S-2

OR 16

003367

T-33-07

MOT

MC6828
MC8507

Note: The dual numbering system emphasis that this device is a bipolar LSI service and directly compatible with the M6800X Microprocessor Family. The Priority Interrupt Controller may be ordered by using either part number.

PRIORITY INTERRUPT CONTROLLER

The MC6828/8507 Priority Interrupt Controller (PIC) is used to add prioritized responses to inputs to microprocessor systems. The performance has been optimized for the M6800X system, but will serve to eliminate input polling routines from any processor system.

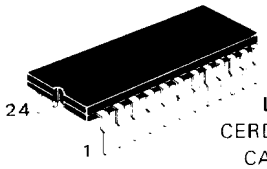
The MC6828/8507 (PIC) modifies the vector ROM addresses that the microprocessor uses to jump to an interrupt routine. The MC6828 provides the user with an additional eight latched interrupt inputs, and it can be cascaded to provide more interrupts.

An interrupt mask prevents any latched interrupt input of lower priority than the mask level from generating an $\overline{IRQ}$ output.

The (PIC) allows for any added decode time by generating a $\overline{Stretch}$ signal which can be used to slow the processor clock while fetching interrupt routine starting addresses. The $\overline{Stretch}$ signal allows the interrupt structure to be designed without concern for faster operation due to improvements in processor speeds.

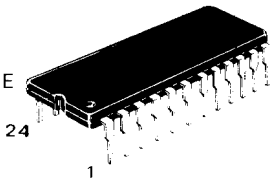
MEGALOGIC

PRIORITY INTERRUPT CONTROLLER

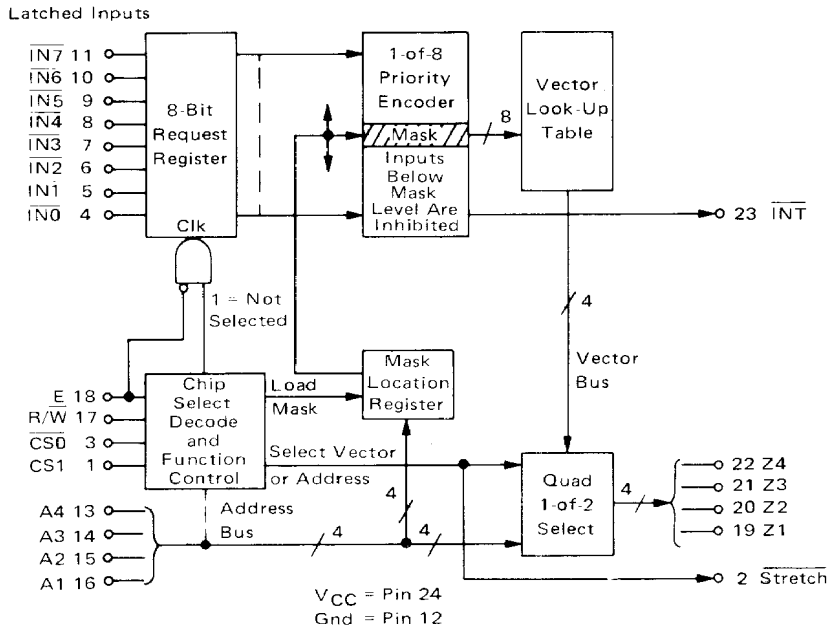


L SUFFIX
CERDIP PACKAGE
CASE 623-03

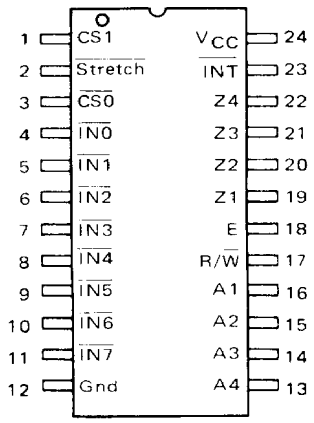
P SUFFIX
PLASTIC PACKAGE
CASE 649-03



BLOCK DIAGRAM



PIN ASSIGNMENT



MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	-0.5 to +7.0	Vdc
Input Voltage	V_{in}	-1.0 to +5.5	Vdc
Output Voltage	V_{OH}	-0.4 to +7.0	Vdc

