May 1975

systems reference and data sheets

MOTOROLA Semiconductor Products Inc.

THE M6800 MICROCOMPUTER FAMILY

The M6800 Family of parts has been designed to achieve several goals in microcomputer architecture. Primarily, these are:

- minimization of required components
- minimization of required support packages
- simplicity of interface
- simplicity of power requirements
- system throughput.

This discussion is intended to demonstrate how these goals have been accomplished by examining the construction and operation of a small microcomputer using M6800 components. Simple examples will also be presented to clarify the operating principles of the system.

A SMALL M6800 MICROCOMPUTER SYSTEM

Figure 1 shows an M6800 Microcomputer System with 1024 bytes of Read Only Memory (ROM) for the storage of instructions and permanent data tables, 128 bytes of Random Access Memory (RAM) for the storage of temporary data, and 2 I/O interfaces. Note that the entire system has been implemented with six NMOS packages from the M6800 family.

MC6800 MICROPROCESSOR BUS INTERFACE

Figure 2 is a symbolic representation of the MC6800 microprocessor showing its controls and its MicroBus interface.

The data and address bus of the processor operate at standard TTL levels and can drive on standard TTL load plus 130 pF. In this application, that drive is sufficient to run with all the other M6800 parts included with no buffers. All the MicroBus parts have this same drive capability and operate at the same standard TTL levels.

FIGURE 1 – A Small M6800 Microcomputer System

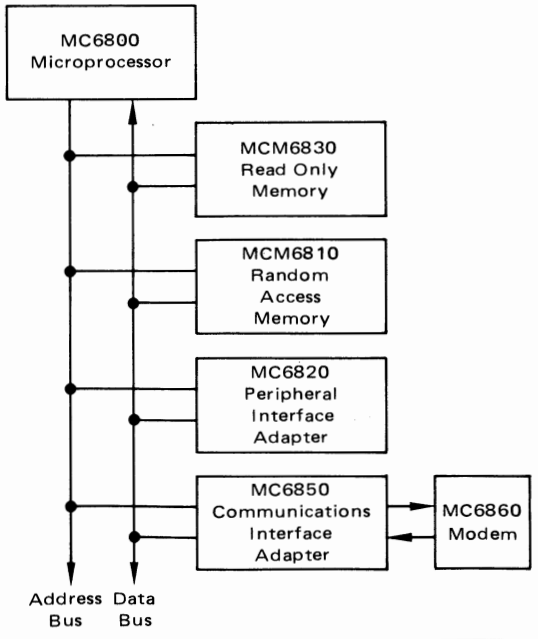
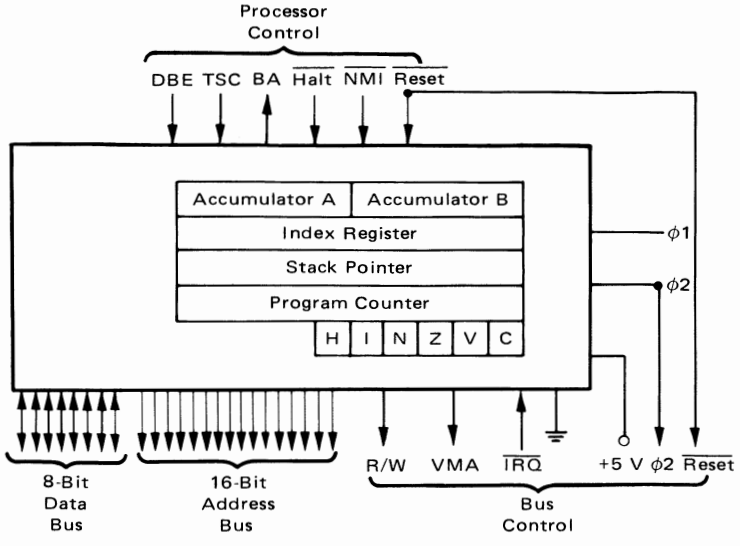


FIGURE 2 – Processor Bus Interface



MicroBus is a trademark of Motorola Inc.

The M6800 system has a single address space. The processor references all other bus components as memory locations. Selection of the various memory or I/O packages is accomplished by the state of the address bus alone. This results in considerable simplification of the required bus controls. The bus control signals generated and received by the processor are:

VMA – which in its active state indicates that a data transfer will take place on the current machine cycle.

R/W – which indicates the direction of data transfers.

$\overline{IRQ}$ – which indicates the presence or absence of an interrupt request from one of the interfaces.

In addition, the following external signals should also be supplied for bus control:

$\phi 2$ – All data transfers take place during the $\phi 2$ clock cycle. Therefore this signal will be used as an enable input for all memories and interfaces.

$\overline{Reset}$ – The interfaces will require this input to insure orderly system start-up.

All of the MicroBus components will operate on the same +5 V power supply and ground.

MC6800 MICROPROCESSOR CONTROLS

The clock for the system will operate at a maximum frequency of 1 MHz. The two phases are non-overlapping square wave complements. At this clock frequency, the MC6800 processor achieves a minimum instruction time of 2 μs. The two clocks are the only inputs to the processor that do not operate at standard TTL levels.

The lines required to control the processor are:

$\overline{Reset}$ – which initializes the processor as well as the interfaces and forces transfer of control to the user-defined starting address.

$\overline{NMI}$ – which is a separate non-maskable interrupt for power-fail situations, real time clock inputs, or other high priority signals.

$\overline{Halt}$ – which is used to externally halt the processor, usually for Direct Memory Access. Halting the processor places the address and R/W lines in their high impedance mode.

BA – which acknowledges that the address and R/W lines are in their high impedance modes.

TSC – which in its active state will place the address and R/W lines in their high impedance mode.

DBE – which in its active state will remove the data bus from its high impedance mode. Since all data transfers take place in the $\phi 2$ clock cycle, it is convenient in this simple system to drive DBE with the $\phi 2$ clock signal.

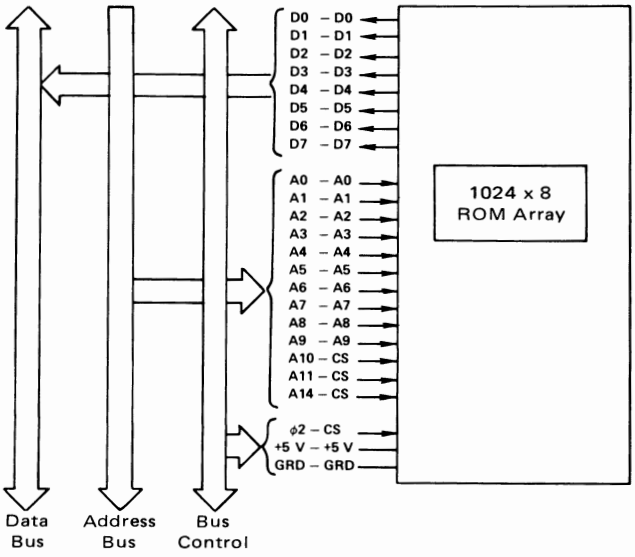
Figure 2 shows all the signals necessary to control the processor and all the pins on the processor package. Note that each signal is on a separate pin. The simplicity of the processor controls results directly from the simplicity of the M6800 MicroBus architecture.

MCM6830 ROM BUS INTERFACE

The ROM bus interface shown in Figure 3 demonstrates the simplicity of interface in the M6800 system. Since all MicroBus components operate at the same TTL levels and with the same drive capability, data, address, and control lines can be connected without buffers. Timing of the memories has been set to permit simple operation at full speed with the processor.

The chip selects of the MCM6830 have been used in this example to partially decode the system address lines. In small and medium sized systems, this partial address decoding will be sufficient to distinguish all packages in the system without using any additional address decoding packages.

FIGURE 3 – MCM6830 ROM Bus Interface

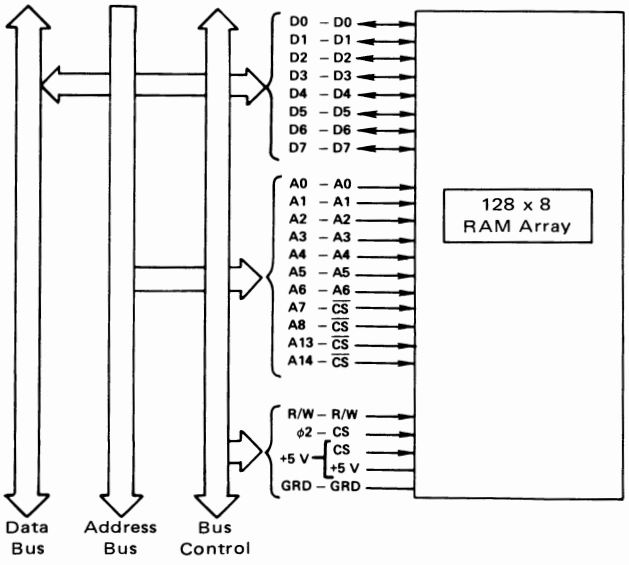


MCM6810 RAM BUS INTERFACE

The RAM bus interface (Figure 4) is as straightforward as that of the ROM. All required outputs may be connected directly without drivers.

The four negative and two positive chip selects on the MCM6810 are used to provide partial address decoding in the system. In this system, the decoding possible with these lines was sufficient to distinguish the RAM.

FIGURE 4 – MCM6810 RAM Bus Interface



MC6820 PIA BUS INTERFACE

The PIA in M6800 systems is used to provide 16 bits of external interface and four control lines at addressable locations in standard system memory. Naturally, the I/O bits are accessed in two words of 8 bits each, but each I/O bit is individually programmable to act as either an input or an output. All operating characteristics of the interface are established by writing from the processor to the Data Direction and Control Registers of the PIA. This is required at the time of system Reset and permitted at any other time.

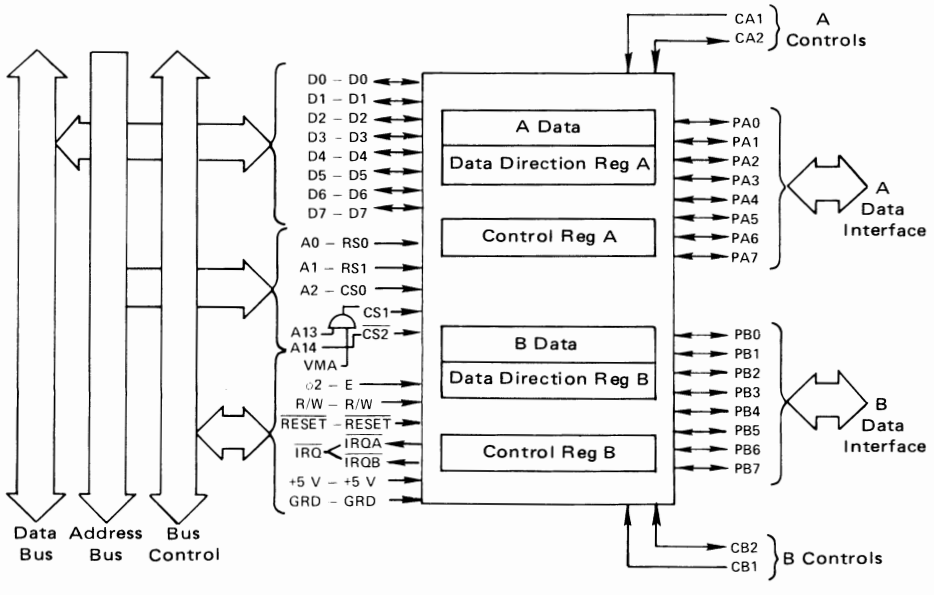
In the previous discussion of the system bus, no mention was made of I/O control lines. There are none on the system bus. All devices connected to the processor have the same interface as a memory — a fundamental characteristic of the MicroBus system. The interface for the PIA is shown in Figure 5. The register select lines (RS0 and RS1) serve the same purpose in the PIA as the address lines do in a memory. For this reason, they are normally connected directly to the low-order address lines of the system. Chip selects are again provided for partial address decoding. In this system, no other decoders are required to distinguish interface packages.

Other lines not used in the memory interface are also shown in the PIA interface. VMA is used to protect the device from a spurious read operation during a machine cycle with no intended memory reference. This is necessary, since a read operation can change the state of the PIA. $\overline{IRQA}$ and $\overline{IRQB}$ from the PIA are OR tied to the system $\overline{IRQ}$ line. Since the PIA may be used to detect incoming interrupt signals on any of its control lines, this connection must be made in order to initiate the interrupt sequence at the processor. The $\overline{IRQ}$ will be pulled down by the PIA following detection of an active transition on any control line which has been enabled as a system interrupt. It will hold $\overline{IRQ}$ down until the interrupt is serviced. Thus no interrupts will ever be lost to the system even if the interrupt mask is set at the processor. Finally, $\overline{RESET}$ is used to initialize the interface.

MC6850 ACIA BUS INTERFACE

The ACIA provides a special purpose serial interface for asynchronous communications. It handles all formatting tasks such as insertion and detection of start, stop, and parity bits and may be programmed for all common serial formats. These operating characteristics are established by writing the appropriate control pattern to the ACIA control register. Register status and all error conditions are indicated in a separate status register.

FIGURE 5 – MC6820 PIA Bus Interface



The bus interface of the ACIA (Figure 6) is the same as for the PIA. No unique controls are required. The interface itself manages the serial communications lines and modem control lines. The $\overline{IRQ}$ output of the ACIA may be used to request service from the processor on the input-register-full or output-register-empty conditions. Since both registers are double-buffered, the processor would have a full word time (33 ms at 300 bps) to respond to these conditions.

MODEM INTERFACE TO THE ACIA

The MC6860 Modem provides digital to FSK and FSK to digital conversion for data rates of 0-600 bps. As shown in Figure 7, it can be interfaced directly to the ACIA to implement a serial data communications link over a voice grade channel.

INPUT/OUTPUT CONCEPTS OF THE M6800 SYSTEM

The I/O concepts of a particular microcomputer will determine the control signals required in the system and interface hardware, the extent to which the processor itself acts as the I/O controller, and the software options available for dealing with data at the ports.

Motorola's approach in the M6800 system has centered on the MicroBus structure in which both memory and I/O interfaces exist in the same address space. The interfaces are individually programmable to provide I/O management. They generate and receive all I/O control signals in the system. They are programmed by control words passed over the system bus. The result of this is minimization of hardware controls required on the system bus together with simplicity and generality of I/O programming.

I/O EXAMPLE

The PIA shown in Figure 8 provides a simple example of the role of the interface in the M6800 system. The eight bits of the A side are used as one parallel byte of input data, while the eight bits of the B side are used as an output byte. These operations are set by placing all zeroes (or ones) in the Data Direction Registers with store operations from the processor. The CA1 line is used as an input Data Ready signal. An active transition (positive in this example) will set its status bit (CRA7) and pull down $\overline{IRQA}$ which is OR tied to the system $\overline{IRQ}$ line. The CA2 line is programmed to go low on a read of the A data and remain low until the next active CA1 transition. Thus it provides a Data Accepted signal. The CB1 line is used as Data Request signal. It will also detect a positive transition, set its status bit (CRB7) and pull down $\overline{IRQB}$ which is OR tied to the system interrupt request line. The CB2 line is programmed to go low on a write of the B data and remain low until the next active transition on CB1. In this manner, it provides an output Data Ready signal. Again, all control line functions are programmed by establishing the patterns shown in the control registers with store operations from the processor

FIGURE 6 – MC6850 ACIA Bus Interface

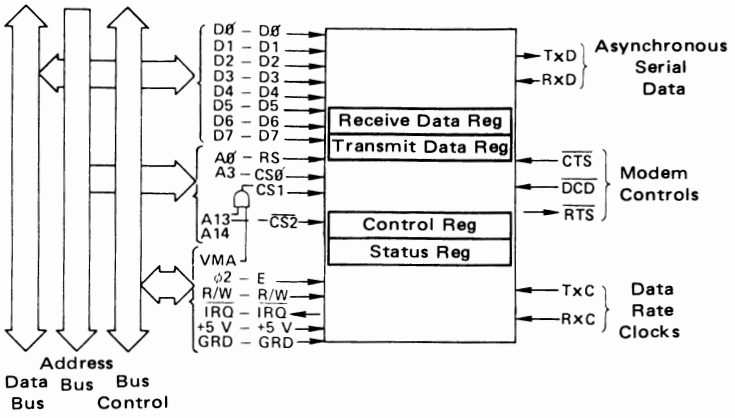


FIGURE 7 – MC6860 Modem to MC6850 ACIA Interface

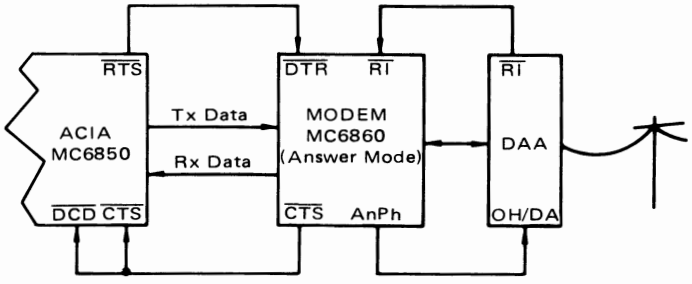
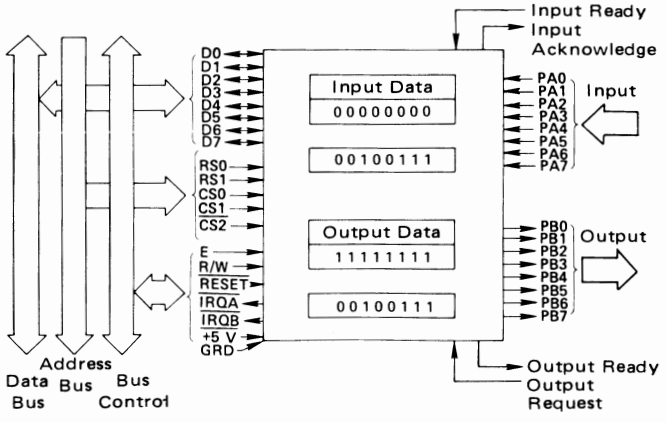


FIGURE 8 – PIA Application Example



Figures 9A thru 9D show a full sequence of I/O operations. Note that all required handshake signals are generated and received locally at the interface. The processor never has to look at the interface until service is required. It never waits for input data to become valid or for output data to be accepted. By reducing processor load in this manner, the PIA significantly increases system throughput.

In summary, the MicroBus concept of the M6800 system simplifies both the hardware and the software required in microcomputer applications. I/O management is provided by individually programmable peripheral interfaces. The processor therefore takes on its appropriate role of system executive – programming the actual operation of the peripheral interfaces and servicing them only when required.

FIGURE 9 – PIA Input/Output Sequence

FIGURE 9A

Data is presented on the A side by an external device. The Input Ready signal sets a status bit and pulls down $\overline{IRQA}$. The Interrupt response routine will identify this interrupt by polling status bits.

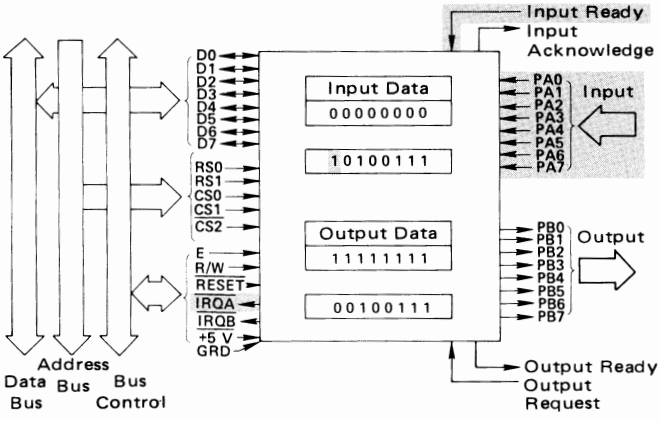


FIGURE 9B

The Interrupt Response routine reads the A data. This action automatically clears the interrupt and sends the Input Acknowledge signal.

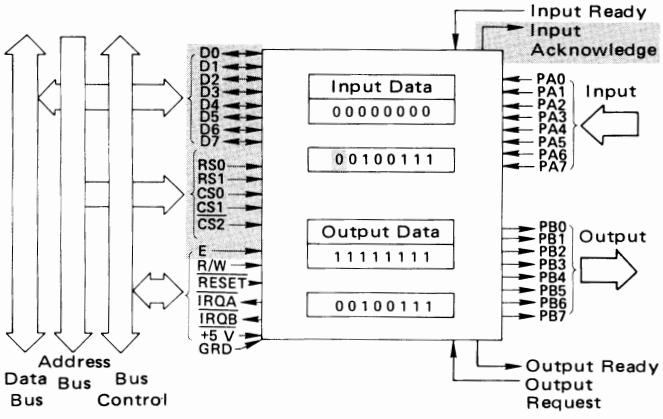


FIGURE 9C

An external device requests data on the B side with Output Request. This sets a status bit and pulls down $\overline{IRQB}$.

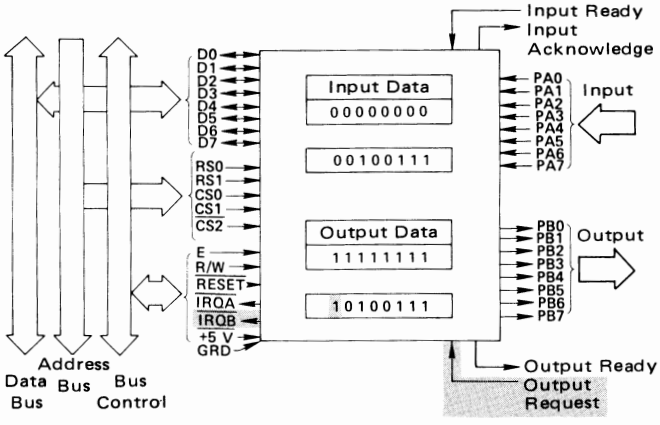
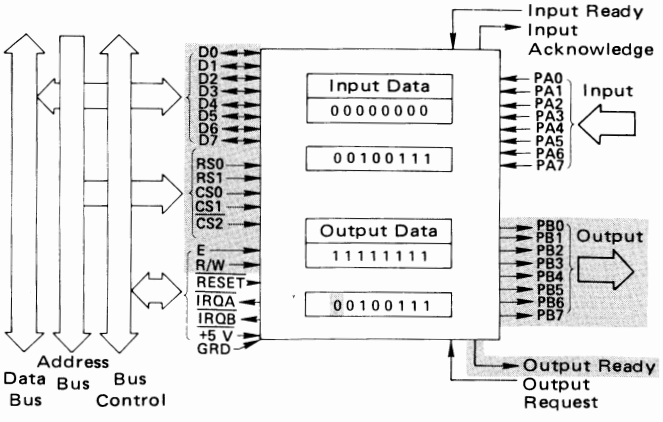


FIGURE 9D

The Interrupt Response routine identifies the interrupt by checking status bits. A read operation is used to clear the interrupt. Writing output data to the B side presents the data to the external device and automatically generates the Output Ready signal.



PROGRAMMING THE PERIPHERALS

In the previous example, two distinct functions were relegated to the software: setting the data direction and control register patterns at the time of system initialization, and handling interrupts.

Setting the data direction and control register patterns establishes the operating characteristics of the I/O device. The initialization program for this example is shown in Figure 10. It will establish the patterns shown in Figure 8. Recall that on $\overline{\text{Reset}}$ all lines of the PIA are initialized as inputs and all interrupts disabled. Thus the program shown only modifies the registers from these states.

FIGURE 10 – Initialization Program

```
*
*       CONFIGURE PIA AT ADDRESS $2004
*       TO ACCEPT INPUT ON A SIDE, PRESENT
*       OUTPUT ON B SIDE, AND HANDLE ALL
*       CONTROLS. ALL REGISTERS ARE ZERO ON RESET
*
RESET   LDA  A  #%00100111    SET CONTROLS
        STA  A  $2005         STORE IN CRA
        COM     $2006         SET B FOR OUTPUT
        STA  A  $2007         STORE CONTROLS IN CRB
        •
        •
        •
```

Handling interrupts in a system such as has been described is a problem of software polling. A polling sequence for this example is shown in Figure 11. The corresponding program is shown in Figure 12.

Such a polling approach is usually the lowest cost alternative for identifying interrupts, but may in some instances be too slow. For such applications, hardware may be added to the system to achieve priority encoding of the various interrupt requests. The encoded value of the interrupt request can then be used as a system address to transfer control to the appropriate response routine. This is referred to as "interrupt vectoring".

FIGURE 11 – Polling Sequence

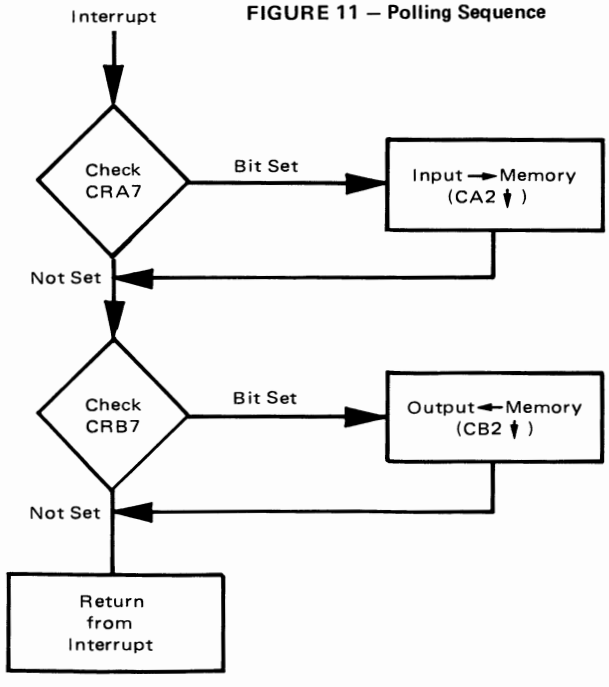


FIGURE 12 – Polling Program

```
*
*       POLL PIA ON INTERRUPT
*       AND SERVICE
*
POLL    LDA  A        $2005
        BPL   POLL2
        •
        •     (INPUT CHARACTER HANDLING)
        •
POLL2   LDA  A        $2007
        BPL   POLL3
        •
        •     (OUTPUT CHARACTER HANDLING)
        •
POLL3   RTI
```

PARTIAL ADDRESS DECODING USING CHIP SELECTS

In any memory system, the chip selects of the memories themselves can be used to achieve a partial decode of the high order addresses. For small systems this may be sufficient to discriminate among all memory locations. As a simple example, consider the memory system shown in Table 1. Each memory section contains 1K bytes and has two chip select inputs. One of the chip selects will be needed for the $\phi 2$ input. The other can be connected to any of the six remaining address lines, permitting six sections of memory to be discriminated as shown in the address table.

