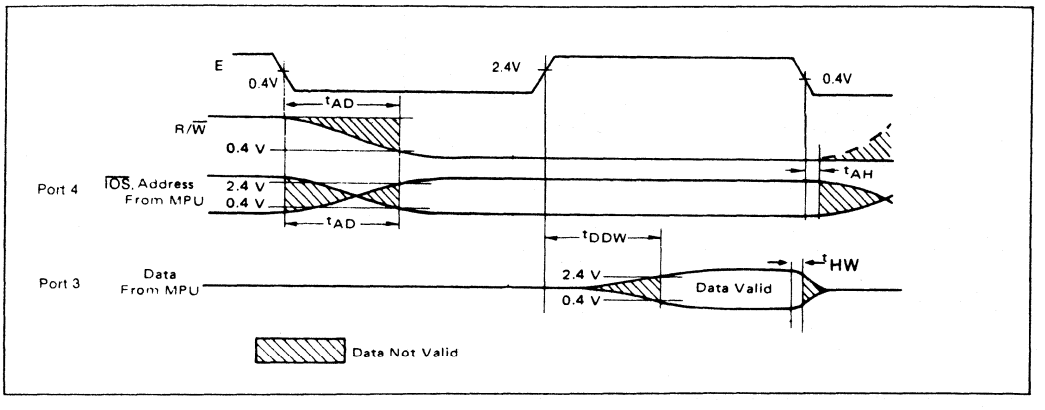


FIGURE 4 — WRITE DATA IN MEMORY OR PERIPHERALS EXPANDED NON-MULTIPLEXED



PORTS 1 AND 2 , AND PORTS 3 AND 4 IN THE SINGLE CHIP MODE

FIGURE 5 — PERIPHERAL DATA SETUP AND HOLD TIMES
(Read Mode)

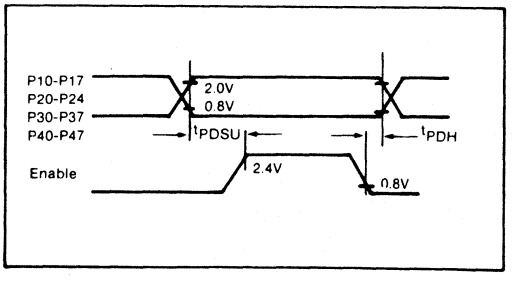


FIGURE 6 — PERIPHERAL CMOS DATA DELAY TIMES
(Write Mode)

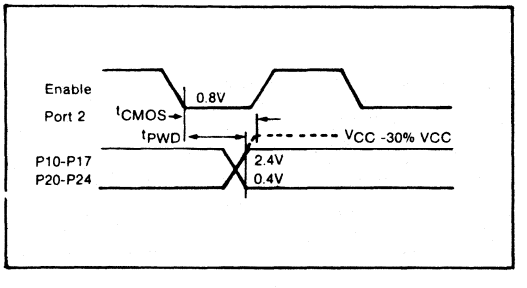


FIGURE 7 — OUTPUT STROBE TIMING — SINGLE CHIP MODE

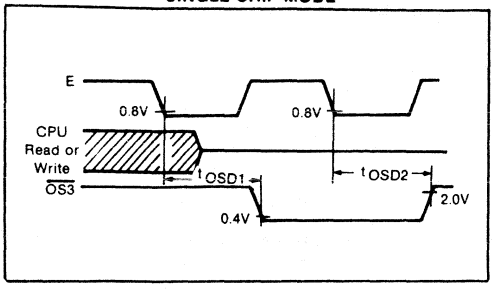


FIGURE 8 — INPUT STROBE TIMING — SINGLE CHIP MODE

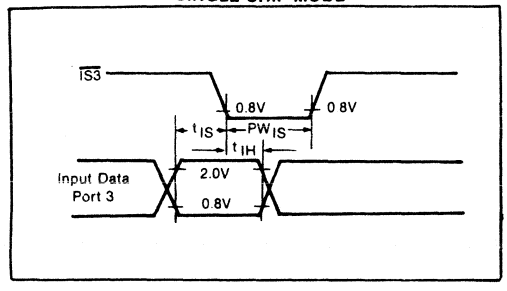


FIGURE 9 — MULTIPLEXED BUS TIMING

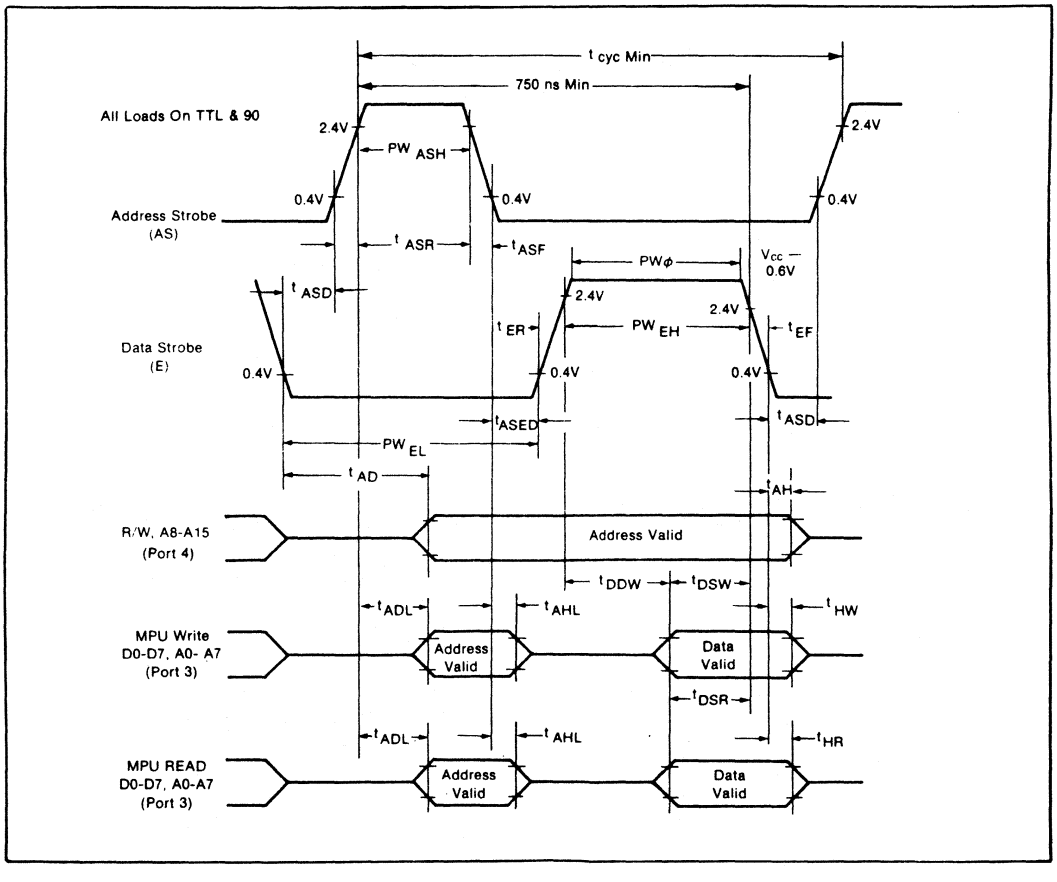


FIGURE 10—CMOS LOAD

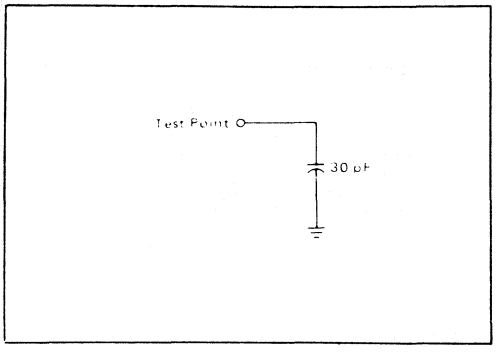
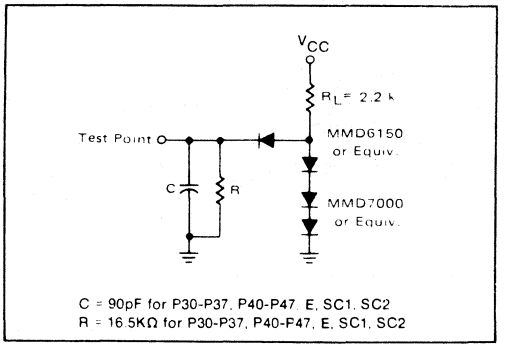


FIGURE 11 — BUS TIMING TEST LOAD AND PORTS 1, 3 AND 4 FOR SINGLE CHIP MODE



C = 90pF for P30-P37, P40-P47, E, SC1, SC2
R = 16.5KΩ for P30-P37, P40-P47, E, SC1, SC2

FIGURE 12 — TEST LOADS FOR PORT 1
Darlington Load
(P10-P17)

C = 40 R_L, R = 12k
Adjust R_L so that I_1 = 3.2 mA
with V_1 = 0.4V and V_{CC}=5.25 V

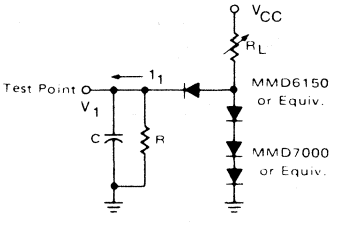


FIGURE 13 — TYPICAL DATA BUS OUTPUT DELAY versus CAPACITIVE LOADING

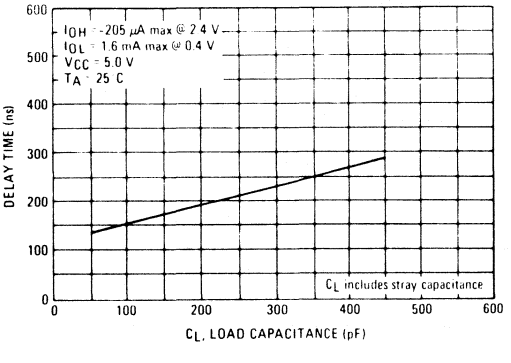
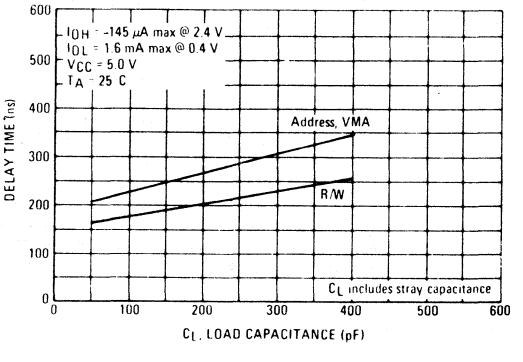


FIGURE 14 — TYPICAL READ/WRITE, VMA AND ADDRESS OUTPUT DELAY versus CAPACITIVE LOADING



SIGNAL DESCRIPTIONS

V_{CC} and Vss

These two pins are used to supply power and ground to the chip. The voltage supplied will be +5 volts ±5%.

XTAL 1 and EXTAL 2

These connections are for a parallel resonant fundamental crystal, AT cut. Divide by 4 circuitry is included with the internal clock, so a 4 MHz crystal may be used to run the system at 1 MHz. The divide by 4 circuitry allows for use of the inexpensive 3.56 MHz Color TV crystal for non-time critical applications. Two 27 pF capacitors are needed from the two crystal pins to ground to insure reliable operation. EXTAL2 may be driven by an external clock source at a 4 MHz rate to run at 1 MHz with a 40/60% duty cycle. It is not restricted to 4 MHz, as it will divide by 4 any frequency less than or equal to 4 MHz. XTAL1 must be grounded if an external clock is used. The following are the recommended crystal parameters:

AT = Cut Parallel Resonance Crystal
C_o = 7 pF MAX
FREQ = 4.0 MHz @ C_L = 24 pF
R_S = 50 ohms MAX
Frequency Tolerance - ±5% to ±0.02%
The best E output "Worst Case Design" tolerance is ±0.05% (500 ppM) using A ±0.02% crystal.

V_{CC} Standby

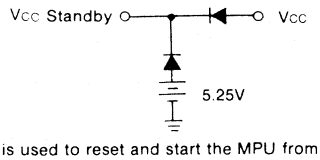
This pin will supply +5 volts -5% to the standby RAM on the chip. The first 64 bytes of RAM will be maintained in the power down mode with 8 mA current max in the ROM version. The circuit of figure 15 can be utilized to assure that V_{CC} Standby does not go below V_{SBB} during power down.

To retain information in the RAM during power down the following procedure is necessary:

1) Write "0" into the RAM enable bit, RAM E. RAM E is bit 6 of the RAM Control Register at location $0014. This disables the standby RAM, thereby protecting it at power down.

2) Keep V_{CC} Standby greater than V_{SBB}.

FIGURE 15—BATTERY BACKUP FOR V_{CC} STANDBY



Reset

This input is used to reset and start the MPU from a power down condition, resulting from a power failure or an initial start-up of the processor. On power up, the reset must be held low for at least 20 ms. During operation, $\overline{\text{Reset}}$, when brought low, must be held low at least 3 clock cycles.

When a high level is detected, the MPU does the following:

a) All the higher order address lines will be forced high.

b) I/O Port 2 bits, 2, 1, and 0 are latched into programmed control bits PC2, PC1 and PC0.

c) The last two (FFFE, FFFF) locations in memory will be used to load the program addressed by the program counter.

d) The interrupt mask bit is set, must be cleared before the MPU can recognize maskable interrupts.

Enable (E)

This supplies the external clock for the rest of the system when the internal oscillator is used. It is a single phase, TTL compatible clock, and will be the divide by 4 result of the crystal frequency. It will drive one TTL load and 90 pF.

Non-Maskable Interrupt ($\overline{\text{NMI}}$)

A low-going edge on this input requests that a non-maskable-interrupt sequence be generated within the processor. As with the Interrupt Request signal, the processor will complete the current instruction that is being executed before it recognizes the $\overline{\text{NMI}}$ signal. The interrupt mask bit in the Condition Code Register has no effect on $\overline{\text{NMI}}$.

In response to an $\overline{\text{NMI}}$ interrupt, the Index Register, Program Counter, Accumulators, and Condition Code Register are stored on the stack. At the end of the sequence, a 16-bit address will be loaded that points to a vectoring address located in memory locations FFFC and FFFD. An address loaded at these locations causes the MPU to branch to a non-maskable interrupt service routine in memory.

A 3.3 kΩ external resistor to V_{CC} should be used for wire-OR and optimum control of interrupts.

Inputs $\overline{\text{IRQ}}$ and $\overline{\text{NMI}}$ are hardware interrupt lines that are sampled during E and will start the interrupt routine on the clock bar following the completion of an instruction.

Interrupt Request ($\overline{\text{IRQ}}$)

This level sensitive input requests that an interrupt sequence be generated within the machine. The processor will wait until it completes the current instruction that is being executed before it recognizes the request. At that time, if the interrupt mask bit in the Condition Code Register is not set, the machine will begin an interrupt sequence. The Index Register, Program Counter, Accumulators, and Condition Code Register are stored on the stack. Next the MPU will respond to the interrupt request by setting the interrupt mask bit high so that no further maskable interrupts may occur. At the end of the cycle, a 16-bit address will be loaded that points to a vectoring address which is located in memory locations FFF8 and FFF9. An address loaded at these locations causes the MPU to branch to an interrupt routine in memory.

The IRQ requires a 3.3 kΩ external resistor to V_{CC} which should be used for wire-OR and optimum control of interrupts. Internal Interrupts will use an internal interrupt line (IRQ2). This interrupt will operate the same as IRQ except that it will use the vector address of FFF0 and FFF7. IRQ1 will have priority over IRQ2 if both occur at the same time. The Interrupt Mask Bit in the condition mode register masks both interrupts. (See Figure 25).

The following pins are available in the Single Chip Mode, and are associated with Port 3 only.

Input Strobe ($\overline{\text{IS3}}$) (SC1)

This sets an interrupt for the processor when the IS3 Enable bit is set. As shown in Figure 8 Input Strobe Timing, IS3 will fall T_{IS} minimum after data is valid on Port 3. If IS3 Enable is set in the I/O Port Control/Status Register, an interrupt will occur. If the latch enable bit in the I/O Control Status Register is set, this strobe will latch the input data from another device when that device has indicated that it has valid data.

Output Strobe ($\overline{\text{OS3}}$) (SC2)

This signal is used by the processor to strobe an external device, indicating valid data is on the I/O pins. The timing for the Output Strobe is shown in Figure 7. I/O Port Control/Status Register is discussed in the following section.

The following pins are available in the Expanded Modes.

Read/Write (R/W) (SC2)

This TTL compatible output signals the peripherals and memory devices whether the MPU is in a Read (high) or a Write (low) state. The normal standby state of this signal is Read (high). This output is capable of driving one TTL load and 90 pF.

I/O Strobe ($\overline{IOS}$) (SC1)

In the expanded non-multiplexed mode of operation, $\overline{IOS}$ internally decodes A9 through A15 as zero's and A8 as a one. This allows external access of the 256 locations from $0100 to $01FF. The timing diagrams are shown as figures 3 and 4.

Address Strobe (AS) (SC1)

In the expanded multiplexed mode of operation address strobe is output on this pin. This signal is used to latch the 8 LSB's of address which are multiplexed with data on Port 3. An 8-bit latch is utilized in conjunction with Address Strobe, as shown in figure 29, Expanded Multiplexed Mode. Address Strobe signals the latch when it is time to latch the address lines so the lines can become data bus lines during the E pulse. The timing for this signal is shown in the MC6801 Bus Timing Figure 9. This signal is also used to disable the address from the multiplexed bus allowing a deselect time, T_{ASD} before the data is enabled to the bus.

MC6801 PORTS

There are four I/O ports on the MC6801 MCU; three 8-bit ports and one 5-bit port. There are two control lines associated with one of the 8-bit ports. Each port has an associated write only Data Direction Register which allows each I/O line to be programmed to act as an input or an output.* A "1" in the corresponding Data Direction Register bit will cause that I/O line to be an output. A "0" in the corresponding Data Direction Register bit will cause that I/O line to be an input. There are four ports: Port 1, Port 2, Port 3, and Port 4. Their addresses and the addresses of their Data Direction registers are given in Table 2.

*The only exception is bit 1 of Port 2, which can either be data input or Timer output.

TABLE 2 — PORT AND DATA DIRECTION REGISTER ADDRESSES

Ports	Port Address	Data Direction Register Address
I/O Port 1	$0002	$0000
I/O Port 2	$0003	$0001
I/O Port 3	$0006	$0004
I/O Port 4	$0007	$0005

I/O Port 1

This is an 8-bit port whose individual bits may be defined as inputs or outputs by the corresponding bit in its data direction register. The 8 output buffers have three-state capability, allowing them to enter a high impedance state when the peripheral data lines are used as inputs. In order to be read properly, the voltage on the input lines must be greater than 2.0 volts for a logic "1" and less than 0.8 volt for a logic "0". As outputs, these lines are TTL compatible and may also be used as a source of up to 1 mA at 1.5 volts to directly drive a Darlington base. After Reset, the I/O lines are configured as inputs. In all three modes, Port 1 is always parallel I/O.

I/O Port 2

This port has five lines that may be defined as inputs or outputs by its data direction register. The 5 output buffers have three-state capability, allowing them to enter a high impedance state when used as an input. In order to be read properly, the voltage on the input lines must be greater than 2.0 volts for a logic "1" and less than 0.8 volt for a logic "0". As outputs, this port has no internal pullup resistors but will drive TTL inputs directly. For driving CMOS inputs, external pullup resistors are required. After Reset, the I/O lines are configured as inputs. Three pins on Port 2 (pins 10, 9 and 8 of the chip) are used to program the mode of operation during reset. The values of these pins at reset are latched into the three MSB's (bits 7, 6, and 5) of Port 2 which are read only. This is explained in the Mode Selection Section.

In all three modes, Port 2 can be configured as I/O and provides access to the Serial Communications Interface and the Timer. Bit 1 is the only pin restricted to data input or Timer output.

I/O Port 3

This is an 8-bit port that can be configured as I/O, a data bus, or an address bus multiplexed with the data bus — depending on the mode of operation hardware programmed by the user at reset. As a data bus, Port 3 is bi-directional. As an input for peripherals, it must be supplied regular TTL levels, that is, greater than 2.0 volts for a logic "1" and less than 0.8 volt for a logic "0".

Its TTL compatible three-state output buffers are capable of driving one TTL load and 90 pf. In the Expanded Modes, after reset, the data direction register is inhibited and data flow depends on the state of the R/W line. The input strobe (IS3) and the output strobe (OS3) used for handshaking are explained later.

In the three modes Port 3 assumes the following characteristics:

Single Chip Mode: Parallel Inputs/Outputs as programmed by its associated Data Direction Register. There are two control lines associated with this port in this mode, an input strobe and an output strobe, that can be used for handshaking. They are controlled by the I/O Port Control/Status Register explained at the end of this section.

Expanded Non-Multiplexed Mode: In this mode Port 3 becomes the data bus (D7-D0).

Expanded Multiplexed Mode: In this mode Port 3 becomes both the data bus (D7-D0) and lower bits of the address bus (A7-A0). An address strobe output is true when the address is on the port.

I/O PORT 3 CONTROL/STATUS REGISTER

	7	6	5	4	3	2	1	0
	IS3	IS3	X	OSS	LATCH	X	X	X
$000F	FLAG	ENABLE			ENABLE			

Bit 0 Not used.
Bit 1 Not used.
Bit 2 Not used.
Bit 3 **Latch Enable.** This controls the input latch for I/O Port 3. If this bit is set high the input data will be latched with the falling edge of the Input Strobe, IS3. This bit is cleared by reset, or CPU Read Port 3.

Bit 4 **(OSS) Output Strobe Select.** This bit will select if the Output Strobe should be generated by a write to I O Port 3 or a read of I O Port 3. When this bit is cleared the strobe is generated by a read Port 3. When this bit is set the strobe is generated by a write Port 3.

Bit 5 Not used.

Bit 6 **IS3 ENABLE.** This bit will be the interrupt caused by IS3. When set to a low level the IS3 FLAG will be set by input strobe but the interrupt will not be generated. This bit is cleared by reset.

Bit 7 **IS3 FLAG.** This is a read only status bit that is set by the falling edge of the input strobe, IS3. It is cleared by a read of the Control Status Register followed by a read or write of I O Port 3. Reset will clear this bit.

I/O Port 4

This is an 8-bit port that can be configured as I/O or as address lines depending on the mode of operation. In order to be read properly, the voltage on the input lines must be greater than 2.0 volts for a logic "1" and less than 0.8 volt for a logic "0".

As outputs, each line is TTL compatible and can drive 1 TTL load and 90 pF. After reset, the lines are configured as inputs. To use the pins as addresses, therefore, they should be programmed as outputs. In the three modes, Port 4 assumes the following characteristics:

Single Chip Mode: Parallel Inputs/Outputs as programmed by its associated Data Direction Register.

Expanded Non-Multiplexed Mode: In this mode Port 4 is configured as the lower order address lines (A7-A0) by writing one s to the data direction register. When all eight address lines are not needed, the remaining lines, starting with the most significant bit, may be used as I O (inputs only).

Expanded Multiplexed Mode: In this mode Port 4 is configured as the high order address lines (A15-A8) by writing one's to the data direction register. When all eight address lines are not needed, the remaining lines, starting with the most significant bit, may be used as I/O (inputs only).

MODE SELECTION

The mode of operation that 6801 will operate in after Reset is determined by hardware that the user must wire on pins 10, 9, and 8 of the chip. These pins are the three LSB's (I O 2, I O 1, and I O 0 respectively) of Port 2. They are latched into programmed control bits PC2, PC1, and PC0 when reset goes high. I/O Port 2 Register is shown below.

	7	6	5	4	3	2	1	0
$0003	PC2	PC1	PC0	I/O 4	I/O 3	I/O 2	I/O 1	I/O 0

An example of external hardware that could be used in the Expanded Non-Multiplexed Mode is given in Figure 16. In the Expanded Non-Multiplexed Mode, pins 10, 9 and 8 are programmed Hi, Lo, Hi respectively as shown.

Couplers between the pins on Port 2 and the peripherals attached may be required. If the lines go to devices which require signals at power up differing from the signals needed to program the 6801's mode, couplers are necessary.

The MC14066B can be used to provide this isolation between the peripheral device and the MCU during reset. Figure 17 shows the logic diagram and truth table for the MC14066B. It is bidirectional and requires no external logic to determine the direction of the information flow. The logic shown insures that the data on the peripheral will not change before it is latched into the MCU and the MCU has started the reset sequence.

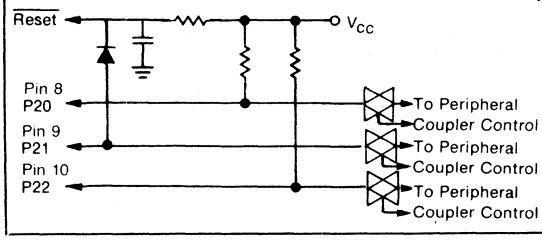
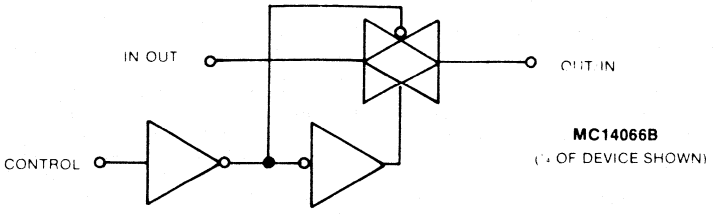


FIGURE 16 — DIODE CONFIGURATION FOR THE EXPANDED NON-MULTIPLEXED MODE

As bits 5, 6 and 7 of Port 2 are read only, the mode cannot be changed through software. The mode selections are shown in Table 3.

P20 refers to Port 2, bit 0.

FIGURE 17—MC14066B QUAD ANALOG, SWITCH/ MULTIPLEXER IN A TYPICAL MC6801 CIRCUIT



CONTROL	SWITCH
0	OFF
1	ON

V CONTROL	Vin TO Vout RESISTANCE
VSS	~ 10⁸ OHMS TYP
VDD	300 OHMS TYP

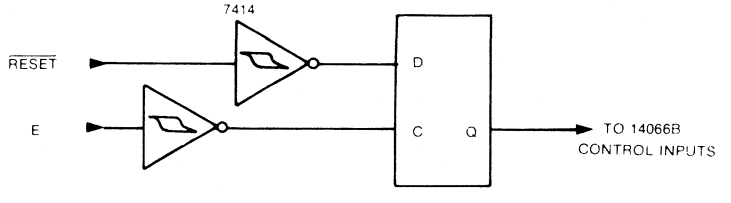
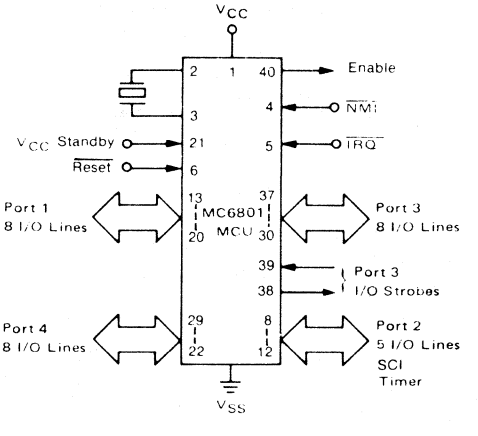


FIGURE 18 — MC6801 MCU SINGLE-CHIP MODE



MC6801 BASIC MODES

The MC6801 is capable of operating in three basic modes; (1) Single Chip Mode. (2) Expanded Multiplexed Mode (compatible with M6800 peripheral family) (3) Expanded Non-Multiplexed Mode.

SINGLE CHIP MODE

In the Single Chip Mode the Ports are configured for I/O.

This is shown in Figure 18 the single Chip Mode. In this mode, Port 3 will have two associated control lines, an input strobe and an output strobe for handshaking data.

EXPANDED NON-MULTIPLEXED MODE

In this mode the MC6801 will directly address M6800 peripherals with no external logic. In this mode Port 3 becomes the data bus. Port 4 becomes the A7-A0 address bus or partial address and I/O (inputs only). Port 2 can be parallel I/O, serial I/O, Timer, or any combination thereof. Port 1 is parallel I/O only. In this mode the MC6801 is expandable to 256 locations. The eight address lines associated with Port 4 may be substituted for I/O (inputs only) if a fewer number of address lines will satisfy the application. (See Figure 19).

FIGURE 19 — MC6801 MCU EXPANDED NON-MULTIPLEXED MODE

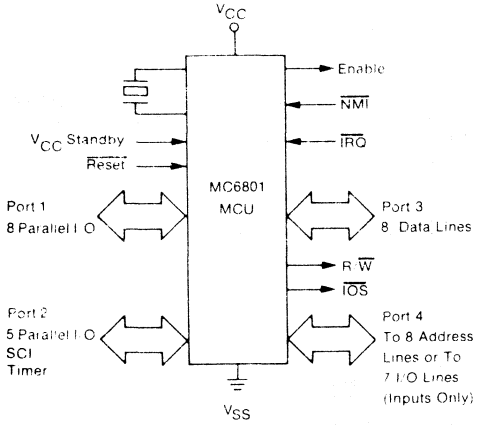
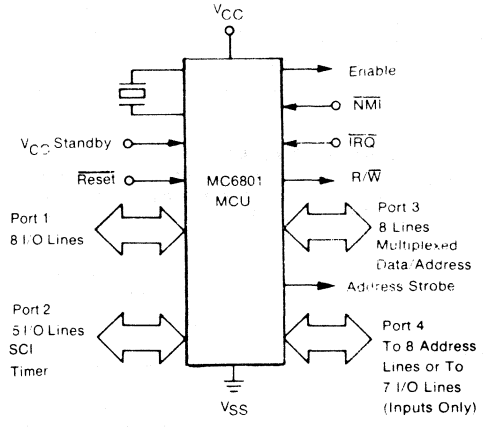


FIGURE 20 — MC6801 MCU EXPANDED MULTIPLEXED MODE



EXPANDED MULTIPLEXED MODE

In this mode Port 4 becomes higher order address lines with an alternative of substituting some of the address lines for I/O (inputs only). Port 3 is the data bus multiplexed with the lower order address lines differentiated by an output called Address Strobe. Port 2 is 5 lines of Parallel I/O, SCI, Timer, or any combination thereof. Port 1 is 8 Parallel I/O lines. In this mode it is expandable to 65K words. (See Figure 20)

TABLE 3 — MODE SELECTS

MODE		PROGRAM CONTROL			ROM	RAM	INTERRUPT VECTORS	BUS
7	SINGLE CHIP	Hi	Hi	Hi	I	I	I	I
6	EXPANDED MULTIPLEXED	Hi	Hi	Lo	I	I	I	Ep/M
5	EXPANDED NON-MULTIPLEXED	Hi	Lo	Hi	I	I	I	Ep
4	SINGLE CHIP TEST	Hi	Lo	Lo	I(2)	I(1)	I	I
3	64K ADDRESS I O	Lo	Hi	Hi	E	E	E	Ep/M
2	PORTS 3 & 4 EXTERNAL	Lo	Hi	Lo	E	I	E	Ep/M
1		Lo	Lo	Hi	I	I	E	Ep/M
0	TEST-DATA OUTPUTTED FROM ROM & RAM TO I O PORT 3	Lo	Lo	Lo	I	I	I*	Ep/M
		Pc2	**Pc1**	**Pco**				

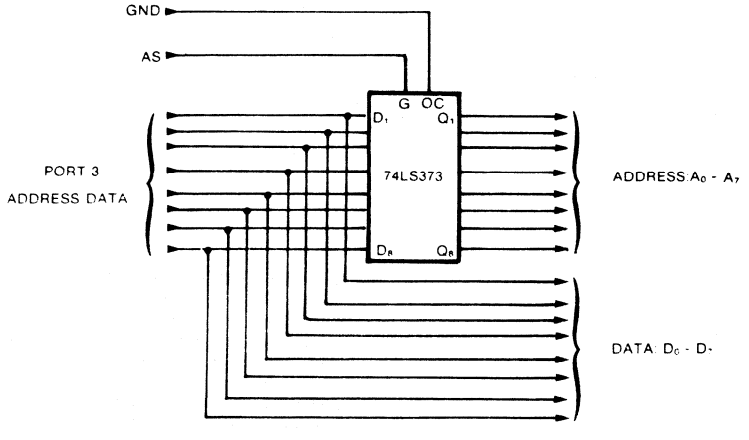
E — EXTERNAL all vectors are external
I — INTERNAL
Ep — EXPANDED
M — MULTIPLEXED

* First two addresses read from external after reset
(1) Address for RAM XX80-XXFF
(2) ROM disabled

Lower order Address Bus Latches

Since the data bus is multiplexed with the lower order address bus in Port 3, latches are required to latch those address bits. The SN74LS373 Transparent octal D-type latch can be used wtih the MC6801 to latch the least significant address byte. Figure 21 shows how to connect the latch to the MC6801. The output control to the LS373 may be connected to ground.

FIGURE 21 — LATCH CONNECTION



FUNCTION TABLE

OUTPUT CONTROL	ENABLE G	ENABLE D	OUTPUT Q
L	H	H	H
L	H	L	L
L	L	X	Q_0
H	X	X	Z

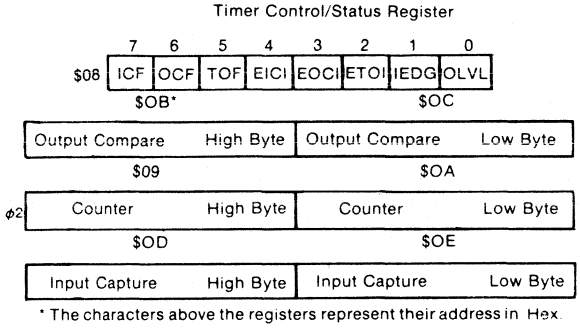
PROGRAMMABLE TIMER

The MC6801 contains an on-chip 16-bit programmable timer which may be used to perform measurements on an input waveform while independently generating an output waveform. Pulse widths for both input and output signals may vary from a few microseconds to many seconds. The timer hardware consists of

- an 8-bit control and status register,
- a 16-bit free running counter,
- a 16-bit output compare register, and
- a 16-bit input capture register

A block diagram of the timer registers is shown in Figure 22.

FIGURE 22 — BLOCK DIAGRAM OF TIMER REGISTERS



* The characters above the registers represent their address in Hex.

Free Running Counter ($0009:000A)

The key element in the programmable timer is a 16-bit free running counter which is driven to increasing values by the MPU ϕ. The counter value may be read by the MPU software at any time. The counter is cleared to zero on RESET and may be considered a read-only register with one exception. Any MPU write to the counter's address ($09) will always result in a preset value of $FFF8 being loaded into the counter regardless of the value involved in the write. This preset feature is intended for testing operation of the part, but may be of value in some applications.

Output Compare Register ($000B:000C)

The Output Compare Register is a 16-bit read/write register which is used to control an output waveform. The contents of this register are constantly compared with the current value of the free running counter. When a match is found, a flag is set (OCF) in the Timer Control and Status Register (TCSR) and the current value of the Output Level bit (OLVL) in the TCSR is clocked to the output level register. Providing the Data Direction Register for Port 2, Bit 1 contains a "1" (output), the output level register value will appear on the pin for Port 2 Bit 1. The values in the Output Compare Register and Output level bit may then be changed to control the output level on the next compare value. The Output Compare Register is set to $FFFF during RESET. The Compare function is inhibited for one cycle following a write to the high byte of the Output Compare Register to insure a valid 16-bit value is in the register before a compare is made.

Input Capture Register ($000D:000E)

The Input Capture Register is a 16-bit read-only register used to store the current value of the free running counter when the proper transition of an external input signal occurs. The input transition change required to trigger the counter transfer is controlled by the Input Edge bit (IEDG) in the TCSR. The Data Direction Register bit for Port 2 Bit 0, should* be clear (zero) in order to gate in the external input signal to the edge detect unit in the timer.

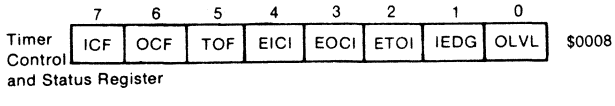
*With Port 2 Bit 0 configured as an output and set to "1", the external input will still be seen by the edge detect unit.

Timer Control and Status Register (TCSR) ($0008)

The Timer Control and Status Register consists of an 8-bit register of which all 8 bits are readable but only the low order 5 bits may be written. The upper three bits contain read-only timer status information and indicate that:

- a proper transition has taken place on the input pin with a subsequent transfer of the current counter value to the input capture register,
- a match has been found between the value in the free running counter and the output compare register, and
- when $0000 is in the free running counter.

Each of the flags may be enabled onto the MC6801 internal bus (IRQ2) with an individual Enable bit in the TCSR. If the I-bit in the MC6801 Condition Code register has been cleared, a priority vectored interrupt will occur corresponding to the flag bit(s) set. A description for each bit follows:



Bit 0 **OLVL** Output Level — This value is clocked to the output level register on an output compare. If the DDR for Port 2 bit 1 is set, the value will appear on the output pin.

Bit 1 **IEDG** Input Edge — This bit controls which transition of an input will trigger a transfer of the counter to the input capture register. The DDR for Port 2 Bit 0 must be clear for this function to operate.
IEDG = 0 Transfer takes place on a negative (high-to-low transition).
IEDG = 1 Transfer takes place on a positive edge (low-to-high transition).

Bit 2 **ETOI** Enable Timer Overflow Interrupt — When **set**, this bit enables IRQ2 to occur on the internal bus for a TOF interrupt; when **clear** the interrupt is inhibited.

Bit 3 **EOCI** Enable Output Compare Interrupt — When **set**, this bit enables IRQ2 to appear on the internal bus for an input capture interrupt; when **clear** the interrupt is inhibited.

Bit 4 **EICI** Enable Input Capture Interrupt — When **set**, this bit enables IRQ2 to occur on the internal bus for an input capture interrupt; when **clear** the interrupt is inhibited.

Bit 5 **TOF** Timer Overflow Flag — This read-only bit is **set** when the counter contains $0000. It is **cleared** by a read of the TCSR (with TOF set) followed by an MPU read of the Counter ($09).

Bit 6 **OCF** Output Compare Flag — This read-only bit is **set** when a match is found between the output compare register and the free running counter. It is **cleared** by a read of the TCSR (with OCF set) followed by an MPU write to the output compare register ($0B or $0C).

Bit 7 **ICF** Input Capture Flag — This read-only status bit is **set** by a proper transition on the input to the edge detect unit; it is **cleared** by a read of the TCSR (with ICF set) followed by an MPU read of the Input Capture Register ($0D).

SERIAL COMMUNICATIONS INTERFACE

The MC6801 contains a full-duplex asynchronous serial communications interface (SCI) on board. Two serial data formats (standard mark/space (NRZ) or Bi-phase) are provided at several different data rates. The controller comprises a transmitter and a receiver which operate independently or each other but in the same data format and at the same data rate. Both transmitter and receiver communicate with the MPU via the data bus and with the outside world via pins 2, 3, and 4 of Port 2. The hardware, software, and registers are explained in the following paragraphs.

Wake-Up Feature

In a typical multi-processor application, the software protocol will usually contain a destination address in the initial byte(s) of the message. In order to permit non-selected MPU's to ignore the remainder of the message, a wake-up feature is included whereby all further interrupt processing may be optionally inhibited until the beginning of the next message. When the next message appears, the hardware re-enables (or "wakes-up") the for the next message. The "wake-up" is automatically triggered by a string of ten consecutive 1's which indicates an idle transmit line. The software protocol must provide for the short idle period between any two consecutive messages.

Programmable Options

The following features of the MC6801 serial I/O section are programmable:

- format — standard mark/space (NRZ) or Bi-phase
- clock — external or internal
- baud rate — one of 4 per given MPU $\phi 2$ clock frequency or external clock X8 input
- wake-up feature — enabled or disabled
- interrupt requests — enabled or masked individually for transmitter and receiver data registers
- clock output — internal clock enabled or disabled to Port 2 (Bit 2)
- Port 2 (bits 3 and 4) — dedicated or not dedicated to serial I/O individually for transmitter and receiver.

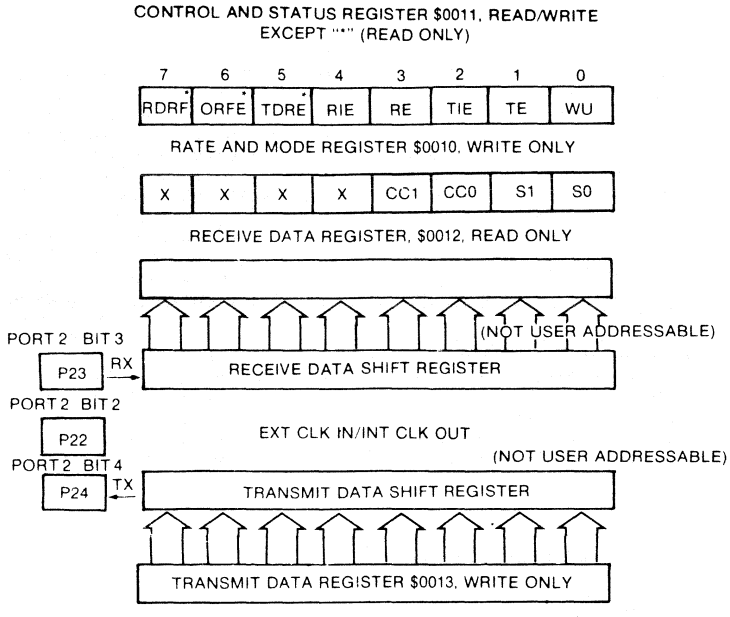
Serial Communications Hardware

The serial communications hardware is controlled by 4 registers as shown in Figure 23. The registers include:

- an 8-bit control and status register
- a 4-bit rate and mode control register (write only)
- an 8-bit read only receive data register and
- an 8-bit write only transmit data register.

In addition to the four registers, the serial I/O section utilizes bit 3 (serial input) and bit 4 (serial output) or Port 2. Bit 2 of Port 2 is utilized if the internal-clock-out or external-clock-in options are selected.

FIGURE 23 — SERIAL I/O REGISTERS



Transmit/Receive Control and Status (TRCS) Register

The TRCS register consists of an 8-bit register of which all 8 bits may be read while only bits 0-4 may be written. The register is initialized to $20 on $\overline{\text{RESET}}$. The bits in the TRCS register are defined as follows:

7	6	5	4	3	2	1	0	
RDRE	ORFE	TDRE	RIE	RE	TIE	TE	WU	ADDR: $0011

Bit 0 **WU** "Wake-up" on Next Message — set by MC6801 software cleared by hardware on receipt of ten consecutive 1's.

Bit 1 **TE** Transmit Enable — set by MC6801/MC68701 to produce preamble of nine consecutive 1's and to enable gating of transmitter output to Port 2, bit 4 regardless of the DDR value corresponding to this bit; when clear, serial I/O has no effect on Port 2 bit 4.

Bit 2 **TIE** Transmit Interrupt Enable — when set, will permit an $\overline{IRQ2}$ interrupt to occur when bit 5 (TDRE) is set; when clear, the TDRE value is masked from the bus.

Bit 3 **RE** Receiver Enable — when set, gates Port 2 bit 3 to input of receiver regardless of DDR value for this bit; when clear, serial I/O has no effect on Port 2 bit 3.

Bit 4 **RIE** Receiver Interrpt Enable — when set, will permit an $\overline{IRQ2}$ interrupt to occur when bit 7 (RDRF) or bit 6 (OR) is set; when clear, the interrupt is masked.

Bit 5 **TDRE** Transmit Data Register Empty — set by hardware when a transfer is made from the transmit data register to the output shift register. The TDRE bit is cleared by reading the status register, then writing a new byte into the transmit data register, TDRE is initialized to 1 by $\overline{RESET}$.

Bit 6 **ORFE** Over-Run-Framing Error — set by hardware when an overrun or framing error occurs (receive only). An overrun is defined as a new byte received with last byte still in Data Register/Buffer. A framing error has occurred when the byte boundaries in bit stream are not synchronized to bit counter. The ORFE bit is cleared by reading the status register, then reading the Receive Data Register, or by $\overline{RESET}$.

Bit 7 **RDRF** Receiver Data Register Full — Set by hardware when a transfer from the input shift register to the receiver data register is made. The RDRF bit is cleared by reading the status register, then reading the Receive Data Register, or by $\overline{RESET}$.

Rate and Mode Control Register

The Rate and Mode Control register controls the following serial I/O variables:

- Baud rate
- format
- clocking source, and
- Port 2 bit 2 configuration

The register consists of 4 bits all of which are write-only and cleared on $\overline{RESET}$. The 4 bits in the register may be considered as a pair of 2-bit fields. The two low order bits control the bit rate for internal clocking and the remaining two bits control the format and clock select logic. The register definition is as follows:

7	6	5	4	3	2	1	0	
X	X	X	X	CC1	CC0	S1	S0	ADDR:$0010

Bit 0 **S0**
Bit 1 **S1**
Speed Select — These bits select the Baud rate for the internal clock. The four rates which may be selected are a function of the MPU $\phi 2$ clock frequency. Table 4 lists the available Baud rates.

Bit 2 **CC0**
Bit 3 **CC1**
Clock Control and Format Select — this 2-bit field controls the format and clock select logic. Table 5 defines the bit field.

TABLE 4 — SCI INTERNAL BAUD RATES

S1,S0	XTAL	4.0 MHz	4.9152 MHz	2.5476 MHz
	$\phi 2$	1.0 MHz	1.2288 MHz	0.6144 MHz
00	$\phi 2 \div 16$	62.5k Bits/s	76.8k Bits/s	38.4k Bits/s
01	$\phi 2 \div 128$	7,812.5 Bits/s	9,600 Bits/s	4,800 Bits/s
10	$\phi 2 \div 1024$	976.6 Bits/s	1,200 Bits/s	600 Bits/s
11	$\phi 2 \div 4096$	244.1 Bits/s	300 Bits/s	150 Bits/s

TABLE 5 — BIT FIELD

CC1, CC0	Format	Clock Source	Port 2 Bit 2	Port 2 Bit 3	Port 2 Bit 4
00	Bi-Phase	Internal	Not Used	**	**
01	NRZ	Internal	Not Used	**	**
10	NRZ	Internal	Output*	Serial Input	Serial Output
11	NRZ	External	Input	Serial Input	Serial Output

*Clock output is available regardless of values for bits RE and TE.
**Bit 3 is used for serial input if RE = "1" in TRCS; bit 4 is used for serial output if TE = "1" in TRCS.

Internally Generated Clock

If the user wishes for the serial I/O to furnish a clock, the following requirements are applicable:

the values of RE and TE are immaterial.

- the values of RE and TE are immaterial.
- CC1, CC0 must be set to 10
- the maximum clock rate will be $\phi \div 16$.
- the clock will be at 1X the bit rate and will have a rising edge at mid-bit.

Externally Generated Clock

If the user wishes to provide an external clock for the serial I/O, the following requirements are applicable:

- the CC1, CC0, field in the Rate and Mode Control Register must be set to 11,
- the external clock must be set to 8 times (X8) the desired baud rate and
- the maximum external clock frequency is 1.3 MHZ.

SERIAL OPERATIONS

The serial I O hardware should be initialized by the MC6801 software prior to operation. This sequence will normally consist of:

- writing the desired operation control bits to the Rate and Mode Control Register and
- writing the desired operational control bits in the Transmit Receive Control and Status Register.

The Transmitter Enable (TE) and Receiver Enable (RE) bits may be left set for dedicated operations.

Transmit Operations

The transmit operation is enabled by the TE bit in the Transmit/Receive Control and Status Register. This bit when set, gates the output of the serial transmit shift register to Port 2 Bit 4 and takes unconditional control over the Data Direction Register value for Port 2, Bit 4.

Following a $\overline{\text{RESET}}$, the user should configure both the Rate and Mode Control Register and the Transmit-Receiver Control and Status Register for desired operation. Setting the TE bit during this procedure initiates the serial output by first transmitting a ten-bit preamble of 1's. Following the preamble, internal synchronization is established and the transmitter section is ready for operation.

At this point one of two situations exist:

a) if the Transmit Data Register is empty (TDRE = 1), a continuous string of ones will be sent indicating an idle line, or

b) if data has been loaded into the Transmit Data Register (TDRE = 0), the word is transferred to the output shift register and transmission of the data word will begin.

During the transfer itself, the 0 start bit is first transmitted. Then the 8 data bits (beginning with bit 0) followed by the stop bit, are transmitted. When the Transmitter Data Register has been emptied, the hardware sets the TDRE flag bit.

If the MC6801 fails to respond to the flag within the proper time, (TDRE is still set when the next normal transfer from the parallel data register to the serial output register should occur) then a 1 will be sent (instead of a 0) at "Start" bit time, followed by more 1's until more data is supplied to the data register. No 0's will be sent while TDRE remains a 1.

The Bi-phase mode operates as described above except that the serial output toggles each bit time, and on 1/2 bit times when a 1 is sent.

Receive Operation

The receive operation is enabled by the RE bit which gates in the serial input through Port 2 Bit 3. The receiver section operation is conditioned by the contents of the Transmit/Receive Control and Status Register and the Rate and Mode Control Register.

The receiver bit interval is divided into 8 sub-intervals for internal synchronization. In the standard, non-Bi-phase mode, the received bit stream is synchronized by the first 0 (space) encountered.

The approximate center of each bit time is strobed during the next 10 bits. If the tenth bit is not a 1 (stop bit) a framing error is assumed, and bit ORFE is set. If the tenth bit is a 1, the data is transferred to the Receiver Data Register, and interrupt flag RDRF is set. If RDRF is still set at the next tenth bit time, ORFE will be set, indicating an over-run has occurred. When the MC6801 responds to either flag (RDRF or ORFE) by reading the status register followed by reading the Data Register, RDRF (or ORFE) will be cleared.

RAM CONTROL REGISTER

This register, which is addressed at $0014, gives status information about the standby RAM. A 0 in the RAM enable bit (RAM E) will disable the standby RAM, thereby protecting it at power down if V_{CC} is held greater than V_{SBB} volts, as explained previously in the signal description for V_{CC} Standby.

$0014	STANDBY BIT		X	X	X	X	X	X
		RAME						

Bit 0 Not Used.
Bit 1 Not Used.
Bit 2 Not used.
Bit 3 Not used.
Bit 4 Not used.
Bit 5 Not used.
Bit 6 The RAM ENABLE control bit allows the user the ability to disable the standby RAM. This bit is set to a logic "one" by reset which enables the standby RAM and can be written to one or zero under program control. When the RAM is disabled, logic "zero", data is read from external memory.
Bit 7 The STANDBY BIT of the control register, $0014, is cleared when the standby voltage is removed. This bit is a read/write status flag that the user can read which indicates that the standby RAM voltage has been applied, and the data in the standby RAM is valid.

FIGURE 24 — MEMORY MAP

The MC6801 provides up to 65k bytes of memory for program and/or data storage. The memory map is shown in Figure 24.

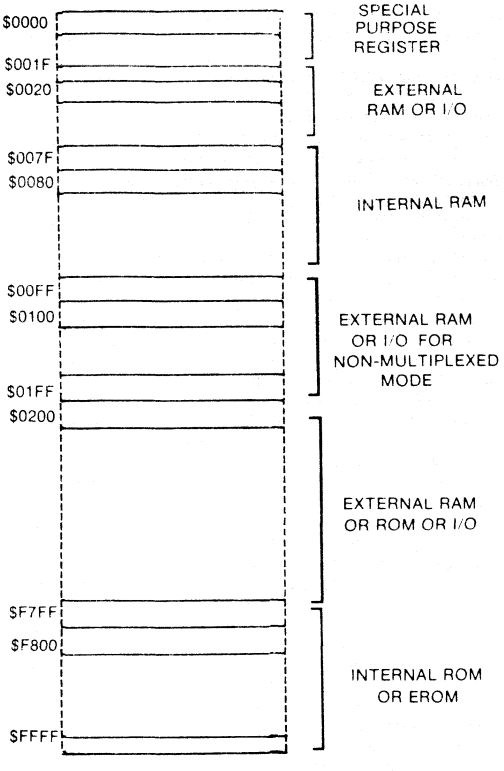


TABLE 6 — SPECIAL REGISTERS

The first 32 bytes are for the special purpose registers as shown in Table 6.

Hex Address	Register
00	Data Direction 1
01	Data Direction 2
02	I/O Port 1
03	I/O Port 2
04	Data Direction 3
05	Data Direction 4
06	I/O Port 3
07	I/O Port 4
08	TCSR
09	Counter High Byte
0A	Counter Low Byte
0B	Output Compare High Byte
0C	Output Compare Low Byte
0D	Input Capture High Byte
0E	Input Capture Low Byte
0F	I/O Port 3 C/S Register
10	Serial Rate and Mode Register
11	Serial Control and Status Register
12	Serial Receiver Data Register
13	Serial Transmit Data Register
14	RAM/EROM Control Register
15-1F	Reserved

FIGURE 25 — MEMORY MAP FOR INTERRUPT VECTORS

	Vector MS	Vector LS	Description
Highest Priority	FFFE	FFFF	Restart
	FFFC	FFFD	Non-Maskable Interrupt
	FFFA	FFFB	Software Interrupt
	FFF8	FFF9	IRQ1 Interrupt Strobe 3
	FFF6	FFF7	IRQ2 Timer Input Capture
	FFF4	FFF5	IRQ2 Timer Output Compare
	FFF2	FFF3	IRQ2 Timer Overflow
Lowest Priority	FFF0	FFF1	IRQ2 Serial I/O Interrupt

Locations $0020 through $007F access external RAM or I/O. Internal RAM is accessed at $0080 through $00FF. The RAM may be alternately selected by mask programming at location $A080. However, if the user desires to access external RAM at those locations he may do so by clearing the RAM ENABLE control bit of the RAM Control Register. In this way an extra 128 bytes of external RAM are available. The first 64 bytes of the 128 bytes of on-chip RAM are provided with a separate power supply. This will maintain the 64 bytes of RAM in the power down mode as explained in the pin description for Vcc Standby.

Locations $0100 through $01FF are available in the Expanded Non-Multiplexed Mode. The eight address lines of Port 4 make this 256 word expandability possible. Those not needed for address lines can be used as input lines instead.

The full range of addresses available to the user is in the Expanded Multiplexed Mode. Locations $0200 through $F7FF can be used as external RAM, external ROM, or I/O. Any higher order bits not required for addressing can be used as I/O as in the Expanded Non-Multiplexed Mode.

The internal ROM is located at $F800 through $FFFF. The decoder for the ROM may be mask programmed on A12, and A13 as zeros or one's to provide for $C800, $D800, $E800 for the ROM address. A12 and A13 may also be don't care in this decoder. The primary address for the ROM will be $F800.

GENERAL DESCRIPTION OF INSTRUCTION SET

The MC6801 is upward object code compatible with the MC6800 as it implements the full M6800 instruction set. The execution times of key instructions have been reduced to increase throughput. In addition, new instructions have been added; these include 16-bit operations and a hardware multiply.

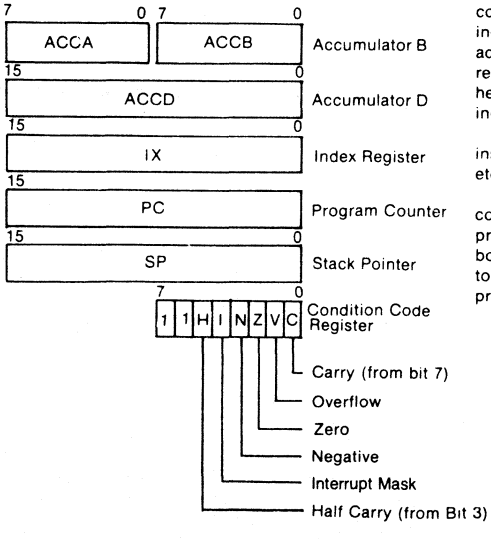
Included in the instruction set section are the following:

- MPU Programming Model (Figure 26)
- Addressing modes
- Accumulator and memory instructions — Table 7
- New instructions
- Index register and stack manipulations — Table 8
- Jump and branch instructions — Table 9
- Special operations — Figure 27
- Condition code register manipulation instructions — Table 10
- Instruction Execution times in machine cycles — Table 11
- Summary of cycle by cycle operation — Table 12

MPU PROGRAMMING MODEL

The programming model for the MC6801 is shown in Figure 26. The double (D) accumulator is physically the same as the A Accumulator concatenated with the B Accumulator so that any operation using accumulator D will destroy information in A and B.

FIGURE 26 — MCU PROGRAMMING MODEL



MPU ADDRESSING MODES

The MC6801 eight-bit microcomputer unit has seven address modes that can be used by a programmer, with the addressing mode a function of both the type of instruction and the coding within the instruction. A summary of the addressing modes for a particular instruction can be found in Table 11 along with the associated instruction execution time that is given in machine cycles. With a clock frequency of 4 MHz, these times would be microseconds.

Accumulator (ACCX) Addressing — In accumulator only addressing, either accumulator A or accumulator B is specified. These are one-byte instructions.

Immediate Addressing — In immediate addressing, the operand is contained in the second byte of the instruction except LDS and LDX which have the operand in the second and third bytes of the instruction. The MCU addresses this location when it fetches the immediate instruction for execution. These are two or three-byte instructions.

Direct Addressing — In direct addressing, the address of the operand is contained in the second byte of the instruction. Direct addressing allows the user to directly address the lowest 256 bytes in the machine i.e., locations zero through 255. Enhanced execution times are achieved by storing data in these locations. In most configurations, it should be a random access memory. These are two-byte instructions.

Extended Addressing — In extended addressing, the address contained in the second byte of the instruction is used as the higher eight-bits of the address of the operand. The third byte of the instruction is used as the lower eight-bits of the address for the operand. This is an absolute address in memory. These are three-byte instructions.

Indexed Addressing — In indexed addressing, the address contained in the second byte of the instruction is added to the index register's lowest eight bits in the MCU. The carry is then added to the higher order eight bits of the index register. This result is then used to address memory. The modified address is held in a temporary address register so there is no change to the index register. These are two-byte instructions.

Implied Addressing — In the implied addressing mode the instruction gives the address (i.e., stack pointer, index register, etc.). These are one-byte instructions.

Relative Addressing — In relative addressing, the address contained in the second byte of the instruction is added to the program counter's lowest eight bits plus two. The carry or borrow is then added to the high eight bits. This allows the user to address data within a range of -125 to +129 bytes of the present instruction. These are two-byte instructions.

TABLE 7—ACCUMULATOR & MEMORY INSTRUCTIONS

ACCUMULATOR AND MEMORY		ADDRESSING MODES																					
		IMMED.			DIRECT			INDEX			EXTEND			INHERENT			5	4	3	2	1	0
Operations	MNEMONIC	OP	~	#	OP	~	#	OP	~	#	OP	~	#	OP	~	#	Boolean/Arithmetic Operation	H	I	N	Z	V	C
Add	ADDA	8B	2	2	9B	3	2	AB	4	2	BB	4	3				A + M → A	↕	•	↕	↕	↕	↕
	ADDB	CB	2	2	DB	3	2	EB	4	2	FB	4	3				B + M → B	↕	•	↕	↕	↕	↕
Add Double	ADDD	C3	4	3	D3	5	2	E3	6	2	F3	6	3				A:B+M:M+1 → A:B	•	•	↕	↕	↕	↕
Add Accumulators	ABA													1B	2	1	A + B → A	↕	•	↕	↕	↕	↕
Add With Carry	ADCA	89	2	2	99	3	2	A9	4	2	B9	4	3				A + M + C → A		•	↕	↕	↕	↕
	ADCB	C9	2	2	D9	3	2	E9	4	2	F9	4	3				B + M + C → B		•	↕	↕	↕	↕
AND	ANDA	84	2	2	94	3	2	A4	4	2	B4	4	3				A M → A	•	•	↕	↕	R	•
	ANDB	C4	2	2	D4	3	2	E4	4	2	F4	4	3				B M → B	•	•	↕	↕	R	•
Bit Test	BIT A	85	2	2	95	3	2	A5	4	2	B5	4	3				A M	•	•	↕	↕	R	•
	BIT B	C5	2	2	D5	3	2	E5	4	2	F5	4	3				B M	•	•	↕	↕	R	•
Clear	CLR							6F	6	2	7F	6	3				00 → M	•	•	R	S	R	R
	CLRA													4F	2	1	00 → A	•	•	R	S	R	R
	CLRB													5F	2	1	00 → B	•	•	R	S	R	R
Compare	CMPA	81	2	2	91	3	2	A1	4	2	B1	4	3				A - M	•	•	↕	↕	↕	↕
	CMPB	C1	2	2	D1	3	2	E1	4	2	F1	4	3				B - M	•	•	↕	↕	↕	↕
Compare Accumulators	CBA													11	2	1	A - B	•	•	↕	↕	↕	↕
Complement, 1's	COM							63	6	2	73	6	3				M → M	•	•	↕	↕	R	S
	COMA													43	2	1	A → A	•	•	↕	↕	R	S
	COMB													53	2	1	B → B	•	•	↕	↕	R	S
Complement, 2's	NEG							60	6	2	70	6	3				00 - M → M	•	•	↕	↕	①	②
(Negate)	NEGA													40	2	1	00 - A → A	•	•	↕	↕	①	②
NEGB														50	2	1	00 - B → B	•	•	↕	↕	①	②
Decimal Adjust, A	DAA													19	2	1	Converts binary add of BCD characters into BCD format	•	•	↕	↕	↕	③
Decrement	DEC							6A	6	2	7A	6	3				M - 1 → M	•	•	↕	↕	④	•
	DECA													4A	2	1	A - 1 → A	•	•	↕	↕	④	•
	DECB													5A	2	1	B - 1 → B	•	•	↕	↕	④	•
Exclusive OR	EORA	88	2	2	98	3	2	A8	4	2	B8	4	3				A ⊕ M → A	•	•	↕	↕	R	•
	EORB	C8	2	2	D8	3	2	E8	4	2	F8	4	3				B ⊕ M → B	•	•	↕	↕	R	•
Increment	INC							6C	6	2	7C	6	3				M + 1 → M	•	•	↕	↕	⑤	•
	INCA													4C	2	1	A + 1 → A	•	•	↕	↕	⑤	•
	INCB													5C	2	1	B + 1 → B	•	•	↕	↕	⑤	•
Load Accumulator	LDAA	86	2	2	96	3	2	A6	4	2	B6	4	3				M → A	•	•	↕	↕	R	•
	LDAB	C6	2	2	D6	3	2	E6	4	2	F6	4	3				M → B	•	•	↕	↕	R	•
Load Double Accumulator	LDAD	CC	3	3	DC	4	2	EC	5	2	FC	5	3				M → A M + 1 → B	•	•	↕	↕	R	•

The Condition Code Register notes are listed after Table 10.

(Continued)

TABLE 7 — Continued

Operations	MNEMONIC	IMMED. OP	IMMED. ~	IMMED. #	DIRECT OP	DIRECT ~	DIRECT #	INDEX OP	INDEX ~	INDEX #	EXTEND OP	EXTEND ~	EXTEND #	INHERENT OP	INHERENT ~	INHERENT #	Boolean/ Arithmetic Operation	5 H	4 I	3 N	2 Z	1 V	0 C
Multiply Unsigned	MUL													3D	10	1	A X B → A:B	•	•	•	•	•	(13)
OR, Inclusive	ORAA	8A	2	2	9A	3	2	AA	4	2	BA	4	3				A + M → A	•	•	↕	↕	R	•
	ORAB	CA	2	2	DA	3	2	EA	4	2	FA	4	3				B + M → B	•	•	↕	↕	R	•
Push Data	PSHA													36	3	1	A → M_{sp}, SP - 1 → SP	•	•	•	•	•	•
	PSHB													37	3	1	B → M_{sp}, SP - 1 → SP	•	•	•	•	•	•
Pull Data	PULA													32	4	1	SP + 1 → SP, M_{sp} → A	•	•	•	•	•	•
	PULB													33	4	1	SP + 1 → SP, M_{sp} → B	•	•	•	•	•	•
Rotate Left	ROL							69	6	2	79	6	3				M, A, B ← C b_7 b_0	•	•	↕	↕	⑥	↕
	ROLA													49	2	1		•	•	↕	↕	⑥	↕
	ROLB													59	2	1		•	•	↕	↕	⑥	↕
Rotate Right	ROR							66	6	2	76	6	3				M, A, B → C b_7 b_0	•	•	↕	↕	⑥	↕
	RORA													46	2	1		•	•	↕	↕	⑥	↕
	RORB													56	2	1		•	•	↕	↕	⑥	↕
Shift Left Arithmetic	ASL							68	6	2	78	6	3				M, A, B ← C b_7 b_0 ← 0	•	•	↕	↕	⑥	↕
	ASLA													48	2	1		•	•	↕	↕	⑥	↕
	ASLB													58	2	1		•	•	↕	↕	⑥	↕
Double Shift Left, Arithmetic	ASLD													05	3	1	C ← ACC A / ACC B ← 0 ; A_7 A_0 B_7 B_0	•	•	↕	↕	⑥	↕
Shift Right Arithmetic	ASR							67	6	2	77	6	3				M, A, B → B_7 B_0 C	•	•	↕	↕	⑥	↕
	ASRA													47	2	1		•	•	↕	↕	⑥	↕
	ASRB													57	2	1		•	•	↕	↕	⑥	↕
Shift Right Logical	LSR							64	6	2	74	6	3				M, A, B → 0 B7 B0 C	•	•	↕	↕	⑥	↕
	LSRA													44	2	1		•	•	↕	↕	⑥	↕
	LSRB													54	2	1		•	•	↕	↕	↕	↕
Double Shift Right Logical	LSRD													04	3	1	0 → ACC A / ACCB → C ; A7 A0 B7 B0	•	•	R	↕	⑥	↕
Store Accumulator	STAA				97	3	2	A7	4	2	B7	4	3				A → M	•	•	↕	↕	R	•
	STAB				D7	3	2	E7	4	2	F7	4	3				B → M	•	•	↕	↕	R	•
Store Double Accumulator	STAD				DD	4	2	ED	5	2	FD	5	3				A → M B → M + 1	•	•	↕	↕	R	•
Subtract	SUBA	80	2	2	90	3	2	A0	4	2	B0	4	3				A - M → A	•	•	↕	↕	↕	↕
	SUBB	C0	2	2	D0	3	2	E0	4	2	F0	4	3				B - M → B	•	•	↕	↕	↕	↕
Double Subtract	SUBD	83	4	3	93	5	2	A3	6	2	B3	6	3				A:B - M:M + 1 → A:B	•	•	↕	↕	↕	↕
Subtract Accumulators	SBA													10	2	1	A - B → A	•	•	↕	↕	↕	↕
Subtract With Carry	SBCA	82	2	2	92	3	2	A2	4	2	B2	4	3				A - M - C → A	•	•	↕	↕	↕	↕
	SBCB	C2	2	2	D2	3	2	E2	4	2	F2	4	3				B - M - C → B	•	•	↕	↕	↕	↕
Transfer Accumulators	TAB													16	2	1	A → B	•	•	↕	↕	R	•
	TBA													17	2	1	B → A	•	•	↕	↕	R	•
Test Zero or Minus	TST							6D	6	2	7D	6	3				M - 00	•	•	↕	↕	RR	R
	TSTB													5D	2	1	B - 00	•	•	↕	↕	R	R

The Condition Code Register notes are listed after Table 10.

ADDED INSTRUCTIONS

In addition to the existing M6800 Instruction Set, the following new instructions are incorporated in the MC6801 Microcomputer.

ABX	Adds the 8-bit unsigned accumulator B to the 16-bit X-Register taking into account the possible carry out of the low order byte of the X-Register.	IX ← IX + ACCB
ADDD	Adds the double precision ACCD* to the double precision value M:M+1 and places the results in ACCD.	ACCD ← (ACCD) + (M:M+1)
ASLD	Shifts all bits of ACCAB one place to the left. Bit 0 is loaded with zero. The C bit is loaded from the most significant bit of ACCD.	
LDD	Loads the contents of double precision memory location into the double accumulator A:B. The condition codes are set according to the data.	ACCD ← (M:M+1)
LSRD	Shifts all bits of ACCD one place to the right. Bit 15 is loaded with zero. The C bit is loaded from the least significant bit to ACCD.	
MUL	Multiplies the 8 bits in accumulator A with the 8 bits in accumulator B to obtain a 16-bit unsigned number in A:B. ACCA contains MSB of result.	ACCD ← ACCA * ACCB
PSHX	The contents of the index register is pushed onto the stack at the address contained in the stack pointer. The stack pointer is decremented by 2.	↓(IXL), SP ← (SP) - 1 ↓(IXL), SP ← (SP) - 1
PULX	The index register is pulled from the stack beginning at the current address contained in the stack pointer +1. The stack pointer is incremented by 2 in total.	SP ← (SP) + 1; IXH SP ← (SP) + 1; IHL
STD	Stores the contents of double accumulator A:B in memory. The contents of ACCD remain unchanged.	M:M + 1 ← (ACCD)
SUBD	Subtracts the contents of M:M + 1 from the contents of double accumulator AB and places the result in ACCD.	ACCAB ← (ACCD) - (M:M + 1)

*ACCD is the 16 bit register (A:B) formed by concatenating the A and B accumulators. The A-accumulator is the most significant byte.

TABLE 8 — INDEX REGISTER AND STACK MANIPULATION INSTRUCTIONS

POINTER OPERATIONS	MNEMONIC	IMMED			DIRECT			INDEX			EXTND			IMPLIED			BOOLEAN/ARITHMETIC OPERATION	COND. CODE REG.					
		OP	~	#	OP	~	#	OP	~	#	OP	~	#	OP	~	#		5 H	4 I	3 N	2 Z	1 V	0 C
Compare Index Reg	CPX	8C	4	3	9C	5	2	AC	6	2	BC	6	3				$X_H - M, X_L - (M + 1)$	•	•	⑦	↕	⑧	•
Decrement Index Reg	DEX													09	3	1	X - 1 → X	•	•	•	↕	•	•
Decrement Stack Pntr	DES													34	3	1	SP - 1 → SP	•	•	•	•	•	•
Increment Index Reg	INX													08	3	1	X + 1 → X	•	•	•	↕	•	•
Increment Stack Pntr	INS													31	3	1	1 SP + 1 → SP	•	•	•	•	•	•
Load Index Reg	LDX	CE	3	3	DE	4	2	EE	5	2	FE	5	3				$M \rightarrow X_H, (M + 1) \rightarrow X_L$	•	•	⑤	↕	R	•
Load Stack Pntr	LDS	8E	3	3	9E	4	2	AE	5	2	BE	5	3				$M \rightarrow SP_H, (M + 1) \rightarrow SP_L$	•	•	⑤	↕	R	•
Store Index Reg	STX				DF	4	2	EF	5	2	FF	5	3				$X_H \rightarrow M, X_L \rightarrow (M + 1)$	•	•	⑤	↕	R	•
Store Stack Pntr	STS				9F	5	2	AF	7	2	BF	6	3				$SP_H \rightarrow M, SP_L \rightarrow (M + 1)$	•	•	⑤	↕	R	•
Index Reg → Stack Pntr	TXS													35	3	1	X - 1 → SP	•	•	•	•	•	•
Stack Pntr → Index Reg	TSX													30	3	1	SP + 1 → X	•	•	•	•	•	•
Add	ABX													3A	3	1	B + X → X	•	•	•	•	•	•
Push Data	PSHX													3C	4	1	$X_L \rightarrow M_{SP}, SP - 1 \rightarrow SP$	•	•	•	•	•	•
																	$X_H \rightarrow H_{SP}, SP - 1 \rightarrow SP$						
Pull Data	PULX													38	5	1	$SP + 1 \rightarrow SP, M_{SP} \rightarrow X_H$	•	•	•	•	•	•
																	$SP + 1 \rightarrow SP, M_{SP} \rightarrow X_L$						

The Condition Code Register notes are listed after Table 10.

TABLE 9 — JUMP AND BRANCH INSTRUCTIONS

		RELATIVE			INDEX			EXTND			IMPLIED				COND. CODE REG.					
															5	4	3	2	1	0
OPERATIONS	MNEMONIC	OP	~	=	OP	~	=	OP	~	=	OP	~	=	BRANCH TEST	H	I	N	Z	V	C
Branch Always	BRA	20	4	2										None	•	•	•	•	•	•
Branch If Carry Clear	BCC	24	4	2										C = 0	•	•	•	•	•	•
Branch If Carry Set	BCS	25	4	2										C = 1	•	•	•	•	•	•
Branch If = Zero	BEQ	27	4	2										Z = 1	•	•	•	•	•	•
Branch If ≥ Zero	BGE	2C	4	2										N ⊕ V = 0	•	•	•	•	•	•
Branch If > Zero	BGT	2E	4	2										Z + (N ⊕ V) = 0	•	•	•	•	•	•
Branch If Higher	BHI	22	4	2										C + Z = 0	•	•	•	•	•	•
Branch If ≤ Zero	BLE	2F	4	2										Z + (N ⊕ V) = 1	•	•	•	•	•	•
Branch If Lower Or Same	BLS	23	4	2										C + Z = 1	•	•	•	•	•	•
Branch If < Zero	BLT	2D	4	2										N ⊕ V = 1	•	•	•	•	•	•
Branch If Minus	BMI	2B	4	2										N = 1	•	•	•	•	•	•
Branch If Not Equal Zero	BNE	26	4	2										Z = 0	•	•	•	•	•	•
Branch If Overflow Clear	BVC	28	4	2										V = 0	•	•	•	•	•	•
Branch If Overflow Set	BVS	29	4	2										V = 1	•	•	•	•	•	•
Branch If Plus	BPL	2A	4	2										N = 0	•	•	•	•	•	•
Branch To Subroutine	BSR	8D	8	2											•	•	•	•	•	•
Jump	JMP				6E	4	2	7E	3	3				See Special Operations	•	•	•	•	•	•
Jump To Subroutine	JSR				AD	8	2	BD	9	3					•	•	•	•	•	•
No Operation	NOP										01	2	1	Advances Prog. Cntr. Only	•	•	•	•	•	•
Return From Interrupt	RTI										3B	10	1		———			(10)	———	
Return From Subroutine	RTS										39	5	1		•	•	•	•	•	•
Software Interrupt	SWI										3F	12	1	See Special Operations	•	•	•	•	•	•
Wait for Interrupt*	WAI										3E	9	1		•	(11)	•	•	•	•

TABLE 10 — CONDITION CODE REGISTER MANIPULATION INSTRUCTIONS

		IMPLIED				COND. CODE REG.					
						5	4	3	2	1	0
OPERATIONS	MNEMONIC	OP	~	=	BOOLEAN OPERATION	H	I	N	Z	V	C
Clear Carry	CLC	0C	2	1	0 → C	•	•	•	•	•	R
Clear Interrupt Mask	CLI	0E	2	1	0 → I	•	R	•	•	•	•
Clear Overflow	CLV	0A	2	1	0 → V	•	•	•	•	R	•
Set Carry	SEC	0D	2	1	1 → C	•	•	•	•	•	S
Set Interrupt Mask	SEI	0F	2	1	1 → I	•	S	•	•	•	•
Set Overflow	SEV	0B	2	1	1 → V	•	•	•	•	S	•
Accumulator A → CCR	TAP	06	2	1	A → CCR	———			(12)	———	
CCR → Accumulator A	TPA	07	2	1	CCR → A	•	•	•	•	•	•

CONDITION CODE REGISTER NOTES: (Bit set if test is true and cleared otherwise)

1 (Bit V) Test: Result = 10000000?
2 (Bit C) Test: Result = 00000000?
3 (Bit C) Test: Decimal value of most significant BCD Character greater than nine? (Not cleared if previously set.)
4 (Bit V) Test: Operand = 10000000 prior to execution?
5 (Bit V) Test: Operand = 01111111 prior to execution?
6 (Bit V) Test: Set equal to result of N⊕C after shift has occurred.
7 (Bit N) Test: Sign bit of most significant (MS) byte = 1?
8 (Bit V) Test: 2's complement overflow from subtraction of MS bytes?
9 (Bit N) Test: Result less than zero? (Bit 15 = 1)
10 (All) Load Condition Code Register from Stack. (See Special Operations)
11 (Bit I) Set when interrupt occurs. If previously set, a Non-Maskable Interrupt is required to exit the wait state.
12 (All) Set according to the contents of Accumulator A.

FIGURE 27 — SPECIAL OPERATIONS

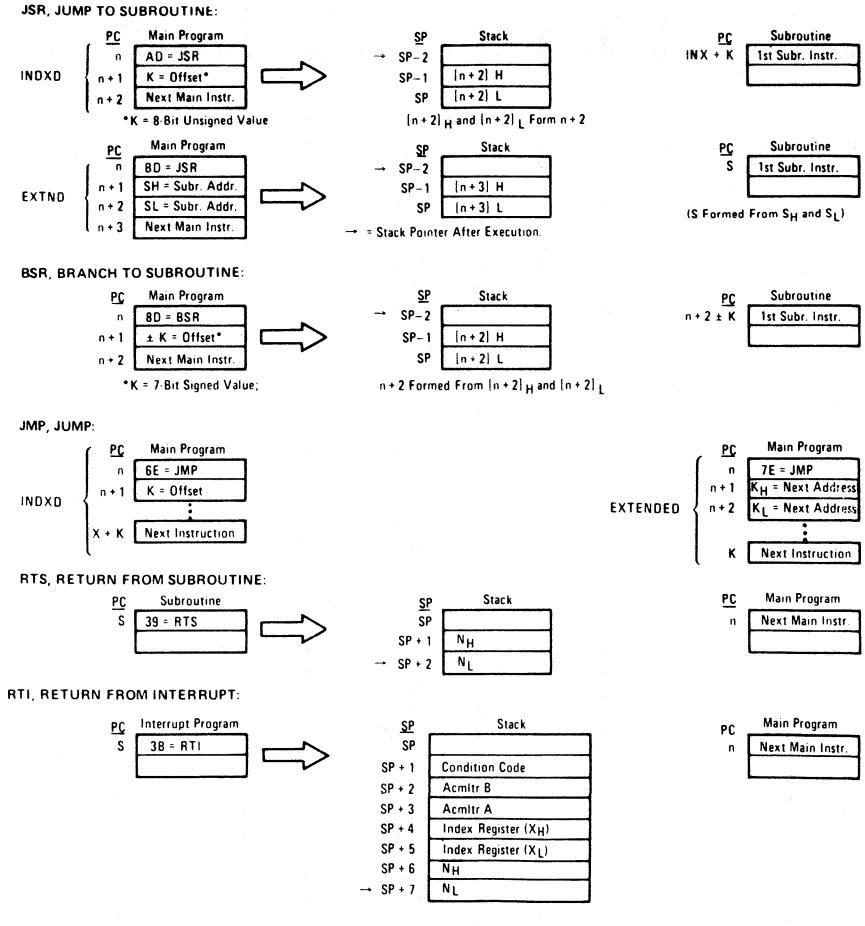


TABLE 11 — INSTRUCTION EXECUTION TIMES IN MACHINE CYCLE

	ACCX	Immediate	Direct	Extended	Indexed	Inherent	Relative		ACCX	Immediate	Direct	Extended	Indexed	Inherent	Relative
ABA	•	•	•	•	•	2	•	INX	•	•	•	•	•	3	•
ABX	•	•	•	•	•	3	•	JMP	•	•	•	3	3	•	•
ADC	•	2	3	4	4	•	•	JSR	•	•	5	6	6	•	•
ADD	•	2	3	4	4	•	•	LDA	•	2	3	4	4	•	•
ADDD	•	4	5	6	6	•	•	LDD	•	3	4	5	5	•	•
AND	•	2	3	4	4	•	•	LDS	•	3	4	5	5	•	•
ASL	2	•	•	6	6	•	•	LDX	•	3	4	5	5	•	•
ASLD	•	•	•	•	•	3	•	LSR	2	•	•	6	6	•	•
ASR	2	•	•	6	6	•	•	LSRD	•	•	•	•	•	3	•
BCC	•	•	•	•	•	•	3	MUL	•	•	•	•	•	10	•
BCS	•	•	•	•	•	•	3	NEG	2	•	•	6	6	•	•
BEQ	•	•	•	•	•	•	3	NOP	•	•	•	•	•	2	•
BGE	•	•	•	•	•	•	3	ORA	•	2	3	4	4	•	•
BGT	•	•	•	•	•	•	3	PSH	3	•	•	•	•	•	•
BHI	•	•	•	•	•	•	3	PSHX	•	•	•	•	•	4	•
BIT	•	2	3	4	4	•	•	PUL	4	•	•	•	•	•	•
BLE	•	•	•	•	•	•	3	PULX	•	•	•	•	•	5	•
BLS	•	•	•	•	•	•	3	ROL	2	•	•	6	6	•	•
BLT	•	•	•	•	•	•	3	ROR	2	•	•	6	6	•	•
BMI	•	•	•	•	•	•	3	RTI	•	•	•	•	•	10	•
BNE	•	•	•	•	•	•	3	RTS	•	•	•	•	•	5	•
BPL	•	•	•	•	•	•	3	SBA	•	•	•	•	•	2	•
BRA	•	•	•	•	•	•	3	SBC	•	2	3	4	4	•	•
BSR	•	•	•	•	•	•	6	SEC	•	•	•	•	•	2	•
BVC	•	•	•	•	•	•	3	SEI	•	•	•	•	•	2	•
BVS	•	•	•	•	•	•	3	SEV	•	•	•	•	•	2	•
CBA	•	•	•	•	•	2	•	STA	•	•	3	4	4	•	•
CLC	•	•	•	•	•	2	•	STD	•	•	4	5	5	•	•
CLI	•	•	•	•	•	2	•	STS	•	•	4	5	5	•	•
CLR	2	•	•	6	6	•	•	STX	•	•	4	5	5	•	•
CLV	•	•	•	•	•	2	•	SUB	•	2	3	4	4	•	•
CMP	•	2	3	4	4	•	•	SUBD	•	4	5	6	6	•	•
COM	2	•	•	6	6	•	•	SWI	•	•	•	•	•	12	•
CPX	•	4	5	6	6	•	•	TAB	•	•	•	•	•	2	•
DAA	•	•	•	•	•	2	•	TAP	•	•	•	•	•	2	•
DEC	2	•	•	6	6	•	•	TBA	•	•	•	•	•	2	•
DES	•	•	•	•	•	3	•	TPA	•	•	•	•	•	2	•
DEX	•	•	•	•	•	3	•	TST	2	•	•	6	6	•	•
EOR	•	2	3	4	4	•	•	TSX	•	•	•	•	•	3	•
INC	2	•	•	6	6	•	•	TXS	•	•	•	•	•	3	•
INS	•	•	•	•	•	3	•	WAI	•	•	•	•	•	9	•

Summary of Cycle by Cycle Operation

Table 12 provides a detailed description of the information present on the Address Bus, Data Bus, and the Read/Write line (R/W) during each cycle for each instruction.

This information is useful in comparing actual with expected results during debug of both software and hardware as the control program is executed. The information is categorized in groups according to addressing mode and number of cycles per instruction. (In general, instructions with the same addressing mode and number of cycles execute in the same manner; exceptions are indicated in the table).

TABLE 12 — CYCLE BY CYCLE OPERATION

ADDRESS MODE & INSTRUCTIONS	CYCLE	CYCLE #	ADDRESS BUS	R/W LINE	DATA BUS
IMMEDIATE					
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	2	1 2	Op Code Address Op Code Address + 1	1 1	Op Code Operand Data
LDS LDX	3	1 2 3	Op Code Address Op Code Address + 1 Op Code Address + 2	1 1 1	Op Code Operand Data (High Order Byte) Operand Data (Low Order Byte)
CPX SUBD ADDD	4	1 2 3 4	Op Code Address Op Code Address + 1 Op Code Address + 2 Address Bus FFFF	1 1 1 1	Op Code Operand Data (High Order Byte) Operand Data (Low Order Byte) Low Byte of Restart Vector
DIRECT					
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	3	1 2 3	Op Code Address Op Code Address + 1 Address of Operand	1 1 1	Op Code Address of Operand Operand Data
STA	3	1 2 3	Op Code Address Op Code Address + 1 Destination Address	1 1 0	Op Code Destination Address Data from Accumulator
LDS LDX LDD	4	1 2 3 4	Op Code Address Op Code Address + 1 Address of Operand Operand Address + 1	1 1 1 1	Op Code Address of Operand Operand Data (High Order Byte) Operand Data (Low Order Byte)
STS STX STD	4	1 2 3 4	Op Code Address Op Code Address + 1 Address of Operand Address of Operand + 1	1 1 0 0	Op Code Address of Operand Register Data (High Order Byte) Register Data (Low Order Byte)
CPX SUBD ADDD	5	1 2 3 4 5	Op Code Address Op Code Address + 1 Operand Address Operand Address + 1 Address Bus FFFF	1 1 1 1 1	Op Code Address of Operand Operand Data (High Order Byte) Operand Data (Low Order Byte) Low Byte of Restart Vector
JSR	5	1 2 3 4 5	Op Code Address Op Code Address + 1 Subroutine Address Stack Pointer Stack Pointer + 1	1 1 1 0 0	Op Code Irrelevant Data First Subroutine Op Code Return Address (Low Order Byte) Return Address (High Order Byte)

(continued)

TABLE 12 — CYCLE BY CYCLE OPERATION
(cont)

ADDRESS MODE & INSTRUCTIONS	CYCLES	CYCLE #	ADDRESS BUS	R/W LINE	DATA BUS
INDEXED					
JMP	3	1	Op Code Address	1	Op Code
		2	Op Code Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	4	1	Op Code Address	1	Op Code
		2	Op Code Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register Plus Offset	1	Operand Data
STA	4	1	Op Code Address	1	Op Code
		2	Op Code Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register Plus Offset	0	Operand Data
LDS LDX LDD	5	1	Op Code Address	1	Op Code
		2	Op Code Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register Plus Offset	1	Operand Data (High Order Byte)
		5	Index Register Plus Offset + 1	1	Operand Data (Low Order Byte)
STS STX STD	5	1	Op Code Address	1	Op Code
		2	Op Code Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register Plus Offset	0	Operand Data (High Order Byte)
		5	Index Register Plus Offset + 1	0	Operand Data (Low Order Byte)
ASL LSR ASR NEG CLR ROL COM ROR DEC TST (1) INC	6	1	Op Code Address	1	Op Code
		2	Op Code Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register Plus Offset	1	Current Operand Data
		5	Address Bus FFFF	1	Low Byte of Restart Vector
		6	Index Register Plus Offset	0	New Operand Data
CPX SUBD ADDD	6	1	Op Code Address	1	Op Code
		2	Op Code Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register + Offset	1	Operand Data (High Order Byte)
		5	Index Register + Offset + 1	1	Operand Data (Low Order Byte)
		6	Address Bus FFFF		Low Byte of Restart Vector
JSR	6	1	Op Code Address	1	Op Code
		2	Op Code Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register + Offset	1	First Subroutine Op Code
		5	Stack Pointer	0	Return Address (Low Order Byte)
		6	Stack Pointer - 1	0	Return Address (High Order Byte)

(continued)

TABLE 12 — CYCLE BY CYCLE OPERATION
(cont)

ADDRESS MODE & INSTRUCTIONS	CYCLES	CYCLE #	ADDRESS BUS	R/W LINE	DATA BUS
EXTENDED					
JMP	3	1	Op Code Address	1	Op Code
		2	Op Code Address + 1	1	Jump Address (High Order Byte)
		3	Op Code Address + 2	1	Jump Address (Low Order Byte)
ADC EOR	4	1	Op Code Address	1	Op Code
ADD LDA		2	Op Code Address + 1	1	Address of Operand
AND ORA		3	Op Code Address + 2	1	Address of Operand (Low Order Byte)
BIT SBC CMP SUB		4	Address of Operand	1	Operand Data
STA A	4	1	Op Code Address	1	Op Code
STA B		2	Op Code Address + 1	1	Destination Address (High Order Byte)
		3	Op Code Address + 2	1	Destination Address (Low Order Byte)
		4	Operand Destination Address	0	Data from Accumulator
LDS	5	1	Op Code Address	1	Op Code
LDX		2	Op Code Address + 1	1	Address of Operand (High Order Byte)
LDD		3	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	Address of Operand	1	Operand Data (High Order Byte)
		5	Address of Operand + 1	1	Operand Data (Low Order Byte)
STS	5	1	Op Code Address	1	Op Code
STX		2	Op Code Address + 1	1	Address of Operand (High Order Byte)
STD		3	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	Address of Operand	0	Operand Data (High Order Byte)
		5	Address of Operand + 1	0	Operand Data (Low Order Byte)
ASL LSR	6	1	Op Code Address	1	Op Code
ASR NEG		2	Op Code Address + 1	1	Address of Operand (High Order Byte)
CLR ROL		3	Op Code Address + 2	1	Address of Operand (Low Order Byte)
COM ROR		4	Address of Operand	1	Current Operand Data
DEC TST (1)		5	Address Bus FFFF	1	Low Byte of Restart Vector
INC		6	Address of Operand	0	New Operand Data
CPX	6	1	Op Code Address	1	Op Code
SUBD		2	Op Code Address + 1	1	Operand Address (High Order Byte)
ADDD		3	Op code Address + 2	1	Operand Address (Low Order Byte)
		4	Operand Address	1	Operand Data (High Order Byte)
		5	Operand Address + 1	1	Operand Data (Low Order Byte)
		6	Address Bus FFFF	1	Low Byte of Restart Vector
JSR	6	1	Op Code Address	1	Op Code
		2	Op Code Address + 1	1	Address of Subroutine (High Order Byte)
		3	Op Code Address + 2	1	Address of Subroutine (Low Order Byte)
		4	Subroutine Starting Address	1	Op Code of Next Instruction
		5	Stack Pointer	0	Return Address (Low Order Byte)
		6	Stack Pointer - 1	0	Address of Operand (High Order Byte)

(continued)

TABLE 12 — CYCLE BY CYCLE OPERATION
(cont)

ADDRESS MODE & INSTRUCTIONS	CYCLES	CYCLES #	ADDRESS BUS	R/W LINE	DATA BUS
INHERENT					
ABA DAA SEC ASL DEC SEI ASR INC SEV CBA LSR TAB CLC NEG TAP CLI NOP TBA CLR ROL TPA CLV ROR TST COM SBA	2	1 2	Op Code Address Op Code Address +1	1 1	Op Code Op Code of Next Instruction
ABX	3	1 2 3	Op Code Address Op Code Address +1 Address Bus FFFF	1 1 1	Op Code Irrelevent Data Low Byte of Restart Vector
ASLD LSRD	3	1 2 3	Op Code Address Op Code Address +1 Address Bus FFFF	1 1 1	Op Code Irrevelant Data Low Byte of Restart Vector
DES INS	3	1 2 3	Op Code Address Op Code Address +1 Previous Register Contents	1 1 1	Op Code Op Code of Next Instruction Irrelevant Data
INX DEX	3	1 2 3	Op Code Address Op Code Address +1 Address Bus FFFF	1 1 1	Op Code Op Code of Next Instruction Low Byte of Restart Vector
PSHA PSHB	3	1 2 3	Op Code Address Op Code Address +1 Stack Pointer	1 1 0	Op Code Op Code of Next Instruction Accumulator Data
ISX	3	1 2 3	Op Code Address Op Code Address +1 Stack Pointer	1 1 1	Op Code Op Code of Next Instruction Irrelevant Data
TXS	3	1 2 3	Op Code Address Op Code Address +1 Address Bus FFFF	1 1 1	Op Code Op Code of Next Instruction Low Byte of Restart Vector
PULA PULB	4	1 2 3 4	Op Code Address Op Code Address +1 Stack Pointer Stack Pointer +1	1 1 1 1	Op Code Op Code of Next Instruction Irrelevant Data
PSHX	4	1 2 3 4	Op Code Address Op Code Address +1 Stack Pointer Stack Pointer -1	1 1 0 0	Op Code Irrelevant Data Index Register (Low Order Byte) Index Register (High Order Byte)
PULX 5	5	1 2 3 4 5	Op Code Address Op Code Address +1 Stack Pointer Stack Pointer +1 Stack Pointer +2	1 1 1 1 1	Op Code Irrelevant Data Irrelevant Data Index Register (High Order Byte) Index Register (Low Order Byte)
BCC BHT BNE BCS BLE BPL BEQ BLS BRA BGE BLT BVC BGT BMT BVS	3	1 2 3	Op Code Address Op Code Address +1 Address Bus FFFF	1 1 1	Op Code Branch Offset Low Byte of Restart Vector
BSR	6	1 2 3 4 5 6	Op Code Address Op Code Address +1 Address Bus FFFF Subroutine Starting Address Stack Pointer Stack Pointer -1	1 1 1 1 0 0	Op Code Branch Offset Low Byte of Restart Vector Op Code of Next Instruction Return Address (Low Order Byte) Return Address (High Order Byte)

FIGURE 28 — MC6801 MCU SINGLE-CHIP DUAL PROCESSOR CONFIGURATION

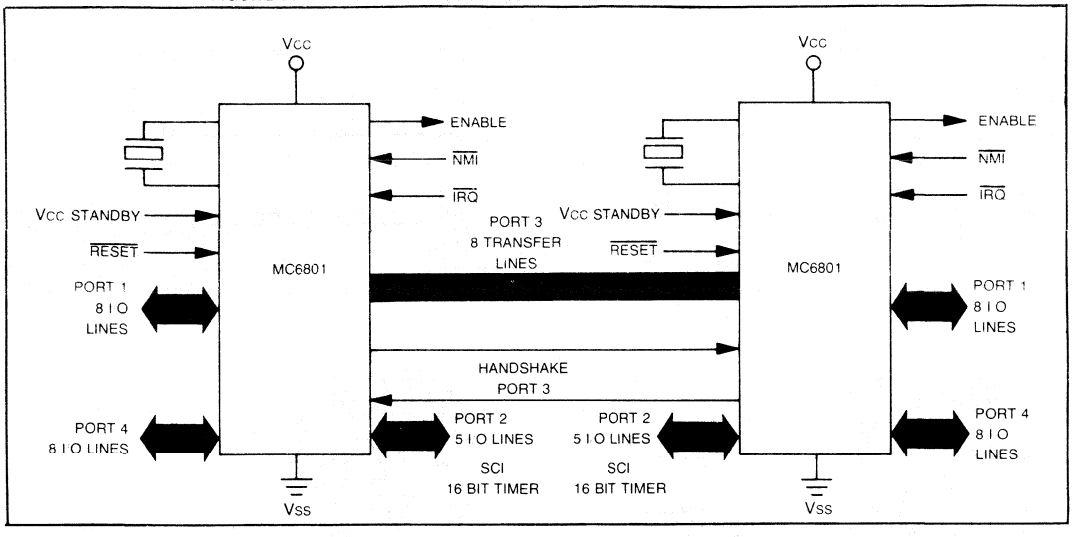


FIGURE 29 — MC6801 MCU
EXPANDED NON-MULTIPLEXED MODE

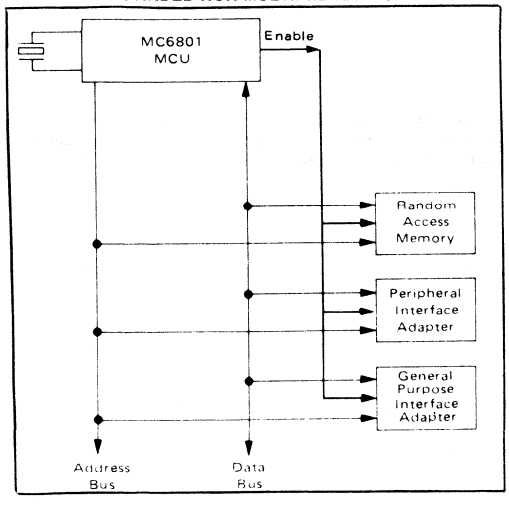
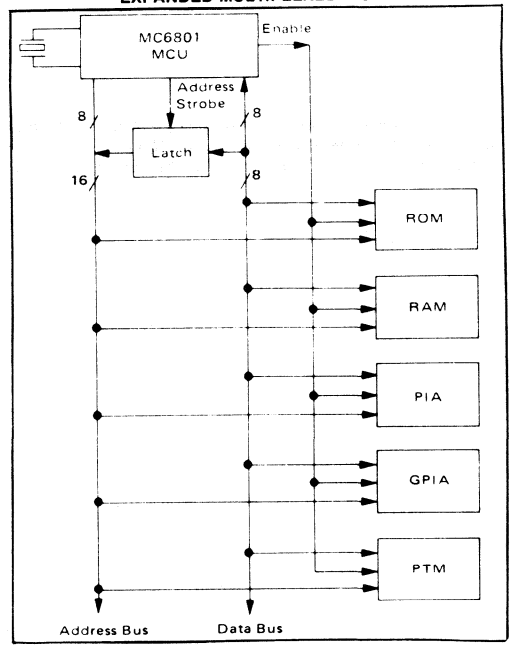
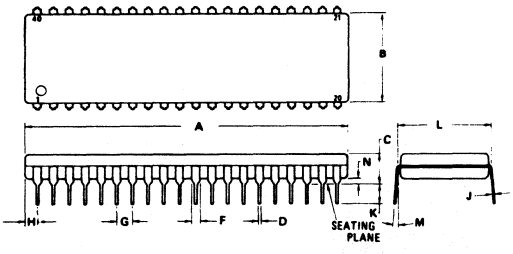


FIGURE 30 — MC6801 MCU
EXPANDED MULTIPLEXED MODE



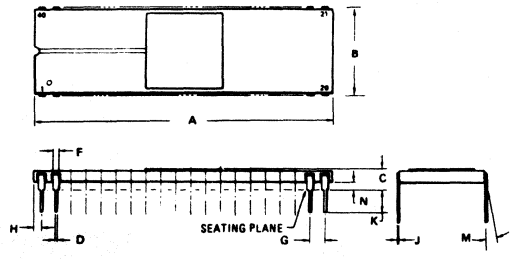
OUTLINE DIMENSIONS



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	50.29	51.31	1.980	2.020
B	14.86	15.62	0.585	0.615
C	2.54	4.19	0.100	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.33	0.008	0.013
K	2.54	4.19	0.100	0.165
M	0°	10°	0°	10°
N	0.51	1.52	0.020	0.060

L SUFFIX
CERAMIC PACKAGE
CASE 715-02

NOTE:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA (AT SEATING PLANE), AT MAX. MAT'L CONDITION.



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	51.82	52.32	2.040	2.060
B	13.72	14.22	0.540	0.560
C	4.57	5.08	0.180	0.200
D	0.36	0.51	0.014	0.020
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	3.05	3.56	0.120	0.140
L	15.24 BSC		0.600 BSC	
M	0°	10°	0°	10°
N	0.51	1.02	0.020	0.040

P SUFFIX
PLASTIC PACKAGE
CASE 711-02

NOTES:
1. LEADS TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION (DIM "D").
2. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.

MOTOROLA

MC6801L1
MC6801P1

Product Preview

MOS

(N-CHANNEL, SILICON-GATE DEPLETION LOAD)

MICROCOMPUTER EVALUATION CHIP

MICROCOMPUTER EVALUATION CHIP

FEATURES

- Three Modes of Operation – Single Chip, Expanded Non-Multiplexed, and Expanded Multiplexed
- Internal 2K ROM Programmed with a Debug Monitor Program
- Communicates with a Terminal Through the Serial Port

DESCRIPTION

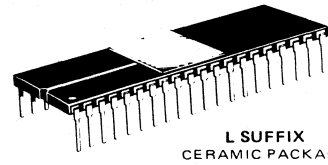
The MC6801L1 is a complete microcomputer on a chip and can be operational with a minimum of external components. It may be used in either the single chip or extended mode. With the Debug Monitor Program the user can debug a program under development and evaluate the different I/O configuration and modes of the MC6801. The 6801 firmware enables the user to:

- Load a Program from Tape
- Dump the Program
- Verify That a Program Was Properly Loaded
- Examine and Change Data in a Memory Location
- Punch (Record) the Program on Tape
- Calculate the Offset for Relative Addressing
- Examine and Change Data in the User's Program Registers and Counter
- Insert, Display, and Remove Breakpoints in the Program
- Free-Run or Trace Through the User's Program

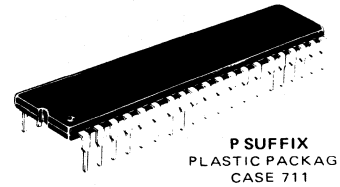
The firmware uses the serial port on the 6801 to provide input, output communications and connect to a tape or digital cassette within the terminal. The I/O functions of the firmware are down-loaded into the RAM to enable the user to change these parameters.

The MC6801L1 is used in the MC6801 Microcomputer Evaluation Module. (See preliminary information on the MEX6801EVM).

● Available December 1978 ●



L SUFFIX
CERAMIC PACKAGE
CASE 715



P SUFFIX
PLASTIC PACKAGE
CASE 711

FIGURE 1 – SINGLE-CHIP MICROCOMPUTER BLOCK DIAGRAM

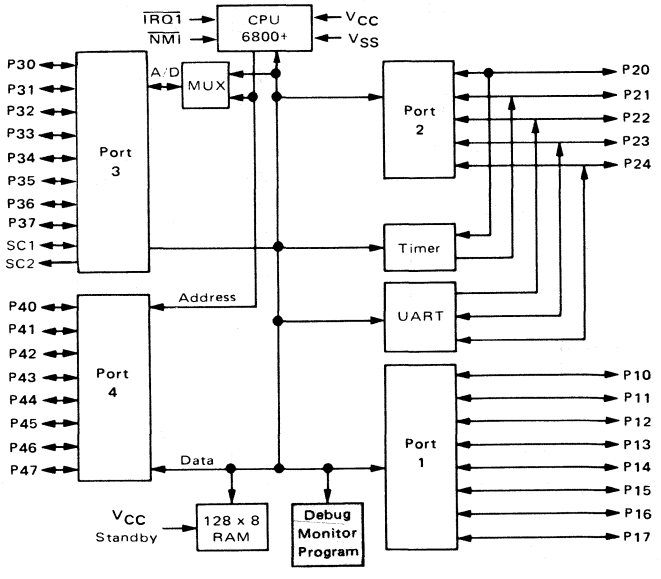
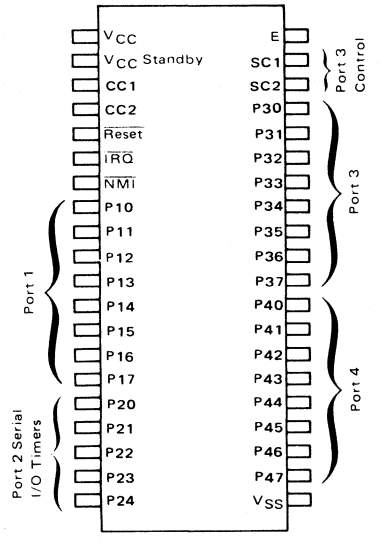


FIGURE 2 – MC6801



This is advance information and specifications are subject to change without notice.

NP-95

MOTOROLA

MC6802

Advance Information

MICROPROCESSOR WITH CLOCK AND RAM

The MC6802 is a monolithic 8-bit microprocessor that contains all the registers and accumulators of the present MC6800 plus an internal clock oscillator and driver on the same chip. In addition, the MC6802 has 128 bytes of RAM on board located at hex addresses 0000 to 007F. The first 32 bytes of RAM, at hex addresses 0000 to 001F, may be retained in a low power mode by utilizing V_{CC} standby, thus facilitating memory retention during a power-down situation.

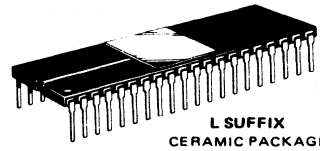
The MC6802 is completely software compatible with the MC6800 as well as the entire M6800 family of parts. Hence, the MC6802 is expandable to 65K words.

- On-Chip Clock Circuit
- 128 x 8 Bit On-Chip RAM
- 32 Bytes of RAM Are Retainable
- Software-Compatible with the MC6800
- Expandable to 65K words
- Standard TTL-Compatible Inputs and Outputs
- 8 Bit Word Size
- 16 Bit Memory Addressing
- Interrupt Capability

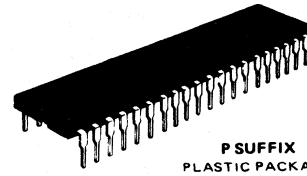
MOS

(N-CHANNEL, SILICON-GATE, DEPLETION LOAD)

MICROPROCESSOR WITH CLOCK AND RAM



L SUFFIX
CERAMIC PACKAGE
CASE 715



P SUFFIX
PLASTIC PACKAGE
CASE 711-03

FIGURE 1 – TYPICAL MICROCOMPUTER

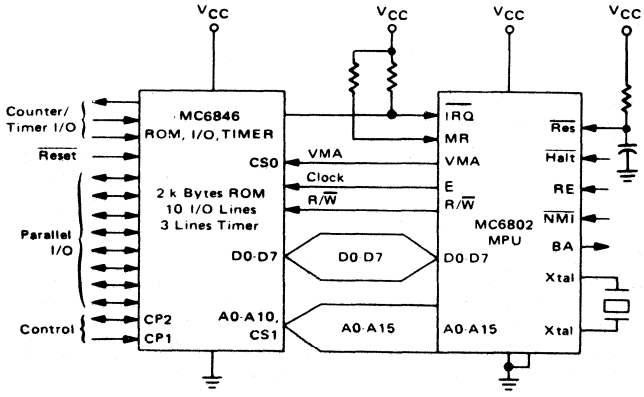
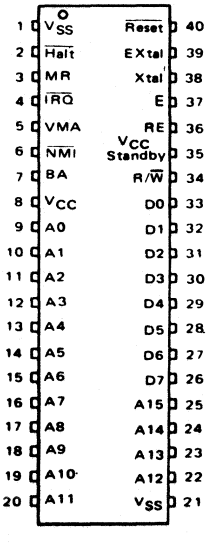


Figure 1 is a block diagram of a typical cost effective microcomputer. The MPU is the center of the microcomputer system and is shown in a minimum system interfacing with a ROM combination chip. It is not intended that this system be limited to this function but that it be expandable with other parts in the M6800 Microcomputer family.

PIN ASSIGNMENT



This is advance information and specifications are subject to change without notice.

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	-0.3 to +7.0	Vdc
Input Voltage	V_{in}	-0.3 to +7.0	Vdc
Operating Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	-55 to +150	°C
Thermal Resistance	θ_{JA}	70	°C/W

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit.

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ± 5%, V_{SS} = 0, T_A = 0 to 70°C unles otherwise noted.)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage	Logic, EXtal $\overline{Reset}$	V_{IH}	V_{SS} + 2.0 V_{SS} + 4.0	–	V_{CC} V_{CC}	Vdc
Input Low Voltage	Logic, EXtal $\overline{Reset}$	V_{IL}	V_{SS} – 0.3 V_{SS} – 0.3	–	V_{SS} + 0.8 V_{SS} + 2.3	Vdc
Input Leakage Current (V_{in} = 0 to 5.25 V, V_{CC} = max)	Logic*	I_{in}	–	1.0	2.5	μAdc
Output High Voltage (I_{Load} = -205 μAdc, V_{CC} = min) (I_{Load} = -145 μAdc, V_{CC} = min) (I_{Load} = -100 μAdc, V_{CC} = min)	 D0-D7 A0-A15, R/$\overline{W}$, VMA, E BA	V_{OH}	 V_{SS} + 2.4 V_{SS} + 2.4 V_{SS} + 2.4	 – – –	 – – –	Vdc
Output Low Voltage (I_{Load} = 1.6 mAdc, V_{CC} = min)		V_{OL}	–	–	V_{SS} + 0.4	Vdc
Power Dissipation		P_D**	–	0.600	1.2	W
Capacitance # (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	D0-D7 Logic Inputs, EXtal A0-A15, R/$\overline{W}$, VMA	C_{in} C_{out}	 – – –	 10 6.5 –	 12.5 10 12	pF pF
Frequency of Operation (Input Clock ÷4) (Crystal Frequency)		f f_{Xtal}	0.1 1.0	– –	1.0 4.0	MHz
Clock Timing Cycle Time		t_{cyc}	1.0	–	10	μs
Clock Pulse Width (Measured at 2.4 V)		$PW_{\phi Hs}$ $PW_{\phi L}$	450	–	9500	ns
Fall Time (Measured between V_{SS} + 0.4 V and V_{SS} + 2.4 V)		t_ϕ	–	–	25	ns

*Except $\overline{IRQ}$ and $\overline{NMI}$, which require 3 kΩ pullup load resistors for wire-OR capability at optimum operation. Does not include EXtal and Xtal, which are crystal inputs.

**In power-down mode, maximum power dissipation is less than 40 mW.

#Capacitances are periodically sampled rather than 100% tested.

READ/WRITE TIMING (Figures 2 through 6; Load Circuit of Figure 4.)

Characteristic	Symbol	Min	Typ	Max	Unit
Address Delay	t_{AD}	–	–	270	ns
Peripheral Read Access Time $t_{acc} = t_{ut} - (t_{AD} + t_{DSR})$ ($t_{ut} = t_{cyc} - t_\phi$)	t_{acc}	–	–	530	ns
Data Setup Time (Read)	t_{DSR}	100	–	–	ns
Input Data Hold Time	t_H	10	–	–	ns
Output Data Hold Time	t_H	30	–	–	ns
Address Hold Time (Address, R/$\overline{W}$, VMA)	t_{AH}	20	–	–	ns
Data Delay Time (Write)	t_{DDW}	–	165	225	ns
Processor Controls Processor Control Setup Time Processor Control Rise and Fall Time (Measured between 0.8 V and 2.0 V)	 t_{PCS} t_{PCr}, t_{PCf}	 200 –	 – –	 – 100	 ns ns

FIGURE 2 – READ DATA FROM MEMORY OR PERIPHERALS

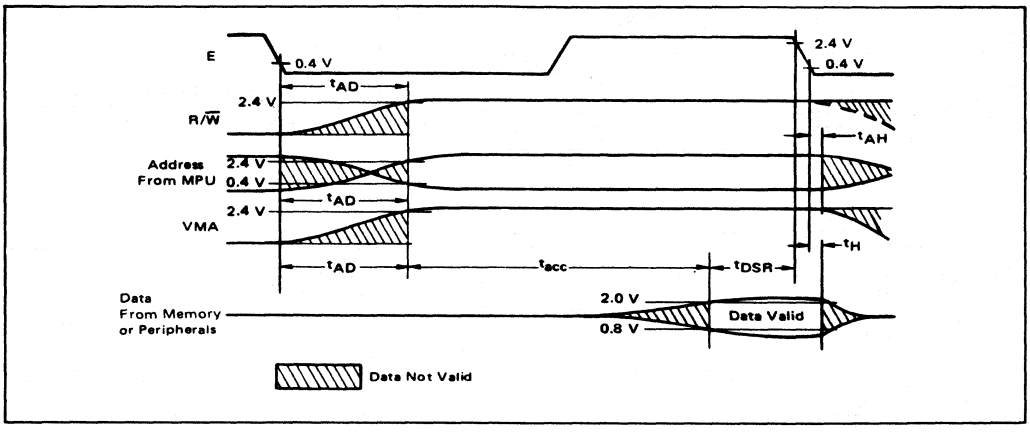


FIGURE 3 – WRITE DATA IN MEMORY OR PERIPHERALS

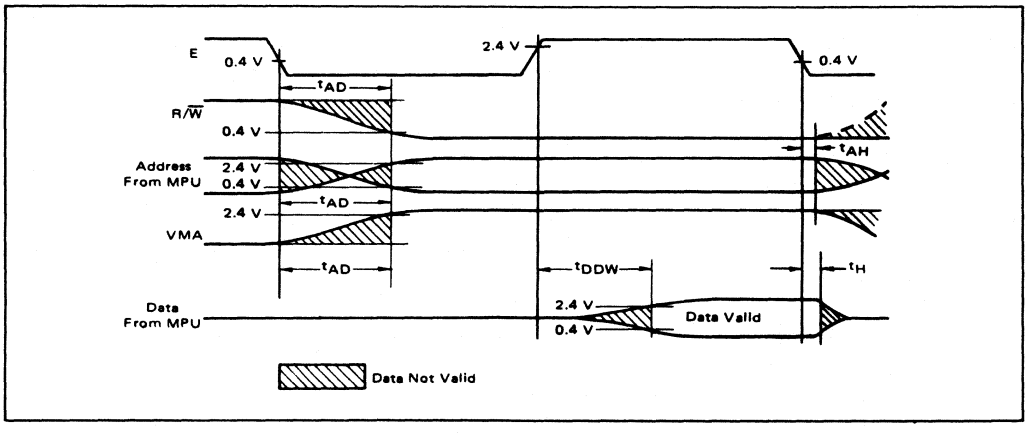


FIGURE 4 – BUS TIMING TEST LOAD

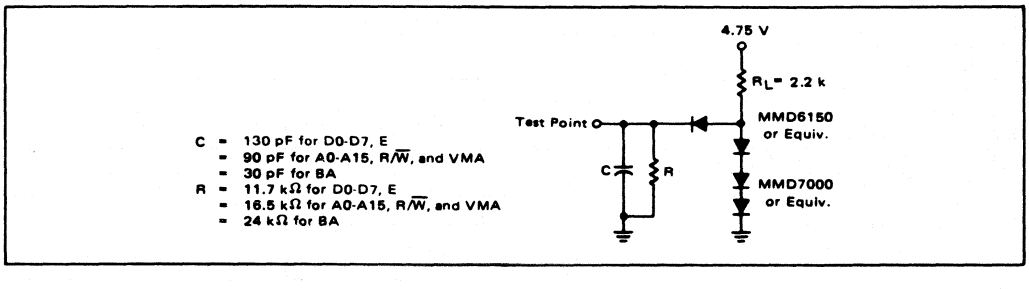


FIGURE 5 – TYPICAL DATA BUS OUTPUT DELAY versus CAPACITIVE LOADING

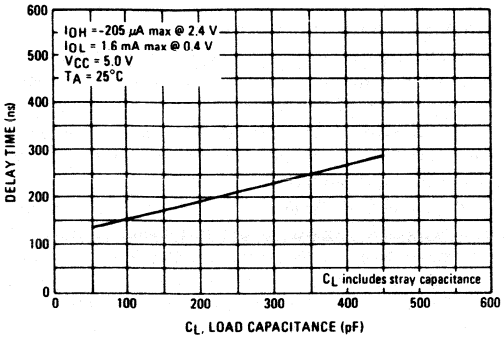


FIGURE 6 – TYPICAL READ/WRITE, VMA, AND ADDRESS OUTPUT DELAY versus CAPACITIVE LOADING

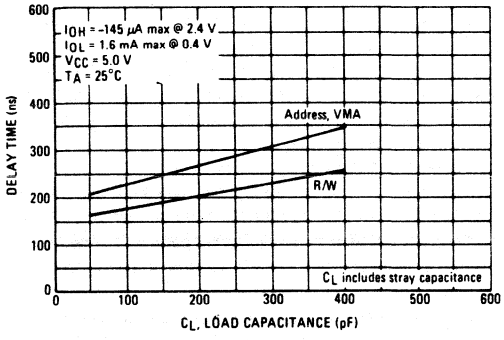
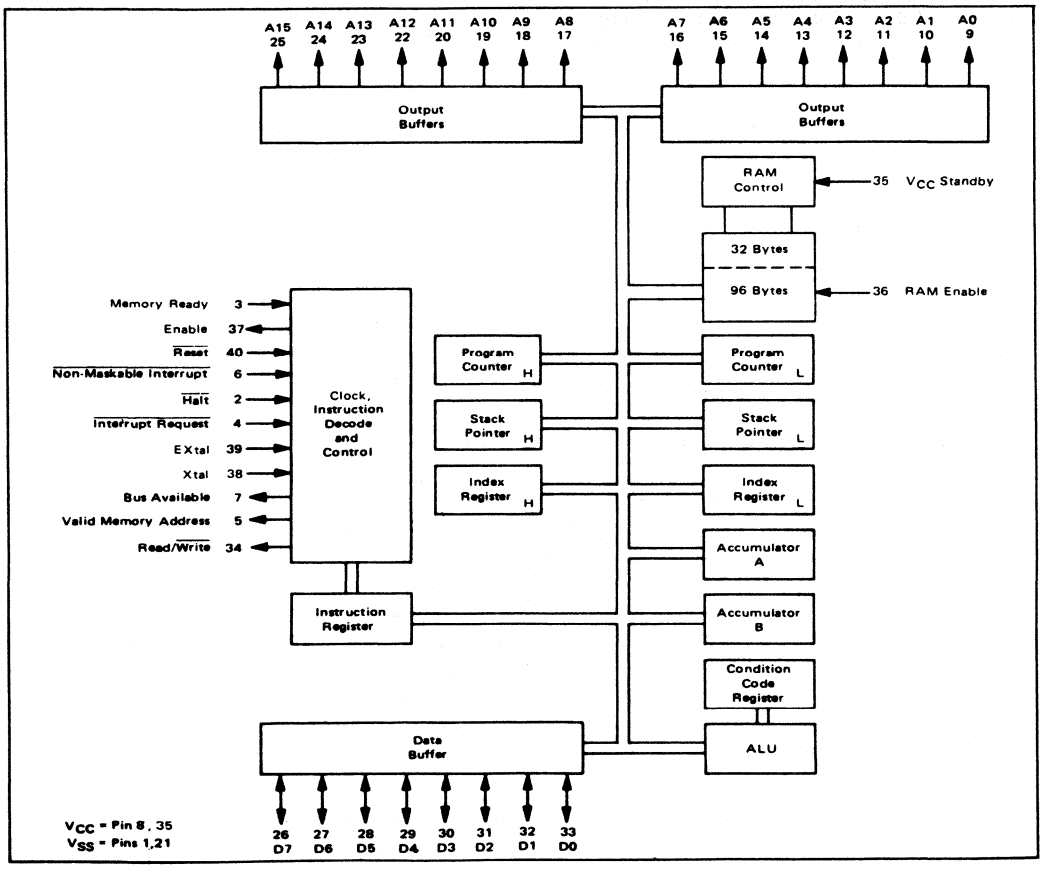


FIGURE 7 – MC6802 EXPANDED BLOCK DIAGRAM



MPU REGISTERS

A general block diagram of the MC6802 is shown in Figure 7. As shown, the number and configuration of the registers are the same as for the MC6800. The 128 x 8 bit RAM has been added to the basic MPU. The first 32 bytes may be operated in a low power mode via a V_{CC} standby. These 32 bytes can be retained during power-up and power-down conditions via the RE signal.

The MPU has three 16-bit registers and three 8-bit registers available for use by the programmer (Figure 8).

Program Counter – The program counter is a two byte (16-bits) register that points to the current program address.

Stack Pointer – The stack pointer is a two byte register that contains the address of the next available location in an external push-down/pop-up stack. This stack is normally a random access Read/Write memory that may have any location (address) that is convenient. In those applications that require storage of information in the stack when power is lost; the stack must be non-volatile.

Index Register – The index register is a two byte register that is used to store data or a sixteen bit memory address for the Indexed mode of memory addressing.

Accumulators – The MPU contains two 8-bit accumulators that are used to hold operands and results from an arithmetic logic unit (ALU).

Condition Code Register – The condition code register indicates the results of an Arithmetic Logic Unit operation: Negative (N), Zero (Z), Overflow (V), Carry from bit 7 (C), and half carry from bit 3 (H). These bits of the Condition Code Register are used as testable conditions for the conditional branch instructions. Bit 4 is the interrupt mask bit (I). The used bits of the Condition Code Register (b6 and b7) are ones.

Figure 9 shows the order of saving the microprocessor status within the stack.

FIGURE 8 – PROGRAMMING MODEL OF THE MICROPROCESSING UNIT

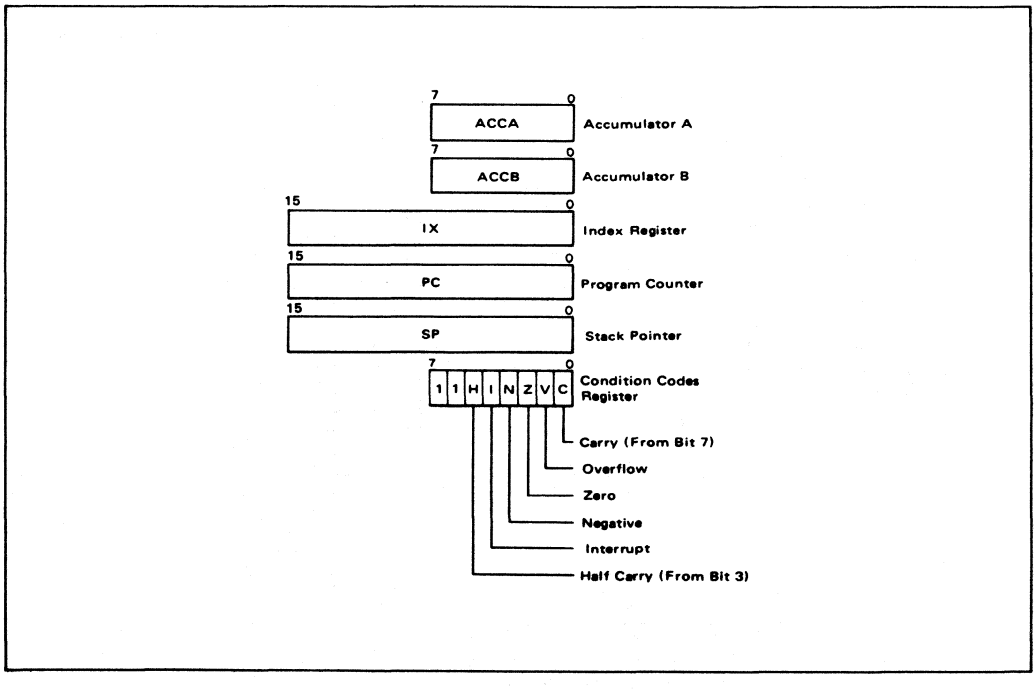
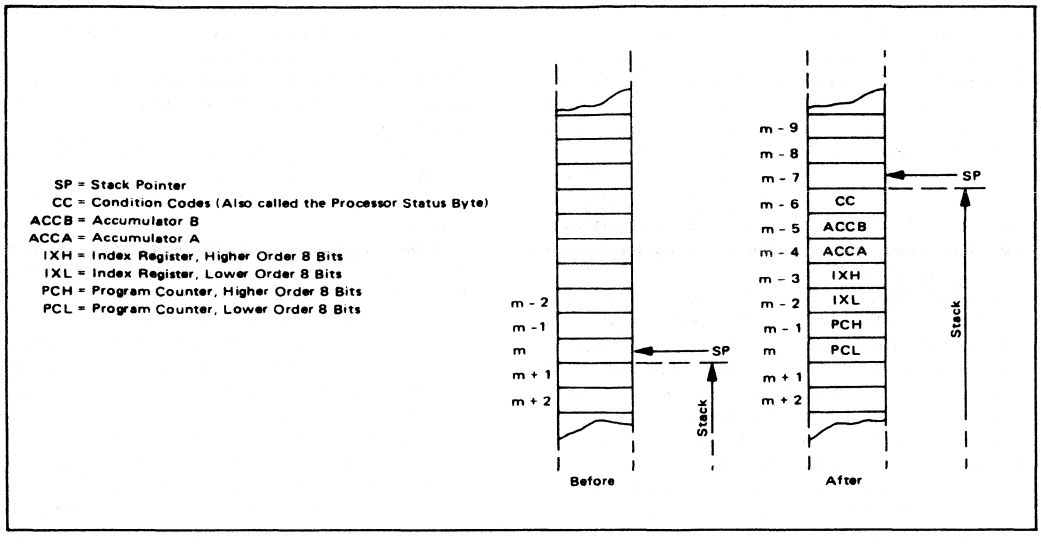


FIGURE 9 – SAVING THE STATUS OF THE MICROPROCESSOR IN THE STACK



MC6802 MPU SIGNAL DESCRIPTION

Proper operation of the MPU requires that certain control and timing signals be provided to accomplish specific functions and that other signal lines be monitored to determine the state of the processor. These control and timing signals for the MC6802 are similar to those of the MC6800 except that TSC, DBE, $\phi 1$, $\phi 2$ input, and two unused pins have been eliminated, and the following signal and timing lines have been added:

RAM Enable (RE)
Crystal Connections EXtal and Xtal
Memory Ready (MR)
V_{CC} Standby
Enable $\phi 2$ Output (E)

The following is a summary of the MC6802 MPU signals:

Address Bus (A0–A15) – Sixteen pins are used for the address bus. The outputs are capable of driving one standard TTL load and 90 pF.

Data Bus (D0–D7) – Eight pins are used for the data bus. It is bidirectional, transferring data to and from the memory and peripheral devices. It also has three-state output buffers capable of driving one standard TTL load and 130 pF.

Data Bus will be in the output mode when the internal RAM is accessed. This prohibits external data entering the MPU. It should be noted that the internal RAM is fully decoded from $0000 to $007F. External RAM at $0000 to $007F must be disabled when internal RAM is accessed.

$\overline{\text{Halt}}$ – When this input is in the low state, all activity in the machine will be halted. This input is level sensitive. In the halt mode, the machine will stop at the end of an instruction, Bus Available will be at a high state, Valid Memory Address will be at a low state. The address bus will display the address of the next instruction.

To insure single instruction operation, transition of the $\overline{\text{Halt}}$ line must not occur during the last 200 ns of E and the $\overline{\text{Halt}}$ line must go high for one Clock cycle.

Halt should be tied high if not used. This is good engineering design practice in general and necessary to insure proper operation of the part.

Read/$\overline{\text{Write}}$ (R/$\overline{\text{W}}$) – This TTL compatible output signals the peripherals and memory devices whether the MPU is in a Read (high) or Write (low) state. The normal standby state of this signal is Read (high). When the processor is halted, it will be in the logical one state. This output is capable of driving one standard TTL load and 90 pF.

Valid Memory Address (VMA) – This output indicates to peripheral devices that there is a valid address on the address bus. In normal operation, this signal should be utilized for enabling peripheral interfaces such as the PIA and ACIA. This signal is not three-state. One standard TTL load and 90 pF may be directly driven by this active high signal.

Bus Available (BA) – The Bus Available signal will normally be in the low state; when activated, it will go to the high state indicating that the microprocessor has stopped and that the address bus is available (but not in a three-state condition). This will occur if the $\overline{\text{Halt}}$ line is in the low state or the processor is in the WAIT state as a result of the execution of a WAIT instruction. At such time, all three-state output drivers will go to their off state and other outputs to their normally inactive level. The processor is removed from the WAIT state by the occurrence of a maskable (mask bit I = 0) or nonmaskable interrupt. This output is capable of driving one standard TTL load and 30 pF.

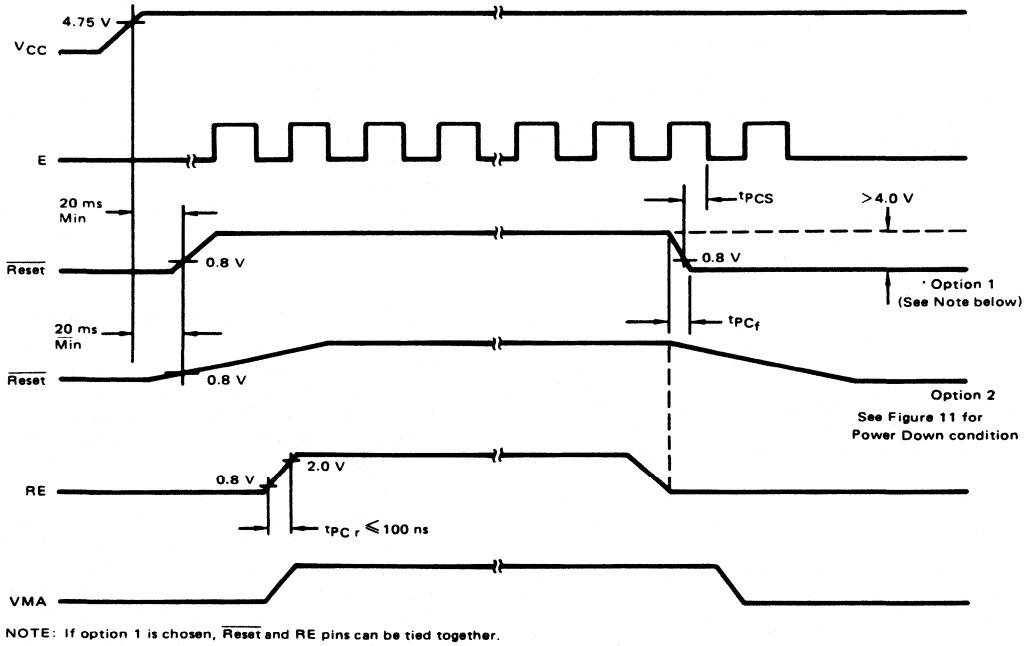
Interrupt Request ($\overline{\text{IRQ}}$) – This level sensitive input requests that an interrupt sequence be generated within the machine. The processor will wait until it completes the current instruction that is being executed before it recognizes the request. At that time, if the interrupt mask bit in the Condition Code Register is not set, the machine will begin an interrupt sequence. The Index Register, Program Counter, Accumulators, and Condition Code Register are stored away on the stack. Next the MPU will respond to the interrupt request by setting the interrupt mask bit high so that no further interrupts may occur. At the end of the cycle, a 16-bit address will be loaded that points to a vectoring address which is located in memory locations FFF8 and FFF9. An address loaded at these locations causes the MPU to branch to an interrupt routine in memory.

The $\overline{\text{Halt}}$ line must be in the high state for interrupts to be serviced. Interrupts will be latched internally while $\overline{\text{Halt}}$ is low.

The $\overline{\text{IRQ}}$ has a high impedance pullup device internal to the chip; however, a 3 kΩ external resistor to V_{CC} should be used for wire-OR and optimum control of interrupts.

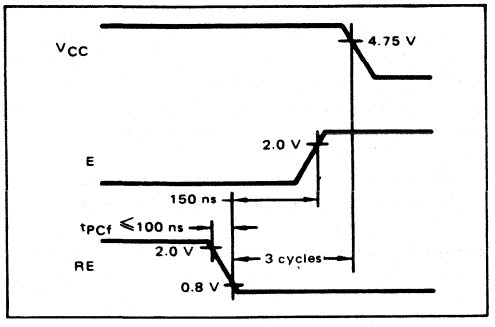
$\overline{\text{Reset}}$ – This input is used to reset and start the MPU from a power-down condition, resulting from a power failure or an initial start-up of the processor. When this line is low, the MPU is inactive and the information in the registers will be lost. If a high level is detected on the input, this will signal the MPU to begin the restart sequence. This will start execution of a routine to initialize the processor from its reset condition. All the higher order address lines will be forced high. For the restart, the last two (FFFE, FFFF) locations in memory will be used to load the program that is addressed by the program counter. During the restart routine, the interrupt mask bit is set and must be reset before the MPU can be interrupted by $\overline{\text{IRQ}}$. Power-up and reset timing and power-down sequences are shown in Figures 10 and 11, respectively.

FIGURE 10 – POWER-UP AND RESET TIMING



NOTE: If option 1 is chosen, $\overline{\text{Reset}}$ and RE pins can be tied together.

FIGURE 11 – POWER-DOWN SEQUENCE



Reset, when brought low, must be held low at least three clock cycles. This allows the MC6802 adequate time to respond internally to the reset. This is independent of the 20 ms power-up reset that is required.

When Reset is released it MUST go through the low-to-high threshold without bouncing, oscillating, or otherwise causing an erroneous Reset (less than three clock cycles). This may cause improper MPU operation until the next valid Reset.

Non-Maskable Interrupt (NMI) – A low-going edge on this input requests that a non-mask-interrupt sequence be generated within the processor. As with the Interrupt Request signal, the processor will complete the current instruction that is being executed before it recognizes the $\overline{NMI}$ signal. The interrupt mask bit in the Condition Code Register has no effect on $\overline{NMI}$.

The Index Register, Program Counter, Accumulators, and Condition Code Register are stored away on the stack. At the end of the cycle, a 16-bit address will be loaded that points to a vectoring address which is located in memory locations FFFC and FFFD. An address loaded at these locations caused the MPU to branch to a non-maskable interrupt routine in memory.

$\overline{NMI}$ has a high impedance pullup resistor internal to the chip; however, a 3 kΩ external resistor to V_{CC} should be used for wire-OR and optimum control of interrupts.

Inputs $\overline{IRQ}$ and $\overline{NMI}$ are hardware interrupt lines that are sampled when E is high and will start the interrupt routine on a low E following the completion of an instruction.

$\overline{NMI}$ should be tied high if not used. This is good engineering design practice in general and necessary to insure proper operation of the part.

Figure 12 is a flowchart describing the major decision paths and interrupt vectors of the microprocessor. Table 1 gives the memory map for interrupt vectors.

RAM Enable (RE) – A TTL-compatible RAM enable input controls the on-chip RAM of the MC6802. When placed in the high state, the on-chip memory is enabled to respond to the MPU controls. In the low state, RAM is disabled. This pin may also be utilized to disable reading and writing the on-chip RAM during a power-down situation. RAM enable must be low three cycles before V_{CC} goes below 4.75 V during power-down.

RE should be tied to the correct high or low state if not used. This is good engineering design practice in general and necessary to insure proper operation of the part.

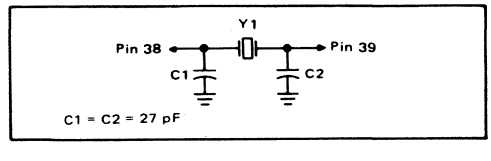
EXtal and Xtal – The MC6802 has an internal oscillator that may be crystal controlled. These connections are for a parallel resonant fundamental crystal. (AT cut.) A divide-by-four circuit has been added to the MC6802 so that a 4 MHz crystal may be used in lieu of a 1 MHz crystal for a more cost-effective system. Pin 39 of the MC6802 may be driven externally by a TTL input signal if a separate clock is required. Pin 38 is to be left open in this mode.

An RC network is not directly usable as a frequency source on pins 38 and 39. An RC network type TTL or CMOS oscillator will work well as long as the TTL or CMOS output drives the MC6802.

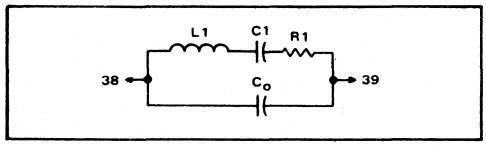
LC networks are not recommended to be used in place of the crystal. Simulating the crystal characteristics with discrete components has not been investigated. It would probably be more expensive than the crystal alone after all costs are considered. Performance would also be worse.

If an external clock is used, it may not be halted for more than 4.5 μs. The MC6802 is a dynamic part except for the internal RAM, and requires the external clock to retain information.

Crystal Specification – The 4.0 MHz specification case is shown below:



The following is the figure and parameters to be supplied to the crystal vendor.



AT – Cut Parallel Resonance Crystal
C_o = 7.0 pF Max
Frequency = 4.0 MHz @ C_L = 24 pF
R1 = 50 ohms Max
Frequency Tolerance – ±5% to ±0.02%

TOLERANCE NOTE:

Critical timing loops may require a better tolerance than ±5%. Because of production deviations and the Temperature Coefficient of the MC6802, the best "worst-case design" tolerance is ±0.05% (500 ppm) using a ±0.02% crystal. If the MC6802 is not going to be used over its entire temperature range of 0°C to 70°C, a much tighter overall tolerance can be achieved.

In those applications where other than a 4.0 MHz crystal is used, the following table gives the designer the crystal parameters to be specified. The table contains the entire spectrum of usable crystals for the MC6802. Crystal frequencies not shown (that lie between 1.0 MHz and 4.0 MHz) may be interpolated from the table.

Y1 Crystal Frequency	C1 and C2	C Load	R1 (Max)	C_o (Max)
4.0 MHz	27 pF	24 pF	50 ohms	7.0 pF
3.58 MHz	27 pF	20 pF	50 ohms	7.0 pF
3.0 MHz	27 pF	18 pF	75 ohms	6.7 pF
2.5 MHz	27 pF	18 pF	75 ohms	6.0 pF
2.0 MHz	33 pF	24 pF	100 ohms	5.5 pF
1.5 MHz	39 pF	27 pF	200 ohms	4.5 pF
1.0 MHz	39 pF	30 pF	250 ohms	4.0 pF

Memory Ready (MR) – MR is a TTL compatible input control signal which allows stretching of E. When MR is high, E will be in normal operation. When MR is low, E may be stretched integral multiples of half periods, thus allowing interface to slow memories. Memory Ready timing is shown in Figure 13.

MR should be tied high if not used. This is good engineering design practice in general and necessary to insure proper operation of the part. A maximum stretch is 10 μs.

Enable (E) – This pin supplies the clock for the MPU and the rest of the system. This is a single phase, TTL compatible clcok. This clock may be conditioned by a Memory Ready Signal. This is equivalent to $\phi 2$ on the MC6800.

V_{CC} Standby (Range = 4.0 V to 5.25 V) – This pin supplies the dc voltage to the first 32 bytes of RAM as well as the RAM Enable (RE) control logic. Thus retention of data in this portion of the RAM on a power-up, power-down, or standby condition is guaranteed. Maximum current drain at 5.25 V is 8 mA.

TABLE 1 – MEMORY MAP FOR INTERRUPT VECTORS

Vector MS	Vector LS	Description
FFFE	FFFF	Restart
FFFC	FFFD	Non-Maskable Interrupt
FFFA	FFFB	Software Interrupt
FFF8	FFF9	Interrupt Request

FIGURE 12 – MPU FLOW CHART

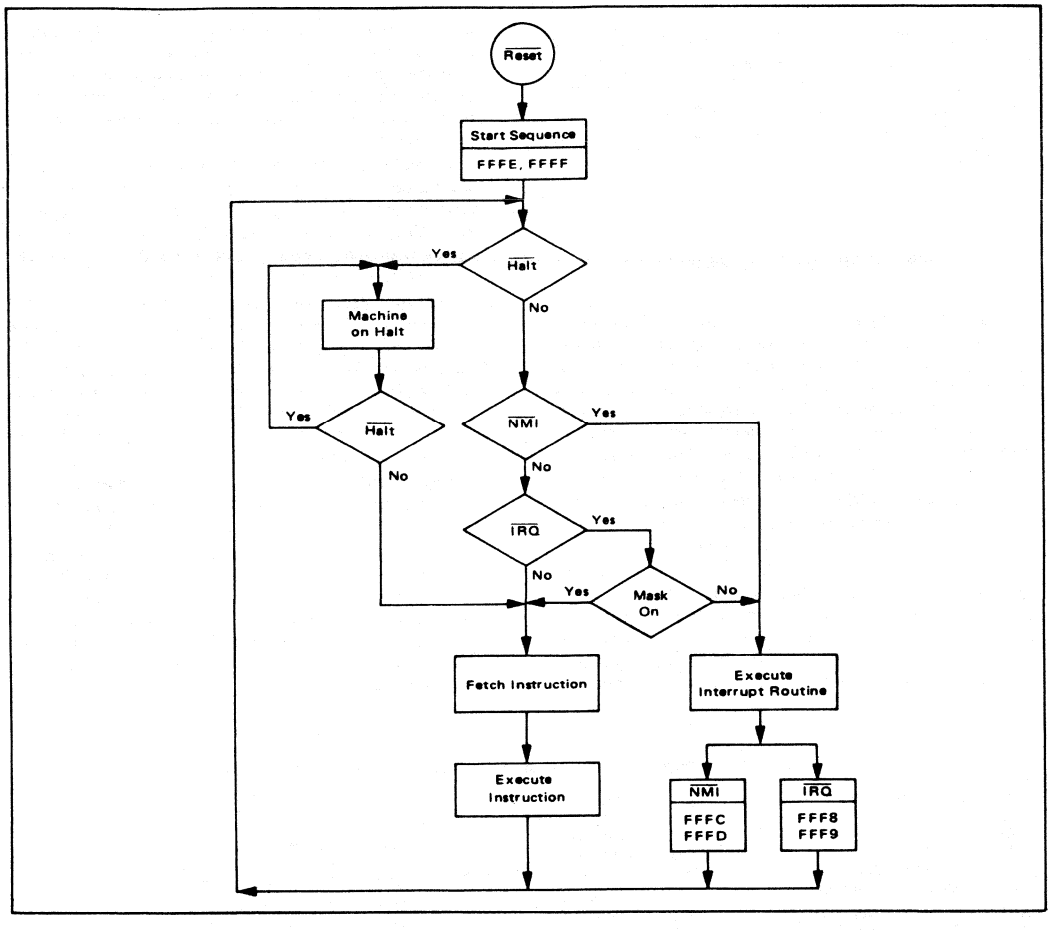
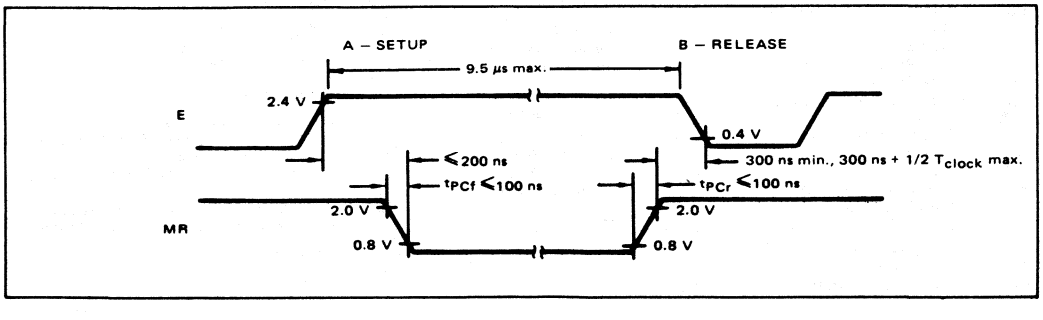


FIGURE 13 – MEMORY READY CONTROL FUNCTION



MPU INSTRUCTION SET

The MC6802 has a set of 72 different instructions. Included are binary and decimal arithmetic, logical, shift, rotate, load, store, conditional or unconditional branch, interrupt and stack manipulation instructions (Tables 2 thru 6). This instruction set is the same as that for the MC6800.

MPU ADDRESSING MODES

The MC6802 eight-bit microprocessing unit has seven address modes that can be used by a programmer, with the addressing mode a function of both the type of instruction and the coding within the instruction. A summary of the addressing modes for a particular instruction can be found in Table 7 along with the associated instruction execution time that is given in machine cycles. With a clock frequency of 1 MHz, these times would be microseconds.

Accumulator (ACCX) Addressing – In accumulator only addressing, either accumulator A or accumulator B is specified. These are one-byte instructions.

Immediate Addressing – In immediate addressing, the operand is contained in the second byte of the instruction except LDS and LDX which have the operand in the second and third bytes of the instruction. The MPU addresses this location when it fetches the immediate instruction for execution. These are two or three-byte instructions.

Direct Addressing – In direct addressing, the address of the operand is contained in the second byte of the instruction. Direct addressing allows the user to directly address the lowest 256 bytes in the machine i.e., locations zero through 255. Enhanced execution times are achieved by storing data in these locations. In most configurations, it should be a random access memory. These are two-byte instructions.

Extended Addressing – In extended addressing, the address contained in the second byte of the instruction is used as the higher eight-bits of the address of the operand. The third byte of the instruction is used as the lower eight-bits of the address for the operand. This is an absolute address in memory. These are three-byte instructions.

Indexed Addressing – In indexed addressing, the address contained in the second byte of the instruction is added to the index register's lowest eight bits in the MPU. The carry is then added to the higher order eight bits of the index register. This result is then used to address memory. The modified address is held in a temporary address register so there is no change to the index register. These are two-byte instructions.

Implied Addressing – In the implied addressing mode the instruction gives the address (i.e., stack pointer, index register, etc.). These are one-byte instructions.

Relative Addressing – In relative addressing, the address contained in the second byte of the instruction is added to the program counter's lowest eight bits plus two. The carry or borrow is then added to the high eight bits. This allows the user to address data within a range of -125 to +129 bytes of the present instruction. These are two-byte instructions.

TABLE 2 – MICROPROCESSOR INSTRUCTION SET – ALPHABETIC SEQUENCE

ABA	Add Accumulators
ADC	Add with Carry
ADD	Add
AND	Logical And
ASL	Arithmetic Shift Left
ASR	Arithmetic Shift Right
BCC	Branch if Carry Clear
BCS	Branch if Carry Set
BEQ	Branch if Equal to Zero
BGE	Branch if Greater or Equal Zero
BGT	Branch if Greater than Zero
BHI	Branch if Higher
BIT	Bit Test
BLE	Branch if Less or Equal
BLS	Branch if Lower or Same
BLT	Branch if Less than Zero
BMI	Branch if Minus
BNE	Branch if Not Equal to Zero
BPL	Branch if Plus
BRA	Branch Always
BSR	Branch to Subroutine
BVC	Branch if Overflow Clear
BVS	Branch if Overflow Set
CBA	Compare Accumulators
CLC	Clear Carry
CLI	Clear Interrupt Mask
CLR	Clear
CLV	Clear Overflow
CMP	Compare
COM	Complement
CPX	Compare Index Register
DAA	Decimal Adjust
DEC	Decrement
DES	Decrement Stack Pointer
DEX	Decrement Index Register
EOR	Exclusive OR
INC	Increment
INS	Increment Stack Pointer
INX	Increment Index Register
JMP	Jump
JSR	Jump to Subroutine
LDA	Load Accumulator
LDS	Load Stack Pointer
LDX	Load Index Register
LSR	Logical Shift Right
NEG	Negate
NOP	No Operation
ORA	Inclusive OR Accumulator
PSH	Push Data
PUL	Pull Data
ROL	Rotate Left
ROR	Rotate Right
RTI	Return from Interrupt
RTS	Return from Subroutine
SBA	Subtract Accumulators
SBC	Subtract with Carry
SEC	Set Carry
SEI	Set Interrupt Mask
SEV	Set Overflow
STA	Store Accumulator
STS	Store Stack Register
STX	Store Index Register
SUB	Subtract
SWI	Software Interrupt
TAB	Transfer Accumulators
TAP	Transfer Accumulators to Condition Code Reg.
TBA	Transfer Accumulators
TPA	Transfer Condition Code Reg. to Accumulator
TST	Test
TSX	Transfer Stack Pointer to Index Register
TXS	Transfer Index Register to Stack Pointer
WAI	Wait for Interrupt

TABLE 3 – ACCUMULATOR AND MEMORY INSTRUCTIONS

OPERATIONS	MNEMONIC	IMMED OP	IMMED ~	IMMED =	DIRECT OP	DIRECT ~	DIRECT =	INDEX OP	INDEX ~	INDEX =	EXTND OP	EXTND ~	EXTND =	IMPLIED OP	IMPLIED ~	IMPLIED =	BOOLEAN/ARITHMETIC OPERATION (All register labels refer to contents)	5 H	4 I	3 N	2 Z	1 V	0 C
Add	ADDA	8B	2	2	9B	3	2	AB	5	2	BB	4	3				A + M → A	↕	•	↕	↕	↕	↕
	ADDB	CB	2	2	DB	3	2	EB	5	2	FB	4	3				B + M → B	↕	•	↕	↕	↕	↕
Add Acmltrs	ABA													1B	2	1	A + B → A	↕	•	↕	↕	↕	↕
Add with Carry	ADCA	89	2	2	99	3	2	A9	5	2	B9	4	3				A + M + C → A	↕	•	↕	↕	↕	↕
	ADCB	C9	2	2	D9	3	2	E9	5	2	F9	4	3				B + M + C → B	↕	•	↕	↕	↕	↕
And	ANDA	84	2	2	94	3	2	A4	5	2	B4	4	3				A · M → A	•	•	↕	↕	R	•
	ANDB	C4	2	2	D4	3	2	E4	5	2	F4	4	3				B · M → B	•	•	↕	↕	R	•
Bit Test	BITA	85	2	2	95	3	2	A5	5	2	B5	4	3				A · M	•	•	↕	↕	R	•
	BITB	C5	2	2	D5	3	2	E5	5	2	F5	4	3				B · M	•	•	↕	↕	R	•
Clear	CLR							6F	7	2	7F	6	3				00 → M	•	•	R	S	R	R
	CLRA													4F	2	1	00 → A	•	•	R	S	R	R
	CLRB													5F	2	1	00 → B	•	•	R	S	R	R
Compare	CMPA	81	2	2	91	3	2	A1	5	2	B1	4	3				A − M	•	•	↕	↕	↕	↕
	CMPB	C1	2	2	D1	3	2	E1	5	2	F1	4	3				B − M	•	•	↕	↕	↕	↕
Compare Acmltrs	CBA													11	2	1	A − B	•	•	↕	↕	↕	↕
Complement, 1's	COM							63	7	2	73	6	3				M̄ → M	•	•	↕	↕	R	S
	COMA													43	2	1	Ā → A	•	•	↕	↕	R	S
	COMB													53	2	1	B̄ → B	•	•	↕	↕	R	S
Complement, 2's (Negate)	NEG							60	7	2	70	6	3				00 − M → M	•	•	↕	↕	①	②
	NEGA													40	2	1	00 − A → A	•	•	↕	↕	①	②
	NEGB													50	2	1	00 − B → B	•	•	↕	↕	①	②
Decimal Adjust, A	DAA													19	2	1	Converts Binary Add. of BCD Characters into BCD Format	•	•	↕	↕	↕	③
Decrement	DEC							6A	7	2	7A	6	3				M − 1 → M	•	•	↕	↕	4	•
	DECA													4A	2	1	A − 1 → A	•	•	↕	↕	4	•
	DECB													5A	2	1	B − 1 → B	•	•	↕	↕	4	•
Exclusive OR	EORA	88	2	2	98	3	2	A8	5	2	B8	4	3				A⊕M → A	•	•	↕	↕	R	•
	EORB	C8	2	2	D8	3	2	E8	5	2	F8	4	3				B⊕M → B	•	•	↕	↕	R	•
Increment	INC							6C	7	2	7C	6	3				M + 1 → M	•	•	↕	↕	⑤	•
	INCA													4C	2	1	A + 1 → A	•	•	↕	↕	⑤	•
	INCB													5C	2	1	B + 1 → B	•	•	↕	↕	⑤	•
Load Acmltr	LOAA	86	2	2	96	3	2	A6	5	2	B6	4	3				M → A	•	•	↕	↕	R	•
	LOAB	C6	2	2	D6	3	2	E6	5	2	F6	4	3				M → B	•	•	↕	↕	R	•
Or, Inclusive	ORAA	8A	2	2	9A	3	2	AA	5	2	BA	4	3				A + M → A	•	•	↕	↕	R	•
	ORAB	CA	2	2	DA	3	2	EA	5	2	FA	4	3				B + M → B	•	•	↕	↕	R	•
Push Data	PSHA													36	4	1	A → M_{SP}, SP − 1 → SP	•	•	•	•	•	•
	PSHB													37	4	1	B → M_{SP}, SP − 1 → SP	•	•	•	•	•	•
Pull Data	PULA													32	4	1	SP + 1 → SP, M_{SP} → A	•	•	•	•	•	•
	PULB													33	4	1	SP + 1 → SP, M_{SP} → B	•	•	•	•	•	•
Rotate Left	ROL							69	7	2	79	6	3				M	•	•	↕	↕	⑥	↕
	ROLA													49	2	1	A	•	•	↕	↕	⑥	↕
	ROLB													59	2	1	B	•	•	↕	↕	⑥	↕
Rotate Right	ROR							66	7	2	76	6	3				M	•	•	↕	↕	⑥	↕
	RORA													46	2	1	A	•	•	↕	↕	⑥	↕
	RORB													56	2	1	B	•	•	↕	↕	⑥	↕
Shift Left, Arithmetic	ASL							68	7	2	78	6	3				M	•	•	↕	↕	⑥	↕
	ASLA													48	2	1	A	•	•	↕	↕	⑥	↕
	ASLB													58	2	1	B	•	•	↕	↕	⑥	↕
Shift Right, Arithmetic	ASR							67	7	2	77	6	3				M	•	•	↕	↕	⑥	↕
	ASRA													47	2	1	A	•	•	↕	↕	⑥	↕
	ASRB													57	2	1	B	•	•	↕	↕	⑥	↕
Shift Right, Logic	LSR							64	7	2	74	6	3				M	•	•	R	↕	⑥	↕
	LSRA													44	2	1	A	•	•	R	↕	⑥	↕
	LSRB													54	2	1	B	•	•	R	↕	⑥	↕
Store Acmltr	STAA				97	4	2	A7	6	2	B7	5	3				A → M	•	•	↕	↕	R	•
	STAB				D7	4	2	E7	6	2	F7	5	3				B → M	•	•	↕	↕	R	•
Subtract	SUBA	80	2	2	90	3	2	A0	5	2	B0	4	3				A − M → A	•	•	↕	↕	↕	↕
	SUBB	C0	2	2	D0	3	2	E0	5	2	F0	4	3				B − M → B	•	•	↕	↕	↕	↕
Subtract Acmltrs.	SBA													10	2	1	A − B → A	•	•	↕	↕	↕	↕
Subtr. with Carry	SBCA	82	2	2	92	3	2	A2	5	2	B2	4	3				A − M − C → A	•	•	↕	↕	↕	↕
	SBCB	C2	2	2	D2	3	2	E2	5	2	F2	4	3				B − M − C → B	•	•	↕	↕	↕	↕
Transfer Acmltrs	TAB													16	2	1	A → B	•	•	↕	↕	R	•
	TBA													17	2	1	B → A	•	•	↕	↕	R	•
Test, Zero or Minus	TST							6D	7	2	7D	6	3				M − 00	•	•	↕	↕	R	R
	TSTA													4D	2	1	A − 00	•	•	↕	↕	R	R
	TSTB													5D	2	1	B − 00	•	•	↕	↕	R	R
																		H	I	N	Z	V	C

LEGEND:

OP	Operation Code (Hexadecimal);	+	Boolean Inclusive OR;
~	Number of MPU Cycles;	⊕	Boolean Exclusive OR;
=	Number of Program Bytes;	M̄	Complement of M;
+	Arithmetic Plus;	→	Transfer Into;
−	Arithmetic Minus;	0	Bit = Zero;
·	Boolean AND;	00	Byte = Zero;
M_{SP}	Contents of memory location pointed to be Stack Pointer;		

Note – Accumulator addressing mode instructions are included in the column for IMPLIED addressing

CONDITION CODE SYMBOLS:

H	Half-carry from bit 3;
I	Interrupt mask
N	Negative (sign bit)
Z	Zero (byte)
V	Overflow, 2's complement
C	Carry from bit 7
R	Reset Always
S	Set Always
↕	Test and set if true, cleared otherwise
•	Not Affected

TABLE 4 – INDEX REGISTER AND STACK MANIPULATION INSTRUCTIONS

		IMMED			DIRECT			INDEX			EXTND			IMPLIED				COND. CODE REG.					
																		5	4	3	2	1	0
POINTER OPERATIONS	MNEMONIC	OP	~	#	OP	~	#	OP	~	#	OP	~	#	OP	~	#	BOOLEAN/ARITHMETIC OPERATION	H	I	N	Z	V	C
Compare Index Reg	CPX	8C	3	3	9C	4	2	AC	6	2	BC	5	3				$X_H - M, X_L - (M + 1)$	•	•	⑦	↕	⑧	•
Decrement Index Reg	DEX													09	4	1	$X - 1 \rightarrow X$	•	•	•	↕	•	•
Decrement Stack Pntr	DES													34	4	1	$SP - 1 \rightarrow SP$	•	•	•	•	•	•
Increment Index Reg	INX													08	4	1	$X + 1 \rightarrow X$	•	•	•	↕	•	•
Increment Stack Pntr	INS													31	4	1	$SP + 1 \rightarrow SP$	•	•	•	•	•	•
Load Index Reg	LDX	CE	3	3	DE	4	2	EE	6	2	FE	5	3				$M \rightarrow X_H, (M + 1) \rightarrow X_L$	•	•	⑨	↕	R	•
Load Stack Pntr	LDS	8E	3	3	9E	4	2	AE	6	2	BE	5	3				$M \rightarrow SP_H, (M + 1) \rightarrow SP_L$	•	•	⑨	↕	R	•
Store Index Reg	STX				DF	5	2	EF	7	2	FF	6	3				$X_H \rightarrow M, X_L \rightarrow (M + 1)$	•	•	⑨	↕	R	•
Store Stack Pntr	STS				9F	5	2	AF	7	2	BF	6	3				$SP_H \rightarrow M, SP_L \rightarrow (M + 1)$	•	•	⑨	↕	R	•
Indx Reg → Stack Pntr	TXS													35	4	1	$X - 1 \rightarrow SP$	•	•	•	•	•	•
Stack Pntr → Indx Reg	TSX													30	4	1	$SP + 1 \rightarrow X$	•	•	•	•	•	•

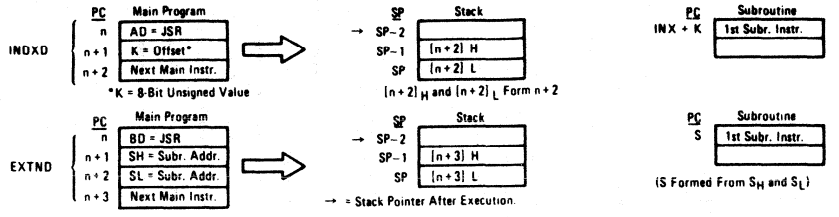
TABLE 5 – JUMP AND BRANCH INSTRUCTIONS

		RELATIVE			INDEX			EXTND			IMPLIED				COND. CODE REG.						
															5	4	3	2	1	0
OPERATIONS	MNEMONIC	OP	~	#	OP	~	#	OP	~	#	OP	~	#	BRANCH TEST	H	I	N	Z	V	C
Branch Always	BRA	20	4	2										None	•	•	•	•	•	•
Branch If Carry Clear	BCC	24	4	2										C = 0	•	•	•	•	•	•
Branch If Carry Set	BCS	25	4	2										C = 1	•	•	•	•	•	•
Branch If = Zero	BEQ	27	4	2										Z = 1	•	•	•	•	•	•
Branch If ≥ Zero	BGE	2C	4	2										$N \oplus V = 0$	•	•	•	•	•	•
Branch If > Zero	BGT	2E	4	2										$Z + (N \oplus V) = 0$	•	•	•	•	•	•
Branch If Higher	BHI	22	4	2										C + Z = 0	•	•	•	•	•	•
Branch If ≤ Zero	BLE	2F	4	2										$Z + (N \oplus V) = 1$	•	•	•	•	•	•
Branch If Lower Or Same	BLS	23	4	2										C + Z = 1	•	•	•	•	•	•
Branch If < Zero	BLT	2D	4	2										$N \oplus V = 1$	•	•	•	•	•	•
Branch If Minus	BMI	2B	4	2										N = 1	•	•	•	•	•	•
Branch If Not Equal Zero	BNE	26	4	2										Z = 0	•	•	•	•	•	•
Branch If Overflow Clear	BVC	28	4	2										V = 0	•	•	•	•	•	•
Branch If Overflow Set	BVS	29	4	2										V = 1	•	•	•	•	•	•
Branch If Plus	BPL	2A	4	2										N = 0	•	•	•	•	•	•
Branch To Subroutine	BSR	8D	8	2											•	•	•	•	•	•
Jump	JMP				6E	4	2	7E	3	3				See Special Operations	•	•	•	•	•	•
Jump To Subroutine	JSR				AD	8	2	BD	9	3					•	•	•	•	•	•
No Operation	NOP										01	2	1	Advances Prog. Cntr. Only	•	•	•	•	•	•
Return From Interrupt	RTI										3B	10	1		—	—	—	⑩	—	—
Return From Subroutine	RTS										39	5	1		•	•	•	•	•	•
Software Interrupt	SWI										3F	12	1	See Special Operations	•	•	•	•	•	•
Wait for Interrupt*	WAI										3E	9	1		•	⑪	•	•	•	•

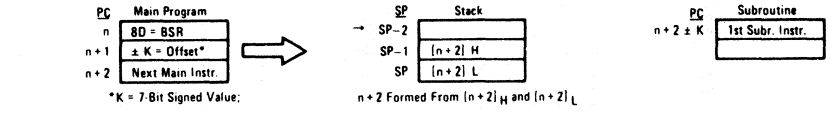
*WAI puts Address Bus, R/W, and Data Bus in the three-state mode while VMA is held low.

SPECIAL OPERATIONS

JSR, JUMP TO SUBROUTINE:

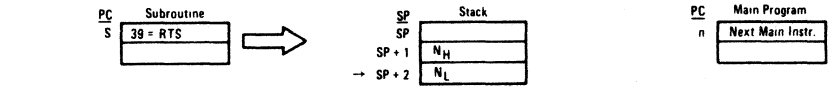


BSR, BRANCH TO SUBROUTINE:



JMP, JUMP:

RTS, RETURN FROM SUBROUTINE:



RTI, RETURN FROM INTERRUPT:

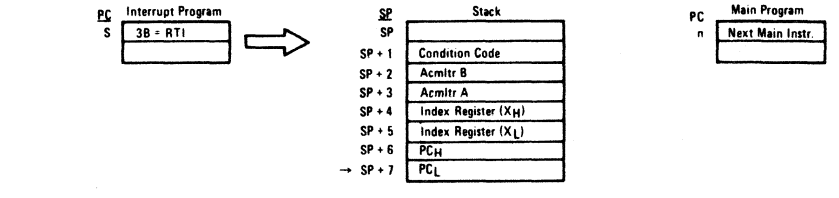


TABLE 6 – CONDITION CODE REGISTER MANIPULATION INSTRUCTIONS

		IMPLIED				COND. CODE REG.					
						5	4	3	2	1	0
OPERATIONS	MNEMONIC	OP	~	=	BOOLEAN OPERATION	H	I	N	Z	V	C
Clear Carry	CLC	0C	2	1	0 → C	•	•	•	•	•	R
Clear Interrupt Mask	CLI	0E	2	1	0 → I	•	R	•	•	•	•
Clear Overflow	CLV	0A	2	1	0 → V	•	•	•	•	R	•
Set Carry	SEC	0D	2	1	1 → C	•	•	•	•	•	S
Set Interrupt Mask	SEI	0F	2	1	1 → I	•	S	•	•	•	•
Set Overflow	SEV	0B	2	1	1 → V	•	•	•	•	S	•
Acmltr A → CCR	TAP	06	2	1	A → CCR	——— (12) ———					
CCR → Acmltr A	TPA	07	2	1	CCR → A	•	•	•	•	•	•

CONDITION CODE REGISTER NOTES: (Bit set if test is true and cleared otherwise)

1 (Bit V) Test: Result = 10000000?
2 (Bit C) Test: Result ≠ 00000000?
3 (Bit C) Test: Decimal value of most significant BCD Character greater than nine? (Not cleared if previously set.)
4 (Bit V) Test: Operand = 10000000 prior to execution?
5 (Bit V) Test: Operand = 01111111 prior to execution?
6 (Bit V) Test: Set equal to result of N⊕C after shift has occurred.
7 (Bit N) Test: Sign bit of most significant (MS) byte = 1?
8 (Cit V) Test: 2's complement overflow from subtraction of MS bytes?
9 (Bit N) Test: Result less than zero? (Bit 15 = 1)
10 (All) Load Condition Code Register from Stack. (See Special Operations)
11 (Bit I) Set when interrupt occurs. If previously set, a Non-Maskable Interrupt is required to exit the wait state.
12 (All) Set according to the contents of Accumulator A.

TABLE 7 – INSTRUCTION ADDRESSING MODES AND ASSOCIATED EXECUTION TIMES
(Times in Machine Cycles)

	(Dual Operand)	ACCX	Immediate	Direct	Extended	Indexed	Implied	Relative
ABA		•	•	•	•	•	2	•
ADC	x	•	2	3	4	5	•	•
ADD	x	•	2	3	4	5	•	•
AND	x	•	2	3	4	5	•	•
ASL		2	•	•	6	7	•	•
ASR		2	•	•	6	7	•	•
BCC		•	•	•	•	•	•	4
BCS		•	•	•	•	•	•	4
BEA		•	•	•	•	•	•	4
BGE		•	•	•	•	•	•	4
BGT		•	•	•	•	•	•	4
BHI		•	•	•	•	•	•	4
BIT	x	•	2	3	4	5	•	•
BLE		•	•	•	•	•	•	4
BLS		•	•	•	•	•	•	4
BLT		•	•	•	•	•	•	4
BMI		•	•	•	•	•	•	4
BNE		•	•	•	•	•	•	4
BPL		•	•	•	•	•	•	4
BRA		•	•	•	•	•	•	4
BSR		•	•	•	•	•	•	8
BVC		•	•	•	•	•	•	4
BVS		•	•	•	•	•	•	4
CBA		•	•	•	•	•	2	•
CLC		•	•	•	•	•	2	•
CLI		•	•	•	•	•	2	•
CLR		2	•	•	6	7	•	•
CLV		•	•	•	•	•	2	•
CMP	x	•	2	3	4	5	•	•
COM		2	•	•	6	7	•	•
CPX		•	3	4	5	6	•	•
DAA		•	•	•	•	•	2	•
DEC		2	•	•	6	7	•	•
DES		•	•	•	•	•	4	•
DEX		•	•	•	•	•	4	•
EOR	x	•	2	3	4	5	•	•

	(Dual Operand)	ACCX	Immediate	Direct	Extended	Indexed	Implied
INC		2	•	•	•	6	7	•
INS		•	•	•	•	•	•	4
INX		•	•	•	•	•	•	4
JMP		•	•	•	3	4	•	
JSR		•	•	•	9	8	•	
LDA	x	•	2	3	4	5	•	
LDS		•	3	4	5	6	•	
LDX		•	3	4	5	6	•	
LSR		2	•	•	6	7	•	
NEG		2	•	•	6	7	•	
NOP		•	•	•	•	•	•	2
ORA	x	•	2	3	4	5	•	
PSH		•	•	•	•	•	•	4
PUL		•	•	•	•	•	•	4
ROL		2	•	•	6	7	•	
ROR		2	•	•	6	7	•	
RTI		•	•	•	•	•	•	10
RTS		•	•	•	•	•	•	5
SBA		•	•	•	•	•	•	2
SBC	x	•	2	3	4	5	•	
SEC		•	•	•	•	•	•	2
SEI		•	•	•	•	•	•	2
SEV		•	•	•	•	•	•	2
STA	x	•	•	4	5	6	•	
STS		•	•	5	6	7	•	
STX		•	•	5	6	7	•	
SUB	x	•	2	3	4	5	•	
SWI		•	•	•	•	•	•	12
TAB		•	•	•	•	•	•	2
TAP		•	•	•	•	•	•	2
TBA		•	•	•	•	•	•	2
TPA		•	•	•	•	•	•	2
TST		2	•	•	6	7	•	
TSX		•	•	•	•	•	•	4
TSX		•	•	•	•	•	•	4
WAI		•	•	•	•	•	•	9

NOTE: Interrupt time is 12 cycles from the end of the instruction being executed, except following a WAI instruction. Then it is 4 cycles.

SUMMARY OF CYCLE BY CYCLE OPERATION

Table 8 provides a detailed description of the information present on the Address Bus, Data Bus, Valid Memory Address line (VMA), and the Read/Write line (R/W) during each cycle for each instruction.

This information is useful in comparing actual with expected results during debug of both software and hardware as the control program is executed. The information is categorized in groups according to Addressing Mode and Number of Cycles per instruction. (In general, instructions with the same Addressing Mode and Number of Cycles execute in the same manner; exceptions are indicated in the table.)

TABLE 8 – OPERATION SUMMARY

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
IMMEDIATE						
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	2	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Operand Data
CPX LDS LDX	3	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Operand Data (High Order Byte)
		3	1	Op Code Address + 2	1	Operand Data (Low Order Byte)
DIRECT						
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	3	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand
		3	1	Address of Operand	1	Operand Data
CPX LDS LDX	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand
		3	1	Address of Operand	1	Operand Data (High Order Byte)
		4	1	Operand Address + 1	1	Operand Data (Low Order Byte)
STA	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Destination Address
		3	0	Destination Address	1	Irrelevant Data (Note 1)
		4	1	Destination Address	0	Data from Accumulator
STS STX	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand
		3	0	Address of Operand	1	Irrelevant Data (Note 1)
		4	1	Address of Operand	0	Register Data (High Order Byte)
		5	1	Address of Operand + 1	0	Register Data (Low Order Byte)
INDEXED						
JMP	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	1	Index Register Plus Offset	1	Operand Data
CPX LDS LDX	6	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	1	Index Register Plus Offset	1	Operand Data (High Order Byte)
		6	1	Index Register Plus Offset + 1	1	Operand Data (Low Order Byte)

TABLE 8 – OPERATION SUMMARY (Continued)

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
INDEXED (Continued)						
STA	6	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	0	Index Register Plus Offset	1	Irrelevant Data (Note 1)
		6	1	Index Register Plus Offset	0	Operand Data
ASL LSR ASR NEG CLR ROL COM ROR DEC TST INC	7	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	1	Index Register Plus Offset	1	Current Operand Data
		6	0	Index Register Plus Offset	1	Irrelevant Data (Note 1)
		7	1/0 (Note 3)	Index Register Plus Offset	0	New Operand Data (Note 3)
STS STX	7	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	0	Index Register Plus Offset	1	Irrelevant Data (Note 1)
		6	1	Index Register Plus Offset	0	Operand Data (High Order Byte)
		7	1	Index Register Plus Offset + 1	0	Operand Data (Low Order Byte)
JSR	8	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer	0	Return Address (Low Order Byte)
		5	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		6	0	Stack Pointer – 2	1	Irrelevant Data (Note 1)
		7	0	Index Register	1	Irrelevant Data (Note 1)
		8	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
EXTENDED						
JMP	3	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Jump Address (High Order Byte)
		3	1	Op Code Address + 2	1	Jump Address (Low Order Byte)
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	1	Address of Operand	1	Operand Data
CPX LDS LDX	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	1	Address of Operand	1	Operand Data (High Order Byte)
		5	1	Address of Operand + 1	1	Operand Data (Low Order Byte)
STA A STA B	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Destination Address (High Order Byte)
		3	1	Op Code Address + 2	1	Destination Address (Low Order Byte)
		4	0	Operand Destination Address	1	Irrelevant Data (Note 1)
		5	1	Operand Destination Address	0	Data from Accumulator
ASL LSR ASR NEG CLR ROL COM ROR DEC TST INC	6	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	1	Address of Operand	1	Current Operand Data
		5	0	Address of Operand	1	Irrelevant Data (Note 1)
		6	1/0 (Note 3)	Address of Operand	0	New Operand Data (Note 3)

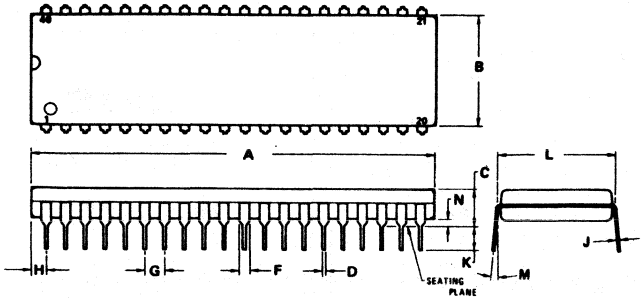
TABLE 8 – OPERATION SUMMARY (Continued)

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
EXTENDED (Continued)						
STS STX	6	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	0	Address of Operand	1	Irrelevant Data (Note 1)
		5	1	Address of Operand	0	Operand Data (High Order Byte)
		6	1	Address of Operand + 1	0	Operand Data (Low Order Byte)
JSR	9	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Subroutine (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Subroutine (Low Order Byte)
		4	1	Subroutine Starting Address	1	Op Code of Next Instruction
		5	1	Stack Pointer	0	Return Address (Low Order Byte)
		6	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		7	0	Stack Pointer – 2	1	Irrelevant Data (Note 1)
		8	0	Op Code Address + 2	1	Irrelevant Data (Note 1)
		9	1	Op Code Address + 2	1	Address of Subroutine (Low Order Byte)
INHERENT						
ABA DAA SEC ASL DEC SEI ASR INC SEV CBA LSR TAB CLC NEG TAP CLI NOP TBA CLR ROL TPA CLV ROR TST COM SBA	2	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
DES DEX INS INX	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	0	Previous Register Contents	1	Irrelevant Data (Note 1)
		4	0	New Register Contents	1	Irrelevant Data (Note 1)
PSH	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	1	Stack Pointer	0	Accumulator Data
		4	0	Stack Pointer – 1	1	Accumulator Data
PUL	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	0	Stack Pointer	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer + 1	1	Operand Data from Stack
TSX	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	0	Stack Pointer	1	Irrelevant Data (Note 1)
		4	0	New Index Register	1	Irrelevant Data (Note 1)
TXS	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	0	Index Register	1	Irrelevant Data
		4	0	New Stack Pointer	1	Irrelevant Data
RTS	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Irrelevant Data (Note 2)
		3	0	Stack Pointer	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer + 1	1	Address of Next Instruction (High Order Byte)
		5	1	Stack Pointer + 2	1	Address of Next Instruction (Low Order Byte)

TABLE 8 – OPERATION SUMMARY (Continued)

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
INHERENT (Continued)						
WAI	9	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	1	Stack Pointer	0	Return Address (Low Order Byte)
		4	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		5	1	Stack Pointer – 2	0	Index Register (Low Order Byte)
		6	1	Stack Pointer – 3	0	Index Register (High Order Byte)
		7	1	Stack Pointer – 4	0	Contents of Accumulator A
		8	1	Stack Pointer – 5	0	Contents of Accumulator B
		9	1	Stack Pointer – 6	1	Contents of Cond. Code Register
RTI	10	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Irrelevant Data (Note 2)
		3	0	Stack Pointer	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer + 1	1	Contents of Cond. Code Register from Stack
		5	1	Stack Pointer + 2	1	Contents of Accumulator B from Stack
		6	1	Stack Pointer + 3	1	Contents of Accumulator A from Stack
		7	1	Stack Pointer + 4	1	Index Register from Stack (High Order Byte)
		8	1	Stack Pointer + 5	1	Index Register from Stack (Low Order Byte)
		9	1	Stack Pointer + 6	1	Next Instruction Address from Stack (High Order Byte)
		10	1	Stack Pointer + 7	1	Next Instruction Address from Stack (Low Order Byte)
SWI	12	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Irrelevant Data (Note 1)
		3	1	Stack Pointer	0	Return Address (Low Order Byte)
		4	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		5	1	Stack Pointer – 2	0	Index Register (Low Order Byte)
		6	1	Stack Pointer – 3	0	Index Register (High Order Byte)
		7	1	Stack Pointer – 4	0	Contents of Accumulator A
		8	1	Stack Pointer – 5	0	Contents of Accumulator B
		9	1	Stack Pointer – 6	0	Contents of Cond. Code Register
		10	0	Stack Pointer – 7	1	Irrelevant Data (Note 1)
		11	1	Vector Address FFFA (Hex)	1	Address of Subroutine (High Order Byte)
		12	1	Vector Address FFFB (Hex)	1	Address of Subroutine (Low Order Byte)
RELATIVE						
BCC BHI BNE BCS BLE BPL BEQ BLS BRA BGE BLT BVC BGT BMI BVS	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Branch Offset
		3	0	Op Code Address + 2	1	Irrelevant Data (Note 1)
		4	0	Branch Address	1	Irrelevant Data (Note 1)
BSR	8	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Branch Offset
		3	0	Return Address of Main Program	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer	0	Return Address (Low Order Byte)
		5	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		6	0	Stack Pointer – 2	1	Irrelevant Data (Note 1)
		7	0	Return Address of Main Program	1	Irrelevant Data (Note 1)
		8	0	Subroutine Address	1	Irrelevant Data (Notes 1, 4)

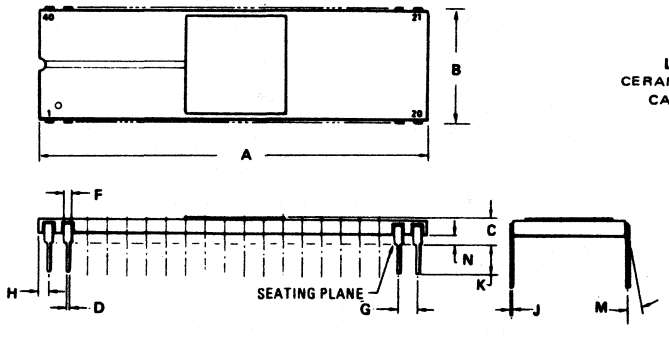
Note 1. If device which is addressed during this cycle uses VMA, then the Data Bus will go to the high impedance three-state condition. Depending on bus capacitance, data from the previous cycle may be retained on the Data Bus.
Note 2. Data is ignored by the MPU.
Note 3. For TST, VMA = 0 and Operand data does not change.
Note 4. MS Byte of Address Bus = MS Byte of Address of BSR Instruction and LS Byte of Address Bus = LS Byte of Sub-Routine Address.



P SUFFIX
PLASTIC PACKAGE
CASE 711-03

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	51.82	52.32	2.040	2.060
B	13.72	14.22	0.540	0.560
C	3.94	5.08	0.155	0.200
D	0.36	0.56	0.014	0.022
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.38	0.008	0.015
K	2.92	3.43	0.115	0.135
L	15.24 BSC		0.600 BSC	
M	0°	15°	0°	15°
N	0.51	1.02	0.020	0.040

NOTES:
1. POSITIONAL TOLERANCE OF LEADS (D), SHALL BE WITHIN 0.25 mm (0.010) AT MAXIMUM MATERIAL CONDITION, IN RELATION TO SEATING PLANE AND EACH OTHER.
2. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSION B DOES NOT INCLUDE MOLD FLASH.
4. CASE OUTLINE 711-02 IS OBSOLETE. NEW STANDARD IS 711-03.



L SUFFIX
CERAMIC PACKAGE
CASE 715-02

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	50.29	51.31	1.980	2.020
B	14.86	15.62	0.585	0.615
C	2.54	4.19	0.100	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.33	0.008	0.013
K	2.54	4.19	0.100	0.165
M	0°	10°	0°	10°
N	0.51	1.52	0.020	0.060

NOTE:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA (AT SEATING PLANE), AT MAX. MAT'L CONDITION.

MOTOROLA

MC6803
Internal-Clock

MC6803E
External-Clock

Product Preview

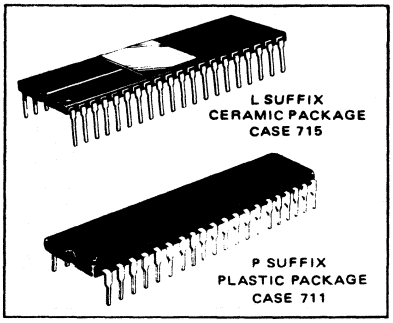
The MC6803 is an 8-bit microcomputer which employs a multiplexed address and data system allowing expandability to 65K words. The MC6803 is object code compatible with the M6800 instruction set and includes improved execution times of key instructions. There are several new 16-bit and 8-bit instructions including an 8 by 8 multiply with 16-bit result. The MC6803 has 128 bytes of RAM, internal clock, UART, parallel I/O, and three function 16-bit timer all on-board. The MC6803E version is also available for use with an external clock. The MC6803 internal clock version requires only the addition of an external crystal for MCU operation. The MC6803 internal clock's divide by four circuitry allows for use of the inexpensive 3.58 MHz color burst crystal. The MC6803 MCU is fully TTL compatible and requires only one +5.0 volt power supply.

- Expanded M6800 Instruction Set
- Full Object Code Compatibility With M6800 MPU's
- Multiplexed Address and Data
- Compatible With Existing M6800 Peripherals
- 8 x 8 Multiply With 16-bit Result
- Up to 13 Parallel I/O Lines
- 128 Bytes On-Board RAM
- On-Board RAM Retainable With V_{CC} Standby
- UART (Serial I/O) On-Board
- 16-bit Timer On-Board
- Internal Clock/Divide by Four Circuitry (MC6803)
- External Clock/Divide by One Circuitry (MC6803E)
- Full TTL Compatibility
- Full Interrupt Capability

MOS

(N-CHANNEL, SILICON-GATE, DEPLETION LOAD)

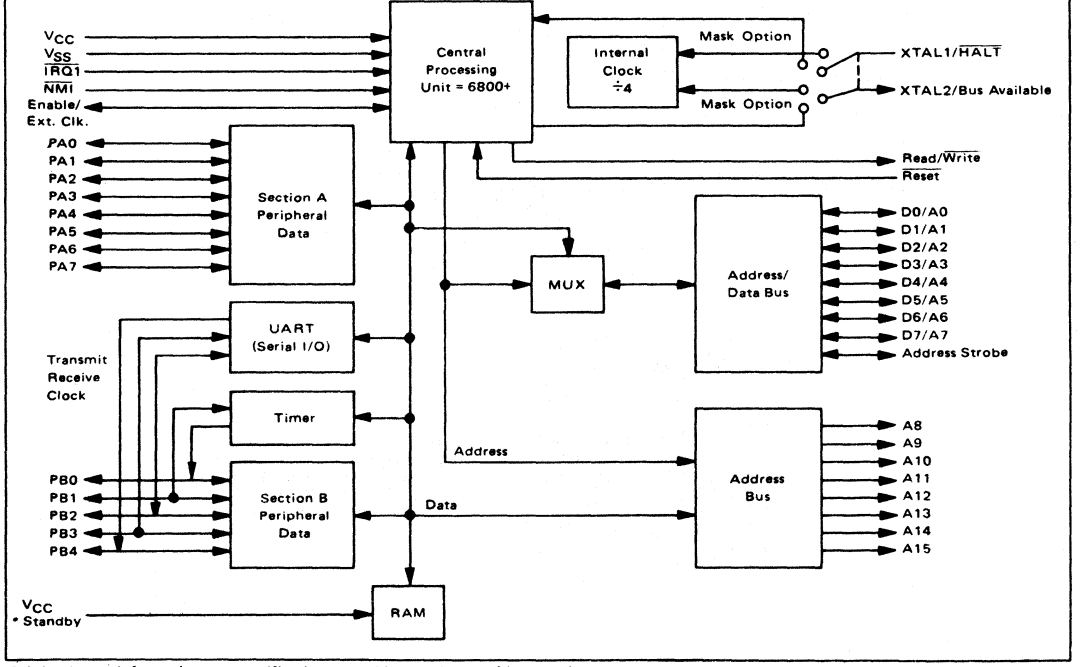
MICROCOMPUTER



L SUFFIX
CERAMIC PACKAGE
CASE 715

P SUFFIX
PLASTIC PACKAGE
CASE 711

FIGURE 1 – BLOCK DIAGRAM



This is advance information and specifications are subject to change without notice.

GENERAL DESCRIPTION

The MC6803 MCU is a completely functional computer that includes a clock, microprocessing unit (MPU), data storage memory, and Input/Output on a single NMOS LSI chip in a 40-pin package. The MC6803 will have an internal clock which will require only the addition of a crystal for operation. This internal clock circuitry will divide the crystal frequency by four allowing for use of the inexpensive 3.58 MHz color burst crystal. The MC6803E version requires the addition of an external clock oscillator for operation.

MC6803/MC6803E Signal Description

Address/Data Bus (A0-A7/D0-D7) – Eight pins are used for the multiplexed address and data bus. These pins provide the lower order address lines plus the eight bit data bus. The data bus is bidirectional for the transfer of data to and from the memory and peripheral devices.

Address Bus (A8-A15) – Eight pins are used for the higher order address for expandability to a full 65K words.

$\overline{\text{Halt}}$ – This input will halt all activity in the machine when it is in the low state. In the halt mode, the machine will stop at the end of an instruction putting Bus Available in its active or true state. The $\overline{\text{Halt}}$ input is available on the MC6803E only. This line is XTAL 1 on the MC6803.

Bus Available (BA) – The Bus Available signal is true to indicate that the microcomputer has stopped and that the address bus is available. The address bus does not become tri-state during halt mode. This will occur if the $\overline{\text{Halt}}$ line is low or if the processor is in the WAIT state as a result of a WAIT instruction. The Bus Available line is available on the MC6803E only. This line is XTAL 2 on the MC6803.

XTAL 1, XTAL 2 – These are inputs on the MC6803 for crystal connection. The crystal frequency will be input through a divide by four circuitry since a 4 MHz crystal is less expensive than a comparable 1 MHz crystal. The XTAL 1 input may be driven by an external TTL or CMOS clock source.

Read/Write (R/$\overline{\text{W}}$) – This output signals the peripherals and memory devices that the MCU is in a Read (high) or Write (low) state.

$\overline{\text{IRQ1}}$ (External $\overline{\text{Interrupt}}$ $\overline{\text{Request}}$) – This input requests that an interrupt sequence be generated within the machine. The MCU will wait until it completes the current instruction that is being executed before it recognizes the request. This input can be masked by software.

$\overline{\text{Reset}}$ – This input is used to reset and start the MCU from a power down condition resulting from a power failure or an initial start-up of the processor.

$\overline{\text{Non-Maskable}}$ $\overline{\text{Interrupt}}$ ($\overline{\text{NMI}}$) – This input requests that a non-mask-interrupt sequence be generated within the processor. As with the $\overline{\text{IRQ1}}$ signal, the processor will complete the current instruction that is being executed before it recognizes the $\overline{\text{NMI}}$ signal. This input cannot be masked by the MCU.

V_{CC} Standby – This pin will supply V_{CC} to the standby RAM on the chip.

V_{CC}, V_{SS} – These two pins supply power and ground to the MCU. The power supplied should be +5.0 volts ±5%.

Enable (E) – This pin will be an output on the MC6803 and will supply the external clock for the rest of the system. This pin is an input on the MC6803E and is used as an input clock with no divide down circuitry.

MC6803/MC6803E PERIPHERAL INTERFACE LINES

The MC6803 provides 8-bit and 5-bit bidirectional busses for interfacing to peripheral devices.

Peripheral Interface A (PA0-PA7) – Each of these peripheral data lines can be programmed to act as an input or output. This is accomplished by setting a "1" in the corresponding Data Direction Register for those lines which are to act as outputs. A "0" in a bit of the Data Direction Register causes the corresponding peripheral data line to act as an input. A register is provided to read and write the Peripheral Interface A lines. A logical "1" written into the register will cause a "high" on the corresponding data line while a "0" results in a low.

Peripheral Interface B (PB0-PB4) – These peripheral data lines can be programmed to act as either inputs or outputs in a similar manner to PA0-PA7. These lines also provide access to the Timer and the UART (Serial I/O). The Timer has two associated lines: 1) Timer Input, 2) Timer Output. The UART (Serial I/O) has three associated lines: 1) Transmitter, 2) Receiver and 3) Clock. Both the Timer and the UART have associated control registers which allow for their selection and access on the Peripheral Interface B lines.

MC6803/MC6803E UART (SERIAL I/O)

The MC6803 and MC6803E have a serial input and output capability with an internal bit rate generator derived from the external crystal or oscillator. This I/O function is software programmable for two types of data formats and full duplex, half duplex, or simplex operation. The two data formats available are the standard mark/space (NRZ) and bi-phase. There are four software addressable registers provided to configure this section, provide status information, and to transmit and receive the serial data. If the internal bit rate is not needed, an external bit rate can be provided by the user. All formats have one start bit and one stop bit with an 8-bit data word.

MC6803/MC6803E TIMER

The MC6803 includes a 16-bit timer with three effectively independent timing functions. The timer is implemented

using a synchronous 16-bit free running counter driven by the $\phi 2$ clock of the MPU, with two additional registers: 1) Input Capture Register and 2) Output Compare Register. The three timing functions are: 1) variable pulse width measurement, 2) variable pulse width generation, and 3) a timer overflow–fixed time flag. These functions allow the MC6803 to be used in any application which requires very accurate time measurement such as industrial controls, automotive systems, A/D converters, modems, real time clocks, and digital VCO's.

MC6803/MC6803E Programming Model

The MC6803 MCU implements the full MC6800 instruction set plus several new instructions. In addition, execution times of key instructions have been reduced to increase throughput. The programming model of the microcomputer is shown in Figure 2.

FIGURE 2 – MC6803 MCU PROGRAMMING MODEL

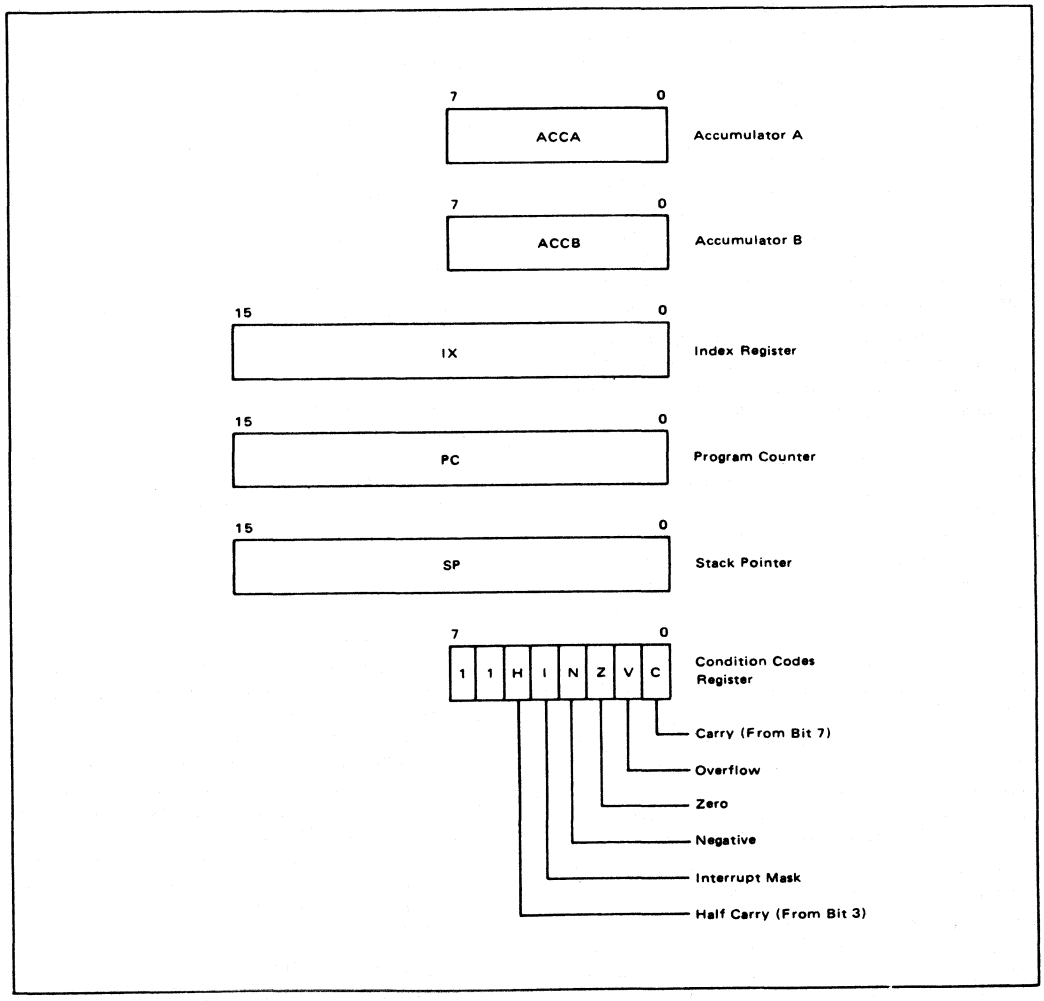


TABLE 1 – MC6803 NEW INSTRUCTIONS

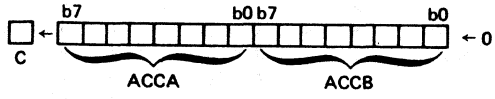
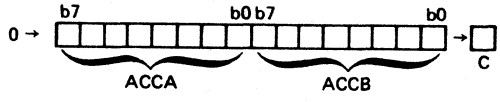
ABX	Adds the 8-bit unsigned accumulator B to the 16-bit X-Register taking into account the possible carry out of the low order byte of the X-Register.	IX ← IX + ACCB
ADDD	Adds the double precision ACCAB to the double precision value M:M+1 and places the results in ACCAB.	ACCAB ← (ACCAB) + (M:M+1)
ASLD	Shifts all bits of ACCAB one place to the left. Bit 0 is loaded with a zero. The C bit is loaded from the most significant bit of ACCAB.	
LDD	Loads the contents of double precision memory location into the double accumulator A:B. The condition codes are set according to the data.	ACCAB ← (M:M+1)
LSRD	Shifts all bits of ACCAB one place to the right. Bit 15 is loaded with zero. The C bit is loaded from the least significant bit of ACCAB.	
MUL	Multiplies the 8 bits in accumulator A with the 8 bits in accumulator B to obtain a 16-bit unsigned number in A:B.ACCA contains MSB of result.	ACCAB ← ACCA * ACCB
PSHX	The contents of the index register is pushed onto the stack at the address contained in the stack pointer. The stack pointer is decremented by 2.	↓ (IXL), SP ← (SP) – 0001 ↓ (IXH), SP ← (SP) – 0001
PULX	The index register is pulled from the stack beginning at the current address contained in the current address contained in the stack pointer +1. The stack pointer is incremented by 2 in total.	SP ← (SP) + 1; ↑ IXH SP ← (SP) + 1; ↑ IXL
STD	Stores the contents of double accumulator A:B in memory. The contents of ACCAB remain unchanged.	M:M+1 ← (ACCAB)
SUBD	Subtracts the contents of M:M+1 from the contents of double accumulator AB and places the result in ACCAB.	ACCAB ← (ACCAB) – (M:M+1)

TABLE 2 – MC6803 MCU PROGRAMMING MODE

The execution time of key instructions has been reduced.
Table 2 shows instruction execution times in machine cycles.

	ACCX	Immediate	Direct	Extended	Indexed	Inherent	Relative
ABA	•	•	•	•	•	2	•
ABX	•	•	•	•	•	3	•
ADC	•	2	3	4	4	•	•
ADD	•	2	3	4	4	•	•
ADDD	•	4	5	6	6	•	•
AND	•	2	3	4	4	•	•
ASL	2	•	•	6	6	•	•
ASLD	•	•	•	•	•	3	•
ASR	2	•	•	6	6	•	•
BCC	•	•	•	•	•	•	3
BCS	•	•	•	•	•	•	3
BEQ	•	•	•	•	•	•	3
BGE	•	•	•	•	•	•	3
BGT	•	•	•	•	•	•	3
BHI	•	•	•	•	•	•	3
BIT	•	2	3	4	4	•	•
BLE	•	•	•	•	•	•	3
BLS	•	•	•	•	•	•	3
BLT	•	•	•	•	•	•	3
BMI	•	•	•	•	•	•	3
BNE	•	•	•	•	•	•	3
BPL	•	•	•	•	•	•	3
BRA	•	•	•	•	•	•	3
BSR	•	•	•	•	•	•	6
BVC	•	•	•	•	•	•	3
BVS	•	•	•	•	•	•	3
CBA	•	•	•	•	•	2	•
CLC	•	•	•	•	•	2	•
CLI	•	•	•	•	•	2	•
CLR	2	•	•	6	6	•	•
CLV	•	•	•	•	•	2	•
CMP	•	2	3	4	4	•	•
COM	2	•	•	6	6	•	•
CPX	•	4	5	6	6	•	•
DAA	•	•	•	•	•	2	•
DEC	2	•	•	6	6	•	•
DES	•	•	•	•	•	3	•
DEX	•	•	•	•	•	3	•
EOR	•	2	3	4	4	•	•
INC	2	•	•	6	6	•	•
INS	•	•	•	•	•	3	•

	ACCX	Immediate	Direct	Extended	Indexed	Inherent	Relative
INX	•	•	•	•	•	3	•
JMP	•	•	•	3	3	•	•
JSR	•	•	5	6	6	•	•
LDA	•	2	3	4	4	•	•
LDD	•	3	4	5	5	•	•
LDS	•	3	4	5	5	•	•
LDX	•	3	4	5	5	•	•
LSR	2	•	•	6	6	•	•
LSRD	•	•	•	•	•	3	•
MUL	•	•	•	•	•	10	•
NEG	2	•	•	6	6	•	•
NOP	•	•	•	•	•	2	•
ORA	•	2	3	4	4	•	•
PSH	3	•	•	•	•	•	•
PSHX	•	•	•	•	4	•	•
PUL	4	•	•	•	•	•	•
PULX	•	•	•	•	•	5	•
ROL	2	•	6	6	•	•	•
ROR	2	•	6	6	•	•	•
RTI	•	•	•	•	•	10	•
RTS	•	•	•	•	•	5	•
SBA	•	•	•	•	•	2	•
SBC	•	2	3	4	4	•	•
SEC	•	•	•	•	•	2	•
SEI	•	•	•	•	•	2	•
SEV	•	•	•	•	•	2	•
STA	•	•	3	4	4	•	•
STD	•	•	4	5	5	•	•
STS	•	•	4	5	5	•	•
STX	•	•	4	5	5	•	•
SUB	•	2	3	4	4	•	•
SUBD	•	4	5	6	6	•	•
SWI	•	•	•	•	•	12	•
TAB	•	•	•	•	•	2	•
TAP	•	•	•	•	•	2	•
TBA	•	•	•	•	•	2	•
TPA	•	•	•	•	•	2	•
TST	2	•	•	6	6	•	•
TSX	•	•	•	•	•	3	•
TXS	•	•	•	•	•	3	•
WAI	•	•	•	•	•	9	•

An MC6803 and an MC6803E system configuration are shown in figures 3 and 4 respectively. The MC6803 MCU is expandable to a full 65K words allowing for a system containing several peripherals plus memory.

FIGURE 3 – MC6803 SYSTEM CONFIGURATION

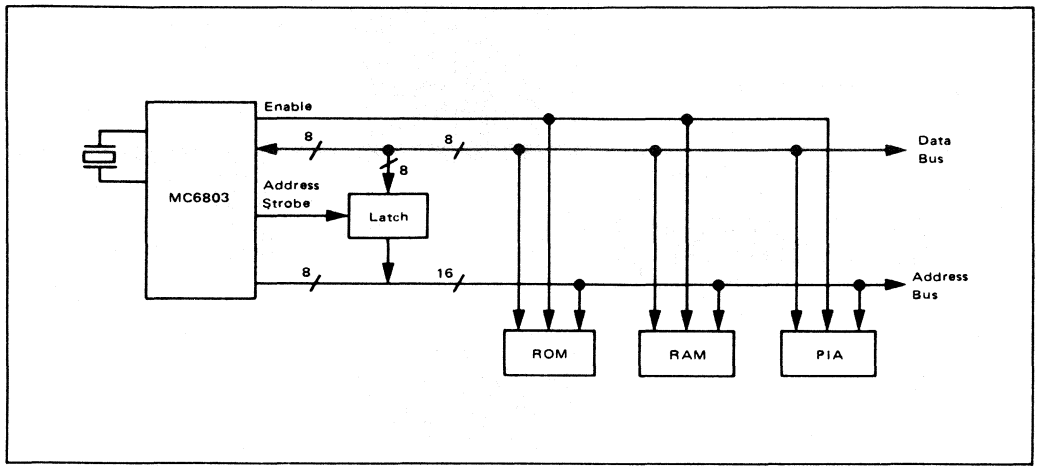
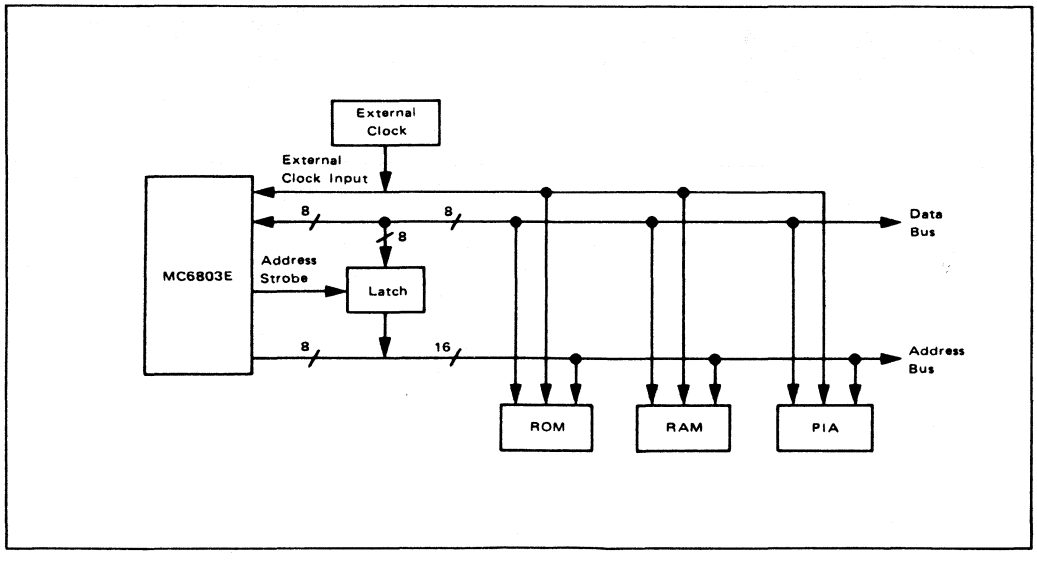
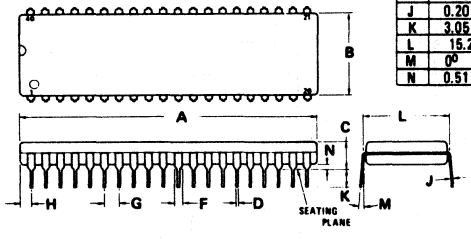


FIGURE 4 – MC6803E SYSTEM CONFIGURATION



OUTLINE DIMENSIONS

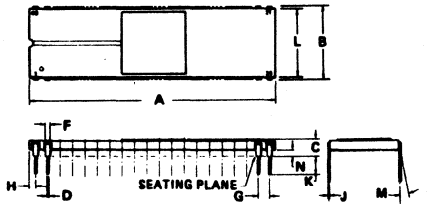
DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	51.82	52.32	2.040	2.060
B	13.72	14.22	0.540	0.560
C	4.57	5.08	0.180	0.200
D	0.36	0.51	0.014	0.020
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	3.05	3.56	0.120	0.140
L	15.24 BSC		0.600 BSC	
M	0°	10°	0°	10°
N	0.51	1.02	0.020	0.040



P SUFFIX
PLASTIC PACKAGE
CASE 711-02

NOTES:
1. LEADS TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION (DIM "D").
2. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	50.29	51.31	1.980	2.020
B	14.86	15.62	0.585	0.615
C	2.54	4.19	0.100	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.33	0.008	0.013
K	2.54	4.19	0.100	0.165
L	14.60	15.37	0.575	0.605
M	0°	10°	0°	10°
N	0.51	1.52	0.020	0.060



CASE 715-02

L SUFFIX
CERAMIC PACKAGE
CASE 715-02

NOTE:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA (AT SEATING PLANE), AT MAX. MAT'L CONDITION.

MOTOROLA

MC6808

Advance Information

MICROPROCESSOR WITH CLOCK

The MC6808 is a monolithic 8-bit microprocessor that contains all the registers and accumulators of the present MC6800 plus an internal clock oscillator and driver on the same chip.

The MC6808 is completely software-compatible with the MC6800 as well as the entire M6800 family of parts. Hence the MC6808 is expandable to 65K words.

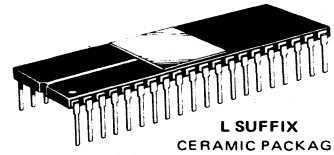
This very cost-effective MPU allows the designer to use the MC6808 in consumer as well as industrial applications without sacrificing industrial specifications.

- On-Chip Clock Circuit
- Software-Compatible with the MC6800
- Expandable to 65K words
- Standard TTL-Compatible Inputs and Outputs
- 8-Bit Word Size
- 16-Bit Memory Addressing
- Interrupt Capability

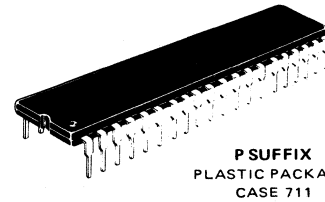
MOS

(N-CHANNEL, SILICON-GATE, DEPLETION LOAD)

MICROPROCESSOR WITH CLOCK

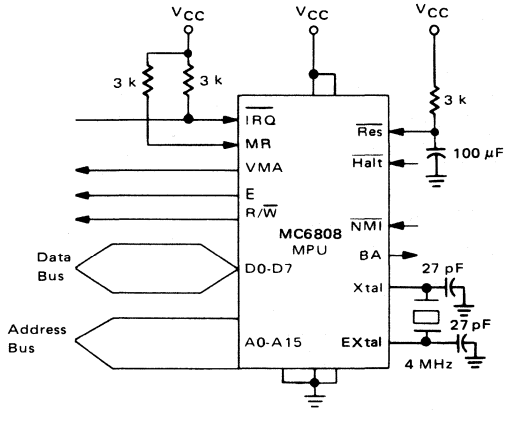


L SUFFIX
CERAMIC PACKAGE
CASE 715

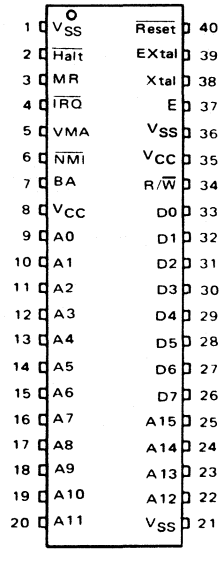


P SUFFIX
PLASTIC PACKAGE
CASE 711

FIGURE 1 – TYPICAL MICROPROCESSOR INTERFACE



PIN ASSIGNMENT



This is advance information and specifications are subject to change without notice.

ADI-805

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	-0.3 to +7.0	Vdc
Input Voltage	V_{in}	-0.3 to +7.0	Vdc
Operating Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	-55 to +150	°C
Thermal Resistance Plastic Ceramic	θ_{JA}	 100 50	°C/W

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit.

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ± 5%, V_{SS} = 0, T_A = 0 to 70°C unles otherwise noted.)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage	Logic, EXtal $\overline{Reset}$	V_{IH}	V_{SS} + 2.0 V_{SS} + 4.0	– –	V_{CC} V_{CC}	Vdc
Input Low Voltage	Logic, EXtal $\overline{Reset}$	V_{IL}	V_{SS} – 0.3 V_{SS} – 0.3	–	V_{SS} + 0.8 V_{SS} + 2.3	Vdc
Input Leakage Current (V_{in} = 0 to 5.25 V, V_{CC} = max)	Logic*	I_{in}	–	1.0	2.5	μAdc
Output High Voltage (I_{Load} = -205 μAdc, V_{CC} = min) (I_{Load} = -145 μAdc, V_{CC} = min) (I_{Load} = -100 μAdc, V_{CC} = min)	 D0-D7 A0-A15, R/$\overline{W}$, VMA, E BA	V_{OH}	 V_{SS} + 2.4 V_{SS} + 2.4 V_{SS} + 2.4	 – – –	 – – –	Vdc
Output Low Voltage (I_{Load} = 1.6 mAdc, V_{CC} = min)		V_{OL}	–	–	V_{SS} + 0.4	Vdc
Power Dissipation		P_D**	–	0.600	1.2	W
Capacitance # (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	 D0-D7 Logic Inputs, EXtal A0-A15, R/$\overline{W}$, VMA	C_{in} C_{out}	 – – –	 10 6.5 –	 12.5 10 12	pF pF
Frequency of Operation (Input Clock ÷ 4) (Crystal Frequency)		f f_{Xtal}	0.1 1.0	– –	1.0 4.0	MHz
Clock Timing Cycle Time Clock Pulse Width (measured at 2.4V) (measured at 0.4V) Fall Time (Measured between V_{SS} + 0.4 V and V_{SS} + 2.4 V)		 t_{cyc} $PW_{\phi H}$ $PW_{\phi L}$ t_ϕ	 1.0 450 450 –	 –	 10 4500 4500 25	 μs ns ns

*Except $\overline{IRQ}$ and $\overline{NMI}$, which require 3 kΩ pullup load resistors for wire-OR capability at optimum operation. Does not include EXtal and Xtal, which are crystal inputs.

#Capacitances are periodically sampled rather than 100% tested.

READ/WRITE TIMING (Figures 2 through 6; Load Circuit of Figure 4.)

Characteristic	Symbol	Min	Typ	Max	Unit
Address Delay	t_{AD}	–	–	270	ns
Peripheral Read Access Time $t_{acc} = t_{ut} - (t_{AD} + t_{DSR})$; $t_{ut} = t_{cyc} - t_\phi$	t_{acc}	–	–	530	ns
Data Setup Time (Read)	t_{DSR}	100	–	–	ns
Input Data Hold Time	t_H	10	–	–	ns
Output Data Hold Time	t_H	30	–	–	ns
Address Hold Time (Address, R/$\overline{W}$, VMA)	t_{AH}	20	–	–	ns
Data Delay Time (Write)	t_{DDW}	–	165	225	ns
Processor Controls Processor Control Setup Time Processor Control Rise and Fall Time (Measured between 0.8 V and 2.0 V)	 t_{PCS} t_{PCr}, t_{PCf}	 200 –	 – –	 – 100	 ns ns
Bus Available Delay Time	t_{BA}	–	–	250	ns

FIGURE 2 – READ DATA FROM MEMORY OR PERIPHERALS

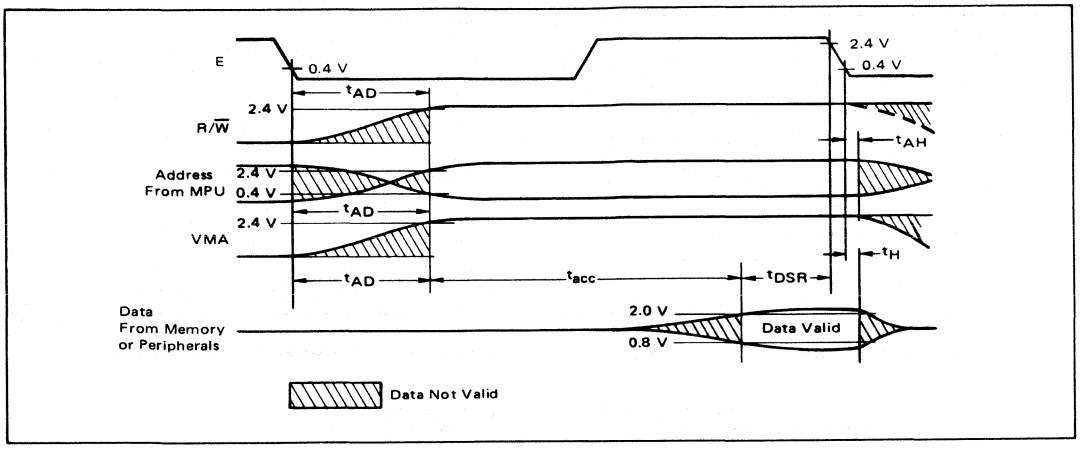


FIGURE 3 – WRITE DATA IN MEMORY OR PERIPHERALS

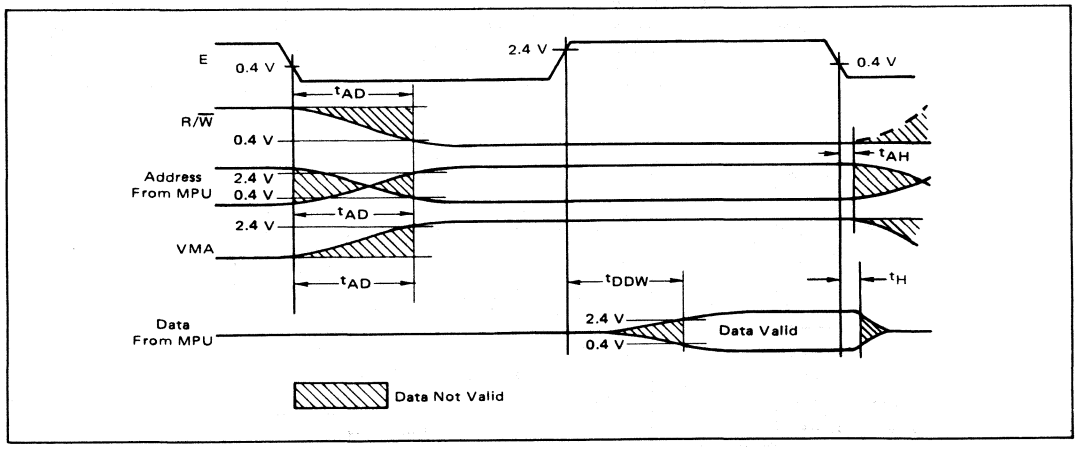


FIGURE 4 – BUS TIMING TEST LOAD

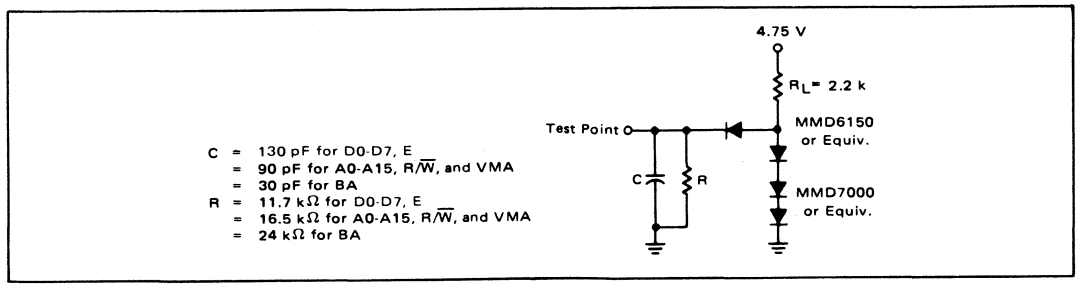


FIGURE 5 – TYPICAL DATA BUS OUTPUT DELAY versus CAPACITIVE LOADING

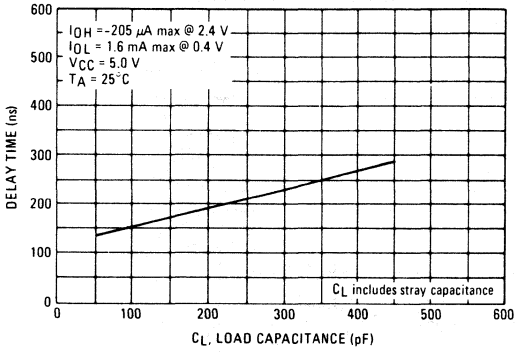


FIGURE 6 – TYPICAL READ/WRITE, VMA, AND ADDRESS OUTPUT DELAY versus CAPACITIVE LOADING

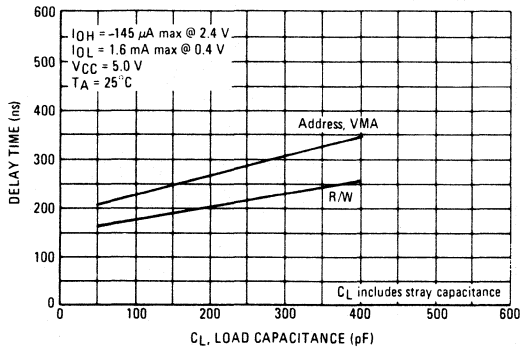
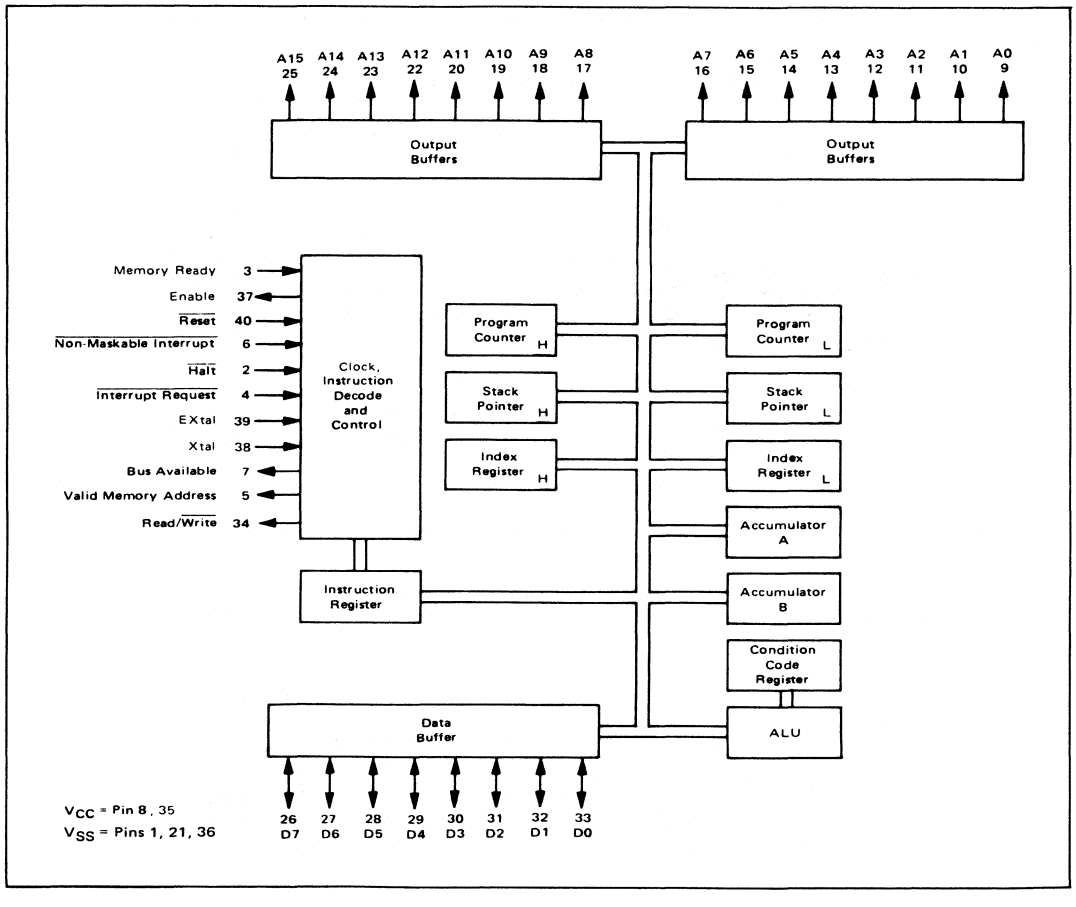


FIGURE 7 – MC6808 EXPANDED BLOCK DIAGRAM



MPU REGISTERS

A general block diagram of the MC6808 is shown in Figure 7. As shown, the number and configuration of the registers are the same as for the MC6800.

The MPU has three 16-bit registers and three 8-bit registers available for use by the programmer (Figure 8).

Program Counter – The program counter is a two byte (16-bits) register that points to the current program address.

Stack Pointer – The stack pointer is a two byte register that contains the address of the next available location in an external push-down/pop-up stack. This stack is normally a random access Read/Write memory that may have any location (address) that is convenient. In those applications that require storage of information in the stack when power is lost, the stack must be non-volatile.

Index Register – The index register is a two byte register that is used to store data or a sixteen bit memory address for the Indexed mode of memory addressing.

Accumulators – The MPU contains two 8-bit accumulators that are used to hold operands and results from an arithmetic logic unit (ALU).

Condition Code Register – The condition code register indicates the results of an Arithmetic Logic Unit operation: Negative (N), Zero (Z), Overflow (V), Carry from bit 7 (C), and half carry from bit 3 (H). These bits of the Condition Code Register are used as testable conditions for the conditional branch instructions. Bit 4 is the interrupt mask bit (I). The used bits of the Condition Code Register (b6 and b7) are ones.

Figure 9 shows the order of saving the microprocessor status within the stack.

FIGURE 8 – PROGRAMMING MODEL OF THE MICROPROCESSING UNIT

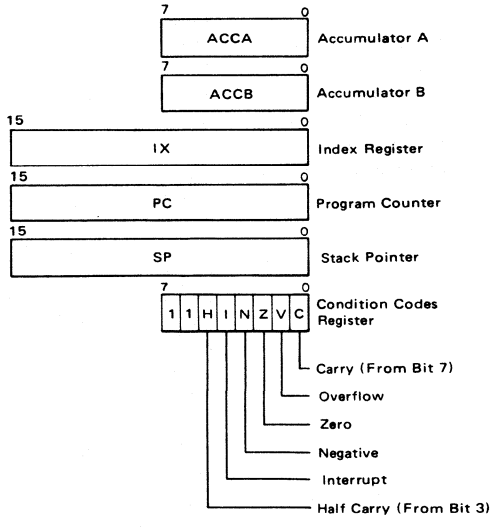
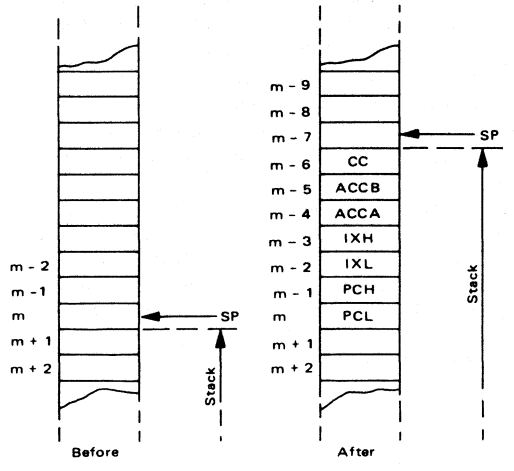


FIGURE 9 – SAVING THE STATUS OF THE MICROPROCESSOR IN THE STACK

SP = Stack Pointer
CC = Condition Codes (Also called the Processor Status Byte)
ACCB = Accumulator B
ACCA = Accumulator A
IXH = Index Register, Higher Order 8 Bits
IXL = Index Register, Lower Order 8 Bits
PCH = Program Counter, Higher Order 8 Bits
PCL = Program Counter, Lower Order 8 Bits



MC6808 MPU SIGNAL DESCRIPTION

Proper operation of the MPU requires that certain control and timing signals be provided to accomplish specific functions and that other signal lines be monitored to determine the state of the processor. These control and timing signals for the MC6808 are similar to those of the MC6800 except that TSC, DBE, $\phi 1$, $\phi 2$ input, and two unused pins have been eliminated, and the following signal and timing lines have been added:

Crystal Connections EXtal and Xtal
Memory Ready (MR)
Enable $\phi 2$ Output (E)

The following is a summary of the MC6808 MPU signals:

Address Bus (A0-A15) – Sixteen pins are used for the address bus. The outputs are capable of driving one standard TTL load and 90 pF.

Data Bus (D0-D7) – Eight pins are used for the data bus. It is bidirectional, transferring data to and from the memory and peripheral devices. It also has three-state output buffers capable of driving one standard TTL load and 130 pF.

$\overline{\text{Halt}}$ – When this input is in the low state, all activity in the machine will be halted. This input is level sensitive. In the halt mode, the machine will stop at the end of an instruction, Bus Available will be at a high state, Valid Memory Address will be at a low state, and all other three-state lines will be in the three-state mode. The address bus will display the address of the next instruction.

To insure single instruction operation, transition of the $\overline{\text{Halt}}$ line must not occur during the last 200 ns of E and the $\overline{\text{Halt}}$ line must go high for one Clock cycle.

Read/$\overline{\text{Write}}$ (R/$\overline{\text{W}}$) – This TTL compatible output signals the peripherals and memory devices whether the MPU is in a Read (high) or Write (low) state. The normal standby state of this signal is Read (high). When the processor is halted, it will be in the logical one state. This output is capable of driving one standard TTL load and 90 pF.

Valid Memory Address (VMA) – This output indicates to peripheral devices that there is a valid address on the address bus. In normal operation, this signal should be utilized for enabling peripheral interfaces such as the PIA and ACIA. This signal is not three-state. One standard TTL load and 90 pF may be directly driven by this active high signal.

Bus Available (BA) — The Bus Available signal will normally be in the low state; when activated, it will go to the high state indicating that the microprocessor has stopped and that the address bus is available but not in three-state. This will occur if the $\overline{\text{Halt}}$ line is in the low state or the processor is in the WAIT state as a result of the execution of a WAIT instruction. At such time, all three-state output drivers will go to their off state and other outputs to their normally inactive level. The processor is removed from the WAIT state by the occurrence of a maskable (mask bit I = 0) or nonmaskable interrupt. This output is capable of driving one standard TTL load and 30 pF.

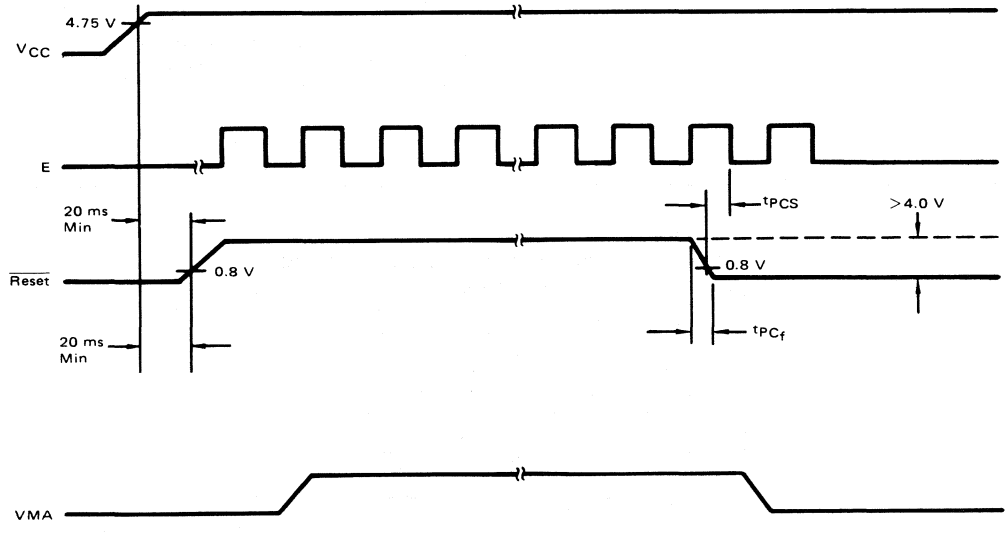
$\overline{\text{Interrupt Request}}$ ($\overline{\text{IRQ}}$) — This level sensitive input requests that an interrupt sequence be generated within the machine. The processor will wait until it completes the current instruction that is being executed before it recognizes the request. At that time, if the interrupt mask bit in the Condition Code Register is not set, the machine will begin an interrupt sequence. The Index Register, Program Counter, Accumulators, and Condition Code Register are stored away on the stack. Next the MPU will respond to the interrupt request by setting the interrupt mask bit high so that no further interrupts may occur. At the end of the cycle, a 16-bit address will be loaded that points to a vectoring address which is located in memory locations FFF8 and FFF9. An address loaded at these locations causes the MPU to branch to an interrupt routine in memory.

The $\overline{\text{Halt}}$ line must be in the high state for interrupts to be serviced. Interrupts will be latched internally while $\overline{\text{Halt}}$ is low.

The $\overline{\text{IRQ}}$ has a high impedance pullup device internal to the chip; however a 3 kΩ external resistor to V_{CC} should be used for wire-OR and optimum control of interrupts.

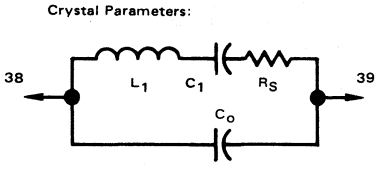
$\overline{\text{Reset}}$ — This input is used to reset and start the MPU from a power down condition, resulting from a power failure or an initial start-up of the processor. When this line is low, the MPU is inactive and the information in the registers will be lost. If a high level is detected on the input, this will signal the MPU to begin the restart sequence. This will start execution of a routine to initialize the processor from its reset condition. All the higher order address lines will be forced high. For the restart, the last two (FFFE, FFFF) locations in memory will be used to load the program that is addressed by the program counter. During the restart routine, the interrupt mask bit is set and must be reset before the MPU can be interrupted by $\overline{\text{IRQ}}$. Power-up and reset timing sequences are shown in Figure 10.

FIGURE 10 — POWER-UP AND RESET TIMING



EXtal and Xtal — The MC 6808 has an internal oscillator that may be crystal controlled. These connections are for a parallel resonant fundamental crystal. (AT cut.) A divide-by-four circuit has been added to the MC6808 so that a 4 MHz crystal may be used in lieu of a 1 MHz crystal for a more cost-effective system. Pin 39 of the MC6808 may be driven externally by a TTL input signal if a separate clock is required. Pin 38 is to be left open in this mode. Crystal parameters to be specified are in Figure 11.

FIGURE 11–CRYSTAL PARAMETERS



AT — Cut Parallel Resonance Crystal
C_o = 7 pF Max.
FREQ = 4.0 MHz @ C_L = 24pF
R_S = 50 ohms Max.
Frequency Tolerance – ± 5% to ± 0.02%
The best E output "Worst Case Design" tolerance is ± 0.05% (500 ppM) using A ± 0.02% crystal.

Non-Maskable Interrupt (NMI) — A low-going edge on this input requests that a non-mask-interrupt sequence be generated within the processor. As with the Interrupt Request signal, the processor will complete the current instruction that is being executed before it recognizes the NMI signal. The interrupt mask bit in the Condition Code Register has no effect on NMI.

The index Register, Program Counter, Accumulators, and Condition Code Register are stored away on the stack. At the end of the cycle, a 16-bit address will be loaded that points to a vectoring address which is located in memory locations FFFC and FFFD. An address loaded at these locations caused the MPU to branch to a non-maskable interrupt routine in memory.

NMI has a high impedance pullup resistor internal to the chip; however a 3 kΩ external resistor to V_{CC} should be used for wire-OR and optimum control of interrupts.

Inputs IRQ and NMI are hardware interrupt lines that are sampled when E is high and will start the interrupt routine on a low E following the completion of an instruction.

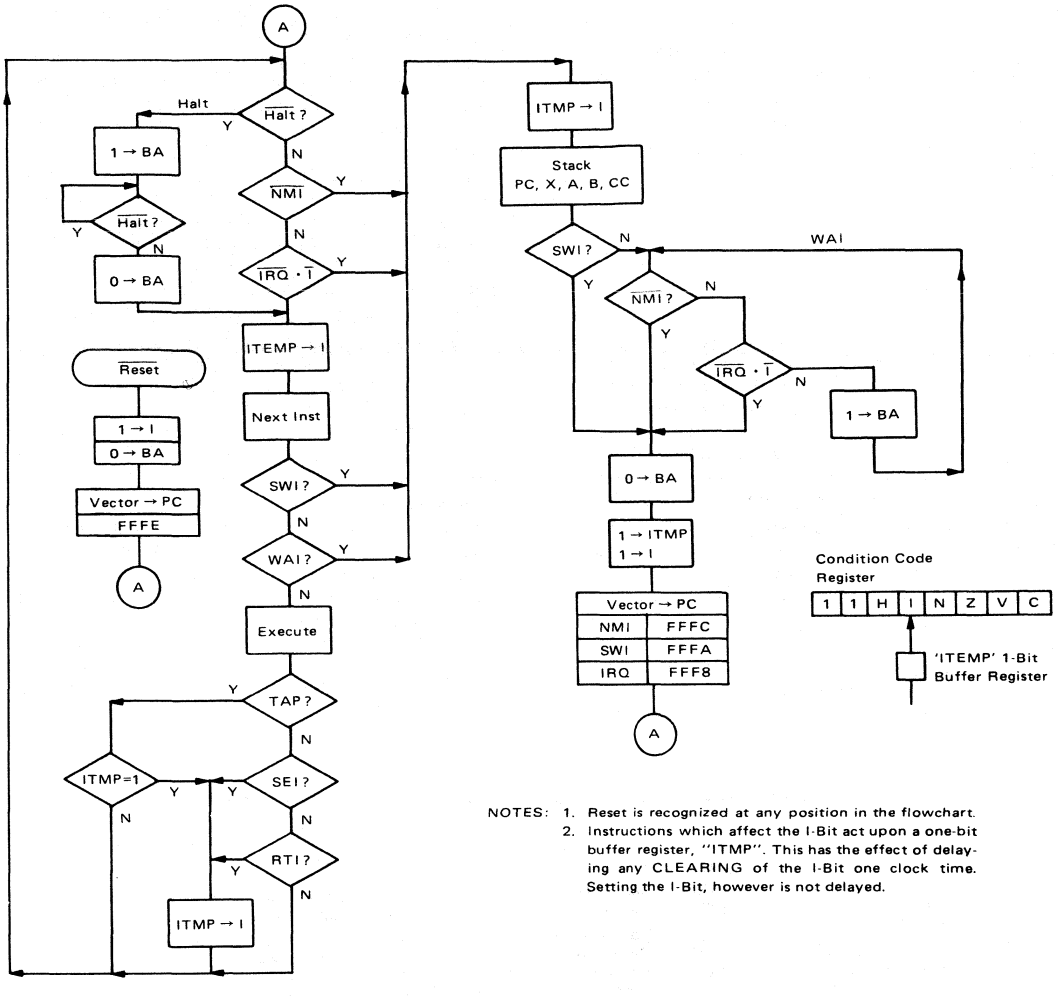
Figure 12 is a flow chart describing the major decision paths and interrupt vectors of the microprocessor. Table 1 gives the memory map for interrupt vectors.

Memory Ready (MR) — MR is a TTL compatible input control signal which allows stretching of E. When MR is high, E will be in normal operation. When MR is low, E may be stretched integral multiples of half periods, thus allowing interface to slow memories. Memory Ready timing is shown in Figure 13.

TABLE 1 – MEMORY MAP FOR INTERRUPT VECTORS

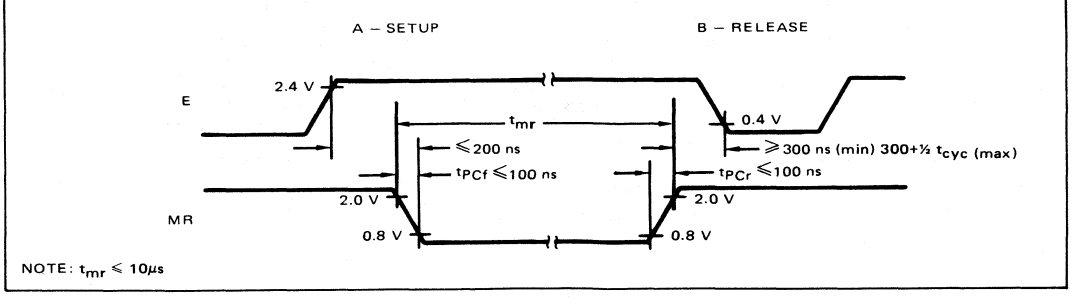
Vector MS	Vector LS	Description
FFFE	FFFF	Restart
FFFC	FFFD	Non-maskable Interrupt
FFFA	FFFB	Software Interrupt
FFF8	FFF9	Interrupt Request

FIGURE 12 – MPU FLOW CHART



NOTES: 1. Reset is recognized at any position in the flowchart.
2. Instructions which affect the I-Bit act upon a one-bit buffer register, "ITMP". This has the effect of delaying any CLEARING of the I-Bit one clock time. Setting the I-Bit, however is not delayed.

FIGURE 13 – MEMORY READY CONTROL FUNCTION



NOTE: $t_{mr} \leqslant 10\mu s$

MPU INSTRUCTION SET

The MC6808 has a set of 72 different instructions. Included are binary and decimal arithmetic, logical, shift, rotate, load, store, conditional or unconditional branch, interrupt and stack manipulation instructions (Tables 2 thru 6). This instruction set is the same as that for the MC6800.

MPU ADDRESSING MODES

The MC6808 eight-bit microprocessing unit has seven address modes that can be used by a programmer, with the addressing mode a function of both the type of instruction and the coding within the instruction. A summary of the addressing modes for a particular instruction can be found in Table 7 along with the associated instruction execution time that is given in machine cycles. With a clock frequency of 1 MHz, these times would be microseconds.

Accumulator (ACCX) Addressing — In accumulator only addressing, either accumulator A or accumulator B is specified. These are one-byte instructions.

Immediate Addressing — In immediate addressing, the operand is contained in the second byte of the instruction except LDS and LDX which have the operand in the second and third bytes of the instruction. The MPU addresses this location when it fetches the immediate instruction for execution. These are two or three-byte instructions.

Direct Addressing — In direct addressing, the address of the operand is contained in the second byte of the instruction. Direct addressing allows the user to directly address the lowest 256 bytes in the machine i.e., locations zero through 255. Enhanced execution times are achieved by storing data in these locations. In most configurations, it should be a random access memory. These are two-byte instructions.

Extended Addressing — In extended addressing, the address contained in the second byte of the instruction is used as the higher eight-bits of the address of the operand. The third byte of the instruction is used as the lower eight-bits of the address for the operand. This is an absolute address in memory. These are three-byte instructions.

Indexed Addressing — In indexed addressing, the address contained in the second byte of the instruction is added to the index register's lowest eight bits in the MPU. The carry is then added to the higher order eight bits of the index register. This result is then used to address memory. The modified address is held in a temporary address register so there is no change to the index register. These are two-byte instructions.

Implied Addressing — In the implied addressing mode the instruction gives the address (i.e., stack pointer, index register, etc.). These are one-byte instructions.

Relative Addressing — In relative addressing, the address contained in the second byte of the instruction is added to the program counter's lowest eight bits plus two. The carry or borrow is then added to the high eight bits. This allows the user to address data within a range of -125 to +129 bytes of the present instruction. These are two-byte instructions.

TABLE 2 — MICROPROCESSOR INSTRUCTION SET — ALPHABETIC SEQUENCE

ABA	Add Accumulators
ADC	Add with Carry
ADD	Add
AND	Logical And
ASL	Arithmetic Shift Left
ASR	Arithmetic Shift Right
BCC	Branch if Carry Clear
BCS	Branch if Carry Set
BEQ	Branch if Equal to Zero
BGE	Branch if Greater or Equal Zero
BGT	Branch if Greater than Zero
BHI	Branch if Higher
BIT	Bit Test
BLE	Branch if Less or Equal
BLS	Branch if Lower or Same
BLT	Branch if Less than Zero
BMI	Branch if Minus
BNE	Branch if Not Equal to Zero
BPL	Branch if Plus
BRA	Branch Always
BSR	Branch to Subroutine
BVC	Branch if Overflow Clear
BVS	Branch if Overflow Set
CBA	Compare Accumulators
CLC	Clear Carry
CLI	Clear Interrupt Mask
CLR	Clear
CLV	Clear Overflow
CMP	Compare
COM	Complement
CPX	Compare Index Register
DAA	Decimal Adjust
DEC	Decrement
DES	Decrement Stack Pointer
DEX	Decrement Index Register
EOR	Exclusive OR
INC	Increment
INS	Increment Stack Pointer
INX	Increment Index Register
JMP	Jump
JSR	Jump to Subroutine
LDA	Load Accumulator
LDS	Load Stack Pointer
LDX	Load Index Register
LSR	Logical Shift Right
NEG	Negate
NOP	No Operation
ORA	Inclusive OR Accumulator
PSH	Push Data
PUL	Pull Data
ROL	Rotate Left
ROR	Rotate Right
RTI	Return from Interrupt
RTS	Return from Subroutine
SBA	Subtract Accumulators
SBC	Subtract with Carry
SEC	Set Carry
SEI	Set Interrupt Mask
SEV	Set Overflow
STA	Store Accumulator
STS	Store Stack Register
STX	Store Index Register
SUB	Subtract
SWI	Software Interrupt
TAB	Transfer Accumulators
TAP	Transfer Accumulators to Condition Code Reg.
TBA	Transfer Accumulators
TPA	Transfer Condition Code Reg. to Accumulator
TST	Test
TSX	Transfer Stack Pointer to Index Register
TXS	Transfer Index Register to Stack Pointer
WAI	Wait for Interrupt

TABLE 3 – ACCUMULATOR AND MEMORY INSTRUCTIONS

OPERATIONS	MNEMONIC	IMMED OP	IMMED ~	IMMED =	DIRECT OP	DIRECT ~	DIRECT =	INDEX OP	INDEX ~	INDEX =	EXTND OP	EXTND ~	EXTND =	IMPLIED OP	IMPLIED ~	IMPLIED =	BOOLEAN/ARITHMETIC OPERATION (All register labels refer to contents)	5 H	4 I	3 N	2 Z	1 V	0 C
Add	ADDA	3B	2	2	9B	3	2	AB	5	2	BB	4	3				A + M → A	↕	●	↕	↕	↕	↕
	ADDB	CB	2	2	DB	3	2	EB	5	2	FB	4	3				B + M → B	↕	●	↕	↕	↕	↕
Add Acmltrs	ABA													1B	2	1	A + B → A	↕	●	↕	↕	↕	↕
Add with Carry	ADCA	89	2	2	99	3	2	A9	5	2	B9	4	3				A + M + C → A	↕	●	↕	↕	↕	↕
	ADCB	C9	2	2	D9	3	2	E9	5	2	F9	4	3				B + M + C → B	↕	●	↕	↕	↕	↕
And	ANDA	84	2	2	94	3	2	A4	5	2	B4	4	3				A · M → A	●	●	↕	↕	R	●
	ANDB	C4	2	2	D4	3	2	E4	5	2	F4	4	3				B · M → B	●	●	↕	↕	R	●
Bit Test	BITA	85	2	2	95	3	2	A5	5	2	B5	4	3				A · M	●	●	↕	↕	R	●
	BITB	C5	2	2	D5	3	2	E5	5	2	F5	4	3				B · M	●	●	↕	↕	R	●
Clear	CLR							6F	7	2	7F	6	3				00 → M	●	●	R	S	R	R
	CLRA													4F	2	1	00 → A	●	●	R	S	R	R
	CLRB													5F	2	1	00 → B	●	●	R	S	R	R
Compare	CMPA	81	2	2	91	3	2	A1	5	2	B1	4	3				A − M	●	●	↕	↕	↕	↕
	CMPB	C1	2	2	D1	3	2	E1	5	2	F1	4	3				B − M	●	●	↕	↕	↕	↕
Compare Acmltrs	CBA													11	2	1	A − B	●	●	↕	↕	↕	↕
Complement, 1's	COM							63	7	2	73	6	3				M̄ → M	●	●	↕	↕	R	S
	COMA													43	2	1	Ā → A	●	●	↕	↕	R	S
	COMB													53	2	1	B̄ → B	●	●	↕	↕	R	S
Complement, 2's (Negate)	NEG							60	7	2	70	6	3				00 − M → M	●	●	↕	↕	①	②
	NEGA													40	2	1	00 − A → A	●	●	↕	↕	①	②
	NEGB													50	2	1	00 − B → B	●	●	↕	↕	①	②
Decimal Adjust, A	DAA													19	2	1	Converts Binary Add. of BCD Characters into BCD Format	●	●	↕	↕	↕	③
Decrement	DEC							6A	7	2	7A	6	3				M − 1 → M	●	●	↕	↕	4	●
	DECA													4A	2	1	A − 1 → A	●	●	↕	↕	4	●
	DECB													5A	2	1	B − 1 → B	●	●	↕	↕	4	●
Exclusive OR	EORA	88	2	2	98	3	2	A8	5	2	B8	4	3				A⊕M → A	●	●	↕	↕	R	●
	EORB	C8	2	2	D8	3	2	E8	5	2	F8	4	3				B⊕M → B	●	●	↕	↕	R	●
Increment	INC							6C	7	2	7C	6	3				M + 1 → M	●	●	↕	↕	⑤	●
	INCA													4C	2	1	A + 1 → A	●	●	↕	↕	⑤	●
	INCB													5C	2	1	B + 1 → B	●	●	↕	↕	⑤	●
Load Acmltr	LDAA	86	2	2	96	3	2	A6	5	2	B6	4	3				M → A	●	●	↕	↕	R	●
	LDAB	C6	2	2	D6	3	2	E6	5	2	F6	4	3				M → B	●	●	↕	↕	R	●
Or, Inclusive	ORAA	8A	2	2	9A	3	2	AA	5	2	BA	4	3				A + M → A	●	●	↕	↕	R	●
	ORAB	CA	2	2	DA	3	2	EA	5	2	FA	4	3				B + M → B	●	●	↕	↕	R	●
Push Data	PSHA													36	4	1	A → M_{SP}, SP − 1 → SP	●	●	●	●	●	●
	PSHB													37	4	1	B → M_{SP}, SP − 1 → SP	●	●	●	●	●	●
Pull Data	PULA													32	4	1	SP + 1 → SP, M_{SP} → A	●	●	●	●	●	●
	PULB													33	4	1	SP + 1 → SP, M_{SP} → B	●	●	●	●	●	●
Rotate Left	ROL							69	7	2	79	6	3				M	●	●	↕	↕	⑥	↕
	ROLA													49	2	1	A	●	●	↕	↕	⑥	↕
	ROLB													59	2	1	B (C b7 b0)	●	●	↕	↕	⑥	↕
Rotate Right	ROR							66	7	2	76	6	3				M	●	●	↕	↕	⑥	↕
	RORA													46	2	1	A	●	●	↕	↕	⑥	↕
	RORB													56	2	1	B (C b7 b0)	●	●	↕	↕	⑥	↕
Shift Left, Arithmetic	ASL							68	7	2	78	6	3				M	●	●	↕	↕	⑥	↕
	ASLA													48	2	1	A (0 C b7 b0)	●	●	↕	↕	⑥	↕
	ASLB													58	2	1	B (0 C b7 b0)	●	●	↕	↕	⑥	↕
Shift Right, Arithmetic	ASR							67	7	2	77	6	3				M	●	●	↕	↕	⑥	↕
	ASRA													47	2	1	A (b7 b0 C)	●	●	↕	↕	⑥	↕
	ASRB													57	2	1	B (b7 b0 C)	●	●	↕	↕	⑥	↕
Shift Right, Logic	LSR							64	7	2	74	6	3				M	●	●	R	↕	⑥	↕
	LSRA													44	2	1	A (0 b7 b0 C)	●	●	R	↕	⑥	↕
	LSRB													54	2	1	B (0 b7 b0 C)	●	●	R	↕	⑥	↕
Store Acmltr.	STAA				97	4	2	A7	6	2	B7	5	3				A → M	●	●	↕	↕	R	●
	STAB				D7	4	2	E7	6	2	F7	5	3				B → M	●	●	↕	↕	R	●
Subtract	SUBA	80	2	2	90	3	2	A0	5	2	B0	4	3				A − M → A	●	●	↕	↕	↕	↕
	SUBB	C0	2	2	D0	3	2	E0	5	2	F0	4	3				B − M → B	●	●	↕	↕	↕	↕
Subtract Acmltrs.	SBA													10	2	1	A − B → A	●	●	↕	↕	↕	↕
Subtr. with Carry	SBCA	82	2	2	92	3	2	A2	5	2	B2	4	3				A − M − C → A	●	●	↕	↕	↕	↕
	SBCB	C2	2	2	D2	3	2	E2	5	2	F2	4	3				B − M − C → B	●	●	↕	↕	↕	↕
Transfer Acmltrs	TAB													16	2	1	A → B	●	●	↕	↕	R	●
	TBA													17	2	1	B → A	●	●	↕	↕	R	●
Test, Zero or Minus	TST							6D	7	2	7D	6	3				M − 00	●	●	↕	↕	R	R
	TSTA													4D	2	1	A − 00	●	●	↕	↕	R	R
	TSTB													5D	2	1	B − 00	●	●	↕	↕	R	R
																		H	I	N	Z	V	C

ADDRESSING MODES

LEGEND:

OP Operation Code (Hexadecimal);
~ Number of MPU Cycles;
= Number of Program Bytes;
+ Arithmetic Plus;
− Arithmetic Minus;
· Boolean AND;
M_{SP} Contents of memory location pointed to be Stack Pointer;
+ Boolean Inclusive OR;
⊕ Boolean Exclusive OR;
M̄ Complement of M;
→ Transfer Into;
0 Bit = Zero;
00 Byte = Zero;

Note – Accumulator addressing mode instructions are included in the column for IMPLIED addressing.

CONDITION CODE SYMBOLS:

H Half-carry from bit 3;
I Interrupt mask
N Negative (sign bit)
Z Zero (byte)
V Overflow, 2's complement
C Carry from bit 7
R Reset Always
S Set Always
↕ Test and set if true, cleared otherwise
● Not Affected

TABLE 4 – INDEX REGISTER AND STACK MANIPULATION INSTRUCTIONS

		IMMED			DIRECT			INDEX			EXTND			IMPLIED				COND. CODE REG.					
																		5	4	3	2	1	0
POINTER OPERATIONS	MNEMONIC	OP	~	=	OP	~	=	OP	~	=	OP	~	=	OP	~	=	BOOLEAN/ARITHMETIC OPERATION	H	I	N	Z	V	C
Compare Index Reg	CPX	8C	3	3	9C	4	2	AC	6	2	BC	5	3				$X_H - M, X_L - (M+1)$	•	•	⑦	↕	⑧	•
Decrement Index Reg	DEX													09	4	1	$X - 1 \rightarrow X$	•	•	•	↕	•	•
Decrement Stack Pntr	DES													34	4	1	$SP - 1 \rightarrow SP$	•	•	•	•	•	•
Increment Index Reg	INX													08	4	1	$X + 1 \rightarrow X$	•	•	•	↕	•	•
Increment Stack Pntr	INS													31	4	1	$SP + 1 \rightarrow SP$	•	•	•	•	•	•
Load Index Reg	LDX	CE	3	3	DE	4	2	EE	6	2	FE	5	3				$M \rightarrow X_H, (M+1) \rightarrow X_L$	•	•	⑨	↕	R	•
Load Stack Pntr	LDS	8E	3	3	9E	4	2	AE	6	2	BE	5	3				$M \rightarrow SP_H, (M+1) \rightarrow SP_L$	•	•	⑨	↕	R	•
Store Index Reg	STX				DF	5	2	EF	7	2	FF	6	3				$X_H \rightarrow M, X_L \rightarrow (M+1)$	•	•	⑨	↕	R	•
Store Stack Pntr	STS				9F	5	2	AF	7	2	BF	6	3				$SP_H \rightarrow M, SP_L \rightarrow (M+1)$	•	•	⑨	↕	R	•
Indx Reg → Stack Pntr	TXS													35	4	1	$X - 1 \rightarrow SP$	•	•	•	•	•	•
Stack Pntr → Indx Reg	TSX													30	4	1	$SP + 1 \rightarrow X$	•	•	•	•	•	•

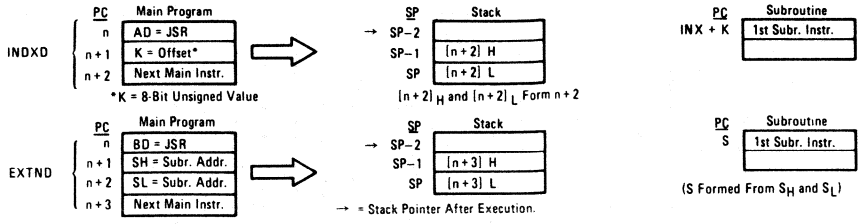
TABLE 5 – JUMP AND BRANCH INSTRUCTIONS

		RELATIVE			INDEX			EXTND			IMPLIED				COND. CODE REG.						
															5	4	3	2	1	0	
OPERATIONS	MNEMONIC	OP	~	=	OP	~	=	OP	~	=	OP	~	=	BRANCH TEST	H	I	N	Z	V	C	
Branch Always	BRA	20	4	2										None	•	•	•	•	•	•	
Branch If Carry Clear	BCC	24	4	2										C = 0	•	•	•	•	•	•	
Branch If Carry Set	BCS	25	4	2										C = 1	•	•	•	•	•	•	
Branch If = Zero	BEQ	27	4	2										Z = 1	•	•	•	•	•	•	
Branch If ≥ Zero	BGE	2C	4	2										$N \oplus V = 0$	•	•	•	•	•	•	
Branch If > Zero	BGT	2E	4	2										$Z + (N \oplus V) = 0$	•	•	•	•	•	•	
Branch If Higher	BHI	22	4	2										C + Z = 0	•	•	•	•	•	•	
Branch If ≤ Zero	BLE	2F	4	2										$Z + (N \oplus V) = 1$	•	•	•	•	•	•	
Branch If Lower Or Same	BLS	23	4	2										C + Z = 1	•	•	•	•	•	•	
Branch If < Zero	BLT	2D	4	2										$N \oplus V = 1$	•	•	•	•	•	•	
Branch If Minus	BMI	2B	4	2										N = 1	•	•	•	•	•	•	
Branch If Not Equal Zero	BNE	26	4	2										Z = 0	•	•	•	•	•	•	
Branch If Overflow Clear	BVC	28	4	2										V = 0	•	•	•	•	•	•	
Branch If Overflow Set	BVS	29	4	2										V = 1	•	•	•	•	•	•	
Branch If Plus	BPL	2A	4	2										N = 0	•	•	•	•	•	•	
Branch To Subroutine	BSR	8D	8	2											•	•	•	•	•	•	
Jump	JMP				6E	4	2	7E	3	3				See Special Operations	•	•	•	•	•	•	
Jump To Subroutine	JSR				AD	8	2	BD	9	3					•	•	•	•	•	•	
No Operation	NOP										01	2	1	Advances Prog. Cntr. Only	•	•	•	•	•	•	
Return From Interrupt	RTI										3B	10	1		———			⑩	———		
Return From Subroutine	RTS										39	5	1		•	•	•	•	•	•	
Software Interrupt	SWI										3F	12	1	See Special Operations	•	•	•	•	•	•	
Wait for Interrupt*	WAI										3E	9	1		•	⑪	•	•	•	•	

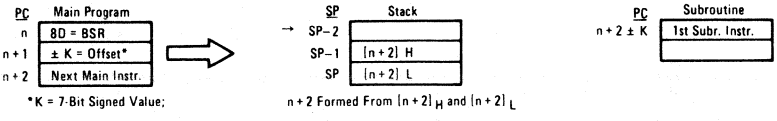
*WAI puts Address Bus, R/W, and Data Bus in the three-state mode while VMA is held low.

SPECIAL OPERATIONS

JSR, JUMP TO SUBROUTINE:

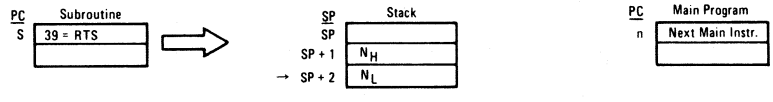


BSR, BRANCH TO SUBROUTINE:



JMP, JUMP:

RTS, RETURN FROM SUBROUTINE:



RTI, RETURN FROM INTERRUPT:

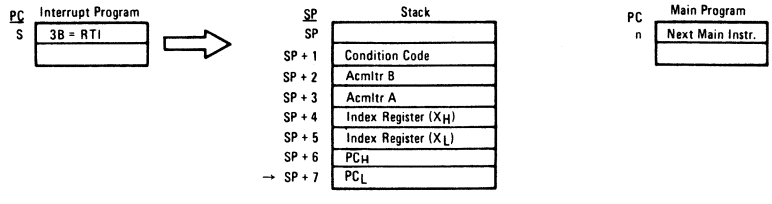


TABLE 6 – CONDITION CODE REGISTER MANIPULATION INSTRUCTIONS

OPERATIONS	MNEMONIC	IMPLIED OP	~	=	BOOLEAN OPERATION	COND. CODE REG. 5 H	4 I	3 N	2 Z	1 V	0 C
Clear Carry	CLC	0C	2	1	0 → C	•	•	•	•	•	R
Clear Interrupt Mask	CLI	0E	2	1	0 → I	•	R	•	•	•	•
Clear Overflow	CLV	0A	2	1	0 → V	•	•	•	•	R	•
Set Carry	SEC	0D	2	1	1 → C	•	•	•	•	•	S
Set Interrupt Mask	SEI	0F	2	1	1 → I	•	S	•	•	•	•
Set Overflow	SEV	0B	2	1	1 → V	•	•	•	•	S	•
Acmltr A → CCR	TAP	06	2	1	A → CCR	———	———	———	(12)	———	———
CCR → Acmltr A	TPA	07	2	1	CCR → A	•	•	•	•	•	•

CONDITION CODE REGISTER NOTES: (Bit set if test is true and cleared otherwise)

1 (Bit V) Test: Result = 10000000?
2 (Bit C) Test: Result ≠ 00000000?
3 (Bit C) Test: Decimal value of most significant BCD Character greater than nine? (Not cleared if previously set.)
4 (Bit V) Test: Operand = 10000000 prior to execution?
5 (Bit V) Test: Operand = 01111111 prior to execution?
6 (Bit V) Test: Set equal to result of N⊕C after shift has occurred.
7 (Bit N) Test: Sign bit of most significant (MS) byte = 1?
8 (Bit V) Test: 2's complement overflow from subtraction of MS bytes?
9 (Bit N) Test: Result less than zero? (Bit 15 = 1)
10 (All) Load Condition Code Register from Stack. (See Special Operations)
11 (Bit I) Set when interrupt occurs. If previously set, a Non-Maskable Interrupt is required to exit the wait state.
12 (All) Set according to the contents of Accumulator A.

TABLE 7 – INSTRUCTION ADDRESSING MODES AND ASSOCIATED EXECUTION TIMES
(Times in Machine Cycles)

	(Dual Operand)	ACCX	Immediate	Direct	Extended	Indexed	Implied	Relative
ABA		•	•	•	•	•	2	•
ADC	x	•	2	3	4	5	•	•
ADD	x	•	2	3	4	5	•	•
AND	x	•	2	3	4	5	•	•
ASL		2	•	•	6	7	•	•
ASR		2	•	•	6	7	•	•
BCC		•	•	•	•	•	•	4
BCS		•	•	•	•	•	•	4
BEA		•	•	•	•	•	•	4
BGE		•	•	•	•	•	•	4
BGT		•	•	•	•	•	•	4
BHI		•	•	•	•	•	•	4
BIT	x	•	2	3	4	5	•	•
BLE		•	•	•	•	•	•	4
BLS		•	•	•	•	•	•	4
BLT		•	•	•	•	•	•	4
BMI		•	•	•	•	•	•	4
BNE		•	•	•	•	•	•	4
BPL		•	•	•	•	•	•	4
BRA		•	•	•	•	•	•	4
BSR		•	•	•	•	•	•	8
BVC		•	•	•	•	•	•	4
BVS		•	•	•	•	•	•	4
CBA		•	•	•	•	•	2	•
CLC		•	•	•	•	•	2	•
CLI		•	•	•	•	•	2	•
CLR		2	•	•	6	7	•	•
CLV		•	•	•	•	•	2	•
CMP	x	•	2	3	4	5	•	•
COM		2	•	•	6	7	•	•
CPX		•	3	4	5	6	•	•
DAA		•	•	•	•	•	2	•
DEC		2	•	•	6	7	•	•
DES		•	•	•	•	•	4	•
DEX		•	•	•	•	•	4	•
EOR	x	•	2	3	4	5	•	•

	(Dual Operand)	ACCX	Immediate	Direct	Extended	Indexed	Implied
INC		2	•	•	6	7	•
INS		•	•	•	•	•	4
INX		•	•	•	•	•	4
JMP		•	•	•	3	4	•
JSR		•	•	•	9	8	•
LDA	x	•	2	3	4	5	•
LDS		•	3	4	5	6	•
LDX		•	3	4	5	6	•
LSR		2	•	•	6	7	•
NEG		2	•	•	6	7	•
NOP		•	•	•	•	•	2
ORA	x	•	2	3	4	5	•
PSH		•	•	•	•	•	4
PUL		•	•	•	•	•	4
ROL		2	•	•	6	7	•
ROR		2	•	•	6	7	•
RTI		•	•	•	•	•	10
RTS		•	•	•	•	•	5
SBA		•	•	•	•	•	2
SBC	x	•	2	3	4	5	•
SEC		•	•	•	•	•	2
SEI		•	•	•	•	•	2
SEV		•	•	•	•	•	2
STA	x	•	•	4	5	6	•
STS		•	•	5	6	7	•
STX		•	•	5	6	7	•
SUB	x	•	2	3	4	5	•
SWI		•	•	•	•	•	12
TAB		•	•	•	•	•	2
TAP		•	•	•	•	•	2
TBA		•	•	•	•	•	2
TPA		•	•	•	•	•	2
TST		2	•	•	6	7	•
TSX		•	•	•	•	•	4
TSX		•	•	•	•	•	4
WAI		•	•	•	•	•	9

NOTE: Interrupt time is 12 cycles from the end of the instruction being executed, except following a WAI instruction. Then it is 4 cycles.

SUMMARY OF CYCLE BY CYCLE OPERATION

Table 8 provides a detailed description of the information present on the Address Bus, Data Bus, Valid Memory Address line (VMA), and the Read/Write line (R/W) during each cycle for each instruction.

This information is useful in comparing actual with expected results during debug of both software and hardware as the control program is executed. The information is categorized in groups according to Addressing Mode and Number of Cycles per instruction. (In general, instructions with the same Addressing Mode and Number of Cycles execute in the same manner; exceptions are indicated in the table.)

TABLE 8 – OPERATION SUMMARY

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
IMMEDIATE						
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	2	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Operand Data
CPX LDS LDX	3	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Operand Data (High Order Byte)
		3	1	Op Code Address + 2	1	Operand Data (Low Order Byte)
DIRECT						
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	3	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand
		3	1	Address of Operand	1	Operand Data
CPX LDS LDX	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand
		3	1	Address of Operand	1	Operand Data (High Order Byte)
		4	1	Operand Address + 1	1	Operand Data (Low Order Byte)
STA	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Destination Address
		3	0	Destination Address	1	Irrelevant Data (Note 1)
		4	1	Destination Address	0	Data from Accumulator
STS STX	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand
		3	0	Address of Operand	1	Irrelevant Data (Note 1)
		4	1	Address of Operand	0	Register Data (High Order Byte)
		5	1	Address of Operand + 1	0	Register Data (Low Order Byte)
INDEXED						
JMP	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	1	Index Register Plus Offset	1	Operand Data
CPX LDS LDX	6	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	1	Index Register Plus Offset	1	Operand Data (High Order Byte)
		6	1	Index Register Plus Offset + 1	1	Operand Data (Low Order Byte)

TABLE 8 – OPERATION SUMMARY (Continued)

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
INDEXED (Continued)						
STA	6	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	0	Index Register Plus Offset	1	Irrelevant Data (Note 1)
		6	1	Index Register Plus Offset	0	Operand Data
ASL LSR ASR NEG CLR ROL COM ROR DEC TST INC	7	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	1	Index Register Plus Offset	1	Current Operand Data
		6	0	Index Register Plus Offset	1	Irrelevant Data (Note 1)
		7	1/0 (Note 3)	Index Register Plus Offset	0	New Operand Data (Note 3)
STS STX	7	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	0	Index Register Plus Offset	1	Irrelevant Data (Note 1)
		6	1	Index Register Plus Offset	0	Operand Data (High Order Byte)
		7	1	Index Register Plus Offset + 1	0	Operand Data (Low Order Byte)
JSR	8	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer	0	Return Address (Low Order Byte)
		5	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		6	0	Stack Pointer – 2	1	Irrelevant Data (Note 1)
		7	0	Index Register	1	Irrelevant Data (Note 1)
		8	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
EXTENDED						
JMP	3	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Jump Address (High Order Byte)
		3	1	Op Code Address + 2	1	Jump Address (Low Order Byte)
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	1	Address of Operand	1	Operand Data
CPX LDS LDX	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	1	Address of Operand	1	Operand Data (High Order Byte)
		5	1	Address of Operand + 1	1	Operand Data (Low Order Byte)
STA A STA B	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Destination Address (High Order Byte)
		3	1	Op Code Address + 2	1	Destination Address (Low Order Byte)
		4	0	Operand Destination Address	1	Irrelevant Data (Note 1)
		5	1	Operand Destination Address	0	Data from Accumulator
ASL LSR ASR NEG CLR ROL COM ROR DEC TST INC	6	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	1	Address of Operand	1	Current Operand Data
		5	0	Address of Operand	1	Irrelevant Data (Note 1)
		6	1/0 (Note 3)	Address of Operand	0	New Operand Data (Note 3)

TABLE 8 – OPERATION SUMMARY (Continued)

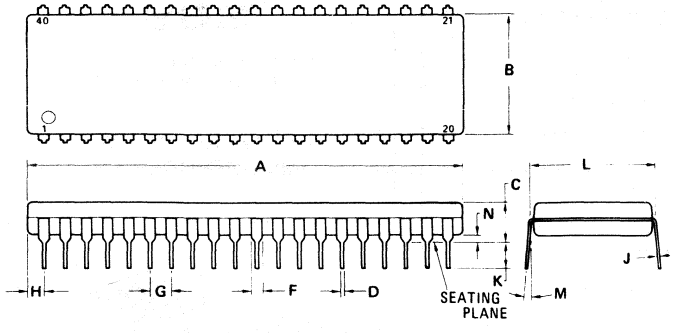
Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
EXTENDED (Continued)						
STS STX	6	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Operand (Low Order Byte)
		4	0	Address of Operand	1	Irrelevant Data (Note 1)
		5	1	Address of Operand	0	Operand Data (High Order Byte)
		6	1	Address of Operand + 1	0	Operand Data (Low Order Byte)
JSR	9	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Address of Subroutine (High Order Byte)
		3	1	Op Code Address + 2	1	Address of Subroutine (Low Order Byte)
		4	1	Subroutine Starting Address	1	Op Code of Next Instruction
		5	1	Stack Pointer	0	Return Address (Low Order Byte)
		6	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		7	0	Stack Pointer – 2	1	Irrelevant Data (Note 1)
		8	0	Op Code Address + 2	1	Irrelevant Data (Note 1)
		9	1	Op Code Address + 2	1	Address of Subroutine (Low Order Byte)
INHERENT						
ABA DAA SEC ASL DEC SEI ASR INC SEV CBA LSR TAB CLC NEG TAP CLI NOP TBA CLR ROL TPA CLV ROR TST COM SBA	2	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
DES DEX INS INX	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	0	Previous Register Contents	1	Irrelevant Data (Note 1)
		4	0	New Register Contents	1	Irrelevant Data (Note 1)
PSH	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	1	Stack Pointer	0	Accumulator Data
		4	0	Stack Pointer – 1	1	Accumulator Data
PUL	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	0	Stack Pointer	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer + 1	1	Operand Data from Stack
TSX	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	0	Stack Pointer	1	Irrelevant Data (Note 1)
		4	0	New Index Register	1	Irrelevant Data (Note 1)
TXS	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	0	Index Register	1	Irrelevant Data
		4	0	New Stack Pointer	1	Irrelevant Data
RTS	5	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Irrelevant Data (Note 2)
		3	0	Stack Pointer	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer + 1	1	Address of Next Instruction (High Order Byte)
		5	1	Stack Pointer + 2	1	Address of Next Instruction (Low Order Byte)

TABLE 8 – OPERATION SUMMARY (Continued)

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
INHERENT (Continued)						
WAI	9	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	1	Stack Pointer	0	Return Address (Low Order Byte)
		4	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		5	1	Stack Pointer – 2	0	Index Register (Low Order Byte)
		6	1	Stack Pointer – 3	0	Index Register (High Order Byte)
		7	1	Stack Pointer – 4	0	Contents of Accumulator A
		8	1	Stack Pointer – 5	0	Contents of Accumulator B
		9	1	Stack Pointer – 6 (Note 4)	1	Contents of Cond. Code Register
RTI	10	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Irrelevant Data (Note 2)
		3	0	Stack Pointer	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer + 1	1	Contents of Cond. Code Register from Stack
		5	1	Stack Pointer + 2	1	Contents of Accumulator B from Stack
		6	1	Stack Pointer + 3	1	Contents of Accumulator A from Stack
		7	1	Stack Pointer + 4	1	Index Register from Stack (High Order Byte)
		8	1	Stack Pointer + 5	1	Index Register from Stack (Low Order Byte)
		9	1	Stack Pointer + 6	1	Next Instruction Address from Stack (High Order Byte)
		10	1	Stack Pointer + 7	1	Next Instruction Address from Stack (Low Order Byte)
SWI	12	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Irrelevant Data (Note 1)
		3	1	Stack Pointer	0	Return Address (Low Order Byte)
		4	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		5	1	Stack Pointer – 2	0	Index Register (Low Order Byte)
		6	1	Stack Pointer – 3	0	Index Register (High Order Byte)
		7	1	Stack Pointer – 4	0	Contents of Accumulator A
		8	1	Stack Pointer – 5	0	Contents of Accumulator B
		9	1	Stack Pointer – 6	0	Contents of Cond. Code Register
		10	0	Stack Pointer – 7	1	Irrelevant Data (Note 1)
		11	1	Vector Address FFFA (Hex)	1	Address of Subroutine (High Order Byte)
		12	1	Vector Address FFFB (Hex)	1	Address of Subroutine (Low Order Byte)
RELATIVE						
BCC BHI BNE BCS BLE BPL BEQ BLS BRA BGE BLT BVC BGT BMI BVS	4	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Branch Offset
		3	0	Op Code Address + 2	1	Irrelevant Data (Note 1)
		4	0	Branch Address	1	Irrelevant Data (Note 1)
BSR	8	1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Branch Offset
		3	0	Return Address of Main Program	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer	0	Return Address (Low Order Byte)
		5	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		6	0	Stack Pointer – 2	1	Irrelevant Data (Note 1)
		7	0	Return Address of Main Program	1	Irrelevant Data (Note 1)
		8	0	Subroutine Address	1	Irrelevant Data (Note 1, Note 5)

Note 1. If device which is addressed during this cycle uses VMA, then the Data Bus will go to the high impedance three-state condition. Depending on bus capacitance, data from the previous cycle may be retained on the Data Bus.
Note 2. Data is ignored by the MPU.
Note 3. For TST, VMA = 0 and Operand data does not change.
Note 4. While the MPU is waiting for the interrupt, Bus Available will go high, VMA is low.
Note 5. MS Byte = MS Byte of BSR instruction address, LS Byte = LS Byte of subroutine address.