THERMAL CHARACTERISTICS

Characteristic		Symbol	Max	Unit
Thermal Resistance	Cerdip	$R_{\theta JA}$	65	°C W
	Plastic		120	

POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$$T_J = T_A + (P_D \cdot \theta_{JA}) \quad (1)$$

Where:

T_A = Ambient Temperature, °C

θ_{JA} = Package Thermal Resistance, Junction-to-Ambient, °C/W

$P_D = P_{INT} + P_{PORT}$

$P_{INT} = I_{CC} \times V_{CC}$, Watts — Chip Internal Power

P_{PORT} = Port Power Dissipation, Watts — User Determined

For most applications $P_{PORT} \ll P_{INT}$ and can be neglected. P_{PORT} may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between P_D and T_J (if P_{PORT} is neglected) is:

$$P_D = K \div (T_J + 273°C) \quad (2)$$

Solving equations 1 and 2 for K gives:

$$K = P_D [T_A + 273°C + (P_D \cdot \theta_{JA})] \quad (3)$$

Where K is a constant pertaining to the particular part. K can be determined from equation (3) by measuring P_D (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 Vdc ±5%, T_A = 0 to 75°C unless otherwise noted)

Characteristic		Symbol	Min	Max	Unit
Input Forward Current (V_{IL} = 0, V_{CC} = 5.25 Vdc)	CS1, E	I_{IL}	—	-75	μAdc
	$\overline{CS0}$, R/$\overline{W}$		—	-150	
	A1 thru A4		—	-225	
	$\overline{IN0}$ thru $\overline{IN7}$		—	-1300	
Input Leakage Current (V_{IH} = 2.4 Vdc, V_{CC} = 5.25 Vdc)	CS1	I_{IH}	—	120	μAdc
	$\overline{CS0}$		—	240	
	A1 thru A4		—	360	
	$\overline{IN0}$ thru $\overline{IN7}$		—	-560	
DC Logic "0" Output Voltage (I_{OL} = 1.6 mAdc, V_{ILT} = 0.8 Vdc, V_{IHT} = 2.0 Vdc, V_{CC} = 4.75 Vdc)	Z1 thru Z4, $\overline{Stretch}$	V_{OL}	—	0.5	Vdc
(I_{OL} = 3.2 mAdc, V_{CC} = 4.75 Vdc)	$\overline{IRQ}$ — Open Collector		—	0.5	
DC Logic "1" Output Voltage (I_{OH} = -0.3 mAdc, V_{ILT} = 0.8 Vdc, V_{IHT} = 2.0 Vdc, V_{CC} = 4.75 Vdc)	Z1 thru Z4, $\overline{Stretch}$	V_{OH}	2.4	—	Vdc
Output Leakage Current (V_{CC} = V_{CEX} = 5.25 Vdc)	$\overline{INT}$	I_{CEX}	—	200	μAdc
Power Supply Drain Current (V_{CC} = 5.0 Vdc, All Inputs Open)		I_{CC}	—	125	mAdc