There are, however, some problems one should be aware of in such systems. First, note that there are several bit patterns which will enable a given section of memory. Section 1, for instance, could be enabled by any of the patterns:

0000 01XX XXXX XXXX
0000 11XX XXXX XXXX
0001 01XX XXXX XXXX
etc.

However, only the first of these discriminates between Section 1 and all other sections. The others are redundant and should be avoided since they would enable two bus components simultaneously. In programming, of course, it is possible to avoid references to redundant addresses. During non-memory reference cycles of the processor, though, the state of the address line is not known. For that reason, when redundancy in addressing is possible, VMA should be gated with one of the enable inputs. In Table 1, it is shown gated with $\phi 2$.

TABLE 1 – HYPOTHETICAL 6K x 8 MEMORY SYSTEM USING CHIP SELECTS FOR ADDRESS DECODING

$\overline{CS2}$ = VMA • $\phi 2$

Memory Section	Unique Address	CS1
1	1024-2047	A10
2	2048-3072	A11
3	4096-5120	A12
4	8192-9216	A13
5	16384-17408	A14
6	32768-33792	A15

Another approach to redundancy is to provide sufficient chip select lines to permit full decoding of a subset of the high order address lines. This is the approach taken for the memories in the M6800 system as shown in Table 2. Consider the ROMs, for instance. One chip select is used for the $\phi 2$ signal, and another for address line A14 which discriminates ROM from other classes of memory. Lines A10 and A11 are fully decoded using the remaining chip selects. While there are still several bit patterns which might enable a given ROM location, none are redundant. This means that it is impossible either by the chance state of the address lines in a non-memory reference cycle or by a programming error to enable two ROM packages simultaneously.

In the RAMs, line A14 is connected to a negative enable, thus discriminating between RAM and ROM; i.e., RAM and ROM addresses will never be redundant. Line A13 is also connected to a negative enable and is used to provide the same discrimination between RAM and I/O devices that line A14 provides between ROM and other classes of memory. Lines A7 and A8 are fully decoded using the remaining enables. Again, despite the fact that there are several bit patterns which may enable a given RAM, none are redundant.

Finally, the I/O devices are selected with line A14 (connected to $\overline{CS2}$) in its low state, line A13 (connected to CS1) in its high state, and one of the set A2-A12 connected to CS0. The I/O devices are always discriminated from other classes of memory using this system, but among themselves, their addresses are potentially redundant. For this reason VMA is shown gated with line A13 to protect against redundant addresses which may appear in non-memory reference cycles. Care should also be taken in programming to avoid the redundant bit patterns.

TABLE 2 – EXPANDED M6800 SYSTEM USING CHIP SELECTS FOR ADDRESS DECODING

RAMs

Address	CS	CS	$\overline{CS}$	$\overline{CS}$	$\overline{CS}$	$\overline{CS}$
0-127	$\phi 2$	+5 V	A14	A13	A7	A8
128-255	$\phi 2$	A7	A14	A13	Gnd	A8
256-383	$\phi 2$	A8	A14	A13	A7	Gnd

ROMs

Address	CS	CS	$\overline{CS}$ or CS		$\overline{CS}$ or CS	
7000-73FF	$\phi 2$	A14	A9	–	A10	–
7400-77FF	$\phi 2$	A14	–	A9	A10	–
7800-7BFF	$\phi 2$	A14	A9	–	–	A10
7C00-7FFF	$\phi 2$	A14	–	A9	–	A10

INTERFACES

Address	E	$\overline{CS2}$	CS1	CS0
2004-2007	$\phi 2$	A14	A13 • VMA	A2
2008-200B	$\phi 2$	A14	A13 • VMA	A3

MOTOROLA Semiconductors
BOX 20912 • PHOENIX, ARIZONA 85036

MC6800

Advance Information

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.

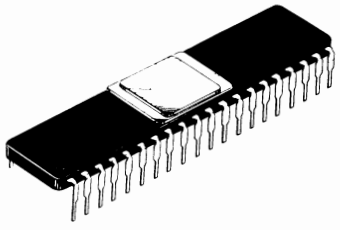
The MC6800 is capable of addressing 65 k bytes of memory with its 16-bit address lines. The 8-bit data bus is bidirectional as well as 3-state, making direct memory addressing and multiprocessing applications realizable.

- Eight-Bit Parallel Processing
- Bi-Directional Data Bus
- Sixteen-Bit Address Bus — 65 k 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
- Clock Rates as High as 1 MHz
- Simple Bus Interface Without TTL
- Halt and Single Instruction Execution Capability

MOS

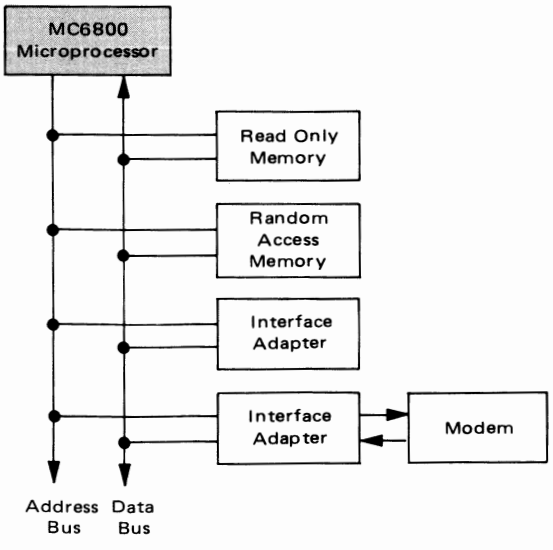
(N-CHANNEL, SILICON-GATE)

MICROPROCESSOR

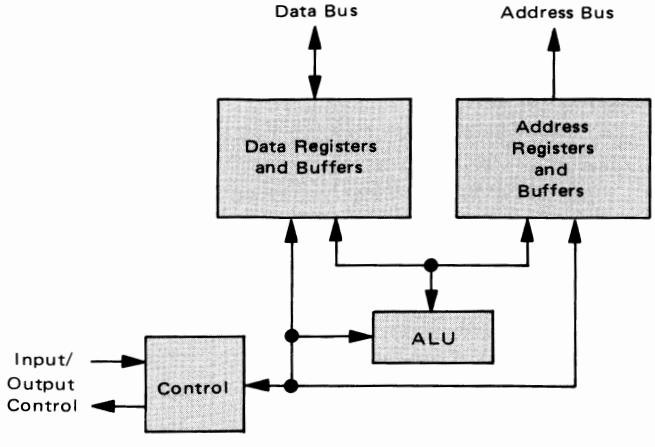


CERAMIC PACKAGE
CASE 699

M6800 MICROCOMPUTER FAMILY BLOCK DIAGRAM



MC6800 MICROPROCESSOR 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

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 =25°C unless otherwise noted.)

Characteristic		Symbol	Min	Typ	Max	Unit
Input High Voltage (Normal Operating Levels)						Vdc
	Logic	V_{IH}	V_{SS} + 2.4	–	V_{CC}	
	$\phi1,\phi2$	V_{IHC}	V_{CC}-0.3	–	V_{CC} + 0.1	
Input Low Voltage (Normal Operating Levels)						Vdc
	Logic	V_{IL}	V_{SS} - 0.3	–	V_{SS} + 0.4	
	$\phi1,\phi2$	V_{ILC}	V_{SS} - 0.1	–	V_{SS} + 0.3	
Clock Overshoot/Undershoot — 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 High Threshold Voltage		V_{IHT}	V_{SS} + 2.0	–	–	Vdc
	$\overline{Reset}$,$\overline{NMI}$,$\overline{Halt}$,$\overline{IRQ}$,Data					
Input Low Threshold Voltage		V_{ILT}	–	–	V_{SS} + 0.8	Vdc
	$\overline{Reset}$,$\overline{NMI}$,$\overline{Halt}$,$\overline{IRQ}$,Data					
Input Leakage Current		I_{in}				μAdc
(V_{in} = 0 to 5.25 V, V_{CC} = 5.25 V)	Logic*		–	–	2.5	
	$\phi1,\phi2$		–	–	100	
Three-State (Off State) Input Current		I_{TSI}				μAdc
(V_{in} = 0.4 to 2.4 V, V_{CC} = max)	Data		–	–	10	
	A0-A15,R/W		–	–	100	
Output High Voltage		V_{OH}	V_{SS} + 2.4	–	–	Vdc
(I_{Load} = -100 μAdc, V_{CC} = min)						
Output Low Voltage		V_{OL}	–	–	V_{SS} + 0.4	Vdc
(I_{Load} = 1.6 mAdc, V_{CC} = min)						
Power Dissipation		P_D	–	0.600	1.2	W
Capacitance #		C_{in}				pF
(V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	Logic		–	–	10	
	Data,TSC		–	–	15	
	$\phi1,\phi2$		80	120	160	
	A0-A15,R/W	C_{out}	–	–	12	pF
Frequency of Operation		f	0.1	–	1.0	MHz
Clock Timing (Figure 1)						
Cycle Time		t_{cyc}	1.0	–	10	μs
Clock Pulse Width		$PW_{\phi H}$				ns
(Measured at V_{CC} - 0.3 V)	$\phi1$		430	–	4500	
	$\phi2$		450	–	4500	
Total $\phi1$ and $\phi2$ Up Time		t_{ut}	940	–	–	ns
Rise and Fall Times	$\phi1,\phi2$	t_r,t_f	5.0	–	50	ns
(Measured between V_{SS} + 0.3 V and V_{CC} - 0.3 V)						
Delay Time or Clock Overlap		t_d	0	–	9100	ns
(Measured at V_{SS} + 0.5 V)						
Overshoot/Undershoot Duration		t_{OS}	0	–	40	ns

*Except $\overline{IRQ}$ and $\overline{NMI}$, which require 3 kΩ pullup load resistors for wire-OR capability at optimum operation.

Capacitances are periodically sampled rather than 100% tested.

READ/WRITE TIMING Figures 2 and 3, f = 1.0 MHz, Loading = 130 pF and one TTL Load except VMA and BA Loading = 30 pF and one TTL Load.

Characteristic	Symbol	Min	Typ	Max	Unit
Read/Write Setup Time from MPU	T_{ASR}	–	100	300	ns
Address Setup Time from MPU	T_{ASC}	–	200	300	ns
Memory Read Access Time $t_{cyc} - (T_{ASC} + T_{DSU} + t_r)$	T_{ACC}	–	–	575	ns
Data Setup Time	T_{DSU}	100	–	–	ns
Address Setup Time from MPU for VMA	T_{VSC}	–	150	300	ns
Data Hold Time	T_H	10	30	–	ns
Enable High Time for DBE Input	T_{EH}	470	–	–	ns
Data Setup Time from MPU	T_{ASD}	–	150	200	ns

FIGURE 1 – CLOCK TIMING WAVEFORM

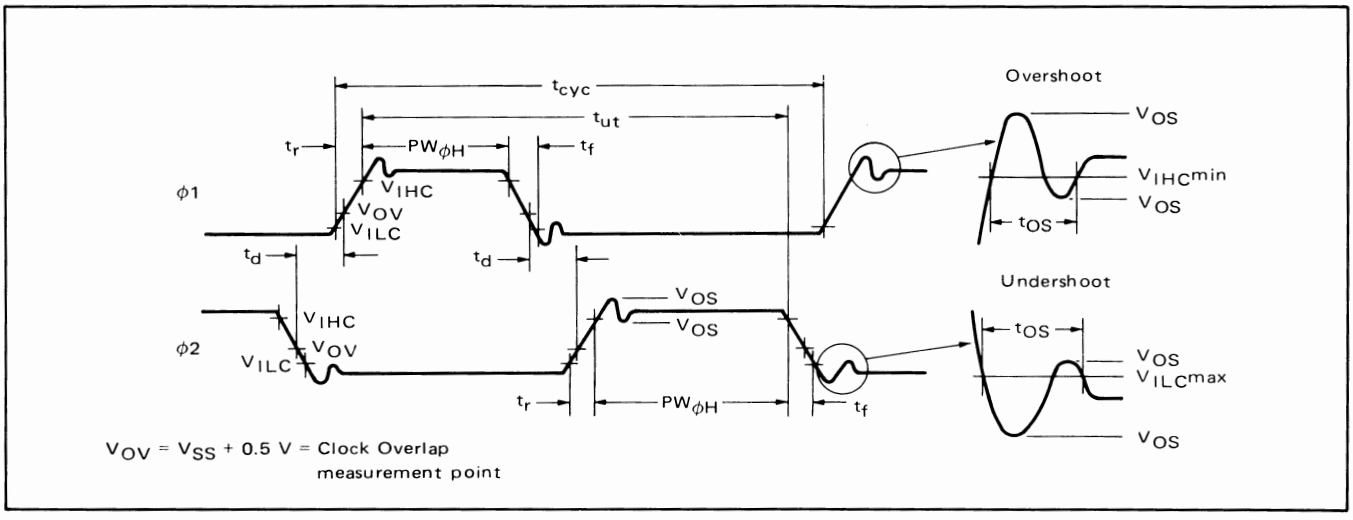


FIGURE 2 – READ DATA FROM MEMORY OR PERIPHERALS

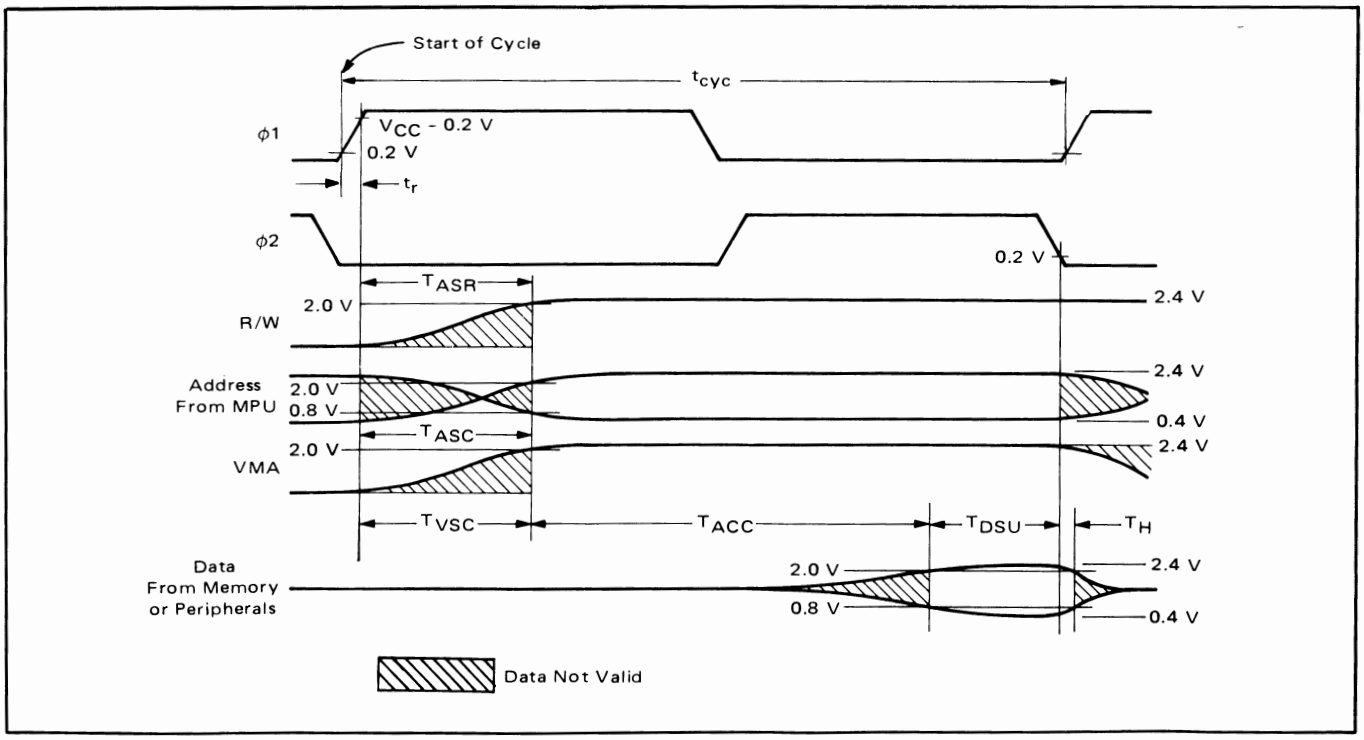
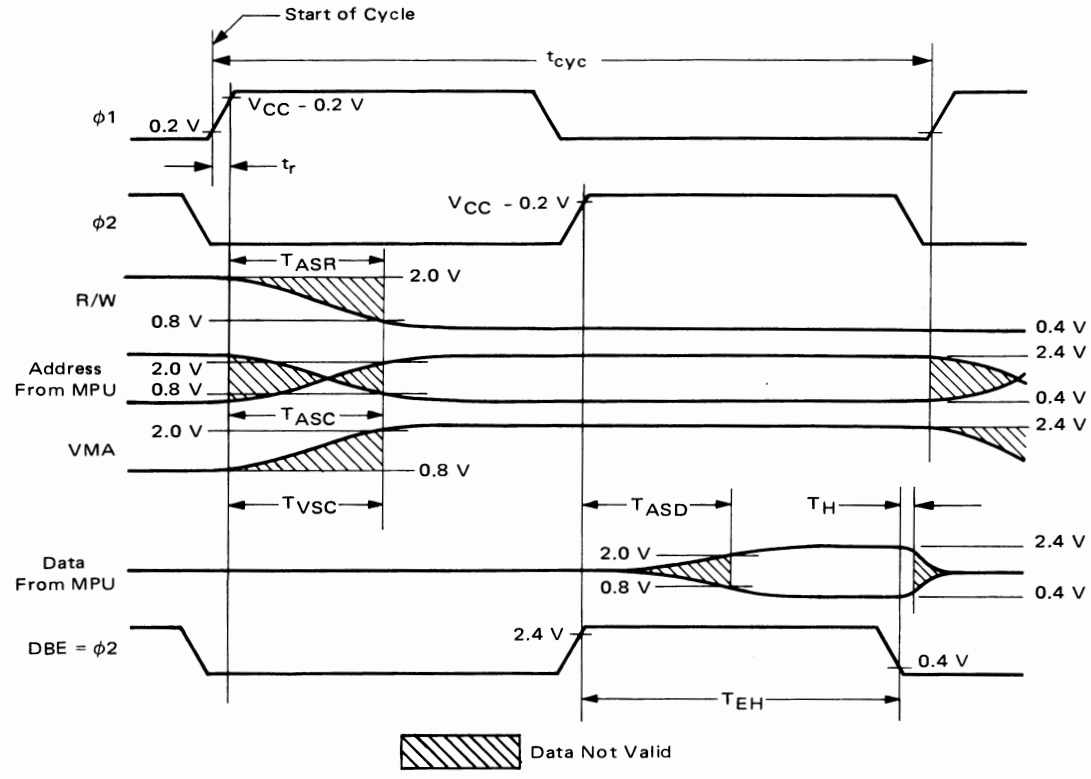
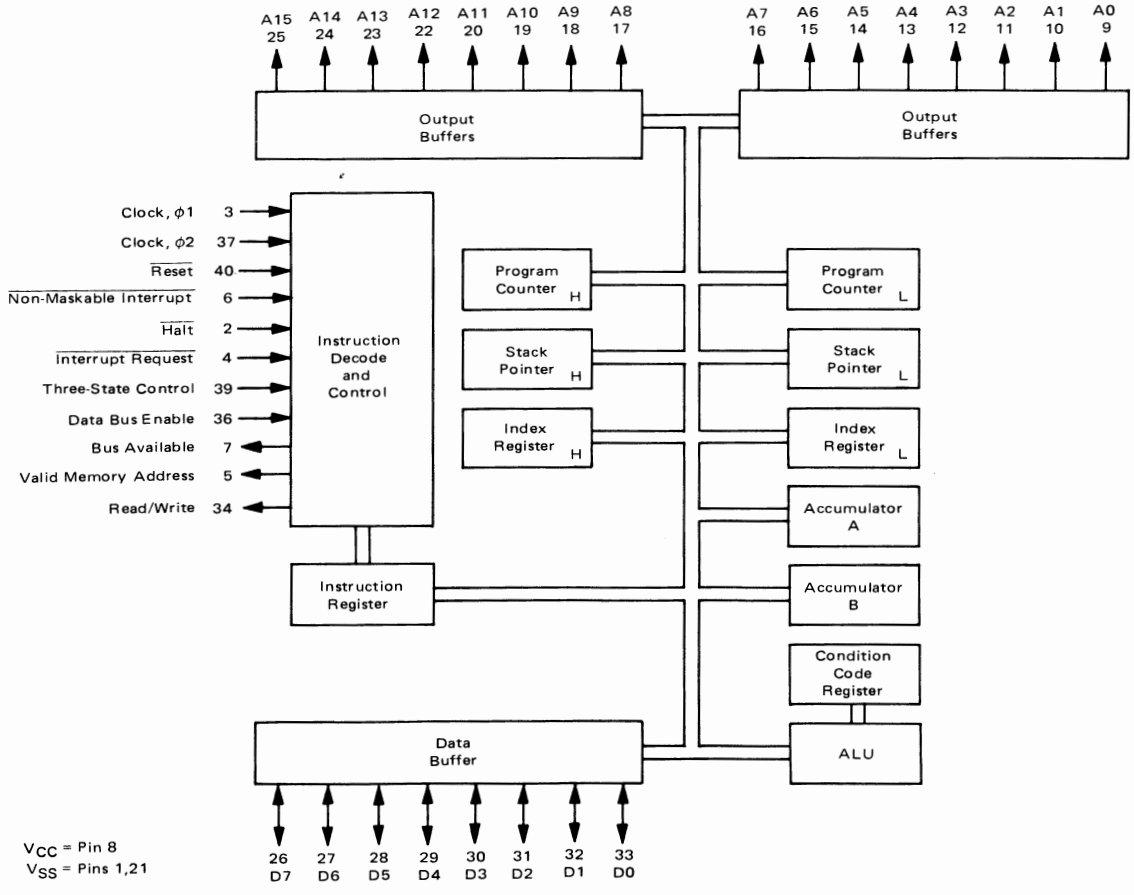


FIGURE 3 – WRITE DATA IN MEMORY OR PERIPHERALS





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 ($\phi 1, \phi 2$) — Two pins are used for a two-phase non-overlapping clock that runs at the V_{CC} voltage level.

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 130 pF. When the output is turned off, it is essentially an open circuit. This permits the MPU to be used in DMA applications.

Data Bus (D0-D7) — Eight pins are used for the data bus. It is bi-directional, 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 one level, Valid Memory Address will be at a zero, and all other three-state lines will be in the three-state mode.

Transition of the $\overline{\text{Halt}}$ line must not occur during the last 250 ns of phase one. To insure single instruction operation, the $\overline{\text{Halt}}$ line must go high for one Phase One Clock cycle.

Three-State Control (TSC) — This input causes all of the address lines and the Read/Write line to go into the off or high impedance state. This state will occur 500 ns after TSC = 2.4 V. The Valid Memory Address and Bus Available signals will be forced low. The data bus is not affected by TSC and has its own enable (Data Bus Enable). In DMA applications, the Three-State Control line should be brought high on the leading edge of the Phase One Clock. The $\phi 1$ clock must be held in the high state and the $\phi 2$ in the low state for this function to operate properly. The address bus will then be available for other devices to directly address memory. Since the MPU is a dynamic device, it can be held in this state for only 5.0 μs or destruction of data will occur in the MPU.

Read/Write (R/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/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 130 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 30 pF may be directly driven by this active high signal.

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.

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 $\overline{\text{Halt}}$ 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 recognized.

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. If a positive edge 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 $\overline{\text{IRQ}}$.

Figure 4 shows the initialization of the microprocessor after restart. $\overline{\text{Reset}}$ must be held low for at least eight clock periods after V_{CC} reaches 4.75 volts. If $\overline{\text{Reset}}$ goes high prior to the leading edge of $\phi 2$, on the next $\phi 1$ the first restart memory vector address (FFFE) will appear on the address lines. This location should contain the higher order eight bits to be stored into the program counter. Following, the next address FFFF should contain the lower order eight bits to be stored into the program counter.

Non-Maskable Interrupt ($\overline{\text{NMI}}$) – A low-going edge on this input requests that a non-mask-interrupt sequence be generated within the processor. As with the $\overline{\text{Interrupt Request}}$ 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 $\overline{\text{NMI}}$.

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 causes the MPU to branch to a non-maskable interrupt routine in memory.

$\overline{\text{NMI}}$ 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 $\overline{\text{IRQ}}$ and $\overline{\text{NMI}}$ are hardware interrupt lines that are sampled during $\phi 2$ and will start the interrupt routine on the $\phi 1$ following the completion of an instruction.

Figure 5 is a flow chart describing the major decision paths and interrupt vectors of the microprocessor. Table 1 gives the memory map for interrupt vectors.

