THOMSON COMPONENTS MOSTEK

TS68HC901

HCMOS MULTI FUNCTION PERIPHERAL

ADVANCE INFORMATION

The TS68HC901 muli-function peripheral (CMFP) is a member of the 68000 Family of peripherals and the CMOS version of the MK68901. The CMFP directly interfaces to the 68000 processor via an asynchronous bus structure and can also support both multiplexed and non multiplexed buses. Both vectored, non vectored and polled interrupt schemes are supported, with the CMFP providing unique vector number generation for each of its 16 interrupt sources. Additionally, handshake lines are provided to facilitate DMAC interfacing.

The TS68HC901 performs many of the functions common to most microprocessor-based systems. The resources available to the user include :

- Eight Individually Programmable I/O Pins with Interrupt Capability
- 16-Source Interrupt Controller with Individual Source Enabling and Masking
- Four Timers, Two of which are Multi-Mode Timers
- Timers may be used as Baud Rate Generators for the Serial Channel
- Single-Channel Full-Duplex Universal Synchronous/Asynchronous Receiver-Transmitter (USART) that Supports Asynchronous and with the Addition of a Polynominal Generator Checker Supports Byte Synchronous Formats.

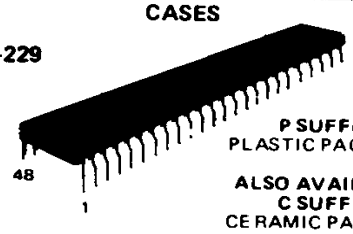
By incorporating multiple functions within the CMFP, the system designer retains flexibility while minimizing device count.
The CMOS technology used for the TS68HC901 reduces also the power consumption of the system.

HCMOS

MULTI-FUNCTION PERIPHERAL

CASES

CB-229



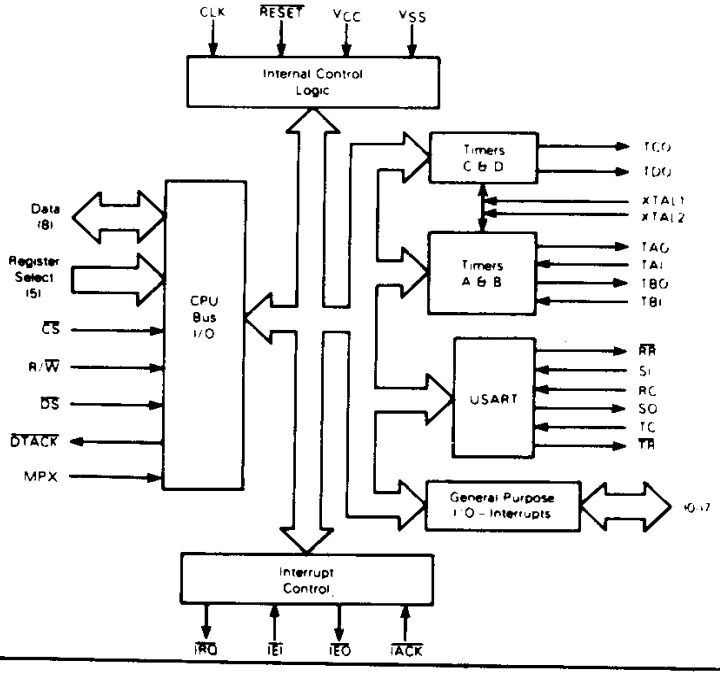
P SUFFIX
PLASTIC PACKAGE

ALSO AVAILABLE
C SUFFIX
CERAMIC PACKAGE

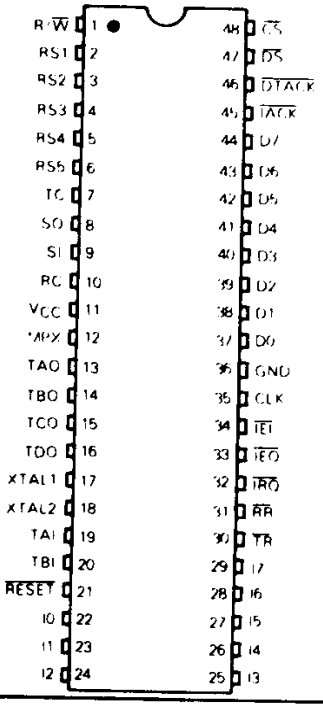
CB-522

FN SUFFIX
PLCC 52

BLOCK DIAGRAM



PIN ASSIGNMENT



SECTION 1
INTRODUCTION

The TS68HC901 multi-function peripheral (CMFP) is a member of the 68000 peripherals. The CMFP directly interfaces to the 68000 processor via an asynchronous bus structure. Both vectored and polled interrupt schemes are supported, with the CMFP providing unique vector number generation for each of its 16 interrupt sources. Additionally, handshake lines are provided to facilitate DMAC interfacing. Refer to block diagram of the TS68HC901.

The TS68HC901 performs many of the functions common to most microprocessor-based systems.

The resources available to the user include:

- Eight Individually Programmable I/O Pins with Interrupt Capability
- 16-Source Interrupt Controller with Individual Source Enabling and Masking
- Four Timers, Two of which are Multi-Mode Timers
- Timers May Be Used as Baud Rate Generators for the Serial Channel
- Single-Channel Full-Duplex Universal Synchronous/Asynchronous Receiver-Transmitter (USART) that Supports Asynchronous and with the Addition of a Polynomial Generator Checker Supports Byte Synchronous Formats

By incorporating multiple functions within the CMFP, the system designer retains flexibility while minimizing device count.

From a programmer's point of view, the versatility of the CMFP may be attributed to its register set. The registers are well organized and allow the CMFP to be easily tailored to a variety of applications. All of the 24 registers are also directly addressable which simplifies programming. The register map is shown in Table 1-1.

Table 1-1. CMFP Register Map

Address							
	Binary						
Hex	RS5	RS4	RS3	RS2	RS1	Abbreviation	Register Name
01	0	0	0	0	0	GPIP	General Purpose I/O Register
03	0	0	0	0	1	AER	Active Edge Register
05	0	0	0	1	0	DDR	Data Direction Register
07	0	0	0	1	1	IERA	Interrupt Enable Register A
09	0	0	1	0	0	IERB	Interrupt Enable Register B
0B	0	0	1	0	1	IPRA	Interrupt Pending Register A
0D	0	0	1	1	0	IPRB	Interrupt Pending Register B
0F	0	0	1	1	1	ISRA	Interrupt In-Service Register A
11	0	1	0	0	0	ISRB	Interrupt In-Service Register B
13	0	1	0	0	1	IMRA	Interrupt Mask Register A
15	0	1	0	1	0	IMRB	Interrupt Mask Register B
17	0	1	0	1	1	VR	Vector Register
19	0	1	1	0	0	TACR	Timer A Control Register
1B	0	1	1	0	1	TBCR	Timer B Control Register
1D	0	1	1	1	0	TCDCR	Timers C and D Control Register
1F	0	1	1	1	1	TADR	Timer A Data Register
21	1	0	0	0	0	TBDR	Timer B Data Register
23	1	0	0	0	1	TCDR	Timer C Data Register
25	1	0	0	1	0	TDDR	Timer D Data Register
27	1	0	0	1	1	SCR	Synchronous Character Register
29	1	0	1	0	0	UCR	USART Control Register
2B	1	0	1	0	1	RSR	Receiver Status Register
2D	1	0	1	1	0	TSR	Transmitter Status Register
2F	1	0	1	1	1	UDR	USART Data Register

NOTE : Hex addresses assume that RS1 connects with A1, RS2 connects with A2, etc... and that DS is connected to LDS on the 68000 or DS is connected to DS on the 68008.

SECTION 2
SIGNAL AND BUS OPERATION DESCRIPTION

This section contains a brief description of the input and output signals. A discussion of bus operation during the various operations is also presented.

Note: The terms **assertion** and **negation** will be used extensively. This is done to avoid confusion when dealing with a mixture of "active low" and "active high" signals. The term assert or assertion is used to indicate that a signal is active or true, independent of whether that level is represented by a high or low voltage. The term negate or negation is used to indicate that a signal is inactive or false.

2.1 SIGNAL DESCRIPTION

The input and output signals can be functionally organized into the groups shown in Figure 2-1. The following paragraphs provide a brief description of the signal and a reference (if applicable) to other sections that contain more detail about its function.

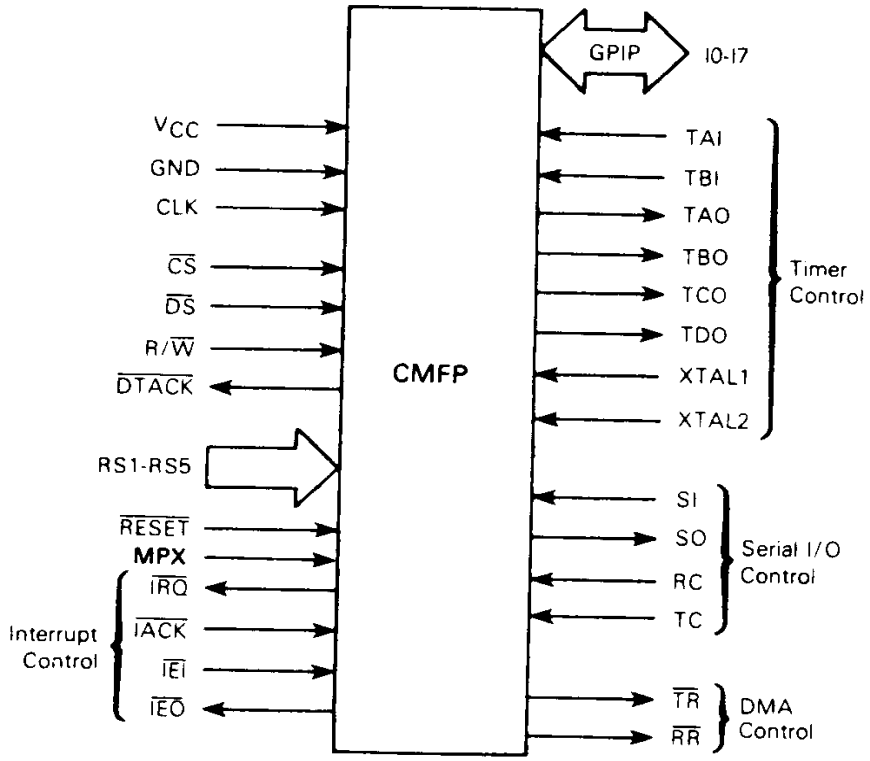


Figure 2-1. Input and Output Signals

2.1.1 V_{CC} and GND

These inputs supply power to the CMFP. The V_{CC} is power at + 5 volts and GND is the ground connection.

2.1.2 Clock (CLK)

The clock input is a single-phase TTL-compatible signal used for internal timing. This input should not be gated off at any time and must conform to minimum and maximum pulse width times. The clock is not necessarily the system clock in frequency nor phase. When the bus is multiplexed (MPX=1), an adress strobe signal is connected to this pin. In the non multiplexed mode (MPX=0), this input is connected to the system clock when used with a 68000 processor type or to V_{SS} (0 V_{DC}) when used with a 6800 processor type.

2.1.3 Asynchronous Bus Control

Asynchronous data transfers are controlled by chip select, data strobe, read/write, and data transfer acknowledge. The low order register select lines, RS1-RS5, select an internal CMFP register for a read or write operation. The reset line initializes the CMFP registers and the internal control signals.

2.1.3.1 CHIP SELECT ($\overline{CS}$). This input activates the CMFP for internal register access.

2.1.3.2 DATA STROBE ($\overline{DS}$). This input is part of the internal chip select and interrupt acknowledge functions. The CMFP must be located on the lower portion of the 16-bit data bus so that the vector number passed to the processor during an interrupt acknowledge cycle will be located in the low byte of the data word. As a result, $\overline{DS}$ must be connected to the processor's lower data strobe if vectored interrupts are to be used. Note that this forces all registers to be located at odd addresses and latches data on the rising edge for writes. This signal is used as $\overline{RD}$ with an Intel processor type.

2.1.3.3 READ/WRITE (R/$\overline{W}$). This input defines a data transfer as a read (high) or a write (low) cycle. This signal is used as $\overline{WR}$ with an Intel processor type.

2.1.3.4. DATA TRANSFER ACKNOWLEDGE ($\overline{DTACK}$). This output signals the completion of the operation phase of a bus cycle to the processor. If the bus cycle is a processor read, the CMFP asserts DTACK to indicate that the information on the data bus is valid. If the bus cycle is a processor to the CMFP, $\overline{DTACK}$ acknowledges the acceptance of the data by the CMFP. $\overline{DTACK}$ will be asserted only by an CMFP that has $\overline{CS}$ or $\overline{IACK}$ (and $\overline{IEI}$) asserted. This signal is not used with a 6800 processor type.

2.1.3.5 REGISTER SELECT BUS (RS1 THROUGH RS5). The lower five bits of the register select bus select an internal CMFP register during a read or write operation.

2.1.3.6 DATA BUS (D0 THROUGH D7). This bidirectional bus is used to receive data from or transmit data to the CMFP's internal registers during a processor read or write cycle. During an interrupt acknowledge cycle, the data bus is used to pass a vector number to the processor. Since the CMFP is an 8-bit peripheral, the CMFP could be located on either the upper or lower portion of the 16-bit data bus (even or odd address). However, during an interrupt acknowledge cycle, the vector number passed to the processor must be located in the low byte of the data word. As a result, D0-D7 of the CMFP must be connected to the low order eight bits of the processor data bus, placing CMFP registers at odd addresses if vectored interrupts are to be used.

2.1.3.7 RESET ($\overline{\text{RESET}}$). This input will initialize the CMFP during power up or in response to a total system reset. Refer to **2.2.3** for further information.

2.1.3.8 MPX. This signal selects the data bus mode:
MPX=0: non multiplexed mode
MPX=1: multiplexed mode. The register select lines RS1-RS5 and the data bus D0-D7 are multiplexed. An adress strobe must be connected to the CLK pin.

2.1.4 Interrupt Control

The interrupt request and interrupt acknowledge signals are handshake lines for a vectored interrupt scheme. Interrupt enable in and the interrupt enable out implement a daisy-chained interrupt structure.

2.1.4.1 INTERRUPT REQUEST ($\overline{\text{IRQ}}$). This output signals the processor that an interrupt is pending from the CMFP. These are 16 interrupt channels that can generate an interrupt request. Clearing the interrupt pending registers (IPRA and IPRB) or clearing the interrupt mask registers (IMRA and IMRB) will cause $\overline{\text{IRQ}}$ to be negated. $\overline{\text{IRQ}}$ will also be negated as the result of an interrupt acknowledge cycle, unless additional interrupts are pending in the CMFP. Refer to **SECTION 3** for further information.

2.1.4.2 INTERRUPT ACKNOWLEDGE ($\overline{\text{IACK}}$). If both $\overline{\text{IRQ}}$ and $\overline{\text{IEI}}$ are active, the CMFP will begin an interrupt acknowledge cycle when $\overline{\text{IACK}}$ and $\overline{\text{DS}}$ are asserted. The CMFP will supply a unique vector number to the processor which corresponds to the interrupt handler for the particular channel requiring interrupt service. In a daisy-chained interrupt structure, all devices in the chain must have a common $\overline{\text{IACK}}$. Refer to **2.2.2** and **3.1.2** for additional information.

2.1.4.3 INTERRUPT ENABLE IN ($\overline{\text{IEI}}$). This input, together with the $\overline{\text{IEO}}$ signal, provides a daisy-chained interrupt structure for a vectored interrupt scheme. $\overline{\text{IEI}}$ indicates that no higher priority device is requesting interrupt service. So, the highest priority device in the chain should have its $\overline{\text{IEI}}$ pin tied low. During an interrupt acknowledge cycle, an CMFP with a pending interrupt is not allowed to pass a vector number to the processor until its $\overline{\text{IEI}}$ pin is asserted. When the daisy-chain option is not implemented, all CMFPs should have their $\overline{\text{IEI}}$ pin tied low. Refer to **3.2** for additional information.

2.1.4.4 INTERRUPT ENABLE OUT ($\overline{IEO}$). This output, together with the $\overline{IEI}$ signal, provides a daisy-chained interrupt structure for a vectored interrupt scheme. The $\overline{IEO}$ of a particular CMFP signals lower priority devices that neither the CMFP nor any other higher-priority device is requesting interrupt service. When a daisy-chain is implemented, $\overline{IEO}$ is tied to the next lower priority device's $\overline{IEI}$ input. The lowest priority device's $\overline{IEO}$ is not connected. When the daisy-chain option is not implemented, $\overline{IEO}$ is not connected. Refer to **3.2** for additional information.

2.1.5 General Purpose I/O Interrupt Lines (I0 Through I7)

This is an 8-bit pin-programmable I/O port with interrupt capability. The data direction register (DDR) individually defines each line as either a high-impedance input or a TTL-compatible output. As an input, each line can generate an interrupt on the user selected transition of the input signal. Refer to **SECTION 4** for further information.

2.1.6 Timer Control

These lines provide internal timing and auxiliary timer control inputs required for certain operating modes. Additionally, the timer outputs are included in this group.

2.1.6.1 TIMER CLOCK (XTAL1 AND XTAL2). This input provides the timing signal for the four timers. A crystal can be connected between the timer clock inputs, XTAL1 and XTAL2, or XTAL1 can be driven with a TTL-level clock while XTAL2 is not connected. The following crystal parameters are suggested:

a) Parallel resonance, fundamental mode AT-cut

b) Frequency tolerance measured with 18 picofarads load (0.1% accuracy) — drive level 10 microwatts

c) Shunt capacitance equals 7 picofarads maximum

d) Series resistance:
   $2.0 < f < 2.7$ MHz; $R_S \leq 300\ \Omega$
   $2.8 < f < 4.0$ MHz; $R_S \leq 150\ \Omega$

2.1.6.2 TIMER INPUTS (TAI AND TBI). These inputs are control signals for timers A and B in the pulse width measurement mode and event count mode. These signals generate interrupts at the same priority level as the general purpose I/O interrupt lines I4 and I3, respectively. While I4 and I3 do not have interrupt capability when the timers are operated in the pulse width measurement mode or the event count mode, I4 and I3 may still be used for I/O. Refer to **5.1.2** and **5.1.3** for further information.

2.1.6.3 TIMER OUTPUTS (TAO, TBO, TCO, AND TDO). Each timer has an associated output which toggles when its main counter counts through 01 (hexadecimal), regardless of which operational mode is selected. When in the delay mode, the timer output will be a square wave with a period equal to two timer cycles. This output signal may be used to supply the universal synchronous/asynchronous receiver-transmitter (USART) baud rate clocks. Timer outputs TAO and TBO may be cleared at any time by writing a one to the reset location in timer control registers A and B. Also, a device reset forces all timer outputs low. Refer to **5.2.2** for additional information.