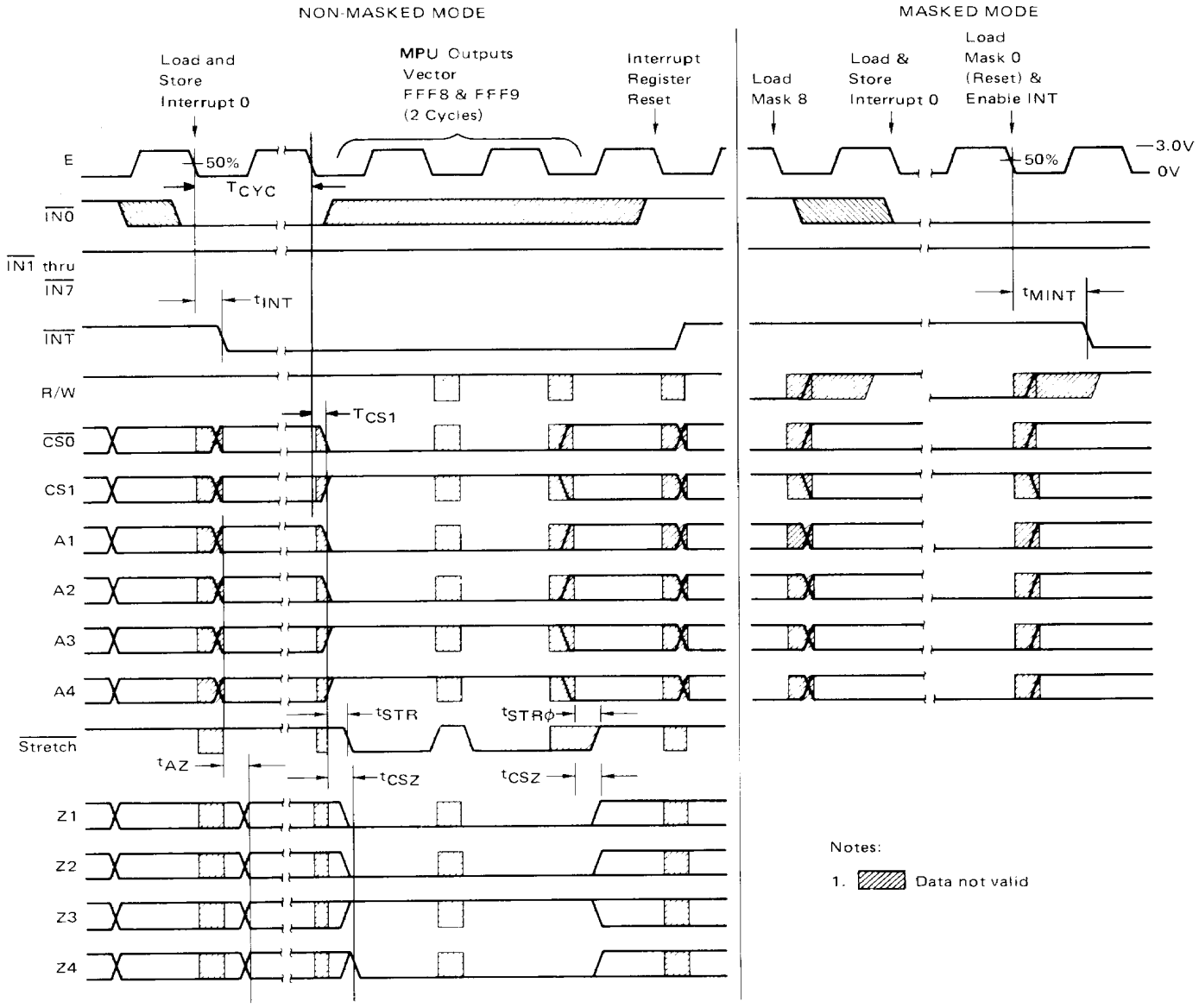
SWITCHING TIMES (V_{CC} = 5.0 Vdc, T_A = 25°C)

Characteristic	Symbol	Min	Max	Unit
A_i to Z_i Delay Time (Not Selected)	t_{AZ}	—	75	ns
A_i to Z_i Delay Time (Selected)	t_{AZ}	—	60	ns
Select* to Z_i Delay Time ($\overline{A1} \cdot \overline{A2} \cdot A3 \cdot A4 \cdot CS0 \cdot CS1$ to Z_i)	t_{CSZ}	—	125	ns
Enable Pulse Width	T_{CYC}	100	—	ns
Enable Low to CS1	T_{CS1}	125	—	ns
Deselect to $\overline{\text{Stretch}}$ High	$t_{STR\phi}$	—	125	ns
Select* to $\overline{\text{Stretch}}$ Delay Time ($\overline{A1} \cdot \overline{A2} \cdot A3 \cdot A4 \cdot \overline{CS0} \cdot CS1$ to $\overline{\text{Stretch}}$)	t_{STR}	—	140	ns
Enable to $\overline{INT}$ Delay Time, Non-Masked Mode	t_{INT}	—	240	ns
Enable to $\overline{INT}$ Delay Time, Masked Mode	$t_{MINT(IN1)}$ $t_{MINT(INJ)}$	— —	360** 200**	ns

*Select = ($\overline{A1} \cdot \overline{A2} \cdot A3 \cdot A4 \cdot \overline{CS0} \cdot CS1 \cdot R \cdot \overline{W}$) which corresponds to FFF8 or FFF9 interrupt response in the M6800 system.