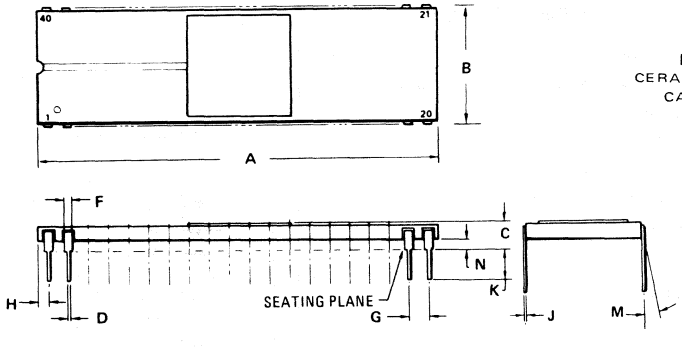
P SUFFIX
PLASTIC PACKAGE
CASE 711-03



NOTES:
1. POSITIONAL TOLERANCE OF LEADS (D), SHALL BE WITHIN 0.25 mm (0.010) AT MAXIMUM MATERIAL CONDITION, IN RELATION TO SEATING PLANE AND EACH OTHER.
2. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSION B DOES NOT INCLUDE MOLD FLASH.
4. 711-02 OBSOLETE, NEW STANDARD 711-03.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	51.69	52.45	2.035	2.065
B	13.72	14.22	0.540	0.560
C	3.94	5.08	0.155	0.200
D	0.36	0.56	0.014	0.022
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.38	0.008	0.015
K	2.92	3.43	0.115	0.135
L	15.24 BSC		0.600 BSC	
M	0^0	15^0	0^0	15^0
N	0.51	1.02	0.020	0.040

L SUFFIX
CERAMIC PACKAGE
CASE 715-02



	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	50.29	51.31	1.980	2.020
B	14.86	15.62	0.585	0.615
C	2.54	4.19	0.100	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.33	0.008	0.013
K	2.54	4.19	0.100	0.165
M	0^0	10^0	0^0	10^0
N	0.51	1.52	0.020	0.060

NOTE:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA (AT SEATING PLANE), AT MAX. MAT'L CONDITION.

MOTOROLA

MC6809(E)
(1.0 MHz)

MC68A09(E)
(1.5 MHz)

MC68B09(E)
(2.0 MHz)

Product Preview

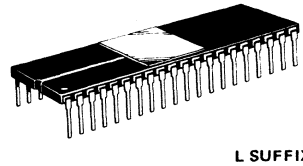
HIGH-PERFORMANCE MICROPROCESSOR

- **MC6800 COMPATIBLE**
  - Hardware — Interfaces With All M6800 Peripherals
  - Software — Upward Compatible Instruction Set and Addressing Modes
- **HARDWARE FEATURES**
  - On-Chip Oscillator (MC6809) 4 X fo Clock
  - Optional ÷1 External Clock Inputs (MC6809E)
  - MRDY Input Extends Data Access Times for Use With Slow Memory
  - $\overline{BREQ}$/TSC Allows Quick Access to Bus for DMA and Memory Refresh
  - Last Instruction Cycle Output for Identification of Opcode Fetch (MC6809E)
  - Fast Interrupt Request Input Stacks Only Program Counter and Condition Code
  - Interrupt Acknowledge Output Allows Vectoring by Device
  - Busy Output Eases Multiprocessor Design (MC6809E)
- **ARCHITECTURAL FEATURES**
  - Two 8-Bit Accumulators Can Be Concatenated to Form One 16-Bit Accumulator
  - Two 16-Bit Index Registers
  - Two 16-Bit Indexable Stack Pointers
  - Direct Page Register Allows Direct Addressing Throughout Memory Space
- **INSTRUCTION SET**
  - Extended Range Branches
  - 16-Bit Arithmetic
  - Push/Pull Any Register or Set of Registers To/From Either Stack
  - 8 X 8 Unsigned Multiply
  - Transfer/Exchange Any Two Registers of Equal Size
  - Enhanced Pointer Register Manipulation
- **ADDRESSING MODES**
  - All MC6800 Modes, Plus PC Relative, Extended Indirect, Indexed Indirect, and PC Relative Indirect
  - Direct Addressing Available for All Memory Access Instructions
  - Index Mode Options Include Accumulator or Up to 16-Bit Constant Offset, and Auto-Increment/Decrement (by 1 or 2) With Any of the Four Pointer Registers

MOS

(N-CHANNEL, SILICON-GATE, DEPLETION LOAD)

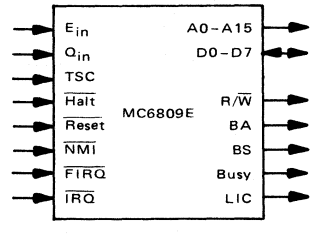
HIGH-PERFORMANCE MICROPROCESSOR

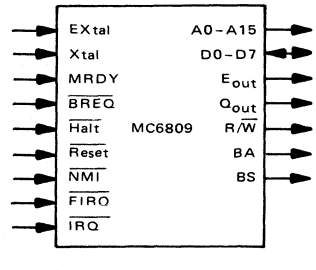


L SUFFIX
CERAMIC PACKAGE
CASE 715

NOT SHOWN:
P SUFFIX
PLASTIC PACKAGE
CASE 711

FIGURE 1 – BLOCK DIAGRAMS





This is advance information and specifications are subject to change without notice.

NP-98 R1

M6809 FEATURES

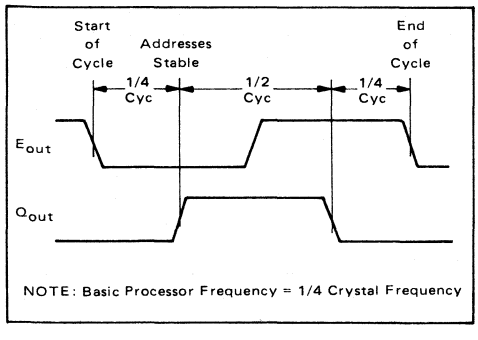
The MC6809 is an advanced processor within the M6800 family offering greater through-put, improved byte-efficiency, and increased adaptability to various software disciplines. These include position independence, reentrancy, recursion, block structuring, and high level language generation. It is compatible with all M6800 peripherals, and is software compatible with the MC6800 at source code level.

This product preview stresses the improvements inherent in the MC6809. The basic operation of the MPU is similar to the MC6800.

Hardware Features — Internal Clock MC6809

For internal clock use, an external crystal is connected between Extal and Xtal. A synchronization signal is available at the E_{out} terminal to be used as a system clock. This signal operates at the basic processor frequency, and is normally connected to the Enable ($\phi 2$) inputs of M6800 family peripherals. A Quadrature output (Q_{out}) provides additional system timing by signifying that addresses and data are stable as shown in Figure 2.

FIGURE 2 — MC6809 TIMING SIGNALS



A Memory Ready (MRDY) input allows extension of data access times for use with slow memories. A logic zero at this input (when E_{out} goes high) causes E_{out} to remain high until MRDY returns high. Stretching is always an integral number of quarter bus cycles, and is limited to a maximum of ten microseconds. The negative transition of Q_{out} is unaffected by MRDY, but further positive transitions of Q_{out} are inhibited for the duration of the clock stretch.

A Bus Request ($\overline{BREQ}$) input allows fast access to the bus for DMA or Memory Refresh. This is a request to temporarily suspend MPU operation and take the MPU off the MOS bus. The BA line will immediately go high (as a result of the trailing edge of E), signifying a Bus Available condition. One-half cycle later, the user may place the DMA device on the MPU buses. This will eliminate bus contentions into DMA.

Hardware Features — External Clock MC6809E

The External Clock mode of the MC6809 is particularly useful when it is desired to synchronize the processor to an externally generated signal. The external clock generator provides an output only at the basic MPU frequency, since the internal frequency dividers of the MC6809 are not used in the external clock mode. E_{in} and Q_{in} signals are required with phasing as shown for E_{out} and Q_{out}, respectively.

A Three-State Control (TSC) input replaces the $\overline{BREQ}$ input of the MC6809, and serves to place the Address and R/$\overline{W}$ line in the high-impedance state for DMA or Memory Refresh. (The Data Bus is in the high-impedance state when E_{in} and Q_{in} are both low.) The E_{out} and Q_{out} terminals are replaced by two status outputs (LIC and Busy).

A last Instruction Cycle output (LIC) is activated during the last cycle of any instruction. The first low cyle after LIC is high signifies that this processor cycle will be an opcode fetch.

A Processor Busy signal (BUSY) facilitates multiprocessor applications. This signal is asserted during MPU Read-Modify-Write instructions, allowing the designer to insure that flags being modified by one processor are not accessed by another simultaneously. The signal is also asserted during execution of double byte instructions and when using the Indirect Addressing modes.

Hardware Features — Either Clock Option

A Fast Interrupt Request ($\overline{FIRQ}$) is added to the $\overline{IRQ}$ and $\overline{NMI}$ inputs available on the MC6800. When a logic zero is recognized at this input, the MC6809 places only the Program Counter and Condition Code Register on the stack prior to accessing the $\overline{FIRQ}$ vector location to obtain the starting location of the $\overline{FIRQ}$ service routine. Either an $\overline{IRQ}$ or Non-Maskable Interrupt stacks the contents of all registers (same as MC6800) prior to vectoring. The three interrupt inputs have separate vector locations, and are prioritized ($\overline{IRQ}$ lowest, $\overline{FIRQ}$ next, $\overline{NMI}$ highest).

An Interrupt Acknowledge function (IACK) indicates that a vector is being fetched as a result of a $\overline{Reset}$, a Software Interrupt (SWI, SWI2, or SWI3), or recognition of $\overline{NMI}$, $\overline{FIRQ}$, or $\overline{IRQ}$. This function allows the program counter to be loaded according to the interrupting device, thus providing full vectored interrupt handling. IACK is denoted by a logic zero at the Bus Available (BA) output in conjunction with a logic one at the Bus Status (BS) output.

A low level on the Halt input causes the MPU to halt at the end of the present instruction. It will remain halted indefinitely without loss of data, until the Halt line is driven high. When the MPU is halted, the BA and BS outputs are driven high to acknowledge the halt, and the Address Bus and Data Bus drivers are made high-impedance. While halted, the MPU cannot respond to

interrupt requests, nor can it be released from reset.

The MC6809 has the capability of entering an idle state under program control via SYNC and CWAI instructions. SYNC Acknowledge can be detected via BA and BS output states. These two status signals are defined in Table 1.

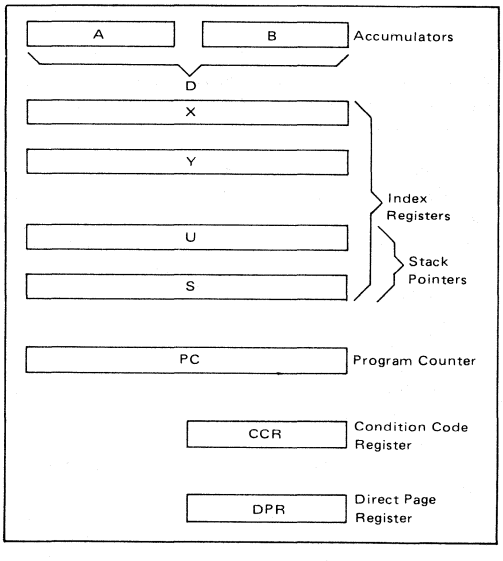
TABLE 1 – STATUS SIGNALS

BA	BS	Function
Ø	Ø	Normal Operation
Ø	1	Interrupt Acknowledge
1	Ø	Sync Acknowledge
1	1	Bus Grant or Halt Acknowledge

PROGRAMMING MODEL

The MC6809 adds three registers to the complement available in the MC6800. These are the Direct Page Register, the User Stack Pointer, and a second (Y) Index Register (shown in Figure 3). In addition to increasing

FIGURE 3 – PROGRAMMING MODEL



the number of registers, the MC6809 features much greater flexibility of register usage. For example, indexed addressing is available using a stack pointer (U or S) as the base register in addition to X or Y. Conversely, the Automatic Increment/Decrement option of indexed instructions allows X and Y to be used as stack pointers if desired.

The enhanced instruction set of the MC6809 also increases register uses. Exchanges, as well as transfers, between any two similar-width registers are allowed. Two's complement addition of the A, B, or D register (or an immediate value) to X, Y, U, or S registers can be implemented via the Load Effective Address instruction. Any register—or set of registers—may be pushed to (pulled from) either stack with a single instruction.

Accumulators

The A and B registers are general-purpose accumulators used for arithmetic calculators and data manipulation. In general, the two registers are identical, although some special purpose instructions apply to one register alone.

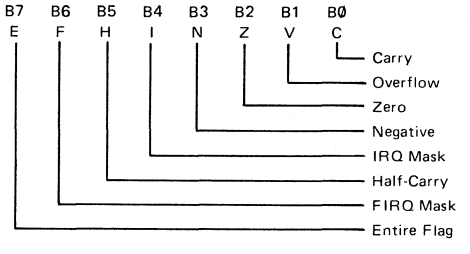
Certain instructions concatenate the A and B registers to form a single 16-bit accumultor. This is referred to as the D register, and is formed with the A register as the Most Significant Byte.

Direct Page Register

The Direct Page Register of the MC6809 serves to enhance the Direct Addressing Mode. The contents of this register appear at the higher order address outputs (A8–A15) during direct addressing instruction execution. This allows the Direct Page to be located at any place in memory under program control. All bits of this register are cleared during Processor Reset. This insures 6800 compatibility.

Condition Code Register

The Condition Code Register is the storage area for processor flags. Bits Ø through 5 are identical to the MC6800. CCR—Bits 6 and 7 are used for FIRQ handling.



Specifically, the E-bit controls the Return from Interrupt (RTI) instruction to insure that the proper number of registers are pulled from the Hardware Stack.

Index Registers

The Index Registers are used in indexed mode addressing. They provide a 16-bit address to be added to an optional offset (of up to 16 bits) to form the effective address of the instruction. The X and Y registers are essentially equivalent in usage and support the same instructions. Either index register is capable of performing the same function as the MC6800 index register, but can also support many additional modes of operation.

Stack Pointers

The Hardware Stack Pointer (S) is used by the processor to automatically store machine states during subroutines and interrupts in a manner similar to that of

the MC6800. A User Stack Pointer (U) is also provided to be controlled exclusively by the program—thus allowing arguments to be passed to and from subroutines with ease.

Both U and S have the same indexed-mode addressing capabilities as the X and Y registers. This allows the MC6809 to be used efficiently as a stack processor—greatly enhancing its ability to support higher level languages. To facilitate use of the stack pointers in indexing mode addressing, the registers always point to the last byte placed on the stack. (The MC6800 stack pointer points to the location in which the next byte will be stored.)

Program Counter

The PC is used by the processor to point to the next instruction to be executed by the processor. Program Counter Relative addressing is allowed on the MC6809, effectively allowing the PC register to be used as an index register. This allows Position Independent Code to be written with greater ease than with other 8-bit processors.

ADDRESSING MODES

The MC6809 retains the Addressing modes available with the MC6800, and adds long relative branches, 16 variations of indexed addressing, program counter relative and extended indirect addressing.

The enhancements to indexed addressing include auto (post) increment, auto (pre) decrement, indexing with 0-, 5-, 8-, or 16-bit two's complement offsets, and indexing with an accumulator as an offset. Also, any of these modes may have one additional level of indirection applied. Any of the four index registers (X, Y, U, or S) may be used as the base register for the indexed addressing modes.

Inherent, Accumulator, Immediate, Direct, Extended

These modes are effectively the same as those available with the MC6800. (The Direct Mode utilizes the DPR to form the most significant address byte as explained previously.)

Indexed

The Indexed Mode of addressing has several options which are selected by the byte following the opcode (postbyte). Bits 5 and 6 of this postbyte are always used to select the pointer register (X, Y, U, or S). Other options are:

Zero Offset – This option allows selection of Auto Increment/Decrement by one or two bits. It is a minimum two-byte instruction (opcode + postbyte).

Constant Offset, ±4 Bits – This option uses Bit 4 of the postbyte as a sign bit and Bits Ø through 3 as a constant offset. It is a minimum two-byte instruction.

Constant Offset, ±7 Bits – This option designates the byte after the postbyte as a two's complement offset. It is a minimum three-byte instruction (opcode + postbyte + offset).

Constant Offset, ±15 Bits – This option designates the two bytes following the postbyte as a two's complement offset. it is a minimum four-byte instruction (opcode + postbyte + two-byte offset).

Accumulator Offset – This option designates the A, B, or D register as a two's complement offset. It is a minimum two-byte instruction.

In all cases, the offset is temporarily added to the contents of the selected pointer register to form an effective address.

Indexed Indirect

Except for the ±4-Bit Constant Offset and the Auto Increment/Decrement by one, all indexed addressing modes may be used with an additional level of indirection. Thus, the address formed by adding the offset to the selected pointer register designates a location containing the effective address of the operand data. Bit 4 of the postbyte is used to select the Indexed Indirect mode. (Note that this bit is used as a sign bit in the ±4-bit constant offset mode.) The number of bytes for a given instruction is the same for either Indexed or Indexed Indirect addressing modes.

Relative

Relative addressing involves adding a signed constant to the contents of the program counter. When used in conjunction with a Branch Instruction, this sum becomes the new Program Counter content if the branch is taken. (If the branch is not taken, the PC advances to the next instruction.)

Relative addressing differs from that contained in the MC6800 due to two important additions. The first of these is that the offset (signed constant) can be either ±7 bits or ±15 bits in length. This allows the program to branch to any location in the memory field.

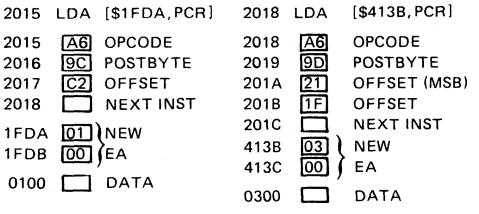
The second important addition to the Relative mode is that it no longer is limited to branch instructions. An Effective Address—which retains the position-independent nature of relative addressing—may be formed by adding a ±7-bit or ±15-bit offset to the program counter. This is, in effect, an Indexed Addressing mode with one of two specific postbytes. (The optional postbytes allow selection of ±7 or ±15-bit offset.) Examples of its use would be:

2015	LDA	-$1FDA,PCR	2018	LDA	$413B,PCR
2015	A6	OPCODE	2018	A6	OPCODE
2016	8C	POSTBYTE	2019	8D	POSTBYTE
2017	C2	OFFSET	201A	21	OFFSET (MSB)
2018		NEXT INST	201B	1F	OFFSET (LSB)
			201C		NEXT INST
1FDA		DATA	413B		DATA

Note that the offset is added to the new value of the program counter, i.e., the location of the next instruction.

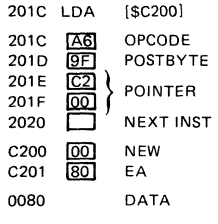
Relative Indirect

This addressing mode is, in effect, Indexed Indirect with the Program Counter used as an index register. One or two bytes (optional) following the postbyte are used to provide a ±7 byte or ±15 bit offset. This signed number is added to the contents of the program counter, forming a pointer to consecutive locations in memory which contain the new effective address. Examples are:



Extended Indirect

This addressing mode is actually another option of Indexed Indirect Addressing. In this case, the two bytes following the postbyte are used as a pointer to consecutive locations in memory which contain the new effective address. An example is:



Absolute Indirect

This mode is exclusively used for Restart and Interrupt vectoring. Servicing of these conditions involves fetching the contents of an exact location in memory to be loaded into the Program Counter.

Instruction Set

A complete listing of the Executable instructions is contained in Tables 2 through 6. Some of the more unique instructions include Load Effective Address, Synchronization with Interrupt, and Exchange Registers. These, along with others not available on the MC6800, are detailed in the following paragraphs.

Load Effective Address – Besides its obvious use, this instruction represents a convenient means of modifying any of the pointer (X, Y, S, or U) registers. The processor forms an Effective Address as dictated by the addressing mode, then loads this value into the designated register rather than outputting the data on the address lines. As an example, LEAX A,X will add the signed number contained in the A register to the contents of the X register, then place this result into the X register.

Synchronize with Interrupt – This instruction causes processing to discontinue until an Interrupt Input ($\overline{NMI}$, $\overline{IRQ}$, or $\overline{FIRQ}$) is activated. When an interrupt occurs, the processor services the interrupt normally if the associated mask is clear. If the interrupt mask is set when the interrupt occurs, the processor exits the Sync mode by continuing to the next instruction. The Bus Available output is activated during the Sync mode, but BS remains low. The instruction is useful for synchronizing the program with a peripheral and for performing DMA under program control.

Exchange and Transfer Registers – Both of these instructions utilize an immediate byte to define the source and destination registers. The only restriction is that both source and destination must be similarly sized registers.

Push/Pull Register(s) – These instructions also use an immediate byte to designate whether the register assigned to a particular bit is to be affected. Thus, a Push instruction followed by a byte containing a "one" in bit 7 causes the Program Counter to be pushed onto a stack. One to eight registers can be pushed or pulled with a single instruction.

Sign Extend – This instruction causes all bits in the A register to take on the value of the Most Significant Bit of the B register.

CWAI – This instruction is similar to the Wait for Interrupt used with the MC6800, but includes an immediate byte to clear condition codes if desired. The CWAI instruction can be used with any of the three interrupt lines, even though CWAI stacks all registers (except S). After stacking is complete, the processor idles until an interrupt occurs.

Those familiar with the MC6800 will note that some instructions are missing from the complement available with the MC6800. Provisions have been made to perform such operations in alternate ways when required. An example might be Decrement X. This will not often be needed with the MC6809 due to the Auto-Decrement option with Indexed Addressing. If needed, however, the operation can be accomplished with an LEAX -1,X instruction. Likewise, the MC6800 instructions to Clear/Set various condition codes are replaced with ANDCC/ORCC. In this manner, the MC6809 uses fewer instruction mnemonics (59 versus 72) than the MC6800, yet is fully software compatible, as well as being considerably more powerful.

TABLE 2 – 8-BIT ACCUMULATOR AND MEMORY INSTRUCTIONS

Mnemonic(s)	Operation	Addressing Modes								
		Implied	Immediate	Direct	Extended	Extended Indirect	Indexed	Indexed Indirect	Relative	Relative Indirect
ADCA, ADCB	Add memory to accumulator with carry	–	X	X	X	X	X	X	X	X
ADDA, ADDB	Add memory to accumulator	–	X	X	X	X	X	X	X	X
ANDA, ANDB	And memory with accumulator	–	X	X	X	X	X	X	X	X
ASL	Arithmetic shift left memory location	–	–	X	X	X	X	X	X	X
ASLA, ASLB	Arithmetic shift left accumulator	X	–	–	–	–	–	–	–	–
ASR	Arithmetic shift right memory location	–	–	X	X	X	X	X	X	X
ASRA, ASRB	Arithmetic shift right accumulator	X	–	–	–	–	–	–	–	–
BITA, BITB	Bit test memory with accumulator	–	X	X	X	X	X	X	X	X
CLR	Clear memory location	–	–	X	X	X	X	X	X	X
CLRA, CLRB	Clear accumulator	X	–	–	–	–	–	–	–	–
CMPA, CMPB	Compare memory with accumulator	–	X	X	X	X	X	X	X	X
COM	Complement memory location	–	–	X	X	X	X	X	X	X
COMA, COMB	Complement accumulator	X	–	–	–	–	–	–	–	–
DAA	Decimal adjust A-accumulator	X	–	–	–	–	–	–	–	–
DEC	Decrement memory location	–	–	X	X	X	X	X	X	X
DECA, DECB	Decrement accumulator	X	–	–	–	–	–	–	–	–
EORA, EORB	Exclusive or memory with accumulator	–	X	X	X	X	X	X	X	X
EXG R1, R2	Exchange R1 with R2 (R1, R2 = A, B, CC, DP)	X	–	–	–	–	–	–	–	–
INC	Increment memory location	–	–	X	X	X	X	X	X	X
INCA, INCB	Increment accumulator	X	–	–	–	–	–	–	–	–
LDA, LDB	Load accumulator from memory	–	X	X	X	X	X	X	X	X
LSL	Logical shift left memory location	–	–	X	X	X	X	X	X	X
LSLA, LSLB	Logical shift left accumulator	X	–	–	–	–	–	–	–	–
LSR	Logical shift right memory location	–	–	X	X	X	X	X	X	X
LSRA, LSRB	Logical shift right accumulator	X	–	–	–	–	–	–	–	–
MUL	Unsigned multiply (AXB → D)	X	–	–	–	–	–	–	–	–
NEG	Negate memory location	–	–	X	X	X	X	X	X	X
NEGA, NEGB	Negate accumulator	X	–	–	–	–	–	–	–	–
ORA, ORB	Or memory with accumulator	–	X	X	X	X	X	X	X	X
ROL	Rotate memory location left	–	–	X	X	X	X	X	X	X
ROLA, ROLB	Rotate accumulator left	X	–	–	–	–	–	–	–	–
ROR	Rotate memory location right	–	–	X	X	X	X	X	X	X
RORA, RORB	Rotate accumulator right	X	–	–	–	–	–	–	–	–
SBCA, SBCB	Subtract memory from accumulator with borrow	–	X	X	X	X	X	X	X	X
STA, STB	Store accumulator to memory	–	–	X	X	X	X	X	X	X
SUBA, SUBB	Subtract memory from accumulator	–	X	X	X	X	X	X	X	X
TST	Test memory location	–	–	X	X	X	X	X	X	X
TSTA, TSTB	Test accumulator	X	–	–	–	–	–	–	–	–
TFR, R1, R2	Transfer R1 to R2 (R1, R2 = A, B, CC, DP)	X	–	–	–	–	–	–	–	–

NOTE: A and B may be pushed to (pulled from) either stack with PSHS, PSHU (PULS, PULU) instructions. See Table 3.

TABLE 3 – 16-BIT ACCUMULATOR AND MEMORY INSTRUCTIONS

Mnemonic(s)	Operation	Addressing Modes								
		Implied	Immediate	Direct	Extended	Extended Indirect	Indexed	Indexed Indirect	Relative	Relative Indirect
ADDD	Add memory to D accumulator	–	X	X	X	X	X	X	X	X
CMPD	Compare memory with D accumulator	–	X	X	X	X	X	X	X	X
EXG D, R	Exchange D with X, Y, S, U, or PC	X	–	–	–	–	–	–	–	–
LDD	Load D accumulator from memory	–	X	X	X	X	X	X	X	X
SEX	Sign Extend	X	–	–	–	–	–	–	–	–
STD	Store D accumulator to memory	–	–	X	X	X	X	X	X	X
SUBD	Subract memory from D accumulator	–	X	X	X	X	X	X	X	X
TFR D, R	Transfer D to X, Y, S, U, or PC	X	–	–	–	–	–	–	–	–
TFR R, D	Transfer X, Y, S, U, or PC to D	X	–	–	–	–	–	–	–	–

TABLE 4 – INDEX REGISTER/STACK POINTER INSTRUCTIONS

Mnemonic(s)	Operation	Addressing Modes								
		Implied	Immediate	Direct	Extended	Extended Indirect	Indexed	Indexed Indirect	Relative	Relative Indirect
CMPS, CMPU	Compare memory with stack pointer	–	X	X	X	X	X	X	X	X
CMPX, CMPY	Compare memory with index register	–	X	X	X	X	X	X	X	X
EXG R1, R2	Exchange D, X, Y, S, U, or PC with D, X, Y, S, U, or PC	X	–	–	–	–	–	–	–	–
LEAS, LEAU	Load effective address into stack pointer	–	–	–	–	X	X	X	X	X
LEAX, LEAY	Load effective address into index register	–	–	–	–	X	X	X	X	X
LDS, LDU	Load stack pointer from memory	–	X	X	X	X	X	X	X	X
LDX, LDY	Load index register from memory	–	X	X	X	X	X	X	X	X
PSHS	Push any register(s) onto hardware stack (except S)	X	–	–	–	–	–	–	–	–
PSHU	Push any register(s) onto user stack (except U)	X	–	–	–	–	–	–	–	–
PULS	Pull any register(s) from hardware stack (except S)	X	–	–	–	–	–	–	–	–
PULU	Pull any register(s) from hardware stack (except U)	X	–	–	–	–	–	–	–	–
STS, STU	Store stack pointer to memory	–	–	X	X	X	X	X	X	X
STX, STY	Store index register to memory	–	–	X	X	X	X	X	X	X
TFR R1, R2	Transfer D, X, Y, S, U, or PC to D, X, Y, S, U, or PC	X	–	–	–	–	–	–	–	–
ABX	Add B-accumulator to X (unsigned)	X	–	–	–	–	–	–	–	–

TABLE 5 – BRANCH INSTRUCTIONS

Mnemonic(s)	Operation	Addressing Modes								
		Implied	Immediate	Direct	Extended	Extended Indirect	Indexed	Indexed Indirect	Relative	Relative Indirect
BCC, LBCC	Branch if carry clear	–	–	–	–	–	–	–	X	–
BCS, LBCS	Branch if carry set	–	–	–	–	–	–	–	X	–
BEQ, LBEQ	Branch if equal	–	–	–	–	–	–	–	X	–
BGE, LBGE	Branch if greater than or equal (signed)	–	–	–	–	–	–	–	X	–
BGT, LBGT	Branch if greater (signed)	–	–	–	–	–	–	–	X	–
BHI, LBHI	Branch if higher (unsigned)	–	–	–	–	–	–	–	X	–
BHS, LBHS	Branch if higher or same (unsigned)	–	–	–	–	–	–	–	X	–
BLE, LBLE	Branch if less than or equal (signed)	–	–	–	–	–	–	–	X	–
BLO, LBLO	Branch if lower (unsigned)	–	–	–	–	–	–	–	X	–
BLS, LBLS	Branch if lower or same (unsigned)	–	–	–	–	–	–	–	X	–
BLT, LBLT	Branch if less than (signed)	–	–	–	–	–	–	–	X	–
BMI, LBMI	Branch if minus	–	–	–	–	–	–	–	X	–
BNE, LBNE	Branch if not equal	–	–	–	–	–	–	–	X	–
BPL, LBPL	Branch if plus	–	–	–	–	–	–	–	X	–
BRA, LBRA	Branch always	–	–	–	–	–	–	–	X	–
BRN, LBRN	Branch never (3, 5 Cycle NOP)	–	–	–	–	–	–	–	X	–
BSR, LBSR	Branch to subroutine	–	–	–	–	–	–	–	X	–
BVC, LBVC	Branch if overflow clear	–	–	–	–	–	–	–	X	–
BVS, LBVS	Branch if overflow set	–	–	–	–	–	–	–	X	–

TABLE 6 – MISCELLANEOUS INSTRUCTIONS

Mnemonic(s)	Operation	Addressing Modes								
		Implied	Immediate	Direct	Extended	Extended Indirect	Indexed	Indexed Indirect	Relative	Relative Indirect
ANDCC	AND condition code register	–	X	–	–	–	–	–	–	–
CWAI	AND condition code register, then wait for interrupt	–	X	–	–	–	–	–	–	–
NOP	No operation	X	–	–	–	–	–	–	–	–
ORCC	OR condition code register	–	X	–	–	–	–	–	–	–
JMP	Jump	–	–	X	X	X	X	X	X	X
JSR	Jump to subroutine	–	–	X	X	X	X	X	X	X
RTI	Return from interrupt	X	–	–	–	–	–	–	–	–
RTS	Return from subroutine	X	–	–	–	–	–	–	–	–
SWI, SWI2, SWI3	Software interrupt (absolute indirect)	X	–	–	–	–	–	–	–	–
SYNC	Synchronize with interrupt line	X	–	–	–	–	–	–	–	–

MOTOROLA *Semiconductor Products Inc.*
3501 ED BLUESTEIN BLVD., AUSTIN, TEXAS 78721 • A SUBSIDIARY OF MOTOROLA INC.

MOTOROLA

MC6821
(1.0 MHz)
MC68A21
(1.5 MHz)
MC68B21
(2.0 MHz)

MOS
(N-CHANNEL, SILICON-GATE, DEPLETION LOAD)

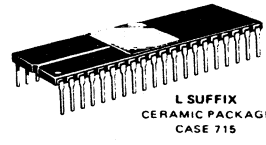
PERIPHERAL INTERFACE ADAPTER

PERIPHERAL INTERFACE ADAPTER (PIA)

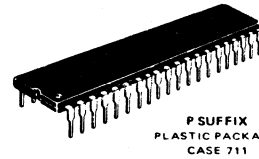
The MC6821 Peripheral Interface Adapter provides the universal means of interfacing peripheral equipment to the MC6800 Microprocessing Unit (MPU). This device is capable of interfacing the MPU to peripherals through two 8-bit bidirectional peripheral data buses and four control lines. No external logic is required for interfacing to most peripheral devices.

The functional configuration of the PIA is programmed by the MPU during system initialization. Each of the peripheral data lines can be programmed to act as an input or output, and each of the four control/interrupt lines may be programmed for one of several control modes. This allows a high degree of flexibility in the over-all operation of the interface.

- 8-Bit Bidirectional Data Bus for Communication with the MPU
- Two Bidirectional 8-Bit Buses for Interface to Peripherals
- Two Programmable Control Registers
- Two Programmable Data Direction Registers
- Four Individually-Controlled Interrupt Input Lines; Two Usable as Peripheral Control Outputs
- Handshake Control Logic for Input and Output Peripheral Operation
- High-Impedance 3-State and Direct Transistor Drive Peripheral Lines
- Program Controlled Interrupt and Interrupt Disable Capability
- CMOS Drive Capability on Side A Peripheral Lines
- Two TTL Drive Capability on All A and B Side Buffers
- TTL-Compatible
- Static Operation

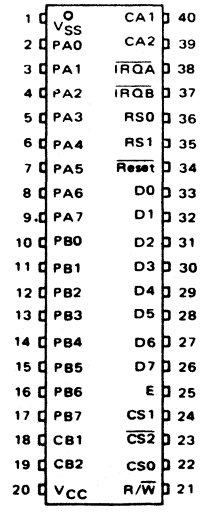


L SUFFIX
CERAMIC PACKAGE
CASE 715



P SUFFIX
PLASTIC PACKAGE
CASE 711

PIN ASSIGNMENT



ORDERING INFORMATION

Speed	Device	Temperature Range
1.0 MHz	MC6821P, L	0 to +70°C
	MC6821CP, CL	-40 to +85°C
MIL-STD-883B MIL-STD-883C	MC6821BQCS MC6821CQCS	- 55 to +125°C
1.5 MHz	MC68A21P, L	0 to +70°C
	MC68A21CP, CL	-40 to +85°C
2.0 MHz	MC68B21P, L	0 to +70°C

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	-0.3 to +7.0	Vdc
Input Voltage	V_{in}	-0.3 to +7.0	Vdc
Operating Temperature Range MC6821, MC68A21, MC68B21 MC6821C, MC68A21C MC6821CQCS, MC6821BQCS	T_A	T_L to T_H 0 to 70 -40 to 85 -55 to 125	 °C °C °C
Storage Temperature Range	T_{stg}	-55 to +150	°C
Thermal Resistance	θ_{JA}	82.5	°C/W

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance.

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ±5%, V_{SS} = 0, T_A = T_L to T_H unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
BUS CONTROL INPUTS (R/$\overline{W}$, Enable, $\overline{Reset}$, RS0, RS1, CS0, CS1, $\overline{CS2}$)					
Input High Voltage	V_{IH}	V_{SS} + 2.0	–	V_{CC}	Vdc
Input Low Voltage	V_{IL}	V_{SS} – 0.3	–	V_{SS} + 0.8	Vdc
Input Leakage Current (V_{in} = 0 to 5.25 Vdc)	I_{in}	–	1.0	2.5	μAdc
Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	C_{in}	–	–	7.5	pF
INTERRUPT OUTPUTS ($\overline{IRQA}$, $\overline{IRQB}$)					
Output Low Voltage (I_{Load} = 3.2 mAdc)	V_{OL}	–	–	V_{SS} + 0.4	Vdc
Output Leakage Current (Off State) (V_{OH} = 2.4 Vdc)	I_{LOH}	–	1.0	10	μAdc
Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	C_{out}	–	–	5.0	pF
DATA BUS (D0-D7)					
Input High Voltage	V_{IH}	V_{SS} + 2.0	–	V_{CC}	Vdc
Input Low Voltage	V_{IL}	V_{SS} – 0.3	–	V_{SS} + 0.8	Vdc
Three-State (Off State) Input Current (V_{in} = 0.4 to 2.4 Vdc)	I_{TSI}	–	2.0	10	μAdc
Output High Voltage (I_{Load} = -205 μAdc)	V_{OH}	V_{SS} + 2.4	–	–	Vdc
Output Low Voltage (I_{Load} = 1.6 mAdc)	V_{OL}	–	–	V_{SS} + 0.4	Vdc
Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	C_{in}	–	–	12.5	pF
PERIPHERAL BUS (PA0-PA7, PB0-PB7, CA1, CA2, CB1, CB2)					
Input Leakage Current (V_{in} = 0 to 5.25 Vdc) R/$\overline{W}$, $\overline{Reset}$, RS0, RS1, CS0, CS1, $\overline{CS2}$, CA1, CB1, Enable	I_{in}	–	1.0	2.5	μAdc
Three-State (Off State) Input Current (V_{in} = 0.4 to 2.4 Vdc) PB0-PB7, CB2	I_{TSI}	–	2.0	10	μAdc
Input High Current (V_{IH} = 2.4 Vdc) PA0-PA7, CA2	I_{IH}	-200	-400	–	μAdc
Darlington Drive Current V_O = 1.5 Vdc PB0-PB7,CB2	I_{OH}	-1.0	–	-10	mAdc
Input Low Current (V_{IL} = 0.4 Vdc) PA0-PA7, CA2	I_{IL}	–	-1.3	-2.4	mAdc
Output High Voltage	V_{OH}				Vdc
(I_{Load} = -200 μAdc) PA0-P7, PB0-PB7, CA2, CB2		V_{SS} + 2.4	–	–	
(I_{Load} = -10 μAdc) PA0-PA7, CA2		V_{CC} - 1.0	–	–	
Output Low Voltage (I_{Load} = 3.2 mAdc)	V_{OL}	–	–	V_{SS} + 0.4	Vdc
Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	C_{in}	–	–	10	pF
POWER REQUIREMENTS					
Power Dissipation	P_D	–	–	550	mW

BUS TIMING CHARACTERISTICS (V_{CC} = 5.0 V ±5%, V_{SS} = 0, T_A = T_L to T_H unless otherwise specified.)

Characteristic	Symbol	MC6821		MC68A21		MC68B21		Unit	Ref. Fig. No.
		Min	Max	Min	Max	Min	Max		
Enable Cycle Time	t_{cycE}	1000	–	666	–	500	–	ns	1
Enable Pulse Width, High	PW_{EH}	450	–	280	–	220	–	ns	1
Enable Pulse Width, Low	PW_{EL}	430	–	280	–	210	–	ns	1
Enable Pulse Rise and Fall Times	t_{Er}, t_{Ef}	–	25	–	25	–	25	ns	1
Setup Time, Address and R/W valid to Enable positive transition	t_{AS}	160	–	140	–	70	–	ns	2, 3
Address Hold Time	t_{AH}	10	–	10	–	10	–	ns	2, 3
Data Delay Time, Read	t_{DDR}	–	320	–	220	–	180	ns	2, 4
Data Hold Time, Read	t_{DHR}	10	–	10	–	10	–	ns	2, 4
Data Setup Time, Write	t_{DSW}	195	–	80	–	60	–	ns	3, 4
Data Hold Time, Write	t_{DHW}	10	–	10	–	10	–	ns	3, 4

FIGURE 1 – ENABLE SIGNAL CHARACTERISTICS

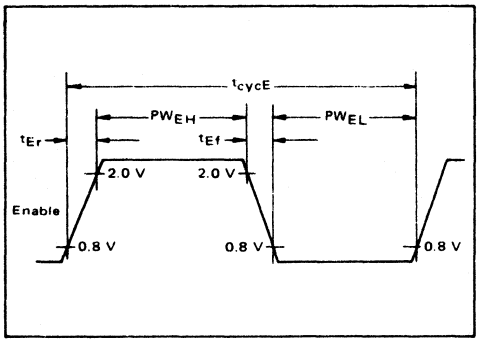


FIGURE 2 – BUS READ TIMING CHARACTERISTICS
(Read Information from PIA)

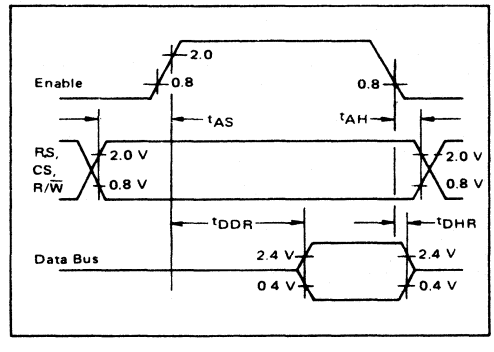


FIGURE 3 – BUS WRITE TIMING CHARACTERISTICS
(Write Information into PIA)

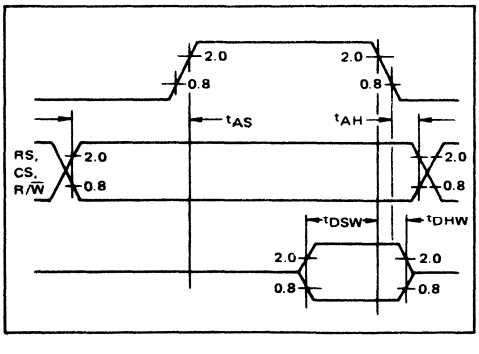
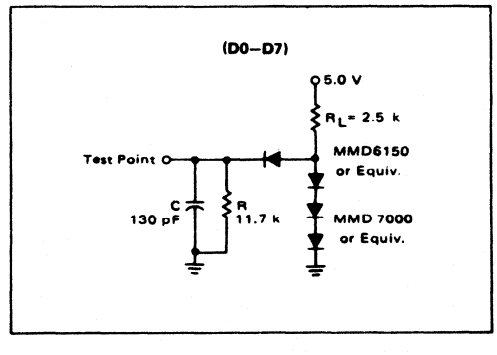


FIGURE 4 – BUS TIMING TEST LOADS



PERIPHERAL TIMING CHARACTERISTICS (V_{CC} = 5.0 V ±5%, V_{SS} = 0 V, T_A = T_L to T_H unless otherwise specified.)

Characteristic	Symbol	MC6821		MC68A21		MC68B21		Unit	Reference Fig. No.
		Min	Max	Min	Max	Min	Max		
Peripheral Data Setup Time	t_{PDSU}	200	–	135	–	100	–	ns	8
Peripheral Data Hold Time	t_{PDH}	0	–	0	–	0	–	ns	8
Delay Time, Enable negative transition to CA2 negative transition	t_{CA2}	–	1.0	–	0.670	–	0.500	µs	5, 9, 10
Delay Time, Enable negative transition to CA2 positive transition	t_{RS1}	–	1.0	–	0.670	–	0.500	µs	5, 9
Rise and Fall Times for CA1 and CA2 input signals	t_r, t_f	–	1.0	–	1.0	–	1.0	µs	5, 10
Delay Time from CA1 active transition to CA2 positive transition	t_{RS2}	–	2.0	–	1.35	–	1.0	µs	5, 10
Delay Time, Enable negative transition to Peripheral Data Valid	t_{PDW}	–	1.0	–	0.670	–	0.5	µs	5, 11, 12
Delay Time, Enable negative transition to Peripheral CMOS Data Valid PA0-PA7, CA2	t_{CMOS}	–	2.0	–	1.35	–	1.0	µs	6, 11
Delay Time, Enable positive transition to CB2 negative transition	t_{CB2}	–	1.0	–	0.670	–	0.5	µs	5, 13, 14
Delay Time, Peripheral Data Valid to CB2 negative transition	t_{DC}	20	–	20	–	20	–	ns	5, 12
Delay Time, Enable positive transition to CB2 positive transition	t_{RS1}	–	1.0	–	0.670	–	0.5	µs	5, 13
Peripheral Control Output Pulse Width, CA2/CB2	PW_{CT}	550	–	550	–	500	–	ns	5, 13
Rise and Fall Time for CB1 and CB2 input signals	t_r, t_f	–	1.0	–	1.0	–	1.0	µs	14
Delay Time, CB1 active transition to CB2 positive transition	t_{RS2}	–	2.0	–	1.35	–	1.0	µs	5, 14
Interrupt Release Time, $\overline{IRQA}$ and $\overline{IRQB}$	t_{IR}	–	1.60	–	1.10	–	0.85	µs	7, 16
Interrupt Response Time	t_{RS3}	–	1.0	–	1.0	–	1.0	µs	7, 15
Interrupt Input Pulse Width	PW_I	500	–	500	–	500	–	ns	15
Reset Low Time*	t_{RL}	1.0	–	0.66	–	0.5	–	µs	17

*The Reset line must be high a minimum of 1.0 µs before addressing the PIA.

FIGURE 5 – TTL EQUIV. TEST LOAD

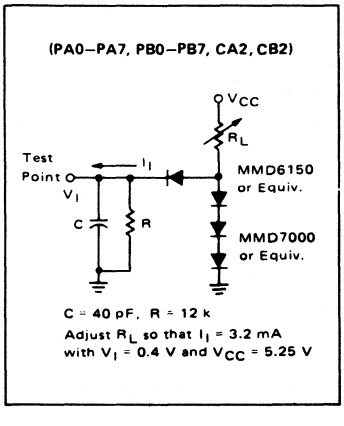


FIGURE 6 – CMOS EQUIV. TEST LOAD

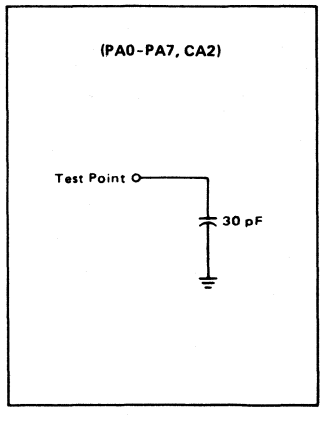


FIGURE 7 – NMOS EQUIV. TEST LOAD

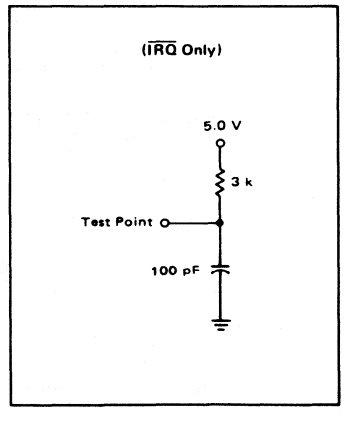


FIGURE 8 – PERIPHERAL DATA SETUP AND HOLD TIMES
(Read Mode)

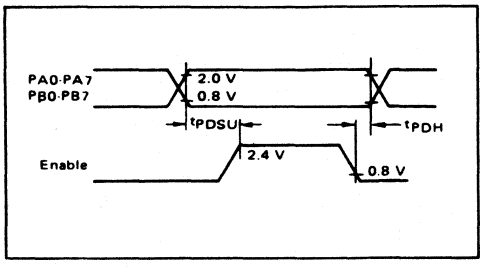


FIGURE 9 – CA2 DELAY TIME
(Read Mode; CRA-5 = CRA-3 = 1, CRA-4 = 0)

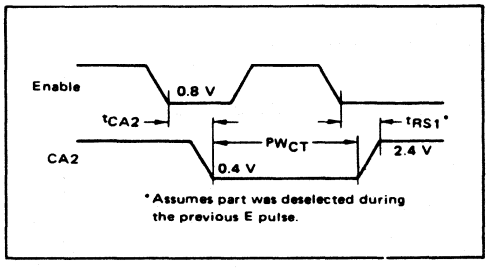


FIGURE 10 – CA2 DELAY TIME
(Read Mode; CRA-5 = 1, CRA-3 = CRA-4 = 0)

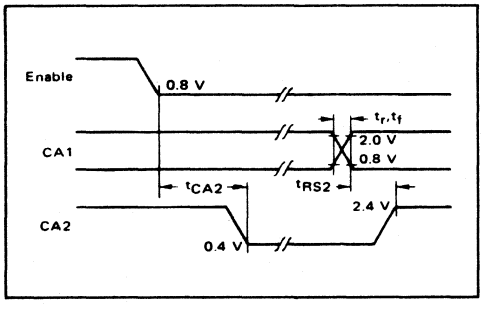


FIGURE 11 – PERIPHERAL CMOS DATA DELAY TIMES
(Write Mode; CRA-5 = CRA-3 = 1, CRA-4 = 0)

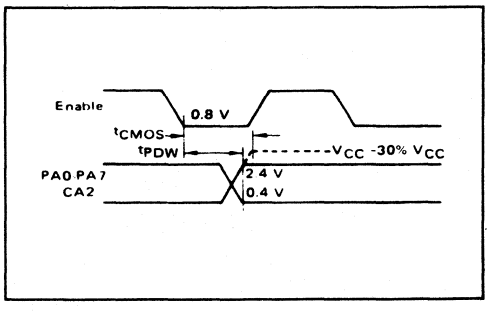


FIGURE 12 – PERIPHERAL DATA AND CB2 DELAY TIMES
(Write Mode; CRB-5 = CRB-3 = 1, CRB-4 = 0)

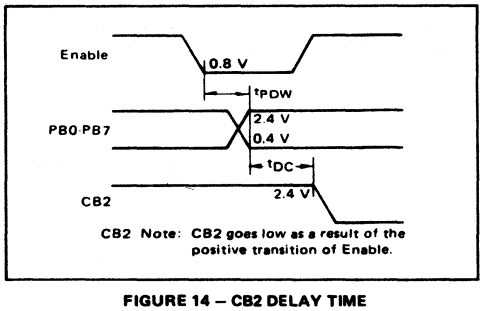


FIGURE 13 – CB2 DELAY TIME
(Write Mode; CRB-5 = CRB-3 = 1, CRB-4 = 0)

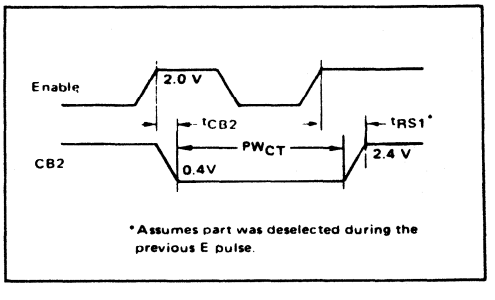


FIGURE 14 – CB2 DELAY TIME
(Write Mode; CRB-5 = 1, CRB-3 = CRB-4 = 0)

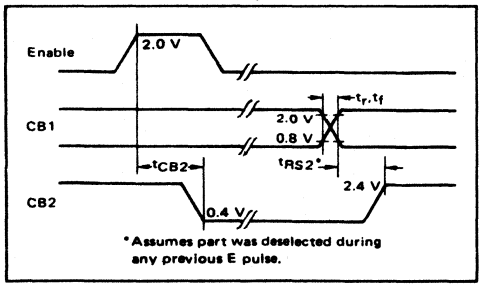


FIGURE 15 – INTERRUPT PULSE WIDTH AND $\overline{IRQ}$ RESPONSE

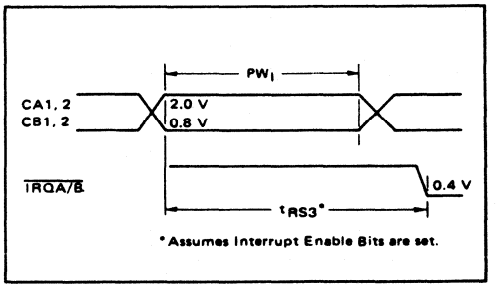


FIGURE 16 – $\overline{IRQ}$ RELEASE TIME

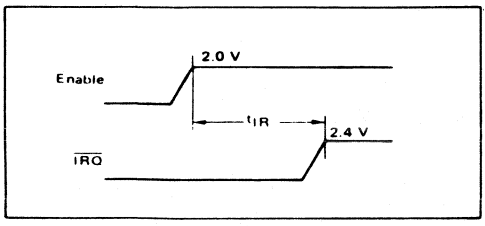
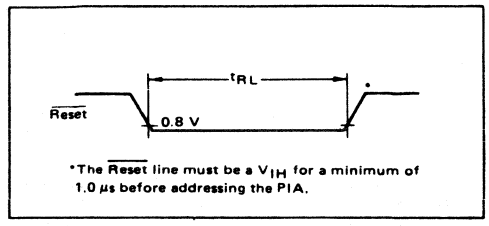
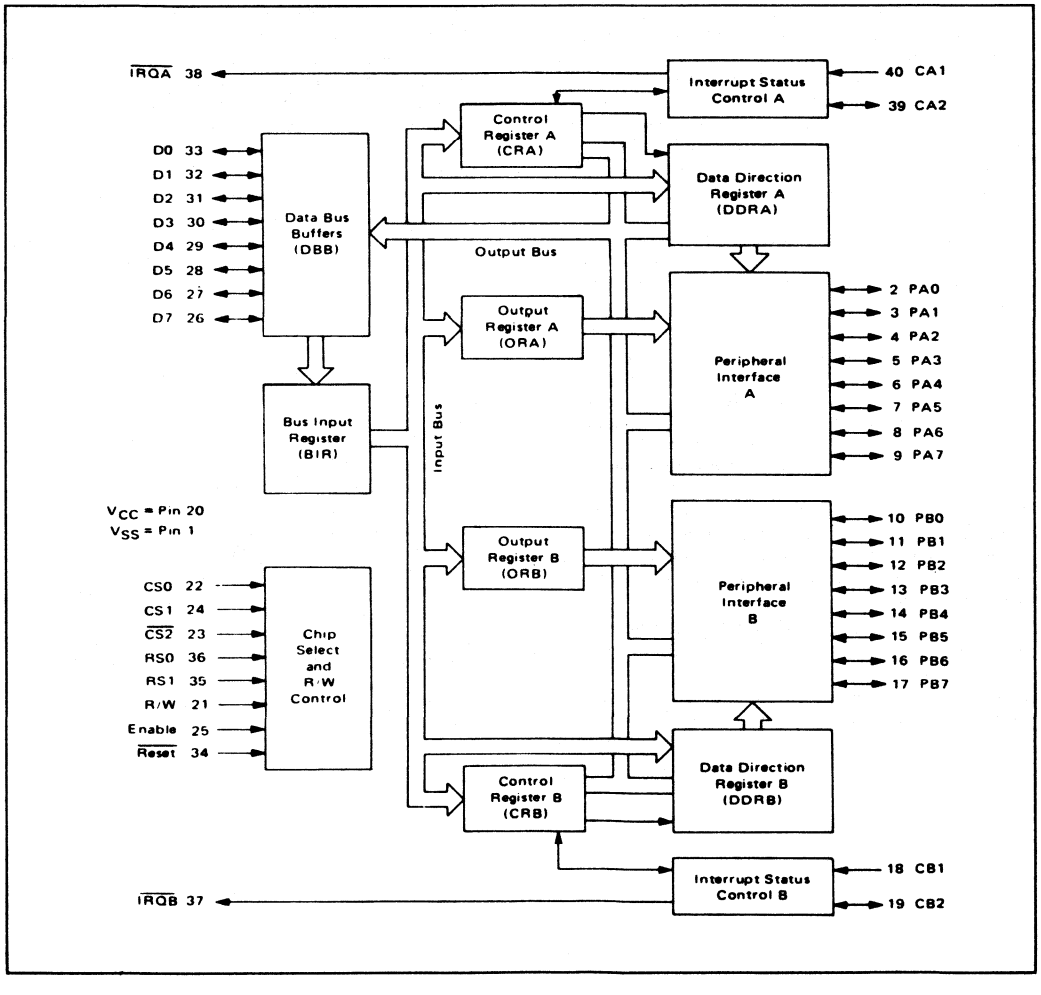


FIGURE 17 – $\overline{RESET}$ LOW TIME



*The $\overline{Reset}$ line must be a V_{IH} for a minimum of 1.0 µs before addressing the PIA.

EXPANDED BLOCK DIAGRAM



PIA INTERFACE SIGNALS FOR MPU

The PIA interfaces to the MC6800 MPU with an eight-bit bi-directional data bus, three chip select lines, two register select lines, two interrupt request lines, read/write line, enable line and reset line. These signals, in conjunction with the MC6800 VMA output, permit the MPU to have complete control over the PIA. VMA should be utilized in conjunction with an MPU address line into a chip select of the PIA.

PIA Bi-Directional Data (D0-D7) – The bi-directional data lines (D0-D7) allow the transfer of data between the MPU and the PIA. The data bus output drivers are three-state devices that remain in the high-impedance (off) state except when the MPU performs a PIA read operation. The Read/Write line is in the Read (high) state when the PIA is selected for a Read operation.

PIA Enable (E) – The enable pulse, E, is the only timing signal that is supplied to the PIA. Timing of all other signals is referenced to the leading and trailing edges of the E pulse. This signal will normally be a derivative of the MC6800 $\phi 2$ Clock.

PIA Read/Write (R/W) – This signal is generated by the MPU to control the direction of data transfers on the Data Bus. A low state on the PIA Read/Write line enables the input buffers and data is transferred from the MPU to the PIA on the E signal if the device has been selected. A high on the Read/Write line sets up the PIA for a transfer of data to the bus. The PIA output buffers are enabled when the proper address and the enable pulse E are present.

$\overline{\text{Reset}}$ – The active low $\overline{\text{Reset}}$ line is used to reset all register bits in the PIA to a logical zero (low). This line can be used as a power-on reset and as a master reset during system operation.

PIA Chip Select (CS0, CS1 and $\overline{\text{CS2}}$) – These three input signals are used to select the PIA. CS0 and CS1 must be high and $\overline{\text{CS2}}$ must be low for selection of the device. Data transfers are then performed under the control of the Enable and Read/Write signals. The chip select lines must be stable for the duration of the E pulse. The device is deselected when any of the chip selects are in the inactive state.

PIA Register Select (RS0 and RS1) – The two register select lines are used to select the various registers inside the PIA. These two lines are used in conjunction with internal Control Registers to select a particular register that is to be written or read.

The register and chip select lines should be stable for the duration of the E pulse while in the read or write cycle.

$\overline{\text{Interrupt Request}}$ ($\overline{\text{IRQA}}$ and $\overline{\text{IRQB}}$) The active low $\overline{\text{Interrupt Request}}$ lines ($\overline{\text{IRQA}}$ and $\overline{\text{IRQB}}$) act to interrupt the MPU either directly or through interrupt priority circuitry. These lines are "open drain" (no load device on the chip). This permits all interrupt request lines to be tied together in a wire-OR configuration.

Each $\overline{\text{Interrupt Request}}$ line has two internal interrupt flag bits that can cause the $\overline{\text{Interrupt Request}}$ line to go low. Each flag bit is associated with a particular peripheral interrupt line. Also four interrupt enable bits are provided in the PIA which may be used to inhibit a particular interrupt from a peripheral device.

Servicing an interrupt by the MPU may be accomplished by a software routine that, on a prioritized basis, sequentially reads and tests the two control registers in each PIA for interrupt flag bits that are set.

The interrupt flags are cleared (zeroed) as a result of an MPU Read Peripheral Data Operation of the corresponding data register. After being cleared, the interrupt flag bit cannot be enabled to be set until the PIA is deselected during an E pulse. The E pulse is used to condition the interrupt control lines (CA1, CA2, CB1, CB2). When these lines are used as interrupt inputs at least one E pulse must occur from the inactive edge to the active edge of the interrupt input signal to condition the edge sense network. If the interrupt flag has been enabled and the edge sense circuit has been properly conditioned, the interrupt flag will be set on the next active transition of the interrupt input pin.

PIA PERIPHERAL INTERFACE LINES

The PIA provides two 8-bit bi-directional data buses and four interrupt/control lines for interfacing to peripheral devices.

Section A Peripheral Data (PA0-PA7) – Each of the peripheral data lines can be programmed to act as an input or output. This is accomplished by setting a "1" in the corresponding Data Direction Register bit for those lines which are to be outputs. A "0" in a bit of the Data Direction Register causes the corresponding peripheral data line to act as an input. During an MPU Read Peripheral Data Operation, the data on peripheral lines programmed to act as inputs appears directly on the corresponding MPU Data Bus lines. In the input mode the internal pullup resistor on these lines represents a maximum of 1.5 standard TTL loads.

The data in Output Register A will appear on the data lines that are programmed to be outputs. A logical "1" written into the register will cause a "high" on the corresponding data line while a "0" results in a "low". Data in Output Register A may be read by an MPU "Read Peripheral Data A" operation when the corresponding lines are programmed as outputs. This data will be read properly if the voltage on the peripheral data lines is greater than 2.0 volts for a logic "1" output and less than 0.8 volt for a logic "0" output. Loading the output lines such that the voltage on these lines does not reach full voltage causes the data transferred into the MPU on a Read operation to differ from that contained in the respective bit of Output Register A.

Section B Peripheral Data (PB0-PB7) – The peripheral data lines in the B Section of the PIA can be programmed to act as either inputs or outputs in a similar manner to PA0-PA7. However, the output buffers driving these lines differ from those driving lines PA0-PA7. They have three-state capability, allowing them to enter a high impedance state when the peripheral data line is used as an input. In addition, data on the peripheral data lines PB0-PB7 will be read properly from those lines programmed as outputs even if the voltages are below 2.0 volts for a "high". As outputs, these lines are compatible with standard TTL and may also be used as a source of up to 1 milliampere at 1.5 volts to directly drive the base of a transistor switch.

Interrupt Input (CA1 and CB1) – Peripheral Input lines CA1 and CB1 are input only lines that set the interrupt flags of the control registers. The active transition for these signals is also programmed by the two control registers.

Peripheral Control (CA2) – The peripheral control line CA2 can be programmed to act as an interrupt input or as a peripheral control output. As an output, this line is compatible with standard TTL; as an input the internal pullup resistor on this line represents 1.5 standard TTL loads. The function of this signal line is programmed with Control Register A.

Peripheral Control (CB2) – Peripheral Control line CB2 may also be programmed to act as an interrupt input or peripheral control output. As an input, this line has high input impedance and is compatible with standard TTL. As an output it is compatible with standard TTL and may also be used as a source of up to 1 milliampere at 1.5 volts to directly drive the base of a transistor switch. This line is programmed by Control Register B.

INTERNAL CONTROLS

There are six locations within the PIA accessible to the MPU data bus: two Peripheral Registers, two Data Direction Registers, and two Control Registers. Selection of these locations is controlled by the RS0 and RS1 inputs together with bit 2 in the Control Register, as shown in Table 1.

TABLE 1 – INTERNAL ADDRESSING

RS1	RS0	Control Register Bit CRA-2	Control Register Bit CRB-2	Location Selected
0	0	1	X	Peripheral Register A
0	0	0	X	Data Direction Register A
0	1	X	X	Control Register A
1	0	X	1	Peripheral Register B
1	0	X	0	Data Direction Register B
1	1	X	X	Control Register B

X = Don't Care

INITIALIZATION

A low reset line has the effect of zeroing all PIA registers. This will set PA0-PA7, PB0-PB7, CA2 and CB2 as inputs, and all interrupts disabled. The PIA must be configured during the restart program which follows the reset.

Details of possible configurations of the Data Direction and Control Register are as follows.

DATA DIRECTION REGISTERS (DDRA and DDRB)

The two Data Direction Registers allow the MPU to control the direction of data through each corresponding peripheral data line. A Data Direction Register bit set at "0" configures the corresponding peripheral data line as an input; a "1" results in an output.

CONTROL REGISTERS (CRA and CRB)

The two Control Registers (CRA and CRB) allow the MPU to control the operation of the four peripheral control lines CA1, CA2, CB1 and CB2. In addition they allow the MPU to enable the interrupt lines and monitor the status of the interrupt flags. Bits 0 through 5 of the two registers may be written or read by the MPU when the proper chip select and register select signals are applied. Bits 6 and 7 of the two registers are read only and are modified by external interrupts occurring on control lines CA1, CA2, CB1 or CB2. The format of the control words is shown in Table 2.

TABLE 2 – CONTROL WORD FORMAT

	7	6	5	4	3	2	1	0
CRA	IRQA1	IRQA2	CA2 Control			DDRA Access	CA1 Control	
CRB	IRQB1	IRQB2	CB2 Control			DDRB Access	CB1 Control	

Data Direction Access Control Bit (CRA-2 and CRB-2) – Bit 2 in each Control register (CRA and CRB) allows selection of either a Peripheral Interface Register or the Data Direction Register when the proper register select signals are applied to RS0 and RS1.

Interrupt Flags (CRA-6, CRA-7, CRB-6, and CRB-7) – The four interrupt flag bits are set by active transitions of signals on the four Interrupt and Peripheral Control lines when those lines are programmed to be inputs. These bits cannot be set directly from the MPU Data Bus and are reset indirectly by a Read Peripheral Data Operation on the appropriate section.

TABLE 3 – CONTROL OF INTERRUPT INPUTS CA1 AND CB1

CRA-1 (CRB-1)	CRA-0 (CRB-0)	Interrupt Input CA1 (CB1)	Interrupt Flag CRA-7 (CRB-7)	MPU Interrupt Request $\overline{IRQA}$ ($\overline{IRQB}$)
0	0	Active	Set high on ↓ of CA1 (CB1)	Disabled — $\overline{IRQ}$ remains high
0	1	Active	Set high on ↓ of CA1 (CB1)	Goes low when the interrupt flag bit CRA-7 (CRB-7) goes high
1	0	Active	Set high on ↑ of CA1 (CB1)	Disabled — $\overline{IRQ}$ remains high
1	1	Active	Set high on ↑ of CA1 (CB1)	Goes low when the interrupt flag bit CRA-7 (CRB-7) goes high

Notes 1 ↑ indicates positive transition (low to high)

2 ↓ indicates negative transition (high to low)

3 The Interrupt flag bit CRA-7 is cleared by an MPU Read of the A Data Register, and CRB-7 is cleared by an MPU Read of the B Data Register.

4 If CRA-0 (CRB-0) is low when an interrupt occurs (Interrupt disabled) and is later brought high, $\overline{IRQA}$ ($\overline{IRQB}$) occurs after CRA-0 (CRB-0) is written to a "one".

Control of CA1 and CB1 Interrupt Input Lines (CRA-0, CRB-0, CRA-1, and CRB-1) – The two lowest order bits of the control registers are used to control the interrupt input lines CA1 and CB1. Bits CRA-0 and CRB-0 are used to enable the MPU interrupt signals $\overline{IRQA}$ and $\overline{IRQB}$, respectively. Bits CRA-1 and CRB-1 determine the active transition of the interrupt input signals CA1 and CB1 (Table 3).

TABLE 4 – CONTROL OF CA2 AND CB2 AS INTERRUPT INPUTS
CRA5 (CRB5) is low

CRA-5 (CRB-5)	CRA-4 (CRB-4)	CRA-3 (CRB-3)	Interrupt Input CA2 (CB2)	interrupt Flag CRA-6 (CRB-6)	MPU Interrupt Request $\overline{IRQA}$ ($\overline{IRQB}$)
0	0	0	↓ Active	Set high on ↓ of CA2 (CB2)	Disabled — $\overline{IRQ}$ remains high
0	0	1	↓ Active	Set high on ↓ of CA2 (CB2)	Goes low when the interrupt flag bit CRA-6 (CRB-6) goes high
0	1	0	↑ Active	Set high on ↑ of CA2 (CB2)	Disabled — $\overline{IRQ}$ remains high
0	1	1	↑ Active	Set high on ↑ of CA2 (CB2)	Goes low when the interrupt flag bit CRA-6 (CRB-6) goes high

Notes: 1. ↑ indicates positive transition (low to high)

2. ↓ indicates negative transition (high to low)

3. The Interrupt flag bit CRA-6 is cleared by an MPU Read of the A Data Register and CRB-6 is cleared by an MPU Read of the B Data Register.

4. If CRA-3 (CRB-3) is low when an interrupt occurs (Interrupt disabled) and is later brought high, $\overline{IRQA}$ ($\overline{IRQB}$) occurs after CRA-3 (CRB-3) is written to a "one".

TABLE 5 – CONTROL OF CB2 AS AN OUTPUT
CRB-5 is high

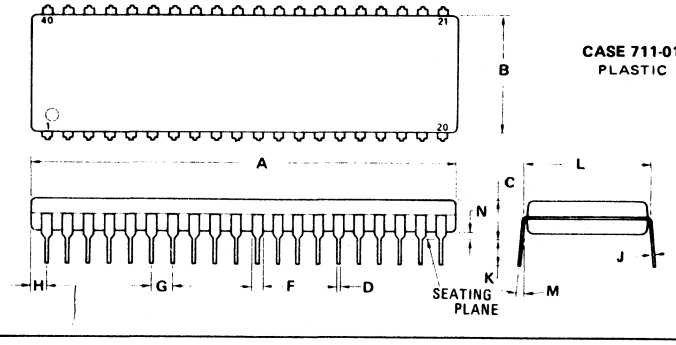
CRB-5	CRB-4	CRB-3	CB2 Cleared	CB2 Set
1	0	0	Low on the positive transition of the first E pulse following an MPU Write "B" Data Register operation.	High when the interrupt flag bit CRB-7 is set by an active transition of the CB1 signal.
1	0	1	Low on the positive transition of the first E pulse after an MPU Write "B" Data Register operation.	High on the positive edge of the first "E" pulse following an "E" pulse which occurred while the part was deselected.
1	1	0	Low when CRB-3 goes low as a result of an MPU Write in Control Register "B".	Always low as long as CRB-3 is low. Will go high on an MPU Write in Control Register "B" that changes CRB-3 to "one".
1	1	1	Always high as long as CRB-3 is high. Will be cleared when an MPU Write Control Register "B" results in clearing CRB-3 to "zero".	High when CRB-3 goes high as a result of an MPU Write into Control Register "B".

Control of CA2 and CB2 Peripheral Control Lines (CRA-3, CRA-4, CRA-5, CRB-3, CRB-4, and CRB-5) – Bits 3, 4, and 5 of the two control registers are used to control the CA2 and CB2 Peripheral Control lines. These bits determine if the control lines will be an interrupt input or an output control signal. If bit CRA-5 (CRB-5) is low, CA2 (CB2) is an interrupt input line similar to CA1 (CB1) (Table 4). When CRA-5 (CRB-5) is high, CA2 (CB2) becomes an output signal that may be used to control peripheral data transfers. When in the output mode, CA2 and CB2 have slightly different characteristics (Tables 5 and 6).

TABLE 6 – CONTROL OF CA-2 AS AN OUTPUT
CRA-5 is high

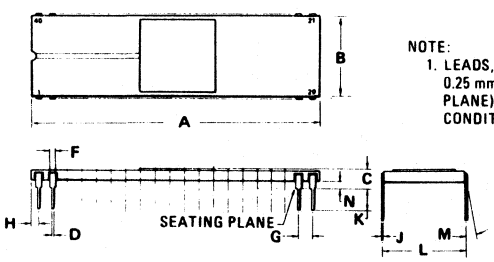
CRA-5	CRA-4	CRA-3	CA2 Cleared	CA2 Set
1	0	0	Low on negative transition of E after an MPU Read "A" Data operation.	High when the interrupt flag bit CRA-7 is set by an active transition of the CA1 signal.
1	0	1	Low on negative transition of E after an MPU Read "A" Data operation.	High on the negative edge of the first "E" pulse which occurs during a deselect.
1	1	0	Low when CRA-3 goes low as a result of an MPU Write to Control Register "A".	Always low as long as CRA-3 is low. Will go high on an MPU Write to Control Register "A" that changes CRA-3 to "one".
1	1	1	Always high as long as CRA-3 is high. Will be cleared on an MPU Write to Control Register "A" that clears CRA-3 to a "zero".	High when CRA-3 goes high as a result of an MPU Write to Control Register "A".

PACKAGE DIMENSIONS



CASE 711-01
PLASTIC

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	51.82	52.32	2.040	2.060
B	13.72	14.22	0.540	0.560
C	4.57	5.08	0.180	0.200
D	0.36	0.51	0.014	0.020
F	1.02	1.52	0.040	0.060
G	2.41	2.67	0.095	0.105
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	3.68	4.19	0.145	0.165
L	14.99	15.49	0.590	0.610
M	0°	10°	0°	10°
N	0.51	1.02	0.020	0.040



NOTE:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA (AT SEATING PLANE), AT MAX. MAT'L CONDITION.

CASE 715-02
CERAMIC

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	50.29	51.31	1.980	2.020
B	14.86	15.62	0.585	0.615
C	2.54	4.19	0.100	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.33	0.008	0.013
K	2.54	4.19	0.100	0.165
L	14.60	15.37	0.575	0.605
M	–	10°	–	10°
N	0.51	1.52	0.020	0.060

MOTOROLA

MC6828
MC8507

Note: The dual numbering system emphasizes that this device is a bipolar LSI device and directly compatible with the M6800 Microprocessor Family. The Priority Interrupt Controller may be ordered by using either part number.

MEGALOGIC

PRIORITY INTERRUPT CONTROLLER

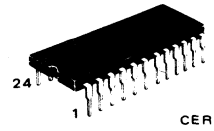
PRIORITY INTERRUPT CONTROLLER

The MC8507/6828 Priority Interrupt Controller (PIC) is used to add prioritized responses to inputs to microprocessor systems. The performance has been optimized for the M6800 system, but will serve to eliminate input polling routines from any processor system.

The MC8507/6828 PIC modifies the ROM address which the processor uses to find the start of the polling or other interrupt service routine. When using the PIC in non-M6800 systems, the address for the service routine must end in . . . 1100x (where x indicates the don't care state of the LSB of the address), and any second byte of the routine address must also end in . . . 1100x.

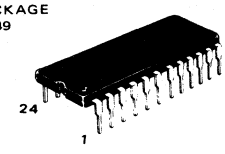
The PIC allows for any added decode time by generating a $\overline{\text{Stretch}}$ signal which can be used to slow the processor clock while fetching interrupt routine starting addresses. The $\overline{\text{Stretch}}$ signal allows the interrupt structure to be designed without concern for faster operation due to improvements in processor speeds.

An interrupt mask prevents any latched interrupt input of lower priority than the mask level from generating an $\overline{\text{IRQ}}$ output.

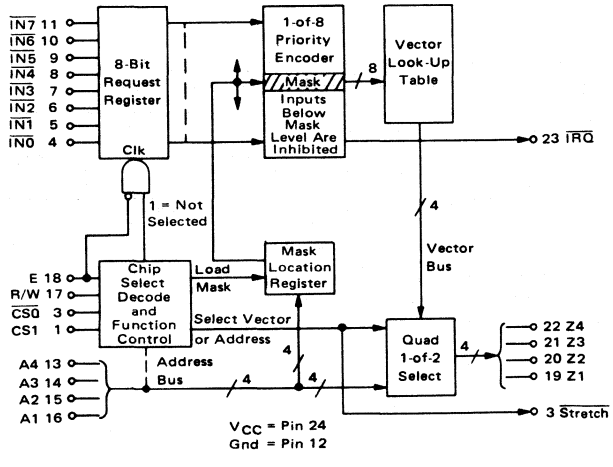


L SUFFIX
CERAMIC PACKAGE
CASE 623

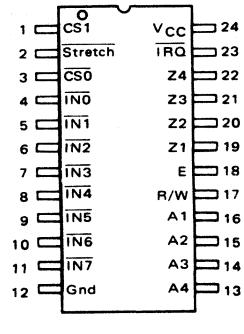
P SUFFIX
PLASTIC PACKAGE
CASE 649



BLOCK DIAGRAM



PIN ASSIGNMENT



MEGALOGIC is a trademark of Motorola Inc.

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	-0.5 to +7.0	Vdc
Input Voltage	V_{in}	-1.0 to +5.5	Vdc
Output Voltage	V_{OH}	-0.4 to +7.0	Vdc
Thermal Resistance	θJA	65	$^{o}C/W$
Operating Temperature Range	T_A	0 to +75	^{o}C
Storage Temperature Range	T_{stg}	-55 to +165	^{o}C

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 Vdc ±5%, T_A = 0 to 75° unless otherwise noted.)

Characteristic		Symbol	Min	Max	Unit
Input Forward Current		I_{IL}			μAdc
(V_{IL} = 0, V_{CC} = 5.25 Vdc)	CS1, E		–	-75	
	$\overline{CS0}$, R/W		–	-150	
	A1 thru A4		–	-225	
	$\overline{IN0}$ thru $\overline{IN7}$		–	-1.300	mAdc
Input Leakage Current		I_{IH}			μAdc
(V_{IH} = 2.4 Vdc, V_{CC} = 5.25 Vdc)	CS1		–	120	
	$\overline{CS0}$		–	240	
	A1 thru A4		–	360	
	$\overline{IN0}$ thru $\overline{IN7}$		–	-560	
Logic "0" Output Voltage		V_{OL}			Vdc
(I_{OL} = 1.6 mAdc, V_{ILT} = 0.8 Vdc, V_{IHT} = 2.0 Vdc, V_{CC} = 4.75 Vdc)					
	Z1 thru Z4, $\overline{Stretch}$		–	0.5	
(I_{OL} = 3.2 mAdc, V_{CC} = 4.75 Vdc)	$\overline{IRQ}$ – Open Collector		–	0.5	
Logic "1" Output Voltage		V_{OH}	2.4	–	Vdc
(I_{OH} = -0.3 mAdc, V_{ILT} = 0.8 Vdc, V_{IHT} = 2.0 Vdc, V_{CC} = 4.75 Vdc)	Z1 thru Z4, $\overline{Stretch}$				
Output Leakage Current		I_{CEX}	–	200	μAdc
(V_{CC} = V_{CEX} = 5.25 Vdc)	$\overline{IRQ}$				
Power Supply Drain Current		I_{CC}	–	125	mAdc
(V_{CC} = 5.0 Vdc, All Inputs Open)					

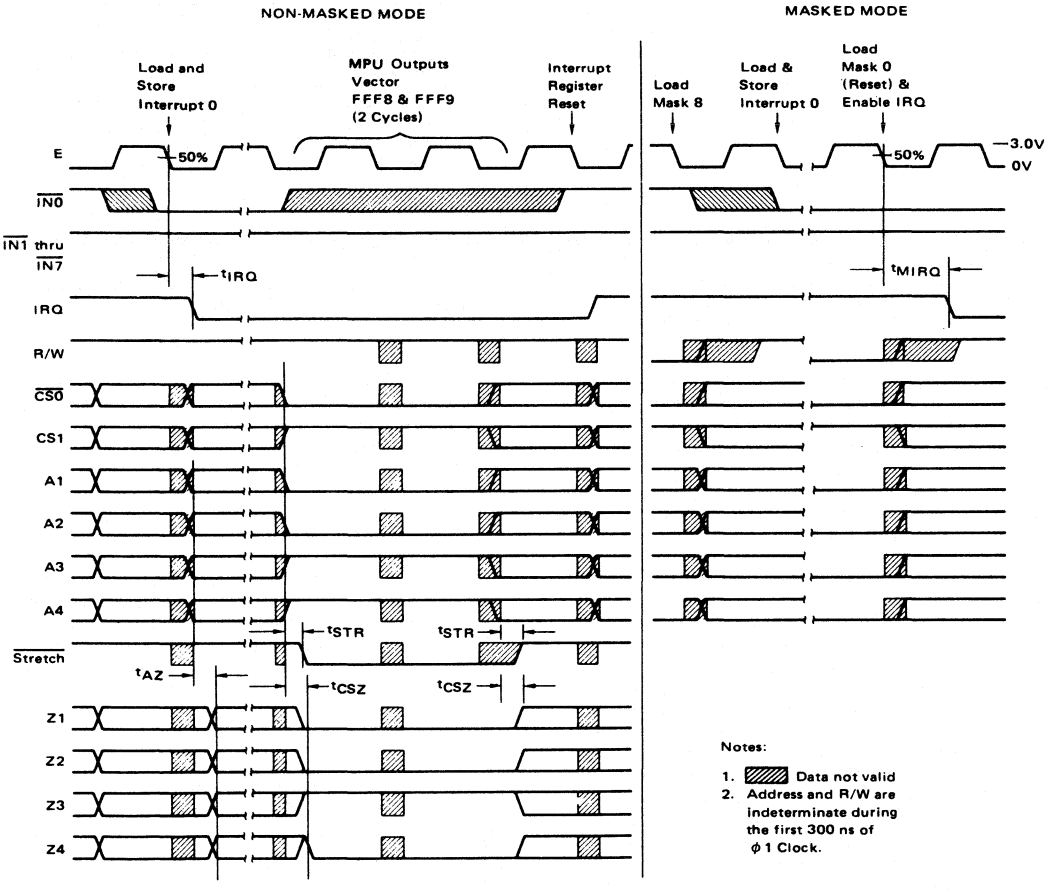
SWITCHING TIMES (V_{CC} = 5.0 Vdc, T_A = 25°C)

Characteristic	Symbol	Min	Typ	Max	Unit
A_i to Z_i Delay Time (Not Selected)	t_{AZ}	–	50	–	ns
Select* to Z_i Delay Time ($\bar{A}1 \cdot \bar{A}2 \cdot A3 \cdot A4 \cdot \overline{CS0} \cdot CS1$ to Z_i)	t_{CSZ}	–	125	–	ns
Select* to $\overline{Stretch}$ Delay Time ($\bar{A}1 \cdot \bar{A}2 \cdot A3 \cdot A4 \cdot \overline{CS0} \cdot CS1$ to $\overline{Stretch}$)	t_{STR}	–	110	–	ns
Enable to $\overline{IRQ}$ Delay Time, Non-Masked Mode	t_{IRQ}	–	220	–	ns
Enable to $\overline{IRQ}$ Delay Time, Masked Mode	t_{MIRQ}	–	**	–	ns

*Select = ($\bar{A}1 \cdot \bar{A}2 \cdot A3 \cdot A4 \cdot \overline{CS0} \cdot CS1 \cdot R/W$) which corresponds to FFF8 or FFF9 interrupt response in the M6800 system.

**Value depends on mask level and stored priority input. Maximum value occurs with mask level 8 and stored interrupt $\overline{IN0}$. Minimum value occurs with mask level J and stored interrupt $\overline{IN(J-1)}$.

FIGURE 1 – FUNCTIONAL WAVEFORMS



OPERATING CHARACTERISTICS

The primary purpose of the Priority Interrupt Controller (PIC) is to generate a modified address to ROM in response to prioritized inputs. With the PIC, each interrupting device is assigned a unique ROM location which contains the starting address of the appropriate service routine. After the MPU detects and responds to an interrupt the PIC directs the MPU to the proper memory location.

The basic functions of the PIC are shown in the block diagram. The 8-bit request register is an edge clocked D-type register with internal 6 k Ω pullup resistors on the interrupt inputs ($\overline{IN0}$ thru $\overline{IN7}$). Note the inputs are active low. The interrupt register is loaded on the falling edge of the enable when the PIC is not selected.

The 1-of-8 priority encoder enables a vector corresponding to the stored interrupt with the highest priority and places it on the vector input port of a data selector. In addition an interrupt request signal $\overline{IRQ}$ is generated to signal the MPU that an interrupt has been detected. The mask location register overides and inhibits all interrupts with priority below the mask level. The mask can be thought of as a movable partition allowing responses to inputs equal to or greater than the mask value. For example if the stored mask level was 4, in-

FIGURE 2 – MC8507 TRUTH TABLE FOR M6800 MICROPROCESSOR SYSTEMS

Active Input		Output When Selected Z4	Z3	Z2	Z1	Equivalent to Bits 1-4 of B0, B1 . . . , B15 Hex Address	Address ROM Bytes Contain Address of:
Highest	$\overline{IN7}$	1	0	1	1	F F F 6 or 7	Priority 7 Routine
	$\overline{IN6}$	1	0	1	0	F F F 4 or 5	Priority 6 Routine
	$\overline{IN5}$	1	0	0	1	F F F 2 or 3	Priority 5 Routine
	$\overline{IN4}$	1	0	0	0	F F F 0 or 1	Priority 4 Routine
	$\overline{IN3}$	0	1	1	1	F F E E or F	Priority 3 Routine
	$\overline{IN2}$	0	1	1	0	F F E C or D	Priority 2 Routine
	$\overline{IN1}$	0	1	0	1	F F E A or B	Priority 1 Routine
Lowest	$\overline{IN0}$	0	1	0	0	F F E 8 or 9	Priority 0 Routine
	None	1	1	0	0	F F F 8 or 9	Default Routine*

*Default routine is the response to interrupt requests not generated by a prioritized input. The default routine may contain polling routines or may be an address in a loop for an interrupt driven system.

puts $\overline{IN0}$, $\overline{IN1}$, $\overline{IN2}$, and $\overline{IN3}$ would not generate an interrupt to the MPU system. The input request register is not affected by the mask, and if the mask is cleared (by loading it with zeros) any previously stored inputs will generate an $\overline{IRQ}$ signal.

The chip select and decode circuitry controls all internal functions of the PIC. The selected mode is defined as the logical AND function $\overline{A1} \cdot \overline{A2} \cdot A3 \cdot A4 \cdot \overline{CS0} \cdot CS1 \cdot R/W$. When the device is not in the selected mode the request register clock is enabled and the address inputs A_i pass directly through the data selector to the Z_i outputs. When the MPU responds to interrupt request $\overline{IRQ}$ and the PIC decodes the select address, the request register is inhibited and the data selector places the vector on the Z outputs. The address delay added to the MPU system is shown in Figure 3. This delay may be critical in some systems. A stretch signal, which indicates the selected mode, is provided for use with special MPU clock drivers to stretch the clock cycle when accessing slow ROMs. The $\overline{CS0}$ input has one less gating level than the remainder of the select decode logic. This allows an external NAND gate to be used for the full address decode without any increase in delay times.

The decode logic also controls the loading of the mask location register. This register will be loaded on the falling edge of the enable pulse when enabled by the logical AND function $\overline{CS0} \cdot CS1 \cdot \overline{R/W}$ (note 1). Contrary to normal read/write operations in MPU systems, the "data" written into the mask register are bits A1 thru A4 of the address bus (see Figure 5). This means that in the load mask mode the data on the data bus is a don't care. However in this mode the ROM will also be accessed and both the ROM and MPU will be driving the data bus. Therefore the read/write line should be used as an active high chip select or enable signal for ROM decoding.

Figure 4 show the typical operation flow diagram for the PIC in an M6800 system. The functional timing for this flow is as shown in the first part of the waveforms in Figure 1. The second half of Figure 1 shows the operation of the mask. Interrupts will be stored even if they are masked. When the mask is released the $\overline{IRQ}$ signal will then be generated.

The influence of the mask register on the priority encoder is shown in the truth table of Figure 5. The actual use of the mask register will vary with the system needs and the imaginative software programmer.

Note 1. Since during normal operation of the MPU the address lines and the R/W line can be in an indeterminate state, VMA should be logically ANDed with one of the chip select inputs of the PIC to prevent erroneous writes into the mask register.

FIGURE 3 – HIGH ROM ADDRESS DELAY ADDED TO M6800 SYSTEM

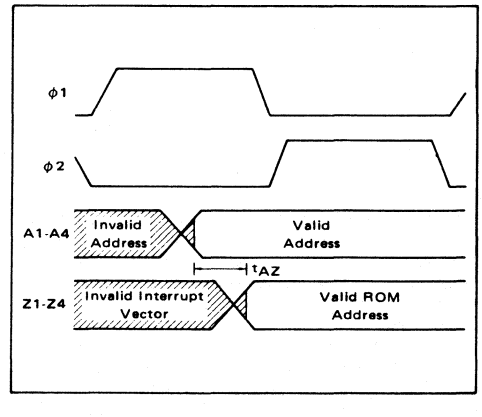


FIGURE 4 – BASIC FUNCTIONAL FLOW CHART

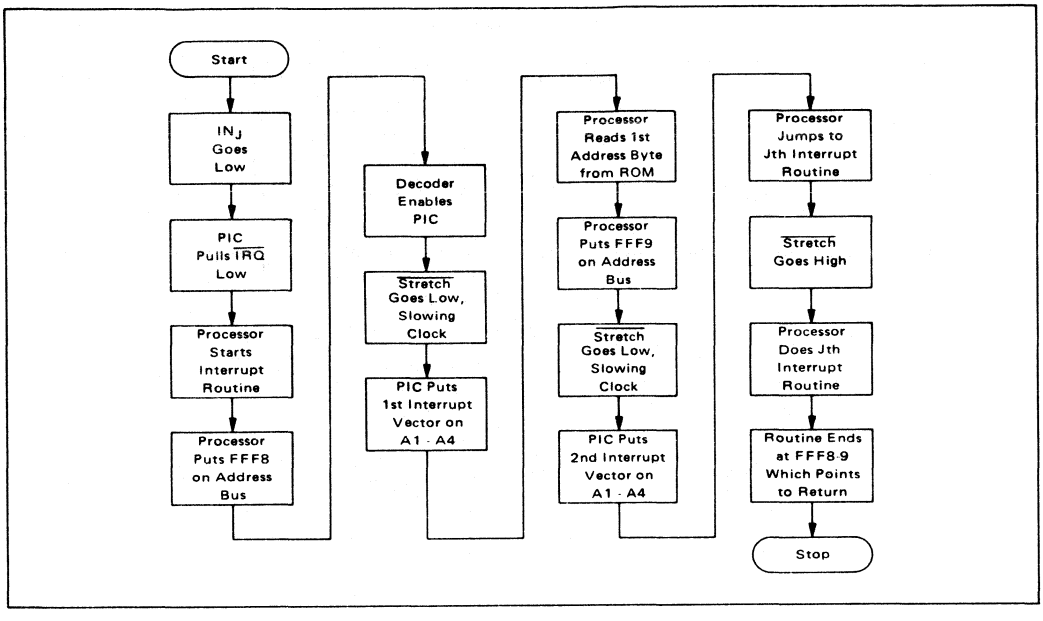
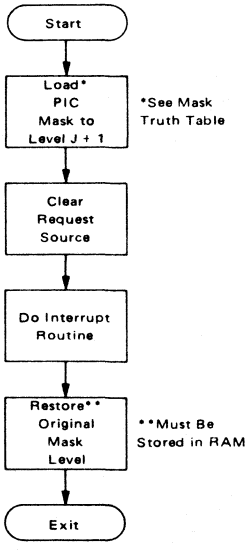


FIGURE 5 – MASK OPERATION

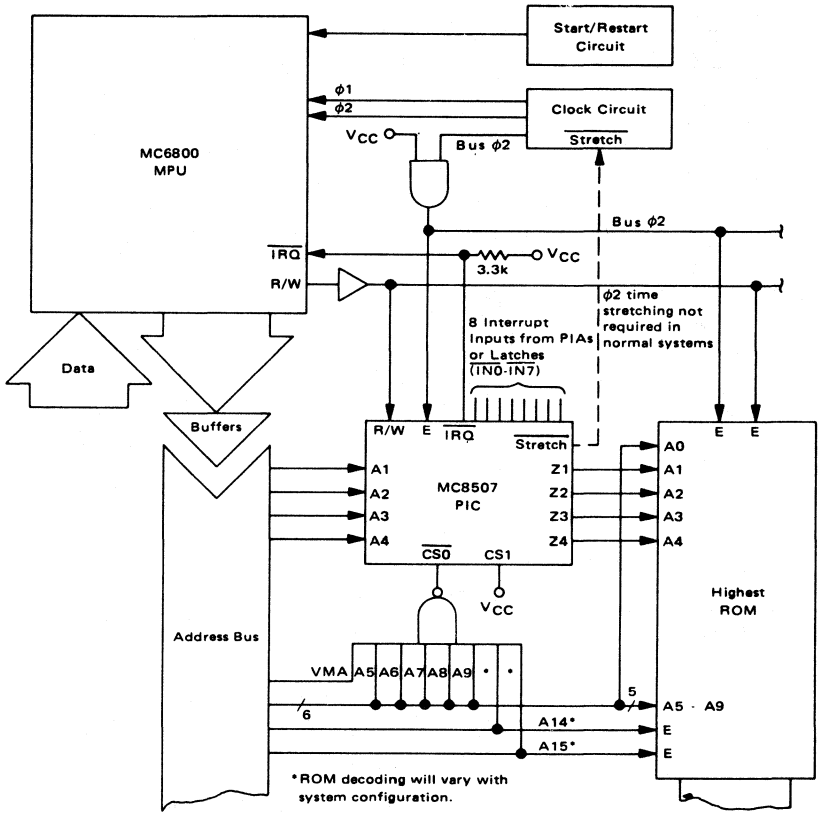
a – MASK FLOW CHART



b – MASK TRUTH TABLE

Mask Register Contents				Response to Priority Inputs 1 = Response to Input, 0 = No Response							
M4	M3	M2	M1	IN7	IN6	IN5	IN4	IN3	IN2	IN1	IN0
1	1	1	1	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	0	0	0	0	0	0	0
0	1	1	0	1	1	0	0	0	0	0	0
0	1	0	1	1	1	1	0	0	0	0	0
0	1	0	0	1	1	1	1	0	0	0	0
0	0	1	1	1	1	1	1	1	0	0	0
0	0	1	0	1	1	1	1	1	1	0	0
0	0	0	1	1	1	1	1	1	1	1	0
0	0	0	0	1	1	1	1	1	1	1	1

FIGURE 6 – TYPICAL M6800 SYSTEM CONFIGURATION



MOTOROLA

MC6840
MC68A40
MC68B40

PROGRAMMABLE TIMER MODULE (PTM)

The MC6840 is a programmable subsystem component of the M6800 family designed to provide variable system time intervals.

The MC6840 has three 16-bit binary counters, three corresponding control registers and a status register. These counters are under software control and may be used to cause system interrupts and/or generate output signals. The MC6840 may be utilized for such tasks as frequency measurements, event counting, interval measuring and similar tasks. The device may be used for square wave generation, gated delay signals, single pulses of controlled duration, and pulse width modulation as well as system interrupts.

- Operates from a Single 5 Volt Power Supply
- Fully TTL Compatible
- Single System Clock Required (Enable)
- Selectable Prescaler on Timer 3 Capable of 4 MHz for the M6840, 6 MHz for the MC68A40 and 8 MHz for the MC68B40
- Programmable Interrupts ($\overline{IRQ}$) Output to MPU
- Readable Down Counter Indicates Counts to Go to Time-Out
- Selectable Gating for Frequency or Pulse-Width Comparison
- $\overline{RESET}$ Input
- Three Asynchronous External Clock and Gate/Trigger Inputs Internally Synchronized
- Three Maskable Outputs

MOS

(N-CHANNEL, SILICON-GATE DEPLETION LOAD)

PROGRAMMABLE TIMER

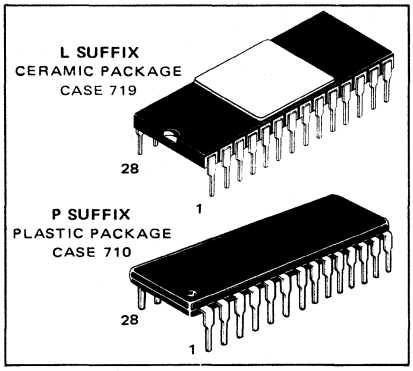
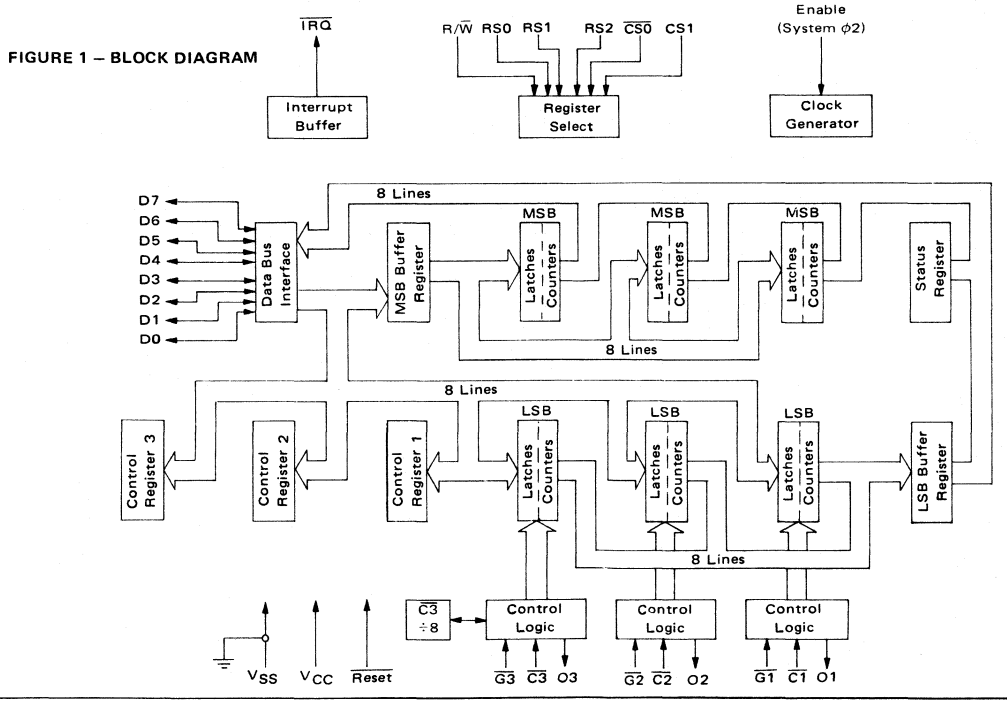


FIGURE 1 – BLOCK DIAGRAM



DS 9802
(Replaces ADI-442)

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	-0.3 to +7.0	Vdc
Input Voltage	V_{in}	-0.3 to +7.0	Vdc
Operating Temperature Range — T_L to T_H MC6840, MC68A40, MC68B40 MC6840C, MC68A40C MC6840BJCS, MC6840CJCS	T_A	T_L T_H 0 to +70 -40 to +85 -55 to +125	°C
Storage Temperature Range	T_{stg}	-55 to +150	°C
Thermal Resistance Plastic Package Ceramic Package	θ_{JA}	 115 60	°C/W

This device contains circitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid coplication of any voltage higher than maximum rated voltages to this high-impedance circuit.

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ±5%, V_{SS} = 0, T_A = T_L to T_H unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage		V_{IH}	V_{SS} +2.0	–	V_{CC}	Vdc
Input Low Voltage		V_{IL}	V_{SS} -0.3	–	V_{SS} +0.8	
Input Leakage Current (V_{in} = 0 to 5.25 V)		I_{in}	–	1.0	2.5	μAdc
Three-State (Off State) Input Current (V_{in} = 0.4 to 2.4 V)	D0–D7	I_{TSI}	–	2.0	10	μAdc
Output High Voltage (I_{load} = -205 μA) (I_{load} = -200 μA)	 D0–D7 Other Outputs	V_{OH}	 V_{SS} +2.4 V_{SS} +2.4	 – –	 – –	Vdc
Output Low Voltage (I_{load} = 1.6 mA) (I_{load} = 3.2 mA)	 D0–D7 O1–O3, $\overline{IRQ}$	V_{OL}	 – –	 – –	 V_{SS} +0.4 V_{SS} +0.4	Vdc
Output Leakage Current (Off State) (V_{OH} = 2.4 Vdc)	$\overline{IRQ}$	I_{LOH}	–	1.0	10	μAdc
Power Dissipation		P_D	–	330	550	mW
Input Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	 D0–D7 All others	C_{in}	 – –	 – –	 12.5 7.5	pF
Output Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	$\overline{IRQ}$ O1, O2, O3	C_{out}	– –	– –	5.0 10	pF

BUS TIMING CHARACTERISTICS

Characteristic	Symbol	MC6840 Min	MC6840 Max	MC68A40 Min	MC68A40 Max	MC68B40 Min	MC68B40 Max	Unit
READ (See Figure 2)								
Enable Cycle Time	t_{cycE}	1.0	10	0.666	10	0.5	10	μs
Enable Pulse Width, High	PW_{EH}	0.45	4.5	0.280	4.5	0.22	4.5	μs
Enable Pulse Width, Low	PW_{EL}	0.43	–	0.280	–	0.21	–	μs
Setup Time, Address and R/$\overline{W}$ valid to enable positive transition	t_{AS}	160	–	140	–	70	–	ns
Data Delay Time	t_{DDR}	–	320	–	220	–	180	ns
Data Hold Time	t_H	10	–	10	–	10	–	ns
Address Hold Time	t_{AH}	10	–	10	–	10	–	ns
Rise and Fall Time for Enable input	t_{Er}, t_{Ef}	–	25	–	25	–	25	ns
WRITE (See Figure 3)								
Enable Cycle Time	t_{cycE}	1.0	10	0.666	10	0.50	10	μs
Enable Pulse Width, High	PW_{EH}	0.45	4.5	0.280	4.5	0.22	4.5	μs
Enable Pulse Width, Low	PW_{EL}	0.43	–	0.280	–	0.21	–	μs
Setup Time, Address and R/$\overline{W}$ valid to enable positive transition	t_{AS}	160	–	140	–	70	–	ns
Data Setup Time	t_{DSW}	195	–	80	–	60	–	ns
Data Hold Time	t_H	10	–	10	–	10	–	ns
Address Hold Time	t_{AH}	10	–	10	–	10	–	ns
Rise and Fall Time for Enable input	t_{Er}, t_{Ef}	–	25	–	25	–	25	ns

AC OPERATING CHARACTERISTICS (See Figures 4–8)

Characteristic	Symbol	MC6840		MC68A40		MC68B40		Unit
		Min	Max	Min	Max	Min	Max	
Input Rise and Fall Times (Figures 4 and 5) $\overline{C}$, $\overline{G}$ and $\overline{Reset}$	t_r, t_f	–	1.0*	–	0.666*	–	0.500*	μs
Input Pulse Width Low (Figure 4) (Asynchronous Mode) $\overline{C}$, $\overline{G}$ and $\overline{Reset}$	PW_L	$t_{cycE}+t_{su}+t_{hd}$	–	$t_{cycE}+t_{su}+t_{hd}$	–	$t_{cycE}+t_{su}+t_{hd}$	–	ns
Input Pulse Width High (Figure 5) (Asynchronous Mode) $\overline{C}$, $\overline{G}$	PW_H	$t_{cycE}+t_{su}+t_{hd}$	–	$t_{cycE}+t_{su}+t_{hd}$	–	$t_{cycE}+t_{su}+t_{hd}$	–	ns
Input Setup Time (Figure 6) (Synchronous Mode) $\overline{C}$, $\overline{G}$ and $\overline{Reset}$, $\overline{C3}$ (÷8 Prescaler Mode only)	t_{su}	200	–	120	–	75	–	ns
Input Hold Time (Figure 6) (Synchronous Mode) $\overline{C}$, $\overline{G}$ and $\overline{Reset}$ $\overline{C3}$ (÷8 Prescaler Mode only)	t_{hd}	50	–	50	–	50	–	ns
Input Pulse Width (Asynchronous Mode) $\overline{C3}$ (÷8 Prescaler Mode only)	PW_L, PW_H	125	–	84	–	62.5	–	ns
Output Delay, O1–O3 (Figure 7)								
(V_{OH} = 2.4 V, Load B) TTL	t_{co}	–	700	–	460	–	340	ns
(V_{OH} = 2.4 V, Load D) MOS	t_{cm}	–	450	–	450	–	340	ns
(V_{OH} = 0.7 V_{DD}, Load D) CMOS	t_{cmos}	–	2.0	–	1.35	–	1.0	μs
Interrupt Release Time	t_{IR}	–	1.2	–	0.9	–	0.7	μs

*t_r and $t_f \leq t_{cycE}$

FIGURE 2 – BUS READ TIMING CHARACTERISTICS
(Read Information from PTM)

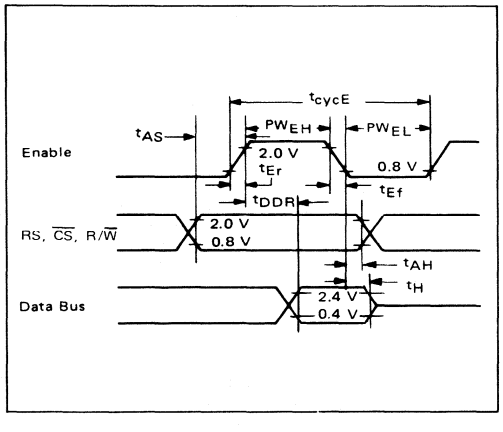


FIGURE 3 – BUS WRITE TIMING CHARACTERISTICS
(Write Information into PTM)

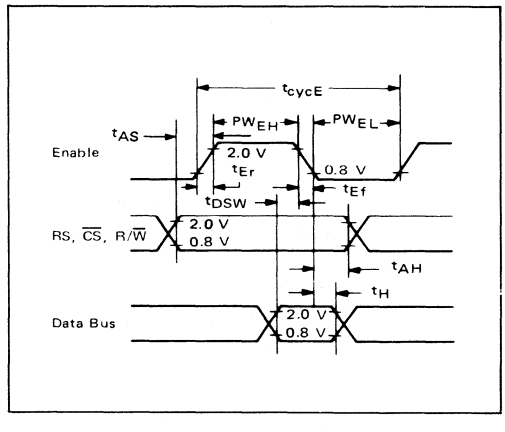


FIGURE 4 – INPUT PULSE WIDTH LOW

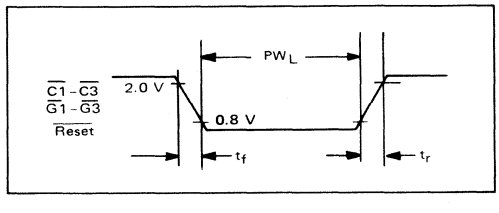


FIGURE 5 – INPUT PULSE WIDTH HIGH

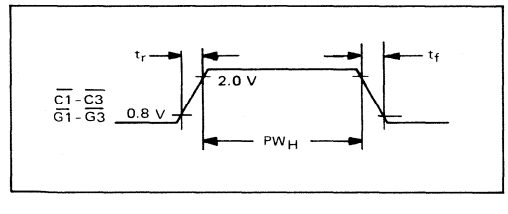


FIGURE 6 – INPUT SETUP AND HOLD TIMES

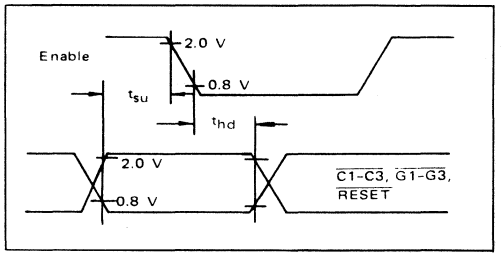


FIGURE 7 – OUTPUT DELAY

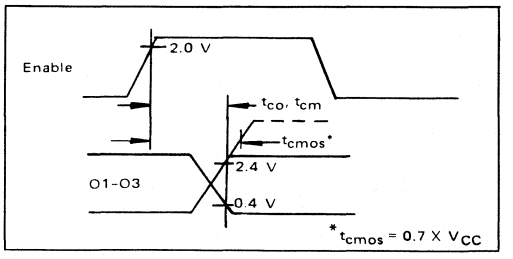


FIGURE 8 – $\overline{IRQ}$ RELEASE TIME

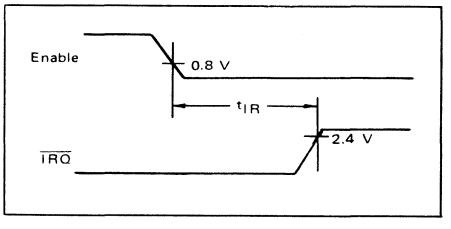
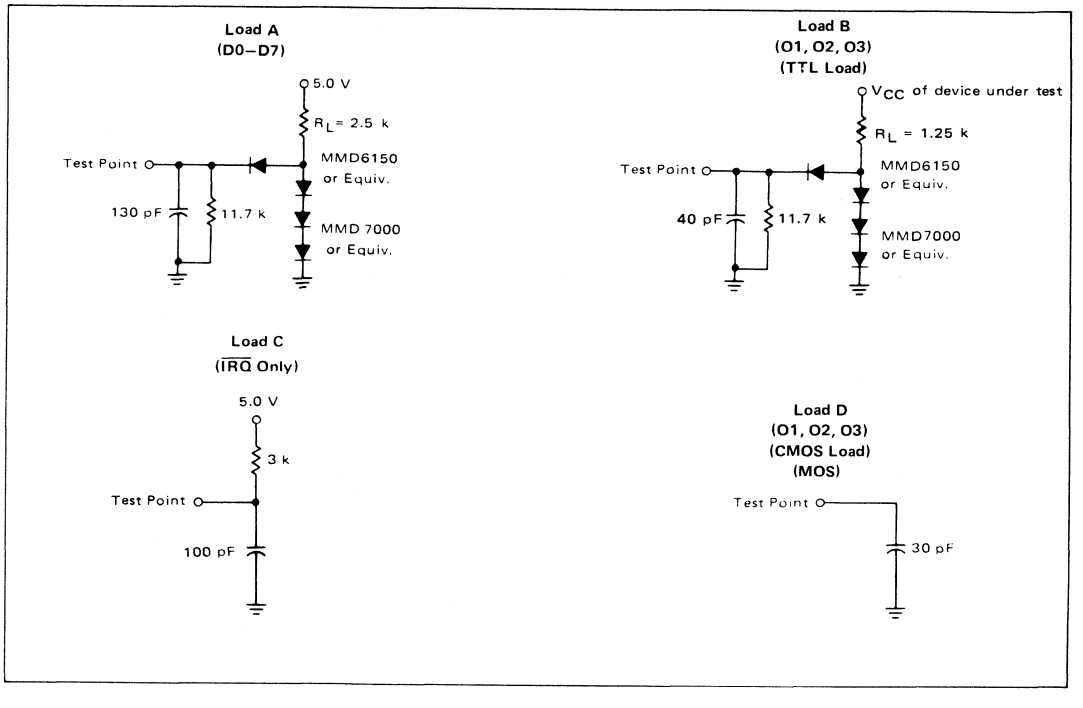


FIGURE 9 – BUS TIMING TEST LOADS



DEVICE OPERATION

The three timers in the MC6840 may be independently programmed to operate in modes which fit a wide variety of applications. The device is fully bus compatible with M6800 systems and is accessed by load and store operations from the MPU in much the same manner as a memory device. In a typical application, a Timer will be loaded by first storing two bytes of data into an associated Counter Latch. This data is then transferred into the counter via a Counter Initialization cycle. The counter decrements on each subsequent clock period which may be an external clock or Enable (System $\phi 2$) until one of several predetermined conditions causes it to halt or recycle. The timers are thus programmable, cyclic in nature, controllable by external inputs or the MPU program, and accessible by the MPU at any time.

BUS INTERFACE

The Programmable Timer Module (PTM) interfaces to the M6800 Bus with an eight-bit bidirectional data bus, two Chip Select lines, a Read/$\overline{\text{Write}}$ line, an Enable (System $\phi 2$) line, an $\overline{\text{Interrupt Request}}$ line, an external $\overline{\text{Reset}}$ line, and three Register Select lines. These signals, in conjunction with the MC6800 VMA output, permit the MPU to control the PTM. VMA should be utilized in conjunction with an MPU address line into a Chip Select of the PTM.

Bidirectional Data (D0–D7) – The bidirectional data lines (D0–D7) allow the transfer of data between the MPU and PTM. The data bus output drivers are three-state devices which remain in the high-impedance (off) state except when the MPU performs a PTM read operation (Read/$\overline{\text{Write}}$ and Enable lines high and PTM Chip Selects activated).

Chip Select ($\overline{\text{CS0}}$, CS1) – These two signals are used to activate the Data Bus interface and allow transfer of data from the PTM. With $\overline{\text{CS0}}$ = 0 and CS1 = 1, the device is selected and data transfer will occur.

Read/$\overline{\text{Write}}$ (R/$\overline{\text{W}}$) – This signal is generated by the MPU to control the direction of data transfer on the Data Bus. With the PTM selected, a low state on the PTM R/$\overline{\text{W}}$ line enables the input buffers and data is transferred from the MPU to the PTM on the trailing edge of the Enable (System $\phi 2$) signal. Alternately, (under the same conditions) R/$\overline{\text{W}}$ = 1 and Enable high allows data in the PTM to be read by the MPU.

Enable (System $\phi 2$) – This signal synchronizes data transfer between the MPU and the PTM. It also performs an equivalent synchronization function on the external clock, reset, and gate inputs of the PTM.

$\overline{\text{Interrupt Request}}$ ($\overline{\text{IRQ}}$) – The active low $\overline{\text{Interrupt Request}}$ signal is normally tied directly (or through priority interrupt circuitry) to the $\overline{\text{IRQ}}$ input of the MPU. This is an "open drain" output (no load device on the chip) which permits other similar interrupt request lines to be tied together in a wire-OR configuration.

The $\overline{\text{IRQ}}$ line is activated if, and only if, the Composite Interrupt Flag (Bit 7 of the Internal Status Register) is asserted. The conditions under which the $\overline{\text{IRQ}}$ line is activated are discussed in conjunction with the Status Register.

External $\overline{\text{Reset}}$ – A low level at this input is clocked into the PTM by the Enable (System $\phi 2$) input. Two Enable pulses are required to synchronize and process the signal. The PTM then recognizes the active "low" or inactive "high" on the third Enable pulse. If the $\overline{\text{Reset}}$ signal is asynchronous, an additional Enable period is required if setup times are not met. The $\overline{\text{Reset}}$ input must be stable High/Low for the minimum time stated in the AC Operating Characteristics.

Recognition of a low level at this input by the PTM causes the following action to occur:

a. All counter latches are preset to their maximal count values.
b. All Control Register bits are cleared with the exception of CR10 (internal reset bit) which is set.
c. All counters are preset to the contents of the latches.
d. All counter outputs are reset and all counter clocks are disabled.
e. All Status Register bits (interrupt flags) are cleared.

Register Select Lines (RS0, RS1, RS2) – These inputs are used in conjunction with the R/W line to select the internal registers, counters and latches as shown in Table 1.

It has been previously stated that the PTM is accessed via MPU Load and Store operations in much the same manner as a memory device. The instructions available with the M6800 family of MPUs which perform operations directly on memory should not be used when the PTM is accessed. These instructions actually fetch a byte from memory, perform an operation, then restore it to the same address location. Since the PTM used the R/$\overline{\text{W}}$ line as an additional register select input, the modified data may not be restored to the same register if these instructions are used.

TABLE 1 – REGISTER SELECTION

Register Select Inputs			Operations	
RS2	RS1	RS0	R/$\overline{W}$ = 0	R/$\overline{W}$ = 1
0	0	0	CR20 = 0 Write Control Register #3 CR20 = 1 Write Control Register #1	No Operation
0	0	1	Write Control Register #2	Read Status Register
0	1	0	Write MSB Buffer Register	Read Timer #1 Counter
0	1	1	Write Timer #1 Latches	Read LSB Buffer Register
1	0	0	Write MSB Buffer Register	Read Timer #2 Counter
1	0	1	Write Timer #2 Latches	Read LSB Buffer Register
1	1	0	Write MSB Buffer Register	Read Timer #3 Counter
1	1	1	Write Timer #3 Latches	Read LSB Buffer Register

CONTROL REGISTER

Three Write-Only registers in the MC6840 are used to modify timer operation to suit a variety of applications. Control Register #2 has a unique address space (RS0 = 1, RS1 = 0, RS2 = 0) and therefore may be written into at any time. The remaining Control Registers (#1 and #3) share the Address Space selected by a logic zero on all Register Select inputs. The least significant bit of Control Register #2 (CR20) is used as an additional addressing bit for Control Registers #1 and #3. Thus, with all Register Selects and R/$\overline{W}$ inputs at logic zero, Control Register #1 will be written into if CR20 is a logic one. Under the same conditions, Control Register #3 will be written into if CR20 is a logic zero. Control Register #3 can also be written into after a $\overline{Reset}$ low condition has occurred, since all control register bits (except CR10) are cleared. Therefore, one may write in the sequence CR3, CR2, CR1.

The least significant bit of Control Register #1 is used as an Internal Reset bit. When this bit is a logic zero, all timers are allowed to operate in the modes prescribed by the remaining bits of the control registers. Writing a "one" into CR10 causes all counters to be preset with the contents of the corresponding counter latches, all counter clocks to be disabled, and the timer outputs and interrupt flags (Status Register) to be reset. Counter Latches and Control Registers are undisturbed by an Internal Reset and may be written into regardless of the state of CR10.

The least significant bit of Control Register #3 is used as a selector for a ÷8 prescaler which is available with Timer #3 only. The prescaler, if selected, is effectively placed between the clock input circuitry and the input to Counter #3. It can therefore be used with either the internal clock (Enable) or an external clock source.

The functions depicted in the foregoing discussions are tabulated on the first row in Table 2 for ease of reference.

TABLE 2 – CONTROL REGISTER BITS

CR10 Internal Reset Bit	CR20 Control Register Address Bit	CR30 Timer #3 Clock Control
0 All timers allowed to operate 1 All timers held in preset state	0 CR#3 may be written 1 CR#1 may be written	0 T3 $\overline{Clock}$ is not prescaled 1 T3 $\overline{Clock}$ is prescaled by ÷8
CRX1*	Timer #X $\overline{Clock}$ Source	
0	TX uses external clock source on $\overline{CX}$ input	
1	TX uses Enable clock	
CRX2	Timer #X Counting Mode Control	
0	TX configured for normal (16-bit) counting mode	
1	TX configured for dual 8-bit counting mode	
CRX3 CRX4 CRX5	Timer #X Counter Mode and Interrupt Control (See Table 3)	
CRX6	Timer #X Interrupt Enable	
0	Interrupt Flag masked on $\overline{IRQ}$	
1	Interrupt Flag enabled to $\overline{IRQ}$	
CRX7	Timer #X Counter Output Enable	
0	TX Output masked on output OX	
1	TX Output enabled on output OX	

*Control Register for Timer 1, 2, or 3, Bit 1.

Control Register Bits CR10, CR20, and CR30 are unique in that each selects a different function. The remaining bits (1 through 7) of each Control Register select common functions, with a particular Control Register affecting only its corresponding timer. For example, Bit 1 of Control Register #1 (CR11) selects whether an internal or external clock source is to be used with Timer #1. Similarly, CR21 selects the clock source for Timer #2, and CR31 performs this function for Timer #3. The function of each bit of Control Register "X" can therefore be defined as shown in the remaining section of Table 2.

Control Register Bit 2 selects whether the binary information contained in the Counter Latches (and subsequently loaded into the Counter) is to be treated as a single 16-bit word or two 8-bit bytes. In the single 16-bit Counter Mode (CRX2 = 0) the counter will decrement to zero after N+1 enabled ($\overline{G}$ = 0) clock periods, where N is defined as the 16-bit number in the Counter Latches. With CRX2 = 1, a similar Time Out will occur after (L+1)·(M+1) enabled clock periods, where L and M, respectively, refer to the LSB and MSB bytes in the Counter Latches.

Control Register Bits 3, 4, and 5 are explained in detail in the Timer Operating Mode section. Bit 6 is an interrupt mask bit which will be explained more fully in conjunction with the Status Register, and bit 7 is used to enable the corresponding Timer Output. A summary of the control register programming modes is shown in Table 3.

STATUS REGISTER/INTERRUPT FLAGS

The MC6840 has an internal Read-Only Status Register which contains four Interrupt Flags. (The remaining four bits of the register are not used, and default to zeros when being read.) Bits 0, 1, and 2 are assigned to Timers 1, 2, and 3, respectively, as individual flag bits, while Bit 7 is a Composite Interrupt Flag. This flag bit will be asserted if any of the individual flag bits is set while Bit 6 of the corresponding Control Register is at a logic one. The conditions for asserting the Composite Interrupt Flag bit can therefore be expressed as:

$$INT = I1 \cdot CR16 + I2 \cdot CR26 + I3 \cdot CR36$$

where INT = Composite Interrupt Flag (Bit 7)
I1 = Timer #1 Interrupt Flag (Bit 0)
I2 = Timer #2 Interrupt Flag (Bit 1)
I3 = Timer #3 Interrupt Flag (Bit 2)

An interrupt flag is cleared by a Timer Reset condition, i.e., External $\overline{Reset}$ = 0 or Internal Reset Bit (CR10) = 1. It will also be cleared by a Read Timer Counter Command provided that the Status Register has previously been read while the interrupt flag was set. This condition on the Read Status Register–Read Timer Counter (RS–RT) sequence is designed to prevent missing interrupts which might occur after the status register is read, but prior to reading the Timer Counter.

An Individual Interrupt Flag is also cleared by a Write Timer Latches (W) command or a Counter Initialization (CI) sequence, provided that W or CI affects the Timer corresponding to the individual Interrupt Flag.

COUNTER LATCH INITIALIZATION

Each of the three independent timers consists of a 16-bit addressable counter and 16 bits of addressable latches. The counters are preset to the binary numbers stored in the latches. Counter initialization results in the transfer of the latch contents to the counter. See notes in Table 4 regarding the binary number N, L, or M placed into the Latches and their relationship to the output waveforms and counter Time-Outs.

Since the PTM data bus is 8-bits wide and the counters are 16-bits wide, a temporary register (MSB Buffer Register) is provided. This "write only" register is for the Most Significant Byte of the desired latch data. Three addresses are provided for the MSB Buffer Register (as indicated in Table 1), but they all lead to the same Buffer. Data from the MSB Buffer will automatically be transferred into the Most Significant Byte of Timer #X when a Write Timer #X Latches Command is performed. So it can be seen that the MC6840 has been designed to allow transfer of two bytes of data into the counter latches provided that the MSB is transferred first.

In many applications, the source of the data will be an MC6800 MPU. It should be noted that the 16-bit store operations of the M6800 family microprocessors (STS and STX) transfer data in the order required by the PTM. A Store Index Register Instruction, for example, results in the MSB of the X register being transferred to the selected address, then the LSB of the X register being written into the next higher location. Thus, either the index register or stack pointer may be transfered directly into a selected counter latch with a single instruction.

A logic zero at the $\overline{Reset}$ input also initializes the counter latches. In this case, all latches will assume a maximum count of $65{,}536_{10}$. It is important to note that an Internal Reset (Bit zero of Control Register 1 Set) has no effect on the counter latches.

COUNTER INITIALIZATION

Counter Initialization is defined as the transfer of data from the latches to the counter with subsequent clearing of the Individual Interrupt Flag associated with the counter. Counter Initialization always occurs when a reset condition ($\overline{Reset}$ = 0 or CR10 = 1) is recognized. It can also occur–depending on Timer Mode–with a Write Timer Latches command or recognition of a negative transition of the $\overline{Gate}$ input.

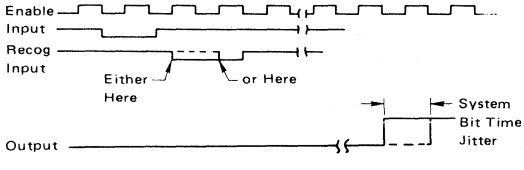
Counter recycling or re-initialization occurs when a negative transition of the clock input is recognized after the counter has reached an all-zero state. In this case, data is transferred from the Latches to the Counter.

ASYNCHRONOUS INPUT/OUTPUT LINES

Each of the three timers within the PTM has external clock and gate inputs as well as a counter output line. The inputs are high impedance, TTL compatible lines and outputs are capable of driving two standard TTL loads.

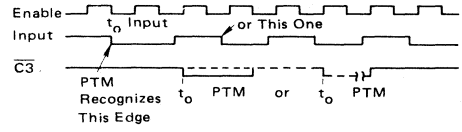
Clock Inputs ($\overline{C1}$, $\overline{C2}$, and $\overline{C3}$) – Input pins $\overline{C1}$, $\overline{C2}$, and $\overline{C3}$ will accept asynchronous TTL voltage level signals to decrement Timers 1, 2, and 3, respectively. The high and low levels of the external clocks must each be stable for at least one system clock period plus the sum of the setup and hold times for the inputs. The asynchronous clock rate can vary from dc to the limit imposed by Enable (System $\phi 2$) Setup, and Hold time.

The external clock inputs are clocked in by Enable (System $\phi 2$) pulses. Three Enable periods are used to synchronize and process the external clock. The fourth Enable pulse decrements the internal counter. This does not affect the input frequency, it merely creates a delay between a clock input transition and internal recognition of that transition by the PTM. All references to $\overline{C}$ inputs in this document relate to internal recognition of the input transition. Note that a clock high or low level which does not meet setup and hold time specifications may require an additional Enable pulse for recognition. When observing recurring events, a lack of synchronization will result in "jitter" being observed on the output of the PTM when using asynchronous clocks and gate input signals. There are two types of jitter. "System jitter" is the result of the input signals being out of synchronization with the Enable (System $\phi 2$), permitting signals with marginal setup and hold time to be recognized by either the bit time nearest the input transition or the subsequent bit time.



"Input jitter" can be as great as the time between input signal negative going transitions plus the system jitter, if the first transition is recognized during one system cycle, and not recognized the next cycle, or vice versa.

External clock input $\overline{C3}$ represents a special case when Timer #3 is programmed to utilize its optional ÷8 prescaler



mode. The maximum input frequency and allowable duty cycles for this case are specified under the AC Operating Characteristics. The output of the ÷8 prescaler is treated in the same manner as the previously discussed clock inputs. That is, it is clocked into the counter by Enable pulses, is recognized on the fourth Enable pulse (provided setup and hold time requirements are met), and must produce an output pulse at least as wide as the sum of an Enable period, setup, and hold times.

Gate Inputs ($\overline{G1}$, $\overline{G2}$, $\overline{G3}$) – Input pins $\overline{G1}$, $\overline{G2}$, and $\overline{G3}$ accept asynchronous TTL-compatible signals which are used as triggers or clock gating functions to Timers 1, 2, and 3, respectively. The gating inputs are clocked into the PTM by the Enable (System $\phi 2$) signal in the same manner as the previously discussed clock inputs. That is, a $\overline{Gate}$ transition is recognized by the PTM on the fourth Enable pulse (provided setup and hold time requirements are met), and the high or low levels of the $\overline{Gate}$ input must be stable for at least one system clock period plus the sum of setup and hold times. All references to $\overline{G}$ transition in this document relate to internal recognition of the input transition.

The $\overline{Gate}$ inputs of all timers directly affect the internal 16-bit counter. The operation of $\overline{G3}$ is therefore independent of the ÷8 prescaler selection.

Timer Outputs (O1, O2, O3) – Timer outputs O1, O2, and O3 are capable of driving up to two TTL loads and produce a defined output waveform for either Continuous or Single-Shot Timer modes. Output waveform definition is accomplished by selecting either Single 16-bit or Dual 8-bit operating modes. The single 16-bit mode will produce a square-wave output in the continous timer mode and will produce a single pulse in the Single-Shot Timer mode. The Dual 8-bit mode will produce a variable duty cycle pulse in both the continuous and single shot Timer modes. One bit of each Control Register (CRX7) is used to enable the corresponding output. If this bit is cleared, the output will remain low (V_{OL}) regardless of the operating mode.

The Continuous and Single-Shot Timer Modes are the only ones for which output response is defined in this data sheet. Refer to the Programmable Timer Fundamentals and Applications manual for a discussion of the output signals in other modes. Signals appear at the outputs (unless CRX7 = 0) during Frequency and Pulse Width comparison modes, but the actual waveform is not predictable in typical applications.

TABLE 3 -- CONTROL REGISTER PROGRAMMING

7	6	5	4	3	2	1	0
X	X	X	X	X	X	X	↕

	Register 1	Register 2	Register 3
0	All Timers Operate	Reg #3 May Be Written	T3 Clk ÷ 1
1	All Timers Preset	Reg #1 May Be Written	T3 Clk ÷ 8

7	6	5	4	3	2	1	0
X	X	X	X	X	X	↕	X

0	External Clock ($\overline{CX}$ Input)
1	Internal Clock (Enable)

7	6	5	4	3	2	1	0
X	X	X	X	X	↕	X	X

0	Normal (16-Bit) Count Mode
1	Dual 8-Bit Count Mode

7	6	5	4	3	2	1	0
X	X	0	0	0	X	X	X

Continuous Operating Mode: $\overline{Gate}$ ↓ or Write to Latches or Reset Causes Counter Initialization

7	6	5	4	3	2	1	0
X	X	0	0	1	X	X	X

Frequency Comparison Mode: Interrupt If $\overline{Gate}$  is < Counter Time Out

7	6	5	4	3	2	1	0
X	X	0	1	0	X	X	X

Continuous Operating Mode: $\overline{Gate}$ ↓ or Reset Causes Counter Intialization

7	6	5	4	3	2	1	0
X	X	0	1	1	X	X	X

Pulse Width Comparison Mode: Interrupt If $\overline{Gate}$  is < Counter Time Out

7	6	5	4	3	2	1	0
1	X	1	0	0	X	X	X

Single Shot Mode: $\overline{Gate}$ ↓ or Write to Latches or Reset Causes Counter Initialization

7	6	5	4	3	2	1	0
X	X	1	0	1	X	X	X

Frequency Comparison Mode: Interrupt If $\overline{Gate}$  is > Counter Time Out

7	6	5	4	3	2	1	0
1	X	1	1	0	X	X	X

Single Shot Mode: $\overline{Gate}$ ↓ or Reset Causes Counter Initialization

7	6	5	4	3	2	1	0
X	X	1	1	1	X	X	X

Pulse Width Comparison Mode: Interrupt If $\overline{Gate}$ is > Counter Time Out

7	6	5	4	3	2	1	0
X	↕	X	X	X	X	X	X

0	Interrupt Flag Masked ($\overline{IRQ}$)
1	Interrupt Flag Enabled ($\overline{IRQ}$)

7	6	5	4	3	2	1	0
↕	X	X	X	X	X	X	X

0	Timer Output Masked
1	Timer Output Enable

NOTE: Reset is Hardware or Software Reset ($\overline{Reset}$ = 0 or CR10 = 1).

TIMER OPERATING MODES

The MC6840 has been designed to operate effectively in a wide variety of applications. This is accomplished by using three bits of each control register(CRX3, CRX4, and CRX5) to defined different operating modes of the Timers. These modes are outlined in Table 4.

TABLE 4 – OPERATING MODES

Control Register			
CRX3	CRX4	CRX5	Timer Operating Mode
0	*	0	Continuous
0	*	1	Single-Shot
1	0	*	Frequency Comparison
1	1	*	Pulse Width Comparison

*Defines Additional Timer Functions

In addition to the four timer modes in Table 4, the remaining control register bit is used to modify counter initialization and enabling or interrupt conditions.

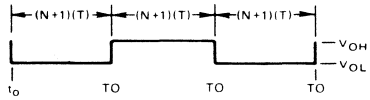
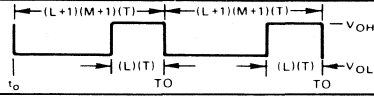
Continuous Operating Mode (Table 5) – Any of the timers in the PTM may be programmed to operate in a continuous mode by writing zeroes into bits 3 and 5 of the corresponding control register. Assuming that the timer output is enabled (CRX7 = 1), either a square wave or a variable duty cycle waveform will be generated at the Timer Output, OX. The type of output is selected via Control Register Bit 2.

Either a Timer Reset (CR10 = 1 or External $\overline{\text{Reset}}$ = 0) condition or internal recognition of a negative transition of the $\overline{\text{Gate}}$ input results in Counter Initialization. A Write Timer Latches command can be selected as a Counter Initialization signal by clearing CRX4.

The counter is enabled by an absence of a Timer Reset condition and a logic zero at the $\overline{\text{Gate}}$ input. The counter will then decrement on the first clock signal recognized during or after the counter initialization cycle. It continues to decrement on each clock signal so long as $\overline{\text{G}}$ remains low and no reset condition exists. A Counter Time Out (the first clock after all counter bits = 0) results in the Individual Interrupt Flag being set and re-initialization of the counter.

A special condition exists for the dual 8-bit mode (CRX2 = 1) if L = 0. In this case, the counter will revert to a mode similar to the single 16-bit mode, except Time Out occurs after M+1 clock pulses. The output, if enabled, goes low during the Counter Initialization cycle and reverses state at each Time Out. The counter remains cyclical (is re-initialized at each Time Out) and the

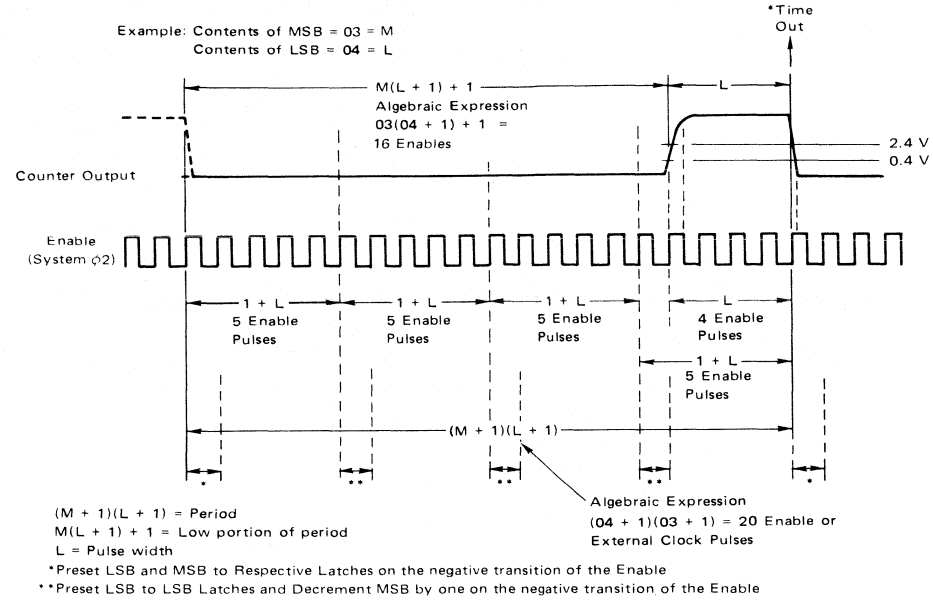
TABLE 5 – CONTINUOUS OPERATING MODES

CONTINUOUS MODE (CRX3 = 0, CRX5 = 0)			
Control Register		Initialization/Output Waveforms	
CRX2	CRX4	Counter Initialization	*Timer Output (OX) (CRX7 = 1)
0	0	$\overline{\text{G}}\downarrow$+W+R	
0	1	$\overline{\text{G}}\downarrow$+R	
1	0	$\overline{\text{G}}\downarrow$+W+R	
1	1	$\overline{\text{G}}\downarrow$+R	

$\overline{\text{G}}\downarrow$ = Negative transition of $\overline{\text{Gate}}$ input.
W = Write Timer Latches Command.
R = Timer Reset (CR10 = 1 or External $\overline{\text{Reset}}$ = 0)
N = 16-Bit Number in Counter Latch.
L = 8-Bit Number in LSB Counter Latch.
M = 8-Bit Number in MSB Counter Latch.
T = Clock Input Negative Transitions to Counter.
t_o = Counter Initialization Cycle.
TO = Counter Time Out (All Zero Condition).

*All time intervals shown above assume the the $\overline{\text{Gate}}$ ($\overline{\text{G}}$) and $\overline{\text{Clock}}$ ($\overline{\text{C}}$) signals are synchronized to Enable (System ϕ2) with the specified setup and hold time requirements.

FIGURE 10 – TIMER OUTPUT WAVEFORM EXAMPLE
(Continuous Dual 8-Bit Mode using Internal Enable)



Individual Interrupt Flag is set when Time Out occurs. If M = L = 0, the internal counters do not change, but the output toggles at a rate of 1/2 the clock frequency.

In the dual 8-bit mode (CRX2 = 1) [Refer to the example in Figure 10] the MSB decrements once for every full countdown of the LSB+1. When the LSB = 0, the MSB is unchanged; on the next clock pulse the LSB is reset to the count in the LSB Latches and the MSB is decremented by 1 (one). The output, if enabled, remains low during and after initialization and will remain low until the counter MSB is all zeroes. The output will go high at the beginning of the next clock pulse. The output remains high until both the LSB and MSB of the counter are all zeroes. At the beginning of the next clock pulse the defined Time Out (TO) will occur and the output will go low. In the normal 16-bit mode the period of the output of the example in Figure 10 would span 1546 clock pulses as opposed to the 20 clock pulses using the Dual 8-bit mode.

The discussion of the Continuous Mode has assumed that the application requires an output signal. It should be noted that the Timer operates in the same manner with the output disabled (CRX7 = 0). A Read Timer Counter command is valid regardless of the state of CRX7.

Single-Shot Timer Mode – This mode is identical to the Continuous Mode with three exceptions. The first of these is obvious from the name--the output returns to a low level after the initial Time Out and remains low until another Counter Initialization cycle occurs. The waveforms available are shown in Table 6.

As indicated in Table 6, the internal counting mechanism remains cyclical in the Single-Shot Mode. Each Time Out of the counter results in the setting of an Individual Interrupt Flag and re-initialization of the counter.

The second major difference between the Single-Shot and Continuous modes is that the internal counter enable is not dependent on the $\overline{\text{Gate}}$ input level remaining in the low state for the Single-Shot mode.

Another special condition is introduced in the Single-Shot mode. If L = M = 0 (Dual 8-bit) or N = 0 (Single 16-bit), the output goes low on the first clock received during or after Counter Initialization. The output remains low until the Operating Mode is changed or nonzero data is written into the Counter Latches. Time Outs continue to occur at the end of each clock period.

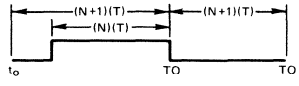
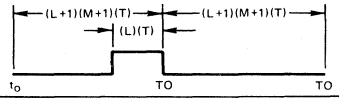
The three differences between Single-Shot and Continuous Timer Modes can be summarized as attributes of the Single-Shot mode:

1. Output is enabled for only one pulse until it is reinitialized.
2. Counter Enable is independent of $\overline{Gate}$.
3. L = M = 0 or N = 0 disables output.

Aside from these differences, the two modes are identical.

Frequency Comparison or Period Measurement Mode (CRX3 = 1, CRX4 = 0) -The Frequency Comparison Mode with CRX5 = 1 is straightforward. If Time Out occurs prior to the first negative transition of the $\overline{Gate}$ input after a Counter Initialization cycle, an Individual Interrupt Flag is set. The counter is disabled, and a Counter Initialization cycle cannot begin until the interrupt flag is cleared and a negative transition on $\overline{G}$ is detected.

TABLE 6 – SINGLE-SHOT OPERATING MODES

SINGLE-SHOT MODE (CRX3 = 0, CRX7 = 1, CRX5 = 1)			
Control Register		**Initialization/Output Waveforms**	
CRX2	**CRX4**	**Counter Initialization**	**Timer Output (OX)**
0	0	$\overline{G}\downarrow$+W+R	
0	1	$\overline{G}\downarrow$+R	
1	0	$\overline{G}\downarrow$+W+R	
1	1	$\overline{G}\downarrow$+R	

Symbols are as defined in Table 5.

Time Interval Modes – The Time Interval Modes are provided for those applications which require more flexibility of interrupt generation and Counter Initialization. Individual Interrupt Flags are set in these modes as a function of both Counter Time Out and transitions of the $\overline{Gate}$ input. Counter Initialization is also affected by Interrupt Flag status.

Refer to the Programmable Timer Fundamentals and Applications manual for a discussion of the output signals in these modes. The counter does operate in either Single 16-bit or Dual 8-bit modes as programmed by CRX2. Other features of the Time Interval Modes are outlined in Table 7.

If CRX5 = 0, as shown in Table 7 and Table 8, an interrupt is generated if $\overline{Gate}$ input returns low prior to a Time Out. If Counter Time-Out occurs first, the counter is recycled and continues to decrement. A bit is set within the timer on the initial Time Out which precludes further individual interrupt generation until a new Counter Initialization cycle has been completed. When this internal bit is set, a negative transition of the $\overline{Gate}$ input starts a new Counter Initialization cycle. (The condition of $\overline{G}\downarrow \cdot \overline{I} \cdot TO$ is satisfied, since a Time Out has occurred and no individual Interrupt has been generated.)

TABLE 7 – TIME INTERVAL MODES

CRX3 = 1			
CRX4	**CRX5**	**Application**	**Condition for Setting Individual Interrupt Flag**
0	0	Frequency Comparison	Interrupt Generated if $\overline{Gate}$ Input Period (1/F) is less than Counter Time Out (TO)
0	1	Frequency Comparison	Interrupt Generated if $\overline{Gate}$ Input Period (1/F) is greater than Counter Time Out (TO)
1	0	Pulse Width Comparison	Interrupt Generated if $\overline{Gate}$ Input "Down Time" is less than Counter Time Out (TO)
1	1	Pulse Width Comparison	Interrupt Generated if $\overline{Gate}$ Input "Down Time" is greater than Counter Time Out (TO)

Any of the timers within the PTM may be programmed to compare the period of a pulse (giving the frequency after calculations) at the $\overline{Gate}$ input with the time period required for Counter Time-Out. A negative transition of the $\overline{Gate}$ input enables the counter and starts a Counter Initialization cycle—provided that other conditions as noted in Table 8 are satisfied. The counter decrements on each clock signal recognized during or after Counter Initialization until an Interrupt is generated, a Write Timer Latches command is issued, or a Timer Reset condition occurs. It can be seen from Table 8 that an interrupt condition will be generated if CRX5 = 0 and the period of the pulse (single pulse or measured separately repetitive pulses) at the $\overline{Gate}$ input is less than the Counter Time Out period. If CRX5 = 1, an interrupt is generated if the reverse is true.

Assume now with CRX5 = 1 that a Counter Initialization has occurred and that the $\overline{Gate}$ input has returned low prior to Counter Time Out. Since there is no Individual Interrupt Flag generated, this automatically starts a new Counter Initialization Cycle. The process will continue with frequency comparison being performed on each $\overline{Gate}$ input cycle until the mode is changed, or a cycle is determined to be above the predetermined limit.

Pulse Width Comparison Mode (CRX3 = 1, CRX4 = 1) This mode is similar to the Frequency Comparison Mode except for a positive, rather than negative, transition of the $\overline{Gate}$ input terminates the count. With CRX5 = 0, an Individual Interrupt Flag will be generated if the zero level pulse applied to the $\overline{Gate}$ input is less than the time period required for Counter Time Out. With CRX5 = 1, the interrupt is generated when the reverse condition is true.

As can be seen in Table 9, a positive transition of the $\overline{Gate}$ input disables the counter. With CRX5 = 0, it is therefore possible to directly obtain the width of any pulse causing an interrupt. Similar data for other Time Interval Modes and conditions can be obtained, if two sections of the PTM are dedicated to the purpose.

TABLE 8 – FREQUENCY COMPARISON MODE

CRX3 = 1, CRX4 = 0				
Control Reg Bit 5 (CRX5)	Counter Initialization	Counter Enable Flip-Flop Set (CE)	Counter Enable Flip-Flop Reset (CE)	Interrupt Flag Set (I)
0	$\overline{G}\downarrow\cdot\overline{I}\cdot(\overline{CE}+TO)+R$	$\overline{G}\downarrow\cdot\overline{W}\cdot\overline{R}\cdot\overline{I}$	W+R+I	$\overline{G}\downarrow$ Before TO
1	$\overline{G}\downarrow\cdot\overline{I}+R$	$\overline{G}\downarrow\cdot\overline{W}\cdot\overline{R}\cdot\overline{I}$	W+R+I	TO Before $\overline{G}\downarrow$

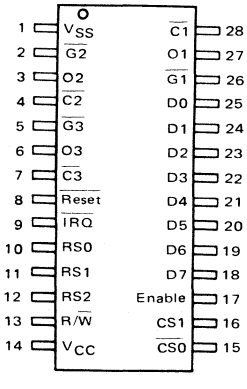
I represents the interrupt for a given timer.

TABLE 9 – PULSE WIDTH COMPARISON MODE

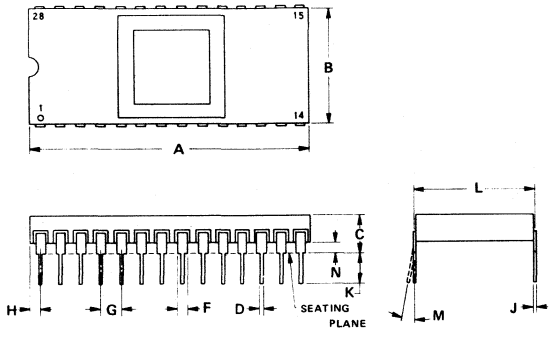
CRX3 = 1, CRX4 = 1				
Control Reg Bit 5 (CRX5)	Counter Initialization	Counter Enable Flip-Flop Set (CE)	Counter Enable Flip-Flop Reset (CE)	Interrupt Flag Set (I)
0	$\overline{G}\downarrow\cdot\overline{I}+R$	$\overline{G}\downarrow\cdot\overline{W}\cdot\overline{R}\cdot\overline{I}$	W+R+I+G	$\overline{G}\uparrow$ Before TO
1	$\overline{G}\downarrow\cdot\overline{I}+R$	$\overline{G}\downarrow\cdot\overline{W}\cdot\overline{R}\cdot\overline{I}$	W+R+I+G	TO Before $\overline{G}\uparrow$

G = Level sensitive recognition of $\overline{Gate}$ input.

PIN ASSIGNMENT



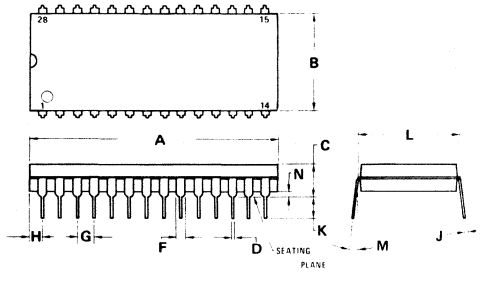
PACKAGE DIMENSIONS



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	35.20	35.92	1.386	1.414
B	14.94	15.34	0.588	0.604
C	3.05	4.19	0.120	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.30	0.008	0.012
K	2.54	4.19	0.100	0.165
L	14.99	15.49	0.590	0.610
M	–	10°	–	10°
N	1.02	1.52	0.040	0.060

NOTES:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIAMETER (AT SEATING PLANE) AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.

L SUFFIX
CERAMIC PACKAGE
CASE 719



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	36.45	37.21	1.435	1.465
B	13.72	14.22	0.540	0.560
C	3.94	5.08	0.155	0.200
D	0.36	0.56	0.014	0.022
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.38	0.008	0.015
K	2.92	3.43	0.115	0.135
L	15.24 BSC		0.600 BSC	
M	0°	15°	0°	15°
N	0.51	1.02	0.020	0.040

NOTES:
1. POSITIONAL TOLERANCE OF LEADS (D), SHALL BE WITHIN 0.25mm(0.010) AT MAXIMUM MATERIAL CONDITION, IN RELATION TO SEATING PLANE AND EACH OTHER.
2. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSION B DOES NOT INCLUDE MOLD FLASH.

P SUFFIX
PLASTIC PACKAGE
CASE 710

MOTOROLA *Semiconductor Products Inc.*
3501 ED BLUESTEIN BLVD., AUSTIN, TEXAS 78721 • A SUBSIDIARY OF MOTOROLA INC.

MOTOROLA

MC6843

Advance Information

FLOPPY DISK CONTROLLER (FDC)

The MC6843 Floppy Disk Controller performs the complex MPU/Floppy interface function. The FDC was designed to optimize the balance between the "Hardware/Software" in order to achieve integration of all key functions and maintain flexibility.

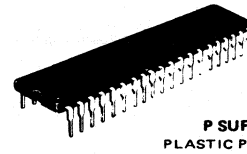
The FDC can interface a wide range of drives with a minimum of external hardware. Multiple drives can be controlled with the addition of external multiplexing rather than additional FDC's.

- Format compatible with IBM3740
- User Programmable read/write format
- Ten powerful macro commands
- Macro End Interrupt allows parallel processing of MPU and FDC
- Controls multiple Floppies with external multiplexing
- Direct interface with MC6800
- Programmable step and settling times enable operation with a wide range of Floppy drives
- Offers both Programmed Controlled I/O (PCIO) and DMA data transfer mode
- Free-Format read or write
- Single 5-volt power supply
- All registers directly accessible

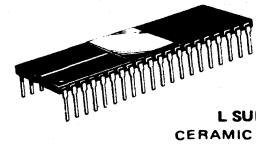
MOS

(N-Channel, Silicon-Gate)

FLOPPY DISK CONTROLLER

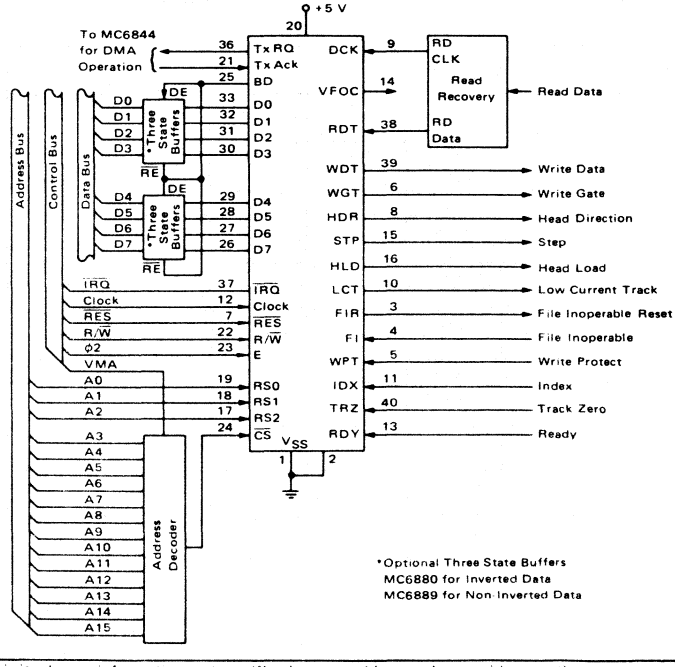


P SUFFIX
PLASTIC PACKAGE
CASE 711



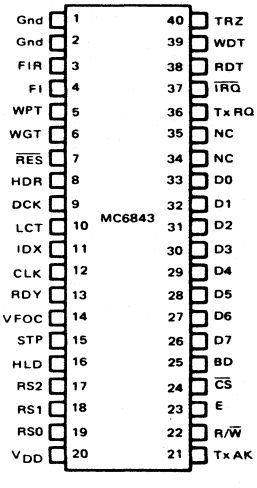
L SUFFIX
CERAMIC PACKAGE
CASE 715

FIGURE 1 – SYSTEM BLOCK DIAGRAM



*Optional Three State Buffers
MC6880 for Inverted Data
MC6889 for Non-Inverted Data

FIGURE 2 – PIN ASSIGNMENT



This is advance information and specifications are subject to change without notice.

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}*	-0.3 to +7.0	Vdc
Input Voltage	V_{in}*	-0.3 to +7.0	Vdc
Operating Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	- 55 to +150	°C

*With respect to V_{SS} (Gnd).

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ±5%, V_{SS} = 0, T_A = 0 to 70°C unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage		V_{IH}	V_{SS} + 2.0	–	V_{CC}	Vdc
Input Low Voltage		V_{IL}	V_{SS} – 0.3	–	V_{SS} + 0.8	Vdc
Input Leakage Current (V_{in} = 0 to 5.5 V)		I_{in}	–	1.0	2.5	μAdc
Three-State (Off-State) Input Current (V_{in} = 0.4 to 2.4 V, V_{CC} = 5.5 V)	D0-D7	I_{TSI}	–10	2.0	10	μAdc
Output High Voltage		V_{OH}				Vdc
(I_{load} = –205 μA)	D0-D7		V_{SS} + 2.4	–	–	
(I_{load} = –100 μA)	Other Outputs		V_{SS} + 2.4	–	–	
Output Low Voltage (I_{load} = 1.6 mA)		V_{OL}	–	–	V_{SS} + 0.4	Vdc
Output Leakage Current (Off-State) (V_{OH} = 2.4 V)	$\overline{IRQ}$	I_{LOH}	–	1.0	10	μA
Power Dissipation		P_D	–	400	–	mW
Input Capacitance		C_{in}				pF
(V_{in} = 0 V, f = 1.0 MHz, T_A = 25°C)	Enable		–	–	10	
	D0-D7		–	–	12.5	
	All others		–	–	10	
Output Capacitance (V_{in} = 0 V, f = 1.0 MHz, T_A = 25°C)	All Outputs	C_{out}	–	–	10	pF
Minimum Clock Pulse Width, Low		PW_{CL}	400	–	–	ns
Minimum Clock Pulse Width, High		PW_{CH}	400	–	–	ns
Clock Frequency		f_c	1.0	–	–	μs
Data Clock Pulse Width, Low		PW_{DCL}	1.3	1.95	–	μs
Data Clock Pulse Width, High		PW_{DCH}	1.3	1.95	–	μs
Data Clock Frequency		f_cD	2.5	4.0	–	μs
Read Data-to-Data Clock Delay Time 1		t_{RDD1}	0.55	1.0	–	μs
Read Data-to-Data Clock Delay Time 2		t_{RDD2}	0.55	1.0	–	μs
Minimum Read Data Pulse Width, High		t_{RDH}	–	2.0	–	μs
Minimum Read Data Pulse Width, Low		t_{RDL}	–	2.0	–	μs
Minimum Index Pulse Width, High		PW_{IDXH}	1.0	–	–	μs
Transfer Request Release Time		t_{TR}	–	–	450	ns
Interrupt Request Release Time		t_{IR}	–	–	1.2	μs
Bus Direction Delay Time		t_{DBD}	–	–	330	ns
Write Data Pulse Width, High (f_c = 1.0 MHz)		PW_{WDH}	–	1.0	–	μs
Minimum Write Data Cycle Time (f_c = 1.0 MHz)		t_{cycWD}	–	2.0	–	μs
Step Pulse Width, High (f_c = 1.0 MHz)		PW_{STP}	–	32	–	μs
Step Cycle Time* (f_c = 1.0 MHz)		t_{cycSTP}	1.0	–	15	ms

*Step (STP) cycle time is programmable.

BUS TIMING CHARACTERISTICS

Characteristic	Symbol	Min	Max	Unit
READ/WRITE (Figure 10)				
Enable Cycle Time	t_{cycE}	1.0	–	μs
Enable Pulse Width, High	PW_{EH}	0.45	25	μs
Enable Pulse Width, Low	PW_{EL}	0.43	–	μs
Setup Time, $\overline{CS}$, RS2-RS0, TxAK and R/$\overline{W}$ valid to Enable positive transition	t_{AS}	160	–	ns
Data Delay Time	t_{DDR}	–	320	ns
Data Hold Time	t_H	10	–	ns
Address Hold Time, $\overline{CS}$, RS2-RS0, TxAK, R/$\overline{W}$	t_{AH}	10	–	ns
Rise and Fall Time for Enable Input	t_{Er}, t_{Ef}	–	25	ns
Data Setup Time	t_{DSW}	195	–	ns
Data Access Time	t_{ACC}	–	480	ns

FIGURE 3 – TEST LOADS

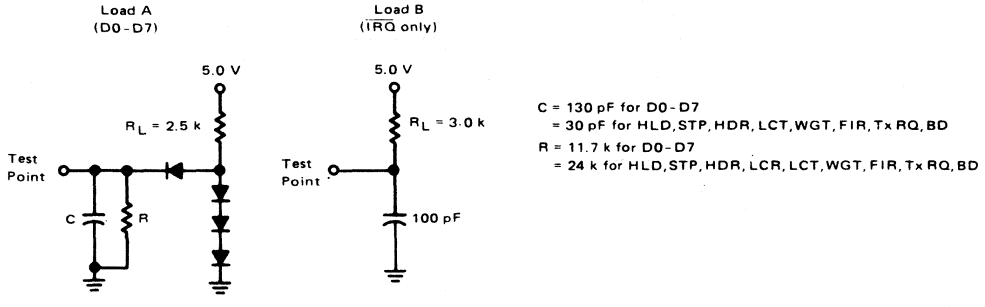


FIGURE 4 – BUS TIMING CHARACTERISTICS–READ/$\overline{WRITE}$

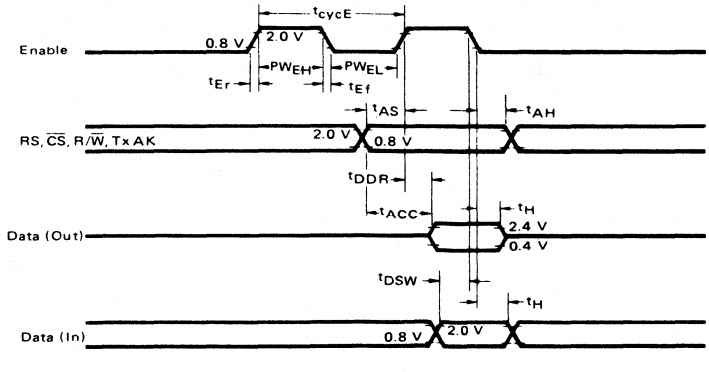


FIGURE 5 – MASTER CLOCK

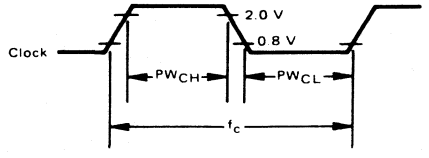


FIGURE 6 – READ DATA TIMING

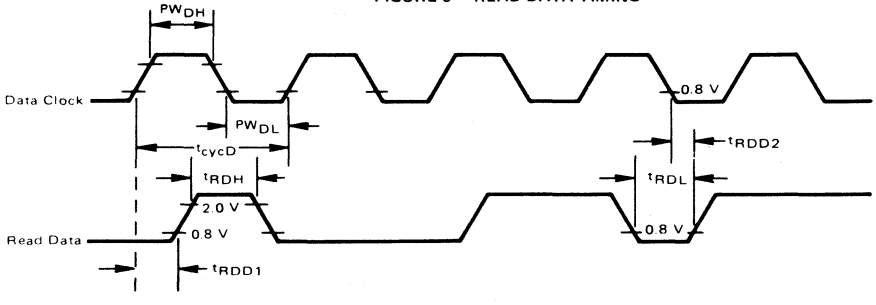


FIGURE 7 – INDEX TIMING

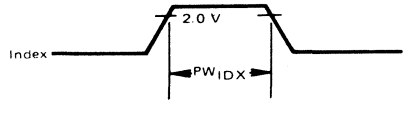


FIGURE 8 – $\overline{IRQ}$ RELEASE TIME

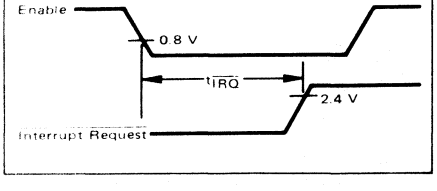


FIGURE 9 – DATA BUS DIRECTION TIMING

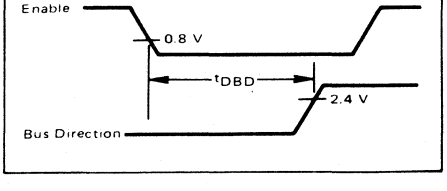


FIGURE 10 – WRITE DATA TIMING

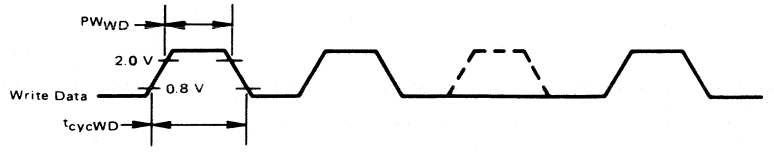


FIGURE 11 – STEP TIMING (PROGRAMMABLE)

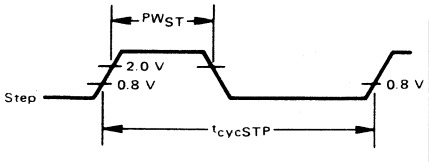


FIGURE 12 – Tx RQ RELEASE TIMING

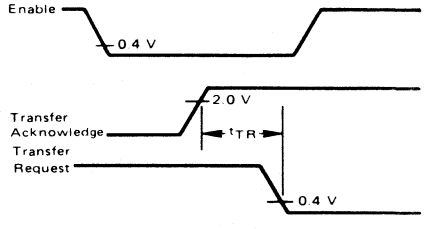
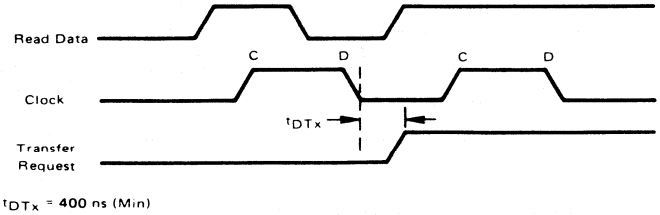


FIGURE 13 – DELAY TIME FROM DATA CLOCK TO TRANSFER REQUEST (t_{DTx})



t_{DTx} = 400 ns (Min)
= 700 ns (Max)

FIGURE 14 – WRITE DATA versus WRITE GATE TIMING

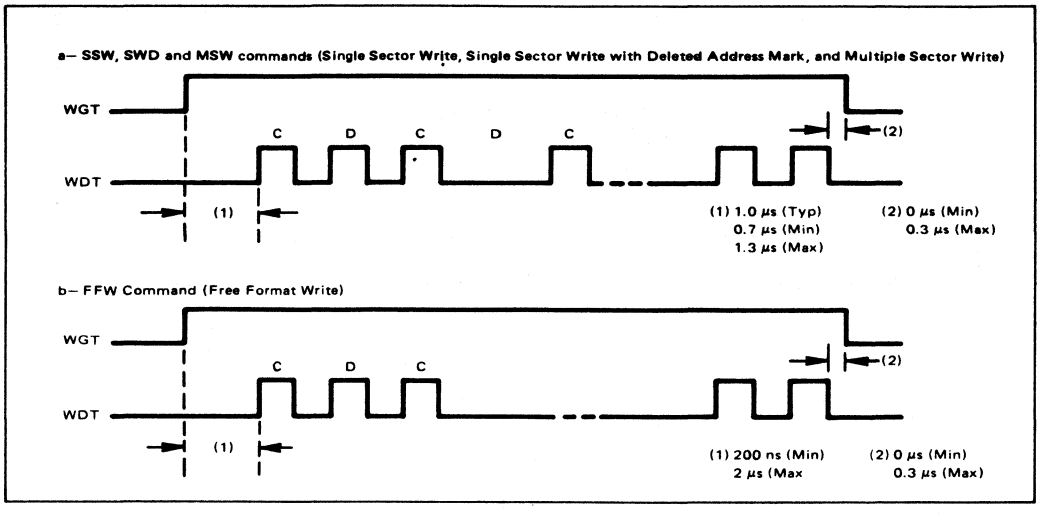


FIGURE 15 – INTERRUPT STATUS REGISTER AND INTERRUPT REQUEST TIMING

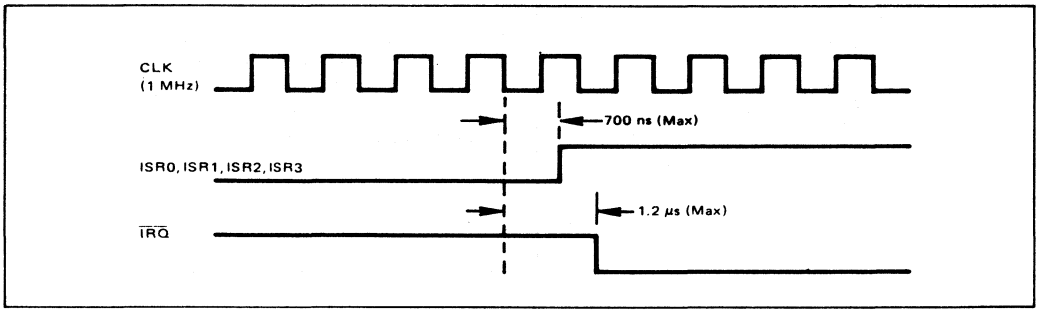
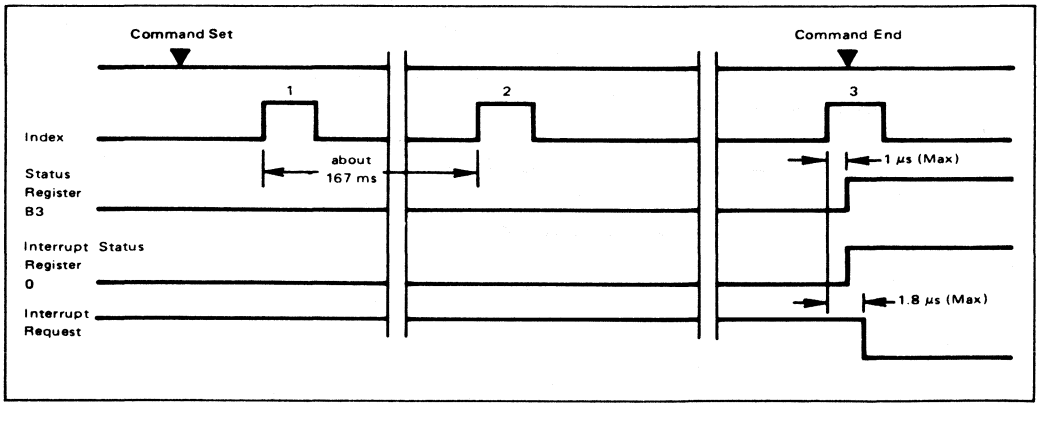


FIGURE 16 – TRACK NOT EQUAL ERROR TIMING



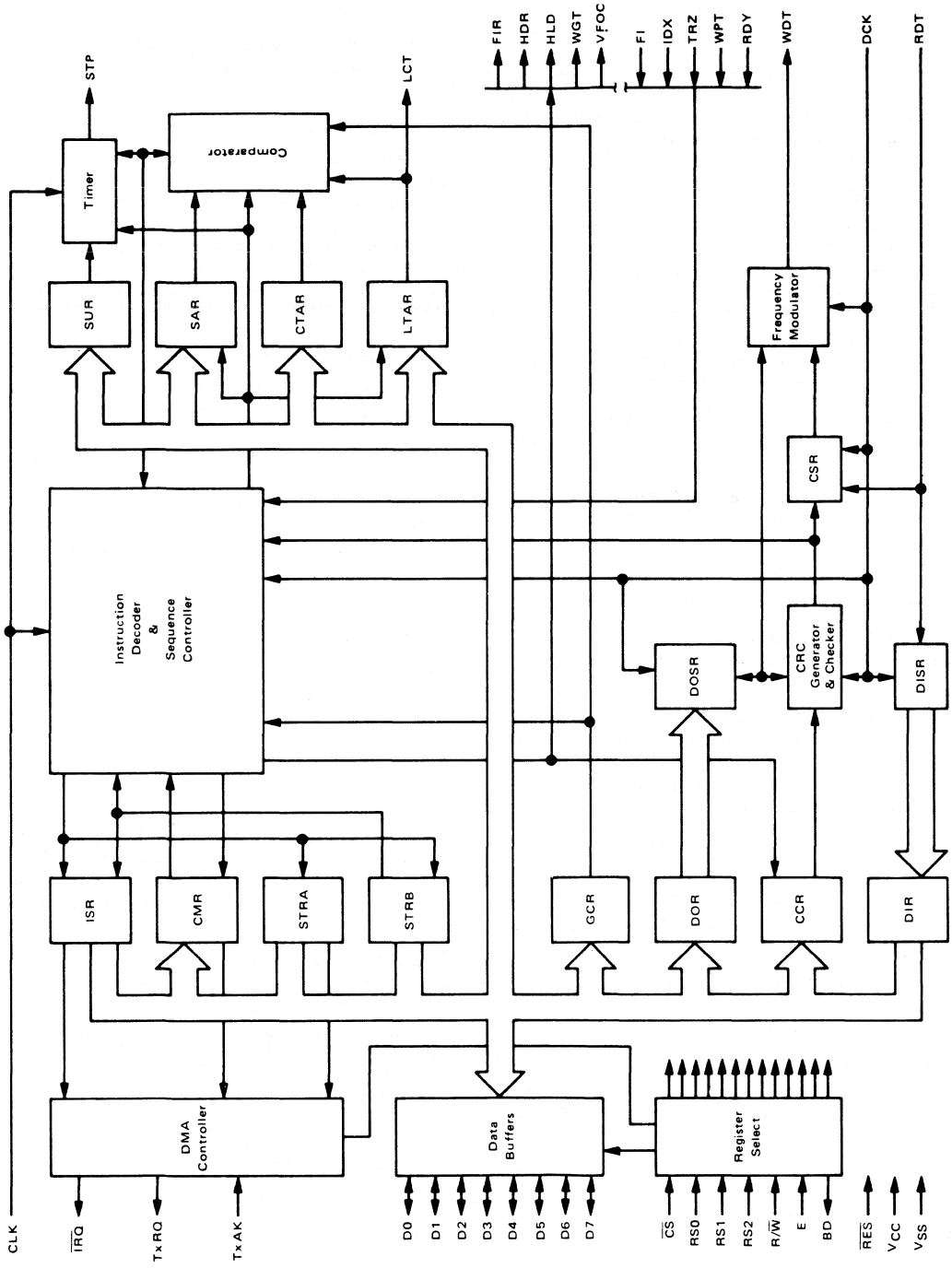


FIGURE 17 – INTERNAL BLOCK DIAGRAM

GENERAL DESCRIPTION

The MC6843 FDC consists of four primary sections; the Register, Serializing, Bus Interface, and Control sections. The following explanation of these sections can be followed in the block diagram of Figure 17.

Register Section

The register section consists of twelve user accessible registers used for controlling a floppy disk drive. All twelve are connected by the internal data bus to allow the processor access to them.

Data Output Register (DOR) – The DOR is an 8 bit register which holds the data to be written onto the disk. The information is stored here by the bus interface.

Data Input Register (DIR) – The data words read from the disk are stored in the 8-bit DIR until read by the bus interface.

Current Track Address Register (CTAR) – CTAR is a 7-bit register containing the address of the track over which the R/W head is currently positioned.

Command Register (CMR) – The macro commands are written to the 8-bit CMR to begin their execution.

Interrupt Status Register (ISR) – The four bits of the ISR represent the four conditions that can cause an interrupt to occur.

Set-Up Register (SUR) – Variable Seek and Settling times are programmed by the SUR. Four bits are used to program the track to track seek time and four bits are used to program the head settling time for the floppy disk drive used with the FDC.

Status Register A (STRA) – The eight bits of STRA are used to indicate the state of the floppy disk interface.

Sector Address Register (SAR) – SAR contains the five bit sector address associated with the current data transfer.

Status Register B (STRB) – The eight error flags of STRB are used to signify error conditions detected by the FDC or generated by the floppy disk drive.

General Count Register (GCR) – The seven bits of GCR contain the destination track address when a SEK (seek) macro command is being executed. If a multi-sector Read or Write macro command is being executed, GCR contains the number of sectors to be read or written.

CRC Control Register (CCR) – The two bits of the CCR are used to enable the CRC and shift the CRC for the Free Format Commands.

Logical Track Address Register (LTAR) – The seven bit track address used for read and write operations is stored in the LTAR by the bus interface.

Serializing Section

The serializing section handles the serial-to-parallel and parallel-to-serial conversions for Read/Write operations as well as CRC generation/checking and the generation/detection of the clock pattern. The Data Output Shift Register (DOSR), Data Input Shift Register (DISR), CRC Generator/Checker, and Clock Shift Register (CSR) comprise the serializing section of the FDC.

Bus Interface

The Bus Interface section provides the timing and control logic that allows the FDC to operate with the 6800 bus, and is comprised of the Data Buffers, DMA Control, and the Register Select circuitry.

Control

The internal timing and control signals which sequence the FDC are derived from the macro instructions by the control section.

MC6843 PIN DESCRIPTION

POWER PINS

V_{DD}: Input

+ 5 volt (±5%) power input.

V_{SS}: Input

Power Supply Ground.

BUS PINS

Master Clock (CLK) Input – The CLK input is used to generate various timing sequences internal to the FDC. The head settling and seek time, as well as the data and data clock timing, are generated from the CLK input signal.

$\overline{\text{Reset}}$ ($\overline{\text{RES}}$) Input – The $\overline{\text{Reset}}$ input is used to initialize the FDC. When $\overline{\text{RES}}$ becomes '0', the state of the outputs is defined by the table below:

Output	State of Output	Output	State of Output
FIR	Low	HLD	Low
WGT	Low	TxRQ	Low
HDR	Low	$\overline{\text{IRQ}}$	High
STP	Low	WDT	Low

Registers which are affected by $\overline{\text{RESET}}$ are shown in Table 7.

$\overline{\text{Interrupt Request}}$ ($\overline{\text{IRQ}}$) Output – The $\overline{\text{IRQ}}$ line is an open drain output that becomes a low level (logic '0')

when the FDC requests an interrupt. Interrupt requests are controlled by the interrupt enables in CMR (Command Register) with the function causing the interrupt shown in ISR (Interrupt Status Register).

Data Bus 0-Data Bus 7 (D0-D7) Bidirectional – The 8 bidirectional data lines allow the transfer of data between the FDC and the controlling system. The output buffers are three-state drivers that are enabled when the FDC is transferring data to the data bus.

Enable (E) Input – The E input to the FDC causes data transfers to occur between the FDC and the system controlling the FDC (MC6800 MPU, DMA Controller, etc.) E must be a logic '1' (high level) for any transfer to be enabled on D0-D7. The E input is normally connected to system $\phi 2$.

Chip Select ($\overline{CS}$) Input – The $\overline{CS}$ input in conjunction with the E input, is used to enable data transfers on D0-D7. E must be a high level and $\overline{CS}$ must be a low level (logic '0') to enable the transfer. The TxAK input being a high level (logic '1') performs a similar function as $\overline{CS}$ being a low level.

Read/Write (R/$\overline{W}$) Input – The R/$\overline{W}$ input is issued by the system controlling the FDC (MC6800 MPU, DMA Controller, etc.) to signify if a read or write operation is to be performed on the FDC. When TxAK is a low level, R/$\overline{W}$ is used in conjunction with $\overline{CS}$ and RS0-RS2 to determine which register is accessed by the bus as shown in Table 1. When TxAK is a high level, R/$\overline{W}$ is used to select either the DOR or DIR to the data bus (see description of TxAK input).

Register Select 0-Register Select 2 (RS0-RS2) Input – RS0-RS2, in conjunction with the R/$\overline{W}$ input, are used to select one of the user accessible registers in the FDC as shown in Table 1.

Transfer Request (TxRQ) Output – TxRQ is used in the DMA mode to request a data transfer from the DMAC. TxRQ is a high level if the FDC is in the DMA mode (CMR bit 5 is set) when a data transfer request occurs (STRA bit 1 is set). It is reset to a low level (logic '0') when TxAK becomes a high level (logic '1'). Data transfer errors will occur if TxAK does not reset TxRQ before the next data transfer is required.

Transfer Acknowledge (TxAK) Input – TxAK is generated by the system controlling the FDC (MC6800 MPU, DMA Controller, etc.) and is a response to a TxRQ issued by the FDC. A high level (logic '1') on TxAK causes the FDC to neglect the state of RS0-RS2 and $\overline{CS}$ causing the FDC to select the DOR (Data Output Register) or DIR (Data Input Register) to the data bus (D0-D7) as shown in Table 2.

TABLE 2 – REGISTER SELECTION FOR DMA TRANSFERS

TxAK	RS0-RS2	$\overline{CS}$	R/$\overline{W}$	Register Selected
1	X	X	1	DOR
1	X	X	0	DIR

This mode of operation is normally used for DMA (Direct Memory Access) transfer with the FDC.

When TxAK is a low level the registers are selected by $\overline{CS}$, R/$\overline{W}$ and RS0-RS2 as shown in Table 1.

Bus Direction (BD) Output – The BD output is provided to control external bidirectional buffers on the data bus (D0-D7) as shown in Figure 1. Its polarity is shown by Table 3.

TABLE 3 – BUS DIRECTION (BD) STATES

TxAK	$\overline{CS}$	BD
1	X	R/$\overline{W}$
0	1	0
0	0	R/$\overline{W}$

(Operation of BD as defined by this chart allows the FDC to function with the DMA Controller MC6844)

TABLE 1: ADDRESS CODES FOR USER ACCESSIBLE REGISTERS

TxAK	RS2	RS1	RS0	R/$\overline{W}$	Registers
0	0	0	0	0	DOR (Data Out Register)
				1	DIR (Data In Register)
0	0	0	1	1/0	CTAR (Current Track Address Register)
0	0	1	0	0	CMR (Command Register)
				1	ISR (Interrupt Status Register)
0	0	1	1	0	SUR (Set Up Register)
				1	STRA (Status Register A)
0	1	0	0	0	SAR (Sector Address Register)
				1	STRB (Status Register B)
0	1	0	1	0	GCR (General Count Register)
0	1	1	0	0	CCR (CRC Control Register)
0	1	1	1	0	LTAR (Logical Track Address Register)

I/O AND CONTROL PINS

Head Load (HLD) Output – HLD is used to notify the disk drive that the R/W head should be loaded (placed in contact with the media). When the FDC is ready for the head to load, HLD is a high level (logic '1'). A low level (logic '0') HLD, indicates the head should be unloaded.

Step (STP) Output – The STP output, in conjunction with HDR, is used to control head movement. A 32 μs wide positive (logic '1') pulse is generated on STP, to move the R/W head one track in the direction defined by the HDR output. The period of the STP signal is programmable by the SUR (Set-Up Register). The number of pulses generated on STP is the difference between the contents of the CTAR (Current Track Address Register) and the GCR (General Count Register) which contains the track address to which the head is to be moved.

Head Direction (HDR) Output – The HDR signal controls the direction of head movement. A high level (logic '1') signifies the head should step to the inside (toward the hub) of the disk. A low level (logic '0') indicates the direction of head movement should be to the outside of the disk.

Low Current Track (LCT) Output – The LCT signal is used to control the level of write current used by the disk drive. LCT is a low level (logic '0') when the write head is positioned over tracks 0-43. If it is over tracks 44-76, LCT is a high level (logic '1'). LCT is determined from the contents of the Current Track Address Register (CTAR).

Write Gate (WGT) Output – When a write operation is being performed, WGT is a logic '1' (high level). For a read operation, WGT is a low level (logic '0').

File Inoperable Reset (FIR) Output – FIR is an output from the FDC to the floppy disk drive to reset it from an inoperable status. If the FI input is a '1', a 1 μs pulse is generated on the FIR output whenever Status Register B is read.

File Inoperable (FI) Input – FI is an input level to the FDC from the drive. A high level indicates the drive is in an inoperable state. Its current state can be examined by reading bit 5 of Status Register B (STRB).

Track Zero (TRZ) Input – The TRZ input is reflected by bit 3 of STRA (Status Register A). The TRZ input must be a high level (logic '1') when the R/W head of the drive is positioned over track zero. A logic '1' on this input inhibits step pulses during a Seek Track Zero command.

Index (IDX) Input – The index input is received from the floppy disk drive and is used to sense the index hole in the disk media. The IDX signal is used to initialize the internal FDC timing. The state of the IDX input is reflected by bit 6 of Status Register A (STRA). A high level (logic '1') is to indicate the index hole is under the index sensor. The index input is used to count the number of disk revolutions while searching for the address ID field (see description of STRB bit 3).

Ready (RDY) Input – The ready input is received from the disk drive and can be read as bit 2 of STRA (Status Register A). A high level (logic '1') indicates the drive is ready and allows the FDC to operate the drive.

Write Protect (WPT) Input – WPT is an input indicating when the media is Write Protected. A high level during an FDC write operation results in a Write Error (STRB bit 6) but the FDC continues to perform the write function. The state of the WPT input can be read by examining bit 4 of the Status Register A (STRA).

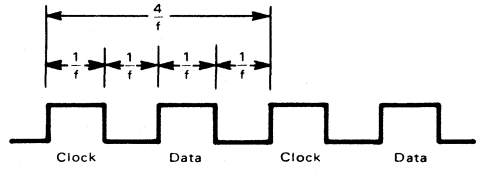
DATA PINS

Data Clock (DCK) Input – DCK is used to clock data from the drive into the FDC. It is generated from the read data received from the drive.

Read Data (RDT) Input – RDT is the serial data input from the drive. The data stream includes both the clock and data bits.

Write Data (WDT) Output – WDT is the double frequency modulated data output from the FDC. The time between clock bits is 4/f where f is the frequency of the clock input. The pulse width for both clock and data is 1/f (see Figure 18). For the normal clock frequency of 1 MHz the clock period is 4 μs, the clock pulse width is 1 μs and the data pulse width is 1 μs. Figure 18 shows the relationship between the WDT output and the frequency of the CLK inputs.

FIGURE 18 – WDT OUTPUT TIMING



f = Frequency of the Clock Input. To insure IBM3740 compatibility the clock frequency must be 1 MHz.

Variable Frequency Oscillator Control (VFOC) Output VFOC is used as a sync signal during system diagnostics. Waveforms are shown in Figure 19.

FIGURE 19 – VARIABLE FREQUENCY OSCILLATOR CONTROL WAVEFORM
(Relation Between WGT and VFOC)

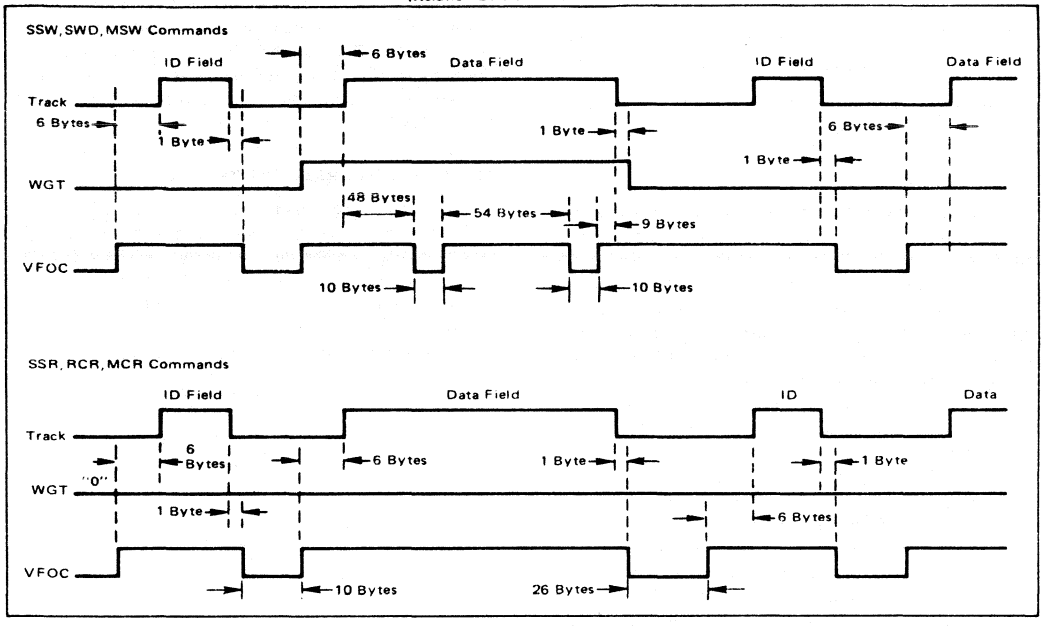
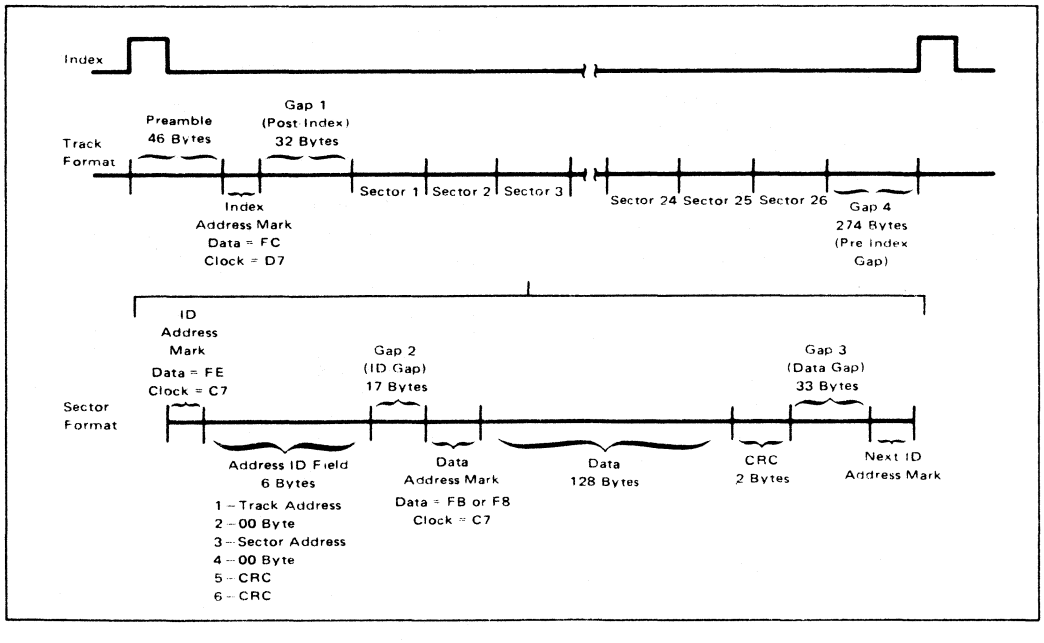


FIGURE 20 – SOFT SECTOR FORMAT



FORMAT

The format used by the MC6843, shown in Figure 20, is compatible with the soft sector format of the IBM 3740.

MACRO COMMAND SET

The macro command set shown in Table 4 is discussed in the following paragraphs.

Seek Track Zero (STZ)

The STZ command causes the R/W head to be released from the suface of the disk (HLD is reset) and positioned above track 00. The FDC issues step pulses on the STP output until the TRZ input becomes a high level or until 83 pulses have been sent to the drive. When the TRZ input becomes high, the step pulses are inhibited on the STP output but the FDC remains busy until all 83 have been generated internally.

If the TRZ input remains low (logic '0') after all 83 pulses have been generated, the seek error flag (STRB bit 4) is set.

After all 83 pulses have been generated, the head is loaded (HLD becomes a '1'). After the settling time specified in the SUR has expired, the settling time complete flag is set (ISR bit 1), Busy STRA-7 is reset, CTAR and GCR are cleared. The head remains in contact with the disk. A command such as RCR (Read CRC) may be issued following a STZ if the head must be released.

Seek (SEK)

The SEK command is used to position the R/W head over the track on which a Read/Write operation is to be performed. The contents of the GCR are taken as the destination address and the content of the CTAR is the source address; therefore, the number of pulses (N) on the STP output are given by:

$$N = |(CTAR) - (GCR)|$$

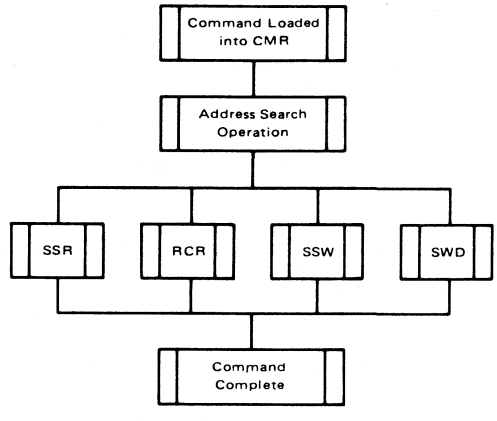
HDR is a '1' for (GCR) > (CTAR) otherwise it is a '0'.

When a SEK command is issued, Busy is set, the head is raised from the disk, HDR is set as described above, and N number of pulses appear on the STP output. After the last step pulse is used, the head is placed in contact with the disk. Once the head settling time has expired, the Settling Time Complete flag (ISR bit 1) is set, Busy is reset, and the contents of the GCR are transferred to the CTAR.

SINGLE SECTOR READ/WRITE COMMANDS

The single sector Read/Write commands (SSR, RCR, SSW, and SWD) are used to Read/Write data from a single 128 byte sector on the disk. As shown in Figure 21 these types of instructions can be divided into two sections. The first section, which is common to all instructions, is the address search operation, while the second section is unique to the requirements of each instruction.

FIGURE 21 – BASIC SINGLE SECTOR COMMAND FLOW CHART



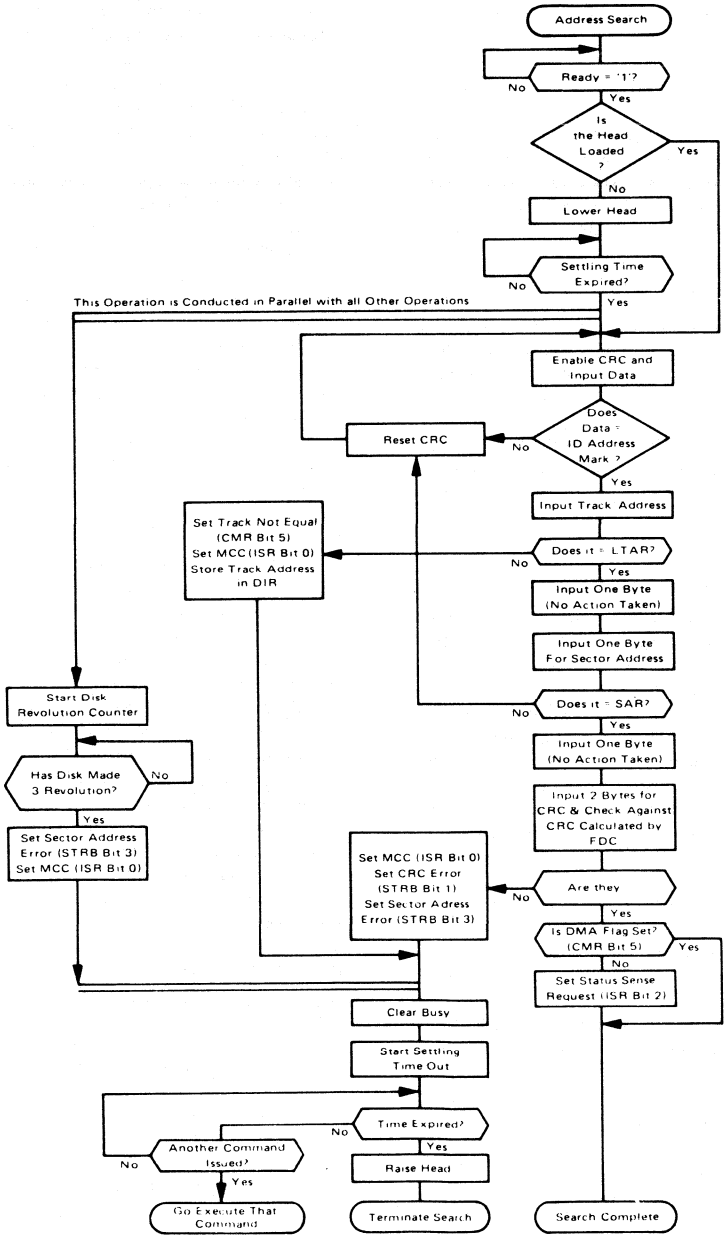
Address Search Operation

The flow chart of Figure 22 shows the operation of the address search operation.

TABLE 4 – MACRO COMMAND SET

			CMR Bits				Hex
			Bit 3	Bit 2	Bit 1	Bit 0	Code
1	STZ	Seek Track Zero	0	0	1	0	2
2	SEK	Seek	0	0	1	1	3
3	SSR	Single Sector Read	0	1	0	0	4
4	SSW	Single Sector Write	0	1	0	1	5
5	RCR	Read CRC	0	1	1	0	6
6	SWD	Single Sector Write with Delete Data Mark	0	1	1	1	7
7	MSW	Multi Sector Write	1	1	0	1	D
8	MSR	Multi Sector Read	1	1	0	0	C
9	FFW	Free Format Write	1	0	1	1	B
10	FFR	Free Format Read	1	0	1	0	A

FIGURE 22 – OPERATIONAL FLOW OF THE ADDRESS SEARCH SEQUENCE



Single Sector Read (SSR)

The single sector read command follows the address search procedure as defined in the previous flowchart. If the search is successful, status sense request is set and the operation continues as described by the flowchart of Figure 23.

Read CRC (RCR)

The RCR command is used to verify that correct data was written on a disk. The operation is the same as for the SSR command with the exception that the data transfer request (STRA bit 0) is not set. The SSR interrupt can be disabled by using the DMA mode.

Single Sector Write (SSW)

Single sector write is used to write 128 bytes of data on the disk. After the command is issued, the address search is performed. The remainder of the instruction's operation is shown in Figure 24.

Single Sector Write with Delete Data Mark (SWD)

The operation flow of SWD is exactly like that of SSW. For SWD, the data pattern of the Data Address Mark becomes F8 instead of FB. The clock pattern remains C7.

Multi-Sector Commands (MSR/MSW)

MSR is used for sequential reading of two or more sectors. If S sectors are to be read, S-1 must be written into the GCR before the command is issued.

The basic operation for the MSR and MSW is the same as that for the SSR and SSW respectively. The basic operation begins with an address search operation, which is followed by a single sector read or write operation. This completes the operation on the first sector. The SAR is incremented, the GCR is decremented, and if no overflow is detected from the GCR (i.e., GCR become negative) the sequence is repeated until S number of sectors are read or written.

The completion of an MSR or MSW is like that of an SSR or SSW command. First MCC is set, after the settling time has expired, Busy is reset, and the head is released.

If a delete data mark is detected during an MSR command, STRA bit 1 (Delete Data Mark Detected) remains set throughout the commands operation.

When a multi-sector instruction is issued, the sum of the SAR and GCR must be less than 27. If SAR + GCR > 26, an address error (STRB bit 3 set) will occur after the contents of SAR becomes greater than 26.

Free Format Write (FFW)

The FFW has two modes of operation which are selected by FWF (Free Format Write Flag) which is data bit 4 of the CMR.

When FWF = '0', the data bits of the DOR are written directly to the disk without first wrting the preamble, address mark, etc. The contents of the DOR are FM modulated with a clock pattern of all ones.

If FWF = '1' the odd bits of the DOR are used as clock bits and even bits are used for data bits. In this mode, the DOSR clock is twice a normal write operation and one byte of DOR is one nibble (four bits of data) on the disk.

The two modes of the FFW command allow formatting a disk with either the IBM3470 format or a user defined format.

After the FFW command is loaded into the CMR, WGT becomes a high level, the contents of DOR are transferred to the DOSR, data transfer request (STRA bit 0) is set, and the serial bit pattern is shifted out on the WDT line. Therefore, DOR must be loaded before the FFW command is issued. Data from the DOR is continually transferred to the DOSR and shifted out on WDT until the CMR has been written with an all zero pattern. When CMR becomes zero, WGT beocmes a low level, but MCC is not set and the R/W head is left in contact with the disk.

Free Format Read (FFR)

FFR is used to input all data (including Address marks) from a disk. Once the FFR command is set into the CMR, the head is loaded and after the settling time has expired the serial data from the FDC is brought into the DISR. After 8 bits have accumulated, it is transferred to the DIR and Data Transfer Request (STRA bit 0) is set.

This operation continues until a zero pattern is stored in the CMR, terminating the FFR command. As in the case of the FFW command, MCC is not set and the head remains in contact with the disk.

The first data that enters the DISR is not necessarily the first bit of a data word since the head may be lowered at any place on the disk. To prevent the FDC from remaining unsynchronized to the data, the FFR command will synchronize to either an ID address mark (FE) or a data Address mark (FB or F8).

REGISTER DEFINITIONS

Data Output Register (DOR) – Hex address 0, write only

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
8 Bits of Data Used for a Disk Write Operation							

When one of the four write macro commands (SSW, SWD, MSW, and FFW) is executed, the information contained in the DOR is loaded into the DOSR, and is shifted out on the WDT line using a double frequency (FM) format.

FIGURE 23 – OPERATIONAL FLOW OF THE SSR COMMAND

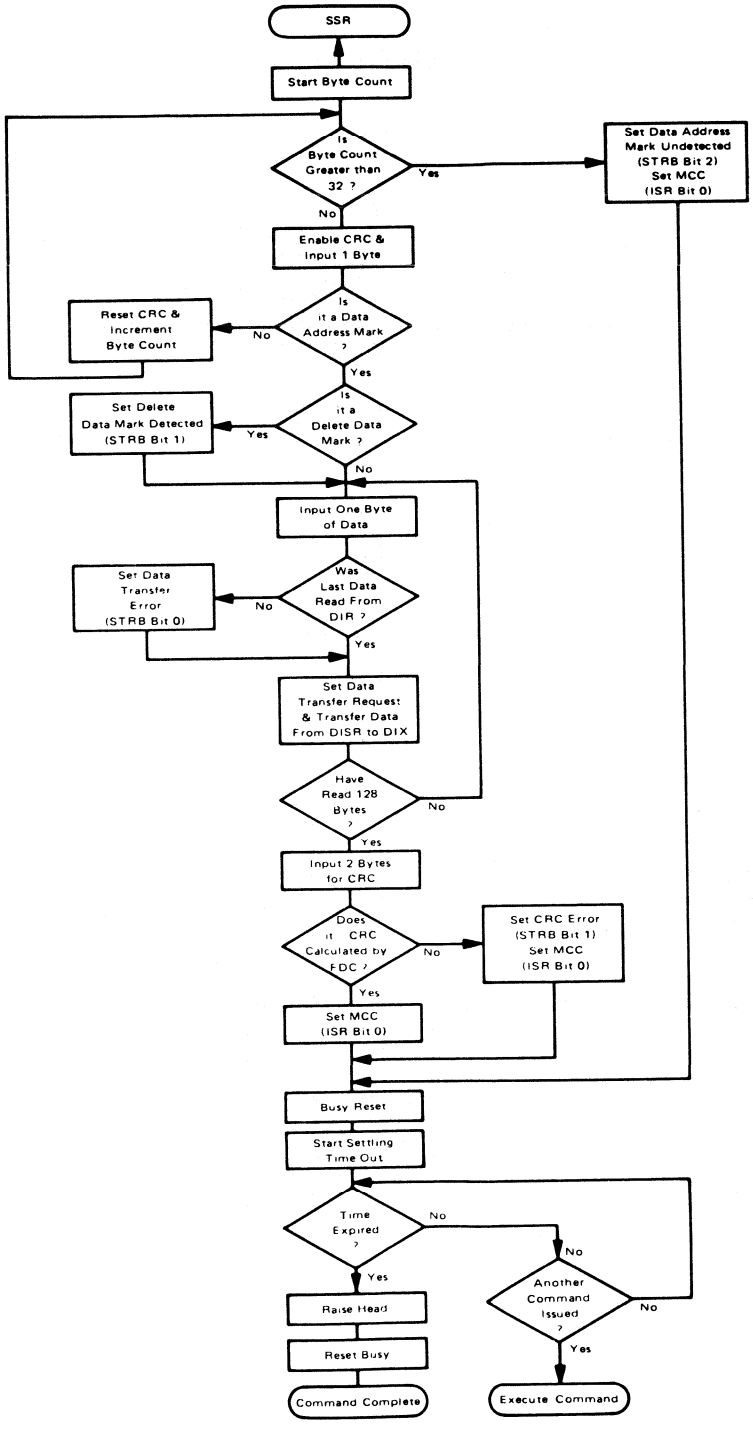
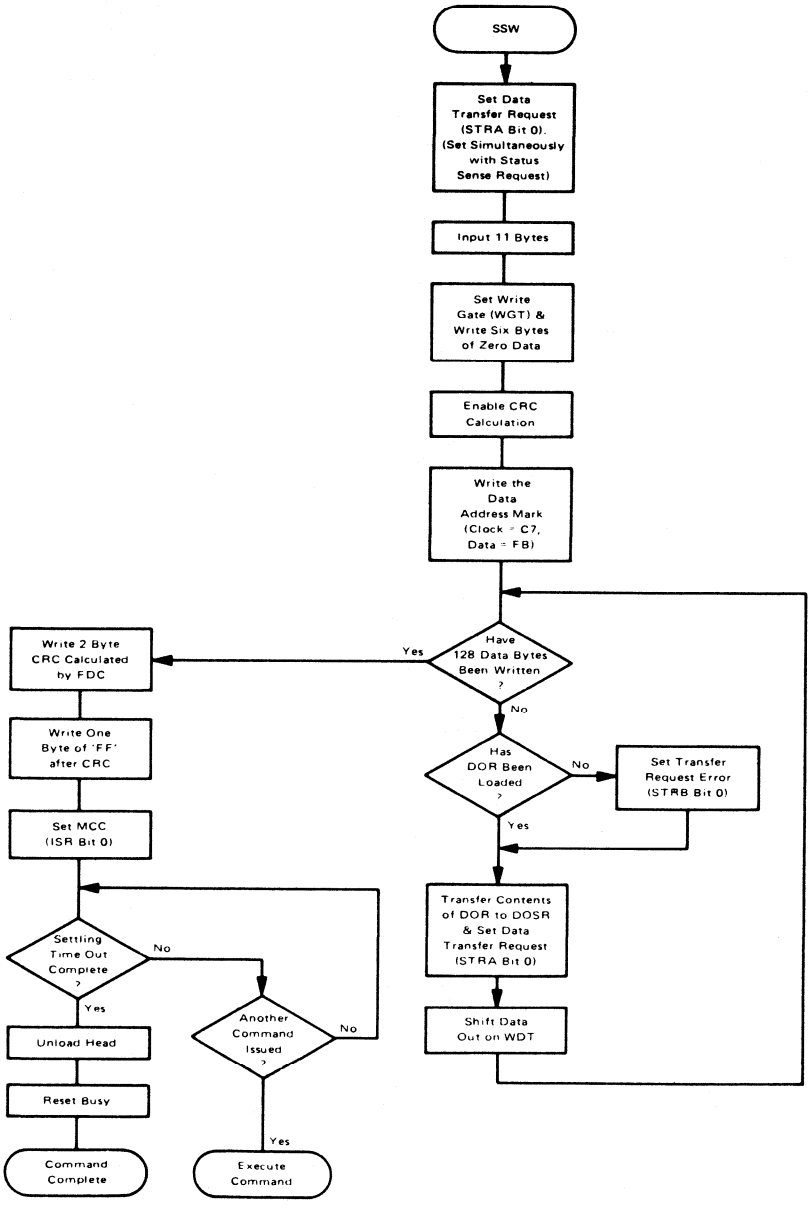


FIGURE 24 – OPERATIONAL FLOW OF THE SSW COMMAND



Data Input Register (DIR) – Hex address 0, read only

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	8 Bits of Data Used for a Disk Read Operation						

One of the three read macro commands (SSR, MSR, FFR) executed, will cause the information on the RDT input to be clocked into the DISR. When 8 clock pulses have occurred, the 8 bits of information in the DISR are transferred to the DIR where it can be read by the bus interface.

Current Track Address (CTAR) – Hex address 1, read/write

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Not Used	7 Bit Track Address of Current Head Position						

The address of the track over which the R/W head is currently positioned is contained in the CTAR. At the end of a SEK command, the contents of the GCR are transferred to the CTAR. CTAR is cleared at the completion of a STZ command. CTAR is a read/write register so that the head position can be updated when several drives are connected to one FDC. Bit 7 is read as a '0'.

Command Register (CMR) – Hex address 2, write only

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3*	Bit 2*	Bit 1*	Bit 0*
Function Interrupt Mask	ISR3 Interrupt Mask	DMA Flag	FWF	Macro Command			

*Bits 0-4 are cleared by $\overline{\text{Reset}}$.

The commands that control the FDC are loaded into the lower four bits of the CMR. Information that controls the data transfer mode and interrupt conditions are loaded into bits four through seven.

Bit 0-Bit 3: Macro Command

The Macro Command to be executed by the FDC is written to bits 0-3.

Bit 4: Free Format Write Flag (FWF)

If a Free Format Write command is issued, the state of bit 4 of the CMR determines what clock source will be used. The FWF is defined in the FFW (Free Format Write) command explanation.

Bit 5: DMA Flag

If bit 5 is a '1' the FDC is in the DMA mode. Bit 5 being a '1' inhibits setting of Status Sense Request (ISR bit 2) thereby preventing its associated interrupt. A logic '1' DMA flag also enables the TxRQ output allowing it to request DMA transfers when the Data Transfer Request flag (STRA bit 0) is set.

A logic '0' DMA flag indicates the program controlled I/O (PC I/O) mode.

Bit 6: ISR3 Mask

CMR bit 6 (ISR3 Mask) is used to control the operation of ISR bit 3. A logic '1' in CMR bit 6 inhibits ISR bit 3 from being set when STRB becomes non-zero. If CMR bit 6 (ISR3 Mask) is a '0' the ISR bit 3 will be set if any bit in STRB becomes set. The setting of ISR bit 3 will cause an interrupt if CMR bit 7 is a '0'.

Bit 7: Function Interrupt Mask

When CMR bit 7 is a logic '1' all interrupts are inhibited except Status Sense Request (ISR bit 2) which can only be inhibited by the DMA flag (CMR bit 5). A logic '0' in CMR bit 7 enables interrupts from ISR0 (Macro Command Complete) and ISR1 (Settling Time Complete), and if the ISR3 Mask is '0', from ISR3.

TABLE 5

Interrupt Status Register (Bits Causing Interrupts)	Command Register Masks That Affect Interrupts		
	CMR7 (Function Interrupt Mask)	CMR6 (ISR3 Mask)	CMR5 (DMA Flag)
ISR0 (Macro Command Complete)	M	X	X
ISR1 (Settling Time Complete)	M	X	X
ISR2 (Status Sense Request)	X	X	M
ISR3 (STRB Conditions)	M	M	X

X = No effect

M = Bits that are used as masks

Interrupt Status Register (ISR) – Hex address 2, read only

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Not Used (Read as '0')				STRB*	Status* Sense Request	Settling* Time Complete	Macro* Command

*Cleared by $\overline{\text{Reset}}$

Bit 0: Macro Command Complete

When an SSR, RCR, SSW, SWD, MSR or MSW Macro Command has completed execution, bit 0 becomes set (logic '1'). If the function interrupts are enabled (bit 7 of CMR is a logic '0'), the conclusion of a Macro Command's execution will cause an interrupt.

Bit 1: Settling Time Complete

Settling time complete is set on SEK and STZ commands to indicate the head has been loaded and the settling time specified in SUR has expired. Since MCC is not set for the SEK or STZ command, settling time complete can be used as an interrupt to signify the SEK or STZ command has finished. Setting Time Complete is not set for any of the R/W commands.

Bit 2: Status Sense Request

For an SSR, SSW, SWD, MSR, or MSW Command, Status Sense Request indicates that the specified address ID field has been detected and verified by a CRC check. This is used as an early indication that data transfers will occur after 18 more byte times. For MSR and MSW commands, it is set for each sector.

In the PC I/O mode, an interrupt occurs when Status Sense Request becomes a logic '1' regardless of the state of the CMR interrupt mask. In the DMA mode, (DMA flag of CMR is set) Status Sense Request is unchanged and does not generate an interrupt when the address ID field has been verified.

BIT 3: STRB

STRB is an 'OR' of all of the bits of Status Register B and is disabled by the STRB interrupt mask in the CMR (CMR bit 6). The equation:

$$STRB = \overline{CMR6} \cdot (STRB0 + STRB1 + STRB2 + STRB3 + STRB4 + STRB5 + STRB6 + STRB7)$$

describes the operation of Bit 3 of the ISR.

ISR0, ISR1, and ISR2 are cleared when the Interrupt Status Register is read, but ISR3 is cleared only after Status Register B has been read.

Set-Up Register (SUR) – Hex address 3, write only

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Track to Track Seek Time				Head Settling Time			

The SUR is not affected by a reset operation; therefore, once it is initialized, the information remains until power is removed from the FDC.

Bit 0-Bit 3: Head Settling Time

The head settling time is used to generate a delay after the head is placed in contact with the disk. This allows the head to stop bouncing before any operations are performed. The delay is programmed by bits 0-3 and is specified by the equation:

$$Delay = \frac{4096}{f} \cdot B$$

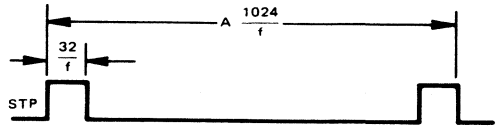
B = Number contained in bits 0-3 of SUR

f = Frequency of CLK input

For IBM3740 compatibility f = 1 MHz and the timing range is 4.096 ms for a "0001" to 61.44 ms for a '1111' A '0000' code prevents Settling Time complete from being set and the FDC must be Reset.

Bit 4-Bit 7: Track to Track Seek Time

The frequency of STP is determined by bit 4-bit 7 of SUR as shown below. If the track to track seek time is 0 the period of STP is 64/f.



A = Number specified in bits 4-7 of SUR.
f = Frequency of clock input.

For IBM compatible operation, f is 1 MHz. This results in an STP pulse width of 32 μs and an STP interval of 1.024 ms for a 0001 to 15.36 ms for a 1111.

Status Register A (STRA) – Hex address 3, read only

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Busy*	Index	Track* Not Equal	Write Protect	Track Zero	Drive Ready	Delete* Data Mark Detected	Data * Transfer Request

*Cleared by $\overline{Reset}$

Bit 0: Data Transfer Request

For a write operation (SSW, SWD, MSW, FFW) the transfer request bit indicates that the DOR is ready to accept the next data word to be written on the disk. If data is not written into the DOR before the last data bit in the DOSR is shifted out to the WDT line, the data transfer error bit (bit 0 of STRB) will be set. After a write command has been issued, the first transfer request occurs simultaneously with the Status Sense Request. For a write operation, transfer request is reset after the DOR has been written from the data bus.

During a read operation (SSR, MSR, FFR) the transfer request bit signifies data from the DISR has been transferred to the DIR. The DIR must be read before the DISR is full again or the data transfer error bit (bit 0 of STRB) will be set. For read operations, transfer request is reset by a read of the DIR.

Bit 1: Delete Data Mark Detected

A Single Sector Read operation that detects a delete data code (F8) instead of a general data code (FB) as a Data Address Mark will set the Delete Data Mark Detected bit. For the MSR command, bit 1 is set the first time an 'F8' code is found and remains set throughout the

execution of the command. Bit 1 is reset whenever an SSR, SSW, SWD, MSR, MSW, or RCR command is issued.

Bit 2: Drive Ready

The Drive Ready bit indicates the state of the Ready input from the floppy disk drive. If a command is issued with Ready at logic '0', its execution will be inhibited until Ready becomes a logic '1'. If ready becomes a '0' during the execution of a command the Hard Error Flag (STRB bit 7) is set.

Bit 3: Track Zero

The state of the Track Zero input from the floppy disk drive is reflected in this bit of STRA. A logic '1' on the Track Zero input inhibits step pulses during an STZ command.

Bit 4: Write Protect

The Write Protect input from the floppy disk drive is reflected by bit 4 of STRA. A high level (logic '1') on the WPT input during the execution of any write command results in a write error (bit 6 of STRB set).

Bit 5: Track Not Equal

If the track address read from the address ID field does not coincide with the address in the LTAR, the Track Not Equal bit is set. Track Not Equal applies to all non-free format read/write commands, and is reset after a non-free format read/write command is issued.

Bit 6: Index

The state of the index input appears in bit 6 of STRA. The index input is used to count the number of disk revolutions while the FDC is looking for the address ID field (see operation of STRB bit 3) during the address search phase of a non-free format read/write command.

Bit 7: Busy

When Busy is a logic '1', the FDC is executing a command and no new commands can be issued.

Sector Address Register (SAR) – Hex address 4, write only

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Not Used			5 Bit Sector Address				

Before a data transfer macro command (SSW, SWD, SSR, RCR, MSW, MSR) is issued, the address of the sector on which the operation is to be performed must be written into the SAR. The address in the sector address byte of an Address ID field of the disk is compared with the contents of the SAR. During an MSW or MSR command, the SAR is incremented after each sector is read or written. When execution is complete, the SAR contains the address of the last sector on which an operation was performed plus one. At the completion of an STZ of SEK command SAR is cleared.

Status Register B(STRB) – Hex address 4, read only

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Hard* Error	Write* Error	File Inop-	Seek* Error	Sector* Address Unde- tected	Data* Mark Unde- tected	CRC* Error	Data* Trans- fer Error

*Cleared by $\overline{\text{Reset}}$

The bits of the STRB represent possible error conditions that may occur during execution of macro commands. Whenever STRB is reset, ISR bit 3 is also reset.

Bit 0: Data Transfer Error

Data Transfer indicates an underflow or overflow of data. If a Write operation is being performed, it signifies that data was not presented to the DOR before the DOSR became empty. In this case, the current contents of the DOR are transferred to the DOSR and the write operation continues. The data transfer error remains set until STRB is read, and the data transfer request remains set until data is written into the DOR. The operation of the CRC is unchanged.

For read commands, a data transfer error indicates that data in the DIR was not read before the next data word from the disk was transferred to the DIR. The read operation continues until sufficient data has been read from the disk to satisfy the requirements of the command (128 bytes for SSR). The error indication remains set until STRB is read, and the transfer request remains set until data is read from the DIR.

Bit 1: CRC Error

A CRC error occurs when the CRC read from the disk does not match that calculated by the FDC on the data it reads from the disk. A CRC error can occur in three different situations; checking the address ID field, checking the data field, and checking the FFR data. (See operation of CCR.)

If the CRC error occurs during the check of an address ID field, Sector Address Undetected (STRB bit 3) will also be indicated (see Table 6). A CRC error of a data field is indicated by a CRC error and no sector address error.

Bit 2: Data Mark Undetected

If a valid data mark is not detected in the data block of a sector, it is indicated by a Data Mark Undetected error. Data Mark Undetected is reset after a non-free format Read/Write command is issued.

Bit 3: Sector Address Undetected

The sector address bit can be set on two conditions; not finding the sector address and a CRC error on an address ID field.

If the disk makes three revolutions during an address search operation and the sector address specified in the sector address register is not found in any of the address ID fields, a sector address undetected condition is indicated.

A CRC error that occurs on an address ID field will set bit 3 also. Table 6 shows how bits 1 and 3 are related.

TABLE 6 – RELATIONSHIP OF CRC ERROR AND SECTOR ADDRESS UNDETECTED

CRC Error (STRB 1)	Sector Address Undetected (STRB 3)	Condition
0	0	No Error
0	1	Sector Address not Detected
1	0	CRC Error on a Data Field
1	1	CRC Error on Address ID Field

Bit 4: Seek Error

An STZ (Seek Track Zero) command that never receives a track zero indication on the track zero input will result in a Seek Error (see description of STZ command).

Bit 5: File Inoperable

The state of the File Inoperable input appears in bit 5. If the File Inoperable input is a '1', a pulse of width 1/f (where f = Frequency of the clock input) is issued on the FIR output when STRB is read. FI is not latched but the input is gated to the bus when STRB is read.

Bit 6: Write Error

If the WPT input becomes a high level (logic '1') during the execution of a write command the write error bit is set.

Bit 7: Hard Error

If the Ready input becomes a '0' during the operation of a command (Busy is set), a hard error indication will result.

General Count Register (GCR) – Hex address 5, write only

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Not Used	7 Bit Count for Track Number on SEK Command and Sector Count for MSR or MSW Command						

The GCR contains the destination track address for the R/W head on an SEK Macro Command. The contents of the GCR are transferred to the CTAR at the end of the SEK Command. For multi-sector read or write operations (MSR, MSW), the GCR contains the number of sectors to be read minus one. During the MSR or MSW execution the GCR is decremented after each sector is read or written.

CRC Control Register (CCR) – Hex address 6, write only

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Not Used						Shift CRC	CRC Enable

The CCR information is used only in the free format commands; for all other commands this register is masked and has no function.

Bit 0: CRC Enable

During an FFW command, CRC Enable is set by software and CRC generation takes effect on the next transfer of data from DOR to DOSR (see figure 25). The CRC generation continues until Shift CRC (CCR bit 1) is set.

For an FFR command, CRC Enable is set by software and CRC generation takes effect on the next data read from DIR. The calculation continues for all data bytes read from DIR until CRC Enable is reset. The bytes read previous to resetting CRC Enable are considered the CRC information bytes and the CRC check is made against them.

Bit 1: Shift CRC

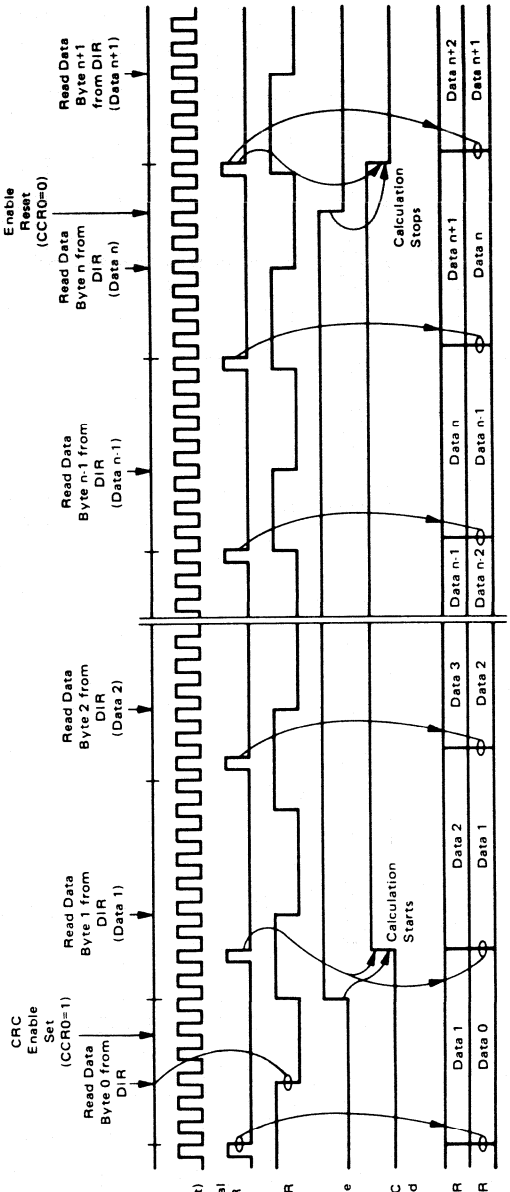
Bit 1 is valid only for the FFW command. After setting, it takes effect on the next transfer of data from DOR to DOSR (see Figure 26). Setting Shift CRC terminates the CRC calculation and causes the CRC calculated on all the data written into DOR up to the setting of bit 1, to be shifted out the WDT output. The CRC calculation will not include any data written to DOR after Shift CRC is set.

LTAR (Logical Track Address) – Hex address 7, write only

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Not Used	7 Bit Logical Track Address						

When a read or write macro command (SSW, SWD, SSR, RCR, MSW, MSR) is issued, the address of the track on which the operation is to be performed must be written into the LTAR. The address in the track address byte of an Address ID field of the disk is compared with the contents of the LTAR. The contents of LTAR are not affected by the execution of any of the commands.

FIGURE 25 – CCR CONTROL REGISTER TIMING FOR AN FFR COMMAND (READ)



CRC Calculation includes Data Byte 1 through Data Byte n.

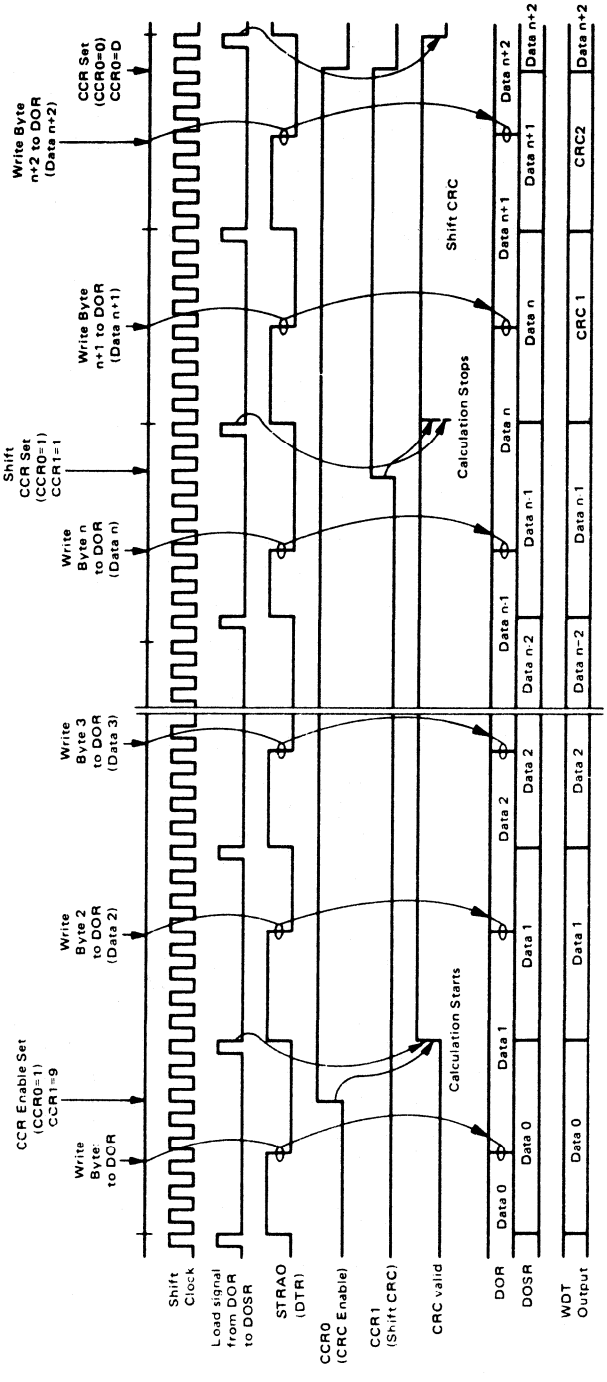


FIGURE 26 – CCR CONTROL REGISTER TIMING FOR AN FFW COMMAND (WRITE)

The CRC Calculation includes Data Byte 1 through Data Byte n 1.

TABLE 7 – PROGRAMMING REFERENCE DATA

Table 7 is a summary of the information in the data sheet and can be used as a reference when programming the MC6843.

Registers	Hex Address	R/$\overline{W}$ Mode	Data Bits							
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DOR	0	WO	8 Bits of Data Used for a Disk Write Operation							
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DIR	0	RO	8 Bits of Data Used for a Disk Read Operation							
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CTAR	1	R/W	Not Used	7 Bit Track Address of Current Head Position						
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CMR	2	WO	Function Interrupt Mask	ISR3 Interrupt Mask	DMA Flag	FWF	* Macro Command	*	*	*
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ISR	2	RO	Not Used				* STRB	Status * Sense Request	Settling * Time Complete	Macro * Command Complete
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SUR	3	WO	Track to Track Seek Time				Head Settling Time			
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STRA	3	RO	* Busy	Index	Track * Not Equal	Write Protect	Track Zero	Drive Ready	Delete * Data Mark Detected	Data * Transfer Request
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SAR	4	WO	Not Used			5 Bit Sector Address				
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STRB	4	RO	* Hard Error	* Write Error	File Inoperable	* Seek Error	* Sector Address Undetected	* Data Mark Undetected	* CRC Error	* Data Transfer
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
GCR	5	WO	Not Used	7 Bit Count for Track Number on SEK or Sector Count for MSR or MSW.						
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CCR	6	WO	Not Used						Shift CRC	CRC Enable
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LTAR	7	WO	Not Used	7 Bit Search Track Address						

RO – Read Only
WO – Write Only
R/W – Read/Write

*Cleared by Reset

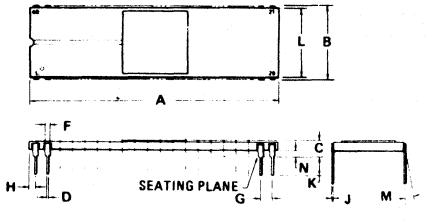
MACRO COMMANDS

Hex Code	Instruction	Hex Code	Instruction
2	STZ	A	FFR
3	SEK	B	FFW
4	SSR	C	MSR
5	SSW	D	MSW
6	RCR		
7	SWD		

TABLE 8

Table 8 is a list of all error flags showing what conditions will cause the error, the instructions for which they are valid, and what conditions reset them.

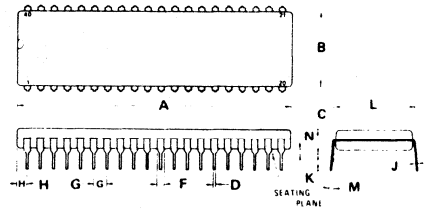
Name	Flag	Set Condition	Reset Condtiion	Command
Track Not Equal	STRA5	The track address that is read from the disk does not coincide with the content of the LTAR.	Upon issuance of SSW, SSR, SWD, RCR, MSR, MSW command	SSW, SSR, SWD, RCR, MSR, MSW
Data Transfer Error	STRB0	During the data transfer between the drive and the MPU or memory, an overrun or underflow occurs.	Reading of STRB	SSW, SSR, SWD, RCR, MSR, MSW, FFR, FFW
CRC Error	STRB1	In checking the CRC of an ID field or a Data field, a CRC error occurs.	Reading of STRB	SSW, SSR, SWD, RCR, MSR, MSW, (FFR)
Data Mark Undetected	STRB2	If data address mark (FB or F8) is not detected within 32 bytes after the address ID field has been detected.	Upon issuance of SSW, SSR, SWD, RCR, MSR, MSW Command	SSR, RCR, MSR
Sector Address Undetected	STRB3	(1) The sector address that coincides with the contents of the SAR does not exist on the track. (2) A CRC error occurs in checking the ID field.	Reading of STRB	SSW, SSR, SWD, RCR, MSR, MSW
Seek Error	STRB4	During a STZ command, the TKZ input remains low after 83 pulses have been issued on the STP output.	Reading of STRB	STZ
FI	STRB5	File Inoperable input is high.	Reading STRB causes the FIR output to go High. This should reset the FI input.	SSW, SWD, MSW, FFW
Write Error	STRB6	The WPT input is high, and a write operation is executed.	Reading of STRB with either WGT or WPT reset.	SSW, SWD, MSW, FFW
Hard Error	STRB7	During the execution of command (Busy is 1) the RDY input becomes low.	Reading of STRB	All commands



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	50.29	51.31	1.980	2.020
B	14.86	15.62	0.585	0.615
C	2.54	4.19	0.100	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.33	0.008	0.013
K	2.54	4.19	0.100	0.165
L	14.60	15.37	0.575	0.605
M	0°	10°	0°	10°
N	0.51	1.52	0.020	0.060

NOTE:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA (AT SEATING PLANE), AT MAX. MAT'L CONDITION.

CASE 715-02



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	51.82	52.32	2.040	2.060
B	13.72	14.22	0.540	0.560
C	4.57	5.08	0.180	0.200
D	0.36	0.51	0.014	0.020
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	3.05	3.56	0.120	0.140
L	15.24 BSC		0.600 BSC	
M	0°	10°	0°	10°
N	0.51	1.02	0.020	0.040

NOTES:
1. LEADS TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION (DIM "D").
2. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.

CASE 711-02

MOTOROLA

MC6844
1.0 MHz
MC68A44
1.5 MHz
MC68B44
2.0 MHz

DIRECT MEMORY ACCESS CONTROLLER (DMAC)

The MC6844 Direct Memory Access Controller (DMAC) performs the function of transferring data directly between memory and peripheral device controllers. It controls the address and data buses in place of the MPU in bus organized systems such as the M6800 Microprocessor System.

The bus interface of the MC6844 includes select, read/write, interrupt, transfer request/grant, and bus interface logic to allow the data transfer over an 8-bit bidirectional data bus. The functional configuration of the DMAC is programmed via the data bus. The internal structure provides for control and handling of four individual channels, each of which is separately configured. Programmable control registers provide control for the transfer location and length, individual channel control and transfer mode configuration, priority of servicing, data chaining, and interrupt control. Status and control lines provide control to the peripheral controllers.

The mode of transfer for each channel can be programmed as cycle-stealing or a burst transfer mode.

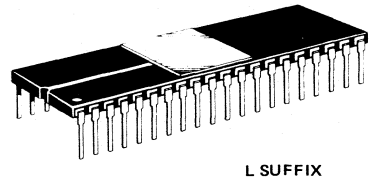
Typical applications would be with the Floppy Disk Controller (FDC) and the Advanced Data Link Controller (ADLC).

- Four DMA Channels, Each Having a 16-Bit Address Register and a 16-Bit Byte Count Register
- 1 M Byte/Sec Maximum Data Transfer Rate
- Selection of Fixed or Rotating Priority Service Control
- Separate Control Bits for Each Channel
- Data Chain Function
- Address Increment or Decrement Update
- Programmable Interrupts and DMA End to Peripheral Controllers

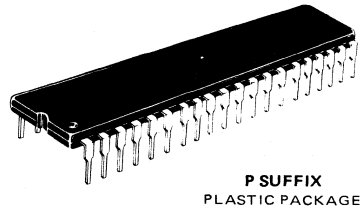
MOS

(N-Channel, Silicon-Gate)

DIRECT MEMORY ACCESS CONTROLLER (DMAC)



L SUFFIX
CERAMIC PACKAGE
CASE 715



P SUFFIX
PLASTIC PACKAGE
CASE 711

FIGURE 1 – M6800 MICROCOMPUTER FAMILY BLOCK DIAGRAM

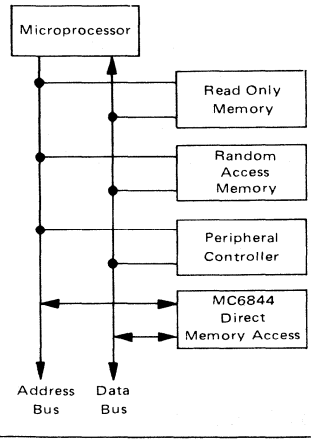
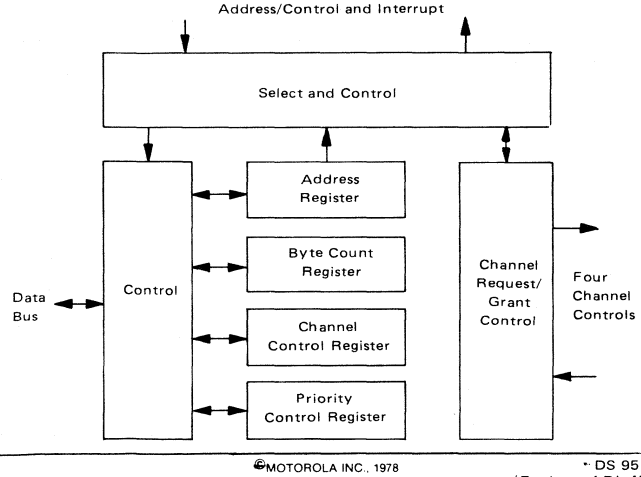


FIGURE 2 – DIRECT MEMORY ACCESS CONTROLLER BLOCK DIAGRAM



* DS 9520
(Replaces ADI-471)

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}*	-0.3 to +7.0	Vdc
Input Voltage	V_{in}*	-0.3 to +7.0	Vdc
Operating Temperature Range MC6844, MC68A44, MC68B44 MC6844C, MC68A44C	T_A	T_L to T_H 0 to +70 -40 to +85	°C
Storage Temperature Range	T_{stg}	-55 to +150	°C
Thermal Resistance Plastic Package Ceramic Package	θ_{JA}	 100 50	°C/W

*In respect to V_{SS}

Permanent device damage may occur if ABSOLUTE MAXIMUM RATINGS are exceeded. Functional operation should be restricted to RECOMMENDED OPERATING CONDITIONS. Exposure to higher than recommended voltages for extended periods of time could affect device reliability.

RECOMMENDED OPERATING CONDITIONS

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	+4.75 to +5.25	Vdc
Input Voltage	V_{IL} V_{IH}	-0.3 to +0.8 2.0 to V_{CC}	Vdc
Operating Ambient Temperature Range	T_A	0 to +70	°C

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ± 5%, V_{SS} = 0, T_A = T_L to T_H unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage		V_{IH}	V_{SS} +2.0	–	V_{CC}	Vdc
Input Low Voltage		V_{IL}	V_{SS} -0.3	–	V_{SS} +0.8	Vdc
Input Leakage Current (V_{in} = 0 to 5.25 V)	TxRQ0-3, φ2 DMA, $\overline{RES}$, DGRNT	I_{in}	–	–	2.5	μAdc
Three-State Leakage Current (V_{in} = 0.4 to 2.4 V)	A0-A15, R/$\overline{W}$ D0-D7	I_{TSI}	-10	–	10	μAdc
Output High Voltage (I_{Load} = -205 μAdc) (I_{Load} = -145 μAdc) (I_{Load} = -100 μAdc)	 D0-D7 A0-15, R/$\overline{W}$ All Others	V_{OH}	 V_{SS} +2.4 V_{SS} +2.4 V_{SS} +2.4	 – – –	 – – –	Vdc
Output Low Voltage (I_{Load} = 1.6 mAdc)		V_{OL}	–	–	V_{SS} +0.4	Vdc
Source Current (V_{in} = 0 Vdc, Figure 10)	$\overline{CS}$/Tx AKB	I_{CSS}	–	10	16	
Power Dissipation		P_D	–	500	1000	mW
Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	 φ2 DMA D0-D7, $\overline{CS}$, A0-A4, R/$\overline{W}$ All Others	C_{in}	 – – –	 – – –	 20 12.5 10	pF
		C_{out}	–	–	12	pF

BUS TIMING CHARACTERISTICS (Load Condition Figure 11)

Characteristic		Symbol	MC6844 Min	MC6844 Max	MC68A44 Min	MC68A44 Max	MC68B44 Min	MC68B44 Max	Unit
READ TIMING (Figure 4)									
Address Setup Time	A0–A4, R/$\overline{W}$, $\overline{CS}$	t_{AS}	160	–	140	–	70	–	ns
Address Input Hold Time	A0–A4, R/$\overline{W}$, $\overline{CS}$	t_{AHI}	10	–	10	–	10	–	ns
Data Delay Time	D0–D7	t_{DDR}	–	320	–	220	–	180	ns
Data Access Time	D0–D7	t_{ACC}	–	480	–	360	–	250	ns
Data Output Hold Time	D0–D7	t_{DHR}	10	–	10	–	10	–	ns
WRITE TIMING (Figure 4)									
Address Setup Time	A0–A4, R/$\overline{W}$, $\overline{CS}$	t_{AS}	160	–	140	–	70	–	ns
Address Input Hold Time	A0–A4, R/$\overline{W}$, $\overline{CS}$	t_{AHI}	10	–	10	–	10	–	ns
Data Setup Time	D0–D7	t_{DSW}	195	–	80	–	60	–	ns
Data Input Hold Time	D0–D7	t_{DHW}	10	–	10	–	10	–	ns

CLOCK TIMING

Characteristic		Symbol	MC6844 Min	MC6844 Max	MC68A44 Min	MC68A44 Max	MC68B44 Min	MC68B44 Max	Unit
ϕ2 DMA (See Figure 4)									
Cycle Time		t_{cyc}	1000	–	666	–	500	–	ns
Pulse Width–High		PW_H	450	–	280	–	220	–	ns
Low		PW_L	430	–	280	–	210	–	
Rise and Fall Time		$t_{\phi r}$, $t_{\phi f}$	–	25	–	25	–	25	ns
DMA TIMING (Load Condition Figure 11)									
Tx RQ Setup Time (Figure 5)									
ϕ2 DMA Rising Edge		t_{TQS1}	120	–	120	–	120	–	ns
ϕ2 DMA Falling Edge		t_{TQS2}	210	–	210	–	210	–	
Tx RQ Hold Time (Figure 5)									
ϕ2 DMA Rising Edge		t_{TQH1}	20	–	10	–	10	–	ns
ϕ2 DMA Falling Edge		t_{TQS2}	20	–	10	–	10	–	
DGRNT Setup Time (Fgiure 6)		t_{DGS}	155	–	125	–	115	–	ns
DGRNT Hold Time (Figure 6)		t_{DGH}	10	–	10	–	10	–	ns
Address Output Delay Time (Figure 15)	A0–A15, R/$\overline{W}$, Tx$\overline{STB}$	t_{AD}	–	270	–	180	–	150	ns
Address Output Hold Time (Figure 15)	A0–A15, R/$\overline{W}$	t_{AHO}	30	–	20	–	20	–	ns
	Tx$\overline{STB}$		35	–	30	–	30	–	
Address Three-State Delay Time (Figure 8)	A0–A15, R/$\overline{W}$	t_{ATSD}	–	720	–	460	–	370	ns
Address Three-State Recovery Time (Figure 8)		t_{ATSR}	–	430	–	280	–	210	ns
Delay Time (Figure 7)	$\overline{DRQH}$, $\overline{DRQT}$	t_{DQD}	–	375	–	250	–	190	ns
Tx AK Delay Time									
ϕ2 DMA Rising Edge (Figure 7)		t_{TKD1}	–	400	–	310	–	250	ns
DGRNT Rising Edge (Figure 10)		t_{TKD2}	–	190	–	160	–	145	
$\overline{IRQ}$/$\overline{DEND}$ Delay Time									
ϕ2 DMA Falling Edge (Figure 8)		t_{DED1}	–	300	–	250	–	210	ns
DGRNT Rising Edge (Figure 10)		t_{DED2}	–	190	–	160	–	145	

FIGURE 3 – EXPANDED BLOCK DIAGRAM

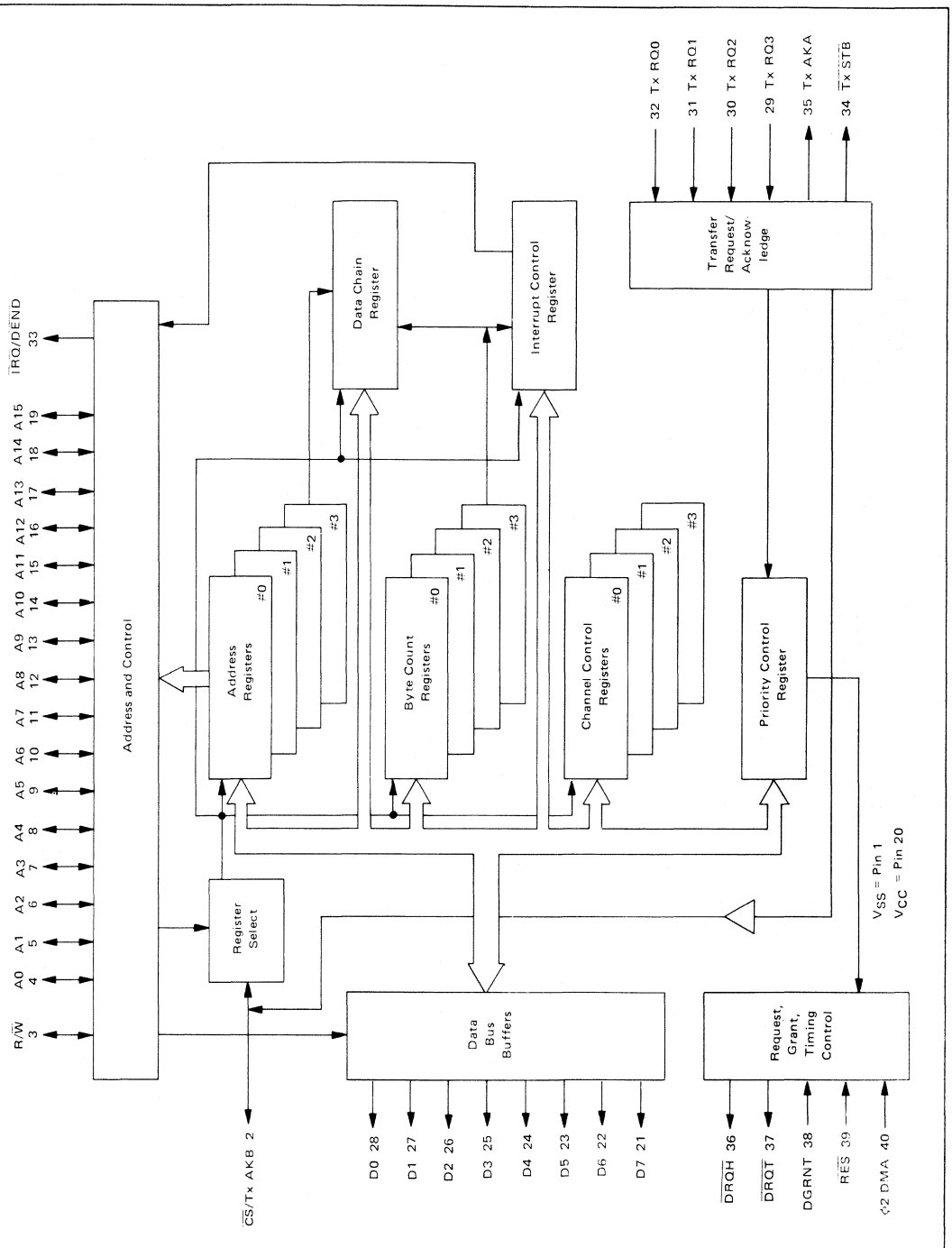


FIGURE 4 – READ/$\overline{\text{WRITE}}$ OPERATION SEQUENCE

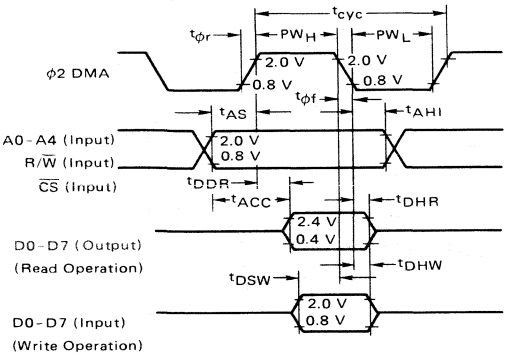


FIGURE 5 – Tx RQ INPUT TIMING

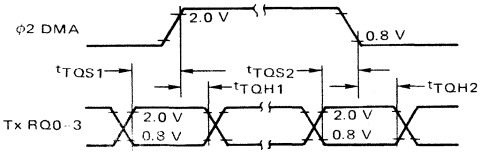


FIGURE 6 – DGRNT INPUT TIMING

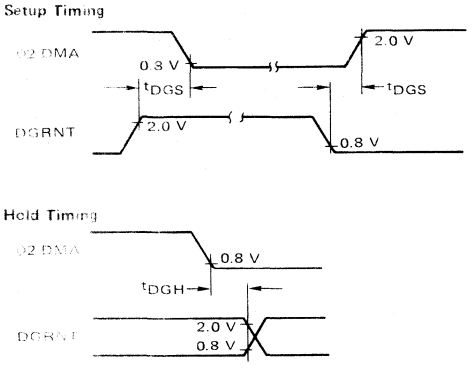


FIGURE 7 – $\overline{\text{DRQH}}$, $\overline{\text{DRQT}}$, Tx AK OUTPUT TIMING

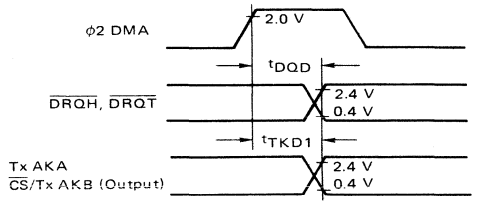


FIGURE 8 – ADDRESS, $\overline{\text{IRQ}}$/$\overline{\text{DEND}}$ OUTPUT TIMING

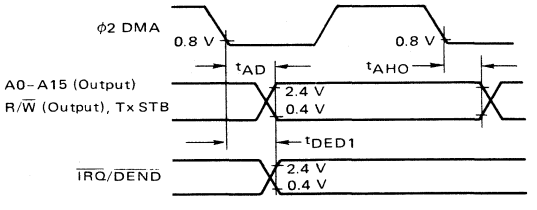


FIGURE 9 – ADDRESS THREE-STATE TIMING

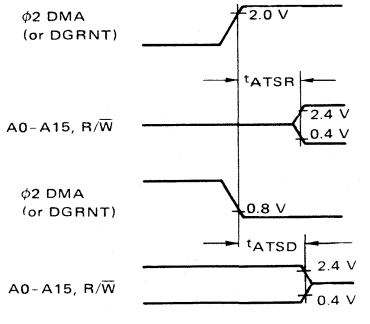


FIGURE 10 – Tx AKB, $\overline{\text{IRQ}}$/$\overline{\text{DEND}}$ OUTPUT TIMING FROM DGRNT INPUT

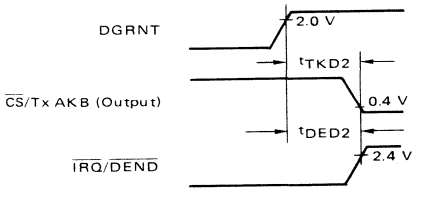
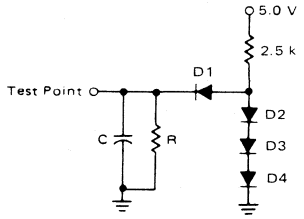
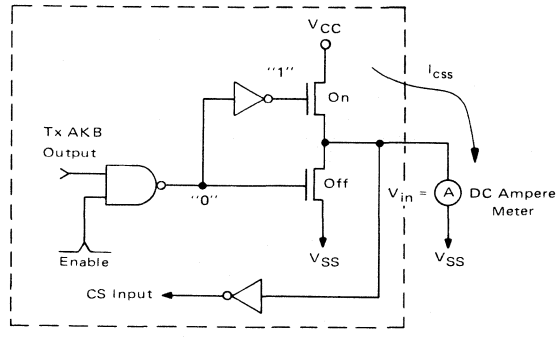


FIGURE 11 – TEST LOADS



Test Pin	C = pF	R = kΩ
D0-D7	130	11.7
A0-A15, R/$\overline{W}$	90	16.5
$\overline{CS}$/Tx AKB	50	24
Others	30	24

FIGURE 12 – $\overline{CS}$/Tx AKB SOURCE CURRENT TEST CIRCUIT



DEVICE OPERATION

The DMAC has fifteen addressable registers, eight of these are sixteen bits in length. Each channel has a separate Address Register and a Byte Count Register, each of which is sixteen bits. There are also four Channel Control Registers. The three General Control Registers common to all four channels are the Priority Control Register, the Interrupt Control Register, and the Data Chain Register.

To prepare a channel for DMA, the Address Registers must be loaded with the starting memory address and the Byte Count Register loaded with the number of bytes to be transferred. The bits in the Channel Control Register establish the direction of the transfer, the mode, and the address increment or decrement after each cycle. Each channel can be set for one of three transfer modes: Three-State Control (TSC) Steal, Halt Steal, or Halt Burst. Two read-only status bits in the Channel Control Register indicate when the channel is busy transferring data and when the DMA transfer is completed.

The Priority Control Register enables the transfer requests from the peripheral controllers and establishes either a fixed priority or rotating priority scheme of servicing these requests.

When the DMA transfer for a channel is complete (the Byte Count Register is zero), a DMA End signal is directed to the peripheral controller and an $\overline{IRQ}$ goes to the MPU. Enabling of these interrupts is done in the Interrupt Control Register. The DMA End/$\overline{IRQ}$ flag bit is read from this register.

Chaining of data transfers is controlled by the Data Chain Register. When enabled, the contents of the Address and Byte Count Registers for channel #3 are put into the registers of the channel selected for chaining when its Byte Count Register becomes zero. This allows for repetitively reading or writing a block of memory.

During the DMA mode, the DMAC controls the address bus and data bus for the system as well as providing the R/$\overline{W}$ line and a signal to be used as VMA. When a peripheral device controller desires a DMA transfer, it is requested by a Transfer Request. Assuming this request is enabled and meets the test of highest priority, the DMAC will issue a DMA Request. When the DMAC receives the DMA Grant, it gives a Transfer Acknowledge to the peripheral device controller, at which time the data is transferred. When the channel's Byte Count Register equals zero, the transfer is complete and a DMA End is given to the peripheral device controller, and an $\overline{IRQ}$ is given to the MPU.

Initialization

During a power-on sequence, the DMAC is reset via the $\overline{RES}$ input. All registers, with the exception of the Address and Byte Count Registers, are set to a logic "0" state. This disables all requests and the Data Chain function while masking all interrupts. The Address, Byte Count, and Channel Control Registers must be programmed before the respective transfer request bit is enabled in the Priority Control Register.

Transfer Modes

There are three ways in which a DMA transfer may be done. The one used is determined by the data transfer rate required, the number of channels attached, and the hardware complexity allowable. Refer to Figures 13 through 15.

Two of the modes, TSC Steal and Halt Steal, are done by cycle- stealing from the MPU. The Three-State Control (TSC) Steal mode is initiated by the DMAC bringing the $\overline{DRQT}$ line LOW. This line goes to the system clock driver which returns a HIGH on DGRNT on the rising edge of the system $\phi 1$ clock. The DGRNT signal must cause the address control and data lines to go to the high impedance state. The DMAC now supplies the address from the Address Register of the channel requesting. It also supplies the R/$\overline{W}$ signal as determined from the Channel Control Register. After one byte is transferred, control is restored to the MPU. This method stretches the $\phi 1$ and $\phi 2$ clocks while the DMAC uses the memory.

The second method of cycle-stealing is the Halt Steal mode. This method actually halts the MPU instead of stretching the $\phi 1$ clock for the transfer period. This mode is initiated by the DMAC bringing the $\overline{DRQH}$ line LOW. This line connects to the MPU $\overline{HALT}$ input. The MPU Bus Available (BA) line is the DGRNT input to the DMAC. While the MPU is halted, its Address Bus, Data Bus, and R/$\overline{W}$ are in the high impedance state. The DMAC now supplies the address and R/$\overline{W}$ line. After one byte is transferred, the $\overline{HALT}$ line is returned HIGH and the MPU regains control. In this mode, the MPU stops internal activity and is removed from the system while the DMAC uses the memory.

The third mode of transfer is the Halt Burst mode. This mode is similar to the Halt Steal mode, except that the transfer does not stop with one byte. The MPU is halted while an entire block of data is transferred. When the channel's Byte Count Register equals zero, the transfer is complete and control is returned to the MPU. This mode gives the highest data transfer rate, at the expense of the MPU being inactive during the transfer period.

FIGURE 13 – TSC STEAL MODE

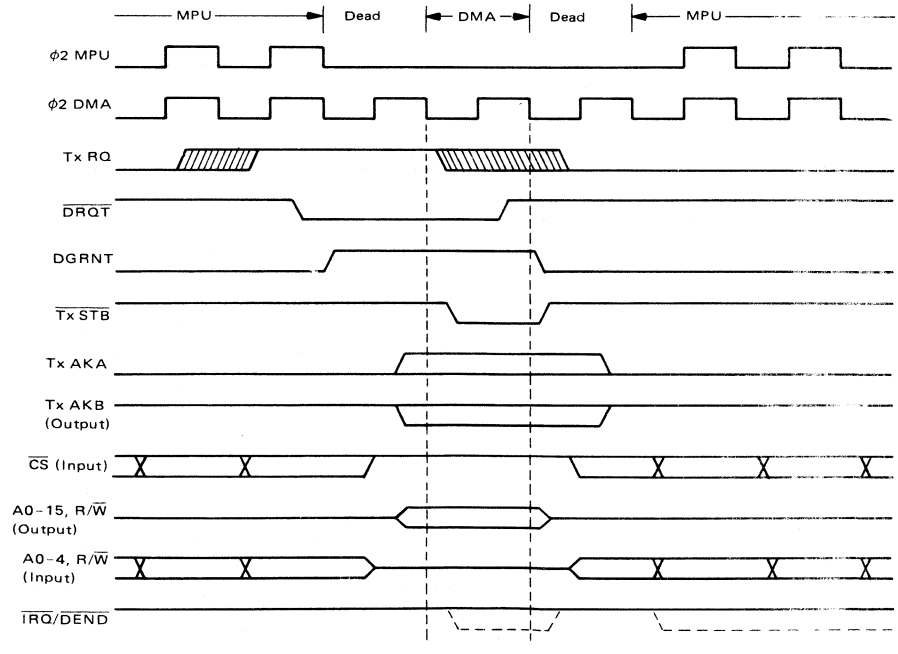


FIGURE 14 – HALT STEAL MODE

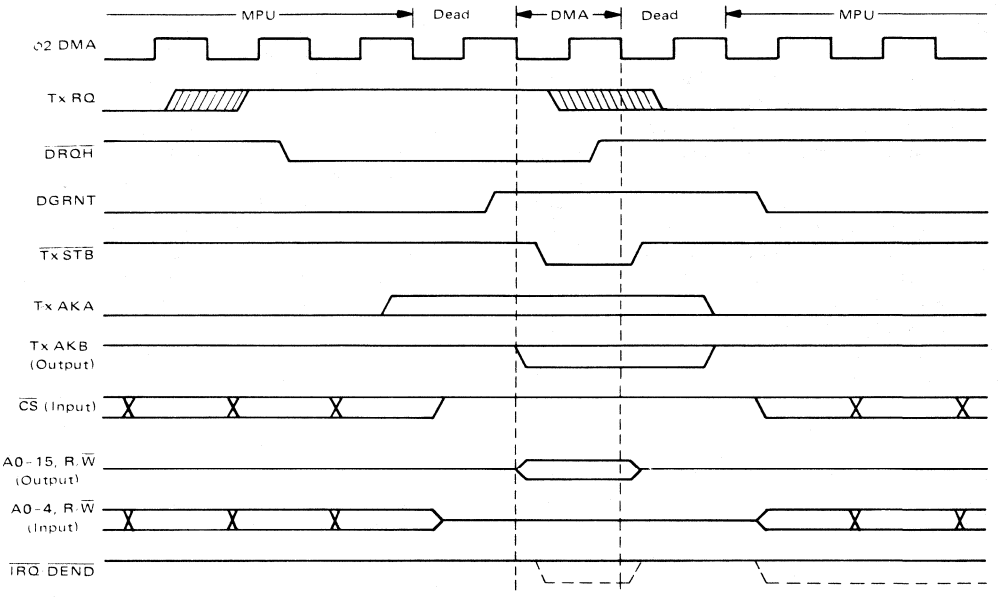
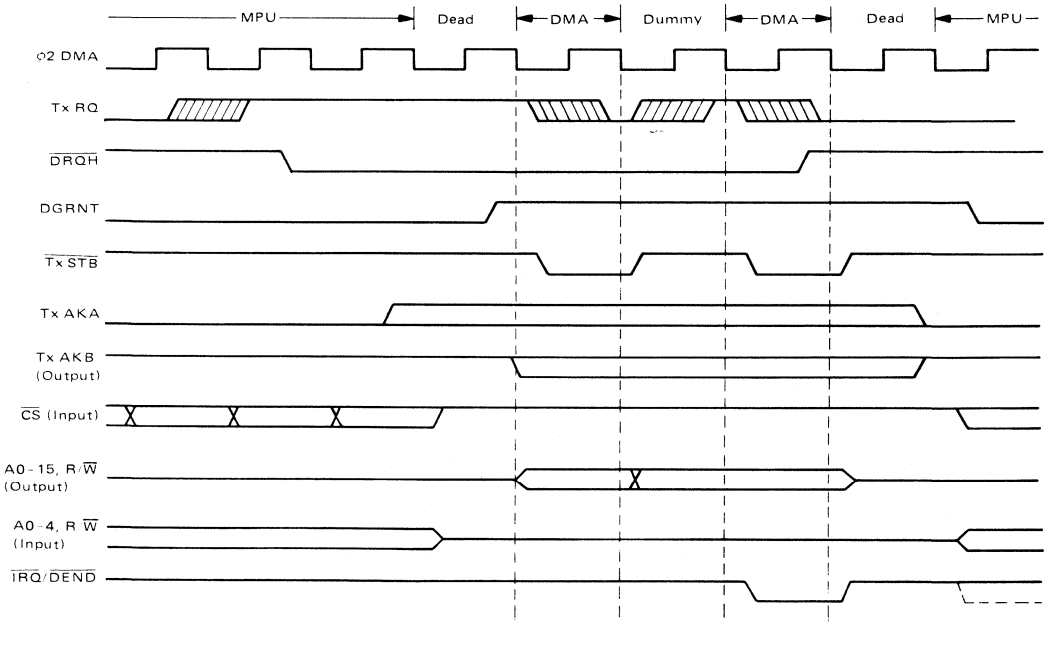


FIGURE 15 -- HALT BURST MODE



INPUT/OUTPUT FUNCTIONS

DMAC Interface Signals for the MPU

The DMAC interfaces with the M6800 MPU through the eight-bit bidirectional data bus, the Chip Select line, five address lines, an Interrupt Request line, the Read/Write line, and the Reset line. These signals, in conjunction with the M6800 VMA output, permit the MPU to have access to the DMAC. Four other lines associated with the MPU and the clock driver are the DRQT, DRQH, DGRNT, and the ϕ2 DMA.

Bidirectional Data (D0-D7). The Bidirectional Data lines (D0-D7) allow for data transfer between the DMAC and the MPU. The data bus output drivers are three-state devices that remain in the high impedance state except when the MPU performs DMAC read operations.

Chip Select/Transfer Acknowledge B ($\overline{CS}$/Tx AKB). This line is multiplexed, serving both as an input and an output. $\overline{CS}$/Tx AKB is an output in the four-channel mode during the DMA transfer. At all other times, it is a high impedance TTL compatible input used to address the DMAC. The DMAC is selected when $\overline{CS}$/Tx AKC is LOW. VMA must be used in generating this input to insure that false selects will not occur. Transfers of data to and from the DMAC are then performed under the control of the ϕ2 DMA, Read/Write, and A0-A4 address lines. In the four-channel mode when Tx AKB is needed, the $\overline{CS}$ gate must have an open-collector output (a pull-up resistor should not be used). In the two-channel mode, $\overline{CS}$/Tx AKB is always an input.

Address Lines A0-A4 (A0-A4). Address lines A0-A4 are both input and output lines. In the MPU mode, these are high impedance inputs used to address the DMAC registers. In the DMA mode, these lines are outputs which are set to the contents of the Address Register of the channel being processed.

Interrupt Request/DMA End ($\overline{IRQ}$/$\overline{DEND}$). Interrupt Request/DMA End is a TTL compatible, active LOW output that is used to interrupt the MPU and to signal the peripheral controller that the data block transfer has ended. If the Interrupt has been enabled, the $\overline{IRQ}$/$\overline{DEND}$ line will go LOW after the last DMA cycle of a transfer. An open collector gate must be connected to DGRNT and $\overline{IRQ}$/$\overline{DEND}$ to prevent false interrupts from the DEND signal when interrupts are not enabled. See Figure 14.

Read/Write (R/$\overline{W}$). Read/Write is a TTL compatible line that is a high impedance input in the MPU mode and an output in the DMA mode. In the MPU mode, it is used to control the direction of data flow through the DMAC's input/output data bus interface. When Read/Write is HIGH (MPU read cycle) and the chip is selected, DMAC data output buffers are turned on and a selected register is read. When it is LOW, the DMAC output drivers are turned off and the MPU writes into a selected register.

In the DMA mode, Read/Write is an output to drive the memory and peripheral controllers. Its state is determined by bit 0 of the Channel Control Register for the channel being serviced. When Read/Write is HIGH, the memory is read and the data from the memory is written into the peripheral controller. When it is LOW, the peripheral controller is read and its data stored in the memory. In the DMA mode, the DMAC data buffers are off, so data is not available on the Data Bus (D0-D7).

Reset ($\overline{RES}$). The $\overline{RES}$ input provides a means of resetting the DMAC from an external source. In the LOW state, the $\overline{RES}$ input causes all registers, with the exception of the Address and Byte Count Registers, to be reset to the logic "0" state. This disables all transfer requests, masks all interrupts, disables the data chain function, and puts each Channel Control Register into the condition of memory write, Halt Steal transfer mode, and address increment.

Transfer Signals to the MPU

Two DMA request output lines and a DMA Grant input line, together with the system clock, synchronize the DMAC with the MPU system.

DMA Request Three-State Control Steal ($\overline{DRQT}$). This active LOW output requests a DMA transfer for a channel configured for the TSC Steal transfer mode. This line is connected to the system clock driver, requesting a ϕ1 clock stretch. It will remain in the LOW state until the transfer has begun.

DMA Request Halt Steal ($\overline{DRQH}$). This active LOW output requests a DMA transfer for a channel programmed for the Halt Steal or Halt Burst mode transfer. This line is connected directly to the MPU $\overline{HALT}$ input and remains LOW until the last byte has begun to be transferred.

DMA Grant (DGRNT). This is a high impedance input to the DMAC, giving it control of the system busses. For the TSC Steal mode, the signal comes from the system clock drive circuit (DMA Grant), indicating that the clock is being stretched. For either of the Halt modes, this signal is the Bus Available from the MPU, indicating that the MPU has halted and turned control of its busses over to the DMAC. For a design involving TSC Steal and Halt mode transfers, this input must be the OR of the clock driven DMA Grant and the MPU BA.

ϕ2 DMA. Transferring in and out of the DMAC registers, sampling of channel request lines and gating of other control signals to the system is done internally in conjunction with the ϕ2 DMA high impedance input. This input must be the system memory clock (non-stretched ϕ2 clock).

Transfer Signals From the Peripheral Controller

Transfer Request (Tx RQ0-3). Each of the four channels has its own high impedance input request for transfer line. The peripheral controller requests a transfer by setting its Tx RQ line HIGH (a logic "1"). The lines are sampled according to the priority and enabling established in the Priority Control Register. In the Steal mode and the first byte of the Halt Burst mode, the Tx RQ signals are tested on the positive edge of $\phi 2$ DMA and the highest priority channel is strobed. Once strobed, the Tx RQs are not tested until that channel's data transfer is finished. In the succeeding bytes of the Halt Burst mode transfer, the Tx RQ is tested on the negative edge of $\phi 2$ DMA, and data is transferred on the next $\phi 2$ DMA cycle if Tx RQ is HIGH.

Transfer Signals to the Peripheral Controller

Two encoded lines select the channel to be serviced. A strobe line acknowledges the request and performs the transfer. The DMA End line signals to the peripheral controller that the DMA transfer is completed.

Transfer Acknowledge A (Tx AKA). The Transfer Acknowledge A (Tx AKA) is a TTL compatible output used in conjunction with the $\overline{CS}$/Tx AKB line to select the channel to be strobed for transfer and to give the DMA End Signal. In the two-channel mode, only Tx AKA is used to select channel =0 or channel =1, and $\overline{CS}$/Tx AKB is always an input.

Chip Select/Transfer Acknowledge B ($\overline{CS}$/Tx AKB). In the DMA mode, this dual purpose line is encoded together with Tx AKA to select the channel being serviced. Table 1 shows the encoding order.

TABLE 1 – ENCODING ORDER

$\overline{CS}$/Tx AKB	Tx AKA	Channel #
0	0	0
0	1	1
1	0	2
1	1	3

$\overline{\text{Transfer Strobe}}$ ($\overline{\text{Tx STB}}$). The $\overline{\text{Transfer Strobe}}$ causes acknowledgement to be given to the peripheral controller and transfers the data to or from the memory. This line is also intended to be the VMA signal for the system in the DMA mode. In a one-channel system, $\overline{\text{Tx STB}}$ may be inverted and run to the peripheral controller's Acknowledge input. In a two- or four-channel system, $\overline{\text{Tx STB}}$ enables the decode of Tx AKA and $\overline{CS}$/Tx AKB to select the device controller to be acknowledged.

$\overline{\text{Interrupt Request}}$/DMA $\overline{\text{End}}$ ($\overline{IRQ}$/$\overline{DEND}$). In the DMA mode, this dual purpose line is LOW for the last byte of transfer, indicating a DMA End. This occurs when the Byte Count register decrements to zero.

This line, through the decode of Tx AKA and $\overline{CS}$/Tx AKB, can be used to strobe a DMA End to each device controller.

Address Lines to the Memory

Address Lines A0-A15 (A0-A15). These output lines are in the high impedance state during the MPU mode. In the DMA mode, these lines are outputs which are set to the contents of the Address Register of the channel being processed.

THE DMAC REGISTERS

All of the fifteen registers in the DMAC are Read/$\overline{\text{Write}}$ registers, although some of the bits are read only status bits.

Address Registers

Each channel has its own individual 16-bit Address Register. Before a DMA transfer is begun, the starting address for the transfer must be loaded into the Address Register. Depending on the state of the Channel Control Register, bit 3, the Address Register will be decremented or incremented after each byte of transfer.

Byte Count Registers

Each channel also has its own Byte Count Register. Before the DMA transfer, this register must be loaded with the number of bytes to be transferred. Since it is 16 bits in length, the transfer can be up to 65,536 bytes of data. The Byte Count Register is decremented at the beginning of each DMA cycle.

Channel Control Registers

The control of each channel's DMA transfer is programmed into its Channel Control Register. Bits 4 and 5 are unused.