FIGURE 4 – INITIALIZATION OF MPU AFTER RESTART

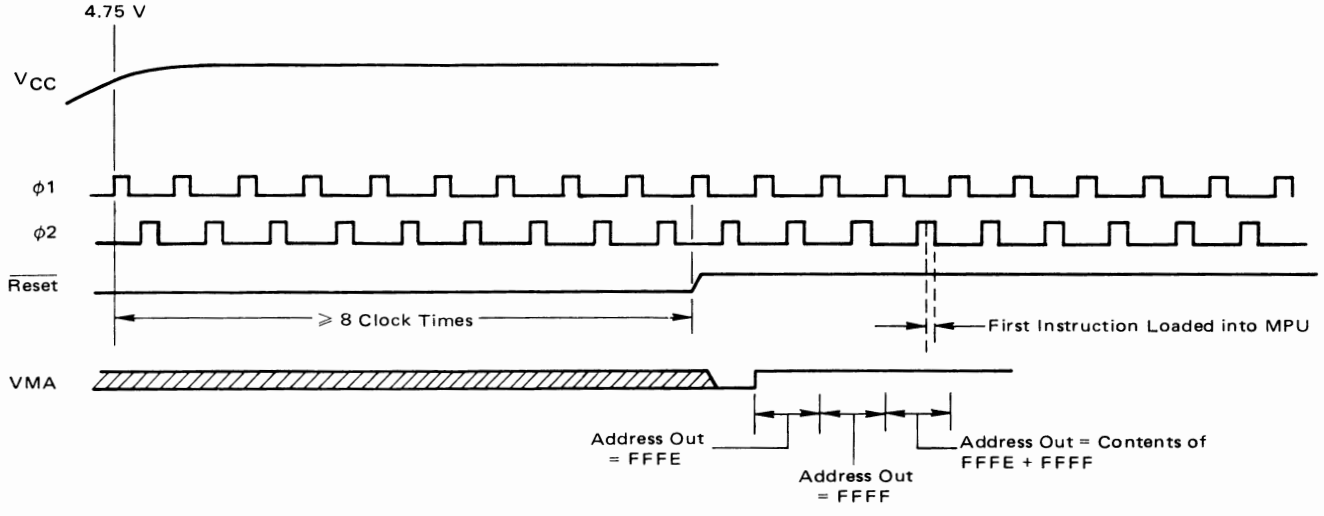
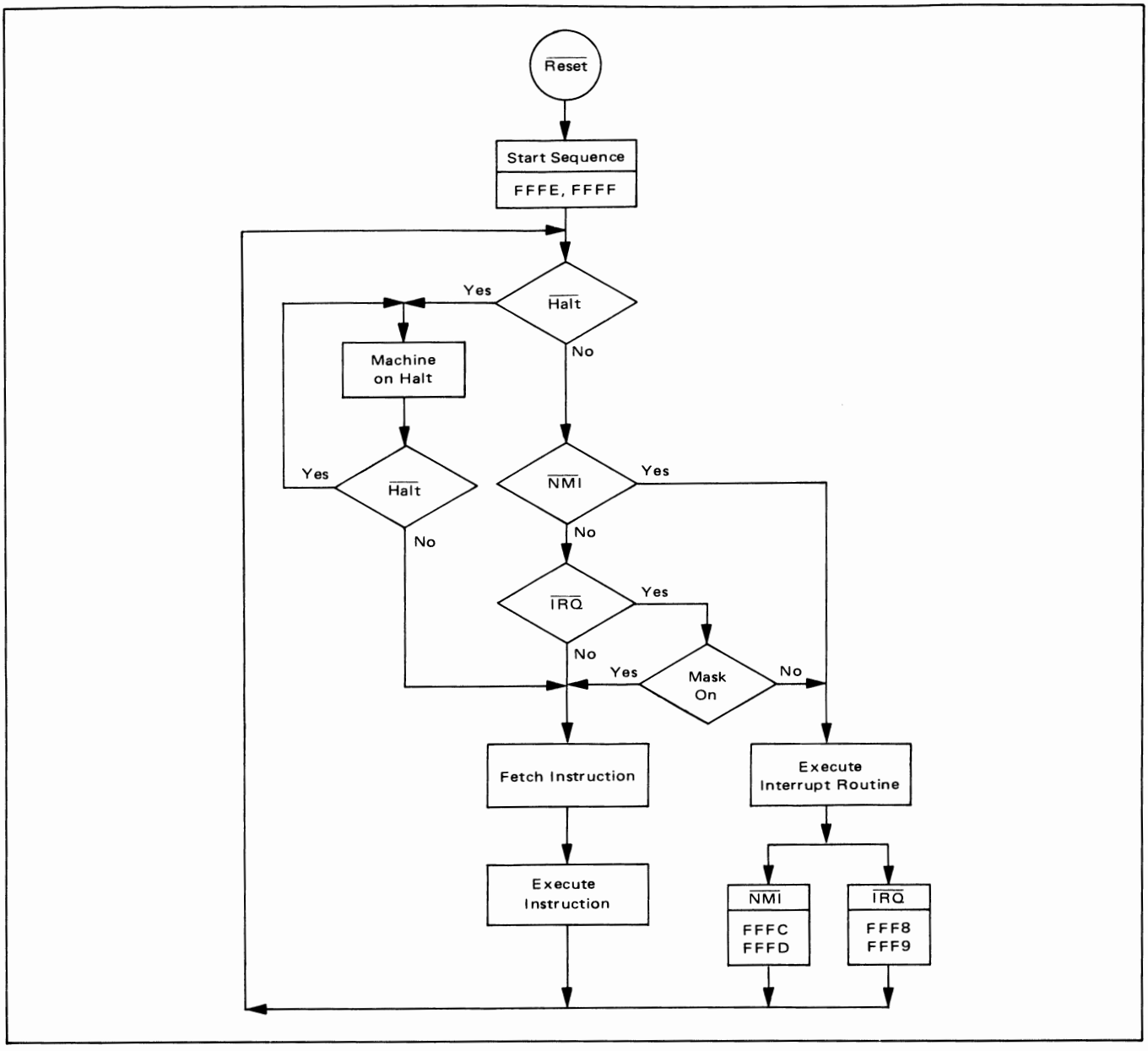


TABLE 1 – MEMORY MAP FOR INTERRUPT VECTORS

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

FIGURE 5 – MPU FLOW CHART



MPU REGISTERS

The MPU has three 16-bit registers and three 8-bit registers available for use by the programmer (Figure 6).

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).

FIGURE 6 – PROGRAMMING MODEL OF THE MICROPROCESSING UNIT

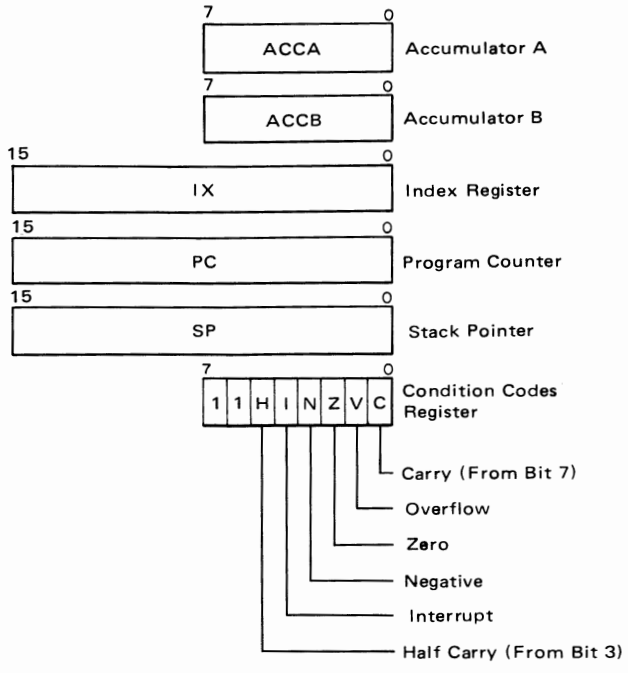
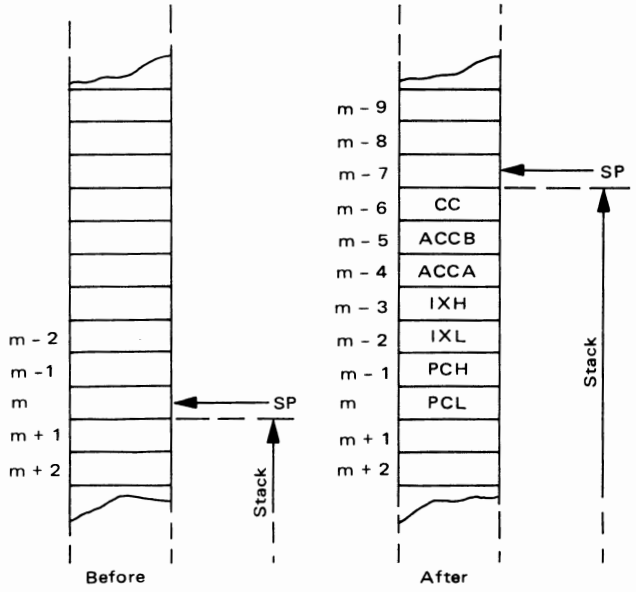


FIGURE 7 – SAVING THE STATUS OF THE MICROPROCESSOR IN THE STACK

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



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 8 shows the order of saving the microprocessor status within the stack.

MPU INSTRUCTION SET

The MC6800 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).

MPU ADDRESSING MODES

The MC6800 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.

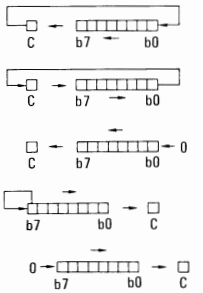
TABLE 2 – MICROPROCESSOR INSTRUCTION SET – ALPHABETIC SEQUENCE

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				$\overline{M}$ → M	•	•	↕	↕	R	S
	COMA													43	2	1	$\overline{A}$ → A	•	•	↕	↕	R	S
	COMB													53	2	1	$\overline{B}$ → 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	•	•	↕	↕	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

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;
$\overline{M}$ 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 4 – INDEX REGISTER AND STACK MANIPULATION INSTRUCTIONS

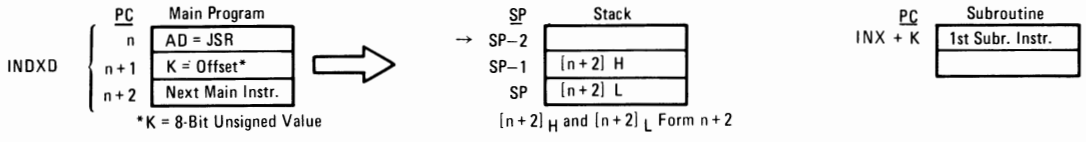
		IMMED			DIRECT			INDEX			EXTND			IMPLIED			BOOLEAN/ARITHMETIC OPERATION	COND. CODE REG.					
																		5	4	3	2	1	0
POINTER OPERATIONS	MNEMONIC	OP	~	#	OP	~	#	OP	~	#	OP	~	#	OP	~	#	BOOLEAN/ARITHMETIC OPERATION	H	I	N	Z	V	C
Compare Index Reg	CPX	8C	3	3	9C	4	2	AC	6	2	BC	5	3				$X_H - M, X_L - (M+1)$	•	•	⑦	↕	⑦	•
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	$SP + 1 \rightarrow X$	•	•	•	•	•	•

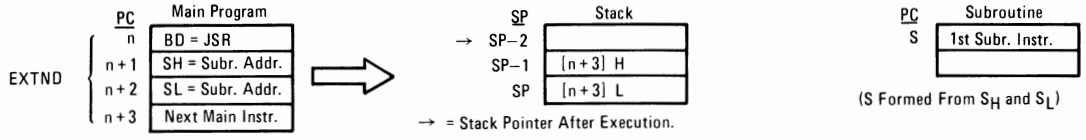
TABLE 5 – JUMP AND BRANCH INSTRUCTIONS

		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										02	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	•	•	•	•

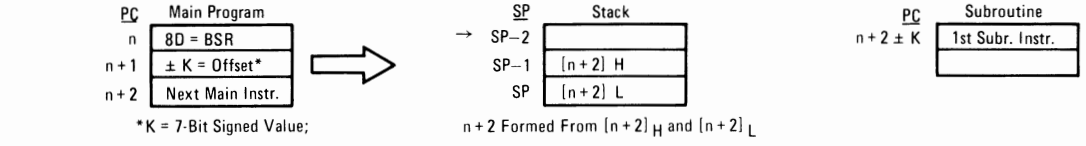
SPECIAL OPERATIONS

JSR, JUMP TO SUBROUTINE:



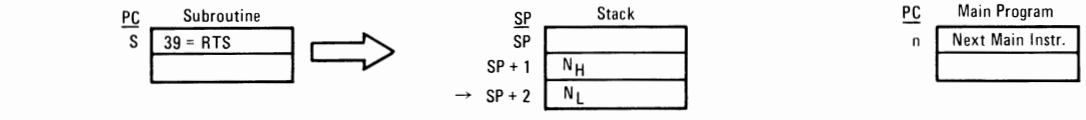


BSR, BRANCH TO SUBROUTINE:



JMP, JUMP:

RTS, RETURN FROM SUBROUTINE:



RTI, RETURN FROM INTERRUPT:

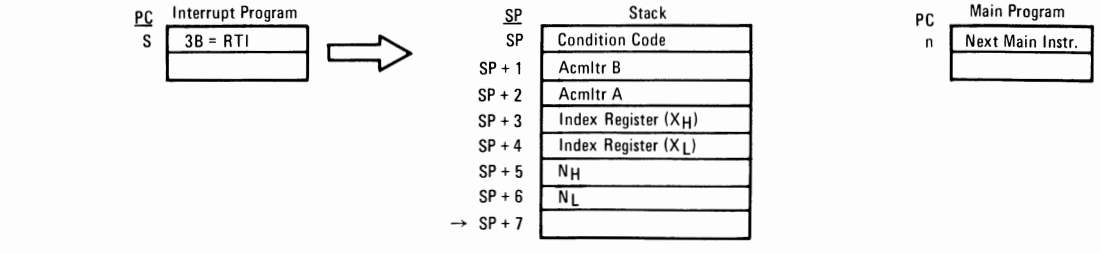


TABLE 6 – CONDITION CODE REGISTER MANIPULATION INSTRUCTIONS

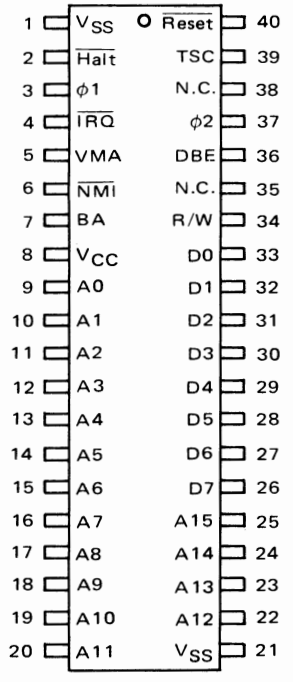
		IMPLIED				COND. CODE REG.					
						5	4	3	2	1	0
OPERATIONS	MNEMONIC	OP	~	#	BOOLEAN OPERATION	H	I	N	Z	V	C
Clear Carry	CLC	0C	2	1	0 → C	•	•	•	•	•	R
Clear Interrupt Mask	CLI	0E	2	1	0 → I	•	R	•	•	•	•
Clear Overflow	CLV	0A	2	1	0 → V	•	•	•	•	R	•
Set Carry	SEC	0D	2	1	1 → C	•	•	•	•	•	S
Set Interrupt Mask	SEI	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)

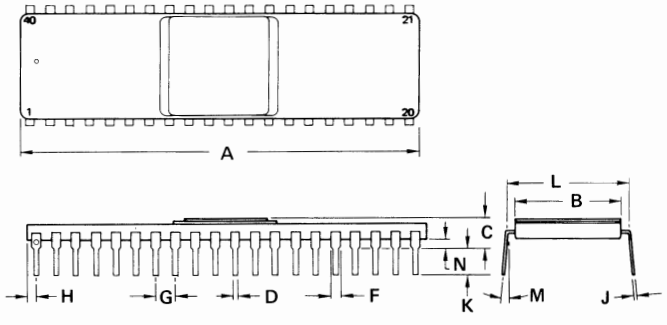
1 (Bit V) Test: Result = 10000000?
2 (Bit C) Test: Result = 00000000?
3 (Bit C) Test: Decimal value of most significant BCD Character greater than nine? (Not cleared if previously set.)
4 (Bit V) Test: Operand = 10000000 prior to execution?
5 (Bit V) Test: Operand = 01111111 prior to execution?
6 (Bit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

PIN ASSIGNMENT



PACKAGE DIMENSIONS

CASE 699-04



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	50.04	51.05	1.970	2.010
B	13.46	14.22	0.530	0.560
C	3.05	3.94	0.120	0.155
D	0.38	0.51	0.015	0.020
F	0.89	1.40	0.035	0.055
G	2.54 BSC		0.100 BSC	
H	0.89	1.40	0.035	0.055
J	0.20	0.28	0.008	0.011
K	3.05	3.68	0.120	0.145
L	14.86	15.87	0.585	0.625
M	–	15°	–	15°
N	0.51	1.14	0.020	0.045

NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE, AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO INSIDE OF LEADS (MEASURED 0.51 mm (0.020) BELOW PACKAGE BASE)

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.

MOTOROLA Semiconductors
BOX 20912, PHOENIX, ARIZONA 85036

MC6820

Advance Information

MOS

(N-CHANNEL, SILICON-GATE)

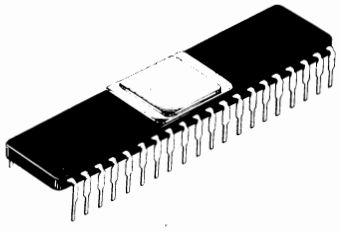
PERIPHERAL INTERFACE ADAPTER

PERIPHERAL INTERFACE ADAPTER (PIA)

The MC6820 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.

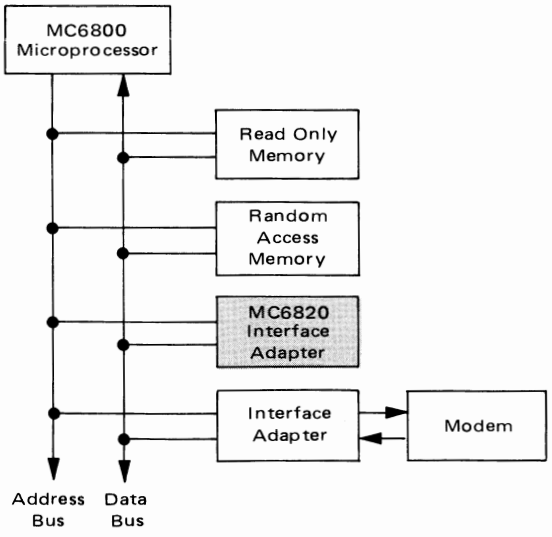
The functional configuration of the PIA is programmed by the MPU during system initialization. Each of the peripheral data lines can be programmed to act as an input or output, and each of the four control/interrupt lines may be programmed for one of several control modes. This allows a high degree of 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 Compatible Peripheral Lines

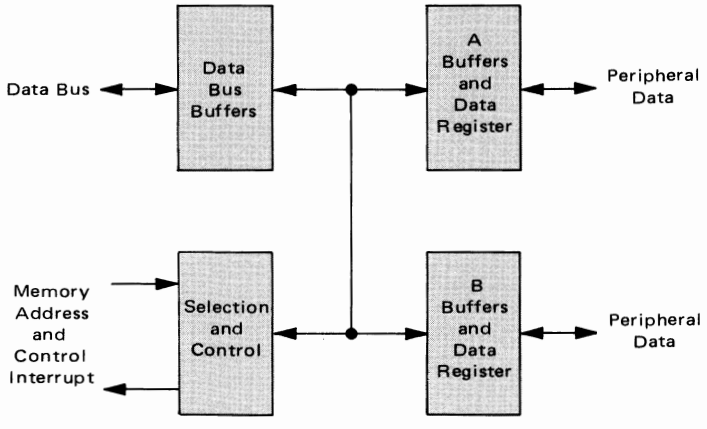


CERAMIC PACKAGE
CASE 699

M6800 MICROCOMPUTER FAMILY BLOCK DIAGRAM



MC6820 PERIPHERAL INTERFACE 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_A	0 to +70	°C
Storage Temperature Range	T_{stg}	-55 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.

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ± 5%, V_{SS} = 0, T_A = 25°C unless otherwise noted.)

Characteristic	Symbol	Min	Typ	Max	Unit
Input High Voltage (Normal Operating Levels)	V_{IH}	V_{SS} + 2.4	–	V_{CC}	Vdc
Input Low Voltage (Normal Operating Levels)	V_{IL}	V_{SS} - 0.3	–	V_{SS} + 0.4	Vdc
Input High Threshold Voltage All Inputs Except Enable	V_{IHT}	V_{SS} + 2.0	–	–	Vdc
Input Low Threshold Voltage All Inputs Except Enable	V_{ILT}	–	–	V_{SS} + 0.8	Vdc
Input Leakage Current (V_{in} = 0 to 5.0 Vdc) R/W, $\overline{\text{Reset}}$, RS0, RS1, CS0, CS1, $\overline{\text{CS2}}$, CA1, CB1, Enable	I_{in}	–	1.0	2.5	μAdc
Three-State (Off State) Input Current (V_{in} = 0.4 to 2.4 Vdc, V_{CC} = max) D0-D7, PB0-PB7, CB2	I_{TSI}	–	2.0	10	μAdc
Input High Current PA0-PA7, CA2 (V_{IH} = 2.4 Vdc)	I_{IH}	-100	-250	–	μAdc
Input Low Current PA0-PA7, CA2 (V_{IL} = 0.4 Vdc)	I_{IL}	–	-1.0	-1.6	mAdc
Output High Voltage (V_{CC} = min, I_{Load} = -100 μAdc, Enable Pulse Width < 25 μs)	V_{OH}	V_{SS} + 2.4	–	–	Vdc
Output Low Voltage (V_{CC} = min, I_{Load} = 1.6 mAdc)	V_{OL}	–	–	V_{SS} + 0.4	Vdc
Output High Current (Sourcing) (V_{OH} = 2.4 Vdc) (V_O = 1.5 Vdc, the current for driving other than TTL, e.g., Darlington Base) PB0-PB7, CB2	I_{OH}	-100 -1.0	-1000 -2.5	– –	μAdc mAdc
Output Low Current (Sinking) (V_{OL} = 0.4 Vdc)	I_{OL}	1.6	–	–	mAdc
Output Leakage Current (Off State) $\overline{\text{IRQA}}$, $\overline{\text{IRQB}}$	I_{off}	–	1.0	10	μAdc
Power Dissipation	P_D	–	300	600	mW
Input Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz) D0-D7, PA0-PA7, PB0-PB7, CA2, CB2 R/W, $\overline{\text{Reset}}$, RS0, RS1, CS0, CS1, $\overline{\text{CS2}}$, CA1, CB1 Enable	C_{in}	 – – –	 – – –	 10 7.0 20	pF
Output Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	C_{out}	–	–	10	pF

READ TIMING CHARACTERISTICS (Figure 1, Loading = 30 pF and one TTL load for PA0-PA7, PB0-PB7, CA2, CB2 = 130 pF and one TTL load for D0-D7, $\overline{\text{IRQA}}$, $\overline{\text{IRQB}}$)

Characteristic	Symbol	Min	Typ	Max	Unit
Delay Time, Address valid to Enable positive transition	T_{AEW}	180	–	–	ns
Delay Time, Enable positive transition to Data valid on bus	T_{EDR}	–	–	395	ns
Peripheral Data Setup Time	T_{PDSU}	300	–	–	ns
Data Bus Hold Time	T_{HR}	10	–	–	ns
Delay Time, Enable negative transition to CA2 negative transition	T_{CA2}	–	–	1.0	μs
Delay Time, Enable negative transition to CA2 positive transition	T_{RS1}	–	–	1.0	μs
Rise and Fall Time for CA1 and CA2 input signals	t_r, t_f	–	–	1.0	μs
Delay Time from CA1 active transition to CA2 positive transition	T_{RS2}	–	–	2.0	μs
Rise and Fall Time for Enable input	t_{rE}, t_{fE}	–	–	25	ns

WRITE TIMING CHARACTERISTICS (Figure 2)

Characteristic	Symbol	Min	Typ	Max	Unit
Enable Pulse Width	T_E	0.470	–	25	μs
Delay Time, Address valid to Enable positive transition	T_{AEW}	180	–	–	ns
Delay Time, Data valid to Enable negative transition	T_{DSU}	300	–	–	ns
Delay Time, Read/Write negative transition to Enable positive transition	T_{WE}	130	–	–	ns
Data Bus Hold Time	T_{HW}	10	–	–	ns
Delay Time, Enable negative transition to Peripheral Data valid	T_{PDW}	–	–	1.0	μs
Delay Time, Enable negative transition to Peripheral Data Valid, CMOS (V_{CC} – 30%) PA0-PA7, CA2	T_{CMOS}	–	–	2.0	μs
Delay Time, Enable positive transition to CB2 negative transition	T_{CB2}	–	–	1.0	μs
Delay Time, Peripheral Data valid to CB2 negative transition	T_{DC}	0	–	1.5	μs
Delay Time, Enable positive transition to CB2 positive transition	T_{RS1}	–	–	1.0	μs
Rise and Fall Time for CB1 and CB2 input signals	t_r, t_f	–	–	1.0	μs
Delay Time, CB1 active transition to CB2 positive transition	T_{RS2}	–	–	2.0	μs

Note: Negative sign indicates outward current flow, positive sign indicates inward flow.