**Value depends on mask level and stored priority input. Maximum value occurs with mask level 7 and stored interrupt IN0. Minimum value occurs with mask level J and stored interrupt INJ).

FIGURE 1 – FUNCTIONAL WAVEFORMS



OPERATING CHARACTERISTICS

The primary purpose of the Priority Interrupt Controller (PIC) is to generate a modified address to ROM in response to prioritized inputs. With the PIC, each interrupting device is assigned a unique ROM location which contains the starting address of the appropriate service routine. After the MPU detects and responds to an interrupt, the PIC directs the MPU to the proper memory location.

The basic functions of the PIC are shown in the block diagram. The 8-bit request register is an edge clocked D-type register with internal 6 kΩ pullup resistors on the interrupt inputs (IN0 thru IN7). Note the inputs are active low. The interrupt register is loaded on the falling edge of the enable when the PIC is not selected.

The 1-of-8 priority encoder enables a vector corresponding to the stored interrupt with the highest priority and places it on the vector input port of a data selector. In addition an interrupt request signal $\overline{INT}$ is generated to signal the MPU that an interrupt has been detected. The mask location register overrides and inhibits all interrupts with priority below the mask level. The mask can be thought of as a movable partition allowing responses to inputs equal to or greater than the mask value. For example if the stored mask level was 4, inputs $\overline{IN0}$, $\overline{IN1}$, $\overline{IN2}$, and $\overline{IN3}$ would not generate an interrupt to the MPU system. The input request register is not affected by the mask, and if the mask is cleared (by loading it with zeros), any previously stored inputs will generate an IRQ signal.

FIGURE 2 — MC6828 TRUTH TABLE FOR M6800 MICROPROCESSOR SYSTEMS

Active Input		Output When Selected Z4	Z3	Z2	Z1	Equivalent to Bits 1-4 of B0, B1 . . . , B15 Hex Address	Address ROM Bytes Contain Address of:
Highest	$\overline{IN7}$	1	0	1	1	F F F 6 or 7	Priority 7 Routine
	$\overline{IN6}$	1	0	1	0	F F F 4 or 5	Priority 6 Routine
	$\overline{IN5}$	1	0	0	1	F F F 2 or 3	Priority 5 Routine
	$\overline{IN4}$	1	0	0	0	F F F 0 or 1	Priority 4 Routine
	$\overline{IN3}$	0	1	1	1	F F E E or F	Priority 3 Routine
	$\overline{IN2}$	0	1	1	0	F F E C or D	Priority 2 Routine
	$\overline{IN1}$	0	1	0	1	F F E A or B	Priority 1 Routine
Lowest	$\overline{IN0}$	0	1	0	0	F F E 8 or 9	Priority 0 Routine
	None	1	1	0	0	F F F 8 or 9	Default Routine*

*Default routine is the response to interrupt requests not generated by a prioritized input. The default routine may contain polling routines or may be an address in a loop for an interrupt driven system.

FIGURE 3 — MC6828 TRUTH TABLE FOR M6809 MICROPROCESSOR SYSTEMS

Active Input		Output When Selected Z4	Z3	Z2	Z1	Equivalent Hex Address	Address ROM Bytes Contain Address of:
Highest	IN7	1	0	1	1	F F D 6 — F F D 7	Priority 7 Routine
	IN6	1	0	1	0	F F D 4 — F F D 5	Priority 6 Routine
	IN5	1	0	0	1	F F D 2 — F F D 3	Priority 5 Routine
	IN4	1	0	0	0	F F D 0 — F F D 1	Priority 4 Routine
	IN3	0	1	1	1	F F C E — F F C F	Priority 3 Routine
	IN2	0	1	1	0	F F C C — F F C D	Priority 2 Routine
	IN1	0	1	0	1	F F C A — F F C B	Priority 1 Routine
	IN0	0	1	0	0	F F C 8 — F F C 9	Priority 0 Routine
Lowest	None	1	1	0	0	F F F 8 — F F F 9	Default Routine ($\overline{IRQ}$)