Read/$\overline{\text{Write}}$ (R/$\overline{W}$), Bit 0. The direction of the DMA transfer is controlled by this bit. When it is HIGH, the peripheral controller reads the memory. When it is LOW, the transfer will be in the opposite direction, thus writing into the memory. The system R/$\overline{W}$ line is in the same state as this R/$\overline{W}$ bit in the DMA mode. The device controller must change the sense of its R/$\overline{W}$ input during the DMA mode.

Burst/Steal, Bit 1. This bit, along with bit 2, selects the mode of the DMA transfer. With bit 1 HIGH, the Burst mode is selected. A LOW selects the Steal mode. Table 2 shows the mode selection.

TSC/Halt, Bit 2. Bit 2 helps select the mode of DMA transfer. When this bit is HIGH, the TSC mode is selected.

When LOW, the Halt mode is selected. Table 2 shows the mode selection. A TSC Burst mode is illegal for M6800-family processors due to restrictions on $\phi 1$ clock stretching for these products.

TABLE 2 – MODE SELECT

Bit 2	Bit 1	DMA Transfer Mode
0	0	Halt Steal
0	1	Halt Burst
1	0	TSC Steal
1	1	(Illegal)

Address Up/Down, Bit 3. Bit 3 controls the change in the Address Register for each DMA cycle. If this bit is LOW, the Address Register will be incremented each time the Byte Count Register decrements. If the bit is HIGH, the Address Register will be decremented.

Busy/Ready Flag, Bit 6. The Busy/Ready Flag is a read only status bit that indicates a DMA transfer is in process on that channel. This bit goes HIGH at the beginning of the transfer and remains so until the $\overline{IRQ}/\overline{DEND}$ has been LOW for one cycle (DMA End). The bit is then reset and the channel can again be configured for a transfer.

DMA End Flag (DEND), Bit 7. The DEND bit indicates a DMA block transfer has ended. This bit is set at the same time the Busy/Ready Flag is reset. The DEND bit is reset by the MPU reading the Channel Control Register. This bit causes an interrupt if enabled in the Interrupt Control Register.

PRIORITY CONTROL REGISTER

The enabling and prioritizing of the transfer requests (Tx RQs) are done in the Priority Control Register. Bits 4 through 6 are unused.

Request Enable (RE0-3), Bits 0-3. The four channels are individually enabled by setting the respective RE bit HIGH. A LOW on any of these bits disables recognition of the transfer request for that channel. The bit number equals the channel number (i.e., bit 2 equals channel #2).

Rotate Control, Bit 7. The DMAC priority service routine is selected by this Rotate Control bit. When it is LOW, the fixed mode is selected. Channel #0 has the highest priority, channel #1 the next highest, and so on down. When this bit is HIGH, a rotating routine is used. This routine states that initially it will be the same as in the fixed mode. But once a channel has been serviced, it moves to the lowest priority and those that were below it advance to the next highest priority.

Interrupt Control Register

An interrupt is caused by a channel completing its DMA block transfer. DEND (Channel Control Register, Bit 7) flags this condition for each channel. Bits 4 through 6 are unused.

DEND IRQ Enable (DIE0-3) Bits 0-3. Each channel is separately enabled to cause the interrupt. A HIGH enables an interrupt from the channel; a LOW masks the interrupt. The bit number equals the channel number (i.e., bit 2 equals channel #2).

DEND IRQ Flag, Bit 7. This read only bit indicates an IRQ is requested of the MPU when it is HIGH. If the interrupt is enabled (DIE = "1") when a channel's DEND flag (Channel Control Register, bit 7) goes HIGH, the DEND IRQ Flag bit also goes HIGH. It is reset by the MPU reading the Channel Control Register that caused the interrupt.

Data Chain Register

Repetitive reading or writing of a block of memory can be done in the data chain function. A DMA transfer cannot be active on channel #3 during the data chain. Bits 4 through 7 are unused.

Data Chain Enable (DCE), Bit 0. The data chain function is enabled when this bit is HIGH.

Data Chain Channel Selects A, B (DCA, DCB), Bits 1 and 2. The channel updated by data chaining is selected by bits 1 and 2, according to the order shown in Table 3.

TABLE 3 – CHANNEL SELECT

DCB Bit 2	DCA Bit 1	Channel #
0	0	0
0	1	1
1	0	2
1	1	(Illegal)

The data chain function is performed by transferring the contents of Channel #3 Address and Byte Count Registers into the respective registers of the channel selected by bits 1 and 2. This transfer is done during the cycle of $\phi 2$ DMA following the Byte Count Register having decremented to zero.

Two/Four Channel Select (2/4), Bit 3. The DMAC is configured to handle two or four channels by bit 3. This bit HIGH selects the 4-channel mode. In this mode, the $\overline{CS}$/Tx AKB becomes a chip select in the MPU mode and a Transfer Acknowledge B for the DMA mode.

With bit 3 LOW, the 2-channel mode is selected, and the $\overline{CS}$/Tx AKB line is always a chip select, both for the MPU and the DMA mode.

TABLE 4 – DMAC PROGRAMMING MODEL

Register	Address (Hex)	Register Content							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Channel Control	1x*	DMA End Flag (DEND)	Busy/Ready Flag	Not Used	Not Used	Address Up/Down	TSC/ Halt	Burst/ Steal	Read/Write (R/$\overline{W}$)
Priority Control	14	Rotate Control	Not Used	Not Used	Not Used	Request Enable #3 (RE3)	Request Enable #2 (RE2)	Request Enable #1 (RE1)	Request Enable #0 (RE0)
Interrupt Control	15	DEND IRQ Flag	Not Used	Not Used	Not Used	DEND IRQ Enable #3 (DIE3)	DEND IRQ Enable #2 (DIE2)	DEND IRQ Enable #1 (DIE1)	DEND IRQ Enable #0 (DIE0)
Data Chain	16	Not Used	Not Used	Not Used	Not Used	Two/Four Channel Select (2/4)	Data Chain Channel Select B	Data Chain Channel Select A	Data Chain Enable

*The x represents the binary equivalent of the channel desired.

Channel Control Register

DEND — **Bit 7** – Is set at end of DMA block transfer; reset by MPU reading the Channel Control Register.

Busy/Ready Flag — **Bit 6** – Status bit set when in transfer; cleared after DMA End.

Address Up/Down — **Bit 3** – HIGH = decrement Address Register for each byte; LOW = increment.

TSC/Halt — **Bit 2** – HIGH = select TSC mode; LOW = Halt modes.

Burst/Steal — **Bit 1** – HIGH = select Burst mode; LOW = Steal modes.

R/$\overline{W}$ — **Bit 0** – HIGH = device controller reads memory; LOW = write into memory.

Priority Control Register

Rotate Control — **Bit 7** – HIGH = use rotate routine; LOW = fixed: 0, 1, 2, 3 priority.

RE0-3 — **Bits 0-3** – HIGH = enable transfer request for the channel; LOW = request disabled.

Interrupt Control Register

DEND IRQ Flag — **Bit 7** – This Flag is set by DENDs in Channel Control Registers when enabled; Reset by reading the Register that caused it to be set.

DIE0-3 — **Bits 0-3** – HIGH = enable IRQ by DEND for the channel; LOW = IRQ masked.

Data Chain Register

Two/Four Channel — **Bit 3** – HIGH = 4-channel mode; LOW = 2-channel.

Data Chain Channel Select — **Bits 2 and 1** – Binary equivalent of channel to be updated by chaining.

Data Chain Enable — **Bit 0** – HIGH = enable Data Chain function; LOW = disabled.

Preparation of a channel for a DMA transfer requires:

1. Load the starting address into the Address Register.
2. Load the number of bytes into the Byte Count Register.
3. Program the Channel Control Register for the transfer characteristics: direction (bit 0), mode (bits 1 and 2), and the address update (bit 3).

The channel is now configured. To enable the transfer request, set the appropriate enable bit (bits 0-3) of the Priority Control Register, as well as the Rotate Control bit.

If an interrupt on DMA End is desired, the enable bit (bits 0-3) of the Interrupt Control Register must be set.

If data chaining for the channel is necessary, it is programmed into the Data Chain Register and the appropriate data must be written into the Address and Byte Count Registers for channel #3.

A comparison of the response times and maximum transfer rates is shown in Table 5. The values shown are for a system clock rate of 1 MHz.

TABLE 5 – TRANSFER RATES

Mode	Response Time (μs)	Maximum Transfer Rate (μs/Byte)
Halt Burst	3.5–15.5*	1
Halt Steal	3.5–15.5*	5–15*
TSC Steal	2.5–3.5	4

*These values will depend on the cycle in process.

The two 8-bit bytes that form the registers in Table 6 are placed in consecutive memory locations, making it very easy to use the MPU index register in programming them.

TABLE 6 – ADDRESS AND BYTE COUNT REGISTERS

Register	Channel	Address (Hex)
Address High	0	0
Address Low	0	1
Byte Count High	0	2
Byte Count Low	0	3
Address High	1	4
Address Low	1	5
Byte Count High	1	6
Byte Count Low	1	7
Address High	2	8
Address Low	2	9
Byte Count High	2	A
Byte Count Low	2	B
Address High	3	C
Address Low	3	D
Byte Count High	3	E
Byte Count Low	3	F

FIGURE 16 – ONE CHANNEL

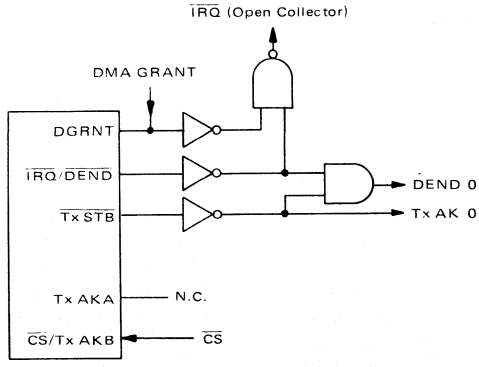


FIGURE 17 – TWO-CHANNEL

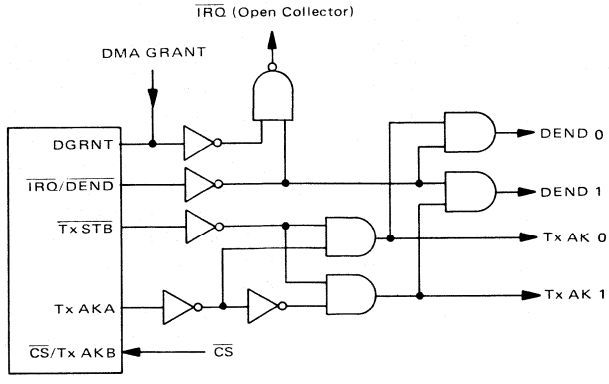
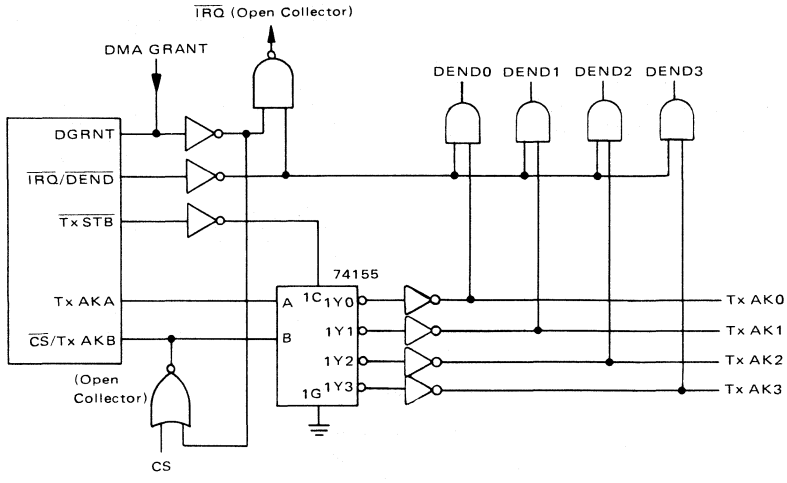


FIGURE 18 – FOUR–CHANNEL



SYSTEM DESCRIPTION

The hardware configuration of the DMAC can be for a 1-, 2-, or 4-channel system. These are shown in Figures 16 through 19.

If the peripheral device controllers do not use the DEND signal, the AND gates generating them are not needed. As mentioned previously, the open collector gate to $\overline{IRQ}$ is to prevent false interrupts from the DEND signal when interrupts are disabled. In the 4-channel mode, the $\overline{CS}$ gate must be open collector so that $\overline{CS}$/Tx AKB can become an output during the DMA cycle.

A typical system design using the MC6800 is shown in Figure 19. System signals are shown on the top; the DMAC control signals are shown on the right. The MC6875 is shown as the clock Generator/Driver. Since the dynamic memory refresh and the DMA control are not separated in the MC6875, it is necessary to have the external Priority Logic to give highest priority to the Refresh Request. Refresh and DMA Grant must not occur during the same cycle.

When using a Halt mode in the DMAC, the MC6875 has no control over the DMA Grant. To prevent a Read or Write during a Refresh cycle, $\phi 2$ DMA must be gated with Refresh Grant.

To be able to use either the TSC Steal and the Halt modes, DGRNT must be the ORed output of Bus Available (BA) and DMA Grant from the Clock Generator/Driver. If only one type is desired, only the one proper line is needed. The MC6875 handles stretching the $\phi 1$ and $\phi 2$ clocks. To three-state the Address and Data Bus and the R/$\overline{W}$ line, the DMA Grant from the Clock Generator/Driver must go to the MC6800's TSC input.

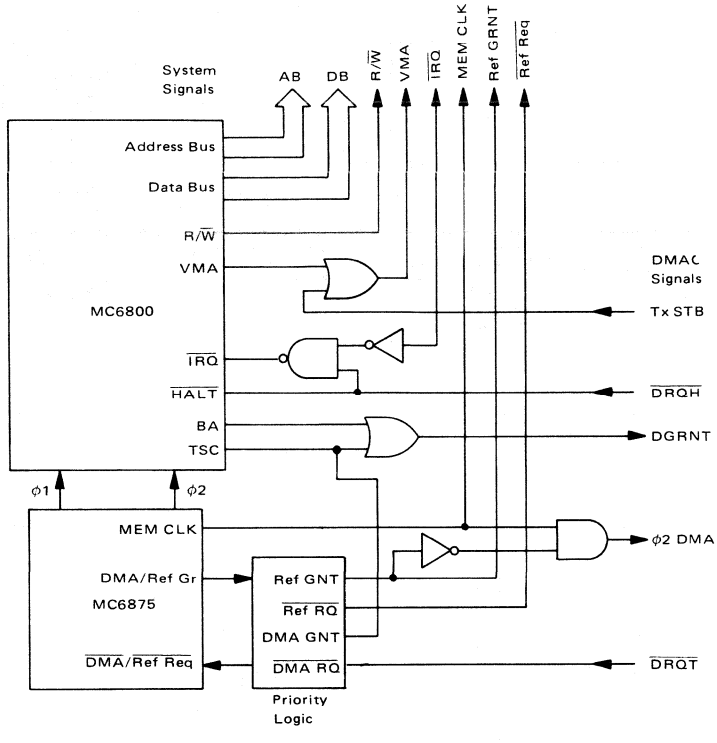
The DMAC's $\overline{DRQH}$ line connects directly to the $\overline{HALT}$ input.

Figure 19 also shows the system $\overline{IRQ}$ is gated with the $\overline{HALT}$ input to get the MPU $\overline{IRQ}$ input. This is necessary only if the Halt modes of transfer are used and the WAI instruction is used along with other system interrupts. If any one of these three is not valid, the system $\overline{IRQ}$ may be connected directly to the MPU $\overline{IRQ}$.

During the DMA cycle, a system VMA signal must be generated by the DMAC. This is done by ORing Tx STB and the MPU VMA line. Another method would be to three-state the MPU VMA by the TSC line input and three-state the Tx STB line by the DGRNT line.

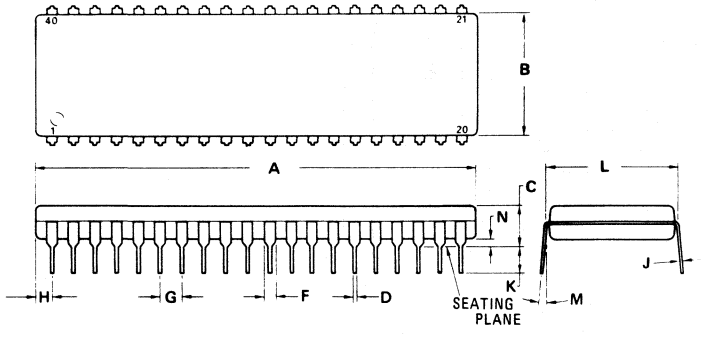
The above gate and line explanations should make it possible to put the DMAC into any system.

FIGURE 19 – TYPICAL SYSTEM



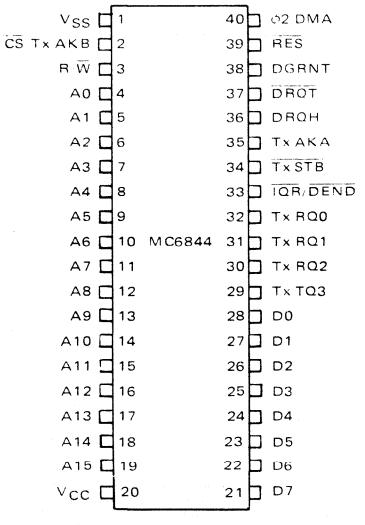
PACKAGE DIMENSIONS

CASE 711-01



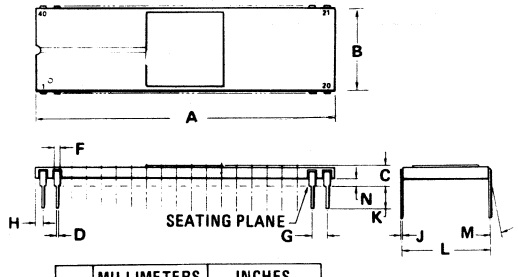
DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	51.82	52.32	2.040	2.060
B	13.72	14.22	0.540	0.560
C	4.57	5.08	0.180	0.200
D	0.36	0.51	0.014	0.020
F	1.02	1.52	0.040	0.060
G	2.41	2.67	0.095	0.105
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	3.68	4.19	0.145	0.165
L	14.99	15.49	0.590	0.610
M	0^0	10^0	0^0	10^0
N	0.51	1.02	0.020	0.040

PIN ASSIGNMENT



PACKAGE DIMENSIONS

CASE 715-02
(CERAMIC)



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	50.29	51.31	1.980	2.020
B	14.86	15.62	0.585	0.615
C	2.54	4.19	0.100	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.33	0.008	0.013
K	2.54	4.19	0.100	0.165
L	14.60	15.37	0.575	0.605
M	–	10^0	–	10^0
N	0.51	1.52	0.020	0.060

NOTE:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA (AT SEATING PLANE), AT MAX. MAT'L CONDITION.

MOTOROLA *Semiconductor Products Inc.*
3501 ED BLUESTEIN BLVD. AUSTIN, TEXAS 78721 • A SUBSIDIARY OF MOTOROLA INC

MOTOROLA

MC6845

Advance Information

CRT CONTROLLER (CRTC)

The MC6845 CRT Controller performs the interface to raster scan CRT displays. It is intended for use in processor-based controllers for CRT terminals in stand-alone or cluster configurations.

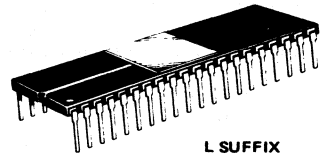
The CRTC is optimized for hardware/software balance in order to achieve integration of all key functions and maintain flexibility. For instance, all keyboard functions, R/W, cursor movements, and editing are under processor control; whereas the CRTC provides video timing and Refresh Memory Addressing.

- Applications include "glass-teletype," smart, programmable, intelligent CRT terminals; video games; information display.
- Alphanumeric, semi-graphic, and full graphic capability.
- Fully programmable via processor data bus. Can generate timing for almost any alphanumeric screen density, e.g. 80 x 24, 72 x 64, 132 x 20, etc.
- Single +5 volt supply. TTL/6800 compatible I/O.
- Hardware scroll (paging or by line or by character)
- Compatible with CPU's and MPU's which provide a means for synchronizing external devices.
- Cursor register and compare circuitry.
- Cursor format and blink are programmable.
- Light pen register.
- Line buffer-less operation. No external DMA required. Refresh Memory is multiplexed between CRTC and MPU.
- Programmable interlace or non-interlace scan.
- 14-bit wide refresh address.

MOS

(N-Channel, Silicon-Gate)

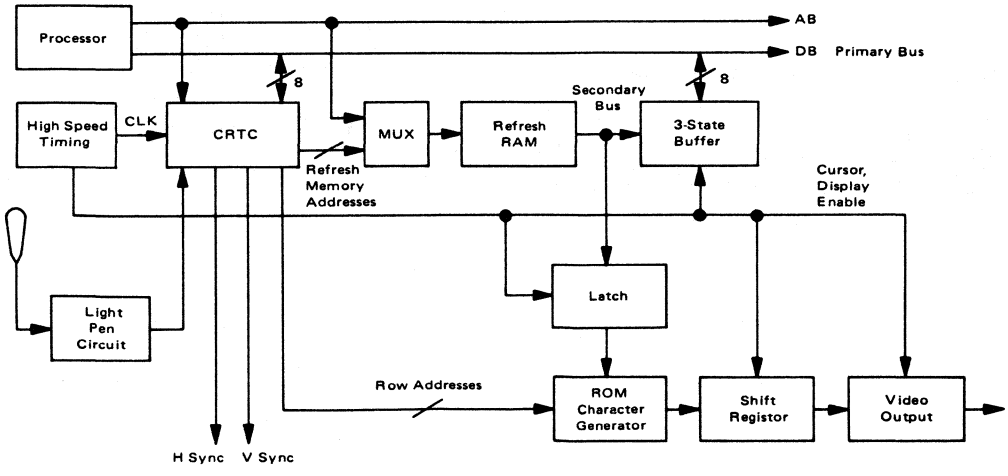
CRT CONTROLLER (CRTC)



L SUFFIX
CERAMIC PACKAGE
CASE 715

NOT SHOWN:
P SUFFIX
PLASTIC PACKAGE
CASE 711

FIGURE 1 – TYPICAL CRT CONTROLLER APPLICATION



This is advance information and specifications are subject to change without notice.

SYSTEM BLOCK DIAGRAM DESCRIPTION

As shown in Figure 1, the primary function of the CRTC is to generate refresh addresses (MA0-MA13), row selects (RA0-RA4), and video monitor timing (HSYNC, VSYNC) and Display Enable. Other functions include an internal cursor register which generates a Cursor output when its contents compare to the current Refresh Address. A light-pen strobe input signal allows capture of Refresh Address in an internal light pen register.

All timing in the CRTC is derived from the Clk input. In alphanumeric terminals, this signal is the character rate. Character rate is divided down from video rate by external High Speed Timing when the video frequency is greater than 3 MHz. Shift Register, Latch, and MUX Control signals are also provided by external High Speed Timing.

The processor communicates with the CRTC through a buffered 8-bit Data Bus by reading/writing into the 18-register file of the CRTC.

The Refresh Memory address is multiplexed between the Processor and CRTC. Data appears on a Secondary Bus which is buffered from the processor Primary Bus. A number of approaches are possible for solving contentions for the Refresh Memory.

1. Processor always gets priority.
2. Processor gets priority access anytime, but can be synchronized by an interrupt to perform accesses only during horizontal and vertical retrace times.
3. Synchronize processor by memory wait cycles.
4. Synchronize processor to character rate (See Figure 2). The 6800 MPU family lends itself to this configuration because it has constant cycle lengths. This method provides zero burden on the processor because there is never a contention for memory. All accesses are *"transparent."*

The secondary data bus concept in no way precludes using the Refresh RAM for other purposes. It looks like any other RAM to the Processor. For example, using Approach 4, a 64K byte RAM Refresh Memory could perform refresh and program storage functions transparently.

MAXIMUM RATINGS

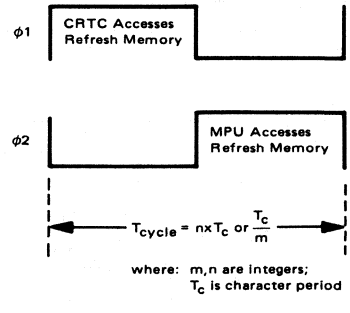
Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}*	-0.3 to +7.0	Vdc
Input Voltage	V_{in}*	-0.3 to +7.0	Vdc
Operating Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	-55 to +150	°C

*With respect to V_{SS} (Gnd).

RECOMMENDED OPERATING CONDITIONS

Characteristics	Symbol	Min	Typ	Max	Unit
Supply Voltage	V_{CC}	4.75	5.0	5.25	Vdc
Input Low Voltage	V_{IL}	-0.3	–	0.8	Vdc
Input High Voltage	V_{IH}	2.0	–	V_{CC}	Vdc

FIGURE 2 – TRANSPARENT REFRESH MEMORY CONFIGURATION TIMING USING 6800 MPU FAMILY



ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ±5%, V_{SS} = 0, T_A = 0 to 70°C unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage		V_{IH}	2.0	–	V_{CC}	Vdc
Input Low Voltage		V_{IL}	-0.3	–	0.8	Vdc
Input Leakage Current		I_{in}	–	1.0	2.5	μAdc
Three-State (V_{CC} = 5.25 V) (V_{in} = 0.4 to 2.4 V)		I_{TSI}	-10	2.0	10	μAdc
Output High Voltage		V_{OH}				Vdc
(I_{load} = -205 μA)	D0-D7		2.4	–	–	
(I_{load} = -100 μA)	Other Outputs		2.4	–	–	
Output Low Voltage (I_{load} = 1.6 mA)		V_{OL}	–	–	0.4	Vdc
Power Dissipation		P_D	–	600	–	mW
Input Capacitance		C_{in}				pF
	D0-D7		–	–	12.5	
	All others		–	–	10	
Output Capacitance	All Outputs	C_{out}	–	–	10	pF
Minimum Clock Pulse Width, Low		P_{WCL}	160	–	–	ns
Minimum Clock Pulse Width, High		P_{WCH}	200	–	–	ns
Clock Frequency		f_c	–	–	2.5	MHz
Rise and Fall Time for Clock Input		t_{cr}, t_{cf}	–	–	20	ns
Memory Address Delay Time		t_{MAD}	–	–	160	ns
Raster Address Delay Time		t_{RAD}	–	–	160	ns
Display Timing Delay Time		t_{DTD}	–	–	300	ns
Horizontal Sync Delay Time		t_{HSD}	–	–	300	ns
Vertical Sync Delay Time		t_{VSD}	–	–	300	ns
Cursor Display Timing Delay Time		t_{CDD}	–	–	300	ns
Light Pen Strobe Minimum Pulse Width		PW_{LPH}	100	–	–	ns
Light Pen Strobe Disable Time		t_{LPD1}	–	–	120	ns
		t_{LPD2}	–	–	0	ns

Note: The light pen strobe must fall to low level before VSYNC pulse rises.

BUS TIMING CHARACTERISTICS

Characteristic	Symbol	Min	Max	Unit
READ/WRITE				
Enable Cycle Time	t_{cycE}	1.0	–	μs
Enable Pulse Width, High	PW_{EH}	0.45	25	μs
Enable Pulse Width, Low	PW_{EL}	0.43	–	μs
Setup Time, $\overline{CS}$ and RS valid to enable positive transition	t_{AS}	160	–	ns
Data Delay Time	t_{DDR}	–	320	ns
Data Hold Time (Read) (write)	t_H	10 10	– –	ns
Address Hold Time	t_{AH}	10	–	ns
Rise and Fall Time for Enable Input	t_{Er}, t_{Ef}	–	25	ns
Data Setup Time	t_{DSW}	195	–	ns
Data Access Time	t_{ACC}	–	480	ns

FIGURE 3 – CRTC TIMING CHART

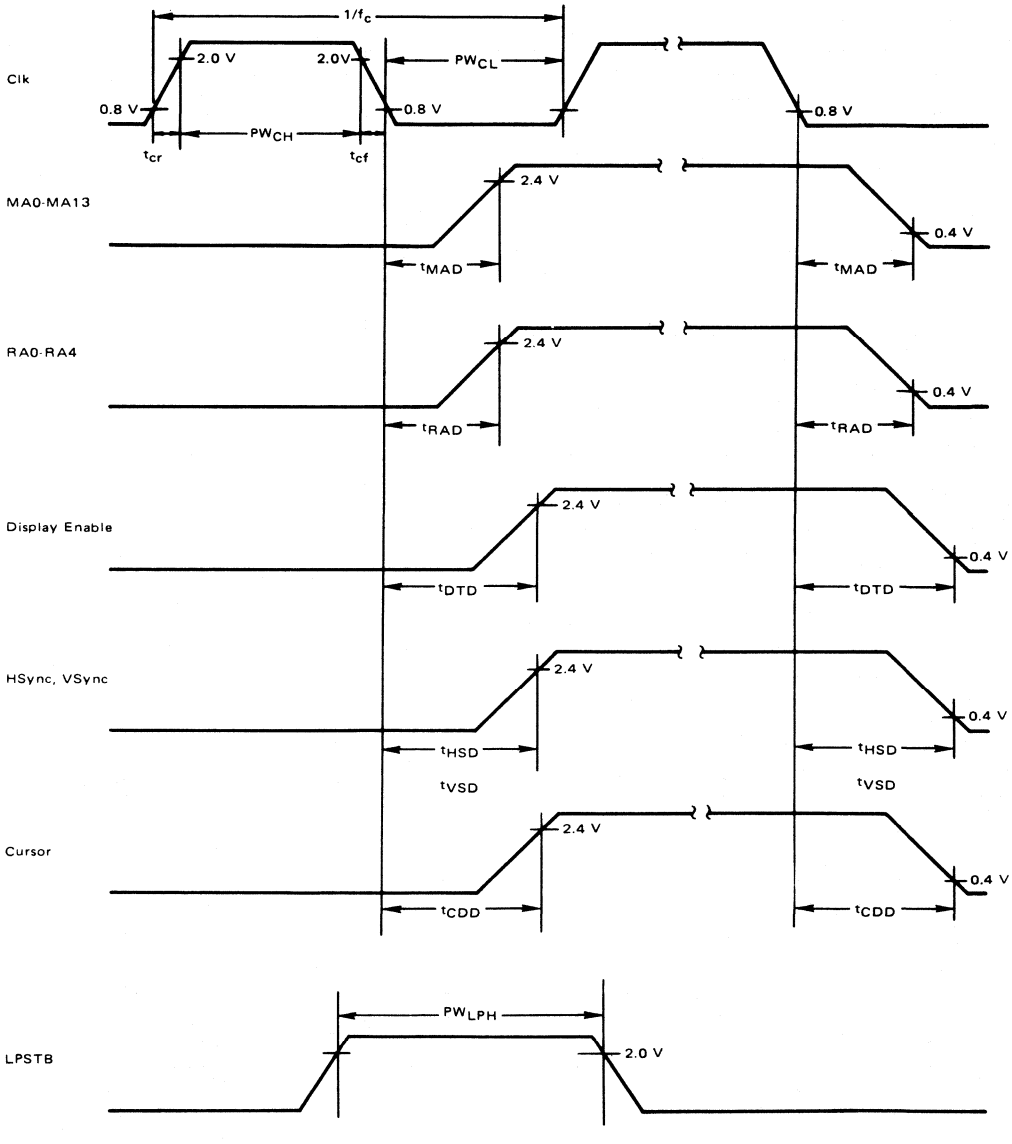


FIGURE 4 – RELATION BETWEEN LPSTB AND REFRESH MEMORY ADDRESS

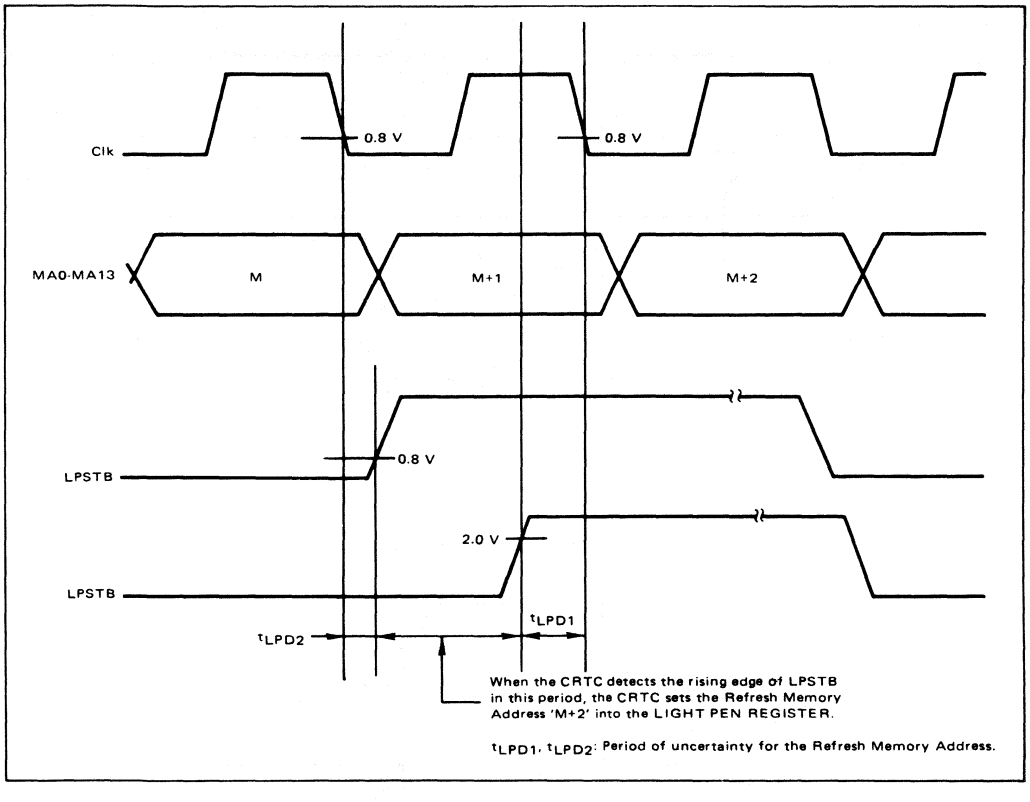
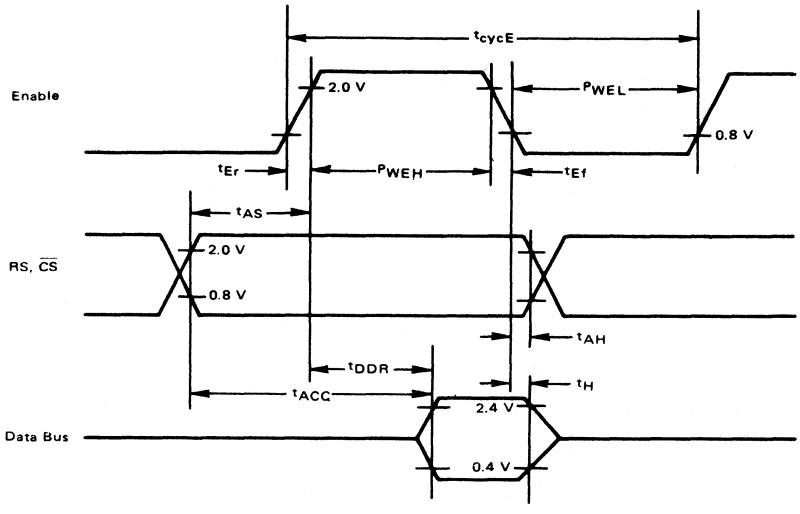


FIGURE 5 – BUS TIMING CHART

5a – Bus Read Timing (Read Information From CRTC)



5b – Bus Write Timing (Write Information Into CRTC)

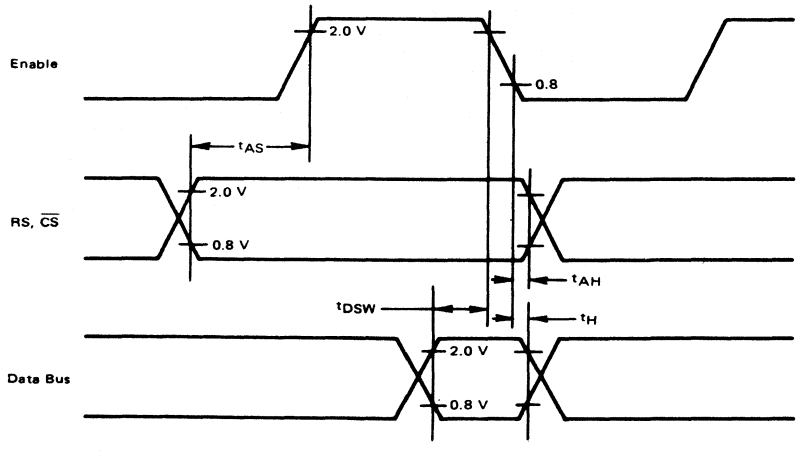


FIGURE 6 – BUS TIMING TEST LOAD

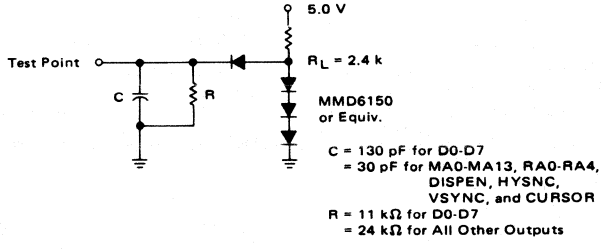
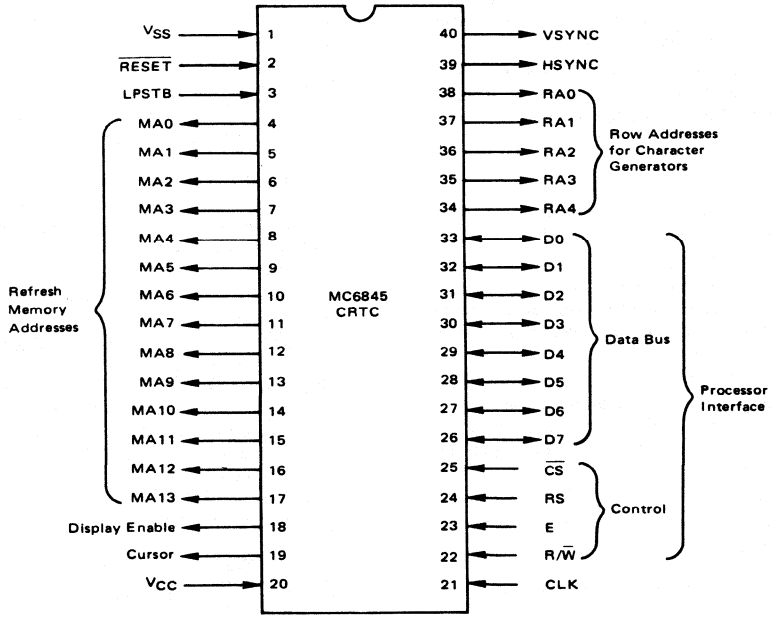


FIGURE 7 – PIN ASSIGNMENT



PIN DESCRIPTION

PROCESSOR INTERFACE

The CRTC interfaces to a processor bus on the bidirectional data bus (D0-D7) using $\overline{CS}$, RS, E, and R/$\overline{W}$ for control signals.

Data Bus (D0-D7) – The bidirectional data lines (D0-D7) allow data transfers between the CRTC internal Register File and the processor. Data bus output drivers are 3-state buffers which remain in the high impedance state except when the processor performs a CRTC read operation. A high level on a data pin is a logical "1."

Enable (E) – The Enable signal is a high impedance TTL/MOS compatible input which enables the data bus input/output buffers and clocks data to and from the CRTC. This signal is usually derived from the processor clock, and the high to low transition is the active edge.

Chip Select ($\overline{CS}$) – The $\overline{CS}$ line is a high impedance TTL/MOS compatible input which selects the CRTC when low to read or write the internal Register File. This signal should only be active when there is a valid stable address being decoded from the processor.

Register Select (RS) – The RS line is a high impedance TTL/MOS compatible input which selects either the Address Register (RS = "0") or one of the Data Registers (RS = "1") of the internal Register File.

Read/Write (R/$\overline{W}$) – The R/$\overline{W}$ line is a high impedance TTL/MOS compatible input which determines whether the internal Register File gets written or read. A write is active low ("0").

CRT CONTROL

The CRTC provides horizontal sync (HS), vertical sync (VS), and Display Enable signals.

Vertical Sync (V SYNC) – This TTL compatible output is an active high signal which drives the monitor directly or is fed to Video Processing Logic for composite generation. This signal determines the vertical position of the displayed text.

Horizontal Sync (H SYNC) – This TTL compatible output is an active high signal which drives the monitor directly or is fed to Video Processing Logic for composite generation. This signal determines the horizontal position of the displayed text.

Display Enable – This TTL compatible output is an active high signal which indicates the CRTC is providing addressing in the active Display Area.

REFRESH MEMORY/CHARACTER GENERATOR ADDRESSING

The CRTC provides Memory Addresses (MA0-MA13) to scan the Refresh RAM. Also provided are Raster Addresses (RA0-RA4) for the character ROM.

Refresh Memory Addresses (MA0-MA13) – These 14 outputs are used to refresh the CRT screen with pages of data located within a 16K block of refresh memory. These outputs drive a TTL load and 30pF. A high level on MA0-MA13 is a logical "1."

Raster Addresses (RA0-RA4) – These 5 outputs from the internal Raster Counter address the Character ROM for the row of a character. These outputs drive a TTL load and 30pF. A high level (on RA0-RA4) is a logical "1."

OTHER PINS

Cursor – This TTL compatible output indicates Cursor Display to external Video Processing Logic. Active high signal.

Clock (CLK) – The CLK TTL/MOS compatible input is used to synchronize all CRT control signals. An external dot counter is used to derive this signal which is usually the character rate in an alphanumeric CRT. The active transition is high to low.

Light Pen Strobe (LPSTR) – This high impedance TTL/MOS compatible input latches the current Refresh Addresses in the Register File. Latching is on the low to high edge and is synchronized internally to character clock.

V_{CC}, Gnd

$\overline{RES}$ – The $\overline{RES}$ input is used to Reset the CRTC. An input low level on $\overline{RES}$ forces CRTC into following status:

(A) All the counters in CRTC are cleared and the device stops the display operation.
(B) All the outputs go down to low level.
(C) Control registers in CRTC are not affected and remain unchanged.

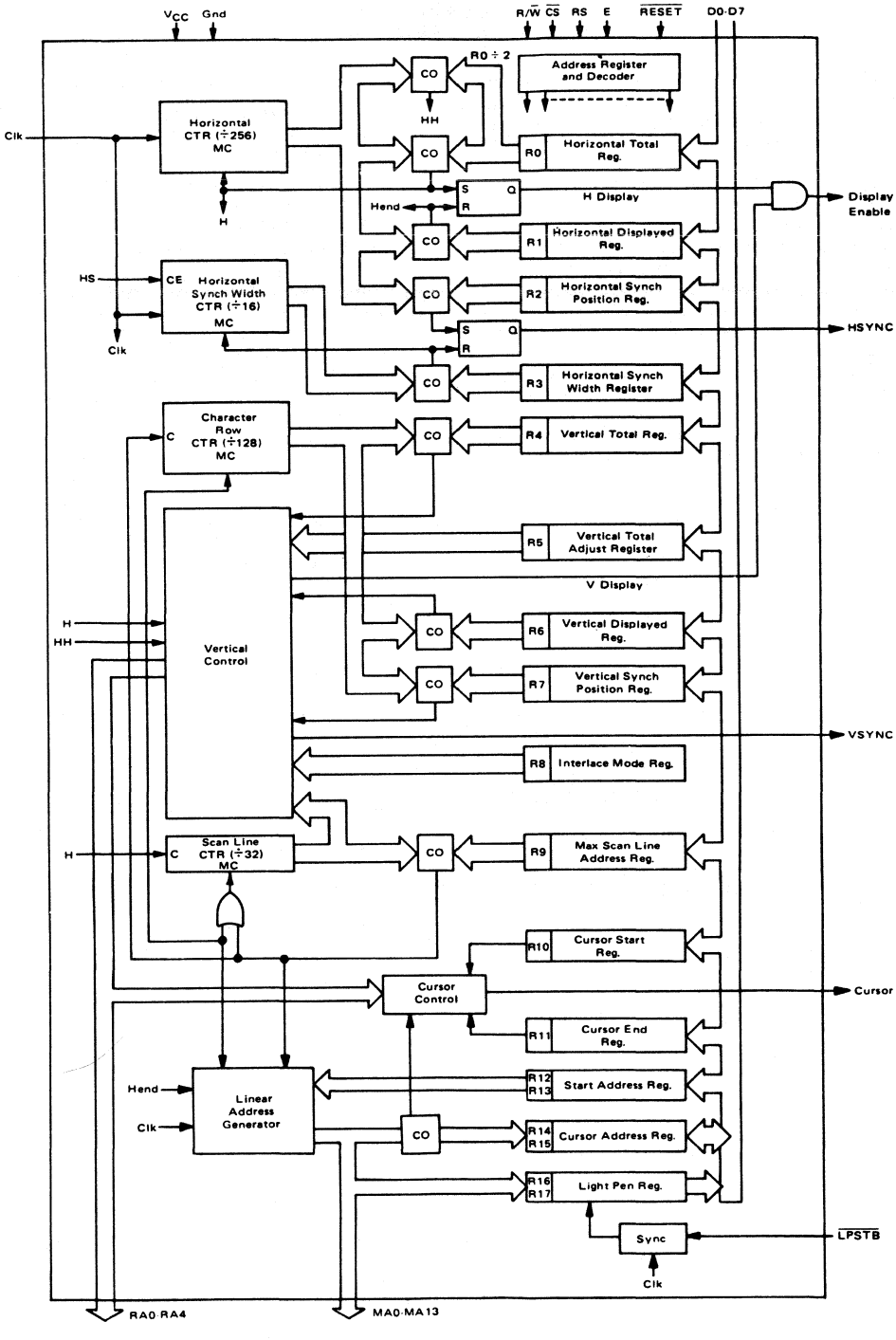
This signal is different from other M6800 family in the following functions:

(A) $\overline{RES}$ signal has capability of reset function only when LPSTB is at low level.
(B) After $\overline{RES}$ has gone down to low level, output signals of MA0-MA13 and RA0-RA4, synchronizing with CLK low level, goes down to low level. (At least 1 cycle CLK signal is necessary for reset.)
(C) The CRTC starts the Display operation immediately after the release of $\overline{RES}$ signal.

TABLE 1 – CRTC Operating Mode

$\overline{RES}$	LPSTB	OPERATING MODE
0	0	Reset
0	1	Test Mode
1	0	Normal Mode
1	1	Normal Mode

FIGURE 8 – CRTC FUNCTIONAL BLOCK DIAGRAM



CRTC DESCRIPTION
(Figure 8: CRTC Block Diagram)

The CRTC consists of programmable horizontal and vertical timing generators, programmable linear address register, programmable cursor logic, light pen capture register, and control circuitry for interface to a processor bus.

All CRTC timing is derived from CLK, usually the output of an external dot rate counter. Coincidence (CO) circuits continuously compare counter contents to the contents of the programmable register file, R0-R17. For horizontal timing generation, comparisons result in: 1) Horizontal sync pulse (HS) of a frequency, position, and width determined by the registers, 2) Horizontal Display Signal of a frequency, position, and duration determined by the registers.

The Horizontal counter produces H clock which drives the Scan Line Counter and Vertical Control. The contents of the Raster Counter are continuously compared to the Max Scan Line Address Register. A coincidence resets the Raster Counter and clocks the Vertical Counter.

Comparisons of Vertical Counter contents and Vertical Registers result in: 1) Vertical sync pulse (VS) of a frequency and positon determined by the registers—the width is fixed at 16 raster lines in the vertical control section and is not programmable, 2) Vertical Display of a frequency and position determined by the registers.

The Vertical Control Logic has other functions.

1. Generate row selects, RA0-RA4, from the Raster Count for the corresponding interlace or non-interlace modes.
2. Extend the number of scan lines in the vertical total by the amount programmed in the Vertical Total Adjust Register.

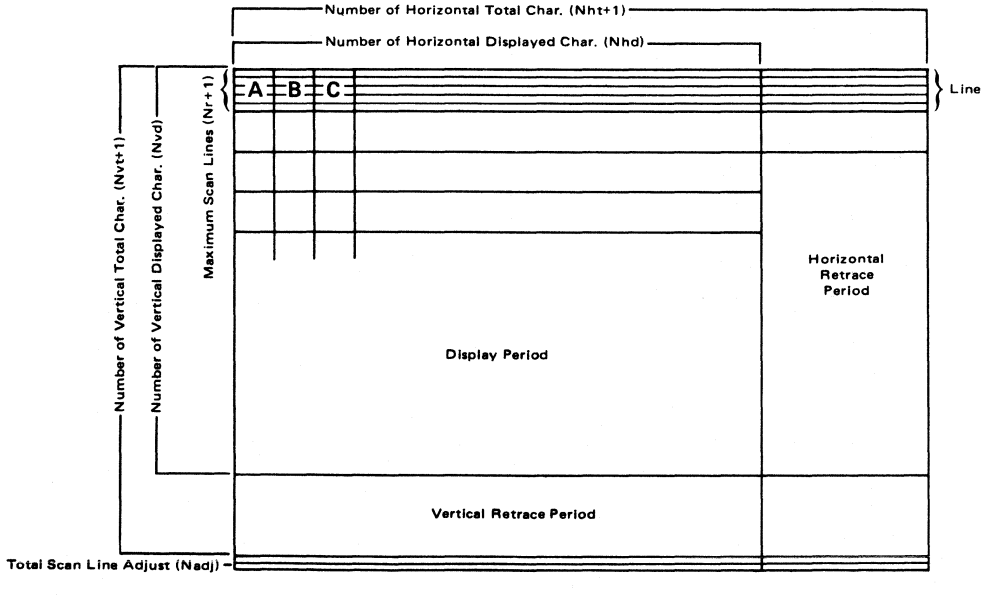
The Linear Address Generator is driven by CLK and locates the relative positions of characters in memory with their positions on the screen. Fourteen lines, MA0-MA13, are available for addressing up to four pages of 4K characters, 8 pages of 2K characters, etc. Using the Start Address Register, hardware scrolling through 16K characters is possible. The Linear Address Generator repeats the same sequence of addresses for each scan line of a character row.

The cursor logic determines the cursor location, size, and blinking rate on the screen. All are programmable.

The light pen strobe going high causes the current contents of the Address Counter to be latched in the Light Pen Register. The contents of the Light Pen Register are subsequently read by the Processor.

Internal CRTC registers are programmed by the processor through the data bus, D0-D7, and the control signals—R/$\overline{W}$, $\overline{CS}$, RS and E.

FIGURE 9 – ILLUSTRATION OF THE CRT SCREEN FORMAT



REGISTER FILE DESCRIPTION
(See Table 2)

Nineteen registers in the CRTC can be accessed by means of the data bus. Register addressing and lengths are shown in Table 2.

ADDRESS REGISTER

The Address Register is a 5 bit write-only register used as an "indirect" or "pointer" register. Its contents are the address of one of the other 18 registers in the file. When RS and $\overline{CS}$ are low, the Address Register itself is addressed. When RS is high, the Register File is accessed.

HORIZONTAL TIMING REGISTERS R0, R1, R2, and R3

Figure 9 shows the visible display area of a typical CRT monitor giving the point of reference for horizontal registers as the left most displayed character position. Horizontal registers are programmed in "character time" units with respect to the reference.

Horizontal Total Register (R0) – This 8 bit write-only register determines the horizontal frequency of HS. It is the total of displayed plus non-displayed character time units minus one.

Horizontal Displayed Register (R1) – This 8 bit write-only register determines the number of displayed characters per horizontal line.

Horizontal Sync Position Register (R2) – This 8 bit write-only register determines the horizontal sync postiion on the horizontal line.

Horizontal Sync Width Register (R3) – This 4 bit write-only register determines the width of the HS pulse. It may not be apparent why this width needs to be programmed. However, consider that all timing widths must be programmed as multiples of the character clock period which varies. If HS width were fixed as an integral number of character times, it would vary with character rate and be out of tolerance for certain monitors. The rate programmable feature allows compensating HS width.

VERTICAL TIMING REGISTERS R4, R5, R6, R7, R8, and R9

The point of reference for vertical registers is the top character position displayed. Vertical registers are programmed in character row times or scan line times.

Vertical Total Register (R4) and Vertical Total Adjust Register (R5) – The vertical frequency of VS is determined by both R4 and R5. The calculated number of character line times is usually an integer plus a fraction to get exactly a 50 or 60 Hz vertical refresh rate. The integer number of character line times minus one is programmed in the 7 bit write-only Vertical Total Register; the fraction is programmed in the 5 bit write-only Vertical Scan Adjust Register as a number of scan line times.

Vertical Displayed Register (R6) – This 7 bit write-only register determines the number of displayed character rows on the CRT screen, and is programmed in character row times.

Vertical Sync Position (R7) – This 7 bit write-only register determines the vertical sync position with respect

TABLE 2 – CRTC INTERNAL REGISTER ASSIGNMENT

$\overline{CS}$	RS	Address Register 4	3	2	1	0	Register #	Register File	Program Unit	Read	Write	Number of Bits 7	6	5	4	3	2	1	0
1	X	X	X	X	X	X	X	–	–	–	–								
0	0	X	X	X	X	X	X	Address Register	–	No	Yes								
0	1	0	0	0	0	0	R0	Horizontal Total	Char.	No	Yes								
0	1	0	0	0	0	1	R1	Horizontal Displayed	Char.	No	Yes								
0	1	0	0	0	1	0	R2	H. Sync Position	Char.	No	Yes								
0	1	0	0	0	1	1	R3	H. Sync Width	Char.	No	Yes								
0	1	0	0	1	0	0	R4	Vertical Total	Char. Row	No	Yes								
0	1	0	0	1	0	1	R5	V. Total Adjust	Scan Line	No	Yes								
0	1	0	0	1	1	0	R6	Vertical Displayed	Char. Row	No	Yes								
W 0	1	0	0	1	1	1	R7	V. Sync Position	Char. Row	No	Yes								
0	1	0	1	0	0	0	R8	Interlace Mode	–	No	Yes								
0	1	0	1	0	0	1	R9	Max Scan Line Address	Scan Line	No	Yes								
0	1	0	1	0	1	0	R10	Cursor Start	Scan Line	No	Yes		B	P				(Note 1)	
0	1	0	1	0	1	1	R11	Cursor End	Scan Line	No	Yes								
0	1	0	1	1	0	0	R12	Start Address (H)	–	No	Yes								
0	1	0	1	1	0	1	R13	Start Address (L)	–	No	Yes								
0	1	0	1	1	1	0	R14	Cursor (H)	–	Yes	Yes								
0	1	0	1	1	1	1	R15	Cursor (L)	–	Yes	Yes								
0	1	1	0	0	0	0	R16	Light Pen (H)	–	Yes	No								
0	1	1	0	0	0	1	R17	Light Pen (L)	–	Yes	No								

Note (1): Bit 5 of the Cursor Start Raster Register is used for blink period control, and Bit 6 is used to select blink or non-blink.

to the reference. It is programmed in character row times.

Interlace Mode Register (R8) – This 2 bit write-only register controls the raster scan mode (see Figure 11). When bit 0 and bit 1 are reset, or bit 0 is reset and bit 1 set, the non-interlace raster scan mode is selected. Two interlace modes are available. Both are interlaced 2 fields per frame. When bit 0 is set and bit 1 is reset, the interlace sync raster scan mode is selected. Also when bit 0 and bit 1 are set, the interlace sync and video raster scan mode is selected.

Maximum Scan Line Address Register (R9) – This 5 bit write-only register determines the number of scan lines per character row including spacing. The programmed value is a max address and is one less than the number of scan lines.

OTHER REGISTERS

Cursor Start Register (R10) – This 7 bit write-only register controls the cursor format (see Figure 10). Bit 5 is the blink timing control. When bit 5 is low, the blink frequency is 1/16 of the vertical field rate, and when bit 5 is high, the blink frequency is 1/32 of the vertical field rate. Bit 6 is used to enable a blink. The cursor start scan line is set by the lower 5 bits.

Cursor End Register (R11) – This 5 bit write-only register sets the cursor end scan line.

Start Address Register (H & L) (R12, R13) – Start Address Register is a 14 bit write-only register which determines the first address put out as a refresh address after vertical blanking. It consists of an 8 bit lower register, and a 6 bit higher register.

Light Pen Register (H & L) (R16, R17) – This 14 bit read-only register is used to store the contents of the Address Register (H & L) when the LPSTB input pulses high. This register consists of an 8 bit lower and 6 bit higher register.

Cursor Register (H & L) (R14, R15) – This 14 bit read/write register stores the cursor location. This register consists of an 8 bit lower and 6 bit higher register.

CURSOR

The Cursor Start and End Registers allow a cursor of up to 32 scan lines in height to be placed on any scan lines of the character block as shown in Figure 10. Using Bits 5 & 6 of the Cursor Start Register, the cursor is programmed with blink periods of 16 or 32 times the field period. Optional non-blink and non-display modes can also be selected. When an external 2X blink on characters is required, it may be necessary to perform cursor blink externally as well so that both blink rates are synchronized. Note that an invert/non-invert cursor is easily implemented by programming the CRTC for blinking cursor and externally inverting the video signal with an exclusive-OR.

The cursor is positioned by changing the contents of registers R14 and R15. The cursor can be placed at any of 16K character positions, thus facilitating hardware paging and scrolling through memory without loss of the cursor's original position.

INTERLACE/NON-INTERLACE DISPLAY MODES

An illustration of the 3 raster scan modes of operation is shown in Figure 11. Normal sync mode is non-interlace. In this mode, each scan line is refreshed at the vertical field rate (e.g., 50 or 60 Hz). Frame time is divided into even and odd alternating fields. The horizontal and vertical timing relationship results in the displacement of scan lines in the odd field with respect to the even field. When the same information is painted in both fields, the mode is called "Interlace Sync;" this is a useful mode for enhancing readability by filling in a character. When the even lines of a character are displayed in the even field and the odd lines in the odd field, the mode is called "Interlace Sync and Video." This last mode effectively doubles the character density on a monitor of a given bandwidth. The disadvantage of both interlace modes is an apparent flicker effect, which can be reduced by careful monitor design.

There are restrictions on the programming of CRTC registers for interlace operation:

1) Horizontal total character count, N_{ht} must be odd (i.e., an even number of character times)
2) For Interlace Sync and Video mode only, the max scan line address, N_{sl}, must be odd (i.e., an even number of scan lines)
3) For Interlace Sync and Video mode only, the Vertical Displayed Total characters must be even. The programmed number, N_{vd}, must be *one-half* the actual number required.
4) For Interlace Sync & Video mode only, the Cursor START and Cursor End Registers must both be even or both odd.

LIGHT PEN

The contents of the CRTC Address Counter are strobed into R16/R17 Light Pen Registers on the next high to low CLK transition after LPSTB goes high. In most systems, the light pen signal would also cause a processor interrupt routine to read R16/R17. Slow light pen response requires the processor software to modify the captured address read from R16/R17 by a calibration factor.

PROGRAMMING CONSIDERATIONS

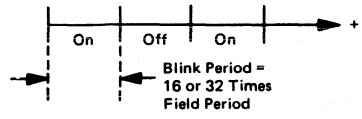
Initialization – Registers R0-R15 must be initialized after power is turned on. The processor normally loads the CRTC registers sequentially from a firmware table. Henceforth, R0-R11 are not changed in most systems. The 6800 program in Table 3 and Figure 12 shows a typical CRTC initialization.

Hardware Scrolling – Registers R12/R13 contents determine which memory location is the first displayed character on the screen. Since the CRTC Linear Address Generator counts from this beginning count, the displayed portion of the screen may be a window on any continuous string of characters within a 16K block or refresh memory. By centering the R12/R13 pointer in the middle of the available memory space, scrolling up or down is possible . . . by line, page, or character.

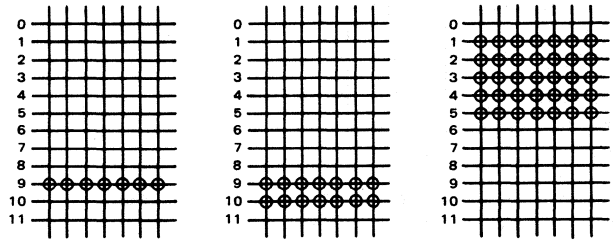
FIGURE 10 – CURSOR CONTROL

Cursor Start Register

B Bit 6	P Bit 5	Cursor Display Mode
0	0	Non-Blink
0	1	Cursor Non-Display
1	0	Blink, 1/16 Field Rate
1	1	Blink, 1/32 Field Rate



Example of Cursor Display Mode



Cursor Start Adr. = 9
Cursor End Adr. = 9

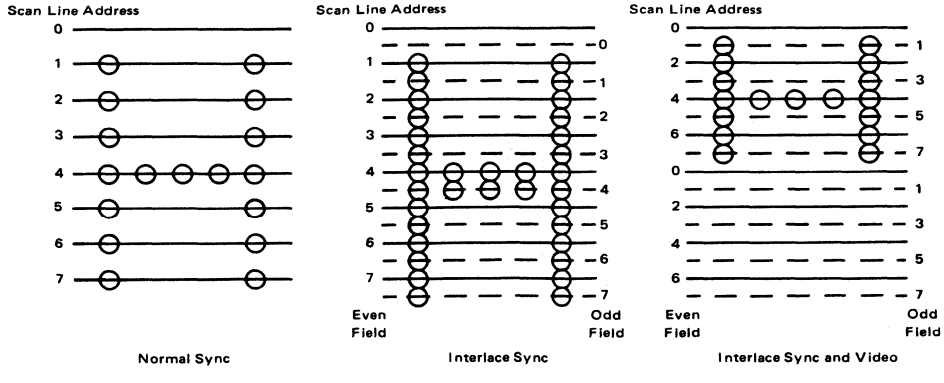
Cursor Start Adr. = 9
Cursor End Adr. = 10

Cursor Start Adr. = 1
Cursor End Adr. = 5

FIGURE 11 – INTERFACE CONTROL

Interlace Mode Register

Bit 1	Bit 0	Mode
0 1	0 0	Normal Sync Mode (Non-Interlace)
0	1	Interlace Sync Mode
1	1	Interlace Sync & Video Mode



Normal Sync

Interlace Sync

Interlace Sync and Video

TABLE 3 – Typical 80 x 24 Screen Format Initialization of CRTC

Reg. #	Register File	Program Unit	Calculation*	Programmed Value	
				Decimal	Hex
R0	H Total	T_c	102 x .527 = 53.76 μs	102 - 1 = 101	N_{ht} = $65
R1	H Displayed	T_c	80 x .527 = 42.16 μs	80	N_{hd} = $50
R2	H Sync Position	T_c	86 x .527 = 45.32 μs	86	N_{hsp} = $56
R3	H Sync Width	T_c	9 x .527 = 4.74 μs	9	N_{hsw} = $09
R4	V Total	T_{cr}	25 x 645.12 = 16.13 ms	25 = 1 = 24	N_{vt} = $18
R5	V Total Adjust	T_{sl}	10 x 53.76 = .54 ms	10	N_{adj} = $0A
R6	V Displayed	T_{cr}	24 x 645.12 = 15.48 ms	24	N_{vd} = $18
R7	V Sync Position	T_{cr}	24 x 645.12 = 15.48 ms	24	N_{vsp} = $18
R8	Interlace Mode	–		–	$00
R9	Max Scan Line Address	T_{sl}		11	N_{sl} = $0B
R10	Cursor Start	T_{sl}		0	$00
R11	Cursor End	T_{sl}		11	$0B
R12	Start Address (H)	–		128	$00
R13	Start Address (L)	–			$80
R14	Cursor (H)	–		128	$00
R15	Cursor (L)	–			$80

Clock Period = T_c = .527 μs

Scan Line Period = T_{sl} = $(N_{ht} + 1) \times T_c$ = 102 x .527 μs = 53.76 μs

Character Row Period = $T_{cr} = N_{sl} \times T_{sl}$ = 12 x 53.76 μs = 645.12 μs

*These are typical values for the Motorola M3000 Monitor; values may vary for other monitors.

FIGURE 12 – INITIALIZATION OF CRTC FOR 80x24 SCREEN FORMAT IN TABLE 3

```
PAGE  001   CRTINT

00001                          NAM     CRTINT
00002 0000                     ORG     $0
00003 0000 5F                  CLR B                 CLEAR COUNTER
00004 0001 CE 0020             LDX     #$20
00005 0004 F7 9000 CRTII       STA B   $9000         CRTC ADDR REG
00006 0007 A6 00               LDA A   0,X
00007 0009 B7 9001             STA A   $9001         ACC TO CRTC REG
00008 000C 08                  INX
00009 000D 5C                  INC B                 INC COUNTER
00010 000E C1 10               CMP B   #$10          LAST CRTC REG?
00011 0010 26 F2               BNE     CRTII
00012 0012 3F                  SWI
00013 0020                     ORG     $20
00014 0020 65          CRTTAB  FCB     $65,$50,$56,$9
00015 0024 18                  FCB     $18,$0A,$18,$18
00016 0028 00                  FCB     0,$0B,0,$0B
00017 002C 0080                FDB     $80,$80
00018      0000                END
CRTII  0004 CRTTAB 0020

TOTAL ERRORS 00000
```

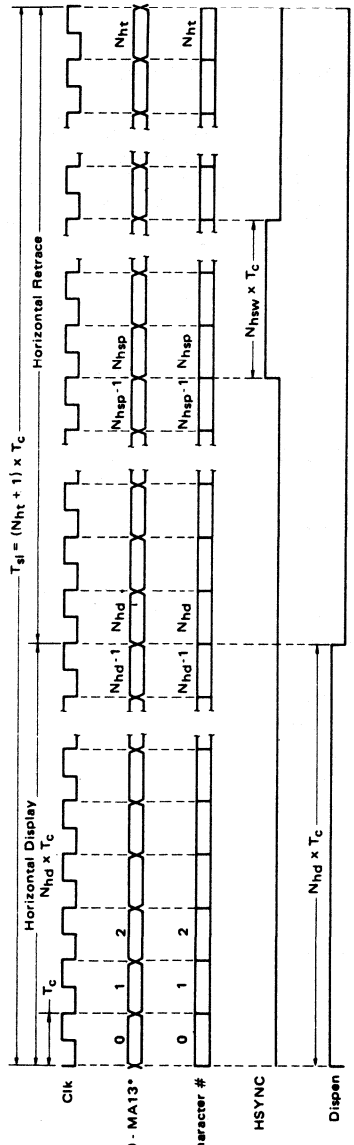
OPERATION OF THE CRTC

Timing Chart of the CRT Interface Signals — Timing charts of CRT interface signals are illustrated in this section with the aid of programmed example of the CRTC. When values listed in Table 4 are programmed into CRTC control registers, the device provides the outputs as shown in the Timing Diagrams (Figures 13 through 15). The screen format of this example is shown in Figure 9. Figure 16 is an illustration of the relation between Refresh Memory Address (MA0-MA13), Raster Address (RA0-RA4) and the position on the screen. In this example, the start address is assumed to be "0".

TABLE 4 — Values Programmed Into CRTC Registers

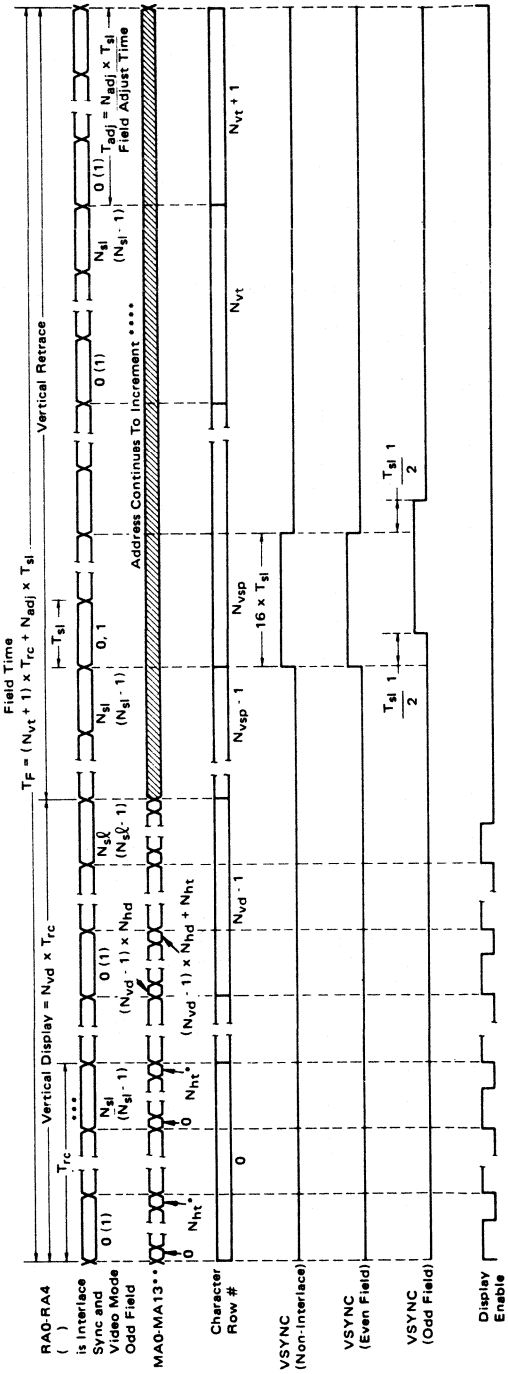
Reg. #	Register Name	Value	Programmed Value
R0	H. Total	$N_{ht}+1$	N_{ht}
R1	H. Displayed	N_{hd}	N_{hd}
R2	H. Sync Position	N_{hsp}	N_{hsp}
R3	H. Sync Width	N_{hsw}	N_{hsw}
R4	V. Total	$N_{vt}+1$	N_{vt}
R5	V. Scan Line Adjust	N_{adj}	N_{adj}
R6	V. Displayed	N_{vd}	N_{vd}
R7	V. Sync Position	N_{vsp}	N_{vsp}
R8	Interlace Mode		
R9	Max. Scan Line Address	N_{sl}	N_{sl}
R10	Cursor Start		
R11	Cursor End		
R12	Start Address (H)	0	
R13	Start Address (L)	0	
R14	Cursor (H)		
R15	Cursor (L)		
R16	Light Pen (H)		
R17	Light Pen (L)		

FIGURE 13 – CRTC HORIZONTAL TIMING



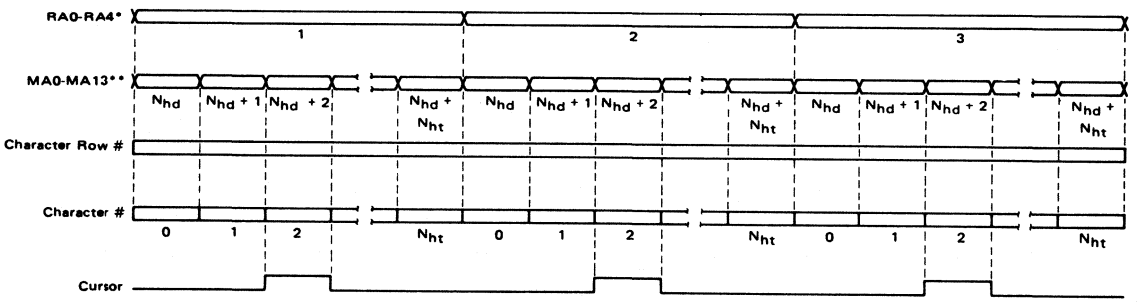
*Timing is shown for first displayed scan row only. See Chart in Figure 16 for other rows. The initial MA is determined by the contents of Start Address Register, R12/R13. Timing is shown for R12/R13 = 0.

FIGURE 14 – CRTC VERTICAL TIMING



*N_{ht} must be an odd number for both interlace modes.
**Initial MA is determined by R12/R13 (Start Address Register), which is zero in this timing example.
***N_{sl} must be an odd number for interlace Sync and Video Mode.
****The present CRTC freezes MA addresses at $N_{vd} \times N_{hd}$ during vertical retrace. A design change is pending to allow MA to free run during vertical retrace time.

FIGURE 15 – CURSOR TIMING

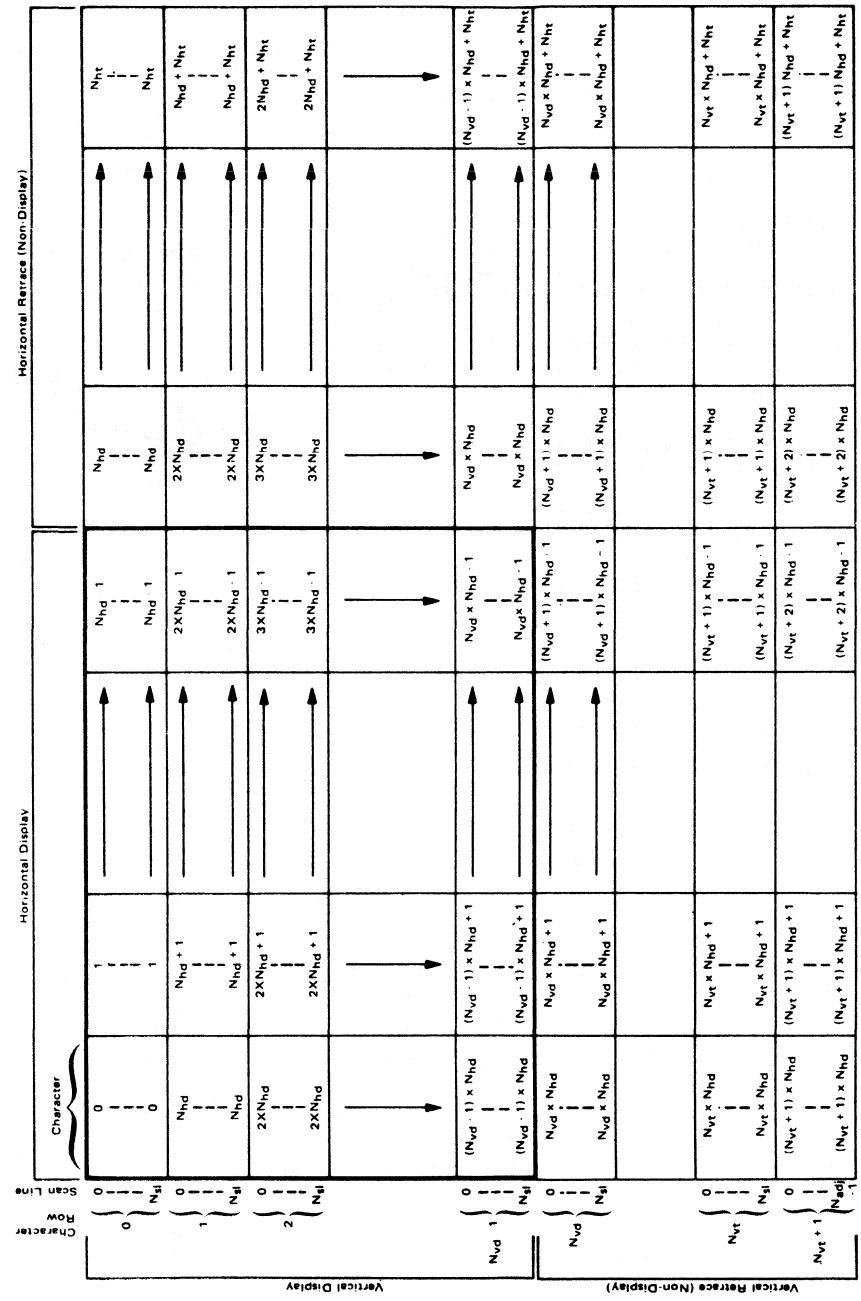


*Timing is shown for non-interlace and interlace sync modes.
Example shown has cursor programmed as:

Cursor Register = $N_{hd} + 2$
Cursor Start = 1
Cursor End = 3

** The initial MA is determined by the contents of Start Address Register, R12/R13. Timing is shown for R12/R13 = 0.

FIGURE 16 – REFRESH MEMORY ADDRESSING (MA0-MA13) STATE CHART



NOTE 1 The initial MA is determined by the contents of start address register, R12/R13. Timing is shown for R12/R13 = 0. Only Non Interlace and Interlace Sync Modes are shown.

NOTE 2 The present CRTC freezes MA addresst at $N_{vd} \times N_{hd}$ during vertical retrace. A design change is pending to allow MA to free run during vertical retrace time.

MOTOROLA

MC6846

Advance Information

ROM–I/O–TIMER

The MC6846 combination chip provides the means, in conjunction with the MC6802, to develop a basic 2-chip microcomputer system. The MC6846 consists of 2048 bytes of mask-programmable ROM, an 8-bit bidirectional data port with control lines, and a 16-bit programmable timer-counter.

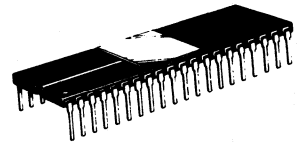
This device is capable of interfacing with the MC6802 (basic MC6800, clock and 128 bytes of RAM) as well as the MC6800 if desired. No external logic is required to interface with most peripheral devices.

- 2048 8-Bit Bytes of Mask-Programmable ROM
- 8-Bit Bidirectional Data Port for Parallel Interface plus Two Control Lines
- Programmable Interval Timer-Counter Functions
- Programmable I/O Peripheral Data, Control and Direction Registers
- Compatible with the Complete M6800 Microcomputer Product Family
- TTL-Compatible Data and Peripheral Lines
- Single 5-Volt Power Supply

MOS

(N-CHANNEL, SILICON-GATE, DEPLETION LOAD)

ROM–I/O–TIMER



L SUFFIX
CERAMIC PACKAGE
CASE 715

NOT SHOWN:
P SUFFIX
PLASTIC PACKAGE
CASE 711

FIGURE 1 – TYPICAL MICROCOMPUTER

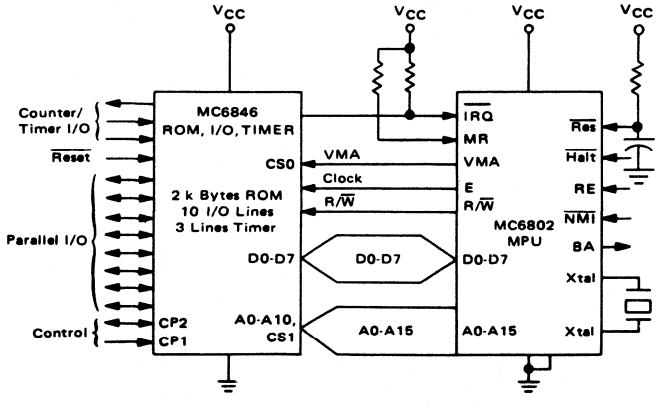
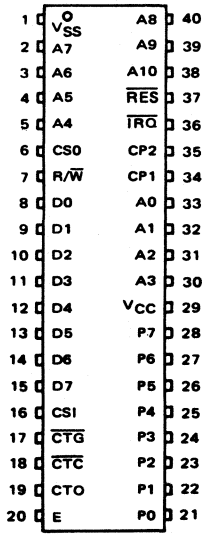


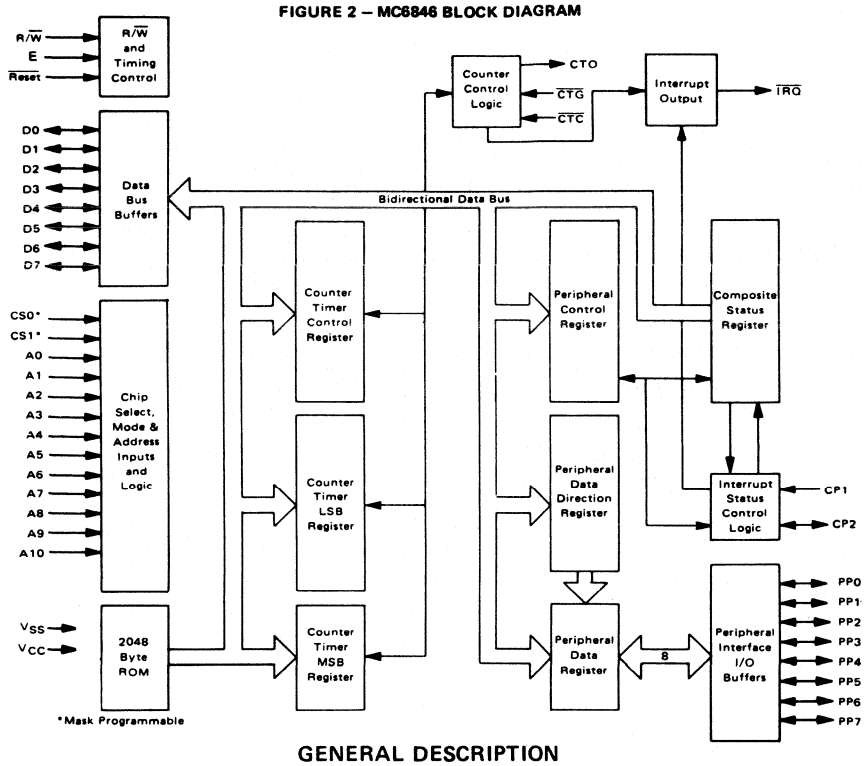
Figure 1 is a block diagram of a typical cost effective microcomputer. The MPU is the center of the microcomputer system and is shown in a minimum system interfacing with a ROM combination chip. It is not intended that this system be limited to this function but that it be expandable with other parts in the M6800 Microcomputer family.

ROM – I/O – TIMER



This is advance information and specifications are subject to change without notice.

FIGURE 2 – MC6846 BLOCK DIAGRAM



GENERAL DESCRIPTION

The MC6846 combination chip may be partitioned into three functional operating sections: programmed storage, timer-counter functions, and a parallel I/O port.

Programmed Storage

The mask-programmable ROM section is similar to other ROM products of the M6800 family. The ROM is organized in a 2048 by 8-bit array to provide read only storage for a minimum microcomputer system. Two mask-programmable chip selects are available for user definition.

Address inputs A0–A10 allow any of the 2048 bytes of ROM to be uniquely addressed. Bidirectional data lines (D0-D7) allow the transfer of data between the MPU and the MC6846.

Timer-Counter Functions

Under software control this 16-bit binary counter may be programmed to count events, measure frequencies, time intervals, or similar tasks. Internal registers associated with the I/O functions may be selected with A0, A1 and A2. It may also be used for square wave generation, single pulses of controlled duration, and gated signals. Interrupts may be generated from a number of conditions selectable by software programming.

The timer/counter control register allows control of the interrupt enable, output enable, selection of an internal or external clock source, a ÷ 8 prescaler, and operating mode. Input pin $\overline{CTC}$ (counter-timer clock) will accept an asynchronous clock pulse to decrement the internal register for the counter-timer. If the divide-by-8 prescaler is used, the maximum clock rate can be four times the master clock frequency with an absolute maximum of 4 MHz. Gate input ($\overline{CTG}$) accepts an asynchronous TTL-compatible signal which may be used as a trigger or gating function to the counter-timer. A counter-timer output (CTO) is also available and is under software control being dependent on the timer control register, the gate input, and the clock source.

Parallel I/O Port

The parallel bidirectional I/O port has functional operational characteristics similar to the B port on the MC6820 PIA. This includes 8 bidirectional data lines and two handshake control signals. The control and operation of these lines are completely software programmable.

The interrupt input (CP1) will set the interrupt flag CSR1 of the composite status register. The peripheral control (CP2) may be programmed to act as an interrupt input (set CSR2) or as a peripheral control output.

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	-0.3 to +7.0	Vdc
Input Voltage	V_{in}	-0.3 to +7.0	Vdc
Operating Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	-55 to +150	°C
Thermal Resistance	θ_{JA}	70	°C/W

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, is is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit.

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ± 5%, V_{SS} = 0, T_A = 0 to 70°C unless otherwise noted.)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage	All Inputs	V_{IH}	V_{SS} + 2.0	–	V_{CC}	Vdc
Input Low Voltage	All Inputs	V_{IL}	V_{SS} – 0.3	–	V_{SS} + 0.8	Vdc
Clock Overshoot – Input High Level		V_{OS}	V_{CC} – 0.5	–	V_{CC} + 0.5	Vdc
– Input Low Level			V_{SS} – 0.5	–	V_{SS} + 0.5	
Input Leakage Current	R/$\overline{W}$, $\overline{Reset}$, CS0, CS1	I_{in}	–	1.0	2.5	μAdc
(V_{in} = 0 to 5.25 Vdc)	CP1, $\overline{CTG}$, $\overline{CTC}$, E, A0-A10					
Three-State (Off State) Input Current	D0-D7	I_{TSI}	–	2.0	10	μAdc
(V_{in} 0.4 to 2.4 Vdc)	PP0-PP7, CP2					
Output High Voltage		V_{OH}				Vdc
(I_{Load} = -205 μAdc,)	D0-D7		V_{SS} + 2.4	–	–	
(I_{Load} = -200 μAdc)	Other Outputs		V_{SS} + 2.4	–	–	
Output Low Voltage		V_{OL}				Vdc
(I_{Load} = 1.6 mAdc)	D0-D7		–	–	V_{SS} + 0.4	
(I_{Load} = 3.2 mAdc)	Other Outputs		–	–	V_{SS} + 0.4	
Output High Current (Sourcing)		I_{OH}				μAdc
(V_{OH} = 2.4 Vdc)	D0-D7		-205	–	–	
	Other Outputs		-200	–	–	
(V_O = 1.5 Vdc, the current for driving other than TTL, e.g., Darlington Base)	CP2, PP0-PP7		-1.0	–	-10	mAdc
Output Low Current (Sinking)		I_{OL}				mAdc
(V_{OL} = 0.4 Vdc)	D0-D7		1.6	–	–	
	Other Outputs		3.2	–	–	
Output Leakage Current (Off State) (V_{OH} = 2.4 Vdc)	$\overline{IRQ}$	I_{LOH}	–	–	10	μAdc
Power Dissipation		P_D	–	–	1000	mW
Capacitance		C_{in}	–	–	20	pF
(V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	D0-D7		–	–	12.5	
	PP0-PP7, CP2		–	–	10	
	A0-A10, R/$\overline{W}$, $\overline{Reset}$, CS0, CS1, CP1, $\overline{CTC}$, $\overline{CTG}$ $\overline{IRQ}$		–	–	7.5	
	PP0-PP7, CP2, CT0	C_{out}		–	5.0	pF
				–	10	
Frequency of Operation		f	0.1	–	1.0	MHz
Clock Timing						
Cycle Time		t_{cycE}	1.0	–	–	μs
Reset Low Time		t_{RL}	2	–	–	μs
Interrupt Release		t_{IR}	–	–	1.6	μs

READ/WRITE TIMING (Figures 3 and 4).

Characteristic	Symbol	Min	Typ	Max	Unit
Enable Pulse Width, Low	PW_{EL}	430	–	–	ns
Enable Pulse Width, High	PW_{EH}	430	–	–	ns
Set Up Time (Address CS0, CS1, R/$\overline{W}$)	t_{AS}	160	–	–	ns
Data Delay Time	t_{DDR}	–	–	320	ns
Data Hold Time	t_H	10	–	–	ns
Address Hold Time	t_{AH}	10	–	–	ns
Rise and Fall Time	t_{Ef}, t_{Er}	–	–	25	ns
Data Set Up Time	t_{DSW}	195	–	–	ns

BUS TIMING

Peripheral I/O Lines

Characteristic	Symbol	Min	Typ	Max	Unit
Peripheral Data Setup	t_{PDSU}	200	–	–	ns
Rise and Fall Times CP1, CP2	t_{Pr}, t_{Pc}	–	–	1.0	µs
Delay Time E to CP2 Fall	t_{CP2}	–	–	1.0	µs
Delay Tme I/O Data CP2 Fall	t_{DC}	20	–	–	ns
Delay Time E to CP2 Rise	t_{RS1}	–	–	1.0	µs
Delay Time CP1 to CP2 Rise	t_{RS2}	–	–	2.0	µs
Peripheral Data Delay	t_{PDW}	–	–	1.0	µs
Peripheral Data Setup Time for Latch	t_{PSU}	100	–	–	ns
Peripheral Data Hold Time for Latch	t_{PDH}	15	–	–	ns

Timer-Counter Lines

Input Rise and Fall Time	CTC and CTG	t_{CR}, t_{CF}	–	–	100	ns
Input Pulse Width High (Asynchronous Mode)		t_{PWH}	t_{cyc} + 250	–	–	ns
Input Pulse Width Low (Asychronous Mode)		t_{PWL}	t_{cyc} + 250	–	–	ns
Input Setup Time (Synchronous Mode)		t_{su}	200	–	–	ns
Input Hold Time (Synchronous Mode)		t_{hd}	60	–	–	ns
Output Delay		t_{CTO}	–	–	1.0	µs

FIGURE 3 – BUS READ TIMING
Read Information from MC6846)

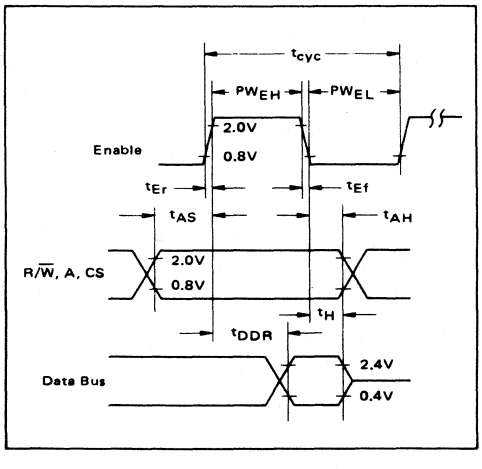


FIGURE 4 – BUS WRITE TIMING
(Write Information from MPU)

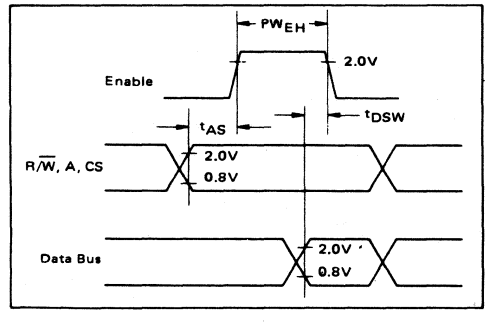


FIGURE 5 – PERIPHERAL PORT LATCH SETUP AND HOLD TIME

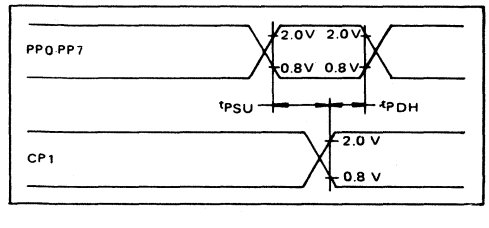


FIGURE 6 – PERIPHERAL DATA AND CP2 DELAY
(Control Mode PCR5=1, PCR4=0, PCR3=1)

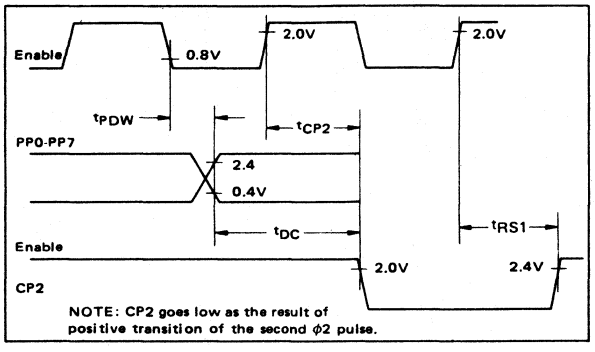


FIGURE 7 – $\overline{IRQ}$ RELEASE TIME

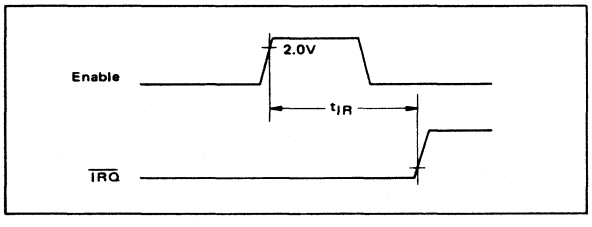


FIGURE 8 – PERIPHERAL PORT SETUP TIME

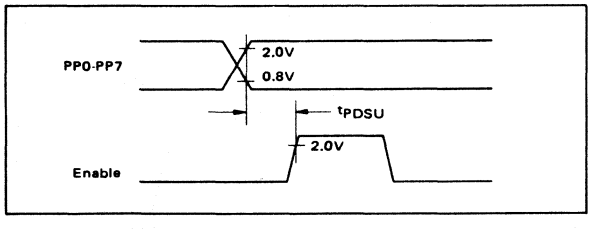


FIGURE 9 – CP2 DELAY TIME
(PCR5=1, PCR4=0, PCR3=0)

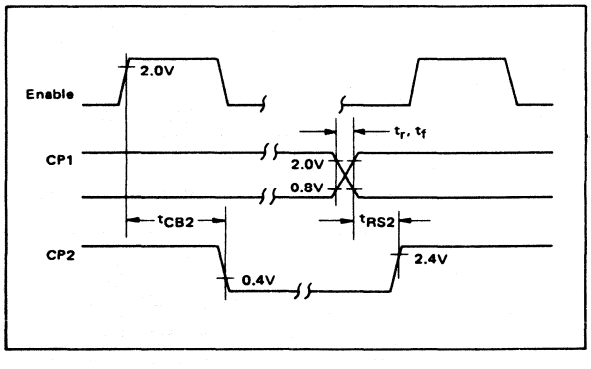


FIGURE 10 – INPUT PULSE WIDTHS

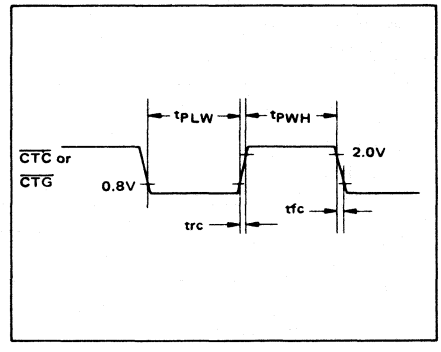


FIGURE 11 – INPUT SET-UP AND HOLD TIMES

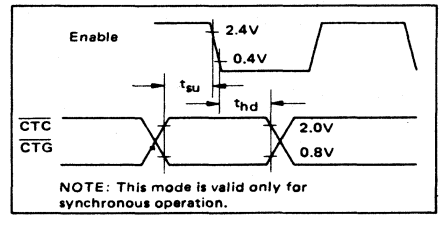


FIGURE 12 – OUTPUT DELAY

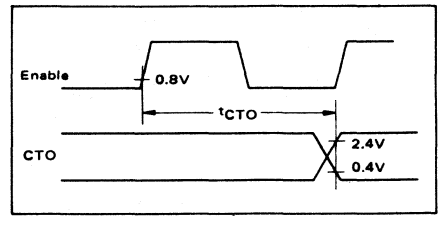
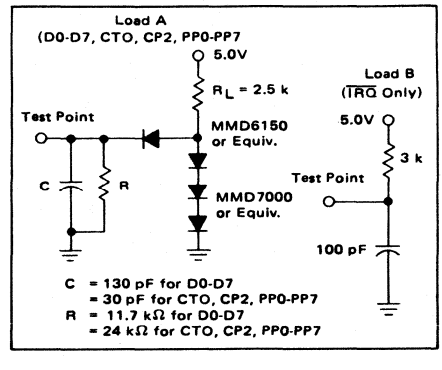


FIGURE 13 – BUS TIMING TEST LOADS



SIGNAL DESCRIPTION

BUS INTERFACE

The MC6846 interfaces to the M6800 Bus via an eight-bit bidirectional data bus, two Chip Select lines, a Read/$\overline{\text{Write}}$ line, and eleven address lines. These signals, in conjunction with the M6800 VMA output, permit the MPU to control the MC6846.

BIDIRECTIONAL DATA BUS (D0-D7)

The bidirectional data lines (D0-D7) allow the transfer of data between the MPU and the MC6846. The data bus output drivers are three-state devices which remain in the high-impedance (Off) state except when the MPU performs an MC6846 register or ROM read (R/$\overline{W}$ = 1 and I/O Registers or ROM selected).

CHIP SELECT (CS0, CS1)

The CS0 and CS1 inputs are used to select the ROM or I/O timer of the MC6846. They are mask programmed to be active high or active low as chosen by the user.

ADDRESS INPUTS (A0-A10)

The Address Inputs allow any of the 2048 bytes of ROM to be uniquely selected when the circuit is operating in the ROM mode. In the I/O-Timer mode, address inputs A0, A1, and A2 select the proper I/O Register, while A3 through A10 (together with CS0 and CS1) can be used as additional qualifiers in the I/O Select circuitry. (See the section on I/O-Timer Select for additional details.)

$\overline{\text{RESET}}$

The active low state of the $\overline{\text{Reset}}$ input is used to initialize all register bits in the I/O section of the device to their proper values. (See the section on Initialization for Reset conditions for timer and peripheral registers.)

ENABLE (ϕ2)

This signal synchronizes data transfer between the MPU and the MC6846. It also performs an equivalent synchronization function on the external clock, reset, and gate inputs of the MC6846 Timer section.

READ/$\overline{\text{WRITE}}$ (R/$\overline{W}$)

This signal is generated by the MPU and is used to control the direction of data transfer on the bidirectional data pins. A low level on the R/$\overline{W}$ input enables the MC6846 input buffers and data is transferred to the circuit during the ϕ2 pulse when the part has been selected. A high level on the R/$\overline{W}$ input enables the output buffers and data is transferred to the MPU during ϕ2 when the part is selected.

$\overline{\text{INTERRUPT REQUEST}}$ ($\overline{\text{IRQ}}$)

The active low $\overline{\text{IRQ}}$ output acts to interrupt the MPU through logic included on the MC6846. This output utilizes an open drain configuration and permits other interrupt request outputs from other circuits to be connected in a wire-OR configuration.

PERIPHERAL DATA (P0-P7)

The peripheral data lines can be individually programmed as either inputs or outputs via the Data Direction Register. When programmed as outputs, these lines will drive two standard TTL loads (3.2 mA). They are also capable of sourcing up to 1.0 mA at 1.5 Volts (Logic "1" output.)

When programmed as inputs, the output drivers associated with these lines enter a three-state (high impedance) mode. Since there is no internal pull-up for these lines, they represent a maximum 10μA load to the circuitry driving them – regardless of logic state.

A logic zero at the $\overline{\text{Reset}}$ input forces the peripheral data lines to the input configuration by clearing the Data Direction Register. This allows the system designer to preclude the possibility of having a peripheral data output connected to an external driver output during power-up sequence.

INTERRUPT INPUT (CP1)

Peripheral input line CP1 is an input-only that sets the Interrupt Flags of the Composite Status register. The active transition for this signal is programmed by the peripheral control register for the parallel port. CP1 may also act as a strobe for the peripheral data register when it is used as an input latch. Details for programming CP1 are in the section on the parallel peripheral port.

PERIPHERAL CONTROL (CP2)

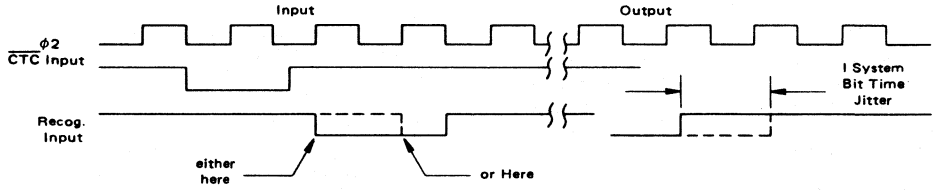
Peripheral Control line CP2 may be programmed to act as an Interrupt input or Peripheral Control output. As an input, this line has high impedance and is compatible with standard TTL voltage levels. As an output, it is also TTL compatible and may be used as a source of 1 mA at 1.5 V to directly drive the base of a Darlington transistor switch. This line is programmed by the Peripheral Control Register.

COUNTER TIMER OUTPUT (CTO)

The Counter Timer Output is software programmable by selected bits in the timer/counter control register. The mode of operation is dependent on the Timer control register, the gate input, and the clock source. The output is TTL compatible.

$\overline{\text{EXTERNAL CLOCK INPUT}}$ ($\overline{\text{CTC}}$)

Input pin $\overline{\text{CTC}}$ will accept asynchronous TTL voltage level signals to be used as a clock to decrement the Timer. The high and low levels of the external clock must be stable for at least one system clock period plus the sum of



the setup and hold times for the inputs. The asynchronous clock rate can vary from dc to the limit imposed by System $\phi 2$, setup, and hold times.

The external clock input is clocked in by Enable (System $\phi 2$) pulses. Three Enable periods are used to synchronize and process the external clock. The fourth Enable pulse decrements the internal counter. This does not affect the input frequency; it merely creates a delay between a clock input transition and internal recognition of that transition by the MC6846. All references to $\overline{CTC}$ inputs in this document relate to internal recognition of the input transition. Note that a clock transition which does not meet setup and hold time specifications may require an additional Enable pulse for recognition.

When observing recurring events, a lack of synchronization will result in either "System jitter" or "Input jitter" being observed on the output of the MC6846 when using an asynchronous clock and gate input signal. "System jitter" is the result of the input signals being out of synchronization with the system $\phi 2$ clock (Enable), permitting signals with marginal set-up and hold time to be recognized by either the bit time nearest the input transition or subsequent bit time. "Input jitter" can be as great as the time between the negative going transitions of the input signal plus the system jitter if the first transition is recognized during one system cycle, and not recognized the next cycle or vice-versa.

GATE INPUTS ($\overline{CTG}$)

The input pin $\overline{CTG}$ accepts an asynchronous TTL-compatible signal which is used as a trigger or a clock gating function to the Timer. The gating input is clocked into the MC6846 by the Enable (System $\phi 2$) signal in the same manner as the previously discussed clock inputs. That is, a $\overline{CTG}$ transition is recognized on the fourth Enable pulse (provided setup and hold time requirements are met), and the high or low levels of the $\overline{CTG}$ input must be stable for at least one system clock period plus the sum of setup and hold times. All references to $\overline{CTG}$ transition in this document relate to internal recognition of the input transition.

The $\overline{CTG}$ input of the timer directly affects the internal 16-bit counter. The operation of $\overline{CTG}$ is therefore independent of the ÷ 8 prescaler selection.

FUNCTIONAL SELECT CIRCUITRY

I/O-TIMER SELECT CIRCUITRY

CS0 and CS1 are user programmable. Any of the four binary combinations of CS0 and CS1 can be used to select the ROM. Likewise, any other combination can be used to select the I/O-Timer. In addition, several address lines are used as qualifiers for the I/O-Timer. Specifically, A3 = A4 = A5 = logical "0". A6 can be programmed to a "1", "0", or don't care. A7 = A8 = A9 = A10 = don't care or one line only may be programmed to a logical "1". Figure 14 outlines in diagrammatic form the available chip select options.

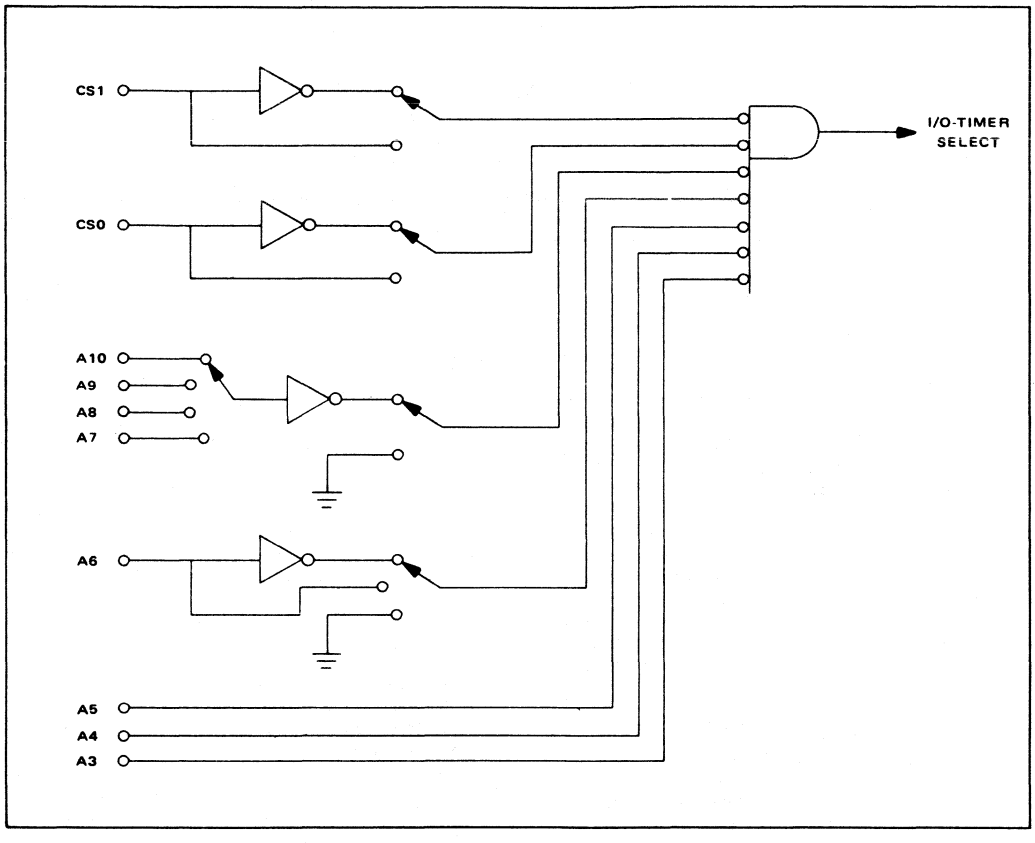
INTERNAL ADDRESSING

Seven I/O Register locations within the MC6846 are accessible to the MPU data bus. Selection of these registers is controlled by A0, A1, and A2 (as shown in Table 1) provided the I/O timer is selected. The combination status register is Read-only; all other Registers are Read and Write.

TABLE 1 – INTERNAL REGISTER ADDRESSES

REGISTER SELECTED	A2	A1	A0
Combination Status Register	0	0	0
Peripheral Control Register	0	0	1
Data Direction Register	0	1	0
Peripheral Data Register	0	1	1
Combination Status Register	1	0	0
Timer Control Register	1	0	1
Timer MSB Register	1	1	0
Timer LSB Register	1	1	1
ROM Address	X	X	X

FIGURE 14 – I/O-TIMER SELECT CIRCUITRY



Initialization

When the $\overline{\text{Reset}}$ input has accepted a low signal, all registers are initialized to the reset state. The data direction and peripheral data registers are cleared. The Peripheral Control Register is cleared except for bit 7 (the $\overline{\text{Reset}}$ bit). This forces the parallel port to the input mode with Interrupts disabled. To remove the $\overline{\text{Reset}}$ condition from the parallel port, a "0" must be written into the Peripheral Control Register bit 7 (PCR7).

The counter latches are preset to their maximal count, the Timer control register bits are reset to zero except for Bit 0 (TCR0 is set), the counter output is cleared, and the counter clock disabled. This state forces the timer counter to remain in an inactive state. The combination status register is cleared of all interrupt flags. During timer initialization, the reset bit (CCR0) must be cleared.

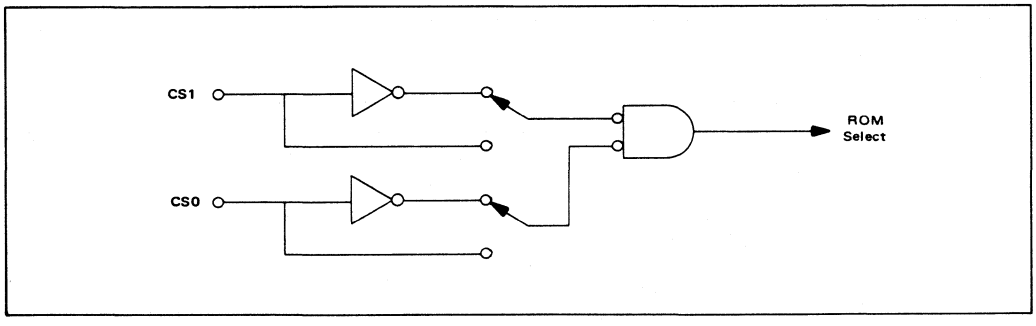
ROM

The Mask Programmable ROM section is similar in operation to other ROM products of the M6800 Microprocessor family. The ROM is organized as 2048 words of 8-bits to provide read-only storage for a minimum microcomputer system. The ROM is active when selected by the unique combination of the chip select inputs.

ROM Select

The active levels of CS0 and CS1 for ROM and I/O select are a user programmable option. Either CS0 or CS1 may be programmed active hi or active low, but different codes must be used for ROM or I/O select. CS0 and CS1 are mask programmed simultaneously with the ROM pattern. The ROM Select Circuitry is shown in Figure 15.

FIGURE 15 – ROM SELECT CIRCUITRY



TIMER OPERATION

The Timer may be programmed to operate in modes which fit a wide variety of applications. The device is fully bus compatible with the M6800 system, and is accessed by Load and Store operations from the MPU.

In a typical application, the timer will be loaded by storing two bytes of data into the counter latch. This data is then transferred into the counter during a Counter Initialization cycle. The counter decrements on each subsequent clock cycle (which may be system $\phi 2$ or an external clock) until one of several predetermined conditions causes it to halt or recycle. Thus the timer is programmable, cyclic in nature, controllable by external inputs or MPU program, and accessible to the MPU at any time.

COUNTER LATCH INITIALIZATION

The Timer consists of a 16-bit addressable counter and two 8 bit addressable latches. The function of the latches is to store a binary equivalent of the desired count value minus one. Counter initialization results in the transfer of the latch contents of the counter. It should be noted that data transfer to the counters is always accomplished via the latches. Thus, the counter latches may be accurately described as a 16-bit "counter initialization data" storage register.

In some modes of operation, the initialization of the latches will cause simultaneous counter initialization (i.e. immediate transfer of the new latch data into the counters). It is, therefore, necessary to insure that all 16 bits of the latches are updated simultaneously. Since the MC6846 data bus is 8 bits wide, a temporary register (MSB Buffer Register) is provided for in the Most Significant Byte of the desired latch data. This is a "write-only" register selected via address lines A0, A1, and A2. Data is transferred directly from the data bus to the MSB Buffer when the chip is selected, R/$\overline{W}$ is low, and the timer MSB register is selected (A0 = "0"; A1 = A2 = "1").

The lower 8 bits of the counter latch can also be referred to as a "write-only" register. Data Bus information will be transferred directly to the LSB of a counter latch when the chip is selected, R/$\overline{W}$ is low and the Timer LSB Register is selected (A0 = A1 = A2 = "1"). Data from the MSB Buffer will automatically be transferred into the Most Significant Byte of the counter latches simultaneously with the transfer of the Data Bus information to the Least Significant Byte of the Counter Latch. For brevity, the conditions for this operation will be referred to henceforth as a "Write Timer Latches Command."

The MC6846 has been designed to allow transfer of two bytes of data into the counter latches from any source, provided the MSB is transferred first. In many applications, the source of data will be an M6800 MPU. It should therefore be noted that the 16-bit store operations of the M6800 family microprocessors (STS and STX) transfer data in the order required by the MC6846. A Store Index Register instruction, for example, results in the MSB of the X register being transferred to the selected address, then the LSB of the X register being written into the next higher location. Thus, either the index register or stack pointer may be transferred directly into a selected counter latch with a single instruction.

A logic zero at the $\overline{\text{Reset}}$ input also initializes the counter latches. All latches will assume maximum count (65, 535) values. It is important to note that an internal $\overline{\text{Reset}}$ (Bit zero of the Timer/Control Register Set) has no effect on the counter latches.

COUNTER INITIALIZATION

Counter Initialization is defined as the transfer of data from the latches to the counter with attendant clearing of the Individual Interrupt Flag associated with the counter. Counter Initialization always occurs when a reset condition (external $\overline{\text{Reset}}$ = 0 or TCRO = 1) is recognized. It can also occur (dependent on The Timer Mode) with a

Write Timer Latches command or recognition of a negative transition of the $\overline{\text{Gate}}$ input.

Counter recycling or reinitialization occurs when a clock input is recognized after the counter has reached an all-zero state. In this case, data is transferred from the Latches to the Counter, but the Interrupt Flag is unaffected.

TIMER CONTROL REGISTER

The Timer Control register (see Table 2) in the MC6846 is used to modify timer operation to suit a variety of applications. The Timer Control Register has a unique address space (A0 = 1, A1 = 0, A2 = 1) and therefore may be written into at any time. The least significant bit of the Control Register is used as an Internal Reset bit. When this bit is a logic zero, all timers are allowed to operate in the modes prescribed by the remaining bits of the timer control register.

Writing "one" into **Timer Control Register Bit 0 (TCR0)** causes the counter to be preset with the contents of the counter latches, all counter clocks are disabled, and the timer output and interrupt flag (Status Register) are reset. The Counter Latch and Timer/Control Register are undisturbed by an Internal Reset and may be written into regardless of the state of TCR0.

Timer Control Register Bit 1 (TCR1) is used to select the clock source. When TCR1 = 0, the external clock input $\overline{\text{CTC}}$ is selected, and when TCR1 = "1", the timer uses system $\phi 2$.

Timer Control Register Bit 2 (TCR2) enables the ÷ 8 prescaler (TCR2 = "1"). In this mode, the clock frequency is divided by eight before being applied to the counter. When TCR2 = "0" the system clock is applied directly to the counter.

TCR3, 4, 5 select the Timer Operating Mode, and are discussed in the next section.

Timer Control Register Bit 6 (TCR6) is used to mask or enable the Timer Interrupt Request. When TCR6 = 0, the Interrupt Flag is masked from the timer. When TCR6 = 1, the Interrupt Flag is enabled into Bit 7 of the Composite Status Register (Composite IRQ Bit), which appears on the $\overline{\text{IRQ}}$ output pin.

Timer Control Register Bit Seven (TCR7) has a special function when the timer is in the Cascaded Single Shot mode. (This function is explained in detail in the section describing the mode.) In all other modes, TCR7 merely acts as an output enable bit. If TCR7 = 0, the Counter Timer Output (CTO) is forced low. Writing a logic one into TCR7 enables CTO.

TIMER OPERATING MODES

The MC6846 has been designed to operate effectively in a wide variety of applications. This is accomplished by using three bits of the control register (TCR3, TCR4, and TCR5) to define different operating modes of the Timer, outlined in Table 3.

TABLE 2 – FORMAT FOR TIMER/COUNTER CONTROL REGISTER

CONTROL REGISTER BIT	STATE	BIT DEFINITION	STATE DEFINITION
TCR0	0	Internal Reset	Timer Enabled
	1		Timer in Preset State
TCR1	0	Clock Source	Timer uses External Clock ($\overline{\text{CTC}}$)
	1		Timer uses $\phi 2$ System Clock
TCR2	0	÷ 8 Prescaler Enabler	Clock is not Prescaled
	1		Clock is prescaled by ÷ 8 Counter
TCR3 TCR4 TCR5	X X X	Operating Mode Selection	See Table 3
TCR6	0	Timer Interrupt Enable	$\overline{\text{IRQ}}$ Masked from Timer
	1		$\overline{\text{IRQ}}$ Enabled from Timer
TCR7	0	Timer Output Enable	Counter Output (CTO) Set LOW
	1		Counter Output Enabled

TABLE 3 – OPERATING MODES

TCR3	TCR4	TCR5	Timer Operating Mode	Counter Initialization	Interrupt Flag Set
0	0	0	Continuous	$\overline{CTG}\downarrow + W + R$	T.O.
0	0	1	Cascaded Single Shot	$\overline{CTG}\downarrow + R$	T.O.
0	1	0	Continuous	$\overline{CTG}\downarrow + R$	T.O.
0	1	1	Normal Single Shot	$\overline{CTG}\downarrow + R$	T.O.
1	0	0	Frequency Comparison	$\overline{CTG}\downarrow \cdot \overline{I} \cdot (W + T.O.) + R$	$\overline{CTG}\downarrow$ Before T.O.
1	0	1		$\overline{CTG}\downarrow \cdot \overline{I} + R$	T.O. Before $\overline{CTG}\downarrow$
1	1	0	Pulse Width Comparison	$\overline{CTG}\downarrow \cdot \overline{I} + R$	$\overline{CTG}\uparrow$ Before T.O.
1	1	1			T.O. Before $\overline{CTG}\uparrow$

R = Reset Condition
W = Write Timer Latches
T.O. = Counter Time Out

$\overline{CTG}\downarrow$ = Negative Transition of Pin 17
$\overline{CTG}\uparrow$ = Positive Transition of Pin 17
$\overline{I}$ = Interrupt Flag (CSR0) = 0

CONTINUOUS OPERATING MODE (TCR3 = 0, TCR5 = 0)

The timer may be programmed to operate in a continuous counting mode by writing zeros into bits 3 and 5 of the timer control register. Assuming that the timer output is enabled (TCR7 = 1), a square wave will be generated at the Timer Output CTO (see Table 4).

Either a Timer Reset (TCR0 = 1 or External $\overline{Reset}$ = 0) condition or internal recognition of a negative transition of the $\overline{CTG}$ input results in Counter Initialization. A Write Timer Latches command can be selected as a Counter Initialization signal by clearing TCR4.

The discussion of the Continuous Mode has assumed the application requires an output signal. It should be noted the Timer operates in the same manner with the output disabled (TCR7 = 0). A Read Timer Counter command is valid regardless of the state of TCR7.

NORMAL SINGLE-SHOT TIMER MODE (TCR3 = 0, TCR4 = 1, TCR5 = 1)

This mode is identical to the Continuous Mode with two exceptions. The first of these is obvious from the name – the output returns to a low-level after the initial Time Out and remains low until another Counter Initialization cycle occurs. The output waveform (CTO) is shown in Figure 16.

As indicated in Figure 16, the internal counting mechanism remains cyclical in the Single-Shot Mode. Each Time Out of the counter results in the setting of an Individual Interrupt Flag and re-initialization of the counter.

The second major difference between the Single-Shot and Continuous modes is that the internal counter enable is not dependent on the $\overline{CTG}$ input level remaining in the low state for the Single-Shot mode. Aside from these differences, the two modes are identical.

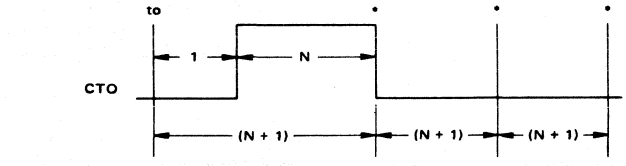
TABLE 4 – CONTINUOUS OPERATING MODES

CONTINUOUS MODE (TCR3 = 0, TCR7 = 1, TCR5 = 0)			
CONTROL REGISTER		INITIALIZATION/OUTPUT WAVEFORMS	
TCR2	TCR4	Counter	Timer Output (2X)
0	0	Initialization $\overline{CTG}\downarrow + \overline{W} + R$	←(N + 1) (T) → ←(N + 1) (T) → ←(N + 1) (T) → V_{OH} V_{OL}
0	1	$\overline{CTG}\downarrow + R$	to T.O. T.O. T.O.

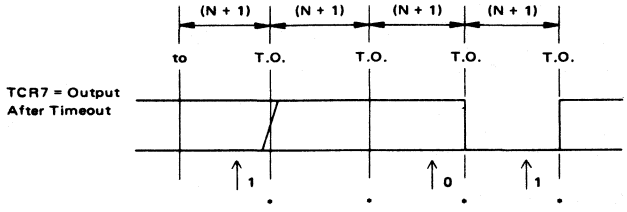
$\overline{CTG}$ = Negative Transition $\overline{GATE}$ Input.
$\overline{W}$ = Write Timer Latches Command.
R = Timer Reset (TCR0 = 1 or External $\overline{RESET}$ = 0)
N = 16 Bit Number in Counter Latch.

T = Period of Clock Input to Counter.
to = Counter Initialization Cycle.
T.O. = Counter Time Out (All Zero Condition).

FIGURE 16 – SINGLE-SHOT MODES



(A) NORMAL SINGLE-SHOT MODE OUTPUT WAVEFORM



(B) CASCADED SINGLE-SHOT MODE OUTPUT WAVEFORM

1 = Write a "1" into TCR-7
0 = Write a "0" into TCR-7

*POINT AT WHICH AN INTERRUPT MAY OCCUR

Note: All time intervals shown above assume the Gate ($\overline{CTG}$) and Clock ($\overline{CTC}$) signals are synchronized to system $\phi 2$ with the specified setup and hold time requirements.

TIME INTERVAL MODES (TCR3 = 1)

The Time Interval Modes are provided for applications requiring more flexibility of interrupt generation and Counter Initialization. The Interrupt Flag is set in these modes as a function of both Counter Time Out and transitions of the $\overline{CTG}$ input. Counter Initialization is also affected by Interrupt Flag status. The output signal is not defined in any of these modes. Other features of the Time Interval Modes are Outlined in Table 5.

TABLE 5 – TIME INTERVAL MODES

TCR3 = 1			
TCR4	TCR5	APPLICATION	CONDITION FOR SETTING INDIVIDUAL INTERRUPT FLAG
0	0	Frequency Comparison	Interrupt Generated if $\overline{CTG}$ Input Period (1/F) is Less Than Counter Time Out (T.O.)
0	1	Frequency Comparison	Interrupt Generated if $\overline{CTG}$ Input Period (1/F) is Greater Than Counter Time Out (T.O.)
1	0	Pulse Width Comparison	Interrupt Generated if $\overline{CTG}$ Input "Down Time" is Less Than Counter Time Out (T.O.)
1	1	Pulse Width Comparison	Interrupt Generated if $\overline{CTG}$ Input "Down Time" is Greater Than Counter Time Out (T.O.)

CASCADED SINGLE-SHOT MODE (TCR3 = 0, TCR4 = 0, TCR5 = 1)

This mode is identical to the single-shot mode with two exceptions. First, the output waveform does not return to a low level and remain low after timeout. Instead, the output level remains at its initialized level until it is re-programmed and changed by timeout. The output level may be changed at any timeout or may have any number of timeouts between changes.

The second difference is the method used to change the output level. Timer Control Register Bit 7 (TCR7) has a special function in this mode. The timer output (CTO) is equal to TCR7 clocked by timeout. At every timeout, the content of TCR7 is clocked to and held at the CTO output. Thus, output pulses of length greater than one timer cycle can be generated by cascading timer cycles and counting timeouts with a software program. (See Figure 16).

An interrupt is generated at each timeout. To cascade timer cycles, the MPU would need an interrupt routine to: 1) count each timeout and determine when to change TCR7; 2) write into TCR7 the state corresponding to the next desired state of the output waveform (only necessary during the last timer cycle before the output is to change state); and 3) clear the interrupt flag by reading the combination status register followed by Read Timer MSB. It is also possible, if desired, to change the length of the timer cycle by reinitializing the timer latches. This allows more flexibility for obtaining desired times.

FREQUENCY COMPARISON MODE (TCR3 = 1, TCR4 = 0)

The timer within the MC6846 may be programmed to compare the period of a pulse (giving the frequency after calculations) at the $\overline{CTG}$ input with the time period required for Counter Time Out. A negative transition of the $\overline{CTG}$ input enables the counter and starts a Counter Initialization cycle – provided that other conditions as noted in Table 6 are satisfied. The counter decrements on each clock signal recognized during or after Counter Initialization until an Interrupt is generated, a Write Timer Latches command is issued, or a Timer Reset condition occurs. It can be seen from Table 6 that an interrupt condition will be generated if TCR5 = 0 and the period of the pulse (single pulse or measured separately repetative pulses) at the $\overline{CTG}$ input is less than the Counter Time Out period. If TCR5 = 1, an interrupt is generated if the reverse is true.

Assume now with TCR5 = 1 that a Counter Initialization has occurred and that the $\overline{CTG}$ input has returned low prior to Counter Time Out. Since there is no Individual Interrupt Flag generated, this automatically starts a new Counter Initialization Cycle. The process will continue with frequency comparison being performed on each $\overline{CTG}$ input cycle until the mode is changed, or a cycle is determined to be above the predetermined limit.

PULSE WIDTH COMPARISON MODE (TCR3 = 1, TCR4 = 1)

This mode is similar to the Frequency Comparison Mode except for the limiting factor being a positive, rather than negative, transition of the $\overline{CTG}$ input. With TCR5 = 0, an Individual Interrupt Flag will be generated if the zero level pulse applied to the $\overline{CTG}$ input is less than the time period required for Counter Time Out. With TCR5 = 1, the interrupt is generated when the reverse condition is true.

As can be seen in Table 7, a positive transition of the $\overline{CTG}$ input disables the counter. With TCR 5 = 0, it is therefore possible to directly obtain the width of any pulse causing an interrupt.

TABLE 6 – FREQUENCY COMPARISON MODE

CRX3 = 1, CRX4 = 0				
Control Reg Bit 5 (CRX5)	Counter Initialization	Counter Enable Flip-Flop Set (CE)	Counter Enable Flip-Flop Reset (CE)	Interrupt Flag Set (I)
0	$\overline{G}\downarrow\cdot\overline{I}\cdot(\overline{CE}+TO\cdot CE)+R$	$\overline{G}\downarrow\cdot\overline{W}\cdot\overline{R}\cdot\overline{I}$	W+R+I	$\overline{G}\downarrow$ Before TO
1	$\overline{G}\downarrow\cdot\overline{I}+R$	$\overline{G}\downarrow\cdot\overline{W}\cdot\overline{R}\cdot\overline{I}$	W+R+I	TO Before $\overline{G}\downarrow$

I represents the interrupt for the timer.

TABLE 7 – PULSE WIDTH COMPARISON MODE

CRX3 = 1, CRX4 = 1				
Control Reg Bit 5 (CRX5)	Counter Initialization	Counter Enable Flip-Flop Set (CE)	Counter Enable Flip-Flop Reset (CE)	Interrupt Flag Set (I)
0	$\overline{G}\downarrow\cdot\overline{I}+R$	$\overline{G}\downarrow\cdot\overline{W}\cdot\overline{R}\cdot\overline{I}$	W+R+I+G	$\overline{G}\uparrow$ Before TO
1	$\overline{G}\downarrow\cdot\overline{I}+R$	$\overline{G}\downarrow\cdot\overline{W}\cdot\overline{R}\cdot\overline{I}$	W+R+I+G	TO Before $\overline{G}\uparrow$

DIFFERENCES BETWEEN THE MC6840 AND THE MC6846 TIMERS

1) Control registers 1 and 3 are buried (access through control register 2 only) in the MC6840 timer. In the MC6846, all registers are directly accessable.

2) The MC6840 has a dual 8 bit continuous mode for generating non-symmetrical waveforms. The MC6846, instead, has a cascaded one shot mode which can accomplish the same function, but also allows the user to generate waveforms longer than one timeout.

3) Because of the different modes, there is a difference in the control registers between the MC6840 and the MC6846.

CONTROL REGISTER BIT	MC6840	MC6846
2	16 bit or dual 8 bit mode control	÷ 8 prescale enable
7	output enable (all modes)	output next state (cascaded one shot mode only), output enable all other modes
0	R_1 internal reset R_2 control register select R_3 timer 3 clock control	internal reset

COMPOSITE STATUS REGISTER

The Composite Status Register (CSR) is a read-only register which is shared by the Timer and the Peripheral Data Port of the MC6846. Three individual interrupt flags in the register are set directly via the appropriate conditions in the timer or peripheral port. The composite interrupt flag – and the $\overline{IRQ}$ Output – respond to these individual interrupts only if corresponding enable bits are set in the appropriate Control Registers. (See Figure 17.) The sequence of assertion is not detected. Setting TCR6 while CSR0 is high will cause CSR7 to be set, for example.

The Composite Interrupt Flag (CSR7) is clear only if all enabled Individual Interrupt Flags are clear. The conditions for clearing CSR1 and CSR2 are detailed in a later section. The Timer Interrupt Flag (CSR0) is cleared under the following conditions:

1) Timer Reset – Internal Reset Bit (TCR0) = 1 or External $\overline{Reset}$ = 0.

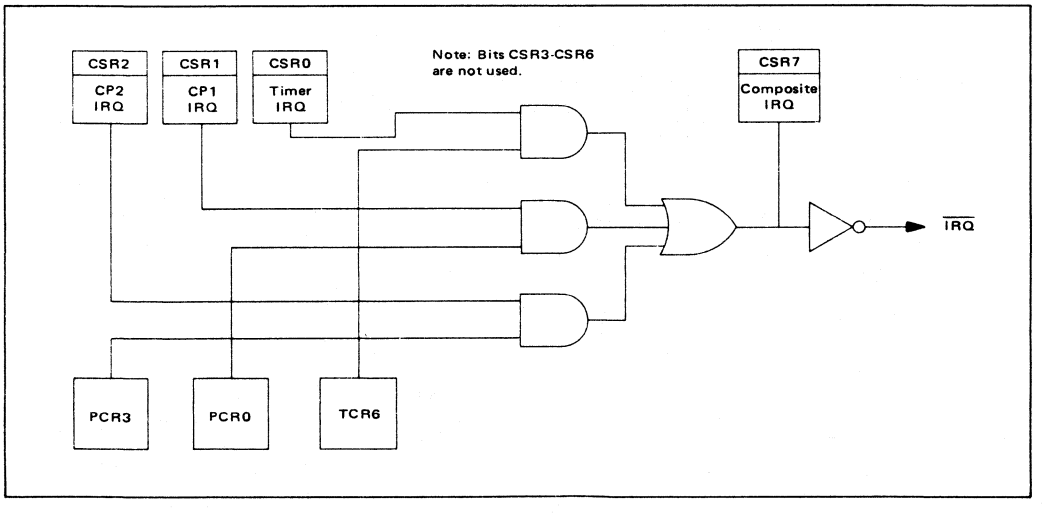
2) Any Counter Initialization condition.

3) A Write Timer Latches command if Time Interval modes (TCR3 = 1) are being used.

4) A Read Timer Counter command, provided this is preceded by a Read Composite Status Register while CSR0 is set. This latter condition prevents missing an Interrupt Request generated after reading the Status Register and prior to reading the counter.

The remaining bits of the Composite Status Register (CSR3-CSR6) are unused. They default to a logic zero when read.

FIGURE 17 – COMPOSITE STATUS REGISTER & ASSOCIATED LOGIC



I/O OPERATION

PARALLEL PERIPHERAL PORT

The peripheral port of the MC6846 contains 8 Peripheral Data lines (P0-P7), two Peripheral Control lines (CP1 and CP2), a Data Direction Register, a Peripheral Data Register, and a Peripheral Control Register. The port also directly affects two bits (CSR1 and CSR2) of the Composite Status Register.

The Peripheral Port is similar to the "B" side of a PIA (MC6820 or MC6821) with the following exceptions:

1) All registers are directly accessible in the MC6846. Data Direction and Peripheral Data in the MC6820/6821 are located at the same address, with Bit Two of the Control Register used for register selection.

2) Peripheral Control Register Bit Two (PCR2) of the MC6846 is used to select an optional input latch function. This option is not available with MC6820/6821 PIA's.

3) Interrupt Flags are located in the MC6846 composite status register rather than Bits 6 and 7 of the Control Register as used in the MC6820/6821.

4) Interrupt Flags are cleared in the MC6820/6821 by reading data from the Peripheral Data Register. MC6846 Interrupt Flags are cleared by either reading or writing to the Peripheral Data Register – provided that this sequence is followed a) Flag Set, b) Read Composite Status Register, c) Read/Write Peripheral Data Register is followed.

5) Bit 6 of the MC6846 Peripheral Control Register is not used. Bit 7 (PCR7) is an Internal Reset Bit not available on the MC6820/6821.

6) The Peripheral Data lines (and CP2) of the MC6846 feature internal current limiting which allows them to directly drive the base of Darlington NPN transistors.

DATA DIRECTION REGISTER

The MPU can write directly to this eight-bit register to configure the Peripheral Data lines as either inputs or outputs. A particular bit within the register (DDRN) is used to control the corresponding Peripheral Data line (PN). With DDRN = 0, PN becomes an input; if DDRN = 1, PN is an output. As an example, writing Hex $OF into the Data Direction Register results in P0 thru P3 becoming outputs and P4 thru P7 being inputs. Hex $55 in the Data Direction Register results in alternate outputs and inputs at the parallel port.

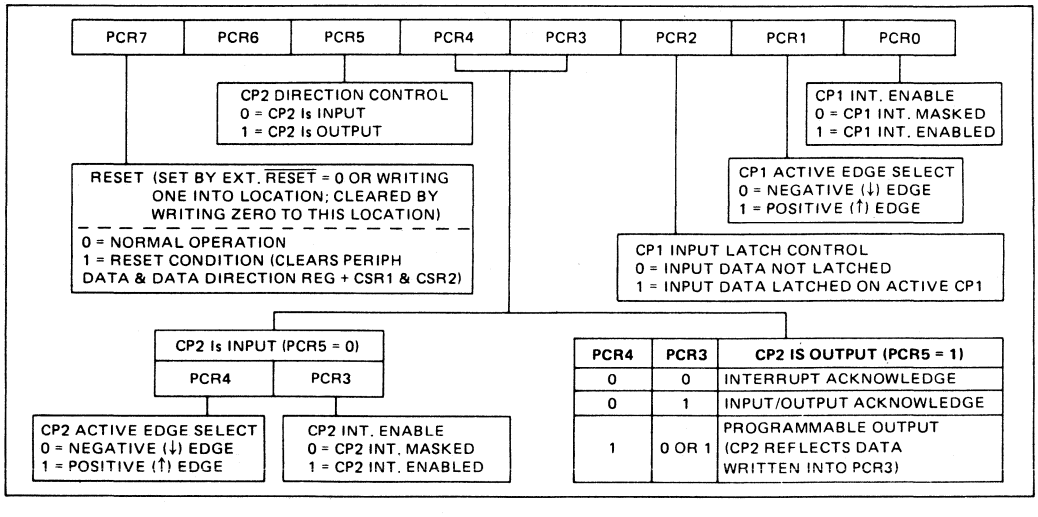
PERIPHERAL DATA REGISTER

This eight-bit register is used for transferring data between the peripheral data port and the MPU. Any bit corresponding to an output line will be used to drive the output buffer associated with that line. Data in these output bits is normally provided by an MPU Write function. (Input bits – those associated with input lines – are unchanged by a Write Command.) Any input bit will reflect the state of the associated input line if the input latch function is deselected. If the Control Register is programmed to provide input latching, the input bit will retain the state at the time CP1 was activated until the Peripheral Data Register is read by the MPU.

PERIPHERAL CONTROL REGISTER

This eight-bit register is used to control the reset function as well as for selection of optional functions of the two peripheral control lines (CP1 and CP2). The Peripheral Control Register functions are outlined in Table 8.

TABLE 8 – PERIPHERAL CONTROL REGISTER FORMAT (Expanded)



PCR4	PCR3	CP2 IS OUTPUT (PCR5 = 1)
0	0	INTERRUPT ACKNOWLEDGE
0	1	INPUT/OUTPUT ACKNOWLEDGE
1	0 OR 1	PROGRAMMABLE OUTPUT (CP2 REFLECTS DATA WRITTEN INTO PCR3)

PERIPHERAL PORT RESET (PCR7)

Bit 7 of the Peripheral Control Register (PCR7) may be used to initialize the peripheral section of the MC6846. When this bit is set high, the peripheral data register, the peripheral data direction register, and the interrupt flags associated with the peripheral port (CSR1 & CSR2) are all cleared. Other bits in the peripheral control register are not affected by PCR7.

PCR7 is set by either a logic zero at the External $\overline{\text{RESET}}$ input or under program control by writing a "one" into the location. In any case, PCR7 may be cleared only by writing a zero into the location while $\overline{\text{RESET}}$ is high. The bit must be cleared to activate the port.

CONTROL OF CP1 PERIPHERAL CONTROL LINE

CP1 may be used as an interrupt request to the MC6846, as a strobe to allow latching of input data, or both. In any case, the input can be programmed to be activated by either a positive or negative transition of the signal. These options are selected via Control Register Bits PCR0, PCR1 & PCR2.

Control Register Bit 0 (PCR0) is used to enable the interrupt transfer circuitry of the MC6846. Regardless of the state of PCR0, an active transition of CP1 causes the Composite Status Register Bit One (CSR1) to be set. If PCR0 = 1, this interrupt will be reflected in the Composite Interrupt Flag (CSR7), and thus at the $\overline{\text{IRQ}}$ output. CSR1 is cleared by a Peripheral Port Reset condition or by either reading or writing to the peripheral data register after the Composite Status Register is read. The latter alternative is conditional -- CSR1 must have been a logic one when the Composite Status Register was last read. This precludes inadvertent clearing of interrupt flags generated between the time the Status Register is read and the manipulation of peripheral data.

Control Register Bit One (PCR1) is used to select the edge which activates CP1. When PCR1 = 0, CP1 is active on negative transitions (high to low). Low to High transitions are sensed by CP1 when PCR1 = 1.

In addition to its use as an interrupt input, CP1 can be used as a strobe to capture input data in an internal latch. This option is selected by writing a one into Peripheral Control Register Bit Two (PCR2). In operation, the data at the pins designated by the Data Direction Register as inputs will be captured by an active transition of CP1. An MPU Read of the Peripheral Data Register will result in the captured data being transferred to the MPU -- and it also releases the latch to allow capture of new data. Note that successive active transitions with no Read Peripheral Data Command between does not update the input latch. Also, it should be noted that use of the input latch function (which can be deselected by writing a zero into PCR2) has no effect on output data. It also does not affect Interrupt function of CP1.

CONTROL OF CP2 PERIPHERAL CONTROL LINE

CP2 may be used as an input by writing a zero into PCR5. In this configuration, CP2 becomes a dual of CP1 in regard to generation of interrupts. An active transition (as selected by PCR4) causes Bit Two of the Composite Status Register to be set. PCR3 is then used to select whether the CP2 transition is to cause CSR7 to be set -- and thereby cause $\overline{\text{IRQ}}$ to go low. CP2 has no effect on the input latch function of the MC6846.

Writing a one into PCR5 causes CP2 to function as an output. PCR4 then determines whether CP2 is to be used in a handshake or programmable output mode. With PCR4 = 1, CP2 will merely reflect the data written into PCR3. Since this can readily be changed under program control, this mode allows CP2 to be a programmable output line in much the same manner as those lines selected as outputs by the Data Direction Register.

The handshaking mode (PCR5 = 1, PCR4 = 0) allows CP2 to perform one of two functions as selected by PCR3. With PCR3 = 1, CP2 will go low on the first Enable (System $\phi 2$) positive transition after a Read or Write to the Peripheral Data Register. This $\overline{\text{Input/Output Acknowledge}}$ signal is released (returns high) on the next positive transition of the Enable signal.

In the $\overline{\text{Interrupt Acknowledge}}$ mode (PCR5 = 1, PCR4 = PCR3 = 0), CP2 is set when CSR1 is set by an active transition of CP1. It is released (goes low) on the first positive transition of Enable after CSR1 has been cleared via an MPU Read or Write to the Peripheral Data Register. (Note that the previously described conditions for clearing CSR1 still apply.)

RESTART SEQUENCE

A typical restart sequence for the MC6846 will include initialization of both the Peripheral Control & Data Direction Registers of the parallel port. It is necessary to set up the Peripheral Control Register first, since PCR7 = 0 is a condition for writing data into the Data Direction Register. (A logic zero at the external $\overline{\text{Reset}}$ input automatically sets PCR7.)

SUMMARY

The MC6846 has several optional modes of operation which allow it to be used in a variety of applications. The following tables are provided for reference in selecting these modes.

TABLE 9 – MC6846 INTERNAL REGISTER ADDRESSES

A2	A1	A0	REGISTER SELECTED
0	0	0	Combination Status Register
0	0	1	Peripheral Control Register
0	1	0	Data Direction Register
0	1	1	Peripheral Data Register
1	0	0	Combination Status Register
1	0	1	Timer Control Register
1	1	0	Timer MSB Register
1	1	1	Timer LSB Register
X	X	X	ROM Address

TABLE 10 – COMPOSITE STATUS REGISTER

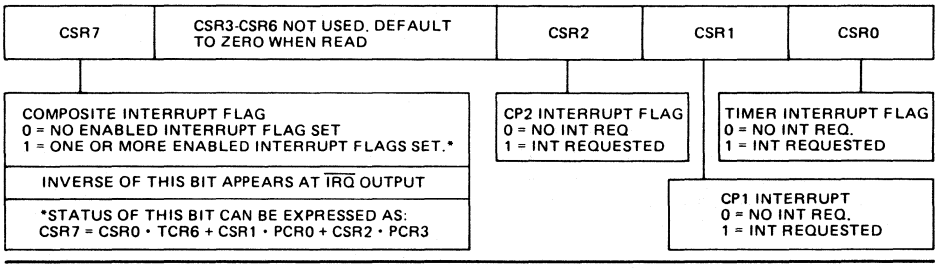
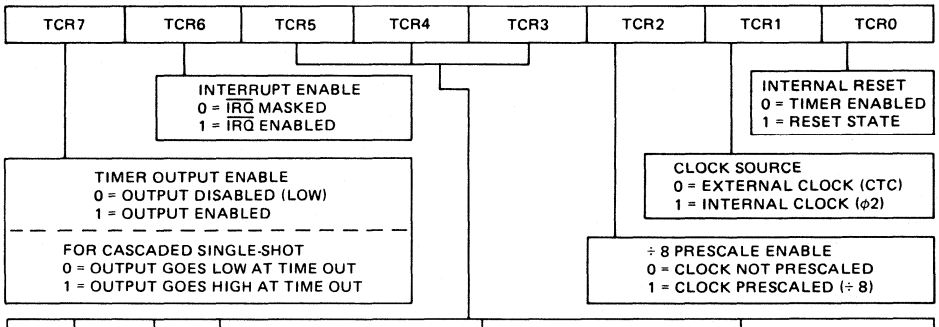


TABLE 11 – TIMER CONTROL REGISTER

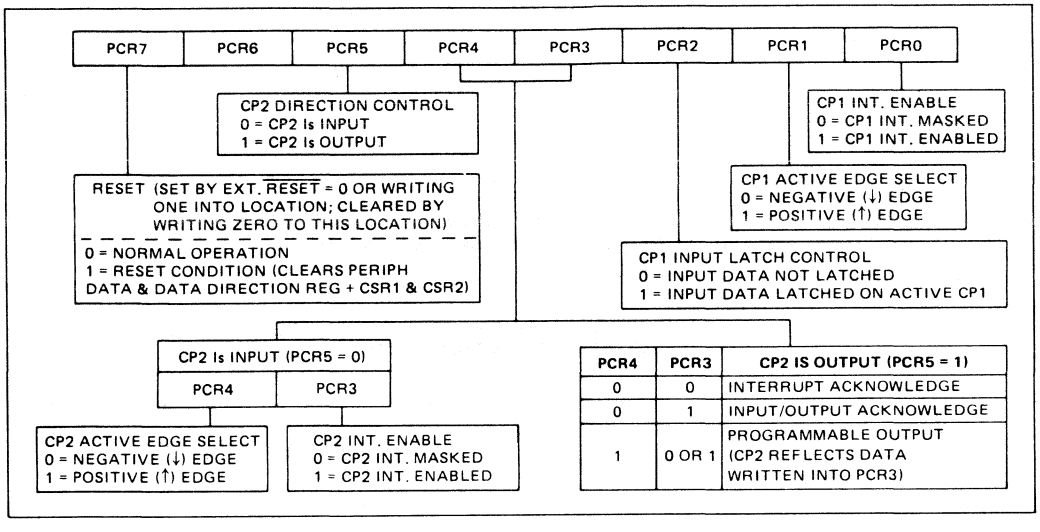


TCR3	TCR4	TCR5	TIMER OPERATING MODE	COUNTER INITIALIZATION	INTERRUPT FLAG SET
0	0	0	CONTINUOUS	$\overline{CTG}\downarrow$ + W + R	T.O.
0	0	1	CASCADED SINGLE SHOT	$\overline{CTG}\downarrow$ + R	T.O.
0	1	0	CONTINUOUS	$\overline{CTG}\downarrow$ + R	T.O.
0	1	1	NORMAL SINGLE SHOT	$\overline{CTG}\downarrow$ + R	T.O.
1	0	0	FREQUENCY COMPARISON	$\overline{CTG}\downarrow \cdot \overline{T} \cdot$ (W + T.O.) + R	$\overline{CTG}\downarrow$ BEFORE T.O.
1	0	1		$\overline{CTG}\downarrow \cdot \overline{T}$ + R	T.O. BEFORE $\overline{CTG}\downarrow$
1	1	0	PULSE WIDTH COMPARISON	$\overline{CTG}\downarrow \cdot \overline{T}$ + R	$\overline{CTG}\uparrow$ BEFORE T.O.
1	1	1			T.O. BEFORE $\overline{CTG}\uparrow$

R = RESET CONDITION
W = WRITE TIMER LATCHES
T.O. = COUNTER TIME OUT

$\overline{CTG}\downarrow$ = NEG TRANSITION OF PIN 17
$\overline{CTG}\uparrow$ = POS TRANSITION OF PIN 17
$\overline{T}$ = INTERRUPT FLAG (CSR0) = 0

TABLE 12 – PERIPHERAL CONTROL REGISTER



CUSTOM PROGRAMMING

By the programming of a single photomask for the MC6846, the customer may specify the content of the memory and the method of enabling the outputs.

Information on the general options of the MC6846 should be submitted on an Organizational Data form such as that shown in Figure 18.

Information for custom memory content may be sent to Motorola in one of two forms (shown in order of preference):

1. Paper tape output of the Motorola M6800 Software.
2. Hexadecimal coding using IBM Punch Cards.

PAPER TAPE

Included in the software packages developed for the M6800 Microcomputer Family is the ability to produce a paper tape output for computerized mask generation. The assembler directives are used to control allocation of memory, to assign values for stored data, and for controlling the assembly process. The paper tape must specify the full 2048 bytes.

Note: Motorola can accept magnetic tape and truth table formats. For further information, contact your local Motorola sales representative.

IBM PUNCH CARDS

The hexadecimal equivalent (from Table 13) may be placed on 80 column IBM punch cards as follows:

Step	Column	
1	12	Byte "0" Hexadecimal equivalent for outputs D7 thru D4 (D7 = M.S.B.)
2	13	Byte "0" Hexadecimal equivalent for outputs D3 thru D0 (D3 = M.S.B.)
3	14-75	Alternate steps 1 and 2 for consecutive bytes.
	76	Blank
4	77-78	Card number (starting 01)
5	79-80	Total number of cards (32)

TABLE 13 – BINARY TO HEXADECIMAL CONVERSION

Binary Data				Hexadecimal Character
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	A
1	0	1	1	B
1	1	0	0	C
1	1	0	1	D
1	1	1	0	E
1	1	1	1	F

FIGURE 18 – FORMAT FOR PROGRAMMING GENERAL OPTIONS

ORGANIZATIONAL DATA
MC6846 COMBINATION ROM-I/O-TIMER

Customer:

Company _____

Part No. _____

Originator _____

Phone No. _____

Motorola Use Only:

Quote: _____

Part No.: _____

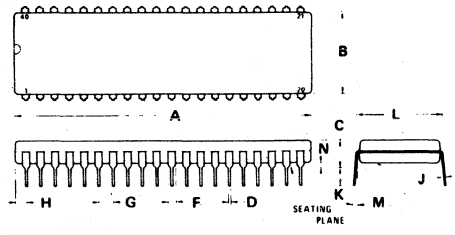
Specif. No.: _____

Enable Options: (ROM ENABLE MUST DIFFER FROM I/O-TIMER)

	ROM SECTION 1	ROM SECTION 0	I/O-TIMER SECTION 1	I/O-TIMER SECTION 0
CS0	☐	☐	☐	☐
CS1	☐	☐	☐	☐

I/O-TIMER SELECT A6 1	0	X	CHECK ONE COLUMN ONLY					
☐	☐	☐						
		A10	X	1	X	X	X	1 ⩾ 2.0V.
		A9	X	X	1	X	X	0 ⩽ 0.8V.
		A8	X	X	X	1	X	X =
		A7	X	X	X	X	1	NOT USED

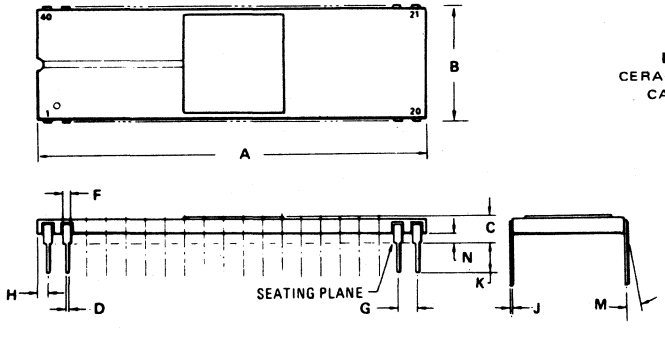
P SUFFIX
PLASTIC PACKAGE
CASE 711-02



DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	51.82	52.32	2.040	2.060
B	13.72	14.22	0.540	0.560
C	4.57	5.08	0.180	0.200
D	0.36	0.51	0.014	0.020
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	3.05	3.56	0.120	0.140
L	15.24 BSC		0.600 BSC	
M	0°	10°	0°	10°
N	0.51	1.02	0.020	0.040

NOTES:
1. LEADS TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION (DIM "D").
2. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.

L SUFFIX
CERAMIC PACKAGE
CASE 715-02



DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	50.29	51.31	1.980	2.020
B	14.86	15.62	0.585	0.615
C	2.54	4.19	0.100	0.165
D	0.38	C.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.33	0.008	0.013
K	2.54	4.19	0.100	0.165
M	0°	10°	0°	10°
N	0.51	1.52	0.020	0.060

NOTE:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA (AT SEATING PLANE), AT MAX. MAT'L CONDITION.

MOTOROLA

MC6847
NONINTERLACE
MC6847Y
INTERLACE

Advance Information

VIDEO DISPLAY GENERATOR (VDG)

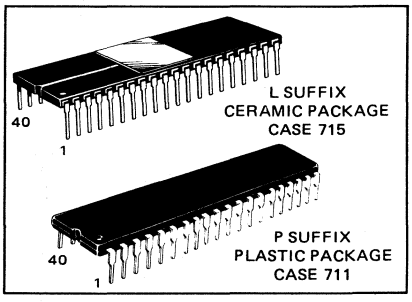
The Motorola MC6847 Video Display Generator (VDG) provides a means of interfacing the Motorola M6800 microprocessor family (or similar products) to a commercially available color or black and white television receiver. Applications of the VDG include video games, bioengineering displays, education, communications and any place graphics are required.

The VDG reads data from memory and produces a composite video signal which will allow the generation of alphanumeric or graphic displays. The generated composite video may be up modulated to either Channel 3 or 4 by using the compatible MC1372 (TV Chroma and Video modulator). The up modulated signal is suitable for application to the antenna of a color TV. A typical TV game is indicated in Figure 1.

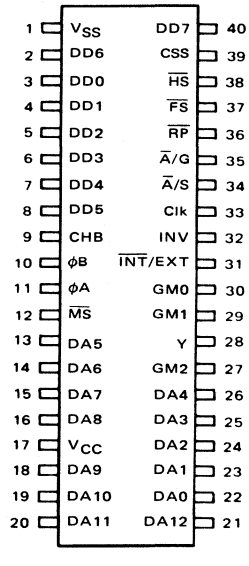
- Generates four different alphanumeric display modes and eight graphic display modes
- Compatible with the M6800 family
- Compatible with the MC1372 modulator
- The alphanumeric modes display 32 characters per line by 16 lines
- An internal multiplexer allows the use of either the internal ROM or an external character generator
- An external character generator can be used to extend the internal character set for "limited graphic" shapes
- A Mask Programmable internal character generator ROM is available on special order (Appendix A)
- One display mode offers 8-color 64 x 32 density graphics in an alphanumeric display mode
- One display mode offers 4-color 64 x 48 density graphics in an alphanumeric display mode
- All alphanumeric modes have a selectable video inverse
- Generates full video signal
- Generates R-Y and B-Y signals for external color modulator
- Full-graphic modes offer 64 x 64, 128 x 64, 128 x 96, 128 x 192, or 256 x 192 densities
- Full-graphic modes allow 2-color or 4-color data structures
- Full-graphic modes use one of two 4-color sets or one of two 2-color sets
- Available in either an interlace mode (NTSC Standard) or a non-interlace mode

MOS
(N-Channel, Silicon-Gate)

VIDEO DISPLAY GENERATOR



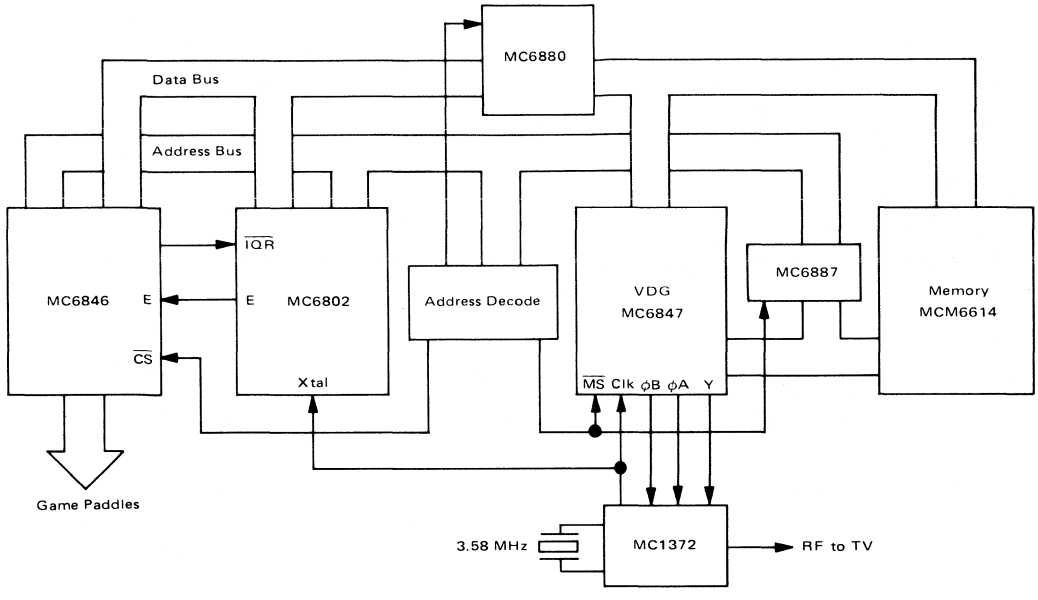
PIN ASSIGNMENT



This is advance information and specifications are subject to change without notice.

ADI-492 R1

FIGURE 1 – BLOCK DIAGRAM OF USE OF THE VDG IN A TV GAME



Mnemonic	Pin Numbers	Function
V_{CC}	17	+5V
V_{SS}	1	Ground
CLK	33	Color burst clock 3.58 MHz (input)
DA0-DA12	22, 23, 24, 25, 26 13, 14, 15, 16, 18, 19, 20, 21	Address lines to display memory, high impedance during memory select ($\overline{MS}$)
DD0-DD5	3, 4, 5, 6, 7, 8	Data from display memory RAM or ROM
DD6, DD7	2, 40	Data from display memory in graphic mode; data also in alpha external mode; color data in alpha semigraphic 4 or 6
φA, φB, Y	11, 10, 28	Chrominance and luminance analog (R-Y, B-Y, Y) output to RF modulator (MC1372)
CHB	9	Chroma bias; reference φA and φB levels
$\overline{RP}$	36	Row preset – Output to provide timing for external character generator.
$\overline{HS}$	38	Horizontal Sync – Output to provide timing for external character generator.
INV	32	Inverts video in all alpha modes
$\overline{INT}$/EXT	31	Switches to external ROM in alpha mode and between SEMIG-4 and SEMIG-6 in semigraphics
$\overline{A}$/S	34	Alpha/Semigraphics; selects between alpha and semigraphics in alpha mode
$\overline{MS}$	12	Memory select forces VDG address buffers to high-impedance state
$\overline{A}$/G	35	Switches between alpha and graphic modes
$\overline{FS}$	37	Field Synchronization goes low at bottom of active display area.
CSS	39	Color set select; selects between two alpha display colors or between two color sets in semigraphics 6 and full graphics
GM0-GM2	30, 29, 27	Graphic mode select; select one of eight graphic modes.

ELECTRICAL SPECIFICATIONS

ABSOLUTE MAXIMUM RATINGS

Rating	Value
Supply Voltage (V_{CC})	-0.3 to + 7.0V
Input Voltage any Pin	-0.3 to + 7.0V
Operating Temperature	0°C to 70°C
Storage Temperature	-65°C to 150°C
Power Dissipation	TBD

DC (STATIC) CHARACTERISTICS – (V_{CC} = 5.0 V ±5%, V_{SS} = 0.0 V, T_A = 0°C to 70°C unless otherwise noted)

Characteristic		Symbol	Min	Typ.	Max.	Unit
Input High Voltage Clk Other Inputs		V_{IH}	 V_{SS} + 2.4 V_{SS} + 2.0	 – –	 V_{CC} V_{CC}	Vdc
Input Low Voltage Clk Other Inputs		V_{IL}	 V_{SS} - 0.3 V_{SS} - 0.3	 – –	 V_{SS} + 0.4 V_{SS} + 0.8	Vdc
Input Leakage Current CLK, GM0-GM2, INV, $\overline{INT}$/EXT, $\overline{MS}$, V_{SS}, DD0-DD7, $\overline{A}$/S, $\overline{A}$/G		I_{in}	–	–	2.5	μAdc
Three-State (Off State) Input Current DA0-DA12		I_{LO}	–	–	10	μAdc
Ohtput High Voltage (C_{Load} = 30 pF, I_{Load} = -100 μA)	$\overline{RP}$, $\overline{HS}$, $\overline{FS}$	V_{OH}	2.4	–	–	Vdc
Output High Voltage (C_{Load} = 55 pF, I_{Load} = -100 μA)	DA0-DA12	V_{OH}	2.4	–	–	Vdc
Output Load Voltage (C_{Load} = 30 pF, I_{Load} = 1.6 mA)	$\overline{RP}$, $\overline{HS}$, $\overline{FS}$	V_{OL}	–	–	V_{SS} + 0.4	Vdc
Output Low Voltage (C_{Load} = 55 pF, I_{Load} = 1.6 mA)	DA0-DA12	V_{OL}	–		V_{SS} + 0.4	Vdc
Output High Current (Sourcing) (V_{OH} = 2.4 V)	All Outputs (except φA, φB, Y, & CHB)	I_{OH}	-100	–	–	μAdc
Output Low Current (Sinking) (V_{OL} = 0.4 Vdc)	All Outputs (except φA, φB, Y, & CHB)	I_{OL}	1.6	–	–	mAdc
Input Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	All Inputs	C_{in}	–	–	7.5	pF
Chroma Bias Voltage (C_{Load} = 20 pF, R Load = 200 k ohm, V_{CC} = 4.75 – 5.25 V)		V_R	–	0.3 V_{CC}	–	Vdc

DC (STATIC) CHARACTERISTICS — (V_{CC} = 5.0 V ± 5%, V_{SS} = 0.0 V, T_A = 0°C to 70°C unless otherwise noted)

Characteristic	Symbol	Min	Typ.	Max.	Unit
Chroma ϕA Voltage Figure 2	$V_{C\phi A}$				Vdc
(C_{Load} = 20 pF, R Load = 200 k ohm)	V_{HI}	–	V_R + 0.1 V_{CC}	–	
	V_0	–	V_R	–	
	V_{LO}	–	V_R – 0.1 V_{CC}	–	
Chroma ϕB Voltage	$V_{C\phi B}$				Vdc
(C_{Load} = 20 pF, R Load = 200 k ohm)		–	V_R + 0.1 V_{CC}	–	
V_0		–	V_R	–	
V_{burst}		–	V_R – 0.05 V_{CC}	–	
V_{LO}		–	V_R – 0.1 V_{CC}	–	
Luminance Y Voltage Figure 2	V_Y				Vdc
(C_{Load} = 20 pF, R_{Load} = 200 k ohm)	V_S	–	0.2 V_{CC}	–	
	V_{BLANK}	–	0.75 V_S	–	
	V_{BLACK}	–	0.7 V_S	–	
Voltage White Low Figure 2	V_{WL}	–	0.62 V_S	–	Vdc
(Voltage White Medium)	V_{WM}	–	0.5 V_S	–	
(Voltage White High)	V_{WH}	–	0.38 V_S	–	

AC (Dynamic) CHARACTERISTICS — V_{CC} = 5.0 V ± 5%, T_A = 0°C to 70°C

Characteristic	Symbol	Min	Typ.	Max.	Unit
Clk Frequency	f	3.579535	3.579545	3.579555	MHz
Clk Duty Cycle	Clk_{dc}	45%	50%	55%	
Chroma Phase Delay Figure 3C					ns
(measured with respect to "Y" output)					
ϕA	t_{YA}	–	200	–	
ϕB	t_{YB}	–	200	–	
Luminance Rise Time Figure 3D	t_{ry}	–	60	–	ns
Luminance Fall Time	t_{fy}	–	50	–	
Chroma Rise and Fall Times Figure 3D					ns
(ϕA Rise Time)	$t_rC\phi A$	–	60	–	
(ϕA Fall Time)	$t_fC\phi A$	–	60	–	
(ϕB Rise Time)	$t_rC\phi B$	–	60	–	
(ϕB Fall Time)	$t_fC\phi B$	–	60	–	
Field Sync. ($\overline{FS}$) Figure 3A	t_{WFS}	–	14.6	–	ms
(Pulse Width)					
Row Present ($\overline{RP}$) Figure 3B					
(Pulse Width)	t_{WRP}	–	0.98	–	μs
(Delay From $\overline{HS}$)	t_{HSRP}	–	0.98	–	μs
Horizontal Sync ($\overline{HS}$) Figure 3B	t_{WHS}	–	4.9	–	μs

FIGURE 2 – VIDEO AND CHROMINANCE RELATIONSHIPS OUTPUT WAVEFORM

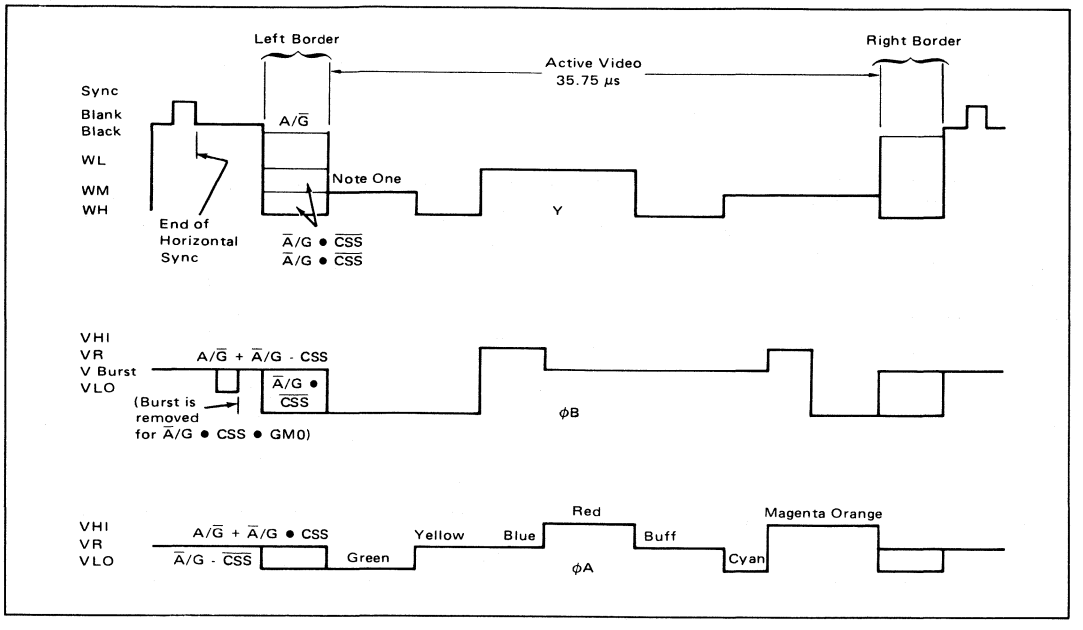
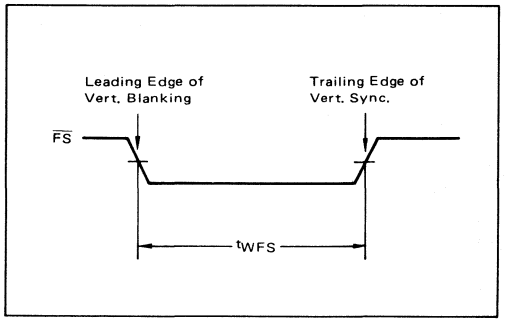
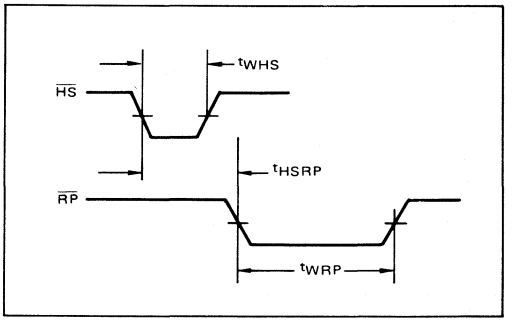


FIGURE 3 – TIMING DIAGRAMS

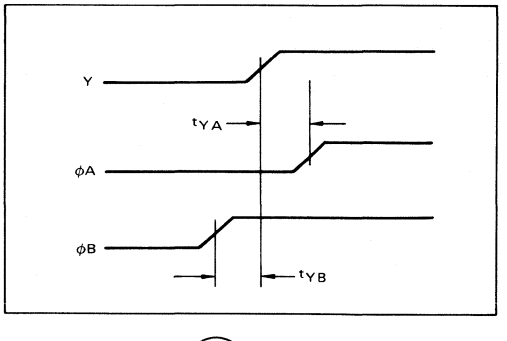
A– Field Sync



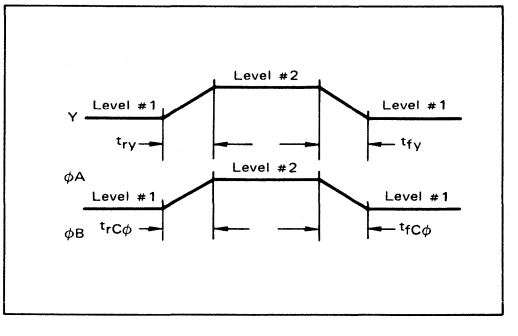
B – Row Preset



C – Chroma Phase Delay



D – Video Rise & Fall Times



VDG SIGNAL DESCRIPTION

Address Output Lines (DA0-DA12) – Thirteen address lines are used by the VDG to scan the display memory. The starting address of the display memory is located at the upper left corner of the display screen. As the television sweeps from the left to right and top to bottom, the VDG increments the RAM display address. These lines are TTL compatible and may be forced into a high impedance state whenever the $\overline{MS}$ pin goes low.

Data Inputs (DD0-DD7) – Eight TTL compatible data lines are used to input data from RAM to be processed by the VDG. The data is interpreted and transformed into luminance Y (Pin 28) and color outputs ϕA and ϕB (Pin 11 and Pin 10).

Power Inputs – V_{CC} requires +5 volts. V_{SS} requires zero volts and is normally ground. The tolerance and current requirements of the VDG are specified in the Electrical Characteristics.

Video Outputs (ϕA, ϕB, Y, CHB) – These four analog outputs are used to transfer luminance and color information to a standard NTSC color television receiver, either via the MC1372 RF modulator or directly into Y, ϕA, ϕB television video inputs.

LUMINANCE (Y) – This six level analog output contains composite sync., blanking and four levels of video luminance.

ϕA – This three level analog output is used in combination with ϕB and Y outputs to specify one of eight colors.

ϕB – This four level analog output is used in combination with ϕA and Y outputs to specify one of eight colors. Additionally, one analog level is used to specify the time of the color burst reference signal.

CHROMA BIAS (CHB) – This pin is an analog output and provides a D.C. reference corresponding to the quiescent value of ϕA and ϕB. CHB is used to guarantee good thermal tracking and minimize the variation between the parts.

Synchronizing Inputs ($\overline{MS}$, CLK)

Three-State Control – ($\overline{MS}$) is a TTL compatible input which, when low, forces the VDG address lines into a high impedance state. This may be done to allow other devices (such as an MPU) to address the display memory (RAM).

Clock (CLK) – The VDG clock input (CLK) requires a 3.579545 MHz (standard) TV crystal frequency square wave. The duty cycle of this clock must be between 45 and 55 percent since it controls the width of alternate dots on the television screen. The MC1372 RF modulator may be used to supply the 3.579545 MHz clock and has provisions for a duty cycle adjustment.

Synchronizing Outputs ($\overline{FS}$, $\overline{HS}$, $\overline{RP}$) – Three TTL compatible outputs provide circuits, exterior to the VDG, with timing references to the following internal VDG states:

FIELD SYNC – ($\overline{FS}$) – The high to low transition of the $\overline{FS}$ output coincides with the end of active display area. During this time interval an MPU may have total access to the display RAM without causing undesired flicker on the screen. The Low to High transition of $\overline{FS}$ coincides with the trailing edge of the vertical synchronization pulse.

HORIZONTAL SYNC – ($\overline{HS}$) – The $\overline{HS}$ pulse is in coincidents with the horizontal synchronization pulse furnished to the television receiver by the VDG. The high to low transition of the $\overline{HS}$ output coincides with the leading edge of the horizontal synchronization pulse.

ROW PRESET – ($\overline{RP}$) – If desired, an external character generator ROM may be used with the VDG. However, an external four bit counter must be added to supply row selection. The counter is clocked by the $\overline{HS}$ signal and cleared by the $\overline{RP}$ signal.

Mode Control Lines (Input) ($\overline{A}$/G, $\overline{A}$/S, $\overline{INT}$/EXT, GM0, GM1, GM2, CSS, INV) – Eight TTL compatible inputs are used to control the operating mode of the VDG. $\overline{A}$/S, $\overline{INT}$/EXT, CSS and INV may be changed on a character by character basis. The CSS pin is used to select between two possible alphanumeric colors; when the VDG is in the alphanumeric mode and between two color sets when the VDG is in the semigraphics 6 and full Graphic mode. Table 1 illustrates the various modes that can be obtained using the mode control lines.

DISPLAY MODES

The VDG is capable of generating 12 distinct display modes (refer to Table 1). The color set selection and invert pins will allow variations on certain modes. The VDG will display two alphanumeric modes with two compatible semigraphic modes or display one of eight full graphic modes. A detailed description of the various modes of operation follows. A summary of major modes can be found in Table 2.

ALPHANUMERIC DISPLAY MODES – All alphanumeric modes occupy an 8 x 12 dot character matrix box and there are 32 x 16 character boxes per TV frame. Each horizontal dot (dot-clock) corresponds to one-half the period duration of the 3.58 MHz clock and each vertical dot is one scan line. One of two colors for the lighted dots may be selected by the color set select pin. An internal ROM will generate 64 ASCII display characters in a standard 5 x 7 box. Six bits of the eight-bit data word are used for the ASCII character generator and the two bits not used can be used to implement inverse video or color switching on a character by character basis. A 512 word display memory is required for this class of display.

The ALPHA SEMIGraphics -4 mode translates bits zero through three into a 4 x 6 dot element in the standard 8 x 12 dot box. Three data bits may be used to select one of eight colors for the entire character box. The extra bit is available to implement mode switching on the fly. A 512 word display memory is required. A density of 64 x 32 elements is available in the display area. The element area is four dot-clocks wide by six lines high.

The ALPHA SEMIGraphic -6 mode maps six 4 x 4 dot elements into the standard 8 x 12 dot alphanumeric box, a screen density of 64 x 48 elements is available. Six bits are used to generate this map and two data bits may be used to select one of four colors in the display box. The element area is four dot-clocks wide by four lines high.

FULL GRAPHIC MODE – There are eight full graphic modes available from the VDG. These modes require 1K to 6K bytes of memory. The eight full-graphic modes include an outside color border in one of two colors depending upon the color set select pin (CSS). The CSS pin selects one of two sets of four colors in the four color graphic modes.

The 64 x 64 Color Graphics Mode – The 64 x 64 color graphics mode generates a display matrix of 64 elements wide by 64 elements high. Each element may be one of four colors. A 1K x 8 display memory is required. Each pictel equals four dot-clocks by three scan lines.

The 128 x 64 Graphics Mode – The 128 x 64 graphics mode generates a matrix 128 elements wide by 64 elements high. Each element may be either ON or OFF. However, the entire display may be one of two colors, selected by using the color set select pin. A 1K x 8 display memory is required. Each pictel equals two dot-clocks by three scan lines.

The 128 x 64 Color Graphics Mode – The 128 x 64 color graphics mode generates a display matrix 128 elements wide by 64 elements high. Each element may be one of four colors. A 2K x 8 display memory is required. Each pictel equals two dot-clocks by three scan lines.

The 128 x 96 Graphics Mode – The 128 x 96 graphics mode generates a display matrix 128 elements wide by 96 elements high. Each element may be either ON or OFF. However, the entire display may be one of two colors selected by using the color select pin. A 2K x 8 display memory is required. Each pictel equals two dot-clocks by two scan lines.

The 128 x 96 Color Graphics Mode – The 128 x 96 color graphics mode generates a display 128 elements wide by 96 elements high. Each element may be one of four colors. A 3K x 8 display memory is required. Each pictel equals two dot-clocks by two scan lines.

The 128 x 192 Graphics Mode – The 128 x 192 graphics mode generates a display matrix 128 elements wide by 192 elements high. Each element may be either ON or OFF, but the ON elements may be one of two colors selected with color set select pin. A 3K x 8 display memory is required. Each pictel equals two dot-clocks by one scan line.

The 128 x 192 Color Graphics Mode – The 128 x 192 color graphics mode generates a display 128 elements wide by 192 elements high. Each element may be one of four colors. A 6K x 8 display memory is required. A detailed description of the VDG modes is given in Table 3. Each pictel equals two dot-clocks by one scan line.

The 256 x 192 Graphics Mode – The 256 x 192 graphics mode generates a display 256 elements wide by 192 elements high. Each element may be either ON or OFF, but the ON element may be one of two colors selected with the color set select pin. A 6K x 8 display memory is required. Each pictel equals one dot-clock by one scan line.

TABLE 1 – TABLE OF MODE CONTROL LINES (INPUTS)

$\overline{A}/G$	$\overline{A}/S$	$\overline{INT}/EXT$	INV	GM2	GM1	GM0	ALPHA/GRAPHIC MODE SELECT
0	0	0	0	X	X	X	Internal Alphanumerics
0	0	0	1	X	X	X	Internal Alphanumerics Inverted
0	0	1	0	X	X	X	External Alphanumerics
0	0	1	1	X	X	X	External Alphanumerics Inverted
0	1	0	X	X	X	X	Semigraphics - 4
0	1	1	X	X	X	X	Semigraphics - 6
1	X	X	X	0	0	0	64 x 64 Color Graphics
1	X	X	X	0	0	1	128 x 64 Graphics
1	X	X	X	0	1	0	128 x 64 Color Graphics
1	X	X	X	0	1	1	128 x 96 Graphics
1	X	X	X	1	0	0	128 x 96 Color Graphics
1	X	X	X	1	0	1	128 x 192 Graphics
1	X	X	X	1	1	0	128 x 192 Color Graphics
1	X	X	X	1	1	1	256 x 192 Graphics

TABLE 2 – SUMMARY OF MAJOR MODES

MAJOR MODE ONE
TABLE OF ALPHA MINOR MODES

Title	Memory	Colors	Display Elements
Alphanumeric (Internal)	512 x 8	2	
Alphanumeric (External)	512 x 8	2	
Alpha Semig-4	512 x 8	8	Box Element
Alpha Semig-6	512 x 8	4	Box Element

MAJOR MODE TWO
TABLE OF MINOR GRAPHICS MODES

Title	Memory	Colors	Comments
64 x 64 Color Graphic	1K x 8	4	Matrix 64 x 64 Elements
128 x 64 Graphics*	1K x 8	2	Matrix 128 elements wide by 64 elements high
128 x 64 Color Graphic	2K x 8	4	
128 x 96 Graphics*	1.5K x 8	2	Matrix 128 elements wide by 96 elements high
128 x 96 Color Graphic	3K x 8	4	
128 x 192 Graphics*	3K x 8	2	Matrix 128 elements wide by 192 elements high
128 x 192 Color Graphic	6K x 8	4	
256 x 192 Graphics*	6K x 8	2	Matrix 256 elements wide by 192 elements high

*Graphics mode turns on or off each element. The color may be one of two.

TABLE 3 – DETAILED DESCRIPTION OF VDG MODES

VDG PINS									COLOR			TV SCREEN		VDG DATA BUS	COMMENTS
$\overline{MS}$	A/$\overline{G}$	A/$\overline{S}$	$\overline{INT}$/EXT	GM2	GM1	GM0	CSS	INV	Character Color	Background	Border	Display Mode	Detail		
1	0	0	0	X	X	X	0	0	Green	Black	Black	32 Characters in columns 16 Characters in rows	8 dots, 12 dots, 7, 5	extra, ASCII code	The ALPHANUMERIC INTERNAL mode uses an internal character generator which contains the following five dot by seven dot characters: @ABCDEFGHIJ KLMNOPQRSTUVWXYZ[\] ↑ ← SP !"#$%&'()*+,-./0123456789:;< =>?. The six bit ASCII code leaves two bits free and these may be externally connected to the mode pins (A/G, A/S, INT/EXT, GM2, GM1, GM0, CSS or INV).
								1	Black	Green					
							1	0	Orange	Black					
								1	Black	Orange					
1	0	0	1	X	X	X	0	0	Green	Black	Black	32 Characters in columns 16 Characters in rows		one row of custom characters	The ALPHANUMERIC EXTERNAL mode uses an external character generator as well as a row counter. Thus, custom character fonts are graphic symbol sets with up to 256 different eight dot x 12 dot "characters" may be displayed.
								1	Black	Green					
							1	0	Orange	Black					
								1	Black	Orange					
1	0	1	0	X	X	X	X	X	Lx C2 C1 C0 Color 0 X X X Black 1 0 0 0 Green 1 0 0 1 Yellow 1 0 1 0 Blue 1 0 1 1 Red 1 1 0 0 Buff 1 1 0 1 Cyan 1 1 1 0 Magenta 1 1 1 1 Orange		Black	64 Display elements in columns 32 Display elements in rows	L_3 L_2 / L_1 L_0 one element	C_2 C_1 C_0 L_3 L_2 L_1 L_0	The SEMIGRAPHICS FOUR mode uses an internal "course graphics" generator in which a rectangle (eight dots by twelve dots) is divided into four equal parts. The luminance of each part is determined by a corresponding bit on the VDG data bus. The color of illuminated parts is determined by three bits.
1	0	1	1	X	X	X	0	X	Lx C1 C0 Color 0 X X Black 1 0 0 Green 1 0 1 Yellow 1 1 0 Blue 1 1 1 Red 0 X X Black 1 0 0 Buff 1 0 1 Cyan 1 1 0 Magenta 1 1 1 Orange		Black	64 Display elements in columns 48 Display elements in rows	L_5 L_4 / L_3 L_2 / L_1 L_0	C_1 C_0 L_5 L_4 L_3 L_2 L_1 L_0	The SEMIGRAPHIC SIX mode is similar to the SEMIGRAPHIC FOUR mode with the following differences: The eight dot by twelve dot rectangle is divided into six equal parts. Color is determined by the two remaining bits.
1	1	X	X	0	0	0	0	X	C1 C0 Color 0 0 Green 0 1 Yellow 1 0 Blue 1 1 Red 0 0 Buff 0 1 Cyan 1 0 Magenta 1 1 Orange		Green Buff	64 Display elements in columns 64 Display elements in rows	E_3 E_2 E_1 E_0	C_1 C_0 C_1 C_0 C_1 C_0 C_1 C_0	The GRAPHICS ONE C mode uses a maximum of 1024 bytes of display RAM in which one pair of bits specifies one picture element.
1	1	X	X	0	0	1	0	X	Lx 0 1	Color Black Green	Green	128 Display elements in columns 64 Display elements in rows	L_7 L_6 L_5 L_4 L_3 L_2 L_1 L_0	L_7 L_6 L_5 L_4 L_3 L_2 L_1 L_0	The GRAPHICS ONE R mode uses a maximum of 1024 bytes of display RAM in which one bit specifies one picture element.
							1	X	0 1	Black Buff	Buff				
1	1	X	X	0	1	0	0	X	Same color as Graphics one C		Green Buff	128 Display elements in columns 64 Display elements in rows	E_3 E_2 E_1 E_0	C_1 C_0 C_1 C_0 C_1 C_0 C_1 C_0	The GRAPHICS TWO C mode uses a maximum of 2048 bytes of display RAM in which one pair of bits specifies one picture element.
							1	X							
1	1	X	X	0	1	1	0	X	Same color as Graphics one R		Green Buff	128 Display elements in columns 96 Display elements in rows	L_7 L_6 L_5 L_4 L_3 L_2 L_1 L_0	L_7 L_6 L_5 L_4 L_3 L_2 L_1 L_0	The GRAPHICS TWO R mode uses a maximum of 1536 bytes of display RAM in which one bit specifies one picture element.
							1	X							
1	1	X	X	1	0	0	0	X	Same color as Graphics one C		Green Buff	128 Display elements in columns 96 Display elements in rows	E_3 E_2 E_1 E_0	C_1 C_0 C_1 C_0 C_1 C_0 C_1 C_0	The GRAPHICS THREE C mode uses a maximum of 3072 bytes of display RAM in which one pair of bytes specifies one picture element.
							1	X							
1	1	X	X	1	0	1	0	X	Same color as Graphics one R		Green Buff	128 Display elements in columns 192 Display elements in rows	L_7 L_6 L_5 L_4 L_3 L_2 L_1 L_0	L_7 L_6 L_5 L_4 L_3 L_2 L_1 L_0	The GRAPHICS THREE R mode uses a maximum of 3072 bytes of display RAM in which one bit specifies one picture element.
							1	X							
1	1	X	X	1	1	0	0	X	Same color as Graphics one C		Green Buff	128 Display elements in columns 192 Display elements in rows	E_3 E_2 E_1 E_0	C_1 C_0 C_1 C_0 C_1 C_0 C_1 C_0	The GRAPHICS SIX C mode uses a maximum of 6144 bytes of display RAM in which one pair of bits specifies one picture element.
							1	X							
1	1	X	X	1	1	1	0	X	Same color as Graphics one R		Green Buff	256 Display elements in columns 192 Display elements in rows	L_7 L_6 L_5 L_4 L_3 L_2 L_1 L_0	L_7 L_6 L_5 L_4 L_3 L_2 L_1 L_0	The GRAPHICS SIX R mode uses a maximum of 6144 bytes of display RAM in which one bit specifies one picture element.
							1	X							

APPENDIX A

Custom MC6847 Ordering Information

A.0 Custom MC6847 Ordering Information

The custom MC6847 specifications may be transmitted to Motorola in any of the following media:

1 PROM(s)
2 Assembler formatted object tape
3 Punched card deck
4 Paper tape of card deck format

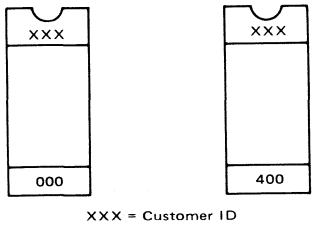
To initiate a ROM pattern for the MC6847 it is necessary to first contact your local field service office, local sales person or your local Motorola representative.

A.1 PROMs

MCM2708 or MCM2716 type PROMs, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. The PROMs must be clearly marked to indicate which PROM corresponds to which address space (000-3FF HEX). See Figure A-1 for recommended marking procedure.

After the PROM(s) are marked they should be placed in conductive IC carriers and securely packed. Do not use styrofoam.

FIGURE A-1



XXX = Customer ID

A.2 Assembler Formatted Object Tape

Cassette tapes produced on a Silent 700 terminal and EXORciser are acceptable.

A.3 Punched Card Deck

The custom MC6847 may be specified for manufacture in the form of standard 80-column punched cards.

The card deck for specifying the Custom MC6847 has the following format:

OPTION CARD
COMMENT CARDS
X CARDS
C CARDS

Option Card — The first card in the deck must be the OPTION CARD. The format is as follows:

Column 1-20: Customer name. Any 20 characters may be used.

Column 25-29: This is a 5-digit number assigned by Motorola. Leave this field blank. It will be punched at Motorola unless otherwise notified.

Column 37-39: Address field base on output listing. The characters HEX or DEC specify the output listing address base as hexadecimal or decimal, respectively. If omitted, DEC is assumed.

Column 41-43: Data field base on output listing. The characters HEX or DEC specify the output listing ROM data base as hexadecimal or decimal, respectively. If omitted, DEC is assumed.

Comment Cards — Comment Cards must have an asterisk (*) in column 1. The remaining 79 columns may contain any letter, number, or character.

X Cards — Five X cards are possible. All X cards have an X in column 1 and one or three or more words each separated by one blank space.

The possibilities are:

1) X SEQUENCE
2) X BASE DEC DEC
3) X BASE DEC HEX
4) X BASE HEX DEC
5) X BASE HEX HEX

Card 1 specifies that there are sequence numbers on each data card that follows. The sequence numbers must be in columns 77-79 of the data cards (C Cards) and must be in decimal, right justified. The numbers must start with 1 (one) and must be in order. The X SEQUENCE Card may appear anywhere within the deck after the Option Card. If it appears within the data card section, data cards encountered before the X SEQUENCE Card will not be checked for sequence numbers. All following cards will be checked. If no X SEQUENCE Card is used, no sequence numbers will be checked.

It is initially assumed that the address and byte count as well as the data specified on the C Cards will be in decimal. An X BASE Card can be used to override this

specification. The second word on the card (that following BASE) specifies the base (either DEC or HEX) of the address and byte count on all following C cards. The last word specifies the base of the data fields on the C Cards. An X BASE Card may appear anywhere within the deck following the OPTION Card. It may be overridden by another X BASE Card. All data cards (C Cards) following an X BASE Card will be interpreted as per that X BASE Card unless another X BASE Card is encountered. If no X BASE Cards are used, it is assumed that all fields on the C Cards are in decimal.

NOTE: Once an X SEQUENCE Card is encountered, all successive cards will be checked for the proper sequence number and unlike X BASE Cards this option cannot thereafter be altered by another X SEQUENCE Card.

C Cards — These cards contain the actual ROM data. All fields are right-justified.

Column 1: C (the letter C)
Column 2-9: ADD
Column 10-12: BYTE
Column 14-16: DATA 1
Column 17-19: DATA 2
• •
• •
• •
Column 76-78: DATA 21
Column 77-79: DATA 22 or SEQUENCE NUMBER

ADD is the address of the first byte of data (DATA 1) contained on that card. Byte is the number of bytes of data to be read from that card. BYTE must be greater than zero and less than 23 (1-22) if no sequence numbers are used, and less than 22 (1-21) if sequence numbers are used. If, for example, there are ten data fields punched on the card, buy BYTE = 2, only the first two will be read. Also, if there are two punched data fields, for example, and BYTE = 6, six ROM locations will be filled from that card. The four unspecified fields will be decoded as zero. ADD and BYTE are always in the same base (HEX or DECIMAL). DATA 1 through DATA N is the data to be placed in the ROM at addresses ADD through ADD + (N-1), respectively.

Any ROM address not filled as a result of reading data from a C Card will be filled with zero. If a particular location has already been specified by a C Card, but a successive C Card also has the data which is to be placed in that location, the second C Card will override the first.

A.4 Paper Tape of Card Deck Format

Punched Paper tape (ASCII) in the same format as cards can also be accepted. However, your order will be processed faster if the data is in card format. After the tape leader there should be a CR LF. Data records should be a full 80 columns, each terminated by a CR LF. Following the last Data record, there should be one more record with the first three characters being EOF, followed by 77 blanks and a CR LF.

CR = Carriage Return
LF = Line Feed

FIGURE A-2

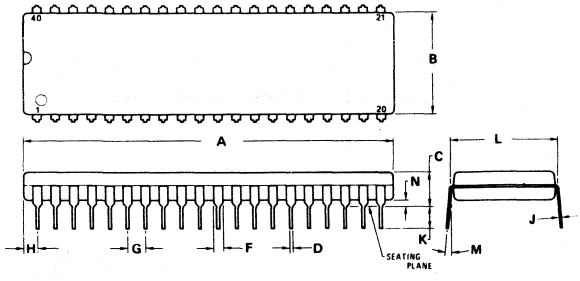
Customer Name _____
Address _____
City _____ State _____ Zip _____
Phone (_____) _____ Extension _____
Contact Ms/Mr _____
Customer Part Number _____

Pattern Media
2708 PROM
2716 PROM
Paper Object Tape
Silent 700 Cassette
Card Deck
Tape of Card Deck
(Note 2) _____

Notes: (2) Other media require prior factory approval

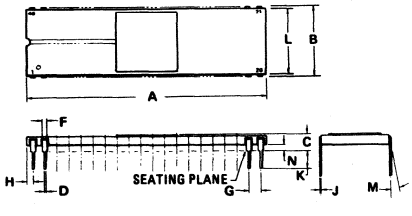
Signature _____
Title _____

P SUFFIX
PLASTIC PACKAGE
CASE 711-02



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	51.82	52.32	2.040	2.060
B	13.72	14.22	0.540	0.560
C	4.57	5.08	0.180	0.200
D	0.36	0.51	0.014	0.020
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	3.05	3.56	0.120	0.140
L	15.24 BSC		0.600 BSC	
M	0°	10°	0°	10°
N	0.51	1.02	0.020	0.040

CASE 711-02



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	50.29	51.31	1.980	2.020
B	14.86	15.62	0.585	0.615
C	2.54	4.19	0.100	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.33	0.008	0.013
K	2.54	4.19	0.100	0.165
L	14.60	15.37	0.575	0.605
M	0°	10°	0°	10°
N	0.51	1.52	0.020	0.060

CASE 715-02

NOTE:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA (AT SEATING PLANE), AT MAX. MAT'L CONDITION.

L SUFFIX
CERAMIC PACKAGE
CASE 715-02

MOTOROLA *Semiconductor Products Inc.*
BOX 20912 • PHOENIX, ARIZONA 85036 • A SUBSIDIARY OF MOTOROLA INC.

11724-1 PRINTED IN USA 12-78 IMPERIAL LITHO B75544 15M

ADI-492-R1

MOTOROLA

MC6850
1.0 MHz
MC68A50
1.5 MHz
MC68B50
2.0 MHz

ASYNCHRONOUS COMMUNICATIONS INTERFACE ADAPTER (ACIA)

The MC6850 Asynchronous Communications Interface Adapter provides the data formatting and control to interface serial asynchronous data communications information to bus organized systems such as the MC6800 Microprocessing Unit.

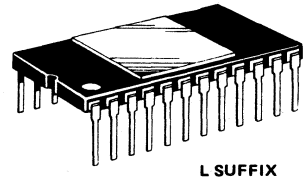
The bus interface of the MC6850 includes select, enable, read/write, interrupt and bus interface logic to allow data transfer over an 8-bit bi-directional data bus. The parallel data of the bus system is serially transmitted and received by the asynchronous data interface, with proper formatting and error checking. The functional configuration of the ACIA is programmed via the data bus during system initialization. A programmable Control Register provides variable word lengths, clock division ratios, transmit control, receive control, and interrupt control. For peripheral or modem operation three control lines are provided. These lines allow the ACIA to interface directly with the MC6860L 0-600 bps digital modem.

- Eight and Nine-Bit Transmission
- Optional Even and Odd Parity
- Parity, Overrun and Framing Error Checking
- Programmable Control Register
- Optional ÷1, ÷16, and ÷64 Clock Modes
- Up to 500 kbps Transmission
- False Start Bit Deletion
- Peripheral/Modem Control Functions
- Double Buffered
- One or Two Stop Bit Operation

MOS

(N-CHANNEL, SILICON-GATE)

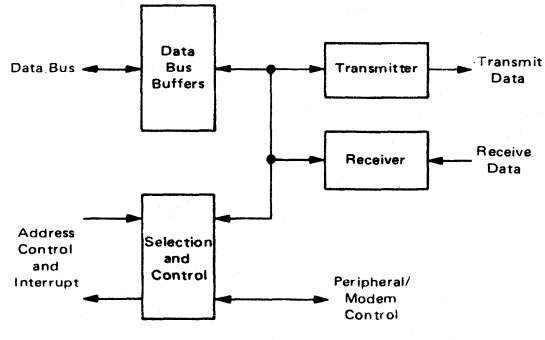
ASYNCHRONOUS COMMUNICATIONS INTERFACE ADAPTER



L SUFFIX
CERAMIC PACKAGE
CASE 716

NOT SHOWN: **P SUFFIX**
PLASTIC PACKAGE
CASE 709

MC6850 ASYNCHRONOUS COMMUNICATIONS INTERFACE ADAPTER BLOCK DIAGRAM



ORDERING INFORMATION

Speed	Device	Temperature Range
1.0 MHz	MC6850P,L	0 to +70°C
	MC6850CP,CL	-40 to +85°C
MIL-STD-883B M1L-STD-883C	MC6850BJCS MC6850CJCS	-55 to +125°C
1.5 MHz	MC68A50P,L	0 to +70°C
	MC68A50CP,CL	-40 to +85°C
2.0 MHz	MC68B50P,L	0 to +70°C

DS 9493 R1

MAXIMUM RATINGS

Rating		Symbol	Value	Unit
Supply Voltage		V_{CC}	-0.3 to +7.0	Vdc
Input Voltage		V_{in}	-0.3 to +7.0	Vdc
Operating Temperature Range		T_A	0 to +70	°C
Storage Temperature Range		T_{stg}	-55 to +150	°C
Thermal Resistance	Plastic	θ_{JA}	120	°C/W
	Ceramic		60	

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit.

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ±5%, V_{SS} = 0, T_A = 0 to 70°C unless otherwise noted.)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage		V_{IH}	V_{SS} + 2.0	–	V_{CC}	Vdc
Input Low Voltage		V_{IL}	V_{SS} -0.3	–	V_{SS} + 0.8	Vdc
Input Leakage Current (V_{in} = 0 to 5.25 Vdc)	R/$\overline{W}$, CS0, CS1, $\overline{CS2}$, Enable	I_{in}	–	1.0	2.5	μAdc
	RS, Rx D, Rx C, $\overline{CTS}$, $\overline{DCD}$					
Three-State (Off State) Input Current (V_{in} = 0.4 to 2.4 Vdc)	D0-D7	I_{TSI}	–	2.0	10	μAdc
Output High Voltage	D0-D7	V_{OH}				Vdc
(I_{Load} = -205 μAdc, Enable Pulse Width <25 μs)			V_{SS} + 2.4	–	–	
(I_{Load} = -100 μAdc, Enable Pulse Width <25 μs)	Tx Data, $\overline{RTS}$		V_{SS} + 2.4	–	–	
Output Low Voltage (I_{Load} = 1.6 mAdc, Enable Pulse Width <25 μs)		V_{OL}	–	–	V_{SS} + 0.4	Vdc
Output Leakage Current (Off State) (V_{OH} = 2.4 Vdc)	$\overline{IRQ}$	I_{LOH}	–	1.0	10	μAdc
Power Dissipation		P_D	–	300	525	mW
Input Capacitance		C_{in}				pF
(V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	D0-D7		–	10	12.5	
	E, Tx Clk, Rx Clk, R/W, RS, Rx Data, CS0, CS1, $\overline{CS2}$, $\overline{CTS}$, $\overline{DCD}$		–	7.0	7.5	
Output Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	$\overline{RTS}$, Tx Data	C_{out}	–	–	10	pF
	$\overline{IRQ}$		–	–	5.0	
Minimum Clock Pulse Width, Low (Figure 1)	÷16, ÷64 Modes	PW_{CL}	600	–	–	ns
Minimum Clock Pulse Width, High (Figure 2)	÷16, ÷64 Modes	PW_{CH}	600	–	–	ns
Clock Frequency	÷1 Mode	f_C	–	–	500	kHz
	÷16, ÷64 Modes		–	–	800	
Clock-to-Data Delay for Transmitter (Figure 3)		t_{TDD}	–	–	1.0	μs
Receive Data Setup Time (Figure 4)	÷1 Mode	t_{RDSU}	500	–	–	ns
Receive Data Hold Time (Figure 5)	÷1 Mode	t_{RDH}	500	–	–	ns
$\overline{Interrupt\ Request}$ Release Time (Figure 6)		t_{IR}	–	–	1.2	μs
$\overline{Request\text{-}to\text{-}Send}$ Delay Time (Figure 6)		t_{RTS}	–	–	1.0	μs
Input Transition Times (Except Enable)		t_r,t_f	–	–	1.0*	μs

*1.0 μs or 10% of the pulse width, whichever is smaller.

BUS TIMING CHARACTERISTICS

Characteristic	Symbol	MC6850 Min	MC6850 Max	MC68A50 Min	MC68A50 Max	MC68B50 Min	MC68B50 Max	Unit
READ (Figures 7 and 9)								
Enable Cycle Time	t_{cycE}	1.0	–	0.666	–	0.500	–	μs
Enable Pulse Width, High	PW_{EH}	0.45	25	0.28	25	0.22	25	μs
Enable Pulse Width, Low	PW_{EL}	0.43	–	0.28	–	0.21	–	μs
Setup Time, Address and R/W valid to Enable positive transistion	t_{AS}	160	–	140	–	70	–	ns
Data Delay Time	t_{DDR}	–	320	–	220	–	180	ns
Data Hold Time	t_H	10	–	10	–	10	–	ns
Address Hold Time	t_{AH}	10	–	10	–	10	–	ns
Rise and Fall Time for Enable input	t_{Er}, t_{Ef}	–	25	–	25	–	25	ns
WRITE (Figures 8 and 9)								
Enable Cycle Time	t_{cycE}	1.0	–	0.666	–	500	–	μs
Enable Pulse Width, High	PW_{EH}	0.45	25	0.28	25	0.22	25	μs
Enable Pulse Width, Low	PW_{EL}	0.43	–	0.28	–	0.21	–	μs
Setup Time, Address and R/W valid to Enable positive transition	t_{AS}	160	–	140	–	70	–	ns
Data Setup Time	t_{DSW}	195	–	80	–	60	–	ns
Data Hold Time	t_H	10	–	10	–	10	–	ns
Address Hold Time	t_{AH}	10	–	10	–	10	–	ns
Rise and Fall Time for Enable input	t_{Er}, t_{Ef}	–	25	–	25	–	25	ns

FIGURE 1 – CLOCK PULSE WIDTH, LOW-STATE

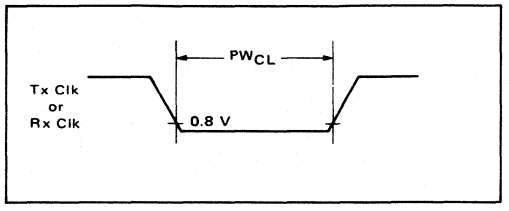


FIGURE 2 – CLOCK PULSE WIDTH, HIGH-STATE

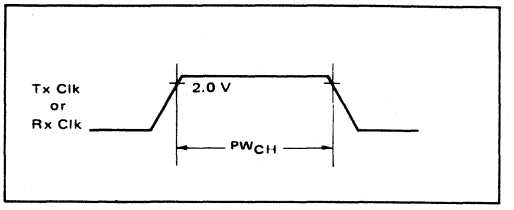


FIGURE 3 – TRANSMIT DATA OUTPUT DELAY

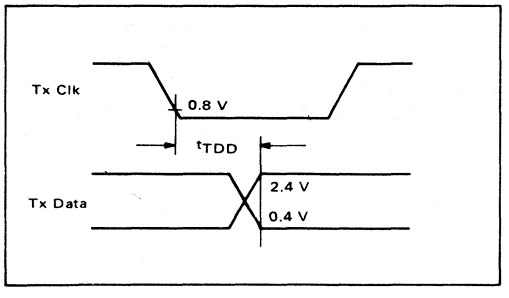


FIGURE 4 – RECEIVE DATA SETUP TIME
(÷1 Mode)

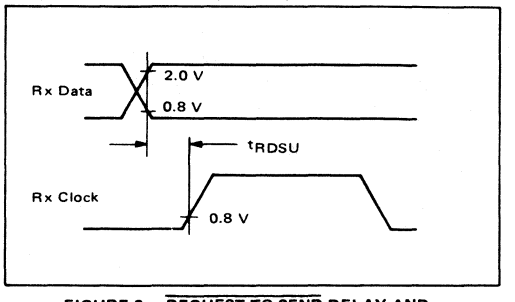


FIGURE 5 – RECEIVE DATA HOLD TIME
(÷1 Mode)

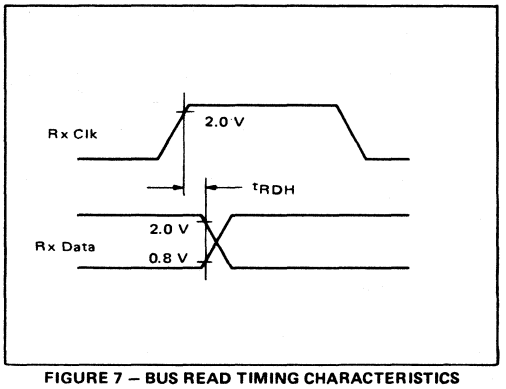


FIGURE 6 – $\overline{\text{REQUEST-TO-SEND}}$ DELAY AND $\overline{\text{INTERRUPT-REQUEST}}$ RELEASE TIMES

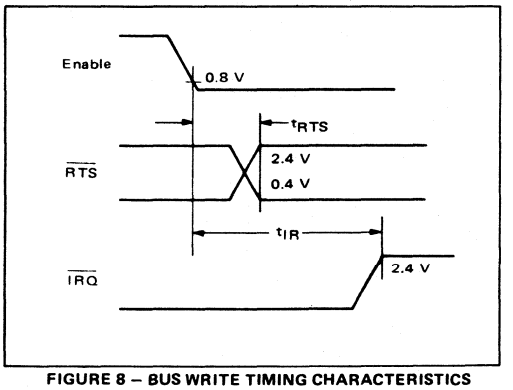


FIGURE 7 – BUS READ TIMING CHARACTERISTICS
(Read information from ACIA)

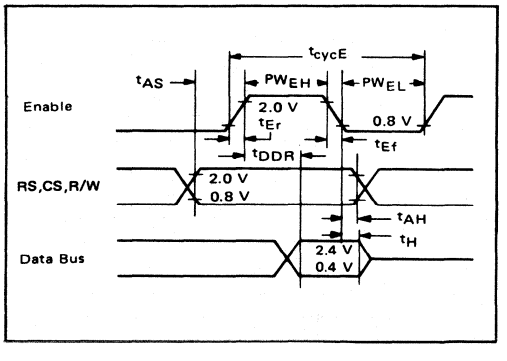


FIGURE 8 – BUS WRITE TIMING CHARACTERISTICS
(Write information into ACIA)

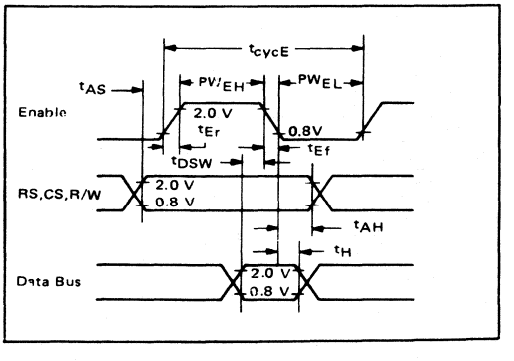
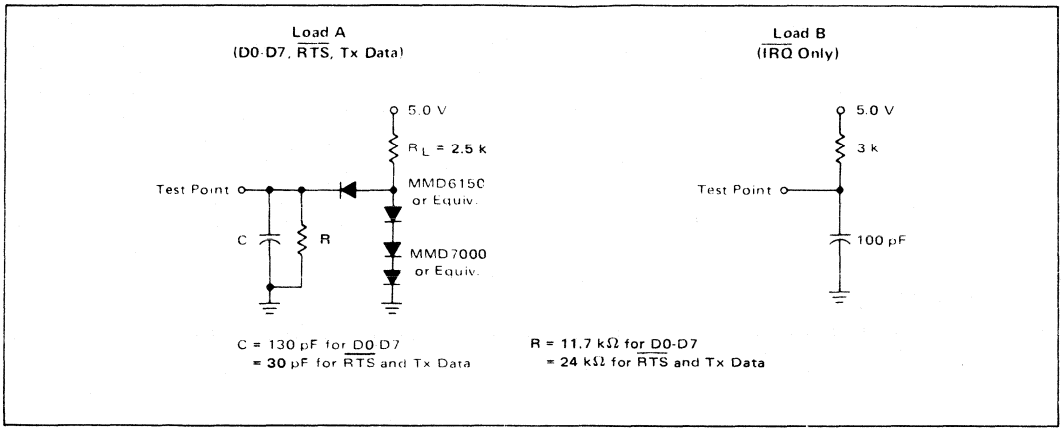
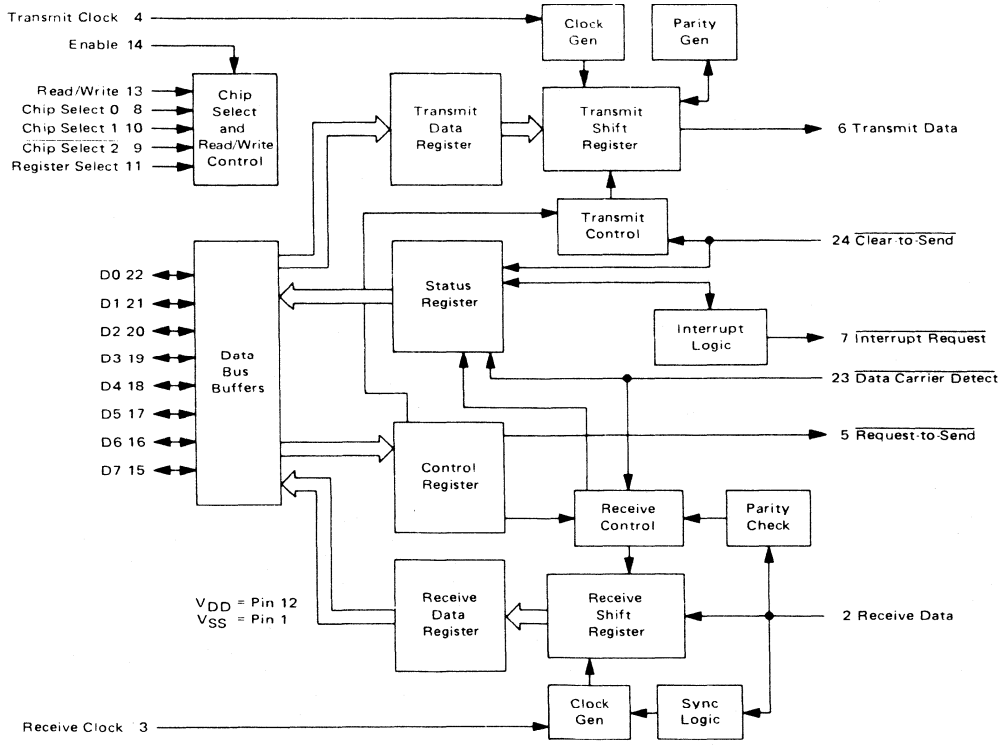


FIGURE 9 – BUS TIMING TEST LOADS



EXPANDED BLOCK DIAGRAM



DEVICE OPERATION

At the bus interface, the ACIA appears as two addressable memory locations. Internally, there are four registers: two read-only and two write-only registers. The read-only registers are Status and Receive Data; the write-only registers are Control and Transmit Data. The serial interface consists of serial input and output lines with independent clocks, and three peripheral/modem control lines.

POWER ON/MASTER RESET

The master reset (CR0, CR1) should be set during system initialization to insure the reset condition and prepare for programming the ACIA functional configuration when the communications channel is required. Control bits CR5 and CR6 should also be programmed to define the state of $\overline{RTS}$ whenever master reset is utilized. The ACIA also contains internal power-on reset logic to detect the power line turn-on transition and hold the chip in a reset state to prevent erroneous output transitions prior to initialization. This circuitry depends on clean power turn-on transitions. The power-on reset is released by means of the bus-programmed master reset which must be applied prior to operating the ACIA. After master resetting the ACIA, the programmable Control Register can be set for a number of options such as variable clock divider ratios, variable word length, one or two stop bits, parity (even, odd, or none), etc.

TRANSMIT

A typical transmitting sequence consists of reading the ACIA Status Register either as a result of an interrupt or in the ACIA's turn in a polling sequence. A character may be written into the Transmit Data Register if the status read operation has indicated that the Transmit Data Register is empty. This character is transferred to a Shift Register where it is serialized and transmitted from the Transmit Data output preceded by a start bit and followed by one or two stop bits. Internal parity (odd or even) can be optionally added to the character and will occur between the last data bit and the first stop bit. After the first character is written in the Data Register, the Status Register can be read again to check for a Transmit Data Register Empty condition and current peripheral status. If the register is empty, another character can be loaded for transmission even though the first character is in the process of being transmitted (because of double buffering). The second character will be automatically transferred into the Shift Register when the first character transmission is completed. This sequence continues until all the characters have been transmitted.

RECEIVE

Data is received from a peripheral by means of the Receive Data input. A divide-by-one clock ratio is provided for an externally synchronized clock (to its data) while the divide-by-16 and 64 ratios are provided for internal synchronization. Bit synchronization in the divide-by-16 and 64 modes is initiated by the detection of the leading mark-to-space transition of the start bit. False start bit deletion capability insures that a full half bit of a start bit has been received before the internal clock is synchronized to the bit time. As a character is being received, parity (odd or even) will be checked and the error indication will be available in the Status Register along with framing error, overrun error, and Receive Data Register full. In a typical receiving sequence, the Status Register is read to determine if a character has been received from a peripheral. If the Receiver Data Register is full, the character is placed on the 8-bit ACIA bus when a Read Data command is received from the MPU. When parity has been selected for an 8-bit word (7 bits plus parity), the receiver strips the parity bit (D7 = 0) so that data alone is transferred to the MPU. This feature reduces MPU programming. The Status Register can continue to be read again to determine when another character is available in the Receive Data Register. The receiver is also double buffered so that a character can be read from the data register as another character is being received in the shift register. The above sequence continues until all characters have been received.

INPUT/OUTPUT FUNCTIONS

ACIA INTERFACE SIGNALS FOR MPU

The ACIA interfaces to the MC6800 MPU with an 8-bit bi-directional data bus, three chip select lines, a register select line, an interrupt request line, read/write line, and enable line. These signals, in conjunction with the MC6800 VMA output, permit the MPU to have complete control over the ACIA.

ACIA Bi-Directional Data (D0-D7) – The bi-directional data lines (D0-D7) allow for data transfer between the ACIA and the MPU. The data bus output drivers are three-state devices that remain in the high-impedance (off) state except when the MPU performs an ACIA read operation.

ACIA Enable (E) – The Enable signal, E, is a high impedance TTL compatible input that enables the bus input/output data buffers and clocks data to and from the ACIA. This signal will normally be a derivative of the MC6800 $\phi 2$ Clock.

Read/Write (R/W) – The Read/Write line is a high impedance input that is TTL compatible and is used to control the direction of data flow through the ACIA's input/output data bus interface. When Read/Write is high (MPU Read cycle), ACIA output drivers are turned on and a selected register is read. When it is low, the ACIA output drivers are turned off and the MPU writes into a selected register. Therefore, the Read/Write signal is used to select read-only or write-only registers within the ACIA.

Chip Select (CS0, CS1, $\overline{CS2}$) – These three high impedance TTL compatible input lines are used to address the ACIA. The ACIA is selected when CS0 and CS1 are high and $\overline{CS2}$ is low. Transfers of data to and from the ACIA are then performed under the control of the Enable signal, Read/Write, and Register Select.

Register Select (RS) – The Register Select line is a high impedance input that is TTL compatible. A high level is used to select the Transmit/Receive Data Registers and a low level the Control/Status Registers. The Read/Write signal line is used in conjunction with Register Select to select the read-only or write-only register in each register pair.

$\overline{\text{Interrupt Request}}$ ($\overline{IRQ}$) – $\overline{\text{Interrupt Request}}$ is a TTL compatible, open-drain (no internal pullup), active low

output that is used to interrupt the MPU. The $\overline{IRQ}$ output remains low as long as the cause of the interrupt is present and the appropriate interrupt enable within the ACIA is set. The IRQ status bit, when high, indicates the $\overline{IRQ}$ output is in the active state.

Interrupts result from conditions in both the transmitter and receiver sections of the ACIA. The transmitter section causes an interrupt when the Transmitter Interrupt Enabled condition is selected (CR5 · $\overline{CR6}$), and the Transmit Data Register Empty (TDRE) status bit is high. The TDRE status bit indicates the current status of the Transmitter Data Register except when inhibited by Clear-to-Send ($\overline{CTS}$) being high or the ACIA being maintained in the Reset condition. The interrupt is cleared by writing data into the Transmit Data Register. The interrupt is masked by disabling the Transmitter Interrupt via CR5 or CR6 or by the loss of $\overline{CTS}$ which inhibits the TDRE status bit. The Receiver section causes an interrupt when the Receiver Interrupt Enable is set and the Receive Data Register Full (RDRF) status bit is high, an Overrun has occurred, or Data Carrier Detect ($\overline{DCD}$) has gone high. An interrupt resulting from the RDRF status bit can be cleared by reading data or resetting the ACIA. Interrupts caused by Overrun or loss of $\overline{DCD}$ are cleared by reading the status register after the error condition has occurred and then reading the Receive Data Register or resetting the ACIA. The receiver interrupt is masked by resetting the Receiver Interrupt Enable.

CLOCK INPUTS

Separate high impedance TTL compatible inputs are provided for clocking of transmitted and received data. Clock frequencies of 1, 16 or 64 times the data rate may be selected.

Transmit Clock (Tx Clk) – The Transmit Clock input is used for the clocking of transmitted data. The transmitter initiates data on the negative transition of the clock.

Receive Clock (Rx Clk) – The Receive Clock input is used for synchronization of received data. (In the ÷ 1 mode, the clock and data must be synchronized externally.) The receiver samples the data on the positive transiton of the clock.

SERIAL INPUT/OUTPUT LINES

Receive Data (Rx Data) – The Receive Data line is a high impedance TTL compatible input through which data is received in a serial format. Synchronization with a clock for detection of data is accomplished internally when clock rates of 16 or 64 times the bit rate are used. Data rates are in the range of 0 to 500 kbps when external synchronization is utilized.

Transmit Data (Tx Data) – The Transmit Data output line transfers serial data to a modem or other peripheral. Data rates are in the range of 0 to 500 kbps when external synchronization is utilized.

PERIPHERAL/MODEM CONTROL

The ACIA includes several functions that permit limited control of a peripheral or modem. The functions included are Clear-to-Send, Request-to-Send and Data Carrier Detect.

Clear-to-Send ($\overline{CTS}$) – This high impedance TTL compatible input provides automatic control of the transmitting end of a communications link via the modem Clear-to-Send active low output by inhibiting the Transmit Data Register Empty (TDRE) status bit.

Request-to-Send ($\overline{RTS}$) – The Request-to-Send output enables the MPU to control a peripheral or modem via the data bus. The $\overline{RTS}$ output corresponds to the state of the Control Register bits CR5 and CR6. When CR6 = 0 or both CR5 and CR6 = 1, the $\overline{RTS}$ output is low (the active state). This output can also be used for Data Terminal Ready ($\overline{DTR}$).

Data Carrier Detect ($\overline{DCD}$) – This high impedance TTL compatible input provides automatic control, such as in the receiving end of a communications link by means of a modem Data Carrier Detect output. The $\overline{DCD}$ input inhibits and initializes the receiver section of the ACIA when high. A low to high transition of the Data Carrier Detect initiates an interrupt to the MPU to indicate the occurrence of a loss of carrier when the Receive Interrupt Enable bit is set.

ACIA REGISTERS

The expanded block diagram for the ACIA indicates the internal registers on the chip that are used for the status, control, receiving, and transmitting of data. The content of each of the registers is summarized in Table 1.

TRANSMIT DATA REGISTER (TDR)

Data is written in the Transmit Data Register during the negative transition of the enable (E) when the ACIA has been addressed and RS · $\overline{R/W}$ is selected. Writing data into the register causes the Transmit Data Register Empty bit in the Status Register to go low. Data can then be transmitted. If the transmitter is idling and no character is being transmitted, then the transfer will take place within one bit time of the trailing edge of the Write command. If a character is being transmitted, the new data character will commence as soon as the previous character is complete. The transfer of data causes the Transmit Data Register Empty (TDRE) bit to indicate empty.

RECEIVE DATA REGISTER (RDR)

Data is automatically transferred to the empty Receive Data Register (RDR) from the receiver deserializer (a shift register) upon receiving a complete character. This event causes the Receive Data Register Full bit (RDRF) in the status buffer to go high (full). Data may then be read through the bus by addressing the ACIA and selecting the Receive Data Register with RS and R/W high when the ACIA is enabled. The non-destructive read cycle causes the RDRF bit to be cleared to empty although

TABLE 1 – DEFINITION OF ACIA REGISTER CONTENTS

Data Bus Line Number	Buffer Address			
	RS • $\overline{R/W}$ Transmit Data Register (Write Only)	RS • R/W Receive Data Register (Read Only)	$\overline{RS}$ • $\overline{R/W}$ Control Register (Write Only)	$\overline{RS}$ • R/W Status Register (Read Only)
0	Data Bit 0*	Data Bit 0	Counter Divide Select 1 (CR0)	Receive Data Register Full (RDRF)
1	Data Bit 1	Data Bit 1	Counter Divide Select 2 (CR1)	Transmit Data Register Empty (TDRE)
2	Data Bit 2	Data Bit 2	Word Select 1 (CR2)	$\overline{\text{Data Carrier Detect}}$ ($\overline{DCD}$)
3	Data Bit 3	Data Bit 3	Word Select 2 (CR3)	$\overline{\text{Clear-to-Send}}$ ($\overline{CTS}$)
4	Data Bit 4	Data Bit 4	Word Select 3 (CR4)	Framing Error (FE)
5	Data Bit 5	Data Bit 5	Transmit Control 1 (CR5)	Receiver Overrun (OVRN)
6	Data Bit 6	Data Bit 6	Transmit Control 2 (CR6)	Parity Error (PE)
7	Data Bit 7***	Data Bit 7**	Receive Interrupt Enable (CR7)	Interrupt Request (IRQ)

* Leading bit = LSB = Bit 0
** Data bit will be zero in 7-bit plus parity modes.
*** Data bit is "don't care" in 7-bit plus parity modes.

the data is retained in the RDR. The status is maintained by RDRF as to whether or not the data is current. When the Receive Data Register is full, the automatic transfer of data from the Receiver Shift Register to the Data Register is inhibited and the RDR contents remain valid with its current status stored in the Status Register.

CONTROL REGISTER

The ACIA Control Register consists of eight bits of write-only buffer that are selected when RS and R/W are low. This register controls the function of the receiver, transmitter, interrupt enables, and the $\overline{\text{Request-to-Send}}$ peripheral/modem control output.

Counter Divide Select Bits (CR0 and CR1) – The Counter Divide Select Bits (CR0 and CR1) determine the divide ratios utilized in both the transmitter and receiver sections of the ACIA. Additionally, these bits are used to provide a master reset for the ACIA which clears the Status Register (except for external conditions on $\overline{CTS}$ and $\overline{DCD}$) and initializes both the receiver and transmitter. Master reset does not affect other Control Register bits. Note that after power-on or a power fail/restart, these bits must be set high to reset the ACIA. After reseting, the clock divide ratio may be selected. These counter select bits provide for the following clock divide ratios:

CR1	CR0	Function
0	0	÷ 1
0	1	÷ 16
1	0	÷ 64
1	1	Master Reset

Word Select Bits (CR2, CR3, and CR4) – The Word Select bits are used to select word length, parity, and the number of stop bits. The encoding format is as follows:

CR4	CR3	CR2	Function
0	0	0	7 Bits + Even Parity + 2 Stop Bits
0	0	1	7 Bits + Odd Parity + 2 Stop Bits
0	1	0	7 Bits + Even Parity + 1 Stop Bit
0	1	1	7 Bits + Odd Parity + 1 Stop Bit
1	0	0	8 Bits + 2 Stop Bits
1	0	1	8 Bits + 1 Stop Bit
1	1	0	8 Bits + Even Parity + 1 Stop Bit
1	1	1	8 Bits + Odd Parity + 1 Stop Bit

Word length, Parity Select, and Stop Bit changes are not buffered and therefore become effective immediately.

Transmitter Control Bits (CR5 and CR6) – Two Transmitter Control bits provide for the control of the interrupt from the Transmit Data Register Empty condition, the $\overline{\text{Request-to-Send}}$ ($\overline{RTS}$) output, and the transmission of a Break level (space). The following encoding format is used:

CR6	CR5	Function
0	0	$\overline{RTS}$ = low, Transmitting Interrupt Disabled.
0	1	$\overline{RTS}$ = low, Transmitting Interrupt Enabled.
1	0	$\overline{RTS}$ = high, Transmitting Interrupt Disabled.
1	1	$\overline{RTS}$ = low, Transmits a Break level on the Transmit Data Output. Transmitting Interrupt Disabled.

Receive Interrupt Enable Bit (CR7) – The following interrupts will be enabled by a high level in bit position 7 of the Control Register (CR7): Receive Data Register Full, Overrun, or a low to high transistion on the $\overline{\text{Data Carrier Detect}}$ ($\overline{DCD}$) signal line.

STATUS REGISTER

Information on the status of the ACIA is available to the MPU by reading the ACIA Status Register. This read-only register is selected when RS is low and R/W is high. Information stored in this register indicates the status of the Transmit Data Register, the Receive Data Register and error logic, and the peripheral/modem status inputs of the ACIA.

Receive Data Register Full (RDRF), Bit 0 – Receive Data Register Full indicates that received data has been transferred to the Receive Data Register. RDRF is cleared after an MPU read of the Receive Data Register or by a master reset. The cleared or empty state indicates that the contents of the Receive Data Register are not current. $\overline{\text{Data Carrier Detect}}$ being high also causes RDRF to indicate empty.

Transmit Data Register Empty (TDRE), Bit 1 – The Transmit Data Register Empty bit being set high indicates that the Transmit Data Register contents have been transferred and that new data may be entered. The low state indicates that the register is full and that transmission of a new character has not begun since the last write data command.

$\overline{\text{Data Carrier Detect}}$ ($\overline{\text{DCD}}$), Bit 2 – The $\overline{\text{Data Carrier Detect}}$ bit will be high when the $\overline{\text{DCD}}$ input from a modem has gone high to indicate that a carrier is not present. This bit going high causes an Interrupt Request to be generated when the Receive Interrupt Enable is set. It remains high after the $\overline{\text{DCD}}$ input is returned low until cleared by first reading the Status Register and then the Data Register or until a master reset occurs. If the $\overline{\text{DCD}}$ input remains high after read status and read data or master reset has occurred, the interrupt is cleared, the $\overline{\text{DCD}}$ status bit remains high and will follow the $\overline{\text{DCD}}$ input.

$\overline{\text{Clear-to-Send}}$ ($\overline{\text{CTS}}$), Bit 3 – The $\overline{\text{Clear-to-Send}}$ bit indicates the state of the $\overline{\text{Clear-to-Send}}$ input from a modem. A low $\overline{\text{CTS}}$ indicates that there is a $\overline{\text{Clear-to-Send}}$ from the modem. In the high state, the Transmit Data Register Empty bit is inhibited and the $\overline{\text{Clear-to-Send}}$ status bit will be high. Master reset does not affect the $\overline{\text{Clear-to-Send}}$ Status bit.

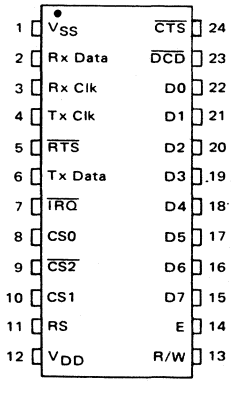
Framing Error (FE), Bit 4 – Framing error indicates that the received character is improperly framed by a start and a stop bit and is detected by the absence of the 1st stop bit. This error indicates a synchronization error, faulty transmission, or a break condition. The framing error flag is set or reset during the receive data transfer time. Therefore, this error indicator is present throughout the time that the associated character is available.

Receiver Overrun (OVRN), Bit 5 – Overrun is an error flag that indicates that one or more characters in the data stream were lost. That is, a character or a number of characters were received but not read from the Receive Data Register (RDR) prior to subsequent characters being received. The overrun condition begins at the midpoint of the last bit of the second character received in succession without a read of the RDR having occurred. The Overrun does not occur in the Status Register until the valid character prior to Overrun has been read. The RDRF bit remains set until the Overrun is reset. Character synchronization is maintained during the Overrun condition. The Overrun indication is reset after the reading of data from the Receive Data Register or by a Master Reset.

Parity Error (PE), Bit 6 – The parity error flag indicates that the number of highs (ones) in the character does not agree with the preselected odd or even parity. Odd parity is defined to be when the total number of ones is odd. The parity error indication will be present as long as the data character is in the RDR. If no parity is selected, then both the transmitter parity generator output and the receiver parity check results are inhibited.

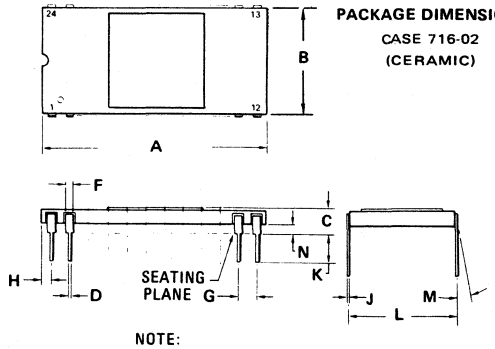
Interrupt Request (IRQ), Bit 7 – The IRQ bit indicates the state of the $\overline{\text{IRQ}}$ output. Any interrupt condition with its applicable enable will be indicated in this status bit. Anytime the $\overline{\text{IRQ}}$ output is low the IRQ bit will be high to indicate the interrupt or service request status. IRQ is cleared by a read operation to the Receive Data Register or a write operation to the Transmit Data Register.

PIN ASSIGNMENT



PACKAGE DIMENSIONS

CASE 716-02
(CERAMIC)



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	29.97	30.99	1.180	1.220
B	14.88	15.62	0.585	0.615
C	3.05	4.19	0.120	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.30	0.008	0.012
K	2.54	4.19	0.100	0.165
L	14.88	15.37	0.585	0.605
M	–	10^0	–	10^0
N	0.51	1.52	0.020	0.060

NOTE:
1. LEADS TRUE POSITIONED WITHIN 0.25mm (0.010) DIA (AT SEATING PLANE) AT MAXIMUM MATERIAL CONDITION.

MOTOROLA Semiconductor Products Inc.

36-A79/5.0 Printed in Switzerland

MOTOROLA

MC6852
(1.0 MHz)

MC68A52
(1.5 MHz)

MC68B52
(2.0 MHz)

SYNCHRONOUS SERIAL DATA ADAPTER (SSDA)

The MC6852 Synchronous Serial Data Adapter provides a bi-directional serial interface for synchronous data information interchange. It contains interface logic for simultaneously transmitting and receiving standard synchronous communications characters in bus organized systems such as the M6800 Microprocessor systems.

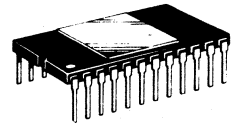
The bus interface of the MC6852 includes select, enable, read/write, interrupt, and bus interface logic to allow data transfer over an 8-bit bi-directional data bus. The parallel data of the bus system is serially transmitted and received by the synchronous data interface with synchronization, fill character insertion/deletion, and error checking. The functional configuration of the SSDA is programmed via the data bus during system initialization. Programmable control registers provide control for variable word lengths, transmit control, receive control, synchronization control, and interrupt control. Status, timing and control lines provide peripheral or modem control.

Typical applications include floppy disk controllers, cassette or cartridge tape controllers, data communications terminals, and numerical control systems.

- Programmable Interrupts from Transmitter, Receiver, and Error Detection Logic
- Character Synchronization on One or Two Sync Codes
- External Synchronization Available for Parallel-Serial Operation
- Programmable Sync Code Register
- Up to 600 kbps Transmission
- Peripheral/Modem Control Functions
- Three Bytes of FIFO Buffering on Both Transmit and Receive
- Seven, Eight, or Nine Bit Transmission
- Optional Even and Odd Parity
- Parity, Overrun, and Underflow Status

MOS
(N-CHANNEL, SILICON-GATE)

SYNCHRONOUS SERIAL DATA ADAPTER



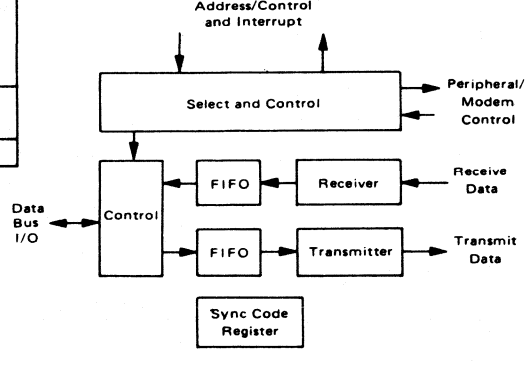
L SUFFIX
CERAMIC PACKAGE
CASE 716

NOT SHOWN: **P SUFFIX**
PLASTIC PACKAGE
CASE 709

ORDERING INFORMATION

Speed	Device	Temperature Range
1.0 MHz	MC6852P,L	0 to 70°C
	MC6852CP,CL	−40 to +85°C
MIL-STD-883B	MC6852BJCS	−55 to +125°C
MIL-STD-883C	MC6852CJCS	
1.5 MHz	MC68A52P,L	0 to +70°C
	MC68A52CP,CL	−40 to +85°C
2.0 MHz	MC68B52P,L	0 to +70°C

MC6852 SYNCHRONOUS SERIAL DATA ADAPTER BLOCK DIAGRAM



MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	−0.3 to +7.0	Vdc
Input Voltage	V_{in}	−0.3 to +7.0	Vdc
Operating Temperature Range—T_L to T_H MC6852, MC68A52, MC68B52 MC6852C, MC68A52C MC6800BJCS, MC6852CJCS	T_A	T_L T_H 0 to +70 −40 to +85 −55 to +125	°C
Storage Temperature Range	T_{stg}	−55 to +150	°C
Thermal Resistance Plastic Package Ceramic Package	θ_{JA}	 120 60	°C/W

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, is is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit.

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ±5%, V_{SS} = 0, T_A = T_L to T_H unless otherwise noted.)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage		V_{IH}	V_{SS} + 2.0		—	Vdc
Input Low Voltage		V_{IL}	—		V_{SS} + 0.8	Vdc
Input Leakage Current (V_{in} = 0 to 5.25 Vdc)	Tx Clk, Rx Clk, Rx Data, Enable, $\overline{\text{Reset}}$, RS, R/$\overline{W}$, $\overline{CS}$, $\overline{DCD}$, $\overline{CTS}$	I_{in}	—	1.0	2.5	μAdc
Three-State (Off State) Input Current (V_{in} = 0.4 to 2.4 Vdc, V_{CC} = 5.25 Vdc)	D0–D7	I_{TSI}	—	2.0	10	μAdc
Output High Voltage (I_{Load} = −205 μAdc, Enable Pulse Width < 25 μs) (I_{Load} = −100 μAdc, Enable Pulse Width < 25 μs)	D0–D7 Tx Data, $\overline{DTR}$, TUF	V_{OH}	V_{SS} + 2.4 V_{SS} + 2.4	— —		Vdc
Output Low Voltage (I_{Load} = 1.6 mAdc, Enable Pulse Width < 25 μs)		V_{OL}	—	—	V_{SS} + 0.4	Vdc
Output Leakage Current (Off State) (V_{OH} = 2.4 Vdc)	$\overline{IRQ}$	I_{LOH}	—	1.0	10	μAdc
Power Dissipation		P_D	—	300	525	mW
Input Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	D0–D7 All Other Inputs	C_{in}	— —	—	12.5 7.5	pF
Output Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	Tx Data, SM/$\overline{DTR}$, TUF $\overline{IRQ}$	C_{out}	—	—	10 5.0	pF

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ± 5%, V_{SS} = 0, T_A = T_L to T_H unless otherwise noted.)

Characteristic	Symbol	MC6852		MC68A52		MC68B52		Unit
		Min	Max	Min	Max	Min	Max	
Minimum Clock Pulse Width, Low (Figure 1)	PW_{CL}	700	—	400	—	280	—	ns
Minimum Clock Pulse Width, High (Figure 2)	PW_{CH}	700	—	400	—	280	—	ns
Clock Frequency	f_C	—	600	—	1000	—	1500	kHz
Receive Data Setup Time (Figure 3, 7)	t_{RDSU}	350	—	200	—	160	—	ns
Receive Data Hold Time (Figure 3)	t_{RDH}	350	—	200	—	160	—	ns
Sync Match Delay Time (Figure 3)	t_{SM}	—	1.0	—	0.666	—	0.500	μs
Clock-to-Data Delay for Transmitter (Figure 4)	t_{TDD}	—	1.0	—	0.666	—	0.500	μs
Transmitter Underflow (Figure 4, 6)	t_{TUF}	—	1.0	—	0.666	—	0.500	μs
$\overline{DTR}$ Delay Time (Figure 5)	t_{DTR}	—	1.0	—	0.666	—	0.500	μs
Interrupt Request Release Time (Figure 5)	t_{IR}	—	1.2	—	0.800	—	0.600	μs
$\overline{\text{Reset}}$ Minimum Pulse Width	t_{Res}	1.0	—	0.666	—	0.500	—	μs
$\overline{CTS}$ Setup Time (Figure 6)	t_{CTS}	200	—	150	—	120	—	ns
$\overline{DCD}$ Setup Time (Figure 7)	t_{DCD}	500	—	350	—	250	—	ns
Input Rise and Fall Times (except Enable) (0.8 V to 2.0 V)	t_r, t_f	—	1.0*	—	1.0*	—	1.0*	μs

*1.0 μs or 10% of the pulse width, whichever is smaller

FIGURE 1 – CLOCK PULSE WIDTH, LOW-STATE

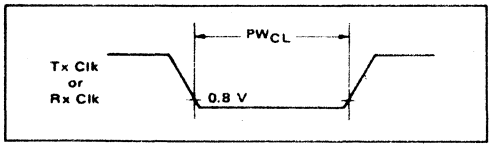
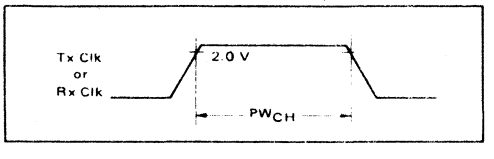


FIGURE 2 – CLOCK PULSE WIDTH, HIGH-STATE



BUS TIMING CHARACTERISTICS

Characteristic	Symbol	MC6852		MC68A52		MC68B52		Unit
		Min	Max	Min	Max	Min	Max	
Enable Cycle Time	t_{cycE}	1.0	–	0.666	–	0.5	–	μs
Enable Pulse Width, High	PW_{EH}	0.45	25	0.28	25	0.22	25	μs
Enable Pulse Width, Low	PW_{EL}	0.43	–	0.28	–	0.21	–	μs
Setup Time, Address and R/$\overline{W}$ valid to Enable positive transition	t_{AS}	160	–	140	–	70	–	ns
Data Delay Time	t_{DDR}	–	320	–	220	–	180	ns
Data Hold Time	t_H	10	–	10	–	10	–	ns
Address Hold Time	t_{AH}	10	–	10	–	10	–	ns
Rise and Fall Time for Enable input	t_{Er}, t_{Ef}	–	25	–	25	–	25	ns
WRITE (Figures 9 and 10)								
Enable Cycle Time	t_{cycE}	1.0	–	0.666	–	0.5	–	μs
Enable Pulse Width, High	PW_{EH}	0.45	25	0.28	25	0.22	25	μs
Enable Pulse Width, Low	PW_{EL}	0.43	–	0.28	–	0.21	–	μs
Setup Time, Address and R/$\overline{W}$ valid to Enable positive transition	t_{AS}	160	–	140	–	70	–	ns
Data Setup Time	t_{DSW}	195	–	80	–	60	–	ns
Data Hold Time	t_H	10	–	10	–	10	–	ns
Address Hold Time	t_{AH}	10	–	10	–	10	–	ns
Rise and Fall Time for Enable input	t_{Er}, t_{Ef}	–	25	–	25	–	25	ns

FIGURE 3 – RECEIVE DATA SETUP AND HOLD TIMES AND SYNC MATCH DELAY TIME

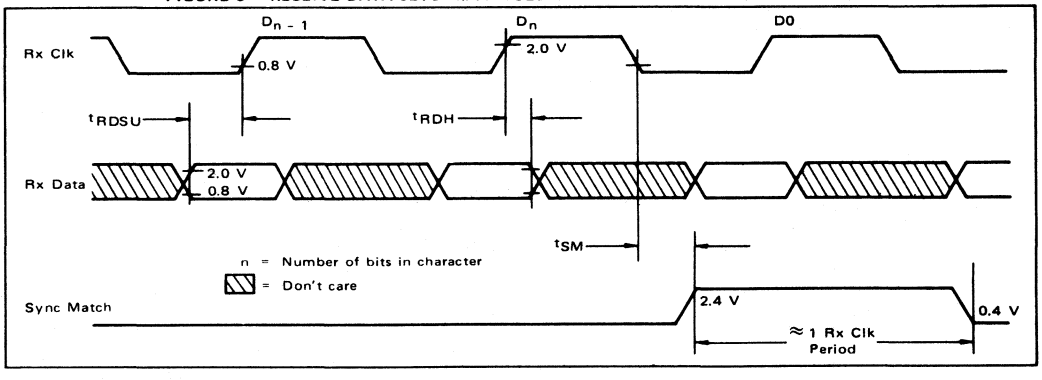


FIGURE 4 – TRANSMIT DATA OUTPUT DELAY AND TRANSMITTER UNDERFLOW DELAY TIME

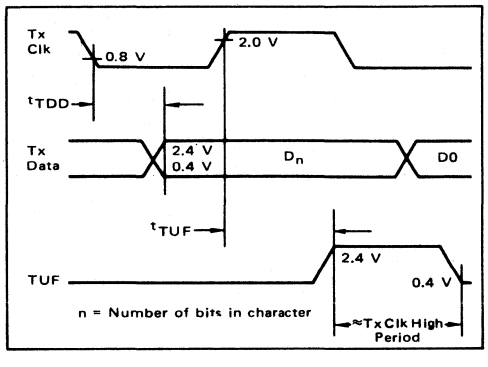


FIGURE 5 – $\overline{\text{DATA TERMINAL READY}}$ AND $\overline{\text{INTERRUPT REQUEST}}$ RELEASE TIMES

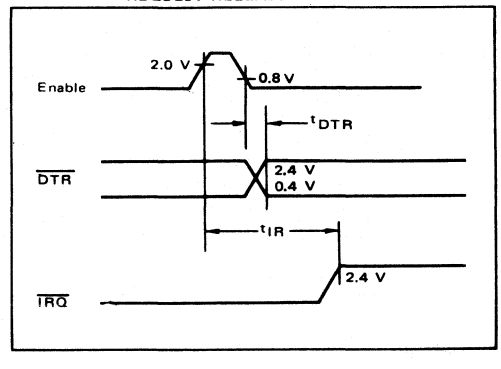


FIGURE 6 – $\overline{\text{CLEAR-TO-SEND}}$ SETUP TIME

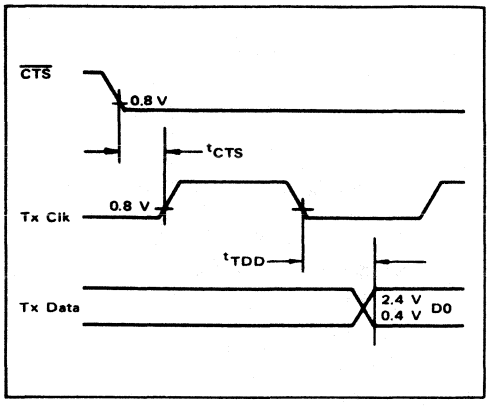


FIGURE 7 – $\overline{\text{DATA CARRIER DETECT}}$ SETUP TIME

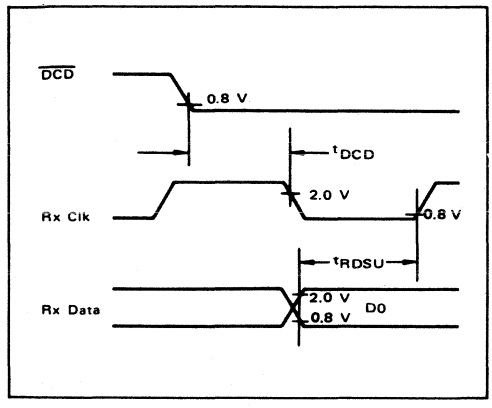


FIGURE 8 – BUS READ TIMING CHARACTERISTICS
(Read information from SSDA)

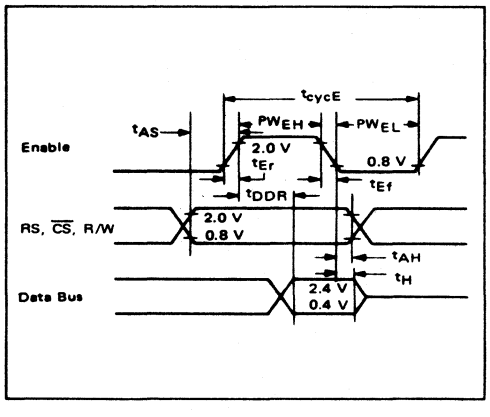


FIGURE 9 – BUS WRITE TIMING CHARACTERISTICS
(Write information into SSDA)

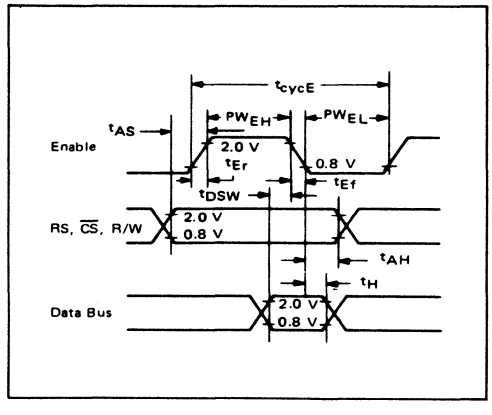
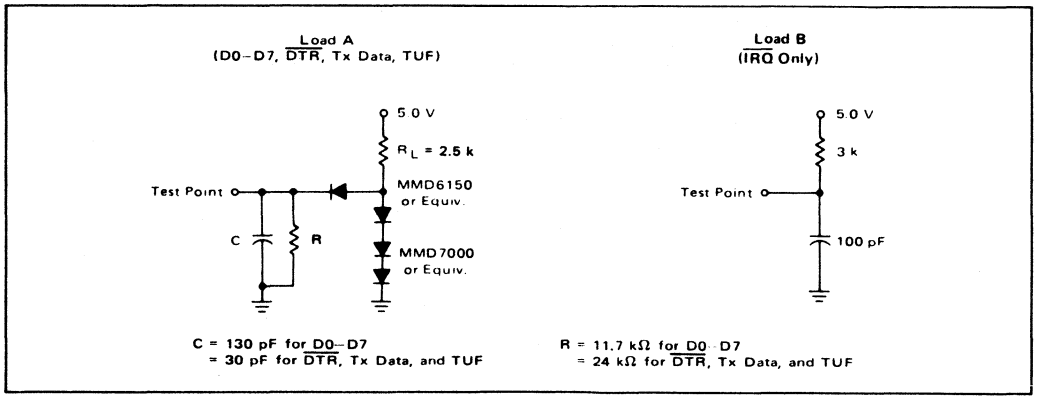
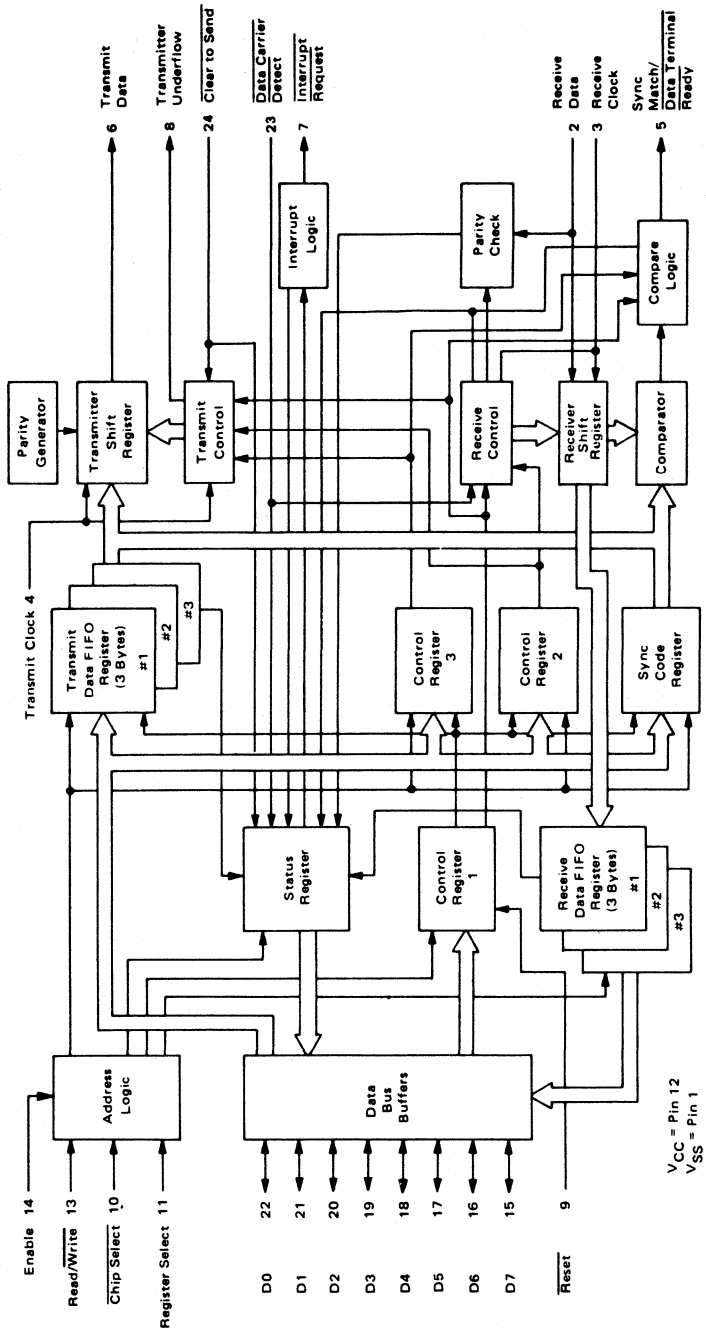


FIGURE 10 – BUS TIMING TEST LOADS



EXPANDED BLOCK DIAGRAM



DEVICE OPERATION

At the bus interface, the SSDA appears as two addressable memory locations. Internally, there are seven registers: two read-only and five write-only registers. The read-only registers are Status and Receive Data; the write-only registers are Control 1, Control 2, Control 3, Sync Code and Transmit Data. The serial interface consists of serial input and output lines with independent clocks, and four peripheral/modem control lines.

Data to be transmitted is transferred directly into the 3-byte Transmit Data First-In First-Out (FIFO) Register from the data bus. Availability of the input to the FIFO is indicated by a bit in the Status Register; once data is entered, it moves through the FIFO to the last empty location. Data at the output of the FIFO is automatically transferred from the FIFO to the Transmitter Shift Register as the shift register becomes available to transmit the next character. If data is not available from the FIFO (underflow condition), the Transmitter Shift Register is automatically loaded with either a sync code or an all "1"s character. The transmit section may be programmed to append even, odd, or no parity to the transmitted word. An external control line ($\overline{\text{Clear-to-Send}}$) is provided to inhibit the transmitter without clearing the FIFO.

Serial data is accumulated in the receiver based on the synchronization mode selected. In the external sync mode, used for parallel-serial operation, the receiver is synchronized by the $\overline{\text{DCD}}$ ($\overline{\text{Data Carrier Detect}}$) input and transfers successive bytes of data to the input of the Receiver FIFO. The single-sync-character mode requires that a match occur between the Sync Code Register and one incoming character before data transfer to the FIFO begins. The two-sync-character mode requires that two sync codes be received in sequence to establish synchronization. Subsequent to synchronization in any mode, data is accumulated in the shift register, and parity is optionally checked. An indication of parity error is carried through the Receiver FIFO with each character to the last empty location. Availability of a word at the FIFO output is indicated by a bit in the Status Register, as is a parity error.

The SSDA and its internal registers are selected by the address bus, Read/$\overline{\text{Write}}$ (R/$\overline{\text{W}}$) and Enable control lines. To configure the SSDA, Control Registers are selected and the appropriate bits set. The Status Register is addressable for reading status.

Other I/O lines, in addition to $\overline{\text{Clear-to-Send}}$ ($\overline{\text{CTS}}$) and $\overline{\text{Data Carrier Detect}}$ ($\overline{\text{DCD}}$), include SM/$\overline{\text{DTR}}$ (Sync Match/$\overline{\text{Data Terminal Ready}}$) and Transmitter Underflow (TUF). The transmitter and receiver each have individual clock inputs allowing simultaneous operation under separate clock control. Signals to the microprocessor are the Data bus and $\overline{\text{Interrupt Request}}$ ($\overline{\text{IRQ}}$).

INITIALIZATION

During a power-on sequence, the SSDA is reset via the $\overline{\text{Reset}}$ input and internally latched in a reset condition to prevent erroneous output transitions. The Sync Code Register, Control Register 2, and Control Register 3 should be programmed prior to the programmed release of the Transmitter and/or Receiver Reset bits; these bits in Control Register 1 should be cleared after the $\overline{\text{Reset}}$ line has gone high.

TRANSMITTER OPERATION

Data is transferred to the transmitter section in parallel form by means of the data bus and Transmit Data FIFO. The Transmit Data FIFO is a 3-byte register whose status is indicated by the Transmitter Data Register Available status bit (TDRA) and its associated interrupt enable bit. Data is transferred through the FIFO on negative edges of Enable (E) pulses. Two data transfer modes are provided in the SSDA: The 1-byte transfer mode provides for writing data to the transmitter section (and reading from the receiver section) one byte at a time. The 2-byte transfer mode provides for writing two data characters in succession.

Data will automatically transfer from the last register location in the Transmit Data FIFO (when it contains data) to the Transmitter Shift Register during the last half of the last bit of the previous character. A character is transferred into the Shift Register by the Transmitter Clock. Data is transmitted *LSB first,* and odd or even parity can be optionally appended. The unused bit positions in short word length characters from the data bus are "don't cares". (Note: The data bus inputs may be reversed for applications requiring the MSB to be transferred first, e.g., IBM format for floppy disks; however, care must be taken to properly program the control registers – Table 1 will have its bit positions reversed.)

When the Shift Register becomes empty, and data is *not* available for transfer from the Transmit Data FIFO, an "underflow" occurs, and a character is inserted into the transmitter data stream to maintain character synchronization. The character transmitted on underflow will be either a "Mark" (all "1"s) or the contents of the Sync Code Register, depending upon the state of the Transmit Sync Code on Underflow control bit. The underflow condition is indicated by a pulse ($\approx$1 Tx Clk high period) on the Underflow output (when in Tx Sync on underflow mode). The Underflow output occurs coincident with the transfer of the last half of the last bit preceding the underflow character. The Underflow status bit is set until cleared by means of the Clear Underflow control bit. This output may be used in floppy disk systems to synchronize write operations and for appending CRCC.

Transmission is initiated by clearing the Transmitter Reset bit in Control Register 1. When the Transmitter Reset bit is cleared, the first *full* positive half-cycle of the Transmit Clock will initiate the transmit cycle, with the transmission of data or underflow characters beginning on the negative edge of the Transmit Clock pulse which started the cycle. If the Transmit Data FIFO was not loaded,

an underflow character will be transmitted (see Figure 4).

The Clear-to-Send ($\overline{CTS}$) input provides for automatic control of the transmitter by means of external system hardware; e.g., the modem $\overline{CTS}$ output provides the control in a data communications system. The $\overline{CTS}$ input resets and inhibits the transmitter section when high, but does not reset the Transmit Data FIFO. The TDRA status bit is inhibited by $\overline{CTS}$ being high in either the one-sync character or two-sync-character mode of operation. In the external sync mode, TDRA is unaffected by $\overline{CTS}$ in order to provide Transmit Data FIFO status for preloading and operating the transmitter under the control of the $\overline{CTS}$ input. When the Transmitter Reset bit (Tx Rs) is set, the Transmit Data FIFO is cleared and the TDRA status bit is cleared. After one E clock has occurred, the Transmit Data FIFO becomes available for new data with TDRA inhibited.

RECEIVER OPERATION

Data and a presynchronized clock are provided to the SSDA receiver section by means of the Receive Data (Rx Data) and Receive Clock (Rx Clk) inputs. The data is a continuous stream of binary data bits without means for identifying character boundaries within the stream. It is, therefore, necessary to achieve character synchronization for the data at the *beginning* of the data block. Once synchronization is achieved, it is assumed to be retained for all successive characters within the block.

Data communications systems utilize the detection of sync codes during the initial portion of the preamble to establish character synchronization. This requires the detection of a single code or two successive sync codes. Floppy disk and cartridge tape units require sixteen bits of defined preamble and cassettes require eight bits of preamble to establish the reference for the start of record. All three are functionally equivalent to the detection of sync codes. Systems which do not utilize code detection techniques require custom logic external to the SSDA for character synchronization and use of the parallel-to-serial (external sync) mode. (Note: The Receiver Shift Register is set to ones when reset.)

SYNCHRONIZATION

The SSDA provides three operating modes with respect to character synchronization: one-sync-character mode, two-sync-character mode, and external sync mode. The external sync mode requires synchronization and control of the receiveing section through the Data Carrier Detect ($\overline{DCD}$) input (see Figure 7). This external synchronization could consist of direct line control from the transmitting end of the serial data link or from external logic designed to detect the start of the message block. The one-sync-character mode searches on a bit-by-bit basis until a match is achieved between the data in the Shift Register and the Sync Code Register. The match indicates character synchronization is complete and will be retained for the message block. In the two-sync-character mode, the receiver searches for the first sync code match on a bit-by-bit basis and then looks for a second *successive* sync code character prior to establishing character synchronization. If the second sync code character is not received, the bit-by-bit search for the first sync code is resumed.

Sync codes received prior to the completion of synchronization (one or two character) are not transferred to the Receive Data FIFO. Redundant sync codes during the preamble or sync codes which occur as "fill characters" can automatically be stripped from the data, when the Strip Sync control bit is set, to minimize system loading. The character synchronization will be retained until cleared by means of the Clear Sync bit, which also inhibits synchronization search when set.

RECEIVING DATA

Once synchronization has been achieved, subsequent characters are automatically transferred into the Receive Data FIFO and clocked through the FIFO to the last empty location by E pulses (MPU System $\phi 2$). The Receiver Data Available status bit (RDA) indicates when data is available to be read from the last FIFO location (#3) when in the 1-byte transfer mode. The 2-byte transfer mode causes the RDA status bit to indicate data is available when the last two FIFO register locations are full. Data being available in the Receive Data FIFO causes an interrupt request if the Receiver Interrupt Enable (RIE) bit is set. The MPU will then read the SSDA Status Register, which will indicate that data is available for the MPU read from the Receive Data FIFO register. The IRQ and RDA status bits are reset by a read from the FIFO. If more than one character has been received and is resident in the Receive Data FIFO, subsequent E clocks will cause the FIFO to update and the RDA and IRQ status bits will again be set. The read data operation for the 2-byte transfer mode requires an intervening E clock between reads to allow the FIFO data to shift. Optional parity is automatically checked as data is received, and the parity status condition is maintained with each character until the data is read from the Receive Data FIFO. Parity errors will cause an interrupt request if the Error Interrupt Enable (EIE) has been set. The parity bit is not transferred to the data bus but must be checked in the Status Register. NOTE: In the 2-byte transfer mode, parity should be checked prior to reading the second byte, since a FIFO read clears the error bit.

Other status bits which pertain to the receiver section are Receiver Overrun and Data Carrier Detect ($\overline{DCD}$). The Overrun status bit is automatically set when a transfer of a character to the Receive Data FIFO occurs and the first register of the Receive Data FIFO is full. Overrun causes an interrupt if Error Interrupt Enable (EIE) has been set. The transfer of the overrunning character into the FIFO causes the previous character in the FIFO input register location to be lost. The Overrun status bit is cleared by reading the Status Register (when the overrun condition is present), followed by a Receive Data FIFO

Register read. Overrun cannot occur and be cleared without providing an opportunity to detect its occurrence via the Status Register.

A positive transition on the $\overline{DCD}$ input causes an interrupt if the EIE control bit has been set. The interrupt caused by $\overline{DCD}$ is cleared by reading the Status Register when the $\overline{DCD}$ status bit is high, followed by a Receive Data FIFO read. The $\overline{DCD}$ status bit will subsequently follow the state of the $\overline{DCD}$ input when it goes low.

INPUT/OUTPUT FUNCTIONS

SSDA INTERFACE SIGNALS FOR MPU

The SSDA interfaces to the MC6800 MPU with an 8-bit bi-directional data bus, a chip select line, a register select line, an interrupt request line, read/write line, an enable line, and a reset line. These signals, in conjunction with the MC6800 VMA output, permit the MPU to have complete control over the SSDA.

SSDA Bi-Directional Data (D0-D7) – The bi-directional data lines (D0-D7) allow for data transfer between the SSDA and the MPU. The data bus output drivers are three-state devices that remain in the high-impedance (off) state except when the MPU performs an SSDA read operation.

SSDA Enable (E) – The Enable signal, E, is a high impedance TTL compatible input that enables the bus input/output data buffers, clocks data to and from the SSDA, and moves data through the FIFO Registers. This signal is normally the continuous MC6800 System $\phi 2$ clock, so that incoming data characters are shifted through the FIFO.

Read/$\overline{\text{Write}}$ (R/$\overline{\text{W}}$) – The Read/$\overline{\text{Write}}$ line is a high impedance input that is TTL compatible and is used to control the direction of data flow through the SSDA's input/output data bus interface. When Read/$\overline{\text{Write}}$ is high (MPU read cycle), SSDA output drivers are turned on if the chip is selected and a selected register is read. When it is low, the SSDA output drivers are turned off and the MPU writes into a selected register. The Read/Write signal is also used to select read-only or write-only registers within the SSDA.

$\overline{\text{Chip Select}}$ ($\overline{\text{CS}}$) – This high impedance TTL compatible input line is used to address the SSDA. The SSDA is selected when $\overline{CS}$ is low. VMA should be used in generating the $\overline{CS}$ input to insure that false selects will not occur. Transfers of data to and from the SSDA are then performed under the control of the Enable signal, Read/Write, and Register Select.

Register Select (RS) – The Register Select line is a high impedance input that is TTL compatible. A high level is used to select Control Registers C2 and C3, the Sync Code Register, and the Transmit/Receive Data Registers. A low level selects the Control 1 and Status Registers (see Table 1).

$\overline{\text{Interrupt Request}}$ ($\overline{\text{IRQ}}$) – $\overline{\text{Interrupt Request}}$ is a TTL compatible, open-drain (no internal pullup), active low output that is used to interrupt the MPU. The $\overline{\text{Interrupt Request}}$ remains low until cleared by the MPU.

$\overline{\text{Reset}}$ Input – The $\overline{\text{Reset}}$ input provides a means of resetting the SSDA from an external source. In the low state, the $\overline{\text{Reset}}$ input causes the following:

1. Receiver Reset (Rx Rs) and Transmitter Reset (Tx Rs) bits are set causing both the receiver and transmitter sections to be held in a reset condition.
2. Peripheral Control bits PC1 and PC2 are reset to zero, causing the SM/$\overline{DTR}$ output to be high.
3. The Error Interrupt Enable (EIE) bit is reset.
4. An internal synchronization mode is selected.
5. The Transmitter Data Register Available (TDRA) status bit is cleared and inhibited.

When $\overline{\text{Reset}}$ returns high (the inactive state), the transmitter and receiver sections will remain in the reset state until the Receiver Reset and Transmitter Reset bits are cleared via the bus under software control. The control Register bits affected by $\overline{\text{Reset}}$ (Rx Rs, Tx Rs, PC1, PC2, EIE, and E/I Sync) cannot be changed when $\overline{\text{Reset}}$ is low.

CLOCK INPUTS

Separate high impedance TTL compatible inputs are provided for clocking of transmitted and received data.

Transmit Clock (Tx Clk)– The Transmit Clock input is used for the clocking of transmitted data. The transmitter shifts data on the negative transition of the clock.

Receive Clock (Rx Clk) – The Receive Clock input is used for clocking in received data. The clock and data must be synchronized externally. The receiver samples the data on the positive transition of the clock.

SERIAL INPUT/OUTPUT LINES

Receive Data (Rx Data) – The Receive Data line is a high impedance TTL compatible input through which data is received in a serial format. Data rates are from 0 to 600 kbps.

Transmit Data (Tx Data) – The Transmit Data output line transfers serial data to a modem or other peripheral. Data rates are from 0 to 600 kbps.

PERIPHERAL/MODEM CONTROL

The SSDA includes several functions that permit limited control of a peripheral or modem. The functions included are $\overline{\text{Clear-to-Send}}$, Sync Match/$\overline{\text{Data Terminal Ready}}$, $\overline{\text{Data Carrier Detect}}$, and Transmitter Underflow.

$\overline{\text{Clear-to-Send}}$ ($\overline{\text{CTS}}$) – The $\overline{CTS}$ input provides a real-time inhibit to the transmitter section (the Tx Data FIFO is not disturbed). A positive $\overline{CTS}$ transition resets the Tx Shift Register and inhibits the TDRA status bit and its associated interrupt in both the one-sync-character and two-

sync-character modes of operation. TDRA is not affected by the $\overline{CTS}$ input in the external sync mode.

The positive transition of $\overline{CTS}$ is stored within the SSDA to insure that its occurrence will be acknowledged by the system. The stored $\overline{CTS}$ information and its associated $\overline{IRQ}$ (if enabled) are cleared by writing a "1" in the Clear. $\overline{CTS}$ bit in Control Register 3 or in the Transmitter Reset bit. The $\overline{CTS}$ status bit subsequently follows the $\overline{CTS}$ input when it goes low.

The $\overline{CTS}$ input provides character timing for transmitter data when in the external sync mode. Transmission is initiated on the negative transition of the first *full* positive clock pulse of the transmitter clock (Tx Clk) after the release of $\overline{CTS}$ (see Figure 6).

Data Carrier Detect ($\overline{DCD}$) – The $\overline{DCD}$ input provides a real-time inhibit to the receiver section (the Rx FIFO is not disturbed). A positive $\overline{DCD}$ transition resets and inhibits the receiver section except for the Receive FIFO and the RDRA status bit and its associated $\overline{IRQ}$.

The positive transition of $\overline{DCD}$ is stored within the SSDA to insure that its occurrence will be acknowledged by the system. The stored $\overline{DCD}$ information and its associated $\overline{IRQ}$ (if enabled) are cleared by reading the Status Register and then the Receiver FIFO, or by writing a "1" into the Receiver Reset bit. The $\overline{DCD}$ status bit subsequently follows the $\overline{DCD}$ input when it goes low. The $\overline{DCD}$ input provides character synchronization timing for the receiver during the external sync mode of operation. The receiver will be initialized and data will be sampled on the positive transition of the first *full* Receive Clock cycle after release of $\overline{DCD}$ (see Figure 7).

Sync Match/Data Terminal Ready (SM/$\overline{DTR}$) – The SM/$\overline{DTR}$ output provides four functions (see Table 1) depending on the state of the PC1 and PC2 control bits. When the Sync Match mode is selected (PC1 = "1", PC2 = "0"), the output provides a one-bit-wide pulse when a sync code is detected. This pulse occurs for each sync code match even if the receiver has already attained synchronization. The SM output is inhibited when PC2 = "1". The $\overline{DTR}$ mode (PC1 = "0") provides an output level corresponding to the complement of PC2 ($\overline{DTR}$ = "0" when PC2 = "1".) (See Table 1.)

Transmitter Underflow (TUF) – The Underflow output indicates the occurrence of a transfer of a "fill character" to the Transmitter Shift Register when the last location (#3) in the Transmit Data FIFO is empty. The Underflow output pulse is approximately a Tx Clk high period wide and occurs during the last half of the last bit of the character preceding the "Underflow" (see Figure 4). The Underflow output pulse does not occur when the Tx Sync bit is in the reset state.

SSDA REGISTERS

Seven registers in the SSDA can be accessed by means of the bus. The registers are defined as read-only or write-only according to the direction of information flow. The Register Select input (RS) selects two registers in each state, one being read-only and the other write-only. The Read/$\overline{Write}$ input (R/$\overline{W}$) defines which of the two selected registers will actually be accessed. Four registers (two read-only and two write-only) can be addressed via the bus at any particular time. These registers and the required addressing are defined in Table 1.

CONTROL REGISTER 1 (C1)

Control Register 1 is an 8-bit write-only register that can be directly addressed from the data bus. Control Register 1 is addressed when RS = "0" and R/W = "0".

Receiver Reset (Rx Rs), C1 Bit 0 – The Receiver Reset control bit provides both a reset and inhibit function to the receiver section. When Rx Rs is set, it clears the receiver control logic, sync logic, error logic, Rx Data FIFO Control, Parity Error status bit, and $\overline{DCD}$ interrupt. The Receiver Shift Register is set to ones. The Rx Rs bit must be cleared after the occurrence of a low level on $\overline{Reset}$ in order to enable the receiver section of the SSDA.

Transmitter Reset ($\overline{Tx}$ Rs), C1 Bit 1 – The Transmitter Reset control bit provides both a reset and inhibit to the transmitter section. When Tx Rs is set, it clears the transmitter control section, Transmitter Shift Register, Tx Data FIFO Control (the Tx Data FIFO can be reloaded after one E clock pulse), the Transmitter Underflow status bit, and the $\overline{CTS}$ interrupt, and inhibits the TDRA status bit (in the one-sync-character and two-sync-character modes). The Tx Rs bit must be cleared after the occurrence of a low level on $\overline{Reset}$ in order to enable the transmitter section of the SSDA. If the Tx FIFO is not preloaded, it must be loaded immediately after the Tx Rs release to prevent a transmitter underflow condition.

Strip Synchronization Characters (Strip Sync), C1 Bit 2 – If the Strip Sync bit is set, the SSDA will automatically strip all received characters which match the contents of the Sync Code Register. The characters used for synchronization (one or two characters of sync) are always stripped from the received data stream.