2.1.7 Serial I/O Control

The full duplex serial channel is implemented by a serial input and output line. The independent receive and transmit sections may be clocked by separate timing signals on the receiver clock input and the transmitter clock input.

2.1.7.1 SERIAL INPUT (SI). This input line is the USART receiver data input. This input is not used in the USART loopback mode. **Refer to 6.3.2 for additional information.**

2.1.7.2 SERIAL OUTPUT (SO). This output line is the USART transmitter data output. This output is driven high during a device reset.

2.1.7.3 RECEIVER CLOCK (RC). This input controls the serial bit rate of the receiver. This signal may be supplied by the timer output lines or by any external TTL-level clock which meets the minimum and maximum cycle times. This clock is not used in the USART loopback mode. Refer to 6.3.2 for additional information.

2.1.7.4 TRANSMITTER CLOCK (TC). This input controls the serial bit rate of the transmitter. This signal may be supplied by the timer output lines or by an external TTL-level clock which meets the minimum and maximum cycle times.

2.1.8 DMA Control

The USART supports DMA transfers through its receiver ready and transmitter ready status lines.

2.1.8.1 RECEIVER READY ($\overline{RR}$). This output reflects the receiver buffer full status for DMA operations.

2.1.8.2 TRANSMITTER READY ($\overline{TR}$). This output reflects the transmitter buffer empty status for DMA operations.

2.1.9 Signal Summary

Table 2-1 is a summary of all the signals discussed in the previous paragraphs.

Table 2-1. Signal Summary

Signal Name	Mnemonic	I/O	Active
Power Input	V_{CC}	Input	High
Ground	GND	Input	Low
Clock	CLK	Input	N/A
Chip Select	$\overline{CS}$	Input	Low
Data strobe	$\overline{DS}$	Input	Low
Read/Write	R/$\overline{W}$	Input	Read – High, Write – Low
Data Transfer Acknowledge	$\overline{DTACK}$	Output	Low
Register Select Bus	RS1-RS5	Input	N/A
Data Bus	D0-D7	I/O	N/A
Reset	$\overline{RESET}$	Input	Low
Interrupt Request	$\overline{IRQ}$	Output	Low
Interrupt Acknowledge	$\overline{IACK}$	Input	Low
Interrupt Enable In	$\overline{IEI}$	Input	Low
Interrupt Enable Out	$\overline{IEO}$	Output	Low
General Purpose I/O – Interrupt Lines	I0-I7	I/O	N/A
Timer Clock	XTAL1, XTAL2	Input	High
Timer Inputs	TAI, TBI	Input	N/A
Timer Outputs	TAO,TBO, TCO, TDO	Output	N/A
Serial Input	SI	Input	N/A
Serial Output	SO	Output	N/A
Receiver Clock	RC	Input	N/A
Transmitter Clock	TC	Input	N/A
Receiver Ready	$\overline{RR}$	Output	Low
Transmitter Ready	$\overline{TR}$	Output	Low
MPX	MPX	Input	N/A

2.2 BUS OPERATION

The following paragraphs explain the control signals and bus operation during data transfer operations and reset.

2.2.1 Data Transfer Operations

Transfer of data between devices involves the following pins:

Register Select Bus – RS1 through RS5

Data Bus – D0 through D7

Control Signals

The address and data buses are separate parallel buses used to transfer data using an asynchronous bus structure. In all cycles, the bus master assumes responsibility for deskewing all signals it issues at both the start and end of a cycle. Additionally, the bus master is responsible for deskewing the acknowledge and data signals from the peripheral devices.

2.2.1.1 READ CYCLE. To read an CMFP register, $\overline{CS}$ and $\overline{DS}$ must be asserted, and R/$\overline{W}$ must be high. The CMFP will place the contents of the register which is selected by the register select bus (RS1 through RS5) on the data bus (D0 through D7) and then assert $\overline{DTACK}$. The register addresses are shown in Table 1-1.

After the processor has latched the data, $\overline{DS}$ is negated. The negation of either $\overline{CS}$ or $\overline{DS}$ will terminate the read operation. The CMFP will drive $\overline{DTACK}$ high and place it in the high-impedance state. Also, the data bus will be in the high-impedance state. The timing for a read cycle is shown in Figure 2-2. Refer to **7.7** for actual timing numbers.

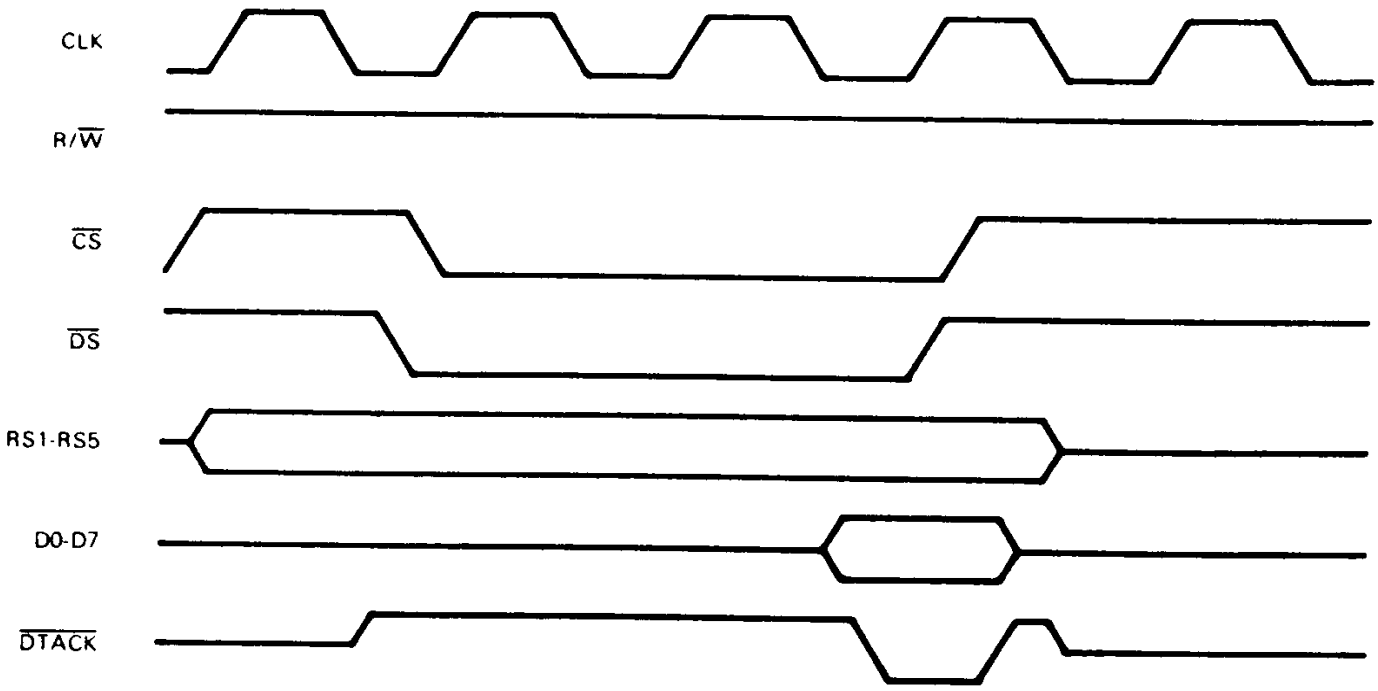


Figure 2-2. Read Cycle Timing

2.2.1.2 WRITE CYCLE. To write a register, $\overline{CS}$ and $\overline{DS}$ must be asserted, and R/$\overline{W}$ must be low. The CMFP will decode the address bus to determine which register is selected (the register map is shown in Table 1-1). Then the register will be loaded with the contents of the data bus and $\overline{DTACK}$ will be asserted.

When the processor recognizes $\overline{DTACK}$, $\overline{DS}$ will be negated. The write cycle is terminated when either $\overline{CS}$ or $\overline{DS}$ is negated. The CMFP will drive $\overline{DTACK}$ high and place it in the high-impedance state. The timing for a write cycle is shown in Figure 2-3. Refer to **7.7** for actual numbers.

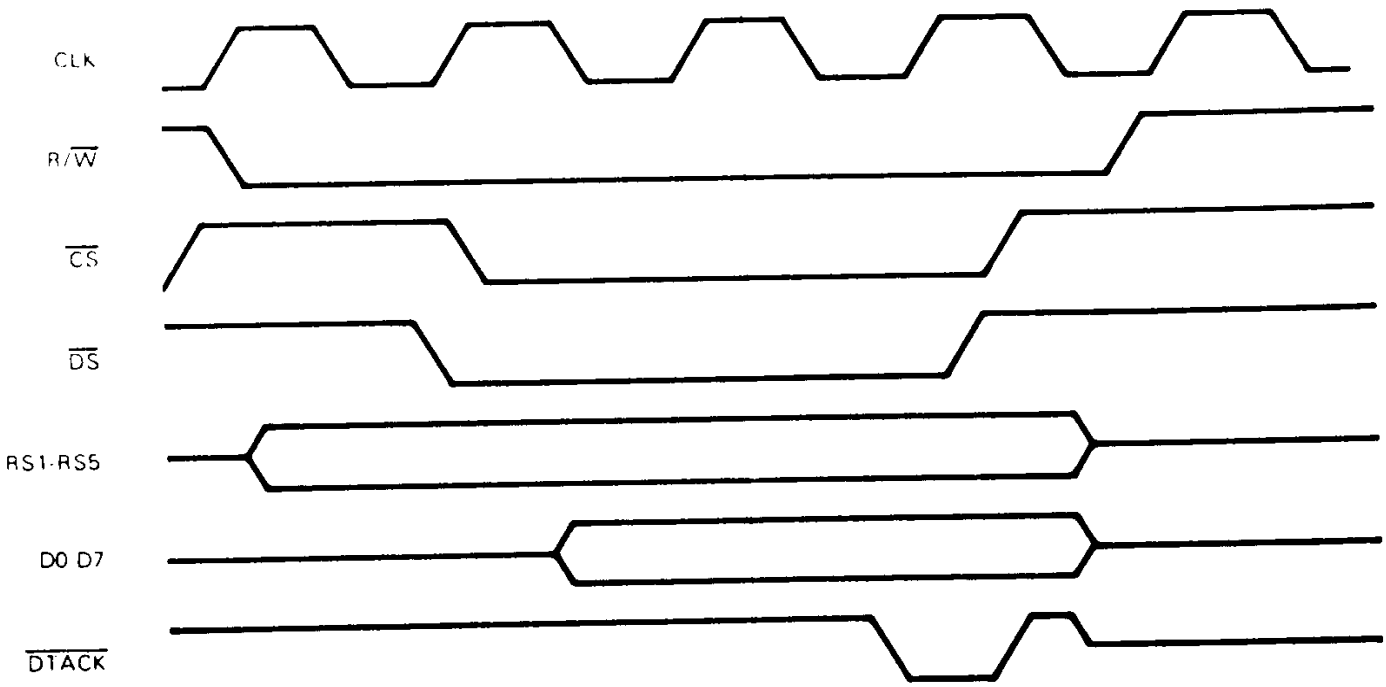


Figure 2-3. Write Cycle Timing

2.2.2 Interrupt Acknowledge Operation

The CMFP has 16 interrupt sources, eight internal sources, and eight external sources. When an interruptrequest is pending, the CMFP will assert $\overline{IRQ}$. In a vectored interrupt scheme, the processor will acknowledge the interrupt request by performing an interrupt acknowledge cycle. $\overline{IACK}$ and $\overline{DS}$ will be asserted. The CMFP responds to the $\overline{IACK}$ signal by placing a vector number on the lower eight bits of the data bus. This vector number corresponds to the $\overline{IRQ}$ handler for the particular interrupt requesting service. The format of this vector number is given in Figure 3-1.

When the CMFP asserts $\overline{DTACK}$ to indicate that valid data is on the bus, the processor will latch the data and terminate the bus cycle by negating $\overline{DS}$. When either $\overline{DS}$ or $\overline{IACK}$ are negated, the CMFP will terminate the interrupt acknowledge operation by driving $\overline{DTACK}$ high and placing it in the high-impedance state. Also, the data bus will be placed in the high-impedance state. $\overline{IRQ}$ will be negated as a result of the $\overline{IACK}$ cycle unless additional interrupts are pending.

The CMFP can be part of a daisy-chain interrupt structure which allows multiple CMFPs to be placed at the same interrupt level by sharing a common $\overline{IACK}$ signal. A daisy-chain priority scheme is implemented with signals $\overline{IEI}$ and $\overline{IEO}$. $\overline{IEI}$ indicates that no higher priority device is requesting interrupt service. $\overline{IEO}$ signals lower priority devices that neither this device nor any higher priority device is requesting service. To daisy-chain CMFPs, the highest priority CMFP has its $\overline{IEI}$ tied low and successive CMFPs have their $\overline{IEI}$ connected to the next higher priority device's $\overline{IEO}$. Note that when the daisy-chain interrupt structure is not implemented, the $\overline{IEI}$ of all CMFPs must be tied low. Refer to **3.2** for additional information.

When the processor initiates an interrupt acknowledge cycle by driving $\overline{IACK}$ and $\overline{DS}$, the CMFP whose $\overline{IEI}$ is low may respond with a vector number if an interrupt is pending. If this device does not have a pending interrupt, $\overline{IEO}$ is asserted which allows the next lower priority device to respond to the interrupt acknowledge. When an CMFP propagates $\overline{IEO}$, it will not drive the data bus nor $\overline{DTACK}$ during the interrupt acknowledge cycle. The timing for an $\overline{IACK}$ cycle is shown in Figure 2-4. Refer to **7.6** for further information.

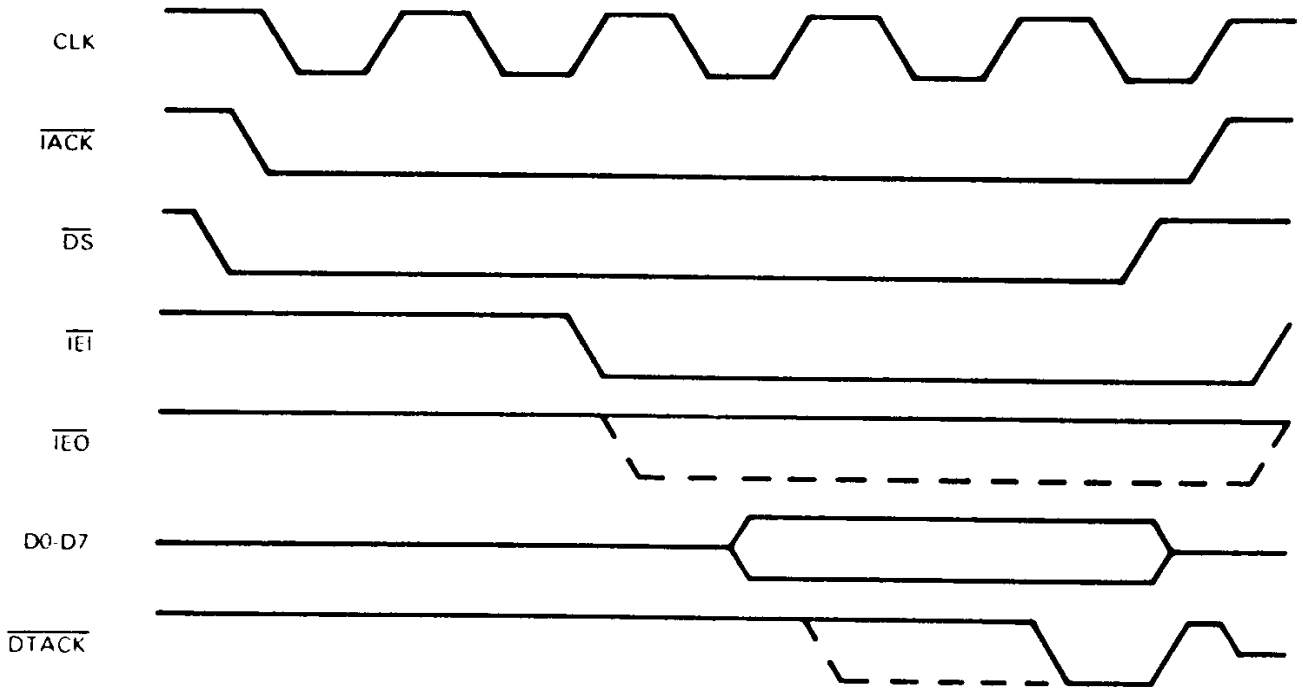


Figure 2-4. $\overline{IACK}$ Cycle Timing

2.2.3 Reset Operation

The reset operation will initialize the CMFP to a known state. The reset operation requires that the $\overline{RESET}$ input be asserted for a minimum of two microseconds. During a device reset condition, all internal CMFP registers are cleared except for the timer data registers (TADR, TBDR, TCDR, and TDDR), the USART data register (UDR), the transmitter status register (TSR) and the interrupt vector register. All timers are stopped and the USART receiver and transmitter are disabled. The interrupt channels are also disabled and any pending interrupts are cleared. In addition, the general purpose interrupt I/O lines are placed in the high-impedance input mode and the timer outputs are driven low. External CMFP signals are negated. The interrupt vector register is initialized to a $0F.

2.2.4 Non Multiplexed mode

In this mode the MPX input must be set to zero, and the TS68HC901 can be used with a 68000 processor type or a 6800 processor type. Refer to figure 7-4, 7-5, 7-8 for the electrical characteristics.

With a 6800 processor type the $\overline{DS}$ pin is connected to the E signal of the processor, the $\overline{DTACK}$ signal is not used and the CLK must be zeroed.

2.2.5 Multiplexed mode

The CMFP can be used either on a MOTOROLA or INTEL bus type. In this case the MPX pin is connected to V_{CC}. The following table gives the signification of the different signals used. A dummy access to the TS68HC901 has to be done before any valid access in order to set up the internal logic of sampling.

Pin	MOTOROLA 6800 type	MOTOROLA Multiplexed	INTEL
48	$\overline{CS}$	$\overline{CS}$	$\overline{CS}$
47	E	$\overline{DS}$	$\overline{RD}$
1	R/$\overline{W}$	R/$\overline{W}$	$\overline{WR}$
35	V_{SS}	AS	ALE

SECTION 3
INTERRUPT STRUCTURE

In a 68000 system, the CMFP will be assigned to one of the seven possible interrupt levels. All interrupt service requests from the CMFP's 16 interrupt channels will be presented at this level. Although, as an interrupt controller, the CMFP will internally prioritize its 16 interrupt sources. Additional interrupt sources may be placed at the same interrupt level by daisy-chaining multiple CMFPs. The CMFPs will be prioritized by their position in the chain.