Chip Select and Stretch

The chip select and decode circuitry controls all internal functions of the PIC. The selected mode is defined as the logical AND function $\overline{A1} \cdot \overline{A2} \cdot A3 \cdot A4 \cdot \overline{CS0} \cdot CS1 \cdot R/\overline{W}$. When the device is not in the selected mode the request register clock is enabled and the address inputs A_i passes directly through the data selector to the Z_i outputs. When the MPU responds to the interrupt request and the PIC decodes the select address, the request register is inhibited and the data selector places the vector on the Z outputs. The address delay added to the MPU system is shown in Figure 4. This delay may be critical in some systems. A stretch signal, which indicates the selected mode, is provided for use with special MPU clock drivers to stretch the clock cycle when accessing slow ROM. This stretch signal is applicable only to those microprocessors which incorporate external clock generators, i.e., 6800, 6809E. The user cannot directly connect stretch to MRDY on the MC6809 or MC6802, as stretching the clock circuit with a signal which is derived from the clock will latch up the MPU. An alternative to this problem is to use a one-shot which would provide the required amount of access time. Figure 8 illustrates a typical such circuit. The $\overline{CS0}$ output has one less gating level than the remainder of the select decode logic. This allows an external NAND gate to be used for the full address decode without any increase in delay times.

Programming the PIC

Changing the priority level, or mask, in the PIC is done by writing to the device. Unlike normal programming of a peripheral where a specific data pattern is written into a selected register, the PIC is programmed by accessing a location determined by A1 through A4 while $R/\overline{W}$ is low.

The decode logic also controls the loading of the mask location register. This register will be loaded on the falling edge of the enable pulse when enabled by the logical AND function $\overline{CS0} \cdot CS1 \cdot R/\overline{W}$ (Note 1). This means that in the load mask mode the data on the data bus is a don't care. However, in this mode the ROM will also be accessed and both the ROM and MPU will be driving the data bus. Therefore the read/write line should be used as an active high chip select or enable signal for ROM decoding.

Figure 5 shows the typical operation flow diagram for the PIC in an M6800 system. The functional timing for this flow is as shown in the first part of the waveforms in Figure 1. The second half of Figure 1 shows the operation of the mask. Interrupts will be stored even if they are masked. When the mask is released the $\overline{INT}$ signal will then be generated.

The influence of the mask register on the priority encoder is shown in the truth table of Figure 6. The actual use of the mask register will vary with the system needs and the imaginative software programmer.

Special Cases

As originally conceived, the PIC was only meant to be used with the MC6800 MPU. With the advent of higher performance/function microprocessors such as the MC6809/MC6802/MC6808, prioritized interrupts are still required. The interrupt vector map for the M6800 is located from FFF8 to FFFF. With the MC6809, this vector map extends downward to FFF0. As can be seen in Figure 2, the normal configuration of the PIC places priority interrupt vectors from FFE8 to FFF7 which conflict with the existing MC6809 interrupt vectors. Figure 1 shows appropriate circuitry required to interface with the MC6809. Figure 3 gives the "modified priority vectors" associated with this hardware modification.

Note 1. Since during normal operation of the MPU the address lines and the $R/\overline{W}$ line can be in an indeterminate state, VMA should be logically ANDed with one of the chip select inputs of the PIC to prevent erroneous writes into the mask register (non-6800 systems).