Clear Synchronization (Clear Sync), C1 Bit 3 – The Clear Sync control bit provides the capability of dropping receiver character synchronization and inhibiting resynchronization. The Clear Sync bit is set to clear and inhibit receiver synchronization in *all* modes and is reset to zero to enable resynchronization.

Transmitter Interrupt Enable (TIE), C1 Bit 4 – TIE enables both the $\overline{Interrupt\ Request}$ output ($\overline{IRQ}$) and Interrupt Request status bit to indicate a transmitter service request. When TIE is set and the TDRA status bit is high, the $\overline{IRQ}$ output will go low (the active state) and the IRQ status bit will go high.

Receiver Interrupt Enable (RIE), C1 Bit 5 – RIE enables both the $\overline{Interrupt\ Request}$ output ($\overline{IRQ}$) and the

TABLE 1 – SSDA PROGRAMMING MODEL

Register	Control Inputs		Address Control		Register Content							
	RS	R/W	AC2	AC1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Status (S)	0	1	X	X	Interrupt Request (IRQ)	Receiver Parity Error (PE)	Receiver Overrun (Rx Ovrn)	Transmitter Underflow (TUF)	Clear-to-Send ($\overline{CTS}$)	Data Carrier Detect ($\overline{DCD}$)	Transmitter Data Register Available (TDRA)	Receiver Data Available (RDA)
Control 1 (C1)	0	0	X	X	Address Control 2 (AC2)	Address Control 1 (AC1)	Receiver Interrupt Enable (RIE)	Transmitter Interrupt Enable (TIE)	Clear Sync	Strip Sync Characters (Strip Sync)	Transmitter Reset (Tx Rs)	Receiver Reset (Rx Rs)
Receive Data FIFO	1	1	X	X	D7	D6	D5	D4	D3	D2	D1	D0
Control 2 (C2)	1	0	0	0	Error Interrupt Enable (EIE)	Transmit Sync Code on Underflow (Tx Sync)	Word Length Select 3 (WS3)	Word Length Select 2 (WS2)	Word Length Select 1 (WS1)	1-Byte/2-Byte Transfer (1-Byte/2-Byte)	Peripheral Control 2 (PC2)	Peripheral Control 1 (PC1)
Control 3 (C3)	1	0	0	1	Not Used	Not Used	Not Used	Not Used	Clear Transmitter Underflow Status (CTUF)	Clear $\overline{CTS}$ Status (Clear $\overline{CTS}$)	One-Sync-Character/ Two-Sync Character Mode Control (1 Sync/ 2 Sync)	External/ Internal Sync Mode Control (E/I Sync)
Sync Code	1	0	1	0	D7	D6	D5	D4	D3	D2	D1	D0
Transmit Data FIFO	1	0	1	1	D7	D6	D5	D4	D3	D2	D1	D0

X = Don't care

STATUS REGISTER

IRQ Bit 7 The IRQ flag is cleared when the source of the IRQ is cleared. The source is determined by the enables in the Control Registers: TIE, RIE, EIE.

Bits 6-0 indicate the SSDA status at a point in time, and can be *reset* as follows:

PE Bit 6 Read Rx Data FIFO, or a "1" into Rx Rs (C1 Bit 0).

Rx Ovrn Bit 5 Read Status and then Rx Data FIFO, or a "1" into Rx Rs (C1 Bit 0).

TUF Bit 4 A "1" into CTUF (C3 Bit 3) or into Tx Rs (C1 Bit 1).

$\overline{CTS}$ Bit 3 A "1" into Clear $\overline{CTS}$ (C3 Bit 2) or a "1" into Tx Rs (C1 Bit1)

$\overline{DCD}$ Bit 2 Read Status and then Rx Data FIFO or a "1" into Rx Rs (C1 Bit 0)

TDRA Bit 1 Write into Tx Data FIFO.

RDA Bit 0 Read Rx Data FIFO.

CONTROL REGISTER 1

AC2, AC1 Bits 7, 6 Used to access other registers, as shown above.

RIE Bit 5 When "1", enables interrupt on RDA (S Bit 0).

TIE Bit 4 When "1", enables interrupt on TDRA (S Bit 1).

Clear Sync Bit 3 When "1", clears receiver character synchronization.

Strip Sync Bit 2 When "1", strips all sync codes from the received data stream.

Tx Rs Bit 1 When "1", resets and inhibits the transmitter section.

Rx Rs Bit 0 When "1", resets and inhibits the receiver section.

CONTROL REGISTER 3

CTUF Bit 3 When "1", clears TUF (S Bit 4), and IRQ if enabled.

Clear $\overline{CTS}$ Bit 2 When "1", clears $\overline{CTS}$ (S Bit 3), and IRQ if enabled.

1 Sync/2 Sync Bit 1 When "1", selects the one-sync-character mode; when "0", selects the two-sync-character mode.

E/I Sync Bit 0 When "1", selects the external sync mode; when "0", selects the internal sync mode.

CONTROL REGISTER 2

EIE Bit 7 When "1", enables the PE, Rx Ovrn, TUF, $\overline{CTS}$, and $\overline{DCD}$ interrupt flags (S Bits 6 through 2).

Tx Sync Bit 6 When "1", allows sync code contents to be transferred on underflow, and enables the TUF Status bit and output. When "0", an all mark character is transmitted on underflow.

WS3, 2, 1 Bits 5-3 Word Length Select

Bit 5 WS3	Bit 4 WS2	Bit 3 WS1	Word Length
0	0	0	6 Bits + Even Parity
0	0	1	6 Bits + Odd Parity
0	1	0	7 Bits
0	1	1	8 Bits
1	0	0	7 Bits + Even Parity
1	0	1	7 Bits + Odd Parity
1	1	0	8 Bits + Even Parity
1	1	1	8 Bits + Odd Parity

1-Byte/2-Byte Bit 2 When "1", enables the TDRA and RDA bits to indicate when a 1-byte transfer can occur; when "0", the TDRA and RDA bits indicate when a 2-byte transfer can occur.

PC2, PC1 Bits 1-0 SM/$\overline{DTR}$ Output Control

Bit 1 PC2	Bit 0 PC1	SM/$\overline{DTR}$ Output at Pin 5
0	0	1
0	1	Pulse ⎍, 1-Bit Wide, on SM
1	0	0
1	1	SM Inhibited, 0

NOTE: When the SSDA is used in applications requiring the MSB of data to be received and transmitted first, the data bus inputs to the SSDA may be reversed (D0 to D7, etc.). Caution must be used when this is done since the bit positions in this table will be reversed, and the parity should not be selected.

Interrupt Request status bit to indicate a receiver service request. When RIE is set and the RDA status bit is high, the $\overline{IRQ}$ output will go low (the active state) and the IRQ status bit will go high.

Address Control 1 (AC1) and Address Control 2 (AC2), C1 Bits 6 and 7 – AC1 and AC2 select one of the write-only registers – Control 2, Control 3, Sync Code, or Tx Data FIFO–as shown in Table 1, when RS = "1" and R/W = "0".

CONTROL REGISTER 2 (C2)

Control Register 2 is an 8-bit write-only register which can be programmed from the bus when the Address Control bits in Control Register 1 (AC1 and AC2) are reset, RS = "1" and R/W = "0".

Peripheral Control 1 (PC1) and Peripheral Control 2 (PC2), C2 Bits 0 and 1 – Two control bits, PC1 and PC2, determine the operating characteristics of the Sync Match/$\overline{DTR}$ output. PC1, when high, selects the Sync Match mode. PC2 provides the inhibit/enable control for the SM/$\overline{DTR}$ output in the Sync Match mode. A one-bit-wide pulse is generated at the output when PC2 is "0", and a match occurs between the contents of the Sync Code Register and the incoming data even if sync is inhibited (Clear Sync bit = "1"). The Sync Match pulse is referenced to the negative edge of Rx Clk pulse causing the match (see Figure 3).

The $\overline{\text{Data Terminal Ready}}$ ($\overline{DTR}$) mode is selected when PC1 is low. When PC2 = "1" the SM/$\overline{DTR}$ output = "0" and vice versa. The operation of PC2 and PC1 is summarized in Table 1.

1-Byte/2-Byte Transfer (1-Byte/2-Byte), C2 Bit 2 – When 1-Byte/2-Byte is set, the TDRA and RDA status bits will indicate the availability of their respective data FIFO registers for a single byte data transfer. Alternately, if 1 Byte/2 Byte is reset, the TDRA and RDA status bits indicate when two bytes of data can be moved without a second status read. An intervening Enable pulse must occur between data transfers.

Word Length Selects (WS1, WS2, WS3), C2 Bits 3, 4, 5 – Word Length Select bits WS1, WS2, and WS3 select word length of 7, 8, or 9 bits including parity as shown in Table 1.

Transmit Sync Code on Underflow (Tx Sync), C2 Bit 6 – When Tx Sync is set, the transmitter will automatically send a sync character when data is not available for transmission. If Tx Sync is reset, the transmitter will transmit a Mark character (including the parity bit position) on underflow. When the underflow is detected, a pulse approximately a Tx Clk high period wide will occur on the underflow output if the Tx Sync bit is set. Internal parity generation is inhibited during underflow except for sync code fill character transmission in 8 bit plus parity word lengths.

Error Interrupt Enable (EIE), C2 Bit 7 – When EIE is set, the IRQ status bit will go high and the $\overline{IRQ}$ output will go low if:

1. A receiver overrun occurs. The interrupt is cleared by reading the Status Register and reading the Rx Data FIFO.
2. $\overline{DCD}$ input has gone to a "1". The interrupt is cleared by reading the Status Register and reading the Rx Data FIFO.
3. A parity error exists for the character in the last location (#3) of the Rx Data FIFO. The interrupt is cleared by reading the Rx Data FIFO.
4. The $\overline{CTS}$ input has gone to a "1". The interrupt is cleared by writing a "1" in the Clear $\overline{CTS}$ bit, C3 bit 2, or by a Tx Reset.
5. The transmitter has underflowed (in the Tx Sync on Underflow mode). The interrupt is cleared by writing a "1" into the Clear Underflow, C3 bit 3, or Tx Reset.

When EIE is a "0", the IRQ status bit and the $\overline{IRQ}$ output are disabled for the above error conditions. A low level on the $\overline{Reset}$ input resets EIE to "0".

CONTROL REGISTER 3 (C3)

Control Register 3 is a 4-bit write-only register which can be programmed from the bus when RS = "1" and R/W = "0" and Address Control bit AC1 = "1" and AC2 = "0".

External/Internal Sync Mode Control (E/I Sync), C3 Bit 0 – When the E/I Sync Mode bit is high, the SSDA is in the external sync mode and the receiver synchronization logic is disabled. Synchronization can be achieved by means of the $\overline{DCD}$ input or by starting Rx Clk at the midpoint of data bit 0 of a character with $\overline{DCD}$ low. Both the transmitter and receiver sections operate as parallel – serial converters in the External Sync mode. The Clear Sync bit in Control Register 1 acts as a receiver sync inhibit when high to provide a bus controllable inhibit. The Sync Code Register can serve as a transmitter fill character register and a receiver match register in this mode. A "low" on the $\overline{Reset}$ input resets the E/I Sync Mode bit placing the SSDA in the internal sync mode.

One-Sync-Character/Two-Sync-Character Mode Control (1 Sync/2 Sync), C3 Bit 1 – When the 1 Sync/2 Sync bit is set, the SSDA will synchronize on a single match between the received data and the contents of the Sync Code Register. When the 1 Sync/2 Sync bit is reset, two *successive* sync characters must be received prior to receiver synchronization. If the second sync character is not detected, the bit by bit search resumes from the first bit in the second character. See the description of the Sync Code Register for more details.

Clear $\overline{CTS}$ Status (Clear $\overline{CTS}$), C3 Bit 2 – When a "1" is written into the Clear $\overline{CTS}$ bit, the stored status and interrupt are cleared. Subsequently, the $\overline{CTS}$ status bit reflects the state of the $\overline{CTS}$ input. The Clear $\overline{CTS}$ control bit does not affect the $\overline{CTS}$ input nor its inhibit of the transmitter section. The Clear $\overline{CTS}$ command bit is self-

clearing, and writing a "0" into this bit is a nonfunctional operation.

Clear Transmit Underflow Status (CTUF), C3 Bit 3 – When a "1" is written into the CTUF status bit, the CTUF bit and its associated interrupt are reset. The CTUF command bit is self-clearing and writing a "0" into this bit is a nonfunctional operation.

SYNC CODE REGISTER

The Sync Code Register is an 8-bit register for storing the programmable sync code required for received data character synchronization in the one-sync-character and two-sync-character modes. The Sync Code Register also provides for stripping the sync/fill characters from the received data (a programmable option) as well as automatic insertion of fill characters in the transmitted data stream. The Sync Code Register is not utilized for receiver character synchronization in the external sync mode; however, it provides storage of receiver match and transmit fill characters.

The Sync Code Register can be loaded when AC2 and AC1 are a "1" and "0", respectively, and R/W = "0" and RS = "1".

The Sync Code Register may be changed after the detection of a match with the received data (the first sync code having been detected) to synchronize with a double-word sync pattern. (This sync code change must occur prior to the completion of the second character.) The sync match (SM) output can be used to interrupt the MPU system to indicate that the first eight bits have matched. The service routine would then change the sync match register to the second half of the pattern. Alternately, the one-sync-character mode can be used for sync codes for 16 or more bits by using software to check the second and subsequent bytes after reading them from the FIFO.

The detection of the sync code can be programmed to appear on the Sync Match/$\overline{DTR}$ output by writing a "1" in PC1 (C2 bit 0) and a "0" in PC2 (C2 bit 1). The Sync Match output will go high for one bit time beginning at the character interface between the sync code and the next character (see Figure 3).

PARITY FOR SYNC CHARACTER

Transmitter

Transmitter does not generate parity for the sync character except 9-bit mode.

9-bit (8-bit + parity) . . . 8-bit sync character + parity
8-bit (7-bit + parity) . . . 8-bit sync character (no parity)
7-bit (6-bit + parity) . . . 7-bit sync character (no parity)

Receiver

At Synchronization

Receiver automatically strips the sync character(s) (two sync characters if '2 sync' mode is selected) which is used to establish synchronization. And parity is not checked for these sync characters.

After Synchronization Is Established

When 'strip sync' bit is selected, the sync characters (fill characters) are stripped and parity is not checked for the stripped sync (fill) characters. When 'strip sync' bit is not selected (low), the sync character is assumed to be normal data and it is transferred into FIFO after parity checking. (When non-parity format is selected, parity is not checked.)

Strip Sync (C1 Bit 2)	WS0–WS2 (Data Format) (C2 Bit 3–5)	
1	X	No transfer of sync code. No parity check of sync code.
0	With Parity	*Transfer data and sync codes. Parity check.
0	Without Parity	*Transfer data and sync codes. No parity check.

*Subsequent to synchronization.

It is necessary to pay attention to the selected sync character in the following cases.

1. Data format is (6 + parity), (7 + parity).
2. Strip sync is not selected (low).
3. After synchronization when sync code is used as a fill character.

Transmitter sends sync character without parity, but receiver checks the parity as if it is normal data. Therefore, the sync character should be chosen to match the parity check selected for the receiver in this special case.

RECEIVE DATA FIRST-IN FIRST-OUT REGISTER (Rx Data FIFO)

The Receive Data FIFO Register consists of three 8-bit registers which are used for buffer storage of received data. Each 8-bit register has an internal status bit which monitors its full or empty condition. Data is always transferred from a full register to an adjacent empty register. The transfer from register to register occurs on E pulses. The RDA status bit will be high when data is available in the last location of the Rx Data FIFO.

In an Overrun condition, the overrunning character will be transferred into the full first stage of the FIFO register and will cause the loss of that data character. Successive overruns continue to overwrite the first register of the FIFO. This destruction of data is indicated by means of the Overrun status bit. The Overrun bit will be set when the overrun occurs and remains set until the Status Register is read, followed by a read of the Rx Data FIFO.

Unused data bits for short word lengths (including the parity bit) will appear as "0"s on the data bus when the Rx Data FIFO is read.

TRANSMIT DATA FIRST-IN FIRST-OUT REGISTER (Tx Data FIFO)

The Transmit Data FIFO Register consists of three 8-bit registers which are used for buffer storage of data to be transmitted. Each 8-bit register has an internal status bit which monitors its full or empty condition. Data is always transferred from a full register to an adjacent empty register. The transfer is clocked by E pulses.

The TDRA status bit will be high if the Tx Data FIFO is available for data.

Unused data bits for short word lengths will be handled as "don't cares". The parity bit is not transferred over the data bus since the SSDA generates parity at transmission.

When an Underflow occurs, the Underflow character will be either the contents of the Sync Code Register or an all "1"s character. The underflow will be stored in the Status Register until cleared and will appear on the Underflow output as a pulse approximately a Tx Clk high period wide.

STATUS REGISTER

The Status Register is an 8-bit read-only register which provides the real-time status of the SSDA and the associated serial data channel. Reading the Status Register is a non-destructive process. The method of clearing status bits depends upon the function each bit represents and is discussed for each bit in the register.

Receiver Data Available (RDA), S Bit 0 – The Receiver Data Available status bit indicates when receiver data can be read from the Rx Data FIFO. The receiver data being present in the last register (#3) of the FIFO causes RDA to be high for the 1-byte transfer mode. The RDA bit being high indicates that the last two registers (#2 and #3) are full when in the 2-byte transfer mode. The second character can be read without a second status read (to determine that the character is available). An E pulse must occur between reads of the Rx Data FIFO to allow the FIFO to shift. Status must be read on a word-by-word basis if receiver data error checking is important. The RDA status bit is reset automatically when data is not available.

Transmitter Data Register Available (TDRA), S Bit 1 – The TDRA status bit indicates that data can be loaded into the Tx Data FIFO Register. The first register (#1) of the Tx Data FIFO being empty will be indicated by a high level in the TDRA status bit in the 1-byte transfer mode. The first two registers (#1 and #2) must be empty for TDRA to be high when in the 2-byte transfer mode. The Tx Data FIFO can be loaded with two bytes without an intervening status read; however, one E pulse must occur between loads. TDRA is inhibited by the Tx Reset or $\overline{\text{Reset}}$. When Tx Reset is set, the Tx Data FIFO is cleared and then released on the next E clock pulse. The Tx Data FIFO can then be loaded with up to three characters of data, even though TDRA is inhibited. This feature allows preloading data prior to the release of Tx Reset. A high level on the $\overline{\text{CTS}}$ input inhibits the TDRA status bit in either sync mode of operation (one-sync-character or two-sync-character). $\overline{\text{CTS}}$ does not affect TDRA in the external sync mode. This enables the SSDA to operate under the control of the $\overline{\text{CTS}}$ input with TDRA indicating the status of the Tx Data FIFO. The $\overline{\text{CTS}}$ input does not clear the Tx Data FIFO in any operating mode.

Data Carrier Detect ($\overline{\text{DCD}}$), S Bit 2 – A positive transition on the $\overline{\text{DCD}}$ input is stored in the SSDA until cleared by reading both Status and Rx Data FIFO. A "1" written into Rx Rs also clears the stored $\overline{\text{DCD}}$ status. The $\overline{\text{DCD}}$ status bit, when set, indicates that the $\overline{\text{DCD}}$ input has gone high. The reading of both Status and Receive Data FIFO allows Bit 2 of subsequent Status reads to indicate the state of the $\overline{\text{DCD}}$ input until the next positive transition.

Clear-to-Send ($\overline{\text{CTS}}$), S Bit 3 – A positive transition on the $\overline{\text{CTS}}$ input is stored in the SSDA until cleared by writing a "1" into the Clear $\overline{\text{CTS}}$ control bit or the Tx Rs bit. The $\overline{\text{CTS}}$ status bit, when set, indicates that the $\overline{\text{CTS}}$ input has gone high. The Clear $\overline{\text{CTS}}$ command (a "1" into C3 Bit 2) allows Bit 3 of subsequent Status reads to indicate the state of the $\overline{\text{CTS}}$ input until the next positive transition.

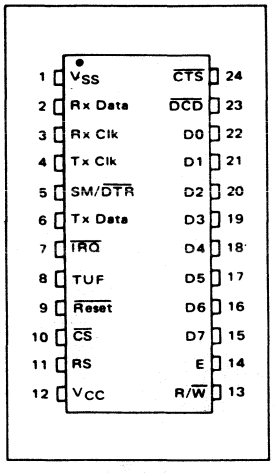
Transmitter Underflow (TUF), S Bit 4 – When data is not available for the transmitter, an underflow occurs and is so indicated in the Status Register (in the Tx Sync on underflow mode). The underflow status bit is cleared by writing a "1" into the Clear Underflow (CTUF) control bit or the Tx Rs bit. TUF indicates that a sync character will be transmitted as the next character. A TUF is indicated on the output *only* when the contents of the Sync Code Register is to be transferred (transmit sync code on underflow = "1").

Receiver Overrun (Rx Ovrn), S Bit 5 – Overrun indicates data has been received when the Rx Data FIFO is full, resulting in data loss. The Rx Ovrn status bit is set when Overrun occurs. The Rx Ovrn status bit is cleared by reading Status followed by reading the Rx Data FIFO or by setting the Rx Rs control bit.

Receiver Parity Error (PE), S Bit 6 – The parity error status bit indicates that parity for the character in the last register of the Rx Data FIFO did not agree with selected parity. The parity error is cleared when the character to which it pertains is read from the Rx Data FIFO or when Rx Rs occurs. The $\overline{\text{DCD}}$ input does not clear the Parity Error or Rx Data FIFO status bits.

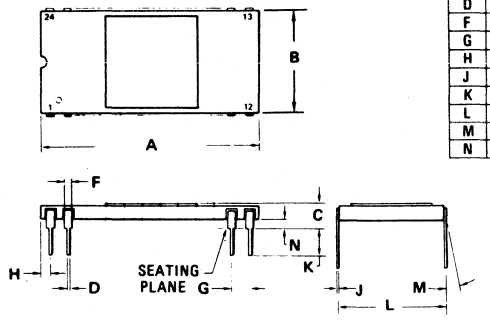
Interrupt Request ($\overline{\text{IRQ}}$), S Bit 7 – The Interrupt Request status bit indicates when the $\overline{\text{IRQ}}$ output is in the active state ($\overline{\text{IRQ}}$ output = "0"). The IRQ status bit is subject to the same interrupt enables (RIE, TIE, and EIE) as the $\overline{\text{IRQ}}$ output. The IRQ status bit simplifies status inquiries for polling systems by providing single bit indication of service requests.

PIN ASSIGNMENT



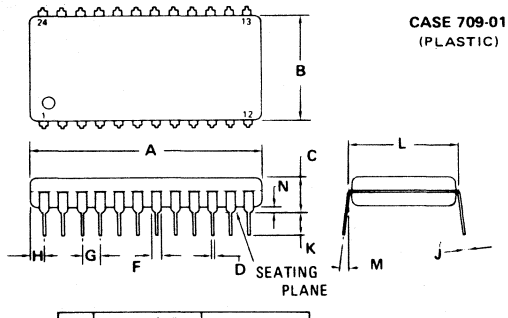
PACKAGE DIMENSIONS

NOTE:
1. LEADS TRUE POSITIONED WITHIN 0.25mm (0.010) DIA (AT SEATING PLANE) AT MAXIMUM MATERIAL CONDITION.



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	29.97	30.99	1.180	1.220
B	14.88	15.62	0.585	0.615
C	3.05	4.19	0.120	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.30	0.008	0.012
K	2.54	4.19	0.100	0.165
L	14.88	15.37	0.585	0.605
M	–	10^0	–	10^0
N	0.51	1.52	0.020	0.060

CASE 716-02
(CERAMIC)



CASE 709-01
(PLASTIC)

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	31.37	32.13	1.235	1.265
B	13.72	14.22	0.540	0.560
C	4.57	5.08	0.180	0.200
D	0.36	0.51	0.014	0.020
F	1.02	1.52	0.040	0.060
G	2.41	2.67	0.095	0.105
H	1.78	2.03	0.070	0.080
J	0.20	0.30	0.008	0.012
K	3.05	3.56	0.120	0.140
L	14.73	15.24	0.580	0.600
M	0^0	10^0	0^0	10^0
N	0.51	1.02	0.020	0.040

NOTES:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION. (DIM. "D")
2. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.

MOTOROLA

MC6854 MC68A54
(1.0 MHz) (1.5 MHz)

MC68B54
(2.0 MHz)

ADVANCED DATA LINK CONTROLLER (ADLC)

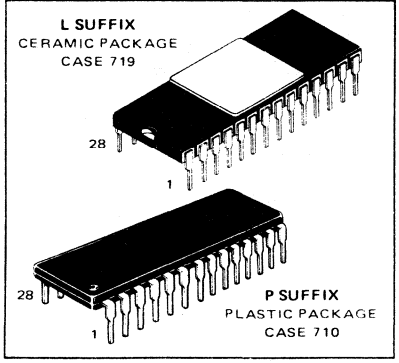
The MC6854 ADLC performs the complex MPU/data communication link function for the "Advanced Data Communication Control Procedure" (ADCCP), High Level Data Link Control (HDLC) and Synchronous Data Link Control (SDLC) standards. The ADLC provides key interface requirements with improved software efficiency. The ADLC is designed to provide the data communications interface for both primary and secondary stations in stand-alone, polling, and loop configurations.

- M6800 Compatible
- Protocol Features
  - Automatic Flag Detection and Synchronization
  - Zero Insertion and Deletion
  - Extendable Address, Control and Logical Control Fields (Optional)
  - Variable Word Length Info Field – 5, 6, 7, or 8 bits
  - Automatic Frame Check Sequence Generation and Check
  - Abort Detection and Transmission
  - Idle Detection and Transmission
- Loop Mode Operation
- Loop Back Self-Test Mode
- NRZ/NRZI Modes
- Quad Data Buffers for Each Rx and Tx
- Prioritized Status Register (Optional)
- MODEM/DMA/Loop Interface
- MIL-STD-883, Class B and C Devices Available

MOS

(N-CHANNEL, SILICON GATE)

ADVANCED DATA LINK CONTROLLER



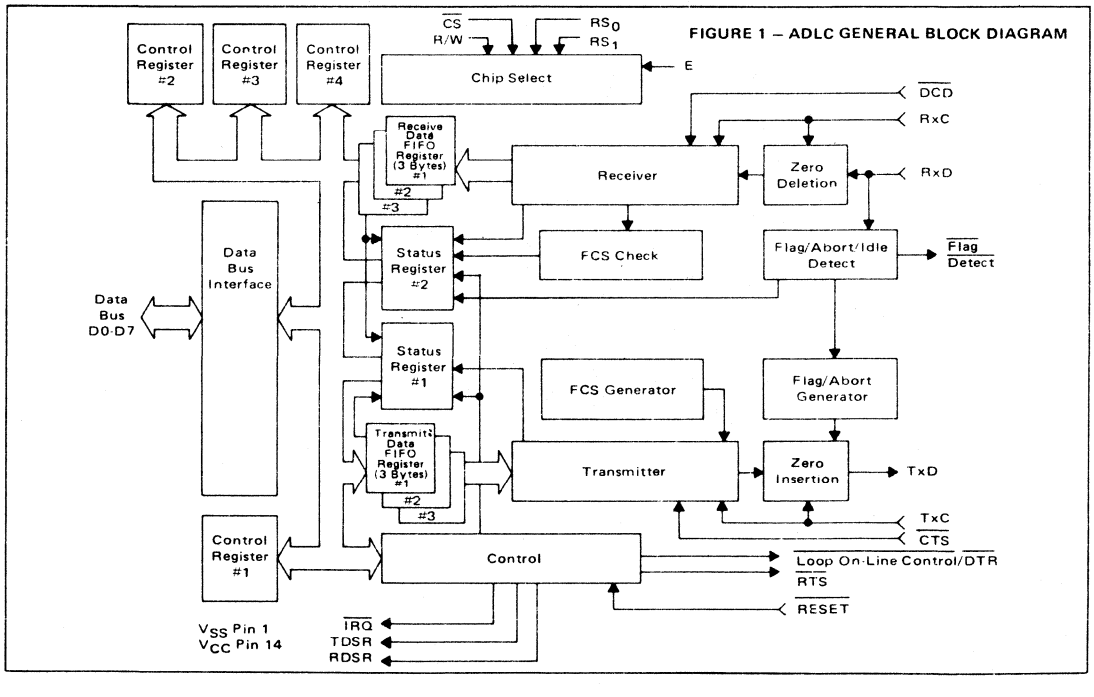


FIGURE 1 – ADLC GENERAL BLOCK DIAGRAM

ORDERING INFORMATION

Speed	Device	Temperature Range
1.0 MHz	MC6854P, L	0 to 70°C
	MC6854CP, CL	-40 to +85°C
MIL-STD-883B	MC6854BQCS	-55 to +125°C
MIL-STD-883C	MC6854CQCS	
1.5 MHz	MC68A54P, L	0 to +70°C
	MC68A54CP, CL	-40 to +85°C
2.0 MHz	MC68B54P, L	0 to +70°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit.

MAXIMUM RATINGS

Rating		Symbol	Value	Unit
Supply Voltage		V_{CC}	-0.3 to +7.0	Vdc
Input Voltage		V_{in}	-0.3 to +7.0	Vdc
Operating Temperature Range		T_A	T_L to T_H	°C
MC6854, MC68A54, MC68B54			0 to 70	
MC6854C, MC68A54C			-40 to 85	
MC6854BTCS, MC6854CTCS			-55 to 125	
Storage Temperature Range		T_{stg}	-65 to +150	°C
Thermal Resistance	Plastic	θ_{JA}	115	°C/W
	Ceramic		60	

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ± 5%, V_{SS} = 0, T_A = T_L to T_H unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage		V_{IH}	V_{SS} + 2.0	–	–	Vdc
Input Low Voltage		V_{IL}	–	–	V_{SS} + 0.8	Vdc
Input Leakage Current (V_{in} = 0 to 5.25 Vdc)	All Inputs Except D0-D7	I_{in}	–	1.0	2.5	μAdc
Three-State (Off State) Input Current (V_{in} = 0.4 to 2.4 Vdc, V_{CC} = 5.25 Vdc)	D0-D7	I_{TSI}	–	2.0	10	μAdc
Output High Voltage		V_{OH}				Vdc
(I_{Load} = -205 μAdc)	D0-D7		V_{SS} + 2.4	–	–	
(I_{Load} = -100 μAdc)	All Others		V_{SS} + 2.4	–	–	
Output Low Voltage (I_{Load} = 1.6 mAdc)		V_{OL}	–	–	V_{SS} + 0.4	Vdc
Output Leakage Current (Off State) (V_{OH} = 2.4 Vdc)	$\overline{IRQ}$	I_{LOH}	–	1.0	10	μAdc
Power Dissipation		P_D	–	–	850	mW
Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)		C_{in}				pF
	D0-D7		–	–	12.5	
	All Other Inputs		–	–	7.5	
	$\overline{IRQ}$	C_{out}	–	–	5.0	pF
	All Others		–	–	10	

Characteristic	Symbol	MC6854		MC68A54		MC68B54		Unit
		Min	Max	Min	Max	Min	Max	
Minimum Clock Pulse Width, Low	PW_{CL}	700	–	450	–	280	–	ns
Minimum Clock Pulse Width, High	PW_{CH}	700	–	450	–	280	–	ns
Clock Frequency	f_C	–	0.66	–	1.0	–	1.5	MHz
Receive Data Setup Time	t_{RDSU}	250	–	200	–	120	–	ns
Receive Data Hold Time	t_{RDH}	120	–	100	–	60	–	ns
Request-to-Send Delay Time	t_{RTS}	–	680	–	460	–	340	ns
Clock-to-Data Delay for Transmitter	t_{TDD}	–	460	–	320	–	250	ns
Flag Detect Delay Time	t_{FD}	–	680	–	460	–	340	ns
$\overline{DTR}$ Delay Time	t_{DTR}	–	680	–	460	–	340	ns
Loop On-Line Control Delay Time	t_{LOC}	–	680	–	460	–	340	ns
RDSR Delay Time	t_{RDSR}	–	540	–	400	–	340	ns
TDSR Delay Time	t_{TDSR}	–	540	–	400	–	340	ns
Interrupt Request Release Time	t_{IR}	–	1.2	–	0.9	–	0.7	μs
$\overline{Reset}$ Minimum Pulse Width	t_{Res}	1.0	–	0.65	–	0.40	–	μs
Input Rise and Fall Times (Except Enable) (0.8 V to 2.0 V)	t_r, t_f	–	1.0*	–	1.0*	–	1.0*	μs

*1.0 μs or 10% of the pulse width, whichever is smaller.

BUS TIMING CHARACTERISTICS (V_{CC} = 5.0 V ± 5%, V_{SS} = 0, T_A = 0 to 70°C unless otherwise noted)

Characteristic	Symbol	MC6854		MC68A54		MC68B54		Unit
		Min	Max	Min	Max	Min	Max	
READ								
Enable Cycle Time	t_{cycE}	1.0	–	0.666	–	0.50	–	μs
Enable Pulse Width, High	PW_{EH}	0.45	–	0.28	–	0.22	–	μs
Enable Pulse Width, Low	PW_{EL}	0.43	–	0.28	–	0.21	–	μs
Setup Time, Address and R/$\overline{W}$ Valid to Enable Positive Transition	t_{AS}	160	–	140	–	70	–	ns
Data Delay Time	t_{DDR}	–	320	–	220	–	180	ns
Data Hold Time	t_H	10	–	10	–	10	–	ns
Address Hold Time	t_{AH}	10	–	10	–	10	–	ns
Rise and Fall Time for Enable Input	t_{Er}, t_{Ef}	–	25	–	25	–	25	ns
WRITE								
Enable Cycle Time	t_{cycE}	1.0	–	0.666	–	0.50	–	μs
Enable Pulse Width, High	PW_{EH}	0.45	–	0.28	–	0.22	–	μs
Enable Pulse Width, Low	PW_{EL}	0.43	–	0.28	–	0.21	–	μs
Set Time, Address and R/$\overline{W}$ Valid to Enable Positive Transition	t_{AS}	160	–	140	–	70	–	ns
Data Setup Time	t_{DSW}	195	–	80	–	60	–	ns
Data Hold Time	t_H	10	–	10	–	10	–	ns
Address Hold Time	t_{AH}	10	–	10	–	10	–	ns
Rise and Fall Time for Enable Input	t_{Er}, t_{Ef}	–	25	–	25	–	25	ns

FIGURE 2 – BUS TIMING TEST LOADS

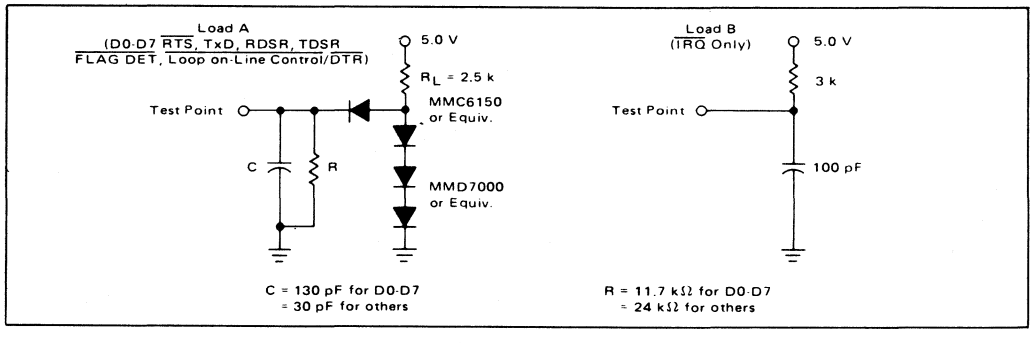


FIGURE 3 – RECEIVER DATA SETUP/HOLD, FLAG DETECT and LOOP ON-LINE CONTROL DELAY TIMING

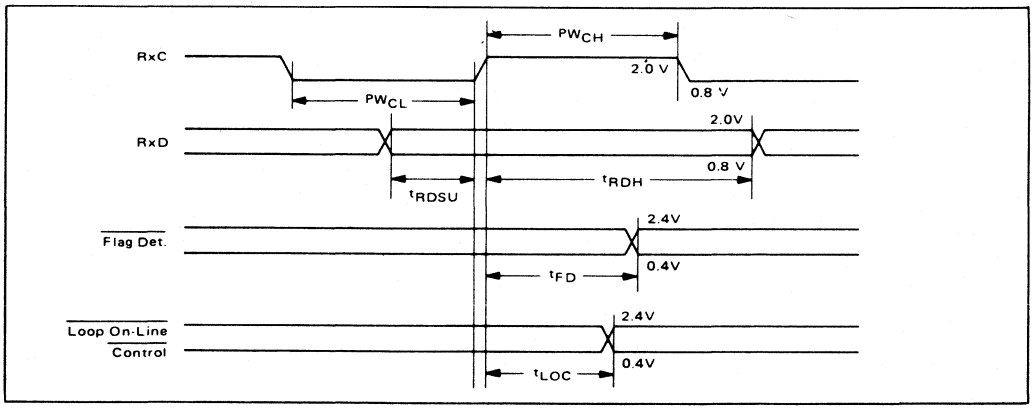


FIGURE 4 – TRANSMIT DATA OUTPUT DELAY AND REQUEST to SEND DELAY TIMING

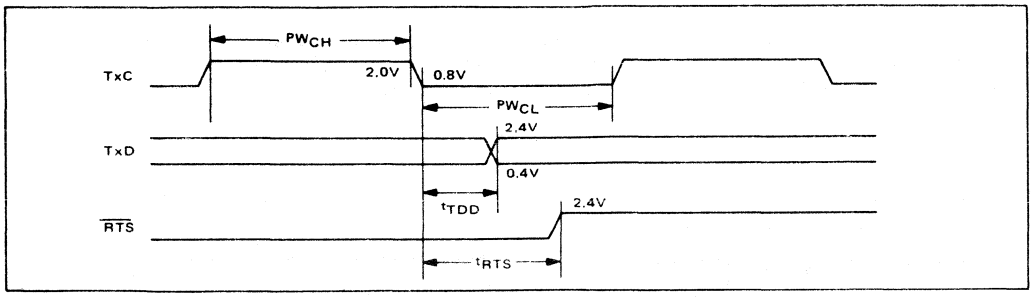


FIGURE 5 – TDSR/RDSR DELAYS, IRQ RELEASE DELAY, RTS and DTR DELAY TIMING

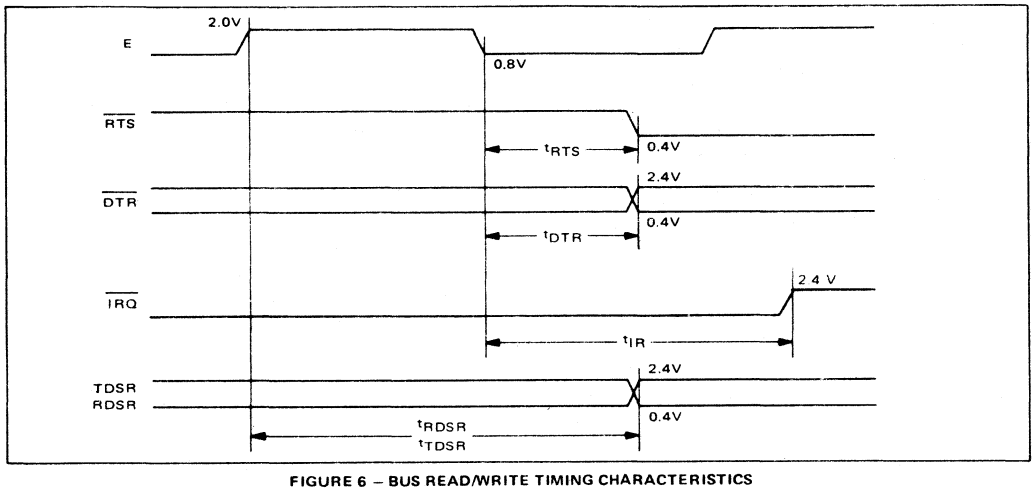
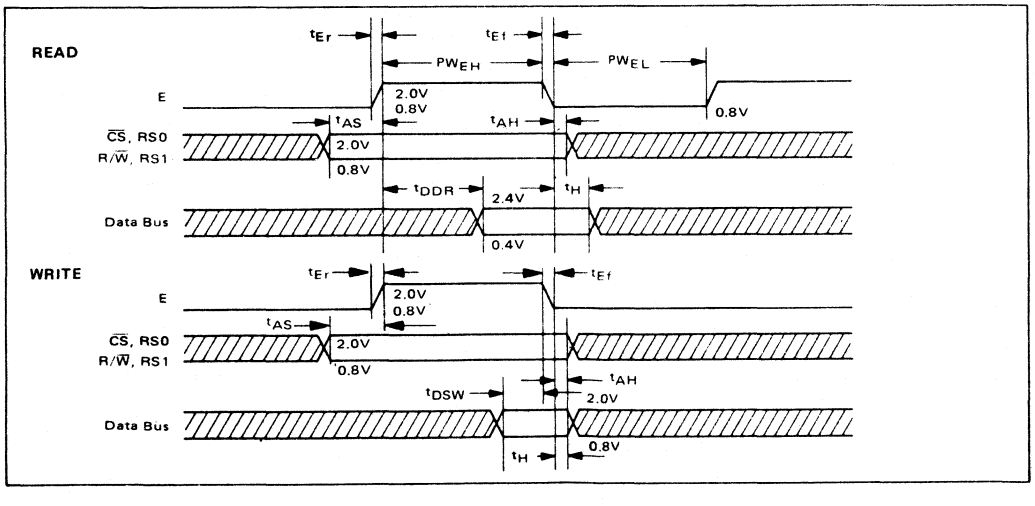


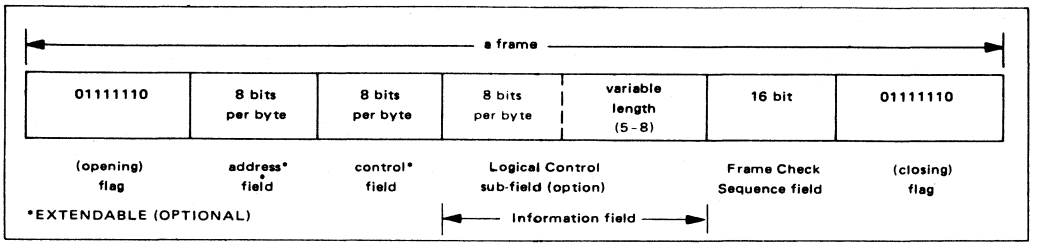
FIGURE 6 – BUS READ/WRITE TIMING CHARACTERISTICS



FRAME FORMAT

The ADLC transmits and receives data (information or control) in a format called a frame. All frames start with an opening flag (F) and end with a closing flag (F). Between the opening flag and closing flag, a frame contains an address field, control field, information field, and frame check sequence field.

FIGURE 7 – DATA FORMAT OF A FRAME



Flag (F) – The flag is the unique binary pattern (01111110). It provides the frame boundary and a reference for the position of each field of the frame.

The ADLC transmitter generates a flag pattern internally and the opening flag and closing flags are appended to a frame automatically. Two successive frames can share one flag for a closing flag of the first frame and for the opening flag of the next frame, if the "FF/"F" control bit in the control register is reset.

The receiver searches for a flag on a bit by bit basis and recognizes a flag at any time. The receiver establishes the frame synchronization with every flag. The flags mark the frame boundary and reference for each field but they are not transferred to the Rx FIFO. The detection of a flag is indicated by the Flag Detect output and by a status bit in the status register.

Order of Bit Transmission – Address, control and information field bytes are transferred between the MPU and the ADLC in parallel by means of the data bus. The bit on D0 (data bus bit 0, pin 22) is serially transmitted first, and the first serially received bit is transferred to the MPU on D0. The FCS field is transmitted and received MSB first.

Address (A) Field – The 8 bits following the opening flag are the address (A) field. The A-field can be extendable if the Auto-Address Extend Mode is selected in control register #3. In the Address Extend Mode, the first bit (bit 0) in every address octet becomes the extend control bit. When the bit is "0", the ADLC assumes another address octet will follow, and when the bit is "1", the address extension is terminated. A "null" address (all "0's") does not extend. In the receiver, the Address Present status bit distinguishes the address field from other fields. When an address byte is available to be read in the receive FIFO register, the Address Present status bit is set and causes an interrupt (if enabled). The Address Present bit is set for every address octet when the Address Extend Mode is used.

Control (C) Field – The 8 bits following the address field is the control (link control) field. When the Extended Control Field bit in control register #3 is selected, the C-field is extended to 16 bits.

Information (I) Field – The I-field follows the C-field and precedes the FCS field. The I-field contains "data" to be transferred but is not always necessarily contained in every frame. The word length of the I-field can be selected from 5 to 8 bits per byte by control bits in control register #4. The I-field will continue until it is terminated by the FCS and closing flag. The receiver has the capability to handle a "partial" last byte. The last information byte can be any word length between 1 and 8 bits. If the last byte in the I-field is less than the selected word length, the receiver will right justify the received bits, fill the remaining bits of the receiver shift register with zeros, and transfer a full byte to the Rx FIFO. Regardless of selected byte length, the ADLC will transfer 8 bits of data to the data bus. Unused bits for word lengths of 5, 6, and 7 will be zeroed.

Logical Control (LC) Field – When the Logical Control Field Select bit in control register #3 is selected, the ADLC separates the I-field into two sub-fields. The first sub-field is the Logical Control field and the following sub-field is the "data" portion of the I-field. The logical control field is 8 bits and follows the C-field, which is extendable by octets, if it is selected. The last bit (bit 7) is the extend control bit, and if it is a "1", the LC-field is extended one octet.

Note: Hereafter the word "Information field" or "I-field" is used as the data potion of the information field, and excludes the logical control field. This is done in order to keep the consistency of the meaning of "Information field" as specified in SDLC, HDLC, and ADCCP standards.

Frame Check Sequence (FCS) Field – The 16 bits preceding the closing flag is the FCS field. The FCS is the "cyclic redundancy check character (CRCC)". The polynomial $x^{16}+x^{12}+x^5+1$ is used both for the transmitter and receiver. Both the transmitter and receiver polynomial registers are initialized to all "1's' prior to calculation of the FCS. The transmitter calculates the FCS on all bits of the address, control, logical control (if selected), and information fields, and transmits the complement of the resulting remainder as FCS. The receiver performs the similar computation on all bits of the address, control, logical control (if selected), information, and received FCS fields and compares the result to F0B8 (Hexadecimal). When the result matches F0B8, the Frame Valid status bit is set in the status register. If the result does not match, the Error status bit is set. The FCS generation, transmission, and checking are performed automatically by the ADLC transmitter and receiver. The FCS field is not transferred to the Rx FIFO.

Invalid Frame – Any valid frames should have at least the A-field, C-field and FCS field between the opening flag and the closing flag. When invalid frames are received, the ADLC handles them as follows:

1) A short frame which has less than 25 bits between flags – The ADLC ignores the short frame and its reception is not reported to the MPU.
2) A frame less than 32 bits between the flags, or a frame 32 bits or more with an extended A-field or C-field that is not completed. – This frame is transferred into the Rx FIFO. The FCS/IF Error status bit indicates the reception of the invalid frame at the end of the frame.
3) Aborted Frame – The frame which is aborted by receiving an abort or DCD failure is also an invalid frame. Refer to "Abort" and "DCD status bit."

Zero Insertion and Zero Deletion – The Zero insertion and deletion, which allows the content of the frame to be transparent, are performed by the ADLC automatically. A binary 0 is inserted by the transmitter after any succession of 5 1's within a frame (A, C, LC, I, and FCS field). The receiver deletes a binary 0 that follows successive 5 continuous 1's within a frame.

Abort – The function of prematurely terminating a data link is called "abort". The transmitter aborts a frame by sending at least 8 consecutive 1's immediately after the Tx Abort control bit in control register #4 is set to a "1". (Tx FIFO is also cleared by the Tx Abort control bit at the same time.) The abort can be extended up to (at least) 16 consecutive 1's, if the Abort Extend control bit in the control register #4 is set when an abort is sent. This feature is useful to force mark idle transmission. Reception of 7 or more consecutive 1's is interpreted as an abort by the receiver. The receiver responds to a received abort as follows:

1) An abort in an "out of frame" condition – An abort during the idle or time fill has no meaning. The abort reception is indicated in the status register as long as the abort condition continues; but neither an interrupt nor a stored condition occurs. The abort indication disappears after 15 or more consecutive 1's are received (Received Idle status is set.)
2) An abort "in frame" after less than 26 bits are received after an opening flag – Under this condition, any field of the aborted frame has not transferred to the MPU yet. The ADLC clears the aborted frame data in the FIFO and clears flag synchronization. Neither an interrupt nor a stored status occurs. The status indication is the same as (1) above.
3) An abort "in frame" after 26 bits or more are received after an opening flag – Under this condition, some fields of the aborted frame might have been transferred onto the data bus. The abort status is stored in the receiver status register and the data of the aborted frame in the ADLC is cleared. The synchronization is also cleared.

Idle and Time Fill – When the transmitter is in an "out of frame" condition (the transmitter is not transmitting a frame), it is in an idle state. Either a series of contiguous flags (time fill) or a mark idle (consecutive 1's on a bit by bit basis) is selected for the transmission in an idle state by the Flag/Mark Idle control bit. When the receiver receives 15 or more consecutive 1's, the Receive Idle status bit is set and causes an interrupt. The flags and mark idle are not transferred to the Rx FIFO.

OPERATION

INITIALIZATION–During a power-on sequence, the ADLC is reset via the $\overline{\text{RESET}}$ input and internally latched in a reset condition to prevent erroneous output transitions. The four control registers must be programmed prior to the release of the reset condition. The release of the reset condition is performed via software by writing a "0" into the Rx RS control bit (receiver) and/or Tx RS control bit (transmitter). The release of the reset condition must be done after the $\overline{\text{RESET}}$ input has gone high.

At any time during operation, writing a "1" into the Rx RS control bit or Tx RS control bit causes the reset condition of the receiver or the transmitter.

TRANSMITTER OPERATION–The Tx FIFO register cannot be pre-loaded when the transmitter is in a reset state. After the reset release, the Flag/Mark Idle control bit selects either the mark idle state (inactive idle) or the Flag "time fill" (active idle) state. This active or inactive mark idle state will continue until data is loaded into the Tx FIFO.

FIGURE 8A – ADLC TRANSMITTER STATE DIAGRAM
(C_ib_i refers to control register bit)

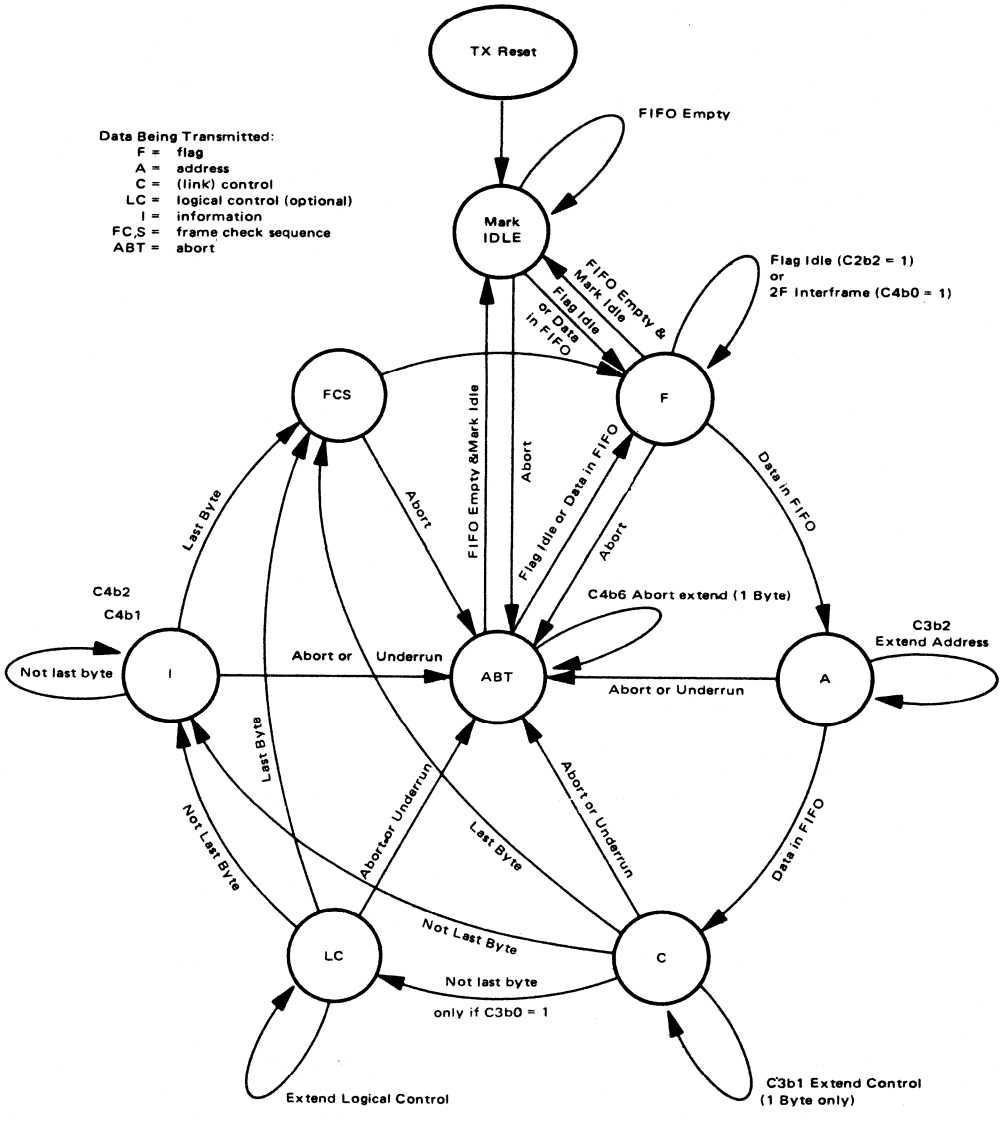
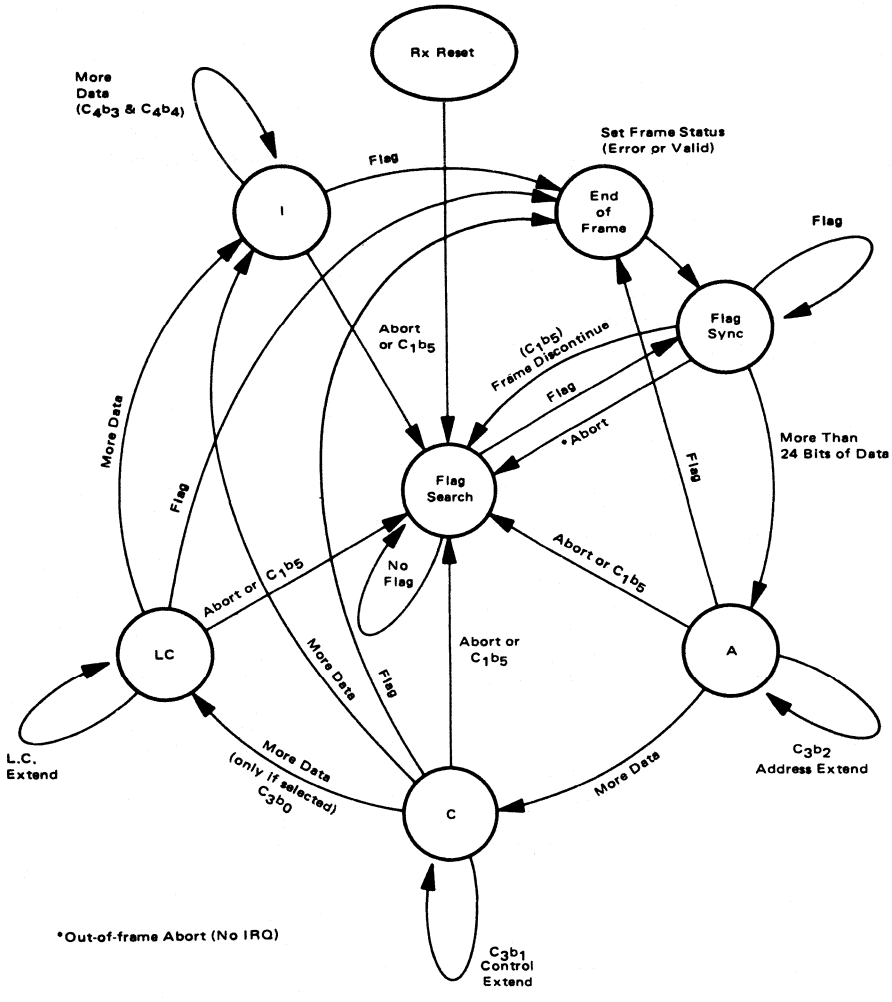


FIGURE 8B – ADLC RECEIVER STATE DIAGRAM



The availability of the Tx FIFO is indicated by the TDRA status bit under the control of the 2Byte/1Byte control bit. TDRA status is inhibited by the Tx RS bit or $\overline{CTS}$ input being high. When the 1 Byte mode is selected, one byte of the FIFO is available for data transfer when TDRA goes high. When the 2 Byte mode is selected, two successive bytes can be transferred when TDRA goes high.

The first byte (Address field) should be written into the Tx FIFO at the "Frame Continue" address. Then the transmission of a frame automatically starts. If the transmitter is in a mark idle state, the transfer of an address causes an opening flag within two or three transmitter clock cycles. If the transmitter has been in a time fill state, the current time fill flag being transmitted is assumed as an opening flag and the address field will follow it.

A frame continues as long as data is written into the Tx FIFO at the "Frame Continue" address. The ADLC internally keeps track of the field sequence in the frame. The frame format is described in the "FRAME FORMAT" section.

The frame is terminated by one of two methods. The most efficient way to terminate the frames from a software standpoint is to write the last data character into the Transmit FIFO "Frame Terminate" address (RS1, RS0 = 11) rather than the Transmit FIFO "Frame Continue" address (RS1, RS0 = 10). An alternate method is to follow the last write of data in the Tx FIFO "Frame Continue" address with the setting of the Transmit Last Data control bit. Either method causes the last character to be transmitted and the FCS field to automatically be appended along with a closing flag. Data for a new frame can be loaded into the Tx FIFO immediately after the old frame data, if TDRA is high. The closing Flag can serve as the opening Flag of the next frame or separate opening and closing Flags may be transmitted. If a new frame is not ready to be transmitted, the ADLC will automatically transmit the Active (Flag) or Inactive (Mark) Idle condition.

If the Tx FIFO becomes empty at any time during frame transmission (the FIFO has no data to transfer into transmitter shift register during transmission of the last half of the next to last bit of a word), an underrun will occur and the transmitter automatically terminates the frame by transmitting an abort. The underrun state is indicated by the Tx Underrun status bit.

Any time the Tx ABORT Control bit is set, the transmitter immediately aborts the frame (transmits at least 8 consecutive 1's) and clears the Tx FIFO. If the Abort Extend Control bit is set at the time, an idle (at least 16 consecutive 1's) is transmitted. An abort or idle in an "out of frame" condition can be useful to gain 8 or 16 bits of delay. (For an example, see "Programming Considerations.")

The $\overline{CTS}$ ($\overline{Clear\text{-}To\text{-}Send}$) input and $\overline{RTS}$ ($\overline{Request\text{-}To\text{-}Send}$) output are provided for a MODEM or other hardware interface.

The TDRA/FC status bit (when selected to be Frame Complete Status) can cause an interrupt upon frame completion (i.e., a flag or abort completion).

Details regarding the inputs and outputs, status bits, control bits, and FIFO operation are described in their respective sections.

RECEIVER OPERATION — Data and a pre-synchronized clock are provided to the ADLC receiver section by means of the Receive Data (RxD) and Receive Clock (RxC) inputs. The data is a continuous stream of binary bits with the characteristic that a maximum of five "1's" can occur in succession unless Abort, Flag, or Idling condition occurs. The receiver continuously (on a bit by bit basis) searches for Flags and Aborts.

When a flag is detected, the receiver establishes frame synchronization to the flag timing. If a series of flags is received, the receiver resynchronizes to each flag.

If the frame is terminated before the internal buffer time expires (the frame data is less than 25 bits after an opening flag), the frame is simply ignored. Noise on the data input (RxD) during time fill can cause this kind of invalid frame.

Once synchronization has been achieved and the internal buffer time (24 bit times) expires data will automatically transfer to the Rx Data FIFO. The Rx Data FIFO is clocked by E to cause received data to move through the FIFO to the last empty register location. The Receiver Data Available status bit (RDA) indicates when data is present in the last register (Reg #3) for the 1 Byte Transfer Mode. The 2 Byte Transfer Mode causes the RDA status bit to indicate data is available when the last two FIFO register locations (Reg #2 and #3) are full. If the data character present in the FIFO is an address octet the status register will exhibit an Address Present status condition. Data being available in the Rx Data FIFO causes an interrupt to be initiated (assuming the receiver interrupt is enabled, RIE="1"). The MPU will read the ADLC Status Register as a result of the interrupt or in its turn in a polling sequence. RDA or Address Present will indicate that receiver data is available and the MPU should subsequently read the Rx Data FIFO register. The interrupt and status bit will then be reset automatically. If more than one character had been received and was resident in the Rx Data FIFO, subsequent E clocks will cause the FIFO to update and the RDA status bit and interrupt will again be SET. In the two byte transfer mode both data bytes may be read on consecutive E cycles. Address Present provides for 1 byte transfers only.

The sequence of each field in the received frame is automatically handled by the ADLC. The frame format is described in the "FRAME FORMAT" section.

When a closing flag is received, the frame is terminated. The 16 bits preceding the closing flag are regarded as the FCS and are not transferred to the MPU. Whatever data is present in the most significant byte portion of the receiver buffer register is right justified and transferred to the Rx FIFO. The frame boundary pointer, which is explained in the "Rx FIFO REGISTER" section, is set simultaneously in the Rx FIFO. The frame boundary pointer sets the Frame Valid status bit (when the frame was completed with no error) or the FCS/IF Error Status bit (when the frame was completed with error) when the last byte of the frame appears at the last location of the Rx FIFO. As long as the Frame Valid or FCS/IF Error status bit is set, the data transfer from the second location of the Rx FIFO to the last location of the Rx FIFO is inhibited.

Any time the Frame Discontinue control bit is set, the ADLC discards the current frame data in the ADLC without dropping flag synchronization. This feature can be used to ignore a frame which is addressed to another station.

The reception of an abort or idle is explained in the "FRAME FORMAT" section. The details regarding the inputs, outputs, status bits, control bits, and Rx FIFO operation are described in their respective sections.

LOOP MODE OPERATION – The ADLC in the loop mode not only performs the transmission and receiving of data frames in the manner previously described but also has additional features for gaining and relinquishing loop control. In Figure 9a, a configuration is shown which depicts loop mode operation. The system configuration shows a primary station and several secondary stations. The loop is always under control of the primary station. When the primary wants to receive data, it transmits a Poll sequence and allows frame transmission to secondary stations on the loop. Each secondary is in series and adds one bit of delay to the loop. Secondary A in the figure receives data from the primary via its Rx Data Input, delays the data 1 bit, and transmits it to secondary B via its Tx Data Output. Secondaries B, C, and D operate in a similar manner. Therefore, data passes through each secondary and is received back by the primary controller.

FIGURE 9A – TYPICAL LOOP CONFIGURATION

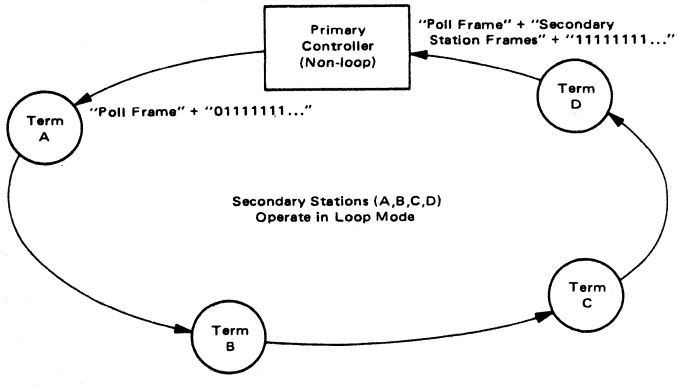
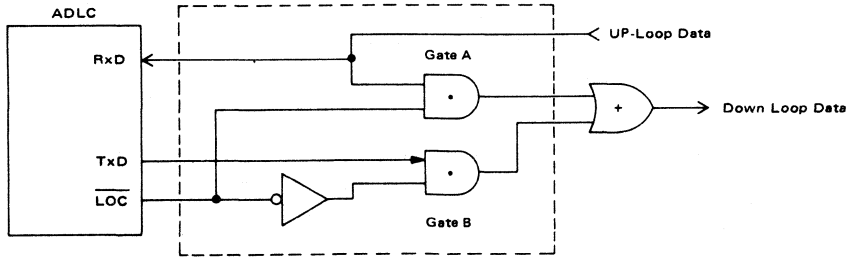


FIGURE 9B – EXAMPLE OF EXTERNAL LOOP LOGIC



Certain protocol rules must be followed in the manner by which the secondary station places itself on-loop (connects its transmitter output to the loop), goes active on the loop (starts transmitting its own station's data on the loop), and goes off the loop (disconnects its transmitter output). Otherwise loop data to other stations down loop would be interfered. The data stream always flows the same way and the order in which secondary terminals are serviced is determined by the hardware configuration. The primary controller times the delay through the loop. Should it exceed n + 1 bit times, where n is the number of secondary terminals on the loop, it will indicate a loop failure. Control is transferred to a secondary by transmitting a "Go Ahead" signal following the closing Flag of a polling frame (request for a response from the secondary) from the primary station. The "Go Ahead" from the primary is a "0" and 7 "1's" followed by mark idling. The primary can abort its response request by interrupting its idle with flags. The secondary should immediately stop transmission and return control back to the primary. When the secondary completes its frame, a closing flag is transmitted followed by all "1's". The primary detects the final 01111111 . . . ("Go Ahead" to the primary) and control is given back to the primary. Note that, if a down-loop secondary (e.g., station D) needs to insert information following an up-loop station (e.g., station A), the go ahead to station D is the last "0" of the closing flag from station A followed by "1's".

The ADLC in the primary station should operate in a non-loop, full duplex mode. The ADLC in the secondaries should operate in a loop mode, monitoring up-loop data on its receiver data input. The ADLC can recognize the necessary sequences in the data stream to automatically go on/off the loop and to insert its own station data. The procedure is the following and is summarized in Table 1. (1) Go On-loop – when the ADLC powers up, the terminal station will be off line. The first task is to become an active terminal on the loop. The ADLC must be connected to a Loop Link via an external switch as shown in Figure 9b. After a hardware reset, the ADLC $\overline{LOC}/\overline{DTR}$ Output will be in the high state and the up-loop receive data repeated through gate A to the down Loop stations. Any Up-Loop transmission will be received by the ADLC. The Loop Mode/Non-Loop Mode Control bit (bit 5 in Control Register 3) must be set to place the ADLC in the Loop Mode. The ADLC now monitors its Rx Data input for a string of seven consecutive "1's" which will allow a station to go on line. The Loop operation may be monitored by use of the Loop Status bit in Status Register 1. After power up and reset, this bit is a zero. When seven consecutive ones are received by the ADLC the $\overline{LOC}/\overline{DTR}$ output will go to a low level, disabling gate A (refer to Figure 9b), enabling gate B and connecting the ADLC Tx Data output to the down Loop stations. The up Loop data is now repeated to the down Loop stations via the ADLC. A one bit delay is inserted in the data (in NRZI mode, there will be a 2 bit delay) as it circulates through the ADLC. The ADLC is now on-line and the Loop Status bit in Status Register 1 will be at a one.

TABLE 1 – SUMMARY OF LOOP MODE OPERATION

STATE	RX SECTION	TX SECTION	LOOP STATUS BIT
OFF-LOOP	Rx section receives data from loop and searches for 7 "1's" (when On-Loop Control bit set) to go ON-LOOP.	Inactive 1) NRZ MODE. Tx data output is maintained "high" (mark). 2) NRZI MODE. Tx data output reflects the Rx data input state delayed by one bit time. (Not normally connected to loop.) The NRZI data is internally decoded to provide error-free transitions to On-Loop mode.	"0"
ON-LOOP	1) When Go-Active on poll bit is set, Rx section searches for 01111111 pattern (the EOP or 'Go Ahead') to become the active terminal on the loop. 2) When On-Loop control bit is reset, Rx section searches for 8 "1's" to go OFF-Loop.	Inactive 1) NRZ MODE. Tx data output reflects Rx data input state delayed one bit time. 2) NRZI MODE. Tx data output reflects Rx data input state delayed 2 bit times.	"1"
ACTIVE	Rx section searches for flag (an interrupt from the loop controller) at Rx data input. Received flag causes $\overline{FD}$ output to go low. IRQ is generated if $\overline{RIE}$ and FDSE control bits are set.	Tx data originates within ADLC until Go Active on Poll bit is reset and a flag or Abort is completed. Then returns to ON-Loop state.	"0"

(2) Go Active after Poll – The receiver section will monitor the up link data for a general or addressed poll command and the Tx FIFO should be loaded with data so that when the go-ahead sequence of a zero followed by seven ones (01111111---) is detected, transmission can be initiated immediately. When the polling frame is detected, the Go-Active-On-Poll control bit must be set (bit 6 in Control Register 3). A minimum of seven bit times are available to set this control bit after the closing flag of the poll. When the Go-Ahead is detected by the receiver, the ADLC will automatically change the seventh one to a zero so that the repeated sequence out gate B in Figure 9b is now an opening flag sequence (01111110). Transmission now continues from the Tx FIFO with data (address, control, etc.) as previously described. When the ADLC has gone active-on-poll, the Loop Status bit in Status Register 1 will go to a zero. The receiver searches for a flag, which indicates that the primary station is interrupting the current operation.

(3) Go Inactive when On-Loop – The Go-Active-On-Poll control bit may be RESET at any time during transmission. When the frame is complete (the closing Flag or abort is transmitted), the Loop is automatically released and the station reverts back to being just a one bit delay in the Loop, repeating up link data. If the Go-Active-On-Poll control bit is not reset by software and the final frame is transmitted (Flag/Mark Idle bit = 0), then the transmitter will mark idle and will not release the loop to up-loop data. A Tx Abort command would have to be used in this case in order to go inactive when on the loop. Also, if the Tx FIFO was not preloaded with data (address, control, etc.) prior to changing the "Go Ahead Character" to a Flag, the ADLC will either transmit flags (active idle character) until data is loaded (when Flag/Mark Idle Control bit is high) or will go into an underrun condition and transmit an Abort (when Flag/Mark Idle control bit is low). When an abort is transmitted, the Go-Active-on-Poll control bit is reset automatically and the ADLC reverts to its repeating mode, (TxD = delayed RxD). When the ADLC transmitter lets go of the loop, the Loop Status bit will return to a "1", indicating normal on-loop retransmission of up-loop data.

(4) Go Off-Loop – The ADLC can drop-off the Loop (go off-line) similar to the way it went on-line. When the Loop On-Line control bit is reset the ADLC receiver section looks for 8 successive "1's" before allowing the $\overline{LOC}/\overline{DTR}$ output to return high (the inactive state). Gate A in Figure 9b will be enabled and gate B disabled allowing the loop to maintain continuity without disturbance. The Loop Status bit will show an off-line condition (logical zero).

INPUT/OUTPUT FUNCTIONS

All inputs of ADLC are high impedance and TTL compatible level inputs. All outputs of the ADLC are compatible with standard TTL. Interrupt Request ($\overline{IRQ}$), however, is an open drain output (no internal pull-up).

INTERFACE FOR MPU

D0-D7 – Bidirectional Data Bus – These data bus I/O ports allow the data transfer between ADLC and system bus. The data bus drivers are three-state devices that remain in the high impedance (off) state except when the MPU performs an ADLC read operation.

E – Enable Clock – E activates the address inputs ($\overline{CS}$, RS0 and RS1) and R/W input and enables the data transfer on the data bus. E also moves data through the Tx FIFO and Rx FIFO. E should be a free running clock such as the MC6800 MPU system clock.

$\overline{CS}$ – Chip Select – An ADLC read or write operation is enabled only when the $\overline{CS}$ input is low and the E clock input is high. ($E \cdot \overline{CS}$).

RS0, RS1 – Register Selects – When the Register Select inputs are enabled by ($E \cdot \overline{CS}$), they select internal registers in conjunction with the Read/Write input and Address Control bit (control register 1, bit 0). Register addressing is defined in table 2.

R/$\overline{W}$ – Read/Write Control Line – The R/$\overline{W}$ input controls the direction of data flow on the data bus when it is enabled by ($E \cdot \overline{CS}$). When R/$\overline{W}$ is high, the I/O Buffer acts as an output driver and as an input buffer when low. It also selects the Read Only and Write Only registers within the ADLC.

$\overline{RESET}$ – Reset Input – The $\overline{RESET}$ Input provides a means of resetting the ADLC from a hardware source. In the "low state," the $\overline{RESET}$ Input causes the following:

* Rx Reset and Tx Reset are SET causing both the Receiver and Transmitter sections to be held in a reset condition.
* Resets the following control bits: Transmit Abort, RTS, Loop Mode, and Loop On-Line/DTR.
* Clears all stored status condition of the status registers.
* Outputs: $\overline{RTS}$ and $\overline{LOC}/\overline{DTR}$ go high. TxD goes to the mark state ("1's" are transmitted.

When $\overline{RESET}$ returns "high" (the inactive state) the transmitter and receiver sections will remain in the reset state until Tx Reset

and Rx Reset are cleared via the data bus under software control. The Control Register bits affected by $\overline{\text{RESET}}$ cannot be changed when $\overline{\text{RESET}}$ is "low".

$\overline{\text{IRQ}}$ — Interrupt Request Output – $\overline{\text{IRQ}}$ will be low if an interrupt situation exists and the appropriate interrupt enable has been set. The interrupt remains as long as the cause for the interrupt is present and the enable is set.

CLOCK AND DATA OF TRANSMITTER AND RECEIVER

TxC — Transmitter Clock Input – The transmitter shifts data on the negative transition of the TxC clock input. When the Loop Mode or Test Mode is selected, TxC should be the the same frequency and phase as the RxC clock. The data rate of the transmitter should not exceed the E frequency.

RxC — Receiver Clock Input – The receiver samples the data on the positive transition of the TxC clock. RxC should be synchronized with receive data externally.

TxD — Transmit Data Output – The serial data from the transmitter is coded in NRZ or NRZI (Zero Compliment) data format.

RxD — Receiver Data Input – The serial data to be received by the ADLC can be coded in NRZ or NRZI (Zero Complement) data format. The data rate of the receiver should not exceed the E frequency. If a partial byte reception is possible at the end of a frame, the maximum data rate of the receiver is indicated by the following relationship:

$$f_{RxC} \leq \frac{1}{2t_E + 300\ ns}$$

where t_E is the period of E.

PERIPHERAL/MODEM CONTROL

$\overline{\text{RTS}}$ — Request to Send Output – The Request-to-Send output is controlled by the Request-to-Send control bit in conjunction with the state of the transmitter section. When the RTS bit goes high, the $\overline{\text{RTS}}$ output is forced low. When the RTS bit returns low, the $\overline{\text{RTS}}$ output remains low until the end of the frame and there is no further data in the Tx FIFO for a new frame. The positive transition of $\overline{\text{RTS}}$ occurs after the completion of a Flag, an Abort, or when the RTS control bit is reset during a mark idling state. When the $\overline{\text{RESET}}$ input is low, the $\overline{\text{RTS}}$ output goes high.

$\overline{\text{CTS}}$ — Clear to Send Input – The $\overline{\text{CTS}}$ input provides a real-time inhibit to the TDRA status bit and its associated interrupt. The positive transition of $\overline{\text{CTS}}$ is stored within the ADLC to insure its occurrence will be acknowledged by the system. The stored CTS information and its associated IRQ (if enabled) are cleared by writing a "1" in the Clear Tx Status bit or in the Transmitter Reset bit.

$\overline{\text{DCD}}$ — Data Carrier Detect Input – The $\overline{\text{DCD}}$ input provides a real-time inhibit to the receiver section. A high level on the $\overline{\text{DCD}}$ input resets and inhibits the receiver register, but data in the Rx FIFO from a previous frame is not disturbed. The positive transition of $\overline{\text{DCD}}$ is stored within the ADLC to insure that its occurrence will be acknowledged by the system. The stored DCD information and its associated IRQ (if enabled) are cleared by means of the Clear Rx Status Control bit or by the Rx Reset bit.

$\overline{\text{LOC/DTR}}$ — Loop On Line Control/Data Terminal Ready output – The $\overline{\text{LOC/DTR}}$ output serves as a $\overline{\text{DTR}}$ output in the non-loop mode or as a Loop Control output in the loop mode. When $\overline{\text{LOC/DTR}}$ output performs the $\overline{\text{DTR}}$ function, it is turned on and off by means of the LOC/DTR control bit. When the LOC/DTR control bit is high the $\overline{\text{DTR}}$ output will be low. In the loop mode the $\overline{\text{LOC/DTR}}$ output provides the means of controlling the external loop interface hardware to go On-line or Off-line. When the LOC/DTR control bit is SET and the loop has "idled" for 7 bit times or more (RxD = 01111111...), the $\overline{\text{LOC/DTR}}$ output will go low (on-line). When the LOC/DTR control bit is low and the loop has "idled" for 8 bit times or more, the $\overline{\text{LOC/DTR}}$ output will return high (off-line). The $\overline{\text{RESET}}$ input being low will cause the $\overline{\text{LOC/DTR}}$ output to be high.

$\overline{\text{FD}}$ — Flag Detect Output – An output to indicate the reception of a flag and initiate an external time-out counter for the loop mode operation. The $\overline{\text{FD}}$ output goes low for one bit time beginning at the last bit of the flag character, as sampled by the receiver clock (RxC).

DMA INTERFACE

RDSR — Receiver Data Service Request Output – The RDSR Output is provided primarily for use in DMA Mode operation and indicates (when high) that the Rx FIFO requests service (RSDR output reflects the RDA status bit). If the prioritized Status Mode is selected,

RDSR will be inhibited when any other receiver status conditions are present. RDSR goes low when the Rx FIFO is read.

TDSR — Transmitter Data Service Request Output — The TDSR Output is provided for DMA mode operation and indicates (when high) that the Tx FIFO requests service. TDSR goes low when the TX FIFO is loaded. TDSR is inhibited by: The Tx Rs control bit being SET, $\overline{RESET}$ being low, or $\overline{CTS}$ being high. If the prioritized status mode is used, Tx underrun also inhibits TDSR. TDSR reflects the TDRA status bit except in the F/C mode.

ADLC REGISTERS

Eight registers in the ADLC can be accessed by means of the MPU data and address buses. The registers are defined as read only or write only according to the direction of information flow. The addresses of these registers are defined in Table 2. The transmitter FIFO register can be accessed by two different addresses, the "Frame Terminate" address and the "Frame Continue" address. (The function of these addresses are discussed in the FIFO section.)

TABLE 2 – REGISTER ADDRESSING

Register Selected	R/W	RS1	RS0	Address Control Bit (C_1b_0)
Write Control Register #1	0	0	0	X
Write Control Register #2	0	0	1	0
Write Control Register #3	0	0	1	1
Write Transmit FIFO (Frame Continue)	0	1	0	X
Write Transmit FIFO (Frame Terminate)	0	1	1	0
Write Control Register #4	0	1	1	1
Read Status Register #1	1	0	0	X
Read Status Register #2	1	0	1	X
Read Receiver FIFO	1	1	X	X

RECEIVER DATA FIRST-IN FIRST-OUT REGISTER

Rx FIFO — The Rx FIFO consists of three 8-bit registers which are used for the buffer storage of received data. Data bytes are always transferred from a full register to an adjacent empty register; and both phases of the E input clock are used for the data transfer. Each register has pointer bits which point the frame boundary. When these pointers appear at the last FIFO location, they update the Address Present, Frame Valid or FCS/IF Error status bits.

The RDA status bit indicates the state of the Rx FIFO. When RDA status bit is "1", the Rx FIFO is ready to be read. The RDA status is controlled by the 2 Byte/1 Byte control bit. When overrun occurs, the data in the first byte of the Rx FIFO are no longer valid.

Both the Rx Reset bit and $\overline{Reset}$ input clear the Rx FIFO. Abort ("in Frame") and a high level on the $\overline{DCD}$ input also clears the Rx FIFO, but the last bytes of the previous frame, which are separated by the frame boundary pointer, are not disturbed.

TRANSMITTER DATA FIRST-IN FIRST-OUT REGISTER

Tx FIFO — The Tx FIFO consists of three 8-bit registers which are used for buffer storage of data to be transmitted. Data is always transferred from a full register to an empty adjacent register; the transfer occurs on both phases of the E input clock. The Tx FIFO can be addressed by two different register addresses, the "Frame Continue" address and the "Frame Terminate" address. Each register has pointer bits which point to the frame boundary. When a data byte is written at the "Frame Continue" address, the pointer of the first FIFO register is set. When a data byte is written at the "Frame Terminate" address, the pointer of the first FIFO register is reset. RxRs control bit or Tx Abort control bit resets all pointers. The pointer will shift through the FIFO. When a positive transition is detected at the third location of FIFO, the transmitter initiates a frame with an open flag. When the negative transition is detected at the third location of FIFO, the transmitter closes a frame, appending the FCS and closing Flag to the last byte.

The Tx last control bit can be used instead of using the "Frame Terminate" address. When the Tx last control bit is set with a "1", the logic searches the last byte location in the FIFO and resets the pointer in the FIFO register.

The status of Tx FIFO is indicated by the TDRA status bit. When TDRA is "1", the Tx FIFO is available for loading data. The TDRA status is controlled by the 2BYTE/1BYTE control bit. The Tx FIFO is reset by both Tx Reset and $\overline{RESET}$ input. During this reset condition or when $\overline{CTS}$ input is high, the TDRA status bit is suppressed and data loading is inhibited.

ADLC INTERNAL REGISTER STRUCTURE

	Bit #	RS1 RS0 = 00 Status Register #1	RS1 RS0 = 01 Status Register #2	RS1 RS0 = 10 Receiver Data Register	RS1 RS0 = 11
Read Only Registers	0	RDA	Address Present	Bit 0	Same as RS1, RS0 = 10
	1	Status #2 Read Request	Frame Valid	Bit 1	
	2	Loop	Inactive Idle Received	Bit 2	
	3	Flag Detected (When Enabled)	Abort Received	Bit 3	
	4	$\overline{CTS}$	FCS Error	Bit 4	
	5	Tx Underrun	$\overline{DCD}$	Bit 5	
	6	TDRA/Frame Complete	Rx Overrun	Bit 6	
	7	IRQ Present	RDA (Receiver Data Available)	Bit 7	

	Bit #	Control Register #1	Control Register #2 ($C_1b_0 = 0$)	Control Register #3 ($C_1b_0 = 1$)	Transmitter Data (Continue Data)	Transmitter Data (Last Data) ($C_1b_0 = 0$)	Control Register #4 ($C_1b_0 = 1$)
Write Only Registers	0	Address Control (AC)	Prioritized Status Enable	Logical Control Field Select	Bit 0	Bit 0	Double Flag/Single Flag Interframe Control
	1	Receiver Interrupt Enable (RIE)	2 Byte/1 Byte Transfer	Extended Control Field Select	Bit 1	Bit 1	Word Length Select Transmit #1
	2	Transmitter Interrupt Enable (TIE)	Flag/Mark Idle	Auto, Address Extension Mode	Bit 2	Bit 2	Word Length Select Transmit #2
	3	RDSR Mode (DMA)	Frame Complete/ TDRA Select	01/11 Idle	Bit 3	Bit 3	Word Length Select Receive #1
	4	TDSR Mode (DMA)	Transmit Last Data	Flag Detected Status Enable	Bit 4	Bit 4	Word Length Select Receive #2
	5	Rx Frame Discontinue	CLR Rx Status	Loop/Non-Loop Mode	Bit 5	Bit 5	Transmit Abort
	6	Rx RESET	CLR Tx Status	Go Active on Poll/Test	Bit 6	Bit 6	Abort Extend
	7	Tx RESET	RTS Control	Loop On-Line Control DTR	Bit 7	Bit 7	NRZI/NRZ

CONTROL REGISTERS

CONTROL REGISTER 1 (CR1)

CONTROL REGISTER 1 (CR1)

				7	6	5	4	3	2	1	0
RS1	RS0	R/W	AC	TxRS	RxRS	Discontinue	TDSR Mode	RDSR Mode	TIE	RIE	AC
0	0	0	X								

b_0 Address Control (AC) – AC provides another RS (Register Select) signal internally. The AC bit is used in conjunction with RS0, RS1 and R/W inputs to select particular registers, as shown in Table 2.

b_1 Receiver Interrupt Enable (RIE) – RIE enables/disables the interrupt request caused by the receiver section. 1...enable, 0...disable.

b_2 Transmitter Interrupt Enable (TIE) – TIE enables/disables the interrupt request caused by the transmitter. 1...enable, 0...disable.

b_3 Receiver Data Service Request Mode (RDSR MODE) – The RDSR MODE bit provides the capability of operation with a bus system in the DMA mode when used in conjunction with the prioritized status mode. When RDSR MODE is set, an interrupt request caused by RDA status is inhibited, and the ADLC does not request data transfer via the $\overline{IRQ}$ output.

b_4 Transmitter Data Service Request Mode (TDSR MODE) – The TDSR MODE bit provides the capability of operation with a bus system in the DMA mode when used in conjunction with the prioritized status mode. When TDSR MODE is set, an interrupt request caused by TDRA status is inhibited, and the ADLC does not request a data transfer via the $\overline{IRQ}$ output.

b_5 Rx Frame Discontinue (DISCONTINUE) – When the DISCONTINUE bit is set, the currently received frame is ignored and the ADLC discards the data of the current frame. The DISCONTINUE bit is automatically reset when the last byte of the frame is discarded. When the ignored frame is aborted by receiving an Abort or DCD failure, the DISCONTINUE bit is also reset.

b_6 Receiver Reset (Rx Rs) – When the Rx Rs bit is "1", the receiver section stays in the reset condition. All receiver sections including the Rx FIFO register and the receiver status bits in both status registers, are reset. (During reset, the stored DCD status is reset but the DCD status bit follows the $\overline{DCD}$ input.) Rx Rs is set by forcing a low level on the $\overline{RESET}$ input or by writing a "1" into this bit from the data bus. Rx Rs must be reset by writing a "0" from the data bus after $\overline{RESET}$ has gone high.

b_7 Transmitter Reset (Tx Rs) – when the Tx Rs bit is "1", the transmitter section stays in the reset condition and transmits marks ("1's"). All transmitter sections, including the Tx FIFO and the transmitter status bits, are reset (FIFO cannot be loaded). During reset, the stored CTS status is reset but the CTS status bit follows the $\overline{CTS}$ input. Tx Rs is set by forcing a low level on the $\overline{RESET}$ input or by writing a "1" from the data bus. It must be reset by writing a "0" after $\overline{RESET}$ has gone high.

CONTROL REGISTER 2 (CR2)

CONTROL REGISTER 2 (CR2)

				7	6	5	4	3	2	1	0
RS1	RS0	R/W	AC	RTS	CLR TxST	CLR RxST	Tx Last	FC/TDRA Select	F/M Idle	2/1 Byte	PSE
0	1	0	0								

b_0 Prioritized Status Enable (PSE) – When the PSE bit is SET, the status bits in both status registers are prioritized as defined in the Status Register section. When PSE is low, the status bits indicate current status without bit suppression by other status bits. The exception to this rule is the $\overline{CTS}$ status bit which always suppresses the TDRA status.

b_1 2 Byte/1 Byte Transfer (2/1 Byte) – When the 2/1 Byte bit is RESET the TDRA and RDA status bits then will indicate the availability of their respective data FIFO registers for a single byte data transfer. Similarly, if 2/1 Byte is set, the TDRA and RDA status bits indicate when two bytes of data can be moved without a second status read.

b_2 Flag/Mark Idle Select (F/M Idle) – The F/M Idle bit selects Flag characters or bit by bit Mark Idle for the time fill or the idle state of the transmitter. When Mark Idle is selected, Go-Ahead code can be generated for loop operation in conjunction with the 01/11 Idle control bit (C_3b_3). 1 . . . Flag time fill, 0 . . . Mark Idle.

b_3 Frame Complete/TDRA Select (FC/TDRA Select) – The FC/TDRA Select bit selects TDRA status or FC status for the TDRA/FC status bit indication. 1 . . . FC status, 0 . . . TDRA status.

b_4 Transmit Last Data (Tx Last) – Tx Last bit provides another method to terminate a frame. When the Tx Last bit is set just after loading a data byte, the ADLC assumes the byte is the last byte and terminates the frame by appending CRCC and a closing Flag. This control bit is useful for DMA operation. Tx Last bit automatically returns to the "0" state.

b_5 Clear Receiver Status (CLR Rx ST) – When a "1" is written into the CLR Rx ST bit, a reset signal is generated for the receiver status bits in status register #1 and #2 (except AP and RDA bits). The reset signal is enabled only for the bits which have been present during the last "read status" operation. The CLR Rx ST bit automatically returns to the "0" state.

b_6 Clear Transmitter Status (CLR Tx ST) – When a "1" is written into CLR Tx ST bit, a reset signal is generated for the transmitter status bits in status register #1 (except TDRA). The reset signal is enabled for the bits whioh have been present during the last "read status" operation. The CLR Tx ST bit automatically returns to the "0" state.

b_7 Request to Send Control (RTS) – The RTS bit when high causes the $\overline{RTS}$ output to be low (the active state). When the RTS bit returns low and data is being transmitted, the $\overline{RTS}$ output remains low until the last character of the frame (the closing Flag or Abort) has been completed and the Tx FIFO is empty. If the transmitter is idling when the RTS bit returns low, the $\overline{RTS}$ output will go high (the inactive state) within two bit times.

CONTROL REGISTER 3 (CR3)

CONTROL REGISTER 3 (CR3)

				7	6	5	4	3	2	1	0
RS1	RS0	R/W	AC	LOC/	GAP/	Loop	FDSE	01/11	AEX	CEX	LCF
0	1	0	1	DTR	TST			Idle			

b_0 Logical Control Field Select (LCF) – The LCF select bit causes the first byte(s) of data belonging to the information field to remain 8 bit characters until the logical control field is complete. The logical control field (when selected) is an automatically extendable field which is extended when bit 7 of a logical control character is a "1". When the LCF Select bit is reset the ADLC assumes no logical control field is present for either the transmit or received data channels. When the logical control field is terminated, the word length of the information data is then defined by WLS_1 and WLS_2.

b_1 Extended Control Field Select (C_{EX}) – When the C_{EX} bit is a "1", the control field is extended and assumed to be 16 bits. When C_{EX} is "0", the control field is assumed to be 8 bits.

b_2 Auto/Address Extend Mode (A_{EX}) – The A_{EX} bit when "low" allows full 8 bits of the address octet to be utilized for addressing because address extension is inhibited. When the A_{EX} bit is "high", bit 0 of address octet equal to "0" causes the Address field to be extended by one octet. The exception to this automatic address field extension is when the first address octet is all "0's" (the Null Address).

b_3 01/11 Idle (01/11 Idle) – The 01/11 Idle Control bit determines whether the inactive (Mark) idle condition begins with a "0" or not. If the 01/11 Idle Control is SET, the closing flag (or Abort) will be followed by a 011111 . . . pattern. This is required of the controller for the "Go Ahead" character in the Loop Mode. When 01/11 is RESET, the idling condition will be all "1's".

b_4 Flag Detect Status Enable (FDSE) – The FDSE bit enables the FD status bit in Status Register #1 to indicate the occurrence of a received Flag character. The status indication will be accompanied by an interrupt if RIE is SET. Flag detection will cause the $\overline{\text{Flag Detect}}$ output to go low for one bit time regardless of the state of FDSE.

b_5 LOOP/NON-LOOP Mode (LOOP) – When the LOOP bit is set, loop mode operation is selected and the GAP/TST control bit, LOC/DTR control bit and $\overline{\text{LOC/DTR}}$ output are selected to perform the loop control functions. When LOOP is reset, the ADLC operates in the point to point data communications mode.

b_6 Go Active On Poll/Test (GAP/TST) – In the Loop Mode – The GAP/TST bit is used to respond to the poll sequence and to begin transmission. When GAP/TST is set, the receiver searches for the "Go Ahead" (or End of Poll, EOP). The receiver "Go Ahead" is converted to an opening Flag and the ADLC starts its own transmission. When GAP/TST is reset during the transmission, the end of the frame (the completion of Flag or Abort) causes the termination of the "go-active-on-poll" operation and the Rx Data to Tx Data link is reestablished. The ADLC then returns to the "loop-on-line" state.

In the Non-loop Mode – The GAP/TST bit is used for self-test purposes. If GAP/TST bit is set, the TxD output is connected to the RxD input internally, and provides a "loop-back" feature. For normal operation, the GAP/TST bit should be reset.

b_7 Loop On-Line Control/DTR Control (LOC/DTR) – In the Loop Mode – The LOC/DTR bit is used to go on-line or to go off-line. When LOC/DTR is set, the ADLC goes to the on-line state after 7 consecutive "1's" occur at the RxD input. When LOC/DTR is reset, the ADLC goes to the "off-line" state after eight consectuive "1's" occur at the RxD input.

In the Non-Loop Mode – The LOC/DTR bit directly controls the Loop On-Line/$\overline{\text{DTR}}$ output state. 1...$\overline{\text{DTR}}$ output goes to low level. 0...$\overline{\text{DTR}}$ output goes to high level.

CONTROL REGISTER 4 (CR4)

CONTROL REGISTER 4 (CR4)

				7	6	5	4	3	2	1	0
RS1	RS0	R/W	AC	NRZI/NRZ	ABT_{EX}	ABT	Rx		Tx		"FF"/F
1	1	0	1				WLS_2	WLS_1	WLS_2	WLS_1	

b_0 Double Flag/Single Flag Interframe Control ("FF"/"F") – The "FF"/"F" Control bit determines whether the transmitter will transmit separate closing and opening Flags when frames are transmitted successively. When the "FF"/"F" control bit is low, the closing flag of the first frame will serve as the opening flag of the second frame. When the bit is high, independent opening and closing flags will be transmitted.

b_1 Transmitter Word Length Select (Tx WLS1 & WLS2)

b_2 – Tx WLS1 and WLS2 are used to select the word length of the transmitter information field. The encoding format is shown in Table 3.

TABLE 3 – I-FIELD CHARACTER LENGTH SELECT

WLS_1	WLS_2	I-Field Character Length
0	0	5 bits
1	0	6 bits
0	1	7 bits
1	1	8 bits

b_3 Receiver Word Length Select (Rx WLS1 & WLS2) –

b_4 Rx WLS1 and WLS2 are used to select the word length of the receiver information field. The encoding format is shown in Table 3.

b_5 Transmit Abort (ABT) – The ABT bit causes an Abort (at least 8 bits of "1" in succession) to be transmitted. The Abort is initiated and the Tx FIFO is cleared when the control bit goes high. Once Abort begins, the Tx Abort control bit assumes the low state.

b_6 Abort Extend (ABT_{EX}) – If ABT_{EX} is set, the abort code initiated by ABT is extended up to at least 16 bits of consecutive "1's", the mark Idle State.

b_7 NRZI (Zero Complement)/NRZ Select (NRZI/NRZ) – NRZI/NRZ bit selects the transmit/receive data format to be NRZI or NRZ in both Loop Mode or Non-Loop mode operation. When the NRZI Mode is selected, a 1 bit delay is added to the transmitted data (TxD) to allow for NRZI encoding. 1...NRZI, 0...NRZ.

NOTE: NRZI coding – The serial data remains in the same state to send a binary "1" and switches to the opposite state to send a binary "0".

STATUS REGISTER

The Status Register #1 is the main status register. The IRQ bit indicates whether the ADLC requests service or not. The S2RQ bit indicates whether any bits in status register #2 request any service. TDRA and RDA, because they are most often used, are located in bit positions that are more convenient to test, RDA reflects the state of the RDA bit in status register #2.

The Status Register #2 provides the detailed status information contained in the S2RQ bit and these bits reflect receiver status. The FD bit is the only receiver status, which is not indicated in status register #2.

The prioritized status mode provides maximum efficiency in searching the status bits and indicates only the most important action required to service the ADLC. The priority trees of both status registers are provided in Figure 10.

Reading the status register is a non-destructive process. The method of clearing status depends upon the bit's function and is discussed for each bit in the register.

STATUS REGISTER 1 (SR1)

STATUS REGISTER 1 (SR1)

				7	6	5	4	3	2	1	0
RS1	RS0	R/W	AC	IRQ	TDRA/FC	TXU	CTS	FD	LOOP	S2RQ	RDA
0	0	1	X								

b_0 Receiver Data Available (RDA) – The RDA status bit reflects the state of the RDA status bit in status Register #2. It provides the means of achieving data transfers of received data in the full Duplex Mode without having to read both status registers.

b_1 Status Register #2 Read Request (S2RQ) – All the status bits (stored conditions) of status register #2 (except RDA bit) are logically ORed and indicated by the S2RQ status bit. Therefore S2RQ indicated that status register #2 needs to be read. When S2RQ is "0", it is not necessary to read status register #2. The bit is cleared when the appropriate bits in Status Register #2 are cleared or when Rx Reset is used.

b_2 Loop Status (LOOP) – The LOOP status bit is used to monitor the loop operation of the ADLC. This bit does not cause an IRQ. When Non-Loop Mode is selected, LOOP bit stays "0". When Loop Mode is selected, the LOOP status bit goes to "1" during "On-Loop" condition. When ADLC is in an "Off-Loop" condition or "Go-Active-On-Poll" condition, the LOOP status bit is a "0".

b_3 Flag Detected (FD) – The FD Status bit indicates that a flag has been received if the Flag Detect Enable control bit has been set. The bit goes high at the last bit of the Flag Character received (when the $\overline{\text{Flag Detect}}$ Output goes low) and is stored until cleared by Clear Rx Status or Rx Reset.

b_4 Clear To Send (CTS) – The $\overline{\text{CTS}}$ input positive transition is stored in the status register and causes an IRQ (if Enabled). The stored CTS condition and its IRQ are cleared by Clear Tx Status control bit or Tx Reset bit. After the stored status is reset, the CTS status bit reflects the state of the $\overline{\text{CTS}}$ input.

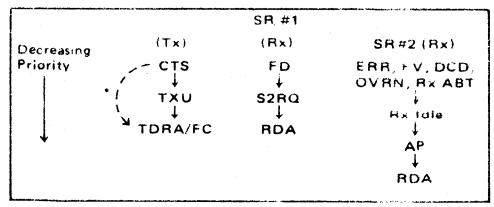
b_5 Transmitter Underrun (TxU) – When the transmitter runs out of data during a frame transmission, an underrun occurs and the frame is automatically terminated by transmitting an Abort. The underrun condition is indicated by the TxU status bit. TxU can be cleared by means of the Clear Tx Status Control bit or by Tx Reset.

b_6 Transmitter Data Register Available/Frame Complete (TDRA/FC) – The TDRA Status bit serves two purposes depending upon the state of the Frame Complete/TDRA Select control bit. When this bit serves as a TDRA status bit, it indicates that data (to be transmitted) can be loaded into the Tx Data FIFO register. The first register (Reg #1) of the Tx Data FIFO being empty (TDRA = "1") will be indicated by the TDRA Status bit in the "1-Byte Transfer Mode." The first two registers (Reg #1 and #2) must be empty for TDRA to be high when in the "2 Byte Transfer Mode." TDRA is inhibited by Tx Reset, or $\overline{\text{CTS}}$ being high.

When the Frame Complete Mode of operation is selected, the TDRA/FC status bit goes high when an abort is transmitted or when a flag is transmitted with no data in the Tx FIFO. The bit remains high until cleared by resetting the TDRA/FC control bit or setting the Tx Reset bit.

b_7 Interrupt Request (IRQ) – The Interrupt Request status bit indicates when the $\overline{\text{IRQ}}$ output is in the active state ($\overline{\text{IRQ}}$ Output = "0"). The IRQ status bit is subject to the same interrupt enables (RIE, TIE) as the $\overline{\text{IRQ}}$ output. The IRQ status bit simplifies status inquiries for polling systems by providing single bit indication of service requests.

FIGURE 10 – STATUS REGISTER PRIORITY TREE (PSE=1)



*Prioritized even when PSE = 0
NOTE: Status bit above will inhibit one below it.

STATUS REGISTER 2 (SR2)

STATUS REGISTER 2 (SR2)

				7	6	5	4	3	2	1	0
RS1	RS0	R/W	AC	RDA	OVRN	DCD	ERR	Rx ABT	Rx Idle	FV	AP
0	1	1	X								

b_0 Address Present (AP) – The AP status bit provides the frame boundary and indicates an Address octet is available in the Rx Data FIFO register. In the Extended Addressing Mode, the AP bit continues to indicate addresses until the Address field is complete. The Address present status bit is cleared by reading data or by Rx Reset.

b_1 Frame Valid (FV) – The FV status bit provides the frame boundary indication to the MPU and also indicates that a frame is complete with no error. The FV status bit is set when the last data byte of a frame is transferred into the last location of the Rx FIFO (available to be read by MPU). Once FV status is set, the ADLC stops further data transfer into the last location of the Rx FIFO (in order to prevent the mixing of two frames) until the status bit is cleared by the Clear Rx Status bit or Rx Reset.

b_2 Inactive Idle Received (Rx Idle) – The Rx Idle status bit indicates that a minimum of 15 consecutive "1's" have been received. The event is stored within the status register and can cause an interrupt. The interrupt and stored condition are cleared by the Clear Rx Status Control bit. The Status bit is the Logical OR of the receiver idling detector (which continues to reflect idling until a "0" is received) and the stored inactive idle condition.

b_3 Abort Received (RxABT) – The RxABT status bit indicates that 7 or more consecutive "1's" have been received. Abort has no meaning under out-of-frame conditions; therefore, no interrupt nor storing of the status will occur unless a Flag has been detected prior to the Abort. An Abort Received when "in frame" is stored in the status register and causes an IRQ. The status bit is the logical OR of the stored conditions and the Rx Abort detect logic, which is cleared after 15 consecutive "1's" have occurred. The stored Abort condition is cleared by the Clear Rx Status Control bit or Rx Reset.

b_4 Frame Check Sequence/Invalid Frame Error (ERR) – When a frame is complete with a cyclic redundancy check (CRC) error or a short frame error (the frame does not have complete Address and Control fields), the ERR status bit is set instead of the Frame Valid status bit. Other functions, frame boundary indication and control function, are exactly the same as for the Frame Valid status bit. Refer to the FV status bit.

b_5 Data Carrier Detect (DCD) – A positive transition on the $\overline{DCD}$ input is stored in the status register and causes an IRQ (if enabled). The stored DCD condition and its IRQ are cleared by the Clear Rx Status Control bit or RX Reset. After stored status is reset, the DCD status bit follows the state of the input. Both the stored DCD condition and the $\overline{DCD}$ input cause the reset of the receiver section when they are high.

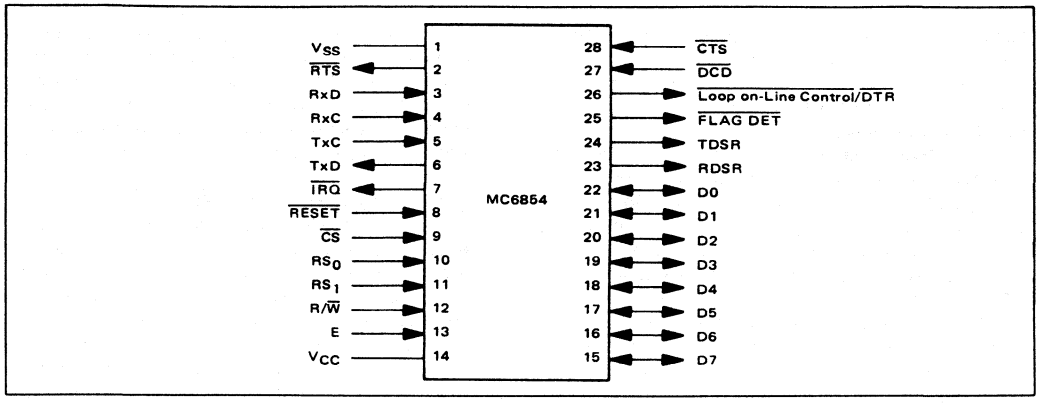
b_6 Receiver Overrun (OVRN) – OVRN status indicates that receiver data has been transferred into the Rx FIFO when it is full, resulting in data loss. The OVRN status is cleared by the Clear Rx Status bit or Rx Reset. Continued overrunning only destroys data in the first FIFO register.

b_7 Receiver Data Available (RDA) – The Receiver Data Available status bit indicates when receiver data can be read from the Rx Data FIFO. When the prioritized status mode is used, the RDA bit indicates that non-Address and non-last data are available in the Rx FIFO. The receiver data being present in the last register of the FIFO causes RDA to be high for the "1 Byte Transfer Mode". The RDA bit being high indicates that the last two registers are full when in the "2 Byte Transfer Mode". The RDA status bit is reset automatically when data is not available.

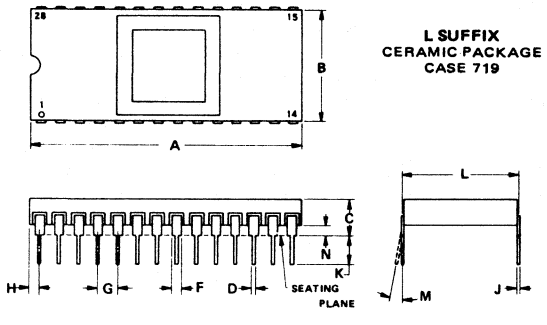
PROGRAMMING CONSIDERATIONS

1. **Status Priority** – When the prioritized status mode is used, it is best to test for the lowest priority conditions first. The lowest priority conditions typically occur more frequently and are the most likely conditions to exist when the processor is interrupted.
2. **Stored vs Present Status** – Certain status bits (DCD, CTS, Rx Abort, and Rx Idle) indicate a status which is the logical OR of a stored and a present condion. It is the stored status that causes an interrupt and which is cleared by a Status Clear control bit. After being cleared, the status register will reflect the present condition of an input or a receiver input sequence.
3. **Clearing Status Registers** – In order to clear an interrupt with the two Status Clear control bits, a particular status condition must be read before it can be cleared. In the prioritized mode, clearing a higher priority condition might result in another $\overline{IRQ}$ caused by a lower priority condition whose status was suppressed when a status register was first read. This guarantees that a status condition is never inadvertently cleared.
4. **Clearing the Rx FIFO** – An Rx Reset will effectively clear the contents of all 3 Rx FIFO bytes. However, the FIFO may contain data from 2 different frames when abort or DCD failure occurs. When this happens, the data from a previously closed frame (a frame whose closing flag has been received) will not be destroyed.
5. **Servicing the Rx FIFO in a 2 Byte Mode** – The procedure for reading the last bytes of data is the same, regardless of whether the frame contains an even or an odd number of bytes. Continue to read 2 bytes until an interrupt occurs that is caused by an end of frame status (FV or ERR). When this occurs, indicating the last byte either has been read or is ready to be read, switch temporarily to the 1-byte mode with no prioritized status (control register 2). Test RDA to indicate whether a 1-byte read should be performed. Then clear the frame end status.
6. **Frame Complete Status and $\overline{RTS}$ Release** – In many cases, a MODEM will require a delay for releasing $\overline{RTS}$. An 8-bit or 16-bit delay can be added to the ADLC $\overline{RTS}$ output by using an Abort. At the end of a transmission, frame complete status will indicate the frame completion. After frame complete status goes high, write "1" into the Abt control bit (and Abt Extend bit if a 16-bit delay is required). After the Abt control bit is set, write "0" into the RTS control bit. The transmitter will transmit eight or sixteen 1's and the $\overline{RTS}$ output will then go high (inactive).
7. **Note to users not using the MC6800** – (a) Care should be taken when performing a write followed by a read on successive E pulses at a high frequency rate. Time must be allowed for status changes to occur. If this is done, the time that E is low between successive write/read E pulses should be at least 500 ns. (b) The ADLC is a completely static part. However, the E frequency should be high enough to move data through the FIFOs and to service the peripheral requirements. Also, the period between successive E pulses should be less than the period of RxC or TxC in order to maintain synchronization between the data bus and the peripherals.

FIGURE 11 – I/O PIN DESCRIPTION



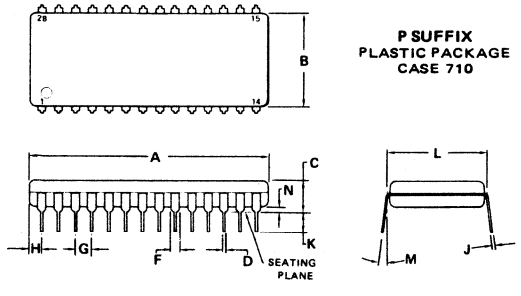
PACKAGE DIMENSIONS



L SUFFIX
CERAMIC PACKAGE
CASE 719

NOTE:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA (AT SEATING PLANE) AT MAXIMUM MATERIAL CONDITION.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	35.20	35.92	1.386	1.414
B	14.43	14.94	0.568	0.588
C	3.05	4.19	0.120	0.165
D	0.41	0.51	0.016	0.020
F	1.12	1.42	0.044	0.056
G	2.54 BSC		0.100 BSC	
H	1.12	1.42	0.044	0.056
J	0.23	0.28	0.009	0.011
K	2.54	4.19	0.100	0.165
L	14.83	15.14	0.584	0.596
M	0°	10°	0°	10°
N	0.51	1.52	0.020	0.060



P SUFFIX
PLASTIC PACKAGE
CASE 710

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	36.32	37.34	1.430	1.470
B	13.72	14.22	0.540	0.560
C	4.57	5.08	0.180	0.200
D	0.38	0.51	0.015	0.020
F	1.02	1.52	0.040	0.060
G	2.41	2.67	0.095	0.105
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	3.05	3.56	0.120	0.140
L	14.99	15.49	0.590	0.610
M	0°	10°	0°	10°
N	0.51	1.02	0.020	0.040

MOTOROLA

MC6860
(0 to 70°C; L or P Suffix)

MC6860C
(-40 to 85°C; L Suffix only)

0-600 bps DIGITAL MODEM

The MC6860 is a MOS subsystem designed to be integrated into a wide range of equipment utilizing serial data communications.

The modem provides the necessary modulation, demodulation and supervisory control functions to implement a serial data communications link, over a voice grade channel, utilizing frequency shift keying (FSK) at bit rates up to 600 bps. The MC6860 can be implemented into a wide range of data handling systems, including stand alone modems, data storage devices, remote data communication terminals and I/O interfaces for minicomputers.

N-channel silicon gate technology permits the MC6860 to operate using a single voltage supply and be fully TTL compatible.

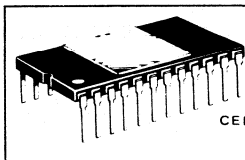
The modem is compatible with the M6800 microcomputer family, interfacing directly with the Asynchronous Communications Interface Adapter to provide low-speed data communications capability.

- Originate and Answer Mode
- Crystal or External Reference Control
- Modem Self Test
- Terminal Interfaces TTL-Compatible
- Full-Duplex or Half-Duplex Operation
- Automatic Answer and Disconnect
- Compatible Functions for 100 Series Data Sets
- Compatible Functions for 1001A/B Data Couplers

MOS

(N-CHANNEL, SILICON-GATE)

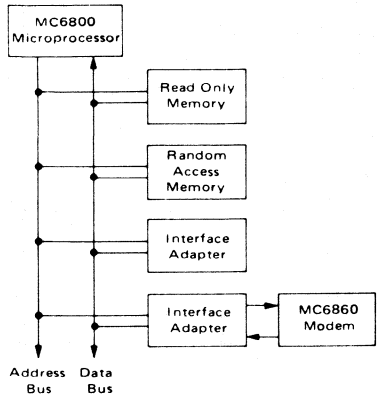
0-600 bps
DIGITAL MODEM



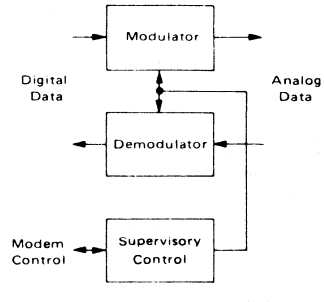
L SUFFIX
CERAMIC PACKAGE
CASE 716

NOT SHOWN: **P SUFFIX**
PLASTIC PACKAGE
CASE 709

M6800 MICROCOMPUTER FAMILY
BLOCK DIAGRAM



MC6860 DIGITAL MODEM
BLOCK DIAGRAM



MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	−0.3 to +7.0	Vdc
Input Voltage	V_{in}	−0.3 to +7.0	Vdc
Operating Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	−55 to +150	°C
Thermal Resistance	θ_{JA}	82.5	°C/W

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit.

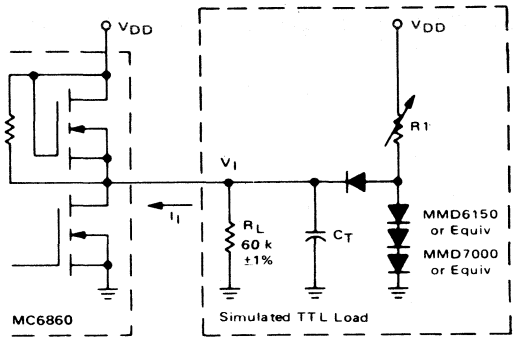
ELECTRICAL CHARACTERISTICS (V_{DD} = 5.0 ±0.25 Vdc, all voltages referenced to V_{SS} = 0, T_A = 0 to 70°C, all outputs loaded as shown in Figure 1 unless otherwise noted.)

Characteristic	Symbol	Min	Typ	Max	Unit
Input High Voltage, All Inputs Except Crystal	V_{IH}	2.0	–	V_{DD}	Vdc
Input Low Voltage, All Inputs Except Crystal	V_{IL}	V_{SS}	–	0.80	Vdc
Crystal Input Voltage (Crystal Input Driven from an External Reference, Input Coupling Capacitor = 200 pF, Duty Cycle = 50 ±5%)	V_{in}	1.5	–	2.0	V_{p-p}
Input Current (V_{in} = V_{SS}) All Inputs Except Rx Car, Tx Data, $\overline{TD}$, TST, $\overline{RI}$, $\overline{SH}$	I_{in}	–	–	-0.2	mAdc
$\overline{RI}$, $\overline{SH}$ Inputs		–	–	-1.6	
Input Leakage Current (V_{in} = 7.0 Vdc, V_{DD} = V_{SS}, T_A = 25°C)	I_{IL}	–	–	1.0	μAdc
Output High Voltage, All Outputs Except An Ph and Tx Car (I_{OH1} = -0.04 mAdc, Load A)	V_{OH1}	2.4	–	V_{DD}	Vdc
Output Low Voltage, All Outputs Except An Ph and Tx Car (I_{OL1} = 1.6 mAdc, Load A)	V_{OL1}	V_{SS}	–	0.40	Vdc
Output High Current, An Ph (V_{OH2} = 0.8 Vdc, Load B)	I_{OH2}	0.30	–		mAdc
Output Low Voltage, An Ph (I_{OL2} = 0, Load B)	V_{OL2}	V_{SS}	–	0.30	Vdc
Input Capacitance (f = 0.1 MHz, T_A = 25°C)	C_{in}	–	5.0	–	pF
Output Capacitance (f = 0.1 MHz, T_A = 25°C)	C_{out}	–	10	–	pF
Transmit Carrier Output Voltage (Load C)	V_{CO}	0.20	0.35	0.50	V(RMS)
Transmit Carrier Output 2nd Harmonic (Load C)	V_{2H}	-25	-32	–	dB
Input Transition Times, All Inputs Except Crystal (Operating in the Crystal Input Mode; from 10% to 90% Points)	t_r	–	–	1.0*	μs
	t_f	–	–	1.0*	
Input Transition Times, Crystal Input (Operating in External Input Reference Mode)	t_r	–	–	30	ns
	t_f	–	–	30	
Output Transition Times, All Outputs Except Tx Car (From 10% to 90% Points)	t_r	–	–	5.0	μs
	t_f	–	–	5.0	
V_{DD} Supply Current (All Inputs at V_{SS} and All Outputs Open)	I_{DD}	–	30	65	mAdc

*Maximum Input Transition Times are ≤ 0.1 x Pulse Width or the specified maximum of 1.0 μs, whichever is smaller.

FIGURE 1 – OUTPUT TEST LOADS

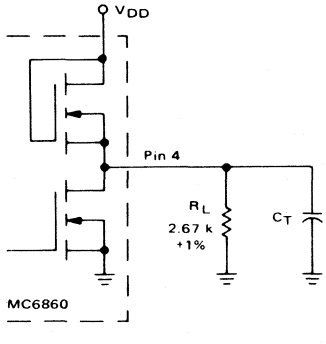
Load A – TTL Output Load for Receive Break, Digital Carrier, Mode, Clear-to-Send, and Receive Data Outputs



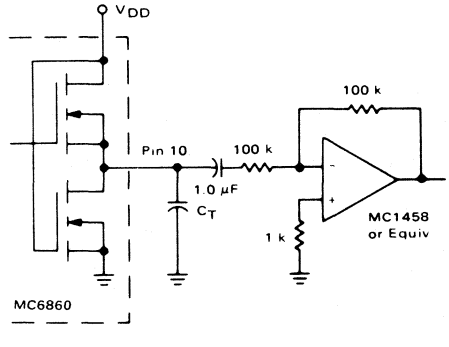
C_T ≈ 20 pF = total parasitic capacitance, which includes probe, wiring, and load capacitances

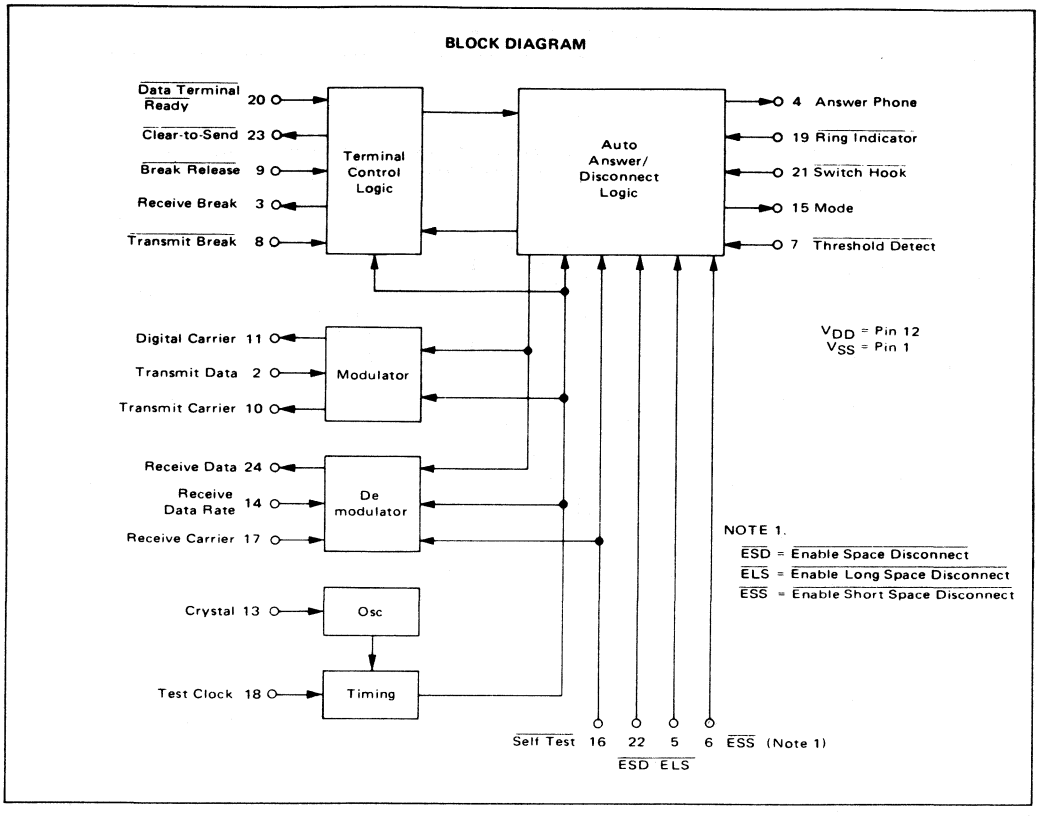
R1 is adjusted for I_I = 1.6 mA at V_I = 0.4 V when output node is disconnected.

Load B – Answer Phone Load



Load C – Transmit Carrier Load





DEVICE OPERATION*

GENERAL

Figure 2 shows the modem and its interconnections. The data to be transmitted is presented in serial format to the modulator for conversion to FSK signals for transmission on the telephone line. The modulator output is buffered before driving the line.

The FSK signal from the remote modem is received via the telephone line and filtered to remove extraneous signals such as the local Transmit Carrier. This filtering can be either a bandpass which passes only the desired band of frequencies or a notch which rejects the known interfering signal. The desired signal is then limited to preserve the axis crossings and fed to the demodulator where the data is recovered from the received FSK carrier.

The Supervisory Control provides the necessary commands and responses for handshaking with the remote modem, along with the interface signals to the data coupler and communication terminal. If the modem is a built-in unit, all input-output (I/O) logic need not be RS-232 compatible. However, if the modem is a stand-alone unit the computer-modem I/O interface must conform to the EIA specification. The use of MC1488 and MC1489A line drivers and receivers will provide the required interface.

*See Tables 1 and 2 for delay time tolerances.

Answer Mode

Automatic answering is first initiated by a receipt of a Ring Indicator ($\overline{RI}$) signal. This can be either a low level for at least 51 ms as would come from a CBS data coupler, or at least 20 cycles of a 20-47 Hz ringing signal (low level $\geq$ 50% of the duty cycle) as would come from a CBT data coupler. The presence of the Ring Indicator signal places the modem in the Answer Mode; if the Data Terminal Ready line is low, indicating the communication terminal is ready to send or receive data, the Answer Phone output goes high. This output is designed to drive a transistor switch which will activate the Off Hook (OH) and

Data Transmission (DA) relays in the data coupler. Upon answering the phone the 2225-Hz Transmit Carrier is turned on.

The originate modem at the other end detects this 2225-Hz signal and after a 450 ms delay (used to disable any echo suppressors in the telephone network) transmits a 1270-Hz signal which the local answering modem detects, provided the amplitude and frequency requirements are met. The amplitude threshold is set external to the modem chip. If the signal level is sufficient the $\overline{TD}$ input should be low for 20 μs at least once every 32 ms. The absence of a threshold indication for a period greater than 51 ms denotes the loss of Receive Carrier and the modem begins hang-up procedures. Hang-up will occur 17 s after $\overline{RI}$ has been released provided the handshaking routine is not re-established. The frequency tolerance during handshaking is ±100 Hz from the Mark frequency.

After the 1270-Hz signal has been received for 150 ms, the Receive Data is unclamped from a Mark condition and data can be received. The Clear-to-Send output goes low 450 ms after the receipt of carrier and data presented to the answer modem is transmitted.

Automatic Disconnect

Upon receipt of a space of 150 ms or greater duration, the modem clamps the Receive Break high. This condition exists until a Break Release command is issued at the receiving station. Upon receipt of a 0.3 s space, with Enable Short Space Disconnect at the most negative voltage (low), the modem automatically hangs up. If Enable Long Space Disconnect is low, the modem requires 1.5 s of continuous space to hang up.

Originate Mode

Upon receipt of a Switch Hook ($\overline{SH}$) command the modem function is placed in the Originate Mode. If the Data Terminal Ready input is enabled (low) the modem will provide a logic high output at Answer Phone. The modem is now ready to receive the 2225-Hz signal from the remote answering modem. It will continue to look for this signal until 17 s after $\overline{SH}$ has been released. Disconnect occurs if the handshaking routine is not established.

Upon receiving 2225 ±100 Hz for 150 ms at an acceptable amplitude, the Receive Data output is unclamped from a Mark condition and data reception can be accomplished. 450 ms after receiving a 2225-Hz signal, a 1270-Hz signal is transmitted to the remote modem. 750 ms after receiving the 2225-Hz signal, the Clear-to-Send output is taken low and data can now be transmitted as well as received.

Initiate Disconnect

In order to command the remote modem to automatically hang up, a disconnect signal is sent by the local modem. This is accomplished by pulsing the normally low Data Terminal Ready into a high state for greater than

FIGURE 2 – TYPICAL MC6860 SYSTEM CONFIGURATION

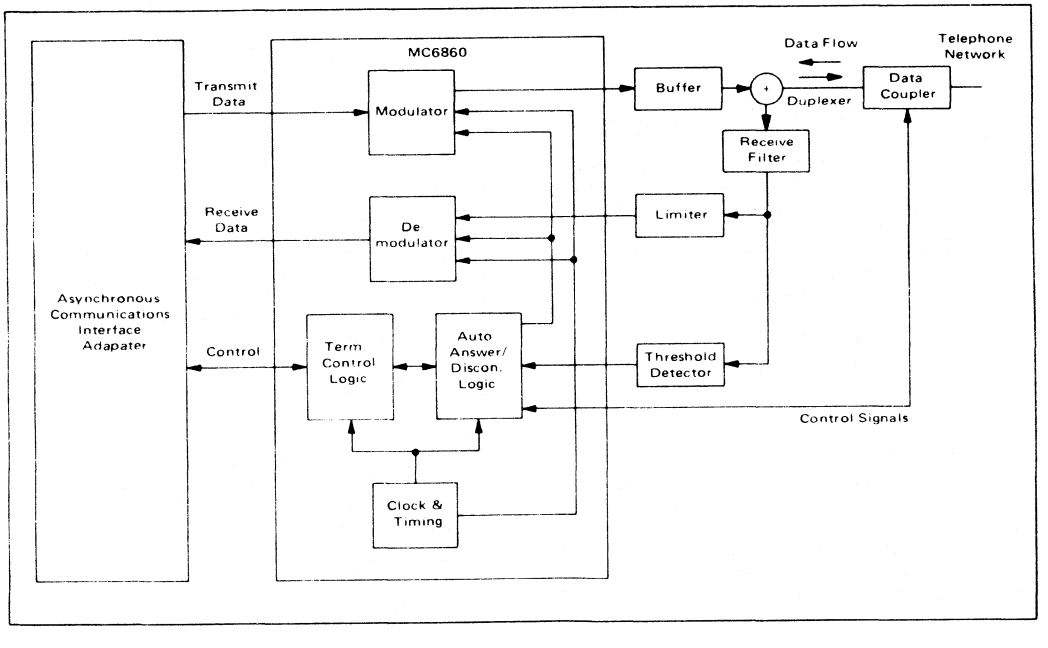
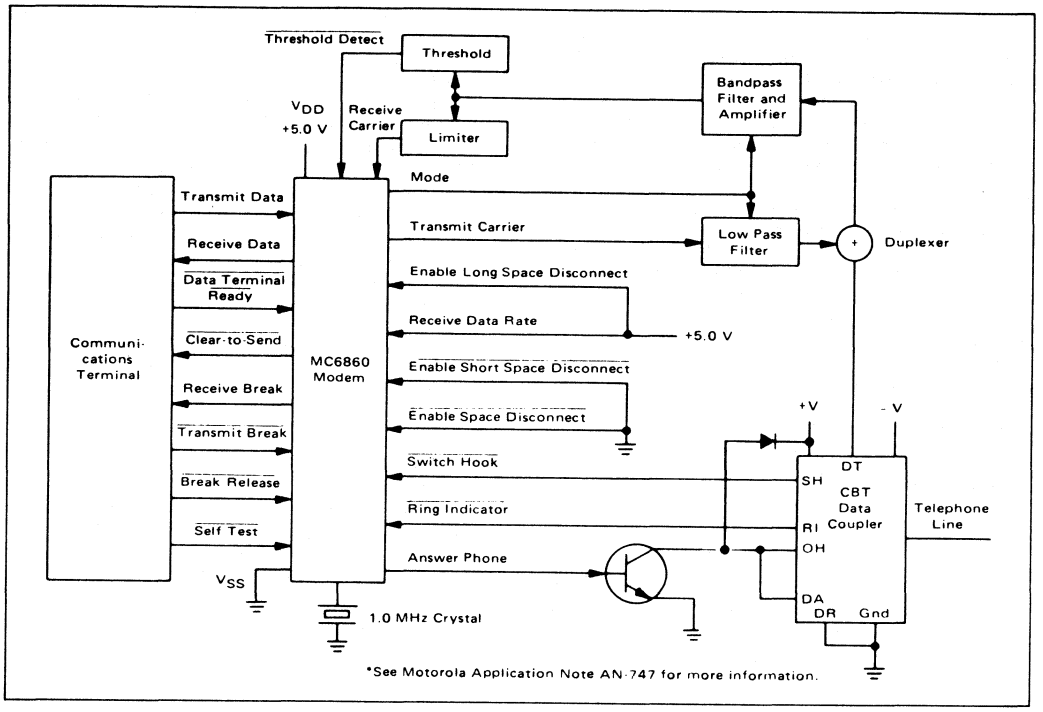


FIGURE 3 – I/O INTERFACE CONNECTIONS FOR MC6860
(ORIGINATE/ANSWER MODEM)



34 ms. The local modem then sends a 3 s continuous space and hangs up provided the $\overline{\text{Enable Space Disconnect}}$ is low. If the remote modem hangs up before 3 s, loss of $\overline{\text{Threshold Detect}}$ will cause loss of $\overline{\text{Clear-to-Send}}$, which marks the line in Answer Mode and turns the carrier off in the Originate Mode.

If $\overline{\text{ESD}}$ is high the modem will transmit data until hang-up occurs 3 s later. $\overline{\text{Receive Break}}$ is clamped 150 ms following the $\overline{\text{Data Terminal Ready}}$ interrupt.

INPUT/OUTPUT FUNCTIONS

Figure 3 shows the I/O interface for the low speed modem. The following is a description of each individual signal:

Receive Carrier (Rx Car)

The Receive Carrier is the FSK input to the demodulator. The local Transmit Carrier must be balanced or filtered out prior to this input, leaving only the Receive Carrier in the signal. The Receive Carrier must also be hard limited. Any half-cycle period greater than or equal to 429 ± 1.0 μs for the low band or 235 ± 1.0 μs for the high band is detected as a space.

$\overline{\text{Ring Indicator}}$ ($\overline{\text{RI}}$)

The modem function will recognize the receipt of a call from the CBT if at least 20 cycles of the 20-47 Hz ringing signal (low level ⩾ 50% of the duty cycle) are present. The CBS $\overline{\text{RI}}$ signal must be level-converted to TTL according to the EIA RS-232 specification before interfacing it with the modem function. The receipt of a call from the CBS is recognized if the $\overline{\text{RI}}$ signal is present for at least 51 ms. This input is held high except during ringing. A $\overline{\text{RI}}$ signal automatically places the modem function in the Answer Mode.

$\overline{\text{Switch Hook}}$ ($\overline{\text{SH}}$)

$\overline{\text{SH}}$ interfaces directly with the CBT and via the EIA RS-232 level conversion for the CBS. An $\overline{\text{SH}}$ signal automatically places the modem function in the Originate Mode.

$\overline{\text{SH}}$ is low during origination of a call. The modem will automatically hang up 17 s after releasing $\overline{\text{SH}}$ if the handshaking routine has not been accomplished.

$\overline{\text{Threshold Detect}}$ ($\overline{\text{TD}}$)

This input is derived from an external threshold detector. If the signal level is sufficient, the $\overline{\text{TD}}$ input must

be low for 20 μs at least once every 32 ms to maintain normal operation. An insufficient signal level indicates the absence of the Receive Carrier; an absence for less than 32 ms will not cause channel establishment to be lost; however, data during this interval will be invalid.

If the signal is present and the level is acceptable at all times, then the threshold input can be low permanently.

Loss of threshold for 51 ms or longer results in a loss of Clear-to-Send. The Transmit Carrier of the originate modem is clamped off and a constant Mark is transmitted from the answer modem.

Receive Data Rate (Rx Rate)

The demodulator has been optimized for signal-to-noise performance at 300 bps and 600 bps. The Receive Data Rate input must be low for 0-600 bps and should be high for 0-300 bps.

Transmit Data (Tx Data)

Transmit Data is the binary information presented to the modem function for modulation with FSK techniques. A high level represents a Mark.

Data Terminal Ready ($\overline{DTR}$)

The Data Terminal Ready signal must be low before the modem function will be enabled. To initiate a disconnect, $\overline{DTR}$ is held high for 34 ms minimum. A disconnect will occur 3 s later.

Break Release ($\overline{Brk\ R}$)

After receiving a 150 ms space signal, the clamped high condition of the Receive Break output can be removed by holding Break Release low for at least 20 μs.

Transmit Break ($\overline{Tx\ Brk}$)

The Break command is used to signal the remote modem to stop sending data.

A Transmit Break (low) greater than 34 ms forces the modem to send a continuous space signal for 233 ms. Transmit Break must be initiated only after $\overline{CTS}$ has been established. This is a negative edge sense input. Prior to initiating $\overline{Tx\ Brk}$, this input must be held high for a minimum of 34 ms.

Enabled Space Disconnect ($\overline{ESD}$)

When $\overline{ESD}$ is strapped low and $\overline{DTR}$ is pulsed to initiate a disconnect, the modem transmits a space for either 3 s or until a loss of threshold is detected, whichever occurs first. If $\overline{ESD}$ is strapped high, data instead of a space is transmitted. A disconnect occurs at the end of 3 s.

Enable Short Space Disconnect ($\overline{ESS}$)

$\overline{ESS}$ is a strapping option which, when low, will automatically hang up the phone upon receipt of a continuous space for 0.3 s. *$\overline{ESS}$ and $\overline{ELS}$ must not be simultaneously strapped low.*

Enable Long Space Disconnect ($\overline{ELS}$)

$\overline{ELS}$ is a strapping option which, when low, will automatically hang up the phone upon receipt of a continuous space for 1.5 s.

Crystal (Xtal)

A 1.0-MHz crystal with the following parameters is required to utilize the on-chip oscillator. A 1.0-MHz square wave can also be fed into this input to satisfy the clock requirement.

Mode:	Parallel
Frequency:	1.0 MHz ±0.1%
Series Resistance:	750 ohms max
Shunt Capacitance:	7.0 pF max
Temperature:	0-70°C
Test Level:	1.0 mW
Load Capacitance:	13 pF

When utilizing the 1.0-MHz crystal, external parasitic capacitance, including crystal shunt capacitance, must be $\leq$ 9 pF at the crystal input. Reliable crystal oscillator start-up requires that the V_{DD} power-on transition time be $>$ 15 milliseconds.

Test Clock (TST)

A test signal input is provided to decrease the test time of the chip. In normal operation this input *must be strapped low.*

Self Test ($\overline{ST}$)

When a low voltage level is placed on this input, the demodulator is switched to the modulator frequency and demodulates the transmitted FSK signal. Channel establishment, which occurred during the initial handshake, is not lost during self test. The Mode Control output changes state during Self Test, permitting the receive filters to pass the local Transmit Carrier.

$\overline{ST}$	$\overline{SH}$	$\overline{RI}$	Mode
H	L	H	H
H	H	L	L
L	L	H	L
L	H	L	H

Answer Phone (An Ph)

Upon receipt of Ring Indicator or Switch Hook signal and Data Terminal Ready, the Answer Phone output goes high [($\overline{SH}$ + $\overline{RI}$) • $\overline{DTR}$]. This signal drives the base of a transistor which activates the Off Hook and Data Transmission control lines in the data coupler. Upon call completion, the Answer Phone signal returns to a low level.

Mode

The Mode output indicates the Answer (low) or Originate (high) status of the modem. This output changes state when a Self Test command is applied.

Clear-To-Send ($\overline{CTS}$)

A low on the $\overline{CTS}$ output indicates the Transmit Data input has been unclamped from a steady Mark, thus allowing data transmission.

Receive Data (Rx Data)

The Receive Data output is the data resulting from demodulating the Receive Carrier. A Mark is a high level.

Receive Break (Rx Brk)

Upon receipt of a continuous 150 ms space, the modem automatically clamps the Receive Break output high. This output is also clamped high until Clear-to-Send is established.

Digital Carrier (FO)

A test signal output is provided to decrease the chip test time. The signal is a square wave at the transmit frequency.

Transmit Carrier (Tx Car)

The Transmit Carrier is a digitally-synthesized sine wave (Figure 4) derived from the 1.0-MHz crystal reference. The frequency characteristics are as follows:

Mode	Data	Transmit Frequency	Tolerance*
Originate	Mark	1270 Hz	-0.15 Hz
Originate	Space	1070 Hz	0.09 Hz
Answer	Mark	2225 Hz	-0.31 Hz
Answer	Space	2025 Hz	-0.71 Hz

*The reference frequency tolerance is not included.

The proper output frequency is transmitted within 3.0 μs following a data bit change with no more than 2.0 μs phase discontinuity. The typical output level is 0.35 V (RMS) into a 100 k-ohm load impedance.

The second harmonic is typically 32 dB below the fundamental (Figure 5).

POWER-ON RESET

Power-on reset is provided on-chip to insure that when power is first applied the Answer Phone output is in the low (inactive) state. This holds the modem in the inactive or idle mode until a $\overline{SH}$ or $\overline{RI}$ signal has been applied. Once power has been applied, a momentary loss of power at a later time may not be of sufficient time to guarantee a chip reset through the power-on reset circuit.

To insure initial power-on reset action, the external parasitic capacitance on $\overline{RI}$ and $\overline{SH}$ should be < 30 pF. Capacitance values > 30 pF may require the use of an external pullup resistor to V_{DD} on these inputs in addition to the pullup devices already provided on chip.

FIGURE 4 – TRANSMIT CARRIER SINE WAVE

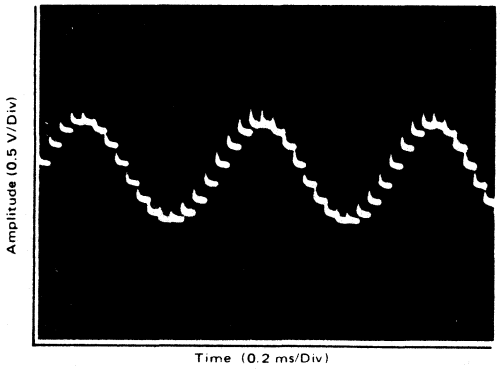
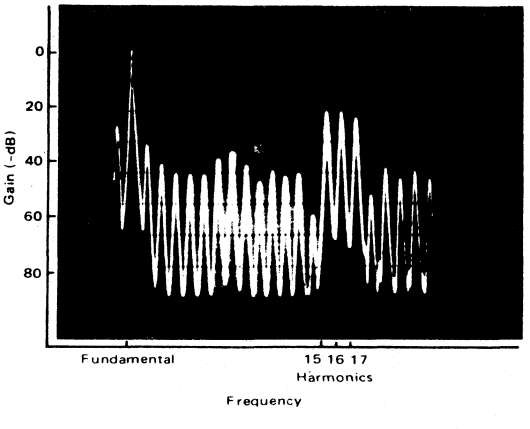


FIGURE 5 – TRANSMIT CARRIER FREQUENCY SPECTRUM



TIMING DIAGRAMS

FIGURE 6 – ANSWER MODE

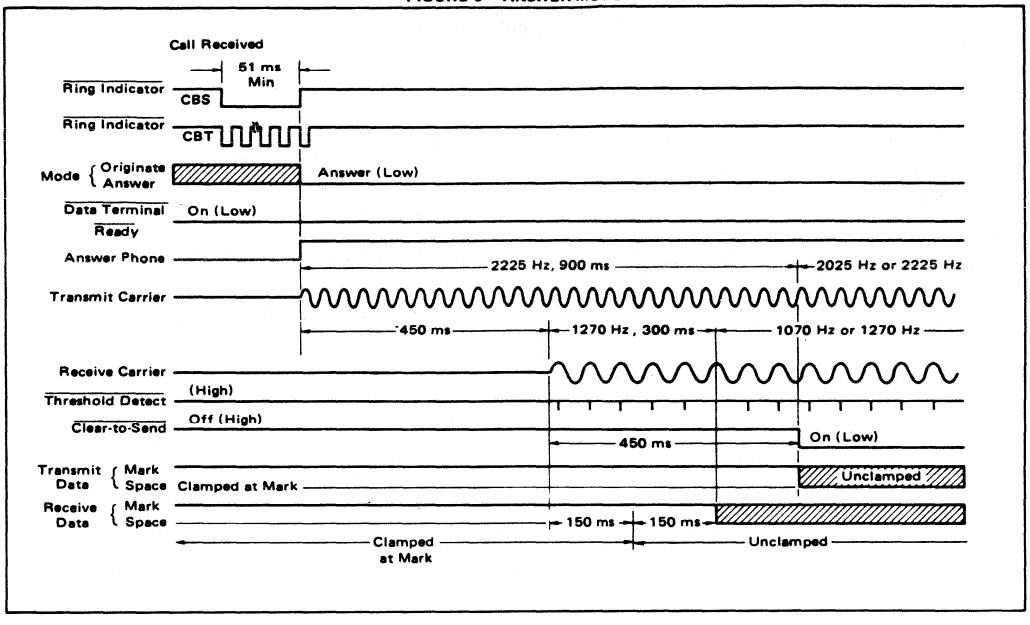


FIGURE 7 – AUTOMATIC DISCONNECT – LONG OR SHORT SPACE

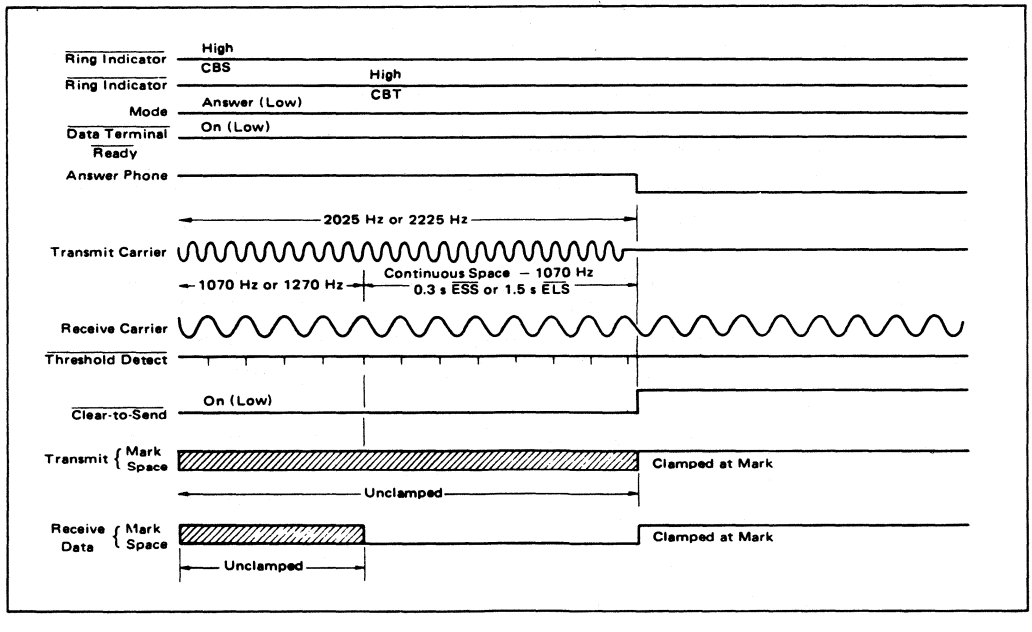


FIGURE 8 – ORIGINATE MODE

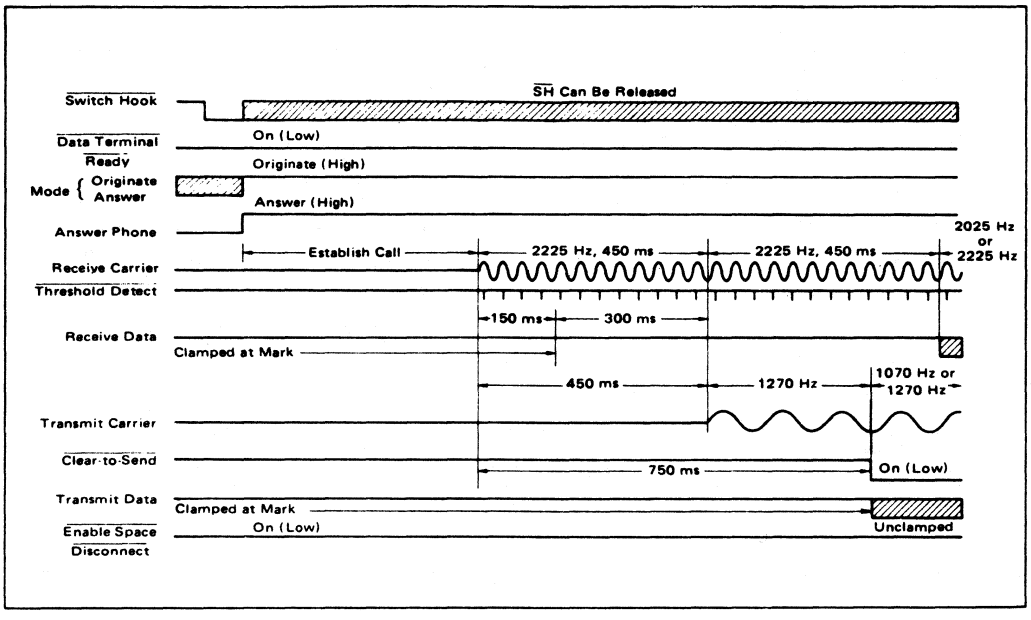


FIGURE 9 – INITIATE DISCONNECT

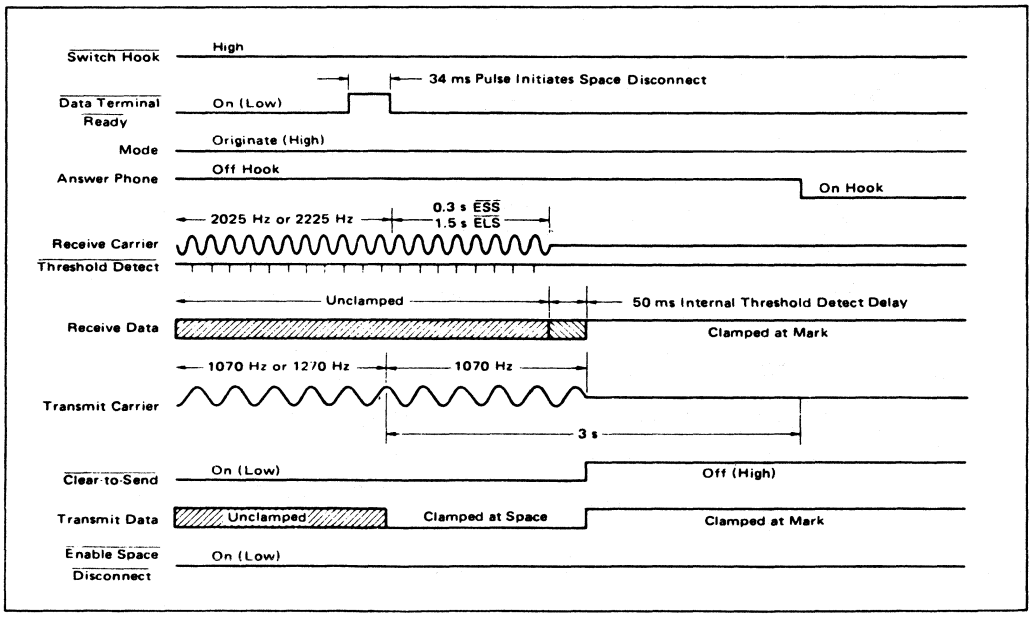
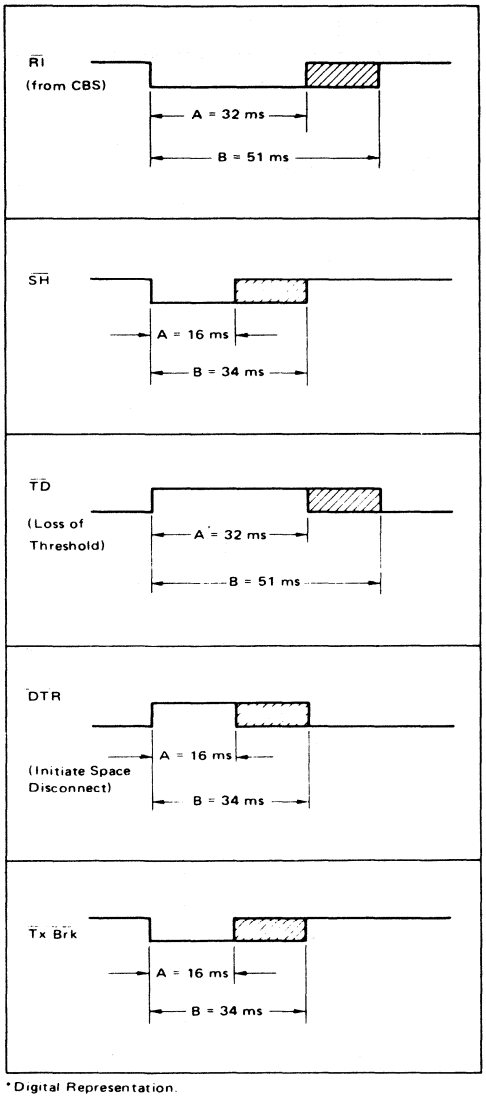


TABLE 1 – ASYNCHRONOUS INPUT PULSE WIDTH AND OUTPUT DELAY VARIATIONS
(Time delays specified do not include the 1-MHz reference tolerance.)

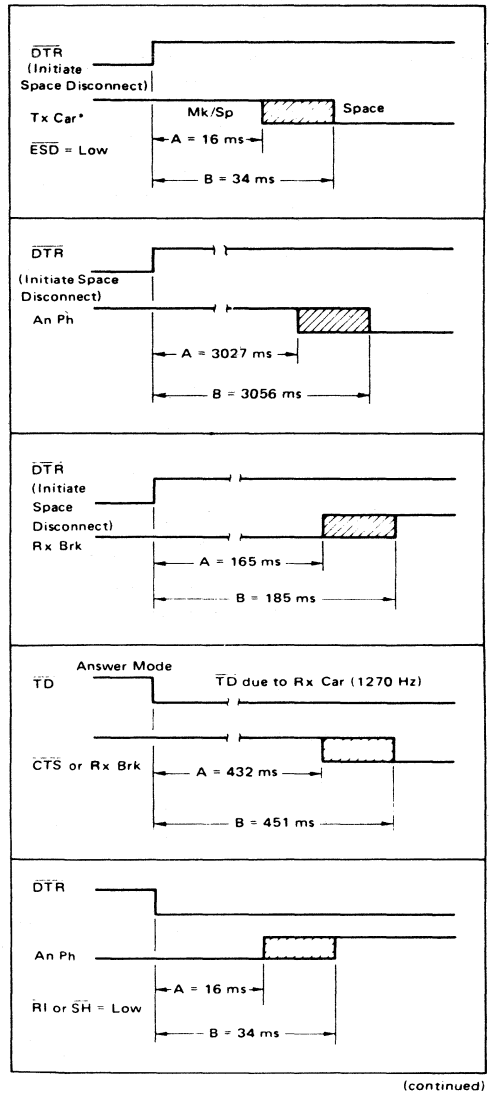
Due to the asynchronous nature of the input signals with respect to the circuit internal clock, a delay variation or input pulse width requirement will exist. Time delay A is the maximum time for which no response will occur. Time delay B is the minimum time required to guarantee an input response. Input signal widths in the cross-hatched region (i.e., greater than A but less than B) may or may not be recognized as valid.

For output delays, time A is the minimum delay before an output will respond. Time B is the maximum delay for an output to respond. Output signal response may or may not occur in the cross-hatched region (i.e., greater than A but less than B).

INPUT PULSES



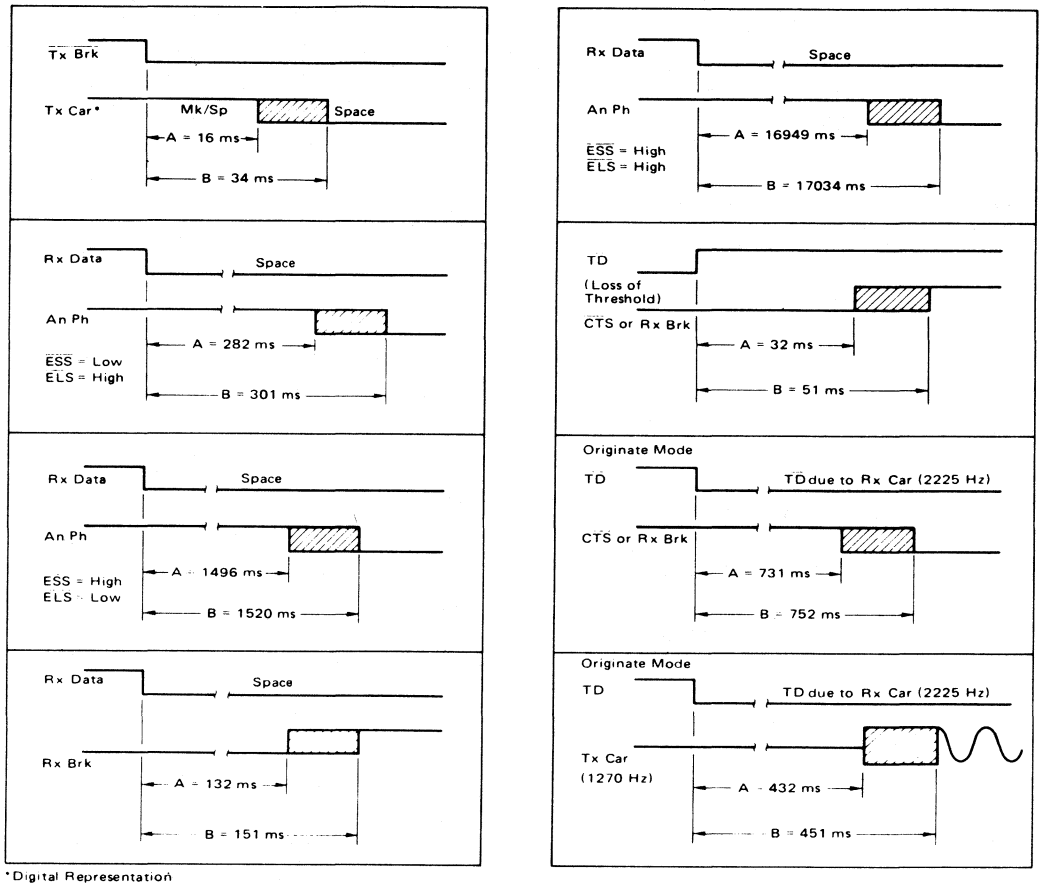
OUTPUT DELAYS



*Digital Representation.

(continued)

TABLE 1 – OUTPUT DELAY VARIATIONS (continued)

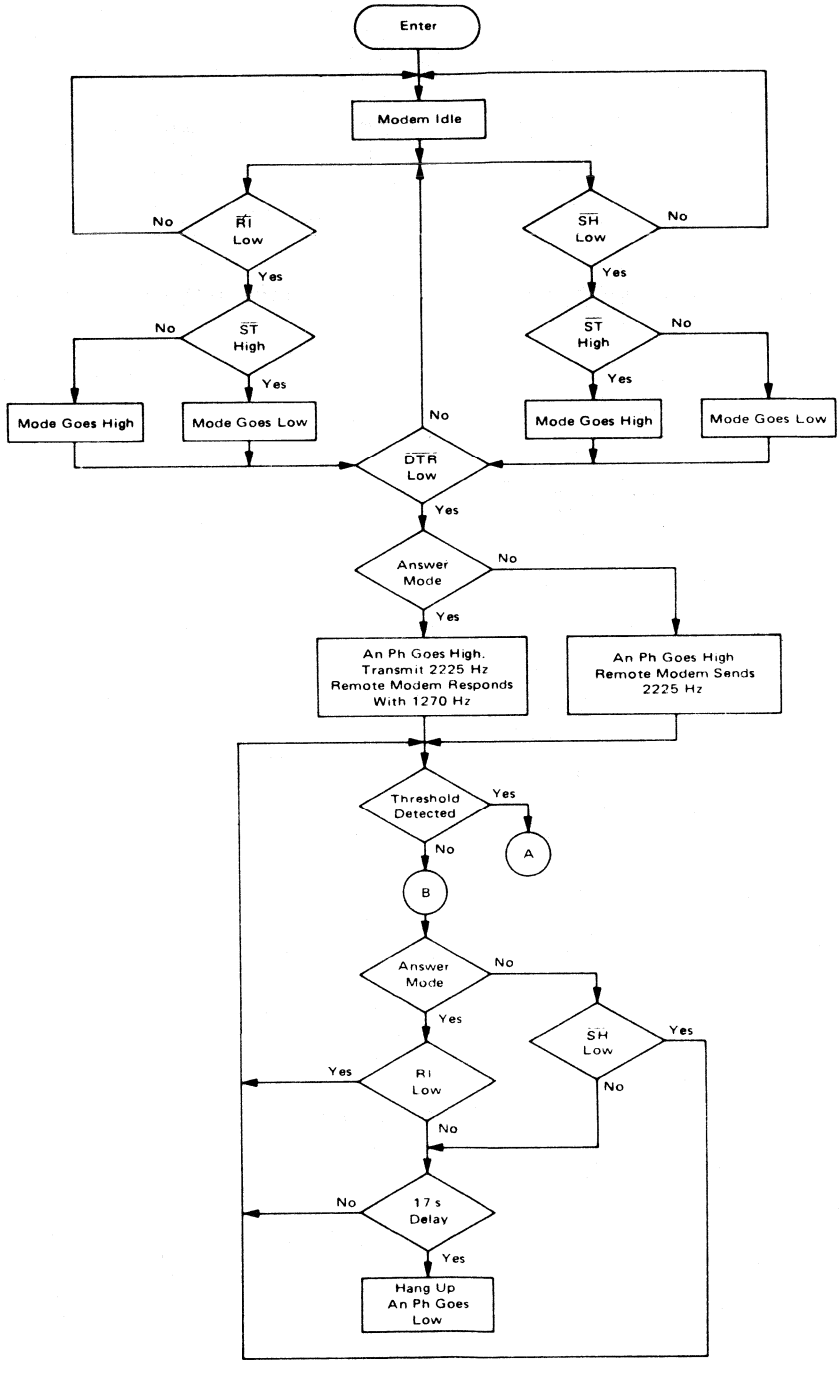


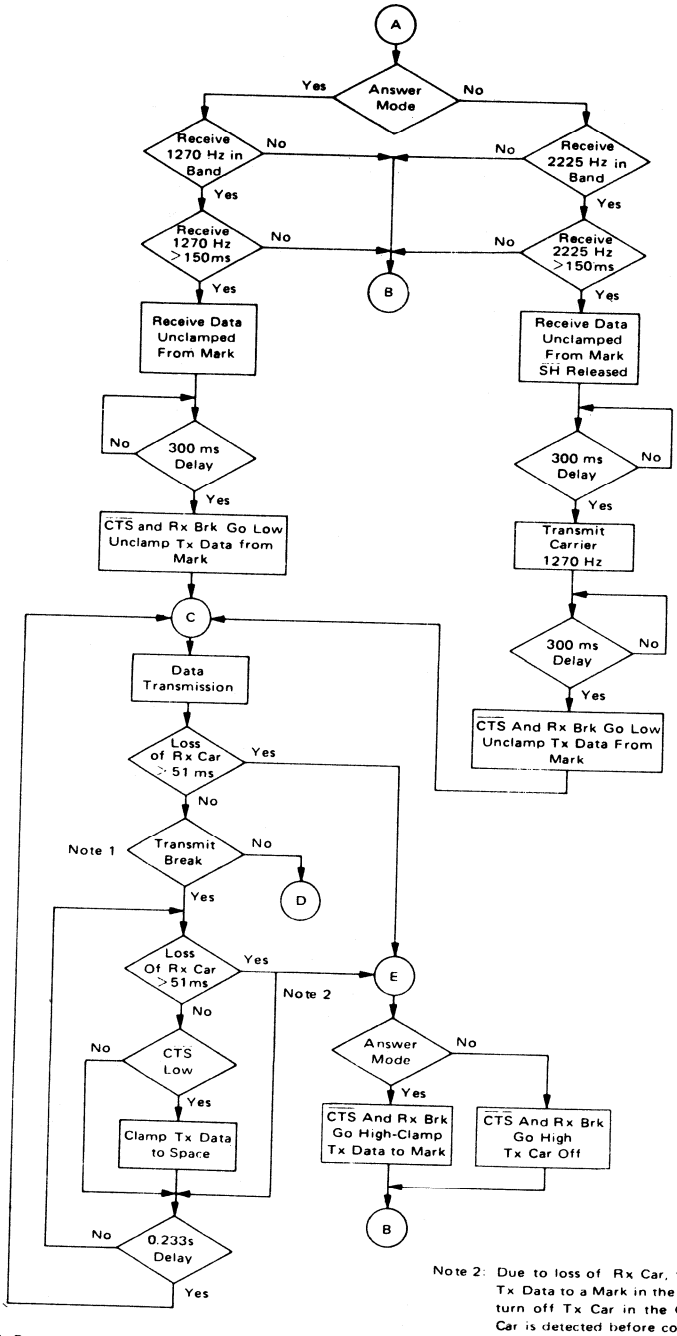
*Digital Representation

TABLE 2 – TRANSMIT BREAK AND DISCONNECT DELAYS

Function Description	Min	Max	Unit
$\overline{Tx\ Brk}$ (Space Duration)	232	235	ms
Space Disconnect (Space Duration) (DTR = High, $\overline{ESD}$ and TD = Low)	3010	3023	ms
Loss of Carrier Disconnect (Measured from positive edge of $\overline{CTS}$ to negative edge of An Ph, with $\overline{RI}$, SH, and TD = High)	16965	17034	ms
Override Disconnect (Measured from positive edge of RI or $\overline{SH}$ to negative edge of An Ph, with TD = High)	16916	17101	ms

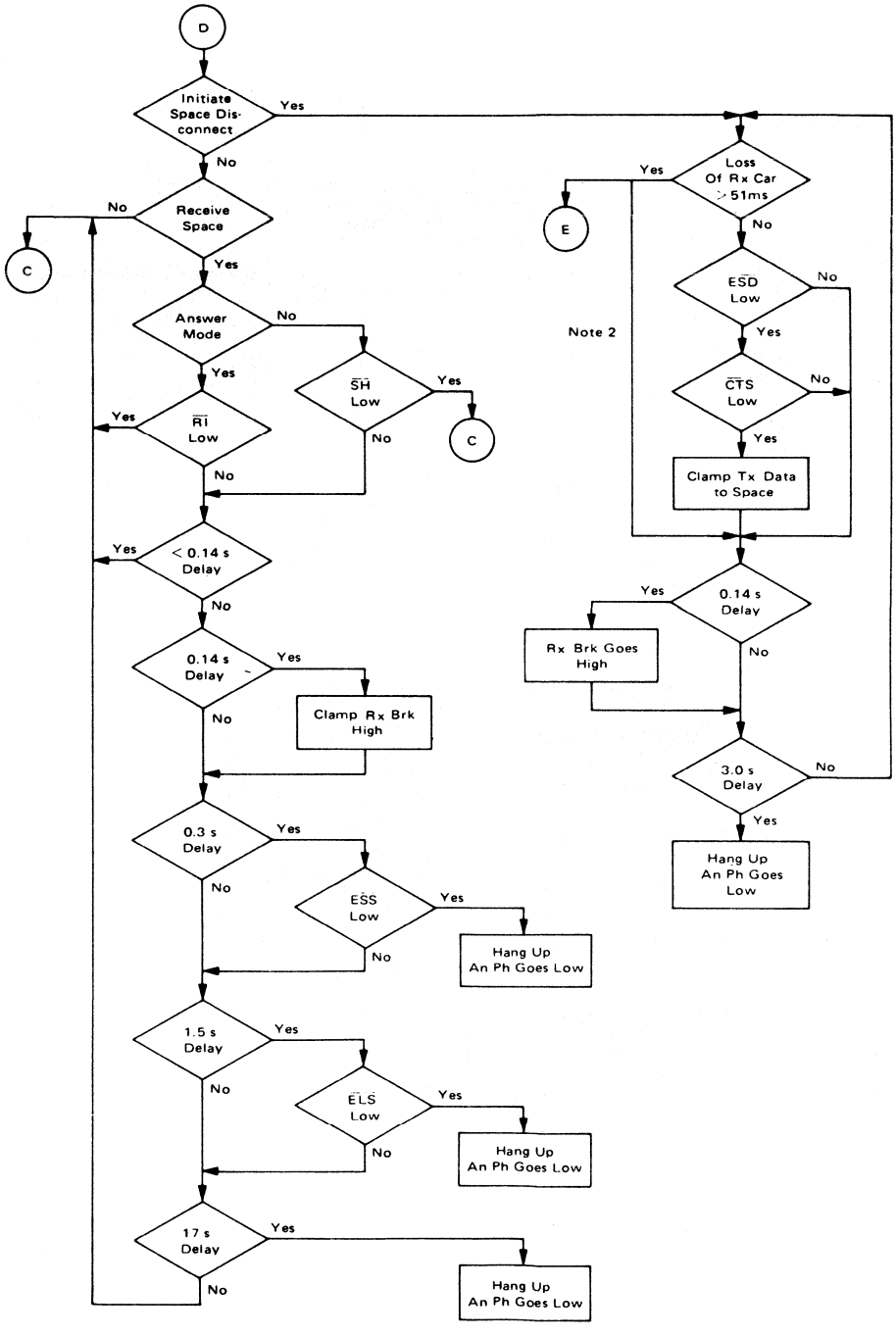
FIGURE 10 – FLOW DIAGRAM



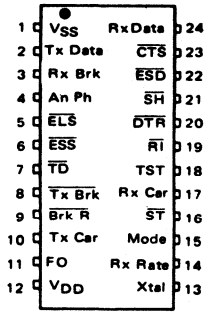


Note 1: Transmit Break, Initiate Space Disconnect, and Receive Space are mutually exclusive events.

Note 2: Due to loss of Rx Car, the modem will clamp Tx Data to a Mark in the Answer Mode and will turn off Tx Car in the Originate Mode. If Rx Car is detected before completion of Tx Brk or Initiate Space Disconnect, normal operation of Tx Brk or Initiate Space Disconnect will continue until completion of their respective time delays.


D
Initiate Space Disconnect
Yes
No
Receive Space
No
C
Yes
Answer Mode
No
Yes
SH Low
Yes
C
No
Yes
RI Low
No
Yes
< 0.14 s Delay
No
0.14 s Delay
Yes
Clamp Rx Brk High
No
0.3 s Delay
Yes
ESS Low
Yes
Hang Up An Ph Goes Low
No
1.5 s Delay
Yes
ELS Low
Yes
Hang Up An Ph Goes Low
No
17 s Delay
Yes
Hang Up An Ph Goes Low
No
Yes
Loss Of Rx Car > 51ms
Yes
E
No
ESD Low
No
Note 2
Yes
CTS Low
No
Yes
Clamp Tx Data to Space
Yes
0.14 s Delay
Yes
Rx Brk Goes High
No
3.0 s Delay
No
Yes
Hang Up An Ph Goes Low

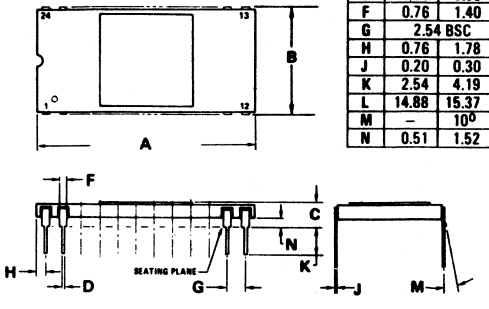
PIN ASSIGNMENT



PACKAGE DIMENSIONS

NOTE:
1. LEADS TRUE POSITIONED WITHIN 0.25mm (0.010) DIA (AT SEATING PLANE) AT MAXIMUM MATERIAL CONDITION.

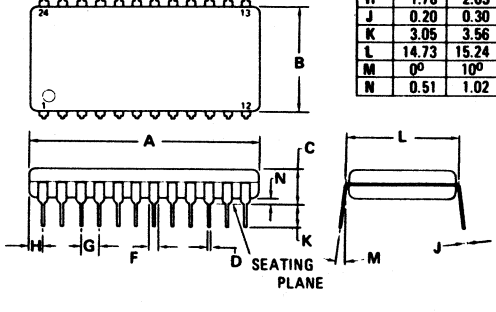
	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	29.97	30.99	1.180	1.220
B	14.88	15.62	0.585	0.615
C	3.05	4.19	0.120	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.30	0.008	0.012
K	2.54	4.19	0.100	0.165
L	14.88	15.37	0.585	0.605
M	–	10^0	–	10^0
N	0.51	1.52	0.020	0.060



CASE 716-02
(CERAMIC)

NOTES:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION. (DIM. "D")
2. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	31.37	32.13	1.235	1.265
B	13.72	14.22	0.540	0.560
C	4.57	5.08	0.180	0.200
D	0.36	0.51	0.014	0.020
F	1.02	1.52	0.040	0.060
G	2.41	2.67	0.095	0.105
H	1.78	2.03	0.070	0.080
J	0.20	0.30	0.008	0.012
K	3.05	3.56	0.120	0.140
L	14.73	15.24	0.580	0.600
M	0^0	10^0	0^0	10^0
N	0.51	1.02	0.020	0.040



CASE 709-01
(PLASTIC)

MOTOROLA

MC6862

(0 to 70°C; L or P Suffix)

MC6862C

(-40 to 85°C; L Suffix only)

Advance Information

2400 bps DIGITAL MODULATOR

The MC6862 is a MOS subsystem designed to be integrated into a wide range of equipment utilizing serial data communication.

The modulator provides the necessary modulation and control functions to implement a serial data communication link over a voice grade channel, utilizing differential phase shift keying (DPSK) at bit rates of 1200 or 2400 bps. Phase options are provided for both the U.S. and international markets. The MC6862 can be implemented into a wide range of data handling systems, including stand-alone modems, data storage devices, remote data communication terminals, and I/O interfaces for counters.

N-channel silicon gate technology permits the MC6862 to operate using a single voltage supply and be fully TTL compatible.

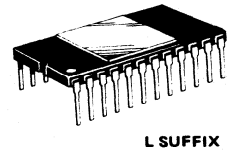
The modulator is compatible with the M6800 microcomputer family, and provides medium-speed data communications capability.

- Clear-to-Send Delay Options
- 511-Bit CCITT Test Pattern
- Terminal Interfaces Are TTL Compatible
- Compatible Functions for 201B/C Data Sets
- CCITT and U.S. Phase Options
- 1200/2400 bps Operation
- Answer Back Tone

MOS

(N-CHANNEL, SILICON-GATE)

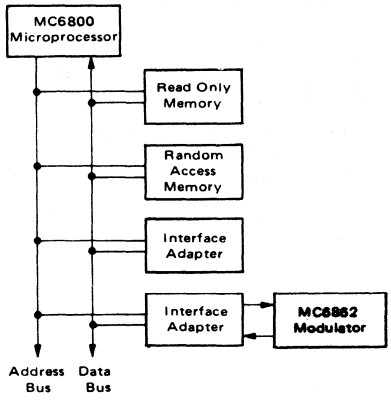
2400 bps MODULATOR



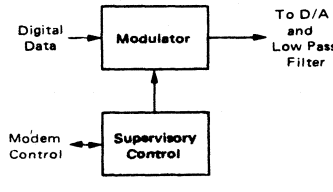
L SUFFIX
CERAMIC PACKAGE
CASE 716

NOT SHOWN: **P SUFFIX**
PLASTIC PACKAGE
CASE 709

M6800 MICROCOMPUTER FAMILY BLOCK DIAGRAM



MC6862 DIGITAL MODULATOR BLOCK DIAGRAM



This is advance information and specifications are subject to change without notice.

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	−0.3 to +7.0	Vdc
Input Voltage	V_{in}	−0.3 to +7.0	Vdc
Operating Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	−55 to +150	°C
Thermal Resistance	θ_{JA}	82.5	°C/W

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit.

ELECTRICAL CHARACTERISTICS (V_{DD} = 5.0 ±0.25 Vdc, V_{SS} = 0, T_A = 0 to 70°C, all outputs loaded as shown in Figure 1 unless otherwise noted.)

Characteristic	Symbol	Min	Typ	Max	Unit
Input High Voltage	V_{IH}	V_{SS} + 2.0	–	V_{DD}	Vdc
Input Low Voltage	V_{IL}	V_{SS}	–	V_{SS} + 0.8	Vdc
Input Current (V_{in} = V_{SS}) CTS1, CTS2, PSS, DRS, $\overline{An\ Bk}$, and $\overline{Tx\ MK}$	I_{in}	–	–	-0.2	mAdc
$\overline{RTS}$ and $\overline{TPE}$		–	–	-1.6	
Input Leakage Current (V_{in} = 5.25 Vdc, V_{DD} = V_{SS})	I_{IL}	–	–	2.5	μAdc
Output High Voltage (I_{OH} = -0.04 mAdc, Load A)	V_{OH1}	V_{SS} + 2.4	–	V_{DD}	Vdc
(I_{OH} = 0.0 mAdc, Load B)	V_{OH2}	V_{DD}-0.5 V	–	V_{DD}	
Output Low Voltage (I_{OL} = 1.6 mAdc, Load A)	V_{OL}	V_{SS}	–	V_{SS} + 0.4	Vdc
Input Capacitance (f = 0.1 MHz, T_A = 25°C)	C_{in}	–	5.0	–	pF
V_{DD} Supply Current (All Inputs at V_{SS} except Pin 13 = 57.6 kHz and ALL Outputs Open)	I_{DD}	–	40	60	mAdc
Input Transition Times, All Inputs Except 1.8432 MHz Input (From 10% to 90% points)	t_r	–	–	1.0*	μs
	t_f	–	–	1.0*	
Input Transition Times, 1.8432 MHz Input (From 0.8 V to 2.0 V)	t_r	–	–	40	ns
	t_f	–	–	40	
Input Clock Duty Cycle, 1.8432 MHz Input (Measured at 1.5 V level)	D.C.	30	–	70	%
Tx Data Setup Time (Figure 2)	t_S	35	–	–	μs
Tx Data Hold Time (Figure 2)	t_H	35	–	–	μs
Output Transition Times (From 10% to 90% Points)	t_r	–	–	5.0	μs
	t_f	–	–	5.0	

*Maximum Input Transition Times are ≤ 0.1 x Pulse Width or the specified maximum of 1.0 μs, whichever is smaller.

FIGURE 1 – OUTPUT TEST LOADS

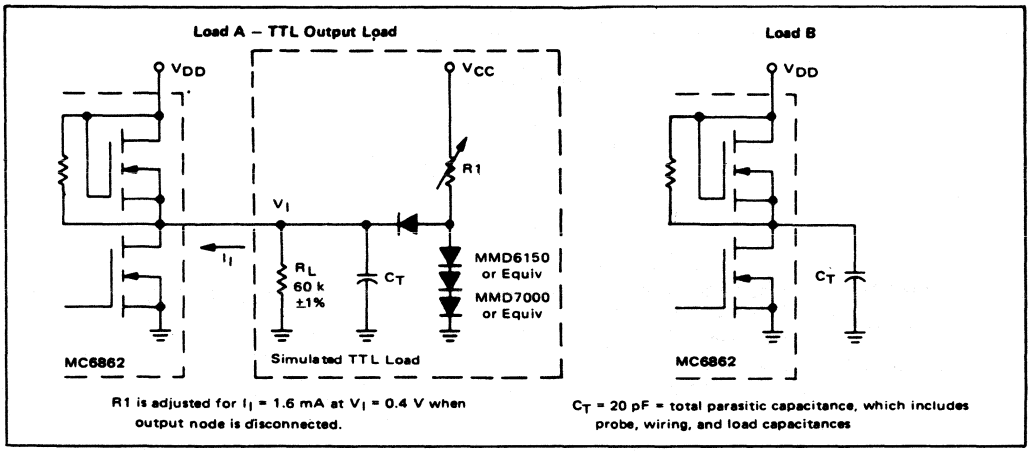
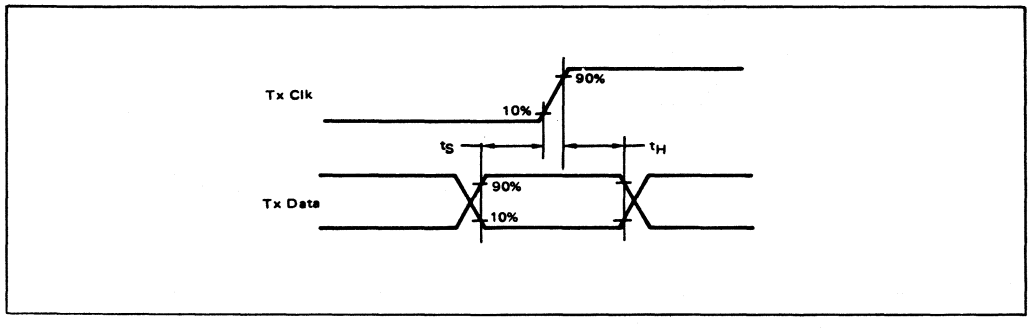
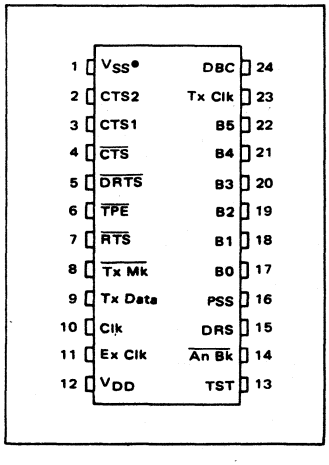


FIGURE 2 – TRANSMIT DATA SETUP AND HOLD TIME

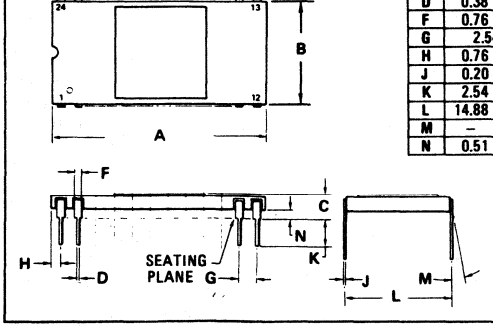


PIN ASSIGNMENT



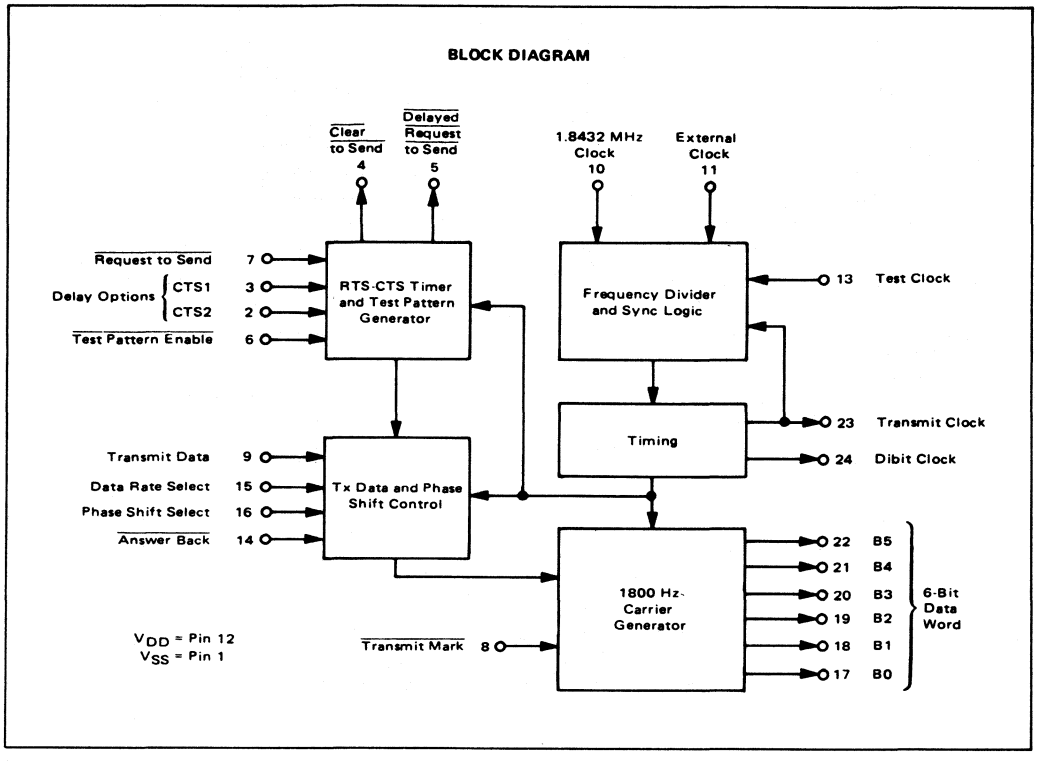
PACKAGE DIMENSIONS

NOTE:
1. LEADS TRUE POSITIONED WITHIN 0.25mm (0.010) DIA (AT SEATING PLANE) AT MAXIMUM MATERIAL CONDITION.



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	29.97	30.99	1.180	1.220
B	14.88	15.62	0.585	0.615
C	3.05	4.19	0.120	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.30	0.008	0.012
K	2.54	4.19	0.100	0.165
L	14.88	15.37	0.585	0.605
M	–	10°	–	10°
N	0.51	1.52	0.020	0.060

CASE 716-02
(CERAMIC)



DEVICE OPERATION

GENERAL

Figure 3 shows the modulator and its intraconnections. The data to be transmitted is presented in synchronous serial format to the modulator for conversion to DPSK signals used in transmission. The modulator output is digital; therefore, a D/A converter and a filter transform the signal to an analog form.

The control functions provide four different Clear-to-Send delay options. An Answer Back tone is available for automatic answering applications. The modulator has a built-in 511-bit pseudorandom pattern generator for use in system diagnostic tests.

INPUT/OUTPUT FUNCTIONS

Request to Send ($\overline{RTS}$)

The $\overline{RTS}$ signal from the data terminal controls transmission from the modulator. A low level on $\overline{RTS}$ activates the modulator data output. A constant mark, for synchronization, is sent during the $\overline{RTS}$ to $\overline{CTS}$ delay interval. Termination of the transmission is accomplished by taking $\overline{RTS}$ high (see Figures 4 and 5).

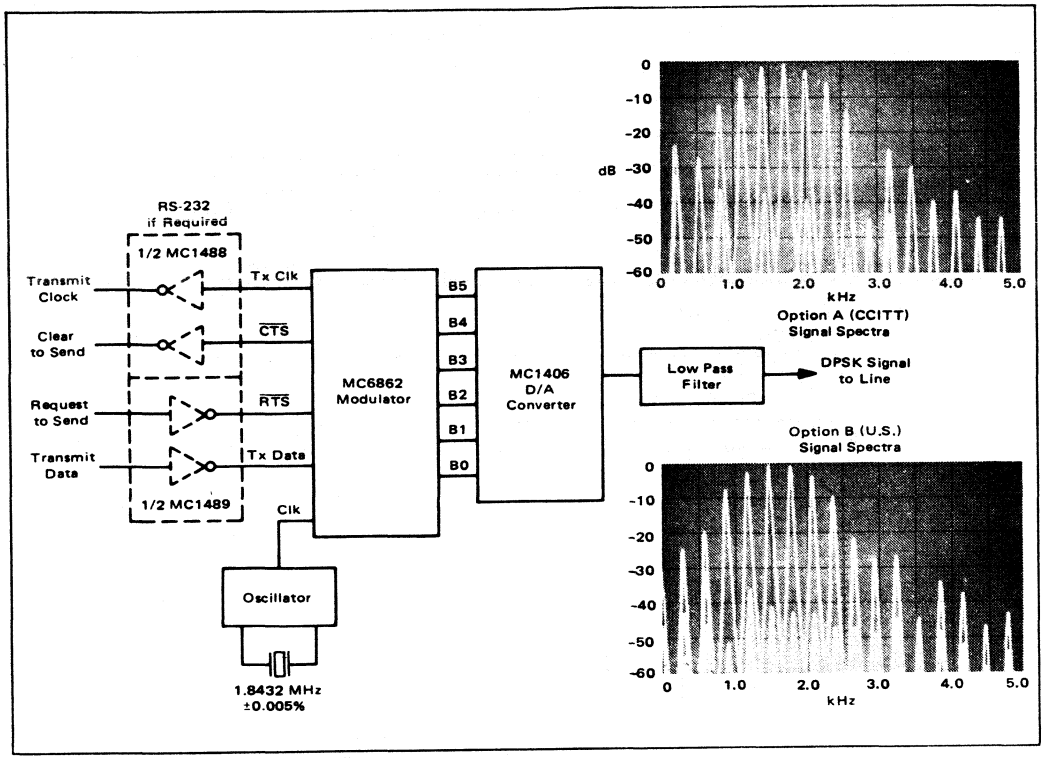
Delayed Request to Send ($\overline{DRTS}$)

This output can be used to control transmission as specified by the Transmit Mark control input. $\overline{DRTS}$ follows the negative transition of $\overline{RTS}$, and goes negative within the 35 μs of the negative transition of $\overline{RTS}$ (Figure 4). The delay from a positive transition of $\overline{RTS}$ to a positive transition of $\overline{DRTS}$ is shown in Figure 5. The $\overline{DRTS}$ delay allows data within the modulator to be transmitted before transmission is inhibited.

Clear to Send ($\overline{CTS}$)

$\overline{CTS}$ follows $\overline{RTS}$ to both the logic 0 and logic 1 levels. The delay from a negative transition of $\overline{RTS}$ to a negative $\overline{CTS}$ transition is selectable by external strapping of CTS1 and CTS2. The delay from a positive transition of $\overline{RTS}$ to a positive $\overline{CTS}$ transition is less than 35 μs.

FIGURE 3 – 2400 BPS MODULATOR INTERFACE



$\overline{CTS}$ will go low within 35 μs after the positive transition of the Dibit Clock (see Figure 4) except when the no-delay option is selected. For the no-delay option, $\overline{CTS}$ follows $\overline{RTS}$ within 35 μs.

$\overline{RTS}$-$\overline{CTS}$ Delay Options (CTS1,CTS2)

The $\overline{RST}$-$\overline{CTS}$ delays are selectable according to the following strapping options.

$\overline{RTS}$-$\overline{CTS}$ Delay	CTS1	CTS2
0.0 + 0.035 ms, -0.0 ms	0	1
8.55 to 9.35 ms	1	0
24.90 to 26.4 ms	1	1
147.0 to 154.0 ms	0	0

Transmit Mark ($\overline{Tx\ Mk}$)

The Transmit Mark control allows the system designer to select whether the Delayed Request to Send activates and deactivates the transmission on the modulator chip or off the chip in the output amplifier.

When $\overline{Tx\ Mk}$ is high, transmission is controlled on the modulator chip, and occurs from the chip only when $\overline{DRTS}$ or Answer Back is in the logic 0 state (see Figure 6).

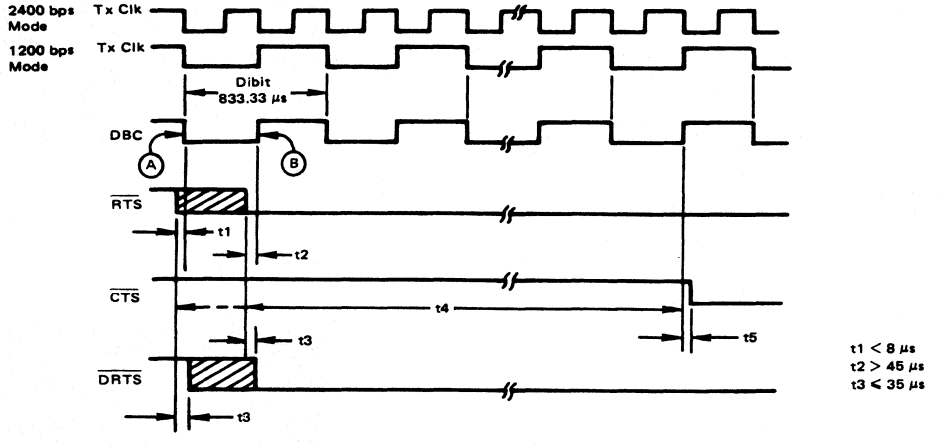
When $\overline{Tx\ Mk}$ is low, transmission is controlled off the modulator chip. In this mode, the modulator chip transmits marks at all times except when data or an Answer Back tone is being transmitted (see Figure 6).

Test Pattern Enable ($\overline{TPE}$)

A 511-bit test pattern generator is contained on the modulator chip. This pattern is in accord with CCITT specification V52. The pattern can be used to scramble input data, or as a test pattern.

The 511-bit test pattern is activated by applying a logic 0 to $\overline{TPE}$. A mark (logic 1) condition on the Transmit Data input with $\overline{TPE}$ activated (logic 0) causes the test pattern to appear at the data output. A space

FIGURE 4 – $\overline{\text{RTS}}$-$\overline{\text{CTS}}$ AND $\overline{\text{RTS}}$-$\overline{\text{DRTS}}$ DELAYS



CTS1	CTS2	t4*	t5
0	1	$0.0^{+0.035\ \text{ms}}_{-0.0\ \text{ms}}$	–
1	0	8.55 to 9.35 ms	< 35 μs
1	1	24.90 to 26.4 ms	< 35 μs
0	0	147.0 to 154.0 ms	< 35 μs

*The reference frequency tolerance is not included.

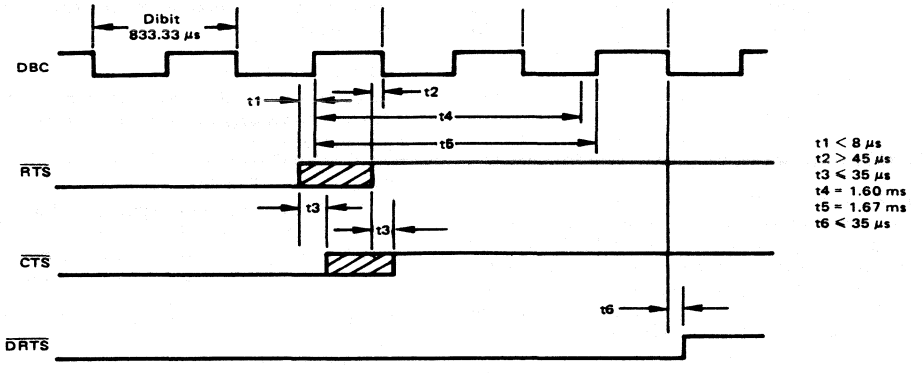
$\overline{\text{RTS}}$-$\overline{\text{CTS}}$ delay options are selected by the CTS1 and CTS2 inputs, and are stated as time delay interval t4. An $\overline{\text{RTS}}$ input signal synchronized about point A as shown and going low within the shaded region, will synchronize $\overline{\text{CTS}}$ with the positive transition of DBC (Dibit Clock), and delay t4 is measured with respect to the negative transition of $\overline{\text{RTS}}$.

$\overline{\text{RTS}}$ signals going low within the shaded region, but synchronized with the positive transition of DBC (point B), will result in the same $\overline{\text{CTS}}$ delay (t4). For this case the negative transition of $\overline{\text{CTS}}$ is synchronized with the negative transition of DBC with delay t4 measured with respect to the negative transition of $\overline{\text{RTS}}$.

$\overline{\text{DRTS}}$ will go low within 35 μs of the negative transition of $\overline{\text{RTS}}$ (shown as time t3). With the exception of the no-delay option, $\overline{\text{CTS}}$ will go low within 35 μs (t5) of the positive transition of DBC, following the t4 delay selected. This applies when $\overline{\text{RTS}}$ is synchronized to Point A as shown.

If $\overline{\text{RTS}}$ goes high and remains high ≥ 20 μs within time interval t4, a reset of the internal $\overline{\text{RTS}}$-$\overline{\text{CTS}}$ timer function will occur. If $\overline{\text{RTS}}$ goes high for less than 20 μs, the circuit may or may not respond to this momentary loss of the $\overline{\text{RTS}}$ signal.

FIGURE 5 – LOSS OF $\overline{RTS}$ TO $\overline{DRTS}$ DELAY

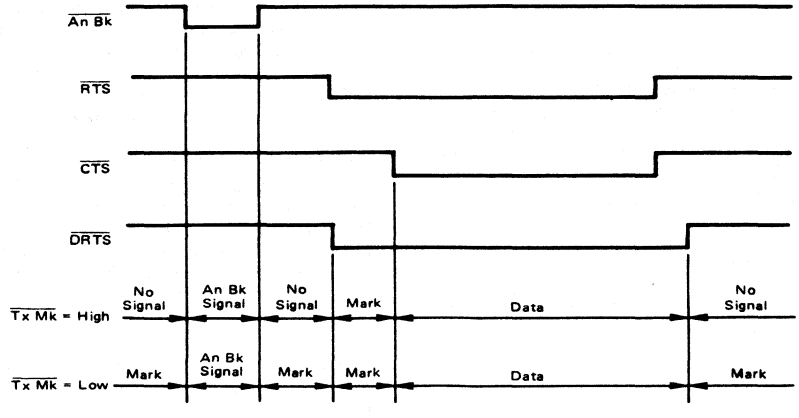


A positive transition of $\overline{RTS}$ after $\overline{CTS}$ has become active can result in different functional characteristics of the $\overline{CTS}$ and $\overline{DRTS}$ output signals, depending on the time duration that $\overline{RTS}$ remains inactive.

Under all conditions, $\overline{CTS}$ will go high within 35 µs (t3) following a positive transition of $\overline{RTS}$. If $\overline{RTS}$ goes high in the shaded region shown (i.e., synchronized to the positive transition of DBC) and remains high beyond the time interval defined as t5, then $\overline{DRTS}$ will go high within 35 µs (t6) of the next negative transition of DBC. If $\overline{RTS}$ were to go low after t5, the $\overline{RTS}$-$\overline{CTS}$ delay times given in Figure 4 will result.

If $\overline{RTS}$ goes high in the shaded region shown, and then returns low within time interval t4, the negative transition of $\overline{CTS}$ will follow within 35 µs, and DRTS will remain in the active or low state. Under these conditions, the normal $\overline{RTS}$-$\overline{CTS}$ delay times are not encountered when $\overline{RTS}$ is reactivated. If $\overline{RTS}$ goes low for less than 20 µs, the circuit may or may not respond to this momentary $\overline{RTS}$ input signal.

FIGURE 6 – TRANSMIT MARK CONTROL



(logic 0) condition on Tx Data with $\overline{TPE}$ activated causes the test pattern data to appear inverted at the data output. Random data applied to Tx Data with $\overline{TPE}$ activated causes the test pattern data to be scrambled (exclusive NORed) with the data, and the result appears at the data output.

The test pattern generator can be enabled only when $\overline{CTS}$ and $\overline{RTS}$ are logic 0. If $\overline{TPE}$ is activated outside this time interval, the previously stated $\overline{RTS}$-$\overline{CTS}$ and $\overline{RTS}$-$\overline{DRTS}$ delays shown in Figures 4 and 5 are not valid.

Data Rate Select (DRS)

The modulator can transmit at either 2400 bps or 1200 bps. Both data rates utilize an 1800 Hz carrier signal and employ phase shifting at 1200 Hz. The 2400 bps rate is obtained by encoding two bits of data into each phase shift. The 2400 Hz rate is selected by applying a logic 1 to the Data Rate Select lead. The 1200 Hz rate is selected by applying a logic 0 to DRS.

Phase Shift Select (PSS)

Option A (CCITT) or Option B (U.S.) phase shift can be selected for 2400 bps operation. The input data format and phase shift relationship for these two options are as follows:

Data	PSS = 0 Option A	PSS = 1 Option B
00	0°	+45°
01	+90°	+135°
11	+180°	+225°
10	+270°	+315°

For 1200 bps operation, Option C (CCITT) or Option D (U.S.) phase shift can be selected:

Data	PSS = 0 Option C	PSS = 1 Option D
0	+90°	+45°
1	+270°	+225°

Option C is selected by applying a logic 0 to the Phase Shift Select lead when the Data Rate Select lead is strapped for 1200 bps operation (logic 0). Option D is selected by applying a logic 1 to PSS with DRS at logic 0. The phase shifts shown are the difference in phase between the signal at the end of one dibit period and the new signal at the beginning of the next dibit.

Transmit Data (Tx Data)

Transmit Data is the serial binary information presented for DPSK modulation. A high level represents a mark. For timing, see Transmit Clock (Figure 4).

Transmit Clock (Tx Clk)

A 2400/1200 Hz Transmit Clock output is provided for the communication terminal. The Transmit Data signal is sampled on the positive transition of Transmit Clock. The Transmit Data to Transmit Clock setup and hold time requirements are shown in the Electrical Characteristics table and in Figure 2.

Dibit Clock (DBC)

A 1200 Hz Dibit Clock identifies the modulation timing. This signal goes negative less than 100 μs prior to the start of dibit modulation.

External Clock (Ex Clk)

A 2400/1200 Hz clock signal applied to the External Clock lead causes Transmit Clock to be synchronized with Ex Clk. This input must have an accuracy within ±0.005%.

When no transitions occur on this input, the internal clock provides the 2400/1200 Hz transmit timing signal. Fast synchronization of Tx Clk to Ex Clk is not provided on the chip. *When Ex Clk is not used it should be tied to either the logic 0 or logic 1 state.*

1.8432 MHz (Clk)

This input must be a square wave with rise and fall times of less than 40 ns and a 50 ±20% duty cycle. The clock accuracy must be within ±0.005%.

$\overline{\text{Answer Back}}$ ($\overline{\text{An Bk}}$)

A logic 0 level applied to $\overline{\text{Answer Back}}$ causes a 2025 Hz carrier to be generated on the modulator chip instead of a phase shifted 1800 Hz carrier. A logic 1 level applied to $\overline{\text{An Bk}}$ enables the modulator to generate the normal phase shifted 1800 Hz carrier signal, as shown in Figure 6. The time delay from a transition on $\overline{\text{An Bk}}$ to the appropriate signal at the modulator chip output is less than 2 ms.

Activation of $\overline{\text{An Bk}}$ (a logic 0) will disable all other operation modes including the $\overline{\text{Tx Mk}}$ function, and will reset $\overline{CTS}$ to an inactive state along with the $\overline{RTS}$-$\overline{CTS}$ internal timer. $\overline{\text{An Bk}}$ should therefore be activated only before initiating $\overline{RTS}$ or after loss of the $\overline{DRTS}$ output signal. The combination of a logic 0 on $\overline{\text{An Bk}}$ with a logic 0 on $\overline{TPE}$ is not used in normal system operation, and hence is used as a reset input during device test.

Digital Output (B0-B5)

These outputs are designed to interface with a six-bit digital-to-analog converter. The resultant signal out of the D/A is the differential phase shift keyed signal quantized at a 14.4 kHz rate. A low pass filter can then be used to smooth the data transitions. B0 is the least significant bit, and the positive level the active state.

Test Clock (TST)

A test signal input is provided to decrease test time of the chip. *In normal operation this input must be strapped low.*

MOTOROLA

MC6870, MC6871 series

actual size

Two-Phase Microprocessor Clocks
Designed to drive the Motorola MC6800 MPU

The *Functional Module* approach to data communications hardware design significantly decreases the time between the "idea" stage and the marketable product.

A fundamental building block in a modular microcomputer system is the 2-phase clock oscillator used to drive the microprocessor. Motorola is uniquely qualified to provide this building block because of expertise in the three relevant fields: oscillator design, quartz crystal technology, and thick film hybrid integrated circuit manufacturing.

This one-of-a-kind expertise has created several clocks designed to drive Motorola's MC6800 Microprocessor. This plug-in unit contains the crystal, the oscillator circuit, the NMOS and TTL drivers, and the waveshaping and interface circuitry; all the components necessary to provide the critical non-overlapping 2-phase waveforms used by the MC6800 MPU.

FEATURES

Clock Module — Each clock module requires only a single 5 volt power supply. The NMOS outputs can drive highly capacitive loads ranging from 80 pf to 160 pf and meet all MPU input waveshape and timing requirements.

Each TTL output signal leads the ϕ_2 NMOS so that additional system device delays can be accommodated. All TTL outputs are buffered so they can drive 5 TTL devices and maintain all output specifications.

Each module is crystal-controlled and is compensated for variations in temperature, voltage, and load. The standard frequency of each model is 1 MHz; however, other frequencies between 250 kHz and 2.5 MHz can be ordered.

Reliability — Decreased Component Count — Thick film hybrids offer a reliability advantage that comes primarily from reduced component count and therefore reduced interconnections. Further, the single hermetic seal on the hybrid package reduces the failure rate whereas in a discrete design a separate sealing process with an associated failure rate is needed for each component.

High Density Packaging — The hybrid MPU clock allows compact microcomputer design. It takes up only 1.34"x .840" space and has a seated height of .200".

Ruggedized Design — Maximum reliability at minimum cost is the result of combining three of Motorola's fields of experience: quartz crystal technology, clock oscillator design, and thick film hybrid integrated circuit manufacturing. Mass automated production techniques assure volume production. Gold plating of all crystals and Class 100 clean room processing testify that no short cuts are taken that might diminish reliability. Environmental testing proves the effectiveness of the rugged design for those applications in which shock and vibration are likely hazards.

Complete Process Control — Motorola is the only totally integrated manufacturer of quartz frequency control devices; full control of all processes from growing, sawing, lapping, and finishing quartz to combining it with other components into an electronic product — the MC6870A, MC6871A, and MC6871B MPU clocks.

Volume Production — Production facilities are oriented to mass automated production techniques. And, if required, capital for expansion is available to meet even greater requirements.

environmental specifications

Temperature Cycle: ±5 ppm max., 0 to 120°C, 3 cycles, 2 hrs. max. each, 25 ±2°C ref.

Shock: 1000G's 0.35 millisec, ½ sine wave, 3 shocks each plane

Vibration: 10-55 Hz, .060" D.A.; 55-2000Hz, 35 G's. Duration Time—12 Hours

Humidity: 85% Rel. Humidity, @ +85°C, 250 Hours

mechanical specifications

Gross Leak Test: All units 100% leak tested in de-ionized H_2O.

Hermetic Sealed Package: Mass spectrometer leak rate less than 2×10^{-8} atmos. cc/sec. of helium.

Seal Strengh: 20 lbs. max. force perpendicular to top and bottom.

Pin Material: Phosphor bronze, ¼ hard, Grade A .00003" thick gold flash finish.

Bend Test: Will withstand maximum bend of 90° reference to base for 1 bend.

Marking Ink: Epoxy, heat cured.

Solvent Resistance: Isopropyl Alcohol Tricholoroethane Freon TMC. No marking or seal destruction. Dipped 1 minute @ +25°C ±5°C in solvent.

Note: (1) Unit can be cleaned by only one type solvent listed.

Note: (2) Ultrasonic degreaser not to be used unless frequency and vibration of cleaner specified.

solderability specifications

Materials:

1.1 Solder: 60% tin and 40% lead.

1.2 Flux: The flux shall be 25 percent by weight of Grade WW rosin and 75 percent by weight of 99 percent isopropyl alcohol.

Procedure:

2.1 Solder Bath: The solder bath shall be maintained at 232 ±6°C.

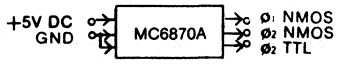
2.2 Solderability: Dip the terminals into the flux to the depth that is to be soldered or to a maximum depth of .025" from the body of the oscillator. Keep them in the flux for at least 5 seconds. Withdraw them from the flux. Dip them immediately into the molten solder to the same depth. Keep them in the molten solder for 2 to 5 seconds. Withdraw them and allow the solder to cool in air.

Requirements:

3.1 The terminals are considered solderable and acceptable for electrical connection purposes if 90 percent of the cold solder surface is uniform and free from breaks and pinholes. The other 10 percent of the cooled solder surface may show only pinholes, voids, or rough spots that are not concentrated in one area.

MC6870A

limited function microprocessor clock
250 kHz to 2.5 MHz



specifications

Rating	Symbol	Value	Unit
Supply Voltage	V_{cc}	5.00±5%	Vdc
Operating Temperature Range	T_A	0 to +70	°C
Storage Temperature	T_{stg}	−55 to +125	°C
Power Supply Drain (max.)	I_{pd}	100	mA

ELECTRICAL CHARACTERISTICS (V_{cc} = 5.0 ± 5%, V_{ss} = O, T_A = 0° to 70°C, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Frequency					
Operating Frequency	f_c	.250		2.5	MHz
Frequency stability (inclusive of calibration tolerance at +25°C, operating temperature, input voltage change, load change, aging, shock and vibration)			±.01		%
NMOS Outputs at 1.0 MHz Operation**					
Pulse Width (meas. at V_{cc} = −.3V dc level)	$T\emptyset_1H$ $T\emptyset_2H$	430 450			ns ns
Logic Levels	V_{OLC} V_{OHC}	V_{ss}-.1 V_{cc}-.3	— —	V_{ss}+.3 V_{cc}+.1	Vdc Vdc
Rise and Fall Times	t_r t_f	5 5	12 12	50 50	ns ns
*Overshoot/Undershoot Logic "1" Logic "0"	 V_{OS}	 V_{cc}-.5 V_{ss}-.5		 V_{cc}+.5 V_{ss}+.5	 Vdc Vdc
Pulse duration of any overshoot or undershoot	T_{OS}			40	ns
Period @ 0.3V dc Level	t_{cyc}		1.00		us
Edge Timing @ V_{cc}=0.3V dc	Tx	940			ns
NMOS Relationship @ +0.5V dc Level	t_{d1} t_{d2}	0 0		8.0	us
TTL Outputs					
In ref. to $\emptyset_2$ NMOS @ 0.3V dc					
$\emptyset_2$ TTL @ +1.4V dc	T_A T_H	15 10	30 25	45 40	ns ns
Logic Levels	V_{OH} V_{OL}	2.4	3.2 .3	 .4	Vdc Vdc
Rise and Fall Times .4V and 2.4V 2.4V and .4V	 t_r t_f			 15 15	 ns ns
Logic "0" Sink (/Gate)	I_{OL}			−1.6	mA
Logic "1" Source (/Gate)	I_{OH}			+40	uA
Current Output Shorted	I_{SC}	−18		−57	mA
Load					
NMOS—Load Capacity $\emptyset_1$, $\emptyset_2$	C_{NMOS}	80	120	160	pf
TTL—No. of Loads				5	ttl
TTL—Load Capacity	C_{TTL}			50	pf

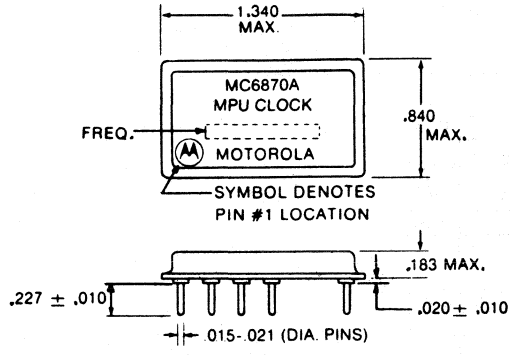
*Into specified test load

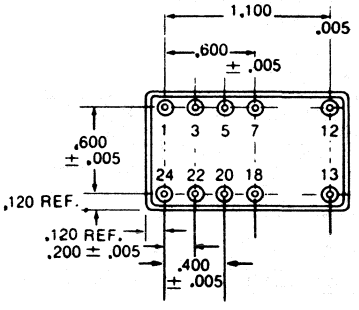
**Apply the following parameters for frequencies other than 1.0 MHz:
$T\phi_1H$=0.5 (P-140) ns
$T\phi_2H$=0.5 (P-100) ns
Tx=(P-60) ns
where P=desired period of operation in nanoseconds

PIN	CONNECTION
1	GND
3	NC
5	$\emptyset_2$ TTL
7	V_{cc} (+5VDC)
12	$\emptyset_2$ NMOS
13	$\emptyset_1$ NMOS
18	GND
20	NC
22	NC
24	NC

Note: All dimensions are in inches

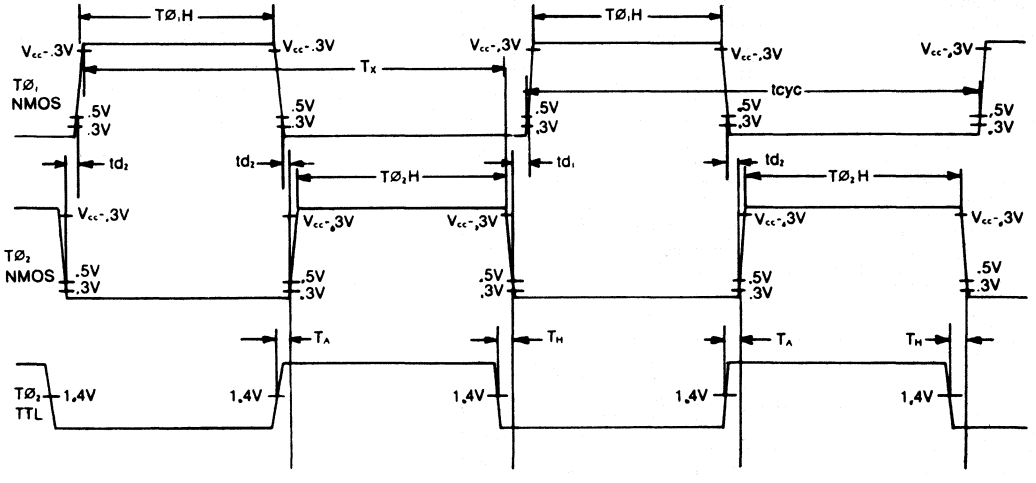
DIMENSIONS



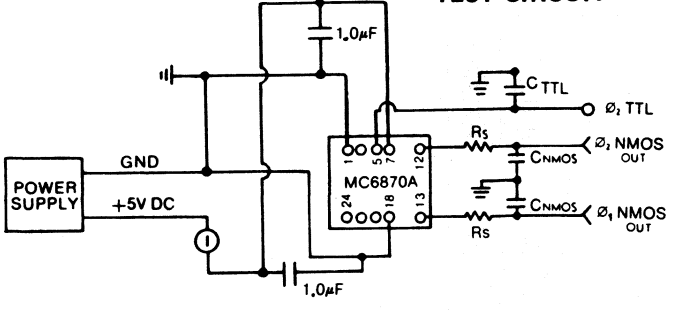


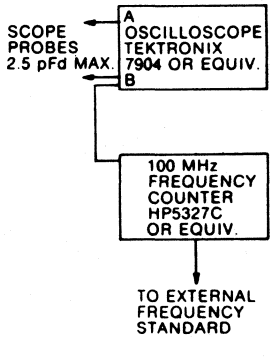
WAVEFORM TIMING

(ALL TIME IN NANOSECONDS)



TEST CIRCUIT





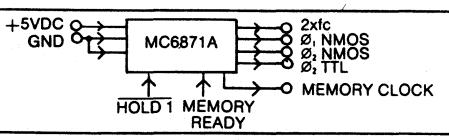
C_{TTL} — MAX CAPACITY 50 pF.

C_{NMOS} — 120 pF ± 40 pF IS THE SPECIFIED MAX. LOAD CAPACITANCE THAT SIMULATES THE MOTOROLA MC6800 MPU INPUT.

Rs — (22Ω) SIMULATES REAL PART OF MPU

MC6871A

full function microprocessor clock
850 kHz to 2.5 MHz



specifications

Rating	Symbol	Value	Unit
Supply Voltage	V_{cc}	5.00±5%	Vdc
Operating Temperature Range	T_A	0 to +70	°C
Storage Temperature	T_{stg}	−55 to +125	°C
Power Supply Drain (max.)	I_{pd}	100	mA

ELECTRICAL CHARACTERISTICS (V_{cc} = 5.0 ± 5%, V_{ss} = O, T_A = 0° to 70°C, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Frequency					
Operating Frequency	f_c	.850		2.5	MHz
Frequency stability (inclusive of calibration tolerance at +25°C, operating temperature, input voltage change, load change, aging, shock and vibration)			±.01		%
NMOS Outputs at 1.0 MHz Operation* **					
Pulse Width (meas. at V_{cc} = −.3V dc level)	$T\emptyset_1H$ $T\emptyset_2H$	430 450			ns ns
Logic Levels	V_{OLC} V_{OHC}	V_{ss}-.1 V_{cc}-.3	— —	V_{ss}+.3 V_{cc}+.1	Vdc Vdc
Rise and Fall Times	t_r t_f	5 5	12 12	50 50	ns ns
*Overshoot/Undershoot Logic "1" Logic "0"	 V_{OS}	 V_{cc}-.5 V_{ss}-.5		 V_{cc}+.5 V_{ss}+.5	 Vdc Vdc
Pulse duration of any overshoot or undershoot	T_{OS}			40	ns
Period @ 0.3V dc Level	t_{cyc}		1.00		us
Edge Timing @ V_{cc}=0.3V dc	Tx	940			ns
NMOS Relationship @ +0.5V dc Level	t_{d1} t_{d2}	0 0		 8.0	 us
TTL Outputs					
In ref. to $\emptyset_2$ NMOS @ 0.3V dc					
$\emptyset_2$ TTL @ 1.4V dc	T_A T_H	15 10	30 25	45 40	ns ns
Memory Clock @ 1.4V dc	T_C T_J	30 20	50 40	70 60	ns ns
2xfc @ 1.4V dc	T_B	40	80	120	ns
Logic Levels	V_{OH} V_{OL}	2.4	3.2 .3	 .4	Vdc Vdc
Rise and Fall Times .4V and 2.4V 2.4V and .4V	 t_r t_f			 15 15	 ns ns
Logic "0" Sink (/Gate)	I_{OL}			−1.6	mA
Logic "1" Source (/Gate)	I_{OH}			+40	uA
Current Output Shorted	I_{SC}	−18		−57	mA
Load					
NMOS—Load Capacity $\emptyset_1$, $\emptyset_2$	C_{NMOS}	80	120	160	pf
TTL—No. of Loads				5	ttl
TTL—Load Capacity	C_{TTL}			50	pf
Logic Inputs ("0" Level Applies HOLD or MEMORY READY)**					
Holds Ø1 NMOS 'High', $\emptyset_2$ NMOS 'Low', $\emptyset_2$ TTL 'Low'	$\overline{\text{HOLD 1}}$	−.2		+.4	Vdc
Holds Ø1 NMOS 'Low', $\emptyset_2$ NMOS 'High', $\emptyset_2$ TTL 'High', and MEMORY CLOCK 'High'	MEMORY READY	−.2		+.4	Vdc

*Into specified test load

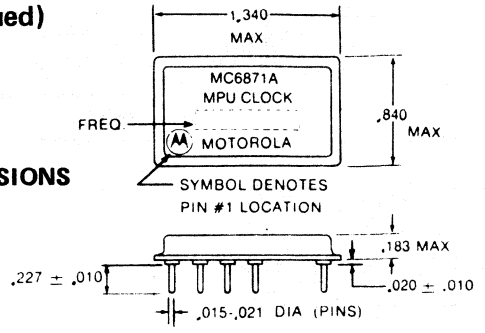
**Must be externally held at "1" level (2.4V min., 5.0V max.) if not used

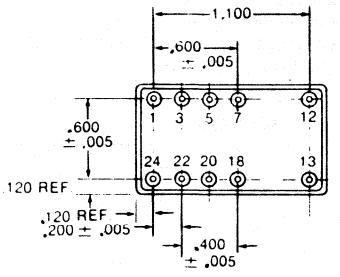
***Apply the following parameters for frequencies other than 1 MHz:
$T\emptyset_1H$=0.5 (P-140) ns
$T\emptyset_2H$=0.5 (P-100) ns
Tx=(P-60) ns
where P=desired period of operation in nanoseconds

PIN	CONNECTION
1	GND
3	MEMORY CLOCK
5	$\emptyset_2$ TTL
7	V_{cc} (+5VDC)
12	$\emptyset_2$ NMOS
13	$\emptyset_1$ NMOS
18	GND
20	$\overline{\text{HOLD 1}}$
22	MEMORY READY
24	2xfc

Note All dimensions are in inches

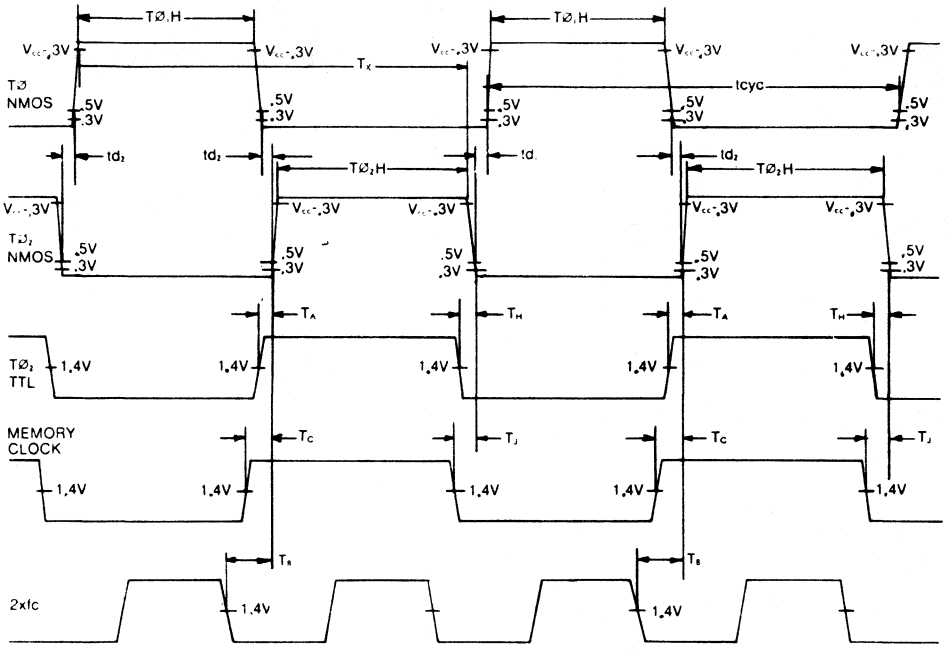
DIMENSIONS



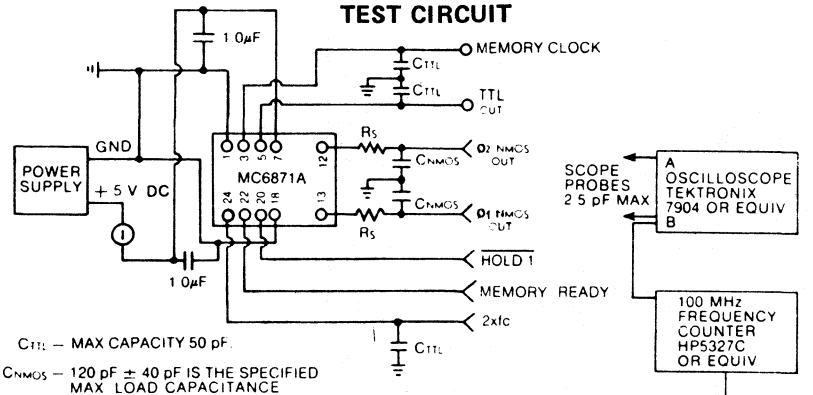


WAVEFORM TIMING

(ALL TIME IN NANOSECONDS)



TEST CIRCUIT



CTTL – MAX CAPACITY 50 pF

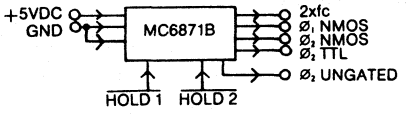
CNMOS – 120 pF ± 40 pF IS THE SPECIFIED MAX LOAD CAPACITANCE THAT SIMULATES THE MOTOROLA MC6800 MPU INPUT

RS – (22Ω) SIMULATES REAL PART OF MPU

*HOLD AND MEMORY READY MUST BE EXTERNALLY HELD AT 1 LEVEL (2.4VDC MIN 5.0VDC MAX) WHEN NOT USED

MC6871B

alternate function microprocessor clock
250 kHz to 2.5 MHz



specifications

Rating	Symbol	Value	Unit
Supply Voltage	V_{cc}	5.00±5%	Vdc
Operating Temperature Range	T_A	0 to +70	°C
Storage Temperature	T_{stg}	−55 to +125	°C
Power Supply Drain (max.)	I_{pd}	100	mA

ELECTRICAL CHARACTERISTICS (V_{cc} = 5.0 ± 5%, V_{ss} = O,T_A = 0° to 70°C, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Frequency					
Operating Frequency Frequency stability (inclusive of calibration tolerance at +25°C, operating temperature, input voltage change, load change, aging, shock and vibration)	f_c	.250	±.01	2.5	MHz %
NMOS Outputs at 1.0 MHz Operation* **					
Pulse Width (meas. at V_{cc} = −.3V dc level)	$T\emptyset_1H$ $T\emptyset_2H$	430 450			ns ns
Logic Levels	V_{OLC} V_{OHC}	V_{ss}-.1 V_{cc}-.3	— —	V_{ss}+.3 V_{cc}+.1	Vdc Vdc
Rise and Fall Times	t_r t_f	5 5	12 12	50 50	ns ns
*Overshoot/Undershoot Logic "1" Logic "0"	V_{OS}	 V_{cc}-.5 V_{ss}-.5		 V_{cc}+.5 V_{ss}+.5	 Vdc Vdc
Pulse duration of any overshoot or undershoot	T_{OS}			40	ns
Period @ 0.3V dc Level	t_{cyc}		1.00		us
Edge Timing @ V_{cc}=0.3V dc	Tx	940			ns
NMOS Relationship @ +0.5V dc	t_{d1} t_{d2}	0 0		 8.0	 us
TTL Outputs					
In ref. to $\emptyset_2$ NMOS @ 0.3V dc					
$\emptyset_2$ TTL @ 1.4V dc	T_A T_H	15 10	30 25	45 40	ns ns
$\emptyset_2$ Ungated @ 1.4V dc	T_C T_J	30 20	50 40	70 60	ns ns
2xfc @ 1.4V dc	T_B	40	80	120	ns
Logic Levels	V_{OH} V_{OL}	2.4	3.2 .3	 .4	Vdc Vdc
Rise and Fall Times .4V and 2.4V 2.4V and .4V	 t_r t_f			 15 15	 ns ns
Logic "0" Sink (/Gate)	I_{OL}			−1.6	mA
Logic "1" Source (/Gate)	I_{OH}			+40	uA
Current Output Shorted	I_{SC}	−18		−57	mA
Load					
NMOS—Load Capacity $\emptyset_1$, $\emptyset_2$	C_{NMOS}	80	120	160	pf
TTL—No. of Loads				5	ttl
TTL—Load Capacity	C_{TTL}			50	pf
Logic Inputs ("0" Level applies HOLD)**					
Holds $\emptyset_1$ NMOS 'High', $\emptyset_2$ NMOS 'Low', $\emptyset_2$ TTL 'Low'	$\overline{HOLD\ 1}$	−.2		+.4	Vdc
Holds $\emptyset_1$ NMOS 'Low', $\emptyset_2$ NMOS 'High', $\emptyset_2$ TTL 'High'	$\overline{HOLD\ 2}$	−.2		+.4	Vdc

*Into specified test load

**Must be externally held at "1" level (2.4V min., 5.0V max.) if not used

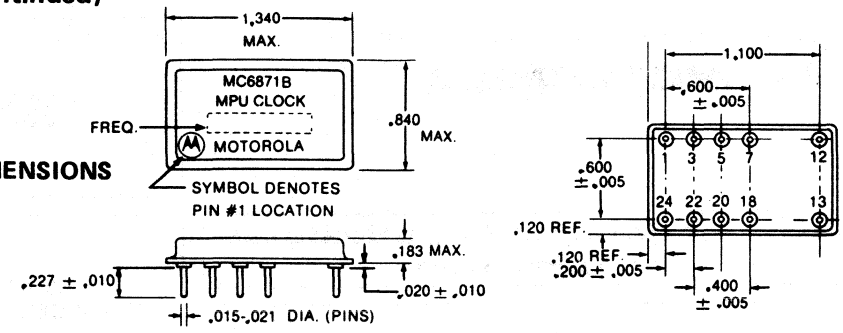
***Apply the following parameters for frequencies other than 1 MHz:
$T\emptyset_1H$=0.5 (P-140) ns
$T\emptyset_2H$=0.5 (P-100) ns
Tx=(P-60) ns
where P=desired period of operation in nanoseconds

PIN	CONNECTION
1	GND
3	$\emptyset_2$ TTL UNGATED
5	$\emptyset_2$ TTL
7	V_{cc} (+5VDC)
12	$\emptyset_2$ NMOS
13	$\emptyset_1$ NMOS
18	GND
20	$\overline{HOLD\ 1}$
22	$\overline{HOLD\ 2}$
24	2xfc

Note: 4xfc available on request

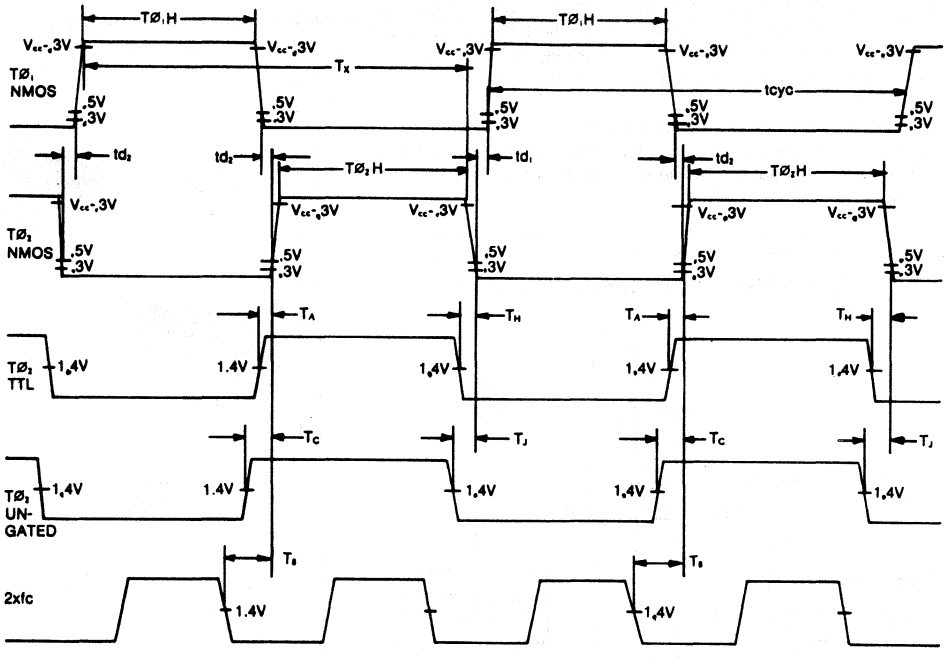
Note. All dimensions are in inches

DIMENSIONS

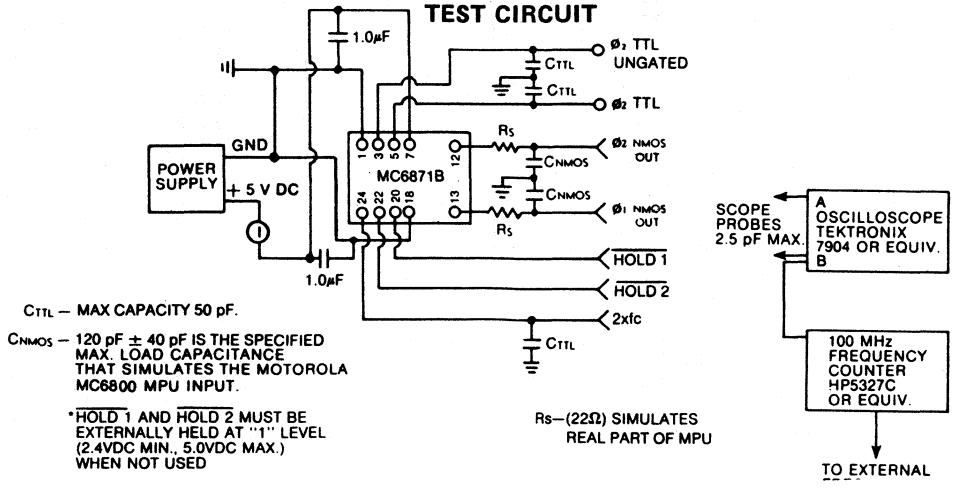


WAVEFORM TIMING

ALL TIME IN NANOSECONDS.



TEST CIRCUIT



CTTL — MAX CAPACITY 50 pF.

CNMOS — 120 pF ± 40 pF IS THE SPECIFIED MAX. LOAD CAPACITANCE THAT SIMULATES THE MOTOROLA MC6800 MPU INPUT.

*$\overline{\text{HOLD 1}}$ AND $\overline{\text{HOLD 2}}$ MUST BE EXTERNALLY HELD AT "1" LEVEL (2.4VDC MIN., 5.0VDC MAX.) WHEN NOT USED

RS—(22Ω) SIMULATES REAL PART OF MPU

MOTOROLA

MC6875

Specifications and Applications Information

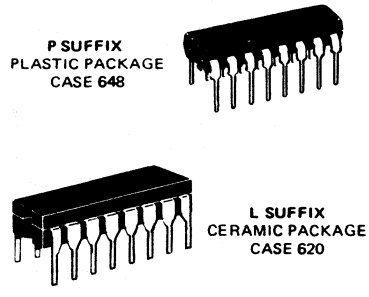
M6800 CLOCK GENERATOR

Intended to supply the non-overlapping $\phi 1$ and $\phi 2$ clock signals required by the microprocessor, this clock generator is compatible with 1.0, 1.5, and 2.0 MHz versions of the MC6800. Both the oscillator and high capacitance driver elements are included along with numerous other logic accessory functions for easy system expansion.

Schottky technology is employed for high speed and PNP-buffered inputs are employed for NMOS compatibility. A single +5 V power supply, and a crystal or RC network for frequency determination are required.

M6800 TWO-PHASE CLOCK GENERATOR/DRIVER

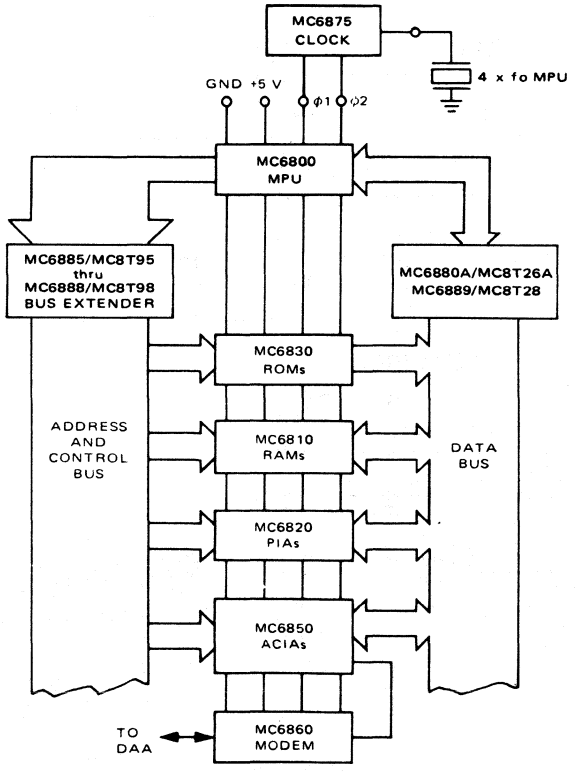
SCHOTTKY MONOLITHIC INTEGRATED CIRCUIT



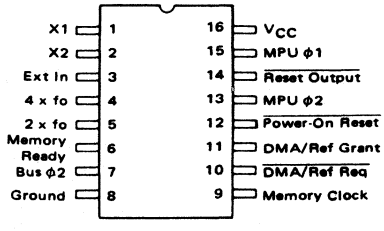
P SUFFIX
PLASTIC PACKAGE
CASE 648

L SUFFIX
CERAMIC PACKAGE
CASE 620

Typical MPU System with Bus Extenders



PIN CONNECTIONS



ORDERING INFORMATION		
Device	Temperature Range	Package
MC6875L	0 to 70°C	Ceramic Dip
MC6875P	0 to 70°C	Plastic DIP

ABSOLUTE MAXIMUM RATINGS (Unless otherwise noted T_A = 25°C.)

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	+7.0	Vdc
Input Voltage	V_I	+5.5	Vdc
Operating Ambient Temperature Range	T_A	0 to +70	°C
Storage Temperature Range Ceramic Package Plastic Package	T_{stg}	 -65 to +150 -55 to +125	°C
Operating Junction Temperature Ceramic Package Plastic Package	T_J	 175 150	°C

RECOMMENDED OPERATING CONDITIONS

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	+4.75 to +5.25	Vdc
Operating Ambient Temperature Range	T_A	0 to +70	°C

ELECTRICAL CHARACTERISTICS

(Unless otherwise noted specifications apply over recommended power supply and temperature ranges. Typical values measured at V_{CC} = 5.0 V and T_A = 25°C.)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage — High Logic State					
MPU ϕ1 and ϕ2 Outputs					V
(V_{CC} = 4.75 V, I_{OHM} = -200 µA)	V_{OHM}	V_{CC} - 0.6	–	–	
(V_{CC} = 5.25 V, I_{OHMK} = +5.0 mA)	V_{OHMK}	–	–	V_{CC} + 1.0	
Bus ϕ2 Output					V
(V_{CC} = 4.75 V, I_{OHB} = -10 mA)	V_{OHB}	2.4	–	–	
(V_{CC} = 5.25 V, I_{OHBK} = +5.0 mA)	V_{OHBK}	–	–	V_{CC} + 1.0	
4 x fo Output					V
(V_{CC} = 4.75 V, V_{IH} = 2.0 V, I_{OH4X} = -500 µA)	V_{OH4X}	2.4	–	–	
2 x fo, DMA/Refresh Grant and Memory Clock Outputs (V_{CC} = 4.75 V, I_{OH} = -500 µA)	V_{OH}	2.4	–	–	V
$\overline{\text{Reset}}$ Output (V_{CC} = 4.75 V, V_{IH} = 3.3 V, $I_{OH\overline{R}}$ = -100 µA)	$V_{OH\overline{R}}$	2.4	–	–	V
Output Voltage — Low Logic State					
MPU ϕ1 and ϕ2 Outputs					V
(V_{CC} = 4.75 V, I_{OLM} = +200 µA)	V_{OLM}	–	–	0.4	
(V_{CC} = 4.75 V, I_{OLMK} = -5.0 mA)	V_{OLMK}	–	–	-1.0	
Bus ϕ2 Output					V
(V_{CC} = 4.75 V, I_{OLB} = +48 mA)	V_{OLB}	–	–	0.5	
(V_{CC} = 4.75 V, I_{OLBK} = -5.0 mA)	V_{OLBK}	–	–	-1.0	
4 x fo Output					V
(V_{CC} = 4.75 V, V_{IL} = 0.8 V, I_{OL4X} = 16 mA)	V_{OL4X}	–	–	0.5	
2 x fo, DMA/Refresh Grant and Memory Clock Outputs (V_{CC} = 4.75 V, I_{OL} = 16 mA)	V_{OL}	–	–	0.5	V
$\overline{\text{Reset}}$ Output (V_{CC} = 4.75 V, V_{IL} = 0.8 V, $I_{OL\overline{R}}$ = 3.2 mA)	$V_{OL\overline{R}}$	–	–	0.5	V
Input Voltage — High Logic State Ext. In, Memory Ready and $\overline{\text{DMA/Refresh Request}}$ Inputs	V_{IH}	2.0	–	–	V
Input Voltage — Low Logic State Ext. In, Memory Ready and $\overline{\text{DMA/Refresh Request}}$ Inputs	V_{IL}	–	–	0.8	V
Input Thresholds — $\overline{\text{Power-On Reset}}$ Input (See Figure 2)					V
Output Low to High	V_{ILH}	–	2.8	3.6	
Output High to Low	V_{IHL}	0.8	1.4	–	
Input Clamp Voltage (V_{CC} = 4.75 V, I_{IC} = -5.0 mA)	V_{IC}	–	–	-1.0	V
Input Current — High Logic State					
Ext. In, Memory Ready and $\overline{\text{DMA/Refresh Request}}$ Inputs (V_{CC} = 4.75 V, V_{IH} = 5.0 V)	I_{IH}	–	–	25	µA
$\overline{\text{Power-On Reset}}$ (V_{CC} = 5.0 V, $V_{IH\overline{R}}$ = 5.0 V)	$I_{IH\overline{R}}$	–	–	50	µA
Input Current — Low Logic State					
Ext. In, Memory Ready and $\overline{\text{DMA/Refresh Request}}$ Inputs (V_{CC} = 5.25 V, V_{IL} = 0.5 V)	I_{IL}	–	–	-250	µA
$\overline{\text{Power-On Reset}}$ Input (V_{CC} = 5.25 V, V_{IL} = 0.5 V)	$I_{IL\overline{R}}$	–	–	-250	µA

OPERATING DYNAMIC POWER SUPPLY CURRENT

Characteristic	Symbol	Min	Typ	Max	Unit
Power Supply Currents (V_{CC} = 5.25 V, f_{osc} = 8.0 MHz, V_{IL} = 0 V, V_{IH} = 3.0 V) Normal Operation (Memory Ready and $\overline{\text{DMA/Refresh Request}}$ Inputs at High Logic State)	I_{CCN}	–	–	150	mA
Memory Ready Stretch Operation (Memory Ready Input at Low Logic State; $\overline{\text{DMA/Refresh Request}}$ Input at High Logic State)	I_{CCMR}	–	–	135	mA
$\overline{\text{DMA/Refresh Request}}$ Stretch Operation (Memory Ready Input at High Logic State; $\overline{\text{DMA/Refresh Request}}$ Input at Low Logic State)	I_{CCDR}	–	–	135	mA

SWITCHING CHARACTERISTICS

(These specifications apply whether the Internal Oscillator (see Figure 9) or an External Oscillator is used (see Figure 10). Typical values measured at V_{CC} = 5.0 V, T_A = 25°C, fo = 1.0 MHz (see Figure 8).

Characteristic	Symbol	Min	Typ	Max	Unit
MPU φ1 AND φ2 CHARACTERISTICS					
Output Period (Figure 3)	t_o	500	–	–	ns
Pulse Width (Figure 3)	t_{PWM}				ns
(fo = 1.0 MHz)		400	–	–	
(fo = 1.5 MHz)		230	–	–	
(fo = 2.0 MHz)		180	–	–	
Total Up Time (Figure 3)	t_{UPM}				ns
(fo = 1.0 MHz)		900	–	–	
(fo = 1.5 MHz)		600	–	–	
(fo = 2.0 MHz)		440	–	–	
Delay Time Referenced to Output Complement (Figure 3) Output High to Low State (Clock Overlap at 1.0 V)	t_{PLHM}	0	–	–	ns
Delay Times Referenced to 2 x fo (Figure 4 MPU φ2 only)					
Output Low to High Logic State	t_{PLHM2X}	–	–	85	ns
Output High to Low Logic State	t_{PHLM2X}	–	–	70	ns
Transition Times (Figure 3)					
Output Low to High Logic State	t_{TLHM}	–	–	25	ns
Output High to Low Logic State	t_{THLM}	–	–	25	ns
BUS φ2 CHARACTERISTICS					
Pulse Width – Low Logic State (Figure 4)	t_{PWLB}				ns
(fo = 1.0 MHz)		430	–	–	
(fo = 1.5 MHz)		280	–	–	
(fo = 2.0 MHz)		210	–	–	
Pulse Width – High Logic State	t_{PWHB}				ns
(fo = 1.0 MHz)		450	–	–	
(fo = 1.5 MHz)		295	–	–	
(fo = 2.0 MHz)		235	–	–	
Delay Times – (Referenced to MPU φ1) (Figure 4)					
Output Low to High Logic State	t_{PLHBM1}				ns
(fo = 1.0 MHz)		480	–	–	
(fo = 1.5 MHz)		320	–	–	
(fo = 2.0 MHz)		240	–	–	
Output High to Low Logic State	t_{PHLBM1}				
(C_L = 300 pF)		–	–	25	
(C_L = 100 pF)		–	–	20	
Delay Times (Referenced to MPU φ2) (Figure 4)					
Output Low to High Logic State	t_{PLHBM2}	-30	–	+25	ns
Output High to Low Logic State	t_{PHLBM2}	0	–	+40	ns
Transition Times (Figure 4)					
Output Low to High Logic State	t_{TLHB}	–		20	ns
Output High to Low Logic State	t_{THLB}	–		20	ns

SWITCHING CHARACTERISTICS (continued)

Characteristic	Symbol	Min	Typ	Max	Unit
MEMORY CLOCK CHARACTERISTICS					
Delay Times (Referenced to MPU $\phi 2$) (Figure 4)					
Output Low to High Logic State	t_{PLHCM}	-50	–	+25	ns
Output High to Low Logic State	t_{PHLCM}	0	–	+40	ns
Delay Times (Referenced to 2 x fo) (Figure 4)					
Output Low to High Logic State	t_{PLHC2X}	–	–	65	ns
Output High to Low Logic State	t_{PHLC2X}	–	–	85	ns
Transition Times (Figure 4)					
Output Low to High State	t_{TLHC}	–	–	25	ns
Output High to Low State	t_{THLC}	–	–	25	ns
2 x fo CHARACTERISTICS					
Delay Times (Referenced to 4 x fo) (Figure 4)					
Output Low to High Logic State	t_{PLH2X}	–	–	50	ns
Output High to Low Logic State	t_{PHL2X}	–	–	65	ns
Delay Time (Referenced to MPU $\phi 1$) (Figure 4)					
Output High to Low Logic State	$t_{PHL2XM1}$				ns
(fo = 1.0 MHz)		365	–	–	
(fo = 1.5 MHz)		220	–	–	
Transition Times (Figure 4)					
Output Low to High Logic State	t_{TLH2X}	–	–	25	ns
Output High to Low Logic State	t_{THL2X}	–	–	25	ns
4 x fo CHARACTERISTICS					
Delay Times (Referenced to Ext. In) (Figure 4)					
Output Low to High Logic State	t_{PLH4X}	–	–	50	ns
Output High to Low Logic State	t_{PHL4X}	–	–	30	ns
Transition Time (Figure 4)					
Output Low to High Logic State	t_{TLH4X}	–	–	25	ns
Output High to Low Logic State	t_{THL4X}	–	–	25	ns
MEMORY READY CHARACTERISTICS					
Set-Up Times (Figure 5)					
Low Input Logic State	t_{SMRL}	55	–	–	ns
High Input Logic State	t_{SMRH}	75	–	–	ns
Hold Time (Figure 5)					
Low Input Logic State	t_{HMRL}	10	–	–	ns
$\overline{\text{DMA/REFRESH REQUEST}}$ CHARACTERISTICS					
Set-Up Times (Figure 6)					
Low Input Logic State	t_{SDRL}	65	–	–	ns
High Input Logic State	t_{SDRH}	75	–	–	ns
Hold Time (Figure 6)					
Low Input Logic State	t_{HDRL}	10	–	–	ns
DMA/REFRESH GRANT CHARACTERISTICS					
Delay Time Referenced to Memory Clock (Figure 6)					
Output Low to High Logic State	t_{PLHG}	-15	–	+25	ns
Output High to Low Logic State	t_{PHLG}	-25	–	+15	ns
Transition Times (Figure 6)					
Output Low to High Logic State	t_{TLHG}	–	–	25	ns
Output High to Low Logic State	t_{THLG}	–	–	25	ns
$\overline{\text{RESET}}$ CHARACTERISTICS					
Delay Time Referenced to $\overline{\text{Power-On Reset}}$ (Figure 7)					
Output Low to High Logic State	$t_{PLH\overline{R}}$	–	–	1000	ns
Output High to Low Logic State	$t_{PHL\overline{R}}$	–	–	250	ns
Transition Times (Figure 7)					
Output Low to High Logic State	$t_{TLH\overline{R}}$	–	–	100	ns
Output High to Low Logic State	$t_{THL\overline{R}}$	–	–	50	ns

DESCRIPTION OF PIN FUNCTIONS

- 4 x fo – A free running oscillator at four times the MPU clock rate useful for a system sync signal.
- 2 x fo – A free running oscillator at two times the MPU clock rate.
- $\overline{\text{DMA/REF REQ}}$ – An asynchronous input used to freeze the MPU clocks in the $\phi 1$ high, $\phi 2$ low state for dynamic memory refresh or cycle steal DMA (Direct Memory Access).
- REF GRANT – A synchronous output used to synchronize the refresh or DMA operation to the MPU.
- MEMORY READY – An asynchronous input used to freeze the MPU clocks in the $\phi 1$ low, $\phi 2$ high state for slow memory interface.
- MPU $\phi 1$, MPU $\phi 2$ – Capable of driving the $\phi 1$ and $\phi 2$ inputs on two MC6800s.
- BUS $\phi 2$ – An output nominally in phase with MPU $\phi 2$ having MC8726A type drive capability.
- MEMORY CLOCK – An output nominally in phase with MPU $\phi 2$ which free runs during a refresh request cycle.
- $\overline{\text{POWER-ON RESET}}$ – A Schmitt trigger input which controls $\overline{\text{Reset}}$. A capacitor to ground is required to set the desired time constant. Internal 50 k resistor to V_{CC}. See General Design Suggestions for Manual Reset Operation.
- $\overline{\text{RESET}}$ – An output to the MPU and I/O devices.
- X1, X2 – Provision to attach a series resonant crystal or RC network.
- EXT IN – Allows driving by an external TTL signal to synchronize the MPU to an external system.

FIGURE 1 – BLOCK DIAGRAM

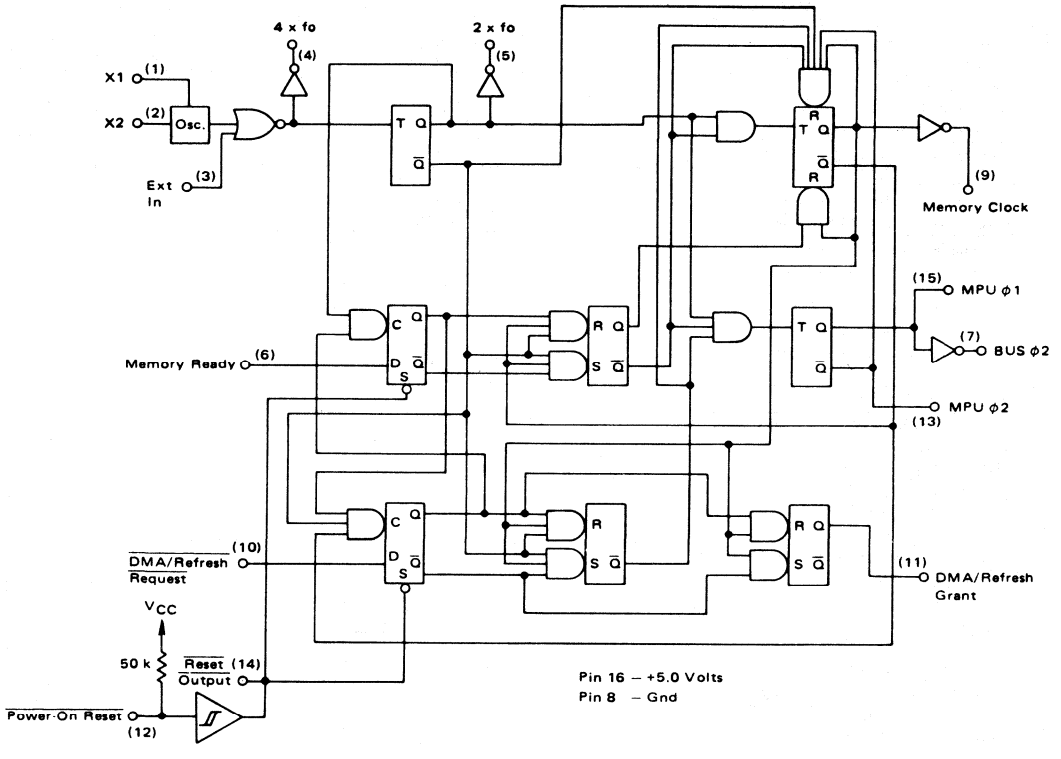


FIGURE 2 – TYPICAL HYSTERESIS CHARACTERISTIC OF $\overline{\text{RESET}}$ FUNCTION

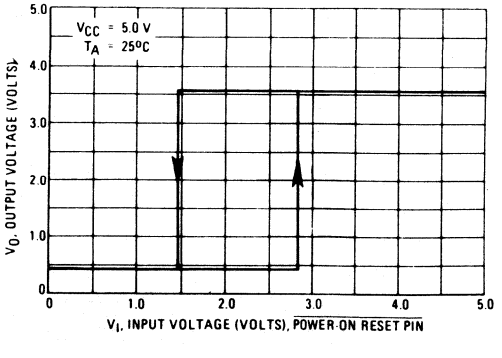
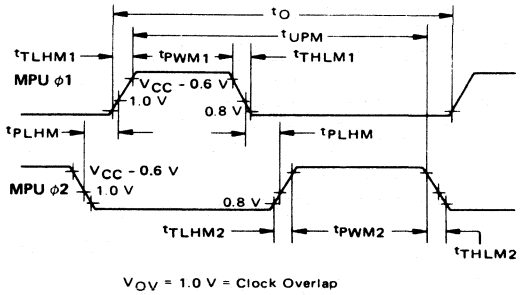


FIGURE 3 – TIMING DIAGRAM FOR MPU φ1 AND φ2



V_{OV} = 1.0 V = Clock Overlap measurement point

FIGURE 4 – TIMING DIAGRAM FOR NON-STRETCHED OPERATION
(Memory Ready and $\overline{\text{DMA/Refresh Request}}$ held high continuously)
Ext. In Input Voltage: 0 V to 3.0 V, f = 8.0 MHz, Duty Cycle = 50%, $t_{TLHEX} = t_{THLEX} = 5.0$ ns

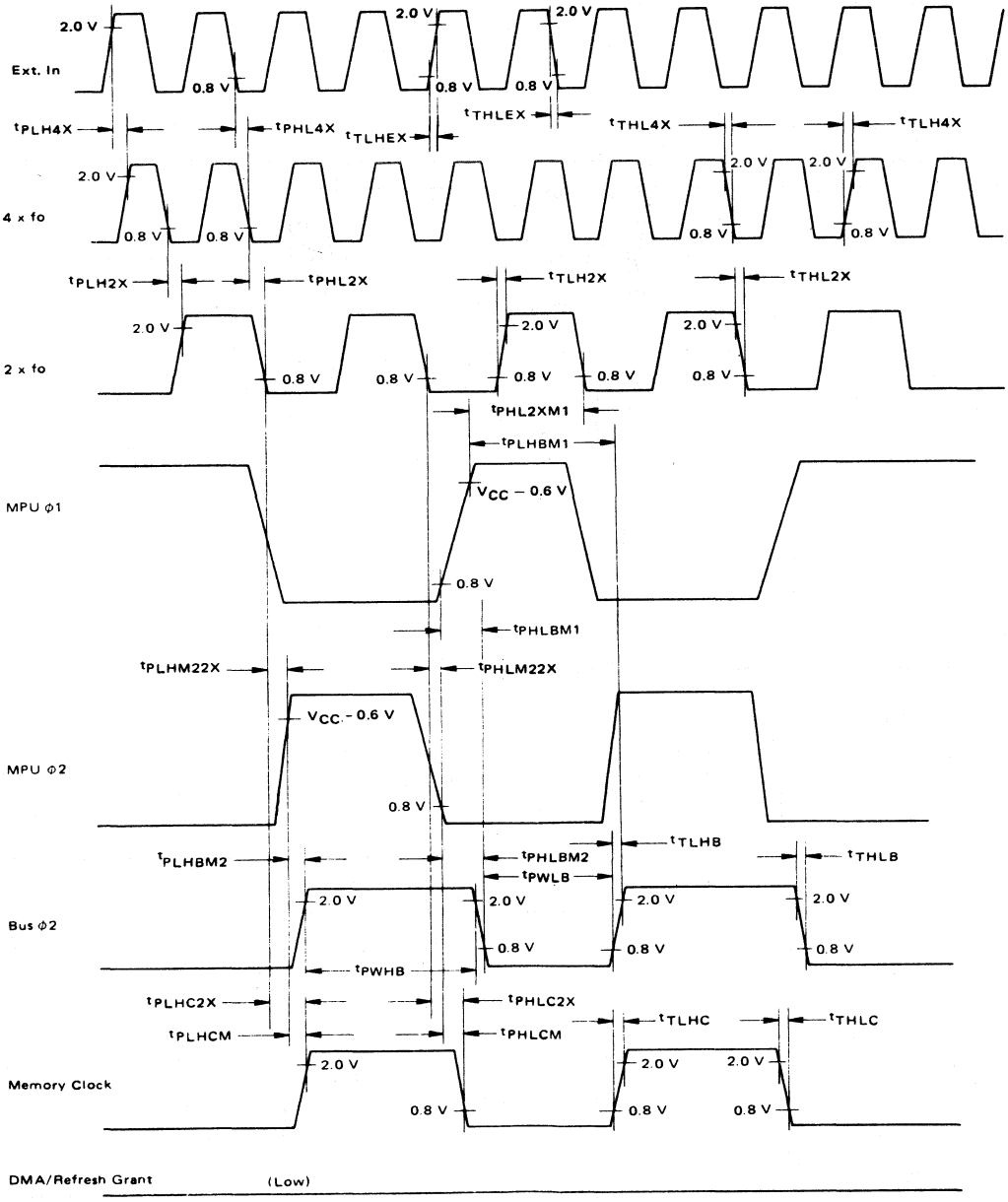


FIGURE 5 – TIMING DIAGRAM FOR MEMORY READY STRETCH OPERATION
(Minimum Stretch Shown)
Input Voltage: 3.0 to 0 V, $t_{THLMR} = t_{TLHMR} = 5.0$ ns

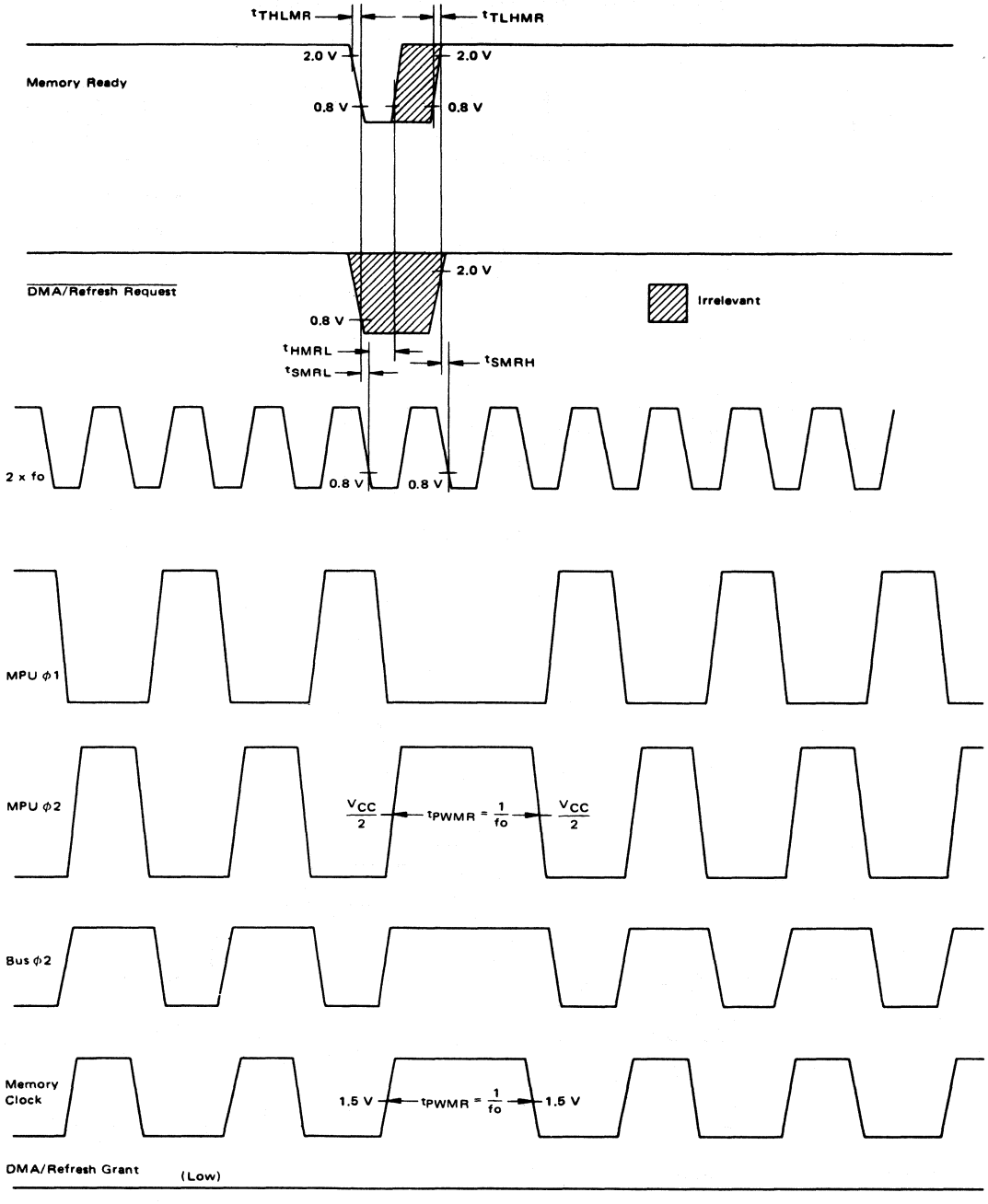


FIGURE 6 – TIMING DIAGRAM FOR $\overline{\text{DMA/REFRESH REQUEST}}$ STRETCH OPERATION
(Minimum Stretch Shown)
Input Voltage: 3.0 to 0 V, $t_{THLDR} = t_{TLHDR} = 5.0$ ns

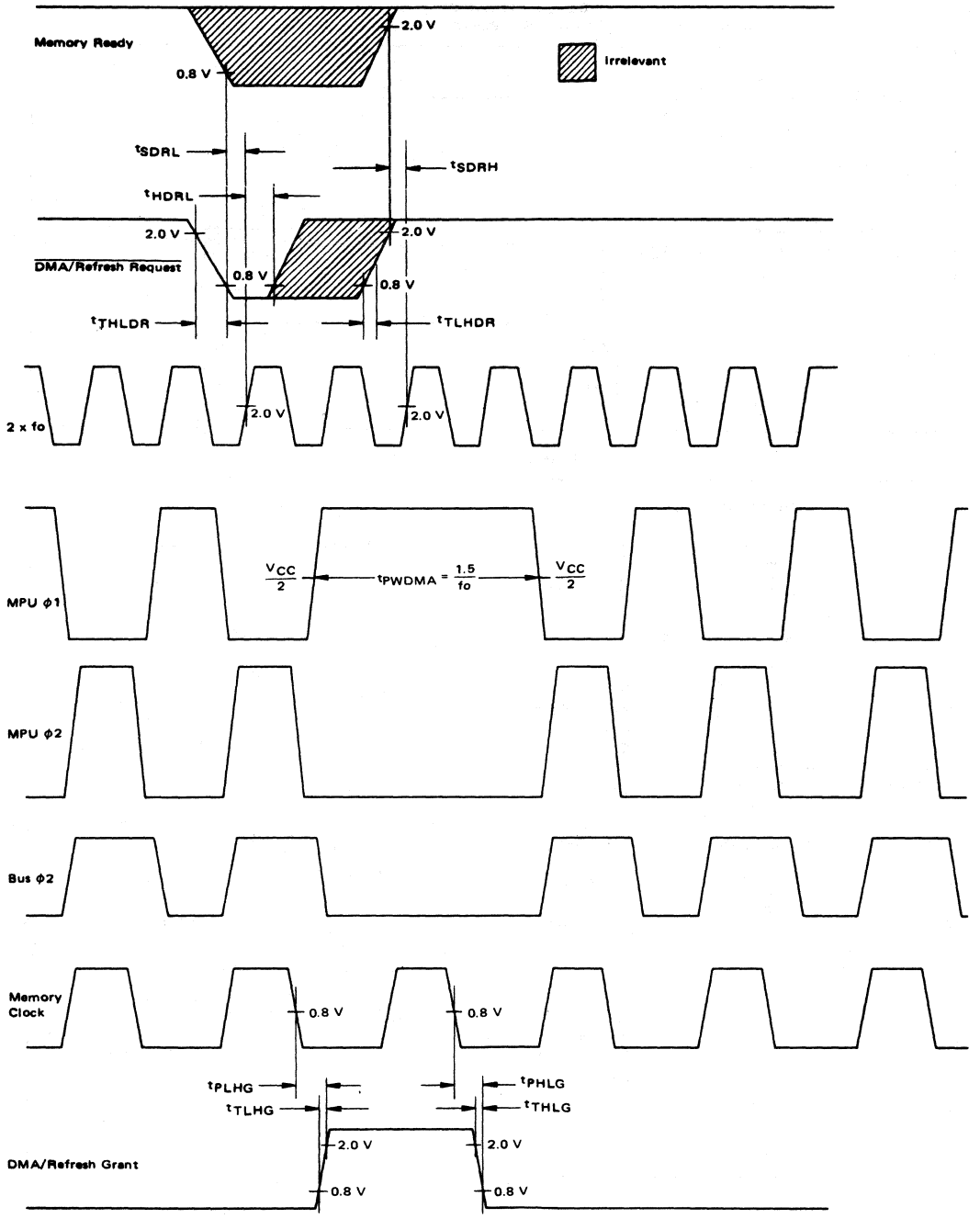


FIGURE 7 – $\overline{\text{POWER ON RESET}}$

Input Voltage: 0 to 5.0 V, f = 100 kHz – Pulse Width = 1.0 μs, $t_{TLH} = t_{THL}$ = 25 ns

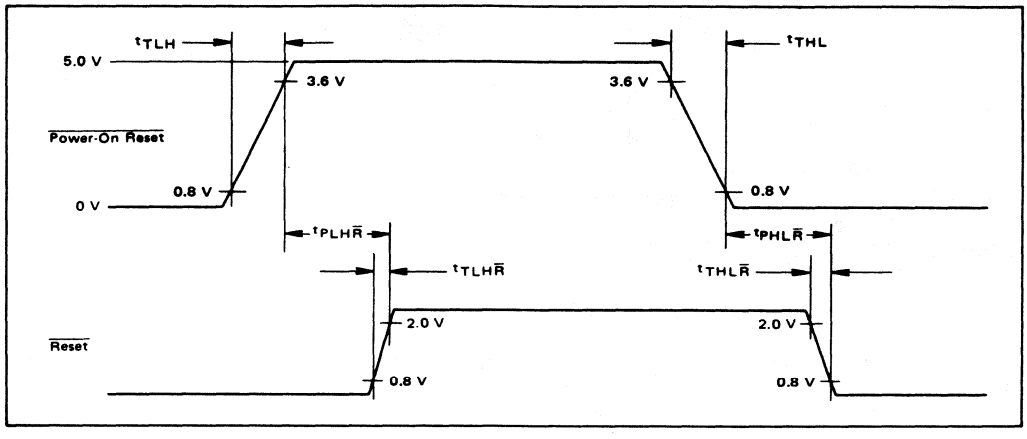
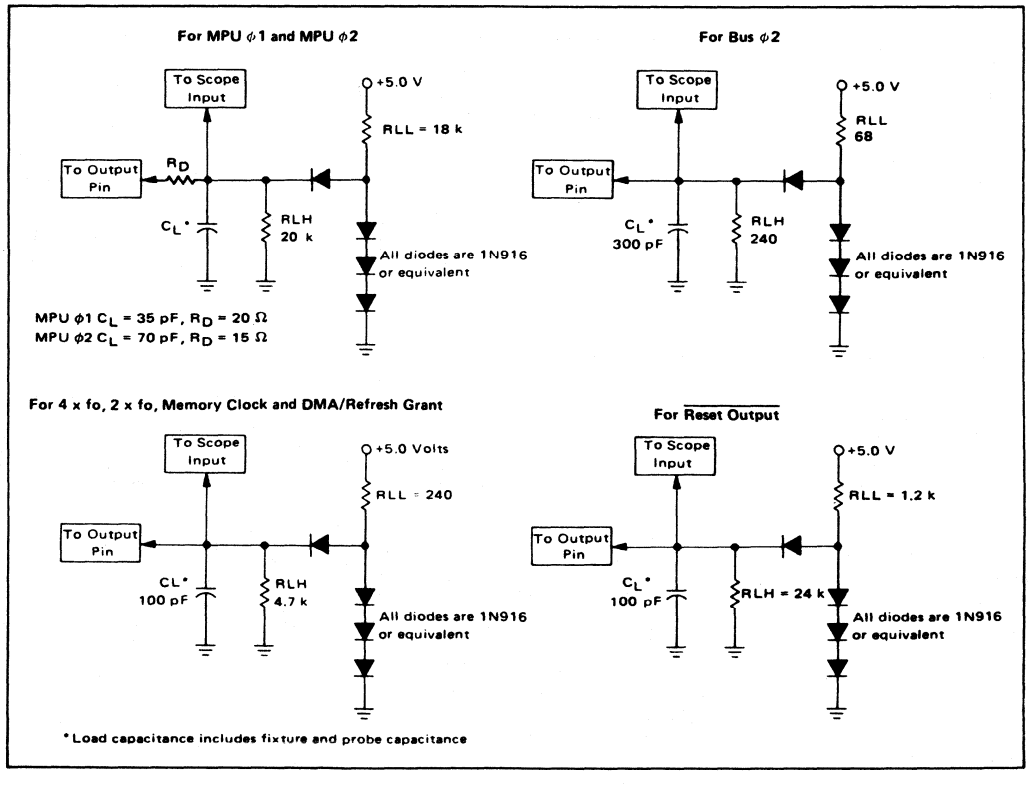


FIGURE 8 – LOAD CIRCUITS



APPLICATIONS INFORMATION

FIGURE 9 – TYPICAL RC FREQUENCY versus VOLTAGE

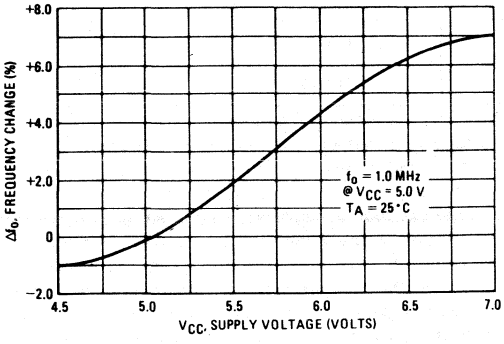


FIGURE 10 – TYPICAL RC FREQUENCY versus TEMPERATURE

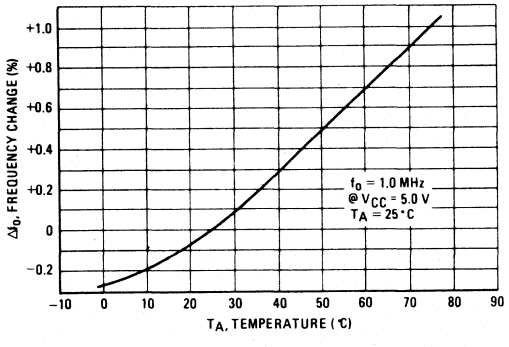
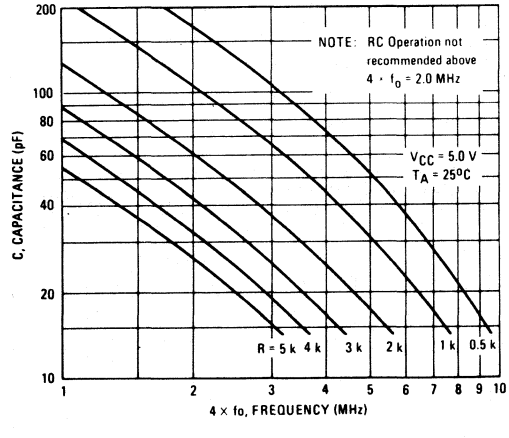


FIGURE 11 – TYPICAL FREQUENCY versus RESISTANCE FOR C VARIABLE



GENERAL

The MC6875 Clock Generator/Driver should be located on the same board and within two inches of the MC6800 MPU. Series damping resistors of 10-30 ohms may be utilized between the MC6875 and the MC6800 on the $\phi 1$ and $\phi 2$ clocks to suppress overshoot and reflections.