FIGURE 1 – READ TIMING CHARACTERISTICS

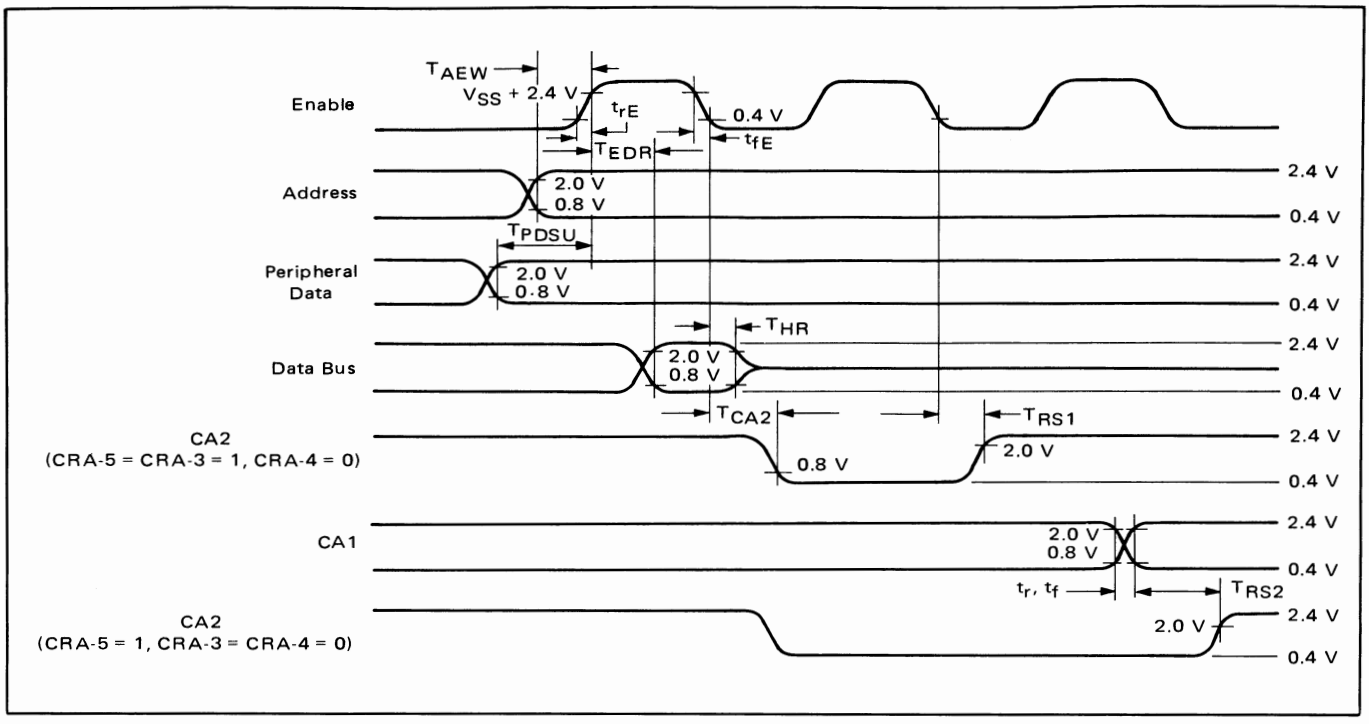
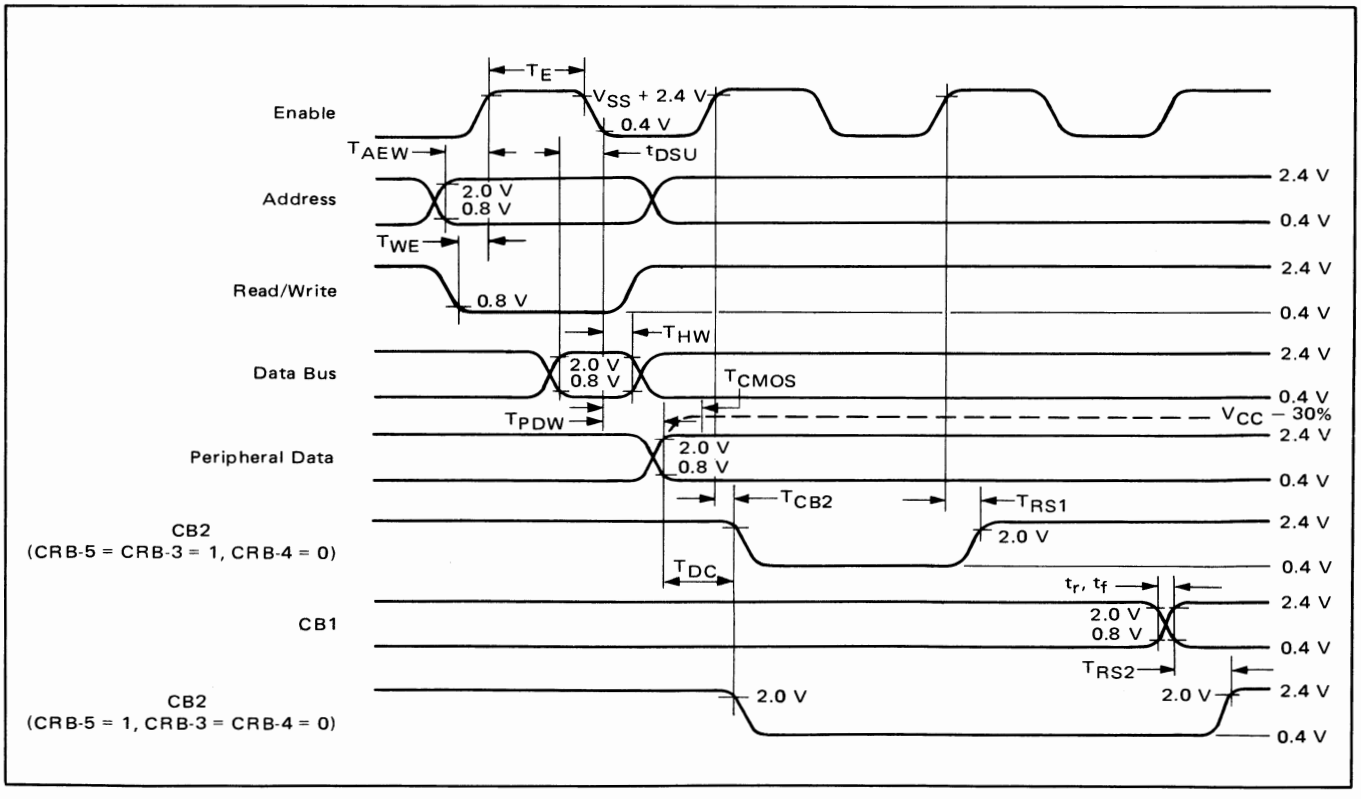
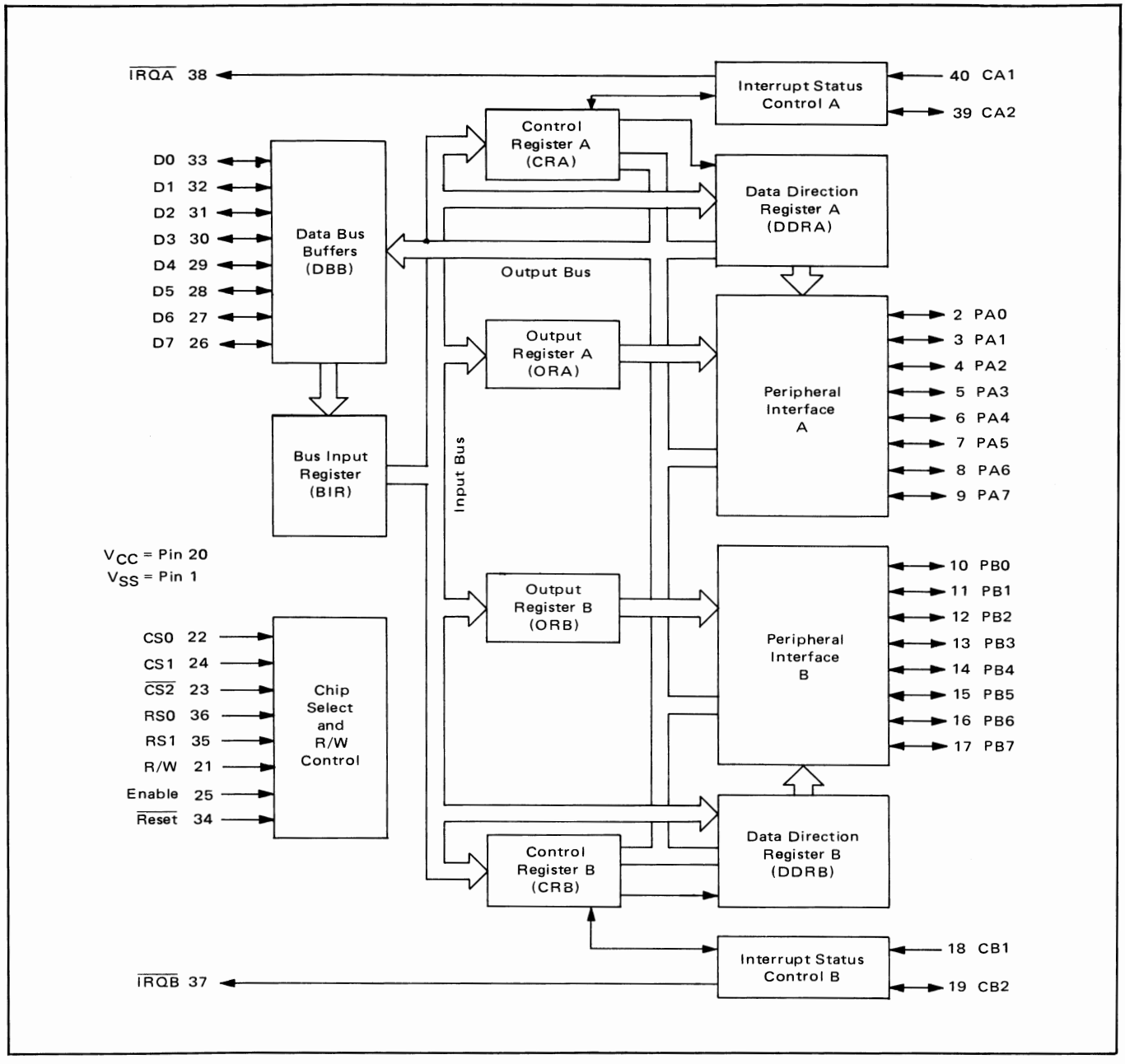


FIGURE 2 – WRITE TIMING CHARACTERISTICS



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 may be utilized to gate the input signals to 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.

The E pulse is used to condition the interrupt/control lines CA1, CA2, CB1, and CB2. At least one E pulse must occur from the inactive edge to the active edge of the input signal to set the interrupt flag, when the lines are used as inputs.

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.

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 select lines should be stable for the duration of the E pulse while in the read or write cycle.

Interrupt Request ($\overline{\text{IRQA}}$ and $\overline{\text{IRQB}}$) – The active low 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 source" (no load device on the chip) and are capable of sinking a current of 1.6 mA from an external source. This permits all interrupt request lines to be tied together in a wire-OR configuration.

Each Interrupt Request line has two internal interrupt flag bits that will cause the 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 Flag is cleared (zeroed) as a result of an MPU Read Peripheral Data Operation.

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 one standard TTL load.

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 one standard TTL load. 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 Status 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 IRQA (IRQB)
0	0	↓ Active	Set high on ↓ of CA1 (CB1)	Disabled — 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 — 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, IRQA (IRQB) occurs on the positive transition of CRA-0 (CRB-0).

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 IRQA and 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 IRQA (IRQB)
0	0	0	↓ Active	Set high on ↓ of CA2 (CB2)	Disabled — 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 — 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, IRQA (IRQB) occurs on the positive transition of CRA-3 (CRB-3).

TABLE 5 – CONTROL OF CB2 AS AN OUTPUT
CRB-5 is high

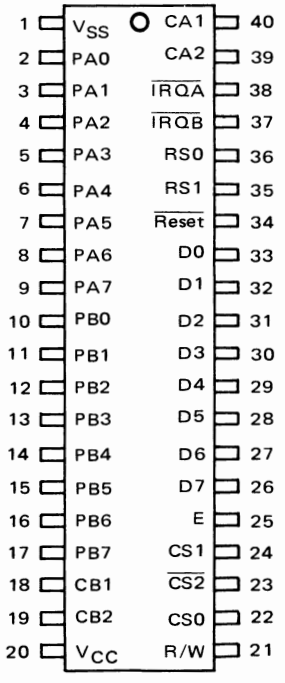
CRB-5	CRB-4	CRB-3	CB2 Cleared	CB2 Set
1	0	0	Low on the positive transition of the first E pulse following an MPU Write "B" Data Register operation.	High when the interrupt flag bit CRB-7 is set by an active transition of the CB1 signal.
1	0	1	Low on the positive transition of the first E pulse following an MPU Write "B" Data Register operation.	High on the positive transition of the next "E" pulse.
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 CA2 AS AN OUTPUT
CRA-5 is high

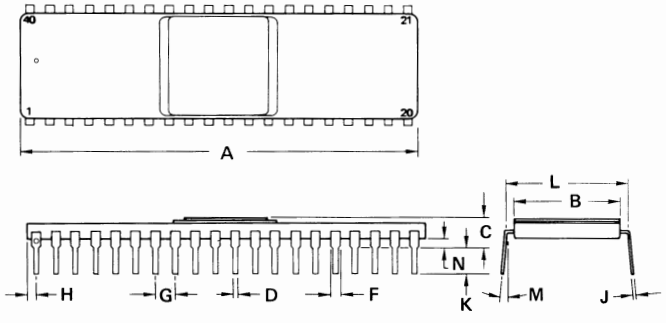
CRA-5	CRA-4	CRA-3	CA2 Cleared	CA2 Set
1	0	0	Low on negative transition of E after an MPU Read "A" Data operation.	High on an active transition of the CA1 signal.
1	0	1	Low immediately after an MPU Read "A" Data operation.	High on the negative edge of the next "E" pulse.
1	1	0	Low when CRA-3 goes low as a result of an MPU Write in Control Register "A".	Always low as long as CRA-3 is low.
1	1	1	Always high as long as CRA-3 is high.	High when CRA-3 goes high as a result of a Write in Control Register "A".

PIN ASSIGNMENT



PACKAGE DIMENSIONS

CASE 699-04



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	50.04	51.05	1.970	2.010
B	13.46	14.22	0.530	0.560
C	3.05	3.94	0.120	0.155
D	0.38	0.51	0.015	0.020
F	0.89	1.40	0.035	0.055
G	2.54 BSC		0.100 BSC	
H	0.89	1.40	0.035	0.055
J	0.20	0.28	0.008	0.011
K	3.05	3.68	0.120	0.145
L	14.86	15.87	0.585	0.625
M	–	15^0	–	15^0
N	0.51	1.14	0.020	0.045

NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE, AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION "L" TO INSIDE OF LEADS (MEASURED 0.51 mm (0.020) BELOW PACKAGE BASE)

MOTOROLA Semiconductors
BOX 20912 • PHOENIX, ARIZONA 85036

MC6850

Advance Information

MOS

(N-CHANNEL, SILICON-GATE)

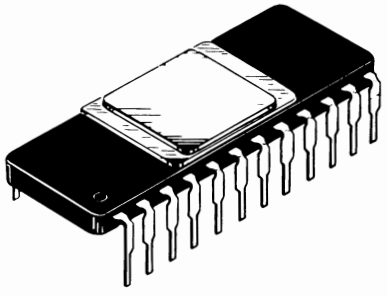
ASYNCHRONOUS COMMUNICATIONS INTERFACE ADAPTER

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.

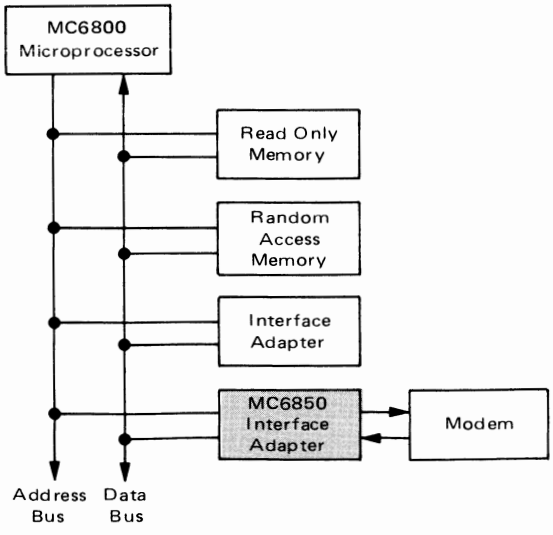
The bus interface of the MC6850 includes select, enable, read/write, interrupt and bus interface logic to allow data transfer over an 8-bit bi-directional data bus. The parallel data of the bus system is serially transmitted and received by the asynchronous data interface, with proper formatting and 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

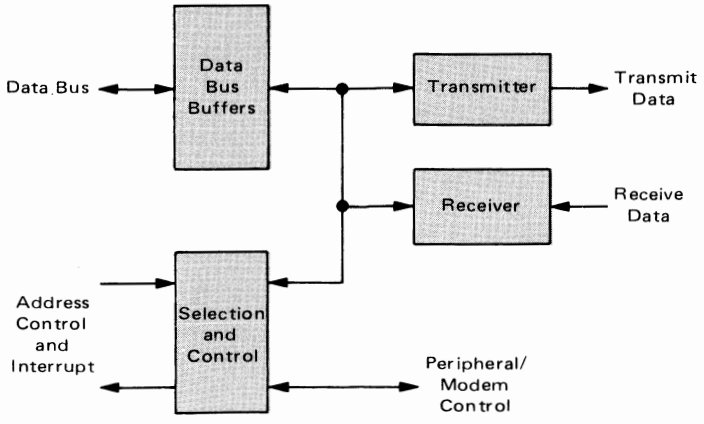


CERAMIC PACKAGE
CASE 684

M6800 MICROCOMPUTER FAMILY BLOCK DIAGRAM



MC6850 ASYNCHRONOUS COMMUNICATIONS INTERFACE 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_A	0 to +70	°C
Storage Temperature Range	T_{stg}	-55 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.

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V ± 5%, all voltages referenced to V_{SS} = 0, T_A = 25°C unless otherwise noted.)

Characteristic	Symbol	Min	Typ	Max	Unit
Input High Voltage (Normal Operating Levels)	V_{IH}	2.4	–	V_{CC}	Vdc
Input Low Voltage (Normal Operating Levels)	V_{IL}	-0.3	–	0.4	Vdc
Input High Threshold Voltage — All Inputs Except Enable	V_{IHT}	2.0	–	–	Vdc
Input Low Threshold Voltage — All Inputs Except Enable	V_{ILT}	–	–	0.8	Vdc
Input Leakage Current (V_{in} = 0 to 5.25 Vdc) — R/W, CS0, CS1, $\overline{CS2}$, Enable	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 (Load A, I_{Load} = -100 μAdc, Enable Pulse Width < 25 μs) — All Outputs Except $\overline{IRQ}$	V_{OH}	2.4	–	–	Vdc
Output Low Voltage (Load A, I_{Load} = 1.6 mAdc, except $\overline{IRQ}$ = Load B, Enable Pulse Width < 25 μs)	V_{OL}	–	–	0.4	Vdc
Output Leakage Current (Off State) — $\overline{IRQ}$	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)	C_{in}				pF
D0-D7		–	–	10	
Tx Clk, Rx Clk, R/W, RS, Rx Data, CS0, CS1, $\overline{CS2}$, $\overline{CTS}$, $\overline{DCD}$		–	–	7.0	
Enable		–	–	10	
Output Capacitance (V_{in} = 0, T_A = 25°C, f = 1.0 MHz)	C_{out}	–	–	10	pF
Minimum Clock Pulse Width, Low (Figure 1)	PW_{CL}	600	–	–	ns
Minimum Clock Pulse Width, High (Figure 2)	PW_{CH}	600	–	–	ns
Clock Frequency	f_C				kHz
÷ 1 Mode		–	–	500	
÷ 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_{IRQR}	–	–	1.2	μs
$\overline{Request\text{-}to\text{-}Send}$ Delay Time (Figure 6)	T_{RTS}	–	–	1.0	μs

BUS TIMING CHARACTERISTICS

READ (Figures 7 and 8)

Characteristic	Symbol	Min	Typ	Max	Unit
Enable Pulse Width	T_E	0.470	–	25	μs
Setup Time, Address valid to Enable positive transition	T_{AEW}	180	–	–	ns
Setup Time, Enable positive transition to Data valid on bus	T_{EDR}	–	–	395	ns
Data Bus Hold Time	T_{HR}	10	–	–	ns
Rise and Fall Time for Enable input	t_{rE}, t_{fE}	–	–	25	ns

WRITE (Figures 7 and 9)

Enable Pulse Width	T_E	0.470	–	25	μs
Setup Time, Address valid to Enable positive transition	T_{AEW}	180	–	–	ns
Setup Time, Data valid to Enable negative transition	T_{DSU}	300	–	–	ns
Setup Time, Read/Write negative transition to Enable positive transition	T_{WE}	130	–	–	ns
Data Bus Hold Time	T_{HW}	10	–	–	ns
Rise and Fall Time for Enable input	t_{rE}, t_{fE}	–	–	25	ns

FIGURE 1 – CLOCK PULSE WIDTH, LOW-STATE

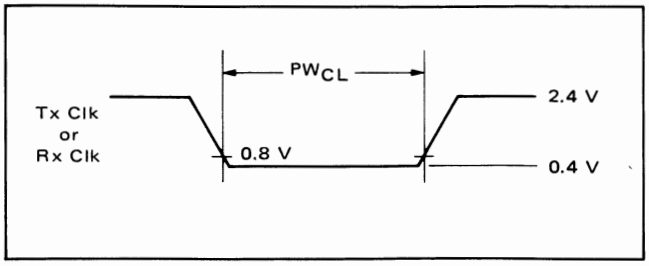


FIGURE 2 – CLOCK PULSE WIDTH, HIGH-STATE

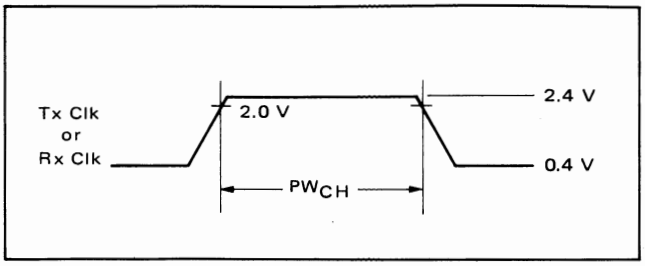


FIGURE 3 – TRANSMIT DATA OUTPUT DELAY

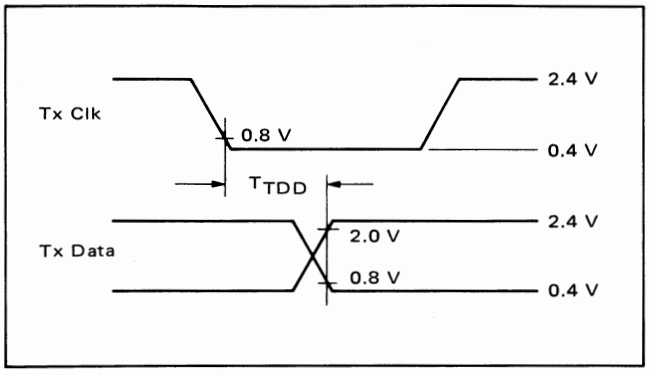


FIGURE 4 – RECEIVE DATA SETUP TIME
(÷1 Mode)

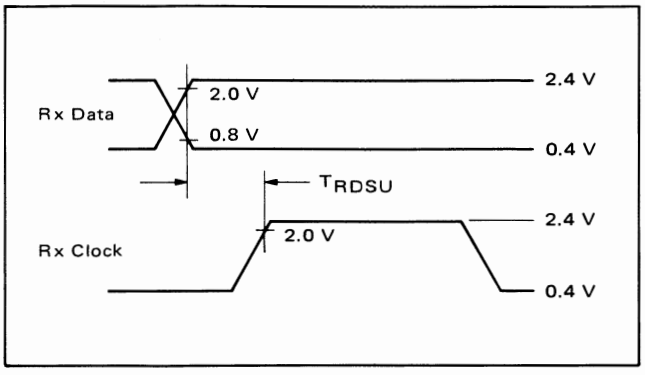


FIGURE 5 – RECEIVE DATA HOLD TIME
(÷1 Mode)

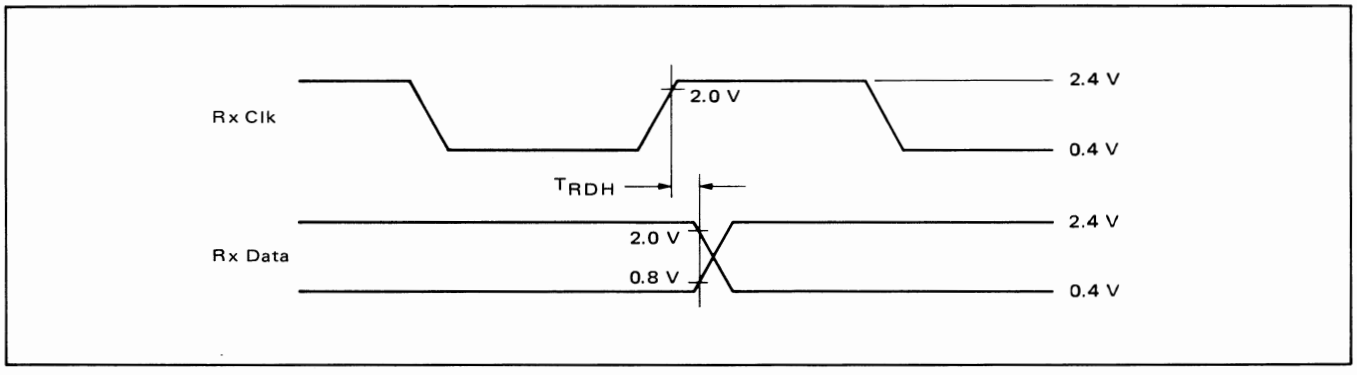


FIGURE 6 – $\overline{\text{REQUEST-TO-SEND}}$ DELAY AND $\overline{\text{INTERRUPT-REQUEST}}$ RELEASE TIMES