FIGURE 4 — HIGH ROM ADDRESS DELAY ADDED TO M6800X SYSTEMS

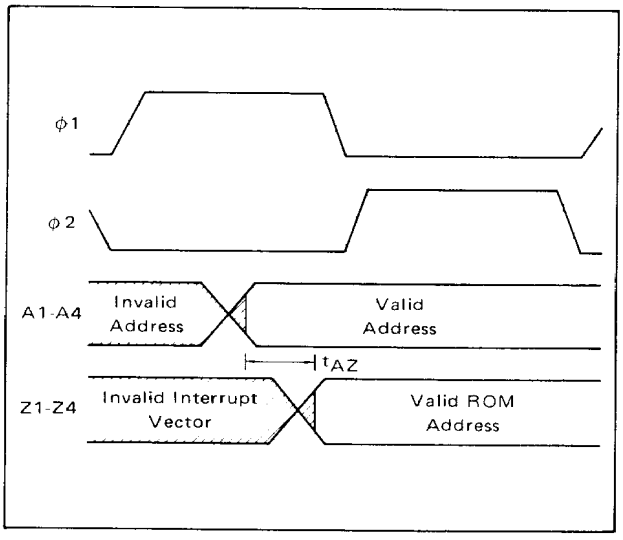


FIGURE 5 — BASIC FUNCTIONAL FLOW CHART

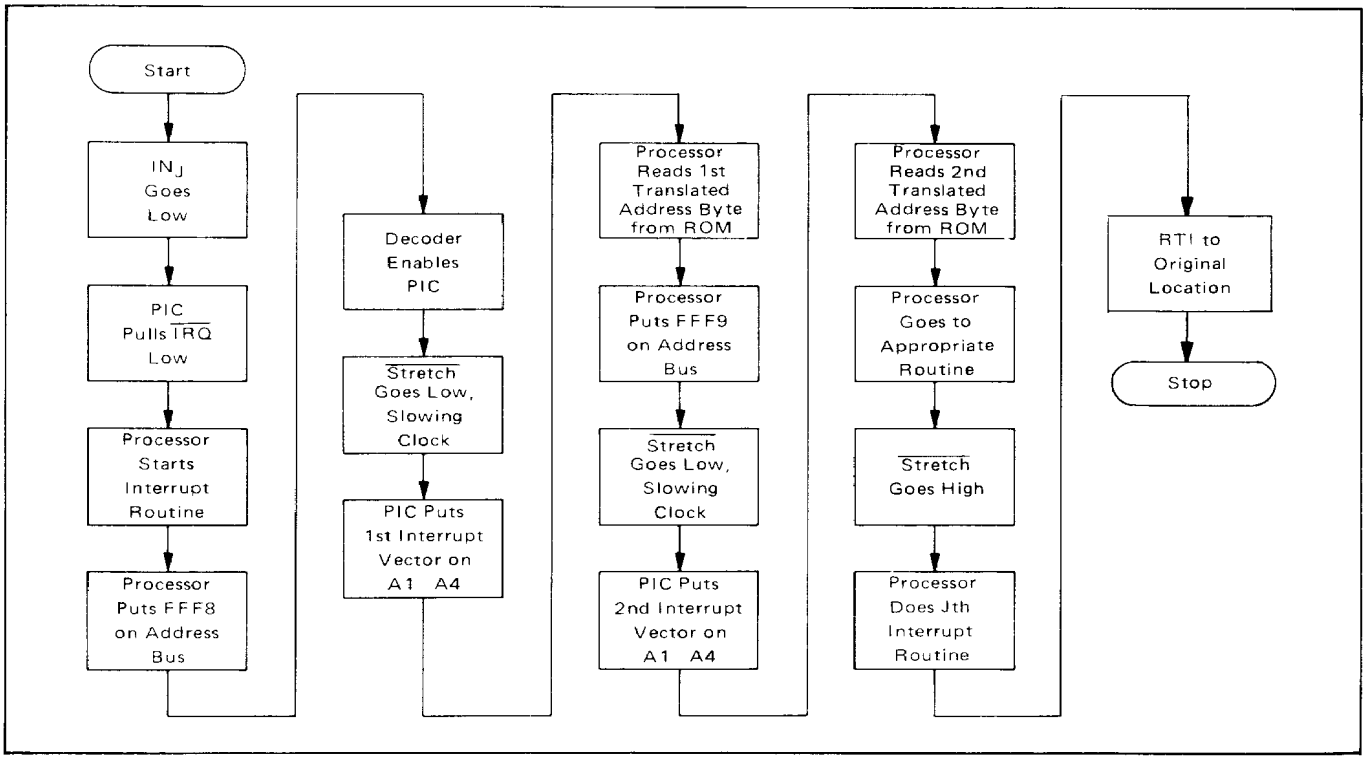
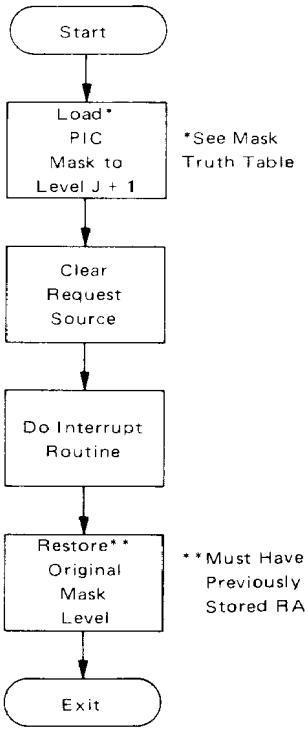