The V_{CC} pin (pin 16) of the MC6875 should be bypassed to the ground pin (pin 8) at the package with a 0.1 μF capacitor. Because of the high peak currents associated with driving highly capacitive loads, an adequately large ground strip to pin 8 should be used on the MC6875. Grounds should be carefully routed to minimize coupling of noise to the sensitive oscillator inputs. Unnecessary grounds or ground planes should be avoided near pin 2 or the frequency determining components. These components should be located as near as possible to the respective pins of the MC6875. Stray capacitance near pin 2 or the crystal, can affect the frequency. The can of the crystal should not be grounded. The ground side of the crystal or the C of the R-C oscillator should be connected as directly as possible to pin 8.

Unused inputs should be connected to V_{CC} or ground. Memory Ready, $\overline{\text{DMA/Refresh Request}}$ and $\overline{\text{Power-On Reset}}$ should be connected to V_{CC} when not used. The External Input should be connected to ground when not used.

OSCILLATOR

A tank circuit tuned to the desired crystal frequency connected between terminals X_1 and X_2 as shown in Figure 12, is recommended to prevent the oscillator from starting at other than the desired frequency. The 1kΩ resistor reduces the Q sufficiently to maintain stable crystal control. Crystal manufacturers may recommend a capacitance (C_L) to be used in series with the crystal for optimum performance at series resonance.

See Figures 9 and 10 for typical oscillator temperature and V_{CC} supply dependence for R-C operation.

FIGURE 12 – OSCILLATOR–CRYSTAL OPERATION

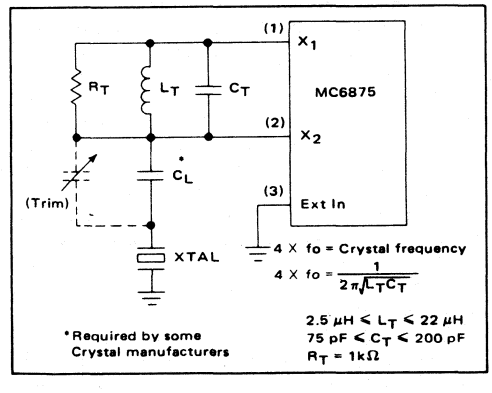
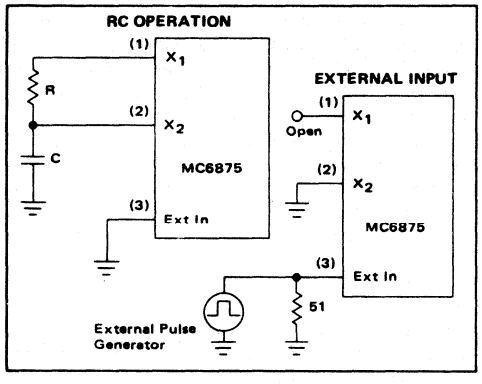


TABLE 1 – OSCILLATOR COMPONENTS

TANK CIRCUIT PARAMETERS		APPROXIMATE CRYSTAL PARAMETERS				CTS KNIGHTS 400 REIMANN AVE. SANDWICH, IL 60548 (815) 786-8411	McCOY ELECT. CO. WATTS & CHESTNUTS STS. MT. HOLLY SPRING, PA 17065 (717) 486-3411	TYCO CRYSTAL PRODUCTS 3940 W. MONTECITO PHOENIX, AZ 85019 (602) 272-7945
L_T μH	C_T pF	R_S Ohms	Co pF	C_1 mpF	fo MHz			
10	150	15-75	3-6	12	4.0	MP-04A * 390 pF	113-31	150-3260
4.7	82	8-45	4-7	23	8.0	MP-080 * 47 pF	113-32	150-3270

Inductors may be obtained from: Coilcraft, Cary, IL 60013 (312) 639-2361

FIGURE 13



To precisely time a crystal to desired frequency, a variable trimmer capacitor in the range of 7 to 40 pF would typically be used. Note it is not a recommended practice to tune the crystal with a parallel load capacitance.

The table above shows typical values for C_T and L_T, typical crystal characteristics, and manufacturers' part numbers for 4.0 and 8.0 megahertz operation.

The MC6875 will function as an R-C oscillator when connected as shown in Figure 13. The desired output frequency (Mϕ1) is approximately:

Formula

$$4 \times fo \approx \frac{320}{C\,(R + .27) + 23}$$

C in picofarads
R in K ohms
4 x fo in Megahertz

(See Figure 11)

It would be desirable to select a capacitor greater than 15 pF to minimize the effects of stray capacitance. It is also desirable to keep the resistor in the 1 to 5 k Ω range. There is a nominal 270 Ω resistor internally at X_1 which is in series with the external R. By keeping the external R as large as possible, the effects due to process variations of the internal resistor on the frequency will be reduced. There will, however, still be some variation in frequency in a production lot both from the resistance variations, external and internal, and process variations of the input switching thresholds. Therefore, in a production system, it is recommended a potentiometer be placed in series with a fixed R between X_1 and X_2.

POWER-ON RESET

As the power to the MC6875 comes up, the Reset Output will be in a high impedance state and will not give a solid V_{OL} output level until V_{CC} has reached 3.5 to 4.0 V. During this time transients may appear on the clock outputs as the oscillator begins to start. This happens at approximately V_{CC} = 3 V. At some V_{CC} level above that, where Reset Output goes low, all the clock outputs will begin functioning normally. This phenomenon of the start-up sequence should not cause any problems except possibly in systems with battery back-up memory. The transients on the clock lines during the time the Reset Output is high impedance could initiate the system in some unknown mode and possibly write into the backup memory system. Therefore in battery backup systems, more elaborate reset circuitry will be required.

Please note that the Power-On Reset input pin of the MC6875 is not suitable for use with a manual MPU reset switch if the DMA/Ref Req or Memory Ready inputs are going to be used. The power on reset circuitry is used to initialize the internal control logic and whenever the input is switched low, the MC6875 is irresponsive to the DMA/Ref Req or Memory Ready inputs. This may result in the loss of dynamic memory and/or possibly a byte of slow static memory. The circuit of Figure 14 is recommended for applications which do not utilize the DMA/Ref Req or Memory Ready inputs. The circuit of Figure 15 is recommended for those applications that do.

FIGURE 14 – MANUAL RESET FOR APPLICATIONS NOT USING DMA/REFRESH REQUEST OR MEMORY READY INPUTS

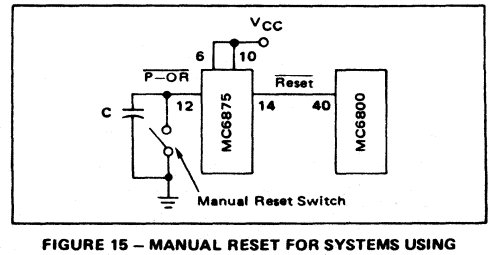
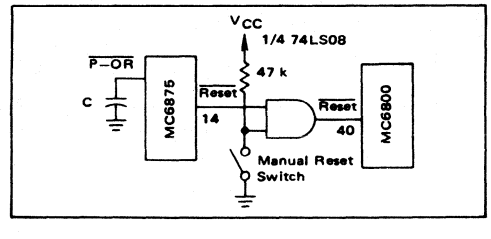
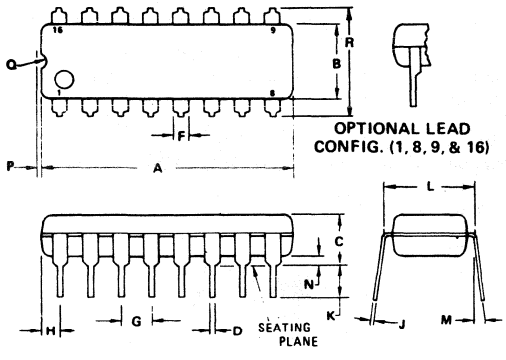


FIGURE 15 – MANUAL RESET FOR SYSTEMS USING DYNAMIC RAM OR SLOW STATIC RAM IN CONJUNCTION WITH MEMORY READY OR DMA/REFRESH REQUEST INPUTS



OUTLINE DIMENSIONS

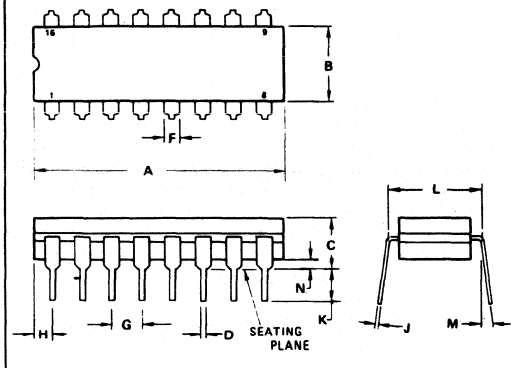
$R_{\theta JA}$ = 100°C/W (Typ)



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	–	22.10	–	0.870
B	6.10	6.60	0.240	0.260
C	–	5.08	–	0.200
D	0.38	0.53	0.015	0.021
F	–	1.78	–	0.070
G	2.54 BSC		0.100 BSC	
H	0.38	2.41	0.015	0.095
J	0.20	0.38	0.008	0.015
K	2.92	–	0.115	–
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	–	0.020	–
R	–	8.26	–	0.325

CASE 648-04

$R_{\theta JA}$ = 100°C/W (Typ)



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	19.05	19.81	0.750	0.780
B	6.22	6.98	0.245	0.275
C	4.06	5.08	0.160	0.200
D	0.38	0.51	0.015	0.020
F	1.40	1.65	0.055	0.065
G	2.54 BSC		0.100 BSC	
H	0.51	1.14	0.020	0.045
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	7.37	7.87	0.290	0.310
M	–	15°	–	15°
N	0.51	1.02	0.020	0.040

CASE 620-02

THERMAL INFORMATION

The maximum power consumption an integrated circuit can tolerate at a given operating ambient temperature, can be found from the equation:

$$P_{D(T_A)} = \frac{T_{J(max)} - T_A}{R_{\theta JA}}$$

Where: $P_{D(T_A)}$ = Power Dissipation allowable at a given operating ambient temperature. This must be greater than the sum of the products of the supply voltages and supply currents at the worst case operating condition.

$T_{J(max)}$ = Maximum Operating Junction Temperature as listed in the Maximum Ratings Section

T_A = Maximum Desired Operating Ambient Temperature

$R_{\theta JA}$ = Thermal Resistance Junction to Ambient

MOTOROLA

MC6880A
MC8T26A

This device may be ordered under either of the above type numbers.

QUAD THREE-STATE BUS TRANSCEIVER

This quad three-state bus transceiver features both excellent MOS or MPU compatibility, due to its high impedance PNP transistor input, and high-speed operation made possible by the use of Schottky diode clamping. Both the −48 mA driver and −20 mA receiver outputs are short-circuit protected and employ three-state enabling inputs.

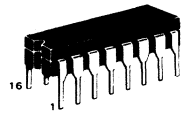
The device is useful as a bus extender in systems employing the M6800 family or other comparable MPU devices. The maximum input current of 200 μA at any of the device input pins assures proper operation despite the limited drive capability of the MPU chip. The inputs are also protected with Schottky-barrier diode clamps to suppress excessive undershoot voltages.

The MC8T26A is identical to the NE8T26A and it operates from a single +5 V supply.

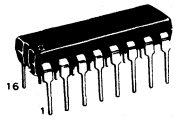
- High Impedance Inputs
- Single Power Supply
- High Speed Schottky Technology
- Three-State Drivers and Receivers
- Compatible with M6800 Family Microprocessor

QUAD THREE-STATE BUS TRANSCEIVER

MONOLITHIC SCHOTTKY INTEGRATED CIRCUITS

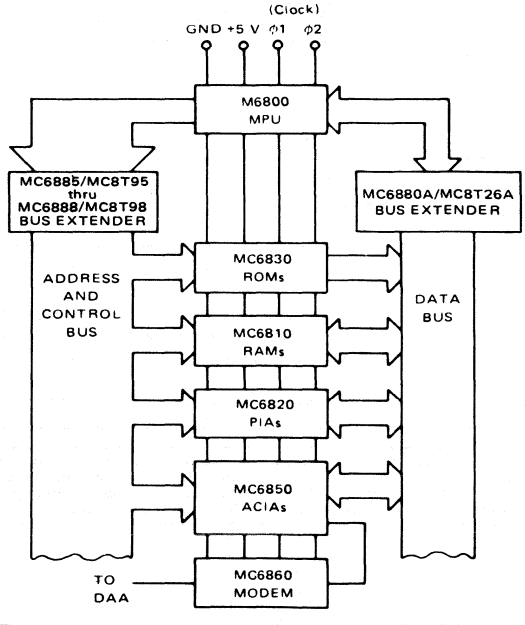


L SUFFIX
CERAMIC PACKAGE
CASE 620

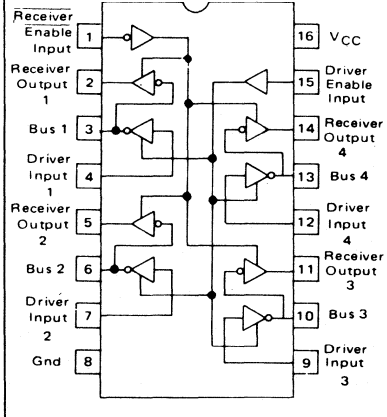


P SUFFIX
PLASTIC PACKAGE
CASE 648

MICROPROCESSOR BUS EXTENDER APPLICATION



PIN CONNECTIONS – MC6880A MC8T26A



ORDERING INFORMATION

Device	Alternate	Temperature Range	Package
MC6880AL	MC8T26AL	0 to +75°C	Ceramic DIP
MC6880AP	MC8T26AP	0 to +75°C	Plastic DIP

MAXIMUM RATINGS (T_A = 25°C unless otherwise noted.)

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	8.0	Vdc
Input Voltage	V_I	5.5	Vdc
Junction Temperature Ceramic Package Plastic Package	T_J	 175 150	°C
Operating Ambient Temperature Range	T_A	0 to +75	°C
Storage Temperature Range	T_{stg}	-65 to +150	°C

ELECTRICAL CHARACTERISTICS (4.75 V ≤ V_{CC} ≤ 5.25 V and 0°C ≤ T_A ≤ 75°C unless otherwise noted.)

Characteristic	Symbol	Min	Typ	Max	Unit
Input Current – Low Logic State					
(Receiver Enable Input, $V_{IL(\overline{RE})}$ = 0.4 V)	$I_{IL(\overline{RE})}$	–	–	-200	μA
(Driver Enable Input, $V_{IL(DE)}$ = 0.4 V)	$I_{IL(DE)}$	–	–	-200	
(Driver Input, $V_{IL(D)}$ = 0.4 V)	$I_{IL(D)}$	–	–	-200	
(Bus (Receiver) Input, $V_{IL(B)}$ = 0.4 V)	$I_{IL(B)}$	–	–	-200	
Input Disabled Current – Low Logic State (Driver Input, $V_{IL(D)}$ = 0.4 V)	$I_{IL(D)\ DIS}$	–	–	-25	μA
Input Current-High Logic State					
(Receiver Enable Input, $V_{IH(RE)}$ = 5.25 V)	$I_{IH(\overline{RE})}$	–	–	25	μA
(Driver Enable Input, $V_{IH(DE)}$ = 5.25 V)	$I_{IH(DE)}$	–	–	25	
(Driver Input, $V_{IH(D)}$ = 5.25 V)	$I_{IH(D)}$	–	–	25	
(Receiver Input, $V_{IH(B)}$ = 5.25 V)	$I_{IH(B)}$	–	–	100	
Input Voltage – Low Logic State					
(Receiver Enable Input)	$V_{IL(\overline{RE})}$	–	–	0.85	V
(Driver Enable Input	$V_{IL(DE)}$	–	–	0.85	
(Driver Input)	$V_{IL(D)}$	–	–	0.85	
(Receiver Input)	$V_{IL(B)}$	–	–	0.85	
Input Voltage – High Logic State					
(Receiver Enable Input)	$V_{IH(\overline{RE})}$	2.0	–	–	V
(Driver Enable Input)	$V_{IH(DE)}$	2.0	–	–	
(Driver Input)	$V_{IH(D)}$	2.0	–	–	
(Receiver Input)	$V_{IH(B)}$	2.0	–	–	
Output Voltage – Low Logic State					
(Bus Driver) Output, $I_{OL(B)}$ = 48 mA)	$V_{OL(B)}$	–	–	0.5	V
(Receiver Output, $I_{OL(R)}$ = 20 mA)	$V_{OL(R)}$	–	–	0.5	
Output Voltage – High Logic State					
(Bus (Driver) Output, $I_{OH(B)}$ = -10 mA)	$V_{OH(B)}$	2.4	3.1	–	V
(Receiver Output, $I_{OH(R)}$ = -2.0 mA)	$V_{OH(R)}$	2.4	3.1	–	
(Receiver Output, $I_{OH(R)}$ = -100 μA, V_{CC} = 5.0 V)		3.5	–	–	
Output Disabled Leakage Current – High Logic State					
(Bus Driver) Output, $V_{OH(B)}$ = 2.4 V)	$I_{OHL(B)}$	–	–	100	μA
(Receiver Output, $V_{OH(R)}$ = 2.4 V)	$I_{OHL(R)}$	–	–	100	
Output Disabled Leakage Current – Low Logic State					
(Bus Output, $V_{OL(B)}$ = 0.5 V)	$I_{OLL(B)}$	–	–	-100	μA
(Receiver Output, $V_{OL(R)}$ = 0.5 V)	$I_{OLL(R)}$	–	–	-100	
Input Clamp Voltage					
(Driver Enable Input $I_{ID(DE)}$ = -12 mA)	$V_{IC(DE)}$	–	–	-1.0	V
(Receiver Enable Input $I_{IC(RE)}$ = +12 mA)	$V_{IC(RE)}$	–	–	-1.0	
(Driver Input $I_{IC(D)}$ = -12 mA)	$V_{IC(D)}$	–	–	-1.0	
Output Short-Circuit Current, V_{CC} = 5.25 V (1)					
(Bus (Driver) Output)	$I_{OS(B)}$	-50	–	-150	mA
(Receiver Output)	$I_{OS(R)}$	-30	–	-75	
Power Supply Current (V_{CC} = 5.25 V)	I_{CC}	–	–	87	mA

(1) Only one output may be short-circuited at a time.

SWITCHING CHARACTERISTICS (Unless otherwise noted, specifications apply at $T_A = 25^oC$ and $V_{CC} = 5.0$ V)

Characteristic	Symbol	Figure	Min	Max	Unit
Propagation Delay Time from Receiver (Bus) Input to High Logic State Receiver Output	$t_{PLH(R)}$	1	–	14	ns
Propagation Delay Time from Receiver (Bus) Input to Low Logic State Receiver Output	$t_{PHL(R)}$	1	–	14	ns
Propagation Delay Time from Driver Input to High Logic State Driver (Bus) Output	$t_{PLH(D)}$	2	–	14	ns
Propagation Delay Time from Driver Input to Low Logic State Driver (Bus) Output	$t_{PHL(D)}$	2	–	14	ns
Propagation Delay Time from $\overline{\text{Receiver Enable}}$ Input to High Impedance (Open) Logic State Receiver Output	$t_{PLZ(RE)}$	3	–	15	ns
Propagation Delay Time from $\overline{\text{Receiver Enable}}$ Input to Low Logic Level Receiver Output	$t_{PZL(RE)}$	3	–	20	ns
Propagation Delay Time from Driver Enable Input to High Impedance Logic State Driver (Bus) Output	$t_{PLZ(DE)}$	4	–	20	ns
Propagation Delay Time from Driver Enable Input to Low Logic State Driver (Bus) Output	$t_{PZL(DE)}$	4	–	25	ns

FIGURE 1 – TEST CIRCUIT AND WAVEFORMS FOR PROPAGATION DELAY FROM BUS (RECEIVER) INPUT TO RECEIVER OUTPUT, $t_{PLH(R)}$ AND $t_{PHL(R)}$

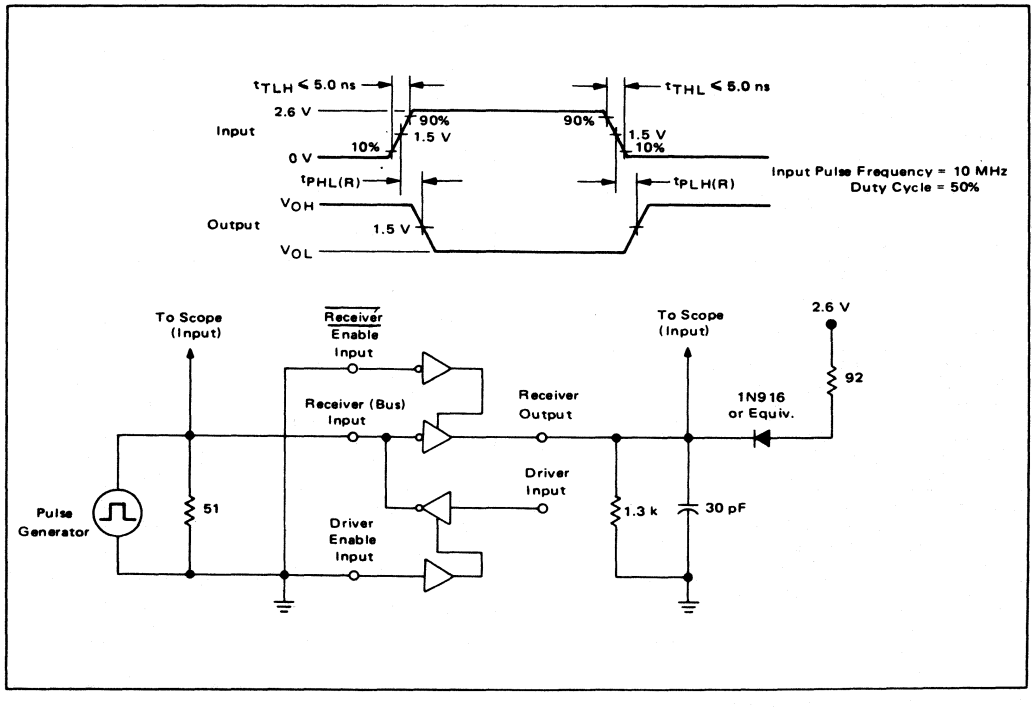


FIGURE 2 – TEST CIRCUIT AND WAVEFORMS FOR PROPAGATION DELAY TIME FROM DRIVER INPUT TO BUS (DRIVER) OUTPUT, $t_{PLH(D)}$ AND $t_{PHL(D)}$

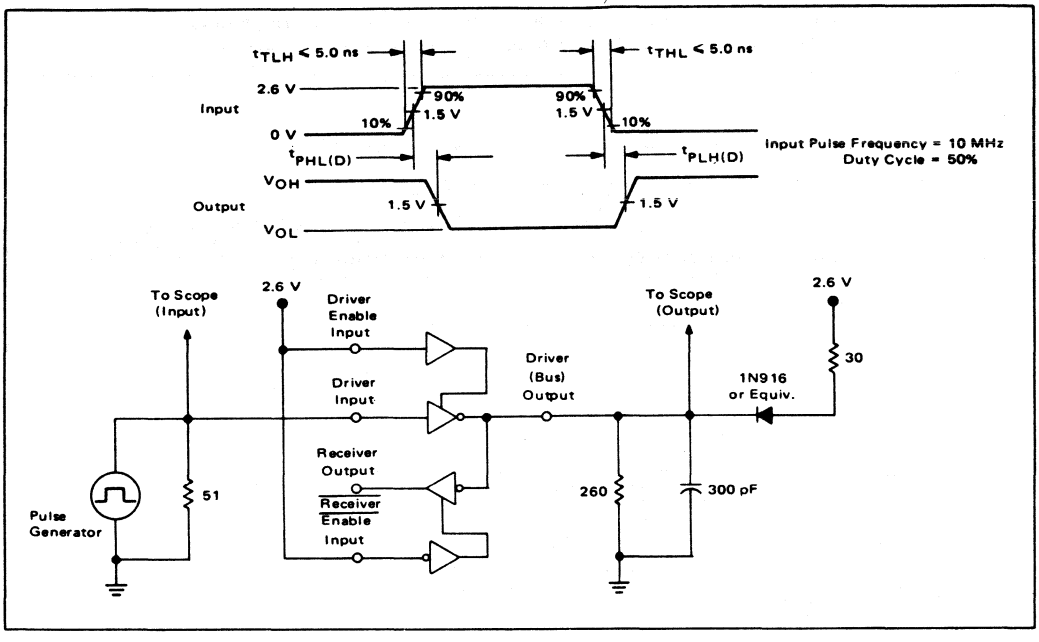


FIGURE 3 – TEST CIRCUIT AND WAVEFORMS FOR PROPAGATION DELAY TIME FROM RECEIVER ENABLE INPUT TO RECEIVER OUTPUT, $t_{PLZ(RE)}$ AND $t_{PZL(RE)}$

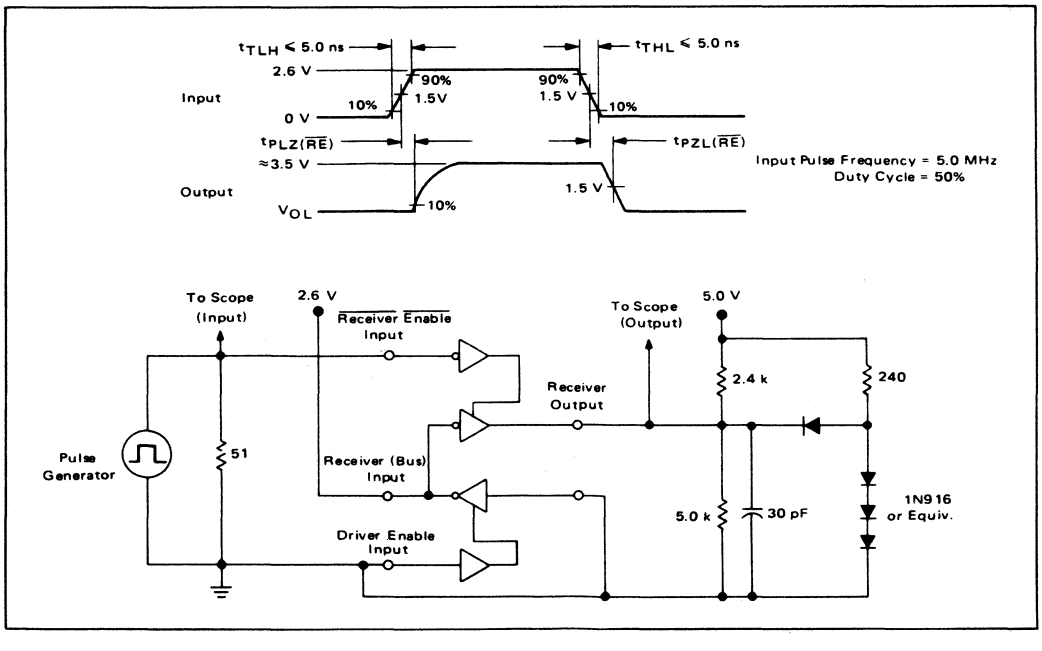


FIGURE 4 – TEST CIRCUIT AND WAVEFORMS FOR PROPAGATION DELAY TIMES FROM DRIVER ENABLE INPUT TO DRIVER (BUS) OUTPUT, $t_{PLZ(DE)}$ AND $t_{PZL(DE)}$

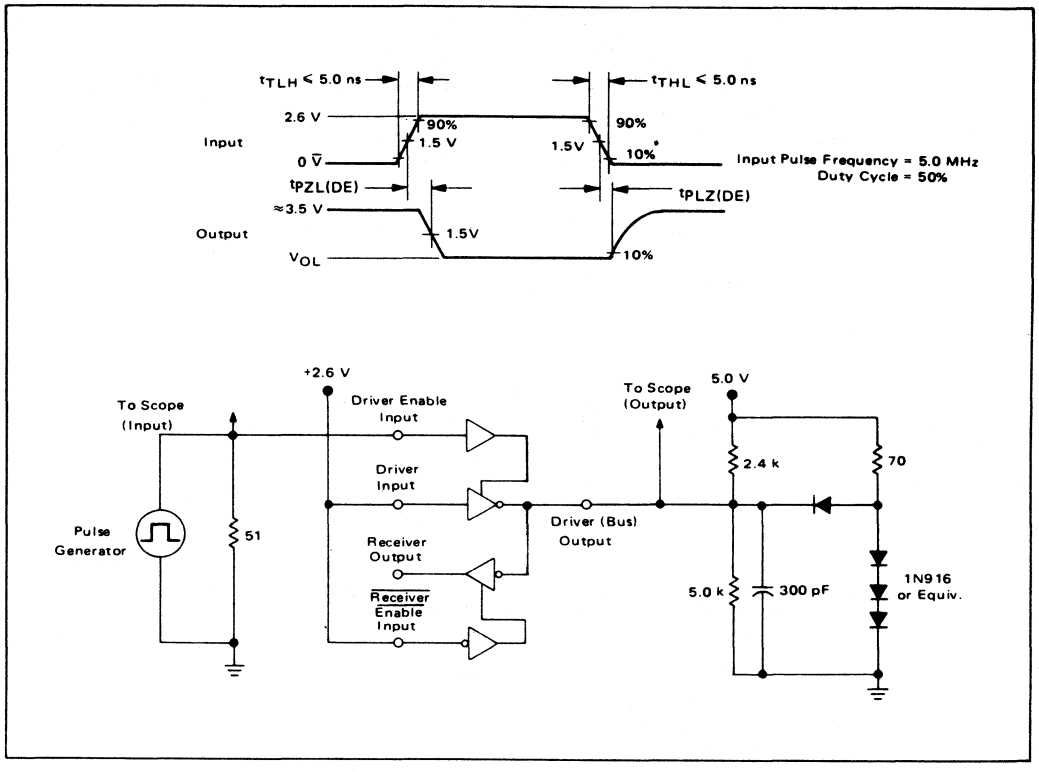
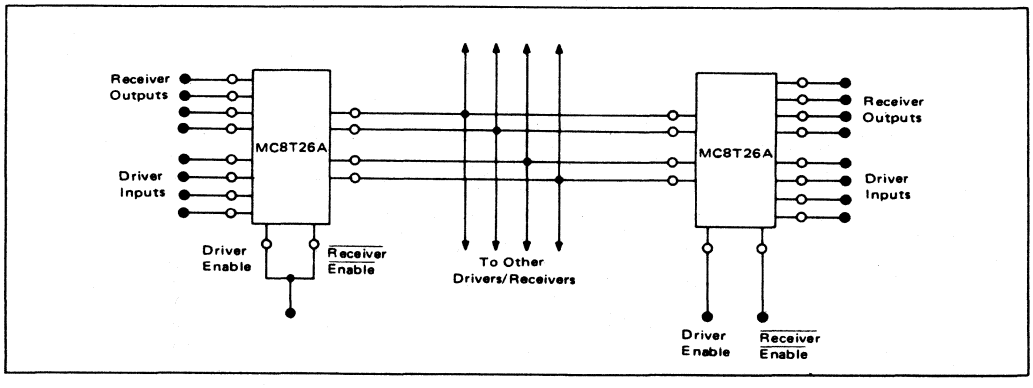
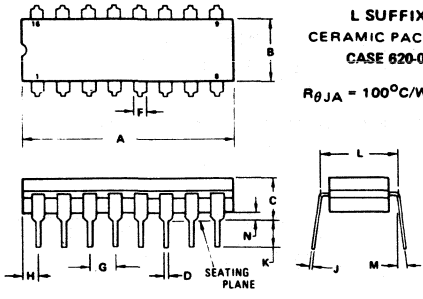


FIGURE 5 – BIDIRECTIONAL BUS APPLICATIONS



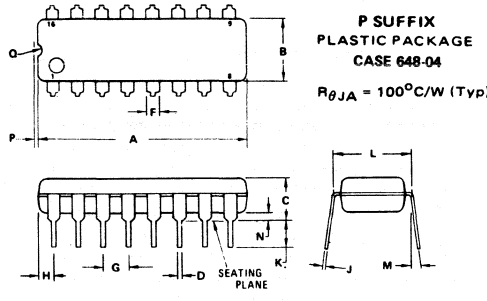


L SUFFIX
CERAMIC PACKAGE
CASE 620-02

$R_{\theta JA}$ = 100°C/W (Typ)

NOTES:
1 LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION
2 PKG. INDEX: NOTCH IN LEAD NOTCH IN CERAMIC OR INK DOT
3 DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	19.05	19.81	0.750	0.780
B	6.22	6.98	0.245	0.275
C	4.06	5.08	0.160	0.200
D	0.38	0.51	0.015	0.020
F	1.40	1.65	0.055	0.065
G	2.54 BSC		0.100 BSC	
H	0.51	1.14	0.020	0.045
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	7.37	7.87	0.290	0.310
M	–	15°	–	15°
N	0.51	1.02	0.020	0.040



P SUFFIX
PLASTIC PACKAGE
CASE 648-04

$R_{\theta JA}$ = 100°C/W (Typ)

NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	–	22.10	–	0.870
B	6.10	6.60	0.240	0.260
C	–	5.08	–	0.200
D	0.38	0.53	0.015	0.021
F	–	1.78	–	0.070
G	2.54 BSC		0.100 BSC	
H	0.38	2.41	0.015	0.095
J	0.20	0.38	0.008	0.015
K	2.92	–	0.115	–
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	–	0.020	–
R	–	8.26	–	0.325

THERMAL INFORMATION

The maximum power consumption an integrated circuit can tolerate at a given operating ambient temperature, can be found from the equation:

$$P_{D(T_A)} = \frac{T_{J(max)} - T_A}{R_{\theta JA(Typ)}}$$

Where: $P_{D(T_A)}$ = Power Dissipation allowable at a given operating ambient temperature. This must be greater than the sum of the products of the supply voltages and supply currents at the worst case operating condition.

$T_{J(max)}$ = Maximum Operating Junction Temperature as listed in the Maximum Ratings Section

T_A = Maximum Desired Operating Ambient Temperature

$R_{\theta JA(Typ)}$ = Typical Thermal Resistance Junction to Ambient

MOTOROLA

MC6881
MC3449

This device may be ordered under either of the above type numbers.

Advance Information

TRIPLE BI-DIRECTIONAL BUS SWITCH

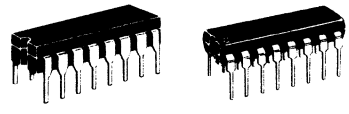
BI-DIRECTIONAL BUS EXTENDER/SWITCH

The MC6881/3449 is a three channel, non-inverting, bi-directional Bus Extender. It is designed to allow the bi-directional exchange of TTL level digital information between a selected pair of ports in a three port network. All three ports of each channel may be forced to a high impedance condition through that channel's $\overline{\text{Enable}}$ input.

Port pair selection and listener/talker status for the three channels is determined through the Control and Select inputs. All inputs are PNP buffered, M6800 Family compatible, and protected with Schottky-Barrier diode clamps to suppress undershoot voltages.

A summary of MC6881/3449 features include:

- Three Channels
- Non-Inverting Data Exchange
- Bi-Directional Operation
- Active Pull-Up with Three-State Capability
- High Impedance Inputs
- TTL Compatible
- High Speed Schottky Technology
- Single Power Supply



L SUFFIX
CERAMIC PACKAGE
CASE 620

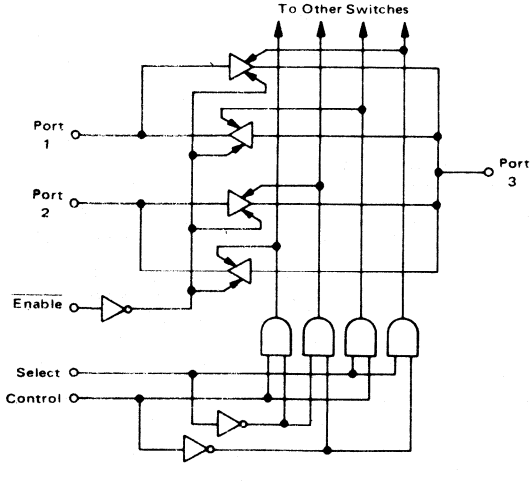
P SUFFIX
PLASTIC PACKAGE
CASE 648

TRUTH TABLE

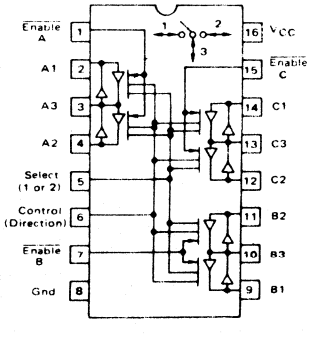
$\overline{\text{Enable}}$	Select	Control	Data Flow
0	0	0	2→3
0	0	1	3→2
0	1	0	1→3
0	1	1	3→1
1	X	X	High Impedance

X Don't Care

FUNCTIONAL DIAGRAM



PIN CONNECTIONS



ORDERING INFORMATION

Device	Alternate	Temperature Range	Package
MC3449L	MC6881L	0 to +70°C	Ceramic DIP
MC3449P	MC6881P	0 to +70°C	Plastic DIP

This is advance information and specifications are subject to change without notice.

MAXIMUM RATINGS (Unless otherwise noted $T_A = 25^oC$.)

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	7.0	Vdc
Input Voltage	V_I	5.5	Vdc
Operating Ambient Temperature Range	T_A	0 to +70	^oC
Storage Temperature Range	T_{stg}	-65 to +150	^oC
Junction Temperature Range Ceramic Package Plastic Package	T_J	 175 150	^oC

Due to the advanced nature of this specification, final electrical limits are not yet given on all parameters. A final version may be obtained after June 1977 by writing:

Motorola Literature Distribution Center
Linear IC – M142
P.O. Box 20912
Phoenix, AZ 85036

ELECTRICAL CHARACTERISTICS (V_{CC} = 4.75 to 5.25 Volts and T_A = 0 to +70°C unless otherwise noted.)

Characteristic	Symbol	Min	Typ	Max	Unit
Input Voltage – Low Logic State	V_{IL}	–	–	0.8	Vdc
Input Voltage – High Logic State	V_{IH}	2.0	–	–	Vdc
Input Current – Low Logic State (V_{IL} = 0.4 V)	I_{IL}	–	–	-100	μA
Input Current – High Logic State	I_{IH}				μA
(V_{IH} = 2.7 V)		–	–	40	
(V_{IH} = 5.25 V)		–	–	100	
Input Clamp Voltage (I_{IC} = -18 mA)	V_{IC}	–	–	-1.5	Vdc
Output Voltage – Low Logic State (I_{OL} = 8.0 mA)	V_{OL}	–	–	0.5	V
Output Voltage – High Logic State (I_{OH} = -400 μA)	V_{OH}	2.7	–	–	V
Output Disabled Leakage Current	I_{OZ}				μA
(V_{OZ} = 0.4 V)		–	–	-40	
(V_{OZ} = 2.7 V)		–	–	40	
(V_{OZ} = 5.25 V)		–	–	100	
Output Short Circuit Current	I_{OS}	-20	–	-55	mA
Crosstalk Current – Low Logic State (V_{IH} = 2.4 V on Node 3, opposite node selected V_{IL} = 0.4 V on node tested)	I_{XL}	–	–	-40	μA
Crosstalk Current – High Logic State (V_{IL} = 0.8 V on Node 3, opposite node selected V_{IH} = 2.4 V on node tested)	I_{XH}	–	–	40	μA
Power Supply Current (V_{IH} = 2.4 V, V_{CC} = 5.25 V)	I_{CC}	–	–	70	mA

SWITCHING CHARACTERISTICS (V_{CC} = 5.0 V and T_A = 25°C unless otherwise noted.)

Propagation Delay Times – Nodes 1, 2, 3					ns
Low-to-High Output	t_{PLH}	–	30	–	
High-to-Low Output	t_{PHL}	–	24	–	
Enable Delay Times					ns
Disabled to High or Low-Logic State	t_{EN}	–	18	–	
High or Low-Logic State to Disabled	t_{DIS}	–	10	–	
Select Delay Times					ns
Third-State to High or Low-Logic State	t_{ON}	–	25	–	
High or Low-Logic State to Third-State	t_{OFF}	–	25	–	
Control Delay Times					ns
Third-State to High or Low-Logic State	t_{ON}	–	25	–	
High or Low-Logic State to Third-State	t_{OFF}	–	25	–	

PROPAGATION DELAY TIME TEST CIRCUITS AND WAVEFORMS

FIGURE 1 – NODE TO OUTPUT

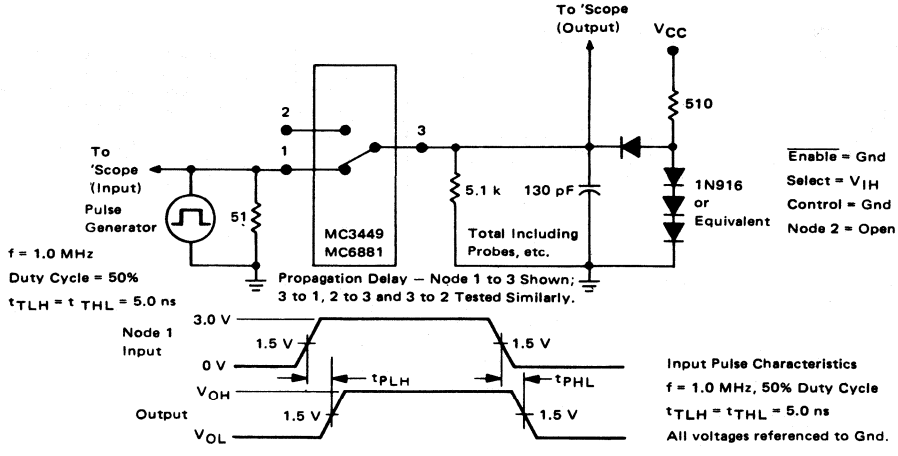
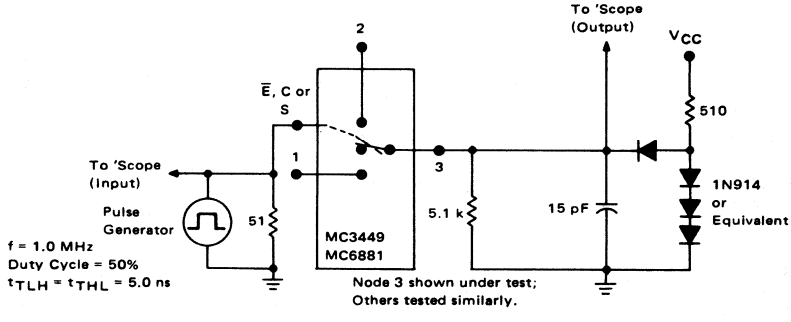
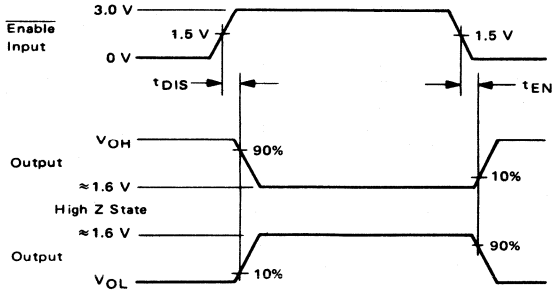


FIGURE 2 – THIRD-STATE

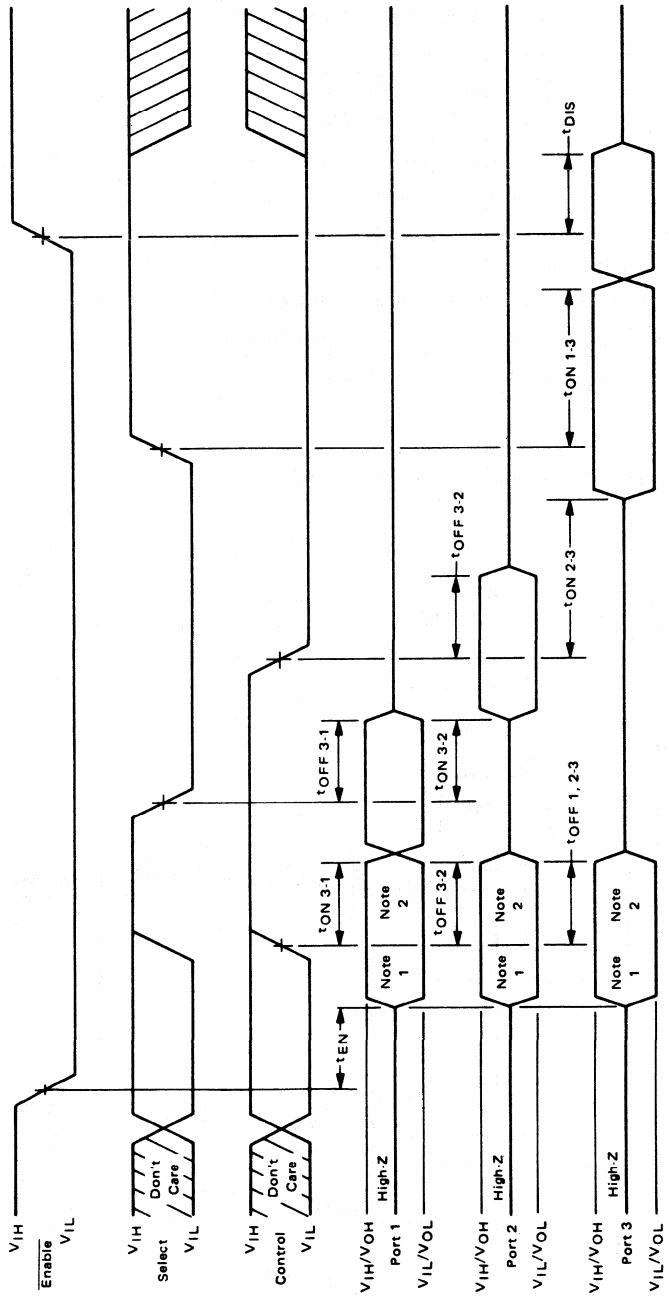




TEST TABLE

Control	Select	Enable	Node 1	Test
V_{IL}	V_{IH}	Pulse	V_{IL}	t_{EN}/t_{DIS}
V_{IL}	V_{IH}	Pulse	V_{IH}	t_{EN}/t_{DIS}

FIGURE 3 – TIMING DIAGRAM



TRUTH TABLE

$\overline{\text{Enable}}$	Select	Control	Data Direction (Note 3)
1	X	X	All Ports are High-Z
0	0	0	Port 2 → Port 3
0	0	1	Port 3 → Port 2
0	1	0	Port 1 → Port 3
0	1	1	Port 3 → Port 1

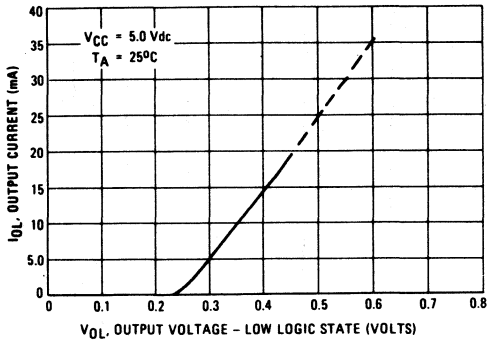
Note 1: Data is transmitted to only 1 of the 3 ports. Which port acts as an output depends on logic state of the select and control pins when the channel is enabled.

Note 2: A port chosen to act as the output is either high or low, depending on the logic state of the port chosen to be the input.

Note 3: The arrow indicates the direction of data flow. Each buffer is non-inverting, so data maintains the same logic state through the buffer.

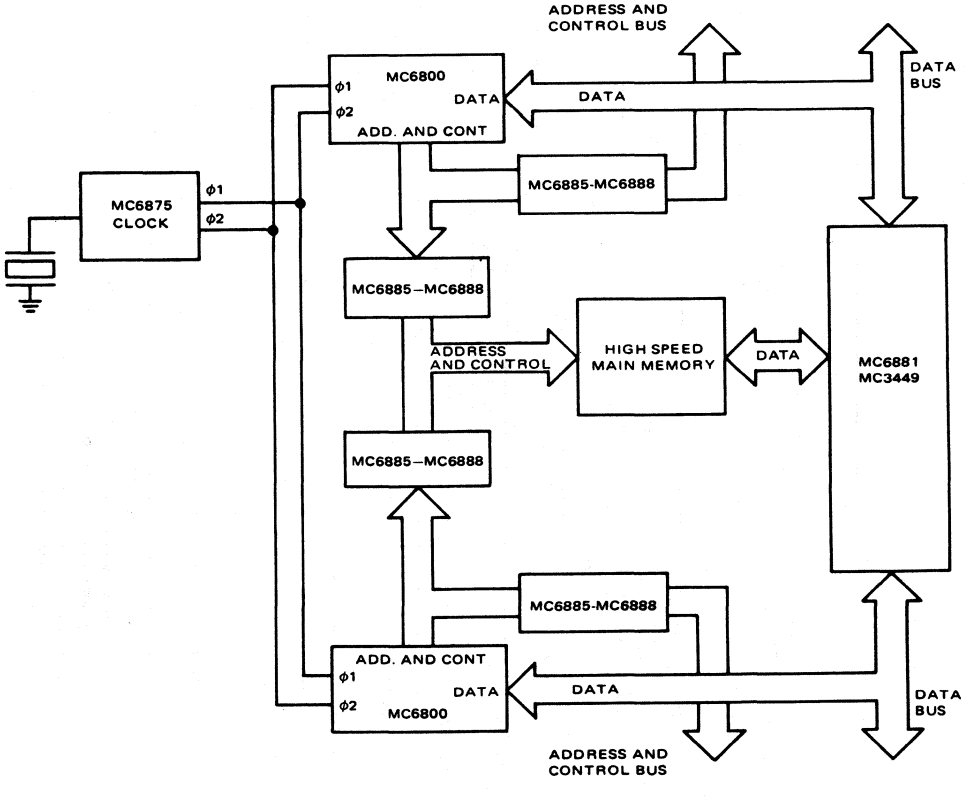
Note 4: t_{ON} is the time from third state to active (high or low) state; t_{OFF} is time from active to third state.

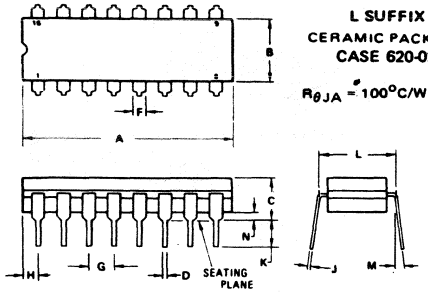
FIGURE 4 – TYPICAL I_{OL} versus V_{OL}



TYPICAL APPLICATION

FIGURE 5 – TWO MPUs SHARING A COMMON MAIN-MEMORY



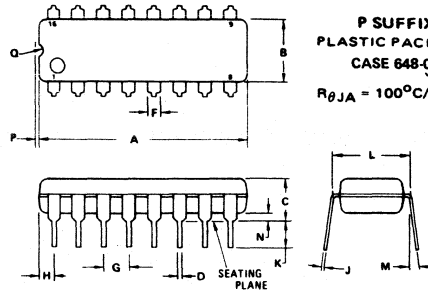


L SUFFIX
CERAMIC PACKAGE
CASE 620-02

$R_{\theta JA}$ = 100°C/W (Typ)

NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION
2. PKG. INDEX: NOTCH IN LEAD NOTCH IN CERAMIC OR INK DOT
3. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	19.05	19.81	0.750	0.780
B	6.22	6.98	0.245	0.275
C	4.06	5.08	0.160	0.200
D	0.38	0.51	0.015	0.020
F	1.40	1.65	0.055	0.065
G	2.54 BSC		0.100 BSC	
H	0.51	1.14	0.020	0.045
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	7.37	7.87	0.290	0.310
M	–	15°	–	15°
N	0.51	1.02	0.020	0.040



P SUFFIX
PLASTIC PACKAGE
CASE 648-04

$R_{\theta JA}$ = 100°C/W (Typ)

NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	–	22.10	–	0.870
B	6.10	6.60	0.240	0.260
C	–	5.08	–	0.200
D	0.38	0.53	0.015	0.021
F	–	1.78	–	0.070
G	2.54 BSC		0.100 BSC	
H	0.38	2.41	0.015	0.095
J	0.20	0.38	0.008	0.015
K	2.92	–	0.115	–
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	–	0.020	–
R	–	8.26	–	0.325

THERMAL INFORMATION

The maximum power consumption an integrated circuit can tolerate at a given operating ambient temperature, can be found from the equation:

$$P_{D(T_A)} = \frac{T_{J(max)} - T_A}{R_{\theta JA(Typ)}}$$

Where: $P_{D(T_A)}$ = Power Dissipation allowable at a given operating ambient temperature. This must be greater than the sum of the products of the supply voltages and supply currents at the worst case operating condition.

$T_{J(max)}$ = Maximum Operating Junction Temperature as listed in the Maximum Ratings Section

T_A = Maximum Desired Operating Ambient Temperature

$R_{\theta JA(Typ)}$ = Typical Thermal Resistance Junction to Ambient

MOTOROLA

MC6882A/MC3482A
MC6882B/MC3482B

This device may be ordered under either of the above type numbers.

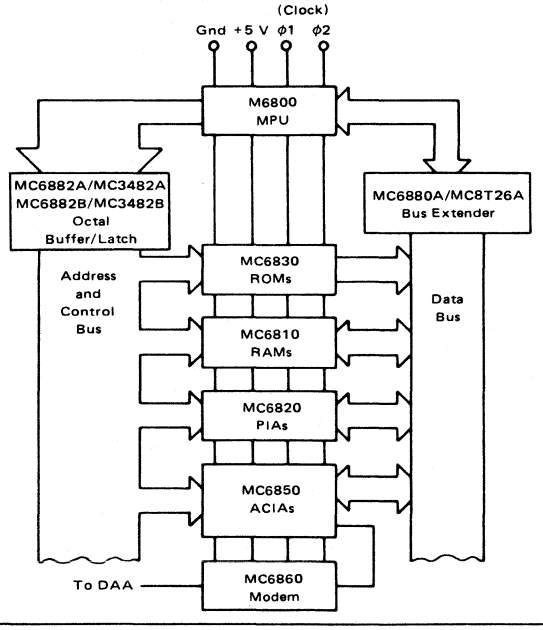
Product Preview

OCTAL THREE-STATE BUFFER/LATCH

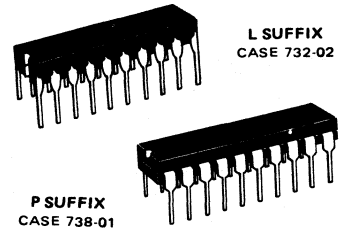
This series of devices combines four features usually found desirable in bus-oriented systems: 1) High impedance logic inputs insure that these devices do not seriously load the bus; 2) Three-state logic configuration allows buffers not being utilized to be effectively removed from the bus; 3) Schottky technology allows high-speed operation; 4) High-impedance output state maintained during power up/down.

- Inverting and Non-Inverting Options of Data
- SN74S373 Function Pinouts
- Eight Transparent Latches/Buffers in a Single Package
- Full Parallel-Access for Loading and Reloading
- Buffered Control Inputs
- All Inputs Have Hysteresis to Improve Noise Rejection
- High Speed — 8.0 ns (Typ)
- Three-State Logic Configuration
- Single +5 V Power Supply Requirement
- Compatible with 74S Logic or M6800 Microprocessor Systems
- High Impedance PNP Inputs Assure Minimal Loading of the Bus

MICROPORCESSOR BUS EXTENDER APPLICATION



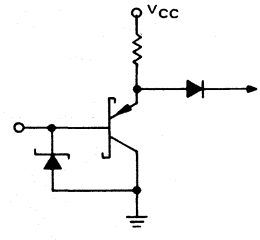
OCTAL THREE-STATE BUFFER/LATCH



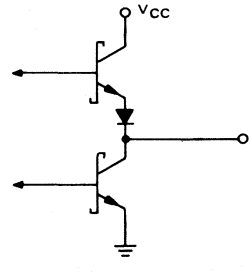
L SUFFIX
CASE 732-02

P SUFFIX
CASE 738-01

INPUT EQUIVALENT CIRCUIT



OUTPUT EQUIVALENT CIRCUIT



ORDERING INFORMATION

(Temperature Range for the following devices = 0 to +75°C.)

Device	Alternate	Package
MC6882AL	MC3482AL	Ceramic DIP
MC6882BL	MC3482BL	Ceramic DIP
MC6882AP	MC3482AP	Plastic DIP
MC6882BP	MC3482BP	Plastic DIP

MAXIMUM RATINGS (T_A = 25°C unless otherwise noted.)

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	8.0	Vdc
Input Voltage	V_I	5.5	Vdc
Operating Ambient Temperature Range	T_A	0 to +75	°C
Storage Temperature Range	T_{stg}	-65 to +150	°C
Operating Junction Temperature Plastic Package Ceramic Package	T_J	 150 175	°C

ELECTRICAL CHARACTERISTICS (Unless otherwise noted, 0°C ≤ T_A ≤ 75°C and 4.75 V ≤ V_{CC} ≤ 5.25 V)

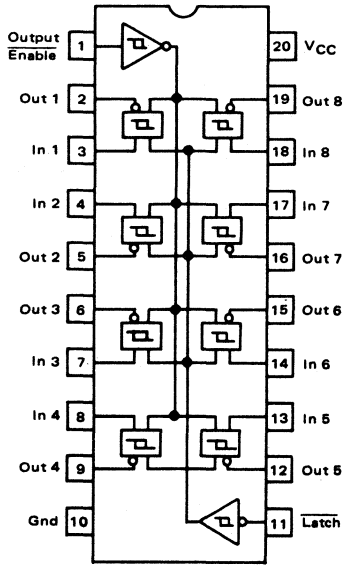
Characteristic	Symbol	Min	Typ	Max	Unit
Input Voltage – High Logic State (V_{CC} = 4.75 V, T_A = 25°C)	V_{IH}	2.0	–	–	V
Input Voltage – Low Logic State (V_{CC} = 4.75 V, T_A = 25°C)	V_{IL}	–	–	0.8	V
Input Current – High Logic State (V_{CC} = 5.25 V, V_{IH} = 2.4 V)	I_{IH}	–	–	40	μA
Input Current – Low Logic State (V_{CC} = 5.25 V, V_{IL} = 0.5 V, $V_{IL(\overline{E})}$ = 0.5 V)	I_{IL}	–	–	-400	μA
Output Voltage – High Logic State (V_{CC} = 4.75 V, I_{OH} = -20 mA)	V_{OH}	2.4	–	–	V
Output Voltage – Low Logic State (I_{OL} = 48 mA)	V_{OL}	–	–	0.5	V
Output Current – High Impedance State (V_{CC} = 5.25 V, V_{OH} = 2.4 V) (V_{CC} = 5.25 V, V_{OL} = 0.5 V)	I_{OZ}	 – –	 – –	 100 -100	μA
Output Short-Circuit Current (V_{CC} = 5.25 V, V_O = 0) (only one output can be shorted at a time)	I_{OS}	-40	-80	-100	mA
Power Supply Current (V_{CC} = 5.25 V)	I_{CC}	–		125	mA
Input Clamp Voltage (V_{CC} = 4.75 V, I_{IC} = -12 mA)	V_{IC}	–	–	-1.2	V

SWITCHING CHARACTERISTICS (V_{CC} = 5.0 V, T_A = 25°C unless otherwise noted.)

Characteristic	Symbol	MC6882A/ MC3482A			MC6882B/ MC3482B			Unit
		Min	Typ	Max	Min	Typ	Min	
Propagation Delay Time – High to Low State (C_L = 50 pF) (C_L = 250 pF) (C_L = 375 pF) (C_L = 500 pF)	t_{PHL}		8.0			10		ns
Propagation Delay Time – Low to High State (C_L = 50 pF) (C_L = 250 pF) (C_L = 375 pF) (C_L = 500 pF)	t_{PLH}		8.0			10		ns
Setup Time	t_{su}		0			0		ns
Hold Time	t_h		11			11		ns
Propagation Delay Time – High State to Third State (C_L = 15 pF)	$t_{PHZ(\overline{E})}$		5.0			7.0		ns
Propagation Delay Time – Low State to Third State (C_L = 15 pF)	$t_{PLZ(\overline{E})}$		9.0			11		ns
Propagation Delay Time – Third State to High State (C_L = 50 pF)	$t_{PZH(\overline{E})}$		7.0			9.0		ns
Propagation Delay Time – Third State to Low State (C_L = 50 pF)	$t_{PZL(\overline{E})}$		12			14		ns

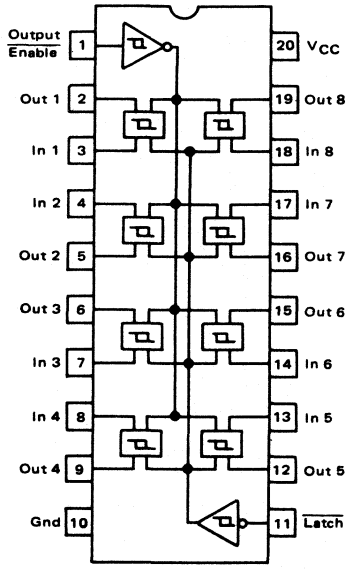
PIN CONNECTIONS AND TRUTH TABLES

MC6882A/MC3482A



Output Enable	Latch	Input	Output
0	1	0	1
0	1	1	0
0	0	X	Q_0
1	X	X	Z

MC68828/MC3482B



Output Enable	Latch	Input	Output
0	1	0	0
0	1	1	1
0	0	X	Q_0
1	X	X	Z

FIGURE 1 – TEST CIRCUIT FOR SWITCHING CHARACTERISTICS

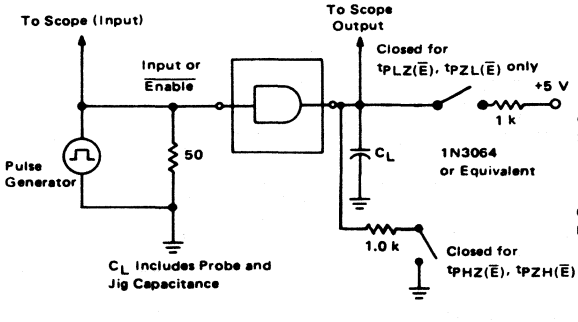


FIGURE 2 – WAVEFORMS FOR PROPAGATION DELAY TIMES INPUT TO OUTPUT

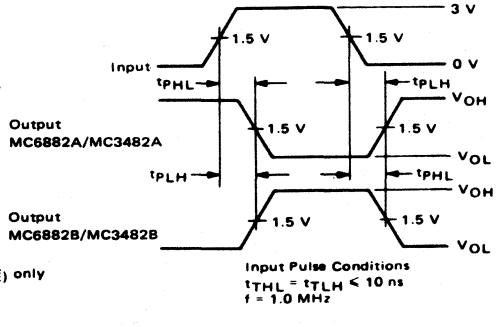
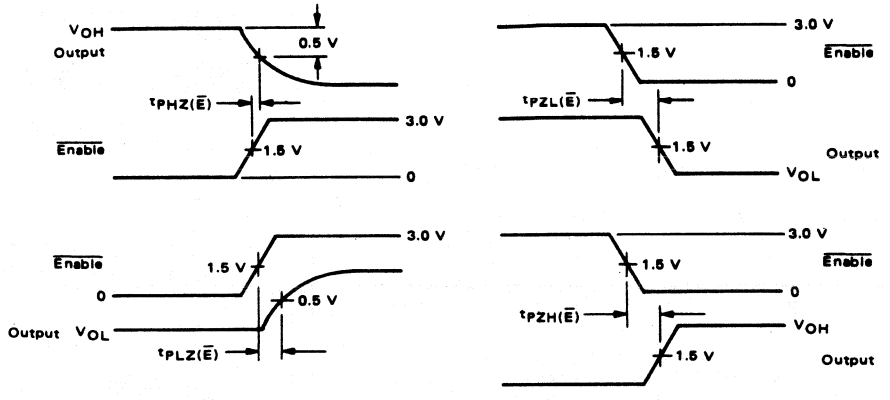


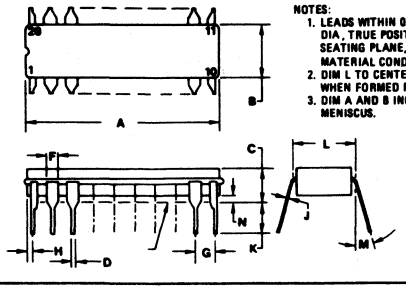
FIGURE 3 – WAVEFORMS FOR PROPAGATION DELAY TIMES – ENABLE TO OUTPUT



H = High-Logic State, L = Low-Logic State, Z = High Impedance State

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	24.38	25.15	0.960	0.990
B	6.86	7.49	0.270	0.295
C	4.32	5.08	0.170	0.200
D	0.38	0.56	0.015	0.022
F	1.40	1.65	0.055	0.065
G	2.54 BSC		0.100 BSC	
H	0.88	1.40	0.035	0.055
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	0.76	0.020	0.030

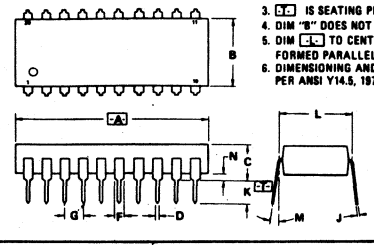
CASE 732-02



NOTES:
1. LEADS WITHIN 0.25 mm (0.010) DIA, TRUE POSITION AT SEATING PLANE, AT MAXIMUM MATERIAL CONDITION.
2. DIM L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIM A AND B INCLUDES MENISCUS.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	25.65	27.18	1.010	1.070
B	6.10	6.60	0.240	0.260
C	3.94	4.19	0.155	0.165
D	0.38	0.56	0.015	0.022
F	1.27	1.78	0.050	0.070
G	2.54 BSC		0.100 BSC	
J	0.20	0.38	0.008	0.015
K	2.79	3.56	0.110	0.140
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	1.02	0.020	0.040

CASE 738-01



NOTES:
1. DIM -A- IS DATUM.
2. POSITIONAL TOL FOR LEADS; ⌖ Ø 0.25 (0.010) Ⓜ T A Ⓜ
3. -T- IS SEATING PLANE.
4. DIM "B" DOES NOT INCLUDE MOLD FLASH.
5. DIM -L- TO CENTER OF LEADS WHEN FORMED PARALLEL.
6. DIMENSIONING AND TOLERANCING PER ANSI Y14.5, 1973.

MOTOROLA

MC6885/MC8T95
MC6886/MC8T96
MC6887/MC8T97
MC6888/MC8T98

This device may be ordered under either of the above type numbers.

HEX THREE-STATE BUFFER INVERTERS

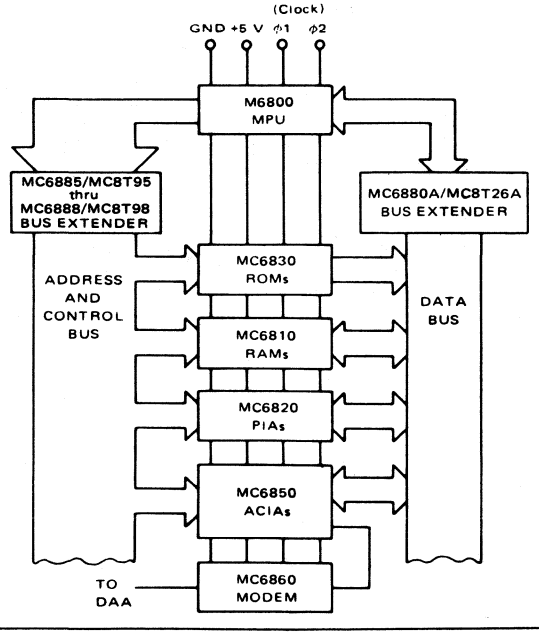
This series of devices combines three features usually found desirable in bus-oriented systems: 1) High impedance logic inputs insure that these devices do not seriously load the bus; 2) Three-state logic configuration allows buffers not being utilized to be effectively removed from the bus; 3) Schottky technology allows high-speed operation.

The devices differ in that the non-inverting MC8T95/MC6885 and inverting MC8T96/MC6886 provide a two-input Enable which controls all six buffers, while the non-inverting MC8T97/MC6887 and inverting MC8T98/MC6888 provide two Enable inputs – one controlling four buffers and the other controlling the remaining two buffers.

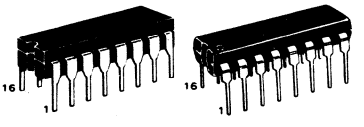
The units are well-suited for Address buffers on the M6800 or similar microprocessor application.

- High Speed – 8.0 ns (Typ)
- Three-State Logic Configuration
- Single +5 V Power Supply Requirement
- Compatible with 74LS Logic or M6800 Microprocessor Systems
- High Impedance PNP Inputs Assure Minimal Loading of the Bus

MICROPROCESSOR BUS EXTENDER APPLICATION

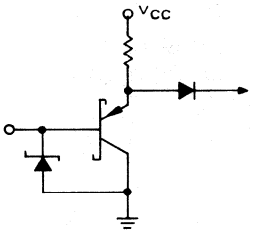


HEX THREE-STATE BUFFER/INVERTERS

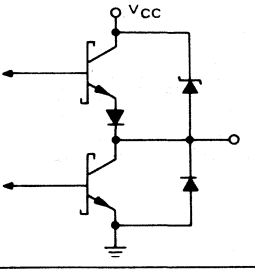


CASE 620 CASE 648

INPUT EQUIVALENT CIRCUIT



OUTPUT EQUIVALENT CIRCUIT



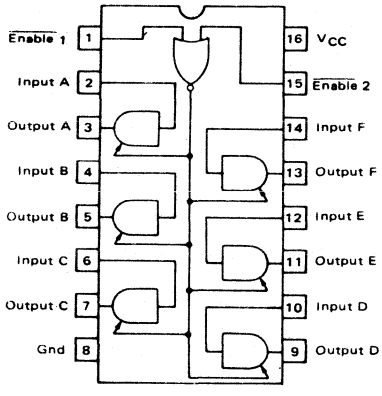
ORDERING INFORMATION

(Temperature Range for the following devices = 0 to +75°C)

DEVICE	ALTERNATE	PACKAGE
MC6885L	MC8T95L	Ceramic DIP
MC6886L	MC8T96L	Ceramic DIP
MC6887L	MC8T97L	Ceramic DIP
MC6888L	MC8T98L	Ceramic DIP
MC6885P	MC8T95P	Plastic DIP
MC6886P	MC8T96P	Plastic DIP
MC6887P	MC8T97P	Plastic DIP
MC6888P	MC8T98P	Plastic DIP

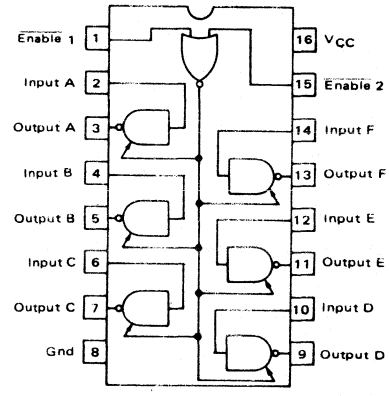
PIN CONNECTIONS AND TRUTH TABLES

MC6885/MC8T95



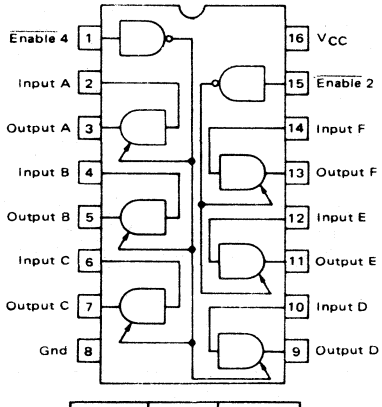
Enable 2	Enable 1	Input	Output
L	L	L	L
L	L	H	H
L	H	X	Z
H	L	X	Z
H	H	X	Z

MC6886/MC8T96



Enable 2	Enable 1	Input	Output
L	L	L	H
L	L	H	L
L	H	X	Z
H	L	X	Z
H	H	X	Z

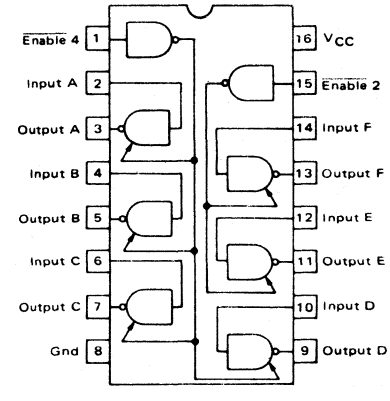
MC6887/MC8T97



Enable	Input	Output
L	L	L
L	H	H
H	X	Z

L = Low Logic State
H = High Logic State
Z = Third (High Impedance) State
X = Irrelevant

MC6888/MC8T98



Enable	Input	Output
L	L	H
L	H	L
H	X	Z

MAXIMUM RATINGS (T_A = 25°C unless otherwise noted.)

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	8.0	Vdc
Input Voltage	V_I	5.5	Vdc
Operating Ambient Temperature Range	T_A	0 to +75	°C
Storage Temperature Range	T_{stg}	-65 to +150	°C
Operating Junction Temperature Plastic Package Ceramic Package	T_J	 150 175	°C

ELECTRICAL CHARACTERISTICS (Unless otherwise noted, $0^{\circ}C \leq T_A \leq 75^{\circ}C$ and $4.75\ V \leq V_{CC} \leq 5.25\ V$)

Characteristic	Symbol	Min	Typ	Max	Unit
Input Voltage – High Logic State (V_{CC} = 4.75 V, T_A = 25°C)	V_{IH}	2.0	–	–	V
Input Voltage – Low Logic State (V_{CC} = 4.75 V, T_A = 25°C)	V_{IL}	–	–	0.8	V
Input Current – High Logic State (V_{CC} = 5.25 V, V_{IH} = 2.4 V)	I_{IH}	–	–	40	μA
Input Current – Low Logic State (V_{CC} = 5.25 V, V_{IL} = 0.5 V, $V_{IL(\overline{E})}$ = 0.5 V)	I_{IL}	–	–	-400	μA
Input Current – High Impedance State (V_{CC} = 5.25 V, $V_{IL(I)}$ = 0.5 V, $V_{IH(\overline{E})}$ = 2.0 V)	$I_{IH(\overline{E})}$	–	–	-40	μA
Output Voltage – High Logic State (V_{CC} = 4.75 V, I_{OH} = -5.2 mA)	V_{OH}	2.4	–	–	V
Output Voltage – Low Logic State (I_{OL} = 48 mA)	V_{OL}	–	–	0.5	V
Output Current – High Impedance State	I_{OZ}				μA
(V_{CC} = 5.25 V, V_{OH} = 2.4 V)		–	–	40	
(V_{CC} = 5.25 V, V_{OL} = 0.5 V)		–	–	-40	
Output Short-Circuit Current (V_{CC} = 5.25 V, V_O = 0) (only one output can be shorted at a time)	I_{OS}	-40	-80	-115	mA
Power Supply Current	I_{CC}				mA
(V_{CC} = 5.25 V) MC8T95, MC8T97, MC6885, MC6887		–	65	98	
MC8T96, MC8T98, MC6886, MC6888		–	59	89	
Input Clamp Voltage (V_{CC} = 4.75 V, I_{IC} = -12 mA)	V_{IC}	–	–	-1.5	V
Output V_{CC} Clamp Voltage (V_{CC} = 0, I_{OC} = 12 mA)	V_{OC}	–	–	1.5	V
Output Gnd Clamp Voltage (V_{CC} = 0, I_{OC} = -12 mA)	V_{OC}	–	–	-1.5	V
Input Voltage (I_I = 1.0 mA)	V_I	5.5	–	–	V

SWITCHING CHARACTERISTICS (V_{CC} = 5.0 V, T_A = 25°C unless otherwise noted.)

		MC8T95/97 MC6885/87			MC8T96/98 MC6886/88			
Characteristic	Symbol	Min	Typ	Max	Min	Typ	Min	Unit
Propagation Delay Time – High to Low State	t_{PHL}							ns
(C_L = 50 pF)		3.0	–	12	4.0	–	11	
(C_L = 250 pF)		–	16	–	–	15	–	
(C_L = 375 pF)		–	20	–	–	18	–	
(C_L = 500 pF)		–	23	–	–	22	–	
Propagation Delay Time – Low to High State	t_{PLH}							ns
(C_L = 50 pF)		3.0	–	13	3.0	–	10	
(C_L = 250 pF)		–	25	–	–	22	–	
(C_L = 375 pF)		–	33	–	–	28	–	
(C_L = 500 pF)		–	42	–	–	35	–	
Transition Time – High to Low State	t_{THL}							ns
(C_L = 250 pF)		–	10	–	–	10	–	
(C_L = 375 pF)		–	11	–	–	13	–	
(C_L = 500 pF)		–	14	–	–	15	–	
Transition Time – Low to High State	t_{TLH}							ns
(C_L = 250 pF)		–	32	–	–	28	–	
(C_L = 375 pF)		–	42	–	–	38	–	
(C_L = 500 pF)		–	60	–	–	53	–	

SWITCHING CHARACTERISTICS (V_{CC} = 5.0 V, T_A = 25°C unless otherwise noted.)

Characteristic	Symbol	MC8T95/97 MC6885/87			MC8T96/98 MC6886/88			Unit
		Min	Typ	Max	Min	Typ	Min	
Propagation Delay Time – High State to Third State (C_L = 5.0 pF)	$t_{PHZ(\overline{E})}$	3.0	–	10	3.0	–	10	ns
Propagation Delay Time – Low State to Third State (C_L = 5.0 pF)	$t_{PLZ(\overline{E})}$	3.0	–	12	5.0	–	16	ns
Propagation Delay Time – Third State to High State (C_L = 50 pF)	$t_{PZH(\overline{E})}$	8.0	–	25	7.0	–	22	ns
Propagation Delay Time – Third State to Low State (C_L = 50 pF)	$t_{PZL(\overline{E})}$	12	–	25	11	–	24	ns

FIGURE 1 – TEST CIRCUIT FOR SWITCHING CHARACTERISTICS

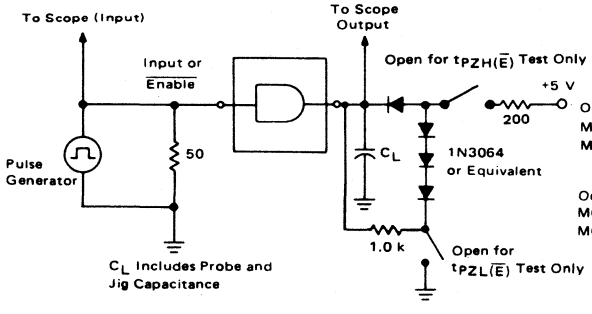


FIGURE 2 – WAVEFORMS FOR PROPAGATION DELAY TIMES INPUT TO OUTPUT

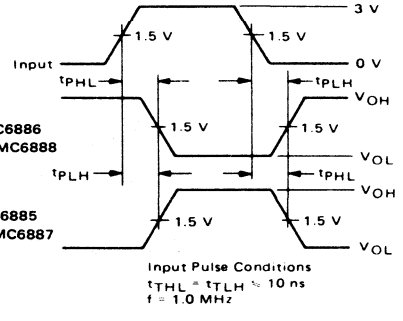
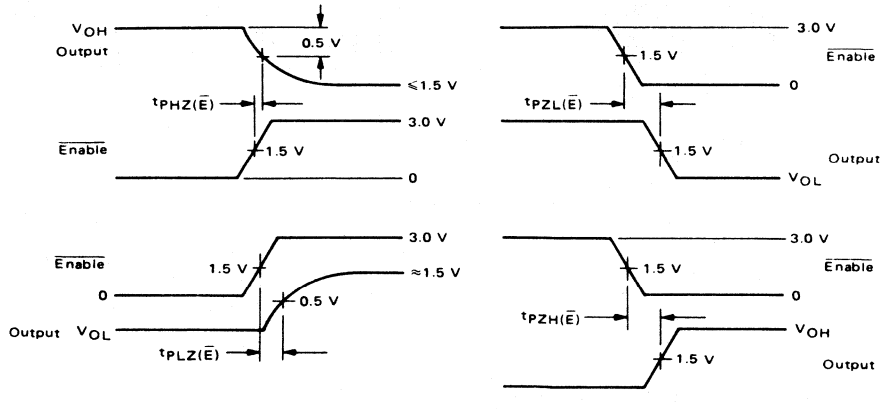
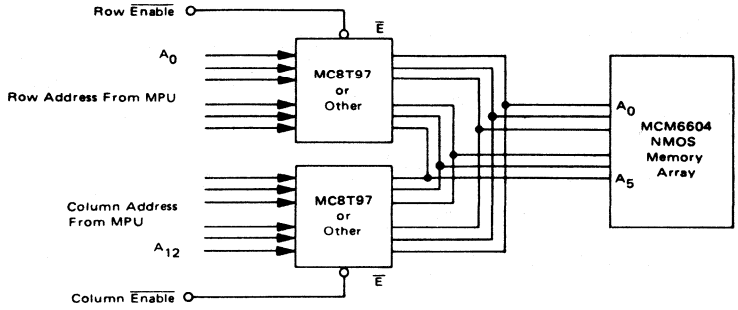


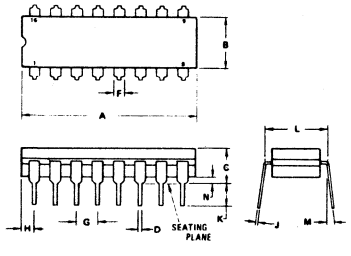
FIGURE 3 – WAVEFORMS FOR PROPAGATION DELAY TIMES – $\overline{\text{ENABLE}}$ TO OUTPUT



H = High-Logic State, L = Low-Logic State, Z = High Impedance State

FIGURE 4 – ADDRESS MULTIPLEXER FOR 16-PIN 4K NMOS MEMORY

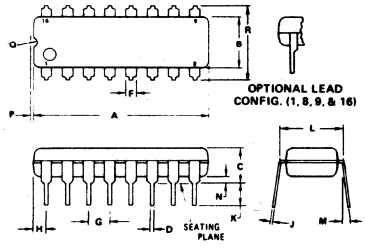




NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION
2. PKG. INDEX: NOTCH IN LEAD NOTCH IN CERAMIC OR INK DOT
3. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	19.05	19.81	0.750	0.780
B	6.22	6.98	0.245	0.275
C	4.06	5.08	0.160	0.200
D	0.38	0.51	0.015	0.020
F	1.40	1.65	0.055	0.065
G	2.54 BSC		0.100 BSC	
H	0.51	1.14	0.020	0.045
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	7.37	7.87	0.290	0.310
M	–	15°	–	15°
N	0.51	1.02	0.020	0.040

L SUFFIX
CERAMIC PACKAGE
CASE 620-02
($R_{\theta JA}$ = 100°C/W Typ)



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	–	22.10	–	0.870
B	6.10	6.60	0.240	0.260
C	–	5.08	–	0.200
D	0.38	0.53	0.015	0.021
F	–	1.78	–	0.070
G	2.54 BSC		0.100 BSC	
H	0.38	2.41	0.015	0.095
J	0.20	0.38	0.008	0.015
K	2.92	–	0.115	–
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	–	0.020	–
R	–	8.26	–	0.325

NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSION "B" DOES NOT INCLUDE MOLD FLASH.
4. "F" DIMENSION IS FOR FULL LEADS. "HALF" LEADS ARE OPTIONAL AT LEAD POSITIONS 1, 8, 9, and 16).
5. DIMENSION "R" TO BE MEASURED AT THE TOP OF THE LEADS (NOT AT THE TIPS).

P SUFFIX
PLASTIC PACKAGE
CASE 648-04
($R_{\theta JA}$ = 100°C/W Typ)

THERMAL INFORMATION

The maximum power consumption an integrated circuit can tolerate at a given operating ambient temperature, can be found from the equation:

$$P_{D(T_A)} = \frac{T_{J(max)} - T_A}{R_{\theta JA(Typ)}}$$

Where: $P_{D(T_A)}$ = Power Dissipation allowable at a given operating ambient temperature. This must be greater than the sum of the products of the supply voltages and supply currents at the worst case operating condition.

$T_{J(max)}$ = Maximum Operating Junction Temperature as listed in the Maximum Ratings Section

T_A = Maximum Desired Operating Ambient Temperature

$R_{\theta JA(Typ)}$ = Typical Thermal Resistance Junction to Ambient

MOTOROLA

MC6889
MC8T28

This device may be ordered under either of the above type numbers.

NON-INVERTING
QUAD THREE-STATE BUS TRANSCEIVER

This quad three-state bus transceiver features both excellent MOS or MPU compatibility, due to its high impedance PNP transistor input, and high-speed operation made possible by the use of Schottky diode clamping. Both the -48 mA driver and -20 mA receiver outputs are short-circuit protected and employ three-state enabling inputs.

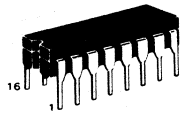
The device is useful as a bus extender in systems employing the M6800 family or other comparable MPU devices. The maximum input current of 200 μA at any of the device input pins assures proper operation despite the limited drive capability of the MPU chip. The inputs are also protected with Schottky-barrier diode clamps to suppress excessive undershoot voltages.

Propagation delay times for the driver portion are 17 ns maximum while the receiver portion runs 17 ns. The MC8T28 is identical to the NE8T28 and it operates from a single +5 V supply.

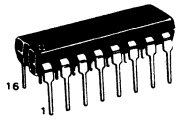
- High Impedance Inputs
- Single Power Supply
- High Speed Schottky Technology
- Three-State Drivers and Receivers
- Compatible with M6800 Family Microprocessor
- Non-Inverting

NON-INVERTING
BUS TRANSCEIVER

MONOLITHIC SCHOTTKY INTEGRATED CIRCUITS

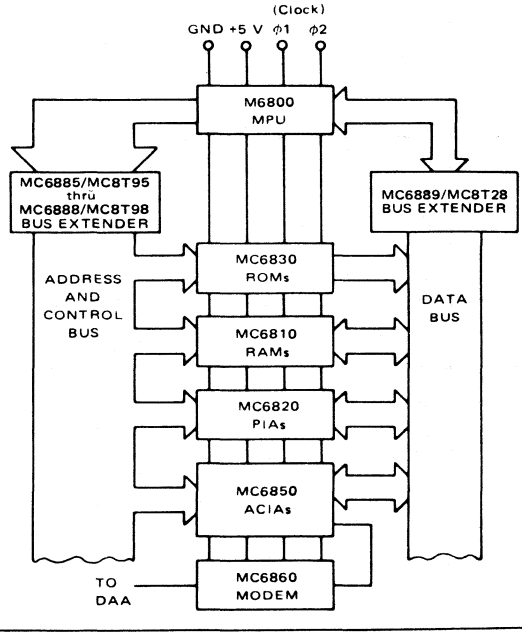


L SUFFIX
CERAMIC PACKAGE
CASE 620

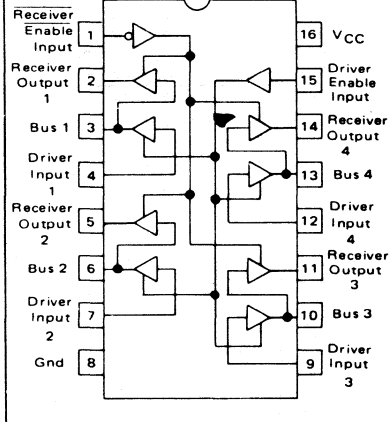


P SUFFIX
PLASTIC PACKAGE
CASE 648

MICROPROCESSOR BUS EXTENDER APPLICATION



PIN CONNECTIONS – MC6889 MC8T28



ORDERING INFORMATION

Device	Alternate	Temperature Range	Package
MC6889L	MC8T28L	0 to +75°C	Ceramic DIP
MC6889P	MC8T28P	0 to +75°C	Plastic DIP

MAXIMUM RATINGS (T_A = 25°C unless otherwise noted.)

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	8.0	Vdc
Input Voltage	V_I	5.5	Vdc
Junction Temperature Ceramic Package Plastic Package	T_J	 175 150	°C
Operating Ambient Temperature Range	T_A	0 to +75	°C
Storage Temperature Range	T_{stg}	-65 to +150	°C

ELECTRICAL CHARACTERISTICS (4.75 V ≤ V_{CC} ≤ 5.25 V and 0°C ≤ T_A ≤ 75°C unless otherwise noted.)

Characteristic	Symbol	Min	Typ	Max	Unit
Input Current – Low Logic State					
(Receiver Enable Input, $V_{IL(RE)}$ = 0.4 V)	$I_{IL(\overline{RE})}$	–	–	-200	μA
(Driver Enable Input, $V_{IL(DE)}$ = 0.4 V)	$I_{IL(DE)}$	–	–	-200	
(Driver Input, $V_{IL(D)}$ = 0.4 V)	$I_{IL(D)}$	–	–	-200	
(Bus (Receiver) Input, $V_{IL(B)}$ = 0.4 V)	$I_{IL(B)}$	–	–	-200	
Input Disabled Current – Low Logic State (Driver Input, $V_{IL(D)}$ = 0.4 V)	$I_{IL(D)\ DIS}$	–	–	-25	μA
Input Current-High Logic State					
(Receiver Enable Input, $V_{IH(RE)}$ = 5.25 V)	$I_{IH(\overline{RE})}$	–	–	25	μA
(Driver Enable Input, $V_{IH(DE)}$ = 5.25 V)	$I_{IH(DE)}$	–	–	25	
(Driver Input, $V_{IH(D)}$ = 5.25 V)	$I_{IH(D)}$	–		25	
Input Voltage – Low Logic State					
(Receiver Enable Input)	$V_{IL(\overline{RE})}$	–	–	0.85	V
(Driver Enable Input	$V_{IL(DE)}$	–	–	0.85	
(Driver Input)	$V_{IL(D)}$	–	–	0.85	
(Receiver Input)	$V_{IL(B)}$	–	–	0.85	
Input Voltage – High Logic State					
(Receiver Enable Input)	$V_{IH(\overline{RE})}$	2.0	–	–	V
(Driver Enable Input)	$V_{IH(DE)}$	2.0	–	–	
(Driver Input)	$V_{IH(D)}$	2.0	–	–	
(Receiver Input)	$V_{IH(B)}$	2.0	–	–	
Output Voltage – Low Logic State					
(Bus Driver) Output, $I_{OL(B)}$ = 48 mA)	$V_{OL(B)}$	–	–	0.5	V
(Receiver Output, $I_{OL(R)}$ = 20 mA)	$V_{OL(R)}$	–	–	0.5	
Output Voltage – High Logic State					
(Bus (Driver) Output, $I_{OH(B)}$ = -10 mA)	$V_{OH(B)}$	2.4	3.1	–	V
(Receiver Output, $I_{OH(R)}$ = -2.0 mA)	$V_{OH(R)}$	2.4	3.1	–	
(Receiver Output, $I_{OH(R)}$ = -100 μA, V_{CC} = 5.0 V)		3.5	–	–	
Output Disabled Leakage Current – High Logic State					
(Bus Driver) Output, $V_{OH(B)}$ = 2.4 V)	$I_{OHL(B)}$	–	–	100	μA
(Receiver Output, $V_{OH(R)}$ = 2.4 V)	$I_{OHL(R)}$	–	–	100	
Output Disabled Leakage Current – Low Logic State					
(Bus Output, $V_{OL(B)}$ = 0.5 V)	$I_{OLL(B)}$	–	–	-100	μA
(Receiver Output, $V_{OL(R)}$ = 0.5 V)	$I_{OLL(R)}$	–	–	-100	
Input Clamp Voltage					
(Driver Enable Input $I_{ID(DE)}$ = -12 mA)	$V_{IC(DE)}$	–	–	-1.0	V
(Receiver Enable Input $I_{IC(RE)}$ = +12 mA)	$V_{IC(RE)}$	–	–	-1.0	
(Driver Input $I_{IC(D)}$ = -12 mA)	$V_{IC(D)}$	–	–	-1.0	
Output Short-Circuit Current, V_{CC} = 5.25 V (1)					
(Bus (Driver) Output)	$I_{OS(B)}$	-50	–	-150	mA
(Receiver Output)	$I_{OS(R)}$	-30	–	-75	
Power Supply Current (V_{CC} = 5.25 V)	I_{CC}	–	–	87	mA

(1) Only one output may be short-circuited at a time.

SWITCHING CHARACTERISTICS (Unless otherwise noted, V_{CC} = 5.0 V and T_A = 25°C)

Characteristic	Symbol	Min	Max	Unit
Propagation Delay Time–Receiver (C_L = 30 pF)	$t_{PLH(R)}$ $t_{PHL(R)}$	–	17 17	ns
Propagation Delay Time–Driver (C_L = 300 pF)	$t_{PLH(D)}$ $t_{PHL(D)}$	–	17 17	ns
Propagation Delay Time–Enable (C_L = 30 pF) – Receiver – Driver Enable (C_L 300 pF)	$t_{PZL(R)}$ $t_{PLZ(R)}$ $t_{PZL(D)}$ $t_{PLZ(D)}$	– – – –	23 18 28 23	ns

FIGURE 1 – TEST CIRCUIT AND WAVEFORMS FOR PROPAGATION DELAY FROM BUS (RECEIVER) INPUT TO RECEIVER OUTPUT, $t_{PLH(R)}$ AND $t_{PHL(R)}$

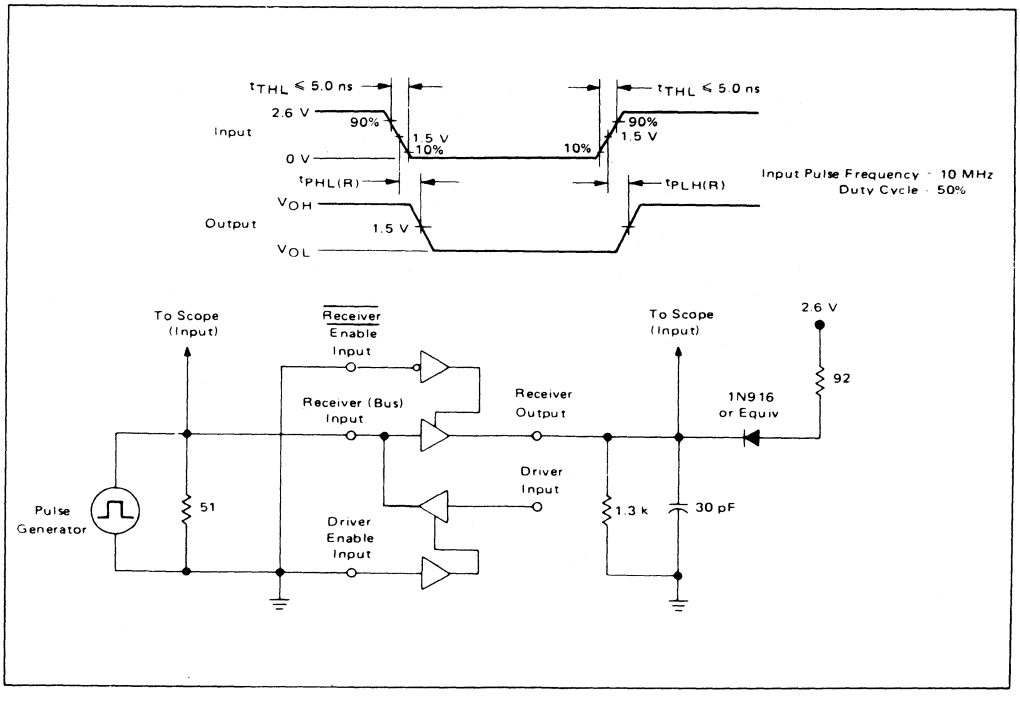


FIGURE 2 – TEST CIRCUIT AND WAVEFORMS FOR PROPAGATION DELAY TIME FROM DRIVER INPUT TO BUS (DRIVER) OUTPUT, $t_{PLH(D)}$ AND $t_{PHL(D)}$

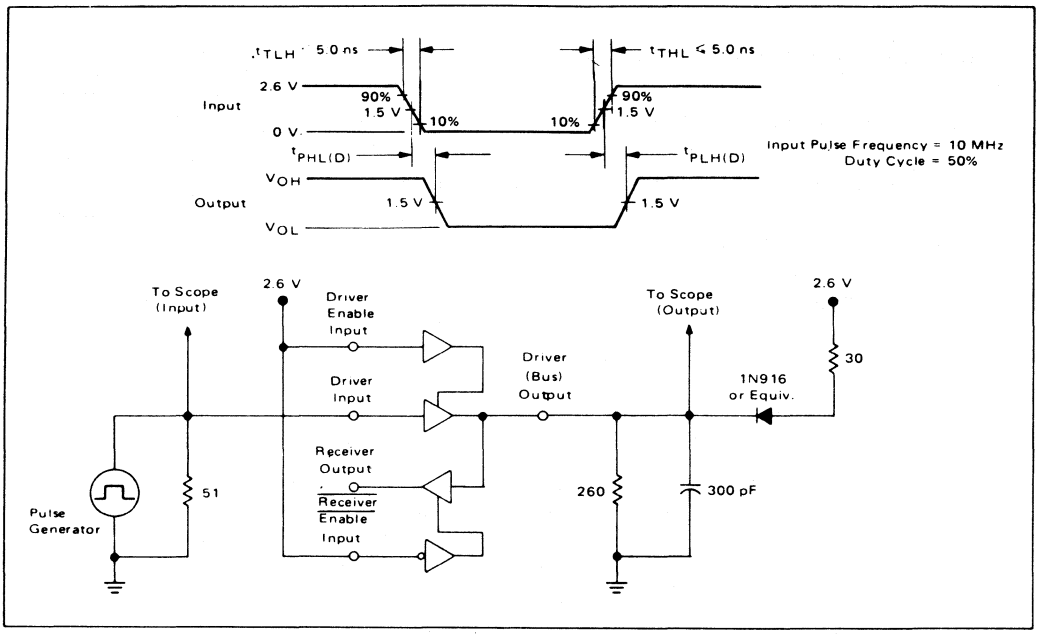


FIGURE 3 – TEST CIRCUIT AND WAVEFORMS FOR PROPAGATION DELAY TIME FROM RECEIVER ENABLE INPUT TO RECEIVER OUTPUT, $t_{PLZ(RE)}$ AND $t_{PZL(RE)}$

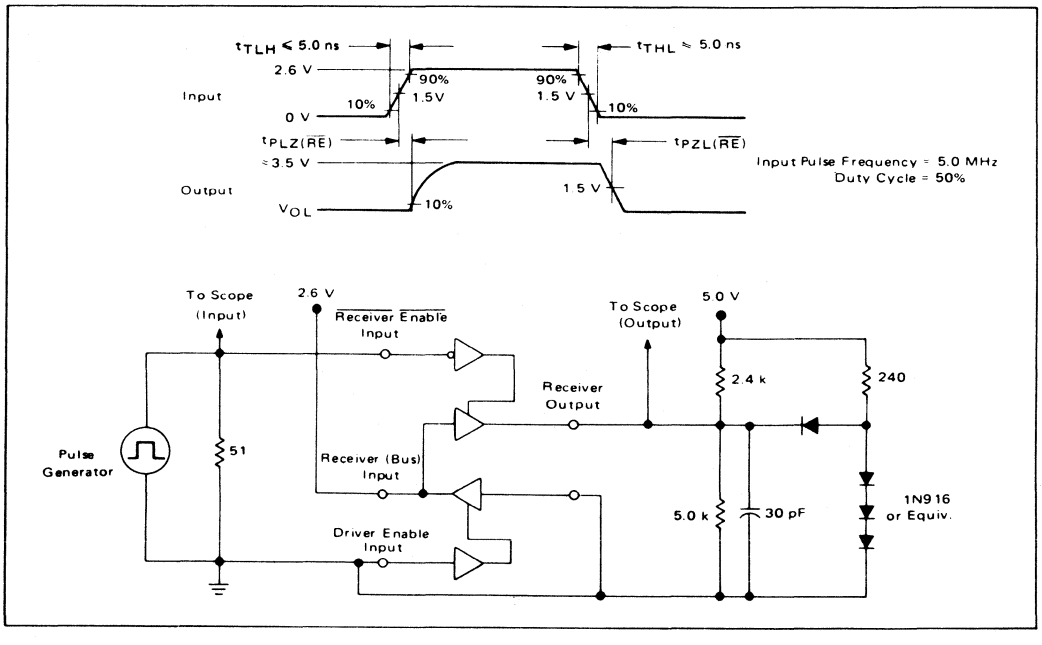


FIGURE 4 – TEST CIRCUIT AND WAVEFORMS FOR PROPAGATION DELAY TIMES FROM DRIVER ENABLE INPUT TO DRIVER (BUS) OUTPUT, $t_{PLZ(DE)}$ AND $t_{PZL(DE)}$

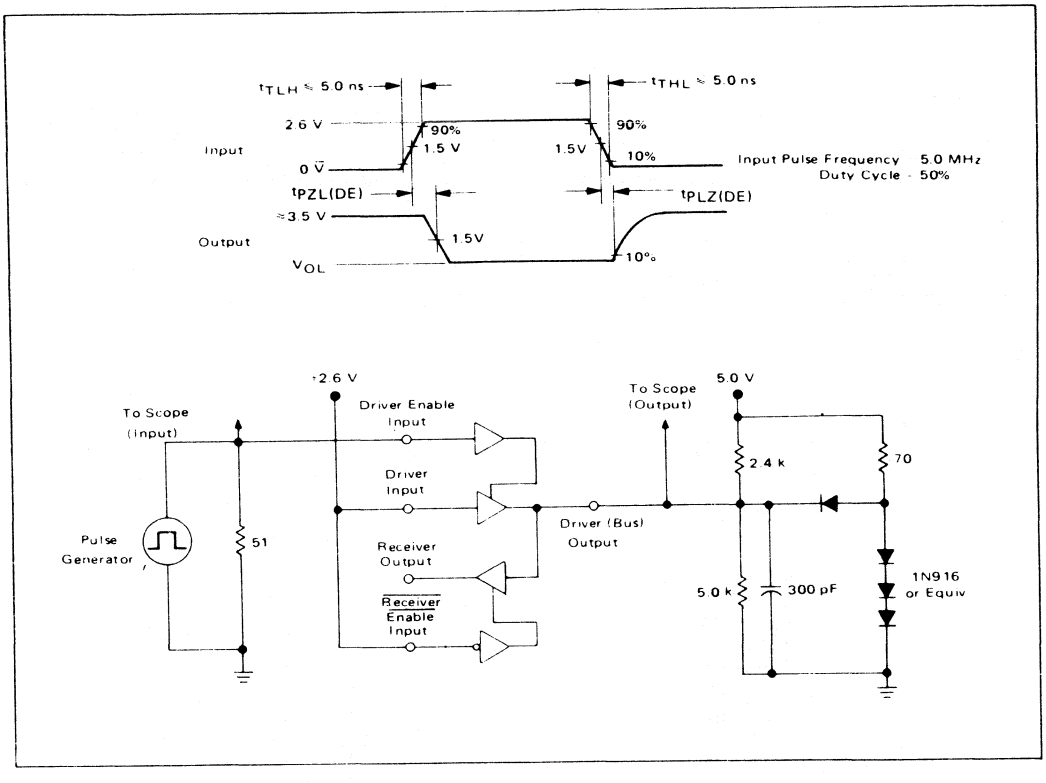
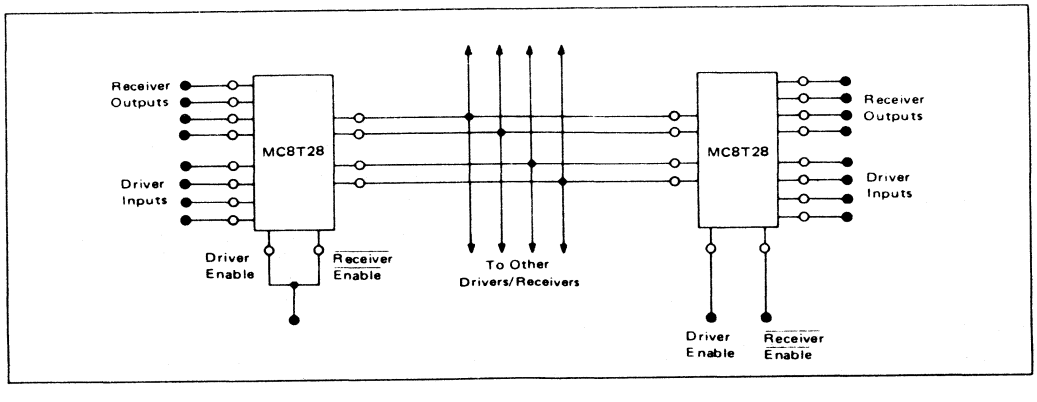
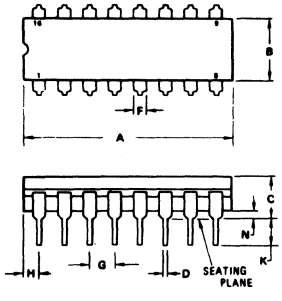


FIGURE 5 – BIDIRECTIONAL BUS APPLICATIONS



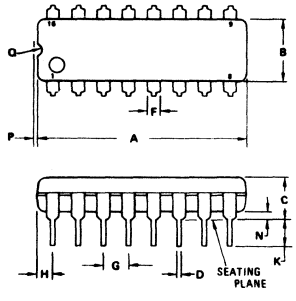


L SUFFIX
CERAMIC PACKAGE
CASE 620-02

$R_{\theta JA}$ = 100°C/W (Typ)

NOTES:
1 LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION
2 PKG. INDEX: NOTCH IN LEAD NOTCH IN CERAMIC OR INK DOT
3 DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	19.05	19.81	0.750	0.780
B	6.22	6.98	0.245	0.275
C	4.06	5.08	0.160	0.200
D	0.38	0.51	0.015	0.020
F	1.40	1.65	0.055	0.065
G	2.54 BSC		0.100 BSC	
H	0.51	1.14	0.020	0.045
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	7.37	7.87	0.290	0.310
M	–	15°	–	15°
N	0.51	1.02	0.020	0.040



P SUFFIX
PLASTIC PACKAGE
CASE 648-04

$R_{\theta JA}$ = 100°C/W (Typ)

NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	–	22.10	–	0.870
B	6.10	6.60	0.240	0.260
C	–	5.08	–	0.200
D	0.38	0.53	0.015	0.021
F	–	1.78	–	0.070
G	2.54 BSC		0.100 BSC	
H	0.38	2.41	0.015	0.095
J	0.20	0.38	0.008	0.015
K	2.92	–	0.115	–
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	–	0.020	–
R	–	8.26	–	0.325

THERMAL INFORMATION

The maximum power consumption an integrated circuit can tolerate at a given operating ambient temperature, can be found from the equation:

$$P_{D(T_A)} = \frac{T_{J(max)} - T_A}{R_{\theta JA(Typ)}}$$

Where: $P_{D(T_A)}$ = Power Dissipation allowable at a given operating ambient temperature. This must be greater than the sum of the products of the supply voltages and supply currents at the worst case operating condition.

$T_{J(max)}$ = Maximum Operating Junction Temperature as listed in the Maximum Ratings Section

T_A = Maximum Desired Operating Ambient Temperature

$R_{\theta JA(Typ)}$ = Typical Thermal Resistance Junction to Ambient

MOTOROLA

MC68488

Advance Information

GENERAL PURPOSE INTERFACE ADAPTER

The MC68488 GPIA provides the means to interface between the IEEE488 standard instrument bus and the MC6800. The 488 instrument bus provides a means for controlling and moving data from complex systems of multiple instruments.

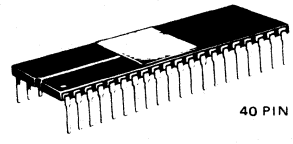
The MC68488 will automatically handle all handshake protocol needed on the instrument bus.

- Single or dual primary address recognition
- Secondary address capability
- Complete source and acceptor handshakes
- Programmable interrupts
- RFD holdoff to prevent data overrun
- Operates with DMA controller
- Serial and parallel polling capability
- Talk-only or listen-only capability
- Selectable automatic features to minimize software
- Synchronization trigger output
- M6800 bus compatible

MOS
(N-CHANNEL, SILICON-GATE DEPLETION LOAD)

GENERAL PURPOSE INTERFACE ADAPTER

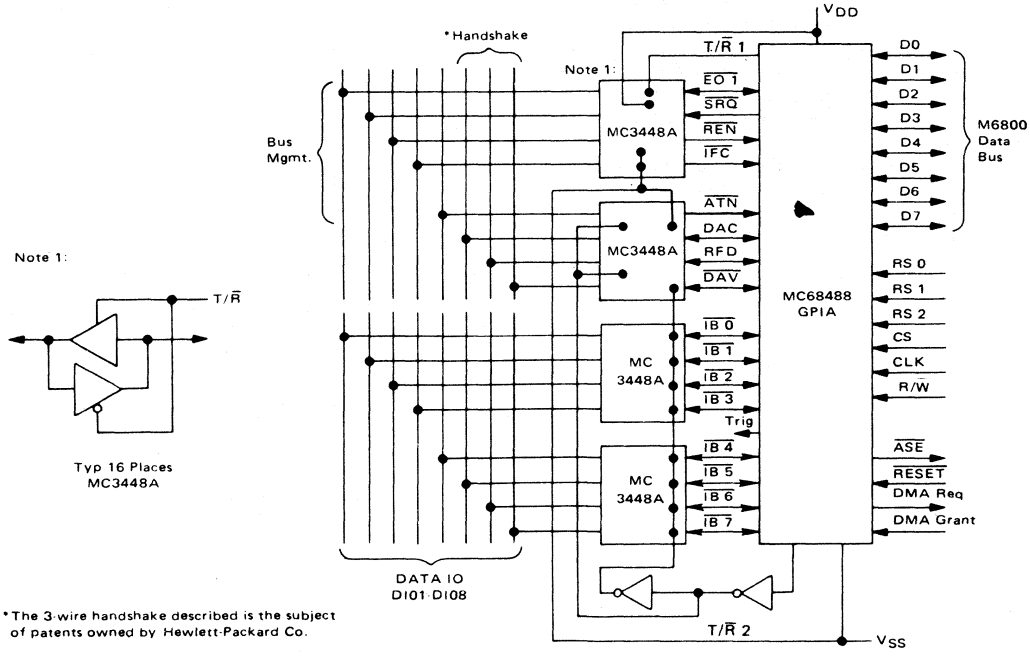
L SUFFIX
CERAMIC PACKAGE
CASE 715



NOT SHOWN:

P SUFFIX
PLASTIC PACKAGE
CASE 711

FIGURE 1



GENERAL DESCRIPTION

The IEEE 488 instrument bus standard is a bit-parallel, byte-serial bus structure designed for communication to and from intelligent instruments. Using this standard, many instruments may be interconnected and remotely and automatically controlled or programmed. Data may be taken from, sent to, or transferred between instruments. A bus controller dictates the role of each device by making the attention line true and sending talk or listen addresses on the instrument bus data lines; those devices which have matching addresses are activated. Device addresses are set into each GPIA from switches or jumpers on a PC board by a microprocessor as a part of the initialization sequence.

When the controller makes the attention line true, instrument bus commands may also be sent to single or multiple GPIA's.

Information is transmitted on the instrument bus data lines under sequential control of the three handshake lines. No step in the sequence can be initiated until the previous step is completed. Information transfer can proceed as fast as the devices can respond, but no faster than the slowest device presently addressed as active. This permits several devices of different speeds to receive the same data concurrently.

The GPIA is designed to work with standard 488-bus driver IC's (MC3448A's) to meet the complete electrical specifications of the IEEE 488 bus. Additionally, a powered-off instrument may be powered-on without disturbing the 488 bus. With some additional logic, the GPIA could be used with other microprocessors.

The MC68488 GPIA has been designed to interface between the MC6800 microprocessor and the complex protocol of the IEEE 488 instrument bus. Many instrument bus protocol functions are handled automatically by the GPIA and require no additional MPU action. Other functions require minimum MPU response due to a large number of internal registers conveying information on the state of the GPIA and the instrument bus.

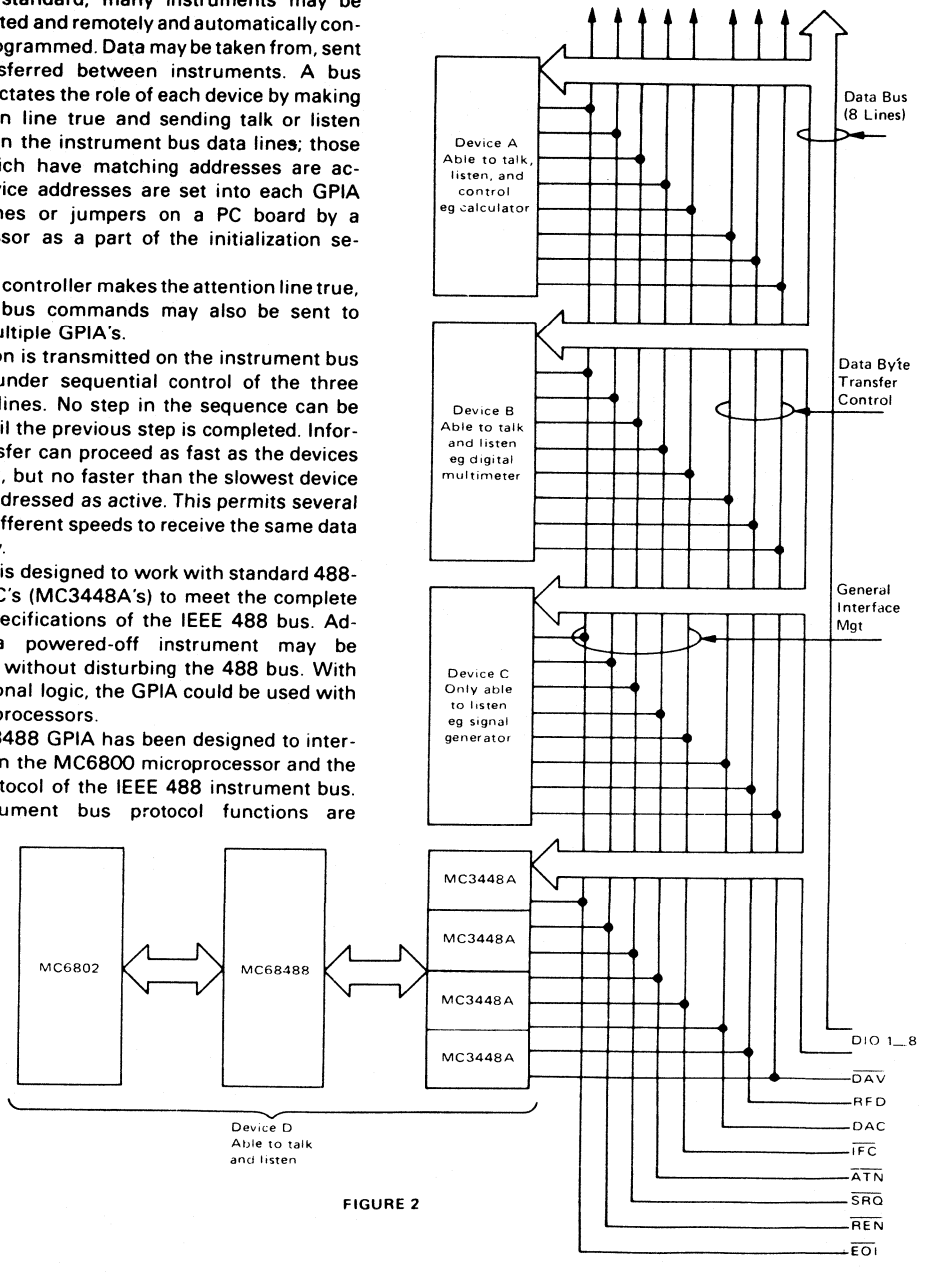


FIGURE 2

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	-0.3 to +7.0	Vdc
Input Voltage	V_{in}	-0.3 to +7.0	Vdc
Operating Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	-55 to +150	°C
Thermal Resistance	θ_{JA}	82.5	°C/W

This device contains circitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit.

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ±5%, V_{SS} = 0, T_A = 0 to 70°C unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage		V_{IH}	V_{SS} +2.0	–	V_{CC}	Vdc
Input Low Voltage		V_{IL}	V_{SS} -0.3	–	V_{SS} +0.8	
Input Leakage Current (V_{in} = 0 to 5.25 V)		I_{in}	–	1.0	2.5	μAdc
Three-State (Off State) Input Current (V_{in} = 0.4 to 2.4 V)	D0–D7	I_{TSI}	–	2.0	10	μAdc
Output High Voltage (I_{load} = -205 μA)	D0–D7	V_{OH}	V_{SS} +2.4	–	–	Vdc
Output Low Voltage (I_{load} = 1.6 mA)	D0–D7	V_{OL}	–	–	V_{SS} +0.4	Vdc
(I_{load} = 3.2 mA)	$\overline{IRQ}$		–	–	V_{SS} +0.4	
Output Leakage Current (Off State) (V_{OH} = 2.4 Vdc)	$\overline{IRQ}$	I_{LOH}	–	1.0	10	μAdc
Power Dissipation		P_D	–	600	–	mW
Input Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	D0–D7	C_{in}	–	–	12.5	pF
	All others		–	–	7.5	

FIGURE 3 – SOURCE and ACCEPTOR HANDSHAKE

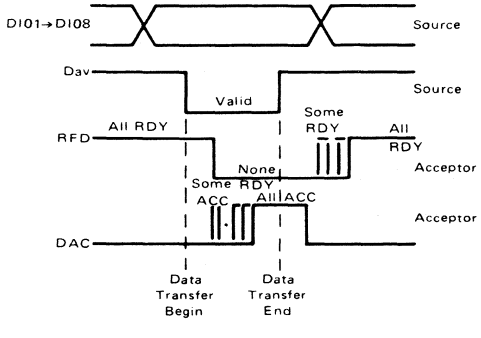
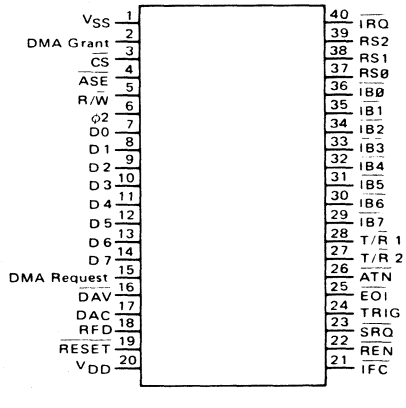


FIGURE 4 – PIN ASSIGNMENT



BUS TIMING CHARACTERISTICS

Characteristic	Symbol	Min	Max	Unit
READ (See Figure 5)				
Enable Cycle Time	t_{cycE}	1.0	–	µs
Enable Pulse Width, High	PW_{EH}	0.45	–	µs
Enable Pulse Width, Low	PW_{EL}	0.43	–	µs
Setup Time, Address and R/$\overline{W}$ valid to enable positive transition	t_{AS}	160	–	ns
Data Delay Time	t_{DDR}	–	320	ns
Data Hold Time	t_H	10	–	ns
Address Hold Time	t_{AH}	10	–	ns
Rise and Fall Time for Enable input	t_{Er}, t_{Ef}	–	25	ns
WRITE (See Figure 6)				
Enable Cycle Time	t_{cycE}	1.0	–	µs
Enable Pulse Width, High	PW_{EH}	0.45	–	µs
Enable Pulse Width, Low	PW_{EL}	0.43	–	µs
Setup Time, Address and R/$\overline{W}$ valid to enable positive transition	t_{AS}	160	–	ns
Data Setup Time	t_{DSW}	195	–	ns
Data Hold Time	t_H	10	–	ns
Address Hold Time	t_{AH}	10	–	ns
Rise and Fall Time for Enable input	t_{Er}, T_{Ef}	–	25	ns
OUTPUT (See Figure 7)				
Output Delay Time				
$\overline{DAV}$, DAC, RFD, $\overline{EOI}$, $\overline{ATN}$ valid	t_{HD}	–	400	ns
T/$\overline{R}$1, T/$\overline{R}$2 valid	$t_{T/\overline{R}1, 2D}$	–	400	ns

FIGURE 5 – BUS READ TIMING CHARACTERISTICS
(Read Information from GPIA)

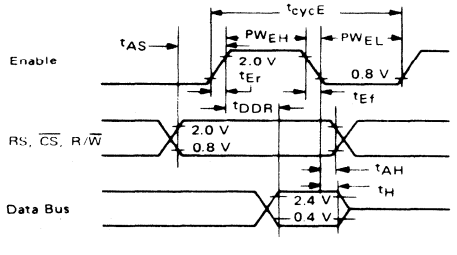


FIGURE 6 – BUS WRITE TIMING CHARACTERISTICS
(Write Information into GPIA)

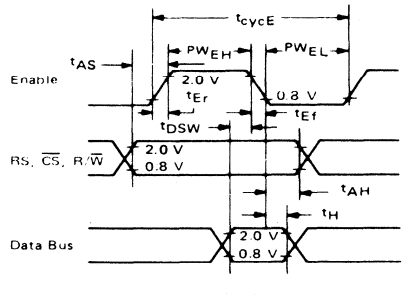
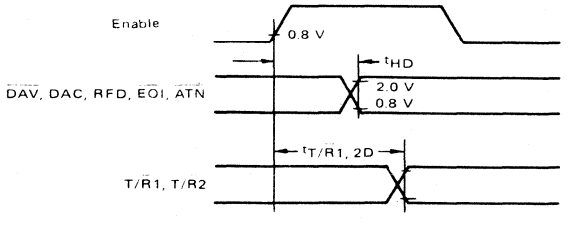


FIGURE 7 – OUTPUT BUS TIMING



AC TIME VALUES

Characteristics		Symbol*	Typ	Unit
Settling Time for Multiple Message	SH	T_1	$\geqslant 2$	μs**
Response to $\overline{ATN}$	SH, AH, T, L	t_2	$\leqslant 200$	ns
Interface Message Accept Time ‡	AH	T_3	>0	§
Response to $\overline{IFC}$ or $\overline{REN}$ False	T, TE, L, LE	t_4	<100	μs
Response to $\overline{ATN} \bullet \overline{EOI}$	PP	t_5	$\leqslant 200$	ns

* Time values specified by a lower case t indicate the maximum time allowed to make a state transition. Time values specified by an upper case T indicate the minimum time that a function must remain in a state before exiting.

**If three-state drivers are used on the $\overline{DIO}$ – $\overline{DAV}$ and $\overline{EOI}$ lines, T_1 may be:
(1) $\geqslant$1100 ns
(2) Or $\geqslant$700 ns if it is known that within the controller $\overline{ATN}$ is driven by a three-state driver.
(3) Or $\geqslant$500 ns for all subsequent bytes following the first sent after each false transition of $\overline{ATN}$ [the first byte must be sent in accordance with (1) or (2)].

‡ Time required for interface functions to accept, not necessarily respond to interface messages.

§ Implementation dependent.

MPU bus clock rate – The current 6800 bus clock is $\leqslant$ 1 MHz but part should operate at 1.5 MHz (design goal), with appropriate settling times (T1).

GPIA/MPU INTERFACE SIGNALS

The MC68488 interfaces to the M6800 MPU with an eight-bit bidirectional data bus, a chip select, Read/$\overline{Write}$ line, $\overline{reset}$ line, three register select lines, an interrupt request line, two DMA control lines, and an address switch enable line.

Bidirectional Data (D0-D7)—The bidirectional data lines allow the transfer of data between the MPU and the GPIA. The data bus output drives are three state devices that remain in the high impedance (off) state except when the MPU performs a GPIA read operation. The Read/$\overline{Write}$ line is in the read state when the GPIA is selected for a read operation.

Chip Select ($\overline{CS}$)—This input signal is used to select the GPIA. $\overline{CS}$ must be low for selection of the device. Chip select decoding is normally accomplished with logic external to the chip.

Read/$\overline{Write}$ Line (R/$\overline{W}$)—This signal is generated by the MPU to control register access and direction of data transfer on the data bus. A low state on the GPIA Read/$\overline{Write}$ allows for the selection of one of seven write only registers when used in conjunction with the register select lines; RS0, RS1, RS2. A high state on the GPIA Read/$\overline{Write}$ allows for the selection of one of eight read only registers when used in conjunction with register select lines RS0, RS1, RS2.

Register Select (RS0, RS1, RS2)—The three register select lines are used to select the various registers inside the GPIA. These three lines are used in conjunction with the Read/$\overline{Write}$ line to select a particular register that is to be written or read. Table 1 shows the register select coding.

Interrupt Request ($\overline{IRQ}$)—The $\overline{IRQ}$ output goes to the common interrupt bus for the MPU. This is an open drain output which is wire-ORed to the $\overline{IRQ}$ bus. The $\overline{IRQ}$ is set false (low) when an enabled interrupt occurs and stays false until the MPU reads from the interrupt status register.

$\overline{Reset}$—The active low $\overline{Reset}$ line is used to initialize the chip during power on start up. $\overline{Reset}$ will be driven by an external power-up reset circuit.

DMA Control Lines (DMA Grant, DMA Request)—The DMA request line is used to signal waiting data when Byte In (BI) or Byte Out (BO) is set high for a DMA controller. The DMA request line is set high if either the BI or BO interrupt flags are set in the Interrupt Status Register (ROW) and the corresponding bits in the Interrupt Mask Register (ROR) are set true. The DMA request line is cleared when the DMA grant is made true. The DMA grant line is used to signal the GPIA that the DMA controller has control of the MPU data and address lines. **DMA Grant must be grounded when not in use!*

Address Switch Enable ($\overline{ASE}$)—The $\overline{ASE}$ output is used to enable the device address switch three state buffers to allow the instrument address switch to be read on the MPU bus.

Clock Input (Clk)—The Clk input is normally a derivative of the MPU $\phi 2$ clock.

TABLE 1 — Register Access

RS2	RS1	RS0	R/W	Register Title	Register Symbol
0	0	0	1	Interrupt Status	R0R
0	0	0	0	Interrupt Mask	R0W
0	0	1	1	Command Status	R1R
0	0	1	0	Unused	—
0	1	0	1	Address Status	R2R
0	1	0	0	Address Mode	R2W
0	1	1	1	Auxiliary Command	R3R
0	1	1	0	Auxiliary Command	R3W
1	0	0	1	Address Switch*	R4R
1	0	0	0	Address	R4W
1	0	1	1	Serial Poll	R5R
1	0	1	0	Serial Poll	R5W
1	1	0	1	Command Pass-Through	R6R
1	1	0	0	Parallel Poll	R6W
1	1	1	1	Data In	R7R
1	1	1	0	Data Out	R7W

*External to MC68488.

GPIA/488 Interface Bus Signals

The GPIA provides a set of eighteen interface signal lines between the M6800 and the IEEE Standard 488 bus.

Signal Lines ($\overline{IB0}$-$\overline{IB7}$)—These bidirectional lines allow for the flow of seven bit ASCII interface messages and device dependent messages. Data appears on these lines in a bit-parallel byte-serial form. These lines are buffered by the MC3448A transceivers and applied to the 488 bus (DIO1-DIO8).

Byte Transfer Lines (DAC, RFD, $\overline{DAV}$)—These lines allow for proper transfer of each data byte on the bus between sources and acceptors. RFD goes passively true indicating that all acceptors are "ready for data." A source will indicate the "data is valid" by pulling $\overline{DAV}$ low. Upon the reception of valid data by all acceptors, DAC will go passively true indicating that the "data has been accepted" by all acceptors.

Bus Management Lines ($\overline{ATN}$, $\overline{IFC}$, $\overline{SRQ}$, $\overline{EOI}$, $\overline{REN}$)—These lines are used to manage an orderly flow of information across the interface lines.

Attention ($\overline{ATN}$)—Is sent true over the interface to disable current talker and listeners freeing the signal lines ($\overline{IB0}$-$\overline{IB7}$). During the $\overline{ATN}$ active state devices monitor the DIO1 for addressing or an interface command. Data flows on the DIO1 lines when $\overline{ATN}$ is inactive (high).

Interface Clear ($\overline{IFC}$)—Is used to put the interface system into a known quiescent state.

Service Request ($\overline{SRQ}$)—Is used to indicate a need for attention in addition to requesting an interruption in the current sequence of events. This indicates to the controller that a device on the bus is in need of service.

Remote Enable ($\overline{REN}$)—is used to select one of two alternate sources of device programming data, local or remote control.

End of Identify ($\overline{EOI}$)—is used to signal the end of a multiple byte transfer sequence and in conjunction with $\overline{ATN}$ executes a parallel polling sequence.

Transmit/$\overline{\text{Receive}}$ Control Signals (T/$\overline{R}$ 1, T/$\overline{R}$ 2)—These two signals are used to control the quad transceivers which drive the interface bus. It is assumed that transceivers equivalent to the MC3448A will be used where each transceiver has a separate transmit/$\overline{\text{receive}}$ control pin. These pins can support one TTL load each. The outputs can then be grouped as shown in Figure 1 with $\overline{SRQ}$ hardwired high to transmit. The transmit/$\overline{\text{receive}}$ inputs of $\overline{REN}$, $\overline{IFC}$, and $\overline{ATN}$ are hardwired low to receive. $\overline{EOI}$ is controlled by T/$\overline{R}$1 through the MC3448A (or an equivalent) allowing it to transmit or receive. T/$\overline{R}$1 operates exactly as T/$\overline{R}$2 except during the parallel polling sequence. During parallel poll $\overline{EOI}$ will be made an input by T/$\overline{R}$1 while $\overline{DAV}$ and $\overline{IB0}$/$\overline{IB7}$ lines are outputs.

GPIA INTERNAL CONTROLS AND REGISTERS*

There are fifteen locations accessible to the MPU data bus which are used for transferring data to control the various functions on the chip and provide current chip status. Seven of these registers are write only and eight registers are read only. The various registers are accessed according to the three least significant bits of the MPU address bus and the status of the Read/Write line. One of the

*NOTE: Upper and lower case type designations will be used with the register bits to indicate remote or local messages respectively.

fifteen registers is external to the IC but an address switch register is provided for reading the address switches. Table 2 shows actual bit contents of each of the registers.

Data-In Register R7R—The data-in register is an actual eight-bit storage register used to move data from the interface bus when the chip is a listener. Reading the register does not destroy information in the data-out register. DAC (data accepted) will remain low until the MPU removes the byte from the data-in register. The chip will automatically finish the handshake by allowing DAC to go high. In RFD (ready for data) holdoff mode, a new handshake is not initiated until a command is sent allowing the chip to release holdoff. This will delay a talker until the available information has been processed.

DATA IN REGISTER
(Read Only)

DI7	DI6	DI5	DI4	DI3	DI2	DI1	DIØ

DIØ DI7 Correspond to DIO1 DIO8 of the 488 1975 Standard and IBØ IB7 of the MC68488

Data-Out Register R7W — The data-out register is an actual eight-bit storage register used to move data out of the chip onto the interface bus. Reading from the data-in register has no effect on the information in the data-out register. Writing to the data-out register has no effect on the information in the data-in register.

DATA OUT REGISTER
(Write Only)

DO7	DO6	DO5	DO4	DO3	DO2	DO1	DOØ

DOØ DO7 Correspond to DIO1 DIO8 of the 488 1975 Standard and IBØ IB7 of the MC68488

Interrupt Mask Register R0W — The Interrupt Mask Register is a seven-bit storage register used to select the particular events that will cause an interrupt to be sent to the MPU. The seven control bits may be set independently of each other. If dsel (bit 7 of the Address Mode Register) is set high CMD bit 2 will interrupt on SPAS or RLC. If dsel is set low CMD will interrupt on UACG, UUCG, and DCAS in addition to RLC and SPAS. The Command Status

TABLE 2

	7	6	5	4	3	2	1	0	
R0W	IRQ	BO	GET		APT	CMD	END	BI	Interrupt "Mask" Reg.
R0R	INT	BO	GET		APT	CMD	END	BI	Interrupt Status Reg.
R1R	UACG	REM	LOK		RLC	SPAS	DCAS	UUCG	Command Status Reg.
R1W									Unused
R2R	ma	to	lo	ATN	TACS	LACS	LPAS	TPAS	Address Status Reg.
R2W	dsel	to	lo		hlde	hlda		apte	Address Mode Reg.
R3R	Reset	DAC	$\overline{\text{DAV}}$	RFD	msa	rtl	ulpa	fget	Auxiliary Command Reg.
R3W		rfdr	feoi	dacr			dacd		
R4R	UD3	UD2	UD1	AD5	AD4	AD3	AD2	AD1	Address Switch Reg.
R4W	lsbe	dal	dat	AD5	AD4	AD3	AD2	AD1	Address Register
R5R	S8	SRQS	S6	S5	S4	S3	S2	S1	Serial Poll Reg.
R5W		rsv							
R6R	B7	B6	B5	B4	B3	B2	B1	B0	Command Pass thru Reg.
R6W	PPR8	PPR7	PPR6	PPR5	PPR4	PPR3	PPR2	PPR1	Parallel Poll Reg.
R7R	D17	D16	D15	D14	D13	D12	DI1	DI0	Data In Register
R7W	DO7	DO6	DO5	DO4	DO3	DO2	DO1	DO0	Data Out Register

Register R1R may then be used to determine which command caused the interrupt. Setting GET bit 5 allows an interrupt to occur on Group Execute Trigger Command. END bit 1 allows an interrupt to occur if $\overline{EOI}$ is true (low) and $\overline{ATN}$ is false (high). APT bit 3 allows an interrupt to occur indicating that a secondary address is available to be examined by the MPU if apte (bit 0 of Address Mode Register) is enabled and listener or talker primary address is received and a Secondary Command Group is received. A typical response for a valid secondary address would be to set msa (bit 3 of Auxiliary Command Register) true and dacr (bit 4 Auxiliary Command Register) true, releasing the DAC handshake. BI indicates that a data byte is waiting in the data-in register. BI is set high when data-in register is full. BO indicates that a byte from the data-out register has been accepted. BO is set when the data-out register is empty. IRQ enabled high allows any interrupt to be passed to the MPU.

INTERRUPT MASK REGISTER

(Write Only)

IRQ	BO	GET	X	APT	CMD	END	BI

IRQ – Mask bit for IRQ pin
BO – Interrupt on byte output
GET – Interrupt on Group Execute Trigger
APT – Interrupt on Secondary Address Pass Through
CMD – Interrupt on SPAS + RLC + dsel (DCAS + UUCG + UACG)
END – Interrupt on EOI and $\overline{ATN}$
BI – Interrupt on byte input

The Interrupt Status Register R0R — The Interrupt Status Register is a seven-bit storage register which corresponds to the interrupt mask register with an additional bit INT bit 7. Except for the INT bit the other bits in the status register are set regardless of the state of the interrupt mask register when the corresponding event occurs. The $\overline{IRQ}$ (MPU interrupt) is cleared when the MPU reads from the register. INT bit 7 is the logical OR of the other six bits ANDed with the respective bit of R0W.

INTERRUPT STATUS REGISTER

(Read Only)

INT	BO	GET	X	APT	CMD	END	BI

INT – Logical OR of all other bits in this register ANDed with the respective bits in the interrupt mask register.
BO – A byte of data has been output
GET – A Group Execute Trigger has occurred
APT – An Address Pass Through has occurred
CMD – SPAS + RLC + $\overline{dsel}$ (DCAS + UUCG + UACG) has occurred
END – An EOI has occurred with $\overline{ATN}$ = 0
BI – A byte has been received

Serial Poll Register R5R/W — The Serial Poll Register is an eight-bit storage register which can be both written into and read by the MPU. It is used for establishing the status byte that the chip sends out when it is serial poll enabled. Status may be placed in bits 0 through 5 and bit 7. Bit 6 rsv (request for service) is used to drive the logic which controls the $\overline{SRQ}$ line on the bus telling the controller that service is needed. This same logic generated the signal SRQS which is substituted in bit 6 position when the status byte is read by the MPU $\overline{IB0}$–$\overline{IB7}$. In order to initiate a rsv (request for service), the MPU sets bit 6 true (generating rsv signal) and this in turn causes the chip to pull down the $\overline{SRQ}$ line. SRQS is the same as rsv when SPAS is false. Bit 6 as read by the MPU will be the SRQS (Service Request State).

SERIAL POLL REGISTER

(Read)

S8	SRQS	S6	S5	S4	S3	S2	S1

S1-S8 – Status bits
SRQS – Bus in Service Request State

SERIAL POLL REGISTER

(Write)

S8	rsv	S6	S5	S4	S3	S2	S1

S1-S8 Status bits
rsv generate a service request

Parallel Poll Register R6W—This register will be loaded by the MPU and the bits in this register will be delivered to the instrument bus IB0-IB7 during PPAS (Parallel Poll Active State). This register powers up in the PP0 (Parallel Poll No Capability) state. The reset bit (Auxiliary Command Register bit 7) will clear this register to the PP0 state.

The parallel poll interface function is executed by this chip using the PP2 subset (Omit Controller Configuration Capability). The controller cannot directly configure the parallel poll output of this chip. This must be done by the MPU. The controller will be able to indirectly configure the parallel poll by issuing an addressed command which has been defined in the MPU software.

PARALLEL POLL REGISTER

(Write Only)

PP8	PP7	PP6	PP5	PP4	PP3	PP2	PP1

Bits delivered to bus during Parallel Poll Active State (PPAS)
Register powers up in the PP0 state
Parallel Poll is executed using the PP2 subset

Address Mode Register R2W — The address mode register is a storage register with six bits for control: to, lo, hlde, hlda, dsel, and apte. The to bit 6 selects the talker/listener and addresses the chip to talk only. The lo bit 5 selects the talker/listener and sets the chip to listen only. The apte bit 0 is used to enable the extended addressing mode. If apte is set low the device goes from the TPAS (Talker Primary

Address State) directly to the TADS (Talker Addressed State). The hlda bit 2 holds off RFD (Ready for Data) on ALL DATA until rfdr is set true. The hlde bit 3 holds off RFD on EOI enabled (low) and ATN not enabled (high). This allows the last byte in a block of data to be continually read as needed. Writing rfdr true (high) will allow the next handshake to proceed.

ADDRESS MODE REGISTER

(Write Only)

dsel	to	lo	X	hdle	hdla	X	apte

dsel – configure for automatic completion of handshake sequence on occurrence of GET, UACG, UUCG, SDC, or DCL commands
to – set to talk-only mode
lo – set to listen-only mode
hdle – Hold-off RFD on end
hdla – Hold-off RFD on all data
apte – Enable the address pass-through feature

Address Status Register R2R — The address status register is not a storage register but simply an eight-bit port used to couple internal signal nodes to the MPU bus. The status flags represented here are stored internally in the logic of the chip. These status bits indicate the addressed state of the talker/listener as well as flags that specify whether the chip is in the talk only or listen only mode. The ATN, bit 4, contains the condition of the Attention Line. The ma signal is true when the chip is in:

TACS — Talker Active State
TADS — Talker Addressed State
LACS — Listener Active State
LADS — Listener Addressed State
SPAS — Serial Poll Active State

ADDRESS STATUS REGISTER

(Read Only)

ma	to	lo	ATN	TACS	LACS	LPAS	TPAS

ma – my address has occurred
to – the talk-only mode is enabled
lo – the listen-only mode is enabled
ATN – The Attention command is asserted
TACS – GPIA is in the Talker Active State
LACS – GPIA is in the Listener Active State
LPAS – GPIA is in the Listener Primary Addressed State
TPAS – GPIA is in the Talker Primary Addressed State

Address Switch Register R4R — The address switch register is external to the chip. There is an enable line ($\overline{ASE}$) to be used to enable three-state drivers connected between the address switches and the MPU. When the MPU addresses the address switch register the enable line directs the switch information to be sent to the MPU. The five least significant bits of the eight-bit register are used to specify the bus address of the device and the remaining three bits may be used at the discretion of the user. The most probable use of one or two of the bits is for controlling the listener only or talk only functions.

ADDRESS SWITCH REGISTER

(Read Only)

UD3	UD2	UD1	AD5	AD4	AD3	AD2	AD1

AD1-AD5 – Device address
UD1-UD3 – User definable bits
When this "register" is addressed, the $\overline{ASE}$ pin is set which allows external address switch information from bus device to be read.

Address Register R4W — The Address Register is an eight-bit storage register. The purpose of this register is to carry the primary address of the device. The primary address is placed in the five least significant bits of the register. If external switches are used for device addressing these are normally read from the Address Switch Register and then placed in the Address Register by the MPU.

AD1 through AD5 bits 0–5 are for the device's address. The lsbe bit 7 is set to enable the Dual Primary Addressing Mode. During this mode the device will respond to two consecutive addresses, one address with AD1 equal to 0 and the other address with AD1 equal to 1. For example, if the device's address is HEX 0F, the Dual Primary Addressing Mode would allow the device to be addressed at both HEX OF and HEX OE. The dal bit 6 is set to disable the listener and the dat bit 5 is set to disable the talker.

This register is cleared by the $\overline{Reset}$ input only (not by the reset bit of the Auxiliary Command Register bit 7).

When $\overline{ATN}$ is enabled and the primary address is received on the $\overline{IBO-7}$ lines, the MC68488 will set bit 7 of the address status register (ma). This places the MC68488 in the TPAS or LPAS.

When $\overline{ATN}$ is disabled the GPIA may go to one of three states: TACS, LACS or SPAS.

ADDRESS REGISTER

(Write Only)

lsbe	dal	dat	AD5	AD4	AD3	AD2	AD1

lsbe – enable dual primary addressing mode
dal – disable the listener
dat – disable the talker
AD1-AD5 – Primary device address, usually read from address switch register
Register is cleared by the $\overline{Reset}$ input pin only.

Auxiliary Command Register R3R/W — Bit 7, reset, initializes the chip to the following states: (reset is set true by external $\overline{Reset}$ input pin and by writing into the register from the MPU).

SIDS—Source Idle State
AIDS—Acceptor Idle State
TIDS—Talker Idle State
LIDS—Listener Idle State

LOCS—Local State
NPRS—Negative Poll Response State
PPIS—Parallel Poll Idle State
PUCS—Parallel Poll Unaddressed to Configure State
PPO—Parallel Poll No Capability

rfdr (release RFD handshake) bit 6 allows for completion of the handshake that was stopped by RFD (Ready For Data) holdoff commands hlda and hlde.

fget (force group execute trigger) bit 0 has the same effect as the GET (Group Execute Trigger) command from the controller.

rtl (return to local) bit 2 allows the device to respond to local controls and the associated device functions are operative.

dacr (release DAC handshake) bit 4 is set high to allow DAC to go passively true. This bit is set to indicate that the MPU has examined a secondary address or an undefined command.

ulpa (upper/lower primary address) bit 1 will indicate the state of the LSB of the address received on the DIO1-8 bus lines at the time the last Primary Address was received. This bit can be read but not written by the MPU.

msa (valid secondary address) bit 3 is set true (high) when TPAS (Talker Primary Addressed State) or LPAS (Listener Primary Addressed State) is true. The chip will become addressed to listen or talk. The primary address must have been previously received.

RFD, $\overline{DAV}$, DAC—(Ready For Data, Data Valid, Data Accepted) bits assume the same state as the corresponding signal on the MC68488 package pins. The MPU may only read these bits. These signals are not synchronized with the MPU clock.

dacd (data accept disable) bit 1 set high by the MPU will prevent completion of the automatic handshake on Addresses or Commands. dacr is used to complete the handshake.

feoi (forced end or identify) bit 5 tells the chip to send $\overline{EOI}$ low. The $\overline{EOI}$ line is then returned high after the next byte is transmitted. NOTE: The following signals are not stored but revert to a false (low) level one clock cycle (MPUϕ2) after they are set true (high):

1. rfdr
2. feoi
3. dacr

These signals can be written but not read by the MPU.

AUXILIARY COMMAND REGISTER

	reset	rfdr	feoi	dacr	msa	rtl	dacd	fget	Write
		DAC	$\overline{DAV}$	RFE			ulpa		Read

reset – initialize the chip to the following status:
(1) all interrupts cleared
(2) following bus states are in effect: SIDS, AIDS, TIDS, LIDS, LOCS, PPIS, PUCS, and PPØ
(3) bit is set by $\overline{Reset}$ input pin
msa – if GPIA is in LPAS or TDAS, setting msa will force GPIA to LADS or TADS
rtl – return to local if local lockout is disabled
ulpa – state of LSB of bus at last primary address receive time
fget – Force group execute trigger command from the MPU has occurred
rfdr – continue handshake stopped by RFD holdoff
feoi – set EOI true, clears after next byte transmitted
dacr – MPU has examined an undefined command or secondary address
dacd – prevents completion of automatic handshake on Addresses or Commands

Command Status Register R1R — The command status register flags commands or states as they occur. These flags or states are simply coupled onto the MPU bus. There are five major address commands. REM shows the remote/local state of the talker/listener. RLC bit 3 is set when a change of state of the remote/local flip-flop occurs and reset when the command status register is read. DCAS bit 1 indicates that either the device clear or selected device clear has been received activating the device clear function. SPAS bit 2 indicates that the SPE command has been received activating the device serial poll function. UACG bit 7 indicates that an undefined address command has been received and depending on programming the MPU decides whether to execute or ignore it. UUCG bit 0 indicates that an undefined universal command has been received.

COMMAND STATUS REGISTER
(Read)

UACG	REM	LOK	X	RLC	SPAS	DCAS	UUCG

UACG – Undefined Addressed Command
REM – Remote Enabled
LOK – Local Lockout Enabled
RLC – Remote/Local State Changed
SPAS – Serial Poll Active State is in effect
DCAS – Device Clear Active State is in effect
UUCG – Undefined Universal Command

Command Pass-Through Register R6R — The command pass through is an eight-bit port with no storage. When this port is addressed by MPU it

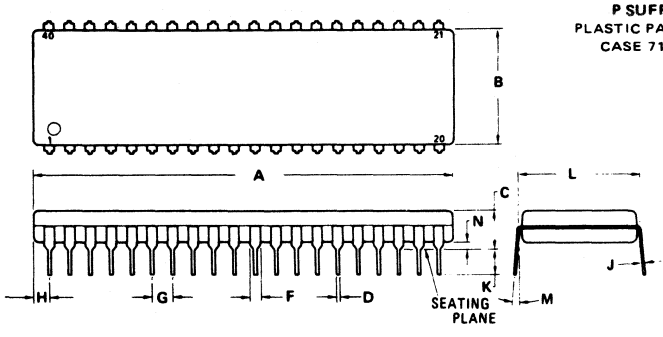
connects the instrument data bus ($\overline{IB0}$–$\overline{IB7}$) to the MPU data bus D0–D7. This port can be used to pass commands and secondary addresses that aren't automatically interpreted through to the MPU for inspection.

COMMAND PASS-THROUGH REGISTER

(Read Only)

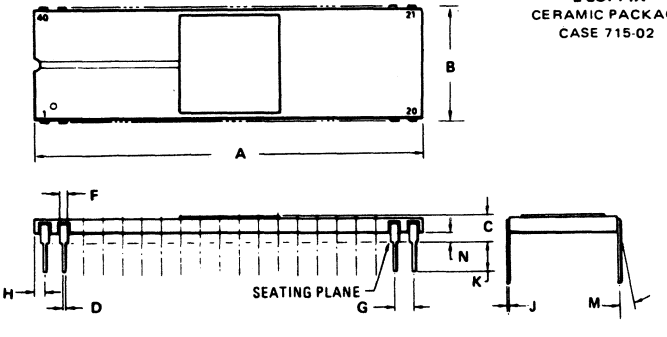
B7	B6	B5	B4	B3	B2	B1	BØ

An eight-bit input port used to pass commands and secondary addresses to MPU which are not automatically interpreted by the GPIA



P SUFFIX
PLASTIC PACKAGE
CASE 711-01

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	51.82	52.32	2.040	2.060
B	13.72	14.22	0.540	0.560
C	4.57	5.08	0.180	0.200
D	0.36	0.51	0.014	0.020
F	1.02	1.52	0.040	0.060
G	2.41	2.67	0.095	0.105
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	3.68	4.19	0.145	0.165
L	14.99	15.49	0.590	0.610
M	0°	10°	0°	10°
N	0.51	1.02	0.020	0.040



L SUFFIX
CERAMIC PACKAGE
CASE 715-02

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	50.29	51.31	1.980	2.020
B	14.86	15.62	0.585	0.615
C	2.54	4.19	0.100	0.165
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
H	0.76	1.78	0.030	0.070
J	0.20	0.33	0.008	0.013
K	2.54	4.19	0.100	0.165
M	0°	10°	0°	10°
N	0.51	1.52	0.020	0.060

NOTE:
1. LEADS, TRUE POSITIONED WITHIN 0.25 mm (0.010) DIA (AT SEATING PLANE), AT MAX. MAT'L CONDITION.

chapter 2 linears for M6800 MPU

MOTOROLA

MC3446

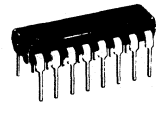
QUAD GENERAL PURPOSE INTERFACE BUS (G.P.I.B.) TRANSCEIVER

The MC3446 is a quad bus transceiver intended for usage in instruments and programmable calculators equipped for interconnection into complete measurement systems. This transceiver allows the bidirectional flow of digital data and commands between the various instruments. The transceiver provides four open-collector drivers and four receivers featuring hysteresis.

- Tailored to Meet the IEEE Standard 488-1975 (Digital Interface for Programmable Instrumentation) and the Proposed IEC Standard on Instrument Interface
- Provides Electrical Compatibility with Hewlett Packard Interface Bus. (HP-IB)
- MOS Compatible with High Impedance Inputs
- Driver Output Guaranteed Off During Power Up/Power Down
- Low Power – Average Power Supply Current = 12 mA
- Termination Resistors Provided

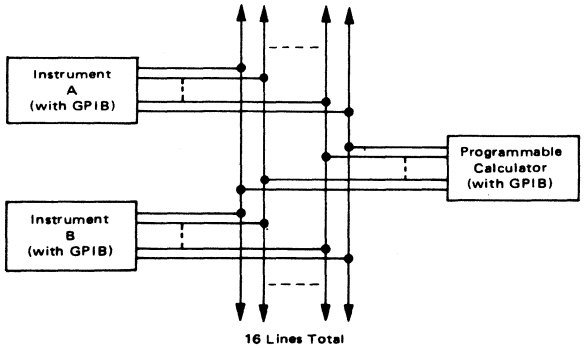
QUAD INTERFACE BUS TRANSCEIVER

SILICON MONOLITHIC INTEGRATED CIRCUIT

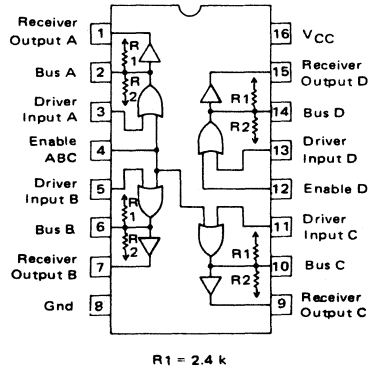


P SUFFIX
PLASTIC PACKAGE
CASE 648

TYPICAL MEASUREMENT SYSTEM APPLICATION



PIN CONNECTIONS



R1 = 2.4 k
R2 = 5.0 k

MAXIMUM RATINGS (T_A = 25°C unless otherwise noted.)

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	7.0	Vdc
Input Voltage	V_I	5.5	Vdc
Driver Output Current	$I_{O(D)}$	150	mA
Junction Temperature	T_J	150	°C
Operating Ambient Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	-65 to +150	°C

ELECTRICAL CHARACTERISTICS (Unless otherwise noted, 4.75 V ≤ V_{CC} ≤ 5.25 V and 0 ≤ T_A ≤ 70°C, typical values are at T_A = 25°C, V_{CC} = 5.0 V)

Characteristic	Symbol	Min	Typ	Max	Unit
DRIVER PORTION					
Input Voltage – High Logic State	$V_{IH(D)}$	2.0	–	–	V
Input Voltage – Low Logic State	$V_{IL(D)}$	–	–	0.8	V
Input Current – High Logic State (V_{IH} = 2.4 V)	$I_{IH(D)}$	–	5.0	20	μA
Input Current – Low Logic State (V_{IL} = 0.4 V, V_{CC} = 5.0 V, T_A = 25°C)	$I_{IL(D)}$	–	0.2	0.36	mA
Input Clamp Voltage (I_{IC} = -12 mA)	$V_{IC(D)}$	–	–	-1.5	V
Output Voltage – High Logic State (1) ($V_{IH(S)}$ = 2.4 V or $V_{IH(D)}$ = 2.0 V)	$V_{OH(D)}$	2.5	3.3	3.7	V
Output Voltage – Low Logic State ($V_{IL(S)}$ = 0.8 V, $V_{IL(D)}$ = 0.8 V, $I_{OL(D)}$ = 48 mA)	$V_{OL(D)}$	–	–	0.4	
Input Breakdown Current ($V_{I(D)}$ = 5.5 V)	$I_{IB(D)}$	–	–	1.0	mA
RECEIVER PORTION					
Input Hysteresis	–	400	900	–	mV
Input Threshold Voltage – Low to High Output Logic State	$V_{ILH(R)}$	–	1.78	2.0	V
Input Threshold Voltage – High to Low Output Logic State	$V_{IHL(R)}$	0.8	0.88	–	V
Output Voltage – High Logic State ($V_{IH(R)}$ = 2.0 V, $I_{OH(R)}$ = -400 μA)	$V_{OH(R)}$	2.4	–	–	V
Output Voltage – Low Logic State ($V_{IL(R)}$ = 0.8 V, $I_{OL(R)}$ = 8.0 mA)	$V_{OL(R)}$	–	–	0.4	V
Output Short Circuit Current ($V_{IH(R)}$ = 2.0 V) (Only one output may be shorted at a time)	$I_{OS(R)}$	4.0	–	14	mA
BUS LOAD CHARACTERISTICS					
Bus Voltage ($V_{IH(E)}$ = 2.4 V, I_{BUS} = -12 mA)	$V_{(BUS)}$	2.5 –	3.3 –	3.7 -1.5	V
Bus Current ($V_{IH(O)}$ = 2.4 V, V_{BUS} = 5.0 V) ($V_{IH(D)}$ = 2.4 V, V_{BUS} = 0.4 V) (V_{BUS} ≤ 5.5 V)	$I_{(BUS)}$	0.7 -1.3 –	– – –	– -3.2 2.5	mA
TOTAL DEVICE POWER CONSUMPTION					
Power Supply Current (All Drivers OFF) (All Drivers ON)	I_{CC}	 – –	 12 32	 19 40	mA

SWITCHING CHARACTERISTICS (V_{CC} = 5.0 V; T_A = 25°C)

Characteristic	Symbol	Min	Typ	Max	Unit
DRIVER PORTION					
Propagation Delay Time from Driver Input to Low Logic State Bus Output	$t_{PHL(D)}$	–	34	50	ns
Propagation Delay Time from Driver Input to High Logic State Bus Output	$t_{PLH(D)}$	–	17	40	ns
Propagation Delay Time from Enable Input to Low Logic State Bus Output	$t_{PHL(E)}$	–	39	50	ns
Propagation Delay Time from Enable Input to High Logic State Bus Output	$t_{PLH(E)}$	–	32	50	ns
RECEIVER PORTION					
Propagation Delay Time from Bus Input to High Logic State Receiver Output	$t_{PLH(R)}$	–	37	50	ns
Propagation Delay Time from Bus Input to Low Logic State Receiver Output	$t_{PHL(R)}$	–	22	40	ns

FIGURE 1 – TEST CIRCUIT AND WAVEFORMS FOR PROPAGATION DELAY TIME FROM RECEIVER INPUT (BUS) TO OUTPUT

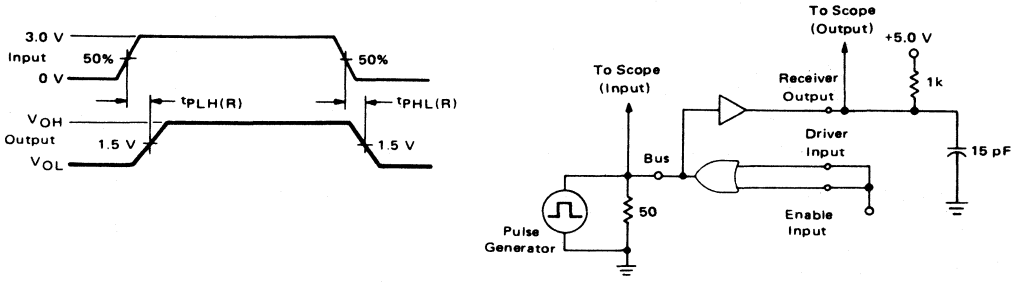


FIGURE 2 – TEST CIRCUIT AND WAVEFORMS FOR PROPAGATION DELAY TIME FROM DRIVER AND COMMON ENABLE INPUTS TO OUTPUT (BUS)

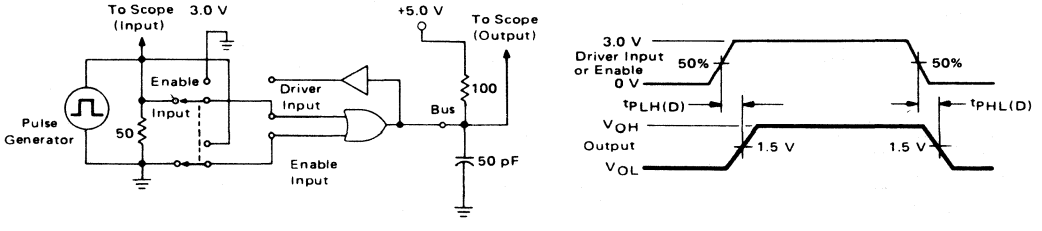


FIGURE 3 – TYPICAL RECEIVER HYSTERESIS CHARACTERISTICS

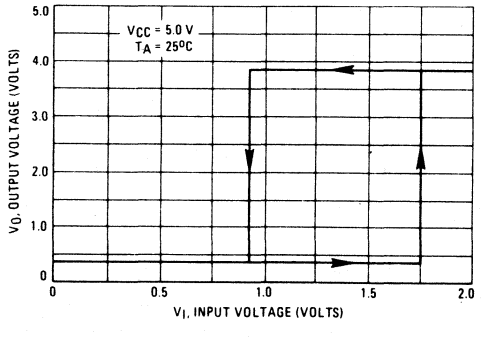
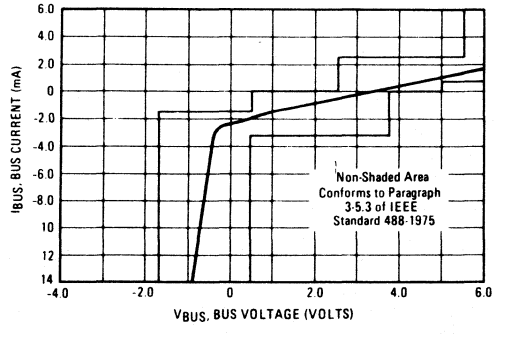
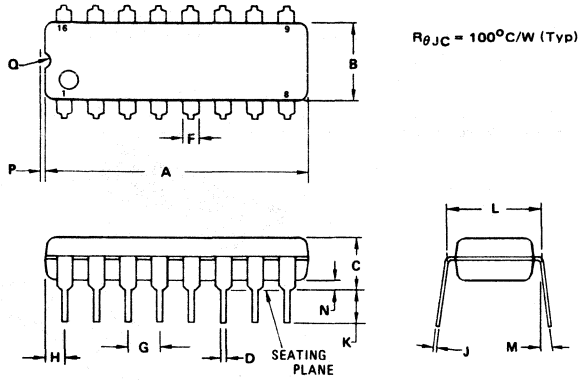


FIGURE 4 – TYPICAL BUS LOAD LINE



OUTLINE DIMENSIONS

P SUFFIX
PLASTIC PACKAGE
CASE 648



$R_{\theta JC}$ = 100°C/W (Typ)

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	20.70	21.34	0.815	0.840
B	6.10	6.60	0.240	0.260
C	4.06	4.57	0.160	0.180
D	0.38	0.51	0.015	0.020
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.32	1.83	0.052	0.072
J	0.20	0.30	0.008	0.012
K	2.92	3.43	0.115	0.135
L	7.37	7.87	0.290	0.310
M	–	10°	–	10°
N	0.51	1.02	0.020	0.040
P	0.13	0.38	0.005	0.015
Q	0.51	0.76	0.020	0.030

NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL

THERMAL INFORMATION

The maximum power consumption an integrated circuit can tolerate at a given operating ambient temperature, can be found from the equation:

$$P_{D(T_A)} = \frac{T_{J(max)} - T_A}{R_{\theta JA(Typ)}}$$

Where: $P_{D(T_A)}$ = Power Dissipation allowable at a given operating ambient temperature. This must be greater than the sum of the products of the supply voltages and supply currents at the worst case operating condition.

$T_{J(max)}$ = Maximum Operating Junction Temperature as listed in the Maximum Ratings Section

T_A = Maximum Desired Operating Ambient Temperature

$R_{\theta JA}$(Typ) = Typical Thermal Resistance Junction to Ambient

MOTOROLA

MC3447

Product Preview

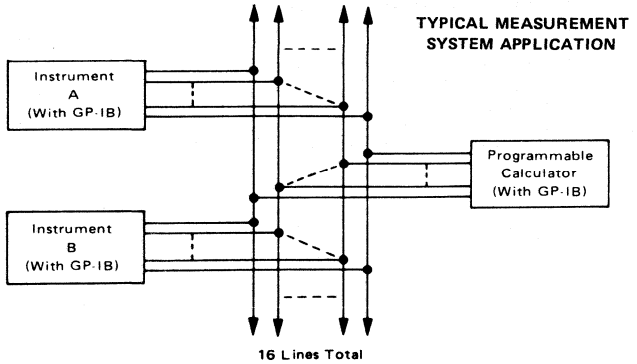
BIDIRECTIONAL INSTRUMENTATION BUS (GP IB) TRANSCEIVER

This bidirectional bus transeiver is intended as the interface between TTL or MOS logic and the IEEE Standard Instrumentation Bus (488-1975, often referred to as GP IB). The required bus termination is internally provided.
Low power consumption has been achieved by trading a minimum of speed for low current drain on non-critical channels. A fast channel is provided for critical ATN and EOI paths.
Each driver/receiver pair forms the complete interface between the bus and an instrument. Either the driver or the receiver of each channel is enabled by a Send/Receive input with the disabled output of the pair forced to a high impedance state.
The receivers have input hysteresis to improve noise margin, and their input loading follows the bus standard specifications.

- Low Power — Average Power Supply Current = 30 mA listen, 75 mA talking
- Eight Driver/Receiver Pairs
- Three-State Outputs
- High Impedance Inputs
- Receiver Hysteresis — 600 mV (Typ)
- Fast Propagation Times — 15 - 20 ns (Typ)
- TTL Compatible Receiver Outputs
- Single +5 Volt Supply
- Open Collector Driver Output with Terminations
- Power Up/Power down Protection (no Invalid Information Transmitted to Bus)
- No Bus Loading when Power is Removed from Device
- Required Termination Characteristics Provided

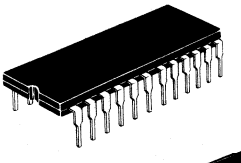
MAXIMUM RATINGS (T_A = 25 °C unless otherwise noted)

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	7.0	Vdc
Input Voltage	V_I	5.5	Vdc
Driver Output Current	$I_{O(D)}$	150	mA
Junction Temperature	T_J	150	°C
Operating Ambient Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	−65 to +150	°C

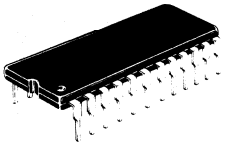


OCTAL BIDIRECTIONAL BUS TRANSCEIVER WITH TERMINATION NETWORKS

SILICON MONOLITHIC INTEGRATED CIRCUIT

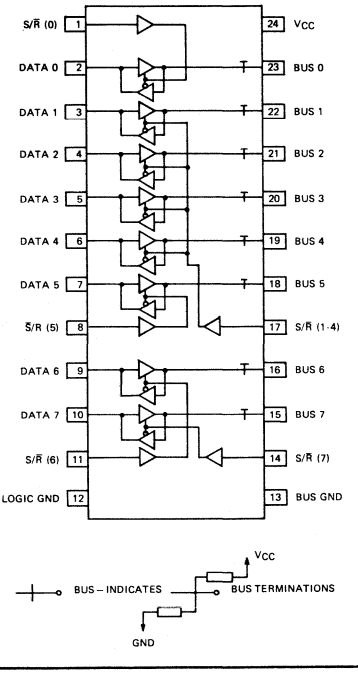


L SUFFIX
Ceramic Package
Case 623



P SUFFIX
Plastic Package
Case 649

PIN ASSIGNMENT



MOTOROLA

MC3448A

BIDIRECTIONAL INSTRUMENTATION BUS (GP-IB) TRANSCEIVER

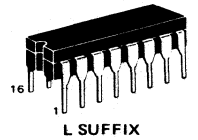
This bidirectional bus transeiver is intended as the interface between TTL or MOS logic and the IEEE Standard Instrumentation Bus (488-1975, often referred to as GP-IB). The required bus termination is internally provided.

Each driver/receiver pair forms the complete interface between the bus and an instrument. Either the driver or the receiver of each channel is enabled by its corresonding Send/Receive input with the disabled output of the pair forced to a high impedance state. An additional option allows the driver outputs to be operated in an open collector or active pull-up configuration. The receivers have input hysteresis to improve noise margin, and their input loading follows the bus standard specifications.

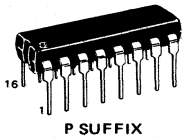
- Four Independent Driver/Receiver Pairs
- Three-State Outputs
- High Impedance Inputs
- Receiver Hysteresis – 600 mV (Typ)
- Fast Propagation Times – 15-20 ns (Typ)
- TTL Compatible Receiver Outputs
- Single +5 Volt Supply
- Open Collector Driver Output Option
- Power Up/Power Down Protection (No Invalid Information Transmitted to Bus)
- No Bus Loading When Power Is Removed From Device
- Required Termination Characteristics Provided

QUAD THREE-STATE BUS TRANSCEIVER WITH TERMINATION NETWORKS

SILICON MONOLITHIC INTEGRATED CIRCUIT



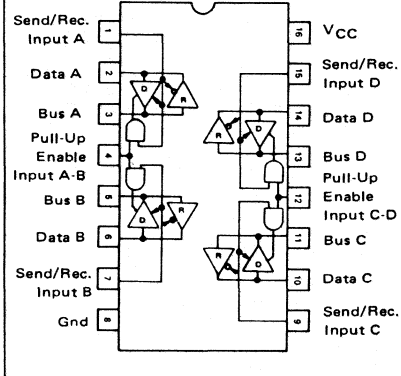
L SUFFIX
CERAMIC PACKAGE
CASE 620

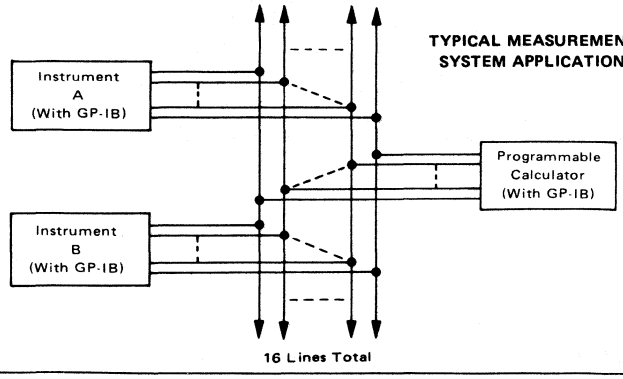


P SUFFIX
PLASTIC PACKAGE
CASE 648

MAXIMUM RATINGS (T_A = 25°C unless otherwise noted)

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	7.0	Vdc
Input Voltage	V_I	5.5	Vdc
Driver Output Current	$I_{O(D)}$	150	mA
Junction Temperature	T_J	150	°C
Operating Ambient Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	-65 to +150	°C





TRUTH TABLE

Send/Rec.	Enable	Info. Flow	Comments
0	X	Bus → Data	—
1	1	Data → Bus	Active Pull Up
1	0	Data → Bus	Open Col.

X = Don't Care

DS 9462
(Replaces NP-51)

ELECTRICAL CHARACTERISTICS

(Unless otherwise noted 4.75 V ≤ V_{CC} ≤ 5.25 V and 0 ≤ T_A ≤ 70°C; typical values are at T_A = 25°C, V_{CC} = 5.0 V)

Characteristic	Symbol	Min	Typ	Max	Unit
Bus Voltage					V
(Bus Pin Open)($V_{I(S/R)}$ = 0.8 V)	$V_{(BUS)}$	2.75	–	3.7	
($I_{(BUS)}$ = -12 mA)	$V_{IC(BUS)}$	–	–	-1.5	
Bus Current	$I_{(BUS)}$				mA
(5.0 V ≤ $V_{(BUS)}$ ≤ 5.5 V)		0.7	–	2.5	
($V_{(BUS)}$ = 0.8 V)		-1.5	–	-3.2	
(V_{CC} = 0 V, 0 V ≤ $V_{(BUS)}$ ≤ 2.75 V)		–	–	-0.04	
Receiver Input Hysteresis ($V_{I(S/R)}$ = 0.8 V)	–	400	600	–	mV
Receiver Input Threshold					V
($V_{I(S/R)}$ = 0.8 V, Low to High)	$V_{ILH(R)}$	–	1.6	1.8	
($V_{I(S/R)}$ = 0.8 V, High to Low)	$V_{IHL(R)}$	0.8	1.0	–	
Receiver Output Voltage – High Logic State ($V_{I(S/R)}$ = 0.8 V, $I_{OH(R)}$ = -800 μA, $V_{(BUS)}$ = 2.0 V)	$V_{OH(R)}$	2.7	–	–	V
Receiver Output Voltage – Low Logic State ($V_{I(S/R)}$ = 0.8 V, $I_{OL(R)}$ = 16 mA, $V_{(BUS)}$ = 0.8 V)	$V_{OL(R)}$	–	–	0.5	V
Receiver Output Short Circuit Current ($V_{I(S/R)}$ = 0.8 V, $V_{(BUS)}$ = 2.0 V)	$I_{OS(R)}$	-15	–	-75	mA
Driver Input Voltage – High Logic State ($V_{I(S/R)}$ = 2.0 V)	$V_{IH(D)}$	2.0	–	–	V
Driver Input Voltage – Low Logic State ($V_{I(S/R)}$ = 2.0 V)	$V_{IL(D)}$	–	–	0.8	V
Driver Input Current – Data Pins ($V_{I(S/R)}$ = $V_{I(E)}$ = 2.0 V)					μA
(0.5 ≤ $V_{I(D)}$ ≤ 2.7 V)	$I_{I(D)}$	-200	–	40	
($V_{I(D)}$ = 5.5 V)	$I_{IB(D)}$	–	–	200	
Input Current – Send/Receive					μA
(0.5 ≤ $V_{I(S/R)}$ ≤ 2.7 V)	$I_{I(S/R)}$	-100	–	20	
($V_{I(S/R)}$ = 5.5 V)	$I_{IB(S/R)}$	–	–	100	
Input Current – Enable					μA
(0.5 ≤ $V_{I(E)}$ ≤ 2.7 V)	$I_{I(E)}$	-200	–	20	
($V_{I(E)}$ = 5.5 V)	$I_{IB(E)}$	–	–	100	
Driver Input Clamp Voltage ($V_{I(S/R)}$ = 2.0 V, $I_{IC(D)}$ = -18 mA)	$V_{IC(D)}$	–	–	-1.5	V
Driver Output Voltage – High Logic State ($V_{I(S/R)}$ = 2.0 V, $V_{IH(D)}$ = 2.0 V, $V_{IH(E)}$ = 2.0 V, I_{OH} = -5.2 mA)	$V_{OH(D)}$	2.5	–	–	V
Driver Output Voltage – Low Logic State (Note 1) ($V_{I(S/R)}$ = 2.0 V, $I_{OL(D)}$ = 48 mA)	$V_{OL(D)}$	–	–	0.5	V
Output Short Circuit Current ($V_{I(S/R)}$ = 2.0 V, $V_{IH(D)}$ = 2.0 V, $V_{IH(E)}$ = 2.0 V)	$I_{OS(D)}$	-30	–	-120	mA
Power Supply Current					mA
(Listening Mode – All Receivers On)	I_{CCL}	–	63	85	
(Talking Mode – All Drivers On)	I_{CCH}	–	106	125	

SWITCHING CHARACTERISTICS (V_{CC} = 5.0 V, T_A = 25°C unless otherwise noted)

Propagation Delay of Driver					ns
(Output Low to High)	$t_{PLH(D)}$	–	10	–	
(Output High to Low)	$t_{PHL(D)}$	–	16	–	
Propagation Delay of Receiver					ns
(Output Low to High)	$t_{PLH(R)}$	–	22	–	
(Output High to Low)	$t_{PHL(R)}$	–	16	–	

NOTE 1. A pending modification of the IEEE 488-1975 Bus Standard changes $V_{OL(D)}$ from 0.4 to 0.5 V maximum to permit the use of Schottky technology.

SWITCHING CHARACTERISTICS (continued) (V_{CC} = 5.0 V, T_A = 25°C unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Propagation Delay Time — Send/Receiver to Data					ns
Logic High to Third State	$t_{PHZ(R)}$	–	9.0	–	
Third State to Logic High	$t_{PZH(R)}$	–	17	–	
Logic Low to Third State	$t_{PLZ(R)}$	–	10	–	
Third State to Logic Low	$t_{PZL(R)}$	–	17	–	
Propagation Delay Time — Send/Receiver to Bus					ns
Logic High to Third State	$t_{PHZ(D)}$	–	12	–	
Third State to Logic High	$t_{PZH(D)}$	–	21	–	
Logic Low to Third State	$t_{PLZ(D)}$	–	11	–	
Third State to Logic Low	$t_{PZL(D)}$	–	19	–	
Turn-On Time — Enable to Bus					ns
Pull-Up Enable to Open Collector	$t_{POFF(E)}$	–	10	–	
Open Collector to Pull-Up Enable	$t_{PON(E)}$	–	12	–	

PROPAGATION DELAY TEST CIRCUITS AND WAVEFORMS

FIGURE 1 — BUS INPUT TO DATA OUTPUT (RECEIVER)

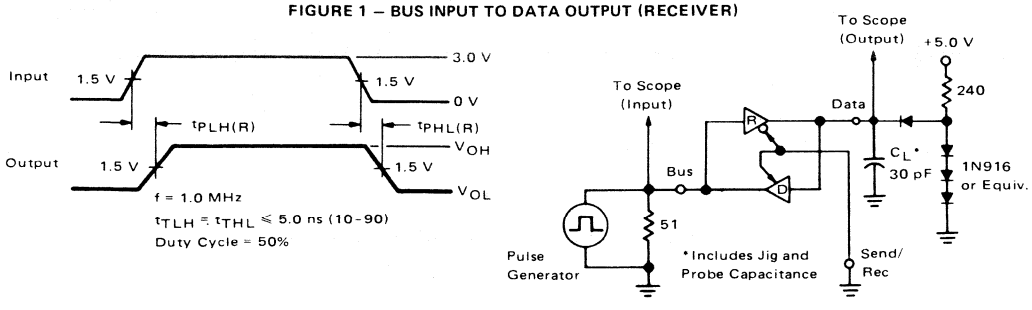


FIGURE 2 — DATA INPUT TO BUS OUTPUT (DRIVER)

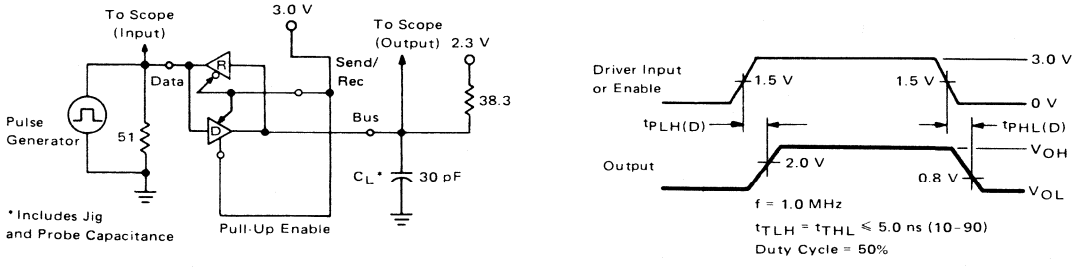


FIGURE 3 — SEND/RECEIVE INPUT TO BUS OUTPUT (DRIVER)

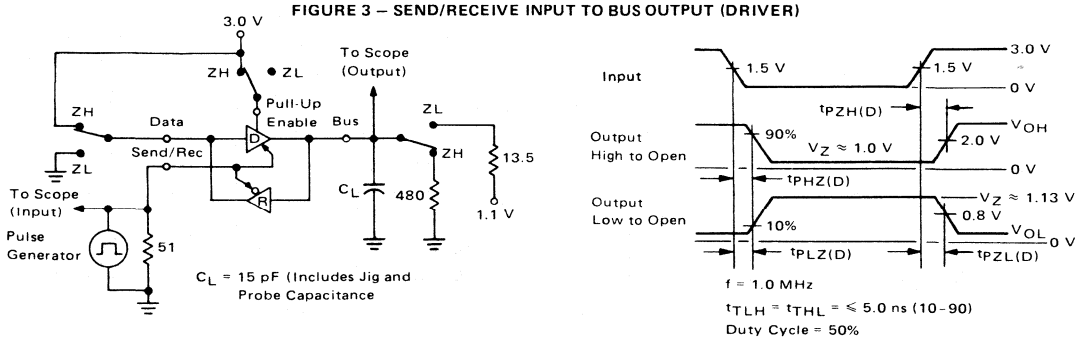
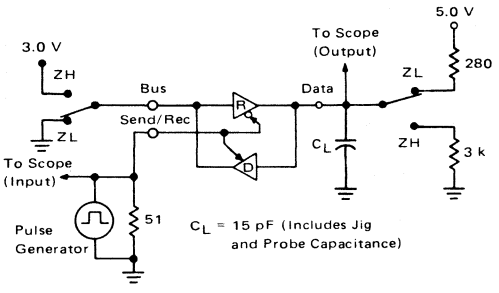


FIGURE 4 – SEND/RECEIVE INPUT TO DATA OUTPUT (RECEIVER)



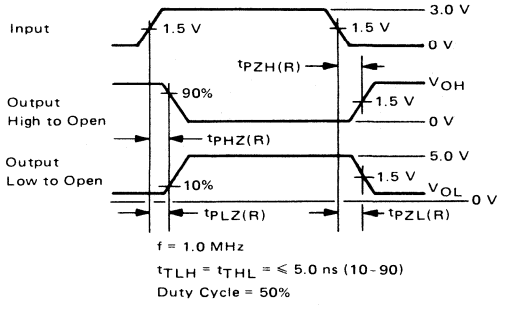
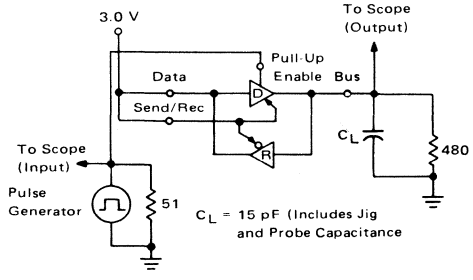


FIGURE 5 – ENABLE INPUT TO BUS OUTPUT (DRIVER)



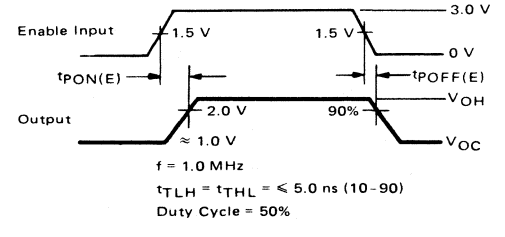


FIGURE 6 – TYPICAL RECEIVER HYSTERESIS CHARACTERISTICS

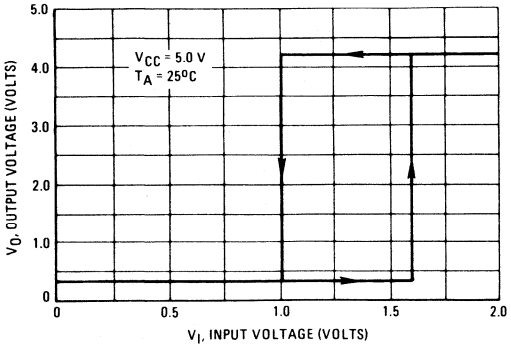


FIGURE 7 – TYPICAL BUS LOAD LINE

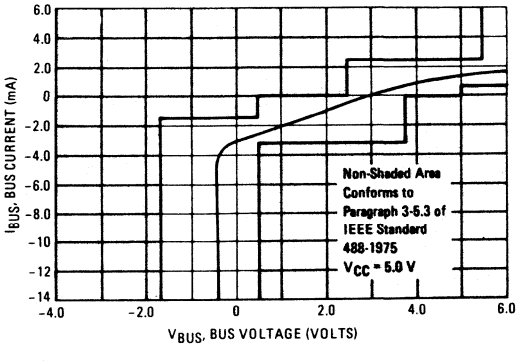
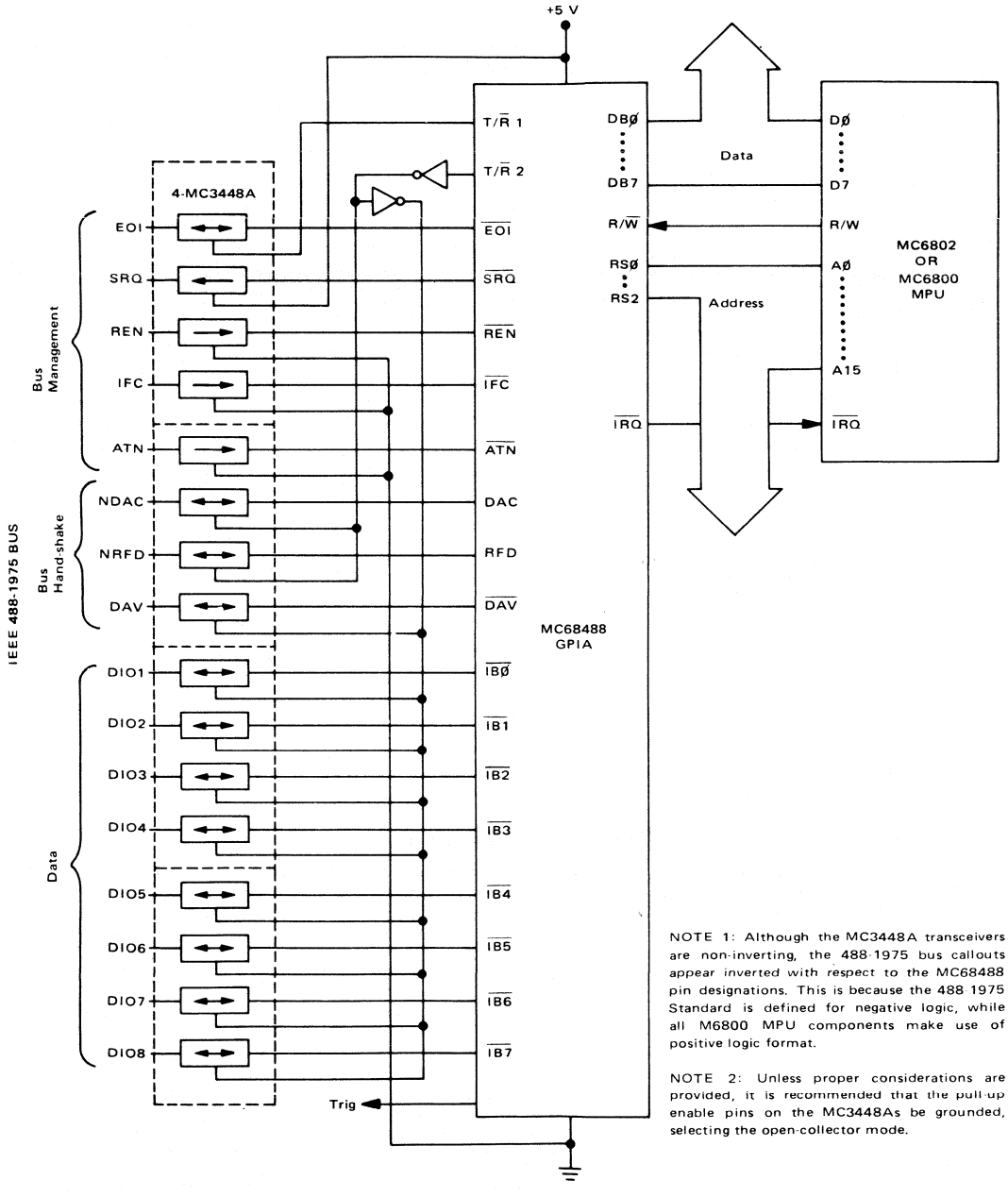


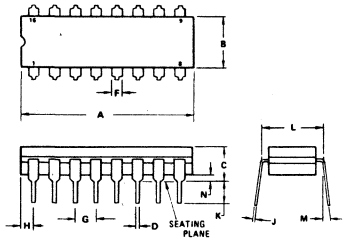
FIGURE 8 – SIMPLE SYSTEM CONFIGURATION



NOTE 1: Although the MC3448A transceivers are non-inverting, the 488-1975 bus callouts appear inverted with respect to the MC68488 pin designations. This is because the 488-1975 Standard is defined for negative logic, while all M6800 MPU components make use of positive logic format.

NOTE 2: Unless proper considerations are provided, it is recommended that the pull-up enable pins on the MC3448As be grounded, selecting the open-collector mode.

OUTLINE DIMENSIONS

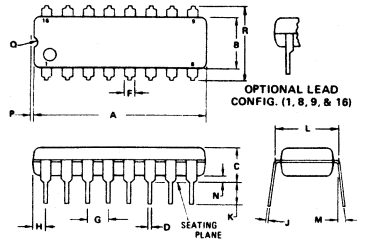


NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION
2. PKG. INDEX: NOTCH IN LEAD NOTCH IN CERAMIC OR INK DOT
3. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	19.05	19.81	0.750	0.780
B	6.22	6.98	0.245	0.275
C	4.06	5.08	0.160	0.200
D	0.38	0.51	0.015	0.020
F	1.40	1.65	0.055	0.065
G	2.54 BSC		0.100 BSC	
H	0.51	1.14	0.020	0.045
J	0.20	0.31	0.008	0.012
K	3.18	0.30	0.125	0.160
L	7.37	7.87	0.290	0.310
M	–	15°	–	15°
N	0.51	1.02	0.020	0.040

CASE 620-02

$R_{\theta JA}$ = 60°C/W (Typical)



NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSION "B" DOES NOT INCLUDE MOLD FLASH.
4. "F" DIMENSION IS FOR FULL LEADS. "HALF" LEADS ARE OPTIONAL AT LEAD POSITIONS 1, 8, 9, and 16).
5. DIMENSION "R" TO BE MEASURED AT THE TOP OF THE LEADS (NOT AT THE TIPS).

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	–	22.10	–	0.870
B	6.10	6.60	0.240	0.260
C	–	5.08	–	0.200
D	0.38	0.53	0.015	0.021
F	–	1.78	–	0.070
G	2.54 BSC		0.100 BSC	
H	0.38	2.41	0.015	0.095
J	0.20	0.38	0.008	0.015
K	2.92	–	0.115	–
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	–	0.020	–
R	–	8.26	–	0.325

CASE 648-04

$R_{\theta JA}$ = 100°C/W (Typical)

THERMAL INFORMATION

The maximum power consumption an integrated circuit can tolerate at a given operating ambient temperature, can be found from the equation:

$$P_{D(T_A)} = \frac{T_{J(max)} - T_A}{R_{\theta JA(Typ)}}$$

Where: $P_{D(T_A)}$ = Power Dissipation allowable at a given operating ambient temperature. This must be greater than the sum of the products of the supply voltages and supply currents at the worst case operating condition.

$T_{J(max)}$ = Maximum Operating Junction Temperature as listed in the Maximum Ratings Section

T_A = Maximum Desired Operating Ambient Temperature

$R_{\theta JA(Typ)}$ = Typical Thermal Resistance Junction to Ambient

MOTOROLA Semiconductor Products Inc.

Printed in Switzerland 60-C78/3.5

chapter 3

other NMOS MPUs

MOTOROLA

MC3870

0° to 70°C

Advance Information

MOS

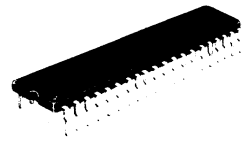
(N-CHANNEL, SILICON-GATE DEPLETION LOAD)

SINGLE-CHIP MICROCONTROLLER

SINGLE-CHIP MICROCONTROLLER

The MC3870 is a complete 8-bit microcomputer on a single MOS integrated circuit. Utilizing ion-implanted, N-channel silicon-gate technology and advanced circuit design techniques, the single-chip 3870 offers maximum cost effectiveness in a wide range of control and logic replacement applications.

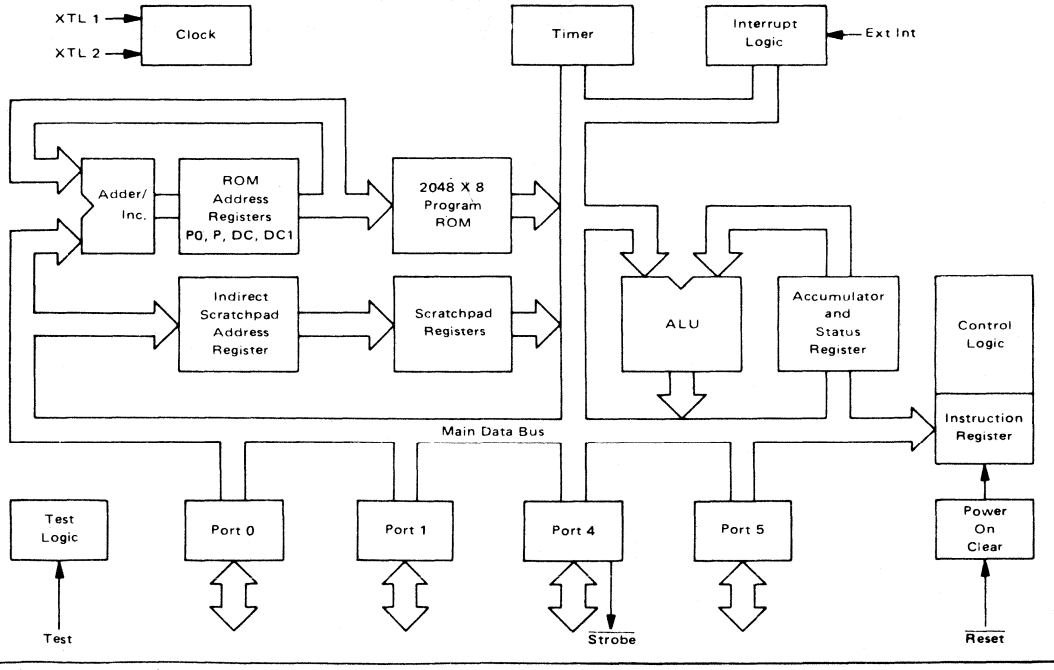
- Software Compatible with F8 Family
- 2048 Byte Mask Programmable ROM
- 64 Byte Scratchpad RAM
- 32 Bits (4 Ports) TTL Compatible I/O
- Programmable Binary Timer
  - Interval Timer Mode
  - Pulse Width Measurement Mode
  - Event Counter Mode
- External Interrupt
- Crystal, LC, RC, External, or Internal Time Base
- Low Power (275 mW Typ.)
- Single +5 Volt ±10% Power Supply



P SUFFIX
PLASTIC PACKAGE
CASE 711

L SUFFIX
CERAMIC PACKAGE
CASE 715
(Available, not shown)

FIGURE 1 - Block Diagram



ELECTRICAL SPECIFICATIONS

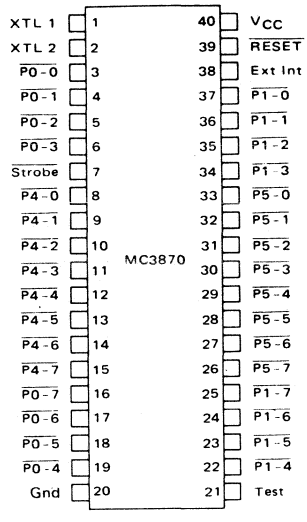
ABSOLUTE MAXIMUM RATINGS*

Temperature under Bias . . . . . 0°C to 70°C
Storage Temperature . . . . . -65°C to +150°C
Voltage on Any Pin with Respect to Ground . . . . . -1.0 V to + 7 V
Power Dissipation . . . . . 1.0 W

DC CHARACTERISTICS (T_A = 0°C to 70°C, V_{CC} = 5 V ± 10%)

Symbol	Parameter	Min	Max	Unit	Test Conditions
I_{CC}	Power Supply Current	–	TBD	mA	Outputs Open
P_D	Power Dissipation	–	TBD	mW	Outputs Open
V_{IHEX}	External Clock Input High Level	2.4	5.8	V	
V_{ILEX}	External Clock Input Low Level	-0.3	0.6	V	
I_{IHEX}	External Clock Input High Current	–	100	μA	V_{IHEX} = 2.4 V
I_{ILEX}	External Clock Input Low Current	–	-100	μA	V_{ILEX} = 0.6 V
V_{IH}	Input High Level	2.0	5.8	V	
V_{IL}	Input Low Level	-0.3	0.8	V	
I_{IH}	Input High Current (except open drain and direct drive I/O ports)	–	100	μA	V_{IH} = 2.4 V Internal Pull-up
I_{IL}	Input Low Current (except open drain and direct drive ports)	–	-1.6	mA	V_{IL} = 0.4 V
I_{LOD}	Leakage Current (open drain ports)	–	10	μA	Pulldown Device Off
I_{OH}	Output High Current (except open drain and direct drive ports)	-100	–	μA	V_{OH} = 2.4 V
I_{OHDD}	Output Drive Current (direct drive ports)	-1.5	-8	mA	V_{OH} = 0.7 V to 1.5 V
I_{OL}	Output Low Current	1.8	–	mA	V_{OL} = 0.4 V
I_{OHS}	Output High Current ($\overline{\text{STROBE}}$ Output)	-300	–	μA	V_{OH} = 2.4 V
I_{OLS}	Output Low Current ($\overline{\text{STROBE}}$ Output)	5.0	–	mA	V_{OL} = 0.4 V

PIN CONNECTIONS



Pin Name	Description	Type
P0 0 P0 7	I/O Port 0	Bidirectional
P1 0 P1 7	I/O Port 1	Bidirectional
P4 0 P4 7	I/O Port 4	Bidirectional
P5 0 P5 7	I/O Port 5	Bidirectional
Strobe	Ready Strobe	Output
Ext Int	External Interrupt	Input
Reset	External Reset	Input
Test	Test Line	Input
XTL 1, XTL 2	Time Base	Input
V_{CC}, Gnd	Power Supply Lines	Input

*Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

AC CHARACTERISTICS (T_A = 0°C to 70°C, V_{CC} = +5 V ± 10%)

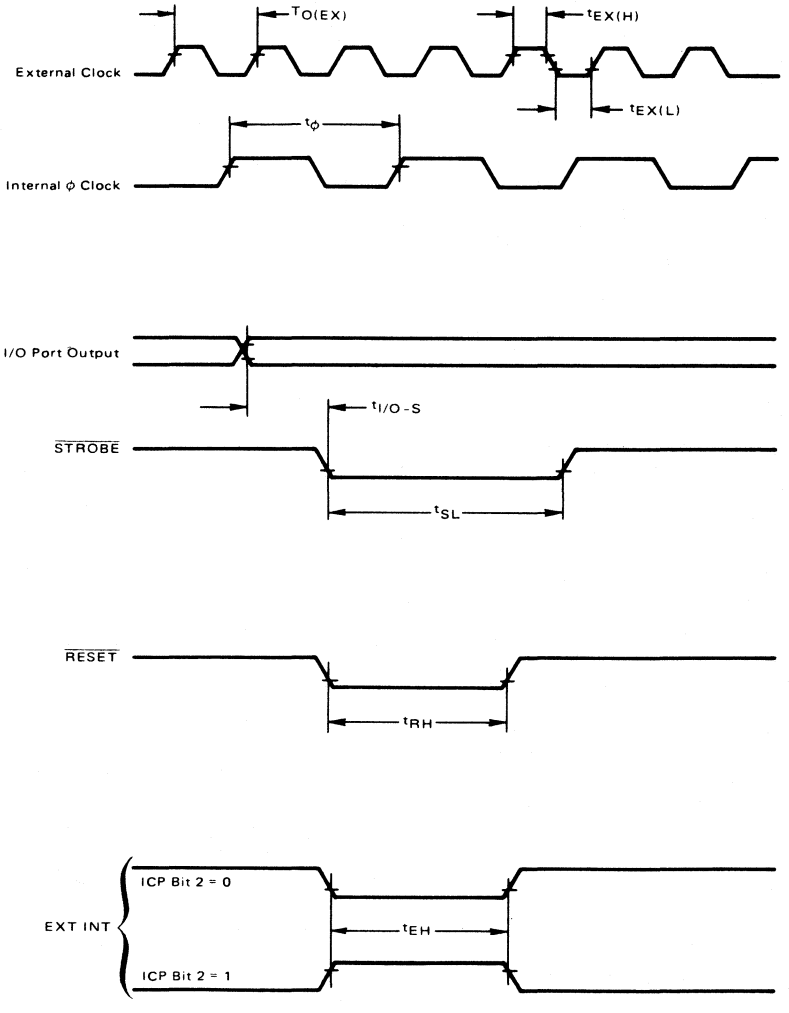
Signal	Symbol	Parameter	Min	Max	Unit	Comments
XTL 1 XTL 2	$t_{0(XTL)}$	Time Base Period, Crystal Mode	250	1000	ns	4MHz-1 MHz
	$t_{0(LC)}$	Time Base Period, LC Mode	250	1000	ns	4MHz-1 MHz
	$t_{0(RC)}$	Time Base Period, RC Mode	250	2000	ns	4 MHz-500 KHz
	$t_{0(INT)}$	Time Base Period, Internal Mode	250	590	ns	4 MHz-1.7 MHz
	$t_{0(EX)}$	Time Base Period, External Mode	250	2500	ns	4 MHz-400 kHz
	$t_{EX(H)}$	External Clock Pulse Width, High	90	2000	ns	
	$t_{EX(L)}$	External Clock Pulse Width, Low	90	2000	ns	
ϕ	$t\phi$	Internal ϕ Clock Period	$2t_0$	Typ.	ns	0.5 μs @ 4 MHz External Time Base
$\overline{\text{STROBE}}$	$t_{I/O-S}$	Port Output to $\overline{\text{STROBE}}$ Delay	$3t\phi$ -1000 Min	$3t\phi$ +250 Max.	ns	Note 1
	t_{SL}	$\overline{\text{STROBE}}$ Pulse Width, Low	$8t\phi$ -250 Min.	$12t\phi$ +250 Max.	ns	
$\overline{\text{RESET}}$	t_{RH}	$\overline{\text{RESET}}$ Hold Time, Low	$6t\phi$ +750 Min.		ns	
EXT INT	t_{EH}	EXT INT Hold Time, Active State	$6t\phi$ +750 Min.		ns	Note 2

NOTES: 1. Load is 50 pF plus 1 standard TTL input.
2. Specification is applicable when the timer is in the Interval Timer Mode. See "Timer Characteristics" for EXT INT requirements when in the Pulse Width Measurement Mode or the Event Counter Mode.

CAPACITANCE (T_A = 25°C, f = 2 MHz)

Symbol	Parameter	Min	Max	Unit	Test Condition
C_{IN}	Input Capacitance: I/O Ports, $\overline{RESET}$, EXT INT	–	7.0	pF	Unmeasured pins returned to GND
C_{XTL}	Input Capacitance: XTL 1, XTL 2	18	23	pF	

AC TIMING DIAGRAMS



NOTE: All measurements are referenced to V_{IL} max., V_{IH} min., V_{OL} max., or V_{OH} min.

FUNCTIONAL PIN DESCRIPTION

$\overline{P0\text{-}0}$–$\overline{P0\text{-}7}$, $\overline{P1\text{-}0}$–$\overline{P1\text{-}7}$, $\overline{P4\text{-}0}$–$\overline{P4\text{-}7}$, and $\overline{P5\text{-}0}$–$\overline{P5\text{-}7}$ are 32 lines which can be individually used as either TTL compatible inputs or as latched outputs.

$\overline{STROBE}$ is a ready strobe associated with I/O Port 4. This pin which is normally high provides a single low pulse after valid data is present on the $\overline{P4\text{-}0}$–$\overline{P4\text{-}7}$ pins during an output instruction.

$\overline{RESET}$ may be used to externally reset the 3870. When pulled low the 3870 will reset. When then allowed to go high the 3870 will begin program execution at program location H '000'.

EXT INT is the external interrupt input. Its active state is software programmable. This input is also used in conjunction with the timer for pulse width measurement and event counting.

XTL 1 and XTL 2 are the time base inputs to which a crystal (1 to 4 MHz), LC network, RC network, or an external single-phase clock may be connected. If timing is not critical, the 3870 will operate from its internal oscillator with no external components.

TEST is an input, used only in testing the 3870. For normal circuit functionality this pin is left unconnected or may be grounded.

V_{CC} is the power supply input (+5 V ±10%).

3870 ARCHITECTURE

This section describes the basic functional elements of the 3870 as shown in the block diagram of Figure 1.

Main Control Logic

The instruction Register (IR) receives the operation code (OP code) of the instruction to be executed from the program ROM via the data bus. During all OP code fetches eight bits are latched into the IR. Some instructions are completely specified by the upper 4 bits of the OP code. In those instructions the lower 4 bits are an immediate register address or an immediate 4-bit operand. Once latched into the IR the main control logic decodes the instruction and provides the necessary control gating signals to all circuit elements.

ROM Address Registers

There are four 11-bit registers associated with the 2K x 8 ROM. These are the Program Counter (PO), the Stack Register (P), the Data Counter (DC) and the Auxiliary Data Counter (DC1). The Program Counter is used to address instructions or immediate operands. P is used to save the contents of PO during an interrupt or subroutine call. Thus P contains the return address at which processing is to resume upon completion of the subroutine or the interrupt routine.

The Data Counter (DC) is used to address data tables. This register is auto-incrementing. Of the two data counters only DC can access the ROM. However, the XDC instruction allows DC and DC1 to be exchanged.

Associated with the address registers is an 11-bit Adder/Incrementer. This logic element is used to increment PO or DC when required and is also used to add displacements to PO on relative branches or to add the data bus contents to DC in the ADC (add data counter) instruction.

2048 X 8 ROM

The microcomputer program and data constants are stored in the program ROM. When a ROM access is required, the appropriate address register (PO or DC) is gated onto the ROM address bus and the ROM output is gated onto the main data bus. The first byte in the ROM is location zero.

Scratchpad and ISAR

The scratchpad provides 64 8-bit registers which may be used as general-purpose RAM memory. The Indirect Scratchpad Address Register (ISAR) is a 6-bit register used to address the 64 registers. All 64 registers may be accessed using ISAR. In addition the lower order 12 registers may also be directly addressed.

ISAR can be visualized as holding two octal digits. This division of ISAR is important, since a number of instructions increment or decrement only the least significant 3 bits of ISAR when referencing scratchpad bytes via ISAR. This makes it easy to reference a buffer consisting of contiguous scratchpad bytes. For example, when the low-order octal digit is incremented or decremented ISAR is incremented from octal 27 (0 '27') to 0 '20' or is decremented from 0 '20' to 0 '27'. This feature of the ISAR is very useful in many program sequences. All six bits of ISAR may be loaded at one time or either half may be loaded independently.

Scratchpad registers 9 through 15 (decimal) are given mnemonic names (J, H, K, and Q) because of special linkages between these registers and other registers such as the Stack Register. These special linkages facilitate the implementatation of multilevel interrupts and subroutine nesting. For example, the instruction LR K,P stores the lower eight bits of the Stack Register into register 13 (K lower or KL) and stores the upper three bits of P into register 12 (K upper or KU).

Arithmetic and Logic Unit (ALU)

After receiving commands from the main control logic, the ALU performs the required arithmetic or logic operations (using the data presented on the two input busses) and provides the result on the result bus. The arithmetic operations that can be performed in the ALU are binary add, decimal adjust, add with carry, decrement, and increment. The logic operations that can be performed are AND, OR, EXCLUSIVE OR, 1's complement, shift

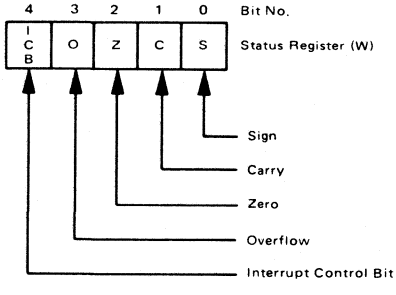
right, and shift left. Besides providing the result on the result bus, the ALU also provides four signals representing the status of the result. These signals, stored in the Status Register (W), represent CARRY, OVERFLOW, SIGN, and ZERO condition of the result of the operation.

Accumulator

The Accumulator (ACC) is the principal register for data manipulation with the 3870. The ACC serves as one input to the ALU for arithmetic or logical operations. The result of ALU operations are stored in the ACC.

The Status Register

The Status Register (also called the W register) holds five status flags as follows.



SUMMARY OF STATUS BITS

$$\text{Overflow} = \text{Carry}_7 \oplus \text{Carry}_6$$

$$\text{Zero} = \overline{ALU_7} \wedge \overline{ALU_6} \wedge \overline{ALU_5} \wedge \overline{ALU_4} \wedge \overline{ALU_3} \wedge \overline{ALU_2} \wedge \overline{ALU_1} \wedge \overline{ALU_0}$$

$$\text{Carry} = \text{Carry}_7$$

$$\text{Sign} = \overline{ALU_7}$$

Interrupt Control Bit

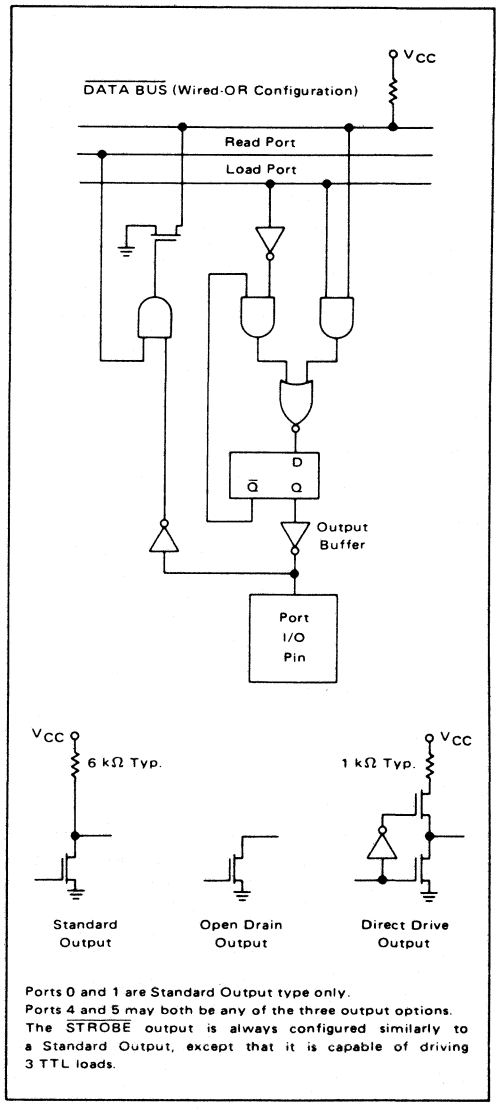
The ICB may be used to allow or disallow interrupts in the 3870. This bit is not the same as the two interrupt enable bits in the Interrupt Control Port (ICP). If the ICB is set and the 3870 interrupt logic communicates an interrupt request to the CPU section, the interrupt will be acknowledged and processed upon completion of the first nonprivileged instruction. If the ICB is cleared, an interrupt request will not be acknowledged or processed until the ICB is set.

I/O Ports

The 3870 provides four complete bidirectional Input/Output ports.

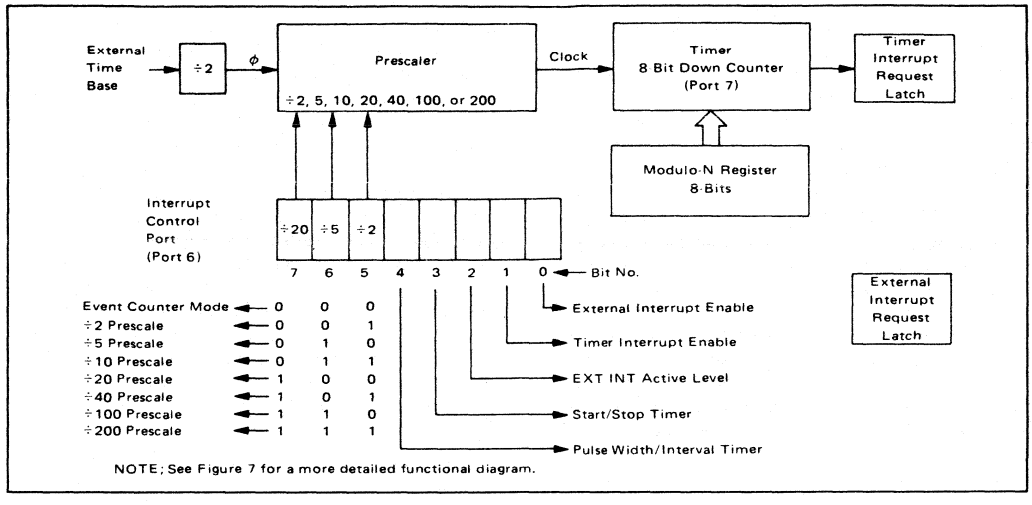
These are ports 0, 1, 4, and 5. In addition, the Interrupt Control Port is addressed as port 6 and the binary timer is addressed as port 7. An output instruction (OUT or OUTS) causes the contents of the ACC to be latched into the addressed port. An input instruction (IN or INS) transfers the contents of the port to the ACC (port 6 is an exception which is described later). The I/O pins on the 3870 are logically inverted. The schematic of an I/O pin and available output drive options are shown in Figure 2.

FIGURE 2 – I/O PIN CONCEPTUAL DIAGRAM WITH OUTPUT BUFFER OPTIONS



Ports 0 and 1 are Standard Output type only.
Ports 4 and 5 may both be any of the three output options.
The $\overline{\text{STROBE}}$ output is always configured similarly to a Standard Output, except that it is capable of driving 3 TTL loads.

FIGURE 3 – TIMER AND INTERRUPT CONTROL PORT BLOCK DIAGRAM



An output ready strobe is associated with port 4. This flag may be used to signal a peripheral device that the 3870 has just completed an output of new data to port 4. The strobe provides a single low pulse shortly after the output operation is completely finished, so either edge may be used to signal the peripheral. $\overline{\text{STROBE}}$ may also be used as an input strobe simply by doing a dummy output of H '00' to port 4 after completing the input operation.

Timer and Interrupt Control Port

The Timer is an 8-bit binary down counter which is software programmable to operate in one of three modes: the Interval Timer Mode, the Pulse Width Measurement Mode, or the Event Counter Mode. As shown in Figure 3, associated with the Timer are an 8-bit register called the Interrupt Control Port, a programmable prescaler, and an 8-bit modulo-N register.

The desired timer mode, prescale value, starting and stopping the timer, active level of the EXT INT pin, and local enabling or disabling of interrupts are selected by outputting the proper bit configuration from the Accumulator to the Interrupt Control Port (port 6) with an OUT or OUTS instruction. Bits within the Interrupt Control Port are defined as follows.

Interrupt Control Port (Port 6)

Bit 0 – External Interrupt Enable
Bit 1 – Timer Interrupt Enable
Bit 2 – EXT INT Active Level
Bit 3 – Start/Stop Timer
Bit 4 – Pulse Width/Interval Timer
Bit 5 – ÷2 Prescale
Bit 6 – ÷5 Prescale
Bit 7 – ÷20 Prescale

A special situation exists when reading the Interrupt Control Port (with an IN or INS instruction). The Accumulator is *not* loaded with the content of the ICP. Instead, Accumulator bits 0 through 6 are loaded with 0s, while bit 7 is loaded with the logic level being applied to the EXT INT pin, thus allowing the status of EXT INT to be determined without the necessity of servicing an external interrupt request. This capability is useful in establishing a high-speed polled handshake procedure or for using EXT INT as an extra input pin, if external interrupts are not required and the Timer is used only in the Interval Timer Mode. However, if it is desirable to read the content of the ICP, then one of the 64 scratch-pad registers may be used to save a copy of whatever is written to the ICP.

The rate at which the timer is clocked in the Interval Timer Mode is determined by the frequency of an internal ϕ clock and by the division value selected for the prescaler. (The internal ϕ clock operates at one-half the external time base frequency.) If ICP bit 5 is set and bits 6 and 7 are cleared, the prescaler divides ϕ by 2. Likewise, if bit 6 or 7 is individually set, the prescaler divides ϕ by 5 or 20, respectively. Combinations of bits 5, 6, and 7 may also be selected. For example, if bits 5 and 7 are set while

6 is cleared, the prescaler will divide by 40. Thus, possible prescaler values are: ÷2, ÷5, ÷10, ÷20, ÷40, ÷100, and ÷200.

Any of three conditions will cause the prescaler to be reset: Whenever the timer is stopped by clearing ICP bit 3, execution of an output instruction to port 7 (the timer is assigned port address 7), or on the trailing edge transition of the EXT INT pin when in the Pulse Width Measurement Mode. These last two conditions are explained in more detail below.

An OUT or OUTS instruction to port 7 will load the content of the Accumulator to both the Timer and the 8-bit modulo-N register, reset the prescaler, and clear any previously stored time interrupt request. As previously noted, the Timer is an 8-bit down counter which is clocked by the prescaler in the Interval Timer Mode and in the Pulse Width Measurement Mode. The prescaler is not used in the Event Counter Mode. The modulo-N register is a buffer whose function is to save the value which was most recently outputted to port 7. The modulo-N register is used in all three timer modes.

Interval Timer Mode

When ICP bit 4 is cleared (logic 0) and at least one prescale bit is set, the Timer operates in the Interval Timer Mode. When bit 3 of the ICP is set, the Timer will start counting down from the modulo-N value. After counting down to H '01', the Timer returns to the modulo-N value at the next count. On the transition from H '01' to H 'N' the Timer sets a timer interrupt request latch. Note that the interrupt request latch is set by the transition to H 'N' and not by the presence of H 'N' in the Timer, thus allowing a full 256 counts, if the modulo-N register is preset to H '00'. If bit 1 of the ICP is set, the interrupt request is passed on to the CPU section of the 3870. However, if bit 1 of the ICP is a logic 0, the interrupt request is not passed on to the CPU section, but the interrupt request latch remains set. If ICP bit 1 is subsequently set, the interrupt request will then be passed on to the CPU section. (Recall from the discussion of the Status Register's Interrupt Control Bit that the interrupt request will be acknowledged by the CPU section only if ICB is set.) Only two events can reset the timer interrupt request latch: When the timer interrupt request is acknowledged by the CPU section, or when a new load of the modulo-N register is performed.

Consider an example in which the modulo-N register is loaded with H '64' (decimal 100). The timer interrupt request latch will be set at the 100th count following the timer start and the timer interrupt request latch will

TIMER CHARACTERISTICS

Definitions:

Error = Indicated Time Value − Actual Time Value
tpsc = tφ x Prescale Value

Interval Timer Mode:

Single interval error, free running (Note 3)	±6tφ
Cumulative interval error, free running (Note 3)	0
Error between two Timer reads (Note 2)	± (tpsc + tφ)
Start Timer to stop Timer error (Notes 1, 4)	+tφ to −(tpsc + tφ)
Start Timer to read Timer error (Notes 1, 2)	−5tφ to −(tpsc + 7tφ)
Start Timer to interrupt request error (Notes 1, 3)	−2tφ to −8tφ
Load Timer to stop Timer error (Note 1)	+tφ to −(tpsc + 2tφ)
Load Timer to read Timer error (Notes 1, 2)	−5tφ to −(tpsc + 8tφ)
Load Timer to interrupt request error (Notes 1, 3)	−2tφ to −9tφ

Pulse Width Measurement Mode:

Measurement accuracy (Note 4)	+tφ to −(tpsc + 2tφ)
Minimum pulse width of EXT INT pin	2tφ

Event Counter Mode:

Minimum active time of EXT INT pin	2tφ
Minimum inactive time of EXT INT pin	2tφ

Notes:

1. All times which entail loading, starting, or stopping the Timer are referenced from the end of the last machine cycle of the OUT or OUTS instruction.
2. All times which entail reading the Timer are referenced from the end of the last machine cycle of the IN or INS instruction.
3. All times which entail the generation of an interrupt request are referenced from the start of the machine cycle in which the appropriate interrupt request latch is set. Additional time may elapse, if the interrupt request occurs during a privileged or multicycle instruction.
4. Error may be cumulative, if operation is repetitively performed.

repeatedly be set on precise 100-count intervals. If the prescaler is set at ÷40, the timer interrupt request latch will be set every 4000 ϕ clock periods. For a 2 MHz ϕ clock (4 MHz time base frequency) this will produce 2 ms intervals.

The range of possible intervals is from 2 to 51,200 ϕ clock periods (1 μs to 25.6 ms for a 2 MHz ϕ clock). However, approximately 50 ϕ periods is a practical minimum because the time between setting the interrupt request latch and the execution of the first instruction of the interrupt service routine is at least 29 ϕ periods. (The response time is dependent upon how many privileged instructions are encountered when the request occurs.) To establish time intervals greater than 51,200 ϕ clock periods is a simple matter of using the timer interrupt service routine to count the number of interrupts, saving the result in one or more of the scratchpad registers until the desired interval is achieved. With this technique virtually any time interval, or several time intervals may be generated.

The Timer may be read at any time and in any mode using an input instruction (IN 7 or INS 7) and may take place "on the fly" without interfering with normal timer operation. Also, the Timer may be stopped at any time by clearing bit 3 of the ICP. The Timer will hold its current contents indefinitely and will resume counting when bit 3 is again set. Recall, however, that the prescaler is reset whenever the Timer is stopped; thus, a series of starting and stoppings will result in a cumulative truncation error.

A summary of other timer errors is given in the timing section of this specification. For a free-running timer in the Interval Timer Mode the time interval between any two interrupt requests may be in error by ±6 ϕ clock periods, although the cumulative error over many intervals is zero. The prescaler and Timer generate precise intervals for setting the timer interrupt request latch, but the time out may occur at any time within a machine cycle. (There are two types of machine cycles: short cycles which consist of 4 ϕ clock periods and long cycles which consist of 6 ϕ clock periods. In the multichip F8 family, there is a signal called the WRITE clock which corresponds to a machine cycle.) Interrupt requests are synchronized with the internal WRITE clock, thus giving rise to the possible ±6 ϕ error. Additional errors may arise due to the interrupt request occurring while a privileged instruction or multicycle instruction is being executed. Nevertheless, for most applications, all of the above errors are negligible, especially if the desired time interval is greater than 1 ms.

Pulse Width Measurement Mode

When ICP bit 4 is set (logic 1) and at least one prescale bit is set, the Timer operates in the Pulse Width Measurement Mode. This mode is used for accurately measuring the duration of a pulse applied to the EXT INT pin. The Timer is stopped and the prescaler is reset whenever EXT INT is at its inactive level. The active level of EXT INT is defined by ICP bit 2: if cleared, EXT INT is active low; if set, EXT INT is active high. If ICP bit 3 is set, the prescaler and Timer will start counting when EXT INT transitions to the active level. When EXT INT returns to the inactive level, the Timer then stops, the prescaler resets, and—*if ICP bit 0 is set*—an external interrupt request latch is set. (Unlike timer interrupts, external interrupts are not latched, if the ICP Interrupt Enable bit is not set.)

As in the Interval Timer Mode, the Timer may be read at any time; may be stopped at any time by clearing ICP bit 3, the prescaler, and ICP bit 1 function as previously described; and, the Timer still functions as an 8-bit binary down counter with the timer interrupt request latch being set on the Timer's transition from H '01' to H 'N'. Note that the EXT INT pin has nothing to do with loading the Timer; its action is that of automatically starting and stopping the Timer and of generating external interrupts. Pulse widths longer than the prescale value times the modulo-N value are easily measured by using the timer interrupt service routine to store the number of timer interrupts in one or more scratchpad registers.

As for accuracy, the actual pulse duration is typically slightly longer than the measured value because the status of the prescaler is not readable and is reset when the Timer is stopped. Thus, for maximum accuracy, it is advisable to use a small division setting for the prescaler.

Event Counter Mode

When ICP bit 4 is cleared and all prescale bits (ICP bits 5, 6, and 7) are cleared, the Timer operates in the Event Counter Mode. This mode is used for counting pulses applied to the EXT INT pin. If ICP bit 3 is set, the Timer will decrement on each transition from the inactive level to the active level of the EXT INT pin. The prescaler is not used in this mode. But, as in the other two timer modes, the Timer may be read at any time; may be stopped at any time by clearing ICP bit 3, ICP bit 1 functions as previously described; and, the timer interrupt request latch is set on the Timer's transition from H '01' to H 'N'.

Normally, ICP bit 0 should be kept cleared in the Event Counter Mode; otherwise, external interrupts will be generated on the transition from the inactive level to the active level of the EXT INT pin.

For the Event Counter Mode, the minimum pulse width required on EXT INT is 2 ϕ clock periods and the minimum inactive time is 2 ϕ clock periods; therefore, the maximum repetition rate is 500 KHz.

External Interrupts

When the timer is in the Interval Timer Mode, the

EXT INT pin is available for nontimer-related interrupts. If ICP bit 0 is set, an external interrupt request latch is set when there is a transition from the inactive level to the active level of EXT INT. (EXT INT is an edge-triggered input.) The interrupt request is latched either until acknowledged by the CPU section or until ICP bit 0 is cleared (unlike timer interrupt requests which remain latched even when ICP bit 1 is cleared). External interrupts are handled in the same fashion when the Timer is in the Pulse Width Measurement Mode or in the Event Counter Mode—except that only in the Pulse Width Measurement Mode the external interrupt request latch is set on the trailing edge of EXT INT, that is, on the transition from the active level to the inactive level.

Interrupt Handling

When either a timer or an external interrupt request is communicated to the CPU section of the 3870, it will be acknowledged and processed at the completion of the first nonprivileged instruction, if the Interrupt Control Bit of the Status Register is set. If the Interrupt Control Bit is not set, the interrupt request will continue until either the Interrupt Control Bit is set and the CPU section acknowledges the interrupt, or until the interrupt request is cleared as previously described.

If there is both a timer interrupt request and an external interrupt request when the CPU section starts to process the requests, the timer interrupt is handled first.

When an interrupt is allowed, the CPU section will request that the interrupting element pass its interrupt vector address to the Program Counter via the data bus. The vector address for a timer interrupt is H '020'. The vector address for external interrupts is H '0A0'. After the vector address is passe the Program Counter, the CPU section sends an acknowledge signal to the appropriate interrupt request latch which clears that latch. The execution of the interrupt service routine will then commence. The return address of the original program is automatically saved in the Stack Register, P.

Power-On Clear

When power is applied to the 3870, the Program Counter and the ICB bit of the W Status Register are cleared. Ports 4, 5, 6, and 7 are loaded with H '00' (thus the I/O pins for ports 4 and 5 are at +V unless external circuitry is forcing the pins to GND). The contents of other registers and ports are undefined. The first program instruction is then fetched from ROM location H '000'.

External Reset

When $\overline{\text{RESET}}$ is taken low, the content of the Program Counter is pushed to the Stack Register and then the Program Counter and the ICB bit of the W Status Register are cleared. The original Stack Register content is lost. As with power-on clear, ports 4, 5, 6, and 7 are loaded with H '00'. The contents of all other registers and ports are unchanged. When $\overline{\text{RESET}}$ is taken high, the first program instruction is fetched from ROM location H '000'.

Test Logic

Special test logic is implemented to allow access to the internal main data bus for test purposes.

In normal operation the TEST pin is unconnected or is connected to GND. When TEST is placed at a TTL level (2.0 V to 2.6 V), port 4 becomes an output of the internal data bus and port 5 becomes a wired-OR input to the internal data bus. The data appearing on the port 4 pins is logically true, whereas input data forced on port 5 must be logically false. When TEST is placed at a high level (6.0 V to 7.0 V), the ports act as above and, additionally, the 2K x 8 program ROM is prevented from driving the data bus. In this mode, operands and instructions may be forced externally through port 5 instead of being accessed from the program ROM. When TEST is in either the TTL state or the high state, $\overline{\text{STROBE}}$ ceases its normal function and becomes a machine cycle clock (identical to the F8 multichip system WRITE clock, except inverted).

Timing complexities render the capabilities associated with the TEST pin impractical for use in a user's application, but these capabilities are thoroughly sufficient to enable Motorola to rapidly test the MC3870.

3870 Clocks

The time base for the 3870 may originate from one of five sources. There are four external modes and one internal mode.

If both XTL 1 and XTL 2 are grounded, the 3870 will activate its internal oscillator.

The four external configurations are shown in Figure 4. There is an internal 20 pF capacitor between XTL 1 and GND and an internal 20 pF capacitor between XTL 2 and GND. Thus, external capacitors are not required. In all external clock modes, the external time base frequency is divided by two to form the internal ϕ clock.

INSTRUCTION SET

The MC3870 executes the entire instruction set of the multichip F8 family. Of course, the STORE instruction is of little use in the 3870 because only Read-Only Memory exists in the addressing range of the Data Counter (the Data Counter will, however, be incremented if a STORE is executed).

A summary of programmable registers and ports is given in Figure 5, followed by a summary of the F8 instruction set.

Also, for convenient reference, a Programming Model of the 3870 is given in Figure 6.

SUPPLEMENTARY NOTES

The Interrupt Control Bit of the W Status Register is automatically reset when an interrupt request is acknowledged. It is then the programmer's responsibility to determine when ICB will again be set (by executing an EI instruction). This action prevents an interrupt service routine from being interrupted unless the programmer so desires.

When reading the Interrupt Control Port (port 6), bit 7 of the Accumulator is loaded with the actual logic level being applied to the EXT INT pin, regardless of the status of ICP bit 2 (the EXT INT Active Level bit). That is, if EXT INT is at +5 V, bit 7 of the Accumulator is set to a logic 1; but, if EXT INT is at GND, then Accumulator bit 7 is reset to logic 0.

In the MC3870 (F8 COMPATIBLE) INSTRUCTION SET summary, the number of cycles shown are "nominal" machine cycles. A nominal machine cycle is defined as 4 ϕ clock periods, thus requiring 2 μs for a 2 MHz ϕ clock frequency (4 MHz external time base frequency).

Also, the summary uses an older nomenclature for register names. The translation is as follows:

PC0 = P0	Program counter
PC1 = P	Stack Register
DC0 = DC	Data Counter
DC1 = DC1	Auxiliary Data Counter

The nomenclature is used in order to be consistent with the assembly language mnemonics.

For the MC3870, execution of an INS or OUTS instruction requires 2 machine cycles for ports 0 and 1, whereas ports 4 and 5 require 4 machine cycles.

When an external reset of the MC3870 occurs. P0 is pushed into P and the old contents of P are lost. It must be noted that an external reset is recognized at the start of a machine cycle and not necessarily at the end of an instruction. Thus, if the MC3870 is executing a multicycle instruction, that instruction is not completed and the contents of P upon reset may not necessarily be the address of the instruction that would have been executed next. It may, for example, point to an immediate operand if the reset occurred during the second cycle of an LI or CI instruction. Additionally, several instructions (JMP, PI, PK, LR P0,Q), as well as the interrupt acknowledge sequence, modify P0 in parts. That is, they alter P0 by first loading one part, then the other, and the entire operation takes more than one cycle. Should reset occur during this modification process, the value pushed into P will be part of the old P0 (the as yet unmodified part) and part of the new P0 (already modified part). Thus care should be taken (perhaps by external gating) to insure that reset does not occur at an undesirable time, if any significance is to be given to the contents of P after a reset occurs.

FIGURE 4 – CLOCK CONFIGURATIONS

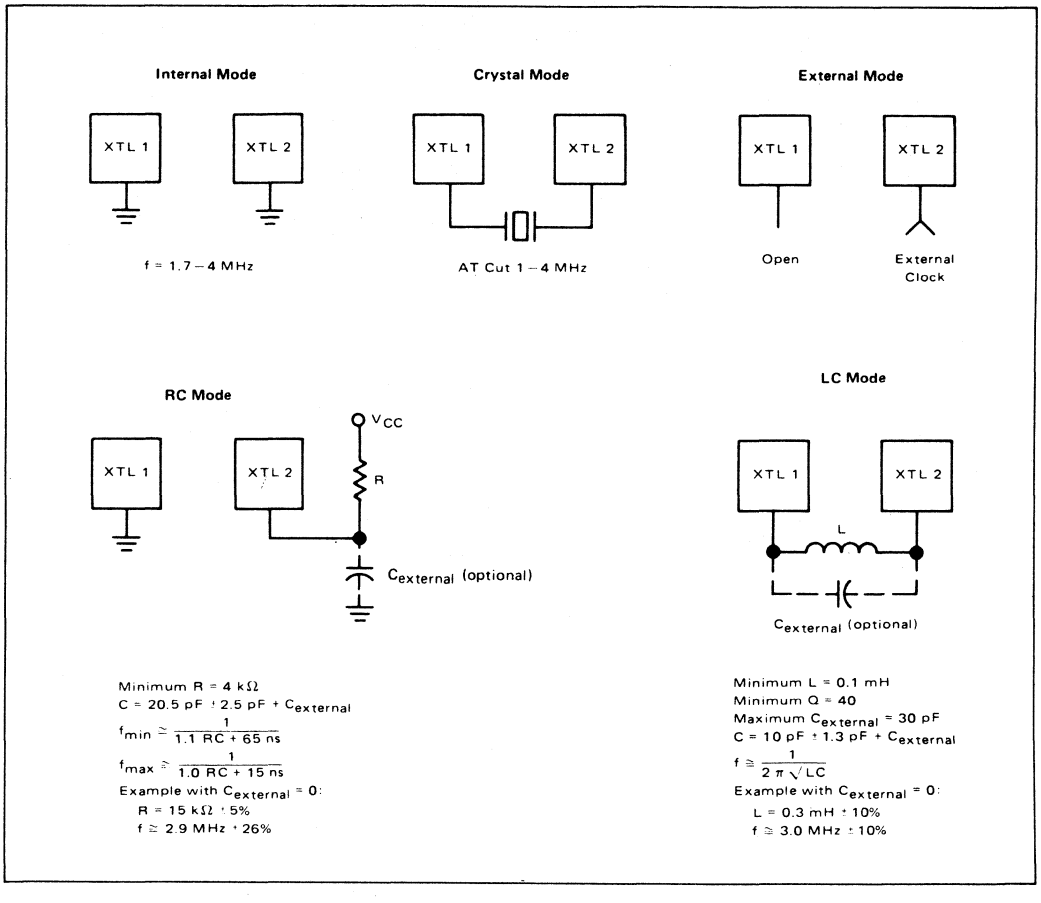
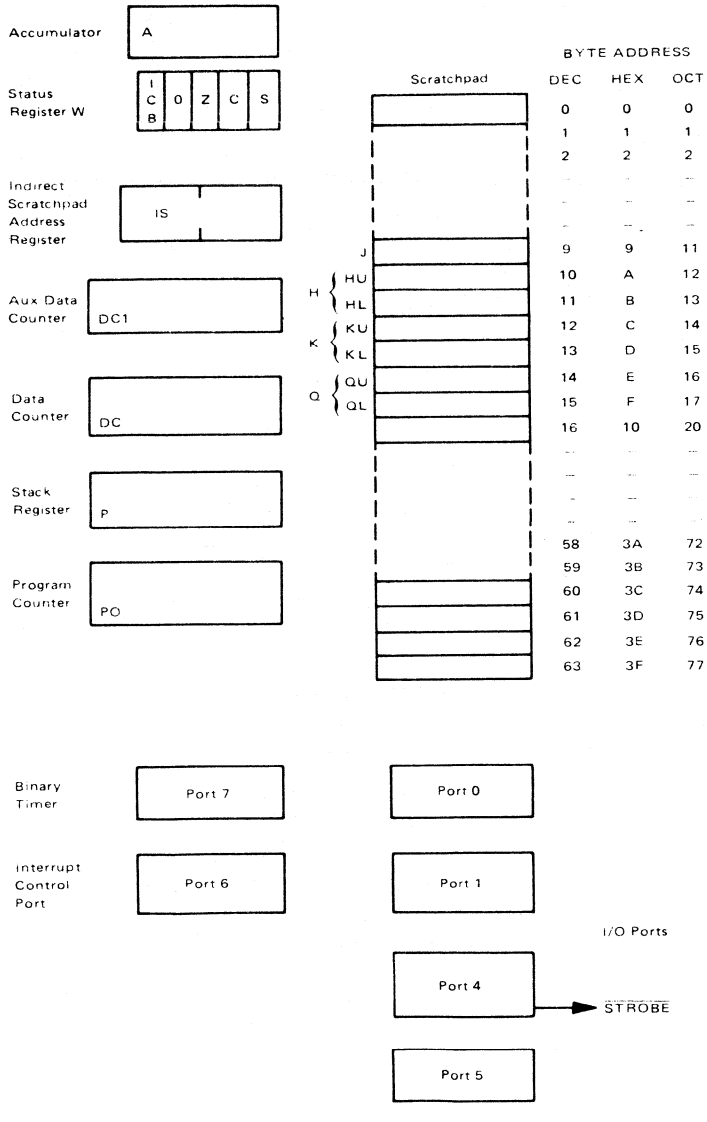


FIGURE 5 – PROGRAMMABLE REGISTERS AND PORTS



MC3870 INSTRUCTION SET
(F8 COMPATIBLE)

ACCUMULATOR GROUP INSTRUCTIONS

Operation	Mnemonic Op Code	Operand	Function	Machine Code	Bytes	Cycles	Status Bits OVF	ZERO	CRY	SIGN
ADD CARRY	LNK		ACC←(ACC) + CRY	19	1	1	1/0	1/0	1/0	1/0
ADD IMMEDIATE	AI	ii	ACC←(ACC) + H'ii'	24ii	2	2.5	1/0	1/0	1/0	1/0
AND IMMEDIATE	NI	ii	ACC←(ACC) ∧ H'ii'	21ii	2	2.5	0	1/0	0	1/0
CLEAR	CLR		ACC←H'00'	70	1	1	–	–	–	–
COMPARE IMMEDIATE	CI	ii	H'ii' + $(\overline{ACC})$ + 1	25ii	2	2.5	1/0	1/0	1/0	1/0
COMPLEMENT	COM		ACC←(ACC) ⊕ H'FF'	18	1	1	0	1/0	0	1/0
EXCLUSIVE OR IMMEDIATE	XI	ii	ACC←(ACC) ⊕ H'ii'	23ii	2	2.5	0	1/0	0	1/0
INCREMENT	INC		ACC←(ACC) + 1	1F	1	1	1/0	1/0	1/0	1/0
LOAD IMMEDIATE	LI	ii	ACC←H'ii'	20ii	2	2.5	–	–	–	–
LOAD IMMEDIATE SHORT	LIS	i	ACC←H'0i'	7i	1	1	–	–	–	–
OR IMMEDIATE	OI	ii	ACC←(ACC) ∨ H'ii'	22ii	2	2.5	0	1/0	0	1/0
SHIFT LEFT ONE	SL	1	SHIFT LEFT 1	13	1	1	0	1/0	0	1/0
SHIFT LEFT FOUR	SL	4	SHIFT LEFT 4	15	1	1	0	1/0	0	1/0
SHIFT RIGHT ONE	SR	1	SHIFT RIGHT 1	12	1	1	0	1/0	0	1
SHIFT RIGHT FOUR	SR	4	SHIFT RIGHT 4	14	1	1	0	1/0	0	1

BRANCH INSTRUCTIONS

In all conditional branches PC_0←(PC_0) + 2 if the test condition is not met. Execution is complete in 3.0 cycles.

Operation	Mnemonic Op Code	Operand	Function	Machine Code	Bytes	Cycles	Status Bits OVF	ZERO	CRY	SIGN
BRANCH ON CARRY	BC	aa	PC_0←[(PC_0) + 1] + H'aa' if CRY = 1	82aa	2	3.5	–	–	–	–
BRANCH ON POSITIVE	BP	aa	PC_0←[(PC_0) + 1] + H'aa' if SIGN = 1	81aa	2	3.5	–	–	–	–
BRANCH ON ZERO	BZ	aa	PC_0←[(PC_0) + 1] + H'aa' if ZERO = 1	84aa	2	3.5	–	–	–	–
BRANCH ON TRUE	BT	taa	PC_0←[(PC_0) + 1] + H'aa' if any test is true	8taa	2	3.5	–	–	–	–

t = TEST CONDITION

2^2	2^1	2^0
ZERO	CRY	SIGN

BRANCH IF NEGATIVE	BM	aa	PC_0←[(PC_0) + 1] + H'aa' if SIGN = 0	91aa	2	3.5	–	–	–	–
BRANCH IF NO CARRY	BNC	aa	PC_0←[(PC_0) + 1] + H'aa' if CARRY = 0	92aa	2	3.5	–	–	–	–
BRANCH IF NO OVERFLOW	BNO	aa	PC_0←[(PC_0) + 1] + H'aa' if OVF = 0	98aa	2	3.5	–	–	–	–
BRANCH IF NOT ZERO	BNZ	aa	PC_0←[(PC_0) + 1] + H'aa' if ZERO = 0	94aa	2	3.5	–	–	–	–
BRANCH IF FALSE TEST	BF	taa	PC_0←[(PC_0) + 1] + H'aa' if all false test bits	9taa	2	3.5	–	–	–	–

t = TEST CONDITION

2^3	2^2	2^1	2^0
OVF	ZERO	CRY	SIGN

BRANCH IF ISAR (LOWER) ≠ 7	BR7	aa	PC_0←[(PC_0) + 1] + H'aa' if ISARL ≠ 7	8Faa	2	2.5	–	–	–	–
			PC_0←(PC_0) + 2 if ISARL = 7			2.0	–	–	–	–
BRANCH RELATIVE	BR	aa	PC_0←[(PC_0 + 1] + H'aa'	90aa	2	3.5	–	–	–	–
JUMP*	JMP	aaaa	PC_0←H'aaaa'	29aaaa	3	5.5	–	–	–	–

*Privileged instruction

MEMORY REFERENCE INSTRUCTIONS In all Memory Reference Instructions, the Data Counter is incremented DC←DC + 1

Operation	Mnemonic OP Code	Operand	Function	Machine Code	Bytes	Cycles	Status Bits OVF	ZERO	CRY	SIGN
ADD BINARY	AM		ACC←(ACC) + [(DC)]	88	1	2.5	1/0	1/0	1/0	1/0
ADD DECIMAL	AMD		ACC←(ACC) + [(DC)]	89	1	2.5	1/0	1/0	1/0	1/0
AND	NM		ACC←(ACC)∧[(DC)]	8A	1	2.5	0	1/0	0	1/0
COMPARE	CM		[(DC)] + ($\overline{ACC}$) + 1	8D	1	2.5	1/0	1/0	1/0	1/0
EXCLUSIVE OR	XM		ACC←(ACC) ⊕ [(DC)]	8C	1	2.5	0	1/0	0	1/0
LOAD	LM		ACC←[(DC)]	16	1	2.5	–	–		–
LOGICAL OR	OM		ACC←(ACC)∨[(DC)]	8B	1	2.5	0	1/0	0	1/0
STORE	ST		(DC)←(ACC)	17	1	2.5	–	–	–	–

ADDRESS REGISTER GROUP INSTRUCTIONS

Operation	Mnemonic OP Code	Operand	Function	Machine Code	Bytes	Cycles	Status Bits OVF	ZERO	CRY	SIGN
ADD to DATA COUNTER	ADC		DC←(DC) + (ACC)	8E	1	2.5	–	–	–	–
CALL to SUBROUTINE*	PK		PC_0U←(r12); PC_0L←(r13); PC_1←(PC_0)	OC	1	4	–	–	–	–
CALL to SUBROUTINE IMMEDIATE*	PI	aaaa	PC_1←(PC_0); PC_0←H'aaaa'	28aaaa	3	6.5	–	–	–	–
EXCHANGE DC	XDC		DC_0⇆DC_1	2C	1	2	–	–	–	–
LOAD DATA COUNTER	LR	DC,Q	DCU←(r14); DCL←(r15)	0F	1	4	–	–	–	–
LOAD DATA COUNTER	LR	DC,H	DCU←(r10); DCL←(r11)	10	1	4	–	–	–	–
LOAD DC IMMEDIATE	DCI	aaaa	DC←H'aaaa'	2Aaaaa	3	6	–	–	–	–
LOAD PROGRAM COUNTER	LR	PO,Q	PC_0U←(r14); PC_0L←(r15)	OD	1	4	–	–	–	–
LOAD STACK REGISTER	LR	P,K	PC_1U←(r12); PC_1L←(r13)	09	1	4	–	–	–	–
RETURN FROM SUBROUTINE*	POP		PC_0←(PC_1)	1C	1	2	–	–	–	–
STORE DATA COUNTER	LR	Q,DC	r14←(DCU); r15←(DCL)	OE	1	4	–	–	–	–
STORE DATA COUNTER	LR	H,DC	r10←(DCU); r11←(DCL)	11	1	4	–	–	–	–
STORE STACK REGISTER	LR	K,P	r12←(PC_1U); r13←(PC_1L)	08	1	4	–	–	–	–

SCRATCHPAD REGISTER INSTRUCTIONS (Refer to Scratchpad Addressing Modes)

Operation	Mnemonic OP Code	Operand	Function	Machine Code	Bytes	Cycles	Status Bits OVF	ZERO	CRY	SIGN
ADD BINARY	AS	r	ACC←(ACC) + (r)	Cr	1	1	1/0	1/0	1/0	1/0
ADD DECIMAL	ASD	r	ACC←(ACC) + (r)	Dr	1	2	1/0	1/0	1/0	1/0
DECREMENT	DS	r	r←(r) + H'FF'	3r	1	1.5	1/0	1/0	1/0	1/0
LOAD	LR	A,r	ACC←(r)	4r	1	1	–	–	–	–
LOAD	LR	A,KU	ACC←(r12)	00	1	1	–	–	–	–
LOAD	LR	A,KL	ACC←(r13)	01	1	1	–	–	–	–
LOAD	LR	A,QU	ACC←(r14)	02	1	1	–	–	–	–
LOAD	LR	A,QL	ACC←(r15)	03	1	1	–	–	–	–
LOAD	LR	r,A	r←(ACC)	5r	1	1	–	–	–	–
LOAD	LR	KU,A	r12←(ACC)	04	1	1	–	–	–	–
LOAD	LR	KL,A	r13←(ACC)	05	1	1	–	–	–	–
LOAD	LR	QU,A	r14←(ACC)	06	1	1	–	–	–	–
LOAD	LR	QL,A	r15←(ACC)	07	1	1	–	–	–	–
AND	NS	r	ACC←(ACC)∧(r)	Fr	1	1	0	1/0	0	1/0
EXCLUSIVE OR	XS	r	ACC←(ACC) ⊕ (r)	Er	1	1	0	1/0	0	1/0

*Privileged Instruction.

MISCELLANEOUS INSTRUCTIONS

Operation	Mnemonic OP Code	Operand	Function	Machine Code	Bytes	Cycles	Status Bits OVF	ZERO	CRY	SIGN
DISABLE INTERRUPT	DI		RESET ICB	1A	1	2	–	–	–	–
ENABLE INTERRUPT*	EI		SET ICB	1B	1	2	–	–	–	–
INPUT	IN	aa	ACC←(INPUT PORT aa)	26aa	2	4	0	1/0	0	1/0
INPUT SHORT	INS	a	ACC←(INPUT PORT a)	Aa	1	4***	0	1/0	0	1/0
LOAD ISAR	LR	IS,A	ISAR←(ACC)	0B	1	1	–	–	–	–
LOAD ISAR LOWER	LISL	a	ISARL←a	1101a**	1	1	–	–	–	–
LOAD ISAR UPPER	LISU	a	ISARU←a	01100a**	1	1	–	–	–	–
LOAD STATUS REGISTER*	LR	W,J	W←(r9)	1D	1	2	1/0	1/0	1/0	1/0
NO-OPERATION	NOP		PC_0←(PC_0) + 1	2B	1	1	–	–	–	–
OUTPUT	OUT	aa	OUTPUT PORT aa←(ACC)	27aa	2	4	–	–	–	–
OUTPUT SHORT	OUTS	a	OUTPUT PORT a←(ACC)	Ba	1	4***	–	–	–	–
STORE ISAR	LR	A,IS	ACC←(ISAR)	0A	1	1	–	–	–	–
STORE STATUS REG	LR	J,W	r9←(W)	1E	1	1	–	–	–	–

*Privileged Instruction
**3-Bit Octal Digit
***2 Machine Cycles for CPU Ports

NOTES

Each lowercase character represents a Hexadecimal digit.
Each cycle equals 4 machine clock periods.
Lowercase denotes variables specified by programmer.

Function Definitions

←	is replaced by
( )	the contents of
(–)	Binary "1"s complement of
+	Arithmetic Add (Binary or Decimal)
⊕	Logical OR exclusive
∧	Logical AND
∨	Logical OR inclusive
H'	Hexadecimal digit

Register Names

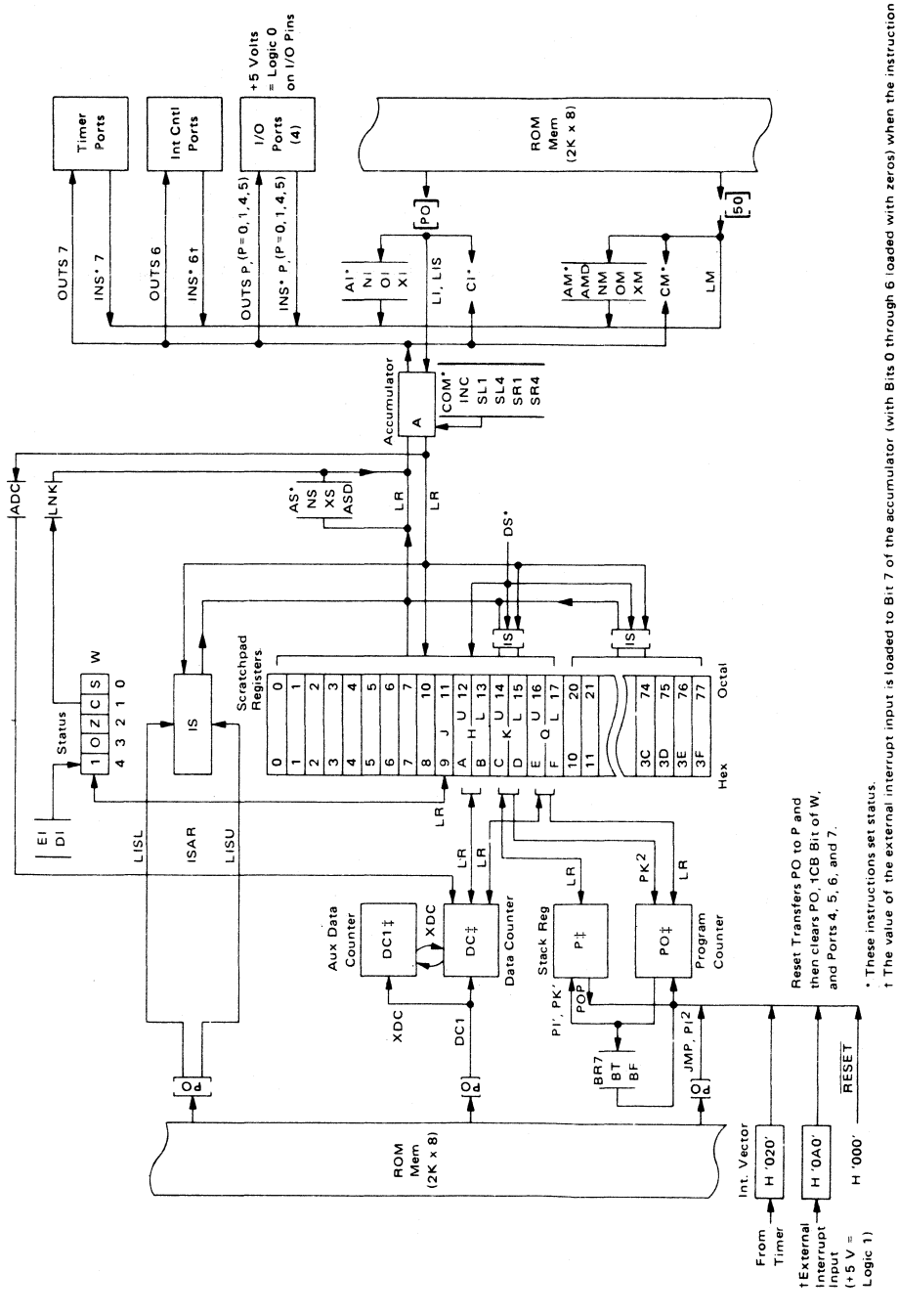
a	Address Variable
A	Accumulator
DC	Data Counter (Indirect Address Register)
DC_0	Data Counter =0 (Indirect Address Register =0)
DC_1	Data Counter =1 (Indirect Address Register =1)
DCL	Least significant 8 bits of Data Counter Addressed
DCU	Most significant 8 bits of Data Counter Addressed
H	Scratchpad Register #10 and #11
i and ii	Immediate Operand
ICB	Interrupt Control Bit
IS	Indirect Scratchpad Address Register
ISAR	Indirect Scratchpad Address Register
ISARL	Least significant 3 bits of ISAR
ISARU	Most significant 3 bits of ISAR
J	Scratchpad Register =9
K	Registers =12 and =13
KL	Register =13
KU	Register =12
PC_0	Program Counter
PC_0L	Least signficant 8 bits of Program Counter
PC_0U	Most significant 8 bits of Program Counter
PC_1	Stack Register
PC_1L	Least significant 8 bits of Program Counter
PC_1U	Most signficant 8 bits of Active Stack Register
Q	Registers =14 and =15
QL	Register =15
QU	Register =14
r	Scratchpad Register (any address through 11)
W	Status Register

Scratchpad Addressing Modes (Machine Code Format)

r C	(Hexadecimal), Register Addressed by ISAR (Unmodified)
r D	(Hexadecimal), Register Addressed by ISAR, ISARL Incremented
r E	(Hexadecimal), Register Addressed by ISAR, ISARL Decremented
r F	(No operation performed)
r 0 through B	(Hexadecimal), Register 0 through 11 addressed directly from the Instruction

Status Register

–	No change in condition
1/0	is set to "1" or "0" depending on conditions
CRY	Carry Flag
OVF	Overflow Flag
SIGN	Sign of Result Flag
ZERO	Zero Flag



* These instructions set status.
† The value of the external interrupt input is loaded to Bit 7 of the accumulator (with Bits 0 through 6 loaded with zeros) when the instruction 'INS 6' is executed. This instruction also sets status.
‡ PO, P, DC1 are 11-bit registers.
NOTE: The Instructions PI and PK are shown in two sequential parts. (PI^1, PI^2 and PK^1, PK^2).

FIGURE 6 – PROGRAMMING MODEL

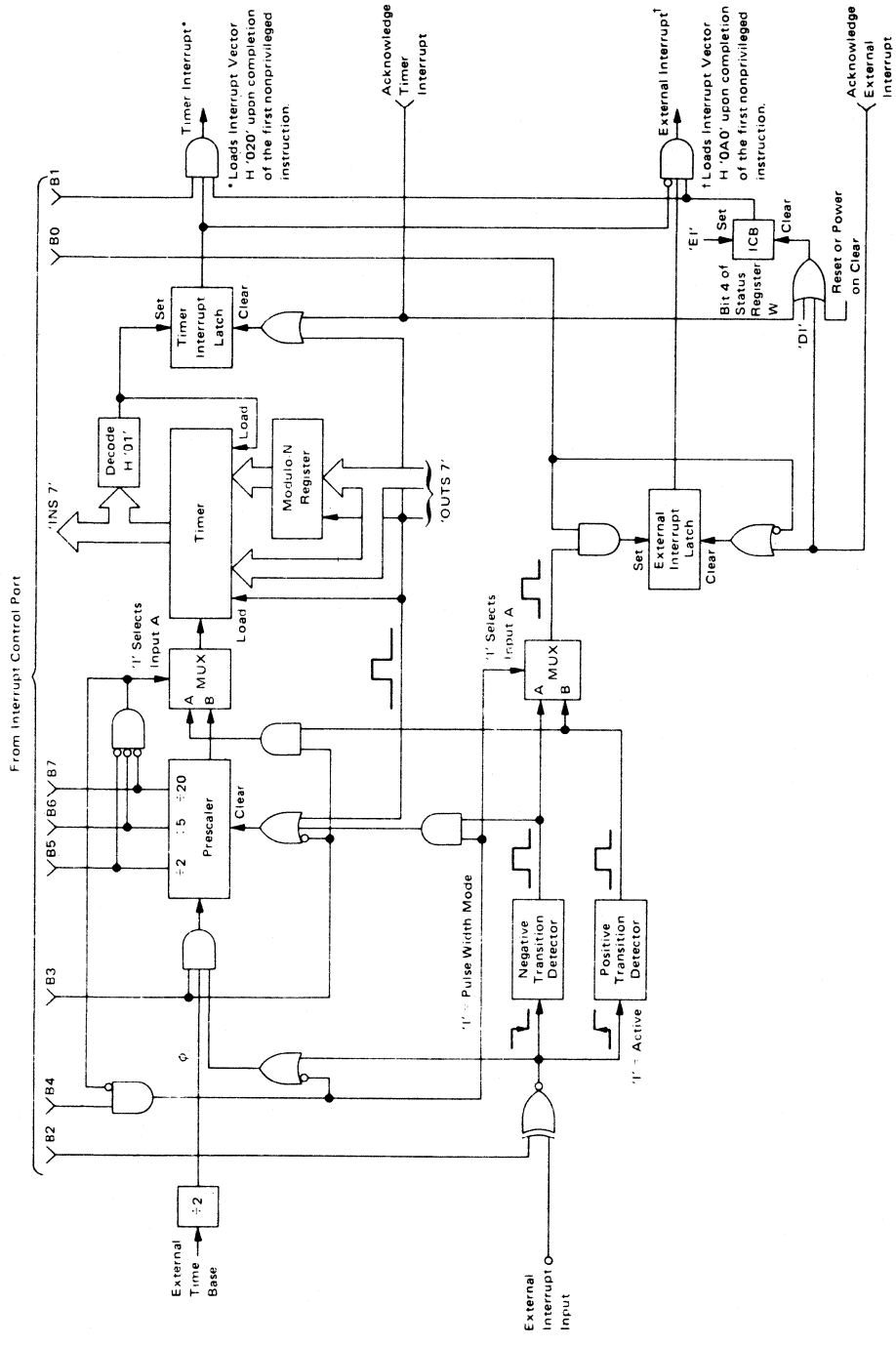


FIGURE 7 – TIMER/INTERRUPT FUNCTIONAL DIAGRAM

APPENDIX

CUSTOM MC3870 ORDERING INFORMATION

A.0 Custom MC3870 Ordering Information

The custom MC3870 specifications may be transmitted to Motorola in any of the following media:

1) 3870 Emulator Module (3870 EM)
2) PROM(s) from the 3870 EM
3) Assembler formatted object tape
4) Punched card deck
5) Paper tape of card deck format

The specification should be formatted and packed as indicated in the appropriate paragraph below and mailed prepaid and insured with a cover letter (see Figure A-2) to:

MOTOROLA INC.
3501 Ed Bluestein Boulevard
Austin, TX 78721

A copy of the cover letter should also be mailed separately.

A.1 3870 Emulator Module (3870 EM)

The 3870 EM circuit board, with the customer program in PROM(s) installed on the board, may be used to specify the custom MC3870 pattern. The circuit board should NEVER be placed in contact with styrofoam materials. The circuit board should then be securely packaged for shipment.

A.2 PROMs

MCM2708 or MCM2716 type PROMs, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. The PROMs must be clearly marked to indicate which PROM corresponds to which address space (000-3FF HEX). See Figure A-1 for recommended marking procedure.

After the PROM(s) are marked and removed from the Emulator, they should be placed in conductive IC carriers and securely packed. Do not use styrofoam. Black IC carriers are provided with the 3870 EM for this purpose.

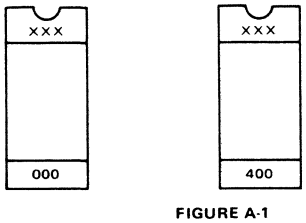


FIGURE A-1

A.3 Assembler Formatted Object Tape

Cassette tapes produced on a Silent 700 terminal using the 3870 EM and EXORciser are acceptable.

A.4 Punched Card Deck

The custom MC3870 may be specified for manufacture in the form of standard 80-column punched cards.

The card deck for specifying the Custom MC3870 has the following format:

OPTION CARD
COMMENT CARDS
X CARDS
C CARDS

Option Card

The first card in the deck must be the OPTION CARD. The format is as follows:

Column 1-20: Customer name. Any 20 characters may be used.

Column 25-29: This is a 5-digit number assigned by Motorola. Leave this field blank. It will be punched at Motorola unless otherwise notified.

Column 32: Port Type. This specifies the interface option of Ports 4 and 5.
1 = Standard TTL
2 = Open Drain
3 = Direct Drive

Column 37-39: Address field base on output listing. The characters HEX or DEC specify the output listing address base as hexadecimal or decimal, respectively. If omitted, DEC is assumed.

Colum 41-43: Data field base on output listing. The characters HEX or DEC specify the output listing ROM data base as hexadecimal or decimal, respectively. If omitted, DEC is assumed.

Comment Cards

Comment Cards must have an asterisk (*) in column 1. The remaining 79 columns may contain any letter, number, or character.

X Cards

Five X cards are possible. All X cards have an X in column 1 and one or three or more words each separated by one blank space.

The possibilities are:

1) X SEQUENCE
2) X BASE DEC DEC
3) X BASE DEC HEX
4) X BASE HEX DEC
5) X BASE HEX HEX

Card 1 specifies that there are sequence numbers on each data card that follows. The sequence numbers must be in columns 77-79 of the data cards (C Cards) and must be in decimal, right justified. The numbers must start with 1 (one) and must be in order. The X SEQUENCE Card may appear anywhere within the deck after the Option Card. If it appears within the data card section, data cards encountered before the X SEQUENCE Card will not be checked for sequence numbers. All following cards will be checked. If no X SEQUENCE Card is used, no sequence numbers will be checked.

It is initially assumed that the address and byte count as well as the data specified on the C Cards will be in decimal. An X BASE Card can be used to override this specification. The second word on the card (that following BASE) specifies the base (either DEC or HEX) of the address and byte count on all following C cards. The last word specifies the base of the data fields on the C Cards. An X BASE card may appear anywhere within the deck following the OPTION Card. It may be overridden by another X BASE Card. All data cards (C Cards) following an X BASE Card will be interpreted as per that X BASE Card unless another X BASE Card is encountered. If no X BASE Cards are used, it is assumed that all fields on the C Cards are in decimal.

NOTE: Once an X SEQUENCE Card is encountered, all successive cards will be checked for the proper sequence number and unlike X BASE Cards this option cannot thereafter be altered by another X SEQUENCE Card.

C Cards

These cards contain the actual ROM data. All fields are right-justified.

Column 1: C (the letter C)
Column 2-9: ADD
Column 10-12: BYTE
Column 14-16: DATA 1
Column 17-19: DATA 2
. .
. .
. .
Column 76-78: DATA 21
Column 77-79: DATA 22 or SEQUENCE NUMBER

ADD is the address of the first byte of data (DATA 1) contained on that card. Byte is the number of bytes of data to be read from that card. BYTE must be greater than zero and less than 23 (1-22) if no sequence numbers are used, and less than 22 (1-21) if sequence numbers are used. If, for example, there are ten data fields punched on the card, but BYTE = 2, only the first two will be read. Also, if there are two punched data fields, for example, and BYTE = 6, six ROM locations will be filled from that card. The four unspecified fields will be decoded as zero. ADD and BYTE are always in the same base (HEX or DECIMAL). DATA 1 through DATA N is the data to be placed in the ROM at addresses ADD through ADD + (N−1), respectively.

Any ROM address not filled as a result of reading data from a C Card will be filled with zero. If a particular location has already been specified by a C Card, but a successive C Card also has the data which is to be placed in that location, the second C Card will override the first.

A.5 Paper Tape of Card Deck Format

Punched Paper tape (ASCII) in the same format as cards can also be accepted. However, your order will be processed faster if the data is in card format. After the tape leader there should be a <CR><LF>. Data records should be a full 80 columns, each terminated by a <CR><LF>. Following the last Data record, there should be one more record with the first three characters being EOF, followed by 77 blanks and a <CR><LF>.

<CR> = Carriage Return
<LF> = Line Feed

CUSTOMER NAME _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

PHONE (_____) _____ EXTENSION _____

CONTACT MS/MR _____

CUSTOMER PART # _____

PORT OPTION (NOTE 1)
- ☐ Standard TTL
- ☐ Open Drain
- ☐ Direct Drive

PATTERN MEDIA
- ☐ 3870 EM
- ☐ 2708 PROM
- ☐ 2716 PROM
- ☐ Paper Object Tape
- ☐ Silent 700 Cassette
- ☐ Card Deck
- ☐ Tape of Card Deck
- ☐ (Note 2) _____

NOTES: (1) Ports 4 and 5 Only
(2) Other Media Require Prior Factory Approval

SIGNATURE _____

TITLE _____

FIGURE A-2

chapter 4

CMOS MCUs/ICUs

MOTOROLA

MC14500B

CMOS LSI

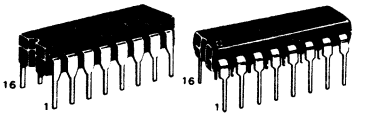
(LOW-POWER COMPLEMENTARY MOS)

INDUSTRIAL CONTROL UNIT

INDUSTRIAL CONTROL UNIT

The MC14500B Industrial Control Unit (ICU) is a single bit CMOS processor. The ICU is designed for use in systems requiring decisions based on successive single bit information. An external ROM stores the control program. With a program counter (and output latches and input multiplexers, if required) the ICU in a system forms a stored program controller that replaces combinatorial logic. Applications include relay logic processing, serial data manipulation and control. The ICU also may control an MPU or be controlled by an MPU.

- 16 Instructions
- DC to 1.0 MHz Operation at V_{DD} = 5 V
- On Chip Clock (Oscillator)
- Executes One Instruction per Clock Cycle
- 3 V to 18 V Operation
- Noise Immunity Typically 45% of V_{DD}
- Quiescent Current 5.0 μAdc Typical at V_{DD} = 5 V
- Capable of Driving One Low-Power Schottky Load or Two Low-Power TTL Loads over Full Temperature Range



L SUFFIX
CERAMIC PACKAGE
CASE 620

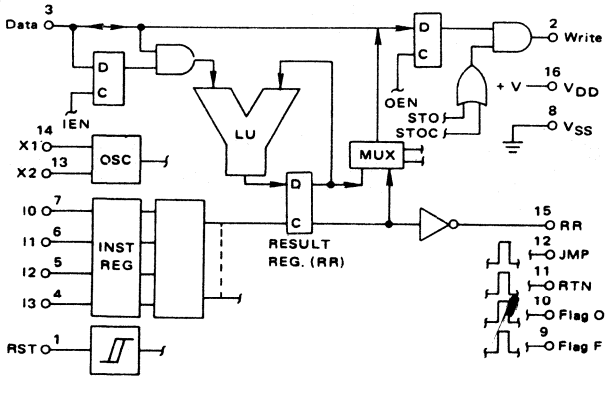
P SUFFIX
PLASTIC PACKAGE
CASE 648

ORDERING INFORMATION

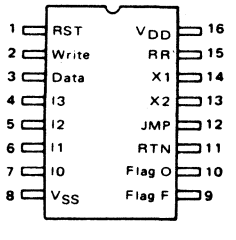
MC14XXXB — Suffix Denotes
- L Ceramic Package
- P Plastic Package
- A Extended Operating Temperature Range
- C Limited Operating Temperature Range

Detailed operation and applications are given in the "MC14500B Industrial Control Unit" handbook.

BLOCK DIAGRAM



PIN ASSIGNMENT



MAXIMUM RATINGS (Voltages referenced to V_{SS})

Rating	Symbol	Value	Unit
DC Supply Voltage	V_{DD}	-0.5 to +18	Vdc
Input Voltage, All Inputs	V_{in}	-0.5 to V_{DD} + 0.5	Vdc
DC Current Drain per Pin	I	10	mAdc
Operating Temperature Range — AL Device CL/CP Device	T_A	-55 to +125 -40 to +85	°C
Storage Temperature Range	T_{stg}	-65 to +150	°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{DD}$.