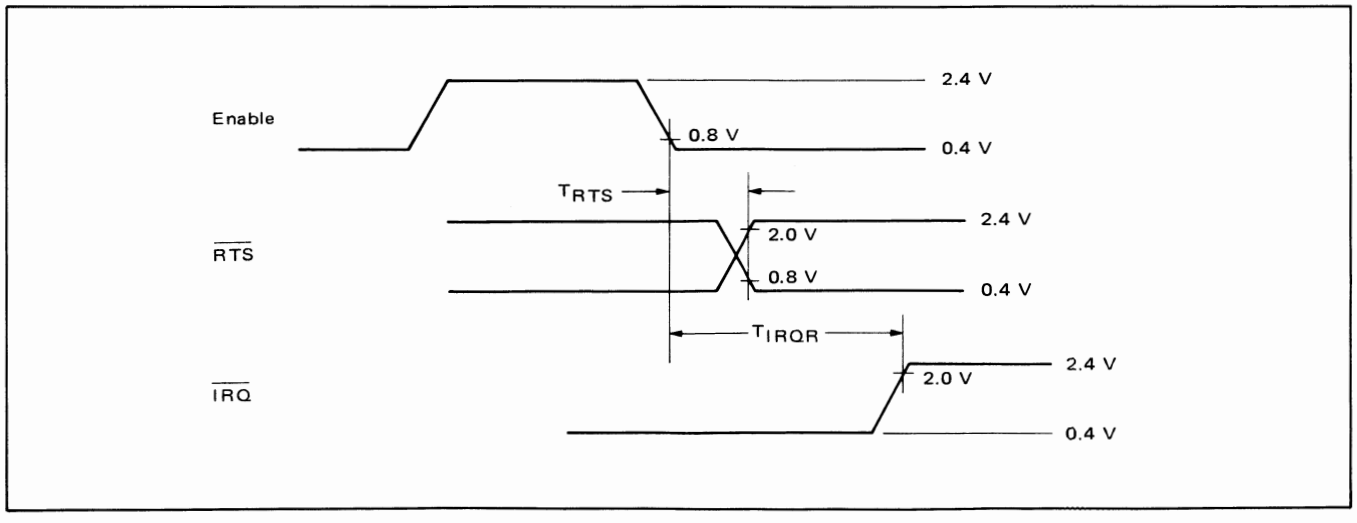


FIGURE 7 – BUS TIMING TEST LOADS

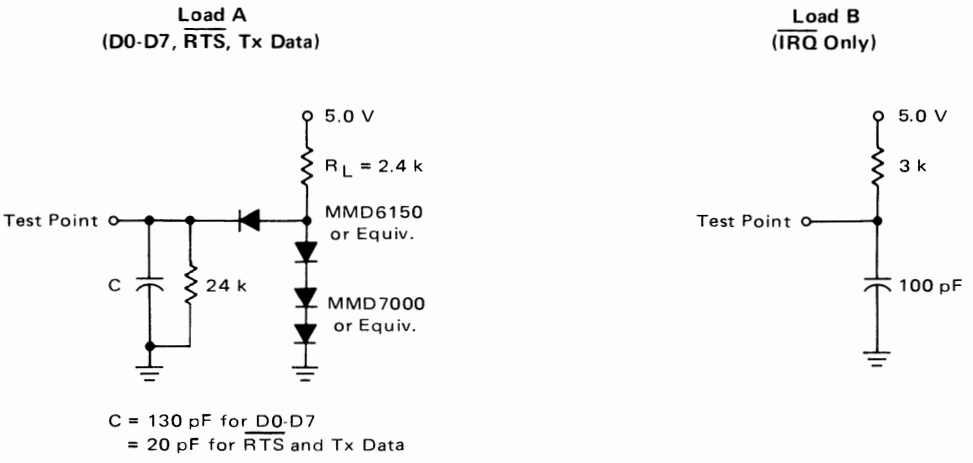


FIGURE 8 – BUS READ TIMING CHARACTERISTICS
(Read information from ACIA)

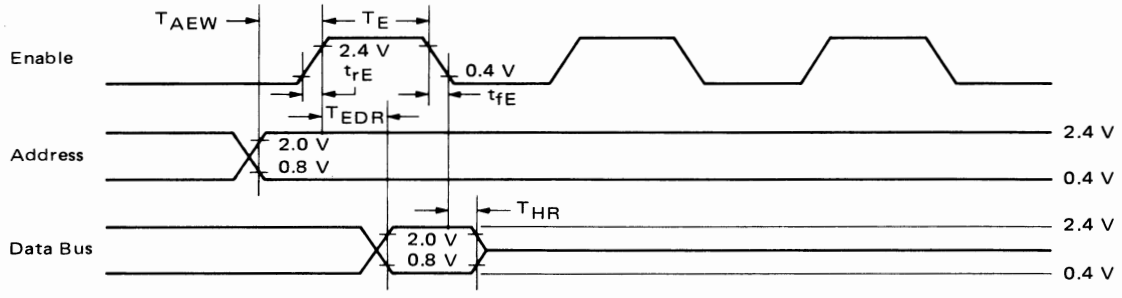
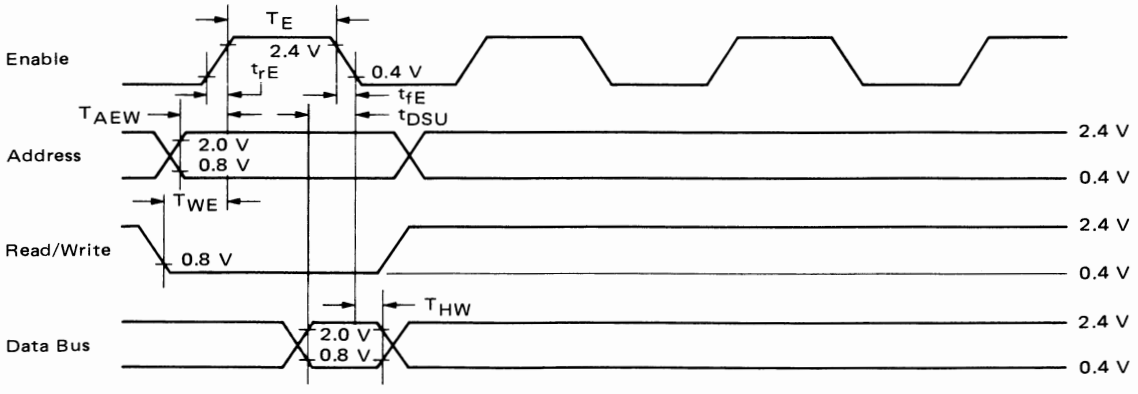
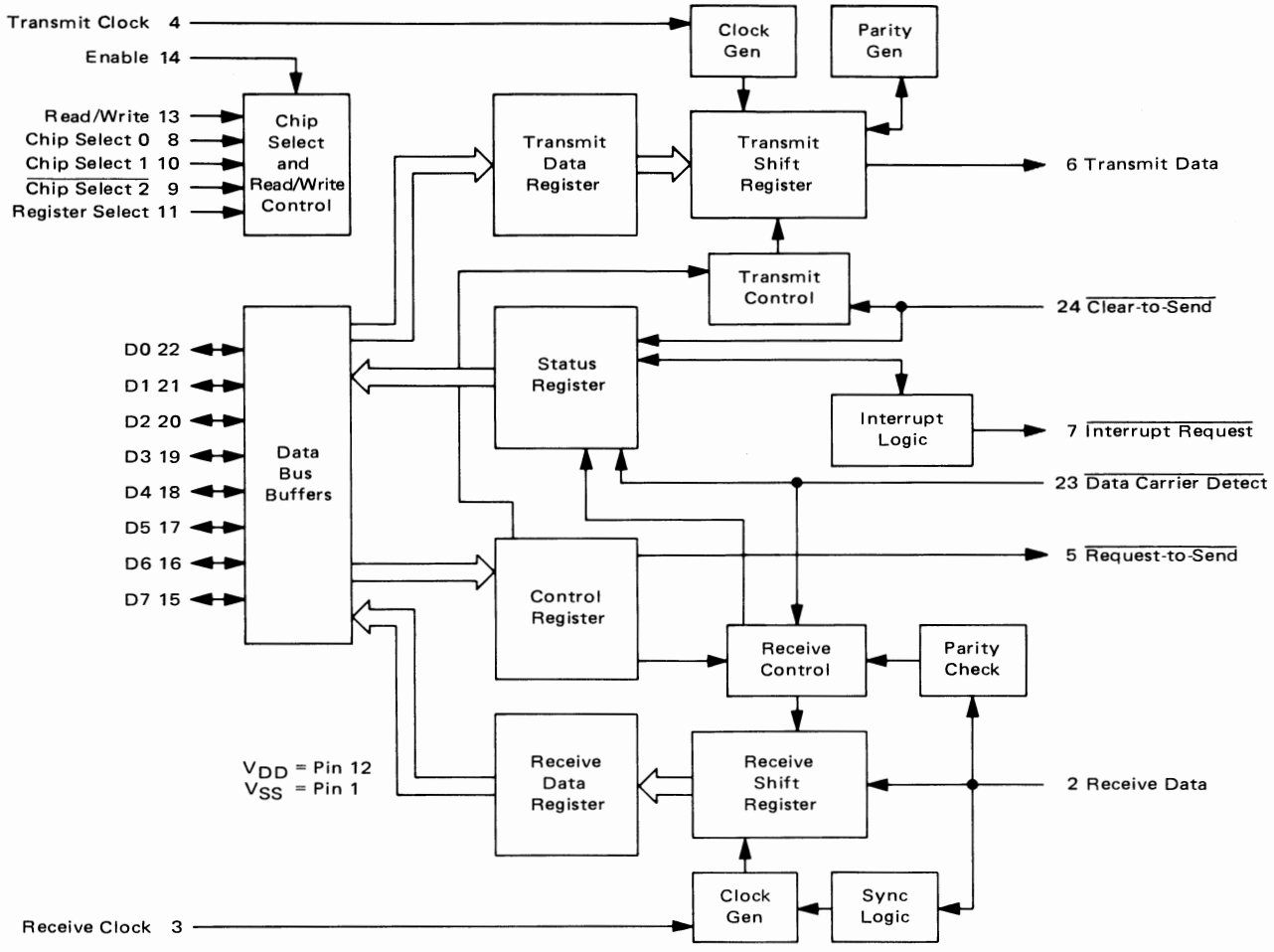


FIGURE 9 – BUS WRITE TIMING CHARACTERISTICS
(Write information into ACIA)



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 $\overline{\text{Interrupt Request}}$ remains low as long as the cause of the interrupt is present and the appropriate interrupt enable within the ACIA is set.

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 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

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)	Data Carrier Detect ($\overline{DCD}$)
3	Data Bit 3	Data Bit 3	Word Select 2 (CR3)	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.

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 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 Request-to-Send 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) – Interrupts will be enabled by a high level in bit position 7 of the Control Register (CR7). Interrupts from the receiver section, Receive Data Register Full being high or by a low to high transistion on the Data Carrier Detect signal line, are enabled or disabled by the Receive Interrupt Enable Bit.

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. 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.

Data Carrier Detect ($\overline{DCD}$), Bit 2 – The Data Carrier Detect bit will be high when the $\overline{DCD}$ 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 $\overline{DCD}$ 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 $\overline{DCD}$ input remains high after read status and read data or master reset have occurred, the $\overline{DCD}$ status bit remains high and will follow the $\overline{DCD}$ input.

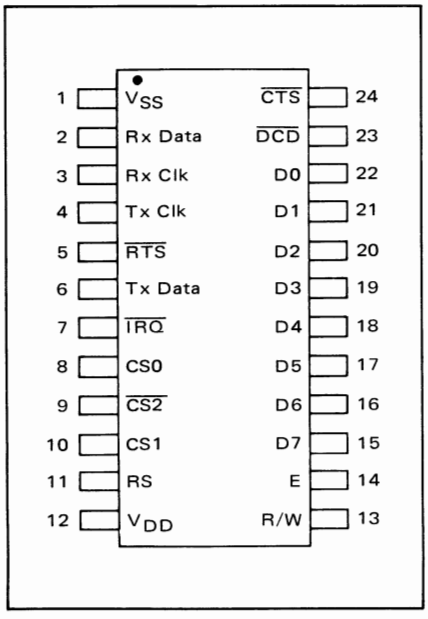
Clear-to-Send ($\overline{CTS}$), Bit 3 – The Clear-to-Send bit indicates the state of the Clear-to-Send input from a modem. A low $\overline{CTS}$ indicates that there is a Clear-to-Send from the modem. In the high state, the Transmit Data Register Empty bit is inhibited and the Clear-to-Send status bit will be high. Master reset does not affect the Clear-to-Send Status bit.

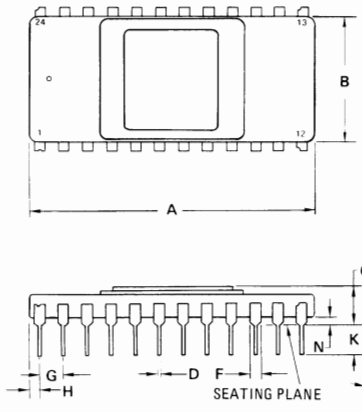
Framing Error (FE), Bit 4 – Framing error indicates that the received character is improperly framed by a start and a stop bit and is detected by the absence of the 1st stop bit. This error indicates a synchronization error, faulty transmission, or a break condition. The framing error flag is set or 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. Overrun is also reset by the 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 $\overline{IRQ}$ output. Any interrupt condition with its applicable enable will be indicated in this status bit. Anytime the $\overline{IRQ}$ output is low the IRQ bit will be high to indicate the interrupt or service request status.





DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	29.34	30.86	1.155	1.215
B	12.70	14.22	0.500	0.560
C	3.05	3.94	0.120	0.155
D	0.38	0.51	0.015	0.020
F	0.89	1.40	0.035	0.055
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	2.92	3.68	0.115	0.145
L	14.86	15.87	0.585	0.625
M	–	15°	–	15°
N	0.51	1.14	0.020	0.045

NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE WITH MAXIMUM MATERIAL CONDITION.
2. LEAD NO. 1 CUT FOR IDENTIFICATION, OR BUMP ON TOP.
3. DIM "L" TO INSIDE OF LEADS. (MEASURED 0.51 mm (0.020) BELOW PKG BASE)

MOTOROLA Semiconductors
BOX 20912 • PHOENIX, ARIZONA 85036

MC6860L

Advance Information

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.

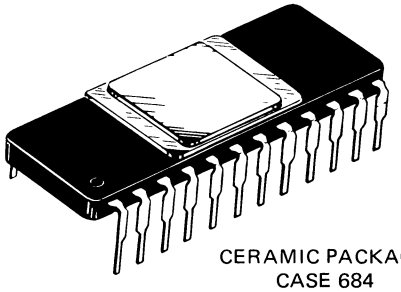
The modem is compatible with the M6800 microcomputer family, interfacing directly with the Asynchronous Communications Interface Adapter to provide low-speed data communications capability.

- 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

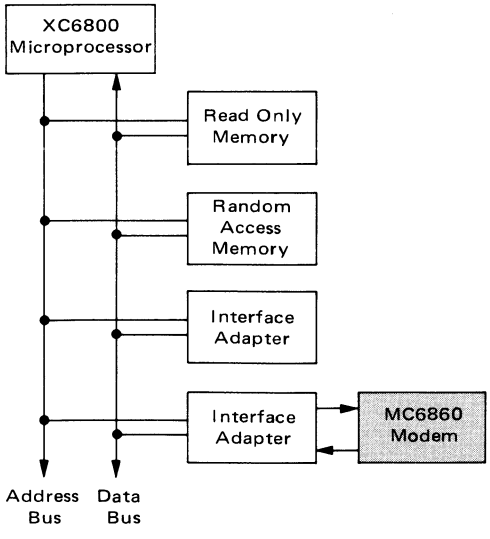
(N-CHANNEL, SILICON-GATE)

0-600 bps
DIGITAL MODEM

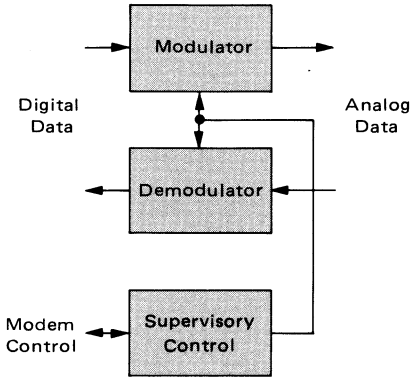


CERAMIC PACKAGE
CASE 684

M6800 MICROCOMPUTER FAMILY
BLOCK DIAGRAM



MC6860 DIGITAL MODEM
BLOCK DIAGRAM



MAXIMUM RATINGS (Voltages referenced to V_{SS}, Pin 1)

Rating	Symbol	Value	Unit
Supply Voltage	V_{DD}	-0.3 to +7.0	Vdc
Data 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

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.

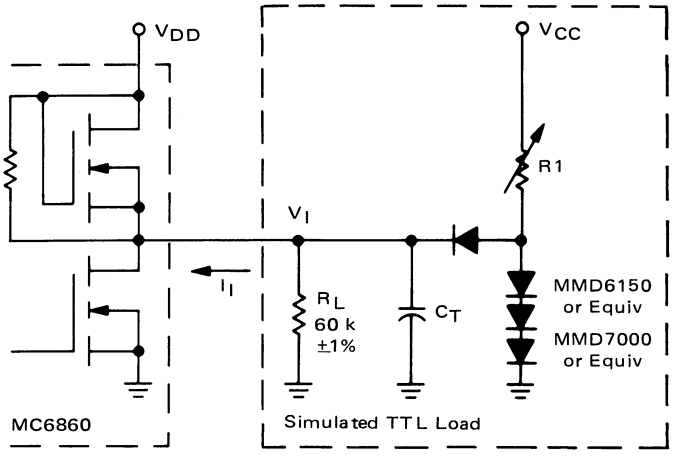
ELECTRICAL CHARACTERISTICS (V_{DD} = 5.0 ±0.25 Vdc, all voltages referenced to V_{SS} = 0, T_A = 0 to 70°C, all outputs loaded as shown in Figure 1 unless otherwise noted.)

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

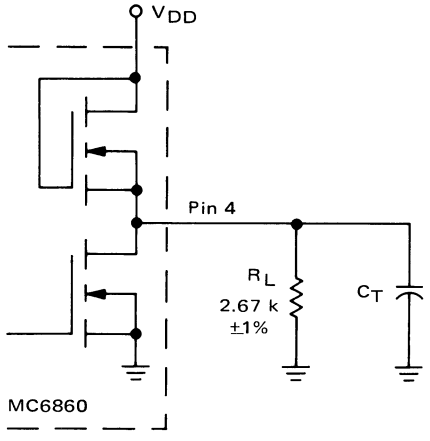
Load A – TTL Output Load for Receive Break, Digital Carrier, Mode, Clear-to-Send, and Receive Data Outputs



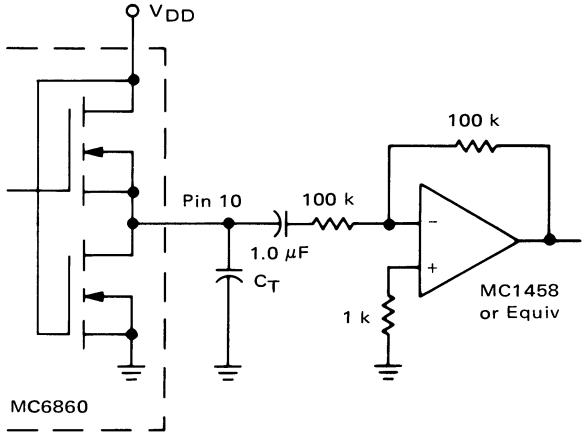
C_T = 20 pF = total parasitic capacitance, which includes probe, wiring, and load capacitances

R1 is adjusted for I_I = 1.6 mA at V_I = 0.4 V when output node is disconnected.

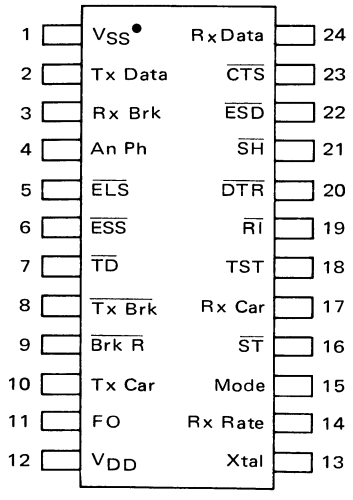
Load B – Answer Phone Load



Load C – Transmit Carrier Load

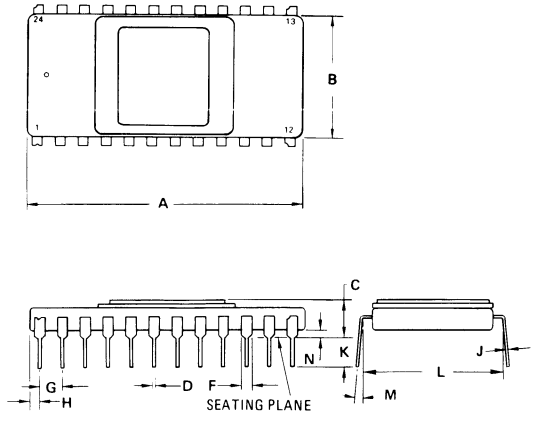


PIN ASSIGNMENT



PACKAGE DIMENSIONS

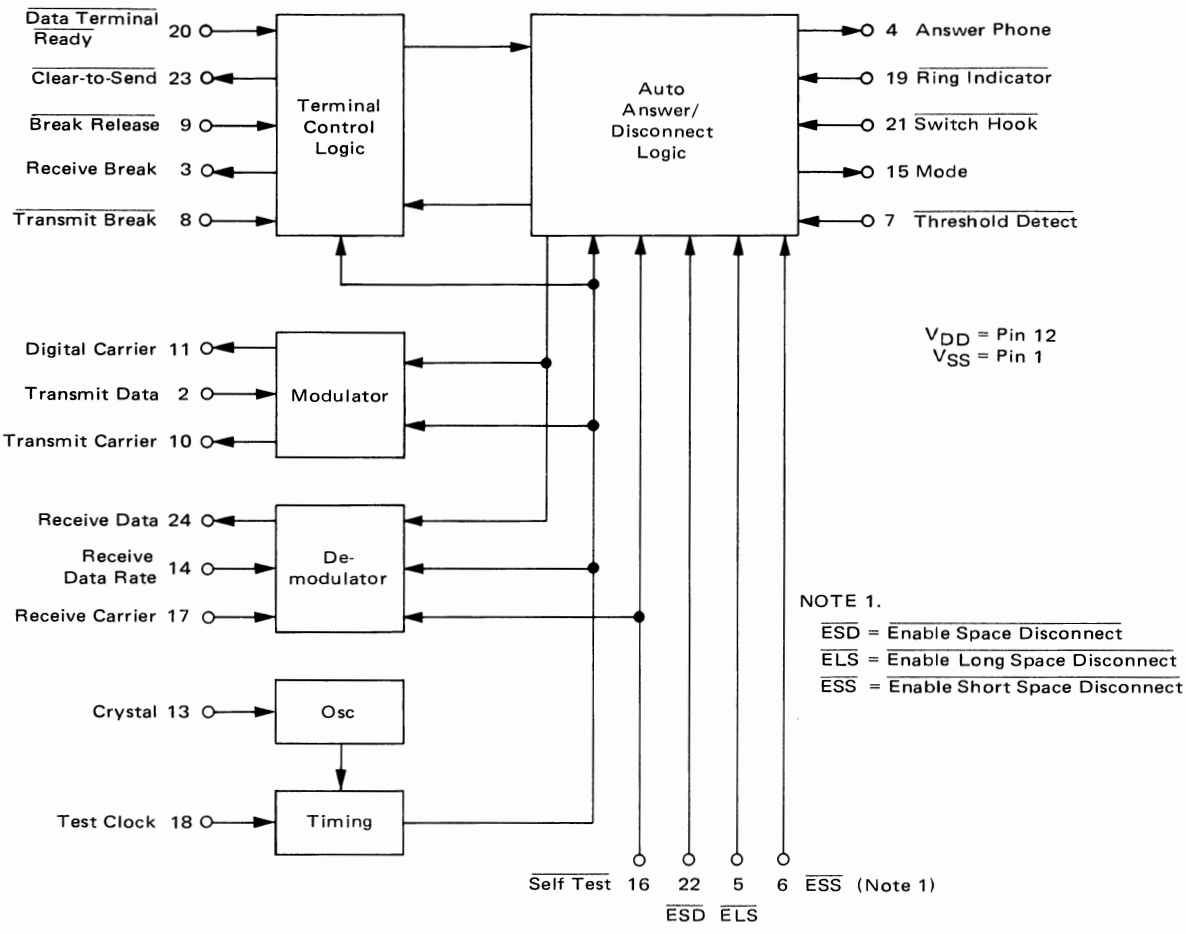
CASE 684-04



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	29.34	30.86	1.155	1.215
B	12.70	14.22	0.500	0.560
C	3.05	3.94	0.120	0.155
D	0.38	0.51	0.015	0.020
F	0.89	1.40	0.035	0.055
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	2.92	3.68	0.115	0.145
L	14.86	15.87	0.585	0.625
M	–	15°	–	15°
N	0.51	1.14	0.020	0.045

NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE WITH MAXIMUM MATERIAL CONDITION.
2. LEAD NO. 1 CUT FOR IDENTIFICATION, OR BUMP ON TOP.
3. DIM "L" TO INSIDE OF LEADS. (MEASURED 0.51 mm (0.020) BELOW PKG BASE)

BLOCK DIAGRAM



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 $\geqslant$ 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