3.1 INTERRUPT PROCESSING

Each CMFP provides individual interrupt capability for its various functions. When an interrupt is received on one of the external interrupt channels or from one of the eight internal sources, the CMFP will request interrupt service. The 16 interrupt channels are assigned a fixed priority so that multiple pending interrupts are serviced according to their relative importance. Since the CMFP can internally generate 16 vector numbers, the unique vector number which corresponds to the highest priority channel that has a pending interrupt is presented to the processor during an interrupt acknowledge cycle. This unique vector number allows the processor to immediately begin execution of the interrupt handler for the interrupting source, decreasing interrupt latency time.

3.1.1 Interrupt Channel Prioritization

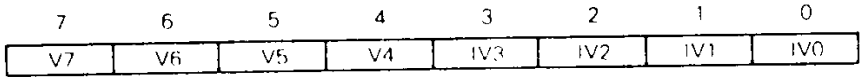
The 16 interrupt channels are prioritized as shown in Table 3-1. General purpose interrupt 7 (I7) is the highest priority interrupt channel and I0 is the lowest priority channel. Pending interrupts are presented to the CPU in order of priority unless they have been masked off. By selectively masking interrupts, the channels are in effect re-prioritized.

Table 3-1. Interrupt Channel Prioritization

Priority	Channel	Description
Highest	1111	General Purpose Interrupt 7 (I7)
	1110	General Purpose Interrupt 6 (I6)
	1101	Timer A
	1100	Receiver Buffer Full
	1011	Receive Error
	1010	Transmit Buffer Empty
	1001	Transmit Error
	1000	Timer B
	0111	General Purpose Interrupt 5 (I5)
	0110	General Purpose Interrupt 4 (I4)
	0101	Timer C
	0100	Timer D
	0011	General Purpose Interrupt 3 (I3)
	0010	General Purpose Interrupt 2 (I2)
	0001	General Purpose Interrupt 1 (I1)
Lowest	0000	General Purpose Interrupt 0 (I0)

3.1.2 Interrupt Vector Number Format

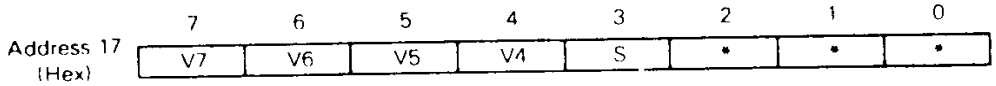
During an interrupt acknowledge cycle, a unique 8-bit vector number is presented to the system which corresponds to the specific interrupt source which is requesting service. The format of the vector is shown in Figure 3-1. The most significant four bits of the interrupt vector number are user programmable. These bits are set by writing the upper four bits of the vector register which is shown in Figure 3-2. The low order bits are generated internally by the TS68HC901. Note that the binary channel number shown in Table 3-1 corresponds to the low order bits of the vector number associated with each channel.



V7-V4 The four most significant bits are copied from the vector register

IV3-IV0 These bits are supplied by the CMFP. They are the binary channel number of the highest priority channel that is requesting interrupt service

Figure 3-1. Interrupt Vector Format



*Unused bits are read as zero

V7-V4 The upper four bits of the vector register are written by the user. These bits become the most significant four bits of the interrupt vector number.

SET a) MPU writes a one
CLEARED a) MPU writes a zero
b) Reset

S In-Service Register Enable. When the S bit is zero, the CMFP is in the automatic end-of-interrupt mode and the in-service register bits are forced low. When the S bit is a one, the CMFP is in the software end-of-interrupt mode and the in-Service register bits are enabled. Refer to **3.4.2** and **3.4.3** for additional information.

tion

SET a) MPU writes a one
CLEARED a) MPU writes a zero
b) Reset

Figure 3-2 Vector Register Format (VR)

3.2 DAISY-CHAINING CMFPs

As an interrupt controller, the TS68HC901 CMFP will support eight external interrupt sources in addition to its eight internal interrupt sources. When a system requires more than eight external interrupt sources to be placed at the same interrupt level, sources may be added to the prioritized structure by daisy-chaining CMFPs. Interrupt sources are prioritized internally within each CMFP and the CMFPs are prioritized by their position in the chain. Unique vector numbers are provided for each interrupt source.

The $\overline{IEI}$ and $\overline{IEO}$ signals implement the daisy-chained interrupt structure. The $\overline{IEI}$ of the highest priority CMFP is tied low and the $\overline{IEO}$ output of this device is tied to the next highest priority CMFP's $\overline{IEI}$. The $\overline{IEI}$ and $\overline{IEO}$ signals are daisy-chained in this manner for all CMFPs in the chain, with the lowest priority CMFP's $\overline{IEO}$ left unconnected. A diagram of an interrupt daisy-chain is shown in Figure 3-3.

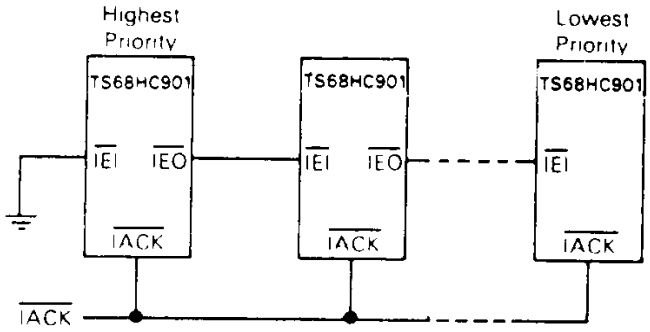


Figure 3-3. Daisy-Chained Interrupt Structure

Daisy-chaining requires that all parts in the chain have a common $\overline{IACK}$. When the common $\overline{IACK}$ is asserted during an interrupt acknowledge cycle, all parts will prioritize interrupts in parallel. When the $\overline{IEI}$ signal to a CMFP is asserted, the part may respond to the $\overline{IACK}$ cycle if it requires interrupt service. Otherwise, the part will assert $\overline{IEO}$ to the next lower priority device. Thus, priority is passed down the chain via $\overline{IEI}$ and $\overline{IEO}$ until a part which has a pending interrupt is reached. The part with the pending interrupt passes a vector number to the processor and does not propagate $\overline{IEO}$.

3.3 INTERRUPT CONTROL REGISTERS

CMFP interrupt processing is managed by the interrupt enable registers A and B, interrupt pending registers A and B, and interrupt mask registers A and B. These registers allow the programmer to enable or disable individual interrupt channels, mask individual interrupt channels, and access pending interrupt status information. In-service registers A and B allow interrupts to be nested as described in **3.4**. The interrupt control registers are shown in Figure 3-4.

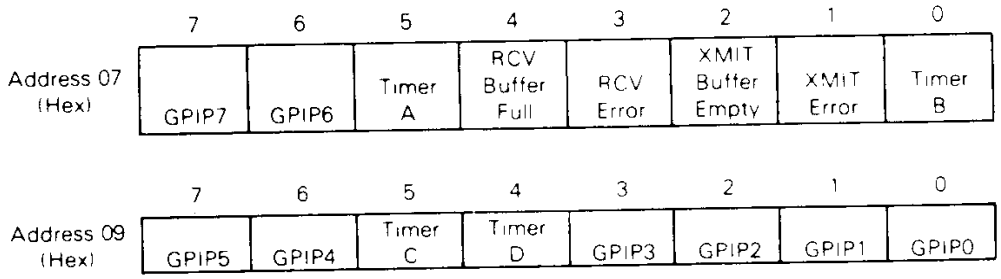
3.3.1 Interrupt Enable Registers

The interrupt channels are individually enabled or disabled by writing a one or zero, respectively, to the appropriate bit of interrupt enable register A (IERA) or interrupt enable register B (IERB). The processor may read these registers at any time.

When a channel is enabled, interrupts received on the channel will be recognized by the CMFP and $\overline{IRQ}$ will be asserted to the processor, indicating that interrupt service is required. On the other hand, a disabled channel is completely inactive; interrupts received on the channel are ignored by the CMFP.

Writing a zero to a bit of interrupt enable register A or B will cause the corresponding bit of interrupt pending register A or B to be cleared. This will terminate all interrupt service requests for the channel and also negate $\overline{IRQ}$, unless interrupts are pending from other sources. Disabling a channel, however, does not affect the corresponding bit in interrupt in-service registers A or B. So, if the CMFP is in the software end-of-interrupt mode (see **3.4.3**) and an interrupt is in service when a channel is disabled, the in-service status bit for that channel will remain set until cleared by software.

(a) Interrupt Enable Registers (IERA and IERB)

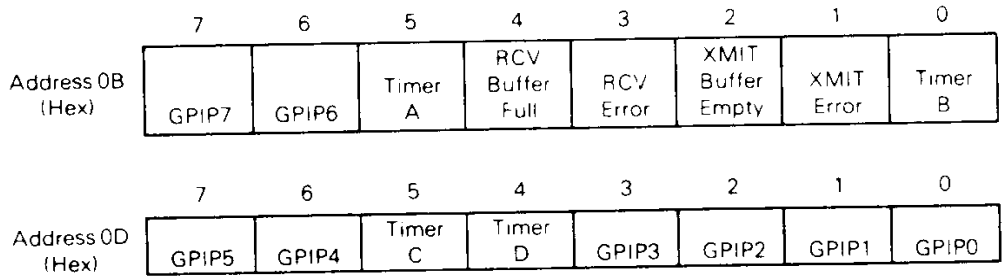


When a bit is a zero, the associated interrupt channel is disabled. When a bit is a one, the associated interrupt channel is enabled.

SET a) MPU writes a one

CLEARED a) MPU writes a zero
b) Reset

(b) Interrupt Pending Registers (IPRA and IPRB)

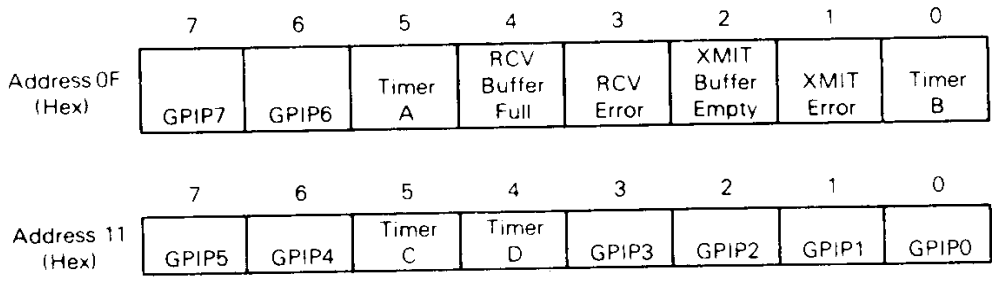


When a bit is a zero, no interrupt is pending on the associated interrupt channel. When a bit is a one, an interrupt is pending on the associated interrupt channel

SET a) Interrupt is received on an enabled interrupt channel

CLEARED a) Interrupt vector for the associated interrupt channel is passed during an $\overline{IACK}$ cycle
b) Associated interrupt channel is disabled
c) MPU writes a zero
d) Reset

(c) Interrupt In-Service Registers (ISRA and ISRB)



When a bit is a zero, no interrupt processing is in progress for the associated interrupt channel. When a bit is a one, interrupt processing is in progress for the associated interrupt channel.

SET a) Interrupt vector number for the associated interrupt channel is passed during an $\overline{IACK}$ cycle and the S bit of the vector register is set.

CLEARED a) Interrupt service is completed for the associated interrupt channel
b) The S bit of the vector register is a zero.
c) MPU writes a zero
d) Reset

Figure 3-4. Interrupt Control Registers (Sheet 1 of 2)

(d) Interrupt Mask Registers (IMRA and IMRB)

	7	6	5	4	3	2	1	0
Address 13 (Hex)	GPIP7	GPIP6	Timer A	RCV Buffer Full	RCV Error	XMIT Buffer Empty	XMIT Error	Timer B
Address 15 (Hex)	GPIP5	GPIP4	Timer C	Timer D	GPIP3	GPIP2	GPIP1	GPIP0

When a bit is zero, interrupts are masked for the associated interrupt channel. When a bit is a one, interrupts are not masked for the associated interrupt channel.

SET a) MPU writes a one

CLEARED a) MPU writes a zero
b) Reset

Figure 3-4. Interrupt Control Registers (Sheet 2 of 2)

3.3.2 Interrupt Pending Registers

When an interrupt is received on an enabled channel, the corresponding interrupt pending bit is set in interrupt pending register A or B (IPRA or IPRB). In a vectored interrupt scheme, this bit will be cleared when the processor acknowledges the interrupting channel and the CMFP responds with a vector number. In a polled interrupt system, the interrupt pending registers must be read to determine the interrupting channel and then the interrupt pending bit is cleared by the interrupt handling routine without performing an interrupt acknowledge sequence.

A single bit of the interrupt pending registers is cleared in software by writing ones to all bit positions except the bit to be cleared. Note that writing ones to IPRA and IPRB has no effect on the contents of the register. A single bit of the interrupt pending registers is also cleared when the corresponding channel is disabled by writing a zero to the appropriate bit of IERA or IERB.

3.3.3 Interrupt Mask Registers

Interrupts are masked for a channel by clearing the appropriate bit in interrupt mask register A or B (IMRA or IMRB). Even though an enabled channel is masked, the channel will recognize subsequent interrupts and set its interrupt pending bit. However, the channel is prevented from requesting interrupt service ($\overline{IRQ}$ to the processor) as long as the mask bit for that channel is cleared.

If a channel is requesting interrupt service at the time that its corresponding bit in IMRA or IMRB is cleared, the request will cease and $\overline{IRQ}$ will be negated, unless another channel is requesting interrupt service. Later, when the mask bit is set, any pending interrupt on the channel will be processed according to the channel's assigned priority. IMRA and IMRB may be read at any time.

3.4 NESTING CMFP INTERRUPTS

In a 68000 vectored interrupt system, the CMFP is assigned to one of seven possible interrupt levels. When an interrupt is received from the CMFP, an interrupt acknowledge for that level is initiated. Once an interrupt is recognized at a particular level, interrupts at that same level or

below are masked by 68000. As long as the processor's interrupt mask is unchanged, the 68000 interrupt structure will prohibit the nesting of interrupts at the same interrupt level. However, additional interrupt requests from the CMFP can be recognized before a previous channel's interrupt service routine is completed by lowering the processor's interrupt mask to the next lower interrupt level within the interrupt handler.

When nesting CMFP interrupts, it may be desirable to permit interrupts on any CMFP channel, regardless of its priority, to preempt or delay interrupt processing of an earlier channel's interrupt service request. Or, it may be desirable to only allow subsequent higher priority channel interrupt requests to supercede previously recognized lower priority interrupt requests. The CMFP interrupt structure provides this flexibility by offering two end-of-interrupt options for vectored interrupt schemes. Note that the end-of-interrupt modes are not active in a polled interrupt scheme.

3.4.1 Selecting The End-Of-Interrupt Mode

In a vectored interrupt scheme, the CMFP may be programmed to operate in either the automatic end-of-interrupt mode or the software end-of-interrupt mode. The mode is selected by writing the S bit of the vector register (see Figure 3-2). When the S bit is programmed to a one, the CMFP is placed in the software end-of-structure mode and when the S bit is a zero, all channels operate in the automatic end-of-interrupt mode.

3.4.2 Automatic End-Of-Interrupt

When an interrupt vector number is passed to the processor during an interrupt acknowledge cycle, the corresponding channel's interrupt pending bit is cleared. In the automatic end-of-interrupt mode, no further history of the interrupt remains in the CMFP. The in-service bits of the interrupt in-service registers (ISRA and ISRB) are forced low. Subsequent interrupts which are received on any CMFP channel will generate an interrupt request to the processor, even if the current interrupt's service routine has not been completed.

3.4.3 Software End-Of-Interrupt

In the software end-of-interrupt mode, the channel's associated interrupt pending bit is cleared and in addition, the channel's in-service bit of in-service register A or B is set when its vector number is passed to the processor during an $\overline{IACK}$ cycle. A higher priority channel may subsequently request interrupt service and be acknowledged, but as long as the channel's in-service bit is set, no lower priority channel may request interrupt service nor pass its vector during an interrupt acknowledge sequence.

While only higher priority channels may request interrupt service, any channel can receive an interrupt and set its interrupt pending bit. Even the channel whose in-service bit is set can receive a second interrupt. However, no interrupt service request is made until its in-service bit is cleared.

The in-service bit for a particular channel can be cleared by writing a zero to its corresponding bit in ISRA or ISRB and ones to all other bit positions. Since bits in the in-service registers can only be cleared in software and not set, writing ones to the registers does not alter their contents. ISRA and ISRB may be read at any time.

SECTION 4
GENERAL PURPOSE INPUT/OUTPUT INTERRUPT PORT

The general purpose interrupt input/output (I/O) port (GPIP) provides eight I/O lines (I0 through I7) that may be operated as either inputs or outputs under software control. In addition, these lines may optionally generate an interrupt on either a positive transition or a negative transition of the input signal. The flexibility of the GPIP allows it to be configured as an 8-bit I/O port or for bit I/O. Since interrupts are enabled on a bit-by-bit basis, a subset of the GPIP could be programmed as handshake lines or the port could be connected to as many as eight external interrupt sources, which would be prioritized by the CMFP interrupt controller for interrupt service.

4.1 6800 INTERRUPT CONTROLLER

The CMFP interrupt controller is particularly useful in a system which has many 6800-type devices. Typically, in a vectored 68000 system, 6800-type peripherals use the autovector which corresponds to their assigned interrupt level since they do not provide a vector number in response to an $\overline{IACK}$ cycle. The autovector interrupt handler must then poll all 6800-type devices at that interrupt level to determine which device is requesting service. However, by tying the $\overline{IRQ}$ output from a 6800-type device to the general purpose I/O interrupt port (GPIP) of a CMFP, a unique vector number will be provided to the processor during an interrupt acknowledge cycle. This interrupt structure will significantly reduce interrupt latency for 6800-type devices and other peripheral devices which do not support vector-by-device.

4.2 GPIP CONTROL REGISTERS

The GPIP is programmed via three control registers shown in Figure 4-1. These registers control the data direction, provide user access to the port, and specify the active edge for each bit of the GPIP which will produce an interrupt. These registers are described in detail in the following paragraphs.

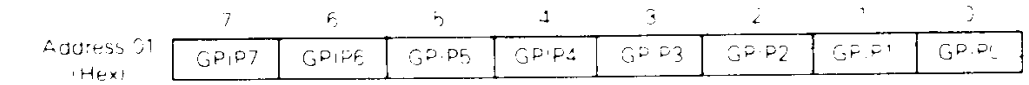
4.2.1 GPIP Data Register

The general purpose I/O data register is used to input or output data to the port. When data is written to the GPIP data register, those pins which are defined as inputs will remain in the high-impedance state. Pins which are defined as outputs will assume the state (high or low) of their corresponding bit in the data register. When the GPIP is read, data will be passed directly from the bits of the data register for pins which are defined as outputs. Data from pins defined as inputs will come from the input buffers.