ELECTRICAL CHARACTERISTICS

Characteristic		Symbol	V_{DD} Vdc	T_{low}* Min	T_{low}* Max	25°C Min	25°C Typ	25°C Max	T_{high}* Min	T_{high}* Max	Unit
Output Voltage	"0" Level	V_{OL}	5.0	–	0.05	–	0	0.05	–	0.05	Vdc
V_{in} = V_{DD} or 0			10	–	0.05	–	0	0.05	–	0.05	
			15	–	0.05	–	0	0.05	–	0.05	
	"1" Level	V_{OH}	5.0	4.95	–	4.95	5.0	–	4.95	–	Vdc
V_{in} = 0 or V_{DD}			10	9.95	–	9.95	10	–	9.95	–	
			15	14.95	–	14.95	15	–	14.95	–	
Input Voltage #	"0 Level"	V_{IL}									Vdc
RST, D, X2											
(V_O = 4.5 or 0.5 Vdc)			5.0	–	1.5	–	2.25	1.5	–	1.5	
(V_O = 9.0 or 1.0 Vdc)			10	–	3.0	–	4.50	3.0	–	3.0	
(V_O = 13.5 or 1.5 Vdc)			15	–	4.0	–	6.75	4.0	–	4.0	
	"1" Level	V_{IH}									Vdc
(V_O = 0.5 or 4.5 Vdc)			5.0	3.5	–	3.5	2.75	–	3.5	–	
(V_O = 1.0 or 9.0 Vdc)			10	7.0	–	7.0	5.50	–	7.0	–	
(V_O = 1.5 or 13.5 Vdc)			15	11.0	–	11.0	8.25	–	11.0	–	
Input Voltage #	"0" Level	V_{IL}									Vdc
I0, I1, I2, I3											
(V_O = 4.5 or 0.5 Vdc)			5.0	–	0.8	–	1.1	0.8	–	0.8	
(V_O = 9.0 or 1.0 Vdc)			10	–	1.6	–	2.2	1.6	–	1.6	
(V_O = 13.5 or 1.5 Vdc)			15	–	2.4	–	3.4	2.4	–	2.4	
	"1" Level	V_{IH}									Vdc
(V_O = 0.5 or 4.5 Vdc)			5.0	2.0	–	2.0	1.9	–	2.0	–	
(V_O = 1.0 or 9.0 Vdc)			10	6.0	–	6.0	3.1	–	6.0	–	
(V_O = 1.5 or 13.5 Vdc)			15	10	–	10	4.3	–	10	–	
Output Drive Current	Source	I_{OH}									mAdc
Data, Write (AL/CL/CP Device)											
(V_{OH} = 4.6 Vdc)			5.0	-1.0	–	-1.0	-2.0	–	-1.0	–	
(V_{OH} = 9.5 Vdc)			10	–	–	–	-6.0	–	–	–	
(V_{OH} = 13.5 Vdc)			15	–	–	–	-12	–	–	–	
(V_{OL} = 0.4 Vdc)	Sink	I_{OL}	5.0	1.6	–	1.6	3.2	–	1.6	–	mAdc
(V_{OL} = 0.5 Vdc)			10	–	–	–	6.0	–	–	–	
(V_{OL} = 1.5 Vdc)			15	–	–	–	12	–	–	–	
Output Drive Current	Source	I_{OH}									mAdc
Other Outputs (AL Device)											
(V_{OH} = 2.5 Vdc)			5.0	-3.0	–	-2.4	-4.2	–	-1.7	–	
(V_{OH} = 4.6 Vdc)			5.0	-0.64	–	-0.51	-0.88	–	-0.36	–	
(V_{OH} = 9.5 Vdc)			10	-1.6	–	-1.3	-2.25	–	-0.9	–	
(V_{OH} = 13.5 Vdc)			15	-4.2	–	-3.4	8.8	–	-2.4	–	
(V_{OL} = 0.4 Vdc)	Sink	I_{OL}	5.0	0.64	–	0.51	0.88	–	0.36	–	mAdc
(V_{OL} = 0.5 Vdc)			10	1.6	–	1.3	2.25	–	0.9	–	
(V_{OL} = 1.5 Vdc)			15	4.2	–	3.4	8.8	–	2.4	–	
Output Drive Current	Source	I_{OH}									mAdc
Other Outputs (CL/CP Device)											
(V_{OH} = 2.5 Vdc)			5.0	-2.5	–	-2.1	-4.2	–	-1.7	–	
(V_{OH} = 4.6 Vdc)			5.0	-0.52	–	-0.44	-0.88	–	-0.36	–	
(V_{OH} = 9.5 Vdc)			10	-1.3	–	-1.1	-2.25	–	-0.9	–	
(V_{OH} = 13.5 Vdc)			15	-3.6	–	-3.0	-8.8	–	-2.4	–	
(V_{OL} = 0.4 Vdc)	Sink	I_{OL}	5.0	0.52	–	0.44	0.88	–	0.36	–	mAdc
(V_{OL} = 0.5 Vdc)			10	1.3	–	1.1	2.25	–	0.9	–	
(V_{OL} = 1.5 Vdc)			15	3.6	–	3.0	8.8	–	2.4	–	

ELECTRICAL CHARACTERISTICS (continued)

Characteristic	Symbol	V_{DD} Vdc	T_{low}* Min	T_{low}* Max	25°C Min	25°C Typ	25°C Max	T_{high}* Min	T_{high}* Max	Unit
Input Current, RST (AL/CL/CP Device)	I_{in}	15	25	–	–	150	–	–	250	μAdc
Input Current (AL Device)	I_{in}	15	–	±0.1	–	±0.00001	±0.1	–	±1.0	μAdc
Input Current (CL/CP Device)	I_{in}	15	–	±0.3	–	±0.00001	±0.3	–	±1.0	μAdc
Input Capacitance (Data)	C_{in}	–	–	–	–	15	–	–	–	pF
Input Capacitance (All Other Inputs) (V_{in} = 0)	C_{in}	–	–	–	–	5.0	7.5	–	–	pF
Quiescent Current (AL Device) (Per Package)	I_{DD}	5.0 10 15	– – –	5.0 10 20	– – –	0.005 0.010 0.015	5.0 10 20	– – –	150 300 600	μAdc
Quiescent Current (CL/CP Device) (Per Package)	I_{DD}	5.0 10 15	– – –	20 40 80	– – –	0.005 0.010 0.015	20 40 80	– – –	150 300 600	μAdc
**Total Supply Current at an External Load Capacitance (C_L) on All Outputs	I_T	–	I_T = (1.5 μA/kHz) f + I_{DD} I_T = (3.0 μA/kHz) f + I_{DD} I_T = (4.5 μA/kHz) f + I_{DD}							μAdc

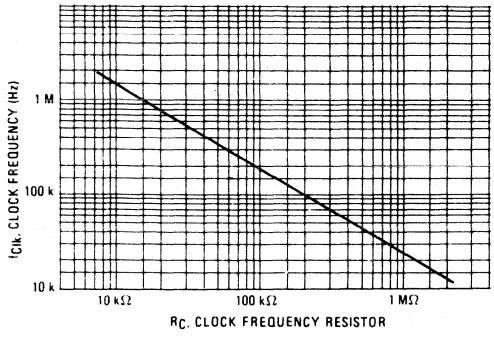
* T_{low} = -55°C for AL Device, -40°C for CL/CP Device.
T_{high} = +125°C for AL Device, +85°C for CL/CP Device.
** The formulas given are for the typical characteristics only at 25°C.
# Noise immunity specified for worst-case input combination.

SWITCHING CHARACTERISTICS (T_A = 25°C; $t_r = t_f$ = 20 ns for X and I inputs; C_L = 50 pF for JMP, X1, RR, Flag O, Flag F; C_L = 130 pF + 1TTL load for Data and Write.)

Characteristic	Symbol	V_{DD} Vdc	All Types Min	All Types Typ	All Types Max	Unit
Propagation Delay Time X1 to RR	t_{dR}	5.0 10 15	– – –	250 125 100	500 250 200	ns
X1 to Flag F, Flag O, RTN, JMP	t_{dF}	5.0 10 15	– – –	200 100 85	400 200 170	ns
X1 to Write	t_{dW}	5.0 10 15	– – –	225 125 100	450 250 200	ns
X1 to Data	t_{dD}	5.0 10 15	– – –	250 120 100	500 240 200	ns
RST to RR	t_{dRRR}	5.0 10 15	– – –	250 125 100	500 250 200	ns
RST to X1	t_{dRX}	5.0 10 15	– – –	450 200 150	Note 1	ns
RST to Flag F, Flag O, RTN, JMP	t_{dRF}	5.0 10 15	– – –	400 200 150	800 400 300	ns
RST to Write, Data	t_{dRW}	5.0 10 15	– – –	450 225 175	900 450 350	ns
Minimum Clock Pulse Width, X1	PW_C	5.0 10 15	– – –	200 100 90	400 200 180	ns
Minimum Reset Pulse Width, RST	PW_R	5.0 10 15	– – –	250 125 100	500 250 200	ns
Setup Time – Instruction	t_{IS}	5.0 10 15	400 250 180	200 125 90	– – –	ns
Data	t_{DS}	5.0 10 15	200 100 80	100 50 40	– – –	ns
Hold Time – Instruction	t_{IH}	5.0 10 15	100 50 50	0 0 0	– – –	ns
Data	t_{DH}	5.0 10 15	200 100 100	100 50 50	– – –	ns

NOTE 1. Maximum Reset Delay may extend to one-half clock period.

FIGURE 1 – TYPICAL CLOCK FREQUENCY versus RESISTOR (R_C)

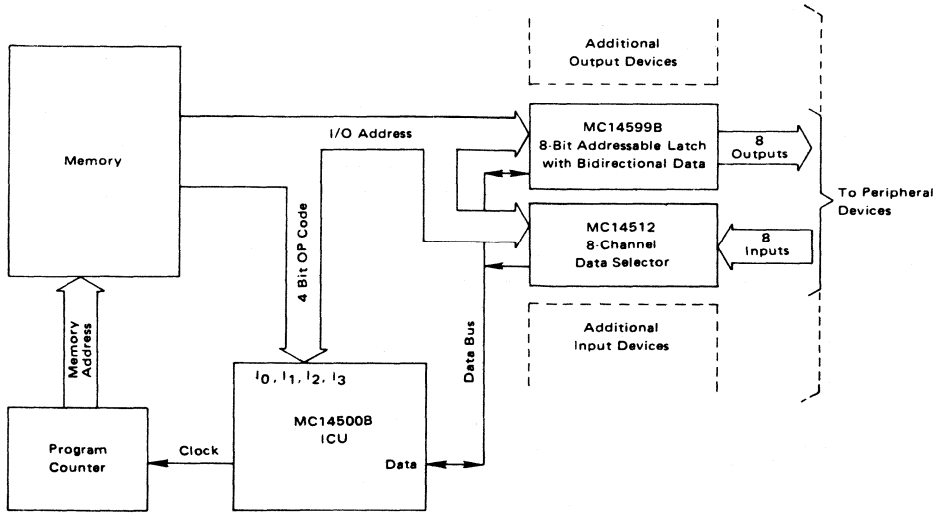


Pin No.	Function	Symbols
1	Chip Reset	RST
2	Write Pulse	Write
3	Data In/Out	Data
4	MSB Instruction Word	I_3
5	Bit 2 Instruction Word	I_2
6	Bit 1 Instruction Word	I_1
7	LSB Instruction Word	I_0
8	Negative Supply (Ground)	V_{SS}
9	Flag on NOP F	Flag F
10	Flag on NOP O	Flag O
11	Subroutine Return Flag	RTN
12	Jump Instruction Flag	JMP
13	Oscillator Input	X2
14	Oscillator Output	X1
15	Result Register	RR
16	Positive Supply	V_{DD}

TABLE 1. MC14500B INSTRUCTION SET

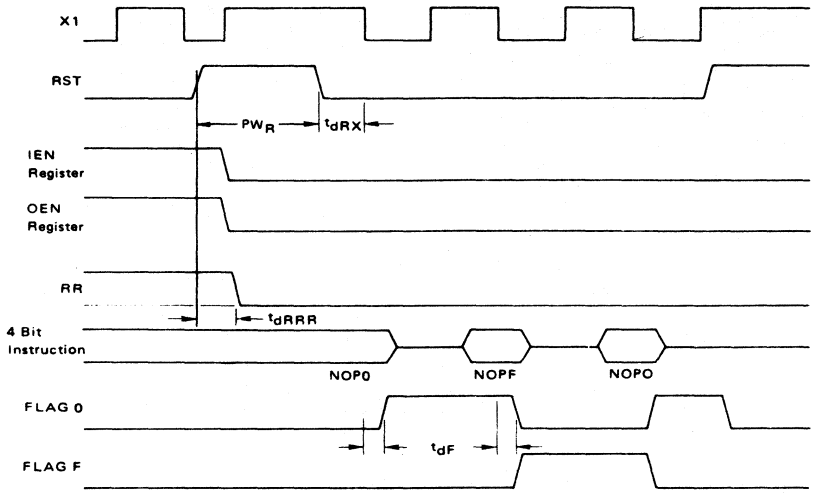
Instruction Code	Mnemonic	Action
0 0000	NOPO	No change in registers. RR → RR, Flag O → ⎍
1 0001	LD	Load result register. Data → RR
2 0010	LDC	Load complement. $\overline{Data}$ → RR
3 0011	AND	Logical AND. RR·Data → RR
4 0100	ANDC	Logical AND complement. RR·$\overline{Data}$ → RR
5 0101	OR	Logical OR. RR + Data → RR
6 0110	ORC	Logical OR complement. RR + $\overline{Data}$ → RR
7 0111	XNOR	Exclusive NOR. If RR = Data, RR → 1
8 1000	STO	Store. RR → Data Pin, Write → ⎍
9 1001	STOC	Store complement. $\overline{RR}$ → Data Pin, Write → ⎍
A 1010	IEN	Input enable. Data → IEN Register
B 1011	OEN	Output enable. Data → OEN Register
C 1100	JMP	Jump. JMP Flag → ⎍
D 1101	RTN	Return. RTN Flag → ⎍ and skip next instruction
E 1110	SKZ	Skip next instruction if RR = 0
F 1111	NOPF	No change in registers. RR → RR, Flag F → ⎍

FIGURE 2 – OUTLINE OF A TYPICAL ORGANIZATION FOR A MC14500B-BASED SYSTEM

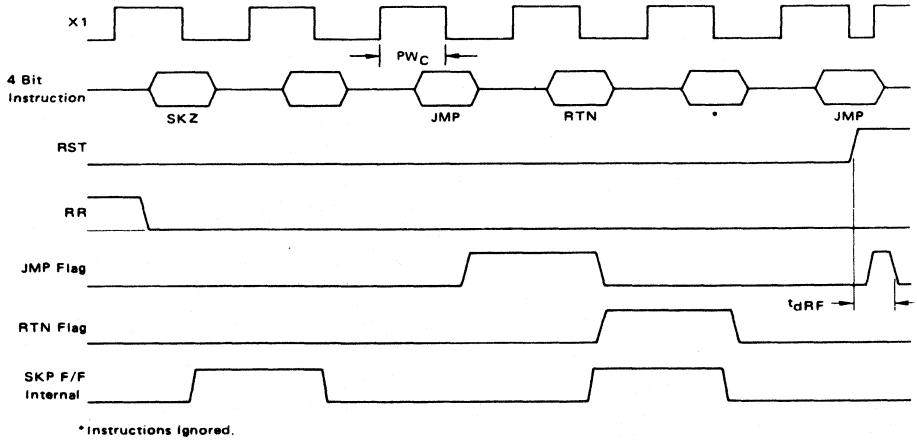


TIMING WAVEFORMS

Instructions NOPO, NOPF
RR, IEN, OEN remain unaffected



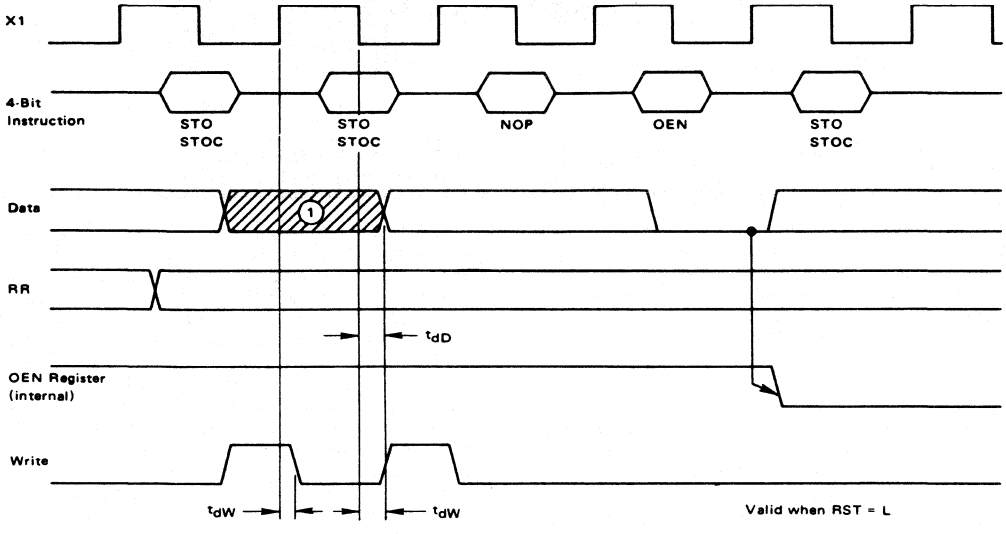
Instructions SKZ, JMP, RTN
RR, IEN, OEN remain unaffected



*Instructions Ignored.

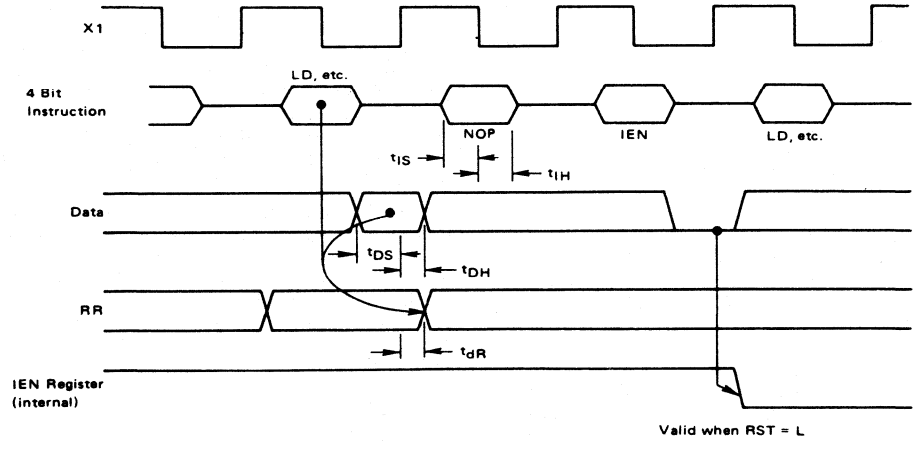
TIMING WAVEFORMS

Instructions STO, STOC, OEN



NOTE 1. Valid output data.

Instructions LD, LDC, AND, ANDC OR, ORC, XNOR, IEN



MOTOROLA

THE MC14500B INDUSTRIAL CONTROL UNIT

A 1-Bit CMOS Microprocessor

Many of the tasks performed with today's classical general-purpose multi-bit microprocessors, or with hardwired logic, are really one-bit tasks. For these, multi-bit processors (which must be programmed to look like a one-bit processor) represent a needless "overkill", while with hardwired logic, the versatility needed to adapt a circuit to a variety of system requirements is not readily available. To perform such tasks in the most efficient and economical manner, Motorola has developed the MC14500B *Industrial Control Unit.*

Applications

The MC14500B, Industrial Control Unit is a one-bit processor that operates over a full voltage and temperature range of CMOS JEDEC B-Series parts. It has as its peripheral components the whole CMOS family of over 100 parts. This allows tailoring a system to an application, and permits a judicious mix of customized hardware and software to be achieved. As a one-bit processor it can be applied in a multitude of systems such as:

- PLC (Low Cost) Programmable Logic Controllers
- Machine Controls
- Numerical Controllers
- Industrial Controls
- Traffic Controllers
- Copier Controllers
- Automatic Test Systems
- Telephone Dialing Systems
- Serial-Bit Stream Communications Systems
- Remote-Bit Stream Controllers
- Serial Data Processors
- Commercial Product Controllers
- Automotive Systems
- Microprogram Control Sequencer
- Peripheral Controllers, Printers, Keyboards, Discs, etc.
- MPU Companion for Unloading Overtaxed μPs

There are functions, however, for which one-bit machines are poorly suited. These functions normally include complex calculations or parallel-word data processing. On the other hand, when the task is decision-and-command oriented, a one-bit machine is an excellent choice. The tasks that are mixed between decisions and calculations will be decided upon by economics, the designer's familiarity with alternatives, and how comfortable the designer is with the alternatives. Under some circumstances, a combination of an MC6800 MPU and an MC14500B ICU may be the best solution.

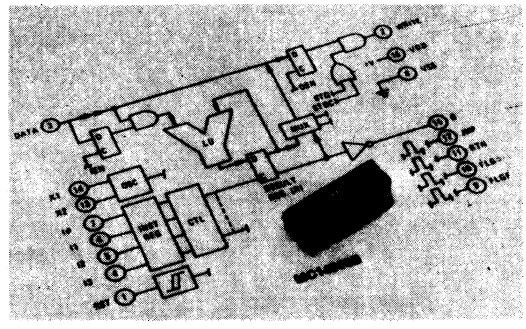


FIGURE 1 – BLOCK DIAGRAM

A 100+-page Handbook is available, detailing the operation and applications of the ICU

The Motorola MC14500B is a single chip, one-bit static CMOS processor optimized for decision-oriented tasks. Many of these decision-oriented tasks were well performed by relays, but with the MC14500B, electronic versatility can now be achieved in cost-effective systems. It is housed in 16-pin packages and features 16 four-bit instructions. The instructions perform logical operations on data appearing on a one-bit bidirectional data line and data in a one-bit accumulating Result Register within the ICU. All operations are performed at the bit level. The ICU has inherent CMOS qualities of high noise-immunity, microwatt power dissipation, the ability to use low-cost, low-current power supplies, 3 to 18 volt operating range, and the ease of battery backup and battery operation.

The ICU is timed by a single phase clock signal, generated by an internal oscillator that uses one external resistor.

Alternatively, it may be driven by an external source. In the external timing mode, the Clk signal is driven into pin 13 of the chip and the Clk signal is available as an output on pin 14 for synchronization with other systems.

In either case, the clock signal is available for synchronization with other systems. Each of the ICU's instructions execute in a single clock period. The clock frequency may be varied over a wide range. At a clock frequency of 1.0 MHz, more than 8300 instructions may be executed in a 60-Hz half-cycle.

The MC14500B instruction set consists of 16 instructions, as shown on page 4, each of which executes in one Clk period. The operating frequency range is from dc (single stepping) to a typical frequency of 1 MHz at 5 volts. The circuit's ability to execute instructions in one clock period, combined with its speed capabilities, allow it to outperform many of the more complex microprocessors for decision-oriented tasks.

Circuit Operation

The MC14500B processor operates synchronously with a single-phase clock. The clock divides the machine cycle into two periods. The first period (Clk High) is the "fetch" period, during which external memory (ROM) supplies the processor with an instruction. The instruction will be latched into the Instruction Register (IR) on the falling edge of the clock signal. The second period (Clk Low) is the "execu-

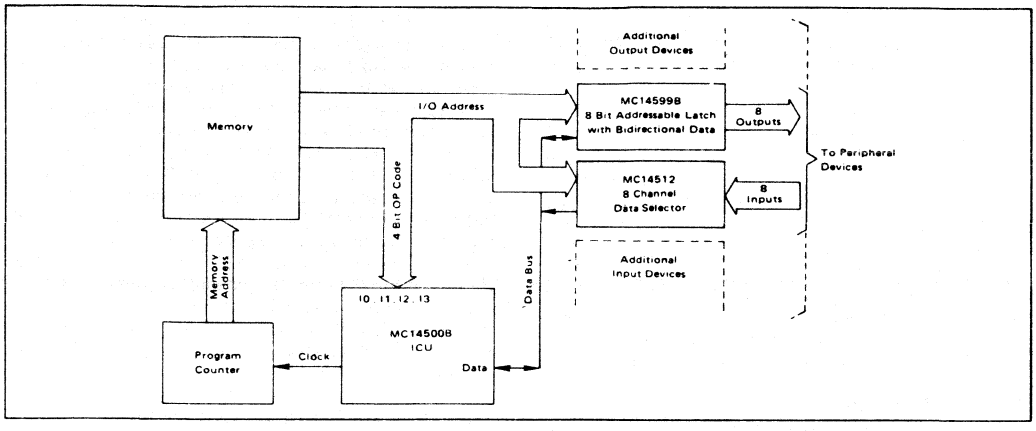


FIGURE 2 – OUTLINE OF A TYPICAL ORGANIZATION FOR A MC14500B-BASED SYSTEM

tion" period, during which the processor performs the command latched in its IR. During the execution period of an *output instruction*, the MC14500B puts the data in the Result Register (RR) onto the data line and raises the Write control line. The data must then be latched by the output circuitry at the end of the execution period of a machine cycle; i.e. on the rising edge of the (Clk) signal.

During the execution period of *input or logical commands*, the data present on the data line at the time the Clk signal is low is accepted by the ICU; the logic operation is then performed and the result is clocked into the Result Register on the rising edge of the clock (Clk) signal. This "split cycle" architecture makes it possible to "interleave" instructions and operands addressed in memory, the advantage being that the address lines of the I/O ports can share the ICU's instruction lines, forming a common bus.

The System

Figure 2 shows a block diagram of a minimal MC14500B system configured with standard B series logic parts. The blocks of the system are:

The MC14500B — which serves as the central controller of the system;

The ROM — which holds the instructions and the operand addresses;

The program counter — used to step the machine through the sequence of instructions stored in memory;

The input selector (demultiplexer) — used to route the addressed input to the MC14500B's one-bit bidirectional data bus;

The output latches — which receive data placed on the 1-bit bidirectional bus by the MC14500B when an output instruction is executed.

The system can easily be expanded in terms of I/O, so long as the memory is sufficiently wide to address the I/O structure.

A CMOS PLC

MC14500B Industrial Control Unit is essentially the monolithic embodiment of the Programmable Logic Controllers (PLC) central architecture. Its Logic Unit (LU) is capable of doing a number of different Boolean functions under the control of instructions latched into the Instruction Register. The LU has two inputs: Data from the "Outside World" and the output of a Result Register.

The output of the LU is latched into the Result Register, where the new result will serve as one of the LU inputs on the next instruction. In general:

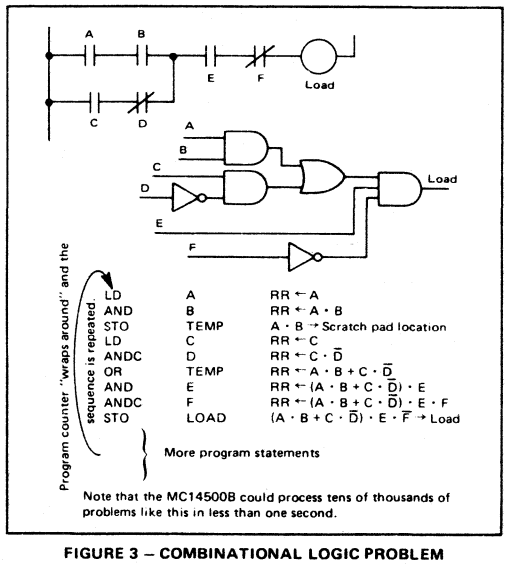


FIGURE 3 – COMBINATIONAL LOGIC PROBLEM

$$RR_{new} = f(data, RR_{old}).$$

After calculating a Boolean equation, the state of the Result Register or its complement is stored in an output latch to activate or deactivate load devices.

Looping Control Structure

A typical problem example is illustrated in Figure 3. The solution shown illustrates a looping control structure. Here, it is seen that the MC14500B continuously samples its inputs and, based on that information, transfers a logic signal to activate or deactivate the load device.

The ICU does not alter the sequential operation of the program counter. It sequentially fetches instructions from memory and when the program counter reaches its highest value it "wraps around," and the entire program is repeated, effecting what is known as a "looping control structure."

Time Invariant Software

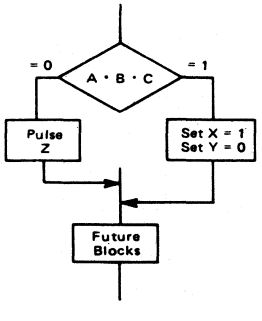
What is important beyond the concept of a "looping control structure" is the fact that the ICU can effect a "conditional jump" without parallel loading the program counter with the Jump address. The concept is that code requiring conditional branching can be written even though all code is fetched from memory sequentially. To accomplish this an output mask is used in the MC14500B. The Output Enabling Register — OEN, may be set or reset with the result of an evaluated logic equation. If the mask is reset, the Write strobe is inhibited and no change can be made in the state of the output devices. The processor can therefore use its logic power to turn off or on whole blocks of instructions. This leads to a looping control structure in which the same sequence of commands are encountered with certain blocks of code being selectively enabled or disabled.

Figure 4 shows a flow chart and program solution of a typical problem. In this example, the ICU uses its OEN mask to effect a "pseudo" branch. It first resolves the question asked in the decisions block (statements 1, 2, and 3), then stores the result (1 = true, 0 = false) in a temporary storage location called (Flag). Next, the ICU loads its output mask with the result of the decision block (statement 5); if the OEN mask was set true, the following two output instructions (statement 6 and 7) would be "active" and the right branch of the decision block would be executed. If the mask was loaded with a zero, statements 6 and 8 would not change the state of the X and Y outputs effecting the Pseudo branch. If the mask was indeed loaded with a zero, this would mean the left branch of the decision block is to be executed. To do this the ICU loads the logical complement of Flag into the output mask (statements 8 and 9). Then if Flag was originally false (i.e. $A \cdot B \cdot C = 0$), the instructions following statement 9 will be active and statement 10 and 11 will create the pulse to be sent to output Z. Statements 12 and 13 force the output mask to the 1 state so that future blocks of code will not inadvertently be turned off.

One of the disadvantages of the conventional conditional jumps as a means of making decisions is that the execution time of the program varies with the state of the input signals. A looping program is inherently time invariant. This allows for synchronization with other systems and other advantages such as stable sync pulses for trouble shooting. In the real-world environment of training people and maintaining products, this can be worth a great deal.

Adding a Conventional Control Structure

In some control applications it may be advantageous to have a control structure like that of a conventional processor, rather than a looping control structure. This situation may occur when the system becomes large and timing is critical. Having the capability to call subroutines also helps to modularize the software. An MC14500B system can be easily modified to incorporate a jumping, conditional branching, and subroutine capability.



	1		LD	A	*RR ← A
	2		AND	B	RR ← A • B
	3		AND	C	RR ← A • B • C
	4		STO	Flag	Set Flag = 1 If Right Branch Taken
5	Right branch		OEN	RR	Output Mask ← (A • B • C) = RR
6	block of		STO	X	1 → X
7	code		STOC	Y	0 → Y
8			LDC	Flag	If Flag was 0 the following code will be active — RR ← Flag, OEN ← Flag
9	Left branch		OEN	RR	
10	block of		STO	Z	Creates Pulse
11	code		STOC	Z	
12			ORC	RR	RR ← 1 = RR + $\overline{RR}$ — Enable Future
13			OEN	RR	Output Mask ← 1 — blocks of code

*RR = Result Register

FIGURE 4 – BLOCK ENABLING STRUCTURE

The MC14500B has three program control instructions which are intended for the purpose of adding conventional jumping, conditional branching, and sub-routine capabilities to an ICU system. These instructions condition the ICU to take the appropriate action and provide the necessary control signals to external logic circuits which actually perform the address modifications.

For More Information . . .

Comprehensive information on these and other topics is contained in the MC14500 ICU Applications Handbook. The book approaches each topic from basic fundamentals and builds on these toward more advanced levels. It is richly illustrated with many examples and includes system schematics and programming examples. Some topics covered are:

1. Basics Concepts
2. Basic Programming and Instruction Set
3. Hardware Systems
4. A Demonstration System
5. Signal conditioning and I/O Interface Circuits
6. Advanced Programming
7. A Comprehensive Example
8. Advanced Hardware Systems
9. Useful Routines

MC14500B OPERATIONS AND CODES

Instruction Code		Mnemonic	Action
Hex	Binary		
0	0000	NOPO	**No Operation Zero** No change in any register. The Flag O output pin gets a one-period pulse beginning on the falling edge of X1 (Clk).
1	0001	LD	**Load** The Result Register is loaded with the state of the data signal.
2	0010	LDC	**Load Complemented** The Result Register is loaded with the complement of the data signal.
3	0011	AND	**AND** The Result Register is loaded with the logical AND of the data signal and the old Result Register state.
4	0100	ANDC	**AND Complement** The Result Register is loaded with the logical AND of the old Result Register state and the complement of the data signal.
5	0101	OR	**OR** The Result Register is loaded with the logical OR of the data signal and the old Result Register state.
6	0110	ORC	**OR Complement** The Result Register is loaded with logical OR of the old Result Register state and the complement of the data signal.
7	0111	XNOR	**Exclusive NOR** The Result Register is loaded with a logical 1 if the old Result Register state and the data signal agree. If the old result and the data signal are not alike, the Result Register is loaded with a logical 0.
8	1000	STO	**Store** The Data pin is driven to the state of the Result Register. The Write pin is driven high for a half period beginning with the fall of X1. The state of the Result Register is not changed.
9	1001	STOC	**Store Complemented** Same as Store, except the data signal is driven with the complement of the Result Register.
A	1010	IEN	**Input Enable** The Input Enable Register is loaded with the state of the data signal.
B	1011	OEN	**Output Enable** The Output Enable Register is loaded with the state of the data signal.
C	1100	JMP	**Jump** A one period pulse is generated at the JMP pin beginning with the falling edge of X1.
D	1101	RTN	**Return** A one period pulse is generated at the RTN pin beginning with the falling edge of X1, and the next instruction is ignored.
E	1110	SKZ	**Skip If Zero** If the Result Register contains a logical 0 at the time of the instruction the next instruction is ignored.
F	1111	NOPF	**No Operation F** No change in any register. Flag F output pin gets a one period pulse beginning on the falling edge of X1 (Clk).

MOTOROLA

MC14 1000
MC14 1200

Product Preview

ONE CHIP MICROCOMPUTER

The MC14 1000 and the MC14 1200 are two members of the MC14 1000 family of CMOS 4-bit microcomputers. They incorporate ROM, RAM, ALU, control, and I/O in a single CMOS monolithic structure. The MC14 1000/2000 can be tailored to its application by internally programmed ROM and output PLA.

The MC14 1000 family is source program compatible, pin-out compatible, and architecturally similar to the PMOS TMS1000 family. These CMOS one-chip microcomputers offer the following additional features not available in the TMS1000:

- Low Power Consumption Suited for Battery-powered or Battery Back-up Systems
- Fully Static Operation
- TTL Compatible – Drives One TTL Load or Four LSTTL Loads
- Clock Frequency to 600 kHz at V_{DD} = 4.75 V
- Single Supply, 3 to 6 Volt Operation
- 16 "R" Outputs (MC14 1200)

FEATURES:

	MC14 1000	MC14 1200
Package Pin Count	28 Pins	40 Pins
Instruction Read Only Memory	1024 X 8 Bits (8,192 Bits)	
Data Random Access Memory	64 X 4 Bits (256 Bits)	
"R" Individually Addressed Output Latches	11	16
"O" Parallel Latched Data Outputs	8 Bits	
Maximum-Rated Voltage	6.5 V	
Working Registers	Static 2-4 Bits Each	
Instruction Set	See Table 1	
On-Chip Oscillator	Yes	
5 V Power Supply/Typical Dissipation	5 V/2.5 mW	
3 V Power Supply/Typical Dissipation	3 V/500 μW	

APPLICATIONS:

- Appliance Controllers
- Calculators
- Toys
- Radio Controllers
- Communications Controllers
- Data Terminals
- Cash Registers
- Heating/Air-Conditioning Controllers
- Remote Sensing System
- Printing Controllers
- Security Systems
- Power Systems Control
- Automotive Control

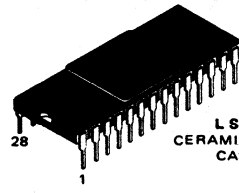
The above applications of the MC14 1000 family demonstrate its wide potential. Motorola will accept customer programs or will contract complete program development given the specifications for the application. Customer hardware and software support is available for developing programs and debugging systems. This consists of one board and a software package using the M6800 EXORciser. Contact your local sales office for details on the support equipment and software.

This is advance information and specifications are subject to change without notice.

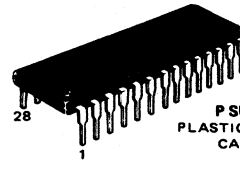
CMOS LSI

(LOW-POWER COMPLEMENTARY MOS)

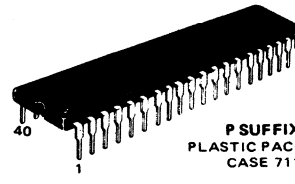
ONE CHIP MICROCOMPUTER



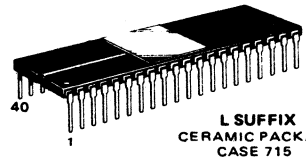
L SUFFIX
CERAMIC PACKAGE
CASE 719



P SUFFIX
PLASTIC PACKAGE
CASE 710



P SUFFIX
PLASTIC PACKAGE
CASE 711



L SUFFIX
CERAMIC PACKAGE
CASE 715

ORDERING INFORMATION

MC14XXXX — Suffix Denotes
- L Ceramic Package
- P Plastic Package

MAXIMUM RATINGS (Voltages referenced to V_{SS})

Rating	Symbol	Value	Unit
DC Supply Voltage	V_{DD}	-0.5 to +6.5	Vdc
Input Voltage, All Inputs	V_{in}	-0.5 to V_{DD} + 0.5	Vdc
DC Current Drain per Pin, All Inputs	I	10	mAdc
DC Current Drain, V_{DD} Pin	I	250	mAdc
DC Current Drain, V_{SS} Pin	I	20	mAdc
Operating Temperature Range	T_A	-40 to +85	°C
Storage Temperature Range	T_{stg}	-65 to +150	°C
Total Power Dissipation @ 25°C MC14 1000 MC14 1200	P_D	400 600	mW

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in} \text{ or } V_{out}) \leq V_{DD}$.

RECOMMENDED OPERATING CONDITIONS (V_{SS} = 0)

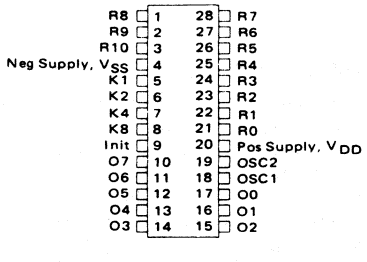
Parameter	Symbol	Value	Unit
DC Supply Voltage — High Speed Clock Full Range Operation	V_{DD}	+4.75 to +6.0 +3.0 to +6.0	Vdc
Clock Frequency — V_{DD} = 5.0 Vdc ± 5% V_{DD} = 3.0 Vdc Min.	f_{Clk}	DC to 600 DC to 200	kHz

ELECTRICAL CHARACTERISTICS (V_{DD} = +5.0 V, V_{SS} Gnd, T_A = 25°C unless otherwise noted)

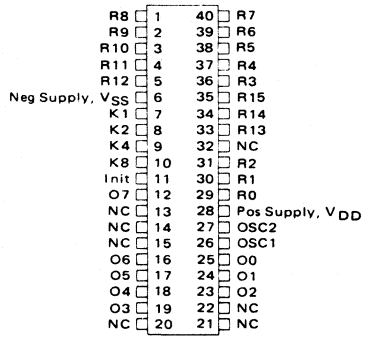
Characteristic	Symbol	Min	Typ	Max	Unit
Input Current — K Inputs (V_{in} = 5.0 V) (V_{in} = 0.0 V)	I_{in}	 75 —	 100 -0.00001	 135 -0.3	µAdc
Output Drive — R and O Outputs (V_{OH} = 2.4 V) (V_{OL} = 0.4 V, T_A = 85°C, V_{DD} = 4.75 V)	 I_{OH} I_{OL}	 -20 1.6	 — —	 — —	mAdc
Average Supply Current (f_{Clk} = 500 kHz)	I_{DD}	—	—	1500	µAdc
Static Supply Current (V_{DD} = 6 V)	I_{DD}	—	60	300	µAdc
Oscillator Frequency (V_{DD} = 4.75 V)	f_{Clk}	No Limit	—	600	kHz
Internal Oscillator Frequency for R_{ext} = 30 kΩ	f_{Clk}	400	500	600	kHz
Input Capacitance — K Inputs	C_{in}	—	—	7.5	pF
Input Capacitance — Clock Input	C_{io}	—	—	30	pF

PIN ASSIGNMENTS

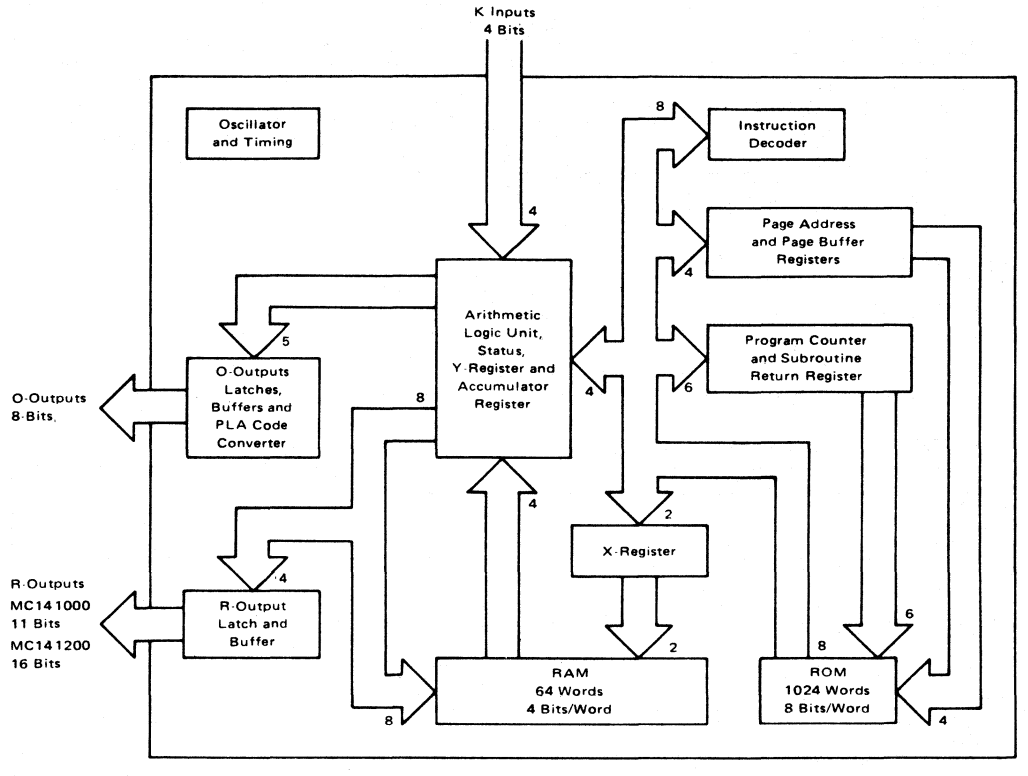
MC14 1000



MC14 1200



LOGIC BLOCK DIAGRAM



The MC141000/1200 ROM program controls data input, storage, processing, and output. The processing of data occurs in the arithmetic logic unit (ALU). K input data enters the ALU and is stored in the 4-bit accumulator. The accumulator output accesses the output latches, the RAM storage cells, and the adder input. Data is stored in the 256-bit RAM, organized into 64 words, 4 bits per word. The 4-bit words are grouped into four 16-word files addressed by a 2-bit X register. A 4-bit Y register addresses one of the 16 words in a file by control signals from the ROM.

The 43 basic instructions handle I/O, constant data from the ROM, bit control, internal data transfer, arithmetic processsing, logic comparison, conditional and nonconditional branching and subroutines.

The designer has access for programming the following functions:

1) ROM – 1024 words of 8 bits
2) Program Logic Array for O outputs
3) Output circuits

PROGRAMMABLE ROM

The 1024 words of the 8-bit ROM are divided into 16 pages of instructions with 64 instructions on each page. The program starts at the top of the sixteenth page. A binary program counter sequentially addresses each ROM instruction on a page. One level of subroutine return address is stored. The page address register (4 bits) stores the current address for one of the 16 ROM pages. To change pages, a constant from the ROM loads into the page buffer register (4 bits), and upon a branch or call, the page buffer loads into the page address register.

RAM

The 64-word by 4-bit RAM comprises 4 files, each file containing 16 four-bit words. The RAM is addressed by the Y register and X register. The Y register selects one of the 16 words in a file and the X register selects one of four files. Any 4-bit word can be read or written. Any selected bit in the RAM can be set, reset, or tested.

INPUT

The 4 data inputs are designated K1, K2, K4, and K8. The R outputs can multiplex inputs for keyboard use. The R outputs can be used for handshake for control ot the input from other devices.

The K inputs are static-protected CMOS inputs with pulldown of about 50 kohms. Thus, an open input is equivalent to logic 0. The circuit is shown in Figure 1.

OUTPUT

The O-outputs comprise the output bus. The R-outputs are used as control lines to scan keyboards and displays, perform handshakes, and interface external logic. The 8 parallel O-outputs are decoded from the 5 bits in the O-output latches. This decoding is defined by the customer and is accomplished in the O-PLA. The 11 R-outputs of the MC14100 and the 16 R-outputs of the MC141200 are individually settable and resettable under program control.

The outputs may be programmed in one of two configurations, either open emitter or active sink NMOS as shown in Figures 2 and 3. The circuits of Figures 2 and 3 sources 20 mA @ V_O = 2.0 Volts and V_{DD} = 5.0 Volts. The circuit of Figure 3 sinks 1.6 mA over temperature to operate one TTL load or four LSTTL loads.

INTERNAL OR EXTERNAL CLOCK

The internal oscillator is controlled by the value of one resistor. This is an improvement over the PMOS TMS1000/1200. For external clocking, the clock signal is connected to Osc. 1. The oscillator circuit works with quartz crystals, ceramic resonators and L-C resonant circuits. Figure 4 shows the typical operation of the oscillator as a function of the programming resistor.

FIGURE 1 – INPUT CIRCUIT WITH PULLDOWN AND STATIC PROTECTION

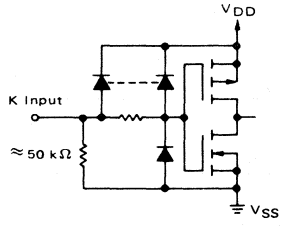


FIGURE 2 – OPEN EMITTER OUTPUT CIRCUIT

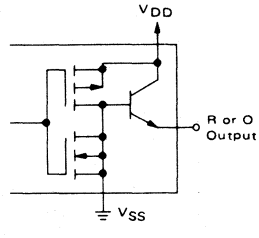
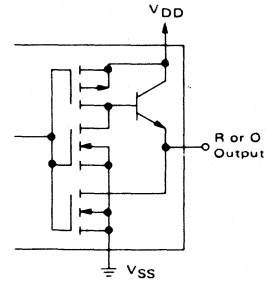


FIGURE 3 – ACTIVE NMOS OUTPUT CIRCUIT



CONNECTION FOR INTERNAL OSCILLATOR

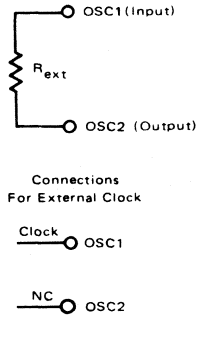


FIGURE 4 – TYPICAL OSCILLATOR FREQUENCY vs EXTERNAL RESISTANCE

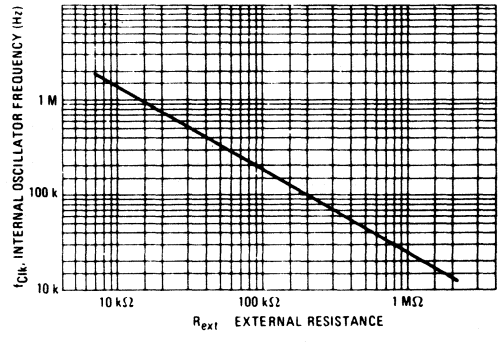


TABLE 1 – Standard Instruction Set

Function	Mnemonic	Status Effects		Description
		CARRY	COMPARED	
Register to Register	TAY			Transfer accumulator to Y register.
	TYA			Transfer Y register to accumulator.
	CLA			Clear accumulator.
Transfer Register to Memory	TAM			Transfer accumulator to memory.
	TAMIY			Transfer accumulator to memory and increment Y register.
	TAMZA			Transfer accumulator to memory and zero accumulator.
Memory to Register	TMY			Transfer memory to Y register.
	TMA			Transfer memory to accumulator.
	XMA			Exchange memory and accumulator.
Arithmetic	AMAAC	√		Add memory to accumulator, results to accumulator. If carry, one to status.
	SAMAN	√		Subtract accumulator from memory, results to accumulator. If no borrow, one to status.
	IMAC	√		Increment memory and load into accumulator. If carry, one to status.
	DMAN	√		Decrement memory and load into accumulator. If no borrow, one to status.
	IA			Increment accumulator, no status effect.
	IYC	√		Increment Y register. If carry, one to status.
	DAN	√		Decrement accumulator. If no borrow, one to status.
	DYN	√		Decrement Y register. If no borrow, one to status.
	A8AAC	√		Add 8 to accumulator, results to accumulator. If carry, one to status.
	A10AAC	√		Add 10 to accumulator, results to accumulator. If carry, one to status.
	A6AAC	√		Add 6 to accumulator, results to accumulator. If carry, one to status.
	CPAIZ	√		Complement accumulator and increment. If then zero, one to status.
Arithmetic Compare	ALEM	√		If accumulator less than or equal to memory, one to status.
	ALEC	√		If accumulator less than or equal to a constant, one to status.
Logical Compare	MNEZ		√	If memory not equal to zero, one to status.
	YNEA		√	If Y register not equal to accumulator, one to status and status latch.
	YNEC		√	If Y register not equal to a constant, one to status.
Bits in Memory	SBIT			Set memory bit.
	RBIT			Reset memory bit.
	TBIT1		√	Test memory bit. If equal to one, one to status.
Constants	TCY			Transfer constant to Y register.
	TCMIY			Transfer constant to memory and increment Y.
Input	KNEZ		√	If K inputs not equal to zero, one to status.
	TKA			Transfer K inputs to accumulator.
Output	SETR			Set R output addressed by Y.
	RSTR			Reset R output addressed by Y.
	TDO			Transfer data from accumulator and status latch to O outputs.
	CLO			Clear O-output register.
RAM 'X' Addressing	LDX			Load 'X' with a constant.
	COMX			Complement 'X'.
ROM Addressing	BR			Branch on status = one.
	CALL			Call subroutine on status = one.
	RETN			Return from subroutine.
	LDP			Load page buffer with constant.

NOTE: If the bits COMPARED are not equal, or if there is a CARRY from the MSB of the adder, status will stay at one. Otherwise, status will go to a zero for one instruction cycle. Branch and Call can only execute successfully when status is a one. The check marks (√) indicate the instructions that affect the status.

chapter 5

bipolar 4-bit slice MPU families

MOTOROLA

MC2901A

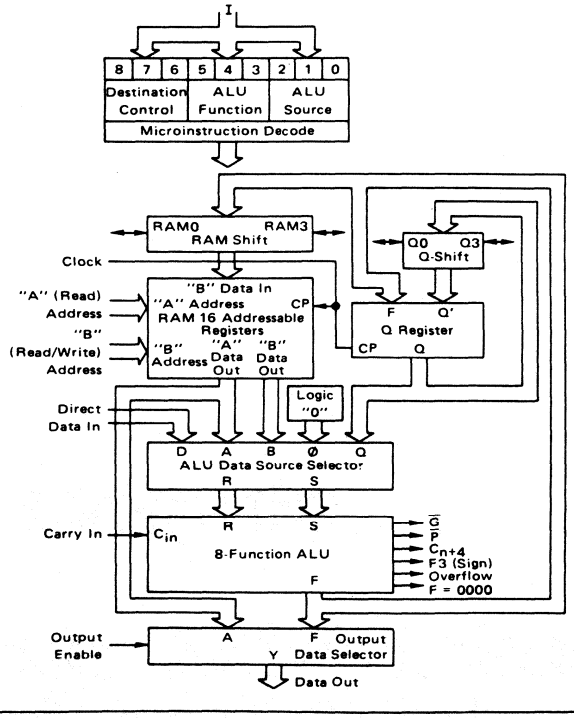
FOUR-BIT BIPOLAR MICROPROCESSOR SLICE

The four-bit bipolar microprocessor slice is designed, as a high-speed cascadable element intended for use in CPUs, peripheral controllers, programmable microprocessors, and numerous other applications. The microinstruction flexibility of the MC2901A will allow efficient emulation of almost any digital computing machine.

The device, as shown in the block diagram below, consists of a sixteen-word by four-bit, two-port RAM; a high-speed ALU; and, the associated shifting, decoding, and multiplexing circuitry. The 9-bit microinstruction word is organized into three groups of three bits each and selects the ALU source operands, the ALU function, and the ALU destination register. The microprocessor is cascadable with full look-ahead or with ripple carry, has three-state outputs, and provides various status flag outputs from the ALU. Advanced low-power Schottky processing is used to fabricate this 40-lead LSI chip.

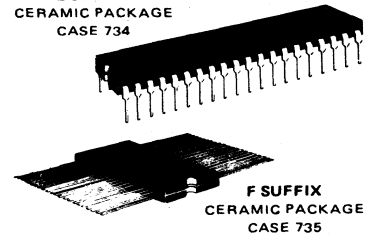
- Plug-in Replacement for MC2901
- 20% to 30% Faster Than MC2901 in Most System Configurations
- Major Speed Improvements in D Input and Carry Paths
- I_{OL} Raised to 20 mA on Y Outputs – 30% More Drive Than MC2901
- I_{CC} Reduced to 190 mA at 125°C – 30% Less Than MC2901
- V_{IL} Raised to 0.8 V Over Full Military Range for Increased Noise Immunity

MICROPROCESSOR SLICE BLOCK DIAGRAM



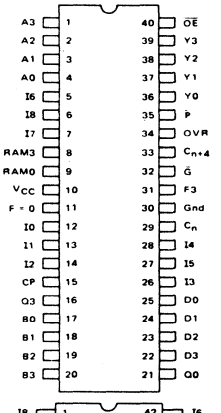
TTL
FOUR-BIT BIPOLAR MICROPROCESSOR SLICE

L SUFFIX
CERAMIC PACKAGE
CASE 734

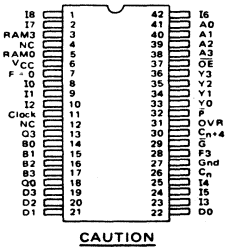


F SUFFIX
CERAMIC PACKAGE
CASE 735

DIP



FLAT PACK



CAUTION

MC2901AFM pinout differs from MC2901FM on pins 4, 11, 12, and 13.

ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Hermetic DIP	0°C to +70°C	MC2901ALC
Hermetic DIP	-55°C to +125°C	MC2901ALM
Hermetic Flat Pack	-55°C to +125°C	MC2901AFM

MAXIMUM RATINGS (above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential	−0.5 V to +7.0 V
DC Voltage Applied to Outputs for High Output State	−0.5 V to +V_{CC} Max
DC Input Voltage	−0.5 V to +5.5 V
DC Output Current, Into Outputs	30 mA
DC Input Current	−30 mA to +5.0 mA

ELECTRICAL CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise noted)

MC2901AXC — T_A = 0°C to +70°C, V_{CC} = 5.0 V ± 5% (Commercial), Min = 4.75 V, Max = 5.25 V

MC2901AXM — T_A = −55°C to +125°C, V_{CC} = 5.0 V ± 10% (Military), Min = 4.50 V, Max = 5.50 V

Parameters	Description	Test Conditions (Note 1)		Min	Typ (Note 2)	Max	Unit
V_{OH}	Output High Voltage	V_{CC} = Min, V_{IN} = V_{IH} or V_{IL}	I_{OH} = −1.6 mA Y0, Y1, Y2, Y3	2.4			Volt
			I_{OH} = −1.0 mA, C_{n+4}	2.4			
			I_{OH} = −800 µA, OVR, $\overline{P}$	2.4			
			I_{OH} = −600 µA, F3	2.4			
			I_{OH} = −600 µA RAM0, 3, Q0, 3	2.4			
			I_{OH} = −1.6 mA, $\overline{G}$	2.4			
I_{CEX}	Output Lekage Current for F = 0 Output	V_{CC} = Min, V_{OH} = 5.5 V, V_{IN} = V_{IH} or V_{IL}				250	µA
V_{OL}	Output Low Voltage	V_{CC} = Min, V_{IN} = V_{IH} or V_{IL}	I_{OL} = 20 mA (Commercial) I_{OL} = 16 mA (Military) Y0, Y1, Y2, Y3			0.5	Volt
			I_{OL} = 16 mA, $\overline{G}$, F = 0			0.5	
			I_{OL} = 10 mA, C_{n+4}			0.5	
			I_{OL} = 8.0 mA, OVR, $\overline{P}$			0.5	
			I_{OL} = 6.0 mA, F3 RAM0, 3, Q0, 3			0.5	
V_{IH}	Input High Level	Guaranteed input logical High voltage for all inputs (Note 6)		2.0			Volt
V_{IL}	Input Low Level	Guaranteed input logical Low voltage for all inputs (Note 6)				0.8	Volt
V_I	Input Clamp Voltage	V_{CC} = Min, I_{IN} = −18 mA				−1.5	Volt
I_{IL}	Input Low Current	V_{CC} = Max, V_{IN} = 0.5 V	Clock, $\overline{OE}$			−0.36	mA
			A0, A1, A2, A3			−0.36	
			B0, B1, B2, B3			−0.36	
			D0, D1, D2, D3			−0.72	
			I0, I1, I2, I6, I8			−0.36	
			I3, I4, I5, I7			−0.72	
			RAM0, 3, Q0, 3 (Note 4)			−0.8	
			C_n			−3.6	
I_{IH}	Input High Current	V_{CC} = Max, V_{IN} = 2.7 V	Clock, $\overline{OE}$			20	µA
			A0, A1, A2, A3			20	
			B0, B1, B2, B3			20	
			D0, D1, D2, D3			40	
			I0, I1, I2, I6, I8			20	
			I3, I4, I5, I7			40	
			RAM0, 3, Q0, 3 (Note 4)			100	
			C_n			200	
I_I	Input High Current	V_{CC} = Max, V_{IN} = 5.5 V				1.0	mA
I_{OZH} I_{OZL}	Off State (High Impedance) Output Current	V_{CC} = Max	Y0, Y1, Y2, Y3 — V_O = 2.4 V			50	µA
			Y0, Y1, Y2, Y3 — V_O = 0.5 V			−50	
			RAM0, 3, Q0, 3 — V_O = 2.4 V (Note 4)			100	
			RAM0, 3, Q0, 3 — V_O = 0.5 V (Note 4)			−800	
I_{OS}	Output Short Circuit Current (Note 3)	V_{CC} = Max +0.5 V, V_O = 0.5 V	Y0, Y1, Y2, Y3, $\overline{G}$	−30		−85	mA
			C_{n+4}	−30		−85	
			OVR, $\overline{P}$	−30		−85	
			F3	−30		−85	
			RAM0, 3, Q0, 3	−30		−85	
I_{CC}	Power Supply Current (Note 5)	V_{CC} = Max	Commercial		160	265	mA
			Military		160	280	

NOTES:

1. For conditions shown as Min or Max, use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0 V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
4. These are three-state outputs internally connected to TTL inputs. Input characteristics are measured with I678 in a state such that the three-state output is Off.
5. Worst case I_{CC} is at minimum temperature.
6. These input levels provide zero noise immunity and should only be tested in a static, noise-free environment.

SWITCHING CHARACTERISTICS OVER OPERATING RANGE

Tables 1, 2, and 3 define the timing characteristics of the MC2901A over the operating voltage and temperature range. The tables are divided into three types of parameters: clock characteristics, combinational delays from inputs to outputs, and setup and hold time requirements. The later table defines the time prior to the end of the cycle (i.e., clock low-to-high transition) that each input must be stable to guarantee that the correct data is written into one of the internal registers.

Measurements are made at 1.5 V with V_{IL} = 0 V and V_{IH} = 3.0 V. For three-state disable tests, C_L = 5.0 pF and measurement is to 0.5 V change on output voltage level. Input rise and fall times are 1ns/V. All outputs are fully loaded.

TABLE 1 -- CYCLE TIME AND CLOCK CHARACTERISTICS

Time	Commercial	Military
Read-Modify-Write Cycle (time from selection of A, B registers to end of cycle)	100 ns	110 ns
Maximum Clock Frequency to Shift Q Register (50% duty cycle) I = 432 or 632	15 MHz	12 MHz
Minimum Clock Low Time	30 ns	30 ns
Minimum Clock High Time	30 ns	30 ns
Minimum Clock Period	100 ns	110 ns

TABLE 2 – COMBINATIONAL PROPAGATION DELAYS

(All in ns, C_L = 50 pF except output disable tests)

	To Output															
	Commercial								Military							
From Input	Y	F3	C_{n+4}	$\overline{G}, \overline{P}$	F = 0 R_L = 470	OVR	Shift Outputs RAM0 RAM3	Shift Outputs Q0 Q3	Y	F3	C_{n+4}	$\overline{G}, \overline{P}$	F = 0 R_L = 470	OVR	Shift Outputs RAM0 RAM3	Shift Outputs Q0 Q3
A, B	80	80	75	65	90	85	95	–	85	85	80	70	100	90	100	–
D (Arithmetic Mode)	45	45	45	35	60	55	65	–	50	50	50	40	65	60	70	–
D (I = X37)(Note 5)	40	40	–	–	55	–	60	–	45	45	–	–	60	–	65	–
C_n	30	35	20	–	50	30	50	–	35	35	25	–	55	35	55	–
I012	55	55	50	45	70	65	75	–	60	60	55	50	75	70	80	–
I345	55	55	55	50	70	65	75	–	60	60	60	55	75	70	80	–
I678	30	–	–	–	–	–	30	30	35	–	–	–	–	–	35	35
$\overline{OE}$ Enable/Disable	35/25	–	–	–	–	–	–	–	40/25	–	–	–	–	–	–	–
A Bypassing ALU (I = 2xx)	45	–	–	–	–	–	–	–	50	–	–	–	–	–	–	–
Clock ↓ (Note 6)	60	60	60	50	75	70	80	30	65	65	65	55	85	75	85	35

TABLE 3 – SET UP AND HOLD TIMES

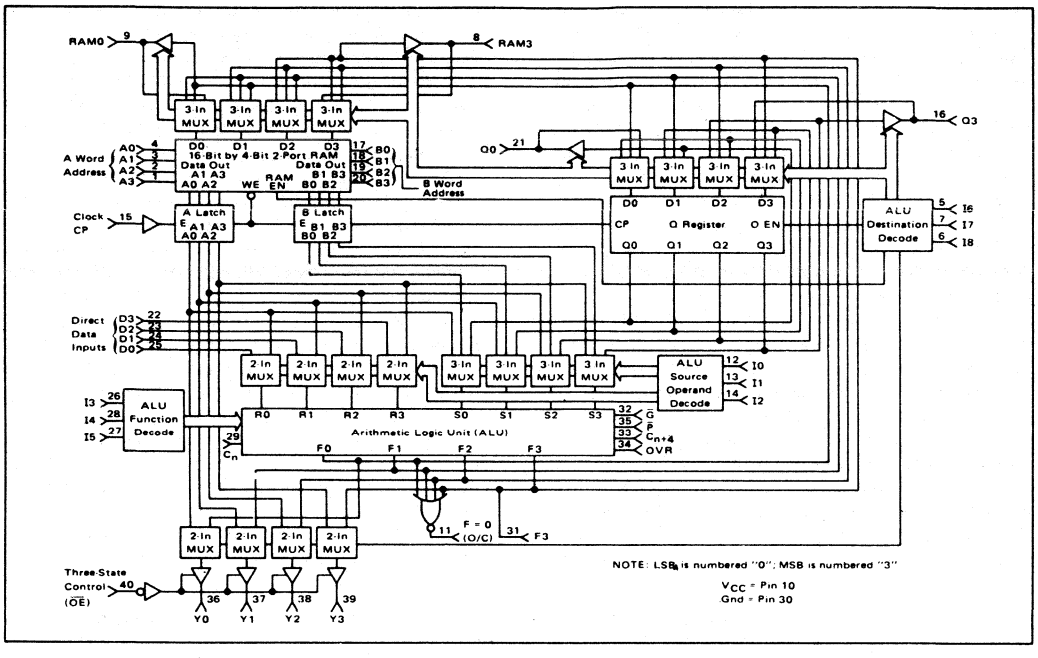
(All in ns. See Note 1)

From Input	Notes	Commercial		Military	
		Setup Time	Hold Time	Setup Time	Hold Time
A, B	2, 4	100	0	110	0
Source	3, 5	$t_{pw}L$ + 30		$t_{pw}L$ + 30	
B Destination	2, 4	$t_{pw}L$ + 15	0	$t_{pw}L$ + 15	0
D (Arithmetic Mode)		70	0	75	0
D (I = X37)(Note 5)		60	0	65	0
C_n		55	0	60	0
I012		80	0	85	0
I345		80	0	85	0
I678	4	$t_{pw}L$ + 30	0	$t_{pw}L$ + 30	0
RAM0, 3, Q0, 3		25	0	25	0

NOTES: 1. See Figure 11. All times relative to clock low-to-high transition.

2. If the B address is used as a source operand, allow for the A, B source setup time. If it is used only for the destination address, use the B Destination setup time.
3. Where two numbers are shown, both must be met.
4. $t_{pw}L$ is the clock low time.
5. D ∨ 0 is the fastest way to load the RAM from the D inputs. This function is obtained with I = 337.
6. Using Q register as source operand in arithmetic mode. Clock is not normally in critical speed path when Q is not a source.

FIGURE 1 – DETAILED MC2901A MICROPROCESSOR BLOCK DIAGRAM



ARCHITECTURE

A detailed block diagram of the bipolar microprogrammable microprocessor structure is shown in Figure 1. The circuit is a four-bit slice cascadable to any number of bits. Therefore, all data paths within the circuit are four bits wide. The two key elements in the Figure 1 block diagram are the 16-word by 4-bit 2-port RAM and the high-speed ALU.

Data in any of the 16 words of the Random Access Memory (RAM) can be read from the A-port of the RAM as controlled by the 4-bit A address field input. Likewise, data in any of the 16 words of the RAM, as defined by the B address field input, can be simultaneously read from the B-port of the RAM. The same code can be applied to the A select field and B select field; in which case, the identical file data will appear at both the RAM A-port and B-port outputs simultaneously.

When enabled by the RAM write enable (RAM-EN), new data is always written into the file (word) defined by the B address field of the RAM. The RAM data input field is driven by a 3-input multiplexer. This configuration is used to shift the ALU output data (F) if desired. This three-input multiplexer scheme allows the data to be shifted up one bit position, shifted down one bit position, or not shifted in either direction.

The RAM A-port data outputs and RAM B-port data outputs drive separate 4-bit latches. These latches hold the RAM data while the clock input is low. This eliminates any possible race conditions that could occur while new data is being written into the RAM.

The high-speed Arithmetic Logic Unit (ALU) can perform three binary arithmetic and five logic operations on the two 4-bit input words, R and S. The R input field is driven from a 2-input multiplexer, while the S input field is driven from a 3-input multiplexer. Both multiplexers also have an inhibit capability; that is, no data is passed. This is equivalent to a "zero" source operand.

Referring to Figure 1, the ALU R-input multiplexer has the RAM A-port and the direct data inputs (D) connected as inputs. Likewise, the ALU S-input multiplexer has the RAM A-port, the RAM B-port and the Q register connected as inputs.

This multiplexer scheme gives the capability of selecting various pairs of the A, B, D, Q, and "0" inputs as source operands to the ALU. These five inputs, when taken two at a time, result in ten possible combinations of source operand pairs. These combinations include AB, AD, AQ, A0, BD, BQ, B0, DQ, D0, and Q0. It is apparent that AD, AQ, and A0 are somewhat redundant with BD,

BQ, and B0 in that if the A address and B address are the same, the identical function results. Thus, there are only seven completely nonredundant source operand pairs for the ALU. The MC2901 microprocessor implements eight of these pairs. The microinstruction inputs used to select the ALU source operands are the I0, I1, and I2 inputs. The definition of I0, I1, and I2 for the eight source operand combinations are as shown in Figure 2. Also shown is the octal code for each selection.

The two source operands not fully described as yet are the D input and Q input. The D input is the four-bit wide direct data field input. This port is used to insert all data into the working registers inside the device. Likewise, this input can be used in the ALU to modify any of the internal data files. The Q register is a separate 4-bit file intended primarily for multiplication and division routines, but it can also be used as an accumulator or holding register for some applications.

The ALU itself is a high-speed arithmetic/logic operator capable of performing three binary arithmetic and five logic functions. The I3, I4, and I5 microinstruction inputs are used to select the ALU function. The definition of these inputs is shown in Figure 3. The octal code is also shown for reference. The normal technique for cascading the ALU of several devices is in a look-ahead carry mode. Carry generate, $\overline{G}$, and carry propagate, $\overline{P}$, are outputs of the device for use with a carry-look-ahead generator such as the MC2902. A carry-out, C_{n+4}, is also generated and is available as an output for use as the carry flag in a status register. Both carry-in (C_n) and carry-out (C_{n+4}) are active high.

The ALU has three other status-oriented outputs. These are F3, F = 0, and overflow (OVR). The F3 output is the most significant (sign) bit of the ALU and can be used to determine positive or negative results without enabling the three-state data outputs. F3 is noninverted with respect to the sign bit output, Y3. The F = 0 output is used for zero detect. It is an open-collector output and can be wire-OR'ed between microprocessor slices. F = 0 is high when all F outputs are low. The overflow output (OVR) is high when overflow exists. That is, when C_{n+3} and C_{n+4} are not the same polarity.

The ALU data output is routed to several destinations. It can be a data output of the device. It can also be stored in the RAM or the Q register. Eight possible combinations of ALU destination functions are available as defined by the I6, I7, and I8 microinstruction inputs. These combinations are shown in Figure 4.

FIGURE 2 – ALU SOURCE OPERAND CONTROL

Micro Code				ALU Source Operands	
Pin 14 I2	Pin 13 I1	Pin 12 I0	Octal Code	R	S
L	L	L	0	A	Q
L	L	H	1	A	B
L	H	L	2	O	Q
L	H	H	3	O	B
H	L	L	4	O	A
H	L	H	5	D	A
H	H	L	6	D	Q
H	H	H	7	D	O

FIGURE 3 – ALU FUNCTION CONTROL

Micro Code					
Pin 27 I5	Pin 28 I4	Pin 26 I3	Octal Code	ALU Function	Symbol
L	L	L	0	R Plus S	R + S
L	L	H	1	S Minus R	S – R
L	H	L	2	R Minus S	R – S
L	H	H	3	R OR S	R ∨ S
H	L	L	4	R AND S	R ∧ S
H	L	H	5	$\overline{R}$ AND S	$\overline{R}$ ∧ S
H	H	L	6	R EX-OR S	R ∀ S
H	H	H	7	R EX-NOR S	$\overline{R \forall S}$

FIGURE 4 – ALU DESTINATION CONTROL

Micro Code				RAM Function		Q-Reg Function		Y	RAM Shifter		Q Shifter	
Pin 6 I8	Pin 7 I7	Pin 5 I6	Octal Code	Shift	Load	Shift	Load	Output	RAM0	RAM3	Q0	Q3
L	L	L	0	X	None	None	F → Q	F	X	X	X	X
L	L	H	1	X	None	X	None	F	X	X	X	X
L	H	L	2	None	F → B	X	None	A	X	X	X	X
L	H	H	3	None	F → B	X	None	F	X	X	X	X
H	L	L	4	Down	F/2 → B	Down	Q/2 → Q	F	F0	IN3	Q0	IN3
H	L	H	5	Down	F/2 → B	X	None	F	F0	IN3	Q0	X
H	H	L	6	Up	2F → B	Up	2Q → Q	F	IN0	F3	IN0	Q3
H	H	H	7	Up	2F → B	X	None	F	IN0	F3	X	Q3

X = Don't care. Electrically, the shift pin is a TTL input internally connected to a three-state output which is in the high-impedance state.
B = Register Addressed by B inputs.
Up is toward MSB; Down is toward LSB.

The four-bit data output field (Y) features three-state outputs and can be directly bus-organized. An output control ($\overline{OE}$) is used to enable the three-state outputs. When $\overline{OE}$ is high, the Y outputs are in the high-impedance state.

A two-input multiplexer is also used at the data output such that either the A-port of the RAM or the ALU outputs (F) are selected at the device Y outputs. This selection is controlled by the I6, I7, and I8 microinstruction inputs. Refer to Figure 4 for the selected output for each microinstruction code combination.

As was discussed previously, the RAM inputs are driven from a three-input multiplexer. This allows the ALU outputs to be entered nonshifted, shifted up one position (X2) or shifted down one position (÷2). The shifter has two ports: one is labeled RAM0 and the other is labeled RAM3. Both of these ports consist of a buffer-driver with a three-state output and an input to the multiplexer. Thus, in the shift up mode, the RAM3 buffer is enabled and the RAM0 multiplexer input is enabled. Likewise, in the shift down mode, the RAM0 buffer and RAM3 input are enabled. In the no-shift mode, both buffers are in the high-impedance state and the multiplexer inputs are not selected. This shifter is controlled from the I6, I7, and I8 microinstruction inputs as defined in Figure 4.

Similarly, the Q register is driven from a 3-input multiplexer. In the no-shift mode, the multiplexer enters the ALU data into the Q register. In either the shift-up or shift-down mode, the multiplexer selects the Q register data appropriately shifted up or down. The Q shifter also has two ports: one is labeled Q0 and the other is Q3. The operation of these two ports is similar to the RAM shifter and is also controlled from I6, I7, and I8 as shown in Figure 4.

The clock input to the MC2901A controls the RAM, the Q register, and the A and B data latches. When enabled, data is clocked into the Q register on the low-to-high transition of the clock. When the clock input is high, the A and B latches are open and will pass whatever data is present at the RAM outputs. When the clock input is low, the latches are closed and will retain the last data entered. If the RAM-EN is enabled, new data will be written into the RAM file (word) defined by the B address field when the clock input is low.

SOURCE OPERANDS AND ALU FUNCTIONS

There are eight source operand pairs available to the ALU as selected by the I0, I1, and I2 instruction inputs. The ALU can perform eight functions: five logic and three arithmetic. The I3, I4, and I5 instruction inputs control this function selection. The carry input, C_n, also affects the ALU results when in the arithmetic mode. The C_n input has no effect in the logic mode. When I0 through I5 and C_n are viewed together, the matrix of Figure 5 results. This matrix fully defines the ALU/source operands function for each state.

The ALU functions can also be examined on a "task" basis, i.e., add, subtract, AND, OR, etc. In the arithmetic mode, the carry will affect the function performed while in the logic mode, the carry will have no bearing on the ALU output. Figure 6 defines the various logic operations that the MC2901A can perform and Figure 7 shows the arithmetic functions of the device. Both carry-in low (C_n = 0) and carry-in high (C_n = 1) are defined in these operations.

FIGURE 5 – SOURCE OPERAND AND ALU FUNCTION MATRIX

		I210 Octal							
		0	1	2	3	4	5	6	7
I543 Octal	ALU Function	ALU Source							
		A, Q	A, B	0, Q	0, B	0, A	D, A	D, Q	D, 0
0	C_n = L R Plus S C_n = H	A + Q A + Q + 1	A + B A + B + 1	Q Q + 1	B B + 1	A A + 1	D + A D + A + 1	D + Q D + Q + 1	D D + 1
1	C_n = L S Minus R C_n = H	Q – A – 1 Q – A	B – A – 1 B – A	Q – 1 Q	B – 1 B	A – 1 A	A – D – 1 A – D	Q – D – 1 Q – D	–D – 1 –D
2	C_n = L R Minus S C_n = H	A – Q – 1 A – Q	A – B – 1 A – B	–Q – 1 –Q	–B –1 –B	–A – 1 –A	D – A – 1 D – A	D – Q – 1 D – Q	D – 1 D
3	R OR S	A ∨ Q	A ∨ B	Q	B	A	D ∨ A	D ∨ Q	D
4	R AND S	A ∧ Q	A ∧ B	0	0	0	D ∧ A	D ∧ Q	0
5	$\overline{R}$ AND S	$\overline{A}$ ∧ Q	$\overline{A}$ ∧ B	Q	B	A	$\overline{D}$ ∧ A	$\overline{D}$ ∧ Q	0
6	R EX-OR S	A ∀ Q	A ∀ B	Q	B	A	D ∀ A	D ∀ Q	D
7	R EX-NOR S	$\overline{A \forall Q}$	$\overline{A \forall B}$	$\overline{Q}$	$\overline{B}$	$\overline{A}$	$\overline{D \forall A}$	$\overline{D \forall Q}$	$\overline{D}$

+ = Plus; – = Minus; ∨ = OR; ∧ = AND; ∀ = EX-OR

LOGIC FUNCTIONS FOR G, P, C_{n+4}, AND OVR

The four signals—G, P, C_{n+4}, and OVR—are designed to indicate carry and overflow conditions when the MC2901A is in the add or subtract mode. Figure 8 indicates the logic equations for these four signals for each of the eight ALU functions. The R and S inputs are the two inputs selected according to Figure 2.

FIGURE 6 – ALU LOGIC MODE FUNCTIONS
(C_n Irrelevant)

Octal I543, I210	Group	Function
4 0	AND	A ∧ Q
4 1		A ∧ B
4 5		D ∧ A
4 6		D ∧ Q
3 0	OR	A ∨ Q
3 1		A ∨ B
3 5		D ∨ A
3 6		D ∨ Q
6 0	EX-OR	A ∀ Q
6 1		A ∀ B
6 5		D ∀ A
6 6		D ∀ Q
7 0	EX-NOR	$\overline{A \forall Q}$
7 1		$\overline{A \forall B}$
7 5		$\overline{D \forall A}$
7 6		$\overline{D \forall Q}$
7 2	INVERT	$\overline{Q}$
7 3		$\overline{B}$
7 4		$\overline{A}$
7 7		$\overline{D}$
6 2	PASS	Q
6 3		B
6 4		A
6 7		D
3 2	PASS	Q
3 3		B
3 4		A
3 7		D
4 2	"ZERO"	0
4 3		0
4 4		0
4 7		0
5 0	MASK	$\overline{A} \wedge Q$
5 1		$\overline{A} \wedge B$
5 5		$\overline{D} \wedge A$
5 6		$\overline{D} \wedge Q$

Definitions (+ = OR)

P0 = R0 + S0 G0 = R0S0

P1 = R1 + S1 G1 = R1S1

P2 = R2 + S2 G2 = R2S2

P3 = R3 + S3 G3 = R3S3

$C4 = G3 + P3G2 + P3P2G1 + P3P2P1G0 + P3P2P1P0C_n$

$C3 = G2 + P2G1 + P2P1G0 + P2\,P1P0C_n$

FIGURE 7 – ALU ARITHMETIC MODE FUNCTIONS

Octal	C_n = 0 (Low)		C_n = 1 (High)	
I543, I210	Group	Function	Group	Function
0 0	ADD	A + Q	ADD plus one	A + Q + 1
0 1		A + B		A + B + 1
0 5		D + A		D + A + 1
0 6		D + Q		D + Q + 1
0 2	PASS	Q	Increment	Q + 1
0 3		B		B + 1
0 4		A		A + 1
0 7		D		D + 1
1 2	Decrement	Q – 1	PASS	Q
1 3		B – 1		B
1 4		A – 1		A
2 7		D – 1		D
2 2	1's Comp	–Q – 1	2's Comp (Negate)	–Q
2 3		–B – 1		–B
2 4		–A – 1		–A
1 7		–D – 1		–D
1 0	Subtract (1's Comp)	Q – A – 1	Subtract (2's Comp)	Q – A
1 1		B – A – 1		B – A
1 5		A – D – 1		A – D
1 6		Q – D – 1		Q – D
2 0		A – Q – 1		A – Q
2 1		A – B – 1		A – B
2 5		D – A – 1		D – A
2 6		D – Q – 1		D – Q

FIGURE 8 – LOGIC FUNCTIONS

I543	Function	Pin 35 $\overline{P}$	Pin 32 $\overline{G}$	Pin 33 C_{n+4}	Pin 34 OVR
0	R + S	$\overline{P3P2P1P0}$	$\overline{G3 + P3G2 + P3P2G1 + P3P2P1G0}$	C4	C3 ∀ C4
1	S – R	Same as R + S equations, but substitute $\overline{R_i}$ for R_i in definitions			
2	R – S	Same as R + S equations, but substitute $\overline{S_i}$ for S_i in definitions			
3	R ∨ S	Low	P3P2P1P0	$\overline{P3P2P1P0} + C_n$	$\overline{P3P2P1P0} + C_n$
4	R ∧ S	Low	$\overline{G3 + G2 + G1 + G0}$	$G3 + G2 + G1 + G0 + C_n$	$G3 + G2 + G1 + G0 + C_n$
5	$\overline{R} \wedge S$	Low	Same as R ∧ S equations, but substitute $\overline{R_i}$ for R_i in definitions		
6	R ∀ S	Same as $\overline{R \forall S}$, but substitute $\overline{R_i}$ for R_i in definitions			
7	$\overline{R \forall S}$	G3 + G2 + G1 + G0	G3 + P3G2 + P3P2G1 + P3P2P1P0	$\overline{G3 + P3G2 + P3P2G1 + P3P2P1P0(G0 + \overline{C_n})}$	See Note

+ = OR

NOTE: $[\overline{P2} + \overline{G2}\,\overline{P1} + \overline{G2}\,\overline{G1}\,\overline{P0} + \overline{G2}\,\overline{G1}\,\overline{G0}C_n] \forall [\overline{P3} + \overline{G3}\,\overline{P2} + \overline{G3}\,\overline{G2}\,\overline{P1} + \overline{G3}\,\overline{G2}\,\overline{G1}\,\overline{P0} + \overline{G3}\,\overline{G2}\,\overline{G1}\,\overline{G0}C_n]$

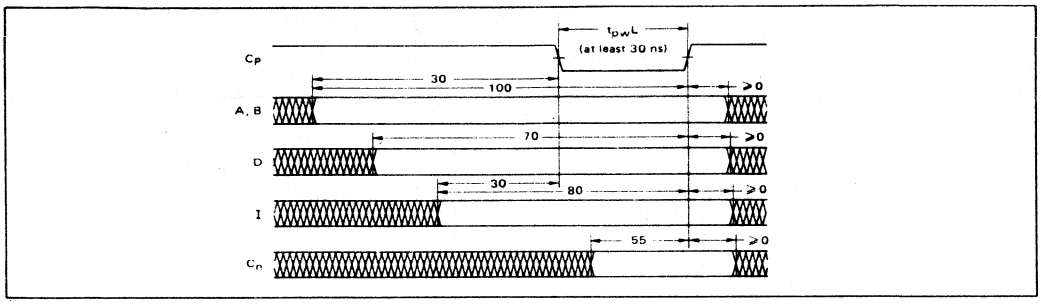
SETUP AND HOLD TIMES

(Minimum Cycles from Each Input)

Setup and hold times are defined relative to the clock low-to-high edge. Inputs must be steady at all times from the setup time prior to the clock, until the hold time after the clock. The setup times allow sufficient time to perform the correct operation on the correct data so that the correct ALU data can be written into one of the registers.

FIGURE 9 – MINIMUM CYCLE TIMES FROM INPUTS

(Numbers shown are minimum data stable times for MC2901ALC, in ns. See Table 3 for detailed information.)



PIN DEFINITIONS

A0–3 The four address inputs to the register stack used to select one register whose contents are displayed through the A-port.

B0–3 The four address inputs to the register stack used to select one register whose contents are displayed through the B-port and into which new data can be written when the clock goes low.

I0–8 The nine instruction control lines to the MC2901, used to determine what data sources will be applied to the ALU (I012), what function the ALU will perform (I345), and what data is to be deposited in the Q-register or the register stack (I678).

Q3 RAM3 A shift line at the MSB of the Q register (Q3) and the register stack (RAM3). Electrically, these lines are three-state outputs connected to TTL inputs internal to the MC2901A. When the destination code on I678 indicates an up shift (octal 6 or 7) the three-state outputs are enabled and the MSB of the Q register is available on the Q3 pin and the MSB of the ALU output is available on the RAM3 pin. Otherwise, the three-state outputs are off (high-impedance) and the pins are electrically LS-TTL inputs. When the destination code calls for a down shift, the pins are used as the data inputs to the MSB of the Q register (octal 4) and RAM (octal 4 or 5).

Q0 RAM0 Shift lines like Q3 and RAM3, but at the LSB of the Q-register and RAM. These pins are tied to the Q3 and RAM3 pins of the adjacent device to transfer data between devices for up and down shifts of the Q register and ALU data.

D0–3 Direct data inputs. A four-bit data field which may be selected as one of the ALU data sources for entering data into the MC2901A. D0 is the LSB.

Y0–3 The four data outputs of the MC2901A. These are three-state output lines. When enabled, they display either the four outputs of the ALU or the data on the A-port of the register stack, as determined by the destination code I678.

$\overline{OE}$ $\overline{\text{Output Enable}}$. When $\overline{OE}$ is high, the Y outputs are off; when $\overline{OE}$ is low, the Y outputs are active (high or low).

$\overline{P}, \overline{G}$ The carry generate and propagate outputs of the MC2901A's ALU. These signals are used with the MC2902 for carry-look-ahead. See Figure 8 for the logic equations.

OVR Overflow. This pin is logically the Exclusive-OR of the carry-in and carry-out of the MSB of the ALU. At the most significant end of the word, this pin indicates that the result of an arithmetic two's complement operation has overflowed into the sign-bit. See Figure 8 for logic equation.

F = 0 This is an open collector output which goes high (off) if the data on the four ALU outputs F0–3 are all low. In positive logic, it indicates the result of an ALU operation is zero.

C_n The carry-in to the MC2901A's ALU.

C_{n+4} The carry-out of the MC2901A's ALU. See Figure 8 for equations.

CP The clock to the MC2901A. The Q register and register stack outputs change on the clock low-to-high transition. The clock low time is internally the write enable to the 16 X 4 RAM which comprises the "master" latches of the register stack. While the clock is low, the "slave" latches on the RAM outputs are closed, storing the data previously on the RAM outputs. This allows synchronous master-slave operation of the register stack.

FIGURE 10 – MC2901A INPUT/OUTPUT CURRENT INTERFACE CONDITIONS

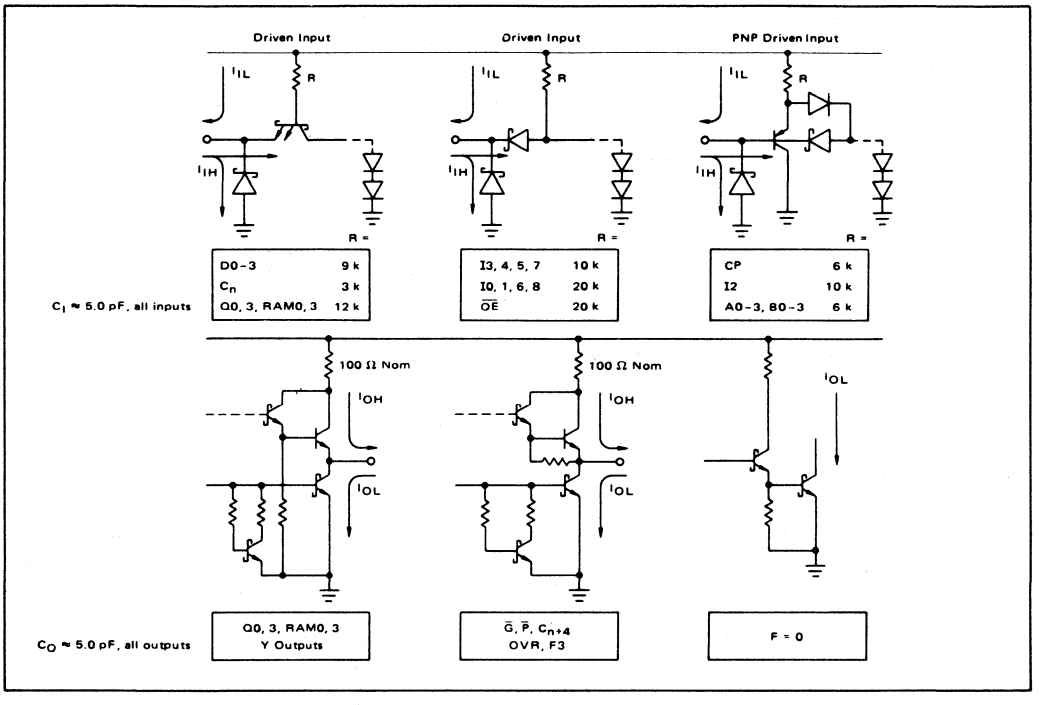
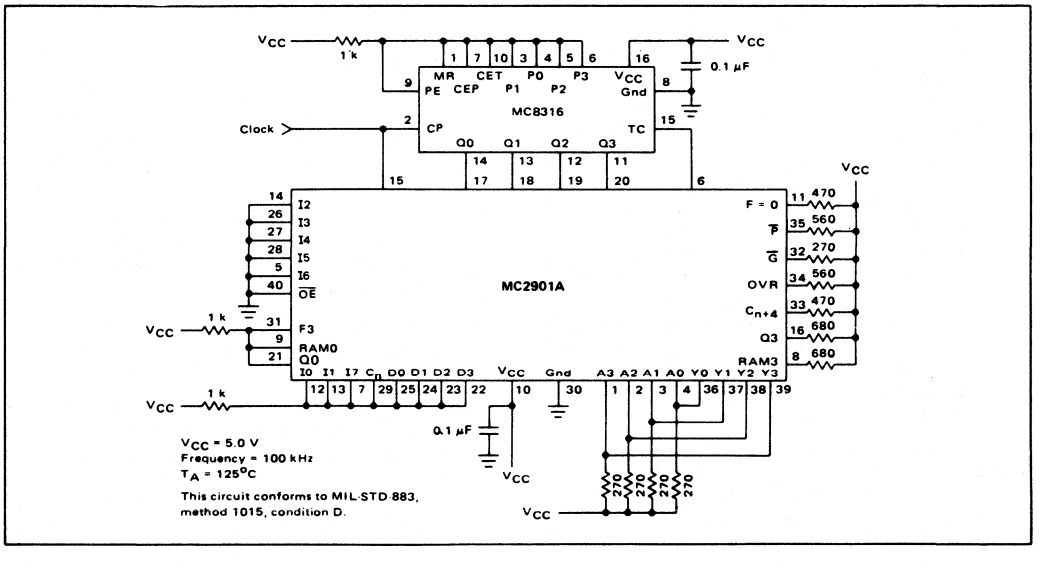
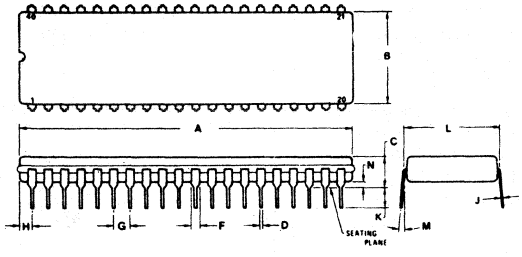


FIGURE 11 – MC2901A BURN-IN CIRCUIT



PACKAGE DIMENSIONS

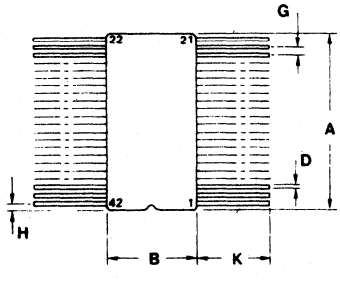


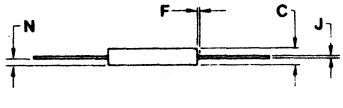
NOTES:
1. LEADS WITHIN 0.25 mm (0.010) DIA OF TRUE POSITION AT SEATING PLANE, AT MAXIMUM MATERIAL CONDITION.
2. DIM L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIM A AND B INCLUDES MENISCUS.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	52.32	53.09	2.060	2.090
B	12.70	13.72	0.500	0.540
C	5.08	5.84	0.200	0.230
D	0.38	0.56	0.015	0.022
F	1.27	1.65	0.050	0.065
G	2.54 BSC		0.100 BSC	
H	2.03	2.41	0.080	0.095
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	15.24 BSC		0.600 BSC	
M	5^0	15^0	5^0	15^0
N	0.51	1.27	0.020	0.050

L SUFFIX
CERAMIC PACKAGE
CASE 734

F SUFFIX
CERAMIC PACKAGE
CASE 735





	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	26.92	27.60	1.060	1.090
B	13.72	14.22	0.540	0.560
C	2.41	3.18	0.095	0.125
D	0.43	0.53	0.017	0.023
F	–	0.38	–	0.015
G	1.27 BSC		0.050 BSC	
H	0.89	1.14	0.035	0.045
J	0.20	0.30	0.008	0.012
K	–	11.94	–	0.470
N	–	1.27	–	0.050

NOTE:
1. LEADS WITHIN 0.25 mm (0.010) TOTAL OF TRUE POSITION AT MAXIMUM MATERIAL CONDITION.

MOTOROLA

MC2902

HIGH-SPEED LOOK-AHEAD CARRY GENERATOR

The MC2902 is a high-speed, look-ahead carry generator which accepts up to four pairs of carry propagate and carry generate signals and a carry input and provides anticipated carries across four groups of binary ALUs. The device also has carry propagate and carry generate outputs which may be used for further levels of look-ahead.

The MC2902 is generally used with the bipolar microprocessor unit to provide look-ahead over word lengths of more than four bits. The look-ahead carry generator can be used with binary ALUs in an active LOW or active HIGH input operand mode by reinterpreting the carry functions. The connections to and from the ALU to the look-ahead carry generator are identical in both cases.

The logic equations provided at the outputs are:

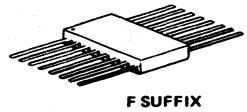
$$C_{n+x} = G_0 + P_0C_n$$
$$C_{n+y} = G_1 + P_1G_0 + P_1P_0C_n$$
$$C_{n+z} = G_2 + P_2G_1 + P_2P_1G_0 + P_2P_1P_0C_n$$
$$G = G_3 + P_3G_2 + P_3P_2G_1 + P_3P_2P_1G_0$$
$$P = P_3P_2P_1P_0$$

FEATURES

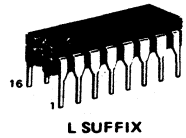
- Provides look-ahead carries across a group of four MC2901 microprocessor ALUs.
- Capability of multi-level look-ahead for high-speed arithmetic operation over large word lengths.
- Typical carry propagation delay of 6.0 ns.

TTL

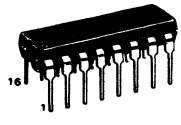
HIGH-SPEED LOOK-AHEAD CARRY GENERATOR



F SUFFIX
CERAMIC PACKAGE
CASE 650

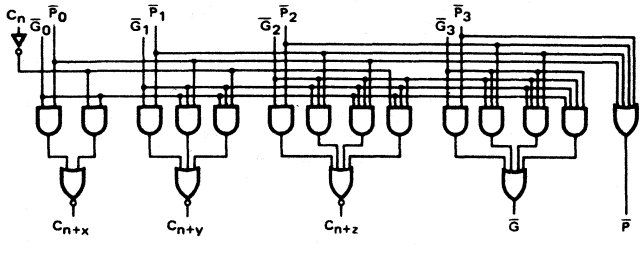


L SUFFIX
CERAMIC PACKAGE
CASE 620

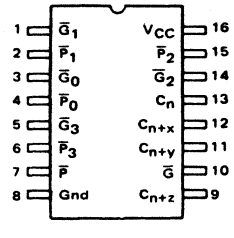


P SUFFIX
PLASTIC PACKAGE
CASE 648

LOGIC DIAGRAM



PIN ASSIGNMENT



ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Molded DIP	0°C to +70°C	MC2902PC
Hermetic DIP	0°C to +70°C	MC2902LC
Hermetic DIP	-55°C to +125°C	MC2902LM
Hermetic Flat Pack	-55°C to +125°C	MC2902FM

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential	−0.5V to +7.0V
DC Voltage Applied to Outputs for HIGH Output State	−0.5V to +V_{CC} max.
DC Input Voltage	−0.5V to +5.5V
DC Output Current, Into Outputs	30 mA
DC Input Current	−30 mA to +5.0 mA

ELECTRICAL CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (Unless Otherwise Noted)

MC2902XC T_A = 0°C to +70°C V_{CC} = 5.0V ±5% (COM'L) MIN. = 4.75V MAX. = 5.25V
MC2902XM T_A = −55°C to +125°C V_{CC} = 5.0V ±10% (MIL) MIN. = 4.50V MAX. = 5.50V

Parameters	Description	Test Conditions (Note 1)		Min.	Typ. (Note 2)	Max.	Units
V_{OH}	Output HIGH Voltage	V_{CC} = MIN., I_{OH} = −0.8mA V_{IN} = V_{IH} or V_{IL}		2.4	3.0		Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN., I_{OL} = 16mA V_{IN} = V_{IH} or V_{IL}			0.2	0.4	Volts
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs		2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs				0.8	Volts
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −12mA				−1.5	Volts
I_{IL} (Note 3)	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.4V	C_n			−3.2	mA
			$\overline{P}_3$			−4.8	
			$\overline{P}_2$			−6.4	
			$\overline{P}_0, \overline{P}_1, \overline{G}_3$			−8.0	
			$\overline{G}_0, \overline{G}_2$			−14.4	
			$\overline{G}_1$			−16	
I_{IH} (Note 3)	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.4V	C_n			80	µA
			$\overline{P}_3$			120	
			$\overline{P}_2$			160	
			$\overline{P}_0, \overline{P}_1, \overline{G}_3$			200	
			$\overline{G}_0, \overline{G}_2$			360	
			$\overline{G}_1$			400	
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 5.5V				1.0	mA
I_{SC}	Output Short Circuit (Note 4)	V_{CC} = MAX., V_{OUT} = 0.0V		−40		−100	mA
I_{CC}	Power Supply Current	V_{CC} = MAX. All Outputs LOW	MIL		62	99	mA
			COM'L		58	94	
		V_{CC} = MAX. All Ouputs HIGH	MIL		37		mA
			COM'L		35		

Notes: 1. For conditions shown as MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0V, 25°C ambient and maximum loading.
3. Actual input currents = Unit Load Current X Input Load Factor (see Loading Rules).
4. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.

SWITCHING CHARACTERISTICS V_{CC} = 5.0V, T_A = 25°C, C_L = 15pF, R_L = 400Ω

Parameter	From (Input)	To (Output)	Test Figure	Test Conditions	Min	Typ	Max	Units
t_{PLH}	C_n	C_{n+j}	2	$\overline{P}_0 = \overline{P}_i = \overline{P}_2 = 0$ V $\overline{G}_0 = \overline{G}_1 = \overline{G}_2 = 4.5$ V		11	14	ns
t_{PHL}						11	14	
t_{PLH}	$\overline{P}_i$	C_{n+j}	3	$\overline{P}_i = 0$ V (j > i) $C_n = \overline{G}_0 = \overline{G}_1 = \overline{G}_2 = 4.5$ V		6.0	8.0	ns
t_{PHL}						6.0	8.0	
t_{PLH}	$\overline{G}_i$	C_{n+j}	3	$\overline{G}_i = 0$ V (j > i) $C_n = \overline{P}_0 = \overline{P}_1 = \overline{P}_2 = 4.5$ V		8.0	10	ns
t_{PHL}						8.0	10	
t_{PLH}	$\overline{P}_i$	$\overline{G}$ or $\overline{P}$	2	$\overline{P}_i = 0$ V (j > i) $C_n = \overline{G}_0 = \overline{G}_1 = \overline{G}_2 = 4.5$ V		11	14	ns
t_{PHL}						11	14	
t_{PLH}	$\overline{G}_i$	$\overline{G}$ or $\overline{P}$	2	$\overline{G}_i = 0$ V (j > i) $C_n = \overline{P}_0 = \overline{P}_1 = \overline{P}_2 = 4.5$ V		12	14	ns
t_{PHL}						12	14	

DEFINITION OF FUNCTIONAL TERMS

C_n Carry-in. The carry-in input to the look-ahead generator. Also the carry-in input to the nth MC2901 microprocessor ALU input.

C_{n+j} Carry-out. (j = x, y, z). The carry-out output to be used at the carry-in inputs of the n+1, n+2, and n+3 microprocessor ALU slices.

G_i P_i Generate and propagate inputs, respectively (i = 0, 1, 2, 3). The carry generate and carry propagate inputs from the n, n+1, n+2 and n+3 microprocessor ALU slices.

G, P Generate and propagate outputs, respectively. The carry generate and carry propagate outputs that can be used with the next higher level of carry look-ahead if used.

LOADING RULES (In Unit Loads)

Input/Output	Pin No.	Input Unit Load	Fan-out Output HIGH	Fan-out Output LOW
$\overline{G_1}$	1	8.0	–	–
$\overline{P_1}$	2	4.0	–	–
$\overline{G_0}$	3	7.2	–	–
$\overline{P_0}$	4	4.0	–	–
$\overline{G_3}$	5	4.0	–	–
$\overline{P_3}$	6	2.4	–	–
$\overline{P}$	7	–	16	8
GND	8	–	–	–
C_{n+z}	9	–	16	8
$\overline{G}$	10	–	16	8
C_{n+y}	11	–	16	8
C_{n+x}	12	–	16	8
C_n	13	1.6	–	–
$\overline{G_2}$	14	7.2	–	–
$\overline{P_2}$	15	3.2	–	–
V_{CC}	16	–	–	–

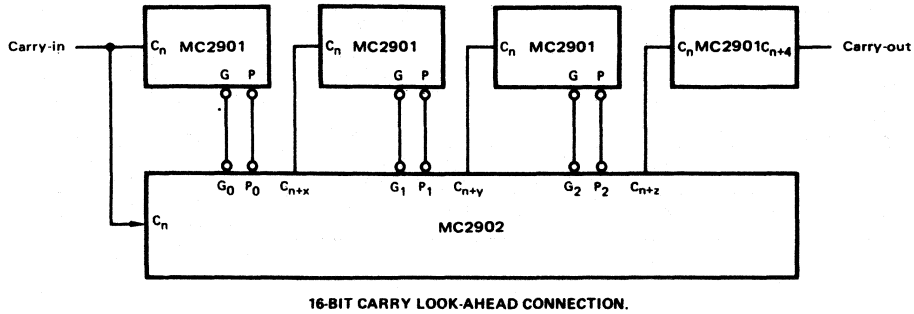
A Schottky TTL Unit Load is defined as 50μA measured at 2.7V HIGH and −2.0mA measured at 0.5V LOW.

TRUTH TABLE

Inputs								Outputs				
C_n	$\overline{G_0}$	$\overline{P_0}$	$\overline{G_1}$	$\overline{P_1}$	$\overline{G_2}$	$\overline{P_2}$	$\overline{G_3}$	$\overline{P_3}$	C_{n+x}	C_{n+y}	C_{n+z}	$\overline{G}$	$\overline{P}$
X	H	H							L				
L	H	X							L				
X	L	X							H				
H	X	L							H				
X	X	X	H	H						L			
X	H	H	H	X						L			
L	H	X	H	X						L			
X	X	X	L	X						H			
X	H	X	L	X						H			
H	X	L	X	L						H			
X	X	X	X	H	H	X					L		
X	X	X	H	H	H	X					L		
X	H	H	H	X	H	X					L		
L	H	X	H	X	H	L					L		
X	X	X	X	X	H	X					H		
X	X	X	X	L	H	L					H		
X	L	X	X	L	H	L					H		
H	X	L	X	L	H	L					H		
	X		X	X	X	H	H	H				H	
	X		X	H	H	X	H	X				H	
	X		H	H	H	H	H	X				H	
	H		X	H	X	X	H	X				H	
	X		X	X	X	X	L	X				L	
	X		X	X	X	L	X	L				L	
	X		L	X	X	X	L	L				L	
	L		X	L	X	L	X	L				L	
		H		X		X		X					H
		X		H		X		X					H
		X		X		H		X					H
		X		X		X		H					H
		L		X		L		L					L

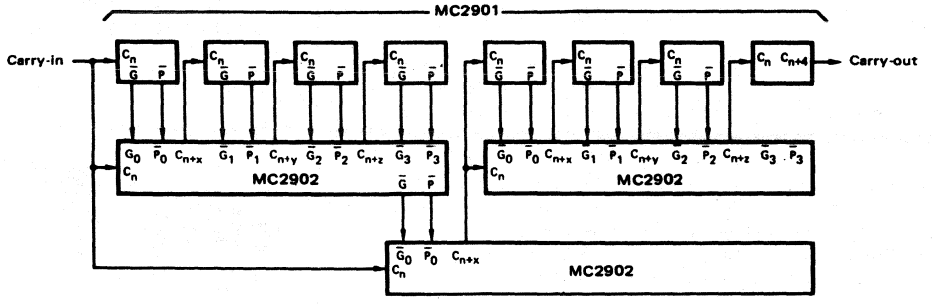
H = HIGH Voltage Level
L = LOW Voltage Level
X = Don't Care

APPLICATIONS



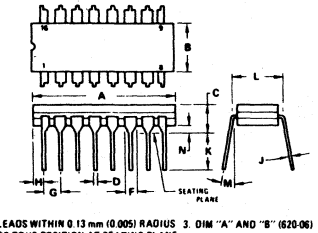
16-BIT CARRY LOOK-AHEAD CONNECTION.

APPLICATIONS (Cont'd.)



32-BIT ALU, THREE LEVEL CARRY LOOK-AHEAD.

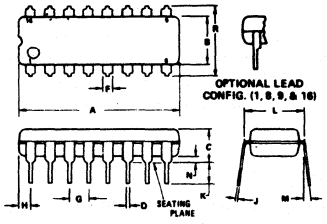
PACKAGE DIMENSIONS



1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. PACKAGE INDEX: NOTCH IN LEAD NOTCH IN CERAMIC OR INK DOT.
3. DIM "A" AND "B" (620-06) DO NOT INCLUDE GLASS RUN-OUT.
4. DIM "L" TO INSIDE OF LEADS (MEASURED 0.51 mm (0.020) BELOW BODY).

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	19.05	19.94	0.750	0.785
B	6.10	7.49	0.240	0.295
C	–	5.08	–	0.200
D	0.38	0.53	0.015	0.021
F	1.40	1.78	0.055	0.070
G	2.54 BSC		0.100 BSC	
H	0.51	1.14	0.020	0.045
J	0.20	0.30	0.008	0.012
K	3.18	5.08	0.125	0.200
L	7.49	8.89	0.295	0.350
M	–	15°	–	15°
N	0.51	1.02	0.020	0.040

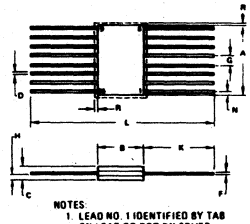
CASE 620



NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSION "B" DOES NOT INCLUDE MOLD FLASH.
4. "F" DIMENSION IS FOR FULL LEADS. "HALF" LEADS ARE OPTIONAL AT LEAD POSITIONS 1, 8, 9, and 16).
5. DIMENSION "R" TO BE MEASURED AT THE TOP OF THE LEADS (NOT AT THE TIPS).

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	–	22.10	–	0.870
B	6.10	6.60	0.240	0.260
C	–	5.08	–	0.200
D	0.38	0.53	0.015	0.021
F	–	1.78	–	0.070
G	2.54 BSC		0.100 BSC	
H	0.38	2.41	0.015	0.095
J	0.20	0.38	0.008	0.015
K	2.92	–	0.115	–
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	–	0.020	–
R	–	8.26	–	0.325

CASE 648



NOTES:
1. LEAD NO. 1 IDENTIFIED BY TAB ON LEAD OR DOT ON COVER.
2. LEADS WITHIN 0.13 mm (0.005) TOTAL OF TRUE POSITION AT MAXIMUM MATERIAL CONDITION.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	9.40	10.16	0.370	0.400
B	6.22	7.24	0.245	0.285
C	1.52	2.03	0.060	0.080
D	0.41	0.48	0.016	0.019
F	0.06	0.15	0.003	0.006
G	1.27 BSC		0.050 BSC	
H	0.64	0.89	0.025	0.035
K	6.35	9.40	0.250	0.370
L	10.92	–	0.745	–
M	–	0.51	–	0.020
R	–	0.38	–	0.015

CASE 650

MOTOROLA

MC2903

Product Preview

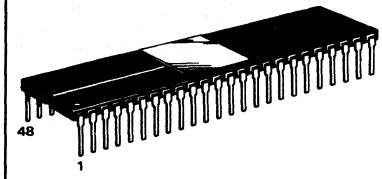
TTL

FOUR-BIT BIPOLAR MICROPROCESSOR SLICE

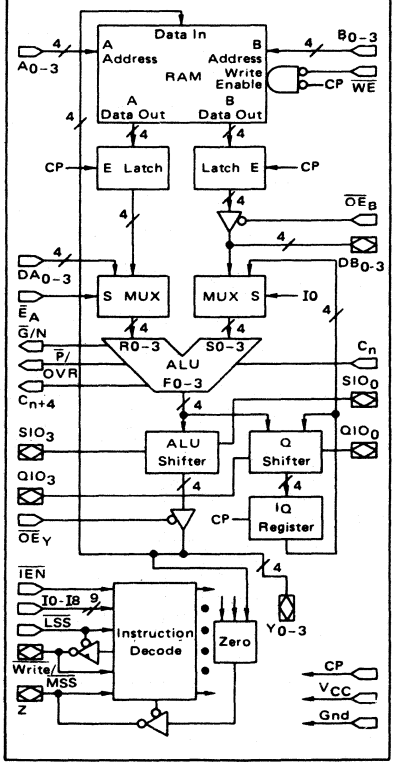
FOUR-BIT BIPOLAR MICROPROCESSOR SLICE

The MC2903 is a four-bit expandable bipolar microprocessor slice. The MC2903 performs all functions performed by the industry standard MC2901A and, in addition, provides a number of significant enhancements that are especially useful in arithmetic-oriented processors. Infinitely expandable memory and three-port, three-address architecture are provided by the MC2903. In addition to its complete arithmetic and logic instruction set, the MC2903 provides a special set of instructions which facilitate the implementation of multiplication, division, normalization, and other previously time-consuming operations. The MC2903 is supplied in a 48 pin dual in-line package.

- **Built-in Parity Generation Circuitry**
  The MC2903 can supply parity across the entire ALU output for use in error detection and CRC code generation.
- **Built-in Sign Extension Circuitry**
  To facilitate operation on different length two's complement numbers, the MC2903 provides the capability to extend the sign at any slice boundary.
- **Expandable Register File**
  Like the MC2901A, the MC2903 contains 16 internal working registers arranged in a two-address architecture. But the MC2903 includes the necessary "hooks" to expand the register file externally to any number of registers.
- **Built-in Multiplication Logic**
  Performing multiplication with the MC2901A requires a few external gates—these gates are contained on-chip in the MC2903. Three special instructions are used for unsigned multiplication, two's complement multiplication, and the last cycle of a two's complement multiplication.
- **Built-in Division Logic**
  The MC2903 contains all logic and interconnects for execution of a non-restoring, multiple-length division with correction of the quotient.
- **Built-in Normalization Logic**
  The MC2903 can simultaneously shift the Q Register and count in a working register. Thus, the mantissa and exponent of a floating point number can be developed using a single microcycle per shift. Status flags indicate when the operation is complete.



BLOCK DIAGRAM



ORDERING INFORMATION		
Package Type	**Temperature Range**	**Order Number**
Hermetic DIP	0°C to +70°C	MC2903LC
Hermetic DIP	-55°C to +125°C	MC2903LM

ARCHITECTURE OF THE MC2903

The MC2903 is a high performance, cascadable, 4-bit bipolar microprocessor slice designed for use in CPUs, peripheral controllers, microprogrammable machines, and numerous other applications. The microinstruction flexibility of the MC2903 allows the efficient emulation of almost any digital computing machine. The 9-bit microinstruction selects the ALU sources, function, and destination. The MC2903 is cascadable with full look-ahead or ripple carry, has three-state outputs, and provides various ALU status flag outputs. Advanced low-power Schottky processing is used to fabricate this 48-pin LSI circuit.

All data paths within the device are four bits wide. As shown in the block diagram, the device consists of a 16-word by 4-bit, two-port RAM with latches on both output ports, a high-performance ALU and shifter, a multipurpose Q Register with shifter input, and a 9-bit instruction decoder.

Two-Port RAM

Any two RAM words addressed at the A and B address ports can be read simultaneously at the respective RAM A and B output ports. Identical data appear at the two output ports when the same address is applied to both address ports. The latches at the RAM output ports are transparent when the clock input, CP, is High and they hold the RAM output data when CP is Low. Under control of the $\overline{OE}_B$ three-state output enable, RAM data can be read directly at the MC2903 DB I/O port.

External data at the MC2903 Y I/O port can be written directly into the RAM, or ALU shifter output data can be enabled onto the Y I/O port and entered into the RAM. Data is written into the RAM at the B address when the write enable input, $\overline{WE}$, is Low and the clock input, CP, is Low.

Arithmetic Logic Unit

The MC2903 high performance ALU can perform seven arithmetic and nine logic operations on two 4-bit operands. Multiplexers at the ALU inputs provide the capability to select various pairs of ALU source operands. The E_A input selects either the DA external data input or RAM output port A for use as one ALU operand and the OE_B and I0 inputs select RAM output port B, DB external data input, or the Q Register content for use as the second ALU operand. Also, during some ALU operations, zeroes are forced at the ALU operand inputs. Thus, the MC2903 ALU can operate on data from two external sources, from an internal and external source, or from two internal sources. Table 1 shows all possible pairs of ALU source operands as a function of the E_A, OE_B, and I0 inputs.

When instruction bits I4, I3, I2, I1, and I0 are Low, the MC2903 executes special functions. Table 4 defines these special functions and the operation which the ALU performs for each. When the MC2903 executes instructions other than the nine special functions, the ALU operation is determined by instruction bits I4, I3, I2, and I1. Table 2 defines the ALU operation as a function of these four instruction bits.

TABLE 1 – ALU OPERAND SOURCES

$\overline{E_A}$	I0	$\overline{OE_B}$	ALU Operand R	ALU Operand B
L	L	L	RAM Output A	RAM Output B
L	L	H	RAM Output A	DB_{0-3}
L	H	X	RAM Output A	Q Register
H	L	L	DA_{0-3}	RAM Output B
H	L	H	DA_{0-3}	DB_{0-3}
H	H	X	DA_{0-3}	Q Register

L = Low, H = High, X = don't care.

TABLE 2 – ALU FUNCTIONS

I4	I3	I2	I1	Hex Code	ALU Functions	
L	L	L	L	0	I0 = L	Special Functions
					I0 = H	F_i = High
L	L	L	H	1	F = S minus R minus 1 plus C_n	
L	L	H	L	2	F = R minus S minus 1 plus C_n	
L	L	H	H	3	F = R plus S plus C_n	
L	H	L	L	4	F = S plus C_n	
L	H	L	H	5	F = $\overline{S}$ plus C_n	
L	H	H	L	6	F = R plus C_n	
L	H	H	H	7	F = $\overline{R}$ plus C_n	
H	L	L	L	8	F_i = Low	
H	L	L	H	9	$F_i = \overline{R}_i$ AND S_i	
H	L	H	L	A	$F_i = R_i$ EXCLUSIVE NOR S_i	
H	L	H	H	B	$F_i = R_i$ EXCLUSIVE OR S_i	
H	H	L	L	C	$F_i = R_i$ AND S_i	
H	H	L	H	D	$F_i = R_i$ NOR S_i	
H	H	H	L	E	$F_i = R_i$ NAND S_i	
H	H	H	H	F	$F_i = R_i$ OR S_i	

L = Low, H = High, i = 0 to 3.

MC2903s may be cascaded in either a ripple carry or lookahead carry fashion. When a number of MC2903s are cascaded, each slice must be programmed to be a most significant slice (MSS), intermediate slice (IS), or least significant slice (LSS) of the array. The carry generate, $\overline{G}$, and carry propagate, $\overline{P}$, signals required for a lookahead carry scheme are generated by the MC2903 and are available as outputs of the least signficant and intermediate slices.

The MC2903 also generates a carry-out signal, C_{n+4}, which is generally available as an output of each slice. Both the carry-in, C_n, and carry-out, C_{n+4}, signals are

active High. The ALU generates two other status outputs. These are negative, N, and overflow, OVR. The N output is generally the most significant (sign) bit of the ALU output and can be used to determine positive or negative results. The OVR output indicates that the arithmetic operation being performed exceeds the available two's complement number range. The N and OVR signals are available as outputs of the most significant slice. Thus, the multipurpose $\overline{G}$/N and $\overline{P}$/OVR outputs indicate $\overline{G}$ and $\overline{P}$ at the least significant and intermediate slices, and sign and overflow at the most significant slice. To some extent, the meaning of the C_{n+4}, $\overline{P}$/OVR, and $\overline{G}$/N signals vary with the ALU function being performed. Refer to Table 5 for an exact definition of these four signals as a function of the MC2903 instruction.

ALU Shifter

Under instruction control, the ALU shifter passes the ALU output (F) non-shifted, shifts it up one bit position (2F), or shifts it down one bit position (F/2). Both arithmetic and logical shift operations are possible. An arithmetic shift operation shifts data around the most significant (sign) bit position of the most significant slice, and a logical shift operation shifts data through this bit position (see Figure A). SIO_0 and SIO_3 are bidirectional serial shift inputs/outputs. During a shift-up operation, SIO_0 is generally a serial shift input and SIO_3 a serial shift output. During a shift-down operation, SIO_3 is generally a serial shift input and SIO_0 a serial shift output.

To some extent, the meaning of the SIO_0 and SIO_3 signals is instruction dependent. Refer to Tables 3 and 4 for an exact definiction of these pins.

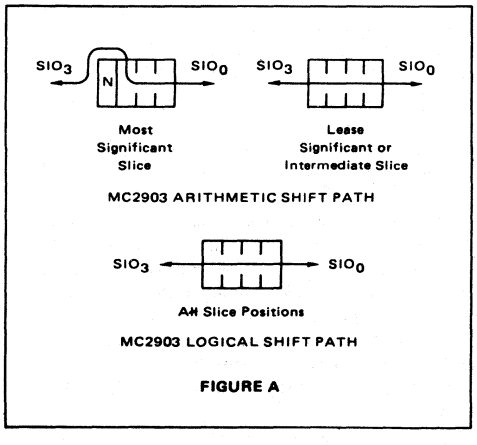


FIGURE A

The ALU shifter also provides the capability to sign extend at slice boundaries. Under instruction control, the SIO_0 (sign) input can be extended through Y0, Y1, Y2, Y3, and propagated to the SIO_3 output.

A cascadable, 5-bit parity generator/checker is designed into the MC2903 ALU shifter and provides ALU error detection capability. Parity for the F0, F1, F2, F3 ALU outputs and SIO_3 input is generated and, under instruction control, is made available at the SIO_0 output. Refer to the MC2903 applications section for a more detailed description of the MC2903 sign extension and parity generation/checking capability.

The instruction inputs determine the ALU shifter operation. Table 4 defines the special functions and the operation the ALU shifter performs for each. When the MC2903 executes instructions other than the nine special functions, the ALU shifter operation is determined by instruction bits I8, I7, I6, I5. Table 3 defines the ALU shifter operations as a function of these four bits.

Q Register

The Q Register is an auxiliary 4-bit register which is clocked on the Low-to-High transition of the CP input. It is intended primarily for use in multiplication and division operations; however, it can also be used as an accumulator or holding register for some applications. The ALU output, F, can be loaded into the Q Register, and/or the Q Register can be selected as the source for the ALU S operand. The shifter at the input to the Q Register provides the capability to shift the Q Register contents up one bit position (2Q) or down one bit position (Q/2). Only logical shifts are performed. QIO_0 and QIO_3 are bidirectional shift serial inputs/outputs. During a Q Register shift-up operation, QIO_0 is a serial shift input and QIO_3 is a serial shift output. During a shift-down operation, QIO_3 is a serial shift inputs and QIO_0 is a serial shift ouptut.

Double-length arithmetic and logical shifting capability is provided by the MC2903. The double-length shift is performed by connecting QIO_3 of the most significant slice to SIO_0 of the least significant slice, and executing an instruction which shifts both the ALU output and the Q Register.

The Q Register and shifter are controlled by the instruction inputs. Table 4 defines the MC2903 special functions and the operations which the Q Register and shifter perform for each. When the MC2903 executes instructions other than the nine special functions, the Q Register and shifter operation is controlled by instruction bits I8, I7, I6, I5. Table 3 defines the Q Register and shifter operation as a function of these four bits.

TABLE 3 – ALU DESTINATION CONTROL FOR I0 OR I1 OR I2 OR I3 OR I4 = High, $\overline{IEN}$ = Low

I8	I7	I6	I5	Hex Code	ALU Shifter Function	SIO_3 Most Sig. Slice	SIO_3 Other Slices	Y3 Most Sig. Slice	Y3 Other Slices	Y2 Most Sig. Slice	Y2 Other Slices	Y1	Y0	SIO_0	$\overline{Write}$	Q Reg and Shifter Function	QIO_3	QIO_0
L	L	L	L	0	Arith F/2 → Y	Input	Input	F3	SIO_3	SIO_3	F3	F2	F1	F0	L	Hold	Hi-Z	Hi-Z
L	L	L	H	1	Log F/2 → Y	Input	Input	SIO_3	SIO_3	F3	F3	F2	F1	F0	L	Hold	Hi-Z	Hi-Z
L	L	H	L	2	Arith F/2 → Y	Input	Input	F3	SIO_3	SIO_3	F3	F2	F1	F0	L	Log Q/2 → Q	Input	Q0
L	L	H	H	3	Log F/2 → Y	Input	Input	SIO_3	SIO_3	F3	F3	F2	F1	F0	L	Log Q/2 → Q	Input	Q0
L	H	L	L	4	F → Y	Input	Input	F3	F3	F2	F2	F1	F0	Parity	L	Hold	Hi-Z	Hi-Z
L	H	L	H	5	F → Y	Input	Input	F3	F3	F2	F2	F1	F0	Parity	H	Log Q/2 → Q	Input	Q0
L	H	H	L	6	F → Y	Input	Input	F3	F3	F2	F2	F1	F0	Parity	H	F → Q	Hi-Z	Hi-Z
L	H	H	H	7	F → Y	Input	Input	F3	F3	F2	F2	F1	F0	Parity	L	F → Q	Hi-Z	Hi-Z
H	L	L	L	8	Arith 2F → Y	F2	F3	F3	F2	F1	F1	F0	SIO_0	Input	L	Hold	Hi-Z	Hi-Z
H	L	L	H	9	Log 2F → Y	F3	F3	F2	F2	F1	F1	F0	SIO_0	Input	L	Hold	Hi-Z	Hi-Z
H	L	H	L	A	Arith 2F → Y	F2	F3	F3	F2	F1	F1	F0	SIO_0	Input	L	Log 2Q → Q	Q3	Input
H	L	H	H	B	Log 2F → Y	F3	F3	F2	F2	F1	F1	F0	SIO_0	Input	L	Log 2Q → Q	Q3	Input
H	H	L	L	C	F → Y	F3	F3	F3	F3	F2	F2	F1	F0	Hi-Z	H	Hold	Hi-Z	Hi-Z
H	H	L	H	D	F → Y	F3	F3	F3	F3	F2	F2	F1	F0	Hi-Z	H	Log 2Q → Q	Q3	Input
H	H	H	L	E	SIO_0 → Y0, Y1, Y2, Y3	SIO_0	SIO_0	SIO_0	SIO_0	SIO_0	SIO_0	SIO_0	SIO_0	Input	L	Hold	Hi-Z	Hi-Z
H	H	H	H	F	F → Y	F3	F3	F3	F3	F2	F2	F1	F0	Hi-Z	L	Hold	Hi-Z	Hi-Z

Parity = F3 ∀ F2 ∀ F1 ∀ F0 ∀ SIO_3

∀ = Exclusive OR

L = Low H = High Hi-Z = High Impedance

Output Buffers

The DB and Y ports are bidirectional I/O ports driven by three-state output buffers with external output enable controls. The Y output buffers are enabled when the $\overline{OE_Y}$ input is Low and are in the high-impedance state when $\overline{OE_Y}$ is High. Likewise, the DB output buffers are enabled when the $\overline{OE_B}$ input is Low and in the high-impedance state when $\overline{OE_B}$ is High.

The zero, Z, pin is an open collector input/output that can be wire-ORed between slices. As an output it can be used as a zero detect status flag and generally indicates that the Y0–3 pins are all Low, whether they are driven from teh Y output buffers or from an external source connected to the Y0–3 pins. To some extent the meaning of this signal varies with the instruction being performed. Refer to Table 5 for an exact definition of this signal as a function of the MC2903 instruction.

Instruction Decoder

The Instruction Decoder generates required internal control signals as a function of the nine instruction inputs, I0–8: the Instruction Enable input, $\overline{IEN}$, the $\overline{LSS}$ input, and the $\overline{Write}/\overline{MSS}$ input/output.

The $\overline{Write}$ output is Low when an instruction which writes data into the RAM is being executed. Refer to Tables 3 and 4 for a definition of the $\overline{Write}$ output as a function of the MC2903 instruction inputs.

When $\overline{IEN}$ is High, the $\overline{Write}$ output is forced High and the Q Register and Sign Compare Flip-Flop contents are preserved. When $\overline{IEN}$ is Low, the $\overline{Write}$ output is enabled and the Q Register and Sign Compare Flip-Flop can be written according to the MC2903 instruction. The Sign Compare Flip-Flop is an on-chip flip-flop which is used during an MC2903 divide operation (see Figure B).

Programming the MC2903 Slice Position

Tying the $\overline{LSS}$ input Low programs the slice to operate as a least signficant slice (LSS) and enables the $\overline{Write}$ output signal onto the $\overline{Write}/\overline{MSS}$ bidirectional I/O pin. When $\overline{LSS}$ is tied High, the $\overline{Write}/\overline{MSS}$ pin becomes an input pin; tying the $\overline{Write}/\overline{MSS}$ pin High programs the slice to operate as an intermediate slice (IS) and tying it Low programs the slice to operate as a most significant slice (MSS).

TABLE 4. SPECIAL FUNCTIONS: $\overline{I0} = \overline{I1} = \overline{I2} = \overline{I3} = \overline{I4}$ = Low, $\overline{IEN}$ = Low

$\overline{I8}$ $\overline{I7}$ $\overline{I6}$ $\overline{I5}$	Hex Code	Special Function	ALU Function	ALU Shifter Function	SIO_3 Most Sig. Slice	SIO_3 Other Slices	SIO_0	Q Reg and Shifter Function	QIO_3	QIO_0	$\overline{Write}$
L L L L	0	Unsigned Multiply	F = S + C_n if Z = L F = R + S + C_n if Z = H	Log F/2 → Y (Note 1)	Hi-Z	Input	F0	Log Q/2 → Q	Input	Q0	L
L L H L	2	Two's Complement Multiply	F = S + C_n if Z = L F = R + S + C_n if Z = H	Log F/2 → Y (Note 2)	Hi-Z	Input	F0	Log Q/2 → Q	Input	Q0	L
L H L L	4	Increment by One or Two	F = S + 1 + C_n	F → Y	Input	Input	Parity	Hold	Hi-Z	Hi-Z	L
L H L H	5	Sign/Magnitude-Two's Complement	F = $\overline{S}$ + C_n if Z = L F = $\overline{S}$ + C_n if Z = H	F → Y (Note 3)	Input	Input	Parity	Hold	Hi-Z	Hi-Z	L
L H H L	6	Two's Complement Multiply, Last Cycle	F = S + C_n if Z = L F = S − R − 1 + C_n if Z = H	Log F/2 → Y (Note 2)	Hi-Z	Input	F0	Log Q/2 → Q	Input	Q0	L
H L L L	8	Single Length Normalize	F = S + C_n	F → Y	F3	F3	Hi-Z	Log 2Q → Q	Q3	Input	L
H L H L	A	Double Length Normalize and First Divide Op	F = S + C_n	Log 2F → Y	R3 ∀ F3	F3	Input	Log 2Q → Q	Q3	Input	L
H H L L	C	Two's Complement Divide	F = S + R + C_n if Z = L F = S − R − 1 + C_n if Z = H	Log 2F → Y	$\overline{R3 \forall F3}$	F3	Input	Log 2Q → Q	Q3	Input	L
H H H L	E	Two's Complement Divide, Correction and Remainder	F = S + R + C_n if Z = L F = S − R − 1 + C_n if Z = H	F → Y	F3	F3	Hi-Z	Log 2Q → Q	Q3	Input	L

NOTES 1. At the most significant slice only, the C_{n+4} signal is internally gated to the Y3 output.
2. At the most significant slice only, F3 ∀ OVR is internally gated to the Y3 output.
3. At the most significant slide only, S3 ∀ F3 is generated at the Y3 output.
4. Op codes 1, 3, 7, 9, B, D, and F are reserved for future use.

L = Low, H = High, X = Don't Care, Hi-Z = High Impedance, ∀ = Exclusive OR, Parity = SIO_3 ∀ F3 ∀ F2 ∀ F1 ∀ F0

MC2903 SPECIAL FUNCTIONS

The MC2903 provides nine special functions which facilitate the implementation of the following operations:

- Single- and double-length normalization
- Two's complement division
- Unsigned and two's complement multiplication
- Conversion between two's complement and sign/magnitude representation
- Incrementation by one or two

Table 4 defines these special functions.

The single-length and double-length normalization functions can be used to adjust a single-precision or double-precision floating point number in order to bring its mantissa within a specified range.

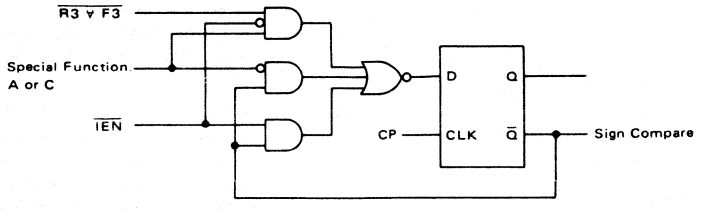
Three special functions which can be used to perform a two's complement, non-restoring divide operation are provided by the MC2903. These functions provide both single- and double-precision divide operations and can be performed in "n" clock cycles, where n is the number of bits in the quotient.

The unsigned multiply special function and the two two's complement multiply special functions can be used to multiply two n-bit, unsigned or two's complement numbers, respectively, in n clock cycles. These functions utilize the conditional add and shift algorithm. During the last cycle of the two's complement multiplication, a conditional subtraction, rather than addition, is performed because the sign bit of the multiplier carries negative weight.

The sign/magnitude-two's complement special function can be used to convert number representation systems. A number expressed in sign/magnitude representation can be converted to the two's complement representation, and vice-versa, in one clock cycle.

The increment-by-one-or-two special function can be used to increment an unsigned or two's complement number by one or two. This is useful in 16-bit word, byte-addressable machines, where the word addresses are multiples of two.

Refer to MC2903 applications section for a more detailed description of these special functions.



The sign compare signal appears at the Z output of the most significant slice during special functions C, D and E, F. Refer to Table 5.

FIGURE B – SIGN COMPARE FLIP-FLOP

TABLE 5 – MC2903 STATUS OUTPUTS

Hex I8 I7 I6 I5	Hex I4 I3 I2 I1	I0	Gi (i = 0 to 3)	Pi (i = 0 to 3)	C_{n+4}	$\overline{F}$/OVR Most Sig. Slice	$\overline{F}$/OVR Other Slices	$\overline{G}$/N Most Sig. Slice	$\overline{G}$/N Other Slices	Z Most Sig. Slice	Z Intermediate Slice	Z Least Sig. Slice
X	0	H	0	1	0	0	0	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	1	X	$\overline{R}_i \wedge S_i$	$\overline{R}_i \vee S_i$	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	2	X	$R_i \wedge \overline{S}_i$	$R_i \vee \overline{S}_i$	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	3	X	$R_i \wedge S_i$	$R_i \vee S_i$	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	4	X	0	S_i	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	5	X	0	$\overline{S}_i$	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	6	X	0	R_i	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	7	X	0	$\overline{R}_i$	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	8	X	0	1	0	0	0	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	9	X	$\overline{R}_i \wedge S_i$	1	0	0	0	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	A	X	$R_i \wedge S_i$	$R_i \vee S_i$	0	0	0	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	B	X	$\overline{R}_i \wedge S_i$	$\overline{R}_i \vee S_i$	0	0	0	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	C	X	$R_i \wedge S_i$	1	0	0	0	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	D	X	$\overline{R}_i \wedge \overline{S}_i$	1	0	0	0	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	E	X	$R_i \wedge S_i$	1	0	0	0	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
X	F	X	$\overline{R}_i \wedge \overline{S}_i$	1	0	0	0	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
0	0	L	0 if Z = L $R_i \wedge S_i$ if Z = H	S_i if Z = L $R_i \vee S_i$ if Z = H	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F3	$\overline{G}$	Input	Input	Q0
2	0	L	0 if Z = L $R_i \wedge S_i$ if Z = H	S_i if Z = L $R_i \vee S_i$ if Z = H	$G \vee PC_n$	$C_{n+4} \forall C_{n+4}$	$\overline{P}$	F3	$\overline{G}$	Input	Input	Q0
4	0	L	See Note 1	See Note 2	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F3	$\overline{G}$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$	$\forall 0 \forall 1 \forall 2 \forall 3$
5	0	L	0	S_i if Z = L $\overline{S}_i$ if Z = H	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$		F3 if Z = L F3 $\forall$ S3 if Z = H	$\overline{G}$	S3	Input	Input
6	0	L	0 if Z = L $\overline{R}_i \wedge S_i$ if Z = H	S_i if Z = L $\overline{R}_i \vee S_i$ if Z = H	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F3	$\overline{G}$	Input	Input	Q0
8	0	L	0	S_i	See Note 3	Q2 $\forall$ Q1	$\overline{P}$	Q3	$\overline{G}$	$\overline{Q}0\overline{Q}1\overline{Q}2\overline{Q}3$	$\overline{Q}0\overline{Q}1\overline{Q}2\overline{Q}3$	$\overline{Q}0\overline{Q}1\overline{Q}2\overline{Q}3$
A	0	L	0	S_i	See Note 4	F2 $\forall$ F1	$\overline{P}$	F3	$\overline{G}$	See Note 5	See Note 5	See Note 5
C	0	L	$R_i \wedge S_i$ if Z = L $\overline{R}_i \wedge S_i$ if Z = H	$R_i \vee S_i$ if Z = L $\overline{R}_i \vee S_i$ if Z = H	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F3	$\overline{G}$	Sign Compare FF Output	Input	Input
E	0	L	$R_i \wedge S_i$ if Z = L $\overline{R}_i \wedge S_i$ if Z = H	$R_i \vee S_i$ if Z = L $\overline{R}_i \vee S_i$ if Z = H	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F3	$\overline{G}$	Sign Compare FF Output	Input	Input

NOTES 1. If $\overline{LSS}$ is Low, G0 = S0 and G1,2,3 = 0. If $\overline{LSS}$ is High, G0,1,2,3 = 0.
2. If $\overline{LSS}$ is Low, P0 = 1 and P1,2,3 = S1,2,3. If $\overline{LSS}$ is High, $P_i = S_i$.
3. At the most significant slice, $C_{n+4} = Q3 \forall Q2$. At other slices, $C_{n+4} = G \vee PC_n$.
4. At the most significant slice, $C_{n+4} = F3 \forall F2$. At other slices, $C_{n+4} = G \vee PC_n$.
5. $Z = \overline{Q}0\overline{Q}1\overline{Q}2\overline{Q}3\overline{F}0\overline{F}1\overline{F}2\overline{F}3$.

L = Low = 0. H = High = 1. $\vee$ = OR. $\wedge$ = AND. $\forall$ = Exclusive OR.
$P = P3P2P1P0$
$G = G3 \vee G2P3 \vee G1P2P3 \vee G0P1P2P3$
$C_{n+3} = G2 \vee G1P2 \vee G0P1P2 \vee C_n P0P1P2$

PIN DEFINITIONS

A_{0-3} Four RAM address inputs which contain the address of the RAM word appearing at the RAM A output port.

B_{0-3} Four RAM address inputs which contain the address of the RAM word appearing at the RAM B output port and into which new data is written when the $\overline{WE}$ input and the CP input are Low.

$\overline{WE}$ The RAM write enable input. If $\overline{WE}$ is Low, data at the Y I/O port is written into the RAM when the CP input is Low. When $\overline{WE}$ is High, writing data into the RAM is inhibited.

DA_{0-3} A four-bit external data input which can be selected as one of the MC2903 ALU operand sources; DA_0 is the least significant bit.

$\overline{EA}$ A control input which, when High, selects DA_{0-3}, and, when Low, selects RAM output A as the ALU R operand.

DB_{0-3} A four-bit external data input/output. Under control of the $\overline{OE_B}$ input, RAM output port B can be directly read on these lines, or input data on these lines can be selected as the ALU S operand.

$\overline{OE_B}$ A control input which, when Low, enables RAM output B onto the DB_{0-3} lines and, when High, disables the RAM output B three-state buffers.

C_n The carry-in input to the MC2903 ALU.

I_{0-8} The nine instruction inputs used to select the MC2903 operation to be performed.

$\overline{IEN}$ The instruction enable input which, when Low, enables the $\overline{Write}$ output and allows the Q Register and the Sign Compare flip-flop to be written. When $\overline{IEN}$ is High, the $\overline{Write}$ output is forced High and the Q Register and Sign Compare flip-flop are in the hold mode.

C_{n+4} This output generally indicates the carry-out of the MC2903 ALU. Refer to Table 5 for an exact definition of this pin.

$\overline{G}/N$ A multipurpose pin which indicates the carry generate, $\overline{G}$, function at the least significant and intermediate slices, and generally indicates the sign, N, of the ALU result at the most significant slice. Refer to Table 5 for an exact definition of this pin.

$\overline{P}/OVR$ A multipurpose pin which indicates the carry propagate, $\overline{P}$, function at the least significant and intermediate slices, and indicates the conventional two's complement overflow, OVR, signal at the most significant slice. Refer to Table 5 for an exact definition of this pin.

Z An open-collector input/output pin which, when High, generally indicates the Y0–3 outputs are all Low. For some special functions, Z is used as an input pin. Refer to Table 5 for an exact definition of this pin.

SIO_0, SIO_3 Bidirectional serial shift inputs/outputs for the ALU shifter. During a shift-up operation, SIO_0 is an input and SIO_3 an output. During a shift-down operation, SIO_3 is an input and SIO_0 is an output. Refer to Tables 3 and 4 for an exact definition of these pins.

QIO_0, QIO_3 Bidirectional serial shift inputs/outputs for the Q shifter which operate like SIO_0 and SIO_3. Refer to Tables 3 and 4 for an exact definition of these pins.

$\overline{LSS}$ An input pin which, when tied Low, programs the chip to act as the least significant slice (LSS) of an MC2903 array and enables the $\overline{Write}$ output onto the $\overline{Write}/\overline{MSS}$ pin. When $\overline{LSS}$ is tied High, the chip is programmed to operate as either an intermediate or most significant slice and the $\overline{Write}$ output buffer is disabled.

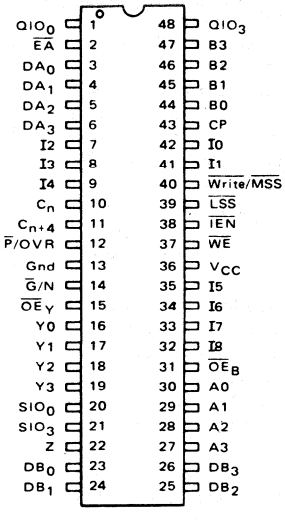
$\overline{Write}/\overline{MSS}$ When $\overline{LSS}$ is tied Low, the $\overline{Write}$ output signal appears at this pin; the $\overline{Write}$ signal is Low when an instruction which writes data into the RAM is being executed. When $\overline{LSS}$ is tied High, $\overline{Write}/\overline{MSS}$ is an input pin; tying it High programs the chip to operate as an intermediate slice (IS) and tying it Low programs the chip to operate as the most significant slice (MSS).

Y_{0-3} Four data inputs/outputs of the MC2903. Under control of the $\overline{OE_Y}$ input, the ALU shifter output data can be enabled onto these lines, or these lines can be used as data inputs when external data is written directly into the RAM.

$\overline{OE_Y}$ A control input which, when Low, enables the ALU shifter output data onto the Y0–3 lines and, when High, disables the Y0–3 three-state output buffers.

CP The clock input to the MC2903. The Q Register and Sign Compare flip-flop are clocked on the Low-to-High transition of the CP signal. When enabled by $\overline{WE}$, data is written in the RAM when CP is Low.

PIN ASSIGNMENT



MOTOROLA

MC2905

QUAD TWO-INPUT OC BUS TRANSCEIVER WITH THREE-STATE RECEIVER

The MC2905 is a high-performance, low-power Schottky bus transceiver intended for bipolar or MOS microprocessor system applicatons. The device consists of four D-type edge-triggered flip-flops with a built-in two-input multiplexer on each. The flip-flop outputs are connected to four open-collector bus drivers. Each bus driver is internally connected to one input of a differential amplifier in the receiver. The four receiver differential amplifier outputs drive four D-type latches that feature three-state outputs.

This LSI bus transceiver is fabricated using advanced low-power Schottky processing. All inputs (except the Bus inputs) are one LS unit load. The open-collector bus output can sink up to 100 mA at 0.8 V maximum. The Bus input differential amplifier contains disconnect protection diodes such that the bus is fail-safe when power is not applied. The bus enable input ($\overline{BE}$) is used to force the driver outputs to the high-impedance state. When $\overline{BE}$ is High, the driver is disabled. The open-collector structure of the driver allows wired-OR operations to be performed on the bus.

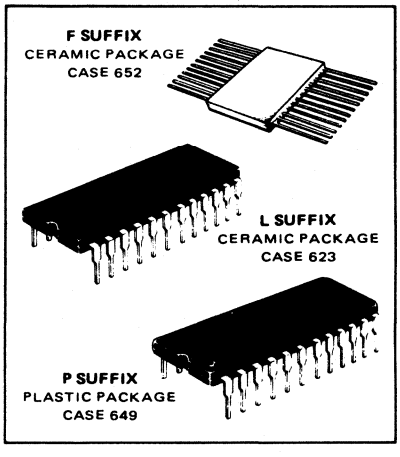
The input register consists of four D-type flip-flops with a buffered common clock and a two-input multiplexer at the input of each flip-flop. A common select input (S) controls the four multiplexers. When S is Low, the A_n data is stored in the register and when S is High, the B_n data is stored. The buffered common clock (DRCP) enters the data into this driver register on the low-to-high transition.

Data from the A or B inputs is inverted at the Bus output. Likewise, data at the Bus input is inverted at the receiver output. Thus, data is noninverted from driver input to receiver output. The four receivers each feature a built-in D-type latch that is controlled from the buffered receiver latch enable ($\overline{RLE}$) input. When the $\overline{RLE}$ input is Low, the latch is open and the receiver outputs will follow the bus inputs (Bus data inverted and $\overline{OE}$ Low). When the $\overline{RLE}$ input is High, the latch will close and retain the present data regardless of the bus input. The four latches have three-state outputs and are controlled by a buffered common three-state control ($\overline{OE}$) input. When $\overline{OE}$ is High, the receiver outputs are in the high-impedance state.

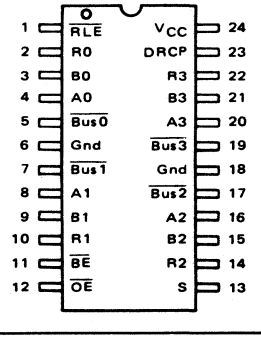
FEATURES

- Quad High-speed LSI Bus-transceiver
- Open-collector Bus Driver
- Two-port Input to D-type Register on Driver
- Bus Driver Output Can Sink 100 mA at 0.8 V Max
- Receiver Has Output Latch for Pipeline Operation
- Three-state Receiver Outputs Sink 12 mA
- Advanced Low-power Schottky Processing
- 100% Reliability Assurance Testing in Compliance With MIL-STD-883

TTL
QUAD TWO-INPUT
OC BUS TRANSCEIVER
WITH THREE-STATE RECEIVER



PIN ASSIGNMENT



ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Molded DIP	0°C to +70°C	MC2905PC
Hermetic DIP	0°C to +70°C	MC2905LC
Hermetic DIP	-55°C to +125°C	MC2905LM
Hermetic Flat Pack	-55°C to +125°C	MC2905FM

MAXIMUM RATINGS (above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential	−0.5 V to +7.0 V
DC Voltage Applied to Outputs for High Output State	−0.5 V to +V_{CC}max
DC Input Voltage	−0.5 V to +5.5 V
DC Output Current, into Outputs (Except Bus)	30 mA
DC Output Current, into Bus	200 mA
DC Input Current	−30 mA to +5.0 mA

ELECTRICAL CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise noted)

MC2905XC − T_A = 0°C to +70°C, V_{CC} = 5.0 V ±5% (Commercial), Min = 4.75 V, Max = 5.25 V
MC2905XM − T_A = −55°C to +125°C, V_{CC} = 5.0 V ±10% (Military), Min = 4.5 V, Max = 5.5 V

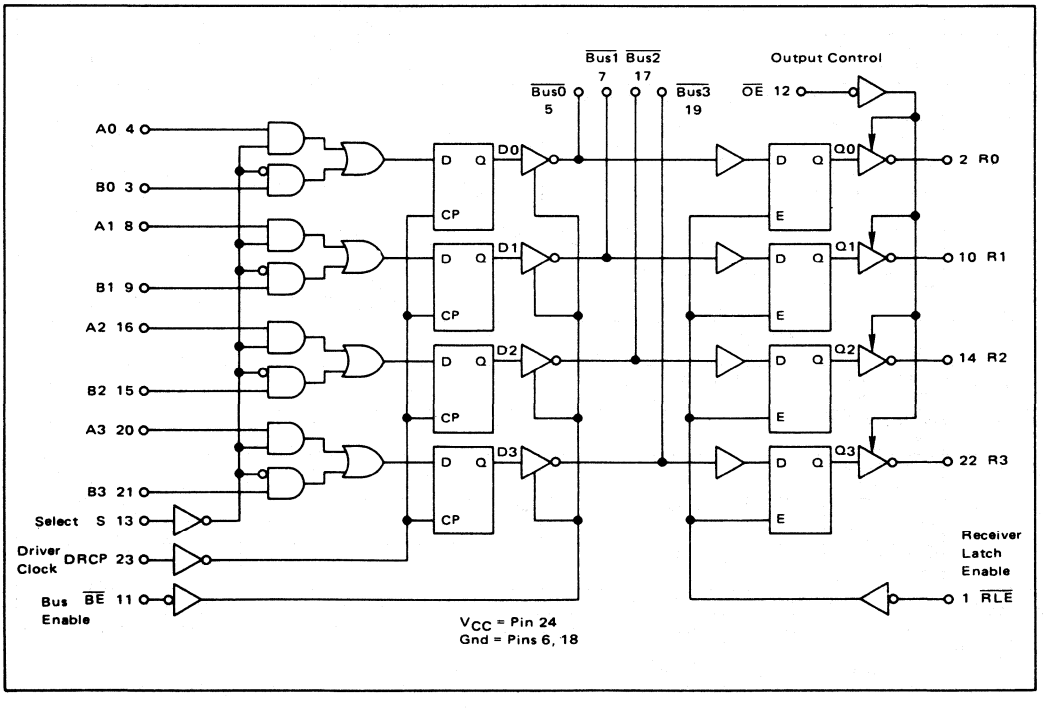
Parameter	Description	Test Conditions (Note 1)			Min	Typ (Note 2)	Max	Unit
BUS INPUT/OUTPUT CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE								
V_{OL}	Bus Output Low Voltage	V_{CC} = Min	I_{OL} = 40 mA			0.32	0.5	Volts
			I_{OL} = 70 mA			0.41	0.7	
			I_{OL} = 100 mA			0.55	0.8	
I_O	Bus Leakage Current	V_{CC} = Max	V_O = 0.4 V				−50	μA
			V_O = 4.5 V	Military			200	
				Commercial			100	
I_{off}	Bus Leakage Current (Power off)	V_O = 4.5 V					100	μA
V_{TH}	Receiver Input High Threshold	Bus Enable = 2.4 V		Military	2.4	2.0		Volts
				Commercial	2.3	2.0		
V_{TL}	Receiver Input Low Threshold	Bus Enable = 2.4 V		Military		2.0	1.5	Volts
				Commercial		2.0	1.6	
DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE								
V_{OH}	Receiver Output High Voltage	V_{CC} = Min, V_{in} = V_{IL} or V_{IH}	Military, I_{OH} = −1.0 mA		2.4	3.4		Volts
			Commercial, I_{OH} = −2.6 mA		2.4	3.4		
V_{OL}	Receiver Output Low Voltage	V_{CC} = Min, V_{in} = V_{IL} or V_{IH}	I_{OL} = 4.0 mA			0.27	0.4	Volts
			I_{OL} = 8.0 mA			0.32	0.45	
			I_{OL} = 12 mA			0.37	0.5	
V_{IH}	Input High Level (Except Bus)	Guaranteed input logical High for all inputs			2.0			Volts
V_{IL}	Input Low Level (Except Bus)	Guaranteed input logical Low for all inputs		Military			0.7	Volts
				Commercial			0.8	
V_I	Input Clamp Voltage (Except Bus)	V_{CC} = Min, I_{in} = −18 mA					−1.2	Volts
I_{IL}	Input Low Current (Except Bus)	V_{CC} = Max, V_{in} = 0.4 V					−0.36	mA
I_{IH}	Input High Current (Except Bus)	V_{CC} = Max, V_{in} = 2.7 V					20	μA
I_{in}	Input High Current (Except Bus)	V_{CC} = Max, V_{in} = 5.5 V					100	μA
I_O	Receiver Off-State Output Current	V_{CC} = Max	V_O = 2.4 V				20	μA
			V_O = 0.4 V				−20	
I_{SC}	Receiver Output Short Circuit Current	V_{CC} = Max			−12		−65	mA
I_{CC}	Power Supply Current	V_{CC} = Max, All inputs = Gnd				69	105	mA

NOTES: 1. For conditions shown as Min or Max, use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0 V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.

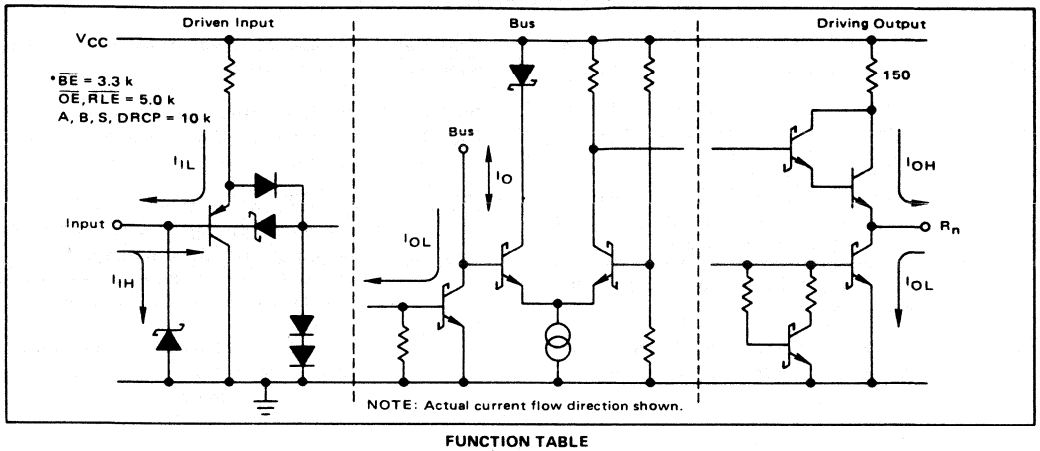
SWITCHING CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

Parameter	Description	Test Conditions	MC2905XM			MC2905XC			Unit
			Min	Typ (Note 2)	Max	Min	Typ (Note 2)	Max	
t_{PHL}	Driver Clock (DRCP) to Bus	C_L (Bus) = 50 pF	–	21	40	–	21	36	ns
t_{PLH}		R_L (Bus) = 50 Ω	–	21	40	–	21	36	
t_{PHL}	Bus Enable ($\overline{BE}$) to Bus		–	13	26	–	13	23	ns
t_{PLH}			–	13	26	–	13	23	
t_s	Data Inputs (A or B)	C_L = 15 pF	25	–	–	23	–	–	ns
t_h		R_L = 2.0 k	8.0	–	–	7.0	–	–	
t_s	Select Input (S)		33	–	–	30	–	–	ns
t_h			8.0	–	–	7.0	–	–	
t_{PW}	Driver Clock (DRCP) Pulse Width (High)		28	–	–	25	–	–	ns
t_{PLH}	Bus to Receiver Output (Latch Enable)		–	18	37	–	18	34	ns
t_{PHL}			–	18	37	–	18	34	
t_{PLH}	Latch Enable to Receiver Output		–	21	37	–	21	34	ns
t_{PHL}			–	21	37	–	21	34	
t_s	Bus to Latch Enable ($\overline{RLE}$)		21	–	–	18	–	–	ns
t_h			7.0	–	–	5.0	–	–	
t_{ZH}	Output Control to Receiver Output		–	14	28	–	14	25	ns
t_{ZL}			–	14	28	–	14	25	
t_{HZ}	Output Control to Receiver Output	C_L = 5.0 pF	–	14	28	–	14	25	ns
t_{LZ}		R_L = 2.0 k	–	14	28	–	14	25	

LOGIC DIAGRAM



INPUT/OUTPUT CURRENT INTERFACE CONDITIONS

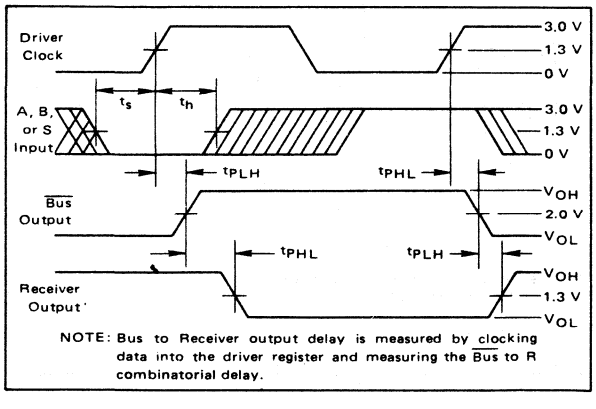


FUNCTION TABLE

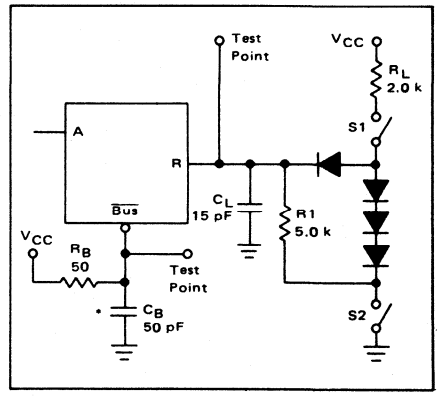
Inputs							Internal to Device		Bus	Output	
S	A_n	B_n	DRCP	$\overline{BE}$	$\overline{RLE}$	$\overline{OE}$	D_n	Q_n	$\overline{Bus_n}$	R_n	Function
X	X	X	X	H	X	X	X	X	Z	X	Driver output disable
X	X	X	X	X	X	H	X	X	X	Z	Receiver output disable
X	X	X	X	H	L	L	X	L	L	H	Driver output disable
X	X	X	X	H	L	L	X	H	H	L	and receive data via Bus input
X	X	X	X	X	H	X	X	NC	X	X	Latch received data
L	L	X	↑	X	X	X	L	X	X	X	Load driver register
L	H	X	↑	X	X	X	H	X	X	X	
H	X	L	↑	X	X	X	L	X	X	X	
H	X	H	↑	X	X	X	H	X	X	X	
X	X	X	L	X	X	X	NC	X	X	X	No driver clock restrictions
X	X	X	H	X	X	X	NC	X	X	X	
X	X	X	X	L	X	X	L	X	H	X	Drive Bus
X	X	X	X	L	X	X	H	X	L	X	

H = High Z = High impedance X = Don't care
L = Low NC = No change ↑ = Low-to-high transition

SWITCHING WAVEFORMS



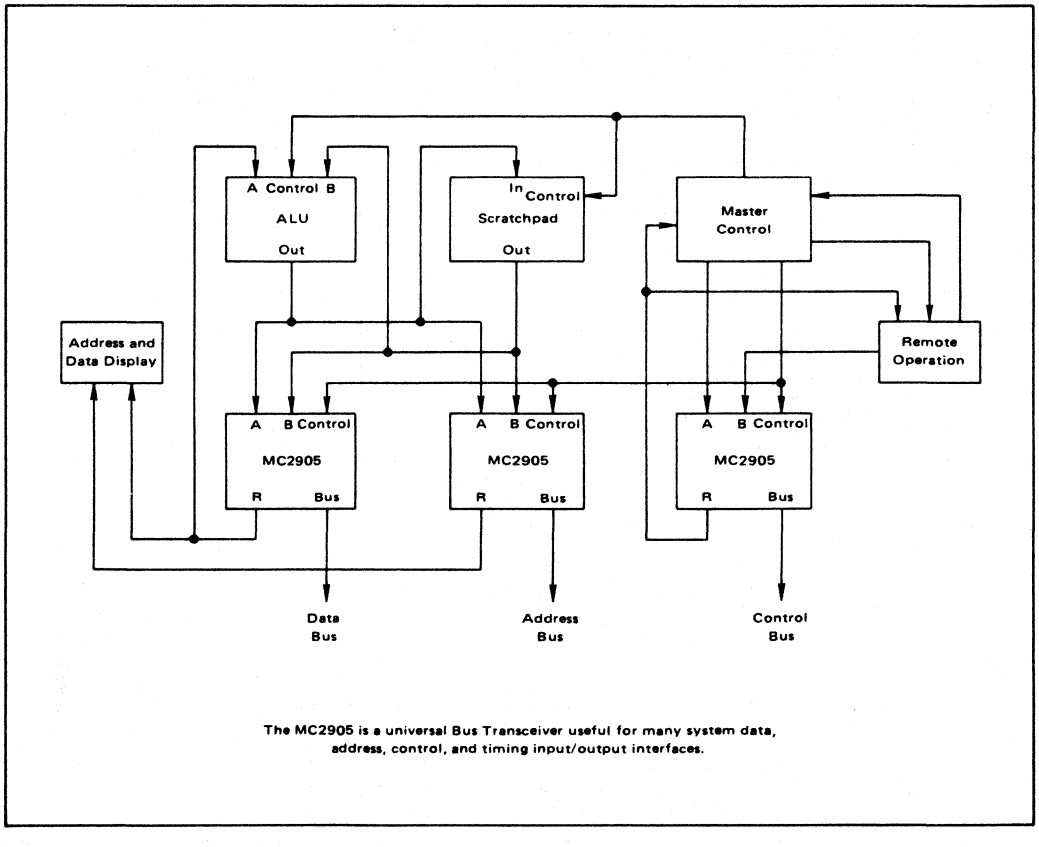
LOAD TEST CIRCUIT



DEFINITIONS OF FUNCTIONAL TERMS

A0, A1, A2, A3	The "A" word data input into the two input multiplexers of the driver register.
B0, B1, B2, B3	The "B" word data input into the two input multiplexers of the driver register.
S	Select. When the select input is Low, the A data word is applied to the driver register. When the select input is High, the B word is applied to the driver register.
DRCP	Driver Clock Pulse. Clock pulse for the driver register.
$\overline{BE}$	$\overline{\text{Bus Enable}}$. When the Bus Enable is High, the four drivers are in the high impedance state.
$\overline{Bus0}$, $\overline{Bus1}$, $\overline{Bus2}$, $\overline{Bus3}$	The four driver outputs and receiver inputs (data is inverted).
R0, R1 R2, R3	The four receiver outputs. Data from the bus is inverted while data from the A or B inputs is non-inverted.
$\overline{RLE}$	$\overline{\text{Receiver Latch Enable}}$. When $\overline{RLE}$ is Low, data on the Bus inputs is passed through the receiver latches. When $\overline{RLE}$ is High, the receiver latches are closed and will retain the data independent of all other inputs.
$\overline{OE}$	$\overline{\text{Output Enable}}$. When the $\overline{OE}$ input is High, the four three-state receiver outputs are in the high impedance state.

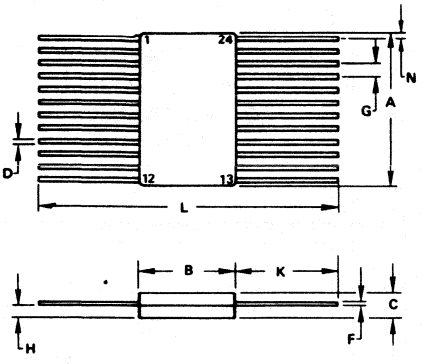
APPLICATIONS



The MC2905 is a universal Bus Transceiver useful for many system data, address, control, and timing input/output interfaces.

PACKAGE DIMENSIONS

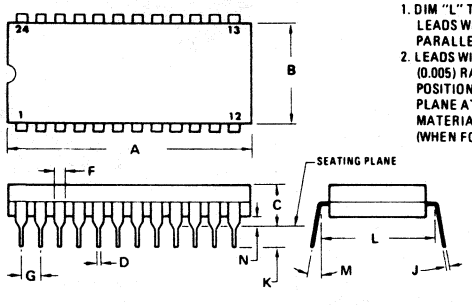
CASE 652



NOTES:
1. LEADS WITHIN 0.25 mm (0.010) TOTAL OF TRUE POSITION AT MAXIMUM MATERIAL CONDITION.

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	14.99	15.49	0.590	0.610
B	9.27	9.91	0.365	0.390
C	1.27	2.03	0.050	0.080
D	0.38	0.48	0.015	0.019
F	0.08	0.15	0.003	0.006
G	1.27 BSC		0.050 BSC	
H	0.69	1.02	0.027	0.040
K	6.35	9.40	0.250	0.370
L	21.97	–	0.865	–
N	0.25	0.63	0.010	0.025

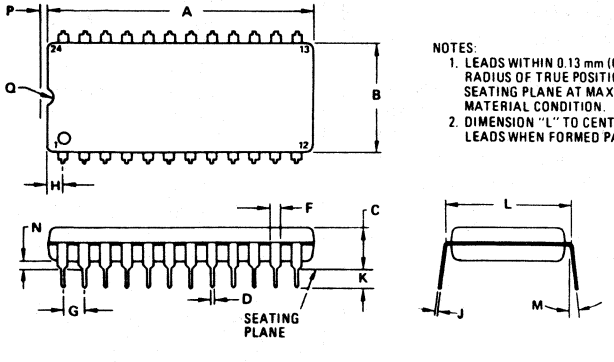
CASE 623



NOTES:
1. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.
2. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION. (WHEN FORMED PARALLEL)

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	31.24	32.26	1.230	1.270
B	12.70	13.72	0.500	0.540
C	4.06	5.59	0.160	0.220
D	0.41	0.51	0.016	0.020
F	1.27	1.52	0.050	0.060
G	2.54 BSC		0.100 BSC	
J	0.20	0.30	0.008	0.012
K	2.29	4.06	0.090	0.160
L	15.24 BSC		0.600 BSC	
M	0°	15°	0°	15°
N	0.51	1.27	0.020	0.050

CASE 649



NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	31.50	32.13	1.240	1.265
B	13.21	13.72	0.520	0.540
C	4.70	5.21	0.185	0.205
D	0.38	0.51	0.015	0.020
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	2.92	3.43	0.115	0.135
L	14.99	15.49	0.590	0.610
M	–	10°	–	10°
N	0.51	1.02	0.020	0.040
P	0.13	0.38	0.005	0.015
Q	0.51	0.76	0.020	0.030

MOTOROLA

MC2906

QUAD TWO-INPUT OC BUS TRANSCEIVER WITH PARITY

The MC2906 is a high-performance, low-power Schottky bus transceiver intended for bipolar or MOS microprocessor system applications. The device consists of four D-type edge-triggered flip-flops with a built-in two-input multiplexer on each. The flip-flop outputs are connected to four open-collector bus drivers. Each bus driver is internally connected to one input of a differential amplifier in the receiver. The four receiver differential amplifier outputs drive four D-type latches. The device also contains a four-bit odd parity checker/generator.

This LSI bus transceiver is fabricated using advanced low-power Schottky processing. All inputs (except the Bus inputs) are one LS unit load. The open-collector bus output can sink up to 100 mA at 0.8 V maximum. The Bus input differential amplifier contains disconnect protection diodes such that the bus is fail-safe when power is not applied. The bus enable input ($\overline{BE}$) is used to force the driver outputs to the high-impedance state. When $\overline{BE}$ is High, the driver is disabled. The open-collector structure of the driver allows wired-OR operations to be performed on the bus.

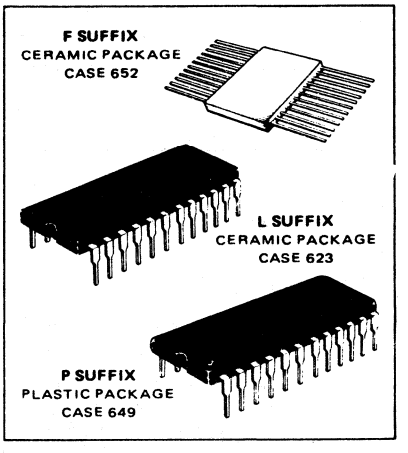
The input register consists of four D-type flip-flops with a buffered common clock and a two-input multiplexer at the input of each flip-flop. A common select input (S) controls the four multiplexers. When S is Low, the A_n data is stored in the register and when S is High, the B_n data is stored. The buffered common clock (DRCP) enters the data into this driver register on the Low-to-High transition.

Data from the A or B inputs is inverted at the Bus output. Likewise, data at the Bus input is inverted at the receiver output. Thus, data is non-inverted from driver input to receiver output. The four receivers each feature a built-in D-type latch that is controlled from the buffered receiver latch enable ($\overline{RLE}$) input. When the $\overline{RLE}$ input is Low, the latch is open and the receiver outputs will follow the bus inputs (Bus data inverted). When the $\overline{RLE}$ input is High, the latch will close and retain the present data regardless of the bus input. The four latches have three-state outputs and are controlled by a buffered common three-state control ($\overline{OE}$) input. When $\overline{OE}$ is High, the receiver outputs are in the high-impedance state.

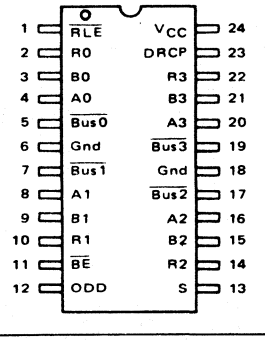
The MC2906 features a built-in four-bit odd parity checker/generator. The bus enable input ($\overline{BE}$) controls whether the parity output is in the generate or check mode. When the bus enable is Low (driver enabled), odd parity is generated based on the A or B field data input to the driver register. When $\overline{BE}$ is High, the parity output is determined by the four latch outputs of the receiver. Thus, if the driver is enabled, parity is generated; and, if the driver is in the high-impedance state, the Bus parity is checked.

- Quad High-speed LSI Bus Transceiver
- Open-collector Bus Driver
- Two-port Input to D-type Register on Driver
- Bus Driver Output Can Sink 100 mA at 0.8 V Max
- Internal Odd 4-bit Parity Checker/Generator
- Receiver Has Output Latch for Pipeline Operation
- Receiver Outputs Sink 12 mA
- Advanced Low-power Schottky Processing
- 100% Reliability Assurance Testing in Compliance With MIL-STD-883

TTL
QUAD TWO-INPUT
OC BUS TRANSCEIVER
WITH PARITY



PIN ASSIGNMENT



ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Molded DIP	0°C to +70°C	MC2906PC
Hermetic DIP	0°C to +70°C	MC2906LC
Hermetic DIP	-55°C to +125°C	MC2906LM
Hermetic Flat Pack	-55°C to +125°C	MC2906FM

MAXIMUM RATINGS (above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential	−0.5 V to +7.0 V
DC Voltage Applied to Outputs for High Output State	−0.5 V to +V_{CC}max
DC Input Voltage	−0.5 V to +5.5 V
DC Output Current, into Outputs (Except Bus)	30 mA
DC Output Current, into Bus	200 mA
DC Input Current	−30 mA to +5.0 mA

ELECTRICAL CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise noted)

MC2906XC − T_A = 0°C to +70°C, V_{CC} = 5.0 V ±5% (Commercial), Min = 4.75 V, Max = 5.25 V
MC2906XM − T_A = −55°C to +125°C, V_{CC} = 5.0 V ±10% (Military), Min = 4.5 V, Max = 5.5 V

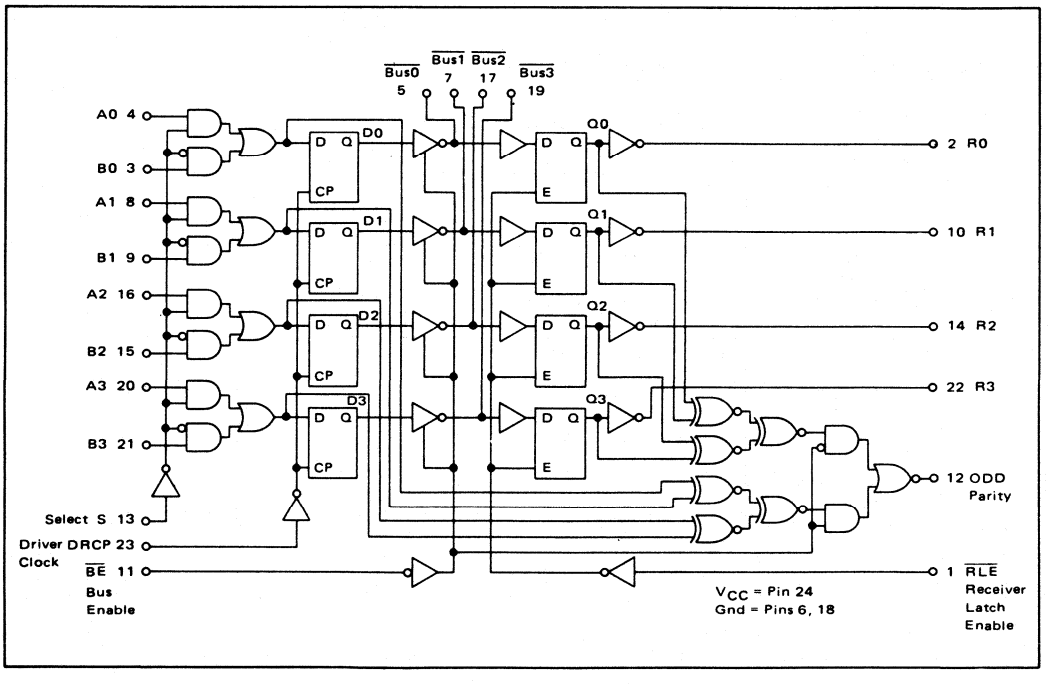
Parameter	Description	Test Conditions (Note 1)		Min	Typ (Note 2)	Max	Unit
BUS INPUT/OUTPUT CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE							
V_{OL}	Bus Output Low Voltage	V_{CC} = Min	I_{OL} = 40 mA		0.32	0.5	Volts
			I_{OL} = 70 mA		0.41	0.7	
			I_{OL} = 100 mA		0.55	0.8	
I_O	Bus Leakage Current	V_{CC} = Max	V_O = 0.4 V			−50	μA
			V_O = 4.5 V, Military			200	
			V_O = 4.5 V, Commercial			100	
I_{off}	Bus Leakage Current (Power off)	V_O = 4.5 V				100	μA
V_{TH}	Receiver Input High Threshold	Bus Enable = 2.4 V	Military	2.4	2.0		Volts
			Commercial	2.3	2.0		
V_{TL}	Receiver Input Low Threshold	Bus Enable = 2.4 V	Military		2.0	1.5	Volts
			Commercial		2.0	1.6	
DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE							
V_{OH}	Receiver Output High Voltage	V_{CC} = Min, V_{in} = V_{IL} or V_{IH}	Military, I_{OH} = −1.0 mA	2.4	3.4		Volts
			Commercial, I_{OH} = −2.6 mA	2.4	3.4		
	Parity Output High Voltage	V_{CC} = Min, I_{OH} = −660 μA, V_{in} = V_{IH} or V_{IL}	Military	2.5	3.4		
			Commercial	2.7	3.5		
V_{OL}	Output Low Voltage (Except Bus)	V_{CC} = Min, V_{in} = V_{IL} or V_{IH}	I_{OL} = 4.0 mA		0.27	0.4	Volts
			I_{OL} = 8.0 mA		0.32	0.45	
			I_{OL} = 12 mA		0.37	0.5	
V_{IH}	Input High Level (Except Bus)	Guaranteed input logical High for all inputs		2.0			Volts
V_{IL}	Input Low Level (Except Bus)	Guaranteed input logical Low for all inputs	Military			0.7	Volts
			Commercial			0.8	
V_I	Input Clamp Voltage (Except Bus)	V_{CC} = Min, I_{in} = −18 mA				−1.2	Volts
I_{IL}	Input Low Current (Except Bus)	V_{CC} = Max, V_{in} = 0.4 V				−0.36	mA
I_{IH}	Input High Current (Except Bus)	V_{CC} = Max, V_{in} = 2.7 V				20	μA
I_{in}	Input High Current (Except Bus)	V_{CC} = Max, V_{in} = 5.5 V				100	μA
I_{SC}	Output Short Circuit Current (Except Bus)	V_{CC} = Max		−12		−65	mA
I_{CC}	Power Supply Current	V_{CC} = Max, All inputs = Gnd			72	105	mA

NOTES: 1. For conditions shown as Min or Max, use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0 V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.

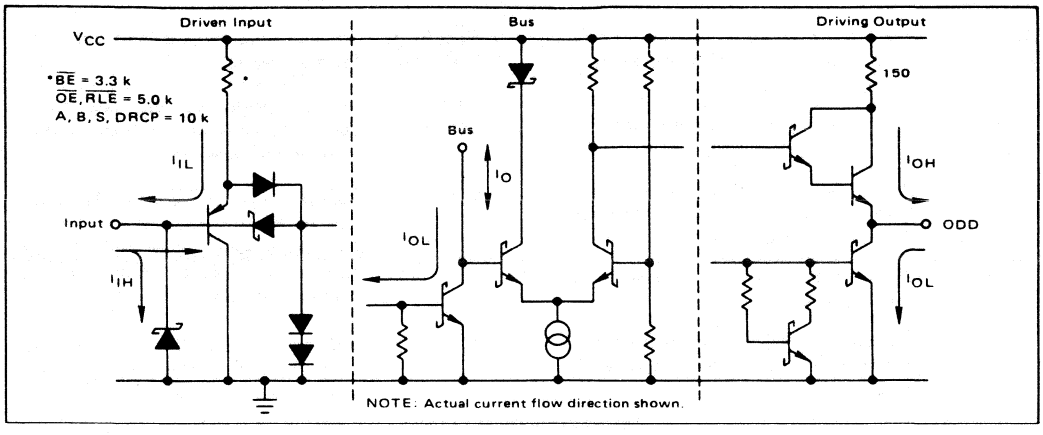
SWITCHING CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

Parameter	Description	Test Conditions	MC2906XM			MC2906XC			Unit
			Min	Typ (Note 2)	Max	Min	Typ (Note 2)	Max	
t_{PHL}	Driver Clock (DRCP) to Bus	C_L (Bus) = 50 pF	–	21	40	–	21	36	ns
t_{PLH}		R_L (Bus) = 50 Ω	–	21	40	–	21	36	
t_{PHL}	Bus Enable ($\overline{BE}$) to Bus		–	13	26	–	13	23	ns
t_{PLH}			–	13	26	–	13	23	
t_s	Data Inputs (A or B)	C_L = 15 pF	25	–	–	23	–	–	ns
t_h		R_L = 2.0 k	8.0	–	–	7.0	–	–	
t_s	Select Input (S)		33	–	–	30	–	–	ns
t_h			8.0	–	–	7.0	–	–	
t_{PW}	Clock Pulse Width (High)		28	–	–	25	–	–	ns
t_{PLH}	Bus to Receiver Output (Latch Enable)		–	18	37	–	18	34	ns
t_{PHL}			–	18	37	–	18	34	
t_{PLH}	Latch Enable to Receiver Output		–	21	37	–	21	34	ns
t_{PHL}			–	21	37	–	21	34	
t_s	Bus to Latch Enable ($\overline{RLE}$)		21	–	–	18	–	–	ns
t_h			7.0	–	–	5.0	–	–	
t_{PLH}	A or B Data to Odd Parity (Driver Enabled)		–	21	40	–	21	36	ns
t_{PHL}			–	21	40	–	21	36	
t_{PLH}	Bus to Odd Parity Output (Driver Inhibited, Latch Enabled)		–	21	40	–	21	36	ns
t_{PHL}			–	21	40	–	21	36	
t_{PLH}	Latch Enable ($RL\overline{E}$) to Odd Parity Output		–	21	40	–	21	36	ns
t_{PHL}			–	21	40	–	21	36	

LOGIC DIAGRAM



INPUT/OUTPUT CURRENT INTERFACE CONDITIONS



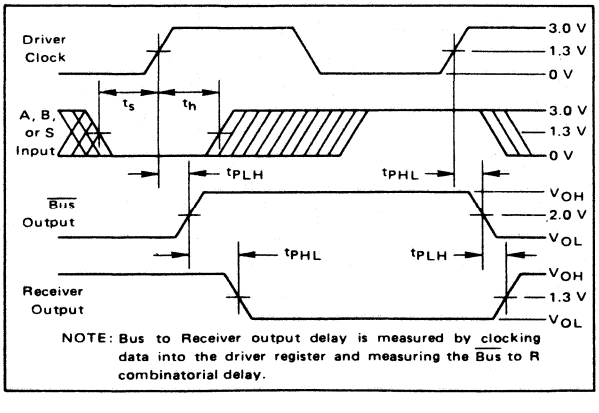
FUNCTION TABLE

Inputs						Internal to Device		Bus	Output	Parity	
S	A_n	B_n	DRCP	$\overline{BE}$	$\overline{RLE}$	D_n	Q_n	$\overline{Bus_n}$	R_n	ODD	Function
X	X	X	X	H	X	X	X	Z	X	Note 1	*Driver output disable
X	X	X	X	H	L	X	L	L	H	Note 1	Driver output disable and receive data via Bus input
X	X	X	X	H	L	X	H	H	L		
X	X	X	X	X	H	X	NC	X	X	X	Latch received data
L	L	X	↑	X	X	L	X	X	X	X	Load driver register
L	H	X	↑	X	X	H	X	X	X	X	
H	X	L	↑	X	X	L	X	X	X	X	
H	X	H	↑	X	X	H	X	X	X	X	
X	X	X	L	X	X	NC	X	X	X	X	No driver clock restrictions
X	X	X	H	X	X	NC	X	X	X	X	
X	X	X	X	L	X	L	X	H	X	Note 2	Drive Bus
X	X	X	X	L	X	H	X	L	X		

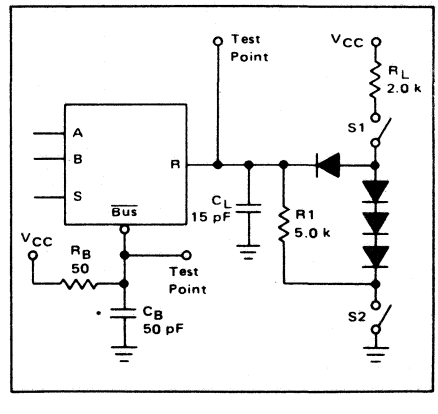
H = High
L = Low
Z = High impedance
NC = No change
X = Don't care
↑ = Low-to-high transition

NOTES: 1. ODD = Q0 ⊕ Q1 ⊕ Q2 ⊕ Q3 when $\overline{BE}$ = H
2. ODD = A0 ⊕ A1 ⊕ A2 ⊕ A3 when $\overline{BE}$ = L, S = L
ODD = B0 ⊕ B1 ⊕ B2 ⊕ B3 when $\overline{BE}$ = L, S = H

SWITCHING WAVEFORMS



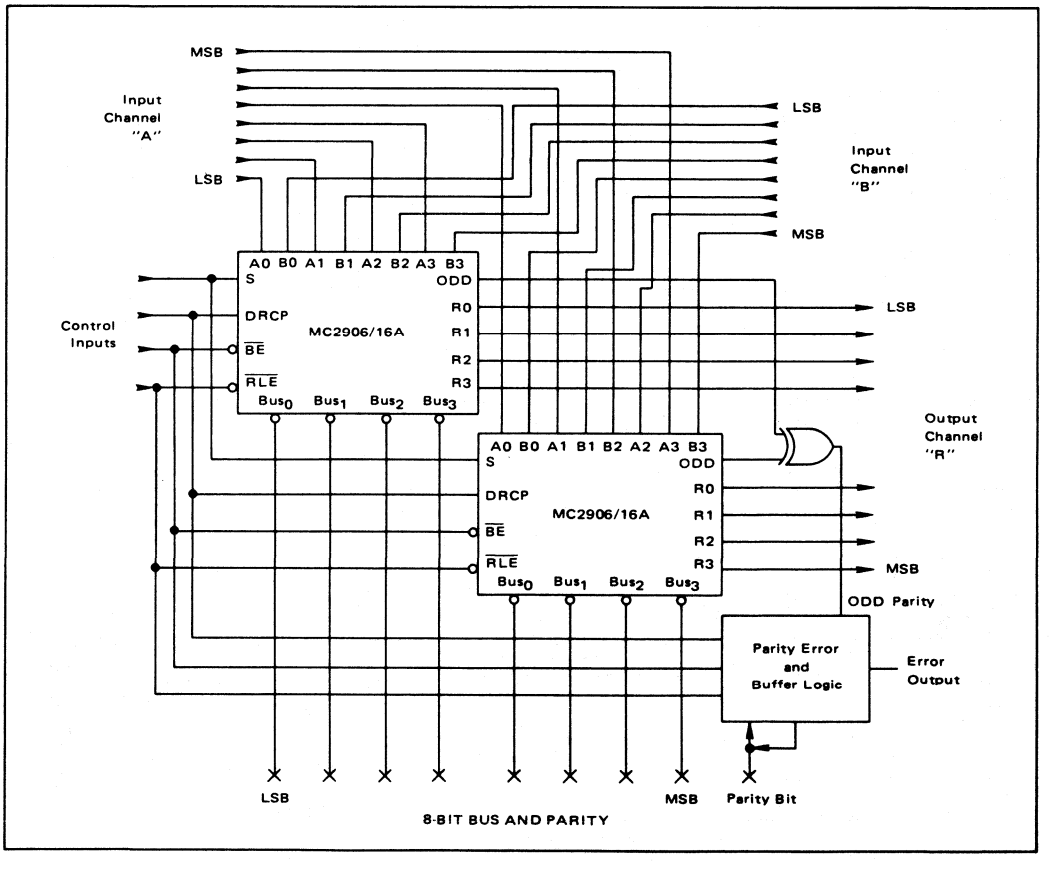
LOAD TEST CIRCUIT



DEFINITIONS OF FUNCTIONAL TERMS

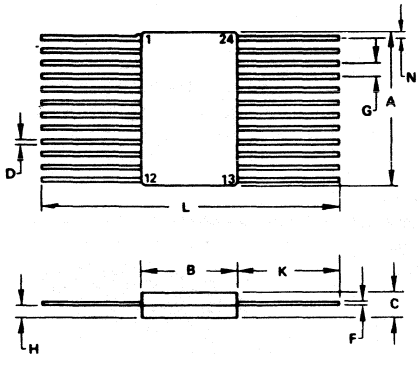
A0, A1, A2, A3	The "A" word data input into the two input multiplexers of the driver register.
B0, B1, B2, B3	The "B" word data input into the two input multiplexers of the driver register.
S	Select. When the select input is Low, the A data word is applied to the driver register. When the select input is High, the B word is applied to the driver register.
DRCP	Driver Clock Pulse. Clock pulse for the driver register.
$\overline{BE}$	$\overline{\text{Bus Enable}}$. When the Bus Enable is High, the four drivers are in the high impedance state.
$\overline{Bus0}$, $\overline{Bus1}$, $\overline{Bus2}$, $\overline{Bus3}$	The four driver outputs and receiver inputs (data is inverted).
R0, R1 R2, R3	The four receiver outputs. Data from the bus is inverted while data from the A or B inputs is non-inverted.
$\overline{RLE}$	$\overline{\text{Receiver Latch Enable}}$. When RLE is Low, data on the Bus inputs is passed through the receiver latches. When RLE is High, the receiver latches are closed and will retain the data independent of all other inputs.
ODD	ODD Parity Output. Generates parity with the driver enabled, checks parity with the driver in the high-impedance state.

8-BIT PERIPHERAL INTERFACE



PACKAGE DIMENSIONS

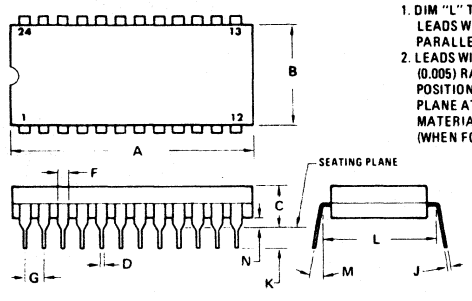
CASE 652



NOTES:
1. LEADS WITHIN 0.25 mm (0.010) TOTAL OF TRUE POSITION AT MAXIMUM MATERIAL CONDITION.

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	14.99	15.49	0.590	0.610
B	9.27	9.91	0.365	0.390
C	1.27	2.03	0.050	0.080
D	0.38	0.48	0.015	0.019
F	0.08	0.15	0.003	0.006
G	1.27 BSC		0.050 BSC	
H	0.69	1.02	0.027	0.040
K	6.35	9.40	0.250	0.370
L	21.97	–	0.865	–
N	0.25	0.63	0.010	0.025

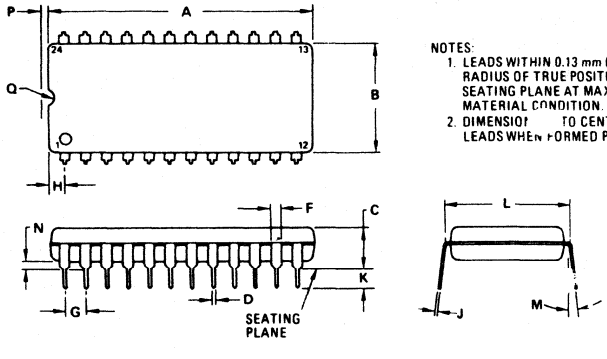
CASE 623



NOTES:
1. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.
2. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION. (WHEN FORMED PARALLEL)

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	31.24	32.26	1.230	1.270
B	12.70	13.72	0.500	0.540
C	4.06	5.59	0.160	0.220
D	0.41	0.51	0.016	0.020
F	1.27	1.52	0.050	0.060
G	2.54 BSC		0.100 BSC	
J	0.20	0.30	0.008	0.012
K	2.29	4.06	0.090	0.160
L	15.24 BSC		0.600 BSC	
M	0°	15°	0°	15°
N	0.51	1.27	0.020	0.050

CASE 649



NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. DIMENSIOI TO CENTER OF LEADS WHEN FORMED PARALLEL.

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	31.50	32.13	1.240	1.265
B	13.21	13.72	0.520	0.540
C	4.70	5.21	0.185	0.205
D	0.38	0.51	0.015	0.020
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	2.92	3.43	0.115	0.135
L	14.99	15.49	0.590	0.610
M	–	10°	–	10°
N	0.51	1.02	0.020	0.040
P	0.13	0.38	0.005	0.015
Q	0.51	0.76	0.020	0.030

MOTOROLA

MC2907

QUAD BUS TRANSCEIVER WITH THREE-STATE RECEIVER AND PARITY

The MC2907 is a high-performance, low-power Schottky bus transceiver intended for bipolar or MOS microprocessor system applications. The device consists of four D-type edge-triggered flip-flops. The flip-flop outputs are connected to four open-collector bus drivers. Each bus driver is internally connected to one input of a differential amplifier in the receiver. The four receiver differential amplifier drive four D-type latches that feature three-state outputs. The device also contains a four-bit odd parity checker/generator.

This LSI bus transceiver is fabricated using advanced low-power Schottky processing. All inputs (except the Bus inputs) are one LS unit load. The open-collector bus output can sink up to 100 mA at 0.8 V maximum. The bus input ($\overline{BE}$) is used to force the driver outputs to the high-impedance state. When $\overline{BE}$ is High, the driver is disabled. The open-collector structure of the driver allows wired-OR operations to be performed on the bus.

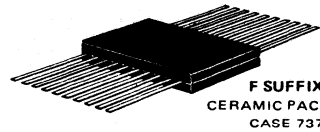
The input register consists of four D-type flip-flops with a buffered common clock. The buffered common clock (DRCP) enters the A_n data into this driver register on the Low-to-High transition.

Data from the A input is inverted at the Bus output. Likewise, data at the Bus input is inverted at the receiver output. Thus, data is non-inverted from driver input to receiver output. The four receivers each feature a built-in D-type latch that is controlled from the buffered receiver latch enable ($\overline{RLE}$) input. When the $\overline{RLE}$ input is Low, the latch is open and the receiver outputs will follow the bus inputs (Bus data inverted and $\overline{OE}$ Low). When the $\overline{RLE}$ input is High, the latch will close and retain the present data regardless of the bus input. The four latches have three-state outputs and are controlled by a buffered common three-state control ($\overline{OE}$) input. When $\overline{OE}$ is High, the receiver outputs are in the high-impedance state.

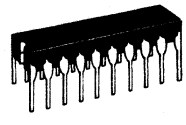
The MC2907 features a built-in four-bit odd parity checker/generator. The bus enable input ($\overline{BE}$) controls whether the parity output is in the generate or check mode. When the bus enable is Low (driver enabled), odd parity is generated based on the A field data input to the driver register. When $\overline{BE}$ is High, the parity output is determined by the four latch outputs of the receiver. Thus, if the driver is enabled, parity is generated; and, if the driver is in the high-impedance state, the Bus parity is checked.

- Quad High-speed LSI Bus Transceiver
- Open-collector Bus Driver
- D-type Register on Driver
- Bus Driver Output Can Sink 100 mA at 0.8 V Max
- Internal Odd 4-bit Parity Checker/Generator
- Receiver Has Output Latch for Pipeline Operation
- Three-state Receiver Outputs Sink 12 mA
- Advanced Low-power Schottky Processing
- 100% Reliability Assurance Testing in Compliance With MIL-STD-883

TTL QUAD BUS TRANSCEIVER WITH THREE-STATE RECEIVER AND PARITY

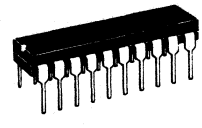


F SUFFIX
CERAMIC PACKAGE
CASE 737

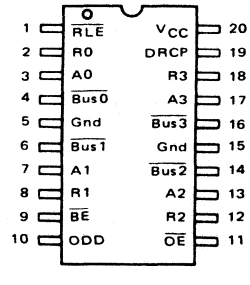


L SUFFIX
CERAMIC PACKAGE
CASE 732

P SUFFIX
PLASTIC PACKAGE
CASE 738



PIN ASSIGNMENT



ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Molded DIP	0°C to +70°C	MC2907PC
Hermetic DIP	0°C to +70°C	MC2907LC
Hermetic DIP	−55°C to +125°C	MC2907LM
Hermetic Flat Pack	−55°C to +125°C	MC2907FM

MAXIMUM RATINGS (above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential	−0.5 V to +7.0 V
DC Voltage Applied to Outputs for High Output State	−0.5 V to +V_{CC}max
DC Input Voltage	−0.5 V to +5.5 V
DC Output Current, into Outputs (Except Bus)	30 mA
DC Output Current, into Bus	200 mA
DC Input Current	−30 mA to +5.0 mA

ELECTRICAL CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise noted)

MC2907XC − T_A = 0°C to +70°C, V_{CC} = 5.0 V ±5% (Commercial), Min = 4.75 V, Max = 5.25 V

MC2907XM − T_A = −55°C to +125°C, V_{CC} = 5.0 V ±10% (Military), Min = 4.5 V, Max = 5.5 V

Parameter	Description	Test Conditions (Note 1)			Min	Typ (Note 2)	Max	Unit
BUS INPUT/OUTPUT CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE								
V_{OL}	Bus Output Low Voltage	V_{CC} = Min	I_{OL} = 40 mA			0.32	0.5	Volts
			I_{OL} = 70 mA			0.41	0.7	
			I_{OL} = 100 mA			0.55	0.8	
I_O	Bus Leakage Current	V_{CC} = Max	V_O = 0.4 V				−50	μA
			V_O = 4.5 V	Military			200	
				Commercial			100	
I_{off}	Bus Leakage Current (Power off)	V_O = 4.5 V					100	μA
V_{TH}	Receiver Input High Threshold	Bus Enable = 2.4 V		Military	2.4	2.0		Volts
				Commercial	2.3	2.0		
V_{TL}	Receiver Input Low Threshold	Bus Enable = 2.4 V		Military		2.0	1.5	Volts
				Commercial		2.0	1.6	
DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE								
V_{OH}	Receiver Output High Voltage	V_{CC} = Min, V_{in} = V_{IL} or V_{IH}	Military, I_{OH} = −1.0 mA		2.4	3.4		Volts
			Commercial, I_{OH} = −2.6 mA		2.4	3.4		
	Parity Output High Voltage	V_{CC} = Min, I_{OH} = −660 μA, V_{in} = V_{IH} or V_{IL}		Military	2.5	3.4		
				Commercial	2.7	3.4		
V_{OL}	Output Low Voltage (Except Bus)	V_{CC} = Min, V_{in} = V_{IL} or V_{IH}	I_{OL} = 4.0 mA			0.27	0.4	Volts
			I_{OL} = 8.0 mA			0.32	0.45	
			I_{OL} = 12 mA			0.37	0.5	
V_{IH}	Input High Level (Except Bus)	Guaranteed input logical High for all inputs			2.0			Volts
V_{IL}	Input Low Level (Except Bus)	Guaranteed input logical Low for all inputs		Military			0.7	Volts
				Commercial			0.8	
V_I	Input Clamp Voltage (Except Bus)	V_{CC} = Min, I_{in} = −18 mA					−1.2	Volts
I_{IL}	Input Low Current (Except Bus)	V_{CC} = Max, V_{in} = 0.4 V					−0.36	mA
I_{IH}	Input High Current (Except Bus)	V_{CC} = Max, V_{in} = 2.7 V					20	μA
I_{in}	Input High Current (Except Bus)	V_{CC} = Max, V_{in} = 5.5 V					100	μA
I_{SC}	Output Short Circuit Current (Except Bus)	V_{CC} = Max			−12		−65	mA
I_{CC}	Power Supply Current	V_{CC} = Max, All inputs = Gnd				75	110	mA
I_O	Off-State Output Current (Receiver Outputs)	V_{CC} = Max	V_O = 2.4 V				20	μA
			V_O = 0.4 V				−20	

NOTES: 1. For conditions shown as Min or Max, use the appropriate value specified under Electrical Characteristics for the applicable device type.

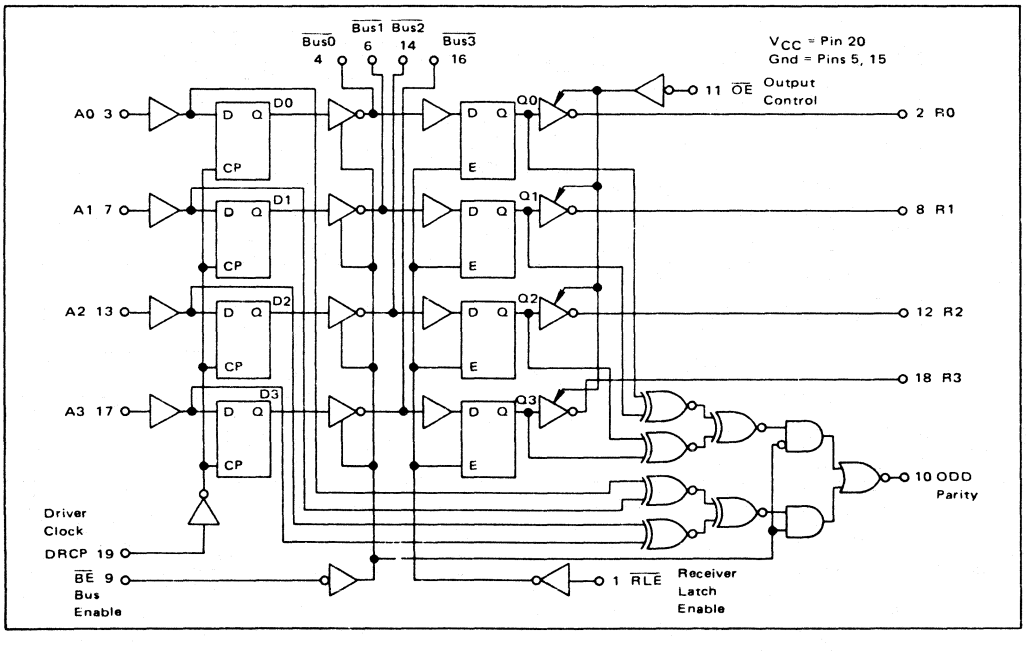
2. Typical limits are at V_{CC} = 5.0 V, 25°C ambient and maximum loading.

3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.

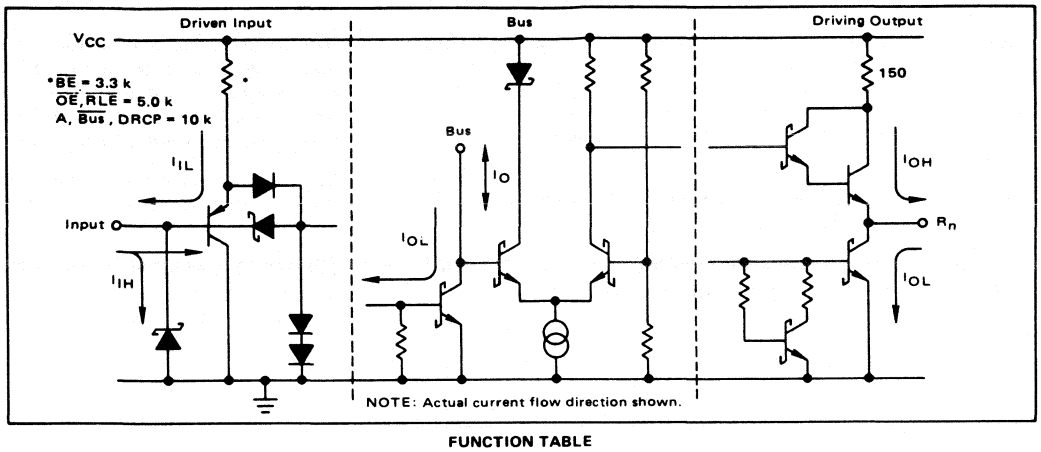
SWITCHING CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

Parameter	Description	Test Conditions	MC2907XM			MC2907XC			Unit
			Min	Typ (Note 2)	Max	Min	Typ (Note 2)	Max	
t_{PHL}	Driver Clock (DRCP) to Bus	C_L (Bus) = 50 pF	–	21	40	–	21	36	ns
t_{PLH}		R_L (Bus) = 50 Ω	–	21	40	–	21	36	
t_{PHL}	Bus Enable ($\overline{BE}$) to Bus		–	13	26	–	13	23	ns
t_{PLH}			–	13	26	–	13	23	
t_s	A Data Inputs	C_L = 15 pF	25	–	–	23	–	–	ns
t_h		R_L = 2.0 k	8.0	–	–	7.0	–	–	
t_{PW}	Clock Pulse Width (High)		28	–	–	25	–	–	ns
t_{PLH}	Bus to Receiver Output (Latch Enabled)		–	18	37	–	18	34	ns
t_{PHL}			–	18	37	–	18	34	
t_{PLH}	Latch Enable to Receiver Output		–	21	37	–	21	34	ns
t_{PHL}			–	21	37	–	21	34	
t_s	Bus to Latch Enable ($\overline{RLE}$)		21	–	–	18	–	–	ns
t_h			7.0	–	–	5.0	–	–	
t_{PLH}	A or B Data to Odd Parity (Driver Enabled)		–	21	40	–	21	36	ns
t_{PHL}			–	21	40	–	21	36	
t_{PLH}	Bus to Odd Parity Out (Driver Inhibited)		–	21	40	–	21	36	ns
t_{PHL}			–	21	40	–	21	36	
t_{PLH}	Latch Enable ($\overline{RLE}$) to Odd Parity Output		–	21	40	–	21	36	ns
t_{PHL}			–	21	40	–	21	36	
t_{ZH}	Output Control to Output		–	14	28	–	14	25	ns
t_{ZL}			–	14	28	–	14	25	
t_{HZ}	Output Control to Output	C_L = 5.0 pF	–	14	28	–	14	25	ns
t_{LZ}		R_L = 2.0 k	–	14	28	–	14	25	

LOGIC DIAGRAM



INPUT/OUTPUT CURRENT INTERFACE CONDITIONS

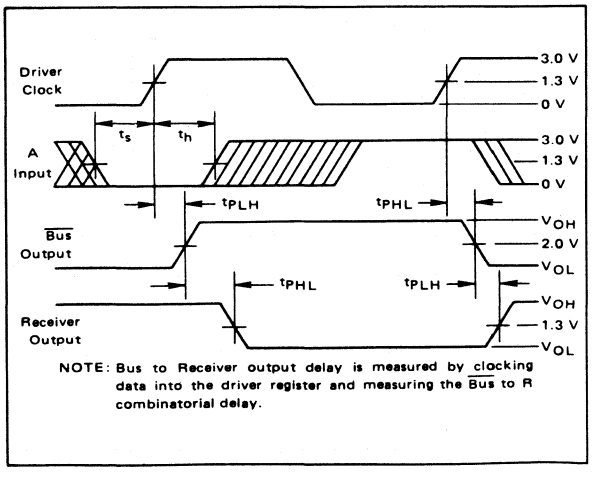


FUNCTION TABLE

		Inputs			Internal to Device		Bus	Output	
A_n	DRCP	$\overline{BE}$	$\overline{RLE}$	$\overline{OE}$	D_n	Q_n	B_n	R_n	Function
X	X	H	X	X	X	X	H	X	Driver output disable
X	X	X	X	H	X	X	X	Z	Receiver output disable
X	X	H	L	L	X	L	L	H	Driver output disable
X	X	H	L	L	X	H	H	L	and receive data via Bus input
X	X	X	H	X	X	NC	X	X	Latch received data
L	↑	X	X	X	L	X	X	X	Load driver register
H	↑	X	X	X	H	X	X	X	
X	L	X	X	X	NC	X	X	X	No driver clock restrictions
X	H	X	X	X	NC	X	X	X	
X	X	L	X	X	L	X	H	X	Drive Bus
X	X	L	X	X	H	X	L	X	

H = High Z = High impedance X = Don't care
L = Low NC = No change ↑ = Low-to-high transition

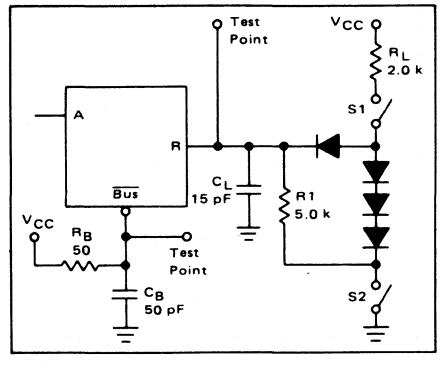
SWITCHING WAVEFORMS



PARITY OUTPUT FUNCTION TABLE

$\overline{BE}$	ODD Parity Output
L	ODD = A0 ⊕ A1 ⊕ A2 ⊕ A3
H	ODD = Q0 ⊕ Q1 ⊕ Q2 ⊕ Q3

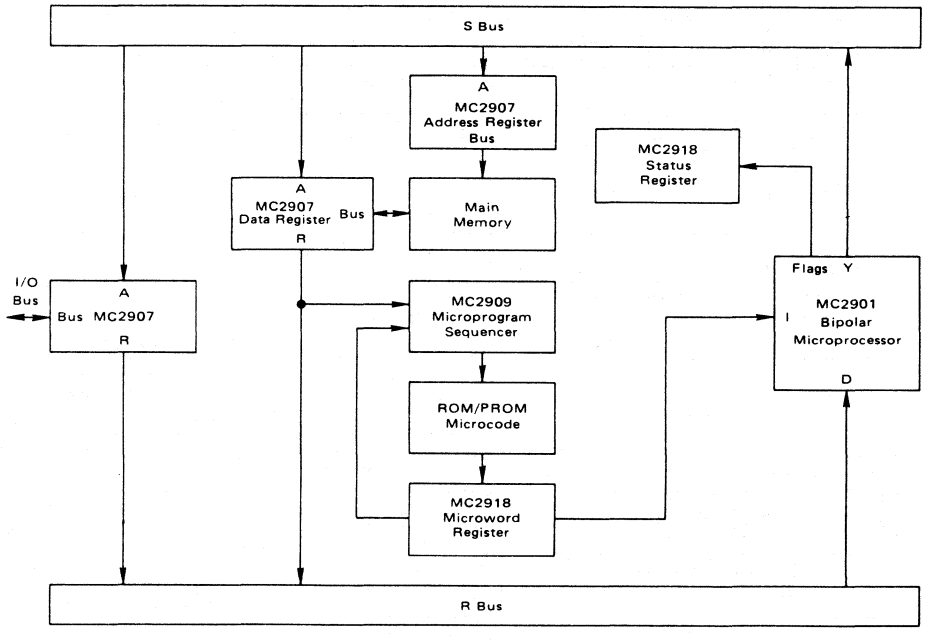
LOAD TEST CIRCUIT



DEFINITIONS OF FUNCTIONAL TERMS

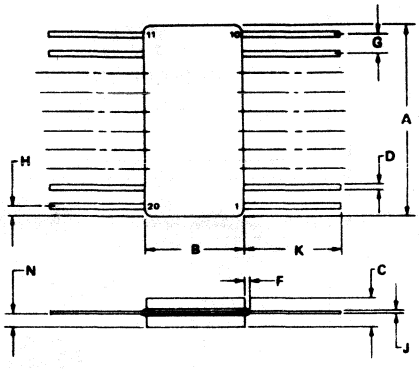
DRCP	Driver Clock Pulse. Clock pulse for the driver register.
$\overline{BE}$	$\overline{\text{Bus Enable}}$. When the Bus Enable is Low, the four drivers are in the high impedance state.
$\overline{Bus0}$, $\overline{Bus1}$, $\overline{Bus2}$, $\overline{Bus3}$	The four driver outputs and receiver inputs (data is inverted).
R0, R1, R2, R3	The four receiver outputs. Data from the bus is inverted while data from the A input is non-inverted.
$\overline{RLE}$	$\overline{\text{Receiver Latch Enable}}$. When $\overline{RLE}$ is Low, data on the Bus inputs is passed through the receiver latches. When $\overline{RLE}$ is High, the receiver latches are closed and will retain the data independent of all other inputs.
ODD	Odd parity output. Generates parity with the driver enabled, checks parity with the driver in the high impedance state.
$\overline{OE}$	$\overline{\text{Output Enable}}$. When the $\overline{OE}$ input is High, the four three-state receiver outputs are in the high impedance state.

APPLICATIONS



The MC2907 can be used as an I/O Bus Transceiver and Main Memory I/O Transceiver in high-speed microprocessor systems.

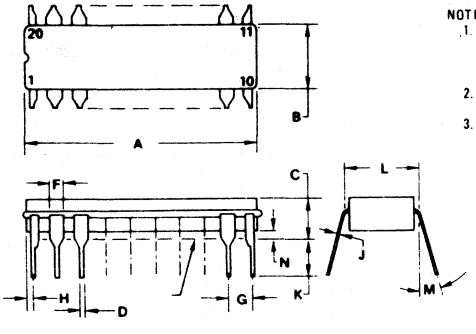
PACKAGE DIMENSIONS



NOTE:
1. LEADS WITHIN 0.25 mm (0.010) TOTAL OF TRUE POSITION AT MAXIMUM MATERIAL CONDITION.

CASE 737

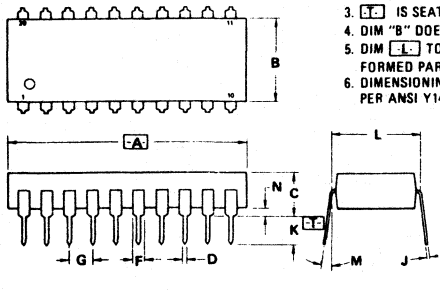
	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	–	13.08	–	0.515
B	5.84	6.60	0.230	0.260
C	1.52	2.16	0.060	0.085
D	0.41	0.46	0.016	0.018
F	–	0.25	–	0.010
G	1.27 BSC		0.050 BSC	
H	1.14	1.40	0.045	0.055
J	0.08	0.13	0.003	0.005
K	–	9.14	–	0.360
N	–	1.02	–	0.040



NOTES:
1. LEADS WITHIN 0.25 mm (0.010) DIA., TRUE POSITION AT SEATING PLANE, AT MAXIMUM MATERIAL CONDITION.
2. DIM L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIM A AND B INCLUDES MENISCUS.

CASE 732

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	24.38	25.15	0.960	0.990
B	6.86	7.49	0.270	0.295
C	4.32	5.08	0.170	0.200
D	0.38	0.56	0.015	0.022
F	1.40	1.65	0.055	0.065
G	2.54 BSC		0.100 BSC	
H	0.89	1.40	0.035	0.055
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	7.62 BSC		0.300 BSC	
M	5°	15°	5°	15°
N	0.51	0.76	0.020	0.030



NOTES:
1. DIM [A] IS DATUM.
2. POSITIONAL TOL FOR LEADS;
   ⌖ Ø 0.25 (0.010) Ⓜ T A Ⓜ
3. [T] IS SEATING PLANE.
4. DIM "B" DOES NOT INCLUDE MOLD FLASH.
5. DIM [L] TO CENTER OF LEADS WHEN FORMED PARALLEL.
6. DIMENSIONING AND TOLERANCING PER ANSI Y14.5, 1973.

CASE 738

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	25.65	27.18	1.010	1.070
B	6.10	6.60	0.240	0.260
C	3.94	4.19	0.155	0.165
D	0.38	0.56	0.015	0.022
F	1.27	1.78	0.050	0.070
G	2.54 BSC		0.100 BSC	
J	0.20	0.38	0.008	0.015
K	2.79	3.56	0.110	0.140
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	1.02	0.020	0.040

MOTOROLA

MC2909
MC2911

TTL
MICROPROGRAM SEQUENCER

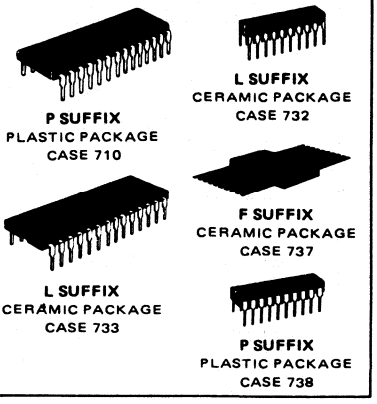
MICROPROGRAM SEQUENCER

The MC2909 is a four-bit-wide address controller intended for sequencing through a series of microinstructions contained in a ROM or PROM. Two MC2909s may be interconnected to generate a twelve-bit address (4K words).

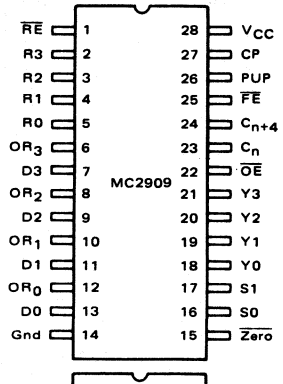
The MC2909 can select an address from any of four sources. They are: 1) a set of external direct inputs (D); 2) external data from the R inputs, stored in an internal register; 3) a four-word-deep push/pop stack; or 4) a program counter register (which usually contains the last address plus one). The push/pop stack includes certain control lines so that it can efficiently execute nested subroutine linkages. Each of the four outputs can be OR'ed with an external input for conditional skip or branch instructions, and a separate line forces the outputs to all zeroes. The outputs are three-state.

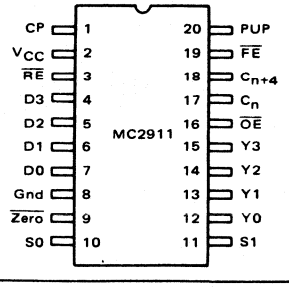
The MC2911 is an identical circuit to the MC2909, except the four OR inputs are removed and the D and R inputs are tied together. The MC2911 is in a 20-pin, 0.3" centers package.

- 4-Bit Slice Cascadable to Any Number of Microwords
- Internal Address Register
- Branch Input for N-Way Branches
- Cascadable 4-Bit Microprogram Counter
- 4 X 4 File with Stack Pointer and Push/Pop Control for Nesting Microsubroutines
- Zero Input for Returning to the Zero Microcode Word
- Individual OR Input for Each Bit for Branching to Higher Microinstructions
- Three-State Outputs
- All Internal Registers Change State on the Low-to-High Transition of the Clock



PIN ASSIGNMENTS





MAXIMUM RATINGS (above which the useful life may be impaired)	
Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential	−0.5 V to +V_{CC} max
DC Input Voltage	−0.5 V to +7.0 V
DC Output Current, Into Outputs	30 mA
DC Input Current	−30 mA to +5.0 mA

ORDERING INFORMATION

Package Type	Temperature Range	MC2909 Order Number	MC2911 Order Number
Molded DIP	0°C to +70°C	MC2909PC	MC2911PC
Hermetic DIP	0°C to +70°C	MC2909LC	MC2911LC
Hermetic DIP	−55°C to +125°C	MC2909LM	MC2911LM
Hermetic Flat Pak	−55°C to +125°C	–	MC2911FM

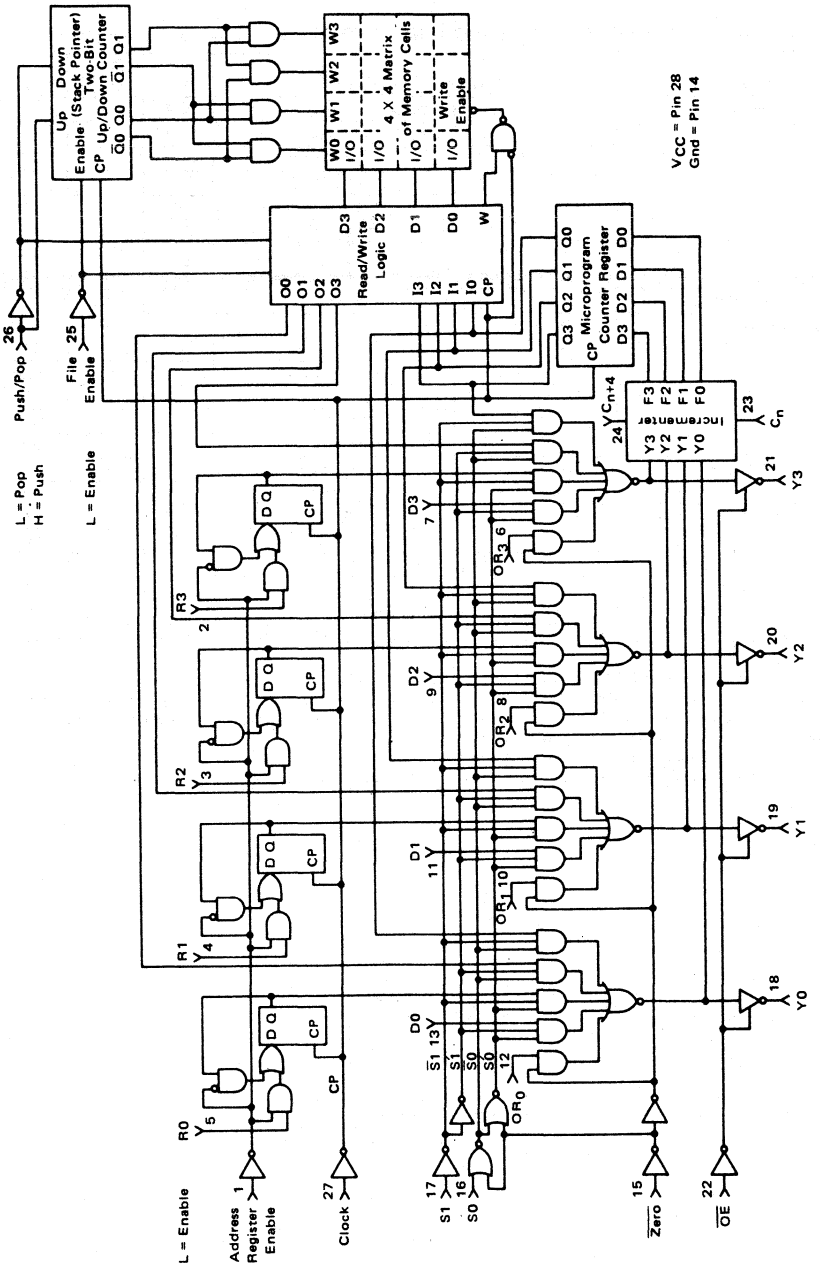


FIGURE 1 – MC2909 MICROPROGRAM SEQUENCER BLOCK DIAGRAM

The MC2911 is an IDENTICAL circuit to the MC2909, except the four OR inputs are removed and the D and R inputs are tied together. The MC2911 is in a 20-pin Dual-in-Line package. See Figure 11.

ELECTRICAL CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise noted)

Parameters	Description	Test Conditions (Note 1)			Min	Typ (Note 2)	Max	Units
V_{OH}	Output High Voltage	V_{CC} = Min, V_{in} = V_{IH} or V_{IL}	Military	I_{OH} = −1.0 mA	2.4	–	–	Volts
			Commercial	I_{OH} = −2.6 mA	2.4	–	–	
V_{OL}	Output Low Voltage	V_{CC} = Min, V_{in} = V_{IH} or V_{IL}	I_{OL} = 4.0 mA		–	–	0.4	Volts
			I_{OL} = 8.0 mA		–	–	0.45	
			I_{OL} = 12 mA (Note 5)		–	–	0.5	
V_{IH}	Input High Level	Guaranteed input logical High voltage for all inputs			2.0	–	–	Volts
V_{IL}	Input Low Level	Guaranteed input logical Low voltage for all inputs		Military	–	–	0.7	Volts
				Commercial	–	–	0.8	
V_I	Input Clamp Voltage	V_{CC} = Min, I_{in} = −18 mA			–	–	−1.5	Volts
I_{IL}	Input Low Current	V_{CC} = Max, V_{in} = 0.4 V	C_n		–	–	−1.08	mA
			Push/Pop, OE		–	–	−0.72	
			Others (Note 6)		–	–	−0.36	
I_{IH}	Input High Current	V_{CC} = Max V_{in} = 2.7 V	C_n		–	–	40	μA
			Push/Pop		–	–	40	
			Others (Note 6)		–	–	20	
I_I	Input High Current	V_{CC} = Max V_{in} = 7.0 V	C_n, Push/Pop		–	–	0.2	mA
			Others (Note 6)		–	–	0.1	
I_{OS}	Output Short Circuit Current (Note 3)	V_{CC} = Max	Y0 – Y3		−30	–	−100	mA
			C_{n+4}		−30		−85	
I_{CC}	Power Supply Current	V_{CC} = Max (Note 4)			–	80	130	mA
I_{OZL} I_{OZH}	Output Off Current	V_{CC} = Max, OE = 2.7 V	V_{Out} = 0.4 V		–	–	−20	μA
			V_{Out} = 2.7 V		–	–	20	

NOTES:
1. For conditions shown as Min or Max, use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0 V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
4. Apply Gnd to C_n, R0, R1, R2, R3, OR_0, OR_1, OR_2, OR_3, D0, D1, D2, and D3. Other inputs open. All outputs open. Measured after a Low-to-High clock transition.
5. The 12 mA output applies only to Y0, Y1, Y2, and Y3.
6. For the MC2911, D_i and R_i are interanlly connected. Loading is doubled (to same values as Push/Pop).

Operating Range	Part Numbers	Power Supply	Temperature Range
Commercial	MC2909PC, LC MC2911PC, LC	5.0 V ± 5%	T_A = 0°C to +70°C
Military	MC2909LM MC2911LM MC2911FM	5.0 V ± 10%	T_C = −55°C to +125°C

SWITCHING CHARACTERISTICS OVER OPERATING RANGE

Tables 1, 2, and 3 define the timing characteristics of the MC2909 and MC2911 over the operating voltage and temperature range. The tables are divided into three types of parameters: clock characteristics, combinational delays from inputs to outputs, and setup and hold time requirements. The latter table defines the time prior to the end of the cycle (i.e., clock Low-to-High transition) that each input must be stable to guarantee that the correct data is written into one of the internal registers.

Measurements are made at 1.5 V with V_{IL} = 0 V and V_{IH} = 3.0 V. For three-state disable tests, C_L = 5.0 pF and measurement is to 0.5 V change on output voltage level.

TABLE 1 – CYCLE TIME AND CLOCK CHARACTERISTICS

Time	Commercial	Military
Minimum Clock Low Time	30	35
Minimum Clock High Time	30	35

TABLE 2 – MAXIMUM COMBINATIONAL PROPAGATION DELAYS

All in ns, C_L = 50 pF (except output disable tests)

From Input	Commercial		Military	
	Y	C_{n+4}	Y	C_{n+4}
D_i	17	30	20	32
S0, S1	30	48	40	50
OR_i	17	30	20	32
C_n	–	14	–	16
$\overline{Zero}$	30	48	40	50
$\overline{OE}$ Low (Enable)	25	–	25	–
$\overline{OE}$ High (Disable)	25	–	25	–
Clock ↑ S1,S0 = LH	43	55	50	62
Clock ↑ S1,S0 = LL	43	55	50	62
Clock ↑ S1,S0 = HL	80	95	90	102

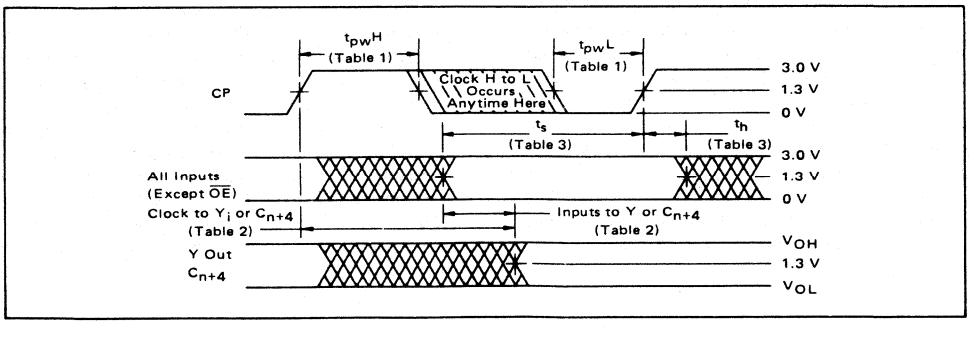
TABLE 3 – GUARANTEED SETUP AND HOLD TIMES

All in ns (Note 1)

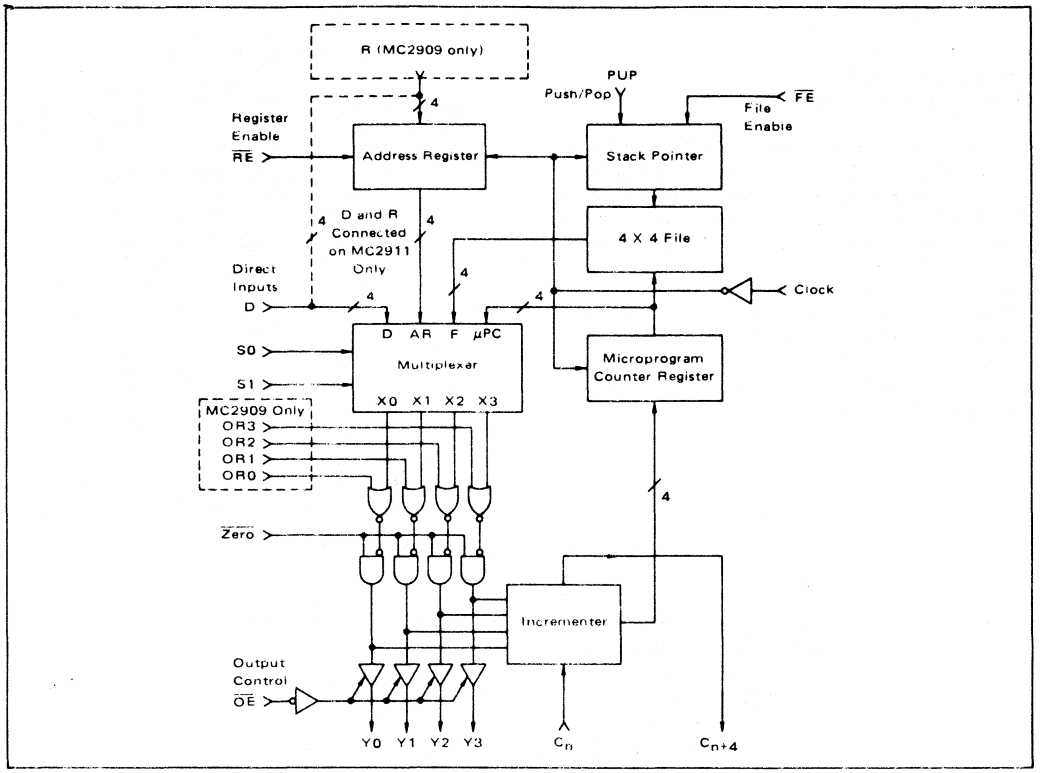
From Input	Notes	Commercial		Military	
		Setup Time	Hold Time	Setup Time	Hold Time
$\overline{RE}$		22	5	22	5
R_i	2	10	5	12	5
Push/Pop		26	6	30	7
$\overline{FE}$		26	5	30	5
C_n		28	5	30	5
D_i	2	30	0	35	3
OR_i		30	0	35	3
S0, S1		45	0	50	0
$\overline{Zero}$		45	0	50	0

NOTES: 1. All times relative to clock Low-to-High transition.
2. On MC2911, R_i and D_i are internally connected together and labeled D_i. Use R_i setup and hold times when D inputs are used to load register.

FIGURE 2 – SWITCHING WAVEFORMS
(See Tables for Specific Values)



MICROPROGRAM SEQUENCER



ARCHITECTURE OF THE MC2909/MC2911

The MC2909/MC2911 are bipolar microprogram sequencers intended for use in high-speed microprocessor applications. The device is a cascadable 4-bit slice such that two devices allow addressing of up to 256-words of microprogram and three devices allow addressing of up to 4K words of microprogram. A detailed logic diagram is shown in Figure 1.

The device contains a four-input multiplexer that is used to select either the address register, direct inputs, microprogram counter, or file as the source of the next microinstruction address. This multiplexer is controlled by the S0 and S1 inputs.

The address register consists of four D-type, edge-triggered flip-flops with a common clock enable. When the address register enable is Low, new data is entered into the register on the clock Low-to-High transition. The address register is available at the multiplexer as a source for the next microinstruction address. The direct input is a 4-bit field of inputs to the multiplexer and can be selected as the next microinstruction address. On the MC2911, the direct inputs are also used as inputs to the register. This allows an N-way branch where N is any word in the microcode.

The MC2902/MC2911 contains a microprogram counter (μPC) that is composed of a 4-bit incrementer followed by a 4-bit register. The incrementer has carry-in (C_n) and carry-out (C_{n+4}) such that cascading to larger word lengths is straightforward. The μPC can be used in either of two ways. When the least significant carry-in to the incrementer is High, the microprogram register is loaded on the next clock cycle with the current Y output word plus one ($Y + 1 \rightarrow \mu PC$). Thus sequential microinstructions can be executed. If this least significant C_n is Low, the incrementer passes the Y output word unmodified and the microprogram reigster is loaded with the same Y word on the next clock cycle ($Y \rightarrow \mu PC$). Thus, the same microinstruction can be executed any number of times by using the least significant C_n as the control.

The last source available at the multiplexer input is the 4 X 4 file (stack). The file is used to provide return address linkage when executing microsubroutines. The file contains a built-in stack pointer (SP) which always points to the last file word written. This allows stack reference operations (looping) to be performed without a push or pop.

The stack pointer operates as an up/down counter with separate push/pop and file enable inputs. When the file enable input is Low and the push/pop input is High, the Push operation is enabled. This causes the stack pointer to increment and the file to be written with the required return linkage—the next microinstruction address following the subroutine jump which initiated the Push.

If the file enable input is Low and the push/pop control is Low, a Pop operation occurs. This implies the usage of the return linkage during this cycle and thus a return from subroutine. The next Low-to-High clock transition causes the stack pointer to decrement. If the file enable is High, no action is taken by the stack pointer regardless of any other input.

The stack pointer linkage is such that any combination of pushes, pops, or stack references can be achieved. One microinstruction subroutine can be performed. Since the stack is four words deep, up to four microsubroutines can be nested.

The $\overline{\text{Zero}}$ input is used to force the four outputs to the binary zero state. When the $\overline{\text{Zero}}$ input is Low, all Y outputs are Low regardless of any other inputs (except $\overline{\text{OE}}$). Each Y output bit also has a separate OR input such that a conditional logic one can be forced at each Y output. This allows jumping to different microinstructions on programmed conditions.

The MC2909/MC2911 feature three-state Y outputs. These can be particularly useful in military designs requiring external Ground Support Equipment (GSE) to provide automatic checkout of the microprocessor. The internal control can be placed in the high-impedance state, and preprogrammed sequences of microinstructions can be executed via external access to the control ROM/PROM.

OPERATION OF THE MC2902/MC2911

Table 4 lists the select codes for the multiplexer. The two bits applied from the microword register (and additional combinational logic for branching) determine which data source contains the address for the next microinstruction. The contents of the selected source will appear on the Y outputs. Table 4 also shows the truth table for the output control and for the control of the push/pop stack. Table 5 shows in detail the effect of S0, S1, $\overline{\text{FE}}$ and PUP on the MC2909. These four signals define what address appears on the Y outputs and what

TABLE 4

ADDRESS SELECTION

Octal	S1	S0	Source for Y Outputs	Symbol
0	L	L	Microprogram Counter	μPC
1	L	H	Address Register	AR
2	H	L	Push/Pop Stack	STK0
3	H	H	Direct Inputs	D_i

OUTPUT CONTROL

OR_i	$\overline{\text{Zero}}$	$\overline{\text{OE}}$	Y_i
X	X	H	Z
X	L	L	L
H	H	L	H
L	H	L	Source selected by S0,S1

Z = High Impedance

SYNCHRONOUS STACK CONTROL

$\overline{\text{FE}}$	PUP	Push/Pop Stack Change
H	X	No Change
L	H	Incrementer Stack Pointer, then Push Current PC onto STK0
L	L	Pop Stack (Decrement Stack Pointer)

the state of all the internal registers will be following the clock Low-to-High edge. In this illustration, the microprogram counter is assumed to contain initially some word J, the address register some word K, and the four words in the push/pop stack contain R_a through R_d.

Table 6 illustrates the execution of a subroutine using the MC2909. The configuration of Figure 3 is assumed. The instruction being executed at any given time is the one contained in the microword register (μWR). The contents of the μWR also controls (indirectly, perhaps) the four signals S0, S1, $\overline{\text{FE}}$, and PUP. The starting address of the subroutine is applied to the D inputs of the MC2909 at the appropriate time.

In the columns on the left is the sequence of microinstructions to be executed. At address J+2, the sequence control portion of the microinstruction contains the command "Jump to the subroutine at A". At the time T2, this instruction is in the μWR and the MC2909 inputs are setup to execute the jump and save the return address. The subroutine address A is applied to the D inputs from the μWR and appears on the Y outputs. The first instruction of the subroutine, I(A), is accessed and is at the inputs of the μWR. On the next clock transistion, I(A) is loaded into the μWR for execution, and the return address J+3 is pushed onto the stack. The return instruction is executed at T5. Table 7 is a similar timing chart showing one subroutine linking to a second, the latter consisting of only one microinstruction.

TABLE 5 – OUTPUT AND INTERNAL NEXT-CYCLE REGISTER STATES FOR MC2909/MC2911

Cycle	S1, S0, $\overline{FE}$, PUP	μPC	AR	STK0	STK1	STK2	STK3	Y_{Out}	Comment	Principal Use
N	0 0 0 0	J	K	Ra	Rb	Rc	Rd	J	Pop Stack	End Loop
N+1	–	J+1	K	Rb	Rc	Rd	Ra	–		
N	0 0 0 1	J	K	Ra	Rb	Rc	Rd	J	Push μPC	Setup Loop
N+1	–	J+1	K	J	Ra	Rb	Rc	–		
N	0 0 1 X	J	K	Ra	Rb	Rc	Rd	J	Continue	Continue
N+1	–	J+1	K	Ra	Rb	Rc	Rd	–		
N	0 1 0 0	J	K	Ra	Rb	Rc	Rd	K	Pop Stack; Use AR for Address	End Loop
N+1	–	K+1	K	Rb	Rc	Rd	Ra	–		
N	0 1 0 1	J	K	Ra	Rb	Rc	Rd	K	Push μPC; Jump to Address in AR	JSR AR
N+1	–	K+1	K	J	Ra	Rb	Rc	–		
N	0 1 1 X	J	K	Ra	Rb	Rc	Rd	K	Jump to Address in AR	JMP AR
N+1	–	K+1	K	Ra	Rb	Rc	Rd	–		
N	1 0 0 0	J	K	Ra	Rb	Rc	Rd	Ra	Jump to Address in STK0; Pop Stack	RTS
N+1	–	Ra+1	K	Rb	Rc	Rd	Ra	–		
N	1 0 0 1	J	K	Ra	Rb	Rc	Rd	Ra	Jump to Address in STK0; Push μPC	
N+1	–	Ra+1	K	J	Ra	Rb	Rc	–		
N	1 0 1 X	J	K	Ra	Rb	Rc	Rd	Ra	Jump to Address in STK0	Stack Ref (Loop)
N+1	–	Ra+1	K	Ra	Rb	Rc	Rd	–		
N	1 1 0 0	J	K	Ra	Rb	Rc	Rd	D	Pop Stack; Jump to Address on D	End Loop
N+1	–	D+1	K	Rb	Rc	Rd	Ra	–		
N	1 1 0 1	J	K	Ra	Rb	Rc	Rd	D	Jump to Address on D; Push μPC	JSR D
N+1	–	D+1	K	J	Ra	Rb	Rc	–		
N	1 1 1 X	J	K	Ra	Rb	Rc	Rd	D	Jump to Address on D	JMP D
N+1	–	D+1	K	Ra	Rb	Rc	Rd	–		

X = Don't Care, 0 = Low, 1 = High. Assume C_n = High.

TABLE 6 – SUBROUTINE EXECUTION

Control Memory		
Execute Cycle	Microprogram Address	Microprogram Instruction
	J–1	–
T0	J	–
T1	J+1	–
T2	J+2	JSR A
T6	J+3	–
T7	J+4	–
	–	–
	–	–
	–	–
	–	–
	–	–
T3	A	I(A)
T4	A+1	–
T5	A32	RTS
	–	–
	–	–
	–	–
	–	–
	–	–
	–	–

Execute Cycle		T0	T1	T2	T3	T4	T5	T6	T7	T8	T9
Clock											
Signals											
MC2909 Inputs (from μWR)	S1, S0	0	0	3	0	0	2	0	0		
	$\overline{FE}$	H	H	L	H	H	L	H	H		
	PUP	X	X	H	X	X	L	X	X		
	D	X	X	A	X	X	X	X	X		
Internal Registers	μPC	J+1	J+2	J+3	A+1	A+2	A+3	J+4	J+5		
	STK0	–	–	–	J+3	J+3	J+3	–	–		
	STK1	–	–	–	–	–	–	–	–		
	STK2	–	–	–	–	–	–	–	–		
	STK3	–	–	–	–	–	–	–	–		
MC2909 Output	Y	J+1	J+2	A	A+1	A+2	J+3	J+4	J+5		
ROM Output	(Y)	I(J+1)	**JSR A**	I(A)	I(A+1)	**RTS**	I(J+3)	I(J+4)	I(J+5)		
Contents of μWR (Instruction Being Executed)	μWR	I(J)	I(J+1)	**JSR A**	I(A)	I(A+1)	**RTS**	I(J+3)	I(J+4)		

C_n = High

TABLE 7 – TWO NESTED SUBROUTINES
Routine B is Only One Instruction.

Control Memory			Execute Cycle		T0	T1	T2	T3	T4	T5	T6	T7	T8	T9
Execute Cycle	Microprogram Address	Microprogram Instruction	Clock											
	J–1	—	Signals											
T0	J	—	MC2909 Inputs (from μWR)	S1, S0	0	0	3	0	0	3	2	0	2	0
T1	J+1	—		$\overline{FE}$	H	H	L	H	H	L	L	H	L	H
T2	J+2	JSR A		PUP	X	X	H	X	X	H	L	X	L	X
T9	J+3	—		D	X	X	Z	X	X	B	X	X	X	X
	—	—	Internal Registers	μPC	J+1	J+2	J+3	A+1	A+2	A+3	B+1	A+4	A+5	J+4
	—	—		STK0	—	—	—	J+3	J+3	J+3	A+3	J+3	J+3	—
	—	—		STK1	—	—	—	—	—	—	J+3	—	—	—
	—	—		STK2	—	—	—	—	—	—	—	—	—	—
T3	A	—		STK3	—	—	—	—	—	—	—	—	—	—
T4	A+1	—	MC2909 Output	Y	J+1	J+2	A	A+1	A+2	B	A+3	A+4	J+3	J+4
T5	A+2	JSR B	ROM Output	(Y)	I(J+1)	**JSR A**	I(A)	I(A+1)	**JSR B**	**RTS**	I(A+3)	**RTS**	I(J+3)	I(J+4)
T7	A+3	—	Contents of μWR (Instruction Being Executed)	μWR	I(J)	I(J+1)	**JSR A**	I(A)	I(A+1)	**JSR B**	**RTS**	I(A+3)	**RTS**	I(J+3)
T8	A+4	RTS												
	—	—												
	—	—												
	—	—												
T6	B	RTS												
	—	—												

C_n = High

DEFINITION OF TERMS

A set of symbols is used in this data sheet to represent various internal and external registers and signals used with the MC2909/MC2911. Since its principle application is as a controller for a microprogram store, it is necessary to define some signals associated with the microcode itself. Figure 3 illustrates the basic interconnection of MC2909, memory, and microinstruction register. The definitions here apply to this architecture.

Inputs to MC2909/MC2911

S1, S0 Control lines for address source selection

$\overline{FE}$, PUP Control lines for push/pop stack

$\overline{RE}$ Enable line for internal address register

OR_i Logic OR inputs on each address output line

$\overline{Zero}$ Logic AND input on the output lines

$\overline{OE}$ Output Enable. When $\overline{OE}$ is High, the Y outputs are Off (high impedance)

C_n Carry-in to the incrementer

R_i Inputs to the internal address register

D_i Direct inputs to the multiplexer

CP Clock input to the AR and μPC register and Push/Pop stack

Outputs from the MC2909/MC2911

Y_i Address outputs from MC2909. (Address inputs to control memory.)

C_{n+4} Carry out from the incrementer

Internal Signals

μPC Contents of the microprogram counter

AR Contents of the address register

STK0–STK3 Contents of the push/pop stack. By definition, the word in the four-by-four file, addressed by the stack pointer is STK0. Conceptually, data is pushed into the stack at STK0; a subsequent push moves STK0 to STK1; a pop implies STK3 → STK2 → STK1 → STK0. Physically, only the stack pointer changes when a push or pop is performed. The data does not move. I/O occurs at STK0.

SP Contents of the stack pointer

External to the MC2909/MC2911

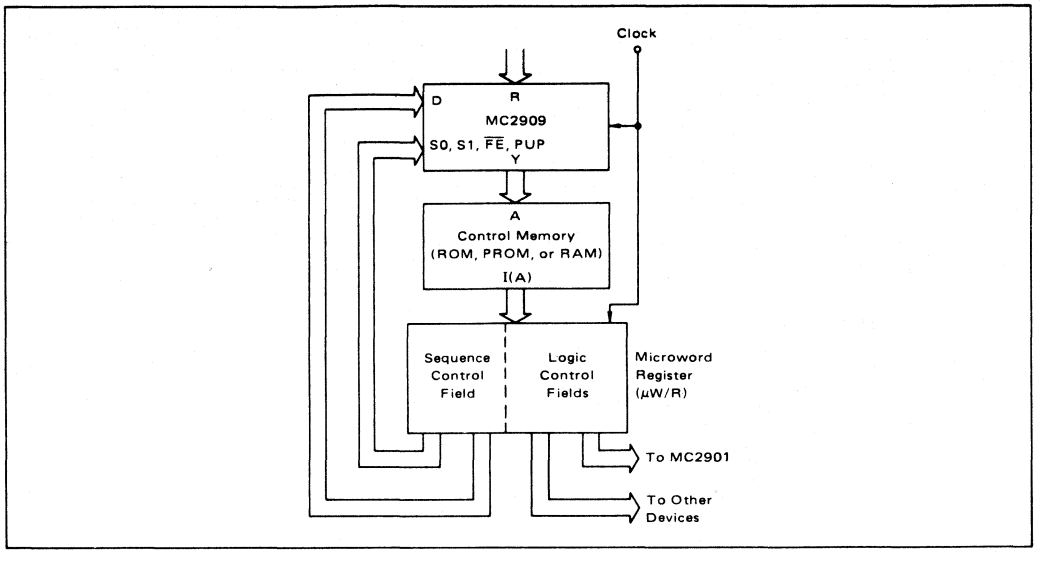
A Address to the control memory

I(A) Instruction in control memory at address A

μWR Contents of the microword register (at output of control memory). The microword register contains the instruction currently being executed.

T_n Time period (cycle) n

FIGURE 3 – MICROPROGRAM SEQUENCER CONTROL



USING THE MC2909/11

The computer control unit (CCU) is generally the single most complicated subsystem in today's digital computer. A CCU is complicated from the conceptualization, design, and implementation viewpoints, because it is the subsystem that controls the internal buses and subsystems of the processor, synchronizes internal and external events, and grants or denies permission to external systems. The MC2909 Microprogram Sequencer is an excellent mechanism for simplifying the CCU design task.

Computer Architecture

A classical computer architecture is shown in Figure 4. The data bus is commonly used by all of the subsystems in the computer. Information, instructions, address operands, data and sometimes control signals are transmitted down the data bus under control of a microprogram. The microprogram selects the source of the data as well as the destination(s) of the data. In a more complicated system there may be a number of data buses.

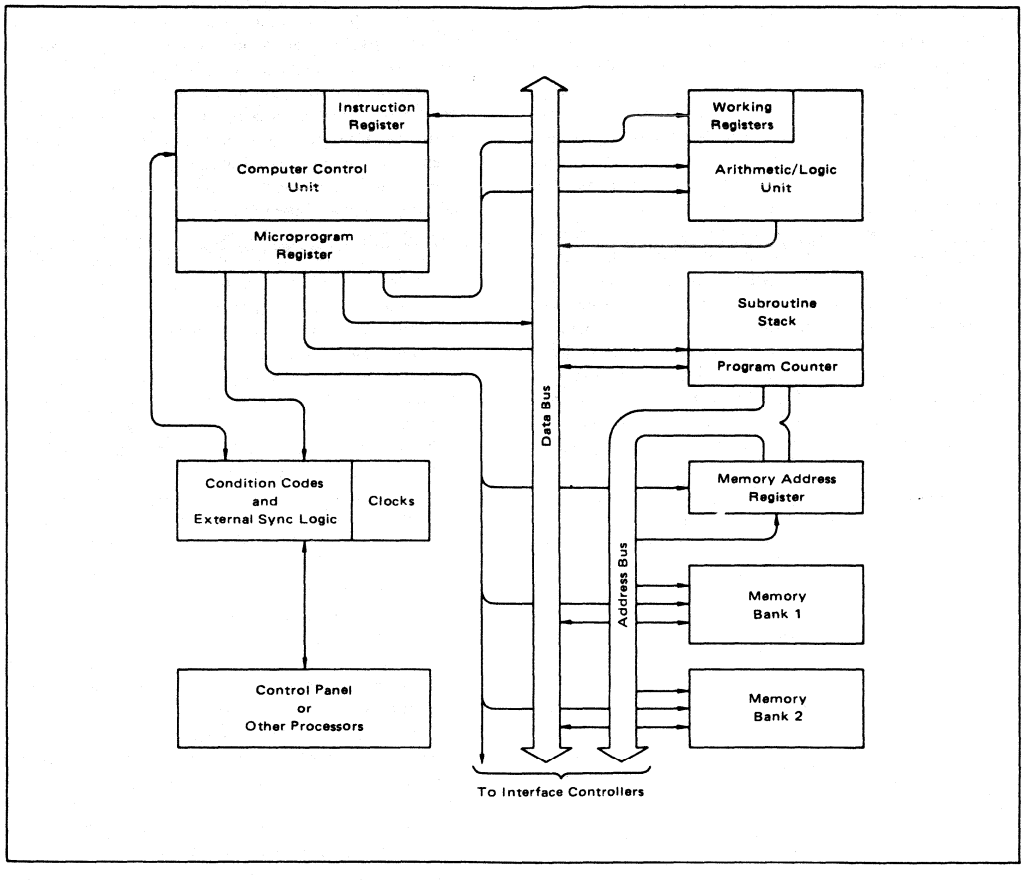
The address bus is typically used to select a word in memory for an internal computer function, or to select an input/output port for an external subsystem or peripheral function. Also selected by microprogram command, the source of the data for the address bus may be the program counter, the memory address reigster, a direct memory address controller, an interface controller, etc.

The arithmetic/logic unit (ALU) is actually that portion of the processor that computes. Depending upon the complexity of the ALU, a large number of different arithmetic functions can be accomplished in various number systems using different representations of those data. The most common minimum sets, however, are the functions (A plus B), (A minus B), and (B minus A) performed in fixed point, two's complement binary form where A and B are the ALU inputs. The logical functions are obtained from the same combinatorial logic array that is used for the arithmetic functions, but it is gated in a different manner. The minimum logical function capability will be (A OR B), (A AND B), and (A EXCLUSIVE-OR B). In addition to these combinatorial logic functions, there are sets of shift and rotate instructions that complete the basic instruction set.

The ALU provides a set of condition codes as a result of the current arithmetic or logical function. These condition codes include such variables as carry-out, A = B, the sign bit, result equals zero, etc. The condition codes, along with other computer status information, are stored in a register for later use by the programmer or computer control unit.

Third generation processors also provide for a general-purpose register set that is available to the programmer to be used to hold variables that are used often—passing arguments to subroutines, referencing memory indirectly,

FIGURE 4 – GENERALIZED COMPUTER ARCHITECTURE



etc. Depending on the architecture of the machine, the general-purpose registers may be selected directly from the operands in the instruction register from an address in the microprogram store, or one of the two sources as determined by a bit in the microprogram store.

The program counter and the memory address register are the two main sources of memory word and I/O address select data on the address bus. The program counter contains the address of the next instruction or instruction operand that is to be fetched from main memory, and the memory address register contains instruction address operands that are necessary to fetch the data required for the execution of the current instruction.

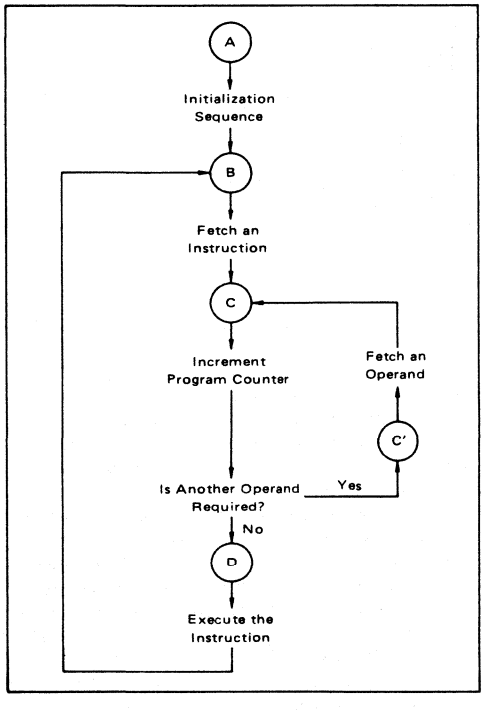
A subroutine address stack is provided to allow the return address linkage to the handled easily when exiting a subroutine. The address stack is a last-in, first-out stack that is controlled by a jump-to-subroutine, Push, or a return-from-subroutine, Pop, instruction from the CCU microprogram word.

Main memory poses an interesting set of problems to computer designers. Random access memory is typically slower than the computer control unit or arithmetic logic unit speed. And, on the other hand, read only memory may be much faster than the control process. The same set of problems is presented to the system by peripheral devices and processes. The computer control unit contends with these problems as well as the problems of synchronizing asynchronous events.

The Control Sequence

The computer control unit contains an instruction register, microprogram storage, and usually a microprogram register. Figure 5 presents a state diagram for a typical computer control unit. The first state of any processor must be an initialization sequence, regardless of its level of complexity or sophistication. The purpose of the initialization sequence is to place all of the system control storage elements in a known state such that control of the process can be started in an orderly manner. For example, registers, condition code, flag, and carry/link flip-flops are either preset to logic "1" or cleared to logic "0". Sometimes a sequence of events takes place such as the initialization of sets of register stacks or main memory. Also, because some peripheral equipment may be involved that may be damaged by randomly changing states at its interface, very close attention must be given to the initialization process within the CCU state machine. A further requirement of this initialization pocess is that clock pulses must be withheld from the initialized hardware in some manner until the initialization procedure is completed.

FIGURE 5 – SIMPLE COMPUTER CONTROL UNIT STATE TRANSITION DIAGRAM



The initialization sequence is usually started by one of three events: application of primary power to the system; either a programmed or operator generated "Master Reset" command; or, an error that the state machine cannot recover from, but can detect. In a power-up generated initialization sequence, care must be given to the circuit that detects the event and generates the timed reset signal. The various power supply filters and loads must be considered as the state machine sequence should not be allowed to start until the entire power system is stable. Furthermore, since some equipment and components may be damaged if they required multiple voltages that are not applied in the proper order, the computer control unit quite often is used to sequence the enabling of power supplies.

State "B" is the first computer minor cycle period. (A minor cycle is one primary clock period in length; characteristically, one microinsturction is executed. A major cycle is composed of one or more minor cycles and describes the completion of a macroinstruction or macroprocessors, i.e., "Add" or "Interrupt".) During this state, the processor may be interrupted, halted, paused, or—in the absence of any of these requests—the computer control unit will fetch a macroinstruction from main memory and load it into the instruction register.

Subsequently, during state "C", the Program Counter will be incremented and the instruction previously fetched will be decoded. If another operand is required for the current instruction, state "C" will be executed the necessary number of times, and the operands will be loaded into the appropriate registers until the requirements of the instruction have been satisfied.

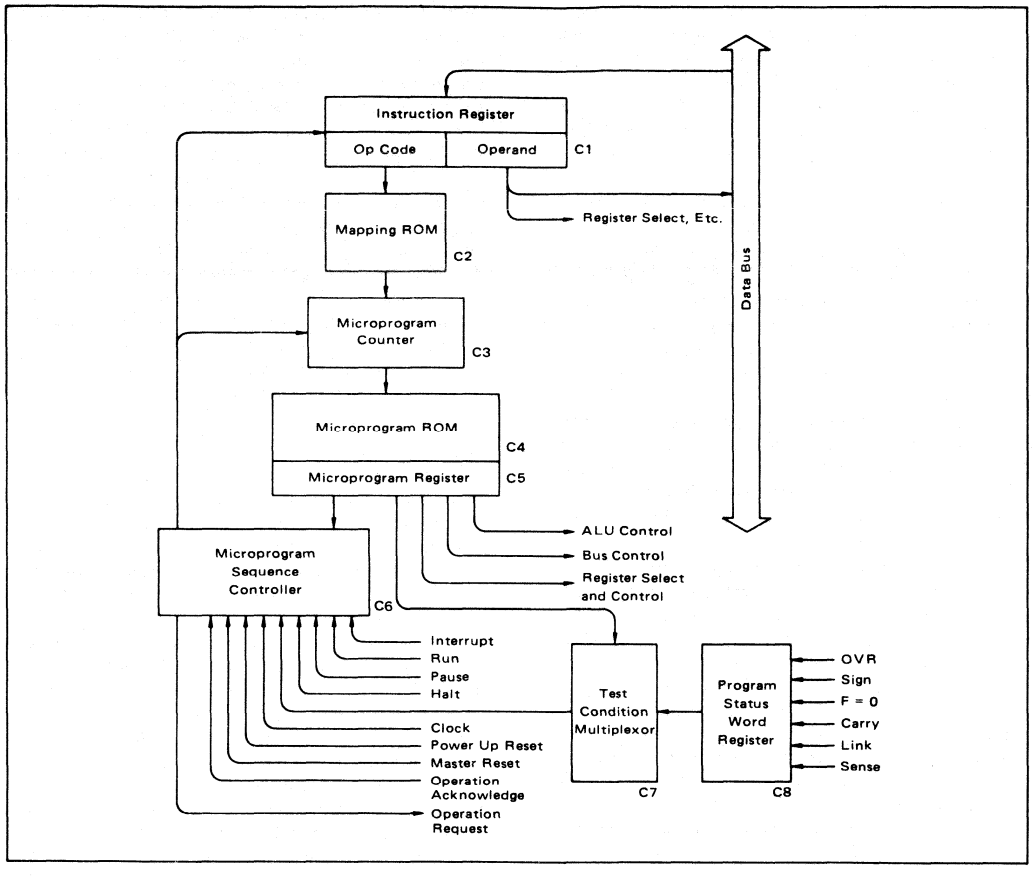
The last state, "D", is where the macroinstruction is executed. As in all of the other states in the process, the instruction execution state may require one or more microinstruction cycles. Having completed this state, control of the CCU will revert to state "B" microcode after a microinstruction branch to the beginning of that sequence has been effected.

CCU Architecture

A functional representation of a computer control unit is presented in Figure 6. To aid in the diagram reference process, the major subsystem components are labeled with the designations C1, to C8.

The instruction register, C1, receives the instruction from main memory via the data bus. The width of the register is generally the same as the memory word and data bus width to conserve processor overhead time. That is, if one clock period is necessary to fetch an instruction and one clock period is used to execute the instruction, that is a much more efficient use of computer time than requiring two or more clock periods to fetch the instruction and only one clock period to execute it. Any time

FIGURE 6 – COMPUTER CONTROL UNIT ARCHITECTURAL SCHEMATIC



required by the processor over and above the instruction execution time is considered overhead.

An instruction is broken down into two or more fields: the opcode, and one or more operands. An opcode (operation code) is the instruction itself. The operands are data used by the computer control unit the execution of the instruction. For example, an operand might be the number of a selected register, a variable to be compared to the accumulator, the address of an input/output port, etc.

Because the operand may be used as data, it must be presented to the data bus via an open collector or three-state transmitter. The operand and its subfields must also be distributed to the other computer subsystems that it serves such as the register selectors in the arithmetic/logic unit. The decoding and use of the opcode, however, is not as obvious conceptually or from an implementation standpoint.

There is usually more than one microinstruction per macroinstruction. And, different classes of macroinstructions almost always require a different number of microprogram steps. The designer that is interested in a computer with only a few instructions may eliminate some hardware by using the opcode from the instruction register, C1, directly as the starting address of the microprogram ROM, C4. This is not only wasteful of opcodes and inflexible, but it means that any change in the instruction set or microprogram perturbs the entire system. To avoid this problem, a mapping ROM may be used.

The output of the mapping ROM, C2, should be wider than the opcode field that is used as the address input. This allows a greater range of starting address for the microprogram ROM, C4. Because ROM/PROM field widths are typically four or eight bits wide, a reasonable choice of width for the mapping ROM with an 8-bit opcode is twelve bits. The starting address is loaded into the microprogram counter, C3, which points to the first microinstruction in the microprogram ROM. When the output of the microprogram ROM stabilizes, it is loaded into the microprogram register, C5.

The use of the microprogram register in this manner is called pipelining. A pipeline register speeds up a state machine of this sort because it allows the address of the microprogram ROM to be changed and its output to settle, while the current microinstruction is being presented to the computer hardware from the microprogram register.

The microprogram sequence controller, C6, has two basic functions: it synchronizes events external to the CCU with the CCU, and it uses the output of the test condition multiplexer to determine whether or not microprogram branches, jumps to subroutine, and returns from subroutine are to be made.

The external signals in the microprogram sequence controller can be classified into five categories: supervisory, condition codes, initialization, synchronization, interrupts, and clocks. Supervisory signals include Run, Halt, and Pause. Run is a latched signal that enables the clock to the entire computer system. Halt disables the clock from the system, but it is only recognized during the instruction fetch microcycle; it, too, is latched. Pause is a level provided to the controller from an outside processor to temporarily suspend CCU control so that the external processor has uncontended access to the computer's resources. Pause is also only recognized during an instruction fetch microcycle.

Condition codes are stored in the program status word register, C8, and presented to the test condition multiplexer, C7, where any of the codes may be selected by one of the microprogram fields in the microprogram register. If true, the output of the test condition multiplexer will enable a branch instruction in the microprogram. The condition codes are loaded into the program status word register after every ALU operation or interrupt request.

Initialization lines include Power-Up Reset and Master Reset. The use of these lines was covered in some detail above.

The synchronization lines include Operation Request and Operation Acknowledge, OPREQ and OPACK. These signals allow external events that may be slower than the CCU to be synchronized to the CCU. For example, when the CCU issues a memory reference instruction, an OPREQ is also generated, and, although the system clock continues to run, it is disabled from the CCU. When the addressed memory bank has achieved its access time and performed the read or write operation, it must generate an OPACK which will be synchronized with the system clock which will, in turn, enable the clock to the CCU. When the memory or I/O cycle times are known and can be controlled, the CCU clock period can be adjusted to preclude the requirement for synchronizing signals.

An interrupt may occur at any time; however, it is only recognized at an instruction fetch microcycle. At the time the interrupt is allowed, the priority encoded interrupt vector is jammed into the program status word register and the microprogram ROM address is forced to the interrupt service routine address. When the interrupt has been serviced, the microprogram counter is returned to instruction fetch minor cycle address and processing resumes.

CCU Instructions

As implied earlier, there are two types of instructions recognized within the CCU, machine language or macroinstructions, and random logic replacement or microinstructions. Macroinstructions reside in main memory, are fetched and loaded into the instruction register and then decoded into microinstructions which directly control the computer's resources.

An example of two different types of macroinstructions may be seen in Table 8a. A 16-bit instruction was defined with a constant length op code defined in the least significant eight bits of the instruction. The remainder of the instruction word, bits 8 through 15, will be defined as a function of instruction type.

The register-to-register instruction has two operand fields that select the source and destination register, Register A and Register B, respectively. That is, the result of an arithmetic/logic function with Registers A and B will be stored in Register B.

The branch instruction's operand is an 8-bit displacement address. With the condition of the branch implicit in the op code, the sum of the current program counter address and the displacement address will be stored in the program counter, if the selected condition is logically true.

A microinstruction word format is depicted in Table 8b. Four bits, b0–3, are used to define the type of microinstruction being executed. The second field of four bits, b4–7, selects the branch condition, if the microinstruction is a branch instruction; enables the interrupt and pause functions, if the microinstruction is a macroinstruction fetch command; and, disables the interrupts at all other times. The third microinstruction field is composed of two 3-bit subfields which are used to define the

TABLE 8 – EXAMPLE MACRO AND MICROINSTRUCTION FIELDS

a. Macroinstructions

REGISTER-TO-REGISTER

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
OP CODE								REG B				REG A			

BRANCH-ON-CONDITION

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
OP CODE								DISPLACEMENT							

b. Microinstructions

ALU AND BRANCH ADDRESS BITS											
Function			Source			Destination			External Gating		
27	26	25	24	23	22	21	20	19	18	17	16

DATA BUS CONTROL					
Source			Destination		
15	14	13	12	11	10

SYNCHRONIZATION LOGIC									
ALU EN	Op Req	Branch Condition				Sequencer Function			
9	8	7	6	5	4	3	2	1	0

source and the destination of data on the data bus. The remaining 12-bit field is defined either as an arithmetic logic unit control field or as a microprogram branch address field, depending on the microinstruction function. Although there are a number of methods for mapping various types of microinstruction control fields into a microinstruction, this straightforward approach will be followed, for the purpose of an implementation example and only one mapped field function will be assumed: ALU control and branch address.

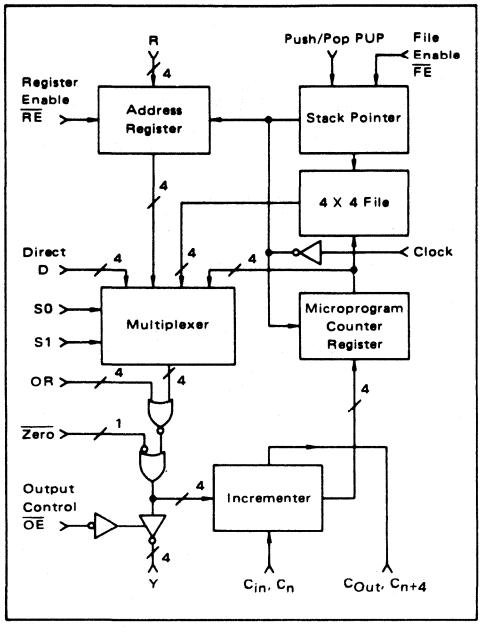
CCU Implementation Using MC2909

As an example, the computer control unit architectural schematic of Figure 6 will be reduced in practice to aid in the illustration of the MC2909 Microprogram Sequencer. The MC2909 is an extremely valuable subsystem component in that it allows the designer to take advantage of the latest microprogramming techniques: microbranching, microsubroutines, and repetitive microinstruction execution. Also, because of the architecture of the component itself, the CCU is inherently faster than a classical implementation of the same function. That is, the classical design may use sequential circuits which must be parallel loaded and sequentially incremented with separate clock pulses, while the MC2909 uses a combinational incrementor outside of the microprogram address bus which is transferred to the microprogram counter on the rising edge of the clock pulse. A detailed specification of the MC2909 is provided in the expanded data sheet and its internal architectural rendering is reproduced here in Figure 7.

The purpose of the MC2909 is to present an address to the microprogram ROM such that a microinstruction may be fetched and executed. In referencing Figure 7, there are four sources of address information available: an address register, a microprogram counter register, a direct or branch input, and a subroutine stack. The address source is chosen by using the one-of-four address multiplexer select lines, S0 and S1. The selected address may then be modified by the OR input lines or the $\overline{\text{Zero}}$ input function before it is presented at the Y address output lines through a three-state buffer.

The OR input lines may be used in one of two manners. Selected OR inputs may be placed at logic "1" which will provide the logical OR of the selected address source and the OR input lines at the Y output. This allows the address to be "masked". If a microprogram instruction of the Skip or Branch classes is being executed and the microinstruction is aligned on an even address microprogram ROM word (the least significant address bit is

FIGURE 7 – MC2909 MICROPROGRAM SEQUENCER ARCHITECTURE



"0"), then the least significant OR input may be controlled by an external test condition multiplexer. If the result of the conditional test was logically false, then the least significant bit may be modified to avoid the execution of the Branch or Skip instruction. All of the unused inputs must be tied to ground. Similarly, if the 2, 3, 4, or n least significant bits of the selected address are "0", the associated OR input lines may be modified for an extended address range skip capability.

Sometimes, in a state machine like a computer control unit, it is desirable to get easily to a predefined state, or address. For instance, if the machine has just been turned on and it is necessary to perform an initialization sequence, or a real-time event occurs where the processor control is required, but the on-going process information may not be destroyed such as an interrupt, the OR inputs may be used. All of the OR inputs must be connected to the output of a positive logic gate so that when the event occurs the output of the gate goes to logical "1", as does the Y output address lines. $\overline{\text{Zero}}$ provides a similar capability, but it must normally be held at logic "1" and only "pulled down" to "0" when the event occurs–causing all of the address output lines to go to "0".

The three-state output buffer that drives the Y-lines may be used nicely to allow automatic test of the memory and register system. That is, if the buffer output control, $\overline{\text{OE}}$, is disabled, the Y-lines go into a high-impedance condition allowing the automatic tester's output lines to be connected directly across the outputs. This capability also allows multiple processors to share the same memory by enabling only one processor's Y-bus at a time.

The address register, as well as all other storage devices on the MC2909, is parallel loaded from the R inputs when the register enable line, $\overline{\text{RE}}$, is low on a positive going clock transition. This is a good register to use when entering the starting address of a microprogram. If selected, the contents of the register are not only presented to the Y outputs, but also to the incrementer.