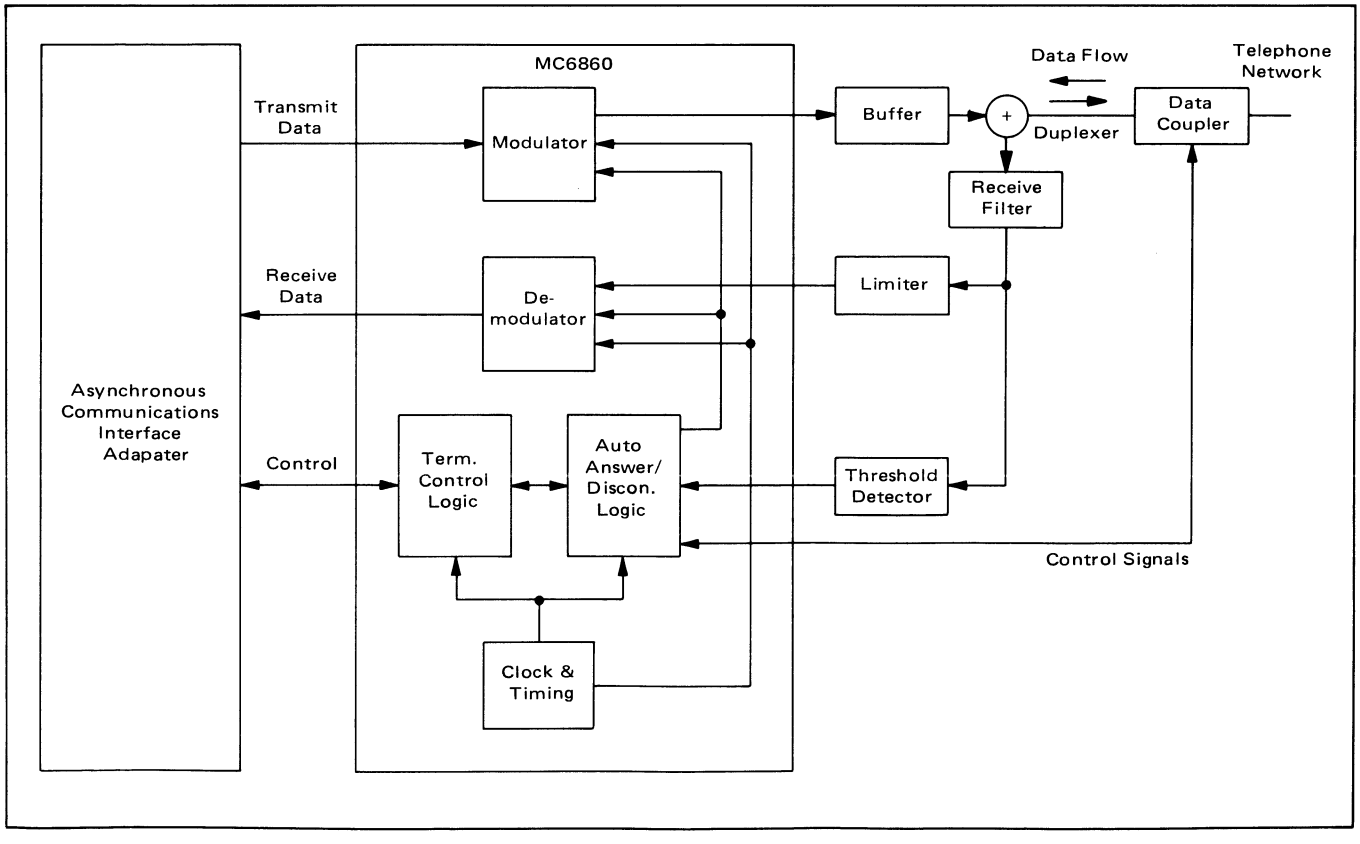
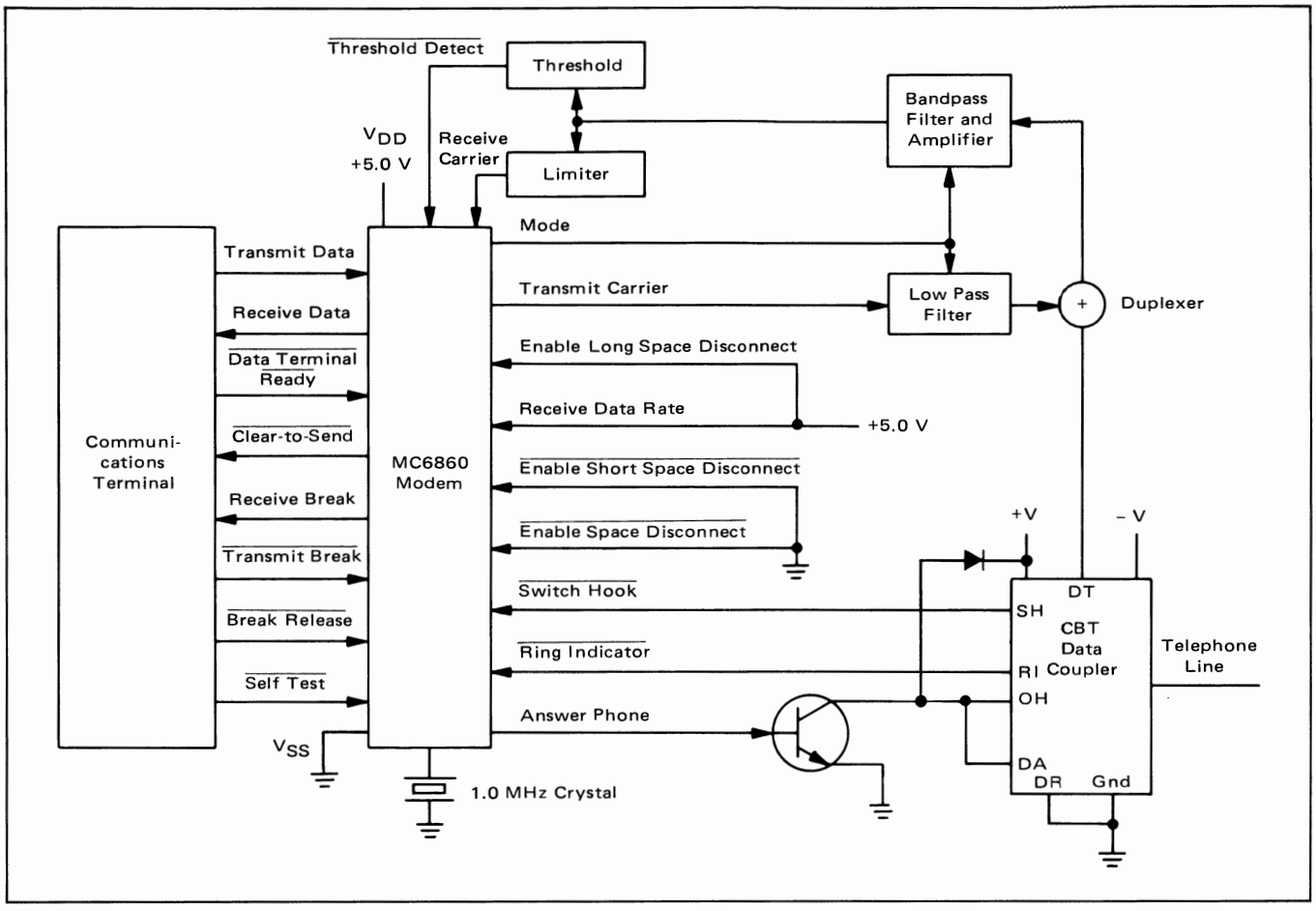


FIGURE 3 – I/O INTERFACE CONNECTIONS FOR MC6860
(ORIGINATE/ANSWER MODEM)



34 ms. The local modem then sends a 3 s continuous space and hangs up provided the Enable Space Disconnect is low. If the remote modem hangs up before 3 s, loss of Threshold Detect will cause loss of Clear-to-Send, which marks the line in Answer Mode and turns the carrier off in the Originate Mode.

If $\overline{ESD}$ is high the modem will transmit data until hang-up occurs 3 s later. Transmit Break is clamped 150 ms following the 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.

Ring Indicator ($\overline{RI}$)

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 $\overline{RI}$ 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 $\overline{RI}$ signal is present for at least 51 ms. This input is held high except during ringing. A $\overline{RI}$ signal automatically places the modem function in the Answer Mode.

Switch Hook ($\overline{SH}$)

$\overline{SH}$ interfaces directly with the CBT and via the EIA RS-232 level conversion for the CBS. An $\overline{SH}$ signal automatically places the modem function in the Originate Mode.

$\overline{SH}$ is low during origination of a call. The modem will automatically hang up 17 s after releasing $\overline{SH}$ if the handshaking routine has not been accomplished.

Threshold Detect ($\overline{TD}$)

This input is derived from an external threshold detector. If the signal level is sufficient, the $\overline{TD}$ 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 $\leqslant$ 9 pF at the crystal input.

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 $[(\overline{SH} + \overline{RI}) \bullet \overline{DTR}]$. 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

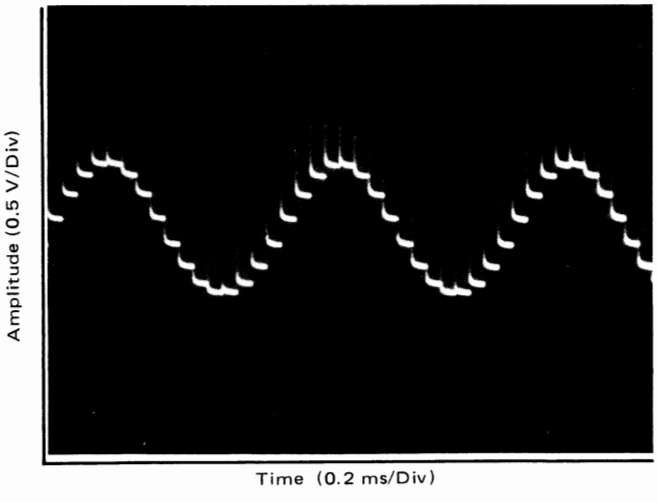
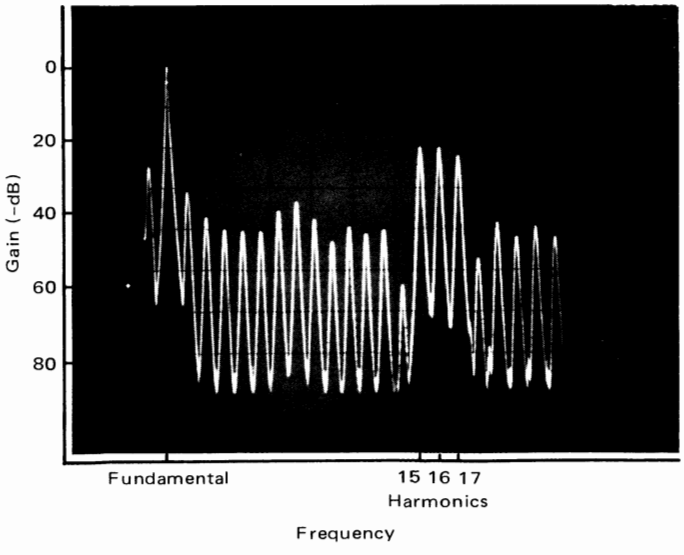


FIGURE 5 – TRANSMIT CARRIER FREQUENCY SPECTRUM



TIMING DIAGRAMS

FIGURE 6 – ANSWER MODE

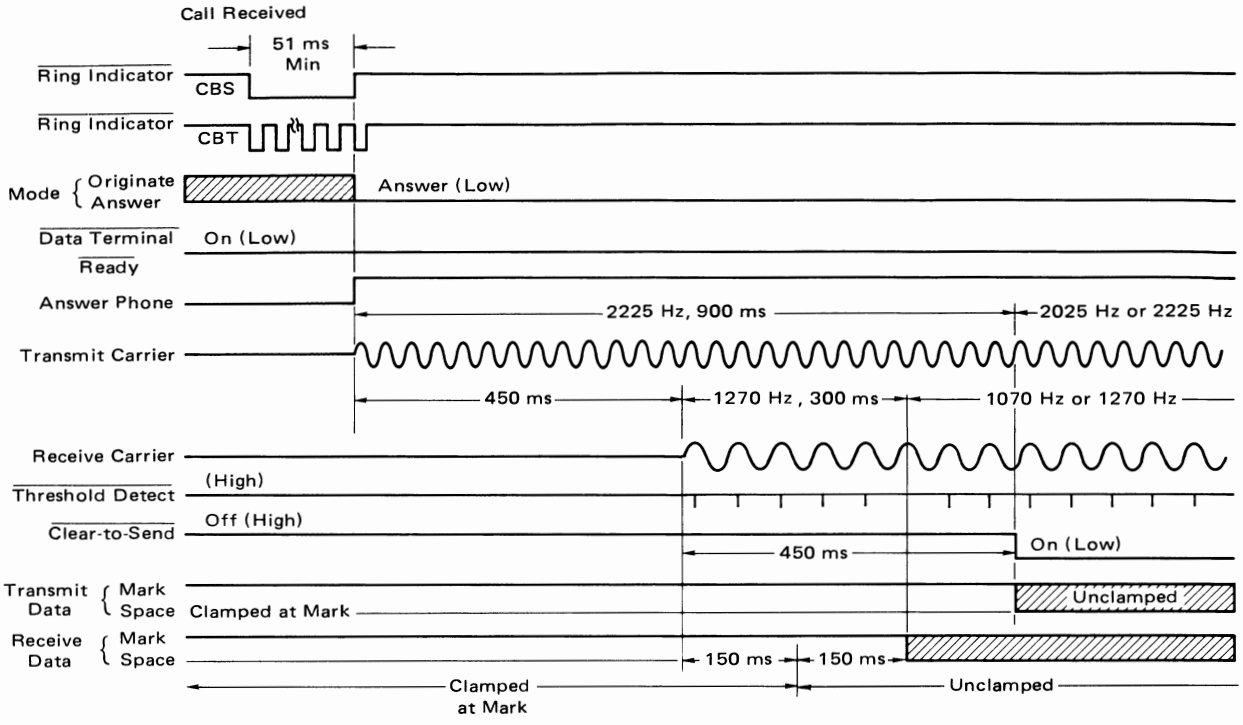


FIGURE 7 – AUTOMATIC DISCONNECT – LONG OR SHORT SPACE

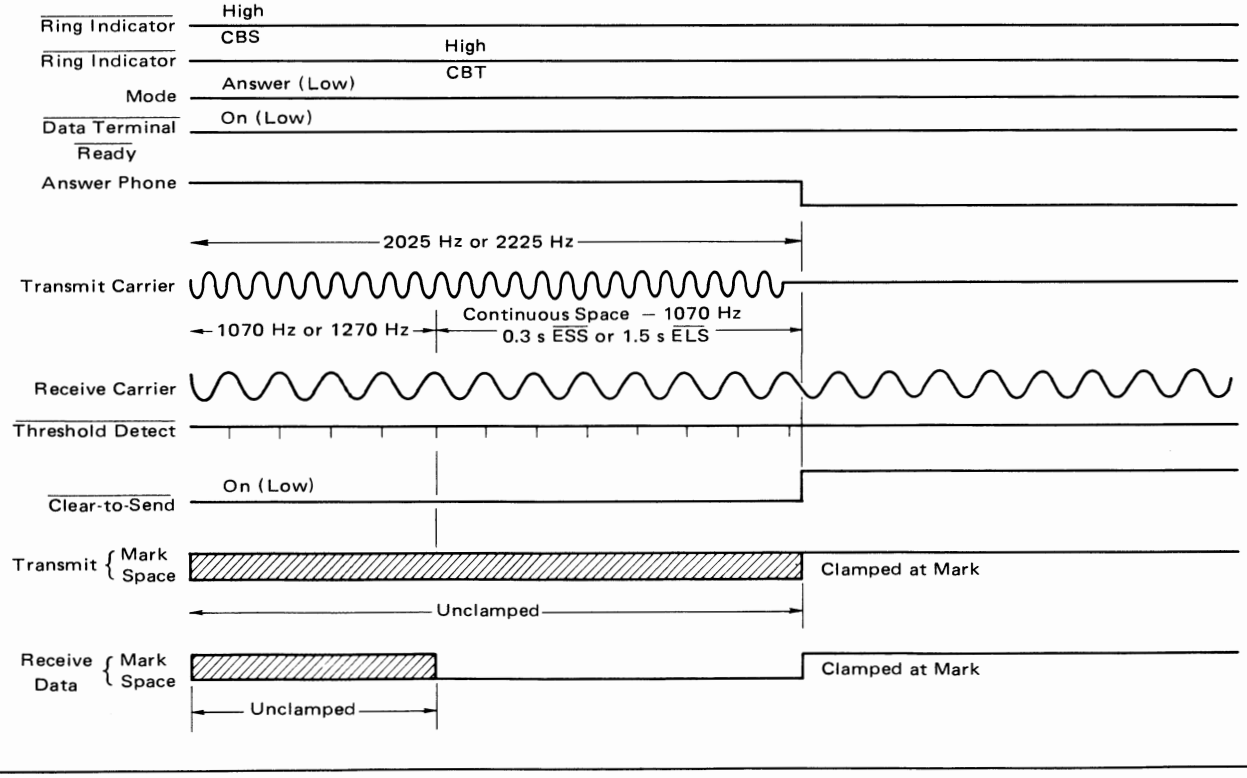


FIGURE 8 – ORIGINATE MODE

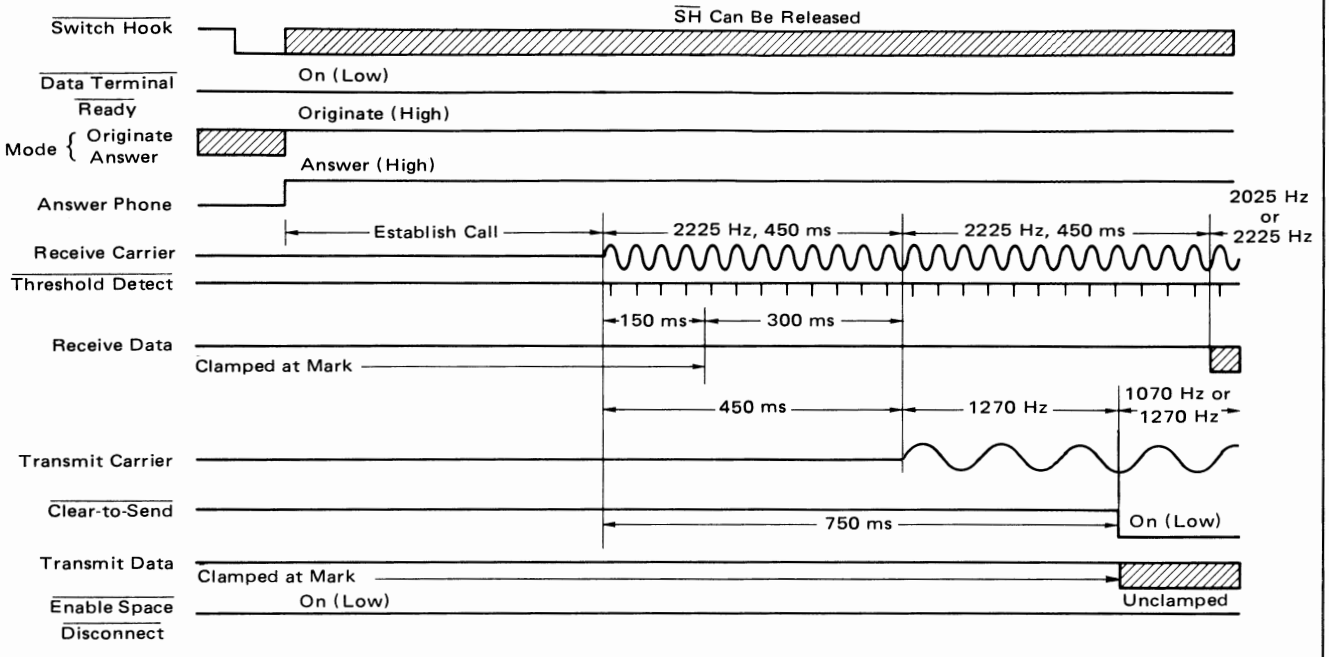


FIGURE 9 – INITIATE DISCONNECT

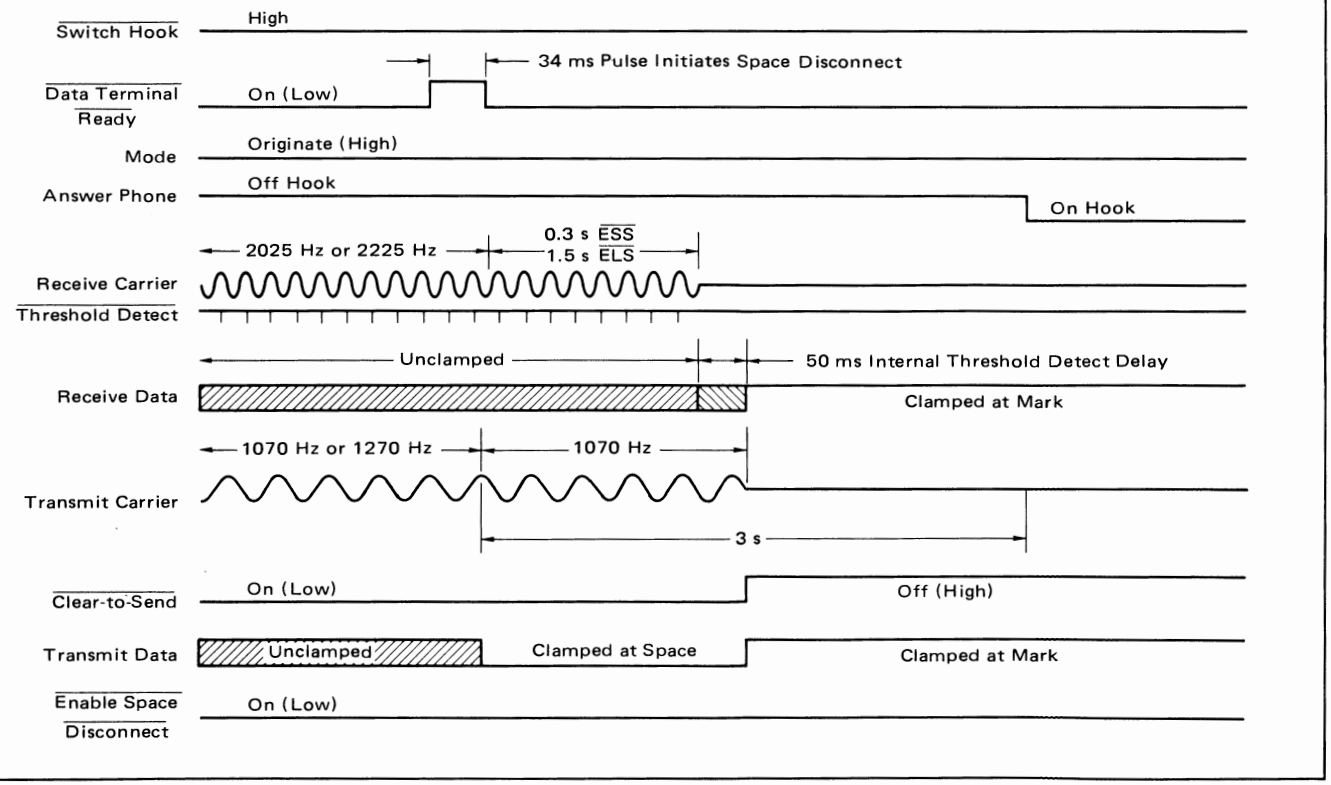
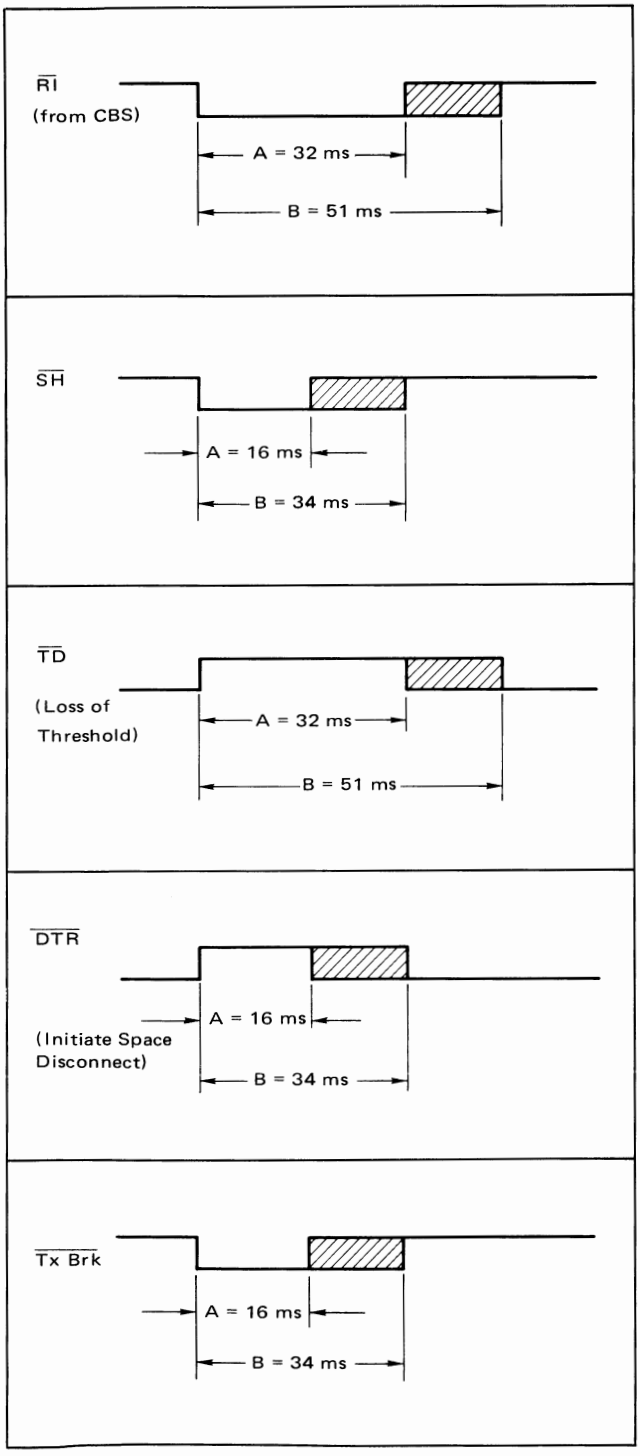


TABLE 1 – ASYNCHRONOUS INPUT PULSE WIDTH AND OUTPUT DELAY VARIATIONS
(Time delays specified do not include the 1-MHz reference tolerance.)