4.2.2 Active Edge Register

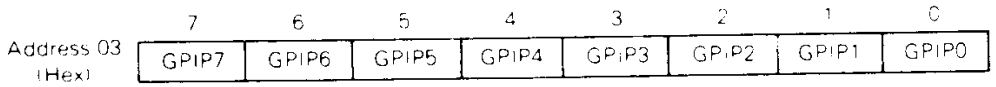
The active edge register (AER) allows each of the GPIP lines to produce an interrupt on either a one-to-zero or a zero-to-one transition. Writing a zero to the appropriate edge bit of the active edge

(a) GPIP Data Register (GPIP)



SET a) MPU writes a one
CLEARED a) MPU writes a zero

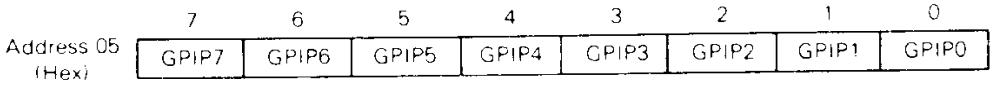
(b) Active Edge Register (AER)



When a bit is a zero, interrupts will be generated on the falling edge of the associated input signal. When a bit is a one, interrupts will be generated on the rising edge of the associated input signal.

SET a) MPU writes a one
CLEARED a) MPU writes a zero
b) Reset

(c) Data Direction Register (DDR)



When a bit is a zero, the associated I/O line is defined to be an input. When a bit is a one, the associated I/O line is defined to be an output.

SET a) MPU writes a one
CLEARED a) MPU writes a zero
b) Reset

Figure 4-1. GPIP Control Registers

register causes the associated input to generate an interrupt on the one-to-zero transition. Writing a one to the edge bit will produce an interrupt on the zero-to-one transition of the corresponding GPIP line.

Note: The transition detector is an exclusive-OR gate whose inputs are the edge bit and the input buffer. As a result, writing the AER may cause an interrupt-producing transition, depending upon the state of the input. So, the AER should be configured before enabling interrupts via the interrupt enable registers (IERA and IERB). Also, changing the edge bit while interrupts are enabled may cause an interrupt on the corresponding channel.

4.2.3 Data Direction Register

The data direction register (DDR) allows the programmer to define I0 through I7 as inputs or outputs by writing the corresponding bit. When a bit of the data direction register is written as a zero, the corresponding interrupt I/O pin will be a high-impedance input. Writing a one to any bit of the data direction register will cause the corresponding pin to be configured as a push-pull output.

SECTION 5
TIMERS

The CMFP contains four 8-bit timers which provide many functions typically required in microprocessor systems. The timers can supply the baud rate clocks for the on-chip serial I/O channel, generate periodic interrupts, measure elapsed time, and count signal transitions. In addition, two timers have waveform generation capability.

All timers are prescaler/counter timers with a common independent clock input (XTAL1 or XTAL2) and are not required to be operated from the system clock. Each timer's output signal toggles when the timer's main counter times out. Additionally, timers A and B have auxiliary control signals which are used in two of the operation modes. An interrupt channel is assigned to each timer and when the auxiliary control signals are used, a separate interrupt channel will respond to transitions on these inputs.

5.1 OPERATION MODES

Timers A and B are full function timers which, in addition to the delay mode, operate in the pulse width measurement mode and the event count mode. Timers C and D are delay timers only. A brief discussion of each of the timer modes follows.

5.1.1 Delay Mode Operation

All timers may operate in the delay mode. In this mode, the prescaler is always active. The prescaler specifies the number of timer clock cycles which must elapse before a count pulse is applied to the main counter. A count pulse causes the main counter to decrement by one. When the timer has decremented down to 01 (hexadecimal), the next count pulse will cause the main counter to be reloaded from the timer data register and a time out pulse will be produced. This time out pulse is coupled to the timer's interrupt channel and, if the channel is enabled, an interrupt will occur. The time out pulse also causes the timer output pin to toggle. The output will remain in this new state until the next time out pulse occurs.

For example, if delay mode with a divide-by-10 prescaler is selected and the timer data register is loaded with 100 (decimal), the main counter will decrement once every 10 timer clock cycles. After 1,000 timer clocks, a time out pulse will be produced. This time out pulse will generate an interrupt if the channel is enabled (IERA, IERB) and in addition, the timer's output line will toggle. The output line will complete one full period every 2,000 cycles of the timer clock.

If the prescaler value is changed while the timer is enabled, the first time out pulse will occur at an indeterminate time no less than one nor more than 200 timer clock cycles. Subsequent time out pulses will then occur at the correct interval.

If the main counter is loaded with 01 (hexadecimal), a time out pulse will occur every time the prescaler presents a count pulse to the main counter. If the main counter is loaded with 00, a time out pulse will occur every 256 count pulses.

5.1.2 Pulse Width Measurement Operation

Besides the delay mode, timers A and B may be programmed to operate in the pulse width measurement mode. In this mode an auxiliary control input is required; timers A and B auxiliary input lines are TAI and TBI. Also, in the pulse width measurement mode, interrupt channels normally associated with I4 and I3 will respond to transitions on TAI and TBI, respectively. General purpose lines I3 and I4 may still be used for I/O. A conceptual circuit of the timers in the pulse width measurement mode is shown in Figure 5-1.

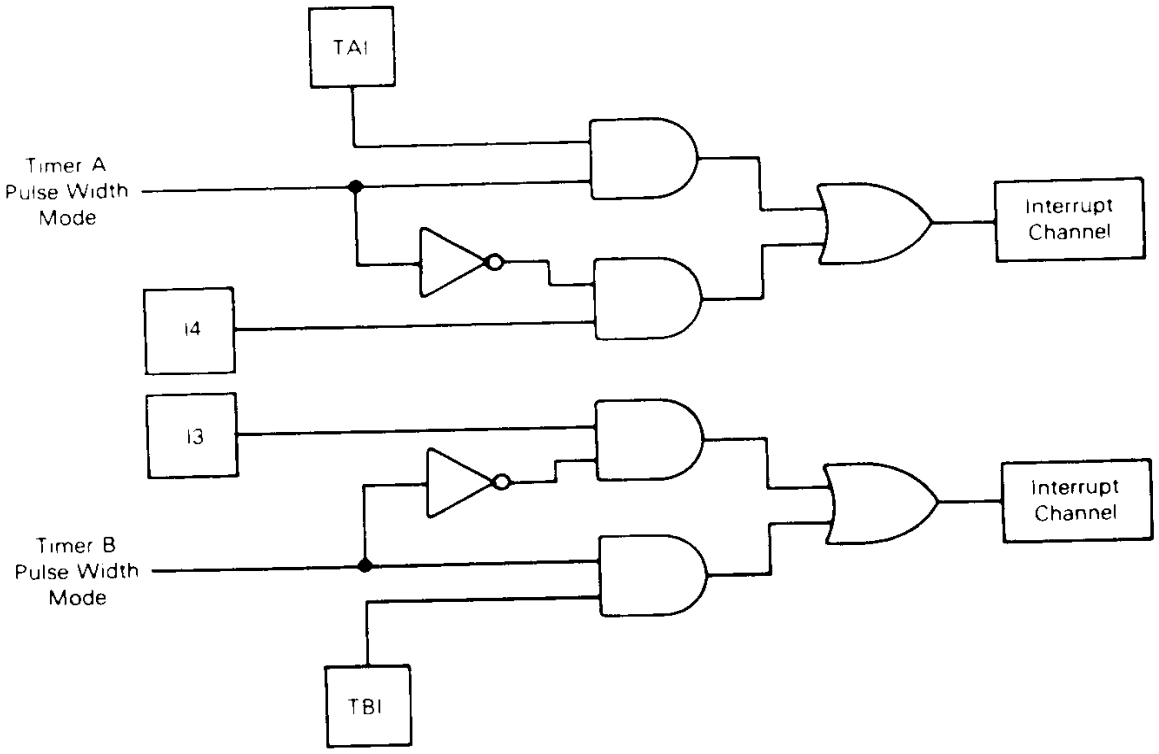


Figure 5-1. Conceptual Circuit of Timers A and B in Pulse Width Measurement Mode

The pulse width measurement mode functions similarly to the delay mode, with the auxiliary control signal acting as an enable to the timer. When the control signal is active, the prescaler and main counter are allowed to operate. When the control signal is negated, the timer is stopped. So, the width of the active pulse on TAI or TBI is measured by the number of timer counts which occur while the timer is allowed to operate.

The active state of the auxiliary input line is defined by the associated interrupt channel's edge bit in the active edge register (AER). GPIP4 of the AER is the edge bit associated with TAI and GPIP3 is associated with TBI. When the edge bit is a one, the auxiliary input will be active high, enabling the timer while the input signal is at a high level. If the edge bit is low, the auxiliary input will be active low and the timer will operate while the input signal is at a low level.

The state of the active edge bit also specifies whether a zero-to-one transition or a one-to-zero transition of the auxiliary input pin will produce an interrupt when the interrupt channel is enabled. In normal operation, programming the active edge bit to a one will produce an interrupt on the zero-to-one transition of the associated input signal. Alternately, programming the edge bit to a zero will produce an interrupt on the one-to-zero transition of the input signal. However, in the pulse width measurement mode, the interrupt generated by a transition on TAI or TBI will occur on the opposite transition as that normally defined by the edge bit.

For example, in the pulse width measurement mode, if the edge bit is a one, the timer will be allowed to run while the auxiliary input TAI is high. When TAI transitions from high to low, the timer will stop and, if the interrupt channel is enabled, an interrupt will occur. By having the interrupt occur on the one-to-zero transition instead of the zero-to-one transition, the processor will be interrupted when the pulse being measured has terminated and the width of the pulse is available from the timer. Therefore, the timers act like a divide-by-prescaler that can be programmed by the timer data register and the timers' A and B control register.

After reading the contents of the timer, the main counter must be reinitialized by writing to the timer data register to allow consecutive pulses to be measured. If the timer is written after the auxiliary input signal is active, the timer will count from the previous contents of the timer data register until it counts through 01 (hexadecimal). At that time, the main counter is loaded with the new value from the timer data register, a time out pulse is generated which will toggle the timer output, and an interrupt may be optionally generated on the timer interrupt channel. Note that the pulse width measured will include counts from before the main counter was reloaded. If the timer data register is written while the pulse is transitioning to the active state, an indeterminate value may be written into the main counter.

Once the timer is reprogrammed for another mode, interrupts will again occur as normally defined by the edge bit. Note that an interrupt may be generated as the result of placing the timer into the pulse width measurement mode or by reprogramming the timer for another mode. Also, an interrupt may be generated by changing the state of the edge bit while in the pulse width measurement mode.

5.1.3 Event Count Mode Operation

In addition to the delay mode and the pulse width measurement mode, timers A and B may be programmed to operate in the event count mode. Like the pulse width measurement mode, the event count mode also requires an auxiliary input signal, TAI or TBI, and the interrupt channels normally associated with I4 and I3 will respond to transitions on TAI and TBI, respectively. General purpose lines I3 and I4 still function normally.

In the event count mode the prescaler is disabled, allowing each active transition on TAI and TBI to produce a count pulse. The count pulse causes the main counter to decrement by one. When the timer counts through 01 (hexadecimal), a time out pulse is generated which will cause the output signal to toggle and may optionally produce an interrupt via the associated timer interrupt channel. The timer's main counter is also reloaded from the timer data register. To count transitions reliably, the input signal may only transition once every four timer clock periods. For this reason, the input signal must have a maximum frequency equal to one-fourth that of the timer clock.

The active edge of the auxiliary input signal is defined by the associated interrupt channel's edge bit. GPIP4 of the AER specifies the active edge for TAI and GPIP3 defines the active edge for TBI. When the edge bit is programmed to a one, a count pulse will be generated on the zero-to-one transition of the auxiliary input signal. When the edge bit is programmed to a zero, a count pulse will be generated on the one-to-zero transition. Also, note that changing the state of the edge bit while the timer is in the event count mode may produce a count pulse.

Besides generating a count pulse, the active transition of the auxiliary input signal will also produce an interrupt on the I3 or I4 interrupt channel, if the interrupt channel is enabled. Typically, in the event count mode, these channels are not enabled since the timer is automatically counting transitions on the input signal. If the interrupt channel were enabled, the number of transitions could be counted in the interrupt routine without requiring the use of the timer.

5.2 TIMER REGISTERS

The four timers are programmed via three control registers and four timer data registers. Control registers TACR and TBCR and timer data registers TADR and TBDR (refer to Figure 5-1) are associated with timers A and B respectively. Timers C and D are controlled by the control register TCDCR and the data registers TCDR and TDDR (refer to Figure 5-2).

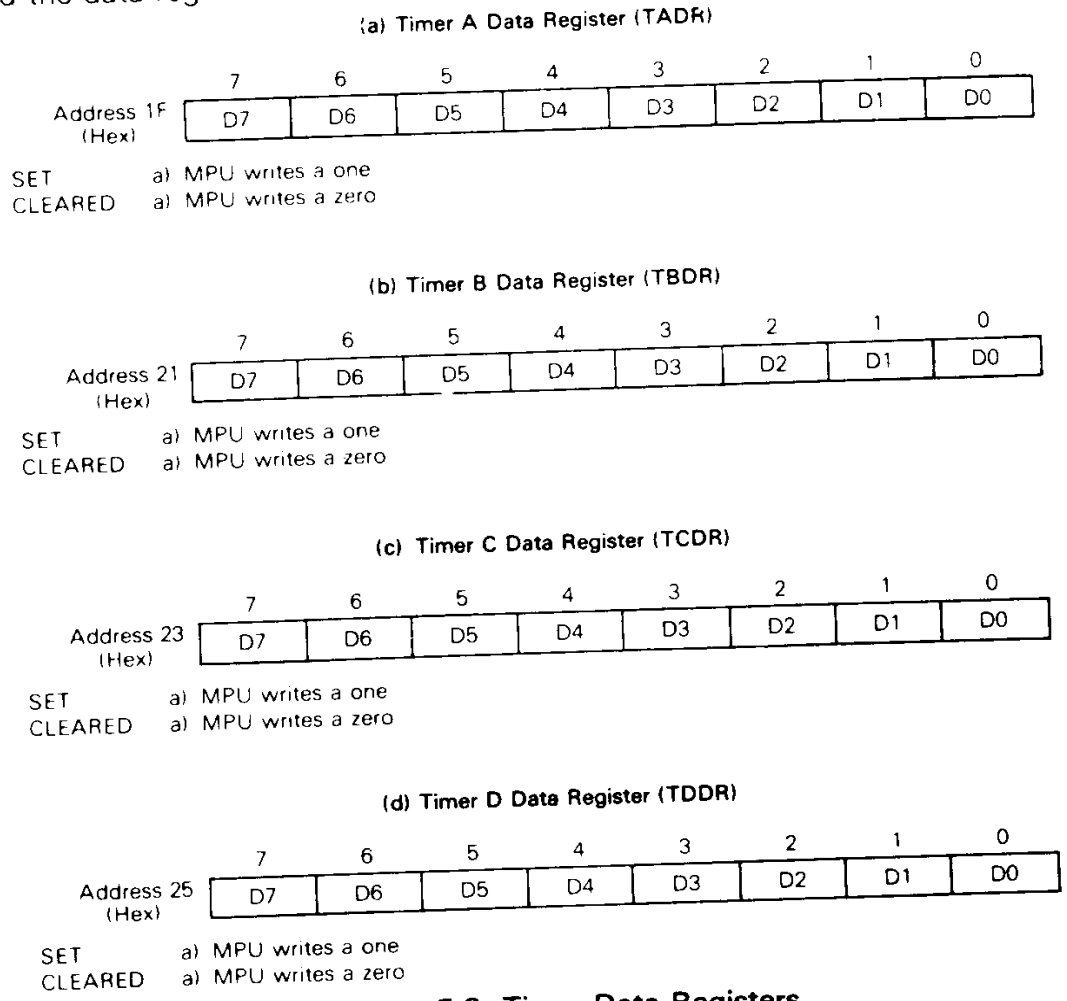


Figure 5-2. Timer Data Registers

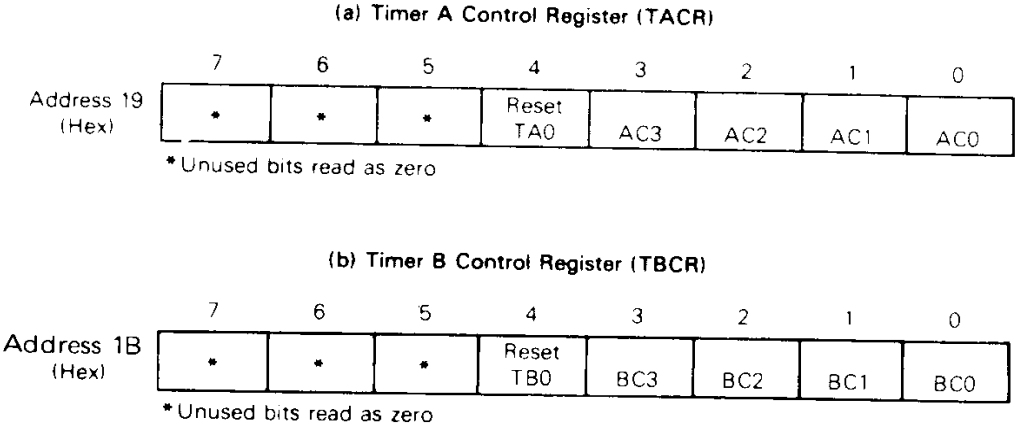
5.2.1 Timer Data Registers

Each timer's main counter is an 8-bit binary down counter. The value of the main counter may be read at any time by reading the timer's data register. The information read is the value of the counter which was captured on the last low-to-high transition of the $\overline{DS}$ pin.

The main counter is initialized by writing to the timer's data register. If the timer is stopped, data is loaded simultaneously into both the timer data register and the main counter. If the timer data register is written while the timer is enabled, the value is not loaded into the timer until the timer counts through 01 (hexadecimal). Writing the timer data register while the timer is counting through 01 (hexadecimal) will cause an indeterminate value to be loaded into the timer's main counter. The four data registers are shown in Figure 5-2.

5.2.2 Timer Control Registers

Bits in the timer control registers select the operation mode, select the prescale value, and disable the timers. Timer control registers TACR and TBCR also have bits which allow the programmer to reset output lines TAO and TBO. These control registers are shown in Figure 5-3.



Reset TAO/TBO — Timer's A and B output lines (TAO and TBO) may be forced low at any time by writing a one to the reset location in TACR and TBCR, respectively. The output will be held low only during the write operation; at the conclusion of the operation, the output will be allowed to toggle in response to a time-out pulse. When resetting TAO and TBO, the remaining bits in the control register must be written with their previous value to avoid altering the operating mode

SET — a) End of write cycle which clears the bit

CLEARED — a) MPU writes a zero
b) Reset

AC3-AC0, BC3-BC0 These bits are decoded to determine the timer operation mode.

Figure 5-3. Timer Control Registers (Sheet 1 of 2)

AC3 BC3	AC2 BC2	AC1 BC1	AC0 BC0	Operation Mode
0	0	0	0	Timer Stopped*
0	0	0	1	Delay Mode, ÷ 4 Prescaler
0	0	1	0	Delay Mode, ÷ 10 Prescaler
0	0	1	1	Delay Mode, ÷ 16 Prescaler
0	1	0	0	Delay Mode, ÷ 50 Prescaler
0	1	0	1	Delay Mode, ÷ 64 Prescaler
0	1	1	0	Delay Mode, ÷ 100 Prescaler
0	1	1	1	Delay Mode, ÷ 200 Prescaler
1	0	0	0	Event Count Mode
1	0	0	1	Pulse Width Mode, ÷ 4 Prescaler
1	0	1	0	Pulse Width Mode, ÷ 10 Prescaler
1	0	1	1	Pulse Width Mode, ÷ 16 Prescaler
1	1	0	0	Pulse Width Mode, ÷ 50 Prescaler
1	1	0	1	Pulse Width Mode, ÷ 64 Prescaler
1	1	1	0	Pulse Width Mode, ÷ 100 Prescaler
1	1	1	1	Pulse Width Mode, ÷ 200 Prescaler

*Regardless of the operation mode, counting is inhibited when the timer is stopped. The contents of the timer's main counter is not affected, although any residual count in the prescaler is lost.

SET — a) MPU writes a one
CLEARED — a) MPU writes a zero; b) Reset

(c) Timers C and D Control Register (TCDCR)

Address 1D (Hex)

7	6	5	4	3	2	1	0
*	CC2	CC1	CC0	*	DC2	DC1	DC0

*Unused bits read as zero

CC2-CC0, DC2-DC0 The bits are decoded to determine the timer operation mode.

CC2 DC2	CC1 DC1	CC0 DC0	Operation Mode
0	0	0	Timer Stopped*
0	0	1	Delay Mode, ÷ 4 Prescaler
0	1	0	Delay Mode, ÷ 10 Prescaler
0	1	1	Delay Mode, ÷ 16 Prescaler
1	0	0	Delay Mode, ÷ 50 Prescaler
1	0	1	Delay Mode, ÷ 64 Prescaler
1	1	0	Delay Mode, ÷ 100 Prescaler
1	1	1	Delay Mode, ÷ 200 Prescaler

*When the timer is stopped, counting is inhibited. The contents of the timer's main counter is not affected, although any residual count in the prescaler is lost.

SET — a) MPU writes a one
CLEARED — a) MPU writes a zero; b) Reset

Figure 5-3. Timer Control Registers (Sheet 2 of 2)

SECTION 6
UNIVERSAL SYNCHRONOUS/ASYNCHRONOUS RECEIVER-TRANSMITTER

The universal synchronous/asynchronous receiver-transmitter (USART) is a single full-duplex serial channel with a double-buffered receiver and transmitter. There are separate receive and transmit clocks and separate receive and transmit status and data bytes. The receive and transmit sections are also assigned separate interrupt channels. Each section has both a normal condition interrupt channel and an error condition interrupt channel. These channels can be optionally disabled from interrupting the processor and instead, DMA transfers can be performed using the receiver ready and transmitter ready external CMFP signals.

6.1 CHARACTER PROTOCOLS

The CMFP USART supports asynchronous and with the aid of a polynomial generator checker (PGC) supports byte synchronous character formats. These formats are selected independently of the divide-by-one and divide-by-16 clock modes.

When the divide-by-one clock mode is selected, synchronization must be accomplished externally. The receiver will sample the serial data on the rising edge of the receiver clock. In the divide-by-16 clock mode, the data is sampled at mid-bit time to increase transient noise rejection.

Also, when the divide-by-16 clock mode is selected, the USART resynchronization logic is enabled. This logic increases the channel's clock skew tolerance. When a valid transition is detected, an internal counter is reset to state zero. Transition checking is then inhibited until state four. Then at state eight, the previous state of the transition checking logic is clocked into the receive shift register.

6.1.1 Asynchronous Format

Variable word length and start/stop bit configurations are available under software control for asynchronous operation. The word length can be five to eight bits and one, one and one-half, or two stop bits can be selected. The user can also select odd, even, or no parity. For character lengths of less than eight bits, the assembled character will consist of the required number of data bits followed by zeros in the unused bit positions and a parity bit, if parity is enabled.

In the asynchronous format, start bit detection is always enabled. New data is not shifted into the receive shift register until a zero bit is received. When the divide-by-16 clock mode is selected, the false start bit logic is also active. Any transition must be stable for three positive receive clock edges to be considered valid. Then a valid zero-to-one transition must not occur for at least eight additional positive clock edges.

6.1.1.1 WAKE-UP FEATURE. In a typical serial loop multiprocessor configuration, the software protocol will usually identify the addressee(s) at the beginning of the message. In order to permit uninterested MPUs to ignore the remainder of the message, wake-up feature is included whereby all further USART receiver flag (and interrupt) processing can be inhibited until its data line goes idle. An USART receiver is re-enabled by an idle string of ten consecutive ones or during reset. Software must provide for the required idle string between consecutive messages and prevent it within messages.

6.1.2 Synchronous Format

When the synchronous character format is selected, the 8-bit synchronous character loaded into the synchronous character register is compared to received serial data until a match is found. Once synchronization is established, incoming data is clocked into the receiver. The synchronous word will be continuously transmitted during an underrun condition. All synchronous characters can be optionally stripped from the receive buffer. Figure 6-1 shows the synchronous character register.

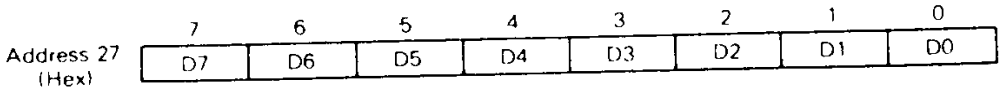


Figure 6-1. Synchronous Character Register (SCR)

The synchronous character is typically written after the data word length is selected, since unused bits in the synchronous character register are zeroed out. When parity is enabled, synchronous word length is the data word length plus one. The CMFP will compute and append the parity bit for the synchronous word when a word length of eight is selected. However, if the word length is less than eight, the user must determine the synchronous word parity and write it into the synchronous character register along with the synchronous character. The CMFP will then transmit the extra bit in the synchronous word as a parity bit.

6.1.3 USART Control Register

The USART control register (UCR) selects the clock mode and the character format for the receive and transmit sections. This register is shown in Figure 6-2.

6.2 RECEIVER

As data is received on the serial input line (SI), it is clocked into an internal 8-bit shift register until the specified number of data bits have been assembled. This character will then be transferred to the receive buffer, assuming that the last word in the receiver buffer has been read. This transfer produces a buffer full interrupt to the processor.

Reading the receive buffer satisfies the buffer full condition and allows a new data word to be transferred to the receive buffer when it is assembled. The receive buffer is accessed by reading the USART data register (UDR). The UDR is simply an 8-bit data register used when transferring data from the CMFP and the CPU.

Each time a word is transferred to the receive buffer, its status information is latched into the receiver status register (RSR). The RSR is not updated again until the data word in the receive buffer has been read. When a buffer full condition exists, the RSR should always be read before the receive buffer (UDR) to maintain the correct correspondance between data and flags. Otherwise, it is possible that after reading the UDR and prior to reading the RSR, a new word could be received and transferred to the receive buffer. Its associated flags would be latched into the RSR, overwriting the flags for the previous data word. Then when the RSR were read to access the status information for the first data word, the flags for the new word would be retrieved.

Address 29 (Hex)	7	6	5	4	3	2	1	0
	CLK	WL1	WL0	ST1	ST0	PE	E/O	WU

CLK — Clock Mode. When this bit is zero, data will be clocked into and out of the receiver and transmitter at the frequency of their respective clocks. When this bit is a one, data will be clocked into and out of the receiver and transmitter at one sixteenth the frequency of their respective clocks. Also, the receiver data transition resynchronization logic will be enabled.

SET • ÷16 — a) MPU writes a one
CLEARED • ÷1 — a) MPU writes a zero; b) Reset

WL0, WL1 — Word Length. These two bits specify the length of the data word exclusive of start bits, stop bits, and parity.

WL1	WL0	Word Length
0	0	8 Bits
0	1	7 Bits
1	0	6 Bits
1	1	5 Bits

SET — a) MPU writes a one
CLEARED — a) MPU writes a zero; b) Reset

ST0, ST1 — Start/Stop Bit and Format Control. These two bits select the number of start and stop bits and also specify the character format.

ST1	ST0	Start Bits	Stop Bits	Format
0	0	0	0	Synchronous
0	1	1	1	Asynchronous
1	0	1	1½	Asynchronous*
1	1	1	2	Asynchronous

* Only used with divide-by-16 clock mode

SET — a) MPU writes a one
CLEARED — a) MPU writes a zero; b) Reset

PE — Parity Enable. When this bit is zero, no parity check will be made and no parity bit will be computed for transmission. When this bit is a one, parity will be checked by the receiver and parity will be calculated and inserted during data transmission. Note that parity is not automatically appended to the synchronous character for word lengths of less than eight bits. In this case, the parity should be written into the synchronous character register along with the synchronous word

SET — a) MPU writes a one
CLEARED — a) MPU writes a zero; b) Reset

E/O — Even/Odd Parity. When this bit is zero, odd parity is selected. When this bit is a one, even parity is selected.

SET — a) MPU writes a one
CLEARED — a) MPU writes a zero; b) Reset

WU — Bit 0 "Wake-up" on idle line. When set, WU enables the wake-up function; it is cleared by ten consecutive ones or during reset. WU will not be set if the line is idle. Refer to 6.1.1.1.

Figure 6-2. USART Control Register (UCR)

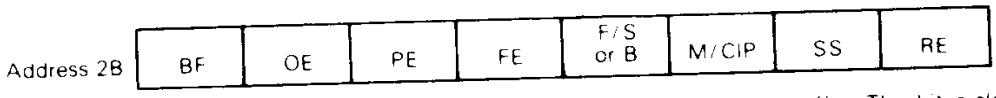
6.2.1 Receiver Interrupt Channels

The USART receive section is assigned two interrupt channels. One indicates the buffer full condition, while the other channel indicates an error condition. Error conditions include overrun, parity error, synchronous found, and break. These interrupting conditions correspond to the BF, OE, PE, and F/S or B bits of the receiver status register. These flags will function as described in **6.2.2** whether the receiver interrupt channels are enabled or disabled.

While only one interrupt is generated per character received, two dedicated interrupt channels allow separate vector numbers to be assigned for normal and abnormal receiver conditions. When a received word has an error associated with it and the error interrupt channel is enabled, an interrupt will be generated on the error channel only. However, if the error channel is disabled, an interrupt for an error condition will be generated on the buffer full interrupt channel along with interrupts produced by the buffer full condition. The receiver status register must always be read to determine which error condition produced the interrupt.

6.2.2 Receiver Status Register

The receiver status register contains the receive buffer full flag, the synchronous strip enable, the receiver enable, and various status information associated with the data word in the receive buffer. The RSR is latched each time a data word is transferred to the receive buffer. RSR flags cannot change again until the data word has been read. The exception is the character in progress flag which monitors when a new word is being assembled in the asynchronous character format. The receiver status register is shown in Figure 6-3.



BF — Buffer Full. This bit is set when a received word is transferred to the receive buffer. This bit is cleared when the receive buffer is read by accessing the USART data register (UDR). This bit is read only.
- SET: a) Received word transferred to buffer
- CLEARED: a) Receive buffer read; b) Reset

OE — Overrun Error. An overrun error occurs when a received word is due to be transferred to the receive buffer, but the receive buffer is full. Neither the receive buffer nor the RSR is overwritten. The OE bit is set after the receive buffer full condition is satisfied by reading the UDR. This error condition will generate an interrupt to the processor. The OE bit is cleared by reading the RSR. New data words will not be assembled until the RSR is read.
- SET: a) Incoming word received and receive buffer full
- CLEARED: a) Receiver status register read; b) Reset

PE — Parity Error. This bit is set when the word transferred to the receive buffer has a parity error. This bit is cleared when the word transferred to the receive buffer does not have a parity error.
- SET: a) Word in receive buffer has a parity error
- CLEARED: a) Word in receive buffer does not have a parity error; b) Reset

FE — Frame Error. A frame error exists when a non-zero data word is not followed by a stop bit in the asynchronous character format. The FE bit is set when the word transferred to the receive buffer has a frame error. The FE bit is cleared when the word transferred to the receive buffer does not have a frame error.
- SET: a) Word in receive buffer has a frame error
- CLEARED: a) Word in receive buffer does not have a frame error; b) Reset

Figure 6-3. Receiver Status Register (RSR) (Sheet 1 of 2)

F/S or B — Found/ Search or Break Detect. In the synchronous character format this bit can be set or cleared in software. When the bit is a zero, the USART receiver is placed in the search mode. The incoming data is compared to the synchronous character register (SCR) and the word length counter is disabled. The F/S bit will automatically be set when a match is found and the word length counter will be enabled. An interrupt will also be produced on the receive error channel.

SET
a) Incoming word matches synchronous character

CLEARED
a) MPU writes a zero
b) Incoming word does not match synchronous character
c) Reset

In the asynchronous character format, this flag indicates a break condition. A break is detected when an all zero data word with no stop bit is received. The break condition continues until a non-zero data bit is received. The B bit is set when the word transferred to the receive buffer is a break indication. A break condition generates an interrupt to the processor. This bit is cleared when a non-zero data bit is received and the break condition has been acknowledged by reading the RSR at least once. An end of break interrupt will be generated when the bit is cleared.

SET
a) Word in receive buffer is a break

CLEARED
a) Break terminates and receiver status register read since beginning of break condition
b) Reset

M or CIP — Match/ Character in Progress. In the synchronous character format, this flag indicates that a synchronous character has been received. The M bit is set when the word transferred to the receive buffer matches the synchronous character register. The M bit is cleared when the word transferred to the receive buffer does not match the synchronous character register.

SET
a) Word transferred to receive buffer matches the synchronous character

CLEARED
a) Word transferred to receive buffer does not match synchronous character
b) Reset

In the asynchronous character format, this flag indicates that a word is being assembled. The CIP bit is set when a start bit is detected. The CIP bit is cleared when the final stop bit has been received.

SET
a) Start bit is detected

CLEARED
a) End of word detected
b) Reset

SS — Synchronous Strip Enable. When this bit is a one, data words that match the synchronous character register will not be loaded into the receive buffer and no buffer full condition will be produced. When this bit is a zero, data words that match the synchronous character register will be transferred to the receive buffer and a buffer full condition will be produced.

SET
a) MPU writes a one

CLEARED
a) MPU writes a zero
b) Reset

RE — Receiver Enable. When this bit is a zero, the receiver will be immediately disabled. All flags will be cleared. When this bit is a one, normal receiver operation is enabled. This bit should not be set to a one until the receiver clock is active.

SET
a) MPU writes a one
b) Transmitter is disabled in auto-turnaround mode

CLEARED
a) MPU writes a zero
b) Reset

Figure 6-3. Receiver Status Register (RSR) (Sheet 2 of 2)

6.2.3 Special Receive Considerations

Certain receive conditions relating to the overrun error flag and the break detect flag require further explanation. Consider the following examples:

1) A break is received while the receive buffer is full.
   This does not produce an overrun condition. Only the B flag will be set after the receiver buffer is read.
2) A new word is received and the receive buffer is full. A break is received before the receive buffer is read.
   Both the B and OE flags will be set when the buffer full condition is satisfied.

6.3 TRANSMITTER

The transmit buffer is loaded by writing to the USART data register (UDR). The data word will be transferred to an internal 8-bit shift register when the last word in the shift register has been transmitted. This will produce a buffer empty condition. If the transmitter completes the transmission of the word in the shift register before a new word is written to the transmit buffer, an underrun error will occur. In the asynchronous character format, the transmitter will send a mark until the transmit buffer is written. In the synchronous character format, the transmitter will continuously send the synchronous character.

The transmit buffer can be loaded prior to enabling the transmitter. After the transmitter is enabled, there is a delay before the first bit is output. The serial output line (SO) should be programmed to be high, low, or high impedance when the transmitter is enabled to force the output line to the desired state until the first bit is shifted out. Note that a one bit will always be transmitted prior to the word in the transmit shift register when the transmitter is first enabled.

When the transmitter is disabled, any word currently being transmitted will continue to completion. However, any word in the transmit buffer will not be transmitted and will remain in the buffer. So, no buffer empty condition will occur. If the buffer is empty when the transmitter is disabled, the buffer empty condition will remain, but no underrun condition will be generated when the word in transmission is completed. If no word is being transmitted when the transmitter is disabled, the transmitter will stop at the next rising edge of the internal shift clock.

In the asynchronous character format, the transmitter can be programmed to send a break. The break will be transmitted once the word currently in the shift register has been sent. If the shift register is empty, the break command will be effective immediately. An END interrupt will be generated at every normal character boundary to aid in timing the break transmission. The break will continue until the break command is cleared.

Any character in the transmit buffer at the start of a break will be transmitted when the break is terminated. If the transmit buffer is empty at the start of a break, it may be written at any time during the break. If the buffer is still empty at the end of the break, an underrun condition will exist.

Disabling the transmitter during a break condition causes the transmitter to cease transmission of the break character at the end of the current character. No end of break stop bit will be transmitted. Even if the transmit buffer is empty, no buffer empty condition will occur nor will an underrun condition occur. Also, any word in the transmit buffer will remain.

6.3.1 Transmitter Interrupt Channels

The USART transmit section is assigned two interrupt channels. One channel indicates a buffer empty condition and the other channel indicates an underrun or end condition. These interrupting conditions correspond to the BE, UE, and END flag bits of the transmitter status register (TSR). The flag bits will function as described in **6.3.2** whether their associated interrupt channel is enabled or disabled.

6.3.2 Transmitter Status Register

The transmitter status register contains various transmitter error flags and transmitter control bits for selecting auto-turnaround and loopback mode. The TSR is shown in Figure 6-4.

Address 2D (Hex)	7	6	5	4	3	2	1	0
	BE	UE	AT	END	B	H	L	TE

BE — Buffer Empty. This bit is set when the word in the transmit buffer is transferred to the transmit shift register. This bit is cleared when the transmit buffer is reloaded by writing to the USART data register (UDR).
SET a) Transmit buffer contents transferred to transmit shift register
CLEARED a) Transmit buffer written

UE — Underrun Error. This bit is set when the word in the transmit shift register has been transmitted before a new word is loaded into the transmit buffer. This bit is cleared by reading the TSR or by disabling the transmitter. This bit does not need to be cleared before writing to the UDR.
SET a) Transmit shift register contents transmitted before transmit buffer written
CLEARED a) Transmitter status register read
b) Transmitter disabled

AT — Auto-Turnaround. When this bit is set, the receiver will be enabled automatically after the transmitter has been disabled and the last character being transmitted is completed.
SET a) MPU writes a one
CLEARED a) Transmitter disabled

END — End of Transmission. When the transmitter is disabled while a character is being transmitted, the END will be set after the character transmission is complete. If no word is being transmitted when the transmitter is disabled, the END bit will be set immediately. The END bit is cleared by reenabling the transmitter.
SET a) Transmitter disabled
CLEARED a) Transmitter enabled

B — Break. This bit has no function in the synchronous character format. In the asynchronous character format, when this bit is set to a one, a break will be transmitted upon the completion of the transmission of any word in the transmit shift register. A break consists of an all zero data word with no stop bit. When this bit is cleared by software, the break indication will cease and normal transmission will resume. Note that when B is set, BE cannot be set.
SET a) MPU writes a one
CLEARED a) MPU writes a zero

H, L — High and Low. These control bits configure the transmitter output (SO) when the transmitter is disabled. These bits also force the transmitter output after the transmitter is enabled until END is cleared.

H	L	Output State
0	0	High Impedance
0	1	Low
1	0	High
1	1	Loopback Mode

Loopback mode internally connects the transmitter output to the receiver input and the transmitter clock to the receiver clock internally. The receiver clock (RC) and the serial input (SI) are not used. When the transmitter is disabled, SO is forced high.
SET a) MPU writes a one
CLEARED a) MPU writes a zero

TE — Transmitter Enable. When this bit is cleared, the transmitter is disabled. The UE bit will be cleared and the END bit will be set. When this bit is set, the transmitter is enabled. The transmitter output will be driven according to the H and L bits until transmission begins. A one bit will be transmitted before the transmission of the word in the transmit shift register is begun.
SET a) MPU writes a one
CLEARED a) MPU writes a zero
b) Reset

Figure 6-4. Transmitter Status Register (TSR)

6.4 DMA OPERATION

USART error conditions are only valid for each character boundary. When the USART performs block data transfers by using the DMA handshake lines $\overline{RR}$ (receiver ready) and $\overline{TR}$ (transmitter ready), errors must be saved and checked at the end of a block. This is accomplished by enabling the error channel for the receiver or transmitter and by masking interrupts for this channel. Once the transfer is complete, interrupt pending register A is read. Any pending receiver or transmitter error indicates an error in the data transfer.

SECTION 7
ELECTRICAL CHARACTERISTICS

This section contains the electrical specifications and associated timing information for the TS68HC901 multi-function peripheral.

7.1 MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	−0.3 to 7.0	V
Input Voltage	V_{in}	−0.3 to 7.0	V
Operating Temperature Range TS68HC901C TS68HC901V TS68HC901M	T_A	T_L to T_H 0 to +70 −40 to +85 −55 to +125	°C
Storage Temperature	T_{stg}	−65 to 150	°C
Power Dissipation	P_D	30	mW

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum-rated voltages to this high-impedance circuit. Reliability of operation is enhanced if unused inputs are tied to an appropriate logic voltage level (e.g., either V_{CC} or GND).

7.2 THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Rating
Thermal Resistance Ceramic Plastic	θ_{JA}	40 TBD	°C/W

7.3 POWER CONSIDERATIONS

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

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

Where:

T_A = Ambient Temperature, °C

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

$P_D = P_{INT} + P_{I/O}$

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

$P_{I/O}$ = Power Dissipation on Input and Output Pins — User Determined

For most applications $P_{I/O} < P_{INT}$ and can be neglected.

An approximate relationship between P_D and T_J (if $P_{I/O}$ is neglected) is:

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

Solving equations 1 and 2 for K gives:

$$K = P_D \bullet (T_A + 273°C) + \theta_{JA} P_D^2 \quad (3)$$

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.

7.4 DC ELECTRICAL CHARACTERISTICS

($T_A = T_L$ to T_H $V_{CC} = -5$ V ±5%, unless otherwise noted)

Characteristic	Symbol	Min	Max	Unit
Input High Voltage except XTAL1, XTAL2	V_{IH}	2.0	V_{DD}+0.3	V
Input High Voltage XTAL1, XTAL2	V_{IH}	V_{DD}-1.5	V_{DD}+0.3	V
Input Low Voltage	V_{IL}	-0.3	0.8	V
Output High Voltage, Except $\overline{DTACK}$ (I_{OH} = -120 µA)	V_{OH}	4.1	–	V
Output Low Voltage, Except $\overline{DTACK}$ (I_{OL} = 2.0 mA)	V_{OL}	–	0.5	V
Power Supply Current (Outputs Open)	I_{LL}	–	3.7	mA
Input Leakage Current (V_{in} = 0 to V_{CC})	I_{LI}	–	±10	µA
Hi-Z Output Leakage Current in Float (V_{out} = 2.4 to V_{CC})	I_{LOH}	–	10	µA
Hi-Z Output Leakage Current in Float (V_{out} = 0.5 V)	I_{LOL}	–	-10	µA
$\overline{DTACK}$ Output Source Current (V_{out} = 2.4 V)	I_{OH}	–	-400	µA
$\overline{DTACK}$ Output Sink Current (V_{out} = 0.5 V)	I_{OL}	–	5.3	mA
Pull Down Resistor	R_{MPX}	1	3	MΩ

7.5 CAPACITANCE (T_A = 25°C, f = 1 MHz, unmeasured pins returned to ground)

Characteristic	Symbol	Min	Max	Unit
Input Capacitance	C_{in}	–	10	pF
Hi-Z Output Capacitance	C_{out}	–	10	pF

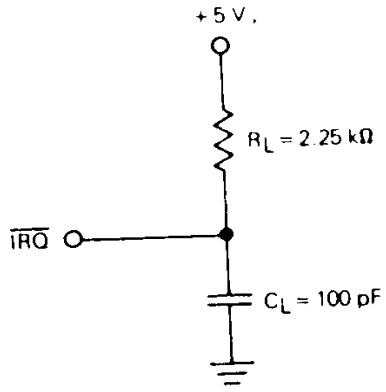
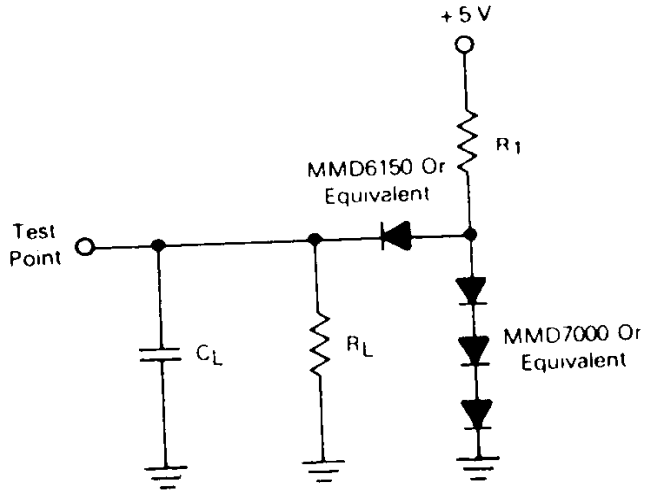


Figure 7-1. $\overline{IRQ}$ Test Load



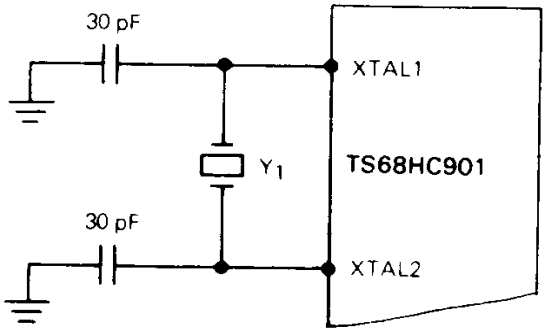
for all outputs except $\overline{DTACK}$
C_L = 100 pF
R_L = 20 kΩ
R_1 = 1.90 kΩ

for $\overline{DTACK}$
C_L = 130 pF
R_L = 6 kΩ
R_1 = 740 Ω

Figure 7-2. Typical Test Load

7.6 CLOCK TIMING

Characteristic	Symbol	Min	Max	Unit
Frequency of Operation	f	1.0	4.0	MHz
Cycle Time	t_{cyc}	250	1000	ns
Clock Pulse Width	t_{CL}, t_{CH}	110	250	ns
Rise and Fall Times	t_{Cr}, t_{Cf}	–	15	ns



Crystal Parameters
Parallel resonance fundamental mode AT cut
$R_S \leq 150\ \Omega$ (f = 2.8 – 4.0 MHz)
$R_S \leq 300\ \Omega$ (f = 2.0 – 2.7 MHz)
C_L = 18 pF, C_M = 0.02 pF, C_R = 5 pF, L_M = 96 MHz
f (typical) = 2.4576 MHz

Figure 7-3. CMFP External Oscillator Components

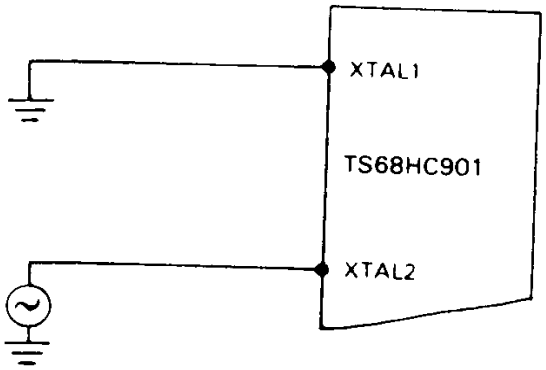


Figure 7-3-1. CMFP External Clock Connection

7.7 AC ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V_{DC} ± 5%, V_{SS} = 0 V_{DC}, T_A = 0°C to 70°C unless otherwise noted). See figures 7-4 through 7-10.

Num	Characteristic	4 MHz		8 MHz		Unit
		Min	Max	Min	Max	
1	$\overline{CS}$, $\overline{DS}$ Width High	50	—	50	—	—
2	R/$\overline{W}$, A1-A5 Valid to Falling $\overline{CS}$ (Setup)	30	—	20	—	ns
3	Data Valid Prior to Falling CLK	280	—	100	—	ns
4 (3)	$\overline{CS}$, $\overline{IACK}$ Valid to falling Clock (Setup)	50	—	50	—	ns
5	CLK Low to $\overline{DTACK}$ Low	—	220	—	90	ns
6	$\overline{CS}$, $\overline{DS}$ or $\overline{IACK}$ High to $\overline{DTACK}$ High	—	60	—	50	ns
7	$\overline{CS}$, $\overline{DS}$ or $\overline{IACK}$ High to $\overline{DTACK}$ Tri-state	—	100	—	100	ns
8	$\overline{DTACK}$ Low to Data Invalid (Hold Time)	0	—	0	—	ns
9	$\overline{CS}$, $\overline{DS}$ or $\overline{IACK}$ High to Data Tri-state	—	50	—	50	ns
10	$\overline{CS}$ or $\overline{DS}$ High to R/$\overline{W}$, A1-A5 Invalid (Hold Time)	0	—	0	—	ns
11 (3,5)	Data Valid from $\overline{CS}$ Low	—	310	—	180	ns
12	Read Data Valid to $\overline{DTACK}$ Low (Setup Time)	50	—	0	—	ns
13	$\overline{DTACK}$ Low to $\overline{DS}$, $\overline{CS}$ or $\overline{IACK}$ High (Hold Time)	0	—	0	—	ns
14	$\overline{IEI}$ Low to Falling $\overline{CLK}$ (Setup)	50	—	50	—	ns
15 (1)	$\overline{IEO}$ Valid from Clock Low (Delay)	—	180	—	120	ns
16	Data Valid from Clock Low (Delay)	—	300	—	180	ns
17	$\overline{IEO}$ Invalid from $\overline{IACK}$ High (Delay)	—	150	—	100	ns
18	$\overline{DTACK}$ Low from Clock High (Delay)	—	180	—	100	ns
19 (1)	$\overline{IEO}$ Valid from $\overline{IEI}$ Low (Delay)	—	100	—	100	ns
20	Data Valid from $\overline{IEI}$ Low (Delay)	—	220	—	140	ns
21	Clock Cycle Time	250	1000	125	1000	ns
22	Clock Width Low	110	—	55	—	ns
23	Clock Width High	110	—	55	—	ns
24 (4)	$\overline{CS}$, $\overline{IACK}$ Inactive to Rising Clock (Setup)	100	—	50	—	ns
25	I/O Minimum Active Pulse Width	100	—	100	—	ns
26	$\overline{IACK}$ Width High/Minimum Delay between two Pulses	2	—	2	—	CLK
27	I/O Data Valid from Rising $\overline{CS}$ or $\overline{DS}$	—	450	—	350	ns
28	Receiver Ready Delay from Falling RC	—	600	—	200	ns
29	Transmitter Ready Delay from Falling TC	—	600	—	200	ns
30 (6)	Timer Output Low from Rising Edge of $\overline{CS}$ or $\overline{DS}$ (A & B) (Reset T_{OUT})	—	450	—	200	ns

Num	Characteristic	4 MHz		8 MHz		Unit
		Min	Max	Min	Max	
31 (2)	T_{OUT} Valid from Internal Timeout	—	2 t_{CLK} + 300	—	2 t_{CLK} + 300	ns
32	Timer Clock Low Time	110	—	55	—	ns
33	Timer Clock High Time	110	—	55	—	ns
34	Timer Clock Cycle Time	250	1000	125	1000	ns
35	$\overline{RESET}$ Low Time	2	—	1.5	—	μs
36	Delay to Falling $\overline{INTR}$ from External Interrupt Active Transition	—	380	—	250	ns
37	Transmitter Internal Interrupt Delay from Falling Edge of TC	550	—	250	—	ns
38	Receiver Buffer Full Interrupt Transition Delay from Rising Edge of RC	800	—	400	—	ns
39	Receiver Error Interrupt Transition Delay from Falling Edge of RC	800	—	400	—	ns
40	Serial In Set Up Time to Rising Edge of RC (Divide by one only)	80	—	50	—	ns
41	Data Hold Time from Rising Edge of RC (Divide by one only)	350	—	100	—	ns
42	Serial Output Data Valid from Falling Edge of TC (÷ 1)	—	440	—	200	ns
43	Transmitter Clock Low Time	500	—	250	—	ns
44	Transmitter Clock High Time	500	—	250	—	ns
45	Transmitter Clock Cycle Time	1.05	*	0.55	*	μs
46	Receiver Clock Low Time	500	—	250	—	ns
47	Receiver Clock High Time	500	—	250	—	ns
48	Receiver Clock Cycle Time	1.05	*	0.55	*	μs
49 (2)	$\overline{CS}$, $\overline{IACK}$, $\overline{DS}$ Width Low	—	80	—	80	T_{CLK}
50	Serial Output Data Valid from Falling Edge of TC (÷ 16)	—	490	—	200	ns
51	Cycle Time	1000	—	—	—	ns
52	Pulse Width, E High	430	—	—	—	ns
53	Pulse Width, E Low	450	—	—	—	ns
54	Address, R/$\overline{W}$ Setup Time Before E	80	—	—	—	ns
55	$\overline{CS}$ Setup Time Before E	80	—	—	—	ns
56	Address Hold Time	10	—	—	—	ns
57	$\overline{CS}$ Hold Time	10	—	—	—	ns
58	Output Data Delay Time (Read)	—	250	—	—	ns
59	Data Hold Time (Read)	0	100	—	—	ns

Num	Characteristic	4 MHz		8 MHz		Unit
		Min	Max	Min	Max	
60	Input Data Setup Time (Write)	280	—	—	—	ns
61	Data Hold Time (Write)	20	—	—	—	ns
62	Cycle Time	800	—	—	—	ns
63	Pulse Width $\overline{DS}$ Low or RD/$\overline{WR}$ High	350	—	—	—	ns
64	Pulse Width DS High or RD/$\overline{WR}$ Low	340	—	—	—	ns
65	Pulse Width AS/ALE High	100	—	—	—	ns
66	Delay AS Fall to $\overline{DS}$ Rise or ALE Fall to RD/$\overline{WR}$ Fall	30	—	—	—	ns
67	Delay $\overline{DS}$ or RD/$\overline{WR}$ Rise to AS/ALE Rise	30	—	—	—	ns
68	R/$\overline{W}$ Setup Time to $\overline{DS}$	100	—	—	—	ns
69	R/$\overline{W}$ Hold Time to $\overline{DS}$	10	—	—	—	ns
70	Address Setup Time to AS/ALE	20	—	—	—	ns
71	Address Hold Time to AS/ALE	20	—	—	—	ns
72	Data Setup Time to $\overline{DS}$ or WR (Write)	280	—	—	—	ns
73	Delay Data to $\overline{DS}$ or RD (Read)	—	250	—	—	ns
74	Data Hold Time to $\overline{DS}$ or WR (Write)	20	—	—	—	ns
75	Data Hold Time to $\overline{DS}$ or RD (Read)	0	100	—	—	ns
76	CE Setup Time to AS/ALE Fall	20	—	—	—	ns
77	CE Hold Time to $\overline{DS}$, RD or $\overline{WR}$	20	—	—	—	ns

Notes:

1. $\overline{IEO}$ only goes low if no acknowledgeable interrupt is pending. If $\overline{IEO}$ goes low, $\overline{DTACK}$ and the data bus remain tri-stated.
2. T_{CLK} refers to the clock applied to the CMFP CLK input pin. t_{CLK} refers to the timer clock signal, regardless of whether that signal comes from the XTAL1/XTAL2 crystal clock inputs or the TAI or TBI timer inputs.
3. If the setup time is not met, $\overline{CS}$ or $\overline{IACK}$ will not be recognized until the next falling CLK.
4. If the setup time is met (for consecutive cycles), the minimum hold-off time of one clock cycle will be obtained. If not met, the hold-off will be two clock cycles.
5. Although $\overline{CS}$ and $\overline{DTACK}$ are synchronized with the clock, the data out during a reed cycle is asynchronous to the clock, relying only on $\overline{CS}$ for timing.
6. Spec. 30 applies to timer outputs TAO and TBO only.

7.7.1 AC ELECTRICAL CHARACTERISTICS - READ CYCLES

($V_{CC} = 5.0 V_{DC} \pm 5\%$, $V_{SS} = 0 V_{DC}$, $T_A = T_L$ to T_H unless otherwise noted)

Num	Characteristic	4 MHz		8 MHz		Unit
		Min	Max	Min	Max	
1	$\overline{CS}$, $\overline{DS}$ Width High	50	—	50	—	—
2	R/$\overline{W}$, A1-A5 Valid to Falling $\overline{CS}$ (Setup)	30	—	20	—	ns
4 (3)	$\overline{CS}$, $\overline{IACK}$ Valid to Falling Clock (Setup)	50	—	50	—	ns
5	CLK Low to $\overline{DTACK}$ Low	—	220	—	90	ns
6	$\overline{CS}$, $\overline{DS}$ or $\overline{IACK}$ High to $\overline{DTACK}$ High	—	60	—	50	ns
7	$\overline{CS}$, $\overline{DS}$ or $\overline{IACK}$ High to $\overline{DTACK}$ Tri-state	—	100	—	100	ns
8	$\overline{DTACK}$ Low to Data Invalid (Hold Time)	0	—	0	—	ns
9	$\overline{CS}$, $\overline{DS}$ or IACK High to Data Tri-state	—	50	—	50	ns
10	$\overline{CS}$ or $\overline{DS}$ High to R/$\overline{W}$, A1-A5 Invalid (Hold Time)	0	—	0	—	ns
11 (3,5)	Data Valid from $\overline{CS}$ Low	—	310	—	180	ns
12	Read Data Valid to $\overline{DTACK}$ Low (Setup Time)	50	—	0	—	ns
13	$\overline{DTACK}$ Low to $\overline{DS}$, $\overline{CS}$ or $\overline{IACK}$ High (Hold Time)	0	—	0	—	ns
24 (4)	$\overline{CS}$, $\overline{IACK}$ Inactive to Rising Clock (Setup)	100	—	50	—	ns

Notes:

3. If the setup time is not met, $\overline{CS}$ or $\overline{IACK}$ will not be recognized until the next falling CLK.
4. If the setup time is met (for consecutive cycles), the minimum hold-off time of one clock cycle will be obtained. If not met, the hold-off will be two clock cycles.
5. Althrough $\overline{CS}$ and $\overline{DTACK}$ are synchronized with the clock, the data out during a reed cycle is asynchronous to the clock, relying only on $\overline{CS}$ for timing.

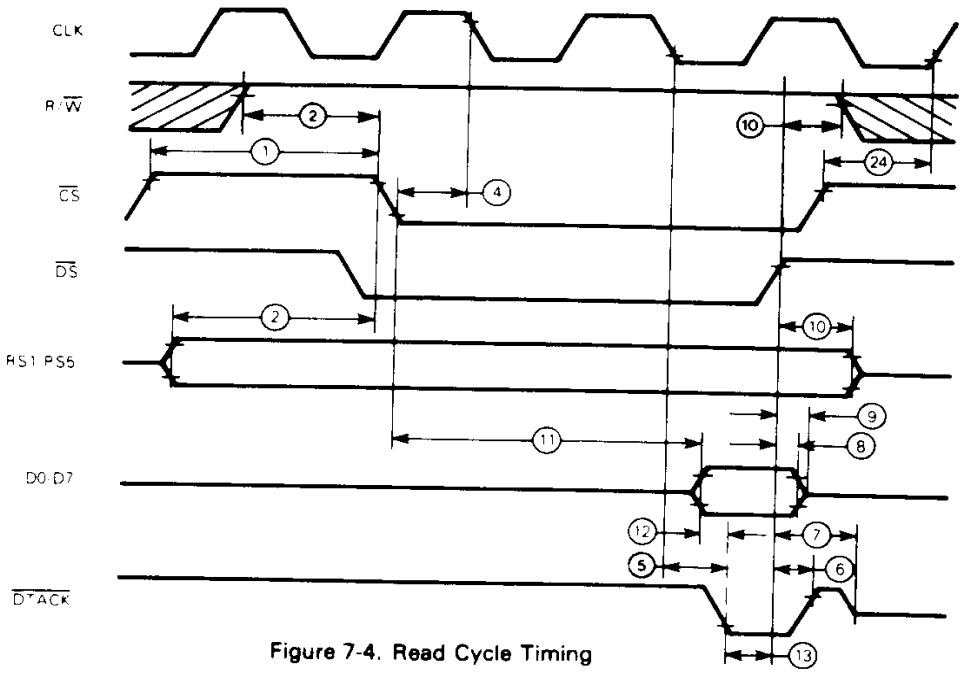


Figure 7-4. Read Cycle Timing

7.7.2 AC ELECTRICAL CHARACTERISTICS - WRITE CYCLES

($V_{CC} = 5.0 V_{DC} \pm 5\%$, $V_{SS} = 0 V_{DC}$, $T_A = T_L$ to T_H unless otherwise noted)

Num	Characteristic	4 MHz		8 MHz		Unit
		Min	Max	Min	Max	
1	$\overline{CS}$, $\overline{DS}$ Width High	50	—	50	—	—
2	R/$\overline{W}$, A1-A5 Valid to Falling $\overline{CS}$ (Setup)	30	—	20	—	ns
3	Data Valid Prior to Falling CLK	280	—	100	—	ns
4 (3)	$\overline{CS}$, $\overline{IACK}$ Valid to Falling Clock (Setup)	50	—	50	—	ns
5	$\overline{CLK}$ Low to $\overline{DTACK}$ Low	—	220	—	90	ns
6	$\overline{CS}$, $\overline{DS}$ or IACK High to $\overline{DTACK}$ High	—	60	—	50	ns
7	$\overline{CS}$, $\overline{DS}$ or $\overline{IACK}$ High to $\overline{DTACK}$ Tri-state	—	100	—	100	ns
8	$\overline{DTACK}$ Low to Data Invalid (Hold Time)	0	—	0	—	ns
10	$\overline{CS}$ or $\overline{DS}$ High to R/$\overline{W}$, A1-A5 Invalid (Hold Time)	0	—	0	—	ns
13	$\overline{DTACK}$ Low to $\overline{DS}$, $\overline{CS}$ or $\overline{IACK}$ High (Hold Time)	0	—	0	—	ns
24 (4)	$\overline{CS}$, $\overline{IACK}$ Inactive to Rising Clock (Setup)	100	—	50	—	ns

Notes:

3. If the setup time is not met, $\overline{CS}$ or $\overline{IACK}$ will not be recognized until the next falling CLK.
4. If the setup time is met (for consecutive cycles), the minimum hold-off time of one clock cycle will be obtained. If not met, the hold-off will be two clock cycles.

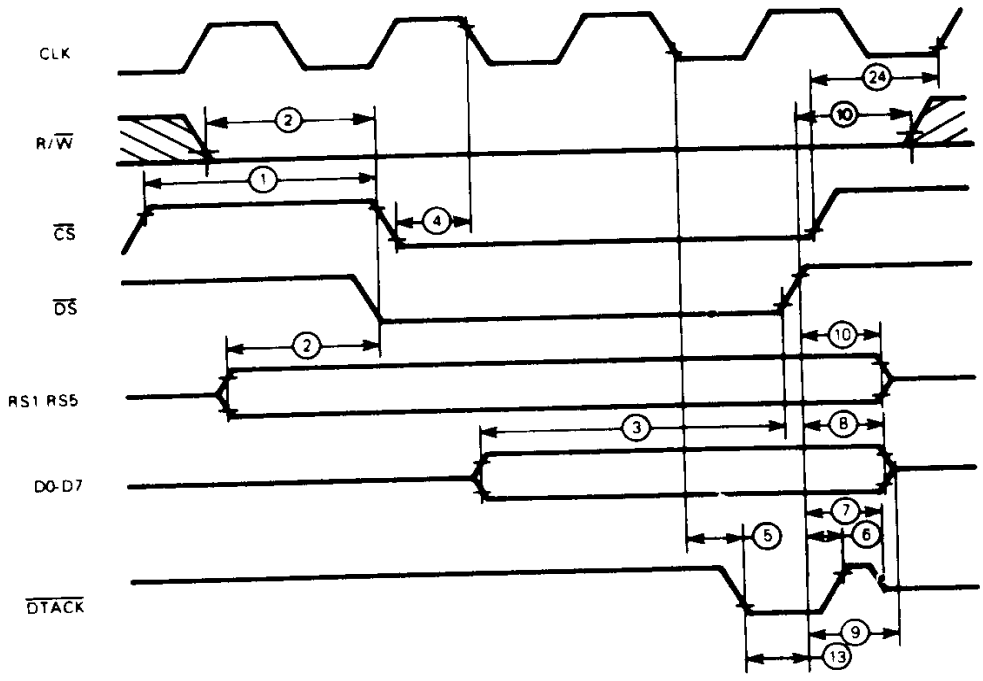


Figure 7-5. Write Cycle Timing

7.7.3 AC ELECTRICAL CHARACTERISTICS - INTERRUPT ACKNOWLEDGE CYCLES

(V_{CC} = 5.0 V_{DC} ± 5 %, V_{SS} = 0 V_{DC}, T_A = T_L to T_H unless otherwise noted)
See Figures 7-6 and 7-7.

Num	Characteristic	4 MHz		8 MHz		Unit
		Min	Max	Min	Max	
4 (3)	$\overline{CS}$, $\overline{IACK}$ Valid to Falling Clock (Setup)	50	—	50	—	ns
5	$\overline{CLK}$ Low to $\overline{DTACK}$ Low	—	220	—	90	ns
6	$\overline{CS}$, $\overline{DS}$ or $\overline{IACK}$ High to $\overline{DTACK}$ High	—	60	—	50	ns
7	$\overline{CS}$, $\overline{DS}$ or $\overline{IACK}$ High to $\overline{DTACK}$ Tri-state	—	100	—	100	ns
9	$\overline{CS}$, $\overline{DS}$ or $\overline{IACK}$ High to Data Tri-state	—	50	—	50	ns
13	$\overline{DTACK}$ Low to $\overline{DS}$, $\overline{CS}$ or $\overline{IACK}$ High (Hold Time)	0	—	0	—	ns
14	$\overline{IEI}$ Low to Falling CLK (Setup)	50	—	50	—	ns
15 (1)	$\overline{IEO}$ Valid from Clock Low (Delay)	—	180	—	120	ns
16	Data Valid from Clock Low (Delay)	—	300	—	180	ns
17	$\overline{IEO}$ Invalid from $\overline{IACK}$ High (Delay)	—	150	—	100	ns
18	$\overline{DTACK}$ Low from Clock High (Delay)	—	180	—	100	ns
19 (1)	$\overline{IEO}$ Valid from $\overline{IEI}$ Low (Delay)	—	100	—	100	ns
20	Data Valid from $\overline{IEI}$ Low (Delay)	—	220	—	140	ns
21	Clock Cycle Time	250	1000	125	1000	ns
22	Clock Width Low	110	—	55	—	ns
23	Clock Width High	110	—	55	—	ns
24 (4)	$\overline{CS}$, $\overline{IACK}$ Inactive to Rising Clock (Setup)	100	—	50	—	ns
26	$\overline{IACK}$ Width High/Minimum Delay between two Pulses	2	—	2	—	CLK

Notes:

1. $\overline{IEO}$ only goes low if no acknowledgeable interrupt is pending. If $\overline{IEO}$ goes low, $\overline{DTACK}$ and the data bus remain tri-stated.

3. If the setup time is not met, $\overline{CS}$ or $\overline{IACK}$ will not be recognized until the next falling CLK.

4. If the setup time is met (for consecutive cycles), the minimum hold-off time of one clock cycle will be obtained. If not met, the hold-off will be two clock cycles.

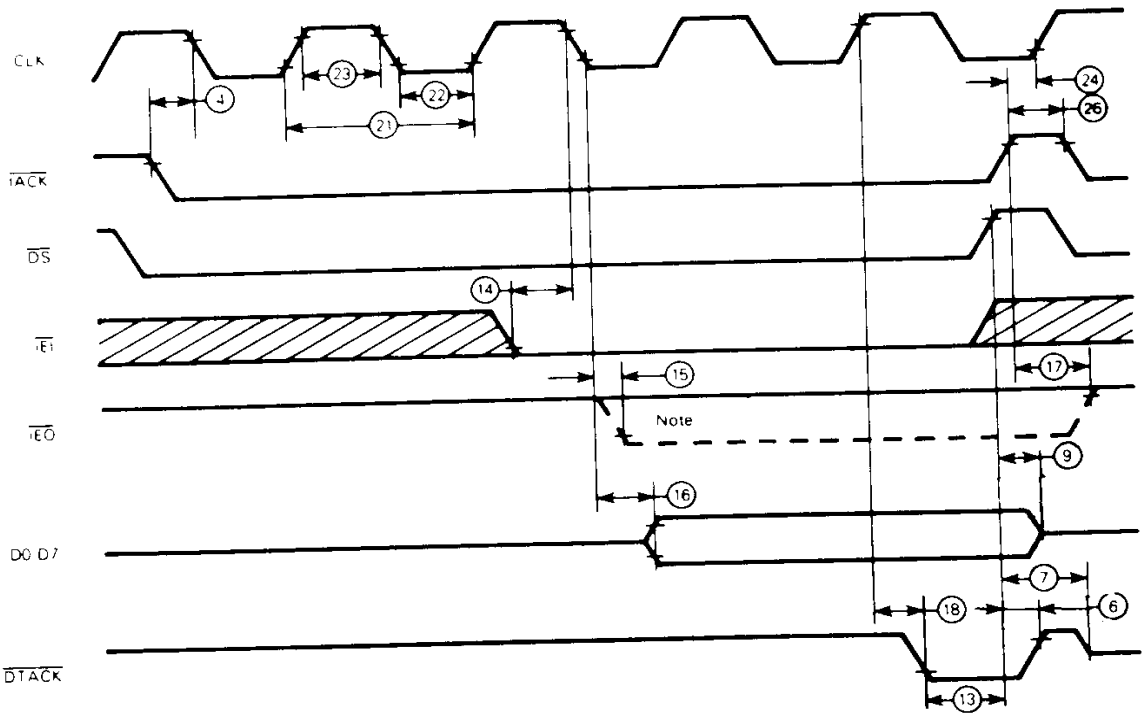
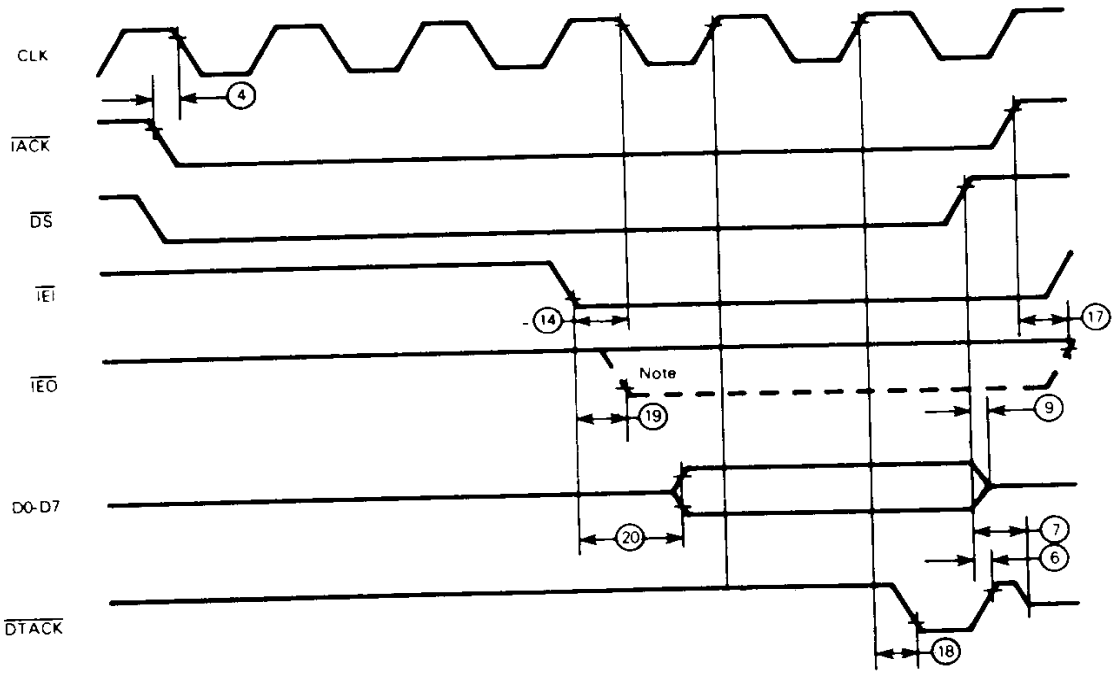


Figure 7-6. Interrupt Acknowledge Cycle ($\overline{IEI}$ Low)



Note: $\overline{CS}$ and $\overline{IACK}$ must be a function of $\overline{DS}$

Figure 7-7. Interrupt Acknowledge Cycle (IEI High)

7.7.4 AC ELECTRICAL CHARACTERISTICS - 6800 INTERFACE TIMING (V_{CC} = 5.0 V_{DC} ± 5%, V_{SS} = 0V_{DC}, T_A = 0°C to 70°C unless otherwise noted). See figure 7-8.

Num	Characteristic	4 MHz		8 MHz		Unit
		Min	Max	Min	Max	
51	Cycle Time	1000	—	—	—	ns
52	Pulse Width, E High	430	—	—	—	ns
53	Pulse Width, E Low	450	—	—	—	ns
54	Address, R/$\overline{W}$ Setup Time Before E	80	—	—	—	ns
55	$\overline{CS}$ Setup Time Before E	80	—	—	—	ns
56	Address Hold Time	10	—	—	—	ns
57	$\overline{CS}$ Hold Time	10	—	—	—	ns
58	Output Data Delay Time (Read)	—	250	—	—	ns
59	Data Hold Time (Read)	0	100	—	—	ns
60	Input Data Setup Time (Write)	280	—	—	—	ns
61	Data Hold Time (Write)	20	—	—	—	ns

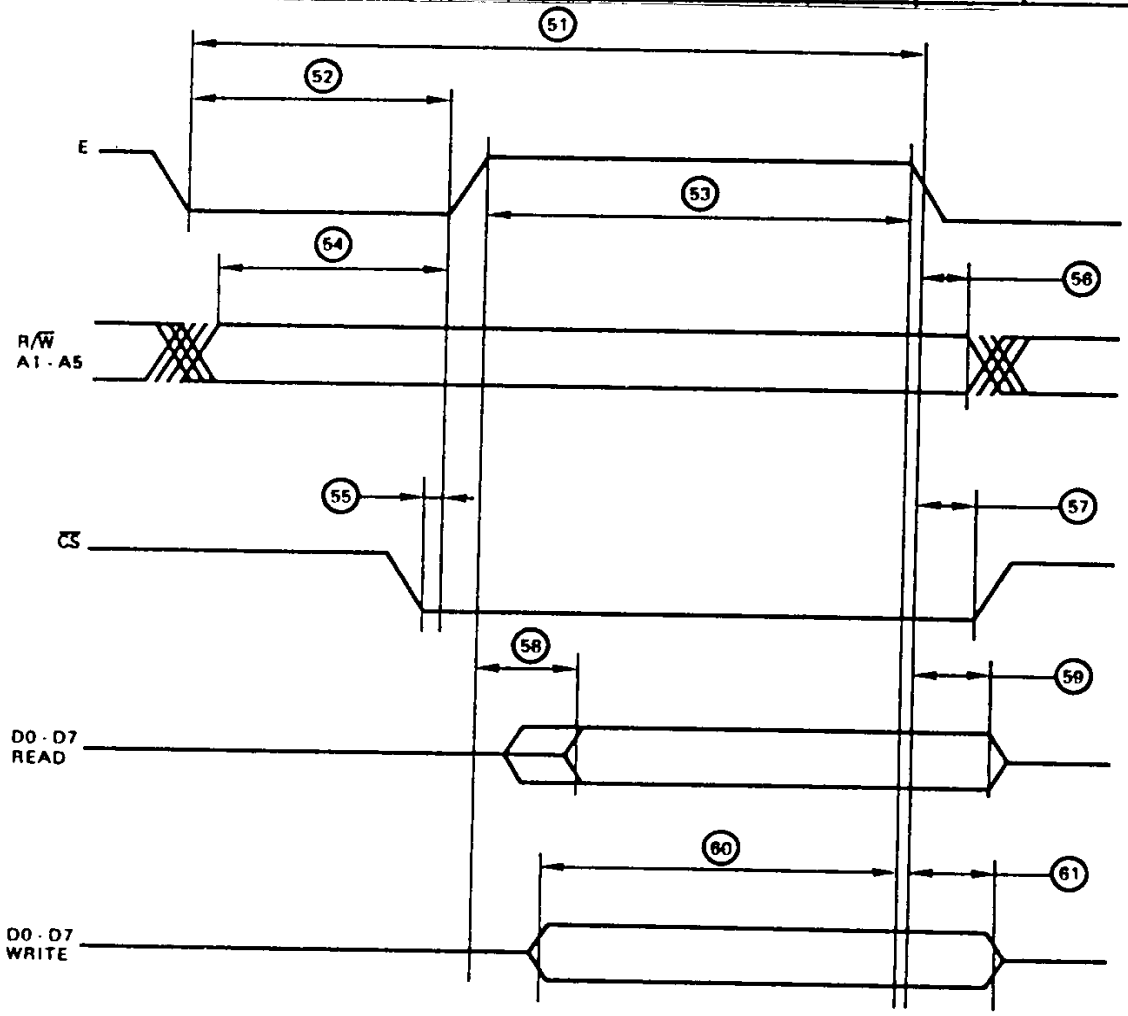


Figure 7-8. 6800 Interfacing Timing

7.7.5 AC ELECTRICAL CHARACTERISTICS - MULTIPLEXED BUS TIMING (V_{CC} = 5.0 V_{DC} ± 5%, V_{SS} = 0 V_{DC}, T_A = 0°C to 70°C unless otherwise noted). See figures 7-9, 7-10.

Num	Characteristic	4 MHz		8 MHz		Unit
		Min	Max	Min	Max	
62	Cycle Time	800	—	—	—	ns
63	Pulse Width $\overline{DS}$ Low or RD/WR High	350	—	—	—	ns
64	Pulse Width DS High or RD/WR Low	340	—	—	—	ns
65	Pulse Width AS/ALE High	100	—	—	—	ns
66	Delay AS Fall to $\overline{DS}$ Rise or ALE Fall to RD/WR Fall	30	—	—	—	ns
67	Delay $\overline{DS}$ or RD/WR Rise to AS/ALE Rise	30	—	—	—	ns
68	R/$\overline{W}$ Setup Time to $\overline{DS}$	100	—	—	—	ns
69	R/$\overline{W}$ Hold Time to $\overline{DS}$	10	—	—	—	ns
70	Address Setup Time to AS/ALE	20	—	—	—	ns
71	Address Hold Time to AS/ALE	20	—	—	—	ns
72	Data Setup Time to $\overline{DS}$ or WR (Write)	280	—	—	—	ns
73	Delay Data to $\overline{DS}$ or RD (Read)	—	250	—	—	ns
74	Data Hold Time to $\overline{DS}$ or WR (Write)	20	—	—	—	ns
75	Data Hold Time to $\overline{DS}$ or RD (Read)	0	100	—	—	ns
76	CE Setup Time to AS/ALE Fall	20	—	—	—	ns
77	CE Hold Time to $\overline{DS}$, RD or WR	20	—	—	—	ns

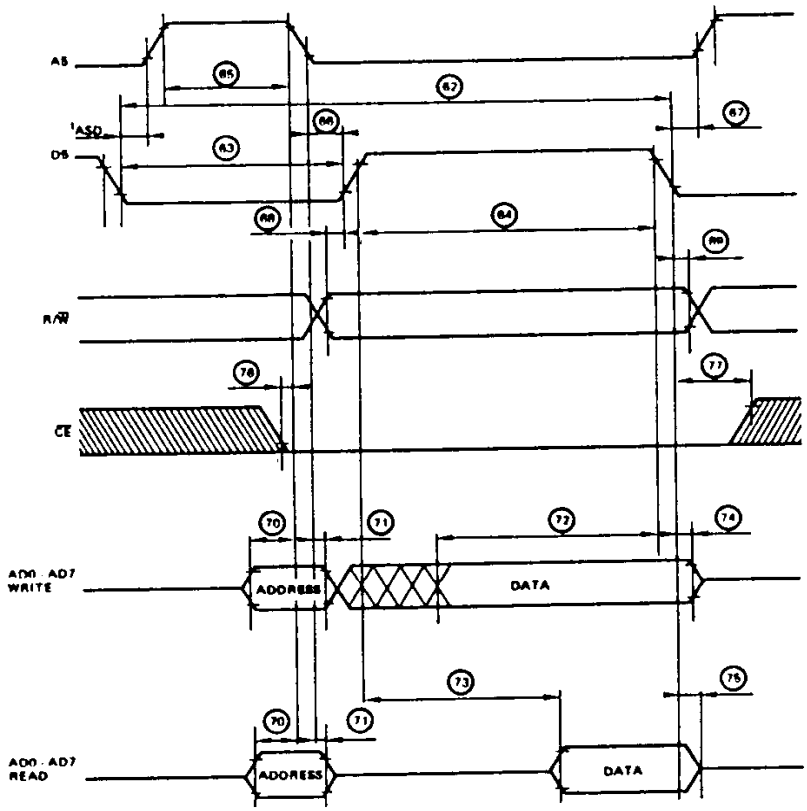


Figure 7-9. Multiplexed Bus Timin Motorola Type

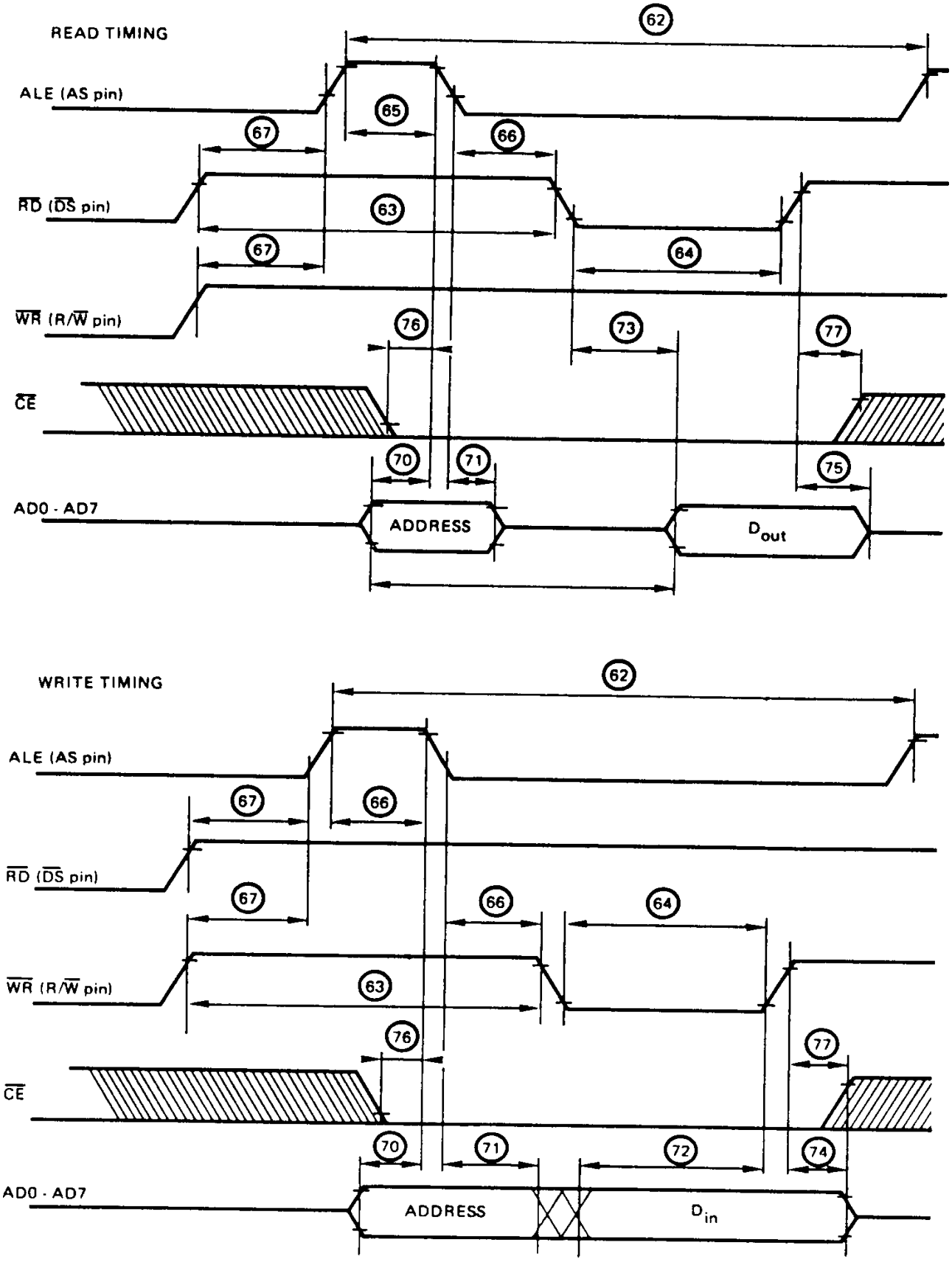
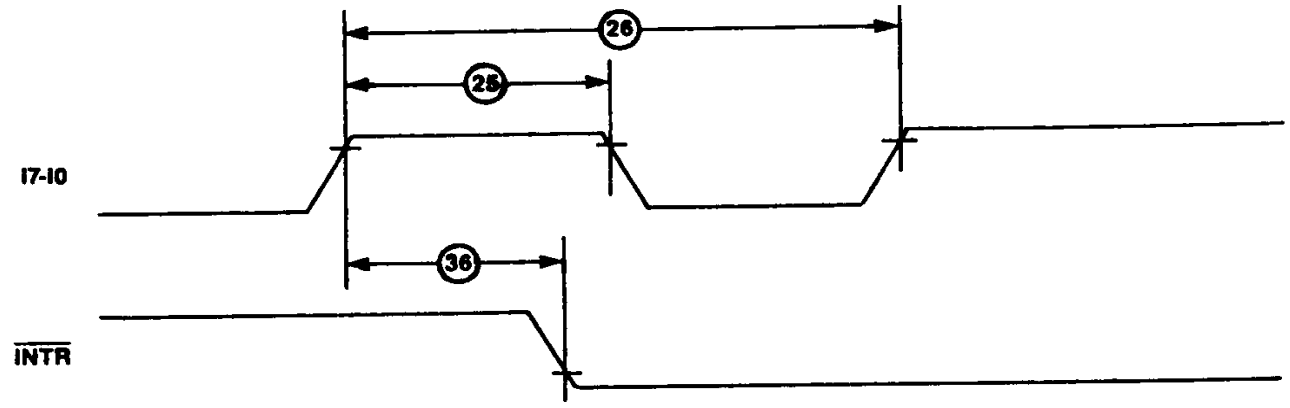


Figure 7-10. Multiplexed Bus Timing - Intel Type



Note: Active edge is assumed to be the rising edge

Figure 7-11. Interrupt Timing

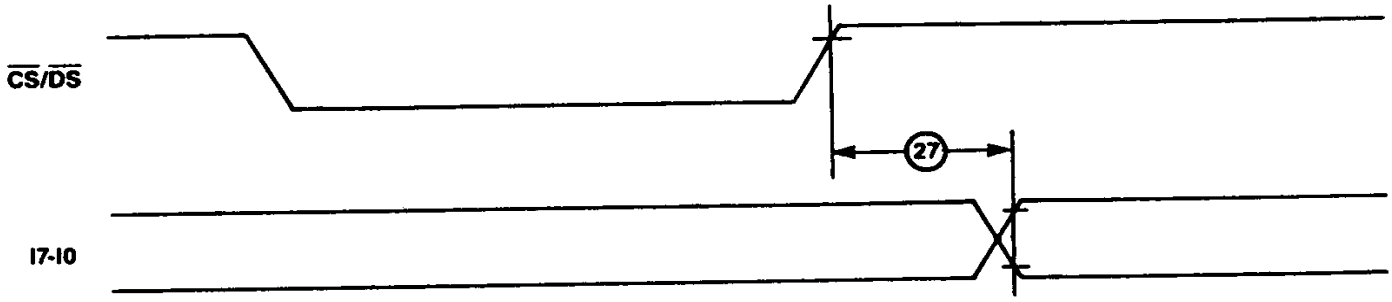


Figure 7-12. Port Timing

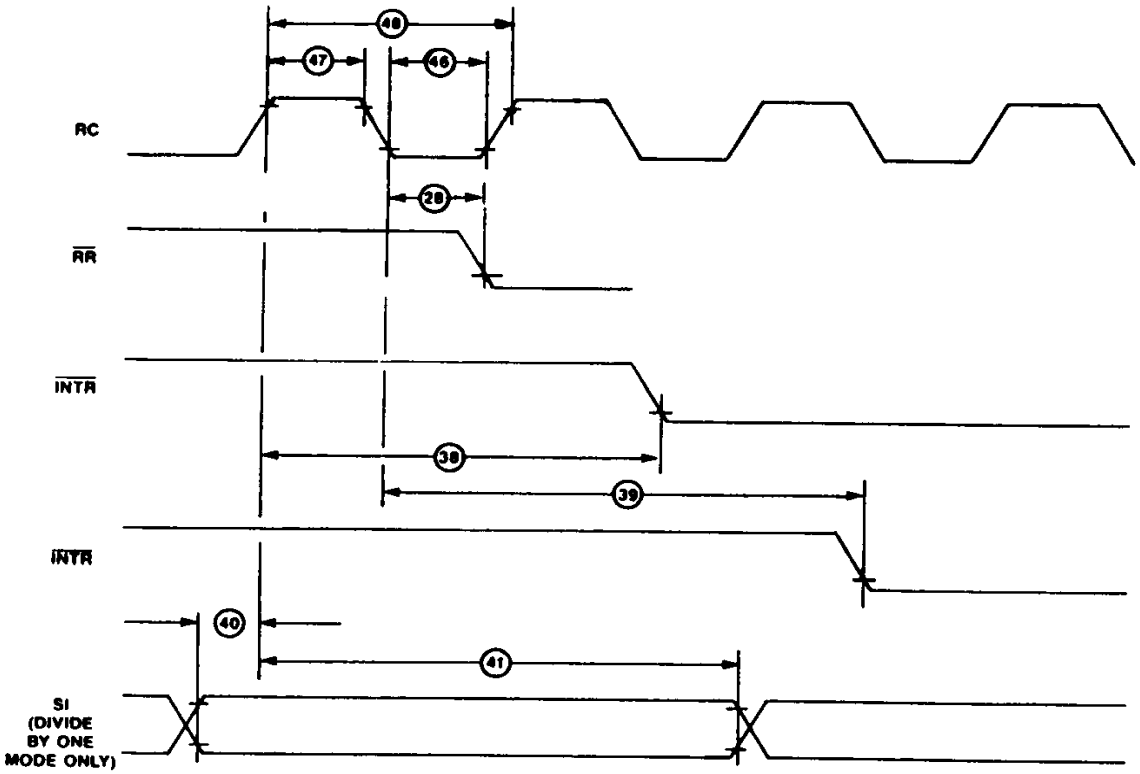


Figure 7-13. Receiver Timing

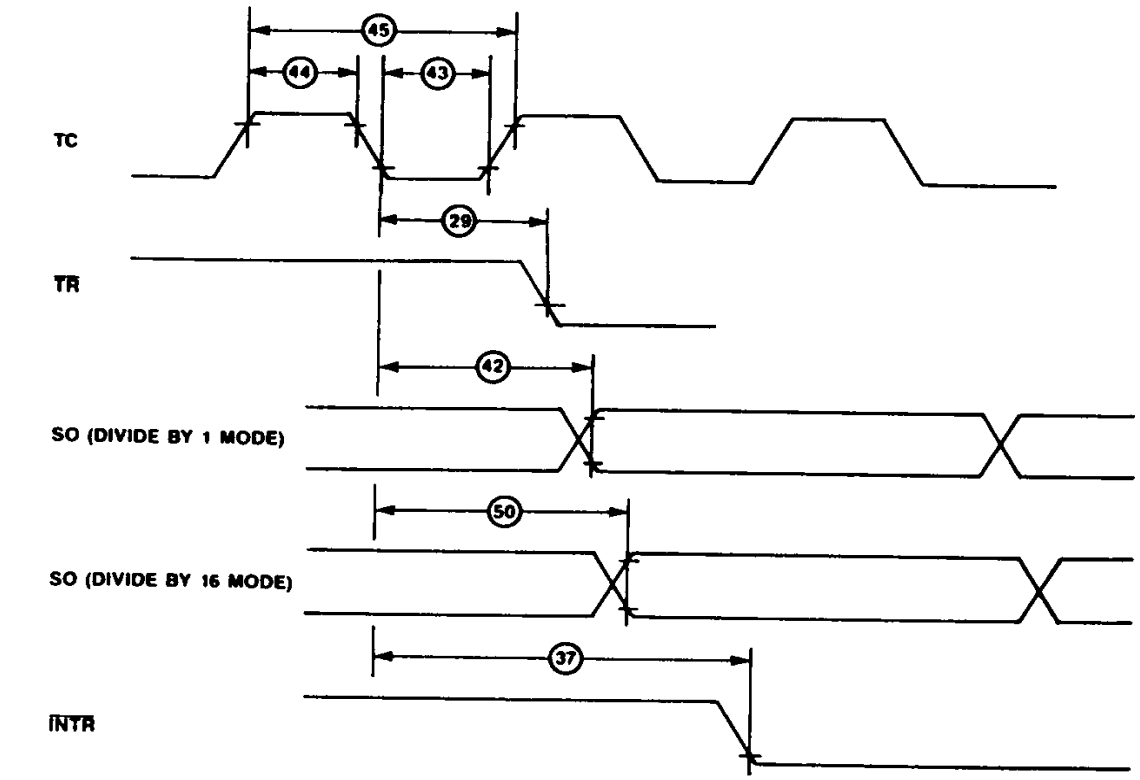