FIGURE 6 — MASK OPERATION

a – MASK FLOW CHART



b – MASK TRUTH TABLE

Mask Register Contents				Response to Priority Inputs 1 = Response to Input, 0 = No Response							
M4	M3	M2	M1	IN7	IN6	IN5	IN4	IN3	IN2	IN1	IN0
1	1	1	1	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	0	0	0	0	0	0	0
0	1	1	0	1	1	0	0	0	0	0	0
0	1	0	1	1	1	1	0	0	0	0	0
0	1	0	0	1	1	1	1	0	0	0	0
0	0	1	1	1	1	1	1	1	0	0	0
0	0	1	0	1	1	1	1	1	1	0	0
0	0	0	1	1	1	1	1	1	1	1	0
0	0	0	0	1	1	1	1	1	1	1	1

FIGURE 7 — TYPICAL SYSTEM CONFIGURATION

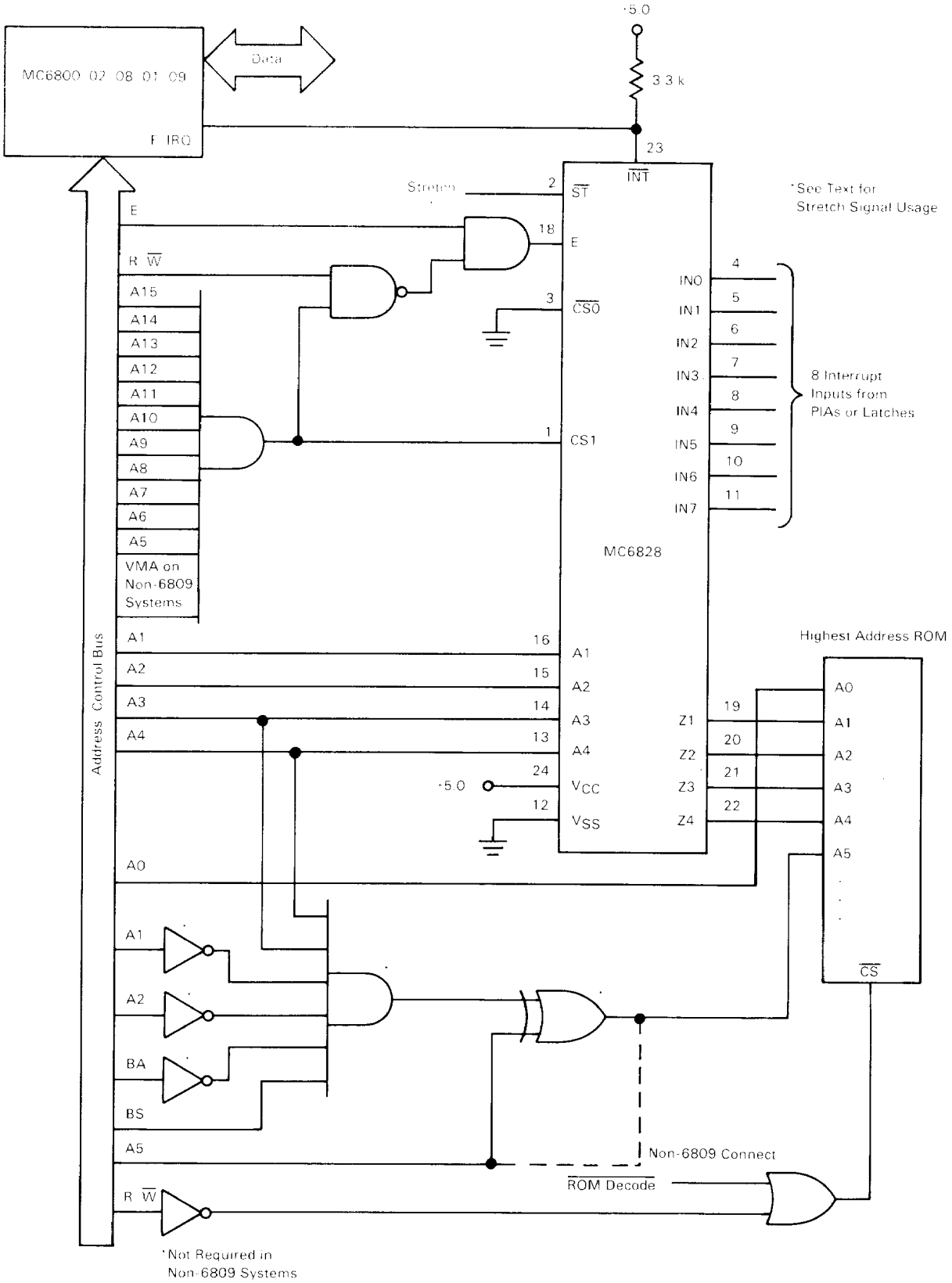
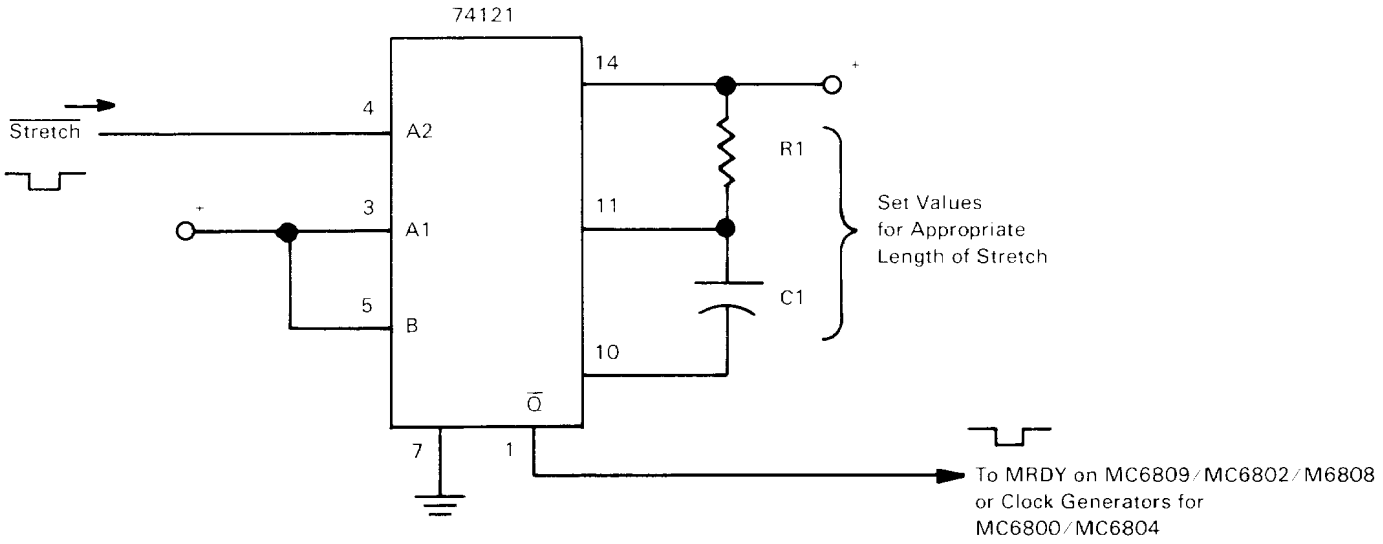


FIGURE 8 — ACCESS TIME EXTENSION USING $\overline{\text{STRETCH}}$ AND MRDY



Ordering Information

MC 68 28 P

Motorola Integrated Circuit
M6800 Family
Device Designations
Package Designation
P = Plastic
L = Ceramic

PACKAGE DIMENSIONS

Cerdip	Case 623-03	L Suffix
Plastic	Case 649-03	P Suffix