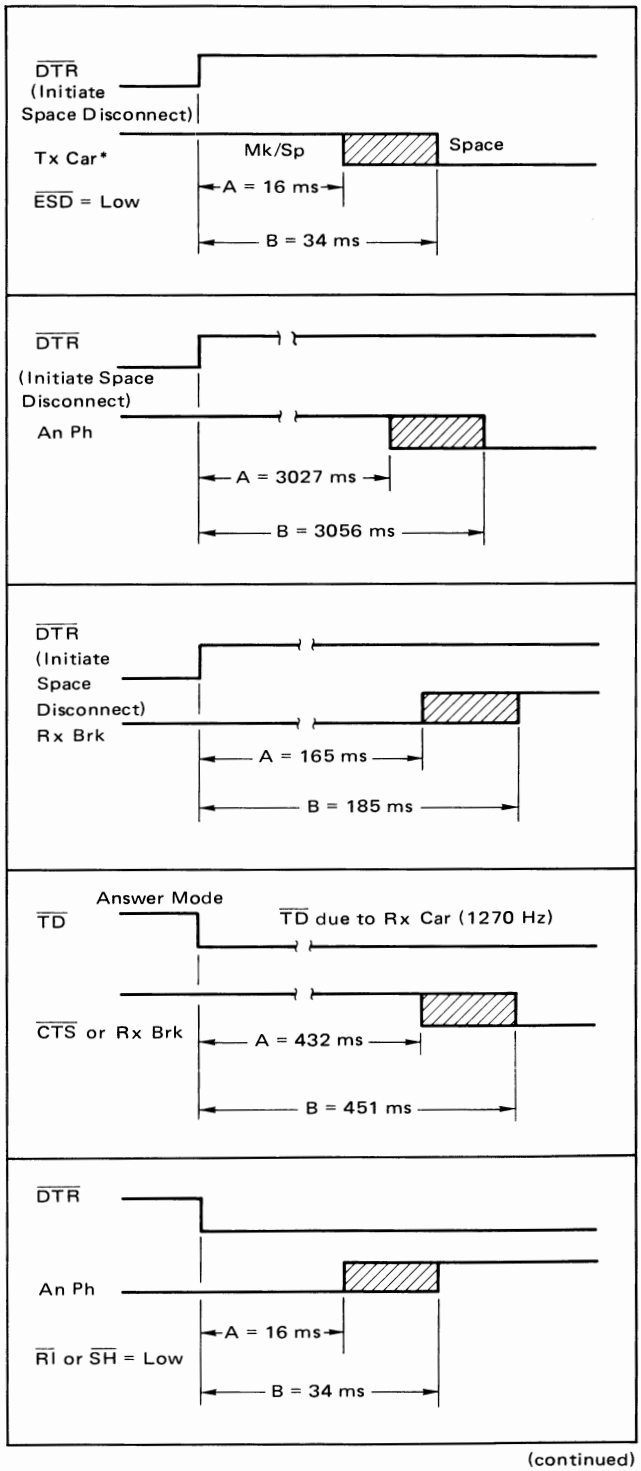
Due to the asynchronous nature of the input signals with respect to the circuit internal clock, a delay variation or input pulse width requirement will exist. Time delay A is the maximum time for which no response will occur. Time delay B is the minimum time required to guarantee an input response. Input 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



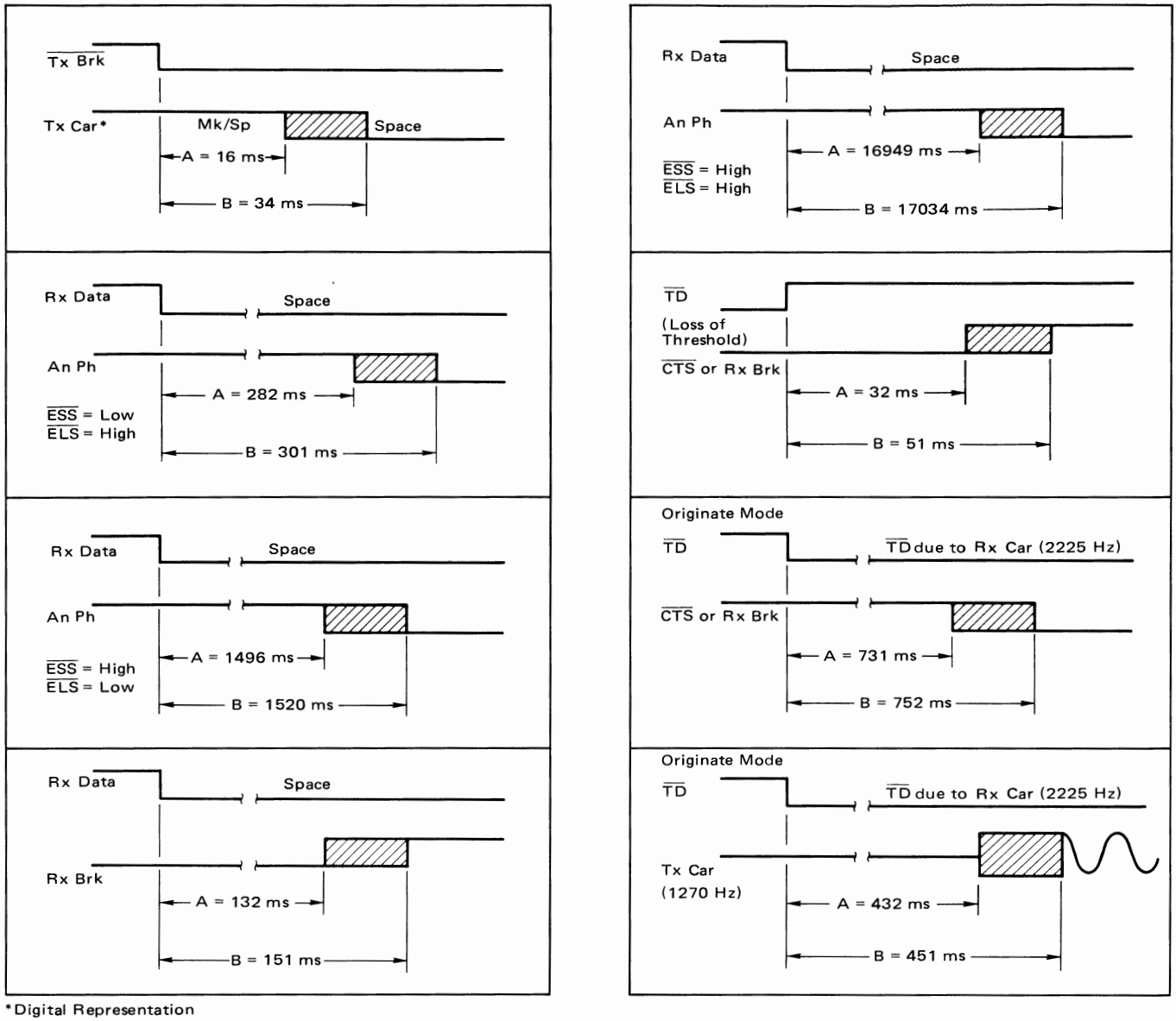
OUTPUT DELAYS



*Digital Representation.

(continued)

TABLE 1 – OUTPUT DELAY VARIATIONS (continued)

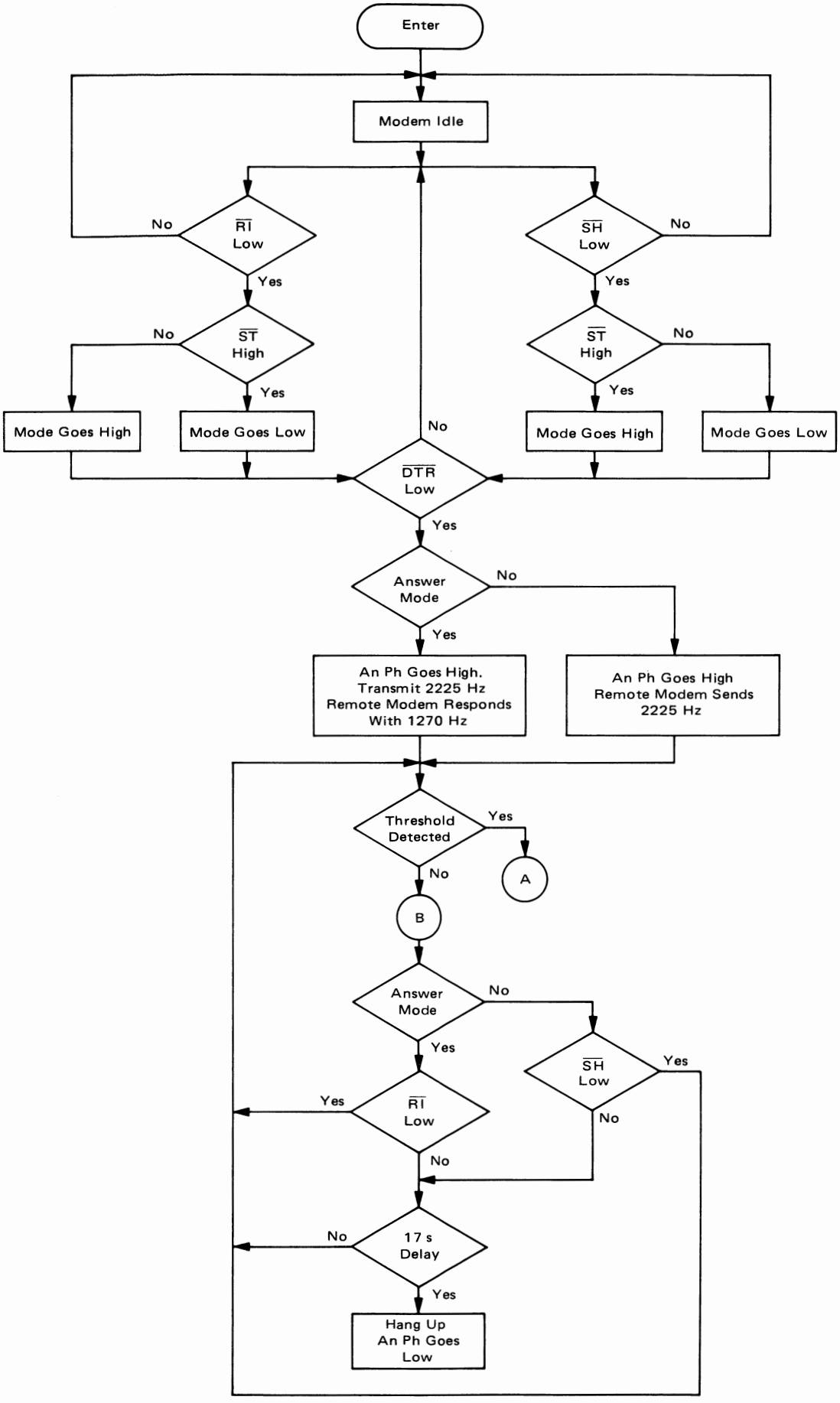


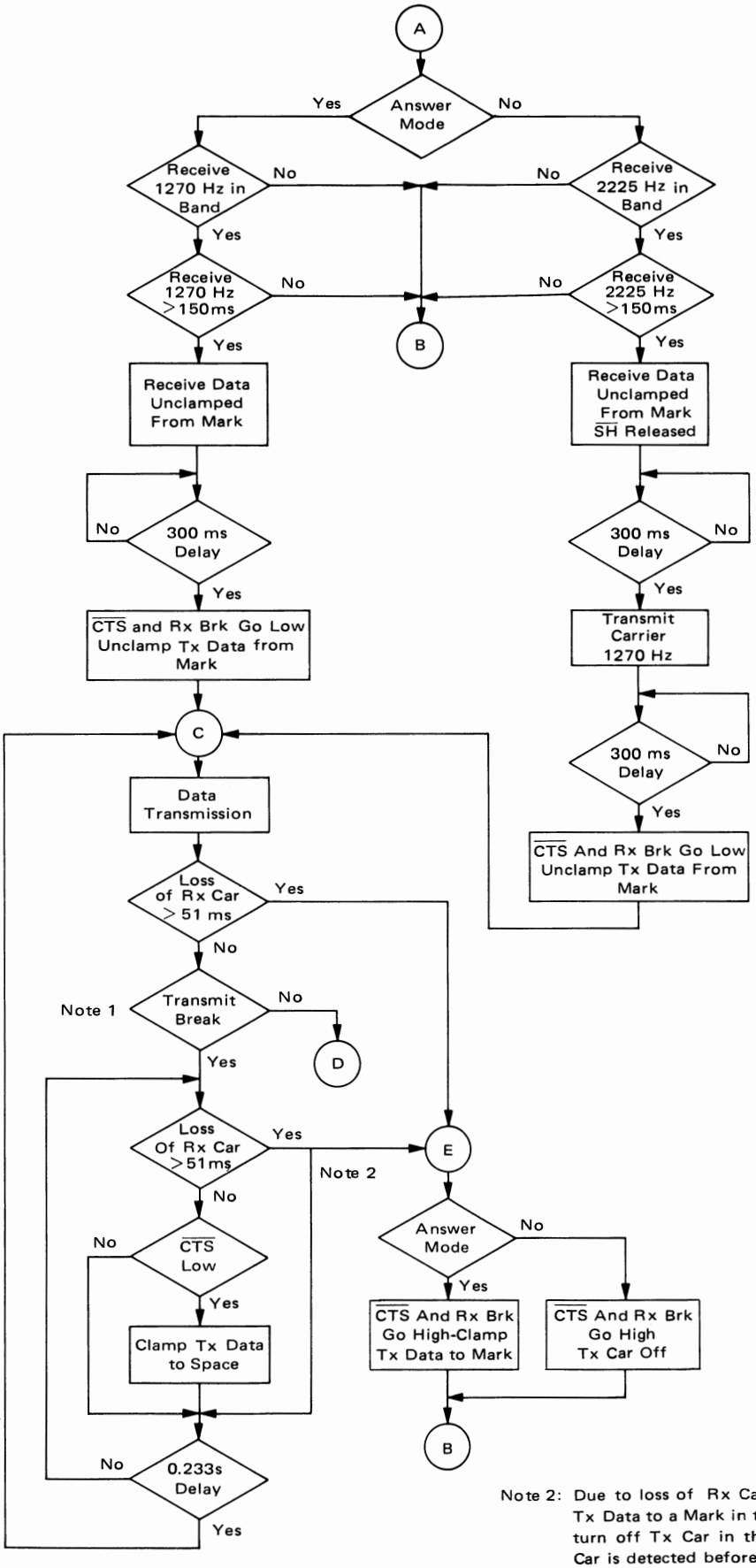
*Digital Representation

TABLE 2 – TRANSMIT BREAK AND DISCONNECT DELAYS

Function Description	Min	Max	Unit
$\overline{\text{Tx Brk}}$ (Space Duration)	232	235	ms
Space Disconnect (Space Duration) ($\overline{\text{DTR}}$ = High, $\overline{\text{ESD}}$ and $\overline{\text{TD}}$ = Low)	3010	3023	ms
Loss of Carrier Disconnect (Measured from positive edge of $\overline{\text{CTS}}$ to negative edge of An Ph, with $\overline{\text{RI}}$, $\overline{\text{SH}}$, and $\overline{\text{TD}}$ = High)	16965	17034	ms
Override Disconnect (Measured from positive edge of $\overline{\text{RI}}$ or $\overline{\text{SH}}$ to negative edge of An Ph, with $\overline{\text{TD}}$ = High)	16916	17101	ms

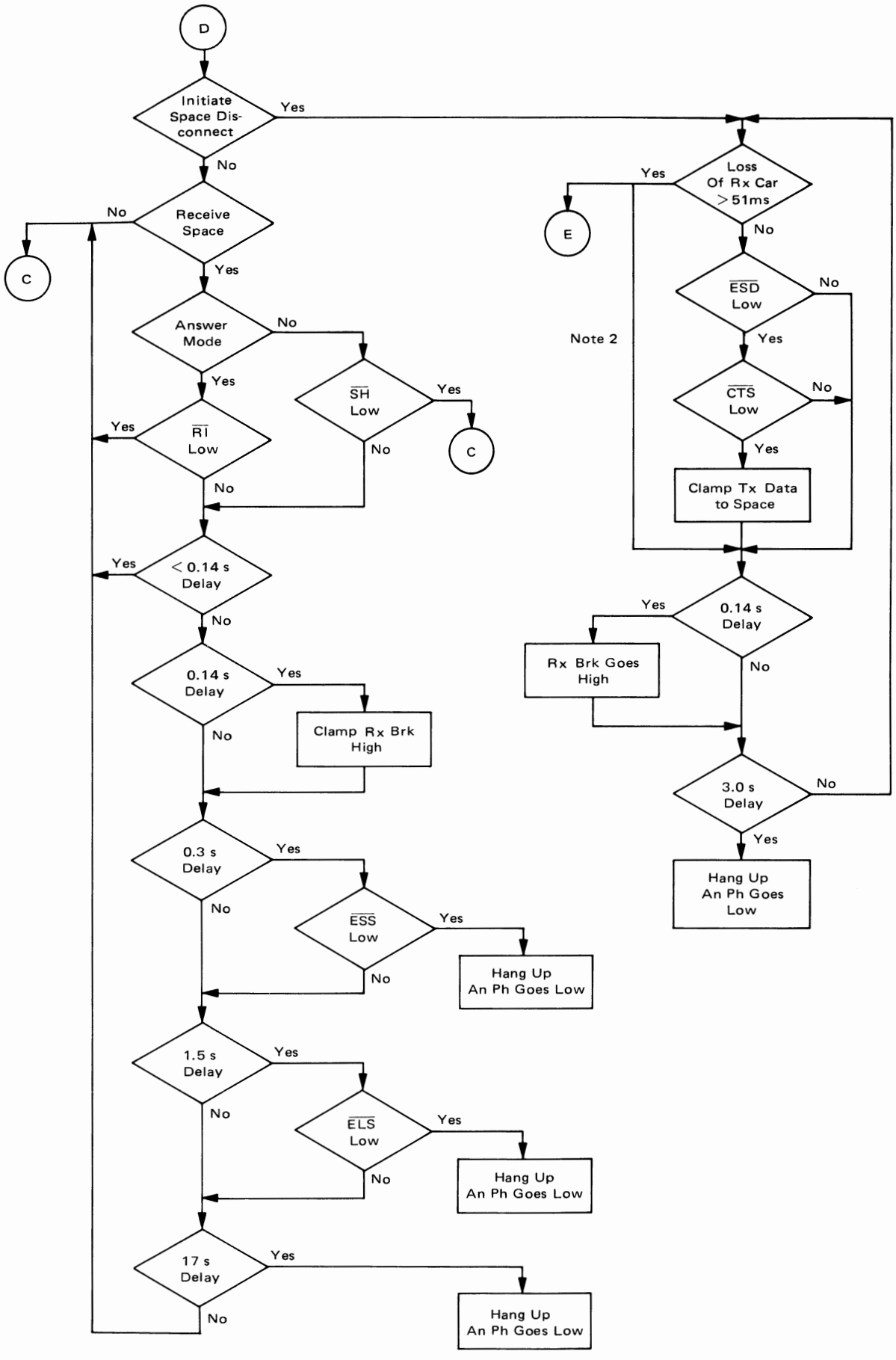
FIGURE 10 – FLOW DIAGRAM





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
< 0.14 s Delay
Yes
No
0.14 s Delay
Yes
Clamp Rx Brk High
No
0.3 s Delay
Yes
No
ESS Low
Yes
Hang Up An Ph Goes Low
No
1.5 s Delay
Yes
No
ELS Low
Yes
Hang Up An Ph Goes Low
No
17 s Delay
Yes
Hang Up An Ph Goes Low
No
Yes
E
Loss Of Rx Car > 51ms
No
ESD Low
No
Yes
Note 2
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

MOTOROLA Semiconductors
BOX 20912 • PHOENIX, ARIZONA 85036

MCM6810L
MCM6810L-1

Advance Information

MOS
(N-CHANNEL, SILICON-GATE)

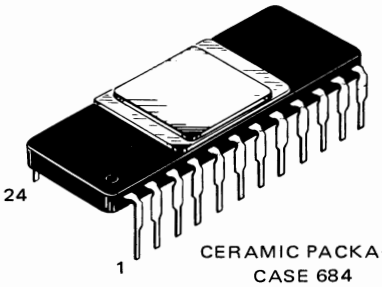
128 X 8-BIT STATIC RANDOM ACCESS MEMORY

128 X 8-BIT STATIC RANDOM ACCESS MEMORY

The MCM6810 is a byte-organized memory designed for use in bus-organized systems. It is fabricated with N-channel silicon-gate technology. For ease of use, the device operates from a single power supply, has compatibility with TTL and DTL, and needs no clocks or refreshing because of static operation.

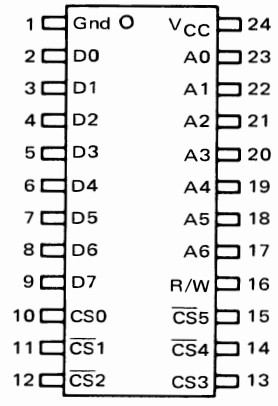
The memory is compatible with the M6800 Microcomputer Family, providing random storage in byte increments. Memory expansion is provided through multiple Chip Select inputs.

- Organized as 128 Bytes of 8 Bits
- Static Operation
- Bi-Directional Three-State Data Input/Output
- Six Chip Select Inputs (Four Active Low, Two Active High)
- Single 5-Volt Power Supply
- TTL Compatible
- Maximum Access Time = 1.0 μs for MCM6810L
  575 ns for MCM6810L-1



CERAMIC PACKAGE
CASE 684

PIN ASSIGNMENT

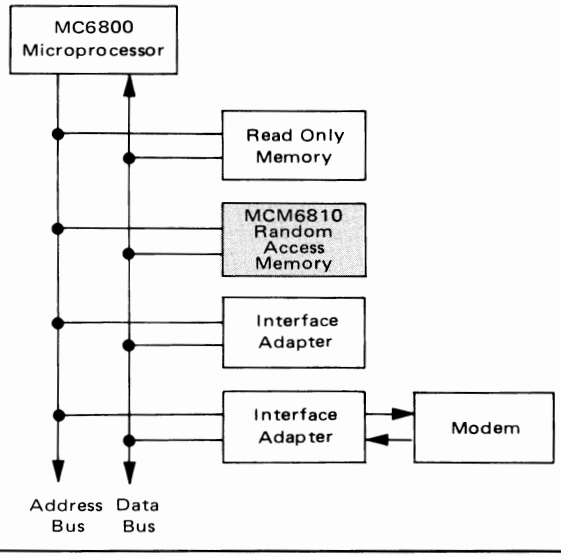


ABSOLUTE MAXIMUM RATINGS (See Note 1)

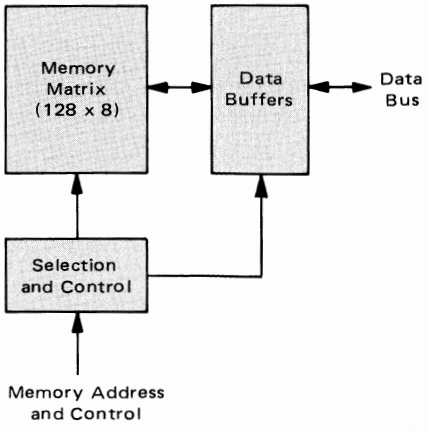
Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	-0.3 to +7.0	Vdc
Input Voltage	V_{in}	-0.3 to +7.0	Vdc
Operating Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	-55 to +150	°C

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.

M6800 MICROCOMPUTER FAMILY BLOCK DIAGRAM



MCM6810 RANDOM ACCESS MEMORY BLOCK DIAGRAM



DC OPERATING CONDITIONS AND CHARACTERISTICS

(Full operating voltage and temperature range unless otherwise noted.)

RECOMMENDED DC OPERATING CONDITIONS

Parameter	Symbol	Min	Typ	Max	Unit
Supply Voltage	V_{CC}	4.75	5.0	5.25	Vdc
Input High Voltage	V_{IH}	2.4	–	5.25	Vdc
Input Low Voltage	V_{IL}	-0.3	–	0.4	Vdc

DC CHARACTERISTICS

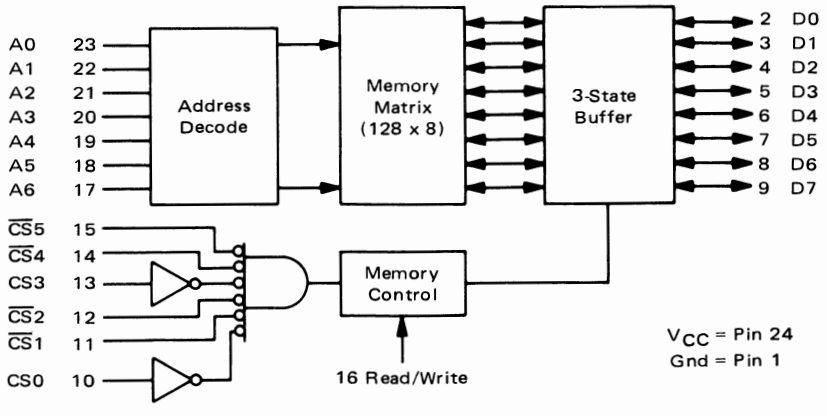
Characteristic	Symbol	Min	Typ	Max	Unit
Input Current (A_n, R/W, CS_n, $\overline{CS}_n$) (V_{in} = 0 to 5.25 V)	I_{in}	–	–	2.5	μAdc
Input High Threshold Voltage	V_{IHT}	2.0	–	–	Vdc
Input Low Threshold Voltage	V_{ILT}	–	–	0.8	Vdc
Output High Voltage (I_{OH} = -100 μA)	V_{OH}	2.4	–	–	Vdc
Output Low Voltage (I_{OL} = 1.6 mA)	V_{OL}	–	–	0.4	Vdc
Output Leakage Current (D0 – D7) (V_O = 2.4 V, CS = 0.4 V, $\overline{CS}$ = 2.4 V)	I_{LOH}	–	–	10	μAdc
Output Leakage Current (D0 – D7) (V_O = 0.4 V, CS = 0.4 V, $\overline{CS}$ = 2.4 V)	I_{LOL}	–	–	10	μAdc
Supply Current (V_{CC} = 5.25 V, T_A = 0°C)	I_{CC}	–	–	130	mAdc

CAPACITANCE (f = 1.0 MHz, T_A = 25°C, periodically sampled rather than 100% tested.)

Characteristic	Symbol	Max	Unit
Input Capacitance	C_{in}	7.5	pF
Output Capacitance	C_{out}	15	pF

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.

BLOCK DIAGRAM



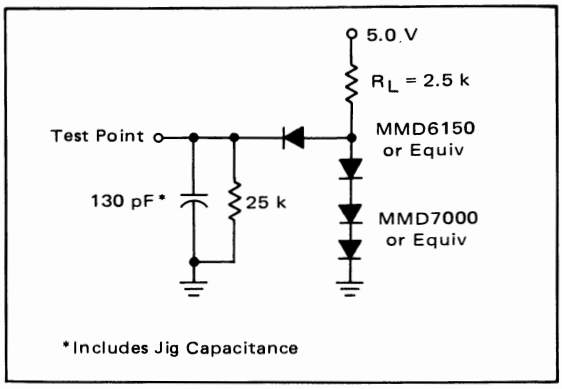
AC OPERATING CONDITIONS AND CHARACTERISTICS

(Full operating voltage and temperature unless otherwise noted.)