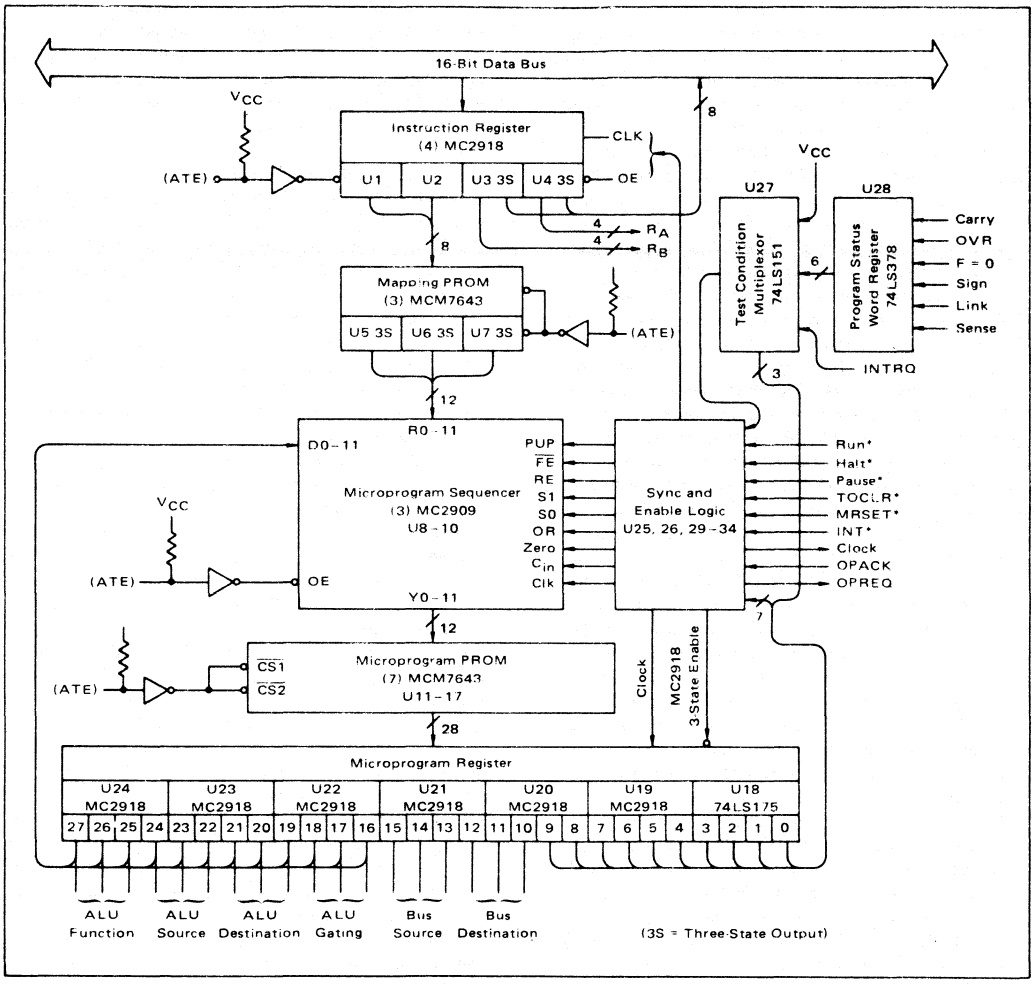
The incrementer is a full-adder provided with an off-chip carry-in signal, C_{in}, and an off-chip carry-out signal, C_{out}, allowing multiple MC2909s to be cascaded. The output from the incrementer is connected to a parallel load input on the microprogram counter register where it is loaded on the rising edge of the next clock pulse. If the microprogram counter is selected as the source address by subsequent microinstructions, it will be incremented by each succeeding clock pulse, thereby stepping through the microprogram.

As described above, it is often valuable to provide a branch instruction and a branch address in a microprogram instruction. The data lines from the branch address field in the microinstrucion may be feedback to the direct input, D, on the MC2909. The source address multiplexer may then select the branch input as the next microinstruction address. This address will be incremented and stored in the microprogram counter register on the next clock pulse which provides the address for the following instruction.

The push/pop, or last-in, first-out stack, allows the microprogrammer to have the same subroutine execution flexibility that machine language programmers have. Heretofore a luxury in almost all computers, microsubroutines may be nested four deep. There is a 4-bit wide by 4-word file whose address is controlled by a 2-bit up/down counter. A push/pop control signal, PUP, determines whether the function being performed is a jump to subroutine, Push, or a return from subroutine, Pop. When the file enable control line, $\overline{\text{FE}}$ is low, the push/pop command will be executed on the next clock pulse rising edge. After the subroutine has been completed, a return to the address immediately following the jump-to-subroutine instruction may be accomplished by selecting the stack as the source address and executing a Pop at the same time.

An example implementation of the computer control unit of Figure 6 using the macro and microinstruction form of Table 8 is depicted in Figure 8. A 16-bit data bus and memory word were assumed as reflected by the instruction register. Four MC2918 4-bit, TTL/three-state

FIGURE 8



output registers, U1–4, are used for the instruction register. The two least significant registers, U1 and U2, contain the op code, while U3 and U4 contain the operand field. The TTL output from the op code register pair is not used, but rather the three-state outputs are connected to the address input of the macroinstruction mapping PROM. If the output enable, OE, of the pair is held low by pulling up the input of an inverter, as shown, then the troubleshooting and automatic testing of the subsystem will be much simpler. In this way, the tester can gain control over the memory system. The three-state output buffers for the operand field are fed back to the eight least significant bits of the data bus so that they may be used to modify the contents of some other register in the system. For ALU functions, the operand field will most likely be used as two 4-bit subfields to specify a source register, R_A, and a source/ destination register, R_B. (In fact, this arrangement works extremely well, if the MC2901 Microprocessor is employed.) The TTL outputs are used for R_A and R_B data.

Selected for their speed and architecture, the mapping PROMs used are three MCM7643s in parallel. With a memory configuration of 256 words by 4 bits, each of the 256 potential op codes has a unique 12-bit starting address which provides the designer with a lot of flexibility for his initial design and an unusally easy task of adding more instructions at a later date.

In turn, the mapping PROM outputs are connected to the address register inputs, R0–11, of the three MC2909 Microprogram Sequencers. The Microprogram Sequencer outputs, Y0–11, provide the address inputs for seven MCM7643 microprogram PROMs. (The output enable lines of the MC2909s should be controlled in the same manner as the instruction register outputs.) Although only 1024 words of microprogram storage are shown, up to 4096 words may be implemented, if necessary. Furthermore, if more than 18-bit microinstruction words are required for the user's task, they may be added as necessary.

The microprogram register consists of one 74LS175 register, U18, and six MC2918 registers, U19–24. The ten least significant bits are used by the synchronization and enable logic. The most significant twelve bits are used for either microsequencer branch address (the TTL outputs of the MC2918s) or for control of the ALU. (The three-state lines are used.) As shown, the ALU control bit fields are specified to control four MC2901s and perform all of the necessary external gating and bit manipulation. The remaining six central bits are provided for data bus source and destination controls and the three-state outputs are used. Whenever the processor is running, the three-state output enable lines are held low, enabling the output. If the processor has been paused, ostensibly for direct memory access, the outputs are disabled so that an external or peripheral processor can gain access to the control line.

The Sync and Enable logic is relatively complex and may be shown better in Figure 9. The two least significant registers in the microprogram registers, U18 and U19, are at the top of the figure. In addition, the remaining two bits from the microprogram register that are used here, ALUEN* and OPREQ, are shown with rectangular boxes around them so that they are easy to see. The control and status bits that emanate from portions of the computer other than the CCU are shown enclosed in ovals. All other signals are generated or used with the CCU.

The four bits stored in microprogram register U18 provide the Microprogram Sequencer function instruction. A 74LS161 was selected for this register because it is synchronous, has an asynchronous clear (enabling power-up reset), and is low-power Schottky. These four bits provide the least significant address bits of an MCM5303 64-word by 8-bit PROM, as well as providing an external event synchronizing signal enable, XSYNC. With XSYNC at logic "1", the external processes that use Pause will be enabled, allowing direct memory access.

The MCM5303 PROM has eight open-collector outputs that must have pull-up resistors added. Seven of the output signals go directly to the three sets of MC2909 control lines. The eighth output line is fed back to enable a gate that drives the fifth PROM address line. The other signal at the AND-gate, U34, is the test condition enable line.

The test condition signal is the result of selecting one of eight processor condition signals using the 74LS151 multiplexer U27. U27 has two outputs, the selected signal and its complement. One of these two signals is selected using (1/2) 74LS51 U29 as the multiplexer. The 4-bit select signal is stored in microprogram register U19. Notice that one of the condition code multiplexer's inputs is tied to V_{CC} which provides for an unconditional branch, if the entire register, U19, is "0". Six of the condition codes (from the ALU) are stored in a 74LS157 register. Every time an ALU function is selected and clocked, as denoted by ALUEN*, the current value of the condition codes are clocked into U28. The eighth condition code bit is the interrupt request signal INTRQ, which is latched externally.

Before proceeding with the rest of the synchronization and enable circuitry, let us consider the programmability aspects of this portion of the state machine. A table of desirable Microprogram Sequencer functions is provided in Table 9. In fact, this table is also the memory map for the Microprogram Sequencer PROM U30. Entries are made in the table by U30 address value. The first 16 entries have the test condition address bit equal to "0". These are the primary instructions; they enable the MC2909 functions. Of the primary instructions, there are only four that have O7, the test condition enable bit set. Three of these are branch instructions and for these microinstructions any condition code may be specified in register U20, except interrupt request, INTRQ. The remaining instruction relates to the macroinstruction fetch process (Figure 5) and only during this microcycle may the CCU be interrupted or paused, as data or instructions moved to system registers under microprogram control may be lost if the microprocess is disturbed. Processed interrupts, by definition, disturb the microprocess and may usurp control of any of the computer's resources.

The only instance when there will be a secondary instruction defined is when bit O7 equals "1" such that, potentially, A5 equals "1". In the event that the condition code test is successful, the four secondary instructions defined in Table 9 will be executed.

If an interrupt was generated during an instruction fetch, the OR output U30,. O6 will assume logic "1". This signal will be logically ANDed by U34 to generate the interrupt acknowledge signal, INTAC. INTRQ or the

FIGURE 9 – SYNC AND ENABLE LOGIC DIAGRAM

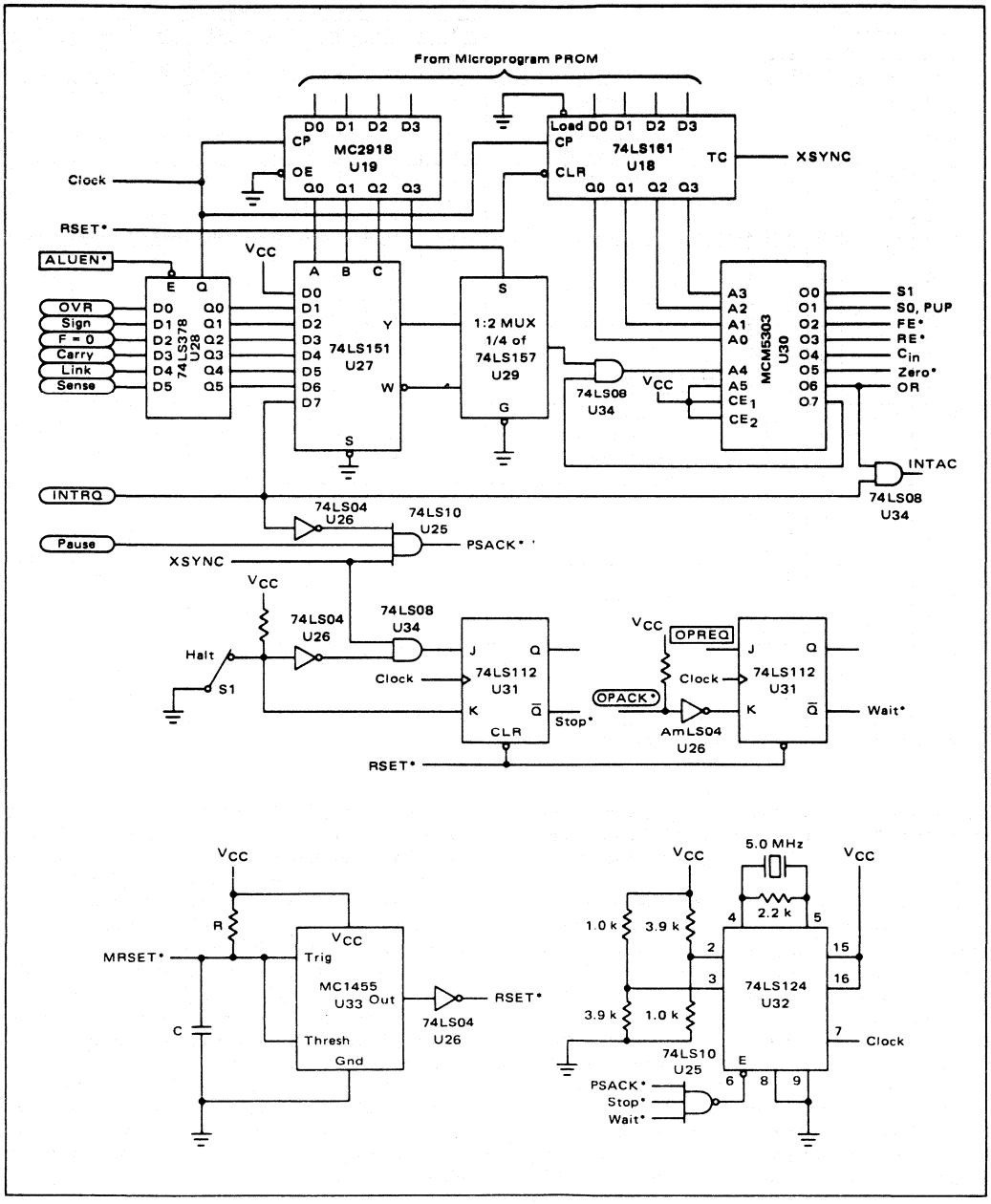


TABLE 9 – MICROSEQUENCER FUNCTION TABLE

Microprogram Sequencer Function						Microprogram Sequencer Control							
						O7	O6	O5	O4	O3	O2	O1	O0
A4	A3	A2	A1	A0	Function Description	Test Enable	OR	$\overline{Zero}$	C_{in}	$\overline{RE}$	$\overline{FE}$	S0, PUP	S1
0	0	0	0	0	Initialize System	L	X	L	H	H	H	X	X
0	0	0	0	1	Branch Test	H	L	H	H	H	H	L	L
0	0	0	1	0	Jump to Subroutine Test	H	L	H	H	H	H	L	L
0	0	0	1	1	Return from Subroutine	H	L	H	H	H	H	L	L
0	0	1	0	0	Execute Program	L	L	H	H	H	H	L	L
0	0	1	0	1	External Carry Control	L	L	H	*	H	H	L	L
0	0	1	1	0		L	L	H	H	H	H	L	L
0	0	1	1	1		L	L	H	H	H	H	L	L
0	1	0	0	0		L	L	H	H	H	H	L	L
0	1	0	0	1	This Group Undefined	L	L	H	H	H	H	L	L
0	1	0	1	0		L	L	H	H	H	H	L	L
0	1	0	1	1		L	L	H	H	H	H	L	L
0	1	1	0	0		L	L	H	H	H	H	L	L
0	1	1	0	1		L	L	H	H	H	H	L	L
0	1	1	1	0	Load Mapped (Starting) Address	L	L	H	H	L	H	H	L
0	1	1	1	1	Fetch Instruction	H	L	H	H	H	H	L	L
1	0	0	0	0	This State Undefined	L	–	–	–	–	–	–	–
1	0	0	0	1	Execute Branch	H	L	H	H	H	H	H	H
1	0	0	1	0	Execute Jump	H	L	H	H	H	L	H	H
1	0	0	1	1	Execute Return	H	L	H	H	H	L	L	H
1	0	1	0	0		L	–	–	–	–	–	–	–
1	0	1	0	1		L	–	–	–	–	–	–	–
1	0	1	1	0		L	–	–	–	–	–	–	–
1	0	1	1	1		L	–	–	–	–	–	–	–
1	1	0	0	0		L	–	–	–	–	–	–	–
1	1	0	0	1	This State Undefined	L	–	–	–	–	–	–	–
1	1	0	1	0		L	–	–	–	–	–	–	–
1	1	0	1	1		L	–	–	–	–	–	–	–
1	1	1	0	0		L	–	–	–	–	–	–	–
1	1	1	0	1		L	–	–	–	–	–	–	–
1	1	1	1	0		L	–	–	–	–	–	–	–
1	1	1	1	1	Service Interrupt or Pause	H	H	H	H	H	H	X	X

*Value of this Bit depends on Logic Implementation. See Text.

absence of the granting of the external synchronization signal, XSYNC, that is generated during the instruction fetch, will preclude the pause acknowledge signal, PSACK*.

An attempt to Halt the processor using an external switch S1 will also be denied unless the current microinstruction cycle is a macroinstruction fetch. Starting the processor by moving switch S1 to the Run position will always be granted and synchronized by U31, because by definition the processor stopped previously at an instruction fetch cycle, which is also the first state which must be executed when the processor is turned on. If Stop, the Q output of U31 is logic "0", the processor will Halt.

Provision has been made to synchronize the relatively fast CCU with relatively slow memory or input/output functions. If a microinstruction causes memory or I/O reference, the microprogrammer must set the OPREQ bit true. This signal is latched up in the second half of flip-flop U31 and stops the processor (Wait = "0") until the address device acknowledges the operation is complete or the data is ready, by pulling down the OPACK* line. U31 synchronizes the event and restarts the processor.

We have discussed a "hard" processor interrupt event, INTRQ, a "soft" interrupt (one where the state of the processor is not disturbed, but operation is suspended), Pause, starting and stopping the processor, Run and Halt, and synchronizing the processor with external events, OPREQ/OPACK*. Let us consider the system clock and system initialization.

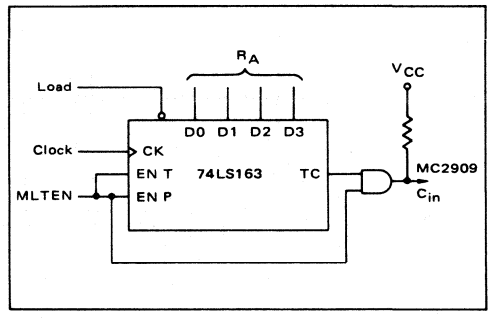
A good choice for a system oscillator is the 74LS124, U32. This device is a dual voltage-controlled oscillator whose timing may be derived by a series mode, fundamental frequency crystal or an RC timing circuit. For high-speed digital circuits, it is necessary to use a crystal.

The output of the oscillator is free running and is presented to an on-chip pulse synchronizer controlled by the enable line of the chip. No partial pulses can be passed by the synchronizer so that control of the oscillator's enable line may be asynchronous. A Pause, Halt, or OPREQ will cause the gated oscillator to shut off until the process is allowed to restart.

Initialization of a state machine is very important. When the power is applied to the system, the system must be disabled until all of the power supply filters have charged and the regulators stabilized. Also, critical storage devices must be preset to a known state. To accomplish this, an MC1455 timer circuit U33 with output buffer U26 is used. The RC value should yield a time-out circuit increment greater than 100 ms in most systems (t = 1.1 RC for this device). Also, by using an external switch or open collector gate to ground an MRSET*, the entire system may be master reset without cycling the power supply off and then on again. Other than clearing flip-flops, the output of the intialization circuit RSET* clears the microprogram register U19. By referring to Table 9 again, it can be seen that microsequencer function A0–3 = 0 provides system initiation, in that U30 output bit O5, Zero*, is low, thereby setting the initial microprogram address to zero and incrementing from there. This allows an initialization microprogram to be stored in the bottom of memory.

The ability to execute the same microinstruction a number of times was referred to previously. Generally, the value of this capability lies outside of the CCU. As an example, let us reconsider the macroinstruction format for a Register-to-Register instruction (Table 8). If the op code is a Shift or Rotate instruction, it would be desirable to allow the programmer to move the data word over a range of one to sixteen bit positions with a single instruction, rather than having to execute the same instruction many times. Since there are two operand subfields, R_A and R_B, let us define the 4-bit value in R_A as the number of bit positions we wish to move the data, and R_B as the general-purpose register that will be affected. (The additional hardware to implement the circuit is shown in Figure 10.)

FIGURE 10 – ITERATIVE MICROINSTRUCTION CONTROL CIRCUIT EXAMPLE



The value in R_A must be parallel loaded into a 4-bit binary counter that has a terminal count flag, TC (when Q0 = Q1 = Q2 = Q3 = 1, TC = 1), such as the 74LS163. The MC2909 control signal RE* that loads the address register must also be applied to the 74LS163 signal LOAD*. Clock is merely the system clock, and MLTEN is a signal that must be supplied by the microprogram to enable this function. MLTEN and TC are connected to an open-collector AND-gate which pulls downthe MC2909 carry-in line until terminal count has been achieved. As a result, the microprogram address does not change until TC equals "1" and then C_{in} equals "1", which increments the microprogram counter, causing the next instruction to be executed. The number of reasons for using this feature are almost unlimited, as are the means to implement the function.

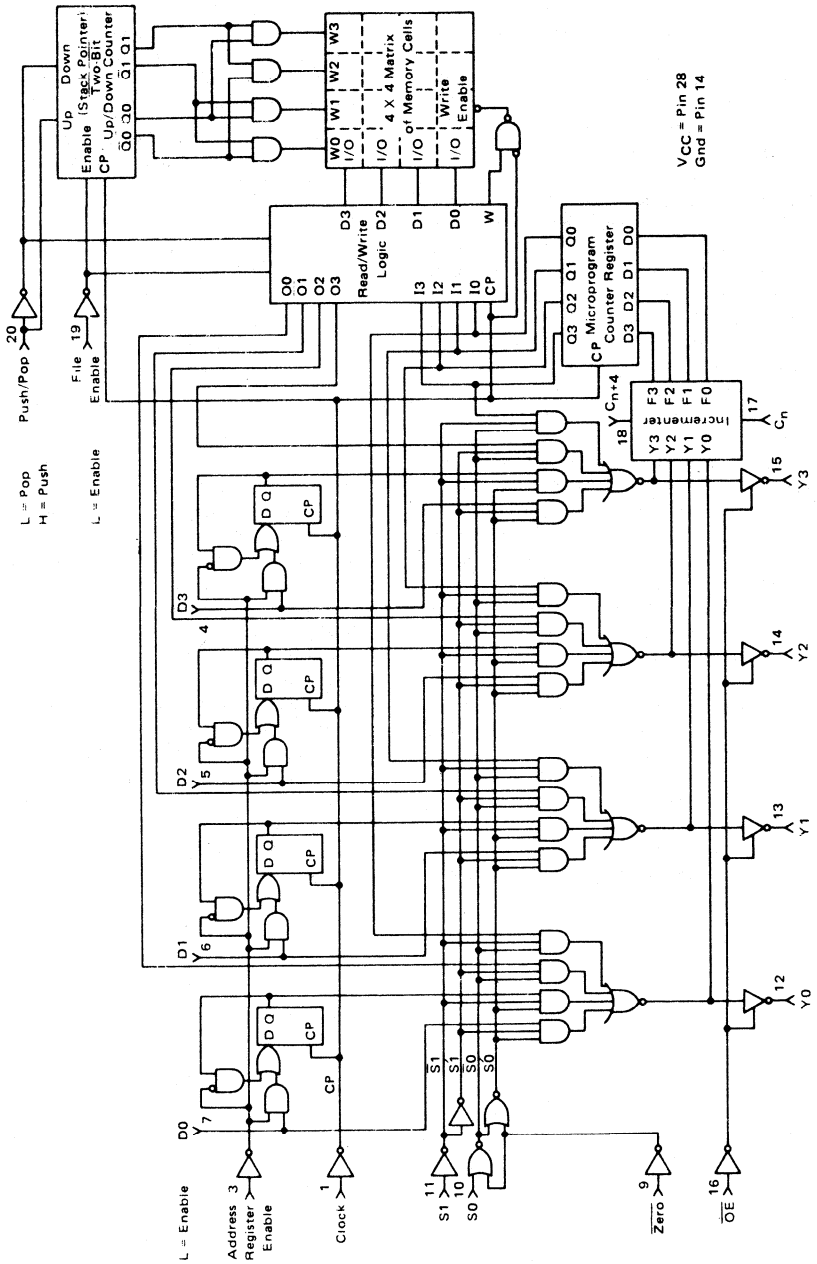
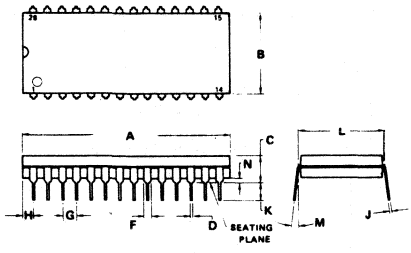


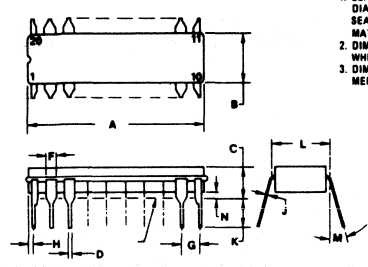
FIGURE 11 – MC2911 MICROPROGRAM SEQUENCER BLOCK DIAGRAM

PACKAGE DIMENSIONS



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	36.32	37.34	1.430	1.470
B	13.72	14.22	0.540	0.560
C	3.94	5.08	0.155	0.200
D	0.36	0.56	0.014	0.022
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.38	0.008	0.015
K	2.92	3.43	0.115	0.135
L	15.24 BSC		0.600 BSC	
M	0°	15°	0°	15°
N	0.51	1.02	0.020	0.040

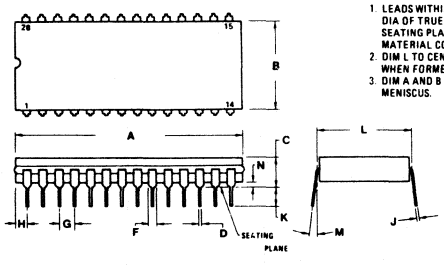
CASE 710-01



NOTES:
1. LEADS WITHIN 0.25 mm (0.010) DIA, TRUE POSITION AT SEATING PLANE, AT MAXIMUM MATERIAL CONDITION.
2. DIM L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIM A AND B INCLUDES MENISCUS.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	24.38	25.15	0.960	0.990
B	6.86	7.49	0.270	0.295
C	4.32	5.08	0.170	0.200
D	0.38	0.56	0.015	0.022
F	1.40	1.65	0.055	0.065
G	2.54 BSC		0.100 BSC	
H	0.89	1.40	0.035	0.055
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	7.62 BSC		0.300 BSC	
M	5°	15°	5°	15°
N	0.51	0.76	0.020	0.030

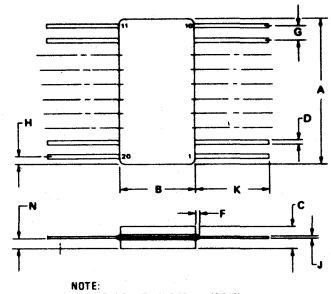
CASE 732-02



NOTES:
1. LEADS WITHIN 0.25 mm (0.010) DIA OF TRUE POSITION OF SEATING PLANE, AT MAXIMUM MATERIAL CONDITION.
2. DIM L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIM A AND B INCLUDES MENISCUS.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	36.83	37.59	1.450	1.480
B	12.70	13.46	0.500	0.530
C	5.08	5.84	0.200	0.230
D	0.38	0.56	0.015	0.022
F	1.27	1.65	0.050	0.065
G	2.54	BSC	0.100	BSC
H	2.03	2.29	0.080	0.090
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	15.24	BSC	0.600	BSC
M	5°	15°	5°	15°
N	0.51	1.27	0.020	0.050

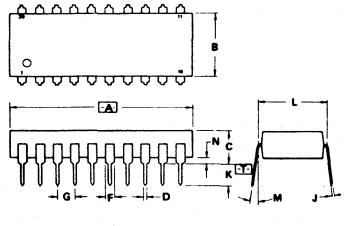
CASE 733-01



NOTE:
1. LEADS WITHIN 0.25 mm (0.010) TOTAL OF TRUE POSITION AT MAXIMUM MATERIAL CONDITION.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	–	13.08	–	0.515
B	5.84	6.60	0.230	0.260
C	1.52	2.16	0.060	0.085
D	0.41	0.46	0.016	0.018
F	–	0.25	–	0.010
G	1.27 BSC		0.050 BSC	
H	1.14	1.40	0.045	0.055
J	0.08	0.13	0.003	0.005
K	–	9.14	–	0.360
N	–	1.02	–	0.040

CASE 737-01



CASE 738-01

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	25.85	27.18	1.010	1.070
B	6.10	6.80	0.240	0.260
C	3.94	4.19	0.155	0.165
D	0.38	0.56	0.015	0.022
F	1.27	1.78	0.050	0.070
G	2.54 BSC		0.100 BSC	
J	0.20	0.38	0.008	0.015
K	2.79	3.56	0.110	0.140
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	1.02	0.020	0.040

NOTES:
1. DIM A IS DATUM.
2. POSITIONAL TOL FOR LEADS, ⌖ Ø 0.25 (0.010) Ⓜ T A Ⓜ
3. T IS SEATING PLANE.
4. DIM "B" DOES NOT INCLUDE MOLD FLASH.
5. DIM L TO CENTER OF LEADS WHEN FORMED PARALLEL.
6. DIMENSIONING AND TOLERANCING PER ANSI Y14.5, 1973.

MOTOROLA

MC2910

Product Preview

TTL

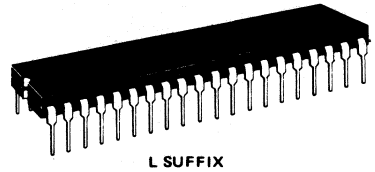
MICROPROGRAM CONTROLLER

MICROPROGRAM CONTROLLER

The MC2910 Microprogram Controller is an address sequencer intended for controlling the sequence of execution of microinstructions stored in microprogram memory. Besides the capability of sequential access, it provides conditional branching to any microinstruction within its 4096-microword range. A last-in, first-out stack provides microsubroutine return linkage and looping capability; there are five levels of nesting of microsubroutines. Microinstruction loop count control is provided with a count capacity of 4096.

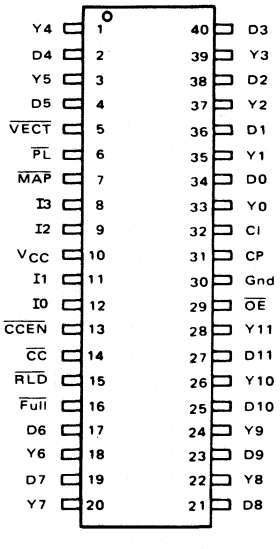
During each microinstruction, the Microprogram Controller provides a 12-bit address from one of four sources: 1) the microprogram address register, μPC, which usually contains an address one greater than the previous address; 2) an external (direct) input, D; 3) a register/counter, R, retaining data loaded during a previous microinstruction; or, 4) a five-deep last-in, first-out stack, F.

- **Twelve Bits Wide**
  Address up to 4096 words of microcode with one chip. All internal elements are a full 12 bits wide.
- **Internal Loop Counter**
  Pre-settable 12-bit down-counter for repeating instructions and counting loop iterations.
- **Four Address Sources**
  Microprogram address may be selected from microprogram counter, branch address bus, five-level push/pop stack, or internal holding register.
- **Sixteen Powerful Microinstructions**
  Executes 16 sequence control instructions, most of which are conditional on external condition input, state of internal loop counter, or both.
- **Output Enable Controls for Three Branch Address Sources**
  Built-in decoder function to enable external devices onto branch address bus. Eliminates external decoder.
- **All Registers Positive Edge-Triggered**
  Simplifies timing problems. Eliminates long setup times.
- **Fast Control from Condition Input**
  Delay from condition code input to address output only 27 ns typical.



L SUFFIX
CERAMIC PACKAGE
CASE 734

PIN ASSIGNMENT



ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Hermetic DIP	0°C to +70°C	MC2910LC
Hermetic DIP	-55°C to +125°C	MC2910LM

FIGURE 1 – BLOCK DIAGRAM

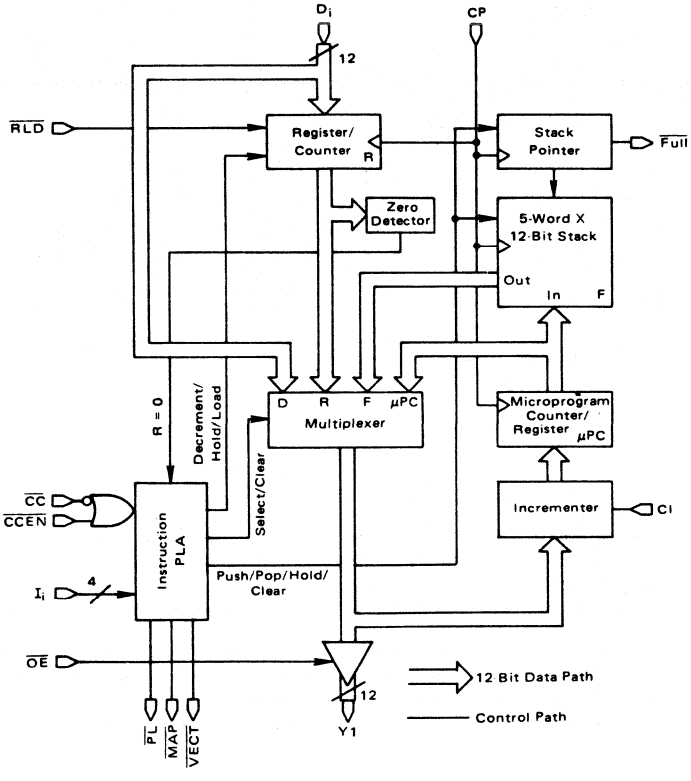


TABLE 1 – INSTRUCTIONS

Hex I3–I0	Mnemonic	Name	Reg/ CNTR Contents	Fail $\overline{CCEN}$ = Low and $\overline{CC}$ = High		Pass $\overline{CCEN}$ = High or $\overline{CC}$ = Low		Reg/ CNTR	Enable
				Y	Stack	Y	Stack		
0	JZ	JUMP ZERO	X	0	Clear	0	Clear	Hold	PL
1	CJS	COND JSB PL	X	PC	Hold	D	Push	Hold	PL
2	JMAP	JUMP MAP	X	D	Hold	D	Hold	Hold	MAP
3	CJP	COND JUMP PL	X	PC	Hold	D	Hold	Hold	PL
4	PUSH	PUSH/COND LD CNTR	X	PC	Push	PC	Push	Note 1	PL
5	JSRP	COND JSB R/PL	X	R	Push	D	Push	Hold	PL
6	CJV	COND JUMP VECTOR	X	PC	Hold	D	Hold	Hold	VECT
7	JRP	COND JUMP R/PL	X	R	Hold	D	Hold	Hold	PL
8	RFCT	REPEAT LOOP, CNTR ≠ 0	≠ 0	F	Hold	F	Hold	Dec	PL
			= 0	PC	Pop	PC	Pop	Hold	PL
9	RPCT	REPEAT PL, CNTR ≠ 0	≠ 0	D	Hold	D	Hold	Dec	PL
			= 0	PC	Hold	PC	Hold	Hold	PL
A	CRTN	COND RTN	X	PC	Hold	F	Pop	Hold	PL
B	CJPP	COND JUMP PL & POP	X	PC	Hold	D	Pop	Hold	PL
C	LDCT	LD CNTR & CONTINUE	X	PC	Hold	PC	Hold	Load	PL
D	LOOP	TEST END LOOP	X	F	Hold	PC	Pop	Hold	PL
E	CONT	CONTINUE	X	PC	Hold	PC	Hold	Hold	PL
F	TWB	THREE-WAY BRANCH	≠ 0	F	Hold	PC	Pop	Dec	PL
			= 0	D	Pop	PC	Pop	Hold	PL

NOTE 1: If $\overline{CCEN}$ = Low and $\overline{CC}$ = High, hold, otherwise load. X = Don't Care

TABLE 2 – PIN FUNCTIONS

Abbreviation	Name	Function
D_i	Direct Input Bit i	Direct input to register/counter and multiplexer. D0 is LSB.
I_i	Instruction Bit i	Selects one of sixteen instructions.
$\overline{CC}$	Condition Code	Used as test criterion. Pass test is a Low on $\overline{CC}$.
$\overline{CCEN}$	Condition Code Enable	Whenever the signal is High, $\overline{CC}$ is ignored and the part operates as through $\overline{CC}$ were true (Low).
CI	Carry-In	Low order carry input to incrementer for microprogram counter.
$\overline{RLD}$	Register Load	When Low forces loading of register/counter, regardless of instruction or condition.
$\overline{OE}$	Output Enable	Three-state control of Y_i outputs.
CP	Clock Pulse	Triggers all internal state changes at Low-to-High edge.
V_{CC}	+5 Volts	
Gnd	Ground	
Y_i	Microrogram Address Bit i	Address to microprogram memory. Y0 is LSB, Y11 is MSB.
$\overline{Full}$	Full	Indicates that five items are on the stack.
$\overline{PL}$	Pipeline Address Enable	Can select #1 source (usually pipeline register) as direct input source.
$\overline{MAP}$	Map Address Enable	Can select #2 source (usually mapping PROM or PLA) as direct input source.
$\overline{VECT}$	Vector Address Enable	Can select #3 source (for example, Interrupt Starting Address) as direct input source.

ARCHITECTURE OF THE MC2910

The MC2910 is a bipolar microprogram controller intended for use in high-speed microprocessor applications. It allows addressing of up to 4K words of microprogram. A block diagram is shown in Figure 1.

The controller contains a four-input multiplexer that is used to select either the register/counter, direct input, microprogram counter, or stack as the source of the next microinstruction address.

The register/counter consists of 12 D-type, edge-triggered flip-flops, with a common clock enable. When its load control, $\overline{RLD}$, is Low, new data is loaded on a positive clock transition. A few instructions include load; in most systems, these instructions will be sufficient, simplifying the microcode. The output of the register/counter is available to the multiplexer as a source for the next microinstruction address. The direct input furnishes a source of data for loading the register/counter.

The MC2910 contains a microprogram counter (μPC) that is composed of a 12-bit incrementer followed by a 12-bit register. The μPC can be used in either of two ways. When the carry-in to the incrementer is High, the microprogram register is loaded on the next clock cycle with the current Y output word plus one ($Y + 1 \rightarrow \mu PC$). Sequential microinstructions are thus executed. When the carry-in is Low, the incrementer passes the Y output word unmodified so that μPC is reloaded with the same Y word on the next clock cycle ($Y \rightarrow \mu PC$). The same microinstruction is thus executed any number of times.

The third source for the multiplexer is the direct (D) input. This source is used for branching.

The fourth source available at the multiplexer input is a 5-word by 12-bit stack (file). The stack is used to provide return address linkage when executing microsubroutines or loops. The stack contains a built-in stack pointer (SP) which always points to the last file word written. This allows stack reference operations (looping) to be performed without a pop.

The stack pointer operates as an up/down counter. During microinstructions 2, 4, and 5, the PUSH operation is performed. This causes the stack pointer to increment and the file to be written with the required return linkage. On the cycle following the PUSH, the return data is at the new location pointed to by the stack pointer.

During five microinstructions, a POP operation may occur. The stack pointer decrements at the next rising clock edge following a POP, effectively removing old information from the top of the stack.

The stack pointer linkage is such that any sequence of pushes, pops, or stack references can be achieved. At RESET (instruction 0), the depth of nesting becomes zero. For each PUSH, the nesting depth increases by one; for each POP, the depth decreases by one. The depth can grow to five. After a depth of five is reached, $\overline{FULL}$ goes Low. Any further PUSHes onto a full stack overwrite information at the top of the stack, but leave the stack pointer unchanged. This operation will usually destroy useful information and is normally avoided. A POP from an empty stack may place non-meaningful data on the Y outputs, but is otherwise safe. The stack pointer remains at zero whenever a POP is attempted from a stack already empty.

The register/counter is operated during three microinstructions (8, 9, 15) as a 12-bit down counter, with result = zero available as a microinstruction branch test criterion. This provides efficient iteration of microinstructions. The register/counter is arranged such that if it is preloaded with a number N and then used as a loop termination counter, the sequence will be executed exactly N + 1 times. During instruction 15, a three-way branch under combined control of the loop counter and the condition code is available.

The device provides three-state Y outputs. These can be particularly useful in designs requiring automatic checkout of the processor. The microprogram controller outputs can be forced into the high-impedance state, and preprogrammed sequences of microinstructions can be executed via external access to the address lines.

OPERATION

Table 1 shows the result of each instruction in controlling the multiplexer which determines the Y outputs, and in controlling the three enable signals, $\overline{PL}$, $\overline{MAP}$, and $\overline{VECT}$. The effect on the register/counter and the stack after the next positive-going clock edge is also shown. The multiplexer determines which internal source drives the Y outputs. The value loaded into μPC is either identical to the Y output, or else one greater, as determined by CI. For each instruction, one and only one of the three outputs PL, MAP, and VECT is Low. If these outputs control three-state enables for the primary source of microprogram jumps (usually part of a pipeline register), a PROM which maps the instruction to a microinstruction starting location, and an optional third source (often a vector from a DMA or interrupt source), respectively, the three-state sources can drive the D inputs without further logic.

Several inputs, as shown in Table 2, can modify instruction execution. The combination CC High and $\overline{CCEN}$ Low is used as a test in 10 of the 16 instructions. $\overline{RLD}$, when Low, causes the D input to be loaded into the register/counter, overriding any HOLD or DEC operation specified in the instruction. $\overline{OE}$, normally Low, may be forced High to remove the Am2910 Y outputs from a three-state bus.

The stack, a 5-word last-in, first-out 12-bit memory, has a pointer which addresses the value presently on the top of the stack. Explicit control of the stack pointer occurs during instruction 0 (RESET), which makes the stack empty by resetting the SP to zero. After a RESET, and whenever else the stack is empty, the contents of the top of stack is undefined until a PUSH occurs. Any POPs performed while the stack is empty put undefined data on the F outputs and leave the stack pointer at zero.

Any time the stack is full (five more PUSHes than POPs have occurred since the stack was last empty), the $\overline{FULL}$ warning output occurs. This signal first appears on the microcycle after a fifth PUSH. No additional PUSH should be attempted onto a full stack; if tried, information within the stack will be overwritten and lost.

MOTOROLA

MC2915A

QUAD THREE-STATE BUS TRANSCEIVER WITH INTERFACE LOGIC

The MC2915A is a high-performance, low-power Schottky bus transceiver intended for bipolar or MOS microprocessor system applications. The device consists of four D-type edge-triggered flip-flops with a built-in two-input multiplexer on each. The flip-flop outputs are connected to four three-state bus drivers. Each bus driver is internally connected to the input of a receiver. The four receiver outputs drive four D-type latches that feature three-state outputs.

This LSI bus transceiver is fabricated using advanced low-power Schottky processing. All inputs (except the Bus inputs) are one LS unit load. The three-state bus output can sink up to 48 mA at 0.5 V maximum. The bus enable input ($\overline{BE}$) is used to force the driver outputs to the high-impedance state. When $\overline{BE}$ is High, the driver is disabled. The V_{OH} and V_{OL} of the bus driver are selected for compatibility with standard and low-power Schottky inputs.

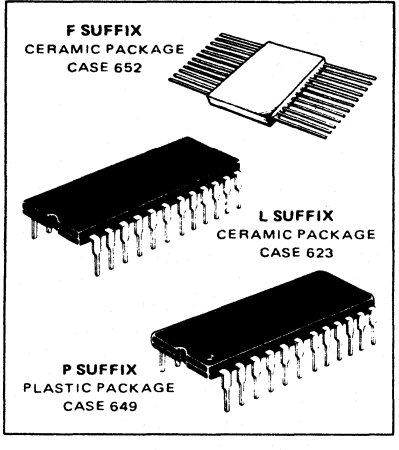
The input register consists of four D-type flip-flops with a buffered common clock and a two-input multiplexer at the input of each flip-flop. A common select input (S) controls the four multiplexers. When S is Low, the A_n data is stored in the register and when S is High, the B_n data is stored. The buffered common clock (DRCP) enters the data into this driver register on the Low-to-High transition.

Data from the A or B inputs is inverted at the Bus output. Likewise, data at the Bus input is inverted at the receiver output. Thus, data is non-inverted from driver input to receiver output. The four receivers each feature a built-in D-type latch that is controlled from the buffered receiver latch enable ($\overline{RLE}$) input. When the $\overline{RLE}$ input is Low, the latch is open and the receiver outputs will follow the bus inputs (Bus data inverted and OE Low). When the $\overline{RLE}$ input is High, the latch will close and retain the present data regardless of the bus input. The four latches have three-state outputs and are controlled by a buffered common three-state control ($\overline{OE}$) input. When $\overline{OE}$ is High, the receiver outputs are in the high-impedance state.

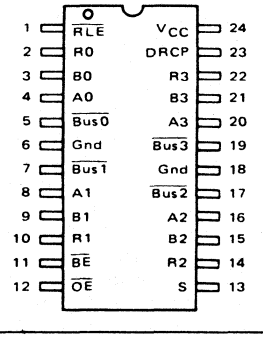
FEATURES

- Quad High-Speed LSI Bus Transceiver
- Three-state Bus Driver
- Two-port Input to D-type Register on Driver
- Bus Driver Output Can Sink 48 mA at 0.5 V Max
- Receiver Has Output Latch for Pipeline Operation
- Three-State Receiver Outputs Sink 12 mA
- Advanced Low-power Schottky Processing
- 100% Reliability Assurance Testing in Compliance With MIL-STD-883
- 3.5 V Minimum Output High Voltage for Direct Interface to MOS Microprocessors

TTL
QUAD THREE-STATE
BUS TRANSCEIVER WITH
INTERFACE LOGIC



PIN ASSIGNMENT



ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Molded DIP	0°C to +70°C	MC2915APC
Hermetic DIP	0°C to +70°C	MC2915ALC
Hermetic DIP	-55°C to +125°C	MC2915ALM
Hermetic Flat Pack	-55°C to +125°C	MC2915AFM

MAXIMUM RATINGS (above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential	−0.5 V to +7.0 V
DC Voltage Applied to Outputs for High Output State	−0.5 V to +V_{CC}max
DC Input Voltage	−0.5 V to +5.5 V
DC Output Current, into Outputs (Except Bus)	30 mA
DC Output Current, into Bus	100 mA
DC Input Current	−30 mA to +5.0 mA

ELECTRICAL CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise noted)

MC2915AXC − T_A = 0°C to +70°C, V_{CC} = 5.0 V ±5% (Commercial), Min = 4.75 V, Max = 5.25 V
MC2915AXM − T_A = −55°C to +125°C, V_{CC} = 5.0 V ±10% (Military), Min = 4.5 V, Max = 5.5 V

BUS INPUT/OUTPUT CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

Parameter	Description	Test Conditions (Note 1)		Min	Typ	Max	Unit
V_{OL}	Bus Output Low Voltage	V_{CC} = Min	I_{OL} = 24 mA			0.4	Volts
			I_{OL} = 48 mA			0.5	
V_{OH}	Bus Output High Voltage	V_{CC} = Min	Commercial, I_{OH} = −20 mA	2.4			Volts
			Military, I_{OH} = −15 mA	2.4			
I_O	Bus Leakage Current (High Impedance)	V_{CC} = Max, Bus Enable = 2.4 V	V_O = 0.4 V			−200	μA
			V_O = 2.4 V			50	
			V_O = 4.5 V			100	
I_{off}	Bus Leakage Current (Power off)	V_O = 4.5 V, V_{CC} = 0 V				100	μA
V_{IH}	Receiver Input High Threshold	Bus Enable = 2.4 V		2.0			Volts
V_{IL}	Receiver Input Low Threshold	Bus Enable = 2.4 V	Commercial			0.8	Volts
			Military			0.7	
I_{SC}	Bus Output Short Circuit Current	V_{CC} = Max, V_O = 0 V		−50	−120	−225	mA

DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

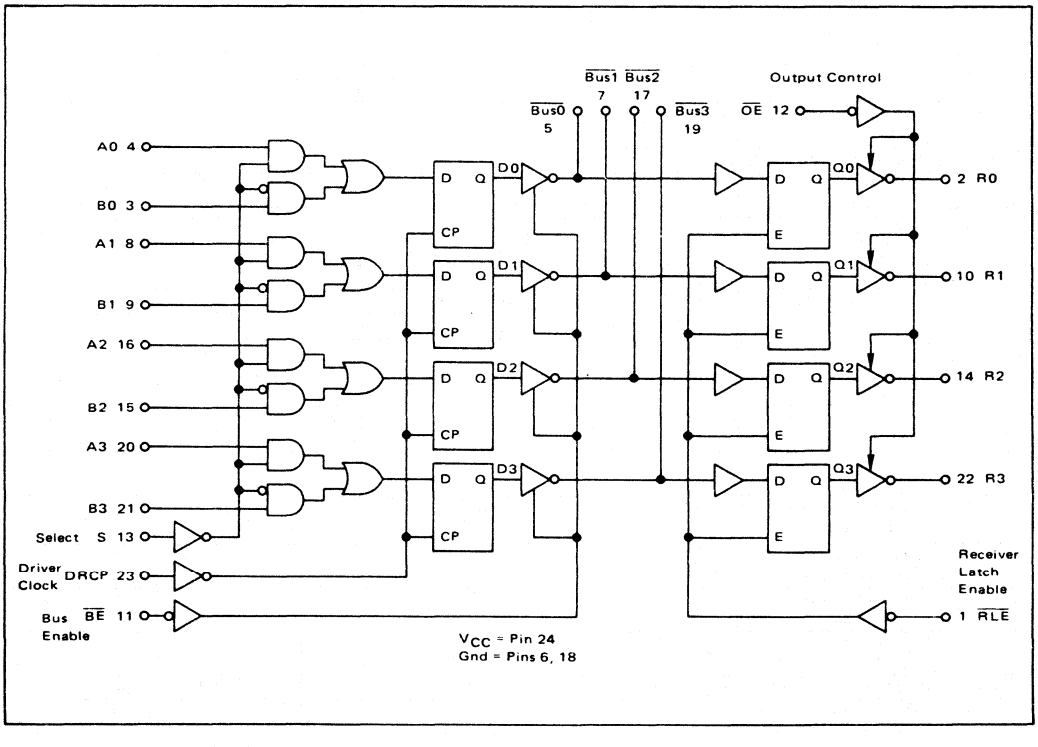
Parameter	Description	Test Conditions (Note 1)		Min	Typ (Note 2)	Max	Unit
V_{OH}	Receiver Output High Voltage	V_{CC} = Min, V_{in} = V_{IL} or V_{IH}	Military, I_{OH} = −1.0 mA	2.4	3.4		Volts
			Commercial, I_{OH} = −2.6 mA	2.4	3.4		
		V_{CC} = 5.0 V, I_{OH} = −100 μA		3.5			
V_{OL}	Output Low Voltage (Except Bus)	V_{CC} = Min, V_{in} = V_{IL} or V_{IH}	I_{OL} = 4.0 mA		0.27	0.4	Volts
			I_{OL} = 8.0 mA		0.32	0.45	
			I_{OL} = 12 mA		0.37	0.5	
V_{IH}	Input High Level (Except Bus)	Guaranteed input logical High for all inputs		2.0			Volts
V_{IL}	Input Low Level (Except Bus)	Guaranteed input logical Low for all inputs	Military			0.7	Volts
			Commercial			0.8	
V_I	Input Clamp Voltage (Except Bus)	V_{CC} = Min, I_{in} = −18 mA				−1.2	Volts
I_{IL}	Input Low Current (Except Bus)	V_{CC} = Max, V_{in} = 0.4 V	$\overline{BE}$, $\overline{RLE}$			−0.72	mA
			All other inputs			−0.36	
I_{IH}	Input High Current (Except Bus)	V_{CC} = Max, V_{in} = 2.7 V				20	μA
I_I	Input High Current (Except Bus)	V_{CC} = Max, V_{in} = 7.0 V				100	μA
I_{SC}	Output Short Circuit Current (Except Bus)	V_{CC} = Max		−30		−130	mA
I_{CC}	Power Supply Current	V_{CC} = Max			63	95	mA
I_O	Off-State Output Current (Receiver Outputs)	V_{CC} = Max	V_O = 2.4 V			50	μA
			V_O = 0.4 V			−50	

NOTES: 1. For conditions shown as Min or Max, use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0 V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.

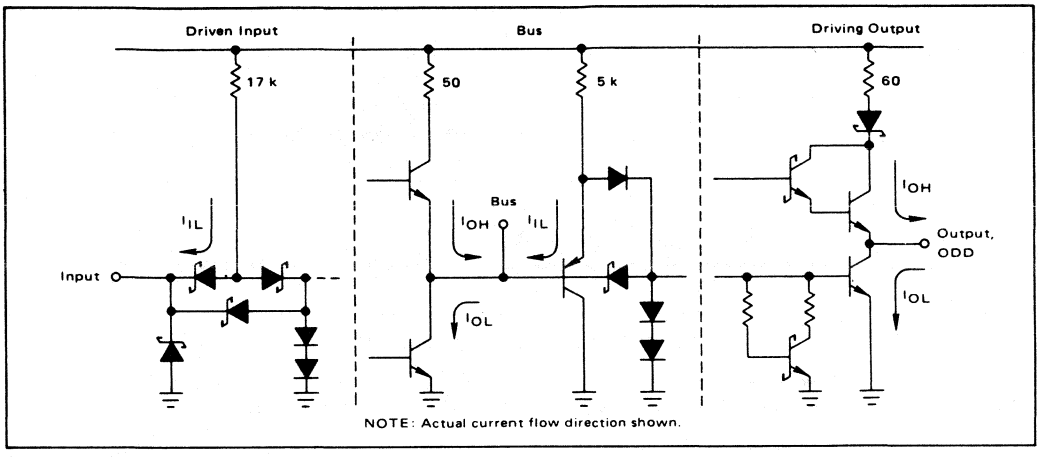
SWITCHING CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

Parameter	Description	Test Conditions	MC2915AXM			MC2915AXC			Unit
			Min	Typ (Note 2)	Max	Min	Typ (Note 2)	Max	
t_{PHL}	Driver Clock (DRCP) to Bus	C_L (Bus) = 50 pF	–	21	36	–	21	32	ns
t_{PLH}		R_L (Bus) = 130 Ω	–	21	36	–	21	32	
t_{ZH},t_{ZL}	Bus Enable ($\overline{BE}$) to Bus		–	13	26	–	13	23	ns
t_{HZ},t_{LZ}			–	13	21	–	13	18	
t_s	Data Inputs (A or B)	C_L = 15 pF	15	–	–	12	–	–	ns
t_h		R_L = 2.0 k	8.0	–	–	6.0	–	–	
t_s	Select Input (S)		28	–	–	25	–	–	ns
t_h			8.0	–	–	6.0	–	–	
t_{PW}	Driver Clock (DRCP) Pulse Width (High)		20	–	–	17	–	–	ns
t_{PLH}	Bus to Receiver Output (Latch Enabled)		–	18	33	–	18	30	ns
t_{PHL}			–	18	30	–	18	27	
t_{PLH}	Latch Enable to Receiver Output		–	21	33	–	21	30	ns
t_{PHL}			–	21	30	–	21	27	
t_s	Bus to Latch Enable ($\overline{RLE}$)		15	–	–	13	–	–	ns
t_h			6.0	–	–	4.0	–	–	
t_{ZH},t_{ZL}	Output Control to Receiver Output		–	14	26	–	14	23	ns
t_{HZ},t_{LZ}		C_L = 5 pF, R_L = 2.0 k	–	14	26	–	14	23	

LOGIC DIAGRAM



INPUT/OUTPUT CURRENT INTERFACE CONDITIONS



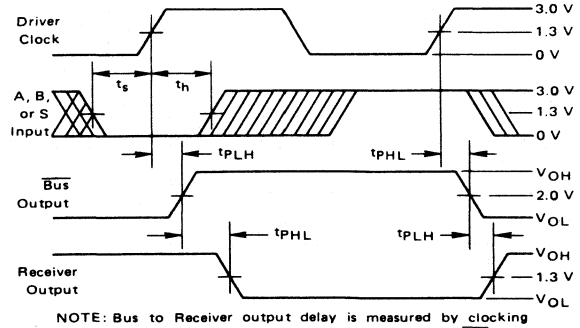
NOTE: Actual current flow direction shown.

FUNCTION TABLE

Inputs							Internal to Device		Bus	Output	
S	A_n	B_n	DRCP	$\overline{BE}$	$\overline{RLE}$	$\overline{OE}$	D_n	Q_n	$\overline{Bus}_n$	R_n	Function
X	X	X	X	H	X	X	X	X	Z	X	Driver output disable
X	X	X	X	X	X	H	X	X	X	Z	Receiver output disable
X	X	X	X	H	L	L	X	L	L	H	Driver output disable and receive data via Bus input
X	X	X	X	H	L	L	X	H	H	L	
X	X	X	X	X	H	X	X	NC	X	X	Latch received data
L	L	X	↑	X	X	X	L	X	X	X	Load driver register
L	H	X	↑	X	X	X	H	X	X	X	
H	X	L	↑	X	X	X	L	X	X	X	
H	X	H	↑	X	X	X	H	X	X	X	
X	X	X	L	X	X	X	NC	X	X	X	No driver clock restrictions
X	X	X	H	X	X	X	NC	X	X	X	
X	X	X	X	L	X	X	L	X	H	X	Drive Bus
X	X	X	X	L	X	X	H	X	L	X	

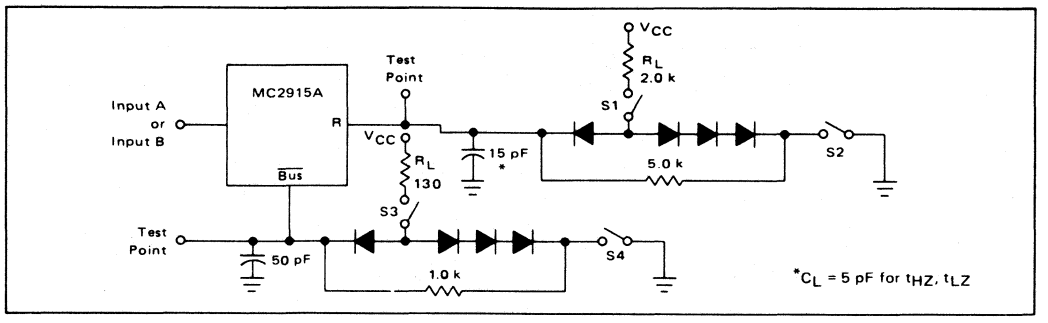
H = High Z = High impedance X = Don't care
L = Low NC = No change ↑ = Low-to-high transition

SWITCHING WAVEFORMS



NOTE: Bus to Receiver output delay is measured by clocking data into the driver register and measuring the $\overline{Bus}$ to R combinatorial delay.

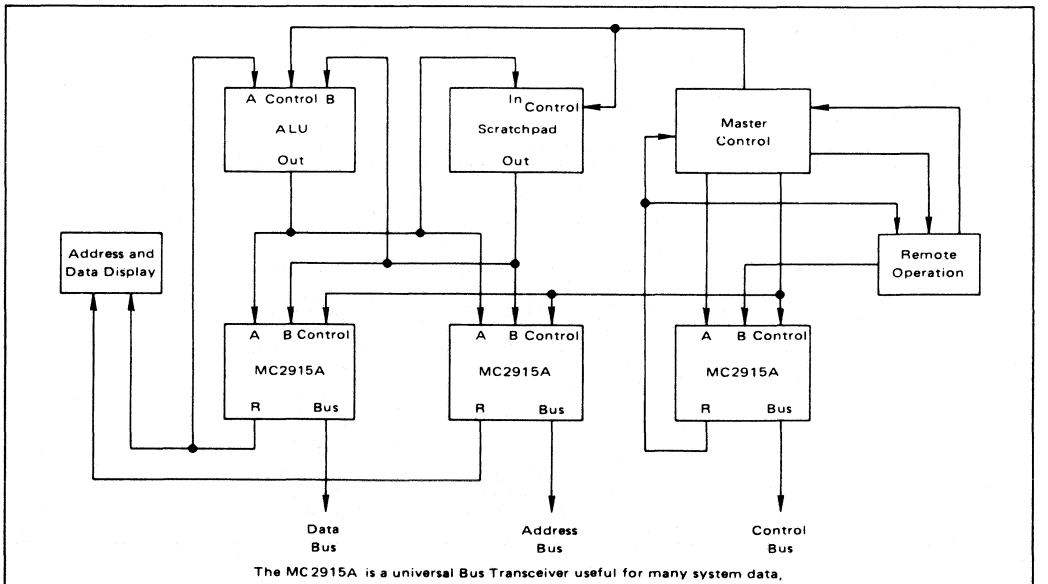
SWITCHING TEST CIRCUIT



DEFINITIONS OF FUNCTIONAL TERMS

A0, A1, A2, A3	The "A" word data input into the two input multiplexers of the driver register.
B0, B1, B2, B3	The "B" word data input into the two input multiplexers of the driver register.
S	Select. When the select input is Low, the A data word is applied to the driver register. When the select input is High, the B word is applied to the driver register.
DRCP	Driver Clock Pulse. Clock pulse for the driver register.
$\overline{BE}$	$\overline{\text{Bus Enable}}$. When the Bus Enable is High, the four drivers are in the high impedance state.
$\overline{Bus0}$, $\overline{Bus1}$, $\overline{Bus2}$, $\overline{Bus3}$	The four driver outputs and receiver inputs (data is inverted).
R0, R1 R2, R3	The four receiver outputs. Data from the bus is inverted while data from the A or B inputs is non-inverted.
$\overline{RLE}$	$\overline{\text{Receiver Latch Enable}}$. When $\overline{RLE}$ is Low, data on the Bus inputs is passed through the receiver latches. When $\overline{RLE}$ is High, the receiver latches are closed and will retain the data independent of all other inputs.
$\overline{OE}$	$\overline{\text{Output Enable}}$. When the $\overline{OE}$ input is High, the four three-state receiver outputs are in the high impedance state.

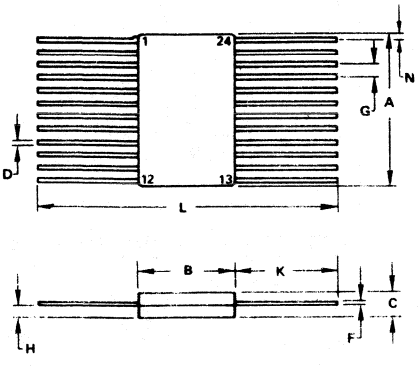
APPLICATIONS



The MC2915A is a universal Bus Transceiver useful for many system data, address, control, and timing input/output interfaces.

PACKAGE DIMENSIONS

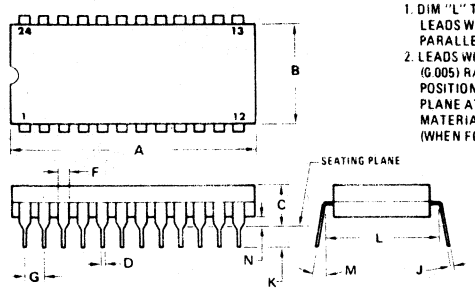
CASE 652



NOTES:
1. LEADS WITHIN 0.25 mm (0.010) TOTAL OF TRUE POSITION AT MAXIMUM MATERIAL CONDITION.

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	14.99	15.49	0.590	0.610
B	9.27	9.91	0.365	0.390
C	1.27	2.03	0.050	0.080
D	0.38	0.48	0.015	0.019
F	0.08	0.15	0.003	0.006
G	1.27 BSC		0.050 BSC	
H	0.69	1.02	0.027	0.040
K	6.35	9.40	0.250	0.370
L	21.97	–	0.865	–
N	0.25	0.63	0.010	0.025

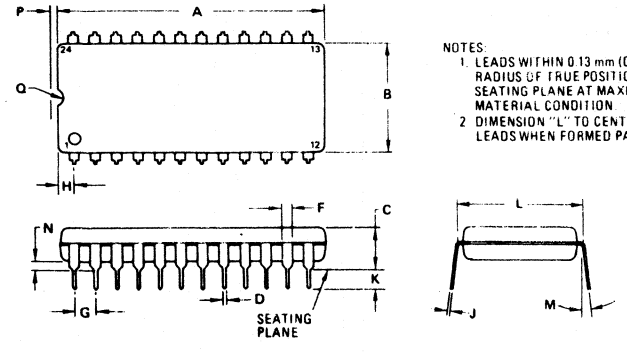
CASE 623



NOTES:
1. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.
2. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION. (WHEN FORMED PARALLEL)

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	31.24	32.26	1.230	1.270
B	12.70	13.72	0.500	0.540
C	4.06	5.59	0.160	0.220
D	0.41	0.51	0.016	0.020
F	1.27	1.52	0.050	0.060
G	2.54 BSC		0.100 BSC	
J	0.20	0.30	0.008	0.012
K	2.29	4.06	0.090	0.160
L	15.24 BSC		0.600 BSC	
M	0°	15°	0°	15°
N	0.51	1.27	0.020	0.050

CASE 649



NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	31.50	32.13	1.240	1.265
B	13.21	13.72	0.520	0.540
C	4.70	5.21	0.185	0.205
D	0.38	0.51	0.015	0.020
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	2.92	3.43	0.115	0.135
L	14.99	15.49	0.590	0.610
M	–	10°	–	10°
N	0.51	1.02	0.020	0.040
P	0.13	0.38	0.005	0.015
Q	0.51	0.76	0.020	0.030

MOTOROLA

MC2916A

QUAD THREE-STATE BUS TRANSCEIVER WITH INTERFACE LOGIC

The MC2916A is a high-perfomance, low-power Schottky bus transceiver intended for bipolar or MOS microprocessor system applications. The device consists of four D-type edge-triggered flip-flops with a built-in two-input multiplexer on each. The flip-flop outputs are connected to four three-state bus drivers. Each bus driver is internally connected to the input of a receiver. The four receiver outputs drive four D-type latches. The device also contains a four-bit odd parity checker/generator.

The LSI bus transceiver is fabricated using advanced low-power Schottky processing. All inputs (except the Bus inputs) are one LS unit load. The three-state bus output can sink up to 48 mA at 0.5 V maximum. The bus enable input ($\overline{BE}$) is used to force the driver outputs to the high-impedance state. When $\overline{BE}$ is High, the driver is disabled.

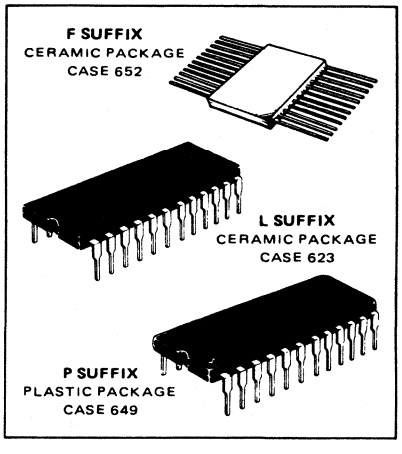
The input reigster consists of four D-type flip-flops with a buffered common clock and a two-input multiplexer at the input of each flip-flop. A common select input (S) controls the four multiplexers. When S is Low, the A_n data is stored in the register and when S is High, the B_n data is stored. The buffered common clock (DRCP) enters the data into this driver register on the Low-to-High transition.

Data from the A or B input is inverted at the Bus output. Likewise, data at the Bus input is inverted at the receiver output. Thus, data is non-inverted from driver input to receiver output. The four receivers each feature a built-in D-type latch that is controlled from the buffered receiver latch enable ($\overline{RLE}$) input. When the $\overline{RLE}$ input is Low, the latch is open and the receiver outputs will follow the bus inputs (Bus data inverted). When the $\overline{RLE}$ input is High, the latch will close and retain the present data regardless of the bus input.

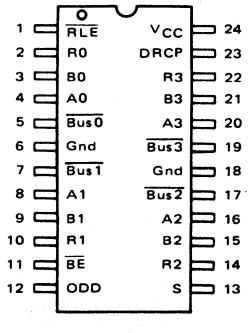
The MC2916A features a built-in four-bit odd parity checker/generator. The bus enable input ($\overline{BE}$) controls whether the parity output is in the generate or check mode. When the bus enable is Low (driver enabled), odd parity is generated based on the A or B field data input to the driver register. When $\overline{BE}$ is High, the parity output is determined by the four latch outputs of the receiver. Thus, if the driver is enabled, parity is generated; and, if the driver is in the high-impedance state, the Bus parity is checked.

- Quad High-speed LSI Bus Transceiver
- Three-state Bus Driver
- Two-port Input to D-type Register on Driver
- Bus Driver Output Can Sink 48 mA at 0.5 V Max
- Internal Odd 4-bit Parity Checker/Generator
- Receiver Has Output Latch for Pipeline Operation
- Receiver Outputs Sink 12 mA
- Advanced Low-power Schottky Processing
- 100% Reliability Assurance Testing in Compliance With MIL-STD-883
- 3.5 V Minimum Output High Voltage for Direct Interface to MOS Microprocessors

TTL
QUAD THREE-STATE
BUS TRANSCEIVER
WITH INTERFACE LOGIC



PIN ASSIGNMENT



ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Molded DIP	0°C to +70°C	MC2916APC
Hermetic DIP	0°C to +70°C	MC2916ALC
Hermetic DIP	-55°C to +125°C	MC2916ALM
Hermetic Flat Pack	-55°C to +125°C	MC2916AFM

MAXIMUM RATINGS (above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential	−0.5 V to +7.0 V
DC Voltage Applied to Outputs for High Output State	−0.5 V to +V_{CC}max
DC Input Voltage	−0.5 V to +5.5 V
DC Output Current, into Outputs (Except Bus)	30 mA
DC Output Current, into Bus	100 mA
DC Input Current	−30 mA to +5.0 mA

ELECTRICAL CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise noted)

MC2916AXC − T_A = 0°C to +70°C, V_{CC} = 5.0 V ±5% (Commercial), Min = 4.75 V, Max = 5.25 V

MC2916AXM − T_A = −55°C to +125°C, V_{CC} = 5.0 V ±10% (Military), Min = 4.5 V, Max = 5.5 V

BUS INPUT/OUTPUT CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

Parameter	Description	Test Conditions (Note 1)		Min	Typ	Max	Unit
V_{OL}	Bus Output Low Voltage	V_{CC} = Min	I_{OL} = 24 mA			0.4	Volts
			I_{OL} = 48 mA			0.5	
V_{OH}	Bus Output High Voltage	V_{CC} = Min	Commercial, I_{OH} = −20 mA	2.4			Volts
			Military, I_{OH} = −15 mA	2.4			
I_O	Bus Leakage Current (High Impedance)	V_{CC} = Max, Bus Enable = 2.4 V	V_O = 0.4 V			−200	μA
			V_O = 2.4 V			50	
			V_O = 4.5 V			100	
I_{off}	Bus Leakage Current (Power off)	V_O = 4.5 V, V_{CC} = 0 V				100	μA
V_{IH}	Receiver Input High Threshold	Bus Enable = 2.4 V		2.0			Volts
V_{IL}	Receiver Input Low Threshold	Bus Enable = 2.4 V	Commercial			0.8	Volts
			Military			0.7	
I_{SC}	Bus Output Short Circuit Current	V_{CC} = Max, V_O = 0 V		−50	−120	−225	mA

DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

Parameter	Description	Test Conditions (Note 1)		Min	Typ (Note 2)	Max	Unit
V_{OH}	Receiver Output High Voltage	V_{CC} = Min, V_{in} = V_{IL} or V_{IH}	Military, I_{OH} = −1.0 mA	2.4	3.4		Volts
			Commercial, I_{OH} = −2.6 mA	2.4	3.4		
		V_{CC} = 5.0 V, I_{OH} = −100 μA		3.5			
V_{OH}	Parity Output High Voltage	V_{CC} = Min, I_{OH} = −660 μA, V_{in} = V_{IH} or V_{IL}	Military	2.5	3.4		Volts
			Commercial	2.7	3.4		
V_{OL}	Output Low Voltage (Except Bus)	V_{CC} = Min, V_{in} = V_{IL} or V_{IH}	I_{OL} = 4.0 mA		0.27	0.4	Volts
			I_{OL} = 8.0 mA		0.32	0.45	
			I_{OL} = 12 mA		0.37	0.5	
V_{IH}	Input High Level (Except Bus)	Guaranteed input logical High for all inputs		2.0			Volts
V_{IL}	Input Low Level (Except Bus)	Guaranteed input logical Low for all inputs	Military			0.7	Volts
			Commercial			0.8	
V_I	Input Clamp Voltage (Except Bus)	V_{CC} = Min, I_{in} = −18 mA				−1.2	Volts
I_{IL}	Input Low Current (Except Bus)	V_{CC} = Max, V_{in} = 0.4 V	$\overline{BE}$, $\overline{RLE}$			−0.72	mA
			All other inputs			−0.36	
I_{IH}	Input High Current (Except Bus)	V_{CC} = Max, V_{in} = 2.7 V				20	μA
I_I	Input High Current (Except Bus)	V_{CC} = Max, V_{in} = 7.0 V				100	μA
I_{SC}	Output Short Circuit Current (Except Bus)	V_{CC} = Max	Receiver	−30		−130	mA
			Parity	−20		−100	
I_{CC}	Power Supply Current	V_{CC} = Max, All inputs = Gnd			75	110	mA

NOTES: 1. For conditions shown as Min or Max, use the appropriate value specified under Electrical Characteristics for the applicable device type.

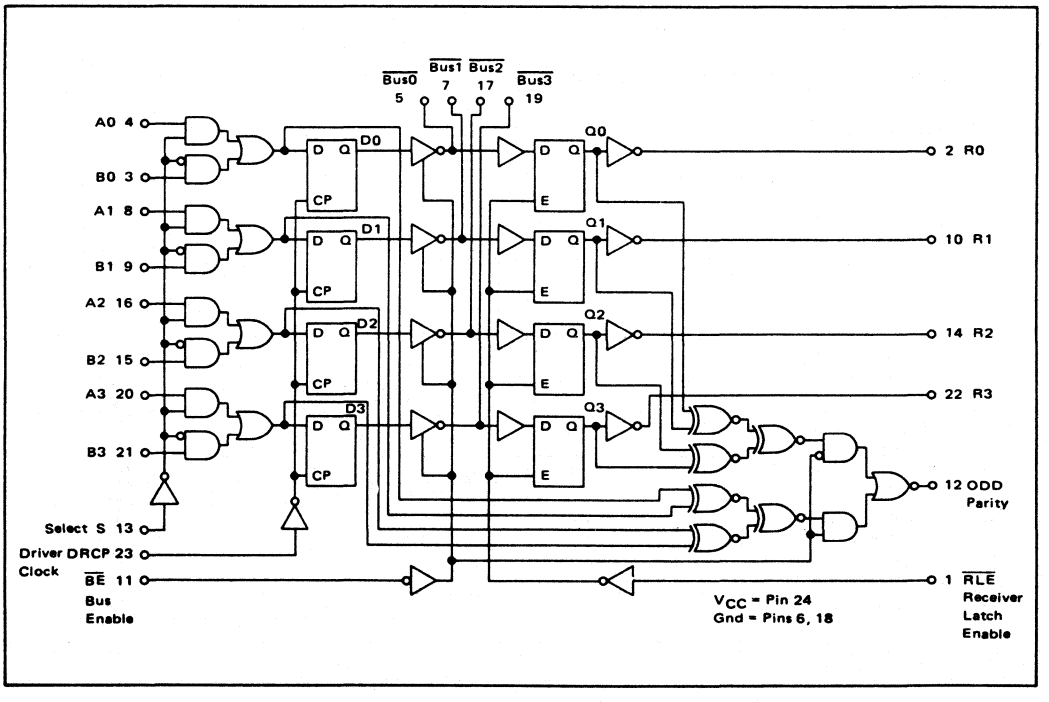
2. Typical limits are at V_{CC} = 5.0 V, 25°C ambient and maximum loading.

3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.

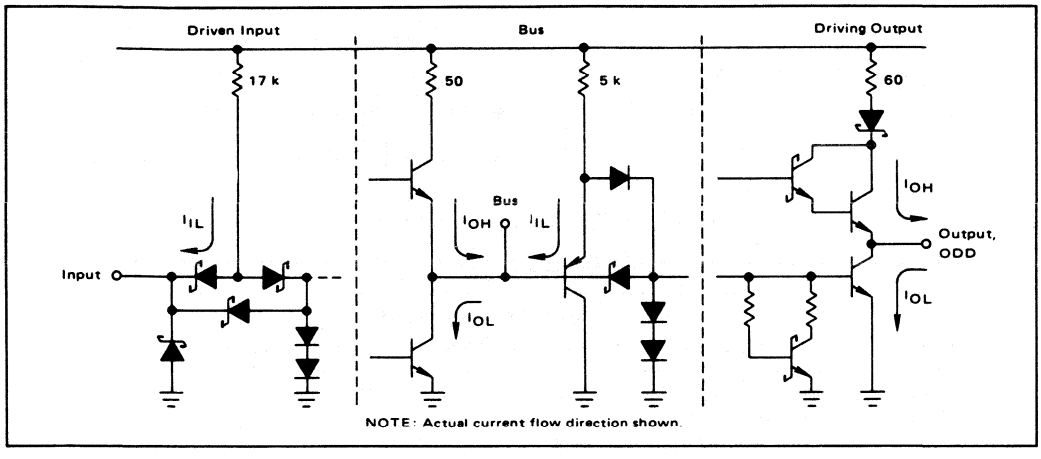
SWITCHING CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

Parameter	Description	Test Conditions	MC2916AXM			MC2916AXC			Unit
			Min	Typ (Note 2)	Max	Min	Typ (Note 2)	Max	
t_{PHL}	Driver Clock (DRCP) to Bus	C_L (Bus) = 50 pF	–	21	36	–	21	32	ns
t_{PLH}		R_L (Bus) = 130 Ω	–	21	36	–	21	32	
t_{ZH},t_{ZL}	Bus Enable ($\overline{BE}$) to Bus		–	13	26	–	13	23	ns
t_{HZ},t_{LZ}			–	13	21	–	13	18	
t_s	Data Inputs (A or B)		15	–	–	12	–	–	ns
t_h			8.0	–	–	6.0	–	–	
t_s	Select Input (S)		28	–	–	25	–	–	ns
t_h			8.0	–	–	6.0	–	–	
t_{PW}	Clock Pulse Width (High)		20	–	–	17	–	–	ns
t_{PLH}	Bus to Receiver Output (Latch Enabled)	C_L = 15 pF	–	18	33	–	18	30	ns
t_{PHL}		R_L = 2.0 k	–	18	30	–	18	27	
t_{PLH}	Latch Enable to Receiver Output		–	21	33	–	21	30	ns
t_{PHL}			–	21	30	–	21	27	
t_s	Bus to Latch Enable ($\overline{RLE}$)		15	–	–	13	–	–	ns
t_h			6.0	–	–	4.0	–	–	
t_{PLH}	A or B Data to Odd Parity (Driver Enabled)		–	32	46	–	32	42	ns
t_{PHL}			–	26	40	–	26	36	
t_{PLH}	Bus to Odd Parity Output (Driver Inhibited, Latch Enabled)		–	21	36	–	21	32	ns
t_{PHL}			–	21	36	–	21	32	
t_{PLH}	Latch Enable (RLE) to Odd Parity Output		–	21	36	–	21	32	ns
t_{PHL}			–	21	36	–	21	32	

LOGIC DIAGRAM



INPUT/OUTPUT CURRENT INTERFACE CONDITIONS

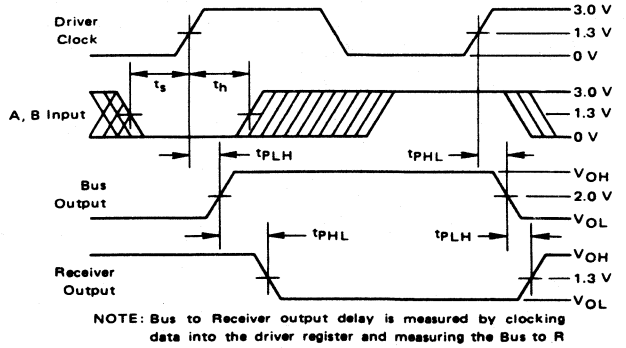


FUNCTION TABLE

Inputs							Internal to Device		Bus	Output	
S	A_n	B_n	DRCP	$\overline{BE}$	$\overline{RLE}$	$\overline{OE}$	D_n	Q_n	$\overline{Bus_n}$	R_n	Function
X	X	X	X	H	X	X	X	X	Z	X	Driver output disable
X	X	X	X	X	X	H	X	X	X	Z	Receiver output disable
X	X	X	X	H	L	L	X	L	L	H	Driver output disable
X	X	X	X	H	L	L	X	H	H	L	and receive data via Bus input
X	X	X	X	X	H	X	X	NC	X	X	Latch received data
L	L	X	↑	X	X	X	L	X	X	X	Load driver register
L	H	X	↑	X	X	X	H	X	X	X	
H	X	L	↑	X	X	X	L	X	X	X	
H	X	H	↑	X	X	X	H	X	X	X	
X	X	X	L	X	X	X	NC	X	X	X	No driver clock restrictions
X	X	X	H	X	X	X	NC	X	X	X	
X	X	X	X	L	X	X	L	X	H	X	Drive Bus
X	X	X	X	L	X	X	H	X	L	X	

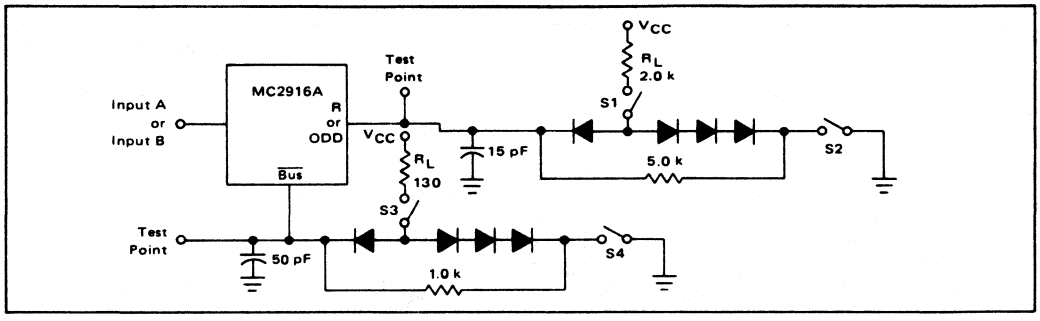
H = High Z = High impedance X = Don't care
L = Low NC = No change ↑ = Low-to-high transition

SWITCHING WAVEFORMS



NOTE: Bus to Receiver output delay is measured by clocking data into the driver register and measuring the Bus to R combinatorial delay.

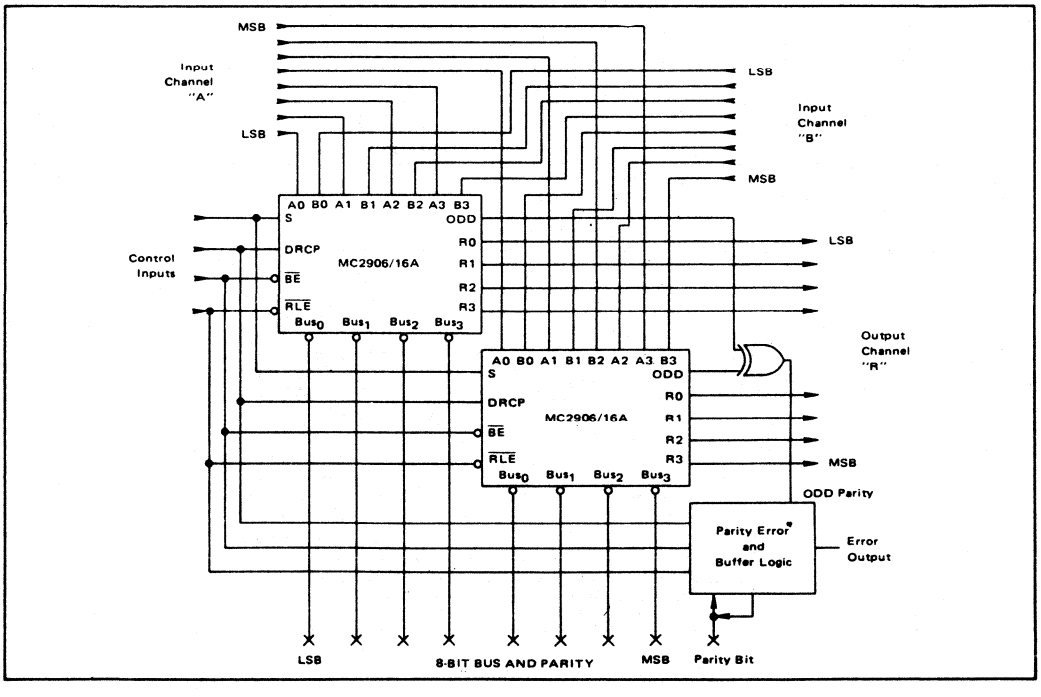
SWITCHING TEST CIRCUIT



DEFINITIONS OF FUNCTIONAL TERMS

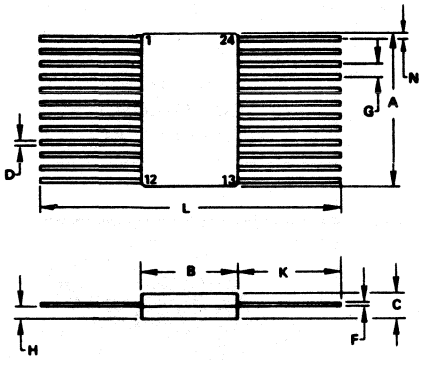
A0, A1, A2, A3	The "A" word data input into the two input multiplexer of the driver register.
B0, B1, B2, B3	The "B" word data input into the two input multiplexers of the driver register.
S	Select. When the select input is Low, the A data word is applied to the driver register. When the select input is High, the B word is applied to the driver register.
DRCP	Driver Clock Pulse. Clock pulse for the driver register.
$\overline{BE}$	$\overline{Bus\ Enable}$. When the Bus Enable is High, the four drivers are in the high impedance state.
$\overline{Bus0}$, $\overline{Bus1}$, $\overline{Bus2}$, $\overline{Bus3}$	The four driver outputs and receiver inputs (data is inverted).
R0, R1, R2, R3	The four receiver outputs. Data from the bus is inverted while data from the A or B inputs is non-inverted.
$\overline{RLE}$	$\overline{Receiver\ Latch\ Enable}$. When $\overline{RLE}$ is Low, data on the Bus inputs is passed through the receiver latches. When $\overline{RLE}$ is High, the receiver latches are closed and will retain the data independent of all other inputs.
$\overline{OE}$	$\overline{Output\ Enable}$. When the $\overline{OE}$ input is High, the four three-state receiver outputs are in the high impedance state.
ODD	Odd parity output generates parity with the driver enabled, checks parity with the driver in the high impedance state.

8-BIT PERIPHERAL INTERFACE



PACKAGE DIMENSIONS

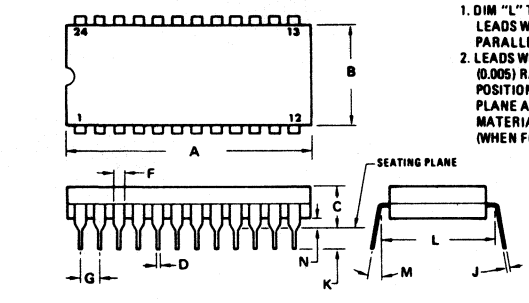
CASE 652



NOTES:
1. LEADS WITHIN 0.25 mm (0.010) TOTAL OF TRUE POSITION AT MAXIMUM MATERIAL CONDITION.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	14.99	15.49	0.590	0.610
B	9.27	9.91	0.365	0.390
C	1.27	2.03	0.050	0.080
D	0.38	0.48	0.015	0.019
F	0.08	0.15	0.003	0.006
G	1.27 BSC		0.050 BSC	
H	0.69	1.02	0.027	0.040
K	6.35	9.40	0.250	0.370
L	21.97	–	0.865	–
N	0.25	0.63	0.010	0.025

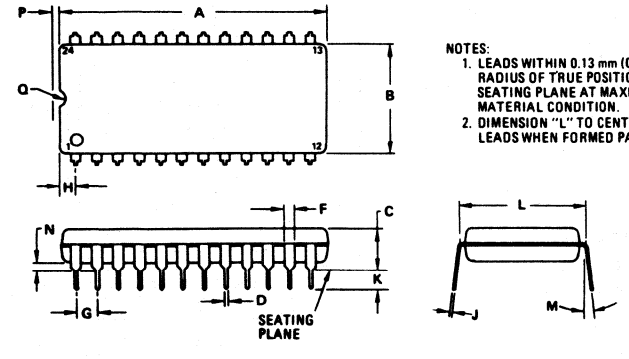
CASE 623



NOTES:
1. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.
2. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION. (WHEN FORMED PARALLEL)

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	31.24	32.26	1.230	1.270
B	12.70	13.72	0.500	0.540
C	4.06	5.59	0.160	0.220
D	0.41	0.51	0.016	0.020
F	1.27	1.52	0.050	0.060
G	2.54 BSC		0.100 BSC	
J	0.20	0.30	0.008	0.012
K	2.29	4.06	0.090	0.160
L	15.24 BSC		0.600 BSC	
M	0°	15°	0°	15°
N	0.51	1.27	0.020	0.050

CASE 649



NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	31.50	32.13	1.240	1.265
B	13.21	13.72	0.520	0.540
C	4.70	5.21	0.185	0.205
D	0.38	0.51	0.015	0.020
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.30	0.008	0.012
K	2.92	3.43	0.115	0.135
L	14.99	15.49	0.590	0.610
M	–	10°	–	10°
N	0.51	1.02	0.020	0.040
P	0.13	0.38	0.005	0.015
Q	0.51	0.76	0.020	0.030

MOTOROLA

MC2917A

QUAD THREE-STATE BUS TRANSCEIVER WITH INTERFACE LOGIC

The MC2917A is a high-performance, low-power Schottky bus transceiver intended for bipolar or MOS microprocessor system applications. The device consists of four D-type edge-triggered flip-flops. The flip-flop outputs are connected to four three-state bus drivers. Each bus driver is internally connected to the input of a receiver. The four receiver outputs drive four D-type latches that feature three-state outputs. The device also contains a four-bit odd parity checker/generator.

The LSI bus transceiver is fabricated using advanced low-power Schottky processing. All inputs (except the Bus inputs) are one LS unit load. The three-state bus output can sink up to 48 mA at 0.5 V maximum. The bus enable input ($\overline{BE}$) is used to force the driver outputs to the high-impedance state. When $\overline{BE}$ is High, the driver is disabled.

The input register consists of four D-type flip-flops with a buffered common clock. The buffered common clock (DRCP) enters the A_n data into this driver register on the Low-to-High transition.

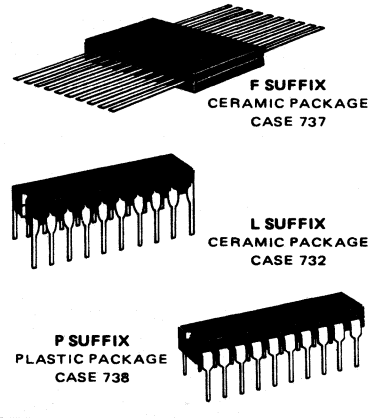
Data from the A input is inverted at the Bus output. Likewise, the data at the Bus input is inverted at the receiver output. Thus, data is non-inverted from driver input to receiver output. The four receivers each feature a built-in D-type latch that is controlled from the buffered receiver latch enable ($\overline{RLE}$) input. When the $\overline{RLE}$ input is Low, the latch is open and the receiver outputs will follow the bus inputs (Bus data inverted and $\overline{OE}$ Low). When the $\overline{RLE}$ input is High, the latch will close and retain the present data regardless of the bus input. The four latches have three-state outputs and are controlled by a buffered common three-state control ($\overline{OE}$) input. When $\overline{OE}$ is High, the receiver outputs are in the high-impedance state.

The MC2917A features a built-in four-bit odd parity checker/generator. The bus enable input ($\overline{BE}$) controls whether the parity output is in the generate or check mode. When the bus enable is Low (driver enabled), odd parity is generated based on the A field data input to the driver register. When $\overline{BE}$ is High, the parity output is determined by the four latch outputs of the receiver. Thus, if the driver is enabled, parity is generated; and, if the driver is in the high-impedance state, the Bus parity is checked.

FEATURES

- Quad High-speed LSI Bus Transceiver
- Three-state Bus Driver
- D-type Register on Driver
- Bus Driver Output Can Sink 48 MA at 0.5 V Max
- Internal Odd 4-bit Parity Checker/Generator
- Receiver Has Output Latch for Pipeline Operation
- Three-state Receiver Outputs Sink 12 mA
- Advanced Low-power Schottky Processing
- 100% Reliability Assurance Testing in Compliance With MIL-STD-883
- 3.5 V Minimum Output High Voltage for Direct Interface to MOS Microprocessors

TTL
QUAD THREE-STATE BUS TRANSCEIVER WITH INTERFACE LOGIC

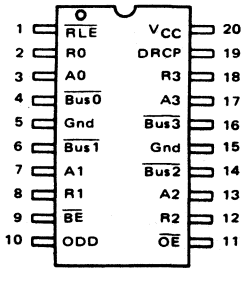


F SUFFIX
CERAMIC PACKAGE
CASE 737

L SUFFIX
CERAMIC PACKAGE
CASE 732

P SUFFIX
PLASTIC PACKAGE
CASE 738

PIN ASSIGNMENT



ORDERING INFORMATION		
Package Type	**Temperature Range**	**Order Number**
Molded DIP	0°C to +70°C	MC2917APC
Hermetic DIP	0°C to +70°C	MC2917ALC
Hermetic DIP	-55°C to +125°C	MC2917ALM
Hermetic Flat Pack	-55°C to +125°C	MC2917AFM

MAXIMUM RATINGS (above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential	−0.5 V to +7.0 V
DC Voltage Applied to Outputs for High Output State	−0.5 V to +V_{CC}max
DC Input Voltage	−0.5 V to +5.5 V
DC Output Current, into Outputs (Except Bus)	30 mA
DC Output Current, into Bus	100 mA
DC Input Current	−30 mA to +5.0 mA

ELECTRICAL CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise noted)

MC2917AXC — T_A = 0°C to +70°C, V_{CC} = 5.0 V ±5% (Commercial), Min = 4.75 V, Max = 5.25 V
MC2917AXM — T_A = −55°C to +125°C, V_{CC} = 5.0 V ±10% (Military), Min = 4.5 V, Max = 5.5 V

BUS INPUT/OUTPUT CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

Parameter	Description	Test Conditions (Note 1)			Min	Typ	Max	Unit
V_{OL}	Bus Output Low Voltage	V_{CC} = Min	I_{OL} = 24 mA				0.4	Volts
			I_{OL} = 48 mA				0.5	
V_{OH}	Bus Output High Voltage	V_{CC} = Min	Commercial, I_{OH} = −20 mA		2.4			Volts
			Military, I_{OH} = −15 mA		2.4			
I_O	Bus Leakage Current (High Impedance)	V_{CC} = Max Bus Enable = 2.4 V	V_O = 0.4 V				−200	μA
			V_O = 2.4 V				50	
			V_O = 4.5 V				100	
I_{off}	Bus Leakage Current (Power off)	V_O = 4.5 V V_{CC} = 0 V					100	μA
V_{IH}	Receiver Input High Threshold	Bus Enable = 2.4 V			2.0			Volts
V_{IL}	Receiver Input Low Threshold	Bus Enable = 2.4 V		Commercial			0.8	Volts
				Military			0.7	
I_{SC}	Bus Output Short Circuit Current	V_{CC} = Max V_O = 0 V			−50	−120	−225	mA

DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

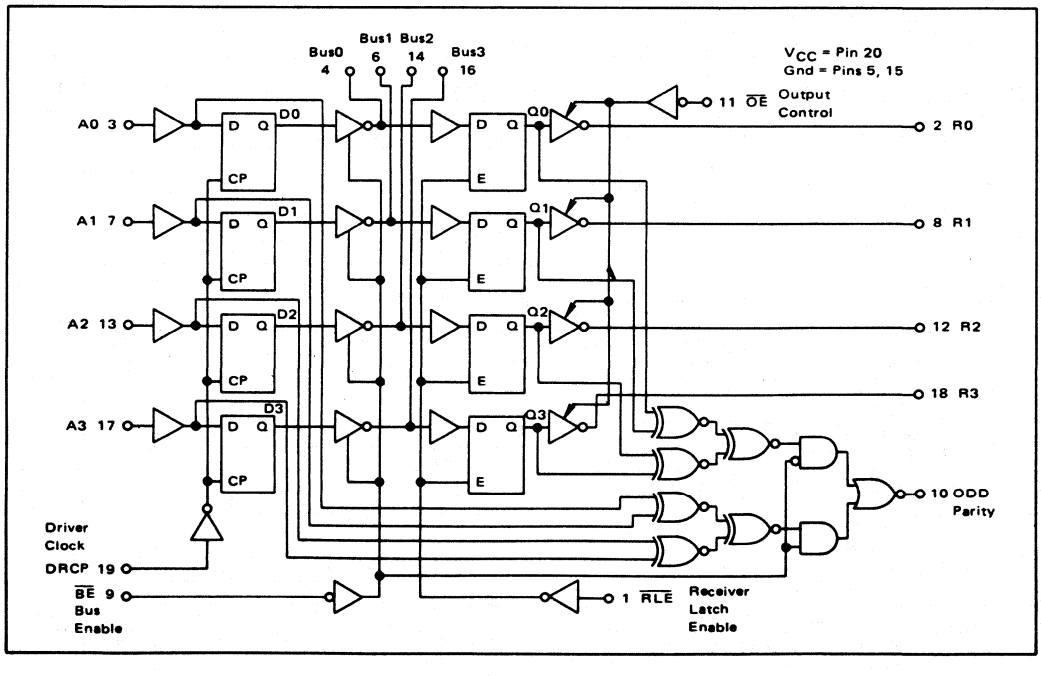
Parameter	Description	Test Conditions (Note 1)			Min	Typ (Note 2)	Max	Unit
V_{OH}	Receiver Output High Voltage	V_{CC} = Min V_{in} = V_{IL} or V_{IH}	Military, I_{OH} = −1.0 mA		2.4	3.4		Volts
			Commercial, I_{OH} = −2.6 mA		2.4	3.4		
		V_{CC} = 5.0 V, I_{OH} = −100 μA			3.5			
V_{OH}	Parity Output High Voltage	V_{CC} = Min, I_{OH} = −660 μA V_{in} = V_{IH} or V_{IL}		Military	2.5	3.4		Volts
				Commercial	2.7	3.4		
V_{OL}	Output Low Voltage (Except Bus)	V_{CC} = Min V_{in} = V_{IL} or V_{IH}	I_{OL} = 4.0 mA			0.27	0.4	Volts
			I_{OL} = 8.0 mA			0.32	0.45	
			I_{OL} = 12 mA			0.37	0.5	
V_{IH}	Input High Level (Except Bus)	Guaranteed input logical High for all inputs			2.0			Volts
V_{IL}	Input Low Level (Except Bus)	Guaranteed input logical Low for all inputs		Military			0.7	Volts
				Commercial			0.8	
V_I	Input Clamp Voltage (Except Bus)	V_{CC} = Min, I_{in} = −18 mA					−1.2	Volts
I_{IL}	Input Low Current (Except Bus)	V_{CC} = Max, V_{in} = 0.4 V	$\overline{BE}$, $\overline{RLE}$				−0.72	mA
			All other inputs				−0.36	
I_{IH}	Input High Current (Except Bus)	V_{CC} = Max, V_{in} = 2.7 V					20	μA
I_I	Input High Current (Except Bus)	V_{CC} = Max, V_{in} = 7.0 V					100	μA
I_{SC}	Output Short Circuit Current (Except Bus)	V_{CC} = Max		Receiver	−30		−130	mA
				Parity	−20		−100	
I_{CC}	Power Supply Current	V_{CC} = Max				63	95	mA
I_O	Off-State Output Current (Receiver Outputs)	V_{CC} = Max	V_O = 2.4 V				50	μA
			V_O = 0.4 V				−50	

NOTES: 1. For conditions shown as Min or Max, use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0 V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.

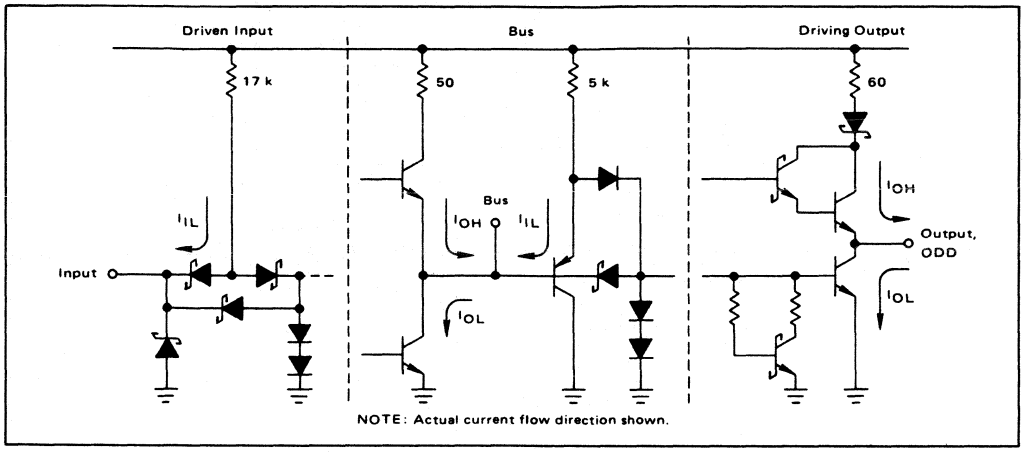
SWITCHING CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

Parameter	Description	Test Conditions	MC2917AXM			MC2917AXC			Unit
			Min	Typ (Note 2)	Max	Min	Typ (Note 2)	Max	
t_{PHL}	Driver Clock (DRCP) to Bus	C_L (Bus) = 50 pF	–	21	36	–	21	32	ns
t_{PLH}		R_L (Bus) = 130 Ω	–	21	36	–	21	32	
t_{ZH},t_{ZL}	Bus Enable ($\overline{BE}$) to Bus		–	13	26	–	13	23	ns
t_{HZ},t_{LZ}			–	13	21	–	13	18	
t_s	A Data Input	C_L = 15 pF	15	–	–	12	–	–	ns
t_h		R_L = 2.0 k	8.0	–	–	6.0	–	–	
t_{PW}	Clock Pulse Width (High)		20	–	–	17	–	–	ns
t_{PLH}	Bus to Receiver Output (Latch Enabled)		–	18	33	–	18	30	ns
t_{PHL}			–	18	30	–	18	27	
t_{PLH}	Latch Enable to Receiver Output		–	21	33	–	21	30	ns
t_{PHL}			–	21	30	–	21	27	
t_s	Bus to Latch Enable ($\overline{RLE}$)		15	–	–	13	–	–	ns
t_h			6.0	–	–	4.0	–	–	
t_{PLH}	A Data to Odd Parity Out (Driver Enabled)		–	32	46	–	32	42	ns
t_{PHL}			–	26	40	–	26	36	
t_{PLH}	Bus to Odd Parity Out (Driver Inhibit)		–	21	36	–	21	32	ns
t_{PHL}			–	21	36	–	21	32	
t_{PLH}	Latch Enable ($\overline{RLE}$) to Odd Parity Output		–	21	36	–	21	32	ns
t_{PHL}			–	21	36	–	21	32	
t_{ZH},t_{ZL}	Output Control to Output		–	14	26	–	14	23	ns
t_{HZ},t_{LZ}		C_L = 5 pF, R_L = 2.0 k	–	14	26	–	14	23	

LOGIC DIAGRAM



INPUT/OUTPUT CURRENT INTERFACE CONDITIONS



NOTE: Actual current flow direction shown.

FUNCTION TABLE

		Inputs			Internal to Device		Bus	Output	
A_n	DRCP	$\overline{BE}$	$\overline{RLE}$	$\overline{OE}$	D_n	Q_n	Bus_n	R_n	Function
X	X	H	X	X	X	X	H	X	Driver output disable
X	X	X	X	H	X	X	X	Z	Receiver output disable
X	X	H	L	L	X	L	L	H	Driver output disable and receive data via Bus input
X	X	H	L	L	X	H	H	L	
X	X	X	H	X	X	NC	X	X	Latch received data
L	↑	X	X	X	L	X	X	X	Load driver register
H	↑	X	X	X	H	X	X	X	
X	L	X	X	X	NC	X	X	X	No driver clock restrictions
X	H	X	X	X	NC	X	X	X	
X	X	L	X	X	L	X	H	X	Drive Bus
X	X	L	X	X	H	X	L	X	

H = High Z = High impedance X = Don't care
L = Low NC = No change ↑ = Low-to-high transition

PARITY OUTPUT FUNCTION TABLE

$\overline{BE}$	Odd Parity Output
L	ODD = A0 ⊕ A1 ⊕ A2 ⊕ A3
H	ODD = Q0 ⊕ Q1 ⊕ Q2 ⊕ Q3

DEFINITIONS OF FUNCTIONAL TERMS

DRCP Driver Clock Pulse. Clock pulse for the driver register.

$\overline{BE}$ $\overline{\text{Bus Enable}}$. When the Bus Enable is Low, the four drivers are in the high impedance state.

Bus0, Bus1, Bus2, Bus3 The four driver outputs and receiver inputs (data is inverted).

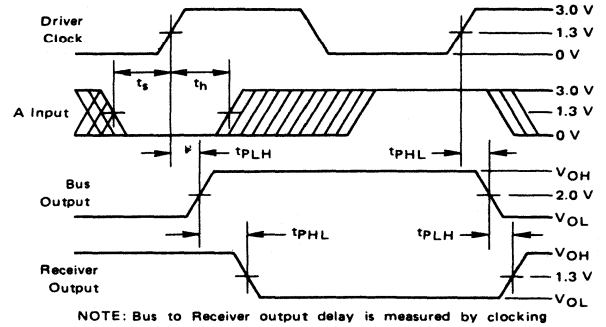
R0, R1, R2, R3 The four receiver outputs. Data from the bus is inverted while data from the A or B inputs is non-inverted.

$\overline{RLE}$ $\overline{\text{Receiver Latch Enable}}$. When $\overline{RLE}$ is Low, data on the Bus inputs is passed through the receiver latches. When $\overline{RLE}$ is High, the receiver latches are closed and will retain the data independent of all other inputs.

ODD Odd parity output generates parity with the driver enabled. Checks parity with the driver in the high impedance state.

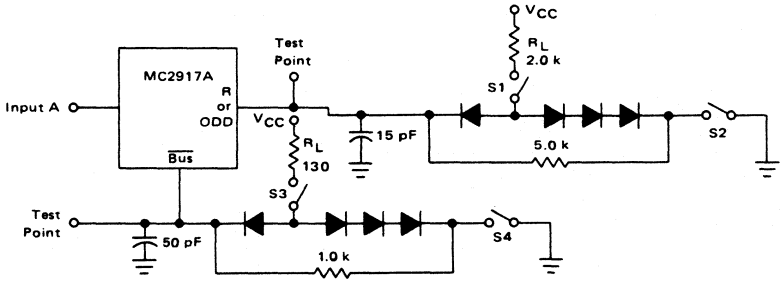
$\overline{OE}$ $\overline{\text{Output Enable}}$. When the $\overline{OE}$ input is High, the four three-state receiver outputs are in the high impedance state.

SWITCHING WAVEFORMS

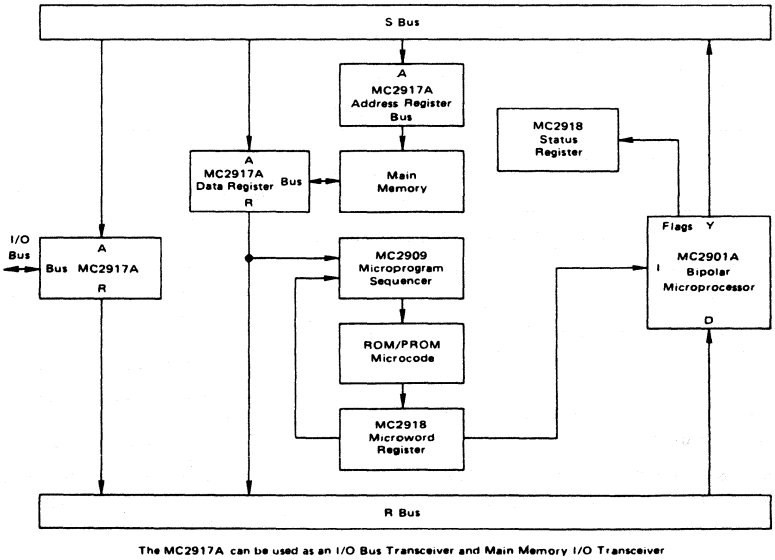


NOTE: Bus to Receiver output delay is measured by clocking data into the driver register and measuring the Bus to R combinatorial delay.

SWITCHING TEST CIRCUIT



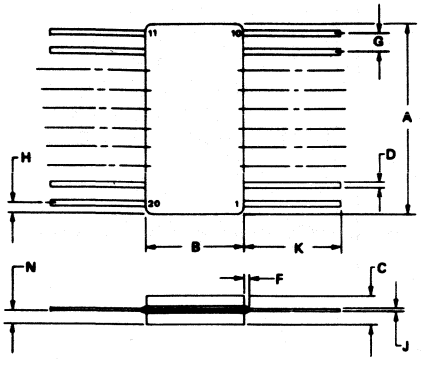
APPLICATIONS



The MC2917A can be used as an I/O Bus Transceiver and Main Memory I/O Transceiver in high speed microprocessor systems.

PACKAGE DIMENSIONS

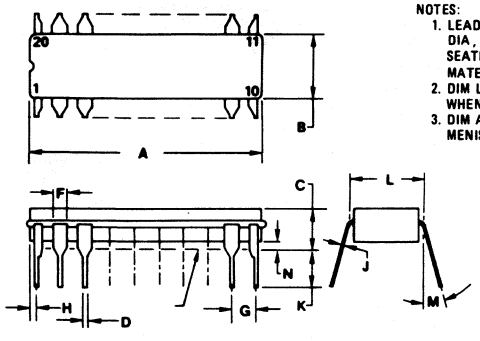
CASE 737



NOTE:
1. LEADS WITHIN 0.25 mm (0.010) TOTAL OF TRUE POSITION AT MAXIMUM MATERIAL CONDITION.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	–	13.08	–	0.515
B	5.84	6.60	0.230	0.260
C	1.52	2.16	0.060	0.085
D	0.41	0.46	0.016	0.018
F	–	0.25	–	0.010
G	1.27 BSC		0.050 BSC	
H	1.14	1.40	0.045	0.055
J	0.08	0.13	0.003	0.005
K	–	9.14	–	0.360
N	–	1.02	–	0.040

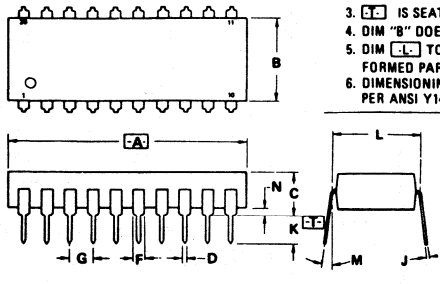
CASE 732



NOTES:
1. LEADS WITHIN 0.25 mm (0.010) DIA., TRUE POSITION AT SEATING PLANE, AT MAXIMUM MATERIAL CONDITION.
2. DIM L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIM A AND B INCLUDES MENISCUS.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	24.38	25.15	0.960	0.990
B	6.86	7.49	0.270	0.295
C	4.32	5.08	0.170	0.200
D	0.38	0.56	0.015	0.022
F	1.40	1.65	0.055	0.065
G	2.54 BSC		0.100 BSC	
H	0.89	1.40	0.035	0.055
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	7.62 BSC		0.300 BSC	
M	5°	15°	5°	15°
N	0.51	0.76	0.020	0.030

CASE 738



NOTES:
1. DIM -A- IS DATUM.
2. POSITIONAL TOL FOR LEADS:
   ⌖ Ø 0.25 (0.010) Ⓜ T A Ⓜ
3. -T- IS SEATING PLANE.
4. DIM "B" DOES NOT INCLUDE MOLD FLASH.
5. DIM -L- TO CENTER OF LEADS WHEN FORMED PARALLEL.
6. DIMENSIONING AND TOLERANCING PER ANSI Y14.5, 1973.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	25.65	27.18	1.010	1.070
B	6.10	6.60	0.240	0.260
C	3.94	4.19	0.155	0.165
D	0.38	0.56	0.015	0.022
F	1.27	1.78	0.050	0.070
G	2.54 BSC		0.100 BSC	
J	0.20	0.38	0.008	0.015
K	2.79	3.56	0.110	0.140
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	1.02	0.020	0.040

MOTOROLA

MC2918

QUAD D REGISTER WITH STANDARD AND THREE-STATE OUTPUTS

New Schottky circuits such as the MC2918 register provide the design engineer with additional flexibility in system configuration—especially with regard to bus structure, organization, and speed. The MC2918 is a quadruple D-type register with four standard totem pole outputs and four three-state bus-type outputs. The 16-pin device also features a buffered common clock (CP) and a buffered common output control ($\overline{OE}$) for the Y outputs. Information meeting the setup and hold requirements on the D inputs is transferred to the Q outputs on the Low-to-High transition of the clock.

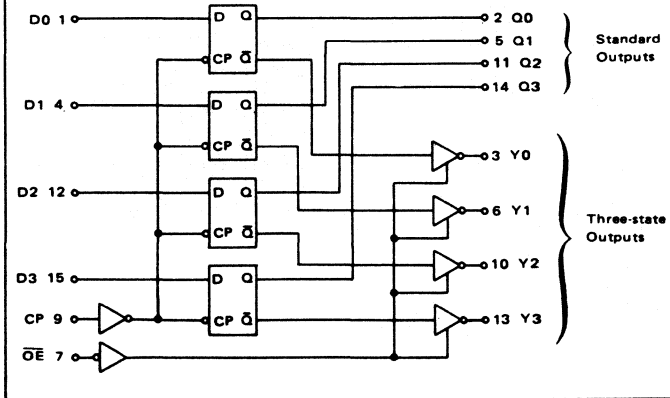
The same data as on the Q outputs is enabled at the three-state Y outputs when the "output control" ($\overline{OE}$) input is Low. When the $\overline{OE}$ input is High, the Y outputs are in the high-impedance state.

The MC2918 register can be used in bipolar microprocessor designs as an address register, status register, instruction register or for various data or microword register applications. Because of the unique design of the three-state output, the device features very short propagation delay from the clock to the Q or Y outputs. Thus, system performance and architectural design can be improved by using the MC2918 register. Other applications of MC2918 register can be found in microprogrammed display systems, communication systems and most general or special purpose digital signal processing equipment.

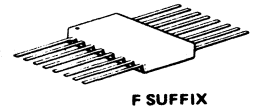
FEATURES

- Advanced Schottky Technology
- Four D-type Flip-flops
- Four Standard Totem Pole Outputs
- Four Three-state Outputs
- 75 MHz Clock Frequency
- 100% Reliability Assurance Testing in Compliance With MIL-STD-883

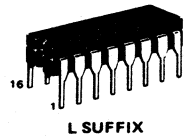
LOGIC DIAGRAM



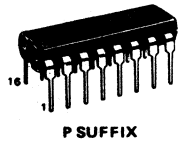
TTL
QUAD D REGISTER WITH STANDARD AND THREE-STATE OUTPUTS



F SUFFIX
CERAMIC PACKAGE
CASE 650

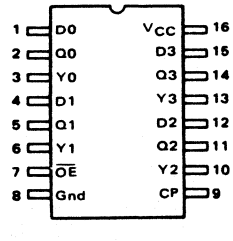


L SUFFIX
CERAMIC PACKAGE
CASE 620



P SUFFIX
PLASTIC PACKAGE
CASE 648

PIN ASSIGNMENT



ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Molded DIP	0°C to +70°C	MC2918PC
Hermetic DIP	0°C to +70°C	MC2918LC
Hermetic DIP	−55°C to +125°C	MC2918LM
Hermetic Flat Pack	−55°C to +125°C	MC2918FM

MAXIMUM RATINGS (above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential	−0.5 V to +7.0 V
DC Voltage Applied to Outputs for High Output State	−0.5 V to +V_{CC} Max
DC Input Voltage	−0.5 V to +5.5 V
DC Output Current, into Outputs	30 mA
DC Input Current	−30 mA to +5.0 mA

ELECTRICAL CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise noted)
MC2918XC − T_A = 0°C to +70°C, V_{CC} = 5.0 V ±5% (Commercial), Min = 4.75 V, Max = 5.25 V
MC2918XM − T_A = −55°C to +125°C, V_{CC} = 5.0 V ±10% (Military), Min = 4.5 V, Max = 5.5 V

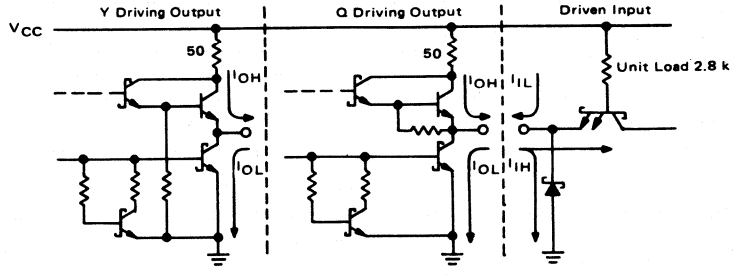
Parameters	Description	Test Conditions (Note 1)				Min	Typ (Note 2)	Max	Units
V_{OH}	Output High Voltage	V_{CC} = Min, V_{IN} = V_{IH} or V_{IL}	Q	I_{OH} = −1 mA	Military	2.5	3.4	–	Volts
					Commercial	2.7	3.4	–	
			Y	Military, I_{OH} = −2 mA		2.4	3.4	–	
				Commercial, I_{OH} = −6.5 mA		2.4	3.4	–	
V_{OL}	Output Low Voltage (Note 6)	V_{CC} = Min, I_{OL} = 20 mA, V_{IN} = V_{IH} or V_{IL}				–	–	0.5	Volts
V_{IH}	Input High Level	Guaranteed input logical High voltage for all inputs				2.0	–	–	Volts
V_{IL}	Input Low Level	Guaranteed input logical Low voltage for all inputs				–	–	0.8	Volts
V_I	Input Clamp Voltage	V_{CC} = Min, I_{in} = −18 mA				–	–	−1.2	Volts
I_{IL} (Note 3)	Input Low Current	V_{CC} = Max, V_{in} = 0.5 V				–	–	−2.0	mA
I_{IH} (Note 3)	Input High Current	V_{CC} = Max, V_{in} = 2.7 V				–	–	50	μA
I_I	Input High Current	V_{CC} = Max, V_{in} = 5.5 V				–	–	1.0	mA
I_O	Y Output Off-State Leakage Current	V_{CC} = Max		V_O = 2.4 V		–	–	50	μA
				V_O = 0.4 V		–	–	−50	
I_{SC}	Output Short Circuit Current (Note 4)	V_{CC} = Max				−40	–	−100	mA
I_{CC}	PowerSupply Current	V_{CC} = Max (Note 5)				–	80	130	mA

NOTES: 1. For conditions shown as Min or Max, use the appropriate value specified under Electrical Characterisitcs for the applicable device type.
2. Typical limits are at V_{CC} = 5.0 V, T_A = 25°C ambient and maximum loading.
3. Actual input currents = Unit Load Current X Input Load Factor (see Loading Rules).
4. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
5. I_{CC} is measured with all inputs at 4.5 V and all outputs open.
6. Measured on Y outputs with Q outputs open. Measured on Q outputs with Y outputs open.

SWITCHING CHARACTERISTICS (T_A = +25°C, V_{CC} = 5.0 V, R_L = 280 Ω)

Parameters	Description		Test Conditions	Min	Typ	Max	Units
t_{PLH}	Clock to Q Output		C_L = 15 pF	–	6.0	9.0	ns
t_{PHL}				–	8.5	13	
t_{PW}	Clock Pulse Width	High		7.0	–	–	ns
		Low		9.0			
t_s	Data			5.0	–	–	ns
t_h	Data			3.0	–	–	ns
t_{PLH}	Clock to Y Output ($\overline{OE}$ Low)			–	6.0	9.0	ns
t_{PHL}				–	8.5	13	
t_{ZH}	Output Control to Output		C_L = 15 pF	–	12.5	19	ns
t_{ZL}				–	12	18	
t_{HZ}			C_L = 5.0 pF	–	4.0	6.0	
t_{LZ}				–	7.0	10.5	
f_{max}	Maximum Clock Frequency		C_L = 15 pF	75	100	–	MHz

SCHOTTKY INPUT/OUTPUT CURRENT INTERFACE CONDITIONS



NOTE: Actual current flow direction shown.

TRUTH TABLE

Inputs			Outputs		
$\overline{OE}$	Clock CP	D	Q	Y	Notes
H	L	X	NC	Z	–
H	H	X	NC	Z	–
H	↑	L	L	Z	–
H	↑	H	H	Z	–
L	↑	L	L	L	–
L	↑	H	H	H	–
L	–	–	L	L	1
L	–	–	H	H	1

L = Low
H = High
X = Don't care

NC = No change
↑ = Low to high transition
Z = High impedance

NOTE 1. When $\overline{OE}$ is Low, the Y output will be in the same logic state as the Q output.

DEFINITIONS OF FUNCTIONAL TERMS

D_n — The four data inputs to the register.

Q_n — The four data outputs of the register with standard totem pole active pull-up outputs. Data is passed non-inverted.

Y_n — The four three-state data outputs of the register. When the three-state outputs are enabled, data is passed non-inverted. A High on the "output control" input forces the Y_n outputs to the high-impedance state.

CP Clock — The buffered common clock for the register. Enters data on the Low-to-High Transition.

$\overline{OE}$ — Output Control. When the $\overline{OE}$ input is High, the Y_n outputs are in th high-impedance state. When the $\overline{OE}$ input is Low, the TRUE register data is present at the Y_n outputs.

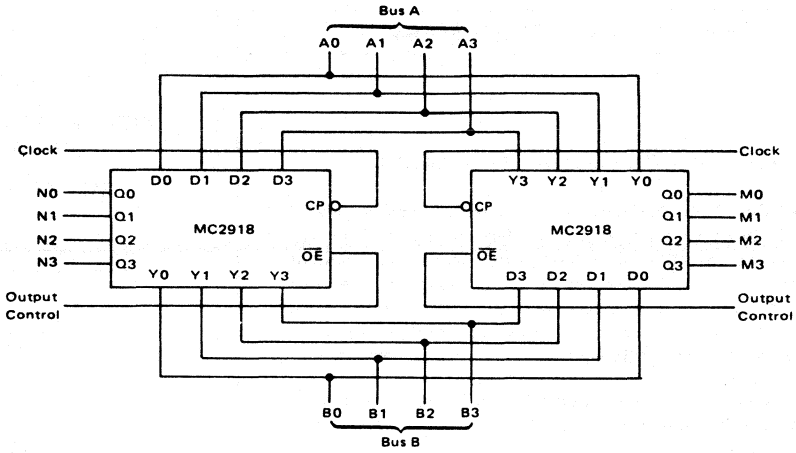
LOADING RULES
(In Unit Loads)

Input/ Output	Pin Numbers	Input Unit Load	Fan-out Output High	Fan-out Output Low
D0	1	1	–	–
Q0	2	–	20	10*
Y0	3	–	40/130	10*
D1	4	1	–	–
Q1	5	–	20	10*
Y1	6	–	40/130	10*
$\overline{OE}$	7	1	–	–
Gnd	8	–	–	–
CP	9	1	–	–
Y2	10	–	40/130	10*
Q2	11	–	20	10*
D2	12	1	–	–
Y3	13	–	40/130	10*
Q3	14	–	20	10*
D3	15	1	–	–
V_{CC}	16	–	–	–

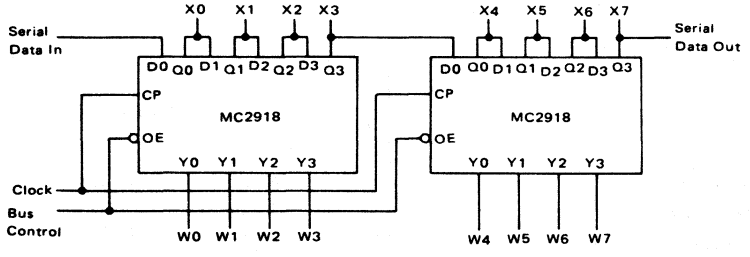
A Schottky TTL unit load is defined as 50 μA measured at 2.7 V High and −2.0 mA measured at 0.5 V Low.

*Fan-out on each Q_n and Y_n output pair should not exceed 15 unit loads (30 mA).

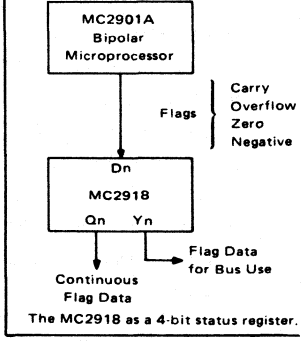
APPLICATIONS



The MC2918 can be connected for bidirectional interface between two buses. The device on the left stores data from the A bus and drives the A bus. The device on the right stores data from the B bus and drives the A bus. The output control is used to place either or both drivers in the high-impedance state. The contents of each register are available for continuous usage at the N and M ports of the device.

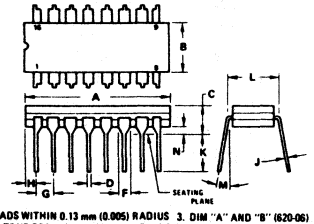


8-bit serial-to-parallel converter with three-state output (W) and direct access to the register word (X).



The MC2918 as a 4-bit status register.

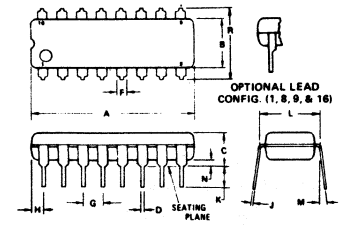
PACKAGE DIMENSIONS



1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. PACKAGE INDEX: NOTCH IN LEAD NOTCH IN CERAMIC OR INK DOT.
3. DIM "A" AND "B" (620-06) DO NOT INCLUDE GLASS RUN-OUT.
4. DIM "L" TO INSIDE OF LEADS (MEASURED 0.51 mm (0.020) BELOW BODY).

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	19.05	19.94	0.750	0.785
B	6.10	7.49	0.240	0.295
C	–	5.08	–	0.200
D	0.38	0.53	0.015	0.021
F	1.40	1.78	0.055	0.070
G	2.54 BSC		0.100 BSC	
H	0.51	1.14	0.020	0.045
J	0.20	0.30	0.008	0.012
K	3.18	5.08	0.125	0.200
L	7.49	8.89	0.295	0.350
M	–	15°	–	15°
N	0.51	1.02	0.020	0.040

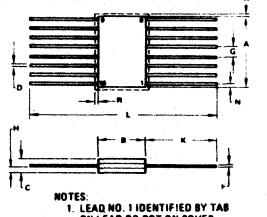
CASE 620



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	–	22.10	–	0.870
B	6.10	6.60	0.240	0.260
C	–	5.08	–	0.200
D	0.38	0.53	0.015	0.021
F	–	1.78	–	0.070
G	2.54 BSC		0.100 BSC	
H	0.38	2.41	0.015	0.095
J	0.20	0.38	0.008	0.015
K	2.92	–	0.115	–
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	–	0.020	–
R	–	8.26	–	0.325

NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSION "B" DOES NOT INCLUDE MOLD FLASH.
4. "F" DIMENSION IS FOR FULL LEADS. "HALF" LEADS ARE OPTIONAL AT LEAD POSITIONS 1, 8, 9, and 16).
5. DIMENSION "R" TO BE MEASURED AT THE TOP OF THE LEADS (NOT AT THE TIPS).

CASE 648



NOTES:
1. LEAD NO. 1 IDENTIFIED BY TAB ON LEAD OR DOT ON COVER.
2. LEADS WITHIN 0.13 mm (0.005) TOTAL OF TRUE POSITION AT MAXIMUM MATERIAL CONDITION.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	9.40	10.16	0.370	0.400
B	6.22	7.24	0.245	0.285
C	1.52	2.03	0.060	0.080
D	0.41	0.48	0.016	0.019
F	0.08	0.15	0.003	0.006
G	1.27 BSC		0.050 BSC	
H	0.64	0.89	0.025	0.035
K	9.65	9.40	0.250	0.370
L	18.92	–	0.745	–
M	–	0.51	–	0.020
R	–	0.38	–	0.015

CASE 650

MOTOROLA

MC29100/MC82100
MC29101/MC82101
(DUAL MARKED)

Product Preview

TTL
FIELD PROGRAMMABLE LOGIC ARRAY
(16 X 8 X 48 FPLA)

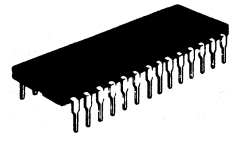
FIELD PROGRAMMABLE LOGIC ARRAY

The MC29100/MC82100 (three-state outputs) and the MC29101/MC82101 (open collector outputs) are bipolar programmable logic arrays, containing 48 product terms (AND terms), and 8 output functions. Each output function can be programmed either true active-High (Fp), or true active-Low ($F\bar{p}$). The true state of the output functions is controlled via an output sum (OR) matrix by a logical combination of 16-input variables, or their complements, up to 48 terms.

Both devices are field-programmable, which means that custom patterns are immediately available by following an appropriate fusing procedure.

The MC29100 and MC29101 are fully TTL compatible, and include a chip-enable clocking input for output de-skewing and inhibit. They feature either open collector or three-state outputs for ease of expansion of product terms and/or input variables.

L SUFFIX
CERAMIC PACKAGE
CASE 733



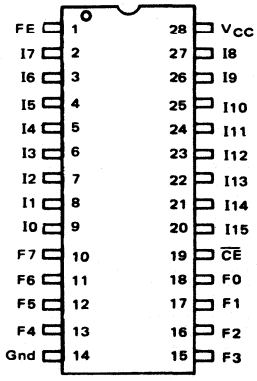
FEATURES

- Field Programmable (Ni-Cr Link)
- Input Variables – 16
- Output Functions – 8
- Product Terms – 48
- Address Access Time – 50 ns, Maximum
- Power Dissipation – 600 mW, Typical
- Input Loading – (–100 μA), Maximum
- Output Option:
  Three-State Outputs –MC29100/MC82100
  Open Collector Outputs – MC29101/MC82101
- Output Disable Function:
  Three-State – Hi-Z
  Open Collector – Hi
- Ceramic DIP

APPLICATIONS

- Large Read Only Memory
- Random Logic
- Code Conversion
- Peripheral Controllers
- Look-Up and Decision Tables
- Microprogramming
- Address Mapping
- Character Generators
- Sequential Controllers

PIN ASSIGNMENT

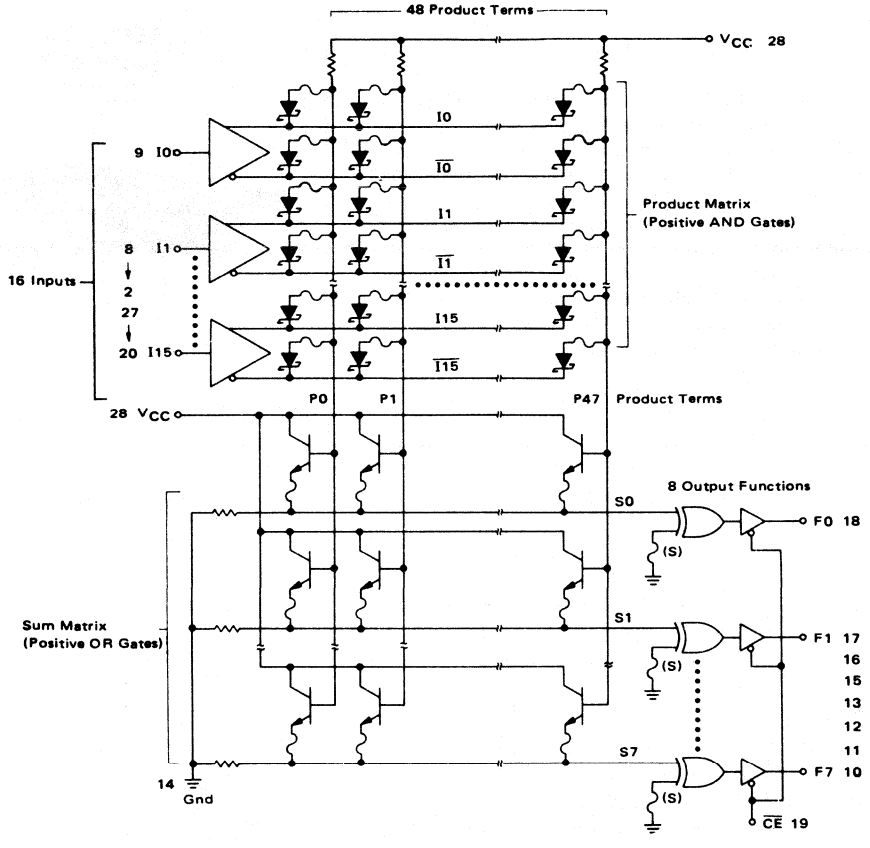


ORDERING INFORMATION

Temperature Range	Device	Device
0°C to +70°C	MC29100LC MC82100LC	MC29101LC MC82101LC
–55°C to +125°C	MC29100LM MC82100LM	MC29101LM MC82101LM

This is advance information and specifications are subject to change without notice.

BLOCK DIAGRAM



TRUTH TABLE

Let $P_n = \Pi_0^{15} (k_m I_m + j_m \overline{I}_m)$; k = 0, 1, X (Don't Care)

n = 0, 1, 2, . . . , 47

where

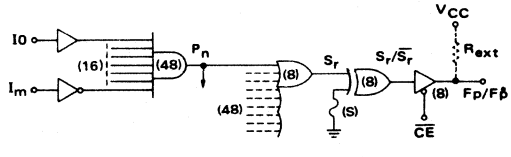
unprogrammed state: $j_m = k_m = 0$

programmed state: $j_m = \overline{k}_m$

$S_r = f(\Sigma_0^{47} P_n)$; $r \equiv p = 0, 1, 2, \ldots, 7$

Mode	P_n	$\overline{CE}$	Fp	$F\hat{p}$	$S_r \stackrel{?}{=} f(P_n)$
Disabled (MC29101/ MC82101)	X	1	1	1	X
Disabled (MC29100/ MC29100)			Hi-Z	Hi-Z	
Read	1	0	1	0	Yes
	0	0	0	1	
	X	0	0	1	No

FPLA TYPICAL LOGIC PATH



NOTE: For each of the 8 outputs, either the function Fp (active High) or $F\hat{p}$ (active Low) is available, but not both. The required function polarity is user-programmable via fuse (S).

$P_n = I0\overline{I}1\overline{I}2I3 \ldots \overline{I}_m$

$S_r = P0 + P1 + P2 + \ldots P_n$

$\overline{S}_r = \overline{P}0 \bullet \overline{P}1 \bullet \overline{P}2 \bullet \ldots \overline{P}_n$

$Fp = (\overline{CE}) + (S_r) = (\overline{CE}) + (P0 + P1 + P2 + \ldots P_n)$ with S = Short

$F\hat{p} = (\overline{CE}) + (\overline{S}_r) = (\overline{CE}) + (\overline{P}0 \bullet \overline{P}1 \bullet \overline{P}2 \bullet \ldots \overline{P}_n)$ with S = Open

MOTOROLA

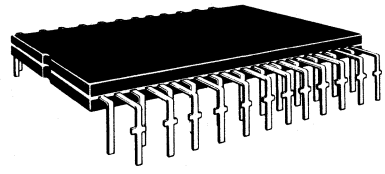
MC10800

INTRODUCTION

The MC10800 4-Bit ALU Slice is an LSI building block for digital processors. This circuit performs the necessary logic and arithmetic functions required to execute the various machine instructions. Each part is 4 bits wide and is "sliced" parallel to data flow. The MC10800 is fully expandable to larger word lengths by connecting circuits in parallel and features three input/output data ports for maximum system flexibility.

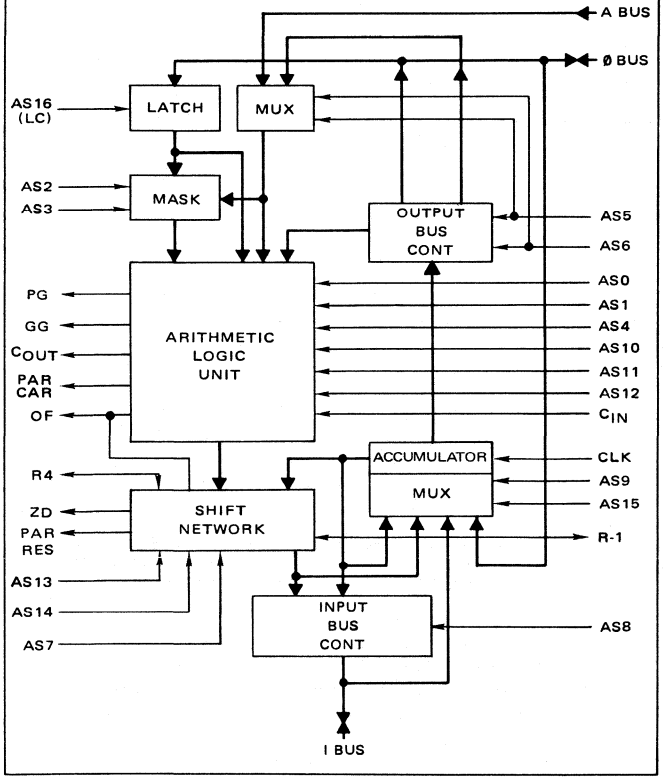
The 4-Bit ALU Slice as shown in the block diagram below contains latch/mask logic, ALU, shift network, accumulator, and bus control logic in a single bipolar circuit. Seventeen select lines are used to control all operations within the part.

MECL—LSI 4-BIT ALU SLICE

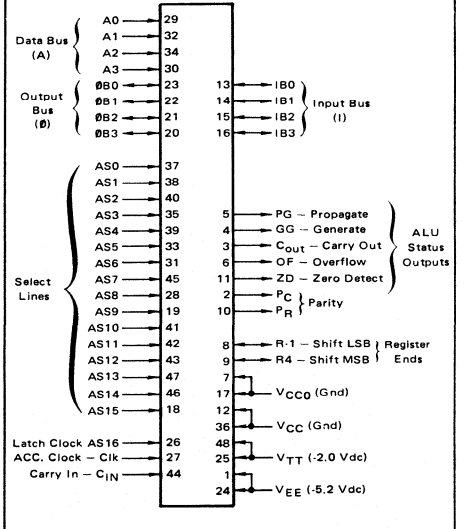


Case 725-01

4-BIT SLICE BLOCK DIAGRAM



INPUT/OUTPUT DIAGRAM



DS9437 R1

TABLE OF CONTENTS

IMPORTANT FEATURES

1. Powerful ALU
   a. Full binary and BCD arithmetic.
   b. A and Ø input data words treated equal.
   c. All logic functions.
   d. Internal lookahead carry with propagate and generate outputs.
2. Internal shift network.
   a. Left and right logic shift.
   b. Sign protect for arithmetic shift.
3. Versatile bus structure.
4. Master-slave accumulator for temporary storage.
5. Interfaces with MECL 10,000 register file circuits.
6. All necessary status outputs: overflow, zero detect, carry out, and sign bit.
7. Parity outputs for binary operations.
8. Full masking of Ø bus to A bus is provided in the latch/mask network.
9. Each part is 4 bits wide and the circuits can be operated in parallel to form any word size in increments of 4 bits.

M10800 LSI FAMILY DEVICES:

P/N	Description
MC10800	4-Bit ALU Slice
MC10801	Microprogram Control Function
MC10802	Timing Function
MC10803	Memory Interface Function

COMPATIBLE MOTOROLA MECL MEMORIES:

MCM10143	8 x 2 Multiport Register File
MCM10144	256 x 1 RAM
MCM10145	16 x 4 RAM
MCM10146	1024 x 1 RAM
MCM10147	128 x 1 RAM
MCM10149	256 x 4 PROM

COMPATIBLE LOGIC:

ECL 10,000: 100 Circuits, Industry-wide

ABSOLUTE MAXIMUM RATINGS (see Note 1)

RATING		SYMBOL	VALUE	UNIT
Supply Voltage (V_{CC} = 0)		V_{EE} V_{TT}	−8 to 0 −4 to 0	Vdc Vdc
Input Voltage (V_{CC} = 0)	Std Bus	V_{in} V_{in}	0 to V_{EE} Note 2	Vdc Vdc
Output Source Current	Cont Surge	I_o I_o	< 50 <100	mAdc mAdc
Storage Temp.		$T_{stg.}$	−55 to +150	°C
Junction Temp.		T_j	165	°C

NOTE: 1. Permanent device damage may occur if absolute maximum ratings are exceeded. Functional operation should be restricted to RECOMMENDED OPERATING CONDITIONS. Exposure to higher than recommended voltages for extended periods of time could affect device realiability.

NOTE: 2. Input voltage limit is V_{CC} to −2 Volts when the bus is used as an input and the output drivers are disabled.

SYSTEM OVERVIEW

Certain basic functional building blocks, as shown in Figure 1, are characteristic of high performance processors. These building blocks can be resolved into LSI circuits which, by proper use of control memory programming and circuit function select lines, will fit a wide range of system requirements. The Motorola M10800 family of LSI processor circuits is designed to provide these functional blocks and not limit the final system to any given system size or architecture.

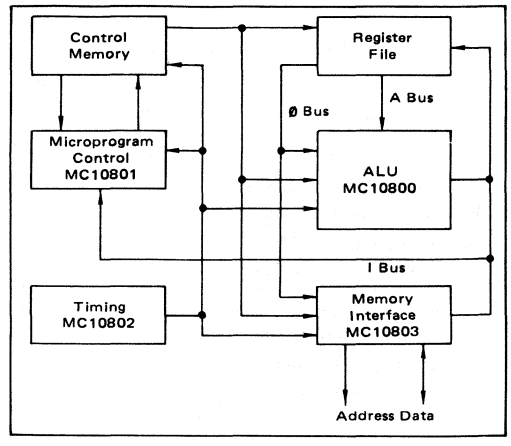
The ALU block in Figure 1 is filled by the MC10800 4-Bit ALU Slice. This circuit combines the mask logic, ALU, shift network, and accumulator to give a very powerful function set. In addition, the data routing paths and data I/O ports allow numerous options when configuring a system. When designing with the MC10800 function set it is possible to accomplish in a single pass what would require mulitple passes with other ALU circuits.

In the M10800 Family, the register file has been made a separate block from the ALU, because modern systems use a wide range of register file sizes – varying between 4 and 256 working registers. With high speed MECL RAM and register file circuits available, the designer is permitted to specify the optimum register file size and configuration for his particular system. Storage registers are available in the MC10801 Microprogram Control Function and MC10803 Memory Interface Function for special purpose functions such as instruction register, status register, program counter, index register, and stack pointer.

Virtually all modern computers use a microprogrammed instruction set. Microprogramming permits emulating machines, updating systems by increasing capability, modifying systems to meet specific customer requirements, and designing to take advantage of existing software. The MC10801 Microprogram Control Function contains the logic needed to address and sequence through microprogram storage. Each MC10801 is 4 bits wide and can be operated in parallel for larger control memory address words. The necessary storage, logic, and I/O is provided to generate next control memory address and handle status, branching, and interrupt functions.

One of the penalties normally paid to gain the advantages of microprogramming is system speed. Each processor instruction requires several microprogram steps. The Motorola M10800 LSI family makes use of MECL 10,000 circuit technology and interfacing to attain fast microprogram cycle times. In addition, other features of the family (such as a powerful ALU function set in the MC10800 and independent memory addressing in the MC10803), minimize the number of microprogram steps per system instruction. With the M10800 bipolar LSI family, it is possible to build fast microprogrammed systems which outperform dedicated hardwired systems using a slower technology.

FIGURE 1 – MICROPROGRAMMED PROCESSOR



The Control Memory block in Figure 1 is a separate section of the system, best selected by the designer. Any microprogram storage included on the LSI circuits results in design constraints. Microprogram storage can vary up to several thousand words, depending on system complexity, and is best built with individual MECL PROMs such as the MCM10149, or MECL RAMs such as the MCM10144 or MCM10146.

Any processor system must have access to external information from such sources as main memory, peripherals, and bulk storage. In the M10800 Family this chore is handled by the M10803 Memory Interface Function. This circuit is 4 bits wide and contains the necessary memory data and address storage. In addition, there are registers and an ALU for performing the various modes of memory addressing.

The MC10802 Timing Function ties the other function blocks together. This part provides the various clock phases as needed and makes it easy to interface to a manual test or control panel. As with other parts in the M10800 Family, the MC10802 is fully programmable for maximum system flexibility.

The Motorola M10800 circuits interface directly with all parts in the MECL 10,000 family. This provides a source for high speed ECL memories and interface circuits for MOS memories. It allows special hardware functions to be constructed for maximum system performance. MECL 10,000 MSI circuits can be used to multiplex status inputs for branch conditions, format priority interrupts, and build high speed array multipliers.

Versatility is a main point of the M10800 Family. The block diagram in Figure 1 is intended to illustrate the purpose of the various LSI functions and not restrict the designer to any particular system configuration or application.

PIN ASSIGNMENTS

Pin Designation	Pin Number	Description
A0	29	Data Bus A — LSB Input
A1	32	Data Bus A — NLSB Input
A2	34	Data Bus A — NMSB Input
A3	30	Data Bus A — MSB Input
ØB0	23	Output Bus — LSB I/O
ØB1	22	Output Bus — NLSB I/O
ØB2	21	Output Bus — NMSB I/O
ØB3	20	Output Bus — MSB I/O
IB0	13	Input Bus — LSB I/O
IB1	14	Input Bus — NLSB I/O
IB2	15	Input Bus — NMSB I/O
IB3	16	Input Bus — MSB I/O
AS0	37	Y Input Mux — Select Input
AS1	38	Y Input Mux — Select Input
AS4	39	Increment/Decrement by 2 — Select Input
AS2	40	X Input Mux — Select Input
AS3	35	X Input Mux — Select Input
AS5	33	Output Bus Control & A Input Mux — Select Input
AS6	31	Output Bus Control & A Input Mux — Select Input
AS10	41	Add/Subtract — Select Input
AS11	42	Binary/BCD — Select Input
AS12	43	Arithmetic/Logic Mode — Select Input
C_{in}	44	Carry Input
C_{out}	3	Carry Output
PG	5	Group Propagate Output
GG	4	Group Generate Output
OF	6	Overflow Output
PC	2	Parity of Carries Output
PR	10	Parity of Result Output
ZD	11	Zero Detect
AS7	45	Shift Network — Source Select Input
AS13	47	Shift Network — Function Select Input
AS14	46	Shift Network — Function Select Input
R4	9	Shift Network — MSB I/O
R-1	8	Shift Network — LSB I/O
AS9	19	Accumulator Mux & Input Bus Control — Select Input
AS15	18	Accumulator Mux & Input Bus Control — Select Input
AS8	28	Input Bus Driver — Enable Input
CLK	27	Accumulator — Clock Input
AS16 (LC)	26	Output Bus Latch — Clock Input
V_{EE}	1	−5.2 Volt Supply
V_{EE}	24	−5.2 Volt Supply
V_{TT}	25	−2.0 Volt Supply
V_{TT}	48	−2.0 Volt Supply
V_{CC}	12	Ground
V_{CC}	36	Ground
V_{CCO}	7	Ground
V_{CCO}	17	Ground

ARCHITECTURAL DESCRIPTION

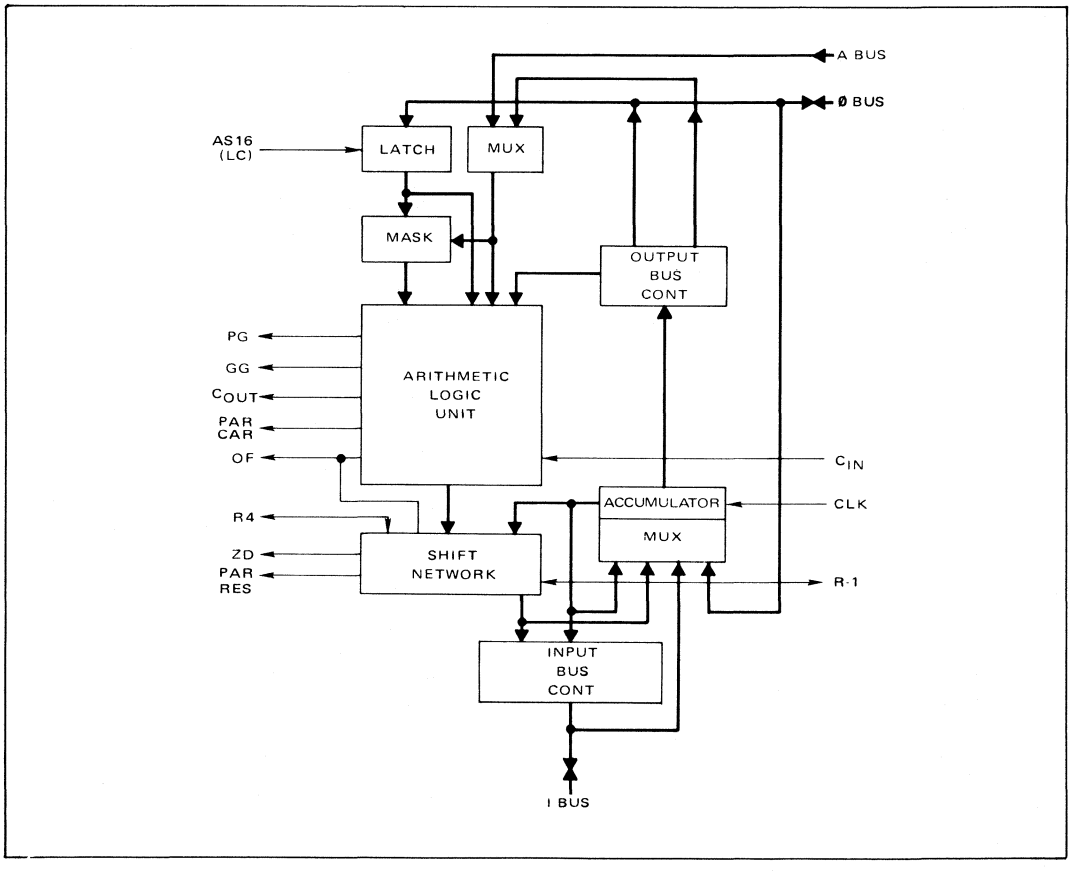
Data enters and exits the MC10800 4-Bit ALU Slice through the A bus, Output (ø bus), and Input (I bus) as shown in Figure 2. The ø bus and I bus are bidirectional, while the A bus is an input port only. These ports are each 4 bits wide requiring that MC10800s be operated in parallel for larger word lengths, single bit data paths C_{in}, C_{out}, R-1, and R4 are used to interconnect parallel MC10800s. The circuit contains two storage elements; the ø bus latch controlled by the latch clock (LC) and the accumulator controlled by the master clock (CLK). The remaining terminals in Figure 2 are status outputs and are used for second level look-ahead carry or for generation of processor condition codes. The individual blocks and I/O terminals in Figure 2 are described below.

Latch/Mask Network

The latch/mask network controls data to one input port of the Arithmetic Logic Unit. The holding latch is positioned to provide temporary storage for data entering through the Output Bus port. The latch clock input (LC) controls the latch operation. When not latched, data ripples through the latch and need not be clocked.

For microprocessor and microcontroller applications, it is desirable to be able to mask data entering a machine. The latch/mask network incorporates this feature. By using the mask select lines, it is possible to mask data on the Output Bus with the A Bus using the logic AND and logic OR functions.

FIGURE 2 – 4-BIT ALU SLICE – MC10800
DATA, STATUS AND CLOCK



Arithmetic Logic Unit

The Arithmetic Logic Unit (ALU) combined with the latch/mask network has the capability of perfoming logic operations, binary arithmetic, and BCD arithmetic on combinations of one, two, or three variables. These variables are the A bus, output bus latch, and accumulator. Variables are treated equally in both binary and BCD formats (A bus minus ø bus and ø bus minus A bus). BCD arithmetic operations are incorporated internally within the ALU and the BCD functions do not compromise system speed.

The ALU incorporates a 9's complement circuit to generate the necessary BCD complement function. The 9's complementer is used with BCD subtract and 9's complement instructions and the circuit is automatically enabled when these functions are selected.

The ALU section of the 4-Bit Slice provides the logic for overflow and carry out. Overflow provides the two's complement overflow of binary addition and subtraction. Overflow is also generated in the shift network and is the exclusive OR of the MSB and NMSB during a shift left operation. Carry out functions for both binary and BCD operations. If second level look-ahead carry is not used, the carry out of one 4-Bit Slice is connected to carry in of the following slice circuit for ripple carry.

The ALU generates the group propagate, group generate, and parity of carry outputs. Group propagate and generate are used for external look-ahead carry between 4-Bit Slice circuits. The propagate and generate outputs operate with both BCD and binary functions. Parity of carries is used for arithmetic error checking and is generated by the exclusive-ORing of C_{in}, carry from the LSB, carry from the NLSB, and carry from the NMSB.

Shift Network

The shift network following the ALU performs the data shift operations within the 4-Bit Slice. Select lines to the shift network control shift left, logic shift right, arithmetic shift right, and ripple through.

The arithmetic shift right provides sign protection for arithmetic shift operations. Only MSB is affected during an arithmetic shift right (towards the LSB), the most significant bit is repeated. R-1 and R4 input/outputs are brought out and are used for shift expansion when interconnecting 4-Bits Slice circuits. The zero detect is derived from the shift network outputs and detects the binary or BCD all zero state. Parity of results is also generated in the shift network. This output, used for parity checking, is generated by exclusive-ORing the shift outputs.

Input Bus Control

The input bus control manipulates the source of data to the Input Bus port. The input bus can receive data from either the shift network or the accumulator. In addition, this control circuit can inhibit data from being routed to the input bus. This allows the input bus to enter data into the accumulator or to be used for other system functions not related to the 4-Bit Slice.

Accumulator/Multiplexer

The master-slave accumulator provides for high speed iterative computer operations. These can include repeated add with accumulated sum, multiply, divide, and multiple shift operations. A multiplexer circuit feeds the accumulator from one of three possible sources as controlled by select lines. These sources are the results of the shift network, the input bus, or the output bus. A fourth condition inhibits the accumulator clock and stored data is retained. Data is entered on the rising (V_{OL} to V_{OH}) clock edge.

Output Bus Control

The output bus control section distributes the outputs of the accumulator to various points in the 4-Bit slice. Select lines route the accumulator to either the A input multiplexer or to the output bus. In addition, the accumulator can be routed to the ALU for mask and compare type operation. A fourth state of the output bus control inhibits the accumulator from going to any of the three above destinations.

A Input

The A input consists of four pins, A0, A1, A2, and A3 which serve as input data paths to the arithmetic logic unit. These inputs are designed to operate in a negative logic data format with a MECL V_{OL} being a logic 1. Because of the BCD functions, the 4-Bit Slice does not directly accept both positive and negative logic formats. The inputs are designated with A0 as the least significant of the 4 bits in the circuit and A3 as the most significant bit.

Output Bus

The output bus consits of four terminals, øB0 through øB3, which function as both data inputs and data outputs. As with the A input, the output bus pins are in negative logic and øB0 is the least significant bit within the part. The output bus when used as an input is routed to the holding latch, and accumulator multiplexer. As an output port, these terminals are used to connect data in the accumulator to the output bus as shown in Figure 2.

Input Bus

The input bus consists of four terminals, IB0 through IB3, which function as both data inputs and data outputs. As with the A buffer and the output bus, the input bus pins are in negative logic and IB0 is the least significant bit within the part. The input bus when used as an input is routed to the accumulator. As an output port, these terminals are used to connect data from either the accumulator or shift network results to the input bus as shown in Figure 2.

Carry In

Carry in, C_{in}, is used to interconnect 4-Bit Slice circuits in a system. For ripple carry, carry in is connected to carry out of the preceding 4-Bit Slice. When look-ahead carry is incorporated, the carry in is connected to the look-ahead carry logic.

Carry in is only used for arithmetic operations and has no effect on any logic operation. The carry in functions for both binary and BCD arithmetic operations. Carry in operates in a negative logic mode with V_{OL} being a logic 1.

Carry Out

Carry out, C_{out}, signals that the calculated value within the ALU has exceeded the maximum capacity of the four ALU output lines. Any binary total over count 15 (1111) or BCD total over count 9 (1001) results in a carry out. When ripple carry is used, carry out is connected to carry in of the following 4-Bit Slice.

Shift Interconnects R-1 and R4

R-1 and R4 are provided to interconnect 4-Bit Slice circutis for shift operations. R-1 and R4 function as both inputs and outputs depending on the shift direction. For a shift left (toward the MSB) R-1 is an input for the R0 bit and R4 is an output for the MSB. For a logic shift right R-1 is an output for the LSB and R4 is an input to R3. MSB is also connected to R4 during a no shift operation and during an arithmetic shift right. This allows R4 to be used as a status output for sign detection. When not used as outputs, the internal drivers for R-1 and R4 are held at a negative logic 1 so the shift interconnects can function as inputs using the MECL emitter dot. See Table 1.

TABLE 1

	I/O FUNCTION	
SHIFT OPERATION	R-1	R4
Shift Left	Shift Input	Shift Output
No Shift	Not Used	MSB Output
Logic Shift Right	Shift Output	Shift Input
Arithmetic Shift Right	Shift Output	MSB Output

Group Propagate and Group Generate

The group propagate, PG, and group generate, GG, outputs are used in conjunction with external look-ahead carry logic for faster system operation. Using this technique, the carry in signals to the 4-Bit Slice circuits are generated faster than with ripple carry. The propagate output goes to the logic 1 when the maximum number value occurs on the ALU outputs. This is count 15 (1111) for binary functions and count 9 (1001) for BCD functions. For binary functions, generate occurs with any value of 16 (10000) or larger and for BCD functions any number value of 10 (10000) or larger.

Group propagate and group generate outputs are used only for arithmetic operations in a system to allow faster generation of carry in signals. They serve no function for ALU logic operations.

Overflow OF

Overflow is used only with two's complement arithmetic and shows that the maximum system word or byte value has been exceeded. In a system, only the overflow output from the 4-Bit Slice operating on the most significant bits of the data word is used.

In addition to overflow caused by an ALU operation, it is possible to have overflow as a result of a shift left (toward the MSB) in the shift network. This happens when the sign bit is changed as a result of the shift left operation.

Normally the overflow of the ALU and shift network are ORed together so that either causes an overflow condition. The exception to this occurs when the accumulator is routed to the shift network inputs. At this time, the ALU overflow is inhibited from the OF output. Overflow is not used with BCD arithmetic.

Zero Detect ZD

Zero detect signals the all zero condition (0000) at the output of the shift network. Zero detect functions for logic operations, binary arithmetic, and BCD arithmetic operations within the ALU. By having the zero detect at the output of the shift network, it is possible to detect zero status after a shift has been performed. Zero detect is defined by the following equation:

$$ZD = \overline{R0} \cdot \overline{R1} \cdot \overline{R2} \cdot \overline{R3}$$

where $\overline{R0}$ through $\overline{R3}$ are the internal outputs from the shift network.

Parity Outputs PAR CAR and PAR RES

Parity bits are used to detect system errors in data handling. With a single parity bit, it is possible to detect a single bit error or any combination of an odd number of bit errors.

For parity checking binary arithmetic operations, two parity points are generated in the MECL 4-Bit Slice. These are parity of carries (PAR CAR) and parity of results (PAR RES). Parity of carries is the parity of the individual bit carries internal to the slice.

$$\text{PAR CAR} = C\,IN \oplus C0 \oplus C1 \oplus C2$$

Parity of results is the parity of the individual result bits at the output of the shift network.

$$\text{PAR RES} = R0 \oplus R1 \oplus R2 \oplus R3$$

Accumulator Clock CLK

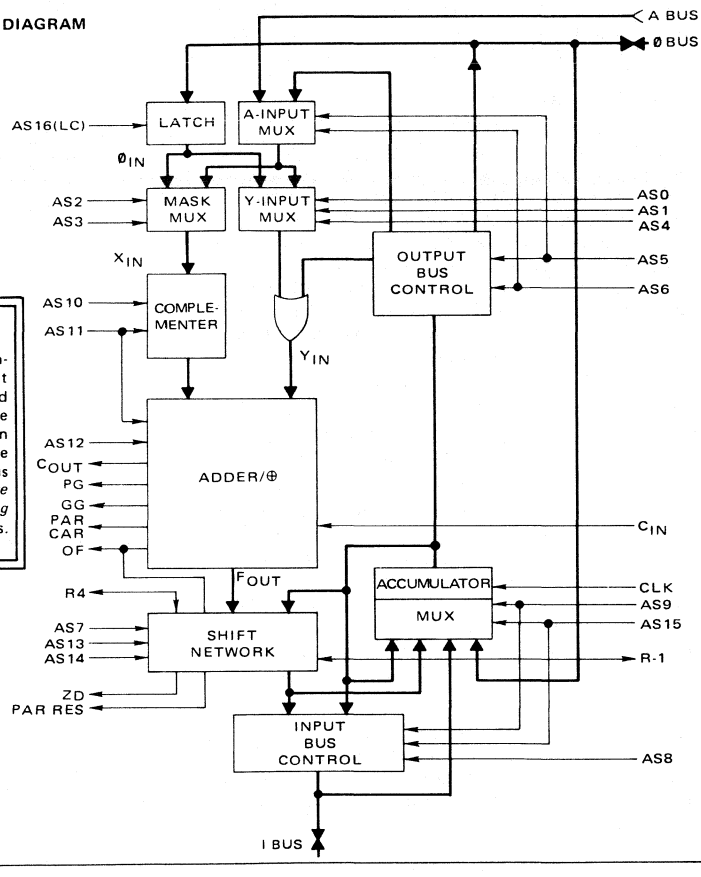
The accumulator is constructed of master-slave flip flops and must be clocked to change stored data. As is characteristic of MECL flip flops, the accumulator is clocked on the positive going (V_{OL} to V_{OH}) clock edge. At that time, data on the accumulator inputs is transferred to the accumulator outputs.

Signals on the accumulator inputs can change at any time with the clock input at either logic state and not change accumulator outputs. The only restriction on changing accumulator inputs is during the set up and hold time near the positive going clock edge.

Latch Clock AS16

Latch clock AS16 controls the storage of data in the holding latch on the output bus. When the latch clock is at V_{IH} data ripples through the latch, interconnecting the output bus with the ALU inputs. When AS16 is at V_{OL} data is stored in the latch and latch outputs are not affected by any changes in information on the output bus.

FIGURE 3 – FUNCTIONAL BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

Seventeen select lines AS0 through AS16 control the flow of data within the MC10800 4-Bit Slice and determine the arithmetic, logic, and shift operations performed on the data. The following information describes the operation of these select lines, then shows how these select lines combine to perform the various MC10800 functions. *The truth tables are expressed in negative logic with V_{OL} being a logic 1 and V_{OH} a logic 0,* Figure 3 applies.

SELECT LINE OPERATION

Y Input Mux Select Inputs AS0 and AS1

AS0 and AS1 control the source of data to the Y-Input mux of the ALU. These select lines allow selection of either the A Input Mux, the output bus, all logic 0 or all logic 1 bits. Table 2 illustrates the operation of these two select lines.

The ALU Y input is also the port for entering accumulator data into the ALU. This is accomplished by setting AS0 and AS1 to a logic 0 and enabling the accumulator with AS5 and AS6 as described in Table 4 and Table 5.

TABLE 2

AS0	AS1	ALU Y INPUT
0	0	LOGIC 0
0	1	OUTPUT BUS LATCH
1	0	A INPUT MUX
1	1	LOGIC 1

Increment/Decrement by 2 Select Input AS4

Select line AS4 is used to give the MECL 4-Bit Slice an increment or decrement by 2 function capability. When this input is held at a logic 1 (V_{OL}) it has no effect on the circuit. When at a logic 0, it is used with AS0 and AS1 to force a code 0010 (plus 2) or 1110 (minus 2) on the Y input of the ALU. In a system, this input would normally be used only with the 4-Bit Slice operating on the least significant bits in a word or byte. However, other slice locations could also be used to add such constants as 2, 32, 34, 512, 544, 546, etc. AS4 operation with AS0 and AS1 is shown in Table 3.

TABLE 3

AS4	ASO	AS1	ALU Y INPUT
1	TABLE 2		DETERMINED BY AS0, AS1
0	0	0	PLUS 2 (0010)
0	1	1	MINUS 2 (1110)

The combinations AS0•$\overline{AS1}$•$\overline{AS4}$ and $\overline{AS0}$•AS1•$\overline{AS4}$ are not normally used. AS0•$\overline{AS1}$•$\overline{AS4}$ results in Y0 = logic 0, Y1 = logic 1, Y2 = A2, and Y3 = A3. $\overline{AS0}$•AS1•$\overline{AS4}$ results in Y0 = logic 0, Y1 = logic 1, Y2 = ØB2, and Y3 = ØB3.

Output Bus Control and A Input Mux Select Inputs AS5 & AS6

Select lines AS5 and AS6 control the destination of the accumulator output. The accumulator can be routed to five locations in the MECL 4-Bit Slice. Three locations, the output bus, the ALU A input, the ALU ACC input are controlled by AS5 and AS6. A fourth state of AS5 and AS6 inhibits the accumulator from any of these three destination. Other destinations for the accumulator are the input bus as described in the section on AS9 and AS15 and to the shift network inputs as controlled by AS7. (Table 9 and Table 11).

When drivers to the output bus are not enabled by AS5 and AS6, they are forced to a logic 1 (V_{OL}) so this bus can take advantage of the MECL emitter dotting. The accumulator input to the ALU requires a logic 0 when not used as a data input.

AS5•$\overline{AS6}$ enables the accumulator on the Ø Bus. The MC10800 can simultaneously output the accumulator contents onto the system Ø Bus and input the accumulator contents to the Ø Bus port of the ALU. Only external data on the Ø Bus is ANDed to the accumulator contents when AS5•$\overline{AS6}$ is selected.

The MC10800 A bus terminals are input only and AS5 and AS6 select either the A bus inputs or the accumulator to the ALU A input.

Logic State $\overline{AS5}$•AS6 operates in conjunction with AS0 and AS1 to enter data into the ALU Y input. The output of the Y-input mux is logically ORed to the accumulator ALU input. Table 5 illustrates the operation of these select lines.

TABLE 4

AS5	AS6	Ø BUS	A IN MUX	ALU
0	0	Ø BUS	A BUS	0
0	1	Ø BUS	A BUS	ACC
1	0	ACC•Ø BUS	A BUS	0
1	1	Ø BUS	ACC	0

TABLE 5

$\overline{AS5}$•AS6	AS0	AS1	ALU Y INPUT
0	TABLE 2		Determined by AS0, AS1
1	0	0	ACCUMULATOR
1	0	1	ACC OR Ø BUS
1	1	0	ACC OR A MUX
1	1	1	LOGIC 1

X Input Mux Select Inputs AS2 and AS3

Select lines AS2 and AS3 control the data path to the other ALU input (X input). These lines can select either the A bus or the output bus. In addition, AS2 and AS3 provide masking capability within the 4-Bit Slice. These select lines control the logic functions – (A bus OR Ø bus) and (A bus AND Ø bus). This allows any bit or bits to be masked to either a logic 1 or 0 with masking information on either the A bus or Ø bus terminals. The advantage of doing masking prior to the ALU is that it allows single pass mask and compare within the 4-Bit Slice. AS2 and AS3 operation is shown in Table 6.

TABLE 6

AS2	AS3	ALU X INPUT
0	0	A MUX AND Ø BUS
0	1	Ø BUS
1	0	A MUX
1	1	A MUX OR Ø BUS

Add/Subtract and Binary/BCD Select Inputs AS10 and AS11

Select line AS10 and add/subtract control enables the complementer. During the add mode the X_{in} data is passed directly through, and during the subtract mode the data is complemented. The complement function is also modified by AS11. The 9's complement is generated for BCD subtract, and data is inverted (1's complement) for binary subtract.

If the ALU is in the logic (Exclusive-OR) mode the complementer is used to selectively invert the X_{in} data. AS11 should be set to the binary mode, and AS10 is used to control inversion of the data.

Arithmetic/Logic Mode Select Input AS12

AS12 is the mode control for the 4-Bit adder. This input determines if the function performed in the ALU is an arithmetic or logic operation. The logic mode disables the carry between bits and the function performed is the Exclusive-OR of the two inputs to the adder.

Shift Network Source Select Input AS7

AS7 controls the information source to the shift network. The MECL 4-Bit Slice is designed to allow shifting data from the accumulator or from the ALU. The accumulator shift operation is useful in multiply and divide add/shift routines. AS7 follows the truth table shown in Table 9.

Shift Network Function Select Inputs AS13 and AS14

AS13 and AS14 control the operation of the shift network following the ALU in the 4-Bit Slice. The four possible operations are: no shift (straight through), shift left one bit, logic shift right one bit, and arithmetic shift right one bit. The truth table for AS13 and AS14 is shown in Table 10.

Shift left shifts each bit at the shift network inputs (F inputs) one bit left (toward the MSB). This operation provides the function for both arithmetic and logic shift left. Logic shift right shifts each bit at the F inputs one bit right (toward the LSB). This shift mode is used in all 4-Bit Slice circuits for logic shift right and all except the slice circuit handling the most significant bit of a word or byte for an arithmetic shift right. During an arithmetic right shift, it is necessary to have sign protection for a number expressed in 2's complement or 1's complement notation. This is accomplished by repeating the most significant bit during and arithmetic shift right.

R-1 is an input for shift left and an output for both logic and arithmetic shift right. This pin is not used for no shift. R4 is an input for logic shift right and an output for all other AS13 and AS14 operations. This feature allows R4 to function as a sign bit status output on the MC10800 having the sign bit as the MSB within the part.

TABLE 7

AS10	AS11	FUNCTION
0	0	SUBTRACT BCD (9's COMPLEMENT)
0	1	SUBTRACT BINARY (INVERT)
1	0	ADD BCD
1	1	ADD BINARY

TABLE 8

AS12	MODE
0	LOGIC (Exclusive-OR)
1	ARITHMETIC

TABLE 9

AS7	SHIFT NETWORK SOURCE
0	ACCUMULATOR
1	ALU

TABLE 10

AS13	AS14	SHIFT OPERATION
0	0	SHIFT LEFT
1	0	NO SHIFT
0	1	LOGIC SHIFT RIGHT
1	1	ARITHMETIC SHIFT RIGHT

Accumulator Mux & Input Bus Control Select Inputs AS9 & AS15

Select lines AS9 and AS15 perform two functions in the MECL 4-Bit Slice. One is to control the source of data to the accumulator, the other to control the source of data to the input bus drivers. The accumulator can store data from three independent points in the 4-Bit Slice. These are the input bus, the output bus, and the shift network outputs. A fourth condition on AS9 and AS15 feeds the accumulator back on itself so the accumulator clock is effectively disabled. This permits storage of data in the accumulator with a continuous system clock entering the slice circuit. The clock disable state of AS9 and AS15 is designed so that only the clock can load information into the accumulator and the accumulator status cannot be altered by the select lines alone.

AS9 and AS15 route either the accumulator or the shift network outputs to the input bus drivers. When the results of the shift network are gated to the accumulator, the accumulator is the source of data to the input bus drivers. For all other combinations AS9 and AS15, the shift network outputs are gated to the bus drivers. The accumulator clock can be disabled when reading the accumulator to the shift network and using the shift network as a feedback path. Table 11 illustrates the operation of AS9 and AS15.

TABLE 11

AS9	AS15	INPUT TO ACCUMULATOR	INPUT BUS SOURCE
0	0	SHIFT RESULTS	ACCUMULATOR
0	1	OUTPUT BUS	SHIFT RESULTS
1	0	INPUT BUS	SHIFT RESULTS
1	1	ACCUMULATOR	SHIFT RESULTS

Input Bus Driver Enable Input AS8

AS8 inhibits and enables the input bus driver circuits. When this select line is at a logic 1 (V_{OL}) the input bus drivers are enabled and data from either the shift network or the accumulator is routed to the input bus. A logic 0 on AS8 disables the input bus drivers so the input bus port can be used to input data or so the input bus can route data independent of the 4-Bit Slice in a system. When disabled the input bus drivers assume a logic 1 state (V_{OL}). Forcing the outputs low permits the use of MECL emitter dotting on the system input bus. The truth table for AS8 is shown in Table 12.

TABLE 12

AS8	INPUT BUS
0	DISABLE OUTPUTS
1	ENABLE OUTPUTS

Accumulator Clock Input CLK and Output Bus Latch Clock Input – AS16

Data is entered into the accumulator on the rising edge of the clock signal. The data source is selected by AS9 and AS15. Latch clock AS16 controls the storage of data in the holding latch on the output bus. When AS16 is a logic 0 (V_{IH}) data ripples through the latch. When AS16 is a logic 1 data is stored in the latch and the latch outputs are not affected by information changes on the output bus. Table 13 is the truth table for AS16.

TABLE 13

AS16	LATCH OPERATION
0	ENABLED
1	LATCHED

ALU LOGIC OPERATION FUNCTION SET

The output bus latch, the A input multiplexer, and the accumulator are sources of data to the ALU. Following the various truth tables of the given select lines a full set of logic operation can be performed in the ALU.

The equivalent block diagram of the ALU for logic operations is shown in Figure 4. The adder is set to the logic mode (AS12 = 0), therefore, F_{OUT} is the Exclusive-OR of selected sources X and Y. The complementer is programmed as a conditional inverter (AS11 = 1) dependent on AS10. The X source is selected by inputs AS2 and AS3. The Y source is selected by AS0 and AS1 (AS4 = 1) and OR-ed with the accumulator (selected by $\overline{AS5}$·AS6). A selected logic function set is shown in Table 14.

Other functions and select line combinations are possible with many redundant operations. Other conditions can be determined from previous truth tables.

FIGURE 4 – BLOCK DIAGRAM OF ALU LOGIC OPERATION

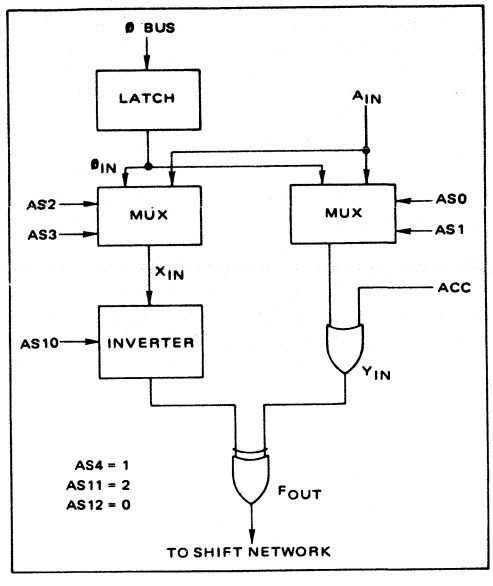
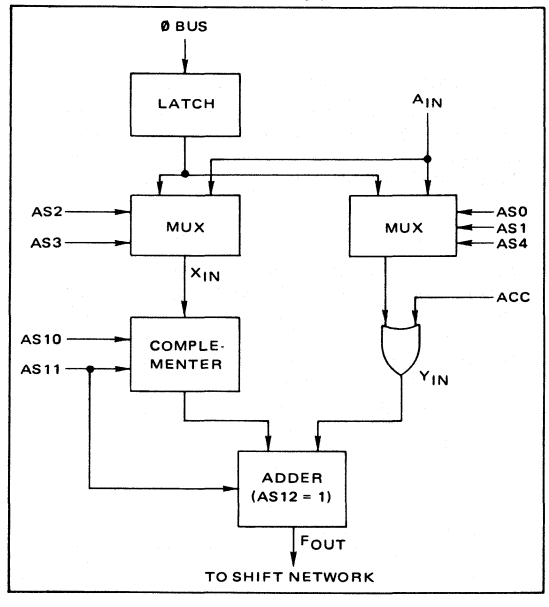


TABLE 14

Y MUX		X MUX		INV	ACC	FUNCTION
AS0	AS1	AS2	AS3	AS10	$\overline{AS5}$·AS6	
0	1	0	1	1	0	LOGIC 0
0	0	1	0	1	0	A
0	0	0	1	1	0	Ø
0	0	1	0	0	0	$\overline{A}$
0	0	0	1	0	0	$\overline{Ø}$
0	0	1	1	1	0	A + Ø
0	1	0	0	0	0	A + $\overline{Ø}$
1	0	0	0	0	0	$\overline{A}$ + Ø
0	0	0	0	1	0	A · Ø
0	1	1	1	1	0	A · $\overline{Ø}$
0	1	0	0	1	0	$\overline{A}$ · Ø
0	1	1	0	1	0	A ⊕ Ø
0	1	1	0	0	0	A ⊕ $\overline{Ø}$
0	0	0	0	0	0	$\overline{A}$ · $\overline{Ø}$
0	0	1	1	0	0	$\overline{A + Ø}$
0	1	0	1	0	0	LOGIC 1
1	0	1	0	1	1	ACC · $\overline{A}$
0	1	0	1	1	1	ACC · $\overline{Ø}$
1	0	1	0	0	1	$\overline{ACC}$ + A
0	1	0	1	0	1	$\overline{ACC}$ + Ø
0	0	1	0	1	1	ACC ⊕ A
0	0	1	0	0	1	ACC ⊕ $\overline{A}$
0	0	0	1	1	1	ACC ⊕ Ø
0	0	0	1	0	1	ACC ⊕ $\overline{Ø}$
0	0	0	0	1	1	ACC ⊕ A·Ø
0	0	0	0	0	1	ACC ⊕ $\overline{A·Ø}$
0	0	1	1	1	1	ACC ⊕ A+Ø
0	0	1	1	0	1	ACC ⊕ $\overline{A+Ø}$

+ = Logical Inclusive OR
· = Logical AND
⊕ = Logical Exclusive OR

FIGURE 5 – BLOCK DIAGRAM OF ALU ARITHMETIC OPERATION



ALU ARITHMETIC OPERATION FUNCTION SET

The block diagram for arithmetic operation is similar to logic operation, however, the complementer and adder go to arithmetic mode. Select input AS12 is set to logic 1 for adder operation, however, AS4 is now used for increment/decrement by 2 and AS11 selects the binary or BCD operation.

The various arithmetic functions in the 4-Bit Slice are determined by the choice of operands to the adder. Most binary functions have a BCD equivalent, however, operands for BCD functions should be valid BCD characters.

Table 15 shows a selected arithmetic function set. Similar to the logic function set other combinations of select lines and operations are possible. These can be generated as needed by the previous truth tables.

TABLE 15

Y MUX		X MUX		±2	COMPLE-MENT	ACC	BINARY FUNCTION (PLUS C_{IN})	BCD FUNCTION (PLUS C_{IN})
AS0	AS1	AS2	AS3	AS4	AS10	$\overline{AS5}$·AS6	AS11 = 1	AS11 = 0
1	0	0	1	1	1	0	A PLUS Ø	A PLUS Ø
1	0	0	1	1	0	0	A PLUS $\overline{Ø}$	A PLUS 9's COMP. Ø
0	1	1	0	1	0	0	Ø PLUS $\overline{A}$	Ø PLUS 9's COMP. A
0	0	1	0	1	1	0	A	A
0	0	0	1	1	1	0	Ø	Ø
0	0	1	0	1	0	0	$\overline{A}$	9's COMP. A
0	0	0	1	1	0	0	$\overline{Ø}$	9's COMP. Ø
1	1	1	0	1	1	0	–1 PLUS A	*
1	1	0	1	1	1	0	–1 PLUS Ø	*
1	1	1	0	0	1	0	–2 PLUS A	*
1	1	0	1	0	1	0	–2 PLUS Ø	*
0	0	1	0	0	1	0	+2 PLUS A	+2 PLUS A
0	0	0	1	0	1	0	+2 PLUS Ø	+2 PLUS Ø
1	0	1	0	1	1	0	A PLUS A	A PLUS A
0	1	0	1	1	1	0	Ø PLUS Ø	Ø PLUS Ø
0	0	1	0	1	1	1	ACC PLUS A	ACC PLUS A
0	0	0	1	1	1	1	ACC PLUS Ø	ACC PLUS Ø
0	0	1	0	1	0	1	ACC PLUS $\overline{A}$	ACC PLUS 9's COMP. A
0	0	0	1	1	0	1	ACC PLUS $\overline{Ø}$	ACC PLUS 9's COMP. Ø
0	0	0	0	1	1	1	ACC PLUS A·Ø	ACC PLUS A·Ø
0	0	0	0	1	0	1	ACC PLUS $A\cdot\overline{Ø}$	ACC PLUS 9's COMP. A·Ø
0	0	1	1	1	1	1	ACC PLUS A + Ø	*
0	0	1	1	1	0	1	ACC PLUS $\overline{A}$ + Ø	*

*Not Defined in BCD

DATA ROUTING FUNCTION SET

Data routing in the MECL 4-Bit Slice covers the routing of data to and from both the shift network and accumulator. Data routing is controlled by select lines AS5, AS6, AS7, AS8, AS9, and AS15. AS5 and AS6 control the output destination of accumulator data, AS7 determines the source of data to the shift network, and AS8 enables and disables the input bus drivers. AS9 and AS15 control the source of data to both the accumulator and input bus drivers. Table 16 shows the truth table for AS7, AS8, AS9, and AS15.

The first four columns show all select line states. The fifth column shows the input source to the accumulator as controlled by AS9 and AS15. The possible accumulator inputs are: (1) ACC which is accumulator feedback on itself for accumulator clock disable. (2) IB which connects the input bus to the accumulator inputs, (3) ØB which connects the output bus to the accumulator inputs, and (4) RES which connects the results of shift network to the accumulator input. The sixth column shows the two possible sources of input data to the shift network. These are from the accumulator (ACC) for accumulator shift operations, and from the ALU function outputs (F). The final column in the table shows the status of the input bus drivers. A logic 0 on AS8 disables the driver circuits so this part can be used to input data or for other system functions not related to the 4-Bit Slice. When enabled, the input bus port will output information from the accumulator or from the results of the shift network.

TABLE 16

				FUNCTION		
AS7	AS8	AS9	AS15	ACC SOURCE	SHIFT SOURCE	INPUT BUS
0	0	0	0	RES	ACC	DISABLE
0	0	0	1	ØB	ACC	DISABLE
0	0	1	0	IB	ACC	DISABLE
0	0	1	1	ACC	ACC	DISABLE
0	1	0	0	RES	ACC	ACC
0	1	0	1	ØB	ACC	RES
0	1	1	0	IB	ACC	RES
0	1	1	1	ACC	ACC	RES
1	0	0	0	RES	F_{OUT}	DISABLE
1	0	0	1	ØB	F_{OUT}	DISABLE
1	0	1	0	IB	F_{OUT}	DISABLE
1	0	1	1	ACC	F_{OUT}	DISABLE
1	1	0	0	RES	F_{OUT}	ACC
1	1	0	1	ØB	F_{OUT}	RES
1	1	1	0	IB	F_{OUT}	RES
1	1	1	1	ACC	F_{OUT}	RES

RECOMMENDED OPERATING CONDITIONS – MC10800

PARAMETER	SYMBOL	VALUE	UNIT
Supply Voltage (V_{CC} = 0 Volts)	V_{TT} V_{EE}	−1.9 to −2.2 −4.68 to −5.72	Vdc Vdc
Operating Temp. (Functional)	T_A	−30 to +85	°C
Output Drive	–	50Ω to −2.0 Vdc	–
Maximum Clock Input Rise and Fall Time (20% to 80%)	t_r, t_f	10	ns
Minimum Clock Pulse Width	PW	5	ns

ELECTRICAL CHARACTERISTICS

Each MECL 10,000 series circuit has been designed to meet the dc specifications shown in the test table, after thermal equilibrium has been established. The circuit is in a test socket or mounted on a printed circuit board and transverse air flow greater than 500 linear fpm is maintained. Outputs are terminated through a 50-ohm resistor to −2.0 volts. Test procedures are shown for only one input, or for one set of input conditions. Other inputs tested in the same manner.

@Test Temperature	TEST VOLTAGE VALUES (Volts) V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{EE}	V_{TT}
30°C	−0.890	1.890	1.205	1.500	5.2	−2.0
+25°C	−0.810	−1.850	1.105	1.475	5.2	−2.0
+85°C	−0.700	1.825	1.035	1.440	5.2	−2.0

Characteristic	Symbol	Pin Under Test	MC10800 TEST LIMITS −30°C Min	−30°C Max	+25°C Min	+25°C Typ	+25°C Max	+85°C Min	+85°C Max	Unit	VOLTAGE APPLIED TO PINS LISTED BELOW: V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{EE}	V_{TT}	(V_{CC}) Gnd
Power Supply Drain Current	I_{EE}	1, 24	–	–	–	195	240	–	–	mAdc	–	–	–	–	1, 24	25, 48	12, 36
	I_{TT}	25, 48	–	–	–	180	199	–	–		–	–	–	–			7, 17
Input Current	I_{inH}	23	–	–	–	–	65	–	–	µAdc	23	–	–	–			
		31	–	–	–	–	350	–	–		31	–	–	–			
		27	–	–	–	–	435	–	–		27	–	–	–			
	I_{inL}	31	–	–	0.5	–	–	–	–	µAdc	–	31	–	–			
Logic "0" Output Voltage	V_{OH}	13	1.060	−0.890	−0.960		−0.810	−0.890	−0.700	Vdc	18,27,30,38,*	–	–	–			
		10	1.060	−0.890	−0.960		−0.810	−0.890	−0.700	Vdc	18,27,30,38,*	–	–	–			
Logic "1" ** Output Voltage	V_{OL}	13	−1.940	−1.675	−1.900		−1.650	−1.875	−1.615	Vdc	30*	–	–	–			
		10	−1.890	1.675	−1.850		−1.650	−1.825	−1.615	Vdc	*	–	–	–			
Logic "0" Threshold Voltage	V_{OHA}	13	−1.080		−0.980			−0.910		Vdc	27,29,**	–	–	18			
		10	−1.080		−0.980			−0.910		Vdc	29,**	–	–	47			
Logic "1" Threshold Voltage	V_{OLA}	13		1.655			−1.630		1.595	Vdc	44,46,**	–	29	–			
		10		1.655			−1.630		1.595	Vdc	29,**	–	47	–			

*V_{IH} on pins 19, 26, 29, 31, 32, 33, 34, 35, 37, 44, 46.
**V_{IH} on pins 19, 26, 30, 31, 32, 33, 34, 35, 37.
*** The bi-directional outputs are specified at −1.90 volts for V_{OL} min.

SETUP AND HOLD TIMES (NANOSECONDS OVER TEMPERATURE RANGE).

Input	Path	Mode	Setup Max	Hold Max
A Bus	→ A MUX → MASK MUX → COMP → ALU → SHIFT → ACC	Arith Subtract	38.0	−15.0
A Bus	→ A MUX → Y MUX → ALU → SHIFT → ACC	Logical	19.0	−5.0
ϕ Bus	→ LATCH → MASK MUX → COMP → ALU → SHIFT → ACC	Arith Subtract	38.0	−15.0
ϕ Bus	→ LATCH → Y MUX → ALU → SHIFT → ACC	Logical	20.0	−5.0
ϕ Bus	→ ACCUMULATOR	Direct	7.0	+5.0
I Bus	→ ACCUMULATOR	Direct	7.0	+5.0
AS0, 1	Y MUX → ALU → SHIFT → ACC	Arith Add	32.0	−15.0
AS4	ALU → SHIFT → ACC	Logical	15.0	0.0
AS3	MASK MUX → COMP → ALU → SHIFT → ACC	Arith Subtract	36.0	−17.0
AS2	MASK MUX → ALU → SHIFT → ACC	Logical (No Comp)	22.0	−5.0
AS5, 6	ALU → SHIFT → ACC		20.0	−5.0
AS7	SHIFT INPUT MUX → SHIFT → ACCUMULATOR	Direct	10.0	+5.0
AS9, 15	ACCUMULATOR INPUT MUX → ACC	Direct	8.0	+7.0
AS10	COMP → ALU → SHIFT → ACC	Arith	35.0	−15.0
AS11	ALU → SHIFT → ACC	Arith	21.0	−2.0
AS12	ALU → SHIFT → ACC		28.0	−10.0
AS12	ALU → SHIFT → ACC		14.0	+2.0
AS13, 14	SHIFT NETWORK → ACCUMULATOR	Direct	16.0	+5.0
CIN	→ ALU → SHIFT → ACC	Arith	19.0	+2.0
R-1, R4	SHIFT NETWORK → ACCUMULATOR	Direct	8.0	+5.0
ϕ Bus	→ LATCH (AS16 – LATCH CLOCK)	Direct	5.0	+6.0

PROPAGATION DELAYS (NANOSECONDS)

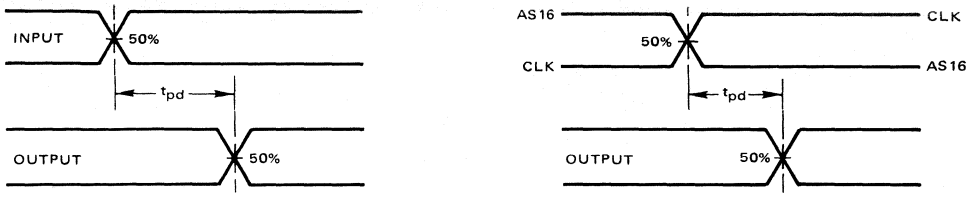
Input	Path			Output	-30°C T_A		+25°C T_A		+85°C T_A	
	Via	Mode	Function		Typ	Max	Typ	Max	Typ	Max
A Bus ØBus	ALU	Arith	Subtract	I Bus	30.0	39.0	32.0	41.0	37.0	49.0
				P_G, G_G	16.0	21.0	17.5	21.0	20.0	27.0
				C_{OUT}	18.0	22.0	19.0	23.0	22.0	28.0
				OF, ZD R-1, R4	27.0	37.0	29.5	39.0	34.0	44.0
				PC, PR	27.0	34.0	29.0	36.0	34.0	41.0
C_{IN}	ALU	Arith	Addition	I Bus	15.0	18.5	16.0	19.5	19.0	24.5
				C_{OUT}	5.0	7.0	5.5	7.5	6.0	8.5
				OF, ZD R-1, R4	12.5	16.0	13.5	17.0	15.5	19.0
				PC, PR	13.5	18.0	14.5	19.0	17.0	23.0
AS0 AS1 AS2 AS3 AS4 AS5 AS6 AS10 AS11 AS12	ALU	Arith	Subtract Accumulator	I Bus	36.0	43.0	38.5	46.5	47.0	64.0
				PG, GG	23.0	30.0	24.0	30.0	30.0	38.0
				C_{OUT}	24.0	32.0	26.0	32.0	31.5	39.0
				OF, ZD R-1, R4	33.0	43.0	36.0	46.0	47.0	60.0
				PC, PR	33.0	40.0	35.0	42.0	44.0	57.0
AS16	ALU	Arith	Subtract	I Bus	33.0	40.0	35.0	43.0	41.0	51.0
				PG, GG	20.0	25.0	21.0	26.0	25.0	32.0
				C_{OUT}	21.5	26.0	23.0	27.5	26.5	33.0
				OF, 2D R-1, R4	30.5	39.0	33.0	42.0	38.0	47.0
				PC, PR	30.5	36.0	33.0	39.0	38.0	47.0
R-1 R4	Shift	Shift Left Shift Right	–	I Bus	7.0	8.5	7.5	9.0	9.0	13.0
AS7 AS13 AS14	Shift	Shift Left Shift Right	–	I Bus	10.0	16.0	10.0	16.0	12.5	18.0
AS9 AS15	Direct	Shift ACC	–	I Bus	8.0	11.0	8.5	11.5	10.0	13.5
AS8	Direct	Enable Disable	–	I Bus	5.5	8.5	6.0	8.5	7.5	10.0
AS5 AS6	Direct	Enable Disable	–	ØBus	7.0	9.5	7.5	9.5	10.0	17.0
CLK	A Bus ALU	Arith	Subtract Accumulator	I Bus	38.5	48.0	41.0	51.0	47.0	57.0
				PG, GG	26.0	36.0	27.5	38.0	31.0	43.0
				C_{OUT}	27.5	38.0	29.0	40.0	32.5	45.0
				OF, ZD R-1, R4	37.0	41.0	39.0	43.0	44.5	49.0
				PC, PR	36.5	44.0	39.0	46.0	44.0	55.0
CLK	ALU	Arith	Add Accumulator	I Bus	34.5	45.0	36.5	47.0	42.5	58.0
CLK	Shift	AS7 = 0	Multiple Shift	I Bus	13.0	17.5	14.0	18.5	16.0	21.0
				OF, ZD R-1, R4	14.0	18.0	14.5	19.0	16.0	23.0
				PC, PR	15.0	20.0	16.0	21.0	18.0	24.0
CLK	Direct	–	Acc to I Bus	I Bus	8.0	11.0	8.5	11.0	10.0	13.0

PROPAGATION DELAYS (NANOSECONDS) (continued)

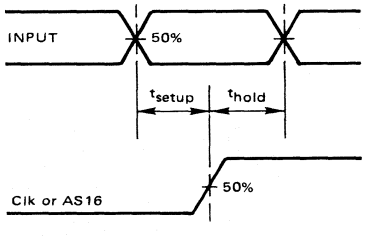
Input	Path Via	Path Mode	Path Function	Output	-30°C T_A Typ	-30°C T_A Max	+25°C T_A Typ	+25°C T_A Max	+85°C T_A Typ	+85°C T_A Max
CLK	Direct	–	Acc to ØBus	ØBus	8.5	12.0	9.0	12.0	10.0	13.0
ØBus	ALU (Mask)	Logic	Without Complement	I Bus	23.0	32.0	25.0	35.0	30.0	45.0
CLK	A Bus (Mask)	Logic	Without Complement	I Bus	33.0	42.0	35.0	43.0	40.0	47.0
CLK	A Bus (Mask)	Logic	With Complement	I Bus	34.5	44.0	37.0	47.0	42.0	55.0
CLK	A Bus (Y Mux)	Logic	Without Complement	I Bus	31.0	39.0	33.0	41.0	37.0	44.0
Output Rise and Fall Time (20% – 80%)				All	3.0	5.0	3.5	5.5	4.0	6.0

SWITCHING WAVEFORMS

PROPAGATION DELAYS



SETUP AND HOLD



TEST PROCEDURE:

a) Establish setup time with long t_{hold}.

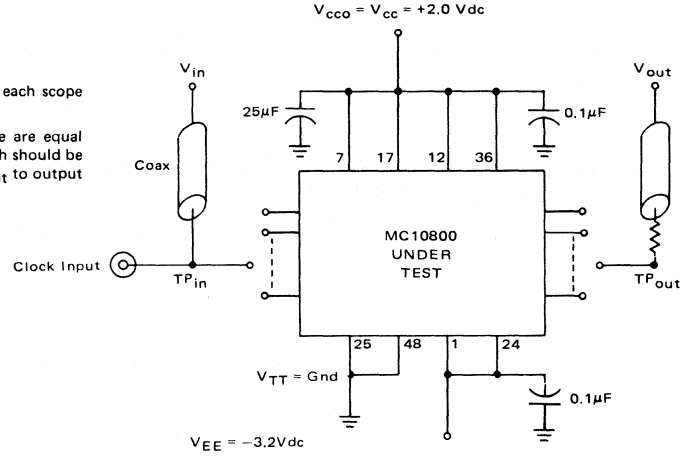
b) Keeping the leading edge of the input constant (t_{setup}) vary the trailing edge of the input to determine t_{hold}.

NOTE: t_{setup} and t_{hold} as defined are positive. Internal delays in the data path may result in a shift of the data waveform to the left, with respect to the clock, resulting in negative hold times.

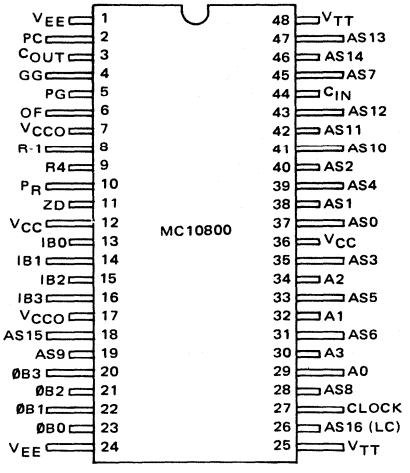
SWITCHING TIME TEST CIRCUIT

50 ohm termination to ground located in each scope channel input.

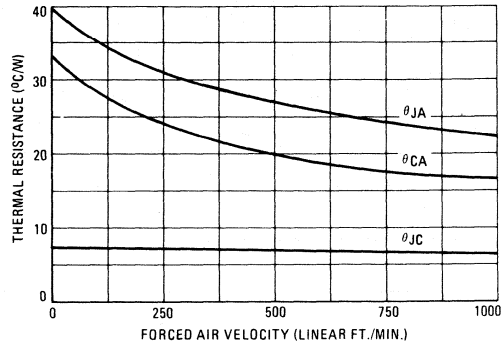
All input and output cables to the scope are equal lengths of 50 ohm coaxial cable. Wire length should be <¼ inch from TP_{in} to input pin and TP_{out} to output pin.



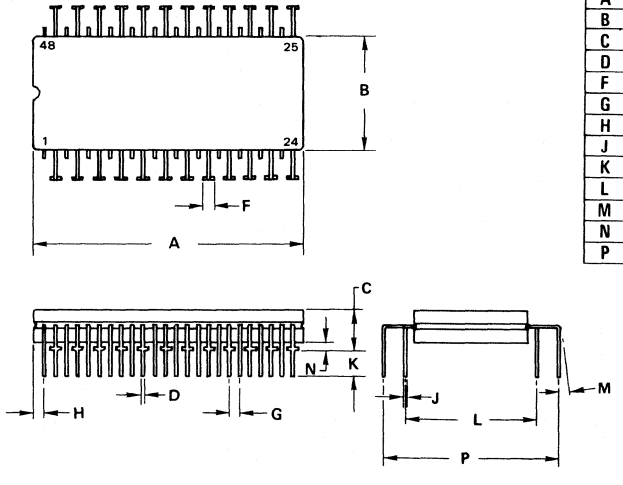
PIN ASSIGNMENT



THERMAL CHARACTERISTICS (TYPICAL)



PACKAGE DIMENSIONS



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	31.24	32.26	1.230	1.270
B	12.70	13.72	0.500	0.540
C	4.57	5.59	0.180	0.220
D	0.38	0.53	0.015	0.021
F	1.14	1.40	0.045	0.055
G	1.27 BSC		0.050 BSC	
H	1.02	1.52	0.040	0.060
J	0.20	0.30	0.008	0.012
K	2.54	3.30	0.100	0.130
L	15.24 BSC		0.600 BSC	
M	–	7°	–	7°
N	0.51	1.52	0.020	0.060
P	20.32 BSC		0.800 BSC	

Case 725-01

A socket for the QUIL package is available from ELECTRONIC MOLDING CORPORATION. (Part number 7178-295-5)

QUIL is a trademark of Motorola Inc.

MOTOROLA

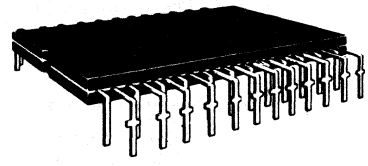
MC10801

INTRODUCTION

The MC10801 Microprogram Control Function is an LSI building block for digital processor systems. This circuit controls machine operations by generating the addresses and sequencing pattern for microprogram control storage. The MC10801 is compatible with a wide range of control memory sizes and organizations. Each part is 4 bits wide and can be connected in parallel for larger memory addresses. Maximum system flexibility is maintained with 5 separate data ports.

The Microprogram Control Function as shown in the block diagram below contains a control memory address register CRO, multipurpose registers CR1-CR3, an incrementer, a subroutine LIFO, and the associated next address, status, and bus control logic in a single MECL Bipolar LSI circuit. Nine select (CS) lines and four instruction inputs (IC) control all operations within the part.

MECL — LSI MICROPROGRAM CONTROL FUNCTION



CASE 725-01

Microprogram Control Function BLOCK DIAGRAM—MC10801

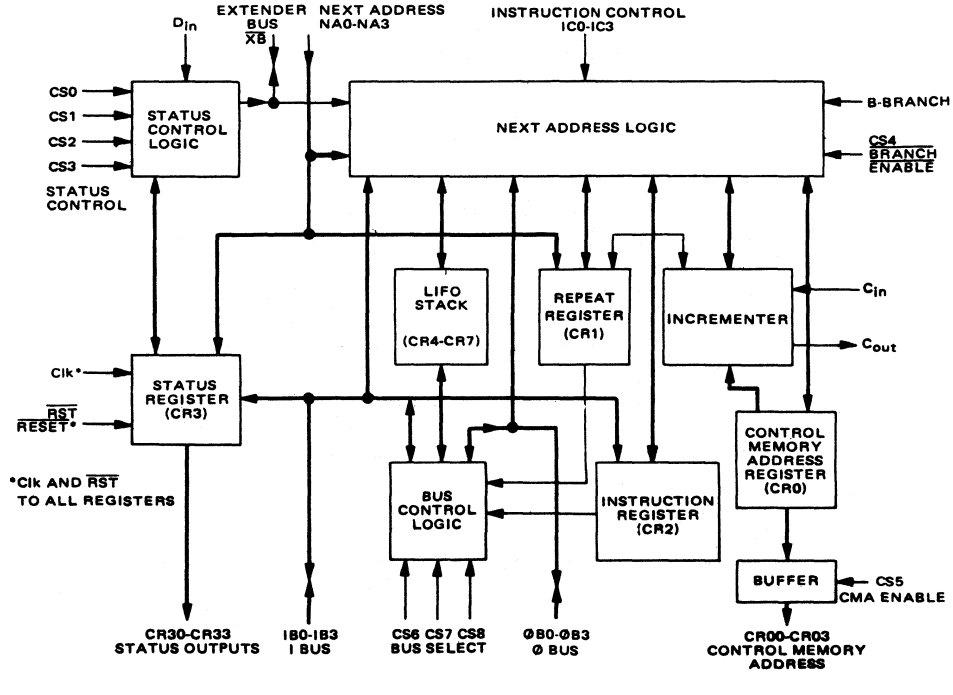
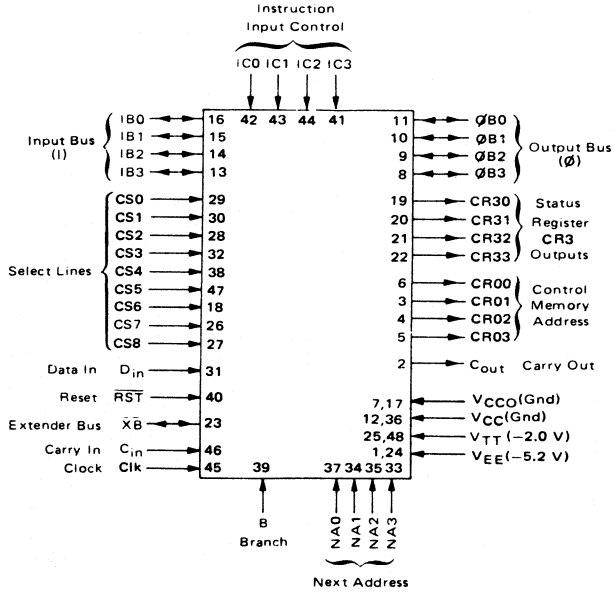


TABLE OF CONTENTS

IMPORTANT FEATURES

1. 16 microprogram sequencing instructions including:
   a. Increment
   b. Direct jumps
   c. Conditional jumps
   d. Subroutining
   e. Conditional Subrouting
2. 4-bit registers expandable with parallel MC10801 circuits.
   a. Microprogram address register – CRO
   b. Repeat register – CR1
   c. Instruction register – CR2
   d. Status register – CR3
3. Expandable 4 X 4 push-pop stack for nesting subroutine – CR4-CR7.
4. Branch inputs for conditional operations and multiway branching
5. Address masking on special instructions.
6. Repeat logic for repeating subroutines or single instructions.
7. All registers are of edge triggered master-slave design.
8. Fully compatible with the MECL 10,000 family.

INPUT/OUTPUT DIAGRAM–MC10801



ABSOLUTE MAXIMUM RATINGS (see Note 1)

RATING		SYMBOL	VALUE	UNIT
Supply Voltage (V_{CC} = 0)		V_{EE} V_{TT}	−8 to 0 −4 to 0	Vdc Vdc
Input Voltage (V_{CC} = 0)	Std Bus	V_{in} V_{in}	0 to V_{EE} Note 2	Vdc Vdc
Output Source Current	Cont Surge	I_o I_o	< 50 <100	mAdc mAdc
Storage Temp.		$T_{stg.}$	−55 to +150	°C
Junction Temp.		T_j	165	°C

NOTE: 1. Permanent device damage may occur if absolute maximum ratings are exceeded. Functional operation should be restricted to RECOMMENDED OPERATING CONDITIONS. Exposure to higher than recommended voltages for extended periods of time could affect device reliability.

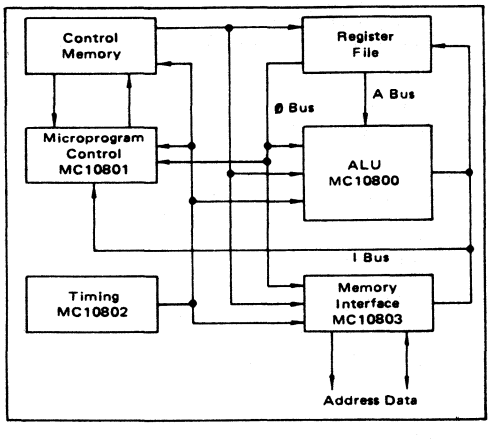
NOTE: 2. Input voltage limit is V_{CC} to −2 Volts when the bus is used as an input and the output drivers are disabled.

SYSTEM OVERVIEW

The Motorola M10800 family of LSI processor circuits has been partitioned into key building block elements as shown in Figure 1. The LSI circuits can be interconnected and programmed for a wide range of processor system applications. Combinations of the various circuits allow expansion to any required data word length or control memory size. Multiple I/O ports on each circuit provide maximum data flow flexibility. The M10800 LSI family is designed to provide functional system blocks without limiting a final system size or architecture.

The M10800 system is designed around a microprogrammed concept for greatest versatility. Microprogramming permits emulating existing machines or software, updating systems by adding more capability, or modifying systems to meet specific customer requirements. The microprogram is contained in the control memory block of Figure 1. Depending on system requirements, this memory can vary from a few hundred words to several thousand. The size and organization of this block is controlled by the system designer and is constructed with MECL PROMs such as the MCM10149, or MECL RAMs such as the MCM10144 or MCM10146.

FIGURE 1 – MICROPROGRAMMED PROCESSOR



In a microprogrammed processor the information for executing a machine function (macroinstruction) is contained within the control memory. Control memory outputs go to the Register File, ALU, and Memory Interface blocks in Figure 1 and control the specific function performed by each section of the processor. The number of control memory steps (microinstructions) required to execute a macroinstruction is determined by complexity of the instruction. For example, a simple register to register add can require only one or two microinstructions, while a more complex multiply or floating point arithmetic calculation requires several control memory words addressed in the proper sequence.

The heart of a microprogrammed system is the microprogram control logic. This block in Figure 1 holds the present control memory word address and controls the sequencing to execute processor operations. Within the Motorola M10800 family, the MC10801 Microprogram Control Function performs this important task. Each circuit is four bits wide and parallel combinations adapt to any required control memory size. A set of sixteen instructions address the MC10801 and control the sequencing of the microprogram storage. Powerful branch and subroutine instructions increase system performance and minimize the amount of control memory required to build a system. The sixteen instructions ease the burden of writing a microprogram by expressing program flow in a manner familiar to assembly language programmers.

Versatility is a key word to describe each circuit in the Motorola M10800 family. The block diagram in Figure 1 and the examples in this data sheet are intended to illustrate ways to use these LSI parts and do not restrict the designer to any particular system configuration or application.

PIN ASSIGNMENTS

Pin Designation	Pin Number	Description
IC0	42	Instruction Control Input
IC1	43	Instruction Control Input
IC2	44	Instruction Control Input
IC3	41	Instruction Control Input
IB0	16	Input Bus – LSB I/O
IB1	15	Input Bus – NLSB I/O
IB2	14	Input Bus – NMSB I/O
IB3	13	Input Bus – MSB I/O
ØB0	11	Output Bus – LSB I/O
ØB1	10	Output Bus – NLSB I/O
ØB2	9	Output Bus – NMSB I/O
ØB3	8	Output Bus – MSB I/O
NA0	37	Next Address – LSB Input
NA1	34	Next Address – NLSB Input
NA2	35	Next Address – NMSB Input
NA3	33	Next Address – MSB Input
CR00	6	Control Memory Address – LSB Output
CR01	3	Control Memory Address – NLSB Output
CR02	4	Control Memory Address – NMSB Output
CR03	5	Control Memory Address – MSB Output
CR30	19	Status Register CR3 Output
CR31	20	Status Register CR3 Output
CR32	21	Status Register CR3 Output
CR33	22	Status Register CR3 Output
CS0	29	Status Register Control – Select Input
CS1	30	Status Register Control – Select Input
CS2	28	Status Register Control – Select Input
CS3	32	Status Register Control – Select Input
CS4	38	Branch Line – Select Input
CS5	47	Control Memory Address – Enable Input
CS6	18	Ø Bus/I Bus Control – Select Input
CS7	26	Ø Bus/I Bus Control – Select Input
CS8	27	Ø Bus/I Bus Control – Select Input
C_{in}	46	Carry Input
C_{out}	2	Carry Output
D_{in}	31	Data Input to CR3
$\overline{B}$	39	Branch Input
$\overline{XB}$	23	Extender Bus
$\overline{RST}$	40	Reset Input
Clk	45	Clock Input
V_{EE}	1	–5.2 Volt Supply
V_{EE}	24	–5.2 Volt Supply
V_{TT}	25	–2.0 Volt Supply
V_{TT}	48	–2.0 Volt Supply
V_{CC}	12	Ground
V_{CC}	36	Ground
V_{CCO}	7	Ground
V_{CCO}	17	Ground

ARCHITECTURAL DESCRIPTION

The MC10801 Microprogram Control Function is composed of 8 master slave registers, CR0 through CR7, as shown in Figure 2. Additional gates, multiplexers, and a next address logic block transfer information to and from these registers. Five 4-bit data ports (CR0, CR3, NA, I Bus, and Ø Bus) are available to enter and output address information. In addition, three single line terminals (B, $\overline{XB}$, and D_{in}) provide status inputs for decisions within the part. Each of the eight registers fills an important function in the storage and generation of control memory addresses. The individual registers and data transfer paths in Figure 2 are described below.

CR0 – CONTROL MEMORY ADDRESS REGISTER

Register CR0 holds the present microprogram control memory address and its outputs are gated to package pins CR00 through CR03. In a system these outputs address the control memory storage block. The next address logic block in Figure 2 generates next address information to the CR0 register inputs. A positive clock edge loads the new control memory address into CR0 which in turn selects the next microinstruction.

NEXT ADDRESS LOGIC

The next address logic block performs 16 sequence instructions as selected by the instruction control lines IC0 through IC3 inputs. These 16 control instructions, see Table 1, determine the source of control memory address information within each MC10801. Possible sources are CR1, CR2, CR4, NA inputs, I Bus, Ø Bus, and the incrementer. During each microcycle the next address block generates a new control memory address in parallel with other processor functions, such as the ALU. Detailed information on the 16 MC10801 instructions follows in the Functional Description section of this data sheet.

FIGURE 2 – FUNCTIONAL BLOCK DIAGRAM

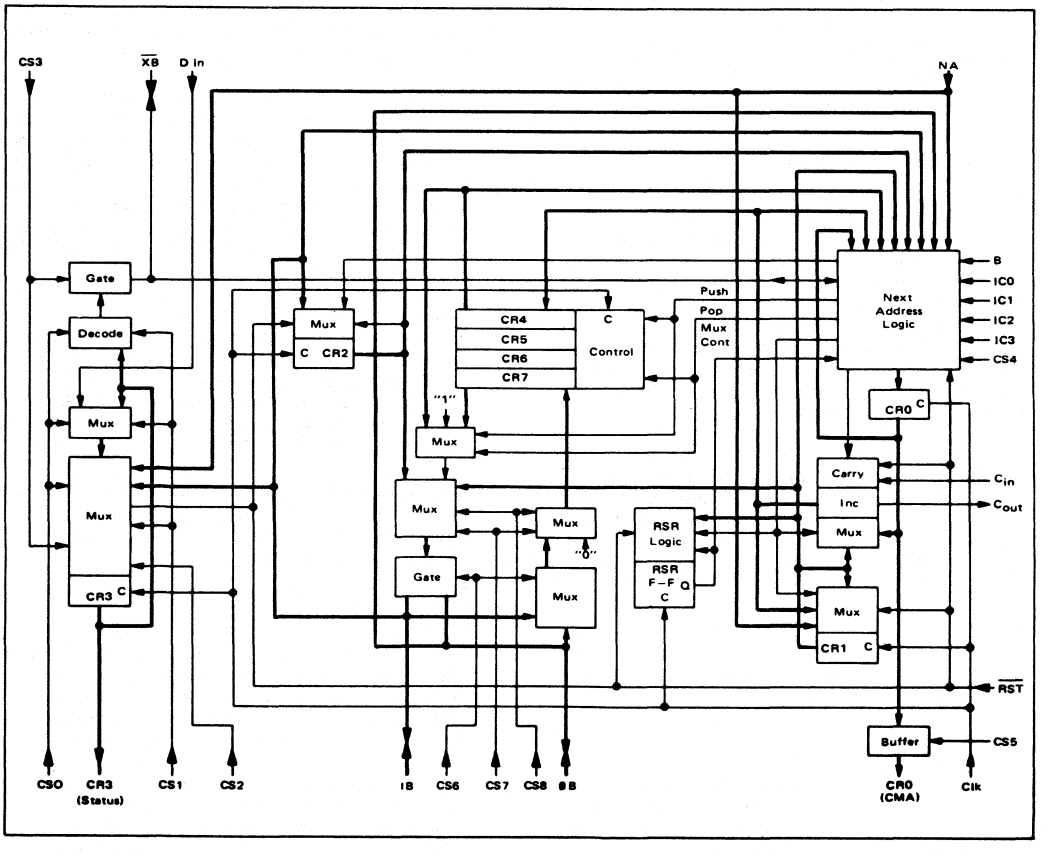


TABLE 1
MC10801 CONTROL INSTRUCTIONS

INC	– Increment
JMP	– Jump to N.A. Inputs
JIB	– Jump to I Bus
JIN	– Jump to I Bus and Load CR2
JPI	– Jump to Primary Instruction (CR2)
JEP	– Jump to External Port (Ø Bus)
JL2	– Jump to N.A. Inputs and Load CR2
JLA	– Jump to N.A. Inputs and Load Address into CR1
JSR	– Jump to Subroutine
RTN	– Return from Subroutine
RSR	– Repeat Subroutine (Load CR1 from N.A. Inputs)
RPI	– Repeat Instruction
BRC	– Branch to N.A. Inputs on Condition; otherwise Increment
BSR	– Branch to Subroutine on Condition; otherwise Increment
ROC	– Return from Subroutine on Condition; otherwise Jump to N.A. Inputs
BRM	– Branch and Modify Address with Branch Inputs (Multiway Branch)

CR1 – REPEAT REGISTER

Register CR1 is primarily designed to be an index counter for repeating single microinstructions or repeating subroutines. This repeat feature is important for multiple shift, multiply, and divide machine instructions. To perform a microprogram repeat sequence, the repeat count is first loaded into CR1 from the NA inputs with a RSR-Repeat Subroutine instruction (Table 1). Each time the selected microinstruction or subroutine is executed, CR1 is automatically incremented. Upon reaching the final repeat count the MC10801 continues to the next microprogram instruction.

A second function performed by CR1 is a control memory address save register. In this mode the present control memory address in CR0 is transferred to CR1 on a JLA-Jump and Load Address instruction (Table 1). At a later time it is possible to return to the stored address by transferring CR1 back to CR0 on a RPI-Repeat Instruction command.

The operation of CR1 is controlled by the next address logic. Possible input sources are the NA inputs, the incrementer, and CR0. CR1 outputs are routed to either CR0, to the incrementer, or to a Bus output.

CR2 – INSTRUCTION REGISTER

Register CR2 is used primarily as an instruction or op code storage register. After fetching a machine instruction, the control memory starting address can be stored in CR2. It can then be used later by transferring the contents of CR2 through the next address logic to the control memory address register CR0. As with register CR1, the operation of CR2 is controlled by the instruction inputs IC0 – IC3 and the next address logic. The I Bus is the source for CR2 and is loaded on either a JIN or JL2 instruction (Table 1). Information is transferred from CR2 to CR0 on a JPI-Jump to Primary instruction.

CR2 is not limited to an instruction register and can be used anytime it is desirable to store a control memory address location for future use. For example, CR2 can store an interrupt vector which may be loaded into CR0 as needed.

CR3 – STATUS REGISTER

Register CR3 is normally used as a status register for storing flag conditions. This 4-bit register can be parallel loaded from either the NA or I Bus inputs. In addition, any single CR3 bit can be set or cleared from the D_{in} input. The CR3 outputs are continuously available on the CR30 to CR33 package pins. The CR3 status information may be used in conjunction with other external information for generating branch conditions.

Any single CR3 bit can be selected and gated onto the $\overline{XB}$ extender bus line. $\overline{XB}$ goes to the next address logic to control branch decisions. When MC10801's are operated in parallel, the $\overline{XB}$ line is common to every part. Therefore, branch decisions can be made independent of which MC10801 circuit contains the selected status bit. The operation of CR3 with respect to the I Bus, NA inputs, D_{in}, and $\overline{XB}$ is controlled by select lines CS0, CS1, CS2, and CS3.

Another use for CR3 is to extend the control memory address. This is accomplished by organizing the control memory in a word-page format. The word address is contained in CR0 and the page address in CR3. With two MC10801's each page can be 256 words (8 CR0 bits) and 16 pages may be addressed with 4 CR3 bits or 256 possible pages using all 8 CR3 bits.

A third use for CR3 is to store all or part of the instruction operation code. In this manner, individual op code bits could be selected onto the $\overline{XB}$ line and tested for secondary decode decisions.

CR4 – CR7 LIFO STACK

Registers CR4 through CR7 are connected as a last-in-first-out (LIFO) stack for nesting subroutines within microprogram. When jumping to a subroutine, the return destination is automatically pushed onto the top of the LIFO (CR4). When returning from subroutine, CR4 is loaded into the control memory address register CR0.

With 4 registers it is possible to nest subroutines up to 4 deep within the LIFO. If additional stack depth is required CR1 can be used as a fifth location or CR7 can be expanded to any length through the I Bus or O Bus ports to additional MECL MSI circuits.

Reading CR7 via the I Bus or Ø Bus during a push of the LIFO stack provides a means for testing when the stack is full. Logic "0" bits are normally stuffed into the bottom of the stack on a "pop" or read operation. Therefore, any information in CR7 would indicate the stack is full.

Push and pop stack operations are controlled by the IC0 – IC3 inputs and the next address logic. In addition, select lines CS6, CS7, and CS8 route information to and from the LIFO via the I Bus and Ø Bus ports.

INCREMENTER

The 4-bit incrementer is used in several of the Table 1 microprogram control instructions. One is the INC-Increment command which linearly steps through a microprogram. A second function is to increment CR1 when it is used as an index counter for repeating microinstructions or subroutines as described in the earlier CR1 section. Increment is also used with the JSR-Jump to Subroutine, BSR-Branch to Subroutine, and JLA-Jump and Load Address commands to generate the proper return address. Operation of the incrementer is controlled by the IC0 – IC3 code and the C_{in} input.

The incrementer is expanded with the carry in (C_{in}) and carry out (C_{out}) terminals when MC10801 circuits are operated in parallel. The carry out of one MC10801 is connected directly to carry in of the circuit handling the next most significant control memory address bits. Carry out of the most significant bit is not required for count operation, but it can be used to signify maximum count value at the incrementer inputs.

Carry in to the least significant MC10801 is connected to a logic "1" for the increment operation. This input is normally hard wired, but in some applications can be system controlled to override the incrementer.

RSR LOGIC AND RSR FLIP FLOP

The repeat subroutine (RSR) logic and flip flop blocks in Figure 2 provide a means for setting the MC10801 in an instruction repeat sequence as described in the previous CR1 section. The RSR flip flop is automatically set when a repeat constant is loaded into CR1 with a RSR-Repeat Subroutine instruction. It is cleared when CR1 reaches the final repeat count. By monitoring the RSR flip flop status, the MC10801 can decide when to perform microinstruction repeats. Additional details of the RSR flip flop operation are explained in the following Functional Description section.

CLK – CLOCK

All registers in the MC10801 Microprogram Control Function are composed of master-slave flip flops and must be clocked to change stored data. A common clock is routed directly to all eight registers. As is characteristic of MECL flip flops, the registers are clocked on the positive going (V_{OL} to V_{OH}) clock edge. At that time data present on the register inputs is stored in the register and is available at the register outputs. Signals on the register inputs can change at any time, with the clock input at either logic state, and not change the register outputs. The only restriction on changing register inputs is during the set up and hold time near the positive going clock edge.

$\overline{RST}$ – RESET

The $\overline{RST}$ input is held at MECL V_{OL} during normal system operation. However, by forcing this input to the MECL V_{OH} level, it is possible to reset all registers in the MC10801. Reset operates in conjunction with the clock and therefore is a synchronous reset. Reset is accomplished in the following sequence. CR0, CR1, CR2, and CR3 are reset on the first clock pulse. The LIFO is connected to the incrementer to which carry-in is inhibited. The LIFO is also forced to a push mode during reset. Therefore, a maximum of five clock pulses reset all MC10801 registers in the following sequence: CR0/CR1/CR2/CR3, CR4, CR5, CR6, CR7.

FUNCTIONAL DESCRIPTION

MICROPROGRAM SEQUENCE CONTROL INSTRUCTIONS IC0-IC3

The MC10801 generates the microprogram address sequencing from 16 control instructions which are encoded on the IC0 – IC3 inputs. Each control instruction determines the data source for the next microprogram control memory address. This next address information is then stored in register CR0 on a positive going clock signal.

The 16 sequence control instructions are each described in Table 2. Table 2 lists these instructions and shows the associated mnemonics, binary select codes, and register transfers. Several instructions require making decisions on the status of the branch (B), extender bus ($\overline{XB}$), RSR flip flop output (RSQ), or select line CS4. Both decision alternatives are given for these instructions.

INC – Increment

The increment command routes the present contents of CR0 through the incrementer, adds C_{in}, and multiplexes the result (CR0 plus C_{in}) to the CR0 register inputs. As with all control instructions, the new address is loaded on a positive clock transition. This instruction is used to linearly step through the microprogram memory. When MC10801s are operated in parallel, C_{in} of a more significant device is connected to C_{out} of the previous MC10801. The least significant C_{in} is normally left floating at a logic "1".

JMP – Jump to Next Address

The JMP command provides for an unconditional jump to another control memory address. The jump destination is directly supplied on the NA inputs, which are normally feedback from control memory. A clock transition transfers address data from the NA inputs to register CR0.

FUNCTIONAL DESCRIPTION

Four instruction control inputs, IC0 – IC3, and nine select lines, CS0 – CS8, control the flow of data within the MC10801 Microprogram Control Function. The following information describes programming these inputs to perform the various circuit functions. *All truth tables are expressed in negative logic with V_{OL} being a logic 1 and V_{OH} a logic 0.*

TABLE 2[5]

	CODE							REGISTER AND FLIP FLOP OUTPUTS [4] V_{OL} ⌐ V_{OH}				
MNEM	IC3	IC2	IC1	IC0	DESCRIPTION	RESET $\overline{RST}$	BRANCH OR REPEAT CONDITION[2]	CR0[7]	CR1	CR2	LIFO STACK CR4 – CR7[6]	RSQ[3]
X	X	X	X	X	RESET CONDITION	0	X	0	0	0	"PUSH" CR0 TO STACK	0
INC	1	1	0	0	INCREMENT	1	X	CR0 plus C_{in}	–	–	–	–
JMP	0	0	1	0	JUMP TO NEXT ADDRESS	1	X	NA	–	–	–	–
JIB	1	0	0	0	JUMP TO I BUS	1	X	IB·NA	–	–	–	–
JIN	1	0	0	1	JUMP TO I BUS & LOAD CR2	1	X	IB·NA	–	IB	–	–
JPI	1	0	1	0	JUMP TO PRIMARY INST	1	X	CR2·NA	–	–	–	–
JEP	1	1	1	0	JUMP TO EXTERNAL PORT	1	X	ØB·NA	–	–	–	–
JL2	0	0	0	1	JUMP & LOAD CR2	1	X	NA	–	IB	–	–
JLA	0	0	1	1	JUMP & LOAD ADDRESS	1	X	NA	CR0 plus C_{in}	–	–	–
JSR	0	0	0	0	JUMP TO SUBROUTINE	1	$\overline{RSQ}$+RIN·$\overline{XB}$=0	NA	–	–	"PUSH" CR0 TO STACK	–
						1	$\overline{RSQ}$+RIN·$\overline{XB}$=1	NA	–	–	"PUSH" CR0 plus C_{in}	–
RTN	1	1	1	1	RETURN FROM SUBROUTINE	1	$\overline{RSQ}$+RIN·$\overline{XB}$=0	CR4	CR1 plus C_{in}	–	"POP" STACK TO CR0	–
						1	$\overline{RSQ}$+RIN·$\overline{XB}$=1	CR4	–	–	"POP" STACK TO CR0	0
RSR	1	1	0	1	REPEAT SUBROUTINE	1	X	CR0 plus C_{in}	NA	–	–	1
RPI	1	0	1	1	REPEAT INSTRUCTION	1	$\overline{RSQ}$+RIN·$\overline{XB}$=0	–	CR1 plus C_{in}	–	–	–
						1	$\overline{RSQ}$+RIN·$\overline{XB}$=1	CR1·NA	–	–	–	0
BRC	0	1	0	1	BRANCH ON CONDITION	1	$\overline{XB}$·(CS4+$\overline{B}$)=0	NA	–	–	–	–
						1	$\overline{XB}$·(CS4+$\overline{B}$)=1	CR0 plus C_{in}	–	–	–	–
BSR	0	1	0	0	BRANCH TO SUBROUTINE	1	$\overline{XB}$·(CS4+$\overline{B}$)=0	NA	–	–	"PUSH" CR0 plus C_{in}	–
						1	$\overline{XB}$·(CS4+$\overline{B}$)=1	CR0 plus C_{in}	–	–	–	–
ROC	0	1	1	1	RETURN ON CONDITION	1	$\overline{XB}$·(CS4+$\overline{B}$)=0	CR4	–	–	"POP" STACK TO CR0	–
						1	$\overline{XB}$·(CS4+$\overline{B}$)=1	NA	–	–	–	–
BRM	0	1	1	0	BRANCH & MODIFY	1	CS4=1	NA	–	–	–	–
						1	CS4=0	CR00=NA0·B CR01=NA1·XB CR02=NA2 CR03=NA3	–	–	–	–

NOTES.
1. X = DON'T CARE STATE
   – = NO CHANGE
2. EQUATIONS APPLY AS SHOWN, WHERE:
   RIN = (CR13·CR12·CR11·CR10)
   $\overline{XB}$ = EXTERNAL EXTENDER BUS NODE (see Table 3)
   $\overline{B}$ = COMPLEMENT OF BRANCH INPUT
3. RSQ = OUTPUT OF RSR FLIP FLOP
4. ALL REGISTERS AND RSR FLIP FLOP CHANGE STATE ON V_{OL} TO V_{OH} (POSITIVE GOING) CLOCK TRANSITION
5. NEGATIVE LOGIC USED THROUGHOUT
6. TABLE 8 SHOWS LIFO STACK TRUTH TABLE
7. CR0 CHIP OUTPUTS ENABLED WHEN CS5=1

JIB – Jump to I Bus

The JIB instruction is a direct jump to address information on the I Bus port. The I Bus is normally an internal data bus in the processor and can be used to input the starting address of a microprogram instruction routine. The I Bus data is modified or "masked" with control memory feedback on the NA inputs. The next address is therefore determined by I Bus ANDed with NA inputs.

JIN – Jump to I Bus and Load CR2

The JIN command routes the I Bus ANDed with NA input to CR0 as does JIB. In addition, JIN loads unmodified I Bus data into register CR2 on the same clock edge. This information in CR2 can be used at a later point in microprogram for primary and secondary program flow modification.

JPI – Jump to Primary Instruction

The JPI command is a jump to the contents of CR2 ANDed with the NA inputs. Register CR2 is loaded with a previous JIN or JL2 instruction. The code stored in CR2 is used to start a new sequence of microinstructions or modify the present microinstruction sequence.

JEP – Jump to External Port

The JEP instruction is a direct jump to information on the Ø Bus port. The Ø Bus data is ANDed with the NA inputs (Ø Bus•NA) prior to entering register CR0. This instruction offers an additional port to enter a starting address or modify information to microprogram flow.

JL2 – Jump to NA Inputs and Load CR2

The JL2 command is a direct jump to the NA inputs and a parallel load of CR2 from the I Bus. This instruction allows CR2 to be loaded during the execution of another microinstruction. This is useful for storing an interrupt vector or a new operation address while finishing a previous microinstruction sequence.

JLA – Jump to NA Inputs and Load CR1

The JLA command is a direct jump to the NA inputs and a parallel load of CR1. CR1 is loaded with the incremented value of CR0 (CR0 plus C_{in}). The JLA instruction can be used to service an interrupt or as an additional form of subroutining.

JSR – Jump to Subroutine

The JSR instruction is an unconditional jump to subroutine. The jump address is provided by the NA inputs which are loaded into register CR0. At the same time, the present CR0 address is routed through the incrementer and "pushed" onto the LIFO stack to CR4.

The JSR command operates in two modes depending upon the status of the RSR flip flop (see Table 2).

1. Non-Repeat mode is used for normal subroutining. The RSR flip flop is clear (RSQ = 0) which causes the present CR0 address to be incremented and pushed onto the stack. That is, CR0 plus $C_{in} \rightarrow$ CR4 and the contents in registers CR4 through CR7 are "pushed down" one location. Upon a return from subroutine, the incremented address puts the control into the main program flow one location below the JSR address.

2. Repeat mode is used for multiple executions of a single subroutine. The RSR flip-flop has been previously set (RSQ = 1) by an RSR instruction.

   The incrementer is disabled and CR0 is loaded into CR4. The stack registers CR4–CR7 are pushed down as before. Upon a return from subroutine, the original JSR address is then returned to CR0 and the JSR is executed again. This repeat cycle will continue until $\overline{XB}$ signifies the final repeat count has been reached.

   For multiple MC10801 configurations, the $\overline{XB}$ line is a common connection between parallel circuits. The RSR flip-flop signifies the repeat mode and $\overline{XB}$ combines with CR1 registers to determine the final repeat count. During a JSR instruction the incrementer is controlled by the following equation:

   $$\text{INTERNAL CARRY\_IN} = C_{in} \cdot (\overline{RSQ} + (CR13 \cdot CR12 \cdot CR11 \cdot CR10) \cdot \overline{XB})$$

Additional information on the $\overline{XB}$ line is found in the following Branch Control and Applications sections.

RTN – Return from Subroutine

The RTN is an unconditional return from subroutine in which the LIFO stack is "popped" and the contents of CR4 are transferred to CR0. Up to 4 levels of nesting are possible with the on-chip stack.

The RTN instruction is used with the JSR instruction for normal subroutining or multiple executions, again dependent on the RSR flip-flop (see Table 2).

1. If RSQ = 0, a normal return is executed. The stack is "popped" and the contents of CR4 are loaded into CR0.

2. If RSQ = 1, the stack is "popped" to CR0 and CR1 is incremented. The RTN will continue in the repeat mode until CR1 is filled with all ones. The RSR flip-flop is reset when all CR1 registers reach full count. As with the JSR command $\overline{XB}$ interconnects parallel MC10801's to determine full count.

RSR – Repeat Subroutine

The RSR command initializes the RSR flip-flop and CR1 for repeating microinstructions or subroutines. During the RSR, CR0 is incremented to the next address location (CR0 plus $C_{in} \rightarrow$ CR0), CR1 is loaded from the N.A. inputs, and the RSR flip-flop is set to a logic "1".

Register CR1 determines the number of times a microinstruction or subroutine will be repeated. Used as a cycle counter, CR1 is incremented until the register contains all ones (final count). For this reason, the repeat count originally loaded into CR1 must be the *2's complement* of the desired count number.

Setting the RSR flip-flop to a logic "1" causes JSR and RTN to repeat subroutines and RPI to repeat single microinstructions.

RPI – Repeat Instruction

The RPI command is used to repeat single microinstructions. In a repeat mode (RSR flip flop set to logic 1 by an RSR instruction), RPI holds the CR0 control memory address constant and increments the CR1 repeat counter. At the final repeat count, all "1"s in CR1, the RSR flip-flop is reset to logic "0" and RPI loads the contents of CR1 ANDed with the N.A. inputs into CR0.

The RPI therefore directly jumps to the new address on the N.A. inputs after the microinstruction repeat sequence is complete. (Note that CR1 remains at all "1"s after completing the repeat sequence.)

$\overline{XB}$ is common to all parallel MC10801s to insure CR1 is full on all circuits.

When not in a repeat mode (RSR flip-flop at logic "0"), the RPI instruction becomes a direct jump to register CR1. CR1 is ANDed with the N.A. inputs and loaded into CR0. In this mode RPI is used with JLA for a single level subroutine, where the return address is: (the starting address plus C_{in}) ANDed with N.A.

BRC – Branch on Condition

The BRC instruction is a conditional jump to the N.A. inputs. The branch decision is determined by the equation:

$$\overline{XB} \cdot (CS4 + \overline{B})$$

where $\overline{XB}$ is the external Extender Bus common to all parallel circuitry and B is the branch input to any MC10801. If the branch equation equals "0", BRC executes a direct jump to N.A. inputs. If the branch equation equals a logic "1", the present control memory address in CR0 is incremented (CR0 plus $C_{in} \rightarrow$ CR0) and the program goes to the next sequential location.

Normally the test bit is applied to an MC10801 branch, B, input. For multiple chip configurations, the $\overline{XB}$ line is connected common so all MC10801 respond to the same branch signal. Select line CS4 is an enable for the B input and selects which MC10801 B input is tested for the branch decision. A selected CR3 bit may also be used for branching as described in Table 4.

BSR – Branch to Subroutine

The BSR is a conditional jump to subroutine. The branch condition is determined by the $\overline{XB}$ line and the B input as with BRC. If $\overline{XB} \cdot (CS4 + \overline{B})$ = logic "0" the BSR jumps to subroutine. The subroutine destination on the N.A. inputs is loaded into CR0, and the present address in CR0 is incremented and pushed into the LIFO stack (CR0 plus $C_{in} \rightarrow$ CR4). If the branch equation equals logic "1", the present control memory address is incremented (CR0 plus $C_{in} \rightarrow$ CR0).

Unlike JSR, the BSR command is unaffected by the RSR flip flop status. Therefore, a BSR subroutine can be nested within a JSR/RTN repeat subroutine sequence without incrementing the CR1 cycle count register. A ROC is then used to return from the BSR jump.

ROC – Return on Condition

The ROC is a conditional return from subroutine. If the branch equation $\overline{XB} \cdot (CS4 + \overline{B}) = 0$, the return is executed by popping the LIFO stack and loading CR4 into CR0. If the equation equals a logic "1", the MC10801 performs a direct jump in the subroutine by loading the N.A. inputs into CR0. ROC operates independent of the RSR flip flop and can be used with BSR to nest subroutines within a repeat sequence.

BRM – Branch and Modify

The BRM instruction is a jump to the N.A. inputs with an address modification by the B and $\overline{XB}$ inputs. The following information is loaded into CR0 with CR03 being the most significant bit in the part.

CR03 = NA3
CR02 = NA2
CR01 = NA1•XB
CR00 = NA0•B

Note that $\overline{XB}$ is inverted as a modifier. This address modification allows multiway branching where the branches are sequential locations.

CS4 overrides the branch modifiers as shown in Table 2. When multiple MC10801s are operated in parallel, CS4 can be used to disable B and $\overline{XB}$ on all but the two least significant address bits.

REPEAT AND BRANCH CONTROL B, $\overline{XB}$, CS4

The Branch (B), Extender Bus ($\overline{XB}$) and Select line CS4 control certain MC10801 instructions to make repeat or conditional jump decisions.

Branch-B and Select line–CS4:

Branch operations BRC, BSR, and ROC use the B input as a source of decision information. Parallel MC10801 circuits determine the branch status from any B input enabled by select line CS4. The selected B input is routed to the $\overline{XB}$ line which is common to all MC10801s and allows parallel circuits to operate as a unit.

A branch decision depends on the following equation: $\overline{XB} \cdot (CS4 + \overline{B})$. Select line CS4 enables the B input when held at a negative logic "0" ($MECL\ V_{OH}$). Branch then occurs if B = logic "1", and the $\overline{XB}$ line extends this branch condition to all parallel circuits. $\overline{XB}$ is a complemented signal to operate properly when wired

together using the emitter dot (negative logic AND function).

The BRM instruction is a special type of branch where the B and $\overline{XB}$ lines determine register CR0 bits as follows:

CR03 = NA3
CR02 = NA2
CR01 = NA1·XB
CR00 = NA0·B

Select line CS4 overrides this use of the branch inputs. The above CR0 inputs are maintained with CS4 = logic "0" and the 4 NA inputs are routed directly to CR0 when CS4 = logic "1". This feature is used with parallel MC10801 circuits to perform a 4-way branch with the two least significant address bits. CS4 disables branch on the more significant circuits. If two MC10801s are used with CS4 = "0" on both chips, 8 way branching is possible with the next microprogram address being NA7 NA6 NA5·$\overline{XB}$ NA4·B2 NA3 NA2 NA1·$\overline{XB}$ NA0·B1 where B1 is the branch input on the lower order chip and B2 is the branch input on the upper order chip.

Repeat operations JSR, RTN, and RPI respond to $\overline{XB}$, but not to B. The functional equation for a repeat decision is $\overline{RSQ}$ + RIN·$\overline{XB}$. Repeat operation is discussed in the preceding MC10801 instruction descriptions and the following applications section.

Extender Bus – $\overline{XB}$:

The $\overline{XB}$ line operates in several modes and can be driven from various parts of the MC10801 or from external circuitry.

The $\overline{XB}$ line is controlled by the B input to insure branch coupling between parallel circuits as described above. Status register CR3 bits can be multiplexed onto $\overline{XB}$ with select lines CS0 through CS3. To make branch decisions, the selected CR3 bit goes to all parallel MC10801s on the $\overline{XB}$ interconnection. Select lines CS0 through CS3 operate independently of the selected MC10801 IC0-IC3 control instruction and must be programmed for the branch.

Repeat register CR1 and the RSR flip-flop control $\overline{XB}$ during a JSR, RTN, or RPI instruction. If RSQ = logic "1" (the MC10801 in a repeat mode) and CR1 signifies a repeat count, $\overline{XB}$ is forced to a logic "0". $\overline{XB}$ going to all parallel MC10801 circuits, couples the cycle count information in CR1 to control the repeat sequence. During a repeat sequence CR3 status bits should be disabled from $\overline{XB}$ to avoid overriding the CR1 cycle count. In a nonrepeat mode, RSQ = logic "0", the $\overline{XB}$ line has no affect on JSR, RTN or RPI instructions.

It is possible to control or modify the $\overline{XB}$ line from an external signal. The $\overline{XB}$ pins of parallel MC10801s are emitter dotted and an external signal can be tied into this connection. The external signal would override internal MC10801 control by forcing a negative logic "0" (MECL V_{OH}) on the $\overline{XB}$ line. This feature is not required for normal MC10801 operation and would be used to produce special branch functions.

Table 3 is a listing of the $\overline{XB}$ status as controlled by the various MC10801 control sequence instructions and select lines CS0, CS1, CS3 and CS4.

TABLE 3

TRUTH TABLE FOR THE $\overline{XB}$ (EXTENDER BUS) LINE

COMMENTS	1 REPEAT FUNCTION	BRANCH DISABLE CS4	INSTRUCTION CONTROL IC3-IC0 MNEMONIC CODE	2 CS3	CS1	CS0	3 $\overline{XB}$
Branch input or repeat function cannot affect the $\overline{XB}$ line on these instructions.	X	X	$\overline{JSR+RPI+RTN+\cdots}$ $\overline{BRC+BSR+ROC}$	1	X	X	1
				0	0	0	$\overline{CR30}$
				0	0	1	$\overline{CR31}$
				0	1	0	$\overline{CR32}$
				0	1	1	$\overline{CR33}$
Branch input cannot affect the $\overline{XB}$ line when CS4=1.	X	1	BRC+BSR+ROC	1	X	X	1
				0	0	0	$\overline{CR30}$
				0	0	1	$\overline{CR31}$
				0	1	0	$\overline{CR32}$
				0	1	1	$\overline{CR33}$
The Branch input is selected onto the $\overline{XB}$ line when CS4=0 and the instruction is a BRC, BSR or ROC.	X	0	BRC+BSR+ROC	1	X	X	$\overline{B}$
				0	0	0	$\overline{B}$·$\overline{CR30}$
				0	0	1	$\overline{B}$·$\overline{CR31}$
				0	1	0	$\overline{B}$·$\overline{CR32}$
				0	1	1	$\overline{B}$·$\overline{CR33}$
If the repeat function =0, the $\overline{XB}$ line is unaffected by JSR, RPI, or RTN.	0	X	JSR+RPI+RTN	1	X	X	1
				0	0	0	$\overline{CR30}$
				0	0	1	$\overline{CR31}$
				0	1	0	$\overline{CR32}$
				0	1	1	$\overline{CR33}$
If the repeat function =1, $\overline{XB}$ is forced to 0 on a JSR, RPI or RTN.	1	X	JSR+RPI+RTN	X	X	X	0

"X" represents a Don't Care Condition

NOTES: 1. (RSQ)·($\overline{CR13}$·$\overline{CR12}$·$\overline{CR11}$·$\overline{CR10}$) = Repeat Function
2. CS3 enables a bit from CR3 to be placed on $\overline{XB}$; CS0 and CS1 select the bit from CR3.
3. The $\overline{XB}$ line can be forced to a "0" from an external chip using the negative logic "AND".

STATUS REGISTER CR3 CONTROL CS0, CS1, CS2, CS3

Register CR3 is primarily used as a storage area for microprogram status information. The contents of this register are continuously available on MC10801 package pins CR30 through CR33. Information can be loaded from the I Bus port, NA inputs, or from the single line input, D_{in}. Select lines CS0 through CS3 and the reset, $\overline{RST}$, input control all CR3 load operations. In addition, CS0, CS1, and CS3 enable CR3 bits onto the $\overline{XB}$ line as described in the preceeding section and Table 3.

CS0 and CS1 select one of the four CR3 bits to be loaded from information on the D_{in} input. This occurs with CS2 = logic "0". CS0 and CSI also select the I Bus or NA inputs for parallel loading CR3. Table 4 shows the truth table for entering information into CR3. As with all MC10801 registers, CR3 is a master-slave design which loads information on a positive going (V_{OL} to V_{OH}) clock edge.

TABLE 4

TRUTH TABLE FOR STATUS REGISTER CR3 AND $\overline{XB}$ AS A FUNCTION OF CS0-CS3

$\overline{RST}$	SELECT LINE INPUTS CS3	CS2	CS1	CS0	REGISTER CR3 OUTPUTS CR33	CR32	CR31	CR30	$\overline{XB}$
0	X	X	X	X	0	0	0	0	–
1	0	0	0	0	–	–	–	DIN	$\overline{CR30}$
1	0	0	0	1	–	–	DIN	–	$\overline{CR31}$
1	0	0	1	0	–	DIN	–	–	$\overline{CR32}$
1	0	0	1	1	DIN	–	–	–	$\overline{CR33}$
1	0	1	0	0	–	–	–	–	$\overline{CR30}$
1	0	1	0	1	–	–	–	–	$\overline{CR31}$
1	0	1	1	0	–	–	–	–	$\overline{CR32}$
1	0	1	1	1	–	–	–	–	$\overline{CR33}$
1	1	0	0	0	–	–	–	DIN	1
1	1	0	0	1	–	–	DIN	–	1
1	1	0	1	0	–	DIN	–	–	1
1	1	0	1	1	DIN	–	–	–	1
1	1	1	0	0	0	0	0	0	1
1	1	1	0	1	IB3	IB2	IB1	IB0	1
1	1	1	1	0	NA3	NA2	NA1	NA0	1
1	1	1	1	1	–	–	–	–	1

"X" represents a Don't Care Condition; "–" represents a NO CHANGE Condition

NOTES: 1. Register CR3 changes state on a V_{OL} to V_{OH} transition at the clock input.
2. The $\overline{XB}$ line can be forced to a "0" due to a branch or repeat condition. Table 3 fully describes $\overline{XB}$.

CR0 OUTPUT BUFFER ENABLE CS5

Select line CS5 provides a gating function on the CR0 control memory address outputs. A logic "1" on CS5 enables CR0 to package pins CR00 through CR03. A logic "0" on CS5 forces the buffer outputs to a logic "1" state. This negative logic 1 (MECL V_{OL}) frees the CR0 output pins and allows for an external source of control memory address information. Note that when the CR0 buffers are disabled, the CR0 information is still available for internal operation. This alternate addressing feature can be used to load writable control storage on power up or for forcing interrupt vectors and overriding normal MC10801 operation. Table 5 shows the truth table for the CS5 input.

TABLE 5

TRUTH TABLE FOR CR0 OUTPUT BUFFER

CS5	OUTPUTS CR00 – CR03
1	ENABLED
0	DISABLED

BUS CONTROL CS6, CS7, CS8

The I Bus and Ø Bus function as I/0 ports for information stored within the MC10801 internal registers. For data output, CS6, CS7, and CS8 select the proper register and enable the bus output drivers. When not used to output data the MC10801 internal bus drivers are forced to a negative logic 1 (MECL V_{OL}) to provide for I Bus and Ø Bus data input operations.

Lines CS6, CS7, and CS8 select data from registers CR1, CR2, or either end of the LIFO stack CR4 and CR7. CS6 selects either the I Bus or the Ø Bus while CS7 and CS8 control the source of output data. Registers CR1 and CR2 are directly selected. However, CR4 and CR7 selection is dependent upon IC0-IC3 control instructions involving the LIFO. CR7 can be read only during a JSR or BSR with branch LIFO push operation. Reset, $\overline{RST}$, results in a LIFO push and also enables CR7 as an output.

LIFO pop operations, as caused by a RTN or ROC with branch, forces a logic 1 state on the I Bus and Ø Bus drivers. Either port can then input information to CR7 as required to extend the stack depth with external circuits. All MC10801 control instructions not involving the LIFO enable CR4 as a possible I Bus or Ø Bus data source. Table 6 shows registers available to the I Bus and Ø Bus as output ports.

TABLE 6

SELECTING THE I BUS AND Ø BUS AS DATA OUTPUTS

INSTRUCTION CONTROL IC0 – IC3 MNEMONIC CODE	$\overline{RST}$	CS7	CS8	CS6 = 0 ØB	CS6 = 0 IB	CS6 = 1 ØB	CS6 = 1 IB
X	X	0	0	1	CR1	CR1	1
JSR+BSR·XB	X	0	1	1	CR7	CR7	1
X	0	0	1	1	CR7	CR7	1
RTN+ROC·XB	1	0	1	1	1	1	1
$\overline{JSR+RTN+(BSR+ROC)\cdot XB}$	1	0	1	1	CR4	CR4	1
X	X	1	0	1	CR2	CR2	1
X	X	1	1	1	1	1	1

X = Don't care

The bus control inputs also select either the I Bus or Ø Bus as input ports to load information into the bottom of the LIFO (CR7). Select line CS6 selects either the I Bus or Ø Bus while CS7 and CS8 in conjunction with a LIFO pop function, RTN or ROC with branch, enables these ports as inputs to CR7. Table 7 shows complete LIFO operation and selection of I Bus and Ø Bus as controlled by the IC0-IC3 control instructions and the CS6-CS8 bus control inputs.

The I Bus automatically becomes an input port to CR0 or CR2 during a JIB, JIN, or JL2 instruction. When using these instructions a logic "1" is normally selected on the I Bus drivers, see Table 6, to avoid a conflict between internal register data and the incoming I Bus information.

CARRY OUT C_{OUT}

The C_{out} line is a direct function of the C_{in} input and the CR1 or CR0 registers as shown in Table 8. Note that when RSQ = 0, C_{out} always monitors the CR0 register independent of the IC0–IC3 instruction inputs.

TABLE 7

TRUTH TABLE FOR THE 4 X 4 LIFO STACK (REGISTERS CR4-CR7)

INSTRUCTION CONTROL IC0 – IC3 MNENONIC CODE							NEXT STATE			
	$\overline{RST}$	$\overline{XB}$	RSQ	CS6	CS7	CS8	CR4	CR5	CR6	CR7
RTN + RPI	0	X	1	X	X	X	CR1	CR4	CR5	CR6
RTN + RPI	0	X	0	X	X	X	CR0	CR4	CR5	CR6
$\overline{RTN + RPI}$	0	X	X	X	X	X	CR0	CR4	CR5	CR6
JSR	1	X	0	X	X	X	CR0 Plus C_{IN}	CR4	CR5	CR6
JSR	1	1	1	X	X	X	CR0 Plus C_{IN}	CR4	CR5	CR6
JSR	1	0	1	X	X	X	CR0	CR4	CR5	CR6
BSR	1	1	X	X	X	X	–	–	–	–
BSR	1	0	X	X	X	X	CR0 Plus C_{IN}	CR4	CR5	CR6
RTN	1	X	X	X	0	0	CR5	CR6	CR7	0
RTN	1	X	X	X	1	X	CR5	CR6	CR7	0
RTN	1	X	X	0	0	1	CR5	CR6	CR7	IB
RTN	1	X	X	1	0	1	CR5	CR6	CR7	ØB
ROC	1	1	X	X	X	X	–	–	–	–
ROC	1	0	X	X	0	0	CR5	CR6	CR7	0
ROC	1	0	X	X	1	X	CR5	CR6	CR7	0
ROC	1	0	X	0	0	1	CR5	CR6	CR7	IB
ROC	1	0	X	1	0	1	CR5	CR6	CR7	ØB
$\overline{JSR+BSR+RTN+ROC}$	1	X	X	X	X	X	–	–	–	–

"X" represents a Don't Care Condition; "–" represents a NO CHANGE Condition

TABLE 8

TRUTH TABLE FOR C_{out}

INSTRUCTION CONTROL IC0 – IC3	RSQ	C_{out}
RPI + RTN	0	C_{in}·CR03·CR02·CR01·CR00
RPI + RTN	1	C_{in}·CR13·CR12·CR11·CR10
$\overline{RPI + RTN}$	X	C_{in}·CR03·CR02·CR01·CR00

APPLICATIONS INFORMATION

The MC10801 fits a wide range of system sizes and applications, and therefore, has no fixed interconnection configuration. The specific system design goals will determine the control memory size, the number of MC10801s, and the interconnection pattern. A typical small processor control section can, however, illustrate use of the MC10801. Figure 3 shows two MC10801s plus microprogram control storage for the processor. Various features are described below:

MEMORY ADDRESSING

Two MC10801s provide increment, direct jump, branch, and subroutine capability for up to 256 words of control memory. Three devices can extend this to 4K words. Control register CR0 outputs are the control memory address.

A second technique to extend memory addressing beyond 256 words is two MC10801s and word-page memory mapping. Status Register CR3 of device B extends the memory size to 16 pages of 256 words each. Increment, direct jump, branch, and subrouting are restricted to within a given page, however, the third MC10801 and Next Address feedback bits from control storage are eliminated. The page address is loaded from the I Bus or NA inputs and controlled via the Status Field.

CONTROL STORAGE

Control storage can be as large as 4K words (16 pages x 256 words) for the example shown. If writable control storage is desired, MECL RAM's (MCM 10144 or MCM 10146) are used. For PROM the MCM 10149 is used.

The word length is the sum of the various control fields existing in the control storage. The Instruction Field equals 4 bits, the Next Address Field equals 8 bits, the Status Field is up to 10 bits, etc. It is not unusual for the word size to be 40 to 80 bits or more including the RF, ALU condition code, and other processor fields.

If system cycle times permit, the word size can be decreased by control field decoding. Small PROMs such as the MCM10139 or discrete logic are used to decode the select line signals. The number of microprogram bits can be reduced, but additional delay in the feedback path is introduced.

MICROPROCESSOR SEQUENCE CONTROL

The control fields feedback from storage to the MC10801s determine the microprogram sequence. The 4-bit Instruction Field selects one of 16 control instructions to generate the next microprogram address. Instruction lines IC0 through IC3 are respectively tied in parallel so that devices A and B perform the same instruction.

The Next Address Field is 8 bits wide – four bits to the most significant device B and the other four bits to device A. The NA inputs are the source of constants, starting addresses, jump and branch vectors, subroutine vectors, and masking information. The data at the NA inputs is used by the MC 10801 and controlled by the Instruction Field and/or the Status Field.

The Status Field can be up to 10 bits wide. The number of bits can be decreased with decoding or with selective functions used in the MC10801.

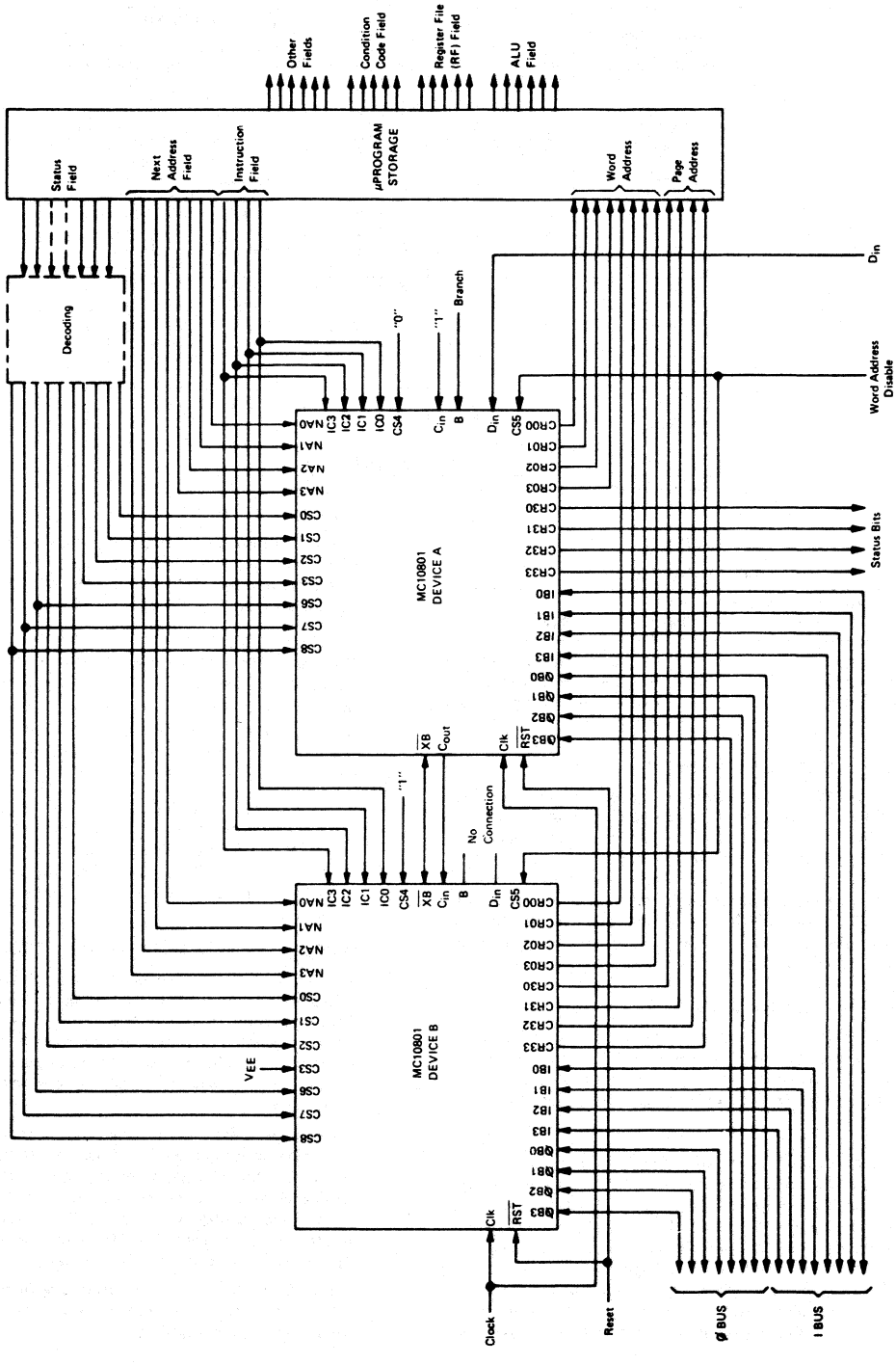


FIGURE 3 – PROGRAMMED CONTROL

Control lines CS0 through CS3 are driven independently because they manipulate register CR3 on each MC10801. Register CR3 is the Page Address register on device B, and is a status bit register on device A. Each CR3 register on the two MC10801's must be controlled independently.

Other connections to the MC10801s include:

1. C_{in} of the least significant device A is a logic "1" for increment functions. C_{out} of device A ripples to C_{in} of device B.

2. The $\overline{XB}$ lines are tied common for parallel branch functions.

3. Branch information is tested on the B input of device A. As a result, CS4 = 1 for device B to disable its branch input because the input is not used.

4. Clock and reset are tied in parallel on both devices.

5. CS5 is the CR0 output disable or the Word Address Disable. This line can be used for writable control storage functions or for interrupt functions.

6. Data can be entered into the CR3 register on a single bit basis using the D_{in} input.

I BUS AND Ø BUS

The data buses are tied to other ports of the processor. Starting addresses, interrupt vectors, and extension of the internal LIFO stack are common uses of these buses. Both the I Bus and Ø Bus are bidirectional, and are controlled by the Status Field and the Instruction Field.

SUBROUTINE & REPEAT FUNCTIONS

Subroutine and repeat operations are important functions of the MC 10801. These can each be illustrated simply.

1. Non-repeat subroutine – an example is illustrated in Figure 4. The address is limited to the word address and is listed in hexadecimal. At address 06, a JSR is executed in which address 17 → CR0, the present address plus 1 → CR4, and the stack is pushed. The subroutine begins at address 17 and ends at address 19 with an RTN. When the RTN is executed, (CR4) → CR0, the stack is popped, and the program jumps to location 07 to continue the program.

FIGURE 4 – NON-REPEAT SUBROUTINE LISTING

ADDRESS	I FIELD	NA FIELD	DESCRIPTION
06	JSR	17	06+1 → CR4; Push Stack; 17 → CR0
07	INC	X	Continue Program
–			
–			
–			
17	INC	X	Beginning of Subroutine
18	INC	X	
19	RTN	X	(CR4) → CR0; Pop Stack

2. Repeat subroutine – Figure 5 shows this example. The instruction flow is similar to the above example except that an RSR must be executed. During the RSR, CR1 is loaded with the 2's complement of 4 which is the number of times the subroutine is to be repeated. In hexadecimal notation, this is FC for 4 cycles. Also RSQ is set to 1 for repeat (in the non-repeat mode RSQ = 0).

The JSR is executed to begin the subroutine operation. During the JSR, the subroutine address 17 → CR0, the present address 06 plus C_{in} internal → CR4 and the stack is pushed. If RSQ = 0 or CR1 = FF, C_{in} internal = 1. Thus for the first 3 cycles when the JSR is executed, the present address 06 is loaded into CR4.

At the end of each subroutine cycle, an RTN is executed or (CR4) → CR0; the stack is pushed; and if RSQ = 1 and CR1 ≠ FF, then CR1 is incremented and if CR1 = FF, 0 → RSQ. In this example for the first 3 cycles, the RTN jumps to 06 (the JSR) and CR1 is incremented finally to FF.

On the final cycle, the JSR is executed with CR1 = FF and address 07 is loaded into CR4. Then, the RTN resets RSQ and jumps to location 07 to end the operation.

For this example with an 8-bit word address, the maximum number of subroutine cycles is 256.

FIGURE 5 – REPEAT SUBROUTINE LISTING

ADDRESS	I FIELD	NA FIELD	DESCRIPTION
05	RSR	FC	1111 1100 → CR1; 1 → RSQ
06	JSR	17	06 + C_{IN} Internal → CR4; Push Stack; 17 → CR0. If RSQ = 0 or CR1 = FF, then C_{IN} Internal = 1.
07	INC	X	Continue Program
–			
–			
17	INC	X	Beginning of Subroutine
18	INC	X	
19	RTN	X	(CR4) → CR0; Pop Stack; if RSQ = 1 and CR1 ≠ FF, then (CR1) +1 → CR1; if CR1 = FF, then 0 → RSQ

3. Repeat Instruction is shown in Figure 6. As in the repeat subroutine, an RSR is executed loading CR1 with = FC and RSQ is set. This sets the number of instruction cycles at 4.

   An RPI then is executed. If RSQ = 1 and CR1≠ FF, then (CR1) plus 1 → CR1 and CR0 → CR0. If CR1 = FF, RSQ is reset and (11)•(FF) → CR0.

   Thus for the first 3 cycles, CR1 is incremented and CR0 stays at the present address.

   During the fourth and final cycle (CR1 = FF from the third cycle) RSQ is reset and CR0 jumps to the Next Address value (11) AND'ed with the value of CR1 (FF) which is all logic 1s. The location 11 now continues the program.

   The maximum repeat cycle is again 256.

FIGURE 6 – REPEAT INSTRUCTION LISTING

ADDRESS	I FIELD	NA FIELD	DESCRIPTION
09	RSR	FC	1111 1100 → CR1; 1 → RSQ
0A	RPI	11	If RSQ = 1 and CR1 ≠ FF, then (CR1) + 1 → CR1, (CR0) → CR0. If CR1 = FF, then 0 → RSQ (11•FF) → CR0
11	INC		Continue Program

RECOMMENDED OPERATING CONDITIONS

PARAMETER	SYMBOL	VALUE	UNIT
Supply Voltage (V_{CC} = 0 Volts)	V_{TT} V_{EE}	−1.9 to −2.2 −4.68 to −5.72	Vdc Vdc
Operating Temp. (Functional)	T_A	−30 to +85	°C
Output Drive	–	50Ω to −2.0 Vdc	–
Maximum Clock Input Rise and Fall Time (20% to 80%)	t_r, t_f	10	ns
Minimum Clock Pulse Width	PW	5	ns

ELECTRICAL CHARACTERISTICS

Each MECL 10,000 series circuit has been designed to meet the dc specifications shown in the test table, after thermal equilibrium has been established. The circuit is in a test socket or mounted on a printed circuit board and transverse air flow greater than 500 linear fpm is maintained. Outputs are terminated through a 50-ohm resistor to −2.0 volts. Test procedures are shown for only one input, or for one set of input conditions. Other inputs tested in the same manner.

@Test Temperature	TEST VOLTAGE VALUES (Volts) V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{EE}	V_{TT}
−30°C	−0.890	−1.890	−1.205	−1.500	−5.2	−2.0
+25°C	−0.810	−1.85	−1.105	−1.475	−5.2	−2.0
+85°C	−0.700	−1.825	−1.035	−1.440	−5.2	−2.0

Characteristic	Symbol	Pin Under Test	TEST LIMITS −30°C Min	−30°C Max	+25°C Min	+25°C Typ	+25°C Max	+85°C Min	+85°C Max	Unit	VOLTAGE APPLIED TO PINS LISTED BELOW: V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{EE}	V_{TT}	(V_{CC}) Gnd
Power Supply Drain Current	I_{EE}	1, 24	–	–	–	200	–	–	–	mAdc	–	–	–	–	1, 24	25, 48	7,12,17,36
	I_{TT}	25, 48	–	–	–	270	–	–	–						1, 24	25, 48	7,12,17,36
Input Current	I_{inH}	23	–	–	–	–	45	–	–	± µAdc	23	–	–	–	1, 24	25, 48	7,12,17,36
		42	–	–	–	–	370	–	–		42						
		40	–	–	–	–	470	–	–		40						
	I_{inL}	18	–	–	0.5	–	–	–	–		–	18	–	–			
Logic "0" Output Voltage	V_{OH}	16	−1.060	−0.890	−0.960	–	−0.810	−0.890	−0.700	Vdc	*18,26,27,45	–	–	–	1, 24	25, 48	7,12,17,36
		2	−1.060	−0.890	−0.960	–	−0.810	−0.890	−0.700	Vdc	41,44,46	40,42,43	–	–	1, 24	25, 48	7,12,17,36
Logic "1"† Output Voltage	V_{OL}	16	−1.940	−1.675	−1.900	–	−1.650	−1.875	−1.615	Vdc	45	26,27,40	–	–	1, 24	25, 48	7,12,17,36
		2	−1.890	−1.675	−1.850	–	−1.650	−1.825	−1.615	Vdc	**42,43,45	40,41,44,46,47	–	–	1, 24	25, 48	7,12,17,36
Logic "0" Threshold Voltage	V_{OHA}	2	−1.080	–	−0.980	–	–	−0.910	–	Vdc	**42,43,45	40,41,44,47	46	–	1, 24	25, 48	7,12,17,36
Logic "1" Threshold Voltage	V_{OLA}	2	–	−1.655	–	–	−1.630	–	−1.595	Vdc	**42,43,45	40,41,44,47	–	46	1, 24	25, 48	7,12,17,36

†The bi-directional outputs are specified at −1.90 volts for V_{OL} min.

Preset Conditions

*PS1: Apply V_{IH} at 37, 43; V_{IL} at 40, 41, 42, 44; then clock once (⌐⌐).

**PS2: Apply V_{IH} at 41, 42, 43, 44; V_{IL} at 33, 34, 35, 37, 40, 47; then clock once (⌐⌐).

SETUP AND HOLD TIMES (NANOSECONDS AT 25°C)

Input	Setup Min	Hold Min
IC0–IC2 (1)	27	–2.0
IC0–IC3 (2, 3)	44	–8.0
NA0–NA3	28	+2.0
I Bus, Ø Bus	25	+1.0
CS0–CS3	35	–2.0
CS4 (4)	23	–2.0
B (4)	21	–1.0
C_{in}	15	+2.0
$\overline{D_{in}}$	20	+2.0
$\overline{RST}$	20	+5.0
$\overline{XB}$	28	–4.0
$\overline{XB}$ (4)	20	–2.0

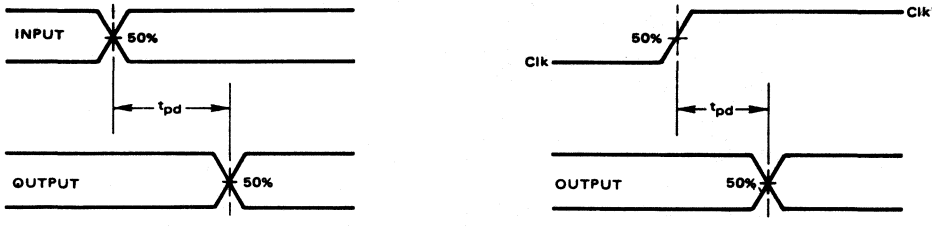
NOTES: (1) All instructions except 2 and 3 below.
(2) BSR, BRC, BRM, or ROC instruction when B · $\overline{CS4}$ = 1.
(3) BSR, BRC, BRM, ROC, JSR, RPI, or RTN instruction when RSQ = 1.
(4) BRM instruction only.

PROPAGATION DELAY TIMES (NANOSECONDS)

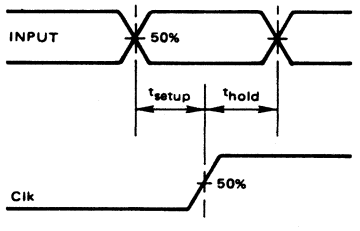
Input	Output	–30°C Typ	–30°C Max	+25°C Typ	+25°C Max	+85°C Typ	+85°C Max
Clock	CR0, CR3	11.1	16.0	11.2	16.0	12.0	17.0
Clock	IB, ØB	16.3	30.0	16.8	30.0	21.2	32.0
Clock	$\overline{XB}$	15.7	20.0	16.1	21.0	18.2	23.0
Clock	C_{out}	13.5	22.0	14.6	23.0	16.3	24.0
C_{in}	C_{out}	3.10	9.00	2.80	7.00	3.40	8.00
IC0–IC3	$\overline{ØB}$	22.7	32.0	23.4	33.0	28.9	38.0
IC0–IC3	$\overline{XB}$	14.9	20.0	15.9	21.0	19.1	24.0
IC0–IC3	C_{out}	17.4	26.0	17.4	26.0	20.9	27.0
CS7, CS8	IB, ØB	14.8	22.0	16.0	24.0	17.6	26.0
CS0–CS4, B	$\overline{XB}$	11.8	17.0	12.4	18.0	15.3	20.0
CS6	IB, ØB	7.00	11.0	6.80	11.0	7.70	12.0
CS5	CR0	5.20	10.0	5.30	10.0	6.10	11.0
$\overline{XB}$	IB, ØB	21.3	29.0	22.2	31.0	24.6	36.0
$\overline{RST}$	IB, ØB	18.8	26.0	19.6	28.0	22.9	31.0
TR, TF	All	6.00	11.0	6.50	11.0	8.30	12.0

SWITCHING WAVEFORMS

PROPAGATION DELAYS



SETUP AND HOLD



TEST PROCEDURE:

a) Establish setup time with long t_{hold}.

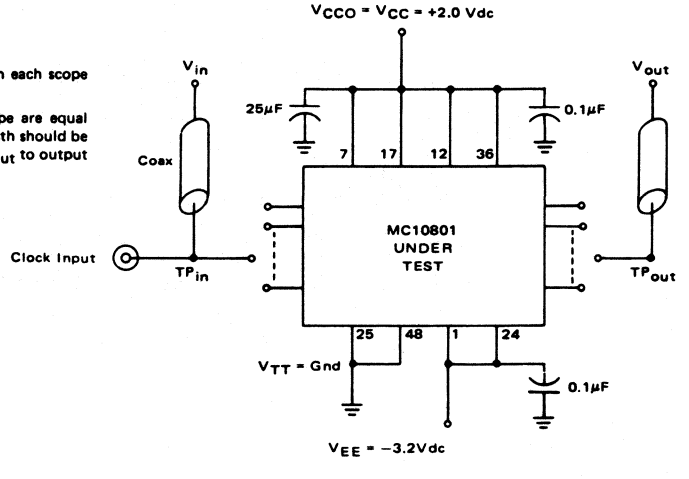
b) Keeping the leading edge of the input constant (t_{setup}) vary the trailing edge of the input to determine t_{hold}.

NOTE: t_{setup} and t_{hold} as defined are positive. Internal delays in the data path may result in a shift of the data waveform to the left, with respect to the clock, resulting in negative hold times.

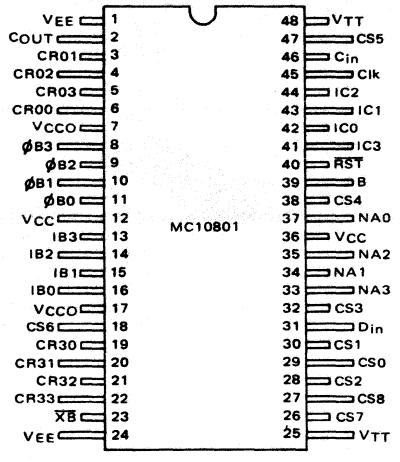
SWITCHING TIME TEST CIRCUIT

50 ohm termination to ground located in each scope channel input.

All input and output cables to the scope are equal lengths of 50 ohm coaxial cable. Wire length should be <¼ inch from TP_{in} to input pin and TP_{out} to output pin.



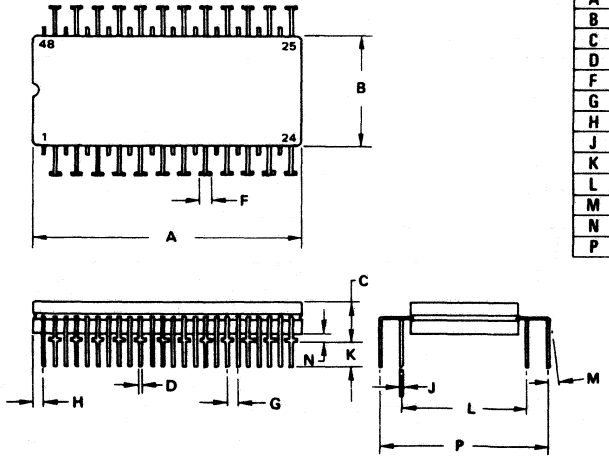
PIN ASSIGNMENT



THERMAL CHARACTERISTICS
(TYPICAL)
@ 500 Linear Ft. Air Flow

θ_{JA} = 26.5° C/W
θ_{JC} = 7° C/W

PACKAGE DIMENSIONS



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	31.24	32.26	1.230	1.270
B	12.70	13.72	0.500	0.540
C	4.57	5.59	0.180	0.220
D	0.38	0.53	0.015	0.021
F	1.14	1.40	0.045	0.055
G	1.27 BSC		0.050 BSC	
H	1.02	1.52	0.040	0.060
J	0.20	0.30	0.008	0.012
K	2.54	3.30	0.100	0.130
L	15.24 BSC		0.600 BSC	
M	–	7°	–	7°
N	0.51	1.52	0.020	0.060
P	20.32 BSC		0.800 BSC	

Case 725-01

A socket for the QUIL package is available from ELECTRONIC MOLDING CORPORATION. (Part number 7178-295-5)

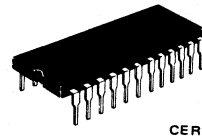
QUIL is a trademark of Motorola Inc.

MOTOROLA

MC10802

Advance Information

MECL – LSI
TIMING
FUNCTION



L SUFFIX
CERAMIC PACKAGE
CASE 623

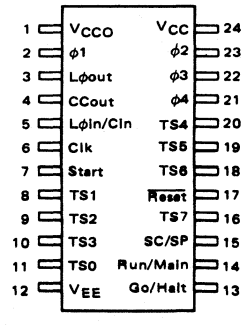
INTRODUCTION

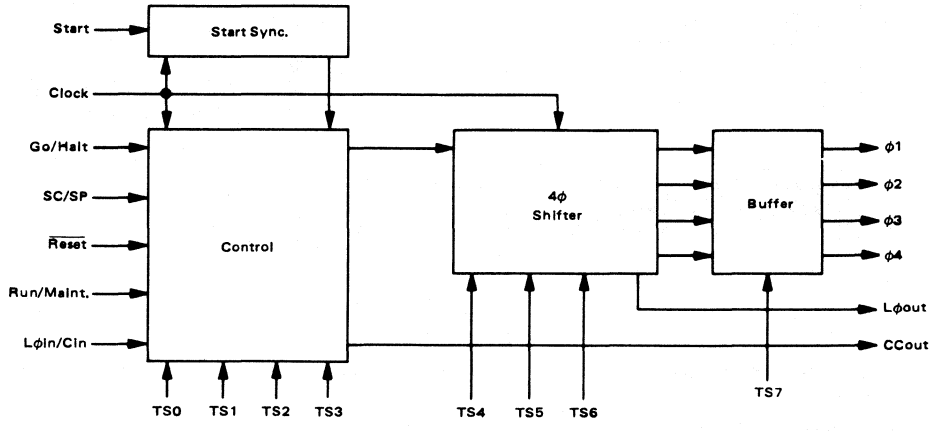
The MC10802 Timing Function is an LSI building block for digital processor systems. This circuit contains the logic and control lines to generate system clock phases and provides for start, stop, and diagnostic operations. Each part is four bits wide and can be connected in series for greater than four phase clock systems.

The Timing Function as shown in the block diagram below is composed of a four phase shifter circuit with buffered outputs. Fifteen input lines combine with Control and Start Sync logic to control all operations within the part.

FEATURES

- Programmable Number of Phases
- Selectable Double-Width Phases Duration
- Start Signal Synchronizer
- Single Cycle Stepping
- Single Phase Stepping
- Asychronous Master Reset
- Cascadable
- Fully Compatible with the MECL 10,000 Family





This is advance information on a new introduction and specifications are subject to change without notice.

FUNCTIONAL DESCRIPTION

The MC10802 is a clock phase generator operating from a single clock source. The number of clock phases (up to four), phase duration, and operation modes are all programmable. Additionally, multiple devices can be cascaded for more than four clock phases. The following description details device operation.

CLOCK PHASE OUTPUTS

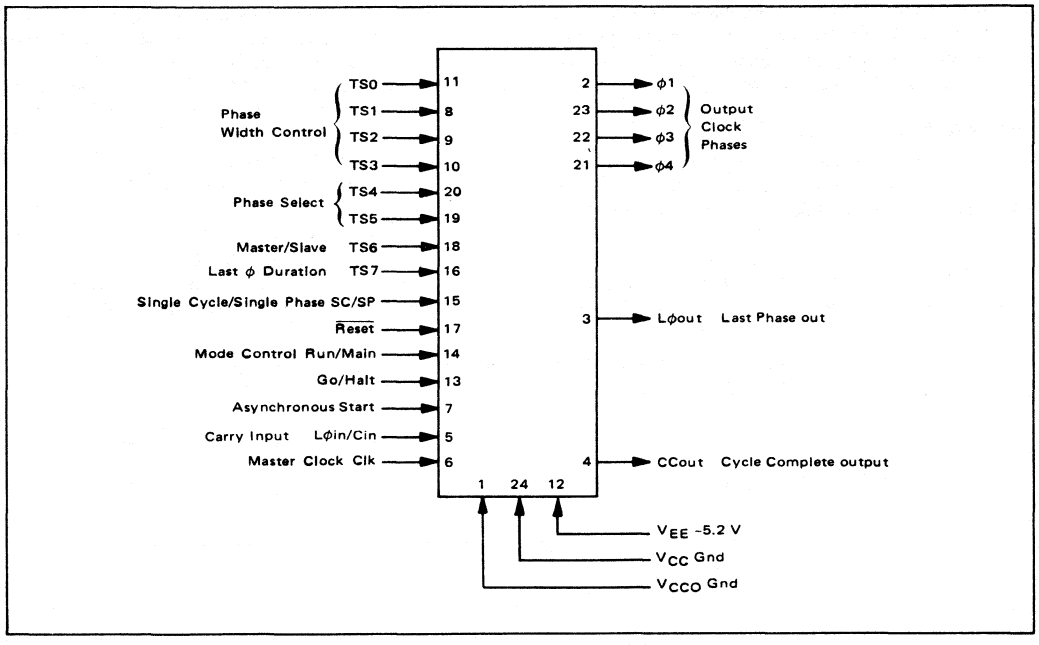
Up to four clock phases are available per circuit. The phase number is programmed by select lines TS4 and TS5 (Table 1). Clock phase outputs **are normally high** (V_{OH}) **and go low while active** (See Figures 5 and 6). The end of each phase is a low to high transition as required to clock MECL master-slave storage devices.

TABLE 1 – PHASE SELECTION

TS4	TS5	Phase Number
H	H	One Phase
L	H	Two Phases
H	L	Three Phases
L	L	Four Phases

The "One Phase" selection is used only with cascaded circuits, for example, five phases with two parts or nine phases with three parts. If used with a single circuit, the MC10802 stops with $\phi 1$ held at a low logic level (V_{OL}). A single phase clock is generated by programming the circuit to two phases, giving the desired clock and its complement on the $\phi 1$ and $\phi 2$ outputs.

INPUT/OUTPUT DIAGRAM-MC10802



ABSOLUTE MAXIMUM RATINGS (See Note)

Rating		Symbol	Value	Unit
Supply Voltage (V_{CC} = 0)		V_{EE}	-8 to 0	Vdc
Input Voltage (V_{CC} = 0)		V_{in}	0 to V_{EE}	Vdc
Output Source Current	Cont	I_o	< 50	mAdc
	Surge	I_o	<100	mAdc
Storage Temp.		$T_{stg.}$	-55 to +150	°C
Junction Temp.		T_j	165	°C

NOTE: Permanent device damage may occur if absolute maximum ratings are exceeded. Functional operation should be restricted to RECOMMENDED OPERATING CONDITIONS. Exposure to higher than recommended voltages for extended periods of time could affect device reliability.

Phase select lines TS4 and TS5 are fixed programmed for most systems; however, the number of selected phases can be dynamically changed if desired. The programming change of TS4 and TS5 must occur on or before the last phase of the smallest of the two selected numbers; that is, changing $\phi_n \rightarrow \phi_n + x$ or $\phi_n + x \rightarrow \phi_n$ must be done during or before ϕ_n and meet circuit setup and hold times. For example, changing from a four phase selection to a two phase selection must occur on or before $\phi 2$.

PULSE WIDTH DURATION

Each clock phase is normally one master clock period in duration. However, any or all selected phases can be extended to double duration. Select lines TS0 through TS3 independently control phase duration of the corresponding output (Table 2).

Similar to phase number selection, phase duration can be dynamically programmed. The associated select line must be stable for the positive going master clock edge that triggers the desired phase. Figure 1 shows an example of stretching $\phi 2$ with associated signals.

OPERATIONAL CONTROL

Control line inputs determine the MC10802 operating mode. System start, stop, and diagnostics can all be controlled via these MC10802 inputs as shown in Table 3.

1. $\overline{\text{RESET}}$ – is an asynchronous input overriding other control inputs. $\overline{\text{Reset}}$ drives all phase outputs to a high logic state. It also resets the circuit so $\phi 1$ will be the first phase after the circuit is initialized. $\overline{\text{Reset}}$ is active with a high logic level on the input. **$\overline{\text{Reset}}$ must be supplied on power up.**

2. MAINTENANCE MODE – is selected with the R/M input in the high logic state. The maintenance mode has four operating patterns as controlled by the G/H and SC/SP inputs (see Table 3). Each maintenance mode operation is initialized by a low, V_L, to high V_H, transition on the START input. This requirement for a start signal is normally used for diagnostic purposes or "front panel" interface. A special synchronizer circuit internal to the MC10802 allows the start signal to be asynchronous to the master clock.

   a. STOP AT END OF CYCLE – is a continuous cycle operation controlled by G/H=L and SC/SP=L. Initialized by a start signal, the MC10802 continues cycling until halted with the G/H input.

TABLE 2 - PHASE DURATION

Select Line		Phase	Duration
TS0	H	1	Single
	L	1	Double
TS1	H	2	Single
	L	2	Double
TS2	H	3	Single
	L	3	Double
TS3	H	4	Single
	L	4	Double

FIGURE 1 – STRETCHING PHASE TWO WITH ASSOCIATED SIGNALS

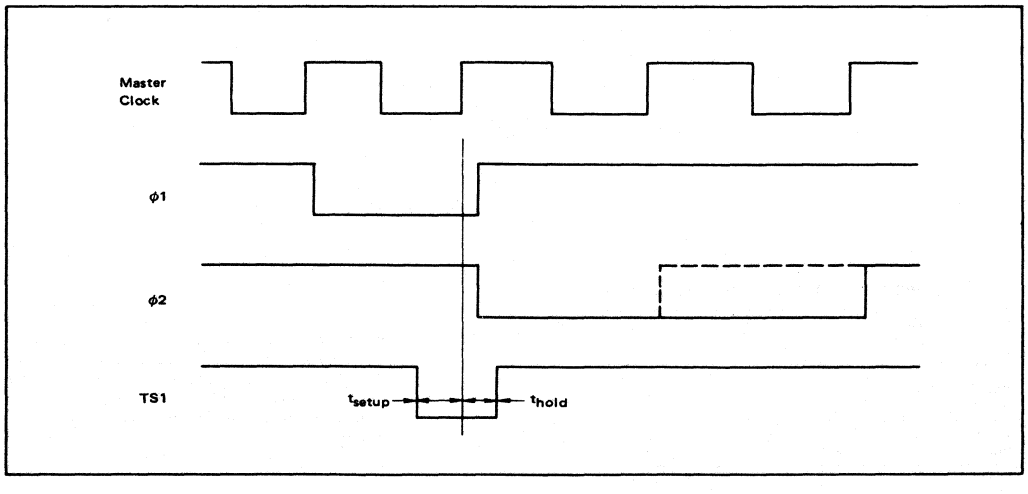


TABLE 3 – OPERATIONAL CONTROL

$\overline{\text{Reset}}$	R/M	G/H	SC/SP	Mode	Operation		Source
H	X	X	X	Reset	X		X
L	H	H	H	Maintenance	Halt	Single Phase	Start ↓
		H	L			Single Cycle	Start ↓
		L	H		Go	Stop on Phase	Initiate-Start ↓ End-G/H ↓
		L	L			Stop at End of Cycle	Initiate-Start ↓ End-G/H ↓
L	L	H	H	Run	Halt	Single Phase	Start ↓
		H	L			Single Cycle	Start ↓
		L	H		Go	Stop on Phase	Initiate-G/H ↘ End-G/H ↓
		L	L			Stop at End of Cycle	Initiate-G/H ↘ End-G/H ↓

X = don't care

When halted, the circuit will complete the programmed phase count and stop. The G/H can go high to stop the part any time during the last set of output phase signals until the minimum setup time before the master clock rising edge that terminates the last phase. The circuit is initialized afterward in this mode by returning G/H to a low state and supplying another start input signal.

b. STOP ON PHASE – is also a continous cycle operation. It is controlled by G/H = L and SC/SP = H. As with the previous Stop at End of Cycle, the MC10802 is initialized with a start signal and continues cycling until halted by taking the G/H input high. However, in this operating mode the circuit stops on the phase during which G/H goes high, again observing minimum set up times.

c. SINGLE CYCLE – is a diagnostic function selected by G/H = H and SC/SP = L. Initially all phase outputs are stationary at a high state. Upon receiving a start signal, the MC10802 goes through one complete cycle of phases, terminating after the last programmed phase (see Figure 5). Each following start signal sequences the circuit through one complete cycle.

d. SINGLE PHASE – is also a diagnostic function and is selected by G/H = H and SC/SP = H. A start input signal advances the MC10802 one output phase from its present position. Several start signals cause the circuit to cycle through the programmed number of phases, moving forward one phase for each start.

If a Single Phase or Stop on Phase operation stops the MC10802 on a phase other than the final selected phase, future operations begin from that point. For example, if a four phase pattern is stopped on $\phi 2$, any new operation other than reset starts with $\phi 3$.

3. RUN MODE – is selected by the R/M input in the low state. The four operations in the run mode differ from the maintenance mode only in the initiation source for Stop on Phase and Stop at End of Cycle (see Table 3). These operations are initiated with the G/H input going to a low state and do not require a start input signal. Stop on Phase and Stop at End of Cycle are terminated by the G/H line going high as with the maintenance mode. Select line changes should be synchronized with the master clock to maintain proper set-up and hold time. Single Cycle and Single Phase operations are exactly the same as in the maintenance mode and require a start signal to initialize the operation.

TS7 controls the output phase duration during a Single Phase or a Stop on Phase Operation. If TS7 is at a logic low state, the phase output returns to a high state after its normal pulse width low. If TS7 is a logic high, the final phase of Stop on Phase or the Single Phase output remains low (Figure 6). Keeping an output low is used for diagnostics during partial computer cycle to trace data, delay clocking of registers, enable buses, etc. If a phase output has been held low, returning TS7 to a logic low causes the phase output to be clocked high with the Master Clock. Table 4 gives the Truth Table for TS7.

The CYCLE COMPLETE OUTPUT (CC) shows the operational status of the MC10802. This output indicates when the circuit is not running and is in the initial condition. Cycle complete goes high as a result of (1) a reset, (2) a completed Stop at End of Cycle operation, or (3) a completed Single Cycle operation. Cycle complete is not given during a running condition as one cycle immediately follows another. Table 5 shows Cycle Complete output status with Figures 5 and 6 giving timing examples.

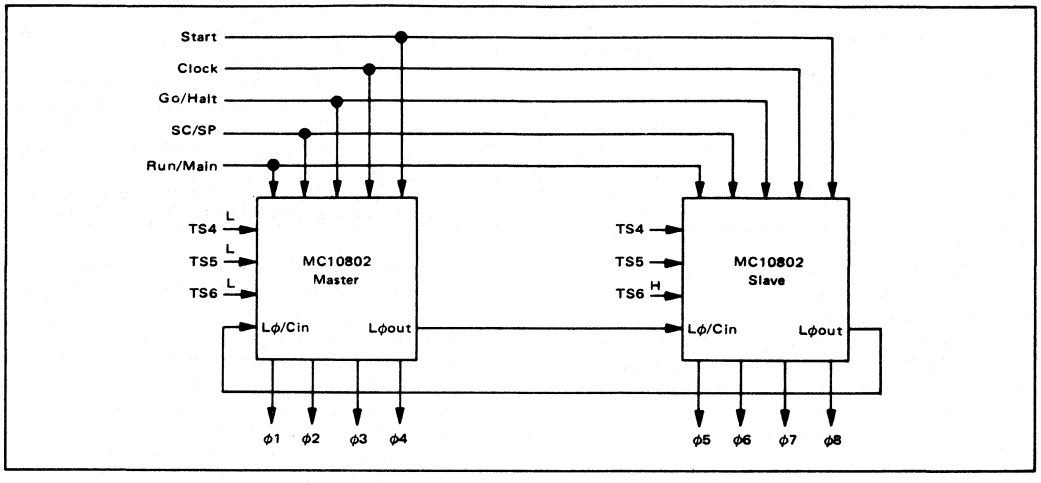
TABLE 4 – TS7 TRUTH TABLE

TS7	Function
L	Normal Phase duration
H	Wide phase duration

TABLE 5 – CYCLE COMPLETE

CC Output	Status
H	Complete
L	Running

FIGURE 2 – CASCADING CIRCUITS



CASCADING DEVICES

Two or more MC10802's can be cascaded for greater than four (4) phases. Figure 2 shows an example of two devices.

Select line TS6 is used to establish a "master-slave" relationship between the cascaded circuits. Only the first device is related as the master stage, all others are programmed as slave stages. A single device is treated as a master stage.

Additionally, the Last Phase Out (Lϕout) and Carry In (Lϕ/Cin) lines are connected for coupling between devices. In the Figure 2 example, the master stage Lϕout is connected to the slave stage Lϕ/Cin, and similarly, the slave stage Lϕout is returned to the master stage Lϕ/Cin. This daisy chain technique can be extended to multiple devices in which the Lϕout of the last device is always returned to the master. **Single device operation requires that Lϕout be returned to Lϕ/Cin of the same circuit.**

Last Phase Out is coincidental with the last programmed phase as selected by TS4 and TS5, Table 1.

TABLE 6 – SELECT LINE TS6

TS6	Function
L	Master Stage (1st)
H	Slave Stage (Other)

If the last phase is double width as selected by TS0 through TS3, Lϕout is present during the last full phase time independent of TS7 (See Figure 6), only for the second half and is one clock period in duration. When using SINGLE PHASE operation, Lϕout is present.

For normal system operation, clock phase programmobility should be limited to the last MC10802 in a chain. The Figure 2 example shows a range of 5 to 8 phases. Programming the first stage results in missing phase outputs; such as ϕ1, ϕ2, ϕ5, ϕ6, etc., if the master device is programmed as a 2ϕ example. This feature can be used for programmable time delay applications (Figure 2 has a total range of 2 through 8).

CIRCUIT OPERATION WAVEFORMS

MC10802 operation can be illustrated by waveforms showing the various timing modes. The following waveforms are based on the circuit being connected as in Figure 3. These examples are designed to show circuit operation and do not represent any particular system clocking patterns.

The Start input goes directly to a start synchronizer circuit built from two master slave flip-flops. This gives the MC10802 an internal start signal one clock period wide that is synchronized to the main clock as shown in Figure 4. Internal start normally occurs on the first positive going clock edge after the start input, but will occasionally fall on the second positive edge due to set up times required for the internal flip flops.

An asynchronous start signal must be high for at least one master clock period to insure the start input is present during a positive going master clock edge. There is no maximum duration for the start input. A start signal synchronized to the master clock must follow the set up and hold times around the positive going master clock edge.

FIGURE 3 – INTERCONNECTIONS

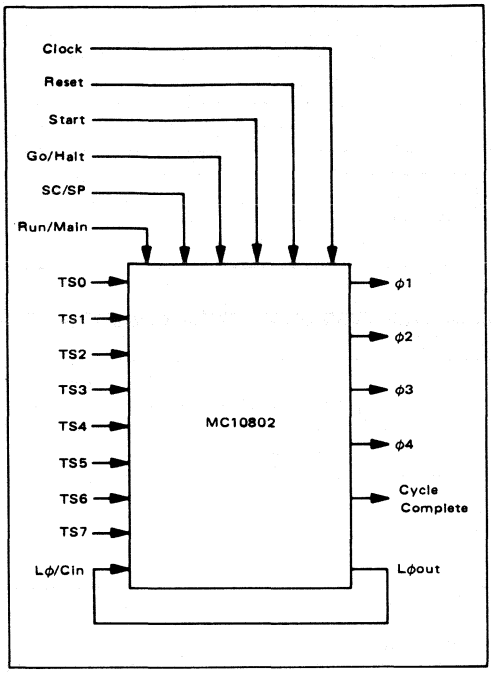


Figure 5 shows operation in the MAINTENANCE mode requiring a start signal to initialize any operation. The circuit has been reset at the start of the sequence, insuring phase one occurs first. With G/H in the GO mode a start input causes the MC10802 to begin continuous operation generating all four phases in this example. A HALT condition on the G/H input causes the circuit to complete the present cycle, SC/SP in the SINGLE CYCLE position, and stop as signified by Cycle Complete. With G/H remaining in the HALT position a start signal causes the MC10802 to go through one complete cycle of four phases and then stop. Both the continuous run or single cycle step can be repeated in any sequence.

Figure 6 shows operation in the **Run** mode which starts operation on the first positive going clock edge after G/H is in the **Go** position. The Circuit will continue to cycle with three phases and a double duration on the third phase as programmed in this example. When G/H is moved to **Halt** the circuit will stop on phase, Φ2 in this illustration, because SC/SP is in the **Single Phase** position. A **Go** input would cause the MC10802 to being operation on the next positive going clock with phase two changing to phase three, etc.

With the circuit at **Halt** and in **Single Phase**, a start signal caused the part to advance one phase as shown in the Figure. When stepping phase-to-phase, it is possible to have either a wide phase pulse or a normal width pulse as controlled by TS7 and shown by the dotted line in Figure 6. Last phase out is present during the second half of a double width last phase and is present continually when halted on the last phase as discussed in the earlier Cascading Circuits section. Finally, in the **Single Phase** mode there is no Cycle Complete signal. The MC10802 operates as if it were continually running with **Halt** being an extended duration of the halted phase.

Examples given in Figures 5 and 6 are only two of the various operating modes. It is possible to have any combination of **Run/Maintenance, Single Cycle/Single Phase,** number of phases and phase durations to meet system requirements.

FIGURE 4 – START SYNCHRONIZER

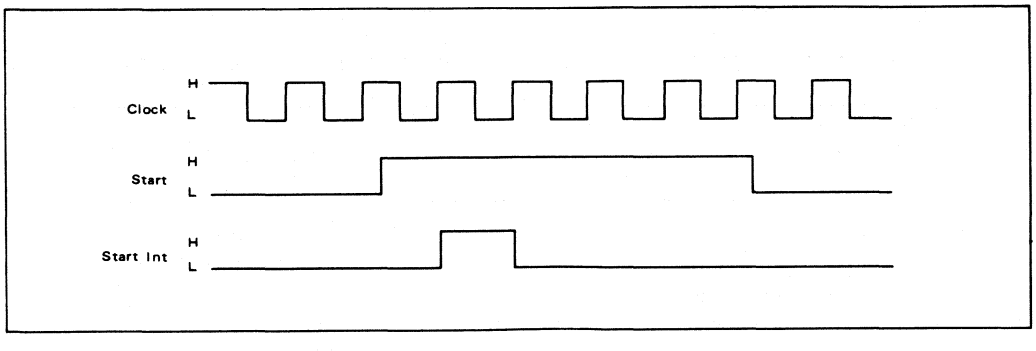


FIGURE 5 – MAINTENANCE MODE WAVEFORMS

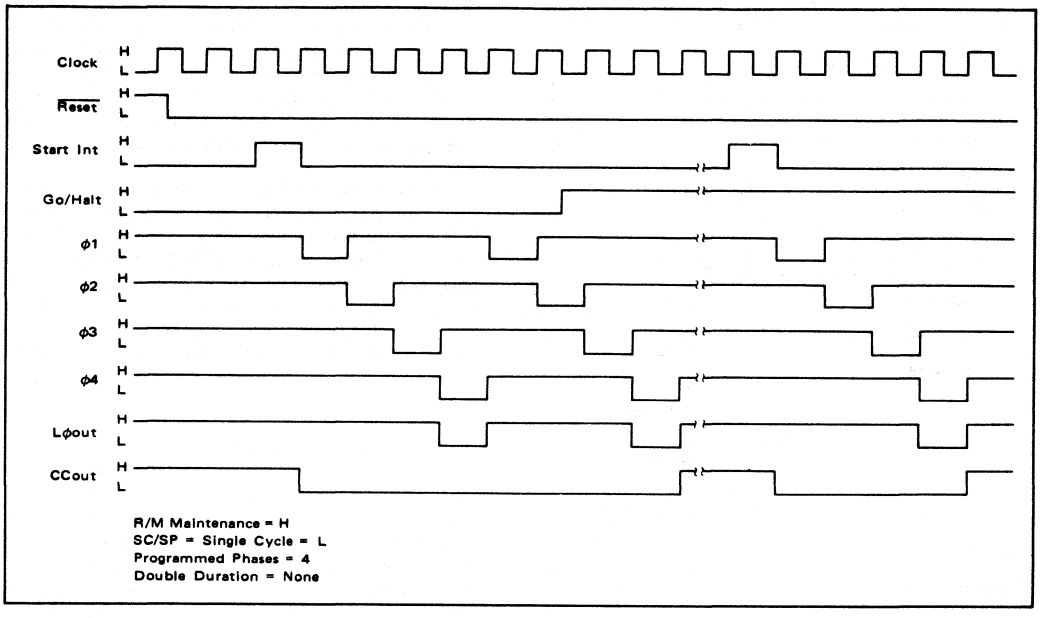
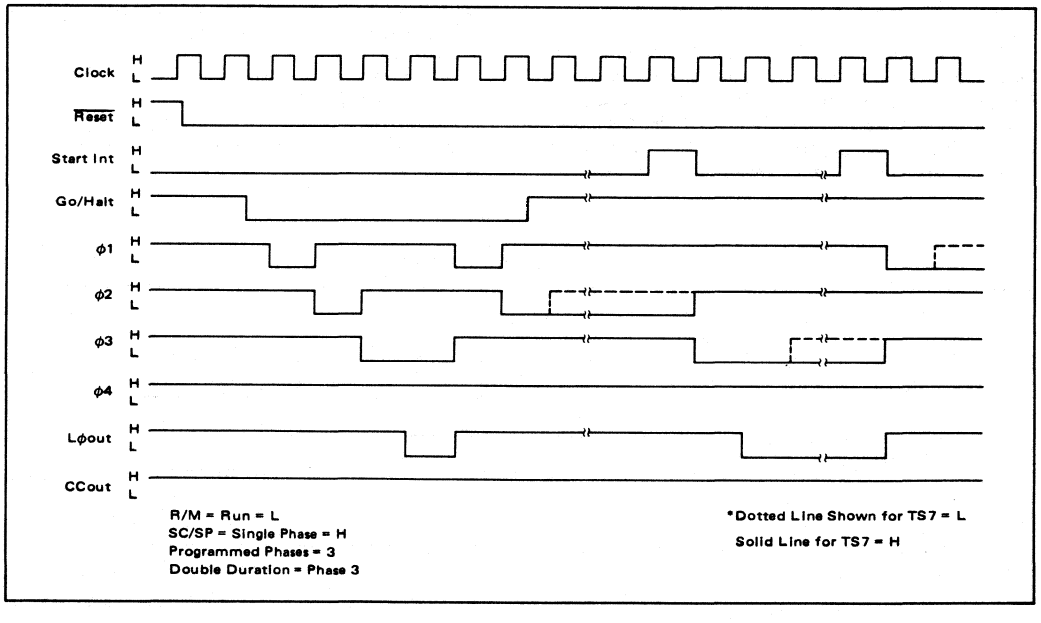


FIGURE 6 – RUN MODE WAVEFORMS



PROPAGATION DELAYS (Nanoseconds at 25°C)

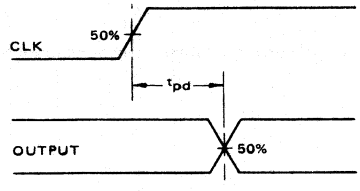
Path	Typ
Master Clock to Phase Outputs	7.5
Master Clock to Lϕout	9.0
Master Clock to CCout	6.4
$\overline{\text{Reset}}$ to Phase Outputs	6.8
$\overline{\text{Reset}}$ to Lϕout	7.5
Maximum Master Clock Frequency	36 MHz

TYPICAL SETUP AND HOLD TIMES (Nanoseconds at 25°C)

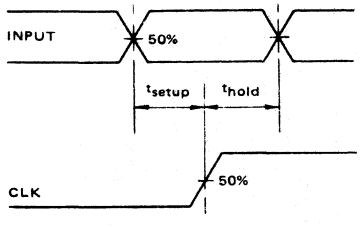
	t_{setup}	t_{hold}
TS4 to Master Clock	8.0	-6.0
TS5 to Master Clock	8.0	-6.0
Go/Halt to Master Clock	-1.5	0.5
Start to Master Clock	-5.0	5.8
Lϕ/Cin to Master Clock	3.0	10

SWITCHING WAVEFORMS

PROPAGATION DELAYS



SETUP AND HOLD



TEST PROCEDURE:

a) Establish setup time with long t_{hold}.

b) Keeping the leading edge of the input constant (t_{setup}) vary the trailing edge of the input to determine t_{hold}.

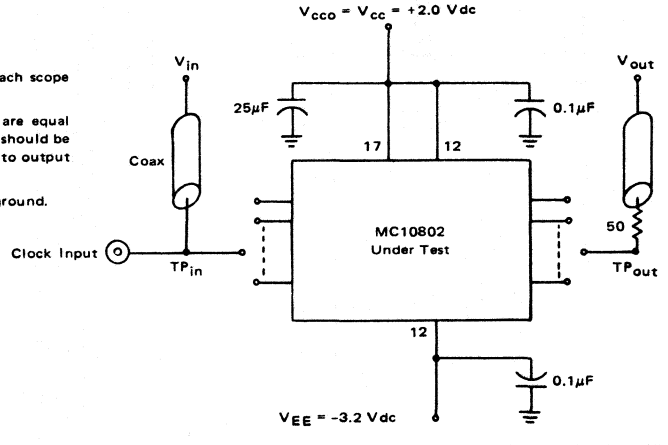
NOTE: t_{setup} and t_{hold} as defined are positive. Internal delays in the data path may result in a shift of the data waveform to the left, with respect to the clock, resulting in negative hold times.

SWITCHING TIME TEST CIRCUIT

50 ohm termination to ground located in each scope channel input.

All input and output cables to the scope are equal lengths of 50 ohm coaxial cable. Wire length should be ≤¼ inch from TP_{in} to input pin and TP_{out} to output pin.

All unused outputs loaded with 50 ohms to ground.



RECOMMENDED OPERATING CONDITIONS – MC10802

Parameter	Symbol	Value	Unit
Supply Voltage (V_{CC} = 0 Volts)	V_{EE}	−4.68 to −5.72	Vdc
Operating Temp. (Functional)	T_A	−30 to +85	°C
Max Output Drive	–	50Ω to −2.0 Vdc	–
Maximum Clock Input Rise and Fall Time (20% to 80%)	t_r, t_f	10	ns
Minimum Clock Pulse Width	PW	5	ns

ELECTRICAL CHARACTERISTICS

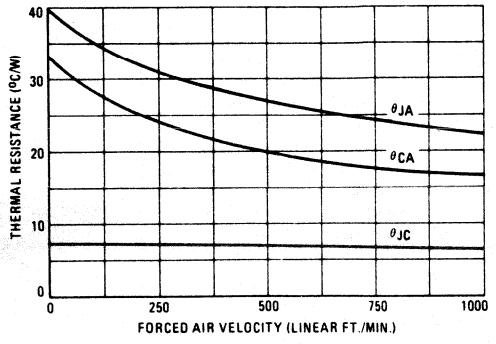
Each MECL 10,000 series circuit has been designed to meet the dc specifications shown in the test table, after thermal equilibrium has been established. The circuit is in a test socket or mounted on a printed circuit board and transverse air flow greater than 500 linear fpm is maintained. Outputs are terminated through a 50-ohm resistor to −2.0 volts. Test procedures are shown for only one input, or for one set of input conditions. Other inputs tested in the same manner.

@ Test Temperature	TEST VOLTAGE VALUES (Volts) $V_{IH\ max}$	$V_{IL\ min}$	$V_{IHA\ min}$	$V_{ILA\ max}$	V_{EE}
−30°C	-.890	-1.890	-1.205	-1.500	-5.2
+25°C	-.810	-1.85	-1.105	-1.475	-5.2
+85°C	-.700	-1.825	-1.035	-1.440	-5.2

Characteristic	Symbol	Pin Under Test	TEST LIMITS −30°C Min	−30°C Max	+25°C Min	+25°C Typ	+25°C Max	+85°C Min	+85°C Max	Unit	VOLTAGE APPLIED TO PINS LISTED BELOW: $V_{IH\ max}$	$V_{IL\ min}$	$V_{IHA\ min}$	$V_{ILA\ max}$	V_{EE}	(V_{CC}) Gnd
Power Supply Drain Current	I_{EE}	12	–	–	–	–	141	–	–	mAdc	–	–	–	–	12	1, 24
Input Current	I_{inH}	5	–	–	–	–	320	–	–	μAdc	5	–	–	–	12	1, 24
		13	–	–	–	–	310	–	–	μAdc	13		–	–	12	1, 24
	I_{inL}	5	–	–	0.5	–	–	–	–	μAdc	–	5	–	–	12	1, 24
Logic "0" Output Voltage	V_{OH}	2	-1.060	-.89	-.960	–	-.810	-.890	-.700	Vdc	*5,8,9,10, 11,18	7,13,14,15, 19,20	–	–	12	1, 24
Logic "1" Output Voltage	V_{OL}	2	-1.89	-1.675	-1.85	–	-1.65	-1.825	-1.615	Vdc	*8,9,10, 11,18	5,7,13,14, 15,19,20	–	–	12	1, 24
Logic "0" Threshold Voltage	V_{OHA}	2	-1.08	–	.980	–	–	-.910	–	Vdc	*8,9,10, 11,18	7,13,14,15, 19,20	5	–	12	1, 24
Logic "1" Threshold Voltage	V_{OLA}	2	–	-1.655	–	–	-1.63	–	-1.595	Vdc	*8,9,10, 11,18	7,13,14,15, 19,20	–	5	12	1, 24

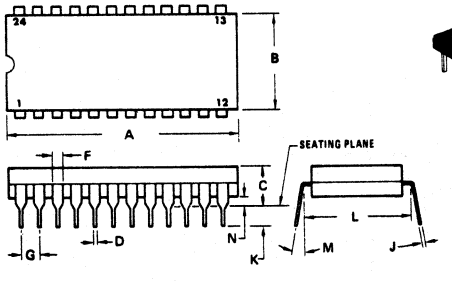
*Apply pulse ⎍ to $\overline{\text{Reset}}$ Pin 17, then
apply pulse ⎍ to CLK Pin 6, then test.

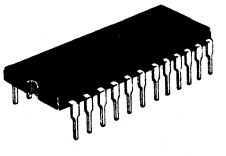
THERMAL CHARACTERISTICS (TYPICAL)



PACKAGE DEMINSIONS

CASE 623-03





DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	31.24	32.26	1.230	1.270
B	12.70	13.72	0.500	0.540
C	4.06	5.59	0.160	0.220
D	0.41	0.51	0.016	0.020
F	1.27	1.52	0.050	0.060
G	2.54 BSC		0.100 BSC	
J	0.20	0.30	0.008	0.012
K	2.29	4.06	0.090	0.160
L	15.24 BSC		0.600 BSC	
M	0°	15°	0°	15°
N	0.51.	1.27	0.020	0.050

NOTES:
1. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.
2. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION. (WHEN FORMED PARALLEL)

MOTOROLA

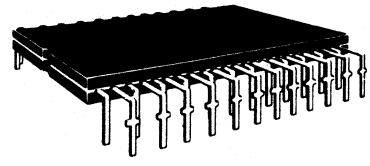
MC10803

INTRODUCTION

The MC10803 Memory Interface Function is an LSI building block for interfacing a high speed processor system to main memory or peripheral equipment. The circuit contains the logic and storage registers for generating memory address and routing incoming or outgoing data. Each part is 4-bits wide and can be connected in parallel to meet wider system I/O word requirements. An internal ALU allows the MC10803 to also assume processor ALU responsibility for many controller applications. Maximum system flexibility is maintained with 5 separate data ports.

The Memory Interface Function as shown in the block diagram below contains six 4-bit registers, an ALU with encoded function/operand select logic, and data transfer circuitry on a single MECL bipolar LSI circuit. Fifteen select (MS) lines control register selection, 13 basic ALU functions, and 17 data transfer operations.

MECL — LSI MEMORY INTERFACE FUNCTION



CASE 725-01

MEMORY INTERFACE FUNCTION
BLOCK DIAGRAM – MC10803

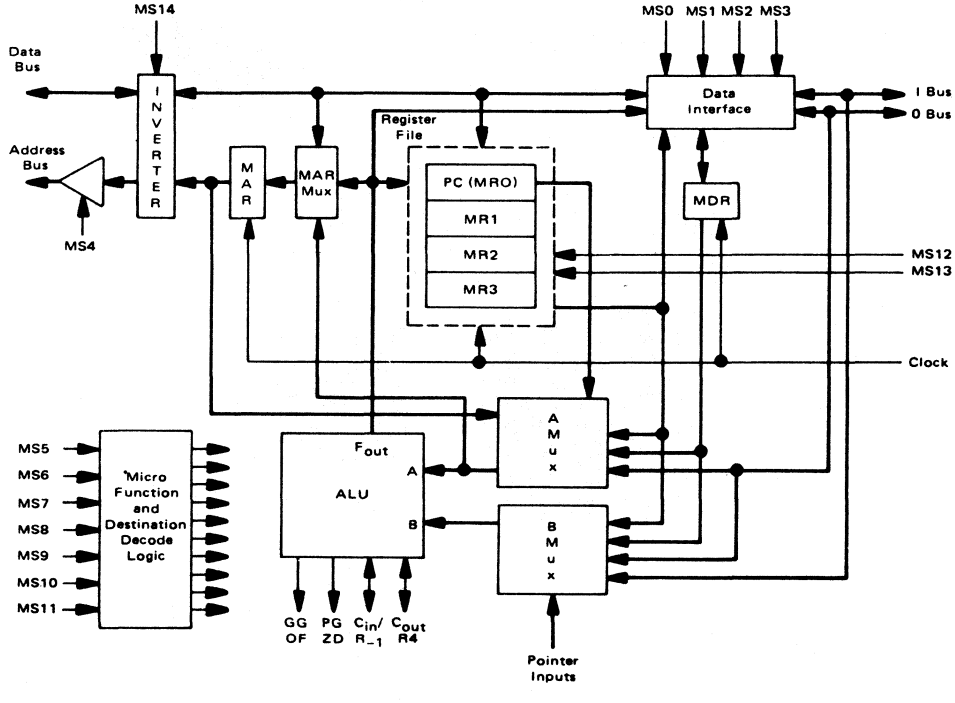
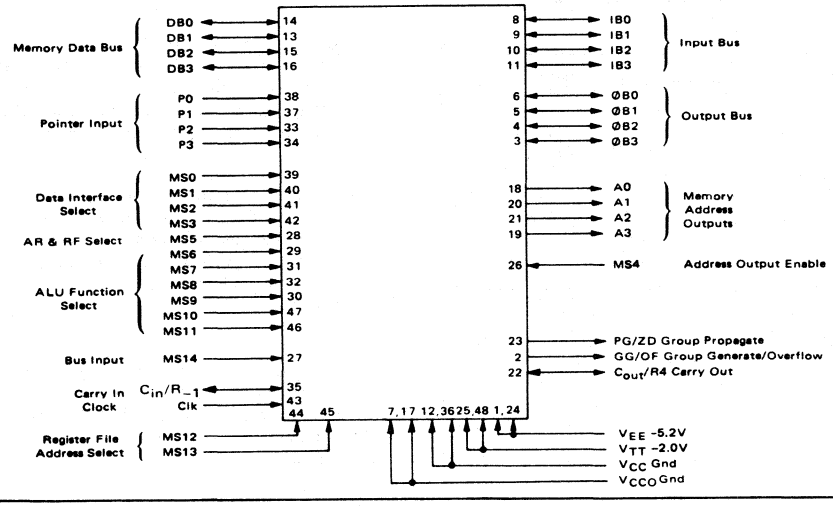


TABLE OF CONTENTS

IMPORTANT FEATURES

1. Internal ALU for address generation
   a. 13 arithmetic, logic and shift functions
   b. 7 separate ALU operands
2. Four word register file
   a. Program counter
   b. 3 general purpose registers for index registers, stack pointers, etc.
3. Memory data register
4. Memory address register
5. 17 data transfer and storage operations
6. 4 bits wide and fully expandable
7. 5 data ports for maximum versatility
8. Internal Register File can be expanded by using External Register File connected to the I Bus and O Bus.
9. Fully compatible with MECL 10000
   a. Power supplies
   b. Logic levels

INPUT/OUTPUT DIAGRAM-MC10803



ABSOLUTE MAXIMUM RATINGS (see Note 1)

RATING		SYMBOL	VALUE	UNIT
Supply Voltage (V_{CC} = 0)		V_{EE} V_{TT}	−8 to 0 −4 to 0	Vdc Vdc
Input Voltage (V_{CC} = 0)	Std Bus	V_{in} V_{in}	0 to V_{EE} Note 2	Vdc Vdc
Output Source Current	Cont Surge	I_o I_o	< 50 <100	mAdc mAdc
Storage Temp.		$T_{stg.}$	−55 to +150	°C
Junction Temp.		T_j	165	°C

NOTE: 1. Permanent device damage may occur if absolute maximum ratings are exceeded. Functional operation should be restricted to RECOMMENDED OPERATING CONDITIONS. Exposure to higher than recommended voltages for extended periods of time could affect device reliability.

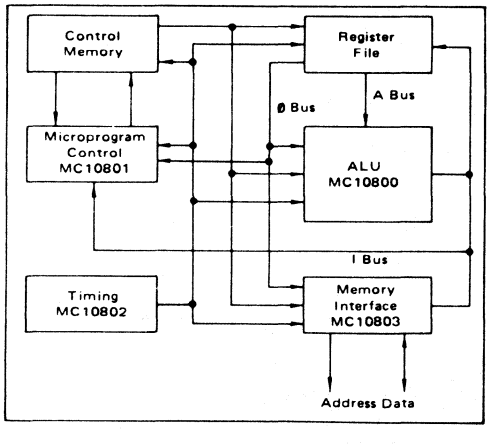
NOTE: 2. Input voltage limit is V_{CC} to −2 Volts when the bus is used as an input and the output drivers are disabled.

SYSTEM OVERVIEW

The Motorola M10800 family of LSI processor circuits combines the cost and size advantages of LSI with system design flexibility. Each family part is a major system building block which can be interconnected and programmed for a wide range of processor applications. Figure 1 illustrates a method of using the various circuits in a general purpose processor. The MC10800 4-Bit ALU Slice performs the various arithmetic, logic, and shift functions. This circuit features full BCD capability and a complete set of status outputs. The MC10801 Microprogram Control Function addresses and sequences through microprogram control memory. A set of 16 control instructions provides for direct jumps, conditional branches, and subroutines within microprogram. The MC10802 Timing Function generates clock phases and features single cycle or single phase clock increment for troubleshooting or diagnostics.

The MC10803 Memory Interface Function interfaces between the LSI processor circuits and other parts of a system. The circuit generates memory addresses and provides for the bidirectional transfer of processor data. The MC10803 represents a step forward for bipolar LSI systems by putting arithmetic capability at I/O address output port. This allows the various modes of memory addressing (relative, indexed, extended, indirect, etc.) to be performed within the MC10803 memory interface block, Figure 1, and not tie up the main ALU. It is also possible to control the program counter or do stack pointer push or pop functions inside the MC10803 again giving the register file and ALU more time for complex computations.

FIGURE 1 – MICROPROGRAMMED PROCESSOR



Within the M10800 family the MC10803 adapts to a wide variety of system functions. The Figure 1 system layout uses the MC10803 parallel with the MC10800 for maximum computational power and throughput. Besides I/O interfacing, the MC10803 ALU can be connected in parallel with the MC10800 for double precision arithmetic.

In larger systems, MC10803 blocks would be operated in parallel for multiple I/O ports and to minimize overhead burden on the processor ALU. Many peripheral controller systems perform high speed data transfer operations and do not need the computational power of the MC10800. These systems can use the MC10803 for both ALU and interface logic to minimize part count.

In some applications, the MC10803 can be used as a microprogram controller with the internal ALU for relative addressing. The circuit then generates addresses for control memory instead of main memory and uses the internal register file for storing subroutines.

The Motorola M10800 circuits interface directly to all parts in the MECL 10,000 family. This provides a source for high speed memories and a complete mix of MSI and SSI circuits. Circuits are available for special hardware functions from high speed multiply to error detection and correction.

Versatility is a main point of the M10800 Family. The block diagram in Figure 1 is intended to illustrate the purpose of the various LSI functions and not restrict the designer to any particular system configuration or application.

PIN ASSIGNMENTS

Pin Designation	Pin Number	Description
DB0	14	Data Bus-LSB I/O
DB1	13	Data Bus-NLSB I/O
DB2	15	Data Bus-NMSB I/O
DB3	16	Data Bus-MSB I/O
A0	18	Address Bus-LSB Output
A1	20	Address Bus-NLSB Output
A2	21	Address Bus-NMSB Output
A3	19	Address Bus-MSB Output
ØB0	6	Output Bus-LSB I/O
ØB1	5	Output Bus-NLSB I/O
ØB2	4	Output Bus-NMSB I/O
ØB3	3	Output Bus-MSB I/O
IB0	8	Input Bus-LSB I/O
IB1	9	Input Bus-NLSB I/O
IB2	10	Input Bus-NMSB I/O
IB3	11	Input Bus-MSB I/O
P0	38	Pointer Input-LSB
P1	37	Pointer Input-NLSB
P2	33	Pointer Input-NMSB
P3	34	Pointer Input-MSB
PG/ZD	23	Group Propagate/Zero Detect
GG/OF	2	Group Generate/Overflow
C_{in}/R_{-1}	35	Carry In/Shift LSB
$C_{out}/R4$	22	Carry Out/Shift MSB
Clk	43	Clock
MS0	39	Data Transfer Select Input
MS1	40	Data Transfer Select Input
MS2	41	Data Transfer Select Input
MS3	42	Data Transfer Select Input
MS4	26	Address Output Enable
MS5	28	AR & RF Select Input
MS6	29	Function Select Input
MS7	31	Function Select Input
MS8	32	Function Select Input
MS9	30	Function Select Input
MS10	47	Function Select Input
MS11	46	Function Select Input
MS12	44	Register File Address Select Input
MS13	45	Register File Address Select Input
MS14	27	Inverter Select Input
V_{EE}	1	-5.2 Volt Supply
V_{EE}	24	-5.2 Volt Supply
V_{TT}	25	-2.0 Volt Supply
V_{TT}	48	-2.0 Volt Supply
V_{CC}	12	Ground
V_{CC}	36	Ground
V_{CCO}	7	Ground
V_{CCO}	17	Ground

ARCHITECTURAL DESCRIPTION

The MC10803 Memory Interface Function contains 6 registers as shown in Figure 2. Three registers are given function designations; memory data register-MDR, memory address register-MAR, and program counter-PC. The other registers can be used as stack pointer, index registers, or for other memory related functions. Register assignments are not fixed and the system designer should utilize the register file as best fits the system requirements. Five 4-bit data ports (I Bus-IB, ØBus-ØB, Data Bus-DB, Address Bus-AB, and Pointer Inputs-P) are available to enter and output information. An ALU performs arithmetic, logic, and shift functions on seven possible operands (MAR, PC, RF, MDR, I BUS, ØBUS, and P). The ALU utilizes only 4 package pins for overflow, zero detect, group generate, group propagate, carry in, carry out, sign bit, shift in, and shift out. The operation of these pins is controlled by the selected ALU function. The various MC10803 sections in Figure 2 are described below.

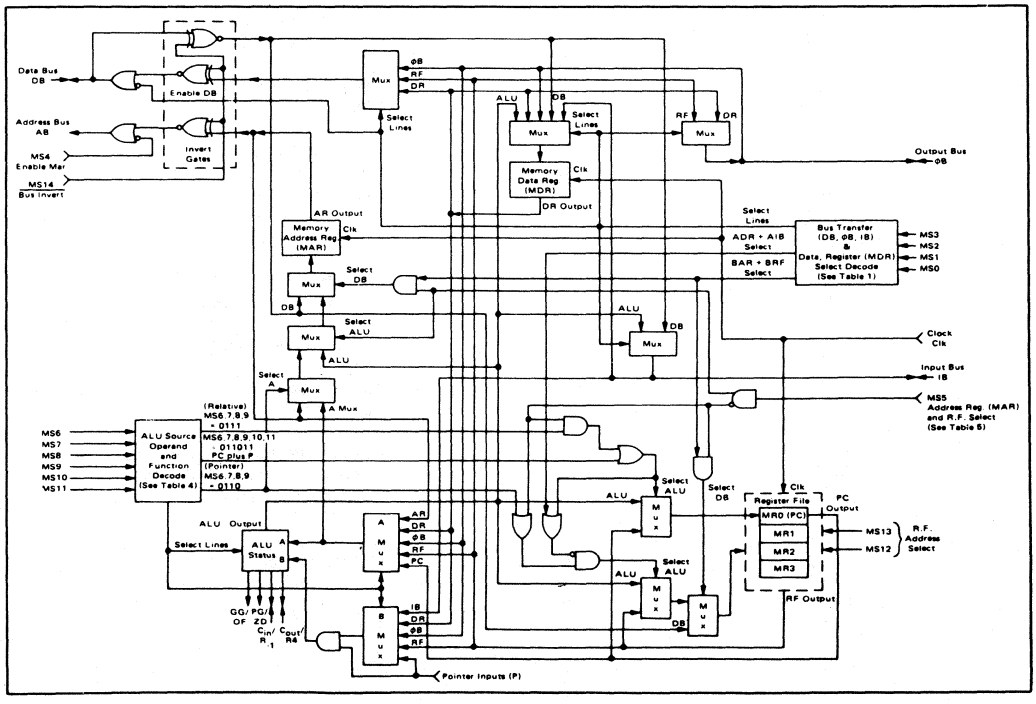
DATA INTERFACE

The data interface controls the transfer of information between various busses and registers within the MC10803.

When information is routed to a register, data is loaded on the positive going (V_{OL} to V_{OH}) clock edge. The 17 possible data transfers are listed below. Transfers numbered 12 and 13 use select line MS5 to select the Data Bus destination.

1. FØB – Register File to Ø Bus
2. RØB – Data Register to Ø Bus
3. AIB – ALU to I Bus
4. BIB – Data Bus to I Bus
5. FDB – Register File to Data Bus
6. ØDB – Ø Bus to Data Bus
7. RDB – Data Register to Data Bus
8. ADR – ALU to Data Register
9. BDR – Data Bus to Data Register
10. IDR – I Bus to Data Register
11. ØDR – Ø Bus to Data Register
12. BRF – Data Bus to Register File (MS5=0)
13. BAR – Data Bus to Address Register (MS5=1)
14. PIØ – Pipeline the I Bus to Ø Bus via the Data Register
15. PBØ – Pipeline from Data Bus to Ø Bus via the Data Register
16. PIB – Pipeline the I Bus to the Data Bus via the Data Register
17. NOP – No transfer occurs

Figure 2. Detailed MC10803 Functional Block Diagram



MR0 THROUGH MR3 – REGISTER FILE

Registers MR0 through MR3 make up a 4-word register file. Each register is 4 bits wide and expands to desired bit size with parallel MC10803 circuits. All registers are constructed from master-slave flip-flops and store information on the positive going clock edge. Register MR0 is given preferential treatment as a program counter. A separate ALU multiplexer input is used for memory addressing with the program counter. The register file can be loaded from the Data Bus or the ALU outputs. Information in the register file can be routed to the 0 Bus, Data Bus, ALU, or Address Register.

MDR – MEMORY DATA REGISTER

The Memory Data Register is the primary storage element for information going to or from the processor on the Data Bus. This register can be loaded from the Ø Bus, ALU, Data Bus, or I Bus. Destinations for information in the Data Register are Data Bus, Ø Bus, Address Register, or ALU. In addition to holding memory data, the MDR can be used as an accumulator when the MC10803 operates as the main ALU or in parallel with the MC10800 for double precision arithmetic.

MAR – MEMORY ADDRESS REGISTER

The Memory Address Register holds the outgoing memory address information. Depending on the type of memory address, this register can be loaded from the Data Bus, ALU outputs, Ø Bus, Data Register, Register File, or Program Counter. The memory address register holds information for the MC10803 address outputs and goes to the ALU for address modification.

ALU – ARITHMETIC LOGIC UNIT

The ALU features 7 basic functions: Add, Subtract, AND, OR, Exclusive OR, Shift Left, and Shift Right. These accomplish a variety of memory related tasks, including masking, bit manipulation, extended addressing, updating program counter, stack push or pop, and generating address offsets. Possible ALU operands are Ø Bus, Register File, I Bus, Data Register, Address Register, Program Counter, or the Pointer (P) inputs. Selection of the ALU function and source operands is controlled by the Microfunction Decode logic from select lines MS6 through MS11.

C_{in}/R_{-1}

During arithmetic functions this pin is the carry in to the least significant ALU bit. During shift right, R_{-1} shifts out the least significant bit. Shift left is an arithmetic operation that is accomplished by adding a word to itself.

$C_{out}/R4$

This pin is the carry out for ALU arithmetic functions. During shift operations it couples the most significant shift bit. The $C_{out}/R4$ pin contains the most significant bit for sign testing during ALU logic operations. The MSB is also brought out on arithmetic shift right because the sign bit is repeated.

PG/ZD

The PG/ZD pin serves as group propagate for fast ALU arithmetic operations and as zero detect for logic and some arithmetic functions. The selection between propagate and zero detect is controlled by the ALU function select line MS9.

GG/OF

The GG/OF pin contains group generate for all ALU arithmetic operations. In addition, the pin can be selected to display 2's complement overflow for selected arithmetic and shift left operations. A conflict between overflow and group generate is avoided by programming only the most significant slice for overflow. (See the applications section for additional information).

DATA BUS, ADDRESS BUS, AND INVERTER

The Data Bus and Address Bus are 4-bit ports. The Data Bus with bidirectional information transfer is ideal for interfacing to memory or peripheral data ports. Transfer of data on this port is controlled by the Data Interface Select lines. When not used as outputs, the data bus drivers are held at a negative logic "1", MECL V_{OL}. The Data Bus port is then available as an input using the MECL emitter-tie feature.

The Address Bus is a 4-bit port designed to output information held in the memory address register. It is placed in the MC10803 for memory and peripheral addressing, but can also be used to output ALU calculations stored in the MAR. The Address Bus is driven from an output buffer controlled by MS4. This select line (when a logic "0") forces the address outputs to a negative logic "1", MECL V_{OL}, and frees the port for other system functions.

An optional inverter controlled by MS14 is placed in series with the Data Bus and the Address Bus. In the invert mode (MS14="0"), incoming or outgoing memory data and outgoing address information is inverted. Otherwise, true data is transferred through the circuit. The invert feature allows a direct interface with any combination of positive or negative logic formats between the processor and I/O busses.

POINTER INPUTS

The P inputs are 4 input pins routed directly to the ALU. They provide a source for pointers, address modifiers, or constants as needed for memory addressing. These inputs also allow for push/pop stack operations and incrementing the program counter in byte oriented memories. When used as a main ALU, the P inputs are available for other system functions such as a register file port.

I BUS AND Ø BUS

The I Bus and Ø Bus are bidirectional data ports designed to interface between the MC10803 Memory Interface Function and other parts of the processor (Register File, ALU, Microprogram Control, etc.). Transfer of information to and from the I Bus and Ø Bus is controlled by the Data Interface select lines MS0 through MS3.

ALU SELECT AND DESTINATION DECODE

Select lines MS6 through MS11 are decoded to determine the ALU operands, the ALU function, while MS5 controls the destination for the ALU output. The ALU function select logic is designed to work in parallel with the Data Interface. This allows simultaneous data and address operations within the MC10803 for maximum system throughput. Programming information on the ALU function and Destination Decoding select lines is given in the following Functional Description tables.

Clk – CLOCK

A single clock input is common to all registers in the MC10803. Each register is built from master-slave flip-flops and loads information on the positive going (V_{OL} to V_{OH}) clock edge. Signals on the register inputs can change at any time with the clock input at either logic state and not change the register outputs. The only restriction to changing register inputs is during the set-up and hold time near the positive going clock edge. Information is held constant in any register not selected to receive a data transfer.

FUNCTIONAL DESCRIPTION

Fifteen select lines, MS0 through MS14, control the flow of information inside the MC10803 Microprogram Control Function. The following information describes programming these inputs to perform the various circuit functions. *All truth tables are expressed in negative logic with V_{OL} being a logic 1 and V_{OH} a logic 0.*

DATA TRANSFER CONTROL – MS0,MS1,MS2,MS3,MS5,MS14

Table 1 lists the 17 MC10803 data transfer functions. In addition to selecting between the BRF and BAR transfers, MS5 is used to select the memory address register data source as described later. MS14 inverts information going to and from the Data Bus. Inverting the Data Bus also inverts information between the memory address register and Address Bus.

TABLE 1. DATA TRANSFER CONTROL

MS 3	MS 2	MS 1	MS 0	MS 5	MS 14	MNEMONIC	OPERATION SOURCE	OPERATION DESTINATION	BUS OUTPUTS DB	BUS OUTPUTS IB	BUS OUTPUTS OB	NEXT STATE DR
0	0	0	0	X	X	NOP	NO OPERATION		1	1	1	–
0	0	0	1	X	X	AIB	ALU	IB	1	ALU	1	–
0	0	1	0	X	X	ØDR	ØB	DR	1	1	1	ØB
0	0	1	1	X	X	ADR	ALU	DR	1	1	1	ALU
0	1	0	0	0	0	BRF*	$\overline{DB}$	RF	1	1	1	–
				0	1		DB	RF	1	1	1	–
				1	0	BAR*	$\overline{DB}$	AR	1	1	1	–
				1	1		DB	AR	1	1	1	–
0	1	0	1	X	0	BIB	$\overline{DB}$	IB	1	$\overline{DB}$	1	–
				X	1		DB	IB	1	DB	1	–
0	1	1	0	X	0	BDR	$\overline{DB}$	DR	1	1	1	$\overline{DB}$
				X	1		DB	DR	1	1	1	DB
0	1	1	1	X	X	IDR	IB	DR	1	1	1	IB
1	0	0	0	X	0	FDB	$\overline{RF}$	DB	$\overline{RF}$	1	1	–
				X	1		RF	DB	RF	1	1	–
1	0	0	1	X	0	RDB	$\overline{DR}$	DB	$\overline{DR}$	1	1	–
				X	1		DR	DB	DR	1	1	–
1	0	1	0	X	0	ØDB	$\overline{ØB}$	DB	$\overline{ØB}$	1	1	–
				X	1		ØB	DB	ØB	1	1	–
1	0	1	1	X	0	PIB	IB	DR	$\overline{DR}$	1	1	IB
							$\overline{DR}$	DB				
				X	1		IB	DR	DR	1	1	IB
							DR	DB				
1	1	0	0	X	X	FØB	RF	ØB	1	1	RF	–
1	1	0	1	X	X	RØB	DR	ØB	1	1	DR	–
1	1	1	0	X	0	PBØ	$\overline{DB}$	DR	1	1	DR	$\overline{DB}$
							DR	ØB				
				X	1		DB	DR	1	1	DR	DB
							DR	ØB				
1	1	1	1	X	X	PIØ	IB	DR	1	1	DR	IB
							DR	ØB				

*These instructions override the ALU destination (see Table 5)

X = Don't care

– = No Change in Register State

ADDRESS OUTPUT ENABLE MS4

MS4 controls the Address Bus output drivers. The disable feature can be used for DMA operation, multiprocessors on a common memory, and other memory bus transfers. Table 2 shows the programming of MS4.

REGISTER FILE ADDRESS SELECT-MS12 AND MS13

MS12 and MS13 are used to select one of the four register file words for all register file (RF) load or read operations. The program counter (PC) operations are a special case that automatically pick MR0 and do not require MS12 or MS13. This does not inhibit selecting MR0 for any RF function. Table 3 gives the truth table for register file selection.

ALU CONTROL – MS6 THROUGH MS11

MS6 through MS11 control the ALU operand and function selection. MS6, MS7, and MS8 select the ALU function. MS9 selects the ALU function and controls the ALU status outputs. MS10 and MS11 control the ALU operand selection. Table 4 shows programming for the various ALU functions and selection of the ALU operands.

TABLE 2 – ADDRESS BUS CONTROL

MS4	MS14	ADDRESS BUS
0	–	Logic 1
1	0	$\overline{MAR}$
1	1	MAR

TABLE 3 – REGISTER FILE SELECTION

MS13	MS12	REGISTER
0	0	MR0 (PC)
0	1	MR1
1	0	MR2
1	1	MR3

TABLE 4. ALU PROGRAMMING*

	MS 6 7 8	MS 6 7 8	MS 6 7 8	MS 6 7 8	MS 6 7 8	MS 6 7 8	MS 6 7 8	MS 6 7 8
	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
MS 9 10 11	SUB $G_G=V\ P_G=Z_D$	ADD $G_G=V\ P_G=Z_D$	ASL $G_G=V\ P_G=Z_D$	POINTER $G_G=G\ P_G=P$	AND $C_{out}=R3\ P_G=Z_D$	OR $C_{out}=R3\ P_G=Z_D$	ASR $C_{out}=R3\ P_G=Z_D$ $G_G=1$	EOR $C_{out}=R3\ P_G=Z_D$
0 0 0	ØB Minus IB·P	ØB Plus IB·P	ØB	ØB Plus P	ØB·(IB·P)	ØB + (IB·P)	ØB	ØB ⊕ (IB·P)
0 0 1	ØB Minus DR·P	ØB Plus DR·P	DR	DR Plus P	ØB·(DR·P)	ØB + (DR·P)	DR	ØB ⊕ (DR·P)
0 1 0	RF Minus ØB·P	RF Plus ØB·P	RF	RF Plus P	RF·(ØB·P)	RF + (ØB·P)	RF	RF ⊕ (ØB·P)
0 1 1	RF Minus DR·P	RF Plus DR·P	AR	PC Plus P	RF·(DR·P)	RF + (DR·P)	AR	RF ⊕ (DR·P)
	$G_G=G_GP_G=P_G$	$G_G=G_GP_G=P_G$	LSL $G_G=G_GP_G=P_G$	RELATIVE $G_G=G_GP_G=P_G$	$C_{out}=R3\ P_G=Z_D$	EORP $C_{out}=R3\ P_G=Z_D$	LSR $G_G=1\ P_G=Z_D$	MODIFY $G_G=G_GP_G=P_G$
1 0 0	ØB Minus IB·P	ØB Plus IB·P	ØB	PC Plus IB·P	ØB · P	ØB ⊕ P	ØB	AR Plus P
1 0 1	ØB Minus DR·P	ØB Plus DR·P	DR	PC Plus DR·P	DR · P	DR ⊕ P	DR	AR Plus DR·P
1 1 0	RF Minus ØB·P	RF Plus ØB·P	RF	PC Plus ØB·P	RF · P	RF ⊕ P	RF	AR Plus ØB·P
1 1 1	RF Minus DR·P	RF Plus DR·P	AR	PC Plus RF·P	AR · P	AR ⊕ P	AR	AR Plus RF·P

*NOTE: 1. "A MUX" Operand is Left Entry, and "B MUX" Operand is Right Entry.
2. Single Operand Instructions use "A MUX".
3. R3 = Sign Detect (MSB of ALU Output)

ALU DESTINATION CONTROL – MS5

The ALU output can be routed to several points in the MC10803. ALU to I Bus and ALU to Data Register transfers are controlled by the Data Interface, Table 1. Other MC10803 operations load the ALU output into the Memory Address Register (AR), Register File (RF), or Program Counter (PC). These transfers are determined by the selected Data Interface transfer, MS5, and the selected ALU function. Table 5 shows programming combinations for routing information to the AR, RF, and PC. Commands that transfer the selected ALU A MUX operand to the address register are designated by the letter A under the AR column.

TABLE 5 – DESTINATION PROGRAMMING

ALU OPERATION	MS 6	7	8	9	10	11	DATA INTERFACE OPERATION (SEE TABLE 1)	MS 5	DESTINATION AR	RF	PC
(POINTER)							BRF	0	A	DB	–
							BAR	1	DB	ALU	–
ØB Plus P	0	1	1	0	0	0					
DR Plus P	0	1	1	0	0	1	ADR or AIB	0	A	–	–
								1	ALU	–	–
RF Plus P	0	1	1	0	1	0	All Others	0	A	ALU	–
								1	ALU	ALU	–
(POINTER)							BRF	0	PC	DB	ALU*
							BAR	1	DB	–	ALU
PC Plus P	0	1	1	0	1	1	All Others	0	PC	–	ALU
								1	ALU	–	ALU
(RELATIVE)							BRF	0	–	DB	ALU*
PC Plus IB·P	0	1	1	1	0	0	BAR	1	DB	–	–
PC Plus DR·P	0	1	1	1	0	1					
PC Plus ØB·P	0	1	1	1	1	0	ADR or AIB	0	–	–	ALU
								1	ALU	–	–
PC Plus RF·P	0	1	1	1	1	1	All Others	0	–	–	ALU
								1	ALU	–	–
SUB	0	0	0	–	–	–	BRF	0	–	DB	–
ADD	0	0	1	–	–	–	BAR	1	DB	–	–
All ASL/LSL	0	1	0	–	–	–					
Others AND	1	0	0	–	–	–	ADR or AIB	0	–	–	–
OR/EORP	1	0	1	–	–	–		1	ALU	–	–
ASR/LSR	1	1	0	–	–	–					
EOR/MOD	1	1	1	–	–	–	All Others	0	–	ALU	–
								1	ALU	–	–

*If the PC is selected (MS12 = 0, MS13 = 0) during the indicated operation, then DB + ALU → PC

– = No change in register state.

NOTE: If MS14 = 0, replace DB with $\overline{DB}$ in Table 5.

APPLICATIONS INFORMATION

Figure 3 shows four MC10803's interconnected in a typical 16-bit configuration. A single MC10179 lookahead carry block gives second level lookahead for maximum performance. Buffers are required between carry out of the lookahead and carry out of the MC10803 so that the higher speed lookahead dominates. The buffer still permits C_{in} to C_{out} transfers required for shift right. Smaller 8 or 12-bit systems use ripple carry and do not require the MC10179 or buffers. Systems larger than 16 bits can use additional MC10179 circuits to maintain maximum performance.

The following section defines the various ALU functions and describes ways to use the part for memory addressing. Refer to Table 4 for programming and choice of operands.

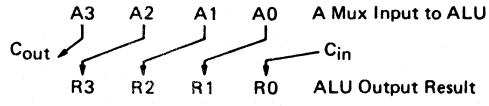
ADD-(OP A) Plus (OP B) · P Plus C_{in} = Result

SUB-(OP A) Plus $\overline{\text{(OP B)}}$ · P Plus C_{in} = Result

Add and subtract perform the logic AND using the B operand and P inputs. This is used for bit masking prior to the arithmetic operation. Lookahead carry is possible when MS9 is equal to logic 1 on the less significant MC10803's, see Figure 3. The most significant slice can have MS9 equal to logic 0 to provide 2's complement overflow without interfering with the MC10179.

ASL,LSL – SHIFT LEFT ONE BIT

The selected operand is shifted left by adding the input to itself. For an arithmetic shift left, MS9 equal logic 0, overflow is generated from a change in the sign bit. As with add and subtract lookahead carry minimizes propagation delay.



ASR,LSR – SHIFT RIGHT ONE BIT

Shift right uses the carry pins for shift coupling. MS9 is held at logic 0 for arithmetic shifting. This holds the sign bit constant and displays the sign on C_{out} of the most significant slice. Group propagate is always zero detect and group generate is held at logic 1 to disable the MC10179 in Figure 3.

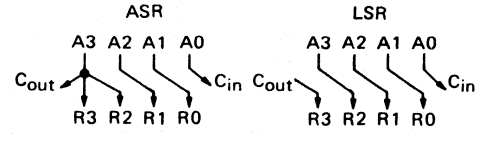
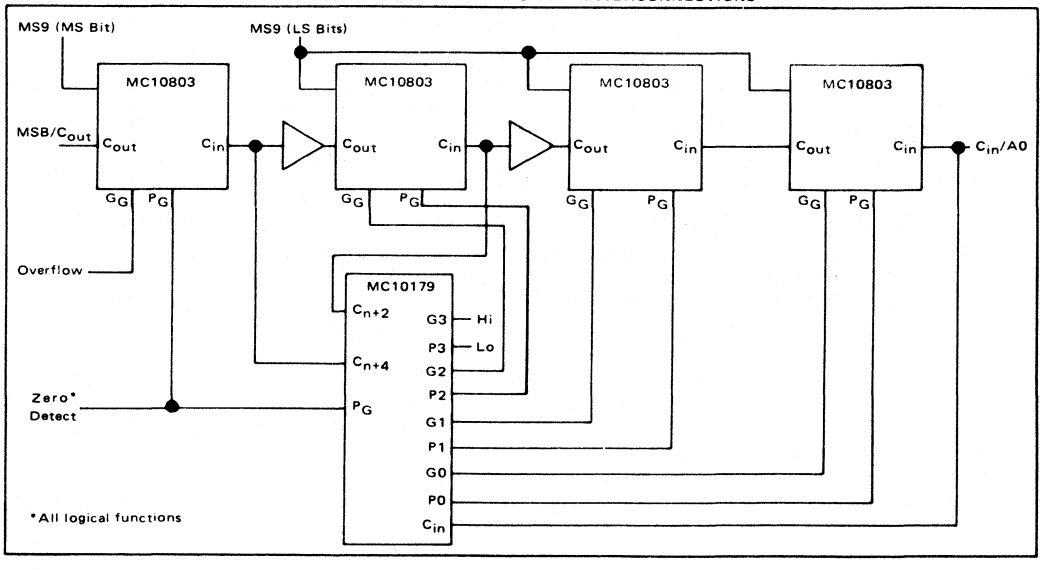


FIGURE 3. 16-BIT LOOKAHEAD CARRY INTERCONNECTIONS



MODIFY PROGRAM COUNTER-PC Plus (OP B) · P Plus C_{in} = Result

MODIFY POINTER-(OP A) Plus P Plus C_{in} = Result

MODIFY ADDRESS REGISTER-AR Plus (OP B) · P Plus C_{in} = Result .

The three preceding functions are normally used with unsigned or positive numbers, and overflow is not generated. Modify Program Counter provides a means for doing a program jump from a variety of sources. The Modify Address Register gives the same jump in program capability without changing the program counter. The Modify functions are especially helpful for stack operations. The stack can directly address memory while it is simultaneously updated in the ALU. Alternately stack information can be updated, routed to the memory address register, and stored in the register file during the same clock cycle. The modify pointer format will take the P inputs and increment or decrement the program counter without disturbing other registers in the MC10803.

AND-(OP A) AND (OP B) AND P = Result

OR-(OP A) OR (OP B) AND P = Result

EOR-(OP A) EXCLUSIVE OR (OP B) AND P = Result

BIT MASK-(OP A) AND P = Result

BIT TOGGLE-(OP A) EXCLUSIVE OR P = Result

Group propagate performs zero detect and C_{out} shows the MSB for all logic functions. AND, OR, and EXOR are general purpose logic functions, allowing the MC-10803 to perform compare and bit manipulation for controller applications. The Bit Mask, in addition to forcing zero bits, can be used to obtain the zero detect and sign bit status after an arithmetic function. This is accomplished when all P Inputs are a logic "1". The Bit Toggle can be used with error detection and correction circuits to correct incoming data from memory or pheripherals. Other applications include encoding or decoding for secure communications or special data formats.

STATUS AND SHIFT I/O LOGIC FUNCTION

Table 6 defines the logic equations for the status and shift conditions for each of the 16 ALU functions. The A and B inputs are selected as shown in Table 4. Table 7 defines the symbols used in the various tables.

TABLE 6 ALU OUTPUT, STATUS & SHIFT I/O

6	7	8	9	ALU FUNCTION	ALU OUTPUT $R_3\,R_2\,R_1\,R_0$	STATUS OR SHIFT I/O GG/OF	PG/ZD	C_{out}/R_4	C_{in}/R_{-1}
0	0	0	0	(SUBTRACT	A plus $\overline{B}$ plus C_{in}	OF	ZD	C_{out} (Output)	C_{in} (Input)
			1	(SEE NOTE 1)		GG	PG		
0	0	1	0	ADD	A plus B plus C_{in}	OF	ZD	C_{out} (Output)	C_{in} (Input)
			1			GG	PG		
0	1	0	0	ASL	A plus A plus C_{in}	$A_3 \oplus A_2$	ZD	C_{out} (Output)	C_{in} (Input)
			1	LSL		GG	PG		
0	1	1	0	POINTER	A plus P plus C_{in}	GG	PG	C_{out} (Output)	C_{in} (Input)
			1	RELATIVE	A plus B plus C_{in}	GG	PG		
1	0	0	0	AND	A · B	$A_3 \cdot \overline{B}_3$	ZD	R_3 (Output)	C_{in} (Input)
			1		A · P	A_3	ZD		
1	0	1	0	OR	A + B	0	ZD	R_3 (Output)	C_{in} (Input)
			1	EORP	A ⊕ P	GG	ZD		
1	1	0	0	ASR	$A_3\,A_3\,A_2\,A_1$	1	ZD	R_3 (Output)	A_0 (Output)
			1	LSR	$R_4\,A_3\,A_2\,A_1$	1	ZD	R_4 (Input)	
1	1	1	0	EOR	A ⊕ B	GG	ZD	R_3 (Output)	C_{in} (Input)
			1	MODIFY	AR plus B plus C_{in}	GG	PG	C_{out} (Output)	

NOTE 1: $\overline{B}_i$ is inserted for B_i in the equations below when the SUBTRACT mode is selected.

DEFINITIONS:

$PG = (A_3+B_3) \cdot (A_2+B_2) \cdot (A_1+B_1) \cdot (A_0+B_0)$

$GG = (A_3 \cdot B_3) + (A_3+B_3) \cdot (A_2 \cdot B_2) + (A_3+B_3) \cdot (A_2+B_2) \cdot (A_1 \cdot B_1) + (A_3+B_3) \cdot (A_2+B_2) \cdot (A_1+B_1) \cdot (A_0 \cdot B_0)$

$C_3 = (A_2 \cdot B_2) + (A_2+B_2) \cdot (A_1 \cdot B_1) + (A_2+B_2) \cdot (A_1+B_1) \cdot (A_0 \cdot B_0) + (A_2+B_2) \cdot (A_1+B_1) \cdot (A_0+B_0) \cdot C_{in}$

$ZD = \overline{R3} \cdot \overline{R2} \cdot \overline{R1} \cdot \overline{R0}$

$C_{out} = GG + PG \cdot C_{in}$

$OF = C_3 \oplus C_{out}$

$= C_3 \cdot (\overline{A}_3 \cdot \overline{B}_3) + \overline{C}_3 \cdot (A_3 \cdot B_3)$

Where R_i = ALU Output, Bit i; i = (LSB)0,1,2,3(MSB)
A_i = A Operand, Bit i; i = (LSB)0,1,2,3(MSB)
B_i = B Operand, Bit i; i = (LSB)0,1,2,3(MSB)
P = Pointer Inputs

PROPAGATION DELAYS (NANOSECONDS)

Path	Via	-30°C		+25°C		+85°C	
		Typ	Max	Typ	Max	Typ	Max
Cin to IB		15	21	16	22	18	24
Cin to 0F, ZD		11	18	12	19	16	25
Cin to Cout		3	8	3	8	3	9
Cout to IB		12	18	13	19	19	23
MS0, 1, 2, 3 to DB, ØB, IB		14	20	15	21	17	23
MS4, MS14 to A		6	12	7	13	9	16
MS6, 7, 8, 9, 10, 11 to PG, GG		21	28	22	29	26	37
MS6, 7, 8, 9, 10, 11 to R_{-1}		20	27	22	28	25	33
MS12, MS13 to DB, ØB		19	24	20	25	22	30
MS12, MS13 to IB	ALU	31	40	34	43	39	48
MS14 to IB		13	19	14	20	17	25
DB to IB		9	15	10	16	11	17
ØB to IB	ALU	24	31	25	32	29	38
ØB to DB		10	14	10	14	11	15
ØB, IB to 0F, ZD		19	26	20	27	22	29
ØB, IB to PG, GG		20	27	21	28	25	34
ØB, IB to R_{-1}		16	22	17	23	20	26
ØB, IB to Cout		15	21	16	22	19	25
Clk to A		10	15	10	15	11	18
Clk to DB, ØB	DR	13	18	14	19	15	20
Clk to PG, GG	RF	25	32	26	33	30	40
Clk to IB	ALU	30	37	32	39	35	42
P to PG, GG		10	16	11	17	12	18
P to 0F, ZD		11	17	12	18	13	19
P to Cout		10	16	11	17	11	18
P to IB	ALU	17	23	17	23	19	26

TABLE 7 DEFINITIONS OF SYMBOLS USED IN THE VARIOUS TABLES

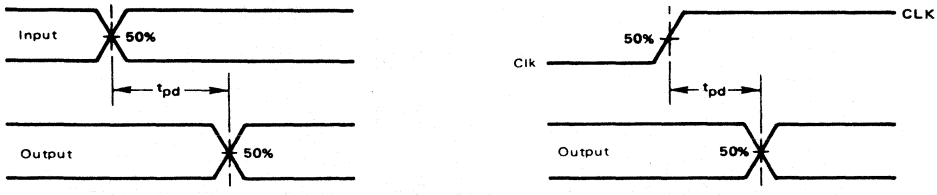
SYMBOL	MEANING
AR	Data Output of the MAR Register
DR	Data Output of the MDR Register
RF	Data Output of One of the Register File Registers as Selected by MS12 and MS13
PC	Data Output of the MRO Register Located in the Register File, PC Data Output is not Dependent on the MS12 or MS13 Select Lines
ALU or R	Data Output of the ALU
A	A Mux Output is the A Operand to the ALU
B	B Mux Output is the B Operand to the ALU (See Table 4 for the Selection of the A & B Operands)
P	Pointer Inputs to ALU
AB	Address Bus
OB	Bidirectional Ø Bus Port
IB	Bidirectional I Bus Port
DB	Bidirectional Data Bus
Plus	2's Complement Addition
Minus	2's Complement Subtraction
	Logical AND
+	Logical OR
⊕	Logical Exclusive OR

SETUP AND HOLD TIMES (NANOSECONDS OVER TEMPERATURE RANGE)

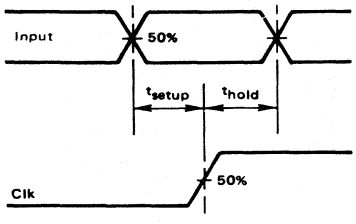
Input	Path	Setup (Min)	Hold (Min)
IB, ØB, DB to Clock	Direct	9.0	4.0
IB, ØB to Clock	via ALU	38	-7.0
C_{in} to Clock	via ALU	26	0
P to Clock	via ALU	30	-2.0
MS0, 1, 2, 3 to Clock	Direct	16	3.0
MS5 to Clock	Direct	11	4.0
MS6, 7, 8 to Clock	via ALU	20	0
MS9, 10 to Clock	via ALU	38	-6.0
MS12, 13 to Clock	Direct	11	+2.0

SWITCHING WAVEFORMS

PROPAGATION DELAYS



SETUP AND HOLD



TEST PROCEDURE:

a) Establish setup time with long t_{hold}.

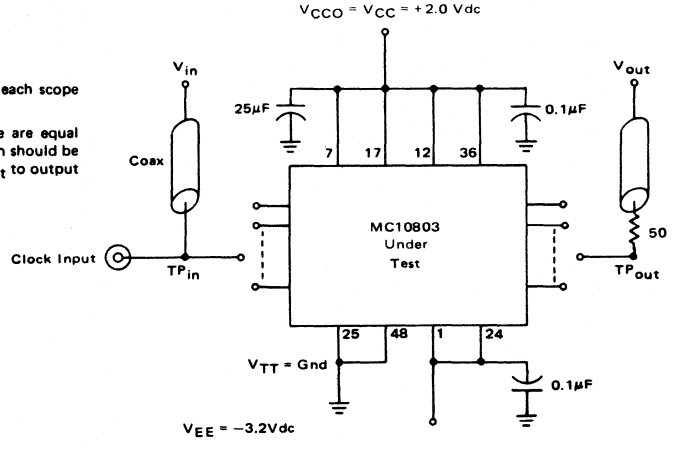
b) Keeping the leading edge of the input constant (t_{setup}) vary the trailing edge of the input to determine t_{hold}.

NOTE: t_{setup} and t_{hold} as defined are positive. Internal delays in the data path may result in a shift of the data waveform to the left, with respect to the clock, resulting in negative hold times.

SWITCHING TIME TEST CIRCUIT

50 ohm termination to ground located in each scope channel input.

All input and output cables to the scope are equal lengths of 50 ohm coaxial cable. Wire length should be <¼ inch from TP_{in} to input pin and TP_{out} to output pin.



RECOMMENDED OPERATING CONDITIONS – MC10800

PARAMETER	SYMBOL	VALUE	UNIT
Supply Voltage (V_{CC} = 0 Volts)	V_{TT} V_{EE}	−1.9 to −2.2 −4.68 to −5.72	Vdc Vdc
Operating Temp. (Functional)	T_A	−30 to +85	°C
Output Drive	–	50Ω to −2.0 Vdc	–
Maximum Clock Input Rise and Fall Time (20% to 80%)	t_r, t_f	10	ns
Minimum Clock Pulse Width	PW	5	ns

ELECTRICAL CHARACTERISTICS

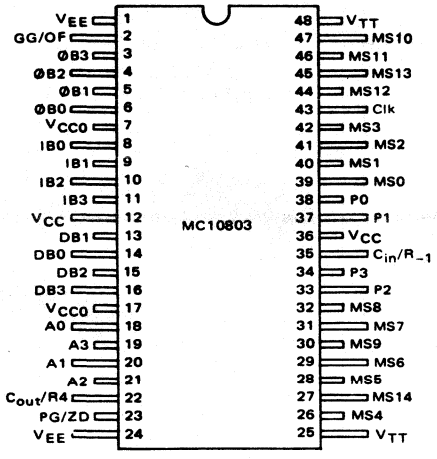
Each MECL 10,000 series circuit has been designed to meet the dc specifications shown in the test table, after thermal equilibrium has been established. The circuit is in a test socket or mounted on a printed circuit board and transverse air flow greater than 500 linear fpm is maintained. Outputs are terminated through a 50-ohm resistor to −2.0 volts. Test procedures are shown for only one input, or for one set of input conditions. Other inputs tested in the same manner.

Characteristic	Symbol	Pin Under Test	TEST LIMITS -30°C Min	-30°C Max	+25°C Min	+25°C Typ	+25°C Max	+85°C Min	+85°C Max	Unit	@ Test Temperature	TEST VOLTAGE VALUES Volts V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{EE}	V_{TT}	(V_{CC}) Gnd
											-30°C	-.890	-1.890	-1.205	-1.500	-5.2	-2	
											+25°C	-.810	-1.85	-1.105	-1.475	-5.2	-2	
											+85°C	-.700	-1.825	-1.035	-1.440	-5.2	-2	
Power Supply Drain Current	I_{EE}	1, 24	–	–	–	209	–	–	–	mAdc	VOLTAGE APPLIED TO PINS LISTED BELOW:	–	–	–	–	1, 24	25, 48	7, 12, 17, 36
	I_{TT}	25, 48	–	–	–	256	–	–	–	mAdc		–	–	–	–	1, 24	25, 48	7, 12, 17, 36
Input Current	I_{inH}	33	–	–	–	–	65	–	–	µAdc		33	–	–	–	1, 24	25, 48	7, 12, 17, 36
		26	–	–	–	–	370	–	–	µAdc		26	–	–	–	1, 24	25, 48	7, 12, 17, 36
		43	–	–	–	–	545	–	–	µAdc		43	–	–	–	1, 24	25, 48	7, 12, 17, 36
	I_{inL}	26	–	–	0.5	–	–	–	–	µAdc		–	26	–	–	1, 24	25, 48	7, 12, 17, 36
Logic "0" Output Voltage	V_{OH}	8	-1.060	-.89	-.960	–	-.810	-.890	-.700	Vdc		14, 40, 42	27, 39, 41	–	–	1, 24	25, 48	7, 12, 17, 36
		23	-1.060	-.89	-.960	–	-.810	-890	-.700	Vdc		*	30, 31, 39	–	–	1, 24	25, 48	7, 12, 17, 36
Logic "1" ** Output Voltage	V_{OL}	8	-1.94	-1.675	-1.90	–	-1.65	-1.875	-1.615	Vdc		40, 42	16, 27, 34, 41	–	–	1, 24	25, 48	7, 12, 17, 36
		23	-1.89	-1.675	-1.85	–	-1.65	-1.825	-1.615	Vdc		*30	31, 39	–	–	1, 24	25, 48	7, 12, 17, 36
Logic "0" Threshold Voltage	V_{OHA}	8	-1.08	–	.980	–	–	-.910	–	Vdc		40, 42	27, 39, 41	14	–	1, 24	25, 48	7, 12, 17, 36
		23	-1.08	–	.980	–	–	-.910	–	Vdc		*	31, 39	–	30	1, 24	25, 48	7, 12, 17, 36
Logic "1" Threshold Voltage	V_{OLA}	8	–	-1.655	–	–	-1.63	–	-1.595	Vdc		40, 42	27, 39, 41	–	14	1, 24	25, 48	7, 12, 17, 36
		23	–	-1.655	–	–	-1.63	–	-1.595	Vdc		*	31, 39	30	–	1, 24	25, 48	7, 12, 17, 36

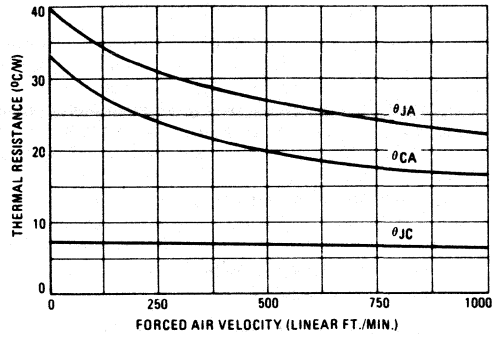
* V_{IH} on pins 3, 4, 5, 6, 29, 32, 35, 40, 41, 42, 46, 47

** The bidirectional outpus are specified at -1.90 volts for V_{OL} min.

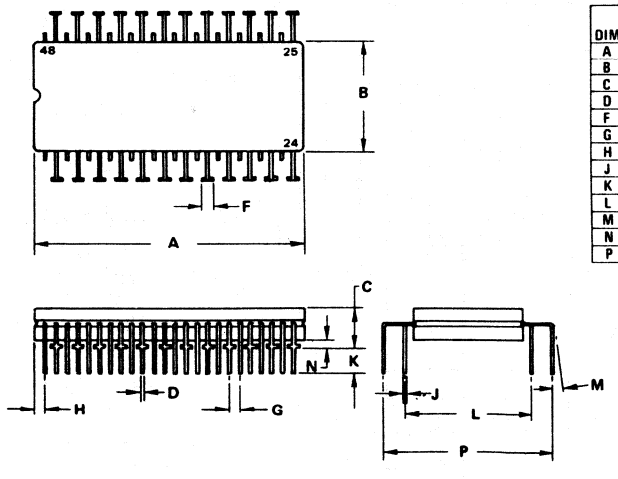
PIN ASSIGNMENT



THERMAL CHARACTERISTICS (TYPICAL)



PACKAGE DIMENSIONS



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	31.24	32.26	1.230	1.270
B	12.70	13.72	0.500	0.540
C	4.57	5.59	0.180	0.220
D	0.38	0.53	0.015	0.021
F	1.14	1.40	0.045	0.055
G	1.27 BSC		0.050 BSC	
H	1.02	1.52	0.040	0.060
J	0.20	0.30	0.008	0.012
K	2.54	3.30	0.100	0.130
L	15.24 BSC		0.600 BSC	
M	–	7°	–	7°
N	0.51	1.52	0.020	0.060
P	20.32 BSC		0.800 BSC	

Case 725-01

A socket for the QUIL package is available from ELECTRONIC MOLDING CORPORATION. (Part number 7178-295-5)

QUIL is a trademark of Motorola Inc.

MOTOROLA

MC10804
MC10805

Advance Information

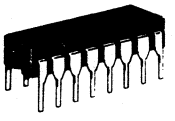
MECL – LSI
ECL/TTL INVERTING
BIDIRECTIONAL
TRANSCEIVERS WITH LATCH

INTRODUCTION

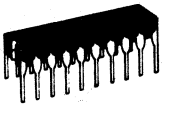
The MC10804 and MC10805 are inverting bidirectional transceivers that interface MECL logic levels with TTL logic levels. Data can be transferred directly in either direction (MECL → TTL or TTL → MECL), and an optional gated latch is also provided. Logic levels are inverted during transfers. The MC10804 is a 4-bit version in the 16-pin package, and the MC10805 is a 5-bit version in the 20-pin package.

The MC10804 and MC10805 are members of the high performance M10800 MECL/LSI processor family. They make it possible to easily interface to MOS memories, TTL compatible peripherals, or existing TTL subsystems.

- Bidirectional Translation
- Power Supplies: +5.0 Volts and -5.2 Volts
- TTL Three-State Outputs
  - Sink 50 mA
  - Source 5 mA
- Standard MECL 50 Ohm Drive Outputs
- Latch – Can Be Bypassed for High Speed
- High Capacitive MOS Drive Capability on MC10805

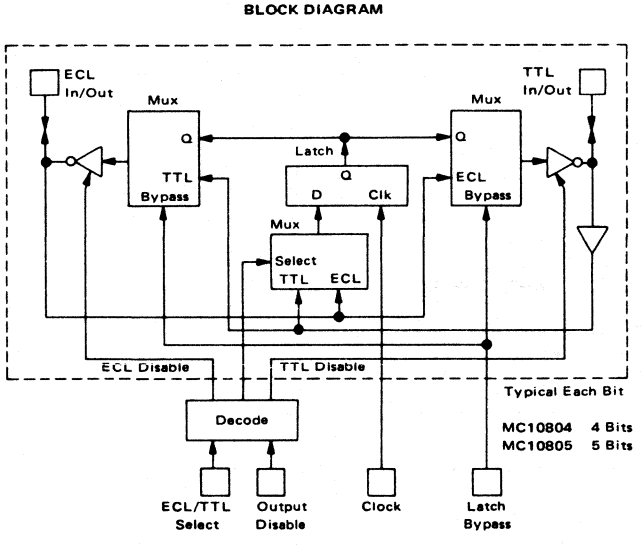


L SUFFIX
CERAMIC PACKAGE
CASE 620
MC10804

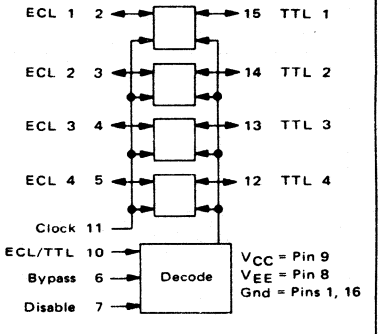


L SUFFIX
CERAMIC PACKAGE
CASE 732
MC10805

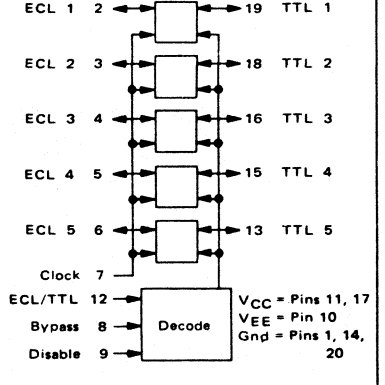
BLOCK DIAGRAM



MC10804



MC10805



This is advance information on a new introduction and specifications are subject to change without notice.

FUNCTIONAL DESCRIPTION

The MC10804 consists of a function decode section, a clock buffer, and four identical bit channels which perform the ECL-TTL translation. Each bit consists of a bidirectional ECL port, a bidirectional TTL port, and a latch. The MC10805 contains the same circuit blocks, but has five instead of four bits translation.

Three logic pins control the function selection. These pins, along with the clock, all operate at standard MECL levels. The block diagram and truth table define the functions. The individual pin descriptions are as follows:

Output Disable

The Output Disable, when at V_{IL}, disables both the ECL and TTL output buffers. That is, both are forced to high-impedance states. When the Output Disable is at V_{IH} the ECL/TTL translation takes place normally, and the appropriate output ports enabled by the ECL/TTL select are active. Regardless of the state of the Output Disable pin, clocked data can be loaded into the latch from the selected input port.

ECL/TTL Select

The ECL/TTL Select pin controls the direction of data transfers. When at V_{IL}, the TTL-to-ECL direction is selected. In this case, the TTL output drivers are disabled, data is input to the latch from the TTL port, and data is output onto the ECL port. When the select pin is at V_{IH}, the ECL-to-TTL direction is selected and the function is the reverse of that just described.

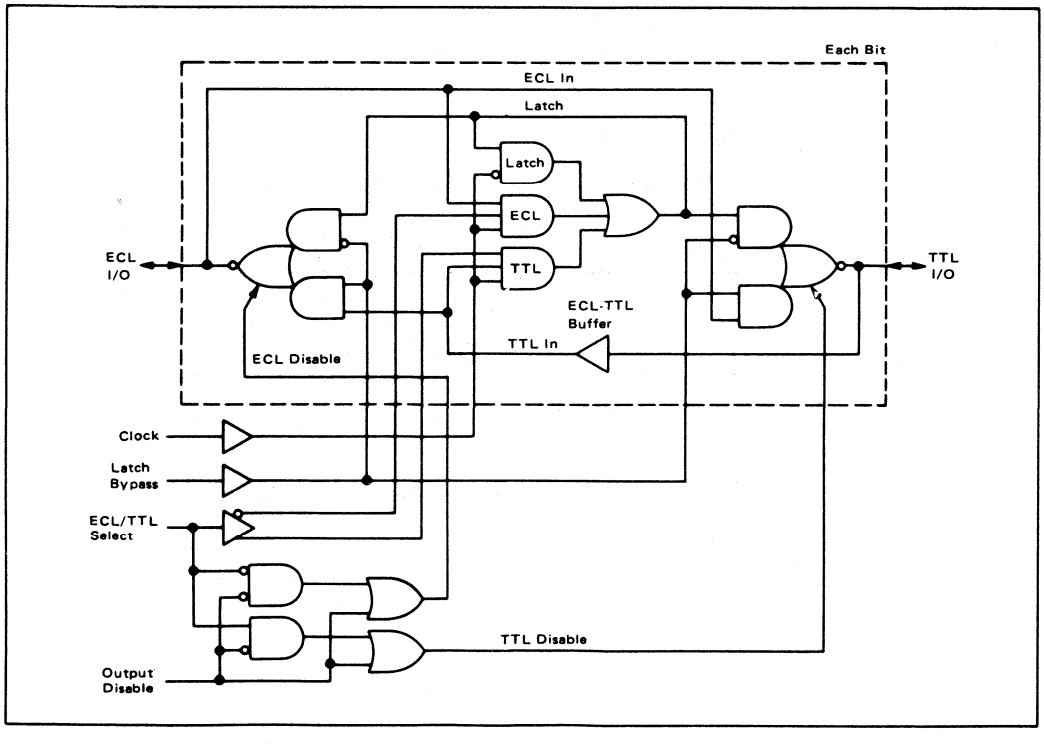
Latch Bypass

The Latch Bypass select line bypasses the latch circuitry for the fast data transfer. When the select line is at V_{IL}, the data is directed to both the latch input and the output buffer simultaneously. This feature enhances the speed of translation because the delay through the latch is bypassed. When the Latch Bypass pin is at V_{IH}, the data must first go into the latch then be sent to the output ports.

Clock

The Clock input is common to all latches and controls the storage of data. When the Clock is at V_{IL} the latch is open and data ripples through from the D input to the Q output. Data is stored or latched on the V_{IL}-to-V_{IH} transition of the Clock input.

NEGATIVE LOGIC DIAGRAM



TRUTH TABLE

SELECT INPUTS (ECL LEVELS, H = -.9V, L = -1.7V)				FUNCTION		
Output Disable	TTL/ECL Select	Latch Bypass	Clock (2)	Latch (1)	TTL I/O (H = 2.4V, L = .5V)	ECL I/O (H = -.9V, L = -1.7V)
H	H	H	H	* Q = H * Q = L	Output = $\overline{Q}$ = L = H	Off Off
H	H	H	L	Q = ECL Input = H L	Output = $\overline{Q}$ = L = H	Input = H = L
H	H	L	H	* *	Output = $\overline{ECL}$ = L = H	Input = H = L
H	H	L	L	Q = ECL Input = H = L	Output = $\overline{ECL}$ = L = H	Input = H = L
H	L	H	H	* Q = H * Q = L	Off Off	Output = $\overline{Q}$ = L = H
H	L	H	L	Q = TTL Input = H = L	Input = H = L	Output = $\overline{Q}$ = L = H
H	L	L	H	* *	Input = H = L	Output = $\overline{TTL}$ = L = H
H	L	L	L	Q = TTL Input = H = L	Input = H = L	Output = $\overline{TTL}$ = L = H
L	H	H	H	*	Off	Off
L	H	H	L	Q = ECL Input = H = L	Off Off	Input = H = L
L	H	L	H	*	Off	Off
L	H	L	L	Q = ECL Input = H = L	Off Off	Input = H = L
L	L	H	H	*	Off	Off
L	L	H	L	Q = TTL Input = H = L	Input = H = L	Off
L	L	L	H	*	Off	Off
L	L	L	L	Q = TTL Input = H = L	Input = H = L	Off

NOTES: (1) * Denotes "NO CHANGE" (2) Latch transfers data when clock is "L" and stores data when clock is "H".

MC10804 SETUP AND HOLD TIMES (NANOSECONDS AT 25°C)

	Setup		Hold	
	Min	Typ	Min	Typ
1. ECL1-4 to clock	–	2.0	–	4.0
2. TTL1-4 to clock	–	3.0	–	3.0
3. ECL/TTL Select to clock	–	4.5	–	1.5

MC10804 PROPAGATION DELAY TIMES (NANOSECONDS AT 25°C)

	Mode	Load	Typical	Max
1. ECL1-4 → TTL1-4	Latch Bypassed	TTL	5.0	–
2. TTL1-4 → ECL1-4	Latch Bypassed		5.5	–
3. ECL1-4 → TTL1-4	Via Latch	TTL	7.5	–
4. TTL1-4 → ECL1-4	Via Latch		8.0	–
5. Latch Bypass → TTL1-4		TTL	7.5	–
6. Latch Bypass → ECL1-4			7.0	–
7. Output Disable → TTL1-4		TTL	8.0	–
8. Output Disable → ECL1-4			7.5	–
9. ECL/TTL Select → TTL1-4		TTL	7.5	–
10. ECL/TTL Select → ECL1-4			7.0	–
11. Clock → TTL1-4		TTL	8.0	–
12. Clock → ECL1-4			7.5	–

MC10805 SETUP AND HOLD TIMES (NANOSECONDS AT 25°C)

	Setup		Hold	
	Min	Typ	Min	Typ
1. ECL1-5	–	2.0	–	4.0
2. TTL1-5	–	3.0	–	3.0
3. ECL/TTL Select	–	4.5	–	1.5

MC10805 PROPAGATION DELAYS (NANOSECONDS AT 25°C)

	Mode	Load	Typical	Max
1. ECL1-5 → TTL1-5	Latch Bypassed	TTL	5.0	–
2. ECL1-5 → TTL1-5	Latch Bypassed	MOS*	14.0	–
3. TTL1-5 → ECL1-5	Latch Bypassed		5.5	–
4. ECL1-5 → TTL1-5	Via Latch	TTL	7.5	–
5. ECL1-5 → TTL1-5	Via Latch	MOS*	16.5	–
6. TTL1-5 → ECL1-5	Via Latch		8.0	–
7. Latch Bypass → TTL1-5		TTL	7.5	–
		MOS*	16.5	
8. Latch Bypass → ECL1-5			7.0	–
9. Output Disable → TTL1-5		TTL	8.0	–
		MOS*	17.0	
10. Output Disable → ECL1-5			7.5	–
11. ECL/TTL Select → TTL1-5		TTL	7.5	–
		MOS*	16.5	
12. ECL/TTL Select → ECL1-5			7.0	–
13. Clock → TTL1-5		TTL	8.0	–
		MOS*	17.0	
14. Clock → ECL1-5			7.5	–

*Load = 150 pF, 413 Ω, 50% to 90%. For other load conditions, see Figure 6.

ELECTRICAL CHARACTERISTICS

Each MECL 10,000 series circuit has been designed to meet the dc specifications shown in the test table, after thermal equilibrium has been established. The circuit is in a test socket or mounted on a printed circuit board and transverse air flow greater than 500 linear fpm is maintained. Outputs are terminated through a 50-ohm resistor to −2.0 volts. Test procedures are shown for only one input, or for one set of input conditions. Other inputs tested in the same manner.

MC10804 RECOMMENDED OPERATING CONDITIONS

PARAMETER	SYMBOL	VALUE	UNIT
Supply Voltage	V_{CC} V_{EE}	+4.5 to +5.5 -4.68 to -5.72	Vdc Vdc
Operating Temp. (Functional)	T_A	-30 to +85	°C
Max Output Drive – ECL	–	50Ω to -2.0 Vdc	–
– TTL	–	V_{CC} = 0.6 V @ 50 mA	–
Maximum Clock Input Rise and Fall Time (20% to 80%)	t_r, t_f	10	ns
Minimum Clock Pulse Width	PW	5	ns

@ Test Temperature	TEST VOLTAGE VALUES (Volts)							
	V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{IHT}	V_{ILT}	V_{CC}	V_{EE}
-30°C	-0.890	-1.890	-1.205	-1.500	+2.0	+0.8	+5.00	-5.20
+25°C	-0.810	-1.850	-1.105	-1.475	+2.0	+0.8	+5.00	-5.20
+85°C	-0.700	-1.825	-1.035	-1.440	+2.0	+0.8	+5.00	-5.20

ELECTRICAL CHARACTERISTICS

Characteristic	Symbol	Pin Under Test	MC10804 TEST LIMITS -30°C Min	-30°C Max	+25°C Min	+25°C Typ	+25°C Max	+85°C Min	+85°C Max	Unit	VOLTAGE APPLIED TO PINS LISTED BELOW V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{IHT}	V_{ILT}	V_{CC}	V_{EE}	Gnd	Output Condition
Negative Power Supply Drain Current	I_{EE}	8	–	–	–	90	–	–	–	mAdc	–	–	–	–	–	–	9	8	1, 16	–
Positive Power Supply Drain Current	I_{CCH}	9	–	–	–	55	–	–	–	mAdc	7, 10	2,3,4,5,6	–	–	–	–	↓	↓	↓	*
	I_{CCL}	9	–	–	–	35	–	–	–	mAdc	–	7	–	–	–	–				–
Input Current	I_{inH}	15	–	–	–	–	45	–	–	µAdc	15	–	–	–	–	–				–
		6	–	–	–	–	350	–	–	µAdc	6	–	–	–	–	–				–
		2	–	–	–	–	485	–	–	µAdc	2	–	–	–	–	–				–
	I_{inL}	6	–	–	0.5	–	–	–	–	µAdc	–	6	–	–	–	–				–
ECL High Output Voltage	V_{OH}	2	-1.060	-.890	-.960	–	-.810	-.890	-.700	Vdc	7, 11	6, 10	–	–	–	15				–
ECL Low Output Voltage	V_{OL}	2	-1.890	-1.675	-1.850	–	-1.650	-1.825	-1.615	Vdc	7, 11	6, 10	–	–	15	–				–
ECL High Threshold Voltage***	V_{OHA}	2	-1.080	–	-.980	–	–	-.910	–	Vdc	7, 11	6	–	10	–	15				–
ECL Low Threshold Voltage	V_{OLA}	2	–	-1.655	–	–	-1.630	–	-1.595	Vdc	7, 11**	10	6	–	–	15				–
ECL Cutoff Voltage	V_{OLZ}	2	–	-1.980	–	–	-1.980	–	-1.980	Vdc	–	7	–	–	–	–				–
TTL High Output Voltage	V_{OHT}	15	+2.500	–	+2.500	–	–	+2.500	–	Vdc	7, 10, 11	2, 6	–	–	–	–				-24 mA
TTL Low Output Voltage	V_{OLT1}	15	–	+.500	–	–	+.500	–	+.500	Vdc	2, 7, 10, 11	6	–	–	–	–				25 mA
	V_{OLT2}	15	–	+.600	–	–	+.600	–	+.600	Vdc	2, 7, 10, 11	6	–	–	–	–				50 mA
TTL High Threshold Voltage***	V_{OHAT}	15	+2.500	–	+2.500	–	–	+2.500	–	Vdc	7, 10	6, 11	–	2	–	–				-24 mA
TTL Low Threshold Voltage	V_{OLAT1}	15	–	+.500	–	–	+.500	–	+.500	Vdc	7, 10	6, 11	2	–	–	–				25 mA
	V_{OLAT2}	15	–	+.600	–	–	+.600	–	+.600	Vdc	7, 10	6, 11	2	–	–	–				50 mA
TTL Cutoff Leakage Current	I_{OHZ}	15	–	-100	–	–	-100	–	-100	µAdc	2	6, 7	–	–	15	–				–
	I_{OLZ}	15	–	-100	–	–	-100	–	-100	µAdc	2	6, 7	–	–	–	15				–
TTL Short-Circuit Current	I_{SC}	15	–	–	–	–	170	–	–	mAdc	7, 10, 11	2, 6	–	–	–	–			1, 15, 16	–

* -5 mA sourced at output pins 12, 13, 14, 15

** Requires the following preset: V_{IH} at pin 7; V_{IL} at pins 6, 10; V_{IHT} at pin 15; then clock once (LH)

*** TTL threshold inputs are the same as V_{IHT} and V_{ILT}

ELECTRICAL CHARACTERISTICS

Each MECL 10,000 series circuit has been designed to meet the dc specifications shown in the test table, after thermal equilibrium has been established. The circuit is in a test socket or mounted on a printed circuit board and transverse air flow greater than 500 linear fpm is maintained. Outputs are terminated through a 50-ohm resistor to −2.0 volts. Test procedures are shown for only one input, or for one set of input conditions. Other inputs tested in the same manner.

MC10805 RECOMMENDED OPERATING CONDITIONS

PARAMETER	SYMBOL	VALUE	UNIT
Supply Voltage	V_{CC} V_{EE}	+4.5 to +5.5 −4.68 to −5.72	Vdc Vdc
Operating Temp. (Functional)	T_A	−30 to +85	°C
Max Output Drive – ECL	–	±50Ω to −2.0 Vdc	–
– TTL_L	–	V_{CC} = 0.6 V @ 50 mA	–
Maximum Clock Input Rise and Fall Time (20% to 80%)	t_r, t_f	10	ns
Minimum Clock Pulse Width	PW	5	ns

*MOS drive is specified by speed vs. C_L curves (MC10805 only)

● Test Temperature	TEST VOLTAGE VALUES (Volts)							
	V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{IHT}	V_{ILT}	V_{CC}	V_{EE}
−30°C	−0.890	−1.890	−1.205	−1.500	+2.0	+0.8	+5.00	−5.20
+25°C	−0.810	−1.850	−1.105	−1.475	+2.0	+0.8	+5.00	−5.20
+85°C	−0.700	−1.825	−1.035	−1.440	+2.0	+0.8	+5.00	−5.20

ELECTRICAL CHARACTERISTICS

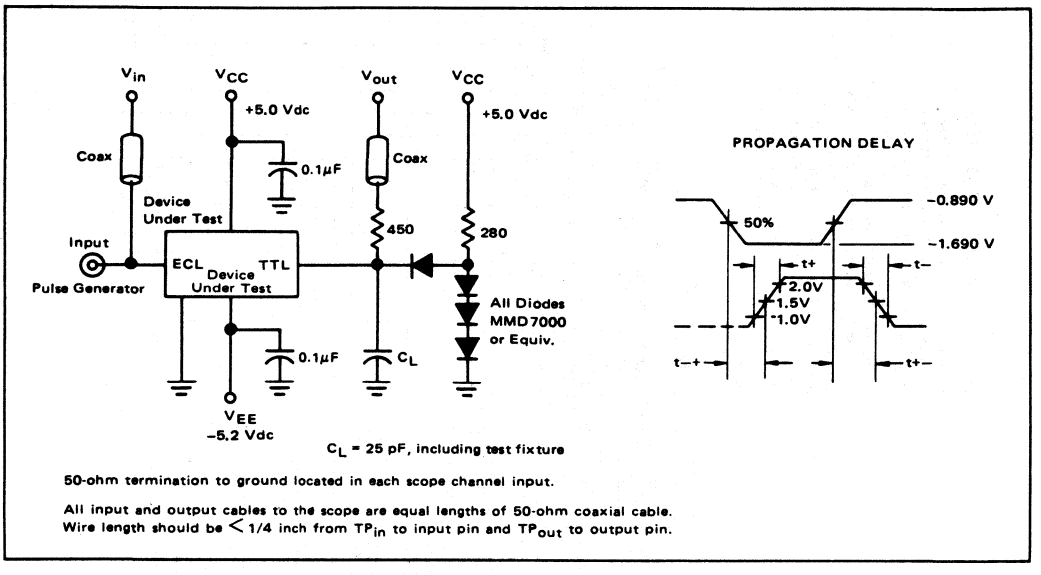
Characteristic	Symbol	Pin Under Test	MC10805 TEST LIMITS −30°C Min	−30°C Max	+25°C Min	+25°C Typ	+25°C Max	+85°C Min	+85°C Max	Unit	VOLTAGE APPLIED TO PINS LISTED BELOW V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{IHT}	V_{ILT}	V_{CC}	V_{EE}	Gnd	Output Condition
Negative Power Supply Drain Current	I_{EE}	10	–	–	–	105	–	–	–	mAdc	–	–	–	–	–	–	11,17	10	1,14 20	–
Positive Power Supply Drain Current	I_{CCH}	11,17	–	–	–	70	–	–	–	mAdc	9,12	2,3,4,5,6,8	–	–	–	–				•
	I_{CCL}	11,17	–	–	–	45	–	–	–	mAdc	–	9	–	–	–	–				–
Input Current	I_{inH}	19	–	–	–	–	45	–	–	µAdc	19	–	–	–	–	–				–
		8	–	–	–	–	350	–	–	µAdc	8	–	–	–	–	–				–
		2	–	–	–	–	485	–	–	µAdc	2	–	–	–	–	–				–
	I_{inL}	8	–	–	0.5	–	–	–	–	µAdc	–	8	–	–	–	–				–
ECL High Output Voltage	V_{OH}	2	-1.060	-.890	-.960	–	-.810	-.890	-.700	Vdc	7,9	8,12	–	–	–	19				–
ECL Low Output Voltage	V_{OL}	2	-1.890	-1.675	-1.850	–	-1.650	-1.825	-1.615	Vdc	7,9	8,12	–	–	19	–				–
ECL High Threshold Voltage***	V_{OHA}	2	-1.080	–	-.980	–	–	-.910	–	Vdc	7,9	8	–	12	–	19				–
ECL Low Threshold Voltage	V_{OLA}	2	–	-1.655	–	–	-1.630	–	-1.595	Vdc	7,9**	12	8	–	–	19				–
ECL Cutoff Voltage	V_{OLZ}	2	–	-1.980	–	–	-1.980	–	-1.980	Vdc	–	9	–	–	–	–				–
TTL High Output Voltage	V_{OHT}	19	+2.500	–	+2.500	–	–	+2.500	–	Vdc	7,9,12	2,8	–	–	–	–				-24 mA
TTL Low Output Voltage	V_{OLT1}	19	–	+.500	–	–	+.500	–	+.500	Vdc	2,7,9,12	8	–	–	–	–				25 mA
	V_{OLT2}	19	–	+.600	–	–	+.600	–	+.600	Vdc	2,7,9,12	8	–	–	–	–				50 mA
TTL High Threshold Voltage***	V_{OHAT}	19	+2.500	–	+2.500	–	–	+2.500	–	Vdc	9,12	7,8	–	2	–	–				-24 mA
TTL Low Threshold Voltage	V_{OLAT1}	19	–	+.500	–	–	+.500	–	+.500	Vdc	9,12	7,8	2	–	–	–				25 mA
	V_{OLAT2}	19	–	+.600	–	–	+.600	–	+.600	Vdc	9,12	7,8	2	–	–	–				50 mA
TTL Cutoff Leakage Current	I_{OHZ}	19	–	-100	–	–	-100	–	-100	µAdc	2	8,9	–	–	19	–				–
	I_{OLZ}	19	–	100	–	–	100	–	100	µAdc	2	8,9	–	–	–	19				–
TTL Short-Circuit Current	I_{SC}	19	–	–	–	–	170	–	–	mAdc	7,9,12	2,8	–	–	–	–			1,14 19,20	–

* −5 mA sourced at output pins 13, 15, 16, 18, 19

** Requires the following preset: V_{IH} at pin 9; V_{IL} at pins 8, 12; V_{IHT} at pin 19; then clock once (⊔)

*** TTL threshold inputs are the same as V_{IHT} and V_{ILT}

FIGURE 1 – SWITCHING TIME TEST CIRCUIT AND WAVEFORMS @ 25°C FOR PROPAGATION DELAY FROM MECL INPUT TO TTL OUTPUT WITH TTL LOAD

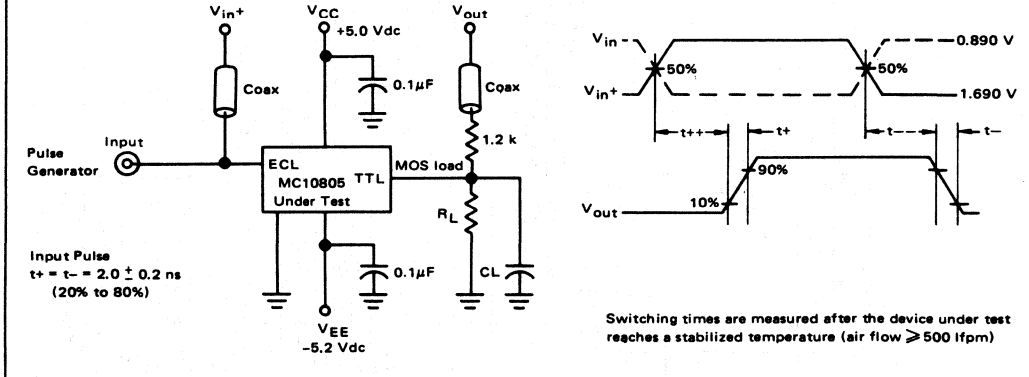


C_L = 25 pF, including test fixture

50-ohm termination to ground located in each scope channel input.

All input and output cables to the scope are equal lengths of 50-ohm coaxial cable. Wire length should be < 1/4 inch from TP_{in} to input pin and TP_{out} to output pin.

FIGURE 2 – SWITCHING TIME TEST CIRCUIT AND WAVEFORMS @ 25°C FOR PROPAGATION DELAY FROM MECL INPUT TO TTL OUTPUT WITH MOS LOAD

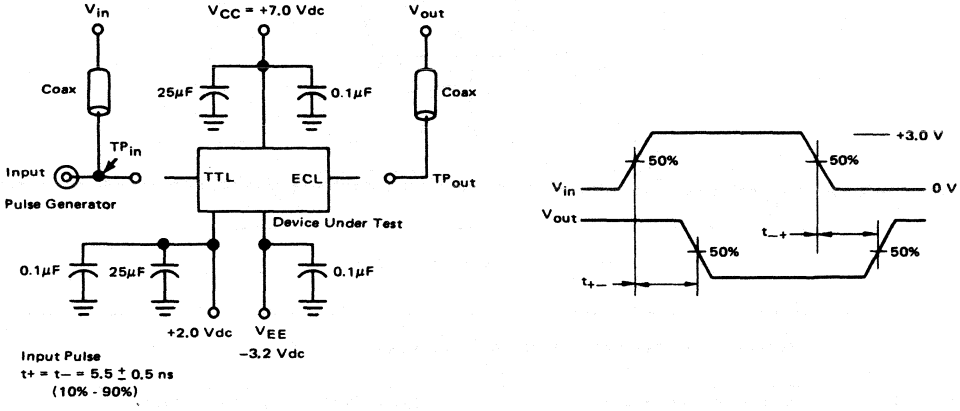


Switching times are measured after the device under test reaches a stabilized temperature (air flow ≥ 500 lfpm)

50-ohm termination to ground located in each scope channel input.

All input and output cables to the scope are equal lengths of 50-ohm coaxial cable. Wire length should be < 1/4 inch from TP_{in} to input pin and TP_{out} to output pin.

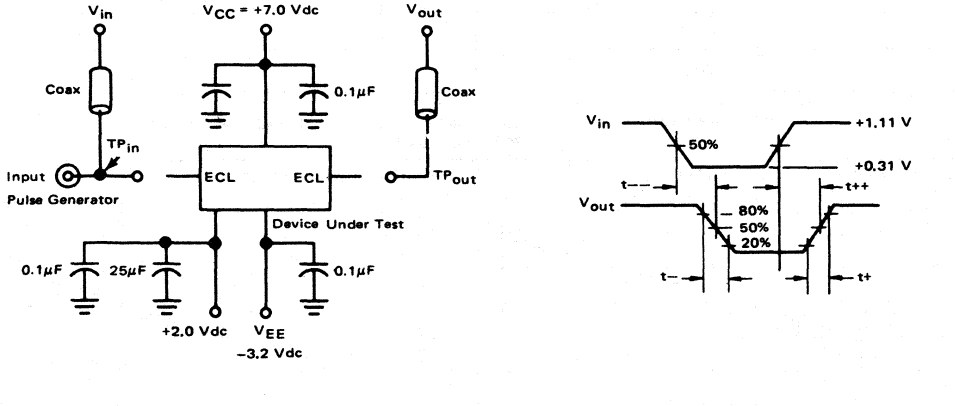
FIGURE 3 – SWITCHING TIME TEST CIRCUIT AND WAVEFORMS @ 25°C FOR PROPAGATION DELAY FROM TTL INPUT TO ECL OUTPUT



50-ohm termination to ground located in each scope channel input.

All input and output cables to the scope are equal lengths of 50-ohm coaxial cable.
Wire length should be < 1/4 inch from TP_{in} to input pin and TP_{out} to output pin.

FIGURE 4 – SWITCHING TIME TEST CIRCUIT AND WAVEFORMS @ 25°C FOR PROPAGATION DELAY FROM ECL SELECT INPUT TO ECL OUTPUT



50-ohm termination to ground located in each scope channel input.

All input and output cables to the scope are equal lengths of 50-ohm coaxial cable.
Wire length should be < 1/4 inch from TP_{in} to input pin and TP_{out} to output pin.

FIGURE 5 – SETUP AND HOLD TIME WAVEFORMS

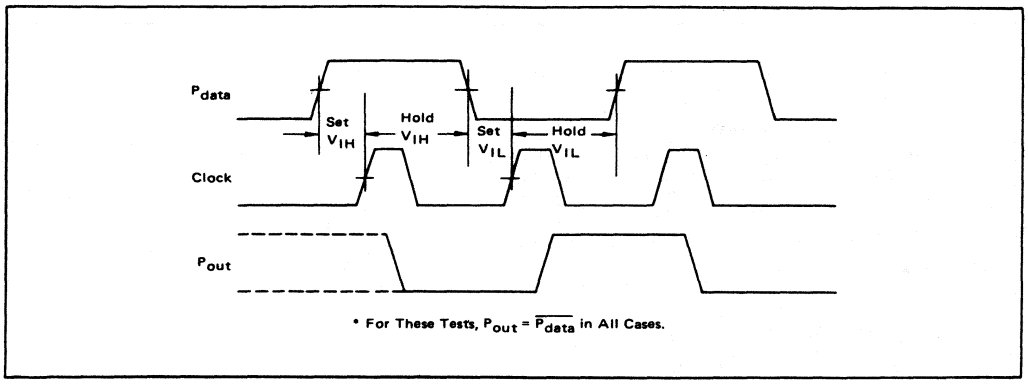


FIGURE 6 – MC10805 ECL → TTL DELAY (Latch Bypassed) versus CAPACITIVE LOAD (T_A = +25°C, V_{CC} = 5.0 V)

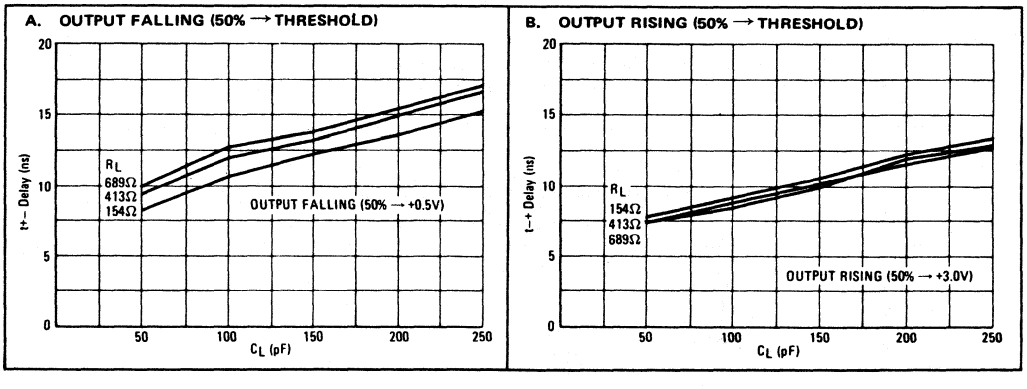
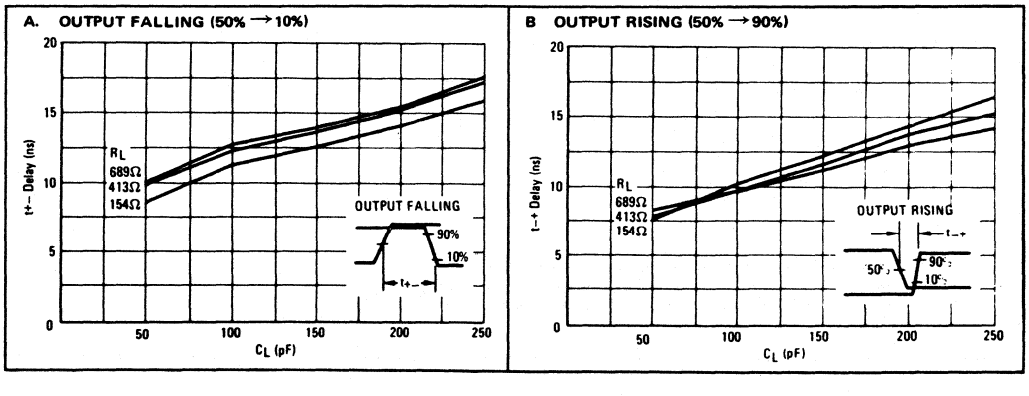
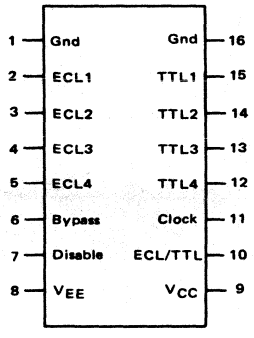


FIGURE 7 – MC10805 ECL → TTL DELAY (Latch Bypassed) versus CAPACITIVE LOAD (T_A = +25°C, V_{CC} = 5.0 V)

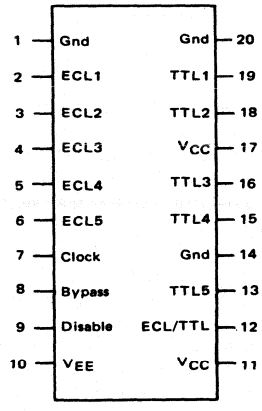


PIN ASSIGNMENTS

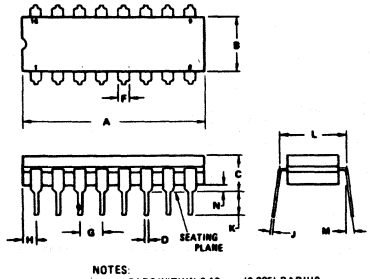
MC10804



MC10805



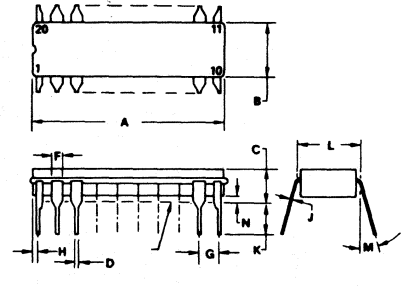
PACKAGE DIMENSIONS



NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION
2. PKG. INDEX: NOTCH IN LEAD NOTCH IN CERAMIC OR INK DOT
3. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	19.05	19.81	0.750	0.780
B	6.22	6.98	0.245	0.275
C	4.06	5.08	0.160	0.200
D	0.38	0.51	0.015	0.020
F	1.40	1.65	0.055	0.065
G	2.54 BSC		0.100 BSC	
H	0.51	1.14	0.020	0.045
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	7.37	7.87	0.290	0.310
M	–	15°	–	15°
N	0.51	1.02	0.020	0.040

CASE 620-02



NOTES:
1. LEADS WITHIN 0.25 mm (0.010) DIA , TRUE POSITION AT SEATING PLANE, AT MAXIMUM MATERIAL CONDITION.
2. DIM L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIM A AND B INCLUDES MENISCUS.

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	24.38	25.15	0.960	0.990
B	6.86	7.49	0.270	0.295
C	4.32	5.08	0.170	0.200
D	0.38	0.56	0.015	0.022
F	1.40	1.65	0.055	0.065
G	2.54 BSC		0.100 BSC	
H	0.89	1.40	0.035	0.055
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	7.62 BSC		0.300 BSC	
M	5°	15°	5°	15°
N	0.51	0.76	0.020	0.030

CASE 732-02

MOTOROLA

MC10806

Advance Information

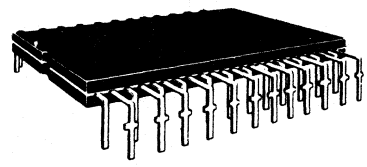
INTRODUCTION

The MC10806 Dual Access Stack is an LSI building block for digital processor systems. This circuit consists of 32 words by 9 bits of memory with two independent address and data ports. The circuit is easily expandable in both the word and bit directions making it ideal in register file, scratch pad, and high-speed buffer applications.

The Dual Access Stack, as shown in the block diagram below, contains a 32 x 9 memory array, two address ports, two 9-bit data input/output ports, two 9-bit output registers, address and data parity checking logic, and two error flip-flops in a single MECL Bipolar LSI circuit. Separate read, write, and output enables exist for each port to control all operations within the part.

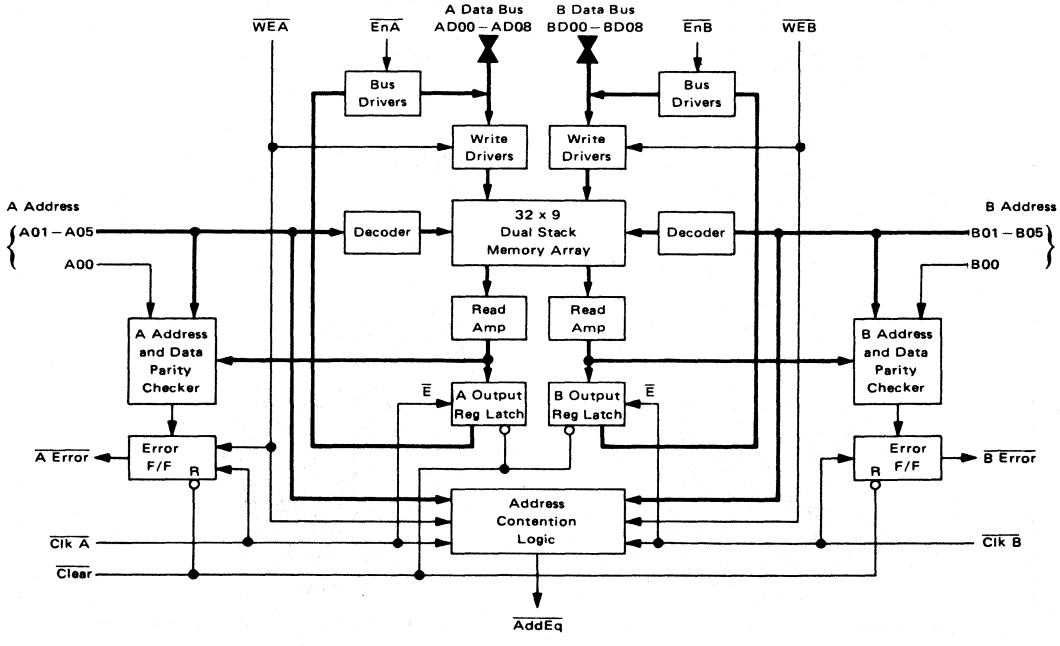
MECL — LSI

DUAL ACCESS STACK



CASE 725-01

DUAL ACCESS STACK BLOCK DIAGRAM – MC10806



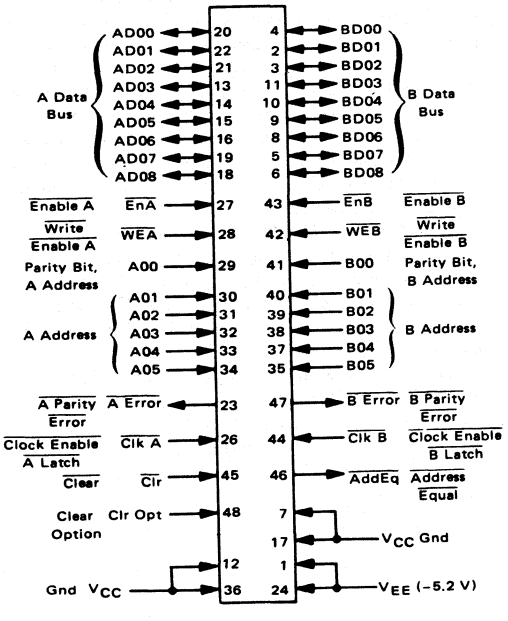
This is advance information on a new introduction and specifications are subject to change without notice.

TABLE OF CONTENTS

IMPORTANT FEATURES

1. 32 x 9 Memory Array
2. Two 9-Bit Output Registers (Latches)
3. Two Independent Address Ports
4. Two Data I/O Ports
5. Address and Data Parity Checking Logic
6. Two Master/Slave Error Flip-Flops
7. Separate Read, Write, and Output Enables for Each Part
8. Each Part is 9-Bits Wide (One Byte) and Can Be Operated in Parallel to Form Any Word Size in Increments of 9 Bits
9. Fully Compatible with the MECL 10,000 Family

INPUT/OUTPUT DIAGRAM



ABSOLUTE MAXIMUM RATINGS (See Note 1)

Rating		Symbol	Value	Unit
Supply Voltage (V_{CC} = 0)		V_{EE}	-8 to 0	Vdc
Input Voltage (V_{CC} = 0)	Std	V_{in}	0 to V_{EE}	Vdc
	Bus	V_{in}	Note 2	Vdc
Output Source Current	Cont	I_{out}	< 50	mAdc
	Surge	I_{out}	< 100	mAdc
Storage Temperature		T_{stg}	-55 to +150	°C
Junction Temperature		T_J	165	°C

NOTES: 1. Permanent device damage may occur, if absolute maximum ratings are exceeded. Functional operation should be restricted to RECOMMENDED OPERATING CONDITIONS. Exposure to higher than recommended voltages for extended periods of time could affect device reliability.

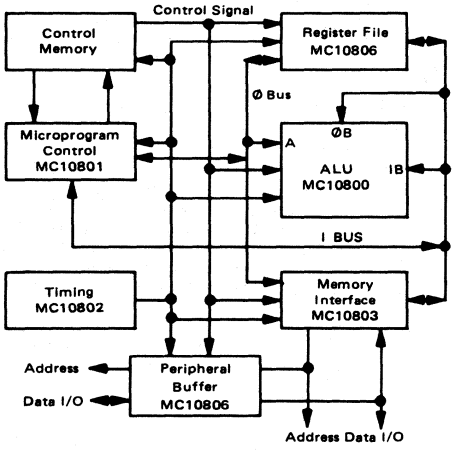
2. Input voltage limit is V_{CC} to -2 Volts when the bus is used as an input and the output drivers are disabled.

SYSTEM OVERVIEW

The Motorola M10800 family of LSI processor circuits combines the cost and size advantages of LSI with system design flexibility. Each family part is a major system building block which can be interconnected and programmed for a wide range of processor applications. Figure 1 illustrates a method of using the various circuits in a general-purpose processor. The MC10800 4-Bit ALU Slice performs the various arithmetic, logic, and shift functions. This circuit features full BCD capability and a complete set of status outputs. The MC10801 Microprogram Control Function addresses and sequences through microprogram control memory. A set of 16 control instructions provides for direct jumps, conditional branches, and subroutines within microprogram. The MC10802 Timing Function generates clock phases and features single-cycle or single-phase clock increment for troubleshooting or diagnostics.

The Register File has been made a separate block so that the designer can optimize the size and configuration for his particular system. The main function of the Register File is to provide storage for addresses and data. Also, the access time of the Register File must be fast in order to efficiently utilize the high speed of the overall processor system.

FIGURE 1 – MICROPROGRAMMED PROCESSOR



The MC10806 Dual Access Stack provides the register-file function in the processor as well as providing a memory buffer interface to peripheral devices. The MC10806 contains 32 words by 9 bits of memory in which 2 words can be independently addressed for read or write operations on two separate data I/O ports. Also, the circuit has the ability to check for parity errors on both the address and the data.

In the Register File block of the processor configuration shown in Figure 1, the two Data I/O ports of the MC10806 are connected to the two internal busses of the processor, the I Bus and the Ø Bus. The addresses and control signals are connected to the Control Memory. In the configuration shown, two locations of register file can be operated on by the ALU when the clock is at "0" with the result placed on the I Bus when the block is at "1", so that it may be written back into the register file in the same microcycle. More details for this configuration are given in the Application section.

In the Peripheral Buffer block, the MC10806 can be used as a temporary buffer for storing data from the processor to the peripheral device or vice versa. One Data I/O and Address port of the MC10806 is connected to the Data I/O and Address port of the MC10803. The other Data I/O and Address port are controlled by the peripheral device independent of the processor. In this application, the processor can read and write data into the peripheral buffer at high speed using the MC10803, while the peripheral device can read and write into buffer at a slower or faster speed independent of the processor. Flag status and interrupt conditions could also be designed into the peripheral interface depending on the application requirements.

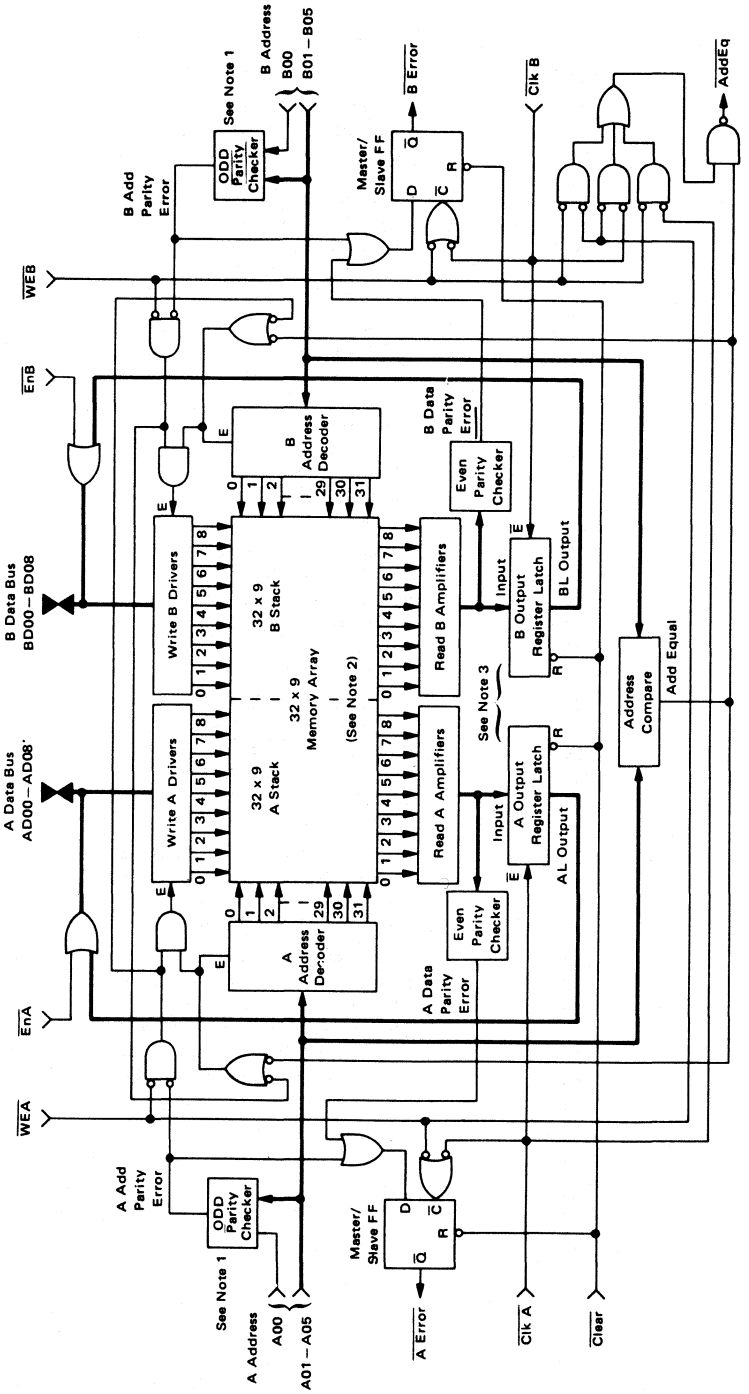
The Motorola M10800 circuits interface directly to all parts in the MECL 10,000 family. This provides a source for high-speed memories and a complete mix of MSI and SSI circuits. Circuits are available for special hardware functions from high-speed multiply to error detection and correction.

Versatility is a key word to describe each circuit in the Motorola M10800 family. The block diagram in Figure 1 and the examples in this data sheet are intended to illustrate ways to use these LSI parts and do not restrict the designer to any particular system configuration or application.

PIN ASSIGNMENTS

Pin Designation	Pin Number	Description
AD00	20	A Data Bus – Bit 0 I/O
AD01	22	A Data Bus – Bit 1 I/O
AD02	21	A Data Bus – Bit 2 I/O
AD03	13	A Data Bus – Bit 3 I/O
AD04	14	A Data Bus – Bit 4 I/O
AD05	15	A Data Bus – Bit 5 I/O
AD06	16	A Data Bus – Bit 6 I/O
AD07	19	A Data Bus – Bit 7 I/O
AD08	18	A Data Bus – Bit 8 I/O
BD00	4	B Data Bus – Bit 0 I/O
BD01	2	B Data Bus – Bit 1 I/O
BD02	3	B Data Bus – Bit 2 I/O
BD03	11	B Data Bus – Bit 3 I/O
BD04	10	B Data Bus – Bit 4 I/O
BD05	9	B Data Bus – Bit 5 I/O
BD06	8	B Data Bus – Bit 6 I/O
BD07	5	B Data Bus – Bit 7 I/O
BD08	6	B Data Bus – Bit 8 I/O
A00	29	A Address – Bit 0 Parity Bit Input
A01	30	A Address – Bit 1 LSB Input
A02	31	A Address – Bit 2 NLSB Input
A03	32	A Address – Bit 3 NLSB Input
A04	33	A Address – Bit 4 NMSB Input
A05	34	A Address – Bit 5 MSB Input
B00	41	B Address – Bit 0 Parity Input
B01	40	B Address – Bit 1 LSB Input
B02	39	B Address – Bit 2 NLSB Input
B03	38	B Address – Bit 3 NLSB Input
B04	37	B Address – Bit 4 NMSB Input
B05	35	B Address – Bit 5 MSB Input
$\overline{\text{EnA}}$	27	Enable A Register Latch to A Data Bus
$\overline{\text{EnB}}$	43	Enable B Register Latch to B Data Bus
$\overline{\text{WEA}}$	28	Write Enable A Data Bus to Memory
$\overline{\text{WEB}}$	42	Write Enable B Data Bus to Memory
$\overline{\text{AError}}$	23	A Address or Data Parity Error Output
$\overline{\text{BError}}$	47	B Address or Data Parity Error Output
$\overline{\text{Clk A}}$	26	Clock Enable A Register Latch Input
$\overline{\text{Clk B}}$	44	Clock Enable B Register Latch Input
$\overline{\text{Clr}}$	45	Clear A and B Error F/Fs and Register Latches
$\overline{\text{AddEq}}$	46	A and B Addresses are Equal and Error condition may exist
Clr Opt	48	Clear Option Input
V_{EE}	1	-5.2 Volt Supply
V_{EE}	24	-5.2 Volt Supply
V_{CC}	12	Ground
V_{CC}	36	Ground
V_{CCO}	7	Ground
V_{CCO}	17	Ground

FIGURE 2 – DETAILED MC10806 FUNCTION BLOCK DIAGRAM (NEGATIVE LOGIC)



NOTES: 1. If A00 is tied to V_{EE}, the A Address Parity Checker will be disabled.
If B00 is tied to V_{EE}, the B Address Parity Checker will be disabled.
2. Stack A and Stack B contain the same information; Data written into Stack A is automatically written into Stack B and vice versa.
3. If Pin 48 is left floating, the A and B Output Registers will be initialized to 0 1111 1111 when $\overline{Clear}$ = 0. If Pin 48 is tied to -2 V (V_{TT}), then the A and B Output Registers will be cleared to all 0s.

ARCHITECTURAL DESCRIPTION

The MC10806 Dual Access Stack as shown in Figure 2 contains a memory array of 32 words by 9 bits that can be read or written via two Address and two Data I/O ports. An output register, read/write control lines, an output enable, and a parity error output (via master/slave flip-flop) exists for each port. In addition, a Clear Input initializes the Parity Error flip-flops and Output Registers. Also, an Address Equal Output is available that indicates when the A and B Addresses are equal and the Write Enable of one port is active concurrently with either the Write Enable or the output latch register clock of the other port.

MEMORY ARRAY

The Memory Array actually contains two stacks with each containing 32 words by 9 bits. Each stack can be addressed independent of the other stack. When information is written into one stack, it is automatically written into the other stack at the same location. Thus, if Stack A is written via the A Bus into all address locations, Stack B will contain identical information. The advantage of this configuration is that the memory can be used as 32 9-bit registers in which reading and writing may be performed using two independent address and data ports.

One port could be writing into one stack location while the other port is reading from another stack location. Also, both ports could be reading or writing simultaneously at different stack locations. However, it is illegal to try to write into the same address from both ports simultaneously. As shown in Figure 2, the A Address Decoder and Write A Drivers are inhibited if the addresses are equal and the Write Enable is active ($\overline{WEB}$ = 0) on the B port. The reason for inhibiting the A Address Decoder is that the word line in the A Stack, corresponding to word line being enabled in the B Stack when writing, must be inhibited in order that the word can be duplicated in the A Stack.

OUTPUT REGISTER LATCH

An Output Register Latch exists for each data port which can be used as a temporary storage register which can be enabled onto the Data Bus at any time. Information in the memory can be read out at any time at the address specified by activating the Clock to the Latch Register ($\overline{Clk}$ A or B = 0). Information in the Register is latched when the Clock is deactivated ($\overline{Clk}$ A or B = 1).

ADDRESS CONTENTION

The $\overline{AddEq}$ output is activated whenever erroneous data is read out of memory or if an illegal write condition occurs. The illegal write condition occurs when writing into the same address from both ports simultaneously. Erroneous data can be read out of memory, if the addresses are equal when writing into one port and activating the clock of the Output Register of the other port. This is due to the inhibiting of the word line (as described earlier) which causes all 1s to be read into the latch. For these reasons the $\overline{AddEq}$ output goes to a "0", if the A and B addresses are equal and the Write Enable of one port is active concurrently with either the Write Enable or the Output Latch Register Clock of the other port.

PARITY CHECKING LOGIC

A master-slave flip-flop on each port is used to check address and data parity errors. The Error output goes to a 0 ($\overline{A\ Error}$ = 0 or $\overline{B\ Error}$ = 0), if there is a parity error when a 0 to 1 transition occurs on the clock (clock to A Error F/F = $\overline{WEA} \cdot \overline{ClkA}$ or the clock to B Error F/F = $\overline{WEB} \cdot \overline{ClkB}$). The Error output goes to a 1, if there is no parity error when a 0 to 1 transition occurs on the clock.

If an address parity error occurs, the contents of the selected address cannot be changed, when writing, in order to facilitate error recovery. The Address Parity Checker can be disabled by connecting the Address Parity Input (A00, B00) to the V_{EE} supply. The data that is read or written into memory is checked for even parity (an even number of 1s must exist in the 9-bit word), if negative logic is used, or odd parity, if positive logic is used. If the parity checking is not required, then a word in memory can consist of 9 bits of data with the Parity Error output ignored.

CLOCK

The clock to the A Error F/F (B Error F/F) is the AND function of the $\overline{WEA}$ ($\overline{WEB}$) and $\overline{ClkA}$ ($\overline{ClkB}$) inputs. The Error F/Fs are master-slave and trigger on a 0 to 1 transition (V_{OH} to V_{OL}) of the clock as defined above.

The A output Register (B Output Register) consists of latches with the $\overline{ClkA}$ ($\overline{ClkB}$) input controlling the clocking. When $\overline{ClkA}$ = 0 ($\overline{ClkB}$ = 0), the data addressed from memory is enabled to the output of the register. When $\overline{ClkA}$ = 1 ($\overline{ClkB}$ = 1), the data in the register is in the latched condition and cannot change.

$\overline{CLEAR}$

The $\overline{Clear}$ input, when a 0, is used to asynchronously reset the error flip-flops and the output registers. This causes the $\overline{A\ Error}$ and $\overline{B\ Error}$ outputs to go to a 1. The output registers are cleared to all 0s if pin 48 is tied to -2 V. If pin 48 is left open, the output register is cleared to 0 1111 1111 (the 0 corresponds to the BIT 00 position in the word).

FUNCTIONAL DESCRIPTION

The MC10806 Dual Access Stack (DAS) can be completely described by the truth tables shown in Tables 1 through 8, together with the block diagram (Figure 2) and the switching waveforms shown in the Electrical Parameters. **All truth tables are expressed in negative logic with V_{OL} being a logic 1 and V_{OH} a logic 0.**

NO OPERATION

Writing into the memory of the DAS is inhibited, if the write enable is not active ($\overline{WEA}$ = 1, $\overline{WEB}$ = 1). Reading from the memory to the output register is inhibited (contents of output register remains unchanged), if the clock is not active ($\overline{ClkA}$ = 1, $\overline{Clk\ B}$ = 1). Also, reading the contents of the Output Register to the Data Bus is inhibited, if the enable line is not active ($\overline{EnA}$ = 1, $\overline{EnB}$ = 1). However, even if no operation is being performed internally the Read Amplifiers will read the data from memory as addressed by the address lines (A01 – A05, B01 – B05).

READ OPERATION

There are three modes in which data can be read onto the data bus as shown in the switching waveforms. In the enable access mode (see Table 7), the activation of the enable line ($\overline{EnA}$ = 0, $\overline{EnB}$ = 0) transfers the contents of the 9-bit Output Register onto the bidirectional data bus. Normally, the contents of the Output Register have been previously loaded from memory making the enable access time very fast.

In the address access mode, the Address lines select the memory location to be accessed and the data appears at the Data Bus after the specified propagation delay has occurred. This assumes that the clock ($\overline{ClkA}$ = 0, $\overline{ClkB}$ = 0) and Enable ($\overline{EnA}$ = 0, $\overline{EnB}$ = 0) lines are both active.

In the third mode for reading data, the activation of the clock (with the addressed location having been set up previously) causes data to appear at the Data Bus (if $\overline{EnA}$ = 0, $\overline{EnB}$ = 0) after the specified propagation delay. Deactivating the clock (0 to 1 transition of $\overline{ClkA}$, $\overline{ClkB}$) causes the data to be latched in the Output Register (see Tables 5 and 6) and the Parity Error flip-flop to be activated ($\overline{A\ Error}$ = 0, $\overline{B\ Error}$ = 0), if there is an address or data parity error (see Tables 2, 3, and 4). When latching the data in the output register by deactivating the clock, the address lines must meet the setup and hold times that are specified.

WRITE OPERATION

The switching waveforms should be referred to in the following description of the write mode of operation. The Enable line, for reading data onto the data bus, must be deactivated ($\overline{EnA}$ = 1, $\overline{EnB}$ = 1) when writing data coming from external sources. However, the Enable line could be activated, if the source for writing data is the internal Output Register. The address should be set up prior to activating the Write Enable line in order that the parity of the address can be checked for error so that writing can be inhibited, if an error occurs (see Table 1). Information on the Data Bus is written into the memory location when the Write Enable is activated ($\overline{WEA}$ = 0, $\overline{WEB}$ = 0). The information must be valid during the setup and hold times referenced to the deactivation of the Write Enable (0 to 1 transition of $\overline{WEA}$, $\overline{WEB}$). Also, the deactivation of the Write Enable causes the Parity Error flip-flop to be activated, if there is an Address or Data Parity Error (see Tables 2, 3, and 4). Note that it is possible to read the data that is being written, into the Output Register by activating the Clock Enable of the same port being written.

PARITY CHECKING

The truth tables for the Parity Error flip-flops are shown in Tables 2, 3, and 4. A description of the parity checking logic is described in the Architectural Description. The Error flip-flops are in the activated state when a logic 0 so that they could be tied together forming the wired-AND function.

INITIALIZATION

Initializing the Error flip-flops and the Output Registers can be accomplished asynchronously by momentarily placing the Clear at a logic 0. Initialization of the Error flip-flops is included in Tables 3 and 4, while Tables 5 and 6 show the initialization states of the Output Registers.

CONFLICTS

An illegal write condition occurs for the dual write condition where both ports are writing into the same address simultaneously. It is the user's responsibility to ensure that writing into both ports simultaneously at the same address does not occur. A write on port A (when Address A = B) inhibits a selection of port B, and a write on port B inhibits a selection on port A. Thus, in a dual write case, theoretically neither writes as shown in Table 1. However, the practical case considers that one port's control signal will occur slightly before the other; thus, the contents of the equal location cannot be guaranteed.

Another possible conflict can occur when writing into one port and activating the Clock of the Output Register of the other port. The internal logic forces all 1s into the Output Register as shown in Table 1. However, if the Clock is held activated a sufficient amount of time after the Write Enable pulse, the contents of the Output

Register will be corrected to the information that was written into memory and a parity error will not occur due to the conflict. A parity error would occur, if the Clock is deactivated while all 1s were being forced into the Output Register, since all 1s is an odd parity condition.

The $\overline{\text{AddEq}}$ output is activated ($\overline{\text{AddEq}}$ = 0) whenever the two possible conflict conditions exist. A truth table is shown in Table 8 listing the various conflict conditions.

TABLE 1 – TRUTH TABLE FOR MEMORY ARRAY AND READ AMPLIFIER OUTPUTS

$\overline{\text{WEA}}$	(See Table 2a) A Address Parity Error	$\overline{\text{WEB}}$	(See Table 2b) B Address Parity Error	Address Equal Condition A01-05=B01-05	Memory Array Contents @ Address A	Memory Array Contents @ Address B	Read A Amp Output	Read B Amp Output	Comments
1	X	1	X	X	–	–	MA00-MA08	MB00-MB08	Read A; Read B
0	0	1	X	0	AD00-AD08	–	AD00-AD08	MB00-MB08	Write A; Read B
	0	1	X	1	AD00-AD08	AD00-AD08	AD00-AD08	All 1s	Write A; Read B Amp = All 1s
0	1	1	X	X	–	–	MA00-MA08	MB00-MB08	Parity Error, No Write A; Read B
1	X	0	0	0	–	BD00-BD08	MA00-MA08	BD00-BD08	Read A; Write B
1	X	0	0	1	BD00-BD08	BD00-BD08	All 1s	BD00-BD08	Read A Amp = All 1s; Write B
1	X	0	1	X	–	–	MA00-MA08	MB00-MB08	Read A; Parity Error, No Write B
0	0	0	0	0	AD00-AD08	BD00-BD08	AD00-AD08	BD00-BD08	Write A; Write B
0	1	0	0	0	–	BD00-BD08	MA00-MA08	BD00-MB08	Parity Error, No Write A; Write B
0	0	0	1	0	AD00-AD08	–	AD00-AD08	MB00-MB08	Write A; Parity Error, No Write B
0	1	0	1	X	–	–	MA00-MA08	MB00-MB08	Parity Error, No Write A; Parity Error, No Write B
0	0	0	0	1	–	–	All 1s	All 1s	Writing is Inhibited
0	1	0	0	1	BD00-BD08	BD00-BD08	All 1s	BD00-BD08	Parity Error, No Write A; Write B
0	0	0	1	1	AD00-AD08	AD00-AD08	AD00-AD08	All 1s	Write A; Parity Error, No Write B

X = Don't Care State – = No Change

NOTE: MA00-MA08 represents the data in the Memory Array at the location addressed by A01-A05. MB00-MB08 represents the data in the Memory Array at the location addressed by B01-B05.

TABLE 2 – TRUTH TABLE FOR ADDRESS PARITY ERROR

a. A Address Parity Error

Parity Bit A00	A Address A01-A05	A Address Parity Error
V_{EE}	X	0
0	Odd Number of 1s	0
1	Odd Number of 1s	1
0	Even Number of 1s	1
1	Even Number of 1s	0

b. B Address Parity Error

Parity Bit B00	B Address B01-B05	B Address Parity Error
V_{EE}	X	0
0	Odd Number of 1s	0
1	Odd Number of 1s	1
0	Even Number of 1s	1
1	Even Number of 1s	0

X = Don't Care Condition

TABLE 3 – TRUTH TABLE FOR $\overline{\text{A ERROR}}$ (OUTPUT OF MASTER/SLAVE FF)

$\overline{\text{Clear}}$	$\overline{\text{Clk A}}$	$\overline{\text{WEA}}$	A Address Parity Error (See Table 2a)	Read A Amp Output (See Table 1)	$\overline{\text{A Error}}$
0	X	X	X	X	1
1	0	X	X	X	–
1	X	0	X	X	–
1	1 or 1 → 0	1 or 1 → 0	X	X	–
1	0 → 1	1	0	Even Number of 1s	1
1	0 → 1	1	1	X	0
1	0 → 1	1	X	Odd Number of 1s	0
1	1	0 → 1	0	Even Number of 1s	1
1	1	0 → 1	1	X	0
1	1	0 → 1	X	Odd Number of 1s	0

X = Don't Care State – = No Change

NOTES: 1. 0 →1 denotes a transition from logic 0 to logic 1 (negative logic).
2. The $\overline{\text{A Error}}$ output changes state on a 0 to 1 transition of $\overline{\text{Clk A}}$ or $\overline{\text{WEA}}$ input.

TABLE 4 – TRUTH TABLE FOR $\overline{\text{B ERROR}}$ (OUTPUT OF MASTER/SLAVE FF)

$\overline{\text{Clear}}$	$\overline{\text{Clk B}}$	$\overline{\text{WEB}}$	B Address Parity Error (See Table 2b)	Read B Amp Output (See Table 1)	$\overline{\text{B Error}}$
0	X	X	X	X	1
1	0	X	X	X	–
1	X	0	X	X	–
1	1 or 1 → 0	1 or 1 → 0	X	X	–
1	0 → 1	1	0	Even Number of 1s	1
1	0 → 1	1	1	X	0
1	0 → 1	1	X	Odd Number of 1s	0
1	1	0 → 1	0	Even Number of 1s	1
1	1	0 → 1	1	X	0
1	1	0 → 1	X	Odd Number of 1s	0

X = Don't Care State – = No Change

NOTES: 1. 0 → 1 denotes a transition from logic 0 to logic 1 (negative logic).
2. The $\overline{\text{B Error}}$ output changes state on a 0 to 1 transition of $\overline{\text{Clk B}}$ or $\overline{\text{WEB}}$ input.

TABLE 5 – TRUTH TABLE FOR A OUTPUT REGISTER LATCH (AL)

Clr Opt ①	$\overline{\text{Clear}}$	$\overline{\text{Clk A}}$	A Output Register Latch (AL)								
			AL00	AL01	AL02	AL03	AL04	AL05	AL06	AL07	AL08
Open	0	X	0	1	1	1	1	1	1	1	1
-2 V	0	X	0	0	0	0	0	0	0	0	0
X	1	1 ②	–	–	–	–	–	–	–	–	–
X	1	0 ③	Read A Amp Output (See Table 1)								

X = Don't Care State – = No Change

NOTES: ① The Clr Opt, Pin 48, may be left "Open" or connected to -2 Volts.
② Information is "latched" when $\overline{\text{Clk A}}$ = 1.
③ The Read A Amp Outputs are enabled to the output of the A Register Latch when $\overline{\text{Clk A}}$ = 0.

TABLE 6 – TRUTH TABLE FOR B OUTPUT REGISTER LATCH (BL)

Clr Opt ①	$\overline{\text{Clear}}$	$\overline{\text{Clk B}}$	B Output Register Latch (BL)								
			BL00	BL01	BL02	BL03	BL04	BL05	BL06	BL07	BL08
Open	0	X	0	1	1	1	1	1	1	1	1
-2 V	0	X	0	0	0	0	0	0	0	0	0
X	1	1 ②	–	–	–	–	–	–	–	–	–
X	1	0 ③	Read B Amp Output (See Table 1)								

X = Don't Care State – = No Change

NOTES: ① The Clr Opt, Pin 48, may be left "Open" or connected to -2 Volts.
② Information is "latched" when $\overline{\text{Clk B}}$ = 1.
③ The Read B Amp Outputs are enabled to the output of the B Register Latch when $\overline{\text{Clk B}}$ = 0.

TABLE 7 – TRUTH TABLES FOR A AND B DATA BUS OUTPUTS

a. A Data Bus Output

$\overline{\text{EnA}}$	A Data Bus AD00-AD08
1	Logic 1s (Disabled)
0	AL00-AL08 (Enabled)

b. B Data Bus Output

$\overline{\text{EnB}}$	B Data Bus BD00-BD08
1	Logic 1s (Disabled)
0	BL00-BL08 (Enabled)

TABLE 8 – TRUTH TABLE FOR $\overline{\text{AddEq}}$ OUTPUT

$\overline{\text{WEA}}$	$\overline{\text{WEB}}$	$\overline{\text{Clk A}}$	$\overline{\text{Clk B}}$	Address Equal A01-05=B01-05	$\overline{\text{AddEq}}$
X	X	X	X	0	1
1	1	X	X	X	1
1	0	1	X	1	1
X	0	0	X	1	0
0	1	X	1	1	1
0	X	X	0	1	0
0	0	X	X	1	0

X = Don't Care State

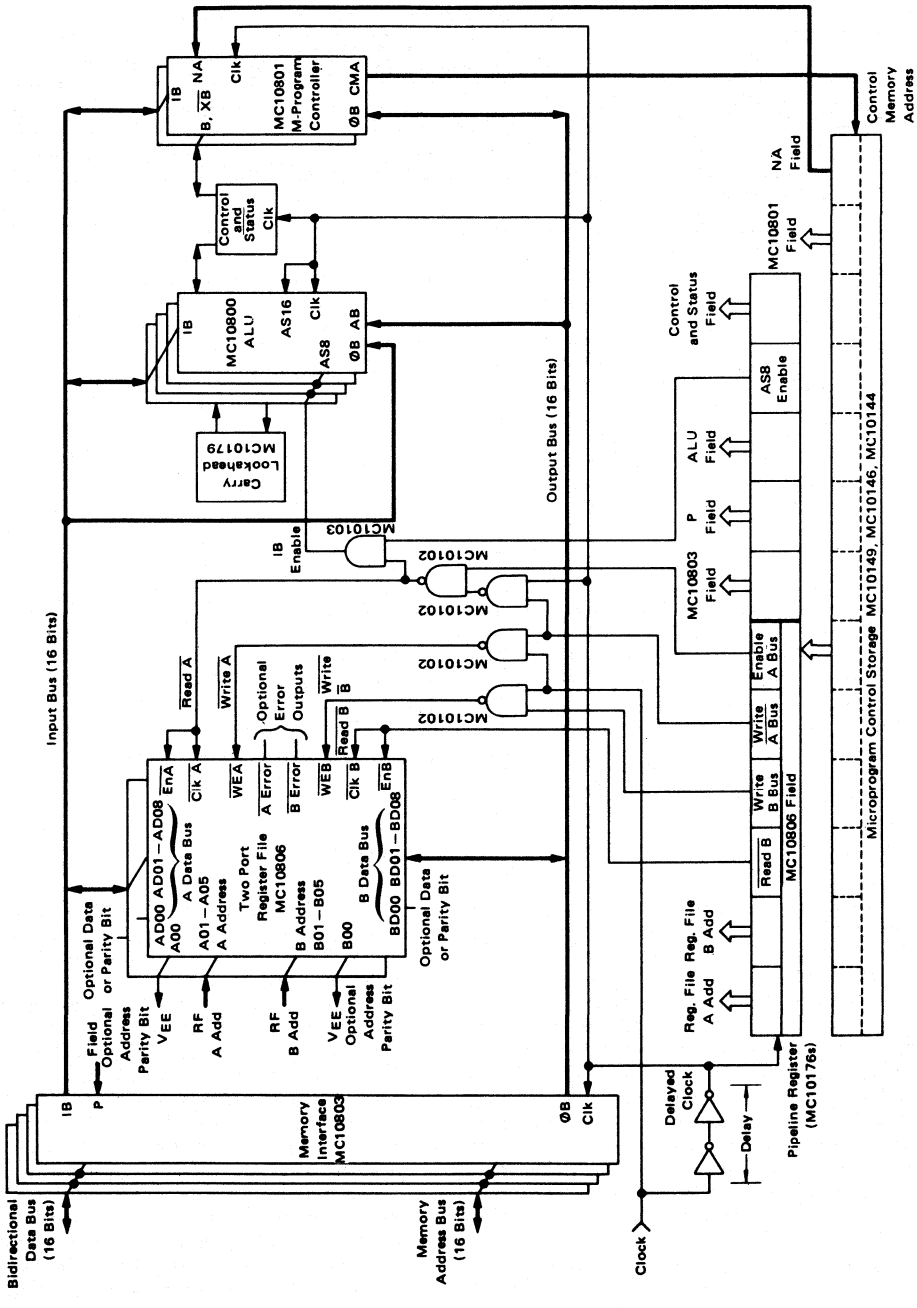
APPLICATION INFORMATION

Figure 3 shows two MC10806s interconnected to form a 32-word by 16-bit Register File in a typical 16-bit pipelined configuration. The Register File is designed so that two words can be read and operated within the ALU with the result being written back into the Register File (A plus B → A) all in the same cycle as shown in the timing diagram in Figure 4. The Clock is delayed about 5 ns in order to satisfy the hold time of data on the Input Bus with respect to the deactivation of the $\overline{\text{Write A}}$ signal. The delay could be made using a delay line or gate delay. The gating of the Clock for reading and writing into the Register File is shown in Figure 3 as well as the interconnections required on the MC10800s for latching the data (AS16) and controlling the enable (AS8) for data to the IB port.

For the A plus B → A operation, the following describes the cycle. First, the microprogram information for the A plus B → A operation is clocked into the Pipelined Register when the Clock goes from a 1 to a 0. After the delay of the register the two address locations A and B are accessed via the Register File with the data appearing on the Input and Output Bus. Then the ALU performs the A plus B operation via the ØB and AB ports of the MC10800. When the Clock goes to a 1, the information on the ØB port is latched in the MC10800; the result of the A plus B operation is enabled to the Input Bus; and, the $\overline{\text{ReadA}}$ line is deactivated (goes to a 1). On the 1 to 0 transition of the Clock, the data is stored in the Register File and a new operation is available at the output of the pipeline register.

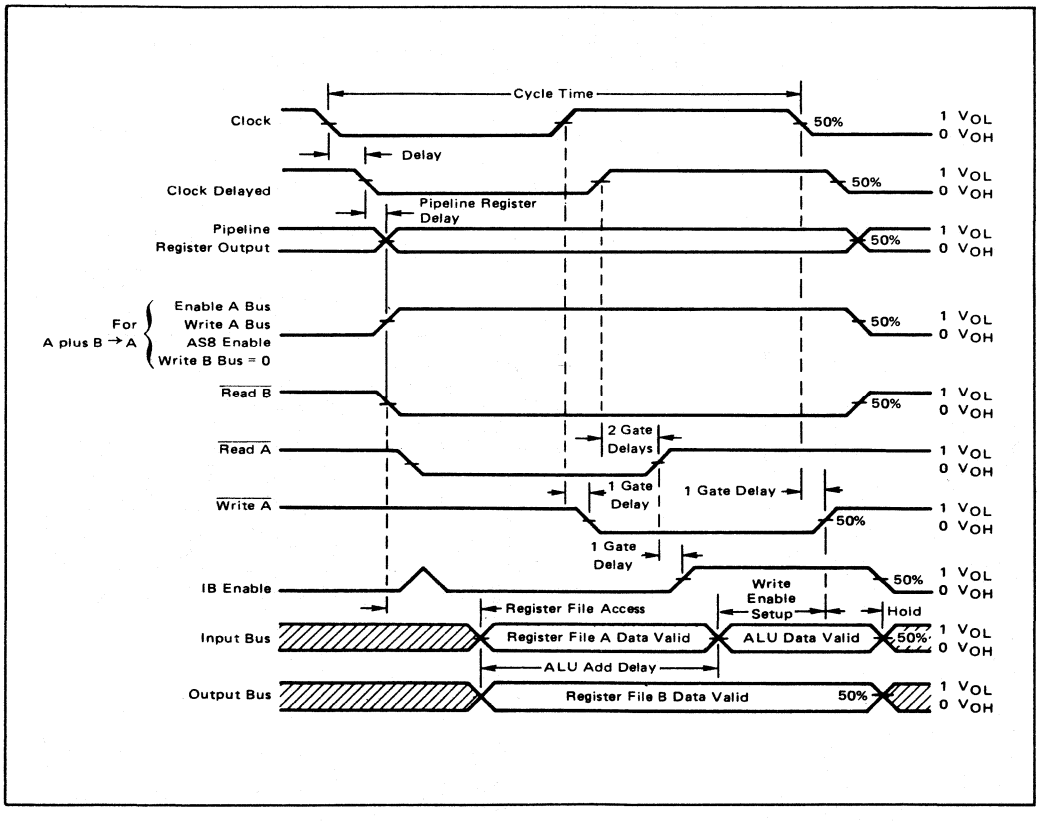
The Register File is also capable of writing into two locations in the same timing cycle. If a third Register File (C Address) was made available from microprogram memory, the operation A plus B → C could be performed in one cycle. This is accomplished by selecting the C Address onto the A Address inputs of the MC10806 when

FIGURE 3 – MC10806 REGISTER FILE APPLICATION USED IN A TYPICAL M10800 SYSTEM



the Clock goes from a 0 to a 1. The $\overline{\text{WriteA}}$ pulse would need to be narrowed using a 4 phase clock system in order to meet the address 4 phase clock system in order to meet the address setup time of the MC10806. In other words, the $\overline{\text{WriteA}}$ should not be activated until the C address is stable, meeting the address setup time of the MC10806. The cycle time for the system shown in Figure 3 is typically 85 ns. Worst case numbers can be calculated from the data sheets of parts used in the system along with the timing diagram information. The architecture of the system can be changed depending on system requirements. Versatility is of prime concern to the system designers and this flexibility is built into the M10800 family.

FIGURE 4 – TIMING DIAGRAM FOR MC10806 REGISTER FILE APPLICATION (FIG. 3)



RECOMMENDED OPERATING CONDITIONS

PARAMETER	SYMBOL	VALUE	UNIT
Supply Voltage (V_{CC} = 0 Volts)	V_{EE}	−4.68 to −5.72	Vdc
Operating Temp. (Functional)	T_A	−30 to +85	°C
Output Drive	–	50Ω to −2.0 Vdc	–
Maximum Clock Input Rise and Fall Time (20% to 80%)	t_r, t_f	10	ns

ELECTRICAL CHARACTERISTICS

Each MECL 10,000 series circuit has been designed to meet the dc specifications shown in the test table, after thermal equilibrium has been established. The circuit is in a test socket or mounted on a printed circuit board and transverse air flow greater than 500 linear fpm is maintained. Outputs are terminated through a 50-ohm resistor to −2.0 volts. Test procedures are shown for only one input, or for one set of input conditions. Other inputs tested in the same manner.

@ Test Temperature	TEST VOLTAGE VALUES (Volts) V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{EE}
−30°C	−0.890	−1.890	−1.205	−1.500	−5.2
+25°C	−0.810	−1.850	−1.105	−1.475	−5.2
+85°C	−0.700	−1.825	−1.035	−1.440	−5.2

Characteristic	Symbol	Pin Under Test	TEST LIMITS −30°C Min	−30°C Max	+25°C Min	+25°C Typ	+25°C Max	+85°C Min	+85°C Max	Unit	VOLTAGE APPLIED TO PINS LISTED BELOW: V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{EE}	(V_{CC}) Gnd
Power Supply Drain Current	I_{EE}	1, 24	–	–	–	330	413	–	–	mAdc	–	–	–	–	1, 24	7, 12, 17, 36
Input Current	I_{inH}	2	–	–	–	–	50	–	–	μAdc	2	27	–	–	1, 24	7, 12, 17, 36
		29	–	–	–	–	310	–	–		29	–	–	–		
		30	–	–	–	–	370	–	–		30	–	–	–		
		27	–	–	–	–	435	–	–		27	–	–	–		
	I_{inL}	29	–	–	0.5	–	–	–	–		–	29	–	–		
Logic "0" Output Voltage	V_{OH}	22	−1.060	−0.890	−0.960	–	−0.810	−0.890	−0.700	Vdc	45, 27	26*	–	–	1, 24	7, 12, 17, 36
		46	−1.060	−0.890	−0.960	–	−0.810	−0.890	−0.700	Vdc	28, 42	**	–	–	1, 24	7, 12, 17, 36
Logic "1" Output Voltage	V_{OL}	22	−1.890	−1.675	−1.850	–	−1.650	−1.825	−1.615	Vdc	45, 27	26	–	–	1, 24	7, 12, 17, 36
		46	−1.890	−1.675	−1.850	–	−1.650	−1.825	−1.615	Vdc	–	26, 28, 42, 44	–	–	1, 24	7, 12, 17, 36
	V_{OLB}	22	–	−1.980	–	–	−1.980	–	−1.980	Vdc	–	27	–	–	1, 24	7, 12, 17, 36
Logic "0" Threshold Voltage	V_{OHA}	22	−1.080	–	−0.980	–	–	−0.910	–	Vdc	45	26*	27	–	1, 24	7, 12, 17, 36
		46	−1.080	–	−0.980	–	–	−0.910	–	Vdc	–	27	–	–	1, 24	7, 12, 17, 36
Logic "1" Threshold Voltage	V_{OLA}	22	–	−1.655	–	–	−1.630	–	−1.595	Vdc	45	26	27	–	1, 24	7, 12, 17, 36
		46	–	−1.655	–	–	−1.630	–	−1.595	Vdc	–	28, 42, 44	–	26	1,24	7, 12, 17, 36
	V_{OLB}	22	–	−1.980	–	–	−1.980	–	−1.980	Vdc	–	–	–	27	1, 24	7, 12, 17, 36

*Pin 48 = −2.0 V

**V_{IH} on Pins 30, 31, 32, 33, 34, 35, 37, 38, 39, 40

SETUP AND HOLD TIME (NANOSECONDS AT 25°C)

Input	Input Reference	$\overline{\text{Clk A}}$ or $\overline{\text{Clk B}}$ (0 ↛1)				WEA or WEB (0↛1)			
		Set Up		Hold		Set Up		Hold	
		Min	Typ	Min	Typ	Min	Typ	Min	Typ
Address A, B	A00-A05 B00-B05	–	12	–	0	–	10*	–	0
Data Bus A, B,	AD00-AD08 BD00-BD08	–	–	–	–	–	14	–	0

* Address setup time is referenced to the 1 → 0 transition of $\overline{\text{WEA}}$ or $\overline{\text{WEB}}$.

IPD, PROPAGATION DELAY TIME (NANOSECONDS AT 25°C)

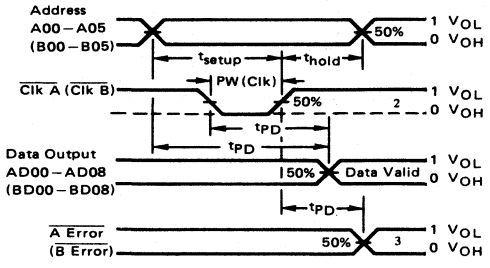
Input	Output	A Data Bus AD00-AD08 B Data Bus BD00-BD08		$\overline{\text{A Error}}$ $\overline{\text{B Error}}$		$\overline{\text{AddEq}}$	
		Typ	Max	Typ	Max	Typ	Max
Address A, B	A01-A05 B01-A05	19	–	–	–	–	–
$\overline{\text{EnA}}$, $\overline{\text{EnB}}$		6	–	–	–	–	–
$\overline{\text{Clk A}}$, $\overline{\text{Clk B}}$		7	–	5.5	–	7.5	–
$\overline{\text{WEA}}$, $\overline{\text{WEB}}$		–	–	–	–	8	–
$\overline{\text{Clear}}$		12	–	–	–	–	–

PULSE WIDTH TIME (NANOSECONDS AT 25°C)

Input	Min	Typ	
$\overline{\text{Clk A}}$, $\overline{\text{Clk B}}$	–	4.5	
$\overline{\text{WEA}}$, $\overline{\text{WEB}}$	–	12	Address A01-A05 ≠ B01-B05 ←
	–	18	Addresses Equal A01-A05 ≠ B01-B05 ←
$\overline{\text{Clear}}$	–	4.5	

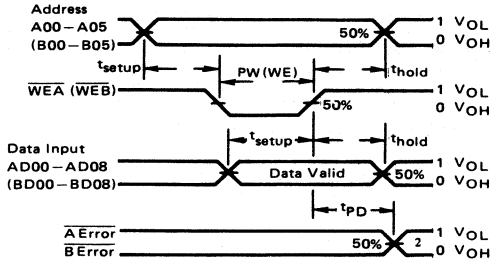
SWITCHING WAVEFORMS

Read Mode — Address Access[1]
(Port A or B or Both)



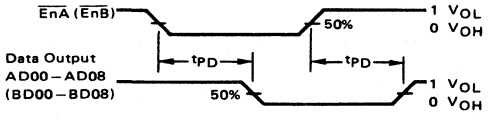
NOTES: 1. $\overline{EnA}$ ($\overline{EnB}$) is maintained at a Logic 0.
2. $\overline{ClkA}$ ($\overline{ClkB}$) can be maintained at a Logic 0 in the Read Mode.
3. $\overline{A\,Error}$ ($\overline{B\,Error}$) goes to a 0, if there is a Parity Error.

Write Mode[1]
(Port A or B or Both)



NOTES: 1. $\overline{EnA}$ ($\overline{EnB}$) is maintained at a Logic 1.
2. A Error (B Error) goes to a 0, if there is a Parity Error.

Read Mode — Enable Access
(Port A or B or Both)



TEST PROCEDURE FOR SETUP AND HOLD

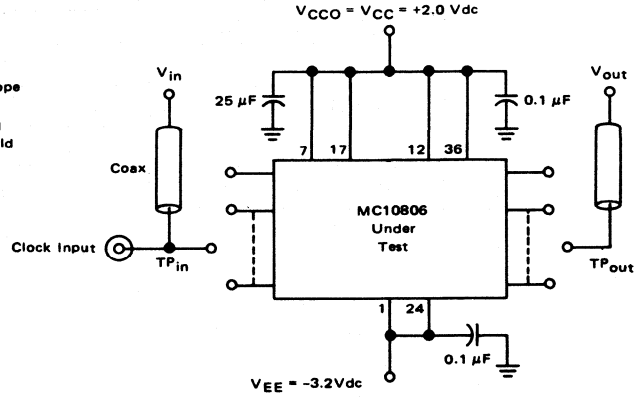
a. Establish setup time with long t_{hold}.
b. Keeping the leading edge of the input constant (t_{setup}) vary the trailing edge of the input to determine t_{hold}.

NOTE: t_{setup} and t_{hold} as defined are positive. Internal delays in the data path may result in a shift of the data waveform to the left, with respect to the clock, resulting in negative hold times.

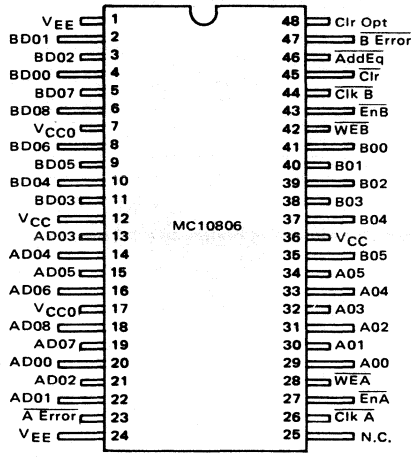
SWITCHING TIME TEST CIRCUIT

50 ohm termination to ground located in each scope channel input.

All input and output cables to the scope are equal lengths of 50 ohm coaxial cable. Wire length should be <¼ inch from TP_{in} to input pin and TP_{out} to output pin.

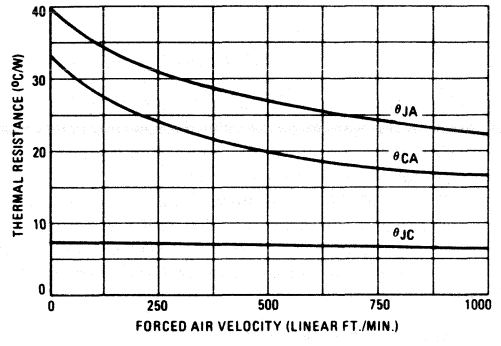


PIN ASSIGNMENT

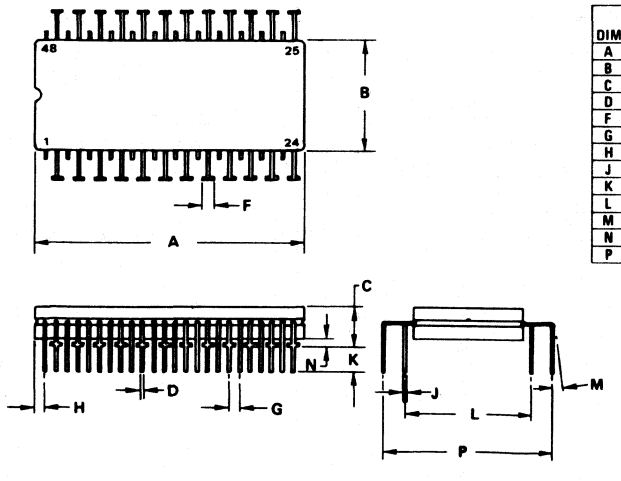


N.C. = No Connection

THERMAL CHARACTERISTICS (TYPICAL)



PACKAGE DIMENSIONS



	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	31.24	32.26	1.230	1.270
B	12.70	13.72	0.500	0.540
C	4.57	5.59	0.180	0.220
D	0.38	0.53	0.015	0.021
F	1.14	1.40	0.045	0.055
G	1.27 BSC		0.050 BSC	
H	1.02	1.52	0.040	0.060
J	0.20	0.30	0.008	0.012
K	2.54	3.30	0.100	0.130
L	15.24 BSC		0.600 BSC	
M	–	7°	–	7°
N	0.51	1.52	0.020	0.060
P	20.32 BSC		0.800 BSC	

Case 725–01

A socket for the QUIL package is available from ELECTRONIC MOLDING CORPORATION. (Part number 7178-295-5)

QUIL is a trademark of Motorola Inc.

MOTOROLA

MC10807

Advanced Information

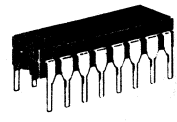
MECL
5-BIT BIDIRECTIONAL
BUS TRANSCEIVER

INTRODUCTION

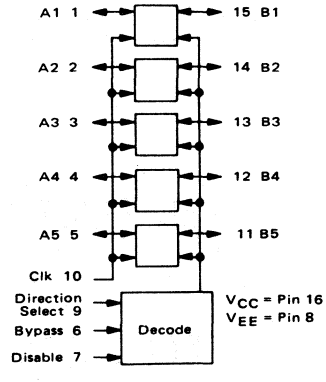
The MC10807 is a 5-bit bidirectional MECL bus transceiver. Data can be transferred directly in either direction (A port → B port or B port → A port), and an optional gated latch is also provided. Operation is shown in the truth table.

The MC10807 is in a 16-pin ceramic package and is a member of the high performance M10800 MECL/LSI processor family. It is designed to provide bidirectional exchange of MECL level signals in multiprocessor installations, and multiplexing of buses to a single processor.

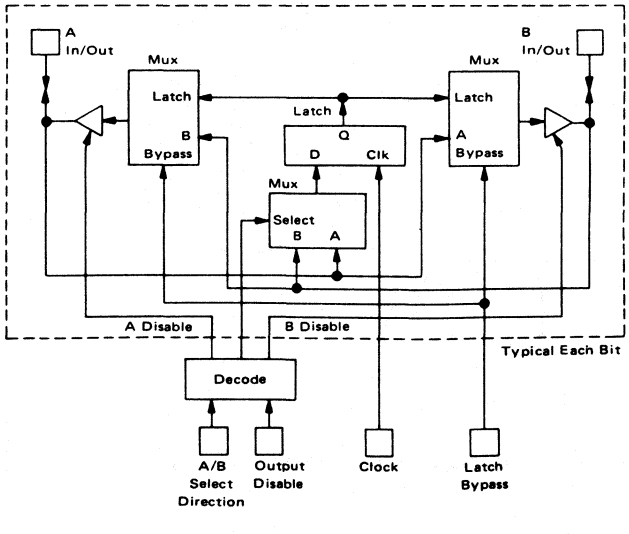
- Bidirectional Data Transfer
- Standard MECL 50 Ohm Drive Outputs
- Latch – Can Be Bypassed for High Speed



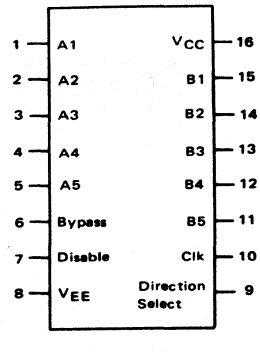
L SUFFIX
CERAMIC PACKAGE
CASE 620



BLOCK DIAGRAM



PIN ASSIGNMENTS



This is advance information on a new introduction and specifications are subject to change without notice.

FUNCTIONAL DESCRIPTION

The MC10807 consists of a function decode section, a clock buffer, and five identical bit channels. Each bit consists of a bidirectional A port, a bidirectional B port, and a latch.

Three logic pins control the function selection. These pins, along with the clock, all operate at standard MECL levels. The block diagram and truth table define the functions. The individual pin descriptions are as follows:

Output Disable

The Output Disable, when at V_{IL}, disables both the A and B port output buffers. That is, both are forced to high-impedance states. When the Output Disable is at V_{IH} the data translation takes place normally, and the appropriate output ports enabled by the direction select are active. Regardless of the state of the Output Disable pin, clocked data can be loaded into the latch from the selected input port.

A/B Direction Select

The A/B Direction Select pin controls the direction of data transfers. When at V_{IL}, the B-to-A direction is selected. In this case, the B output drivers are disabled, data is input to the latch from the B port, and data is output onto the A port. When the select pin is at V_{IH}, the A-to-B direction is selected and the function is the reverse.

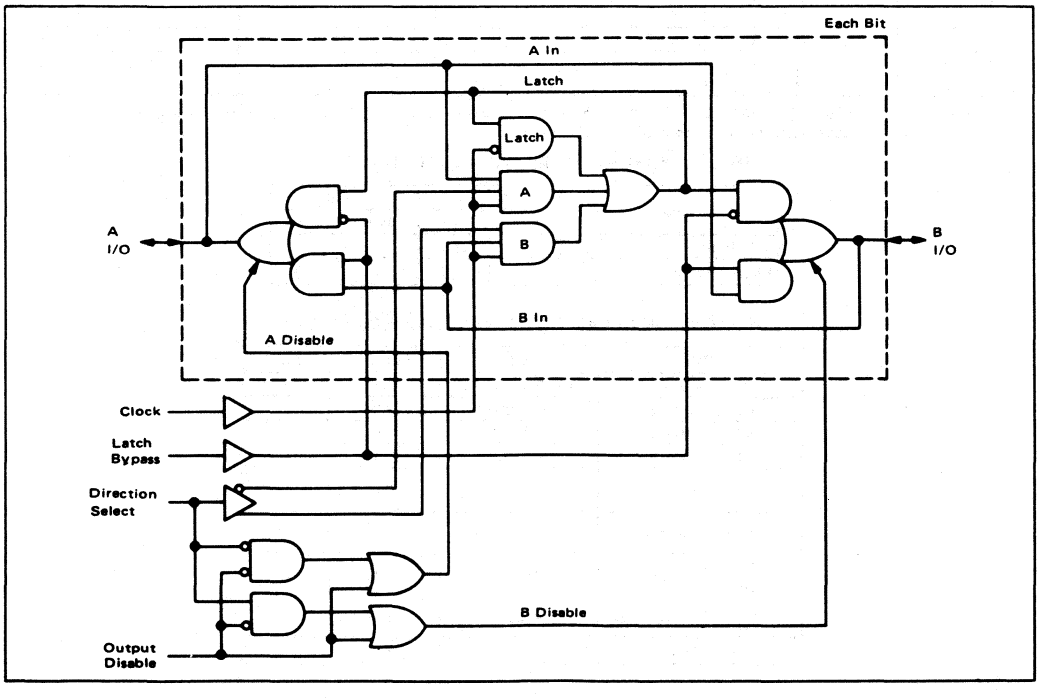
Latch Bypass

The Latch Bypass select line bypasses the latch circuitry for the fast data transfer. When the select line is at V_{IL}, the data is directed to both the latch input and the output buffer simultaneously. This feature enhances the speed of translation because the delay through the latch is bypassed. When the Latch Bypass pin is at V_{IH}, the data must first go into the latch then be sent to the output ports.

Clock

The Clock input is common to all latches and controls the storage of data. When the Clock is at V_{IL} the latch is open and data ripples through from the D input to the Q output. Data is stored or latched on the V_{IL}-to-V_{IH} transition of the Clock input.

NEGATIVE LOGIC DIAGRAM



TRUTH TABLE

SELECT INPUTS (ECL LEVELS, H = -.9V, L = -1.7V)				FUNCTION		
Disable	Direction Select	Bypass	Clock (2)	Latch (1)	B I/O**	A I/O**
H	H	H	H	• Q = H • Q = L	Output = Q = H = L	Off Off
H	H	H	L	Q = A Input = H = L	Output = Q = H = L	Input = H = L
H	H	L	H	• •	Output = A = H = L	Input = H = L
H	H	L	L	Q = A Input = H = L	Output = A = H = L	Input = H = L
H	L	H	H	• Q = H • Q = L	Off Off	Output = Q = H = L
H	L	H	L	Q = B Input = H = L	Input = H = L	Output = Q = H = L
H	L	L	H	• •	Input = H = L	Output = B = H = L
H	L	L	L	Q = B Input = H = L	Input = H = L	Output = B = H = L
L	H	H	H	•	Off	Off
L	H	H	L	Q = A Input = H = L	Off Off	Input = H = L
L	H	L	H	•	Off	Off
L	H	L	L	Q = A Input = H = L	Off Off	Input = H = L
L	L	H	H	•	Off	Off
L	L	H	L	Q = B Input = H = L	Input = H = L	Off
L	L	L	H	•	Off	Off
L	L	L	L	Q = B Input = H = L	Input = H = L	Off

NOTES: (1) • Denotes "NO CHANGE" (2) Latch transfers data when clock is "L" and stores data when clock is "H"
(3) **Output driver is disabled to V_{OLZ} during "Off" state.

SETUP AND HOLD TIMES (NANOSECONDS AT 25°C)

Input	Setup		Hold	
	Min	Typ	Min	Typ
A1-5, B1-5	–	2.0	–	4.0
Direction Select	–	4.5	–	1.5

PROPAGATION DELAY TIMES (NANOSECONDS AT 25°C)

Path	Mode	Typical	Max
A1-5 → B1-5	Latch Bypassed	3.9	–
B1-5 → A1-5	Latch Bypassed	3.9	–
A1-5 → B1-5	Via Latch	6.4	–
B1-5 → A1-5	Via Latch	6.4	–
Bypass → Output		6.4	–
Disable → Output		4.9	–
Direction Select → Output	Direct	7.4	–
Direction Select → Output	Via Latch	8.9	–
Clock → Output		7.0	–

RECOMMENDED OPERATING CONDITIONS

Parameter	Symbol	Value	Unit
Supply Voltage (V_{CC} = 0 Volts)	V_{EE}	-4.68 to -5.72	Vdc
Operating Temperature (Functional)	T_A	-30 to +85	°C
Output Drive	–	50 Ω to -2.0 Vdc	–
Maximum Clock Input Rise and Fall Time (20% to 80%)	t_r, t_f	10	ns
Minimum Clock Pulse Width	PW	5	ns

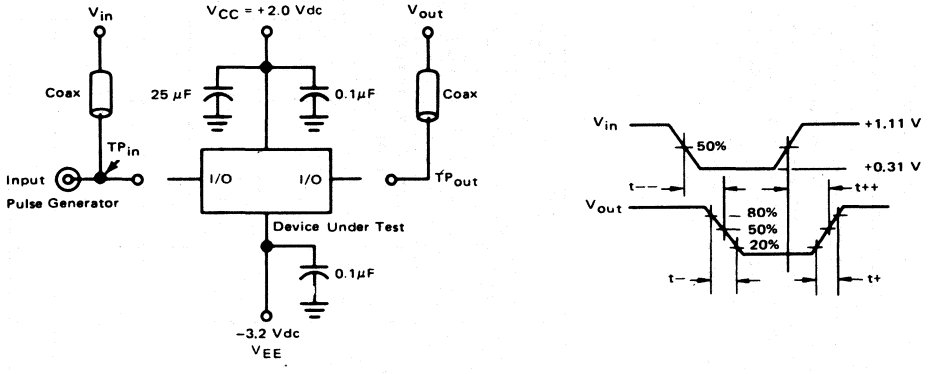
ELECTRICAL CHARACTERISTICS

Each MECL 10,000 series circuit has been designed to meet the dc specifications shown in the test table, after thermal equilibrium has been established. The circuit is in a test socket or mounted on a printed circuit board and transverse airflow greater than 500 linear fpm is maintained. Outputs are terminated through a 50-ohm resistor to -2.0 volts. Test procedures are shown for only one input, or for one set of input conditions. Other inputs tested in the same manner.

@ Test Temperature	TEST VOLTAGE VALUES (Volts) V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{EE}
-30°C	-0.0890	-1.890	-1.205	-1.500	-5.2
+25°C	-0.810	-1.850	-1.105	-1.475	-5.2
+85°C	-0.700	-1.825	-1.035	-1.440	-5.2

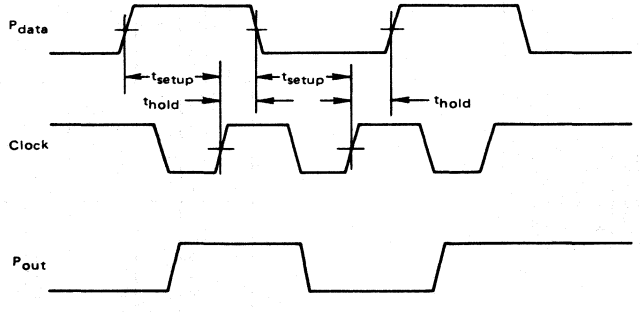
Characteristic	Symbol	Pin Under Test	MC10807 TEST LIMITS -30°C Min	-30°C Max	+25°C Min	+25°C Typ	+25°C Max	+85°C Min	+85°C Max	Unit	VOLTAGE APPLIED TO PINS LISTED BELOW: V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{EE}	(V_{CC}) Gnd
Power Supply Drain Current	I_{EE}	8	–	–	–	87	–	–	–	mAdc	–	–	–	–	8	16
Input Current	I_{inH}	6	–	–	–	–	350	–	–	μAdc	6	–	–	–	8	16
		1	–	–	–	–	410	–	–		1	–	–	–	8	16
	I_{inL}	6	–	–	0.5	–	–	–	–	μAdc	–	6	–	–	8	16
Logic "0" Output Voltage	V_{OH}	1	-1.060	-0.890	-0.960	–	-0.810	-0.890	-0.700	Vdc	7, 10, 15	6, 9	–	–	8	16
Logic "1" Output Voltage	V_{OL}	1	-1.890	-1.675	-1.850	–	-1.650	-1.825	-1.615	Vdc	7, 10	6, 9, 15	–	–	8	16
Logic "0" Threshold Voltage	V_{OHA}	1	-1.080	--	0.980	–	–	-0.910	–	Vdc	7, 10	6, 9	15	–	8	16
Logic "1" Threshold Voltage	V_{OLA}	1	–	-1.655	–	–	-1.630	–	-1.595	Vdc	7, 10	6, 9	–	15	8	16
Output Cutoff Voltage	V_{OLZ}	1	–	-1.980	--	–	-1.980	–	-1.980	Vdc	–	7	–	–	8	16

SWITCHING TIME TEST CIRCUIT AND WAVEFORMS @ 25°C FOR PROPAGATION DELAY

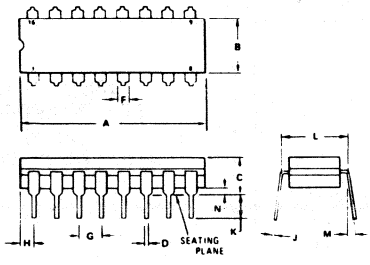


50-ohm termination to ground located in each scope channel input.
All input and output cables to the scope are equal lengths of 50-ohm coaxial cable.
Wire length should be < 1/4 inch from TP_{in} to input pin and TP_{out} to output pin.

SETUP AND HOLD TIME WAVEFORMS



PACKAGE DIMENSIONS



DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	19.05	19.81	0.750	0.780
B	6.22	6.98	0.245	0.275
C	4.06	5.08	0.160	0.200
D	0.38	0.51	0.015	0.020
F	1.40	1.65	0.055	0.065
G	2.54 BSC		0.100 BSC	
H	0.51	1.14	0.020	0.045
J	0.20	0.31	0.008	0.012
K	3.18	0.30	0.125	0.160
L	7.37	7.87	0.290	0.310
M	–	15°	–	15°
N	0.51	1.02	0.020	0.040

CASE 620-02

NOTES
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION
2. PKG. INDEX: NOTCH IN LEAD NOTCH IN CERAMIC OR INK DOT
3. DIM "L" TO CENTER OF LEADS WHEN FORMED PARALLEL

MOTOROLA

MC10808

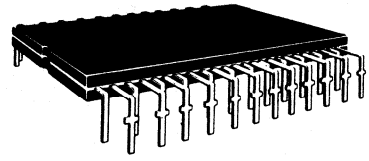
Advance Information

INTRODUCTION

The MC10808 Programmable Multi-Bit Shifter is an LSI building block for shifting data in a high-speed processor system. The circuit is essential when performing floating point operations for pre-normalization or alignment of exponents.

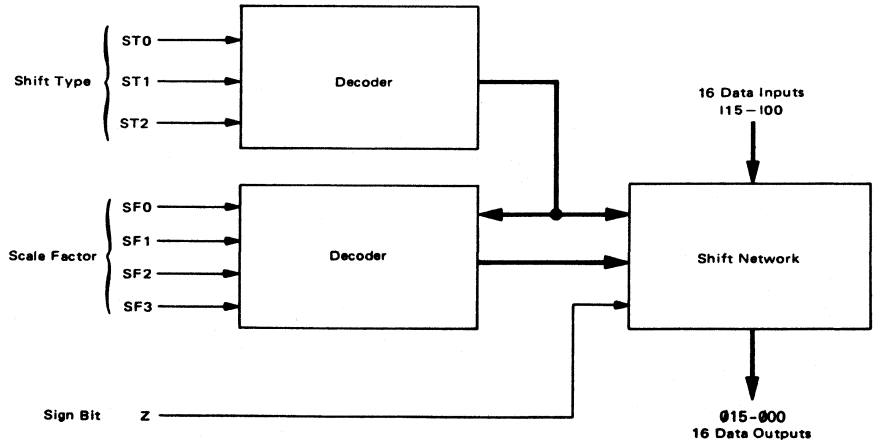
The Programmable Multi-Bit Shifter as shown in the block diagram contains a 16-bit shift network that is fully expandable in a shifter array to handle practically any number of bits. The shift type function select contains arithmetic, logic, and rotate shifting. Four scale factor inputs are provided for specifying the number of positions that the input data is to be shifted or rotated. A sign bit is also provided for arithmetic shift operations.

MECL — LSI PROGRAMMABLE 16-BIT SHIFTER FUNCTION



CASE 725-01

BLOCK DIAGRAM



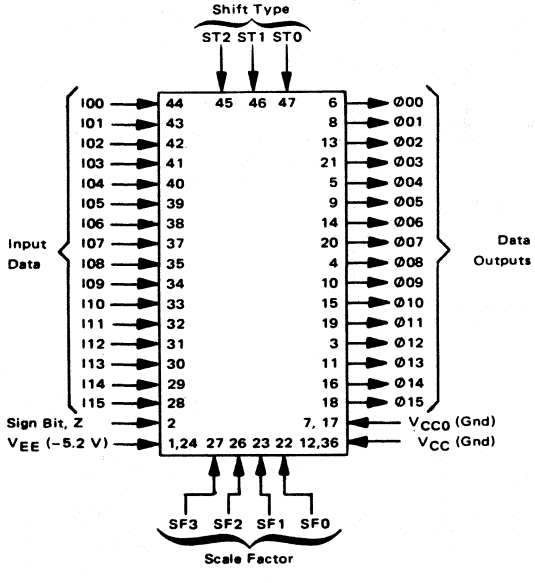
This is advance information and specifications are subject to change without notice.

TABLE OF CONTENTS

IMPORTANT FEATURES

1. Three hundred gate complexity reduces package count considerably while increasing system speed.
2. Sixteen separate data inputs and sixteen data outputs are available with only two levels of gating separating them for high-speed operation. The sign bit input goes through only one level of gating.
3. Three shift type select lines are used to select eight different shift functions including shift left, shift right, and rotate.
4. Four scale factor inputs select the number of positions (in binary and 2's complement for shift right and shift left) the data is to be shifted.
5. Sign bit input is used for arithmetic shifting and for sign extend operations. Also, the sign bit is used in logic shifting for use in both positive and negative logic systems.
6. The outputs may be disabled for array expansions by selecting the "ODA" function.
7. High-speed operation of 6 ns typ delay from Data-In to Data-Out, 6 ns typ delay from Sign Bit to Data-Out, and 12 ns typ delay from the select lines to the Data-Out.
8. Two different shifter arrays can be built. One array requires only two package delays for a shifter requiring up to 256 bits. The other array requires only one package delay but more packages. A 64-bit shifter requires ten MC10808s with two package delays or sixteen MC10808s with one package delay.
9. Fully compatible with the MECL 10,000 family.

INPUT/OUTPUT DIAGRAM – MC10808



ABSOLUTE MAXIMUM RATINGS (See Note)

Rating		Symbol	Value	Unit
Supply Voltage (V_{CC} = 0)		V_{EE}	-8 to 0	Vdc
Input Voltage (V_{CC} = 0)		V_{in}	0 to V_{EE}	Vdc
Output Source Current	Cont	I_o	< 50	mAdc
	Surge		< 100	
Storage Temperature		T_{stg}	-55 to +150	°C
Junction Temperature		T_j	165	

NOTE: Permanent device damage may occur if absolute maximum ratings are exceeded. Functional operation should be restricted to RECOMMENDED OPERATING CONDITIONS. Exposure to higher than recommended voltages for extended periods of time could affect device reliability.

SYSTEM OVERVIEW

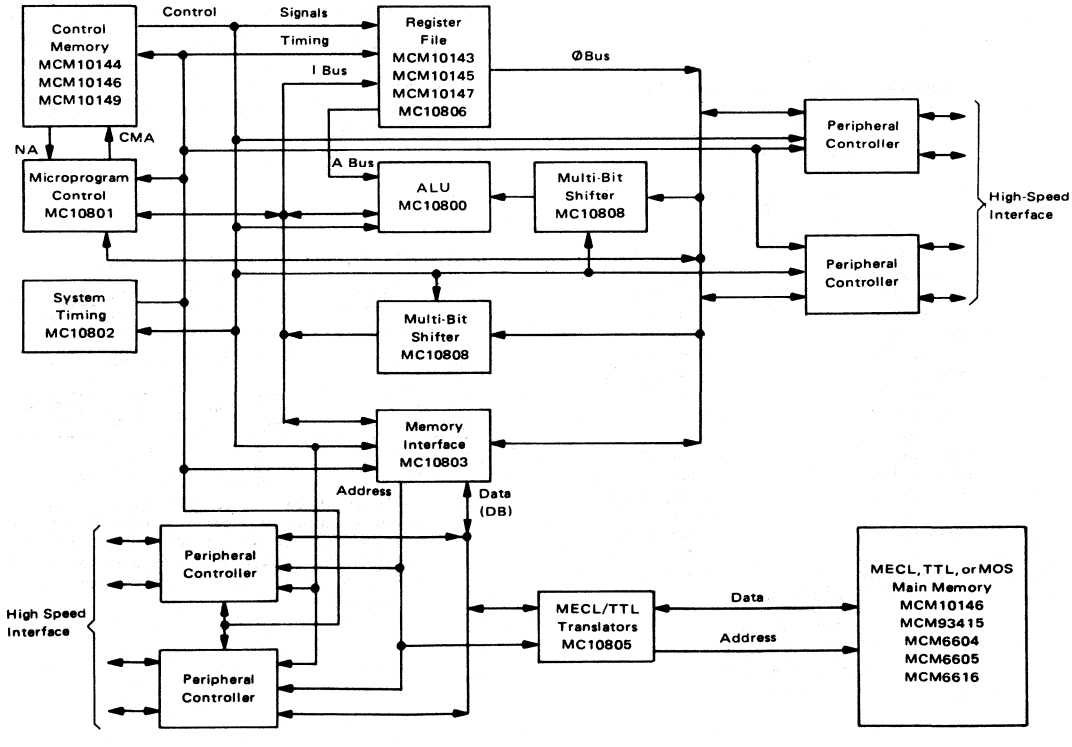
The Motorola M10800 family of LSI processor circuits combines the cost and size advantages of LSI with system design flexibility. Each family part is a major system building block which can be interconnected and programmed for a wide range of processor applications. Figure 1 illustrates a method of using the various circuits in a general purpose processor. The MC10800 4-Bit ALU Slice performs the various arithmetic, logic, and shift functions. This circuit features full BCD capability and a complete set of status outputs. The MC10801 Microprogram Control Function addresses and sequences through microprogram control memory. A set of 16 control instructions provides for direct jumps, conditional branches, and subroutines within microprogram. The MC10802 Timing Function generates clock phases and features single cycle or single phase clock increment for troubleshooting or diagnostics. The MC10803 Memory Interface Function interfaces between the LSI processor circuits and other parts of a system. The circuit generates memory addresses and provides for the bidirectional transfer of processor data.

In systems that use floating point data manipulation in high-speed systems, the MC10808 provides the shifting required for prenormalization or alignment of exponents. There are two ways that the MC10808 can be interconnected into the system as shown in Figure 1. One way is to connect the data input of the MC10808 to the Ø Bus and the output to the I Bus. Then the accumulator of the M10800, the external register file, or the register file of the M10803 can be shifted onto the I Bus and loaded back into the accumulator or register file. If the shifter input is connected to Ø Bus and if the output is connected to the Ø Bus input of the MC10800, then data in register file can be shifted before it is operated on by the ALU of the MC10800. Then the result may be loaded back into the register file using the I Bus.

The MC10808 performs the shift operation asynchronously with the number of shift positions determined by the shift type select and the scale factor. The speed for an equivalent operation using shift registers is a function of the number of clock pulses required to perform the shift.

Versatility is a main point of the M10800 family. The block diagram in Figure 1 is intended to illustrate the purpose of the various LSI functions and not restrict the designer to any particular system configuration or application.

FIGURE 1 – TYPICAL SYSTEM CONFIGURATION



PIN ASSIGMENTS

Pin Designation	Pin Number	Description
I00	44	Input Data – Bit 0 (LSB)
I01	43	Input Data – Bit 1
I02	42	Input Data – Bit 2
I03	41	Input Data – Bit 3
I04	40	Input Data – Bit 4
I05	39	Input Data – Bit 5
I06	38	Input Data – Bit 6
I07	37	Input Data – Bit 7
I08	35	Input Data – Bit 8
I09	34	Input Data – Bit 9
I10	33	Input Data – Bit 10
I11	32	Input Data – Bit 11
I12	31	Input Data – Bit 12
I13	30	Input Data – Bit 13
I14	29	Input Data – Bit 14
I15	28	Input Data – Bit 15 (MSB)
Ø00	6	Output Data – Bit 0 (LSB)
Ø01	8	Output Data – Bit 1
Ø02	13	Output Data – Bit 2
Ø03	21	Output Data – Bit 3
Ø04	5	Output Data – Bit 4
Ø05	9	Output Data – Bit 5
Ø06	14	Output Data – Bit 6
Ø07	20	Output Data – Bit 7
Ø08	4	Output Data – Bit 8
Ø09	10	Output Data – Bit 9
Ø10	15	Output Data – Bit 10
Ø11	19	Output Data – Bit 11
Ø12	3	Output Data – Bit 12
Ø13	11	Output Data – Bit 13
Ø14	16	Output Data – Bit 14
Ø15	18	Output Data – Bit 15
ST0	47	Shift Type – Select Input
ST1	46	Shift Type – Select Input
ST2	45	Shift Type – Select Input
SF0	22	Scale Factor – LSB Input
SF1	23	Scale Factor – NLSB Input
SF2	26	Scale Factor – NMSB Input
SF3	27	Scale Factor – MSB Input
Z	2	Sign Bit
V_{CC}	12	Ground
V_{CC}	36	Ground
V_{CCO}	7	Ground
V_{CCO}	17	Ground
V_{EE}	1	-5.2 Volt Supply
V_{EE}	24	-5.2 Volt Supply

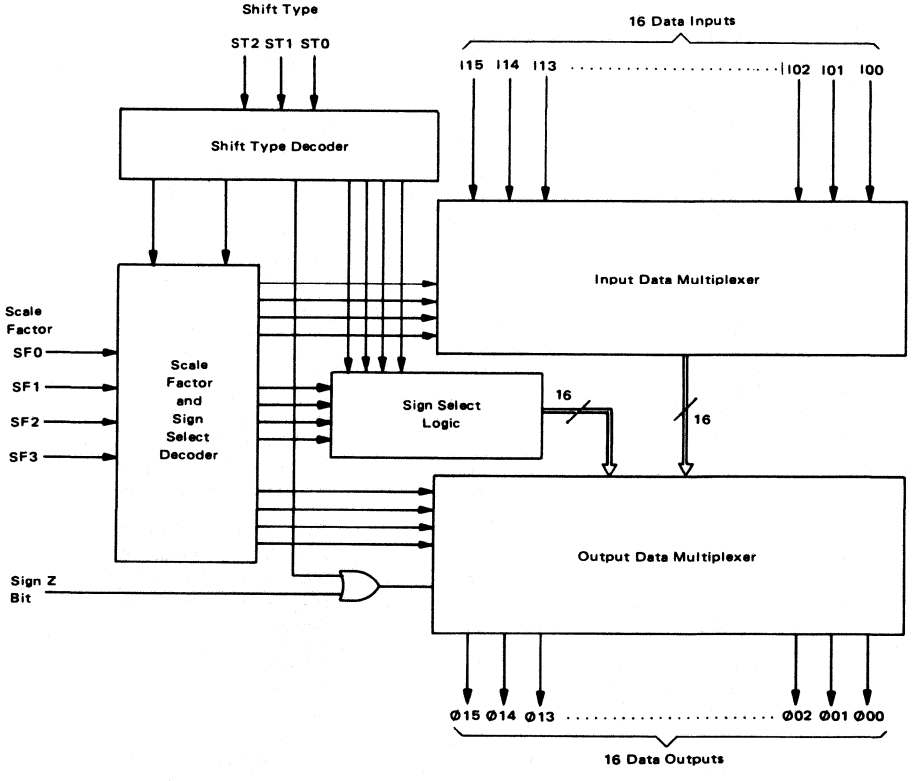
Pins 25 and 48 not used

ARCHITECTURAL DESCRIPTION

A functional block diagram of the MC10808 Programmable Multi-Bit Shifter is shown in Figure 2. There are sixteen inputs and sixteen outputs with two levels of gating to perform the shifting. There are three shift type select inputs—ST2, ST1, and ST0—that are used to select the appropriate shifting function. Also, four scale factor inputs specify the number of positions that the input data should be shifted or rotated. A sign bit input is used for arithmetic shift right or left and sign extend operations. The outputs can be disabled for wire-ANDing (negative logic) other device outputs by selecting the "ODA" command.

The input multiplexers do a right rotate only of 0, 1, 2, or 3 positions and the output multiplexers do a right rotate only of 0, 4, 8, or 12 positions. The sign select logic can force the Z input (see Figure 2) into the data output positions as selected by the shift type and the sign select scale factor. In this manner the function select (see Table 1) can be implemented to perform all the functions listed such as arithmetic shift right and left, rotate right and left, 2's complement right and left, sign extend, and output disable.

FIGURE 2 – FUNCTIONAL BLOCK DIAGRAM



MC10808 INPUT/OUTPUT SIGNALS

Pin Designation/Name	Function
Z/Sign Bit	The sign bit is used to specify the sign during an arithmetic shift or sign extend operation. The sign bit is usually connected to the most signficiant bit of the data word.
ST0, ST1, ST2/Shift Type Select	A description of the shift type select is described in Table 1.
SF0–SF3/Scale Factor Select	The truth tables for the scale factor versus various shift types are described in Tables 2 through 9.
I00–I15/Input Data	There are 16 input data lines for shifting to the output.
Ø00–Ø15/Output Data	There are 16 output data lines that contain the shifted data from the inputs.

FUNCTIONAL DESCRIPTION

Tables 1 through 9 describe the logical operation of the Programmable Multi-Bit Shifter (PMS) along with the block diagram in Figure 2. There are eight different shift types that may be selected as shown in Table 1. The following is a description of the various shift types. **All truth tables are expressed in negative logic with VOL being a logic 1 and VOH a logic 0.**

ALS–Arithmetic Shift Left

As shown in Table 2, the input data is shifted to the left with the vacated bit positions filled with the logic level existing at the sign bit input. The sign bit input is connected to either VOL or VOH depending if the system uses positive or negative logic.

ARS–Arithmetic Shift Right

As shown in Table 3, the input data is shifted to the right with the vacated bit positions filled with the sign bit for arithmetic shifting. For logic shifting, the sign bit input is connected to either VOL or VOH depending if the system uses positive or negative logic.

RLT–Rotate LefT

As shown in Table 4, the input data is rotated to the left as specified by the binary number specified by the scale factor.

RRT–Rotate RighT

This function is very useful in building shifter arrays requiring 16, 32, 64, 128, and 256 bits. As shown in Table 5, the input data is rotated to the right a number of positions as specified by the scale factor.

SRC–Shift Right Using 2's Complement

This function is very useful in shifter arrays having a one package delay that requires a shift left operation. One example is shown in Figure 4, a 32-bit shift/rotate array. Table 6 shows the truth table for the SRC function with the inputs shifted to the right as selected by the 2's complement of the scale factor. Negative logic 1s fill the vacated positions.

SLC–Shift Left Using 2's Complement

This function is very useful in implementing shifter arrays (one package delay) requiring a shift right operation. The SLC function is used to extend the number of bits to any number larger than 16 bits when performing an arithmetic or logic shift right. The SLC function can also be used in shifter arrays for performing the shift left operaton where the scale factor is in 2's complement notation. Table 7 shows the truth table for the SLC function. Note that logic 1s fill the vacated positions.

ODA–Output DisAble

This function is used to disable the outputs to a negative logic 1 as shown in Table 8.

SBO–Sign Bit is placed at all Outputs

This function places the sign bit at all the outputs regardless of the scale factor, as shown in Table 9. It is very useful in extending the sign in shifter arrays when performing an arithmetic shift right.

TABLE 1. SHIFT TYPE SELECT DESCRIPTION (NEGATIVE LOGIC)

Shift Type			Symbol	Description
ST2	ST1	ST0		
0	0	0	SBO	Sign Bit is placed at all Outputs. The Ø15 through Ø00 are filled with the sign bit for use in arithmetic shifting.
0	0	1	ODA	Output DisAble. The Ø15 through Ø00 outputs are forced to logic 1s, so that the device can be wire-ANDed to other device outputs.
0	1	0	SLC	Shift Left using 2's Complement. The I15—I00 inputs are shifted to the left at the Ø15—Ø00 outputs as selected by the 2's complement of the scale factor inputs SF3—SF0. The vacated bit positions are filled with negative logic 1s.
0	1	1	SRC	Shift Right using 2's Complement. The I15—I00 inputs are shifted to the right at the Ø15—Ø00 outputs as selected by the 2's complement of the scale factor inputs SF3—SF0. The vacated bit positions are filled with negative logic 1s.
1	0	0	RRT	Rotate RighT. The I15-I00 inputs are rotated to the right at the Ø15-Ø00 outputs as selected by the binary number specified by the scale factor inputs.
1	0	1	RLT	Rotate LefT. The I15-I00 inputs are rotated to the left at the Ø15-Ø00 outputs as selected by the binary number specified by the scale factor inputs.
1	1	0	ARS	Arithmetic Shift Right. The I15—I00 inputs are shifted to the right at the Ø15—Ø00 outputs as selected by the binary number specified by the scale factor inputs. The vacated bit positions are filled with the sign bit.
1	1	1	ALS	Arithmetic Shift Left. The I15—I00 inputs are shifted to the left at the Ø15—Ø00 outputs as selected by the binary number specified by the scale factor inputs. The vacated bit positions are filled with the sign bit.

TABLE 2. OUTPUT TRUTH TABLE FOR ARITHMETIC SHIFT LEFT (ST2 = 1, ST1 = 1, ST0 = 1)

Scale Factor				Output														
SF3	SF2	SF1	SF0	Ø15	Ø14	Ø13	Ø12	Ø11	Ø10	Ø09	Ø08	Ø07	Ø06	Ø05	Ø04	Ø03	Ø02	Ø01	Ø00
0	0	0	0	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00
0	0	0	1	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	Z
0	0	1	0	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	Z	Z
0	0	1	1	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	Z	Z	Z
0	1	0	0	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	Z	Z	Z	Z
0	1	0	1	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	Z	Z	Z	Z	Z
0	1	1	0	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	Z	Z	Z	Z	Z	Z
0	1	1	1	I08	I07	I06	I05	I04	I03	I02	I01	I00	Z	Z	Z	Z	Z	Z	Z
1	0	0	0	I07	I06	I05	I04	I03	I02	I01	I00	Z	Z	Z	Z	Z	Z	Z	Z
1	0	0	1	I06	I05	I04	I03	I02	I01	I00	Z	Z	Z	Z	Z	Z	Z	Z	Z
1	0	1	0	I05	I04	I03	I02	I01	I00	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
1	0	1	1	I04	I03	I02	I01	I00	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
1	1	0	0	I03	I02	I01	I00	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
1	1	0	1	I02	I01	I00	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
1	1	1	0	I01	I00	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
1	1	1	1	I00	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z

TABLE 3. OUTPUT TRUTH TABLE FOR ARITHMETIC SHIFT RIGHT (ST2 = 1, ST1 = 1, ST0 = 0)

Scale Factor				Output															
SF3	SF2	SF1	SF0	Ø15	Ø14	Ø13	Ø12	Ø11	Ø10	Ø09	Ø08	Ø07	Ø06	Ø05	Ø04	Ø03	Ø02	Ø01	Ø00
0	0	0	0	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00
0	0	0	1	Z	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01
0	0	1	0	Z	Z	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02
0	0	1	1	Z	Z	Z	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03
0	1	0	0	Z	Z	Z	Z	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04
0	1	0	1	Z	Z	Z	Z	Z	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05
0	1	1	0	Z	Z	Z	Z	Z	Z	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06
0	1	1	1	Z	Z	Z	Z	Z	Z	Z	I15	I14	I13	I12	I11	I10	I09	I08	I07
1	0	0	0	Z	Z	Z	Z	Z	Z	Z	Z	I15	I14	I13	I12	I11	I10	I09	I08
1	0	0	1	Z	Z	Z	Z	Z	Z	Z	Z	Z	I15	I14	I13	I12	I11	I10	I09
1	0	1	0	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	I15	I14	I13	I12	I11	I10
1	0	1	1	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	I15	I14	I13	I12	I11
1	1	0	0	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	I15	I14	I13	I12
1	1	0	1	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	I15	I14	I13
1	1	1	0	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	I15	I14
1	1	1	1	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	I15

TABLE 4. OUTPUT TRUTH TABLE FOR LEFT ROTATE (ST2 = 1, ST1 = 0, ST0 = 1)

Scale Factor				Output															
SF3	SF2	SF1	SF0	Ø15	Ø14	Ø13	Ø12	Ø11	Ø10	Ø09	Ø08	Ø07	Ø06	Ø05	Ø04	Ø03	Ø02	Ø01	Ø00
0	0	0	0	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00
0	0	0	1	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15
0	0	1	0	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14
0	0	1	1	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13
0	1	0	0	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12
0	1	0	1	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11
0	1	1	0	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10
0	1	1	1	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09
1	0	0	0	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08
1	0	0	1	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07
1	0	1	0	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06
1	0	1	1	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05
1	1	0	0	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04
1	1	0	1	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03
1	1	1	0	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02
1	1	1	1	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01

TABLE 5. OUTPUT TRUTH TABLE FOR RIGHT ROTATE (ST2 = 1, ST1 = 0, ST0 = 0)

Scale Factor				Output															
SF3	SF2	SF1	SF0	Ø15	Ø14	Ø13	Ø12	Ø11	Ø10	Ø09	Ø08	Ø07	Ø06	Ø05	Ø04	Ø03	Ø02	Ø01	Ø00
0	0	0	0	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00
0	0	0	1	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01
0	0	1	0	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02
0	0	1	1	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03
0	1	0	0	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04
0	1	0	1	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05
0	1	1	0	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06
0	1	1	1	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08	I07
1	0	0	0	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09	I08
1	0	0	1	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10	I09
1	0	1	0	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11	I10
1	0	1	1	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12	I11
1	1	0	0	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13	I12
1	1	0	1	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14	I13
1	1	1	0	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15	I14
1	1	1	1	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	I15

TABLE 6. OUTPUT TRUTH TABLE FOR SHIFT RIGHT, 2'S COMPLEMENT (ST2 = 0, ST1 = 1, ST0 = 1)

Scale Factor				Output														
SF3	SF2	SF1	SF0	Ø15	Ø14	Ø13	Ø12	Ø11	Ø10	Ø09	Ø08	Ø07	Ø06	Ø05	Ø04	Ø03	Ø02	Ø01	Ø00
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	I15
0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	I15	I14
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	I15	I14	I13
0	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	I15	I14	I13	I12
0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	I15	I14	I13	I12	I11
0	1	1	0	1	1	1	1	1	1	1	1	1	1	I15	I14	I13	I12	I11	I10
0	1	1	1	1	1	1	1	1	1	1	1	1	I15	I14	I13	I12	I11	I10	I09
1	0	0	0	1	1	1	1	1	1	1	1	I15	I14	I13	I12	I11	I10	I09	I08
1	0	0	1	1	1	1	1	1	1	1	I15	I14	I13	I12	I11	I10	I09	I08	I07
1	0	1	0	1	1	1	1	1	1	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06
1	0	1	1	1	1	1	1	1	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05
1	1	0	0	1	1	1	1	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04
1	1	0	1	1	1	1	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03
1	1	1	0	1	1	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02
1	1	1	1	1	I15	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01

TABLE 7. OUTPUT TRUTH TABLE FOR SHIFT LEFT, 2'S COMPLEMENT (ST2 = 0, ST1 = 1, ST0 = 0)

Scale Factor				Output														
SF3	SF2	SF1	SF0	Ø15	Ø14	Ø13	Ø12	Ø11	Ø10	Ø09	Ø08	Ø07	Ø06	Ø05	Ø04	Ø03	Ø02	Ø01	Ø00
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	I00	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	0	I01	I00	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	1	I02	I01	I00	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	0	0	I03	I02	I01	I00	1	1	1	1	1	1	1	1	1	1	1	1
0	1	0	1	I04	I03	I02	I01	I00	1	1	1	1	1	1	1	1	1	1	1
0	1	1	0	I05	I04	I03	I02	I01	I00	1	1	1	1	1	1	1	1	1	1
0	1	1	1	I06	I05	I04	I03	I02	I01	I00	1	1	1	1	1	1	1	1	1
1	0	0	0	I07	I06	I05	I04	I03	I02	I01	I00	1	1	1	1	1	1	1	1
1	0	0	1	I08	I07	I06	I05	I04	I03	I02	I01	I00	1	1	1	1	1	1	1
1	0	1	0	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	1	1	1	1	1	1
1	0	1	1	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	1	1	1	1	1
1	1	0	0	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	1	1	1	1
1	1	0	1	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	1	1	1
1	1	1	0	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	1	1
1	1	1	1	I14	I13	I12	I11	I10	I09	I08	I07	I06	I05	I04	I03	I02	I01	I00	1

TABLE 8. OUTPUT TRUTH TABLE FOR OUTPUT DISABLE (ODA) (ST2 = 0, ST1 = 0, ST0 = 1)

Scale Factor				Output														
SF3	SF2	SF1	SF0	Ø15	Ø14	Ø13	Ø12	Ø11	Ø10	Ø09	Ø08	Ø07	Ø06	Ø05	Ø04	Ø03	Ø02	Ø01	Ø00
X	X	X	X	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

X = Don't Care

TABLE 9. OUTPUT TRUTH TABLE FOR SIGN BIT PLACED AT ALL OUTPUTS (SBO) (ST2 = 0, ST1 = 0, ST0 = 0)

Scale Factor				Output														
SF3	SF2	SF1	SF0	Ø15	Ø14	Ø13	Ø12	Ø11	Ø10	Ø09	Ø08	Ø07	Ø06	Ø05	Ø04	Ø03	Ø02	Ø01	Ø00
X	X	X	X	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z

APPLICATION INFORMATION

Table 10 shows a function select that can be used in building shift/rotate arrays with only two select lines labeled X and Y. The important functions that are normally required in a shifter are included such as Rotate Right, Logic and Arithmetic Right, and Logic Left. A rotate left can be obtained by using the rotate right command and changing the scale factor select lines to the 2's complement of binary number to be rotated. A rotate left command could be added to Table 10; however, three function select lines would be required.

Figures 3 through 7 show some examples of implementing shift arrays using the selected commands shown in Table 10. **These examples are actually applicable to positive logic systems even though the drawings are shown using negative logic.** The reason is that negative logic 1s (VOH) are shifted into the vacated bit positions for logic shift right and left which is the correct format for positive logic systems.

Two different shifter arrays can be built to satisfy the selected commands shown in Table 10. Figures 3, 4, 5, and 6a show the design for implementing a 16-, 32-, 48-, and 64-bit shift/rotate array with only one package delay. The truth table and equations for designing a 64-bit shift/rotate array are shown in Figure 6b. Other bit configurations can be designed using these techniques. Any bit configuration could be designed by adjusting the scale factor inputs on certain devices in the array during rotate and shift left commands. The secret to the design of the shift/rotate array with one package delay is the wire "ANDing" of the common bit outputs and the 2's complement shift left and right operations.

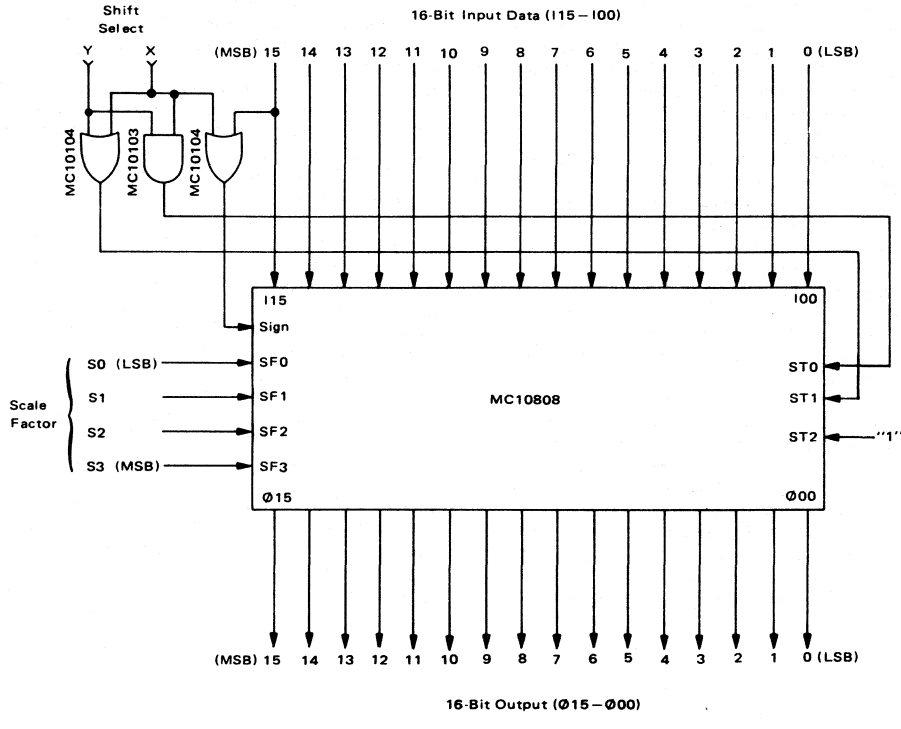
The other shifter array requires two package delays from data-in to data-out. Figure 7 is an example of a 64-bit shift/rotate with two package delays. Note that only three additional gates are required along with only ten of the shifter devices to complete the design. The top array, devices (1) through (6), shifts the input data right or left, or right rotates the data 0, 1, 2, or 3 positions as defined by the two least significant scale factor inputs, S0 and S1. The bottom array, devices (7) through (10), completes the shift array by the unique interconnections of the outputs of devices (1) through (6) to inputs of devices (7) through (10). The four upper scale factor inputs, S2 through S5, select the shifting of the input data in multiples of four positions at a time. The interconnection and design of the shifting array for devices (1) through (10) is unique to the shift function (X and Y) code shown in Table 10. The result is a highly flexible design that can be implemented with many variations in shift functions. Using the same techniques shown in Figure 7, a 128-bit shift/rotate array with two package delays could be built with twenty-four MC10808s or a 256-bit shift/rotate array which would require sixty-three MC10808s and only one additional gate package.

TABLE 10. FUNCTION SELECT DESCRIPTION THAT CAN BE USED WITH VARIOUS SHIFT/ROTATE ARRAYS

Function Select		Symbol	Description
Y	X		
0	0	RRS	Rotate Right Shift. The input data is rotated to the right at the outputs as selected by the binary number specified by the scale factor inputs.
0	1	LSR	Logic Shift Right. The input data is shifted to the right at the outputs as selected by the binary number specified by the scale factor inputs. The vacated bit positions are filled with logic 1s (negative logic).
1	0	ASR	Arithmetic Shift Right. The input data is shifted to the right at the outputs as selected by the binary number specified by the scale factor inputs. The vacated bit positions are filled with the sign bit.
1	1	LSL	Logic Shift Left. The input data is shifted to the left at the outputs as selected by the binary number specified by the scale factor inputs. The vacated bit positions are filled with logic 1s (negative logic).

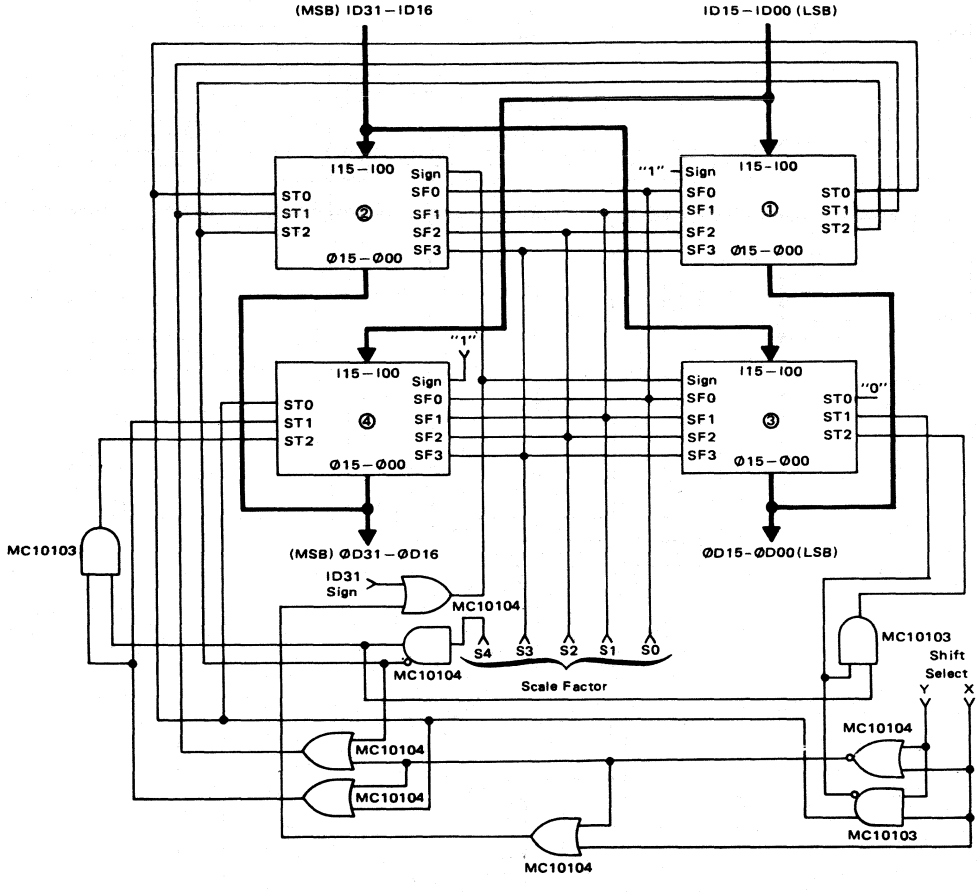
NOTE: The function select shown above is used in the shift/rotate arrays shown in Figures 3 through 7.

FIGURE 3 – 16-BIT SHIFT/ROTATE ARRAY



Negative logic is used.

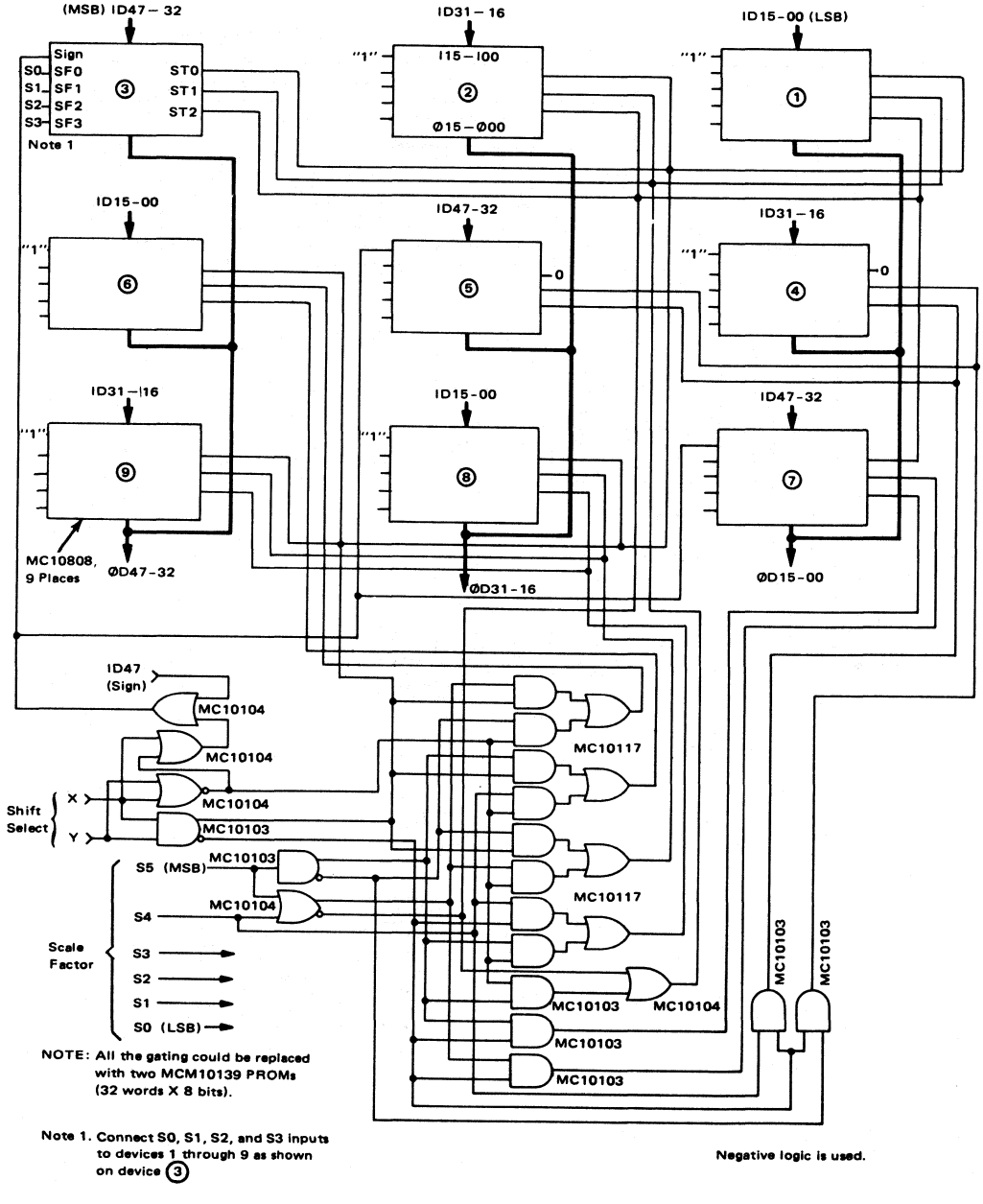
FIGURE 4 – 32-BIT SHIFT/ROTATE ARRAY



Negative Logic is used.

Shift Select		
Y	X	Shift Function
0	0	Rotate Right
0	1	Logic Right
1	0	Arithmetic Right
1	1	Logic Left

FIGURE 5 – 48-BIT SHIFT/ROTATE ARRAY WITH ONE PACKAGE DELAY



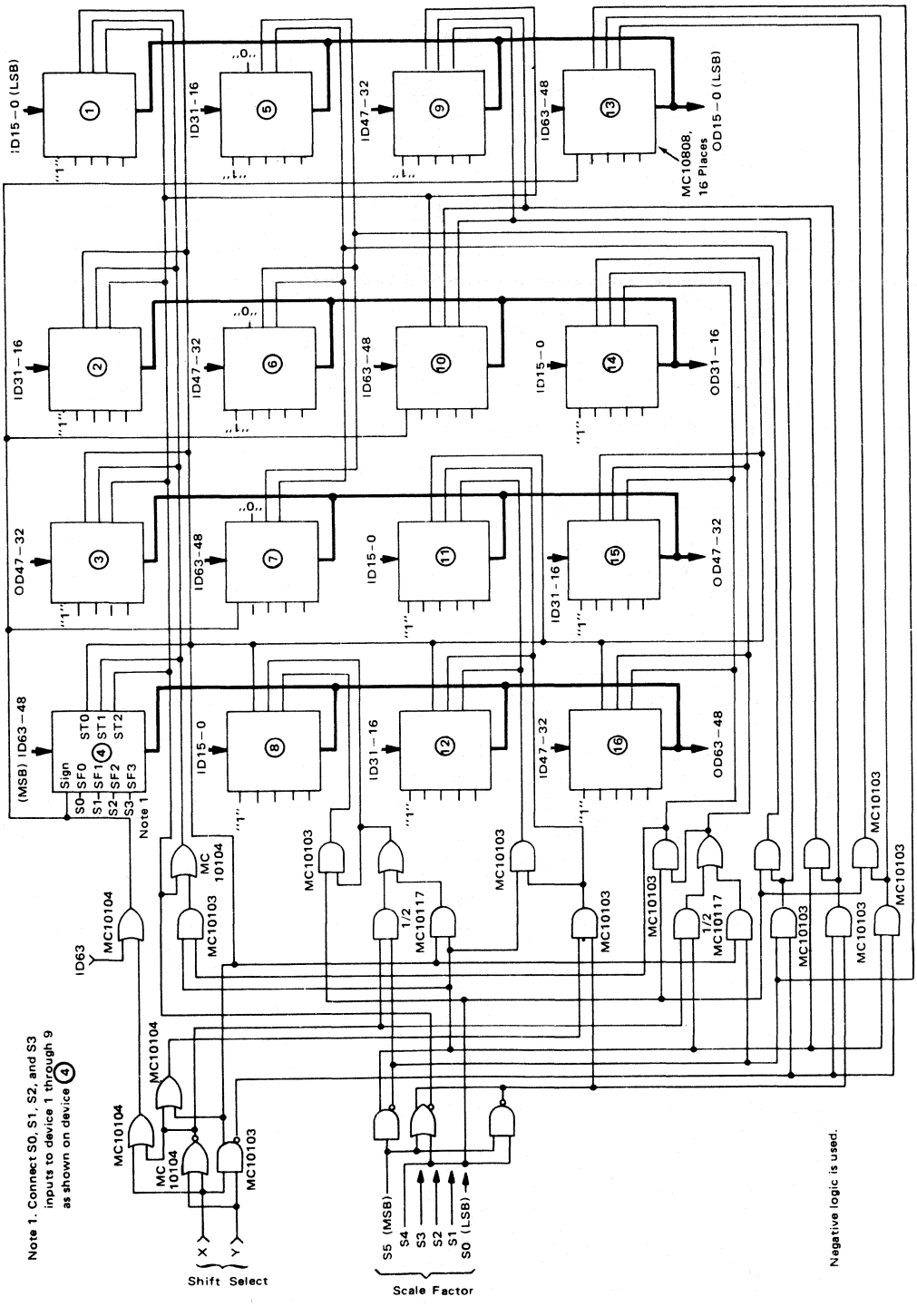


FIGURE 6a. 64-BIT SHIFT/ROTATE ARRAY WITH ONE PACKAGE DELAY

FIGURE 6b. TRUTH TABLE FOR 64-BIT SHIFTER (See Figure 6a)

Shift Select (See Table 10)			Scale Factor Bits		Sign Bit Devices 4, 7, 10, 13				Devices 1, 2, 3, 4			Device 8			Devices 5, 6, 7			Devices 11, 12			Devices 9, 10			Devices 14, 15, 16			Device 13		
Mnemonic	Y	X	S5	S4	Z4	Z7	Z10	Z13	ST2	ST1	ST0	ST2	ST1	ST0	ST2	ST1	ST0	ST2	ST1	ST0	ST2	ST1	ST0	ST2	ST1	ST0	ST2	ST1	ST0
RRS	0	0	0	0	1	1	1	1	1	1	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	1
ARS	1	0	0	0	ID63	–	–	–	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	1
LLS	1	1	0	0	1	1	1	1	1	1	1	0	0	1	0	0	0	0	0	1	0	0	1	0	1	1	0	0	1
LRS	0	1	0	0	1	1	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	1
RRS	0	0	0	1	1	1	1	1	0	0	0	1	1	0	1	1	0	0	1	0	0	1	0	0	0	0	0	0	1
ARS	1	0	0	1	ID63	ID63	–	–	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1
LLS	1	1	0	1	1	1	1	1	0	0	1	0	0	1	0	0	0	0	1	1	0	0	0	1	1	1	0	0	1
LRS	0	1	0	1	1	1	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1
RRS	0	0	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	0	1	1	0	0	1	0	0	1	0
ARS	1	0	1	0	ID63	ID63	ID63	–	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0
LLS	1	1	1	0	1	1	1	1	0	0	1	0	1	1	0	0	0	1	1	1	0	0	0	0	0	1	0	0	0
LRS	0	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0
RRS	0	0	1	1	1	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	0
ARS	1	0	1	1	ID63	ID63	ID63	ID63	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
LLS	1	1	1	1	1	1	1	1	0	0	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
LRS	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

– = Don't Care

$Z_4 = ID63 + [X + (\overline{X+Y})]$

$Z_7 = Z_4$

$Z_{10} = Z_4$

$Z_{13} = Z_4$

Device 8

$$\begin{cases} ST0_8 = X \cdot Y \\ ST1_8 = S5 \cdot (X \cdot Y) + \overline{S5} \cdot (\overline{X+Y}) \\ ST2_8 = ST1_8 \cdot S4 \end{cases}$$

Devices 11, 12

$$\begin{cases} ST0_{11} = X \cdot Y \\ ST1_{11} = (S5 \oplus S4) \cdot (\overline{X \oplus Y}) \\ ST2_{11} = ST1_{11} \cdot S5 \end{cases}$$

Devices 14, 15, 16

$$\begin{cases} ST0_{14} = X \cdot Y \\ ST1_{14} = \overline{S5} \cdot (X \cdot Y) + S5 \cdot (\overline{X+Y}) \\ ST2_{14} = ST1_{14} \cdot S4 \end{cases}$$

Devices 1, 2, 3, 4

$$\begin{cases} ST0_1 = X \cdot Y \\ ST1_1 = (\overline{S5+S4}) + ST2_{14} \cdot S5 \\ ST2_1 = (\overline{S4+S4}) \end{cases}$$

Device 13

$$\begin{cases} ST1_{13} = S5 \cdot (\overline{XY}) \\ ST2_{13} = ST1_{13} \cdot S4 \\ ST0_{13} = \overline{S5} \end{cases}$$

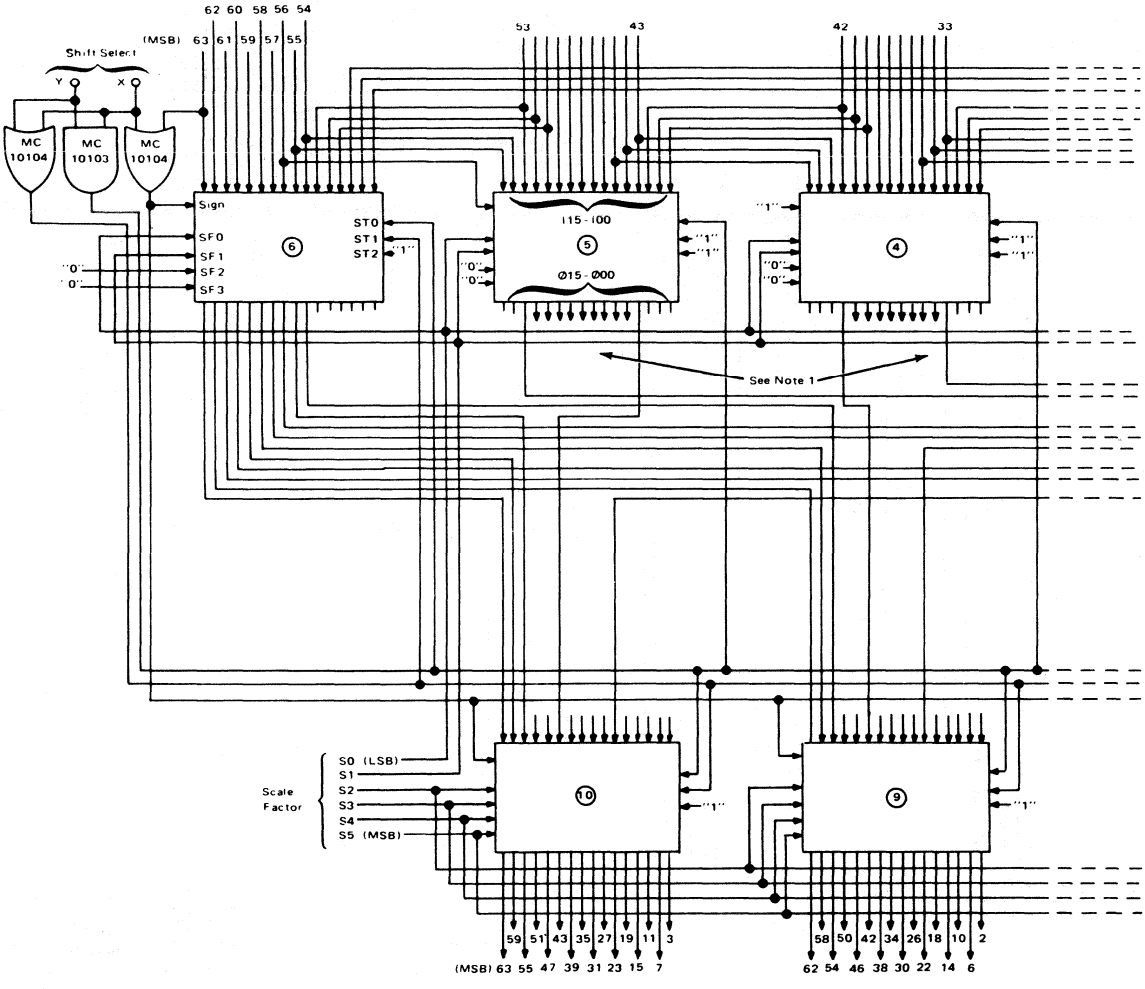
Devices 9, 10

$$\begin{cases} ST1_9 = (S5 \oplus S4) \cdot \overline{XY} \\ ST2_9 = ST1_9 \cdot S5 \\ ST0_9 = S5 \cdot S4 \end{cases}$$

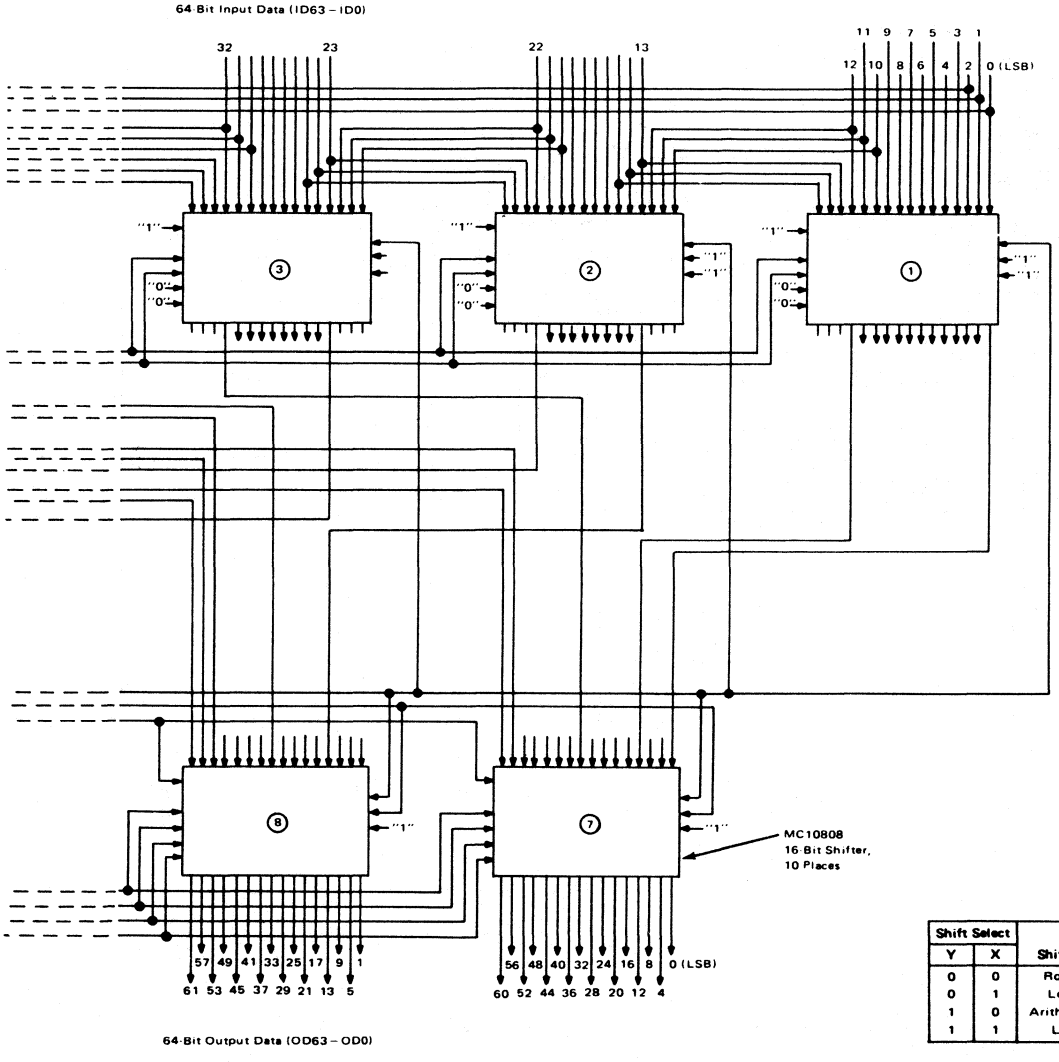
Devices 5, 6, 7

$$\begin{cases} ST1_5 = \overline{S5} \cdot (\overline{X \cdot Y}) \\ ST2_5 = ST1_5 \cdot S4 \end{cases}$$

FIGURE 7 – 64-BIT SHIFT/ROTATE ARRAY



NOTE 1 The outputs of devices ① through ⑤ are connected to devices ⑦ through ⑩ in the same manner as shown for connecting the outputs of device ⑥.



Shift Select		Shift Function
Y	X	
0	0	Rotate Right
0	1	Logic Right
1	0	Arithmetic Right
1	1	Logic Left

RECOMMENDED OPERATING CONDITIONS

Parameter	Symbol	Value	Unit
Supply Voltage (V_{CC} = 0 Volts)	V_{EE}	-4.68 to -5.72	Vdc
Operating Temperature (Functional)	T_A	-30 to +85	°C
Output Drive	–	50 Ω to -2.0 Vdc	–

ELECTRICAL CHARACTERISTICS

Each MECL 10,000 series circuit has been designed to meet the dc specifications shown in the test table, after thermal equilibrium has been established. The circuit is in a test socket or mounted on a printed circuit board and transverse airflow greater than 500 linear fpm is maintained. Outputs are terminated through a 50-ohm resistor to -2.0 volts. Test procedures are shown for only one input, or for one set of input conditions. Other inputs tested in the same manner.

@ Test Temperature	TEST VOLTAGE VALUES (Volts) V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{EE}
-30°C	-0.890	-1.890	-1.205	-1.500	-5.2
+25°C	-0.810	-1.850	-1.105	-1.475	-5.2
+85°C	-0.700	-1.825	-1.035	-1.440	-5.2

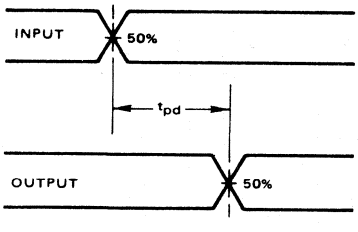
Characteristic	Symbol	Pin Under Test	MC10808 TEST LIMITS -30°C Min	-30°C Max	+25°C Min	+25°C Typ	+25°C Max	+85°C Min	+85°C Max	Unit	VOLTAGE APPLIED TO PINS LISTED BELOW V_{IHmax}	V_{ILmin}	V_{IHAmin}	V_{ILAmax}	V_{EE}	(V_{CC}) Gnd
Power Supply Drain Current	I_{EE}	1,24	–	–	–	278	348	–	–	mAdc	–	–	–	–	1,24	–
Input Current	I_{inH}	22	–	–	–	–	330	–	–	μAdc	22	–	–	–	1,24	7,12,
		2	–	–	–	–	390	–	–	↓	2	–	–	–	↓	17,36
		44	–	–	–	–	435	–	–		44	–	–	–		↓
	I_{inL}	44	–	–	0.5	–	–	–	–		–	44	–	–		
Logic "0" Output Voltage	V_{OH}	6	-1.060	-0.890	-0.960	–	-0.810	-0.890	-0.700	Vdc	44,45, 46,47	–	–	–	1,24	7,12, 17,36
Logic "1" Output Voltage	V_{OL}	6	-1.890	-1.675	-1.850	–	-1.650	-1.825	-1.615	Vdc	45,46, 47	44	–	–	1,24	7,12, 17,36
Logic "0" Threshold Voltage	V_{OHA}	6	-1.080	–	-0.980	–	–	-0.910	–	Vdc	44,46, 47	–	45	–	1,24	7,12, 17,36
Logic "1" Threshold Voltage	V_{OLA}	6	–	-1.655	–	–	-1.630	–	-1.595	Vdc	45,46, 47	–	–	44	1,24	7,12, 17,36

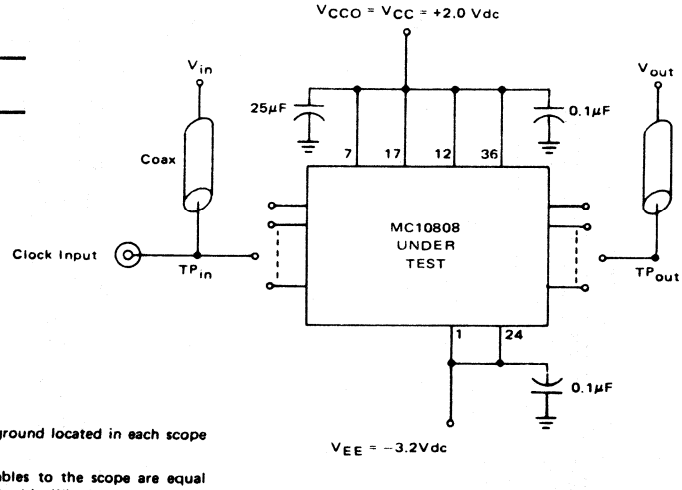
PROPAGATION DELAY TIMES
(Nanoseconds at 0 to 70°C to
Data Outputs (Ø15–Ø00)

	Typ
Data Inputs, I15–I00	6
Sign Bit, Z	6
Scale Factor, SF0–SF3	12
Shift Type, ST0–ST2	12

SWITCHING TIME TEST CIRCUIT AND WAVEFORMS

PROPAGATION DELAYS

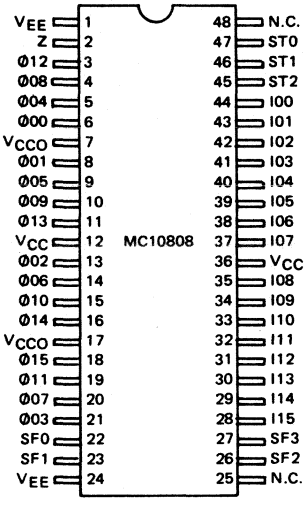




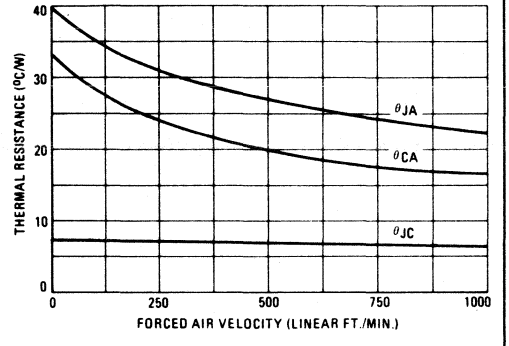
50 ohm termination to ground located in each scope channel input.

All input and output cables to the scope are equal lengths of 50 ohm coaxial cable. Wire length should be <¼ inch from TP_{in} to input pin and TP_{out} to output pin.

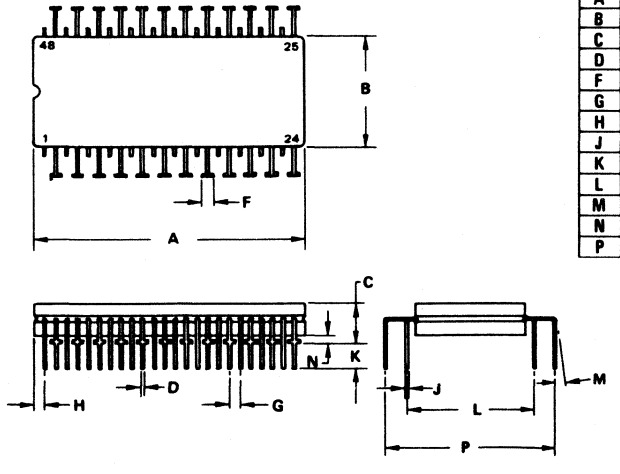
PIN ASSIGNMENT



THERMAL CHARACTERISTICS
(TYPICAL)



PACKAGE DIMENSIONS

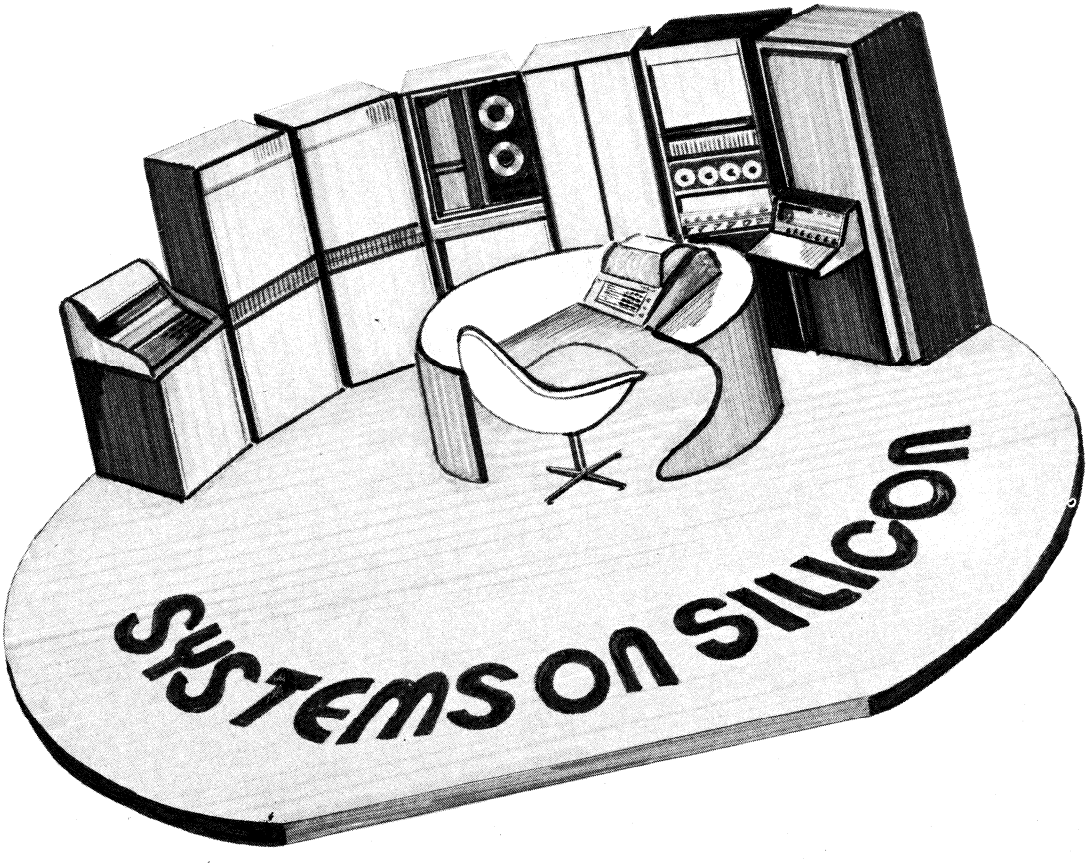


DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	31.24	32.26	1.230	1.270
B	12.70	13.72	0.500	0.540
C	4.57	5.59	0.180	0.220
D	0.38	0.53	0.015	0.021
F	1.14	1.40	0.045	0.055
G	1.27 BSC		0.050 BSC	
H	1.02	1.52	0.040	0.060
J	0.20	0.30	0.008	0.012
K	2.54	3.30	0.100	0.130
L	15.24 BSC		0.600 BSC	
M	—	7°	—	7°
N	0.51	1.52	0.020	0.060
P	20.32 BSC		0.800 BSC	

Case 725-01

A socket for the QUIL package is available from ELECTRONIC MOLDING CORPORATION. (Part number 7178-295-5)

QUIL is a trademark of Motorola Inc.


SYSTEMS ON SILICON