RECOMMENDED AC OPERATING CONDITIONS

Parameter	Symbol	Min	Unit
Address Setup Time	t_{AS}	30	ns
Address Hold Time	t_{AH}	0	ns
Chip Select Pulse Width MCM6810L MCM6810L-1	t_{CS}	 800 400	ns

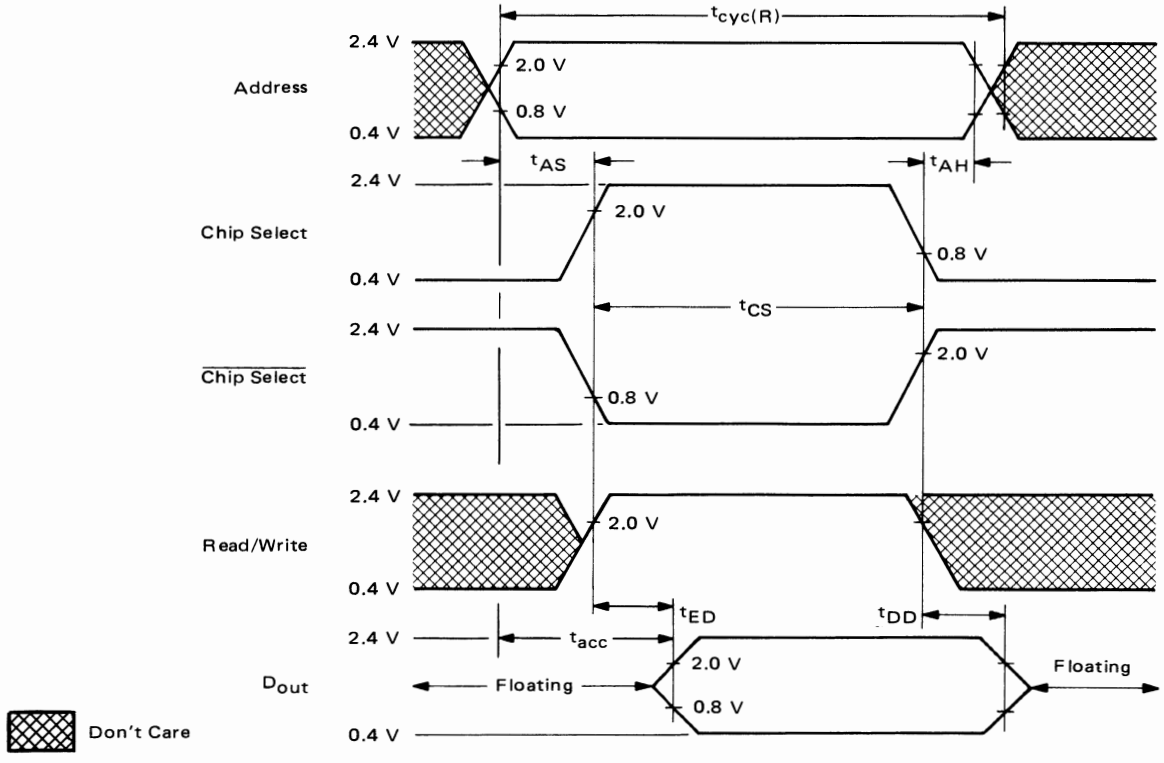
FIGURE 1 – AC TEST LOAD



READ CYCLE (All timing with $t_r = t_f = 20$ ns, Load of Figure 1)

Characteristic		Symbol	Min	Max	Unit
Read Cycle Time	MCM6810L MCM6810L-1	$t_{cyc(R)}$	1000 575	– –	ns ns
Output Enable Delay Time	MCM6810L MCM6810L-1	t_{ED}	– –	400 300	ns ns
Output Disable Delay Time	MCM6810L MCM6810L-1	t_{DD}	10 10	200 150	ns ns
Read Access Time	MCM6810L MCM6810L-1	t_{acc}	– –	1000 575	ns ns

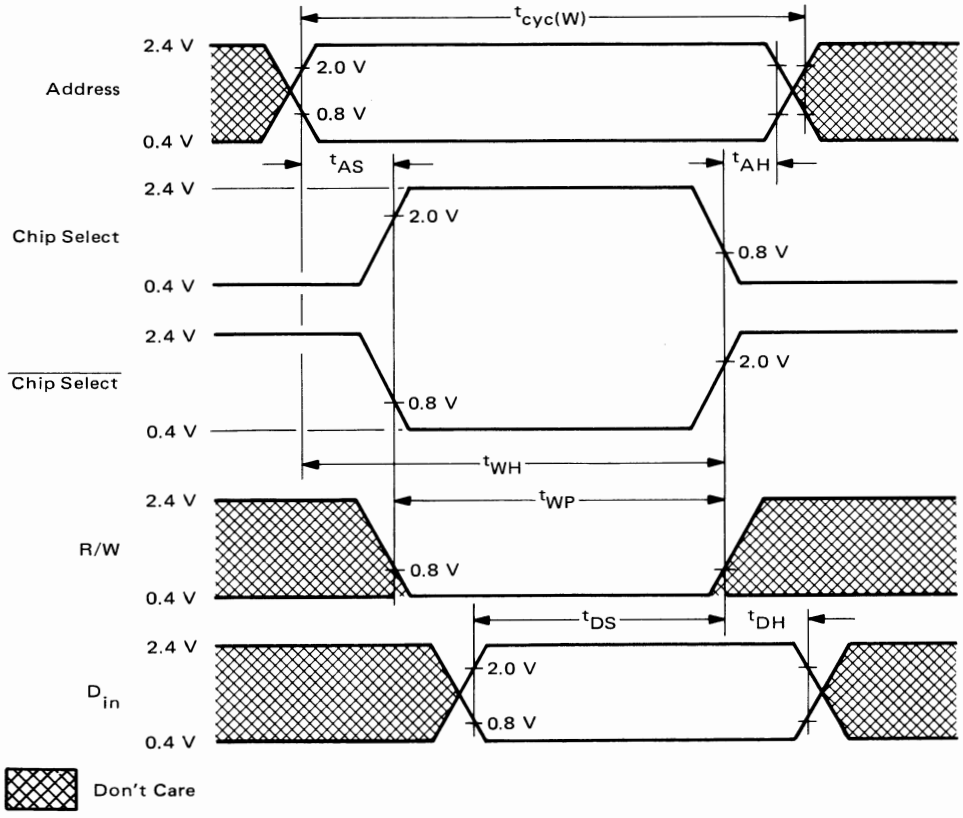
READ CYCLE TIMING



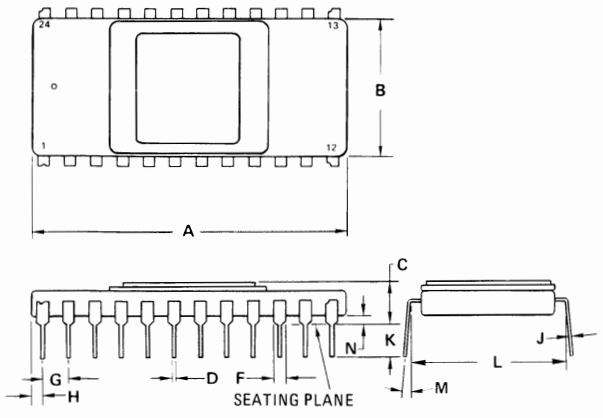
WRITE CYCLE (All timing with $t_r = t_f = 20$ ns, Load of Figure 1)

Characteristic		Symbol	Min	Max	Unit
Write Cycle Time	MCM6810L	$t_{cyc(W)}$	1000	–	ns
	MCM6810L-1		500	–	ns
Write Pulse Width	MCM6810L	t_{WP}	800	–	ns
	MCM6810L-1		400	–	ns
Write Pulse Hold Time	MCM6810L	t_{WH}	1000	–	ns
	MCM6810L-1		500	–	ns
Data Setup Time	MCM6810L	t_{DS}	500	–	ns
	MCM6810L-1		300	–	ns
Data Hold Time		t_{DH}	0	–	ns

WRITE CYCLE TIMING



PACKAGE DIMENSIONS



DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	29.34	30.86	1.155	1.215
B	12.70	14.22	0.500	0.560
C	3.05	3.94	0.120	0.155
D	0.38	0.51	0.015	0.020
F	0.89	1.40	0.035	0.055
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	2.92	3.68	0.115	0.145
L	14.86	15.87	0.585	0.625
M	–	15°	–	15°
N	0.51	1.14	0.020	0.045

NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE WITH MAXIMUM MATERIAL CONDITION.
2. LEAD NO. 1 CUT FOR IDENTIFICATION, OR BUMP ON TOP.
3. DIM "L" TO INSIDE OF LEADS. (MEASURED 0.51 mm (0.020) BELOW PKG BASE)

CASE 684-04

MOTOROLA Semiconductors
BOX 20912 • PHOENIX, ARIZONA 85036

MCM6830L

Advance Information

MOS

(N-CHANNEL, SILICON-GATE)

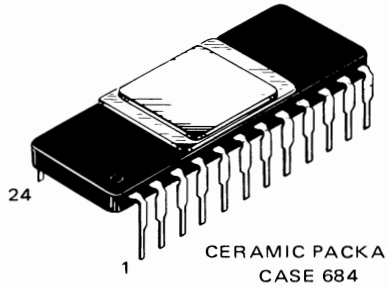
1024 X 8-BIT
READ ONLY MEMORY

1024 X 8-BIT READ ONLY MEMORY

The MCM6830 is a mask-programmable byte-organized memory designed for use in bus-organized systems. It is fabricated with N-channel silicon-gate technology. For ease of use, the device operates from a single power supply, has compatibility with TTL and DTL, and needs no clocks or refreshing because of static operation.

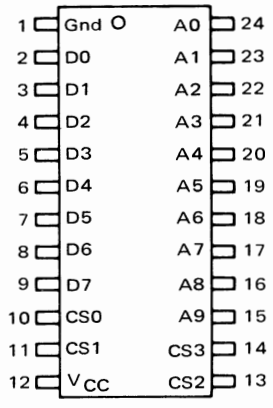
The memory is compatible with the M6800 Microcomputer Family, providing read only storage in byte increments. Memory expansion is provided through multiple Chip Select inputs. The active level of the Chip Select inputs and the memory content are defined by the customer.

- Organized as 1024 Bytes of 8 Bits
- Static Operation
- Three-State Data Output
- Four Chip Select Inputs (Programmable)
- Single 5-Volt Power Supply
- TTL Compatible
- Maximum Access Time = 575 ns



CERAMIC PACKAGE
CASE 684

PIN ASSIGNMENT

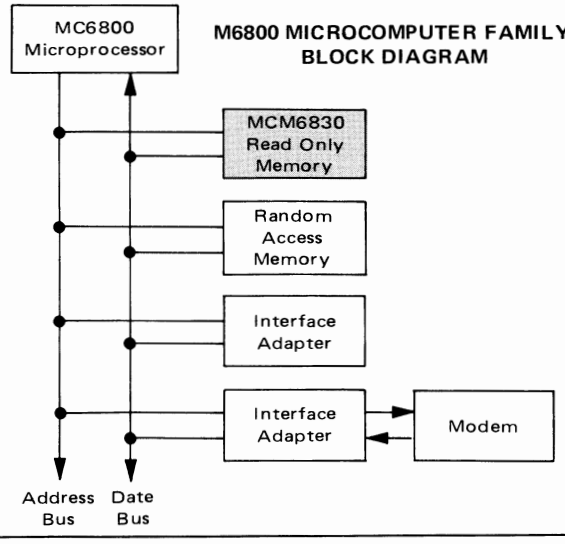


ABSOLUTE MAXIMUM RATINGS (See Note 1)

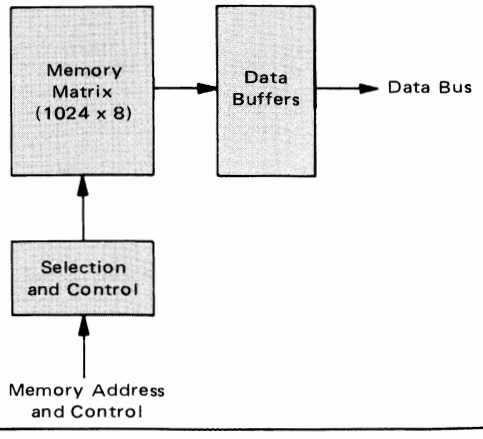
Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	-0.3 to +7.0	Vdc
Input Voltage	V_{in}	-0.3 to +7.0	Vdc
Operating Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	-55 to +150	°C

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.

M6800 MICROCOMPUTER FAMILY BLOCK DIAGRAM



MCM6830 READ ONLY MEMORY BLOCK DIAGRAM



DC OPERATING CONDITIONS AND CHARACTERISTICS

(Full operating voltage and temperature range unless otherwise noted.)

RECOMMENDED DC OPERATING CONDITIONS

Parameter	Symbol	Min	Typ	Max	Unit
Supply Voltage	V_{CC}	4.75	5.0	5.25	Vdc
Input High Voltage	V_{IH}	2.4	–	5.25	Vdc
Input Low Voltage	V_{IL}	-0.3	–	0.4	Vdc

DC CHARACTERISTICS

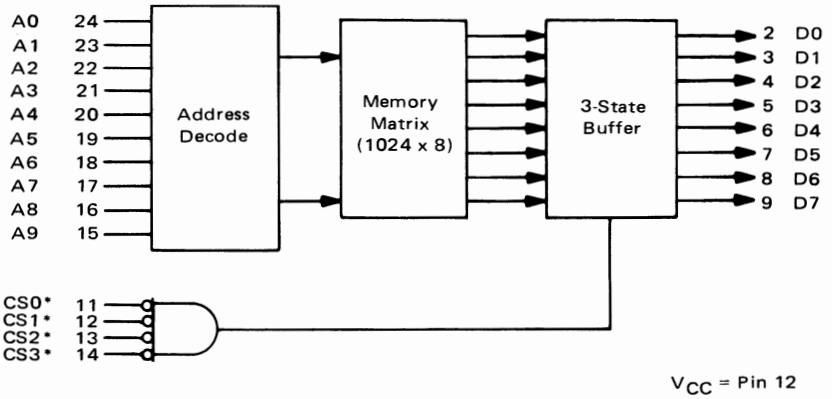
Characteristic	Symbol	Min	Typ	Max	Unit
Input Current (V_{in} = 0 to 5.25 V)	I_{in}	–	–	2.5	μAdc
Input High Threshold Voltage	V_{IHT}	2.0	–	–	Vdc
Input Low Threshold Voltage	V_{ILT}	–	–	0.65	Vdc
Output High Voltage (I_{OH} = -100 μA)	V_{OH}	2.4	–	–	Vdc
Output Low Voltage (I_{OL} = 1.6 mA)	V_{OL}	–	–	0.45	Vdc
Output Leakage Current (V_O = 2.4 V, CS = 0.4 V, $\overline{CS}$ = 2.4 V)	I_{LOH}	–	–	10	μAdc
Output Leakage Current (V_O = 0.4 V, CS = 0.4 V, $\overline{CS}$ = 2.4 V)	I_{LOL}	–	–	10	μAdc
Supply Current (V_{CC} = 5.25 V, T_A = 0°C)	I_{CC}	–	–	150	mAdc

CAPACITANCE (f = 1.0 MHz, T_A = 25°C, periodically sampled rather than 100% tested.)

Characteristic	Symbol	Max	Unit
Input Capacitance	C_{in}	7.5	pF
Output Capacitance	C_{out}	15	pF

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.

BLOCK DIAGRAM



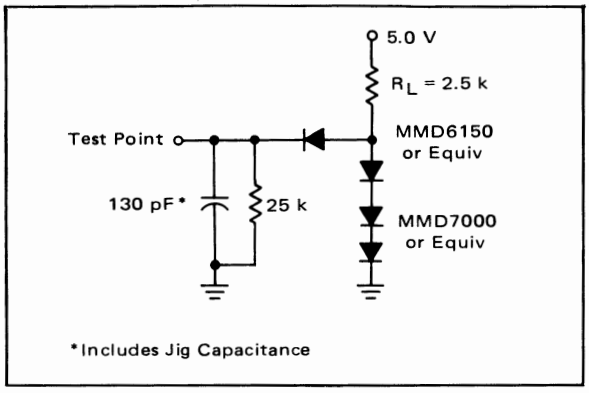
*Active level defined by the customer.

V_{CC} = Pin 12
Gnd = Pin 1

AC OPERATING CONDITIONS AND CHARACTERISTICS

(Full operating voltage and temperature unless otherwise noted.)

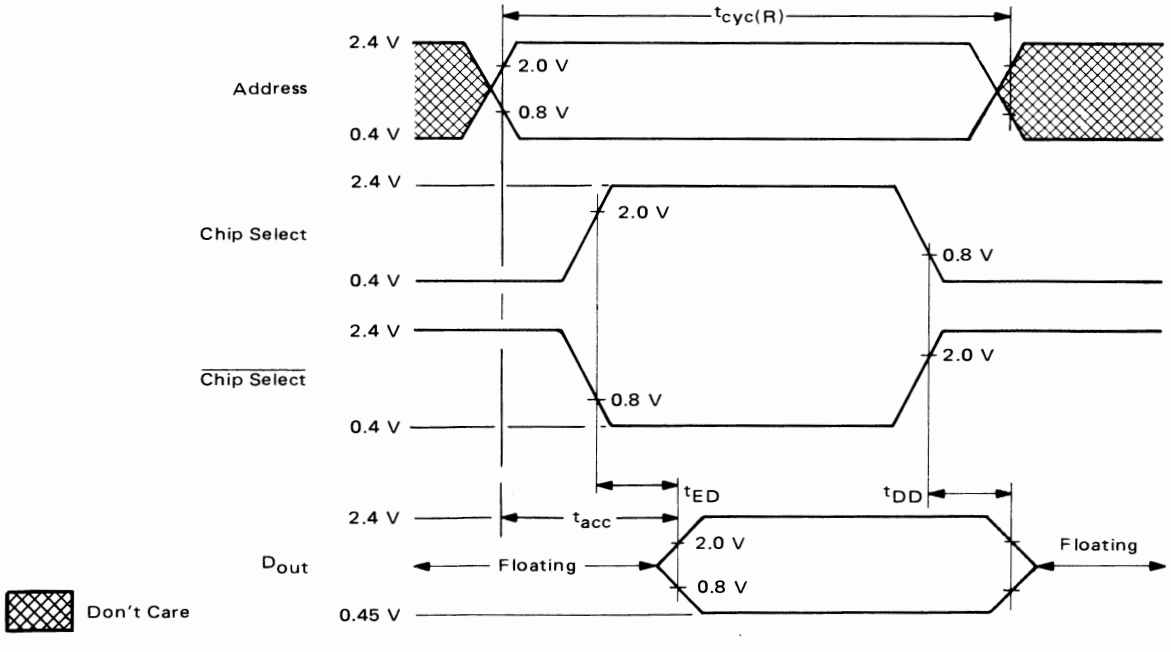
FIGURE 1 – AC TEST LOAD



READ CYCLE (All timing with $t_r = t_f = 20$ ns, Load of Figure 1)

Characteristic	Symbol	Min	Max	Unit
Read Cycle Time	$t_{cyc(R)}$	575	–	ns
Output Enable Delay Time	t_{ED}	–	300	ns
Output Disable Delay Time	t_{DD}	10	150	ns
Read Access Time	t_{acc}	–	575	ns

READ CYCLE TIMING



CUSTOM PROGRAMMING

By the programming of a single photomask for the MCM6830, the customer may specify the content of the memory and the method of enabling the outputs.

Information on the general options of the MCM6830 should be submitted on an Organizational Data form such as that shown in Figure 3.

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 procedure for generating and verifying a system is shown in Figure 4. 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 1024 bytes.

Note: Motorola can accept magnetic tape and truth table table formats. For further information, contact your local Motorola sales representative.

FIGURE 2 – BINARY TO HEXADECIMAL CONVERSION

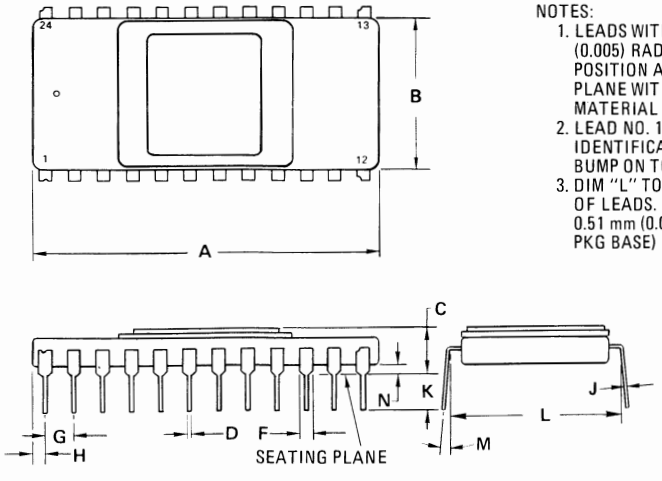
Binary Data				Hexadecimal Character
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	A
1	0	1	1	B
1	1	0	0	C
1	1	0	1	D
1	1	1	0	E
1	1	1	1	F

IBM PUNCH CARDS

The hexadecimal equivalent (from Figure 2) 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	14-73	Alternate steps 1 and 2 for consecutive bytes.
4	77-78	Card number (starting 01)
5	79-80	Total number of cards (32)

PACKAGE DIMENSIONS



NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE WITH MAXIMUM MATERIAL CONDITION.
2. LEAD NO. 1 CUT FOR IDENTIFICATION, OR BUMP ON TOP.
3. DIM "L" TO INSIDE OF LEADS. (MEASURED 0.51 mm (0.020) BELOW PKG BASE)

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	29.34	30.86	1.155	1.215
B	12.70	14.22	0.500	0.560
C	3.05	3.94	0.120	0.155
D	0.38	0.51	0.015	0.020
F	0.89	1.40	0.035	0.055
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	2.92	3.68	0.115	0.145
L	14.86	15.87	0.585	0.625
M	–	15^0	–	15^0
N	0.51	1.14	0.020	0.045

CASE 684-04

FIGURE 3 – FORMAT FOR PROGRAMMING GENERAL OPTIONS

ORGANIZATIONAL DATA
MCM6830 MOS READ ONLY MEMORY

Customer:

Company _____

Part No. _____

Originator _____

Phone No. _____

Motorola Use Only:

Quote: _____

Part No.: _____

Specif. No.: _____

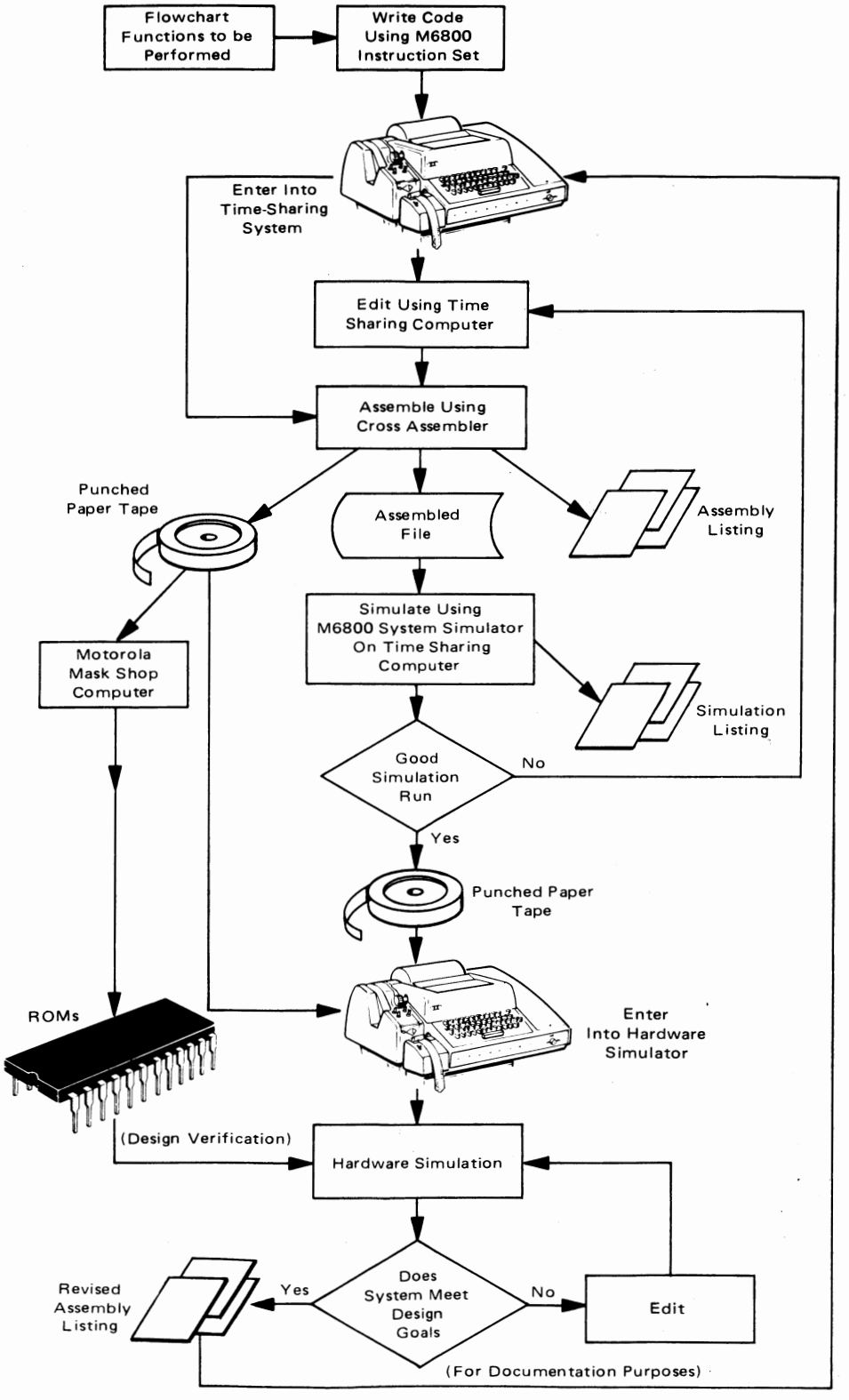
Enable Options:

	1	0
CS0	☐	☐
CS1	☐	☐
CS2	☐	☐
CS3	☐	☐

Input Logic Levels:

1 is most positive
0 is most negative
X is don't care

FIGURE 4 – SYSTEM DESIGN AND VERIFICATION PROCEDURE



MOTOROLA Semiconductor Products Inc.
BOX 20912 • PHOENIX, ARIZONA 85036 • A SUBSIDIARY OF MOTOROLA INC.
9701-7 PRINTED IN USA 1-76 IMPERIAL LITHO 854720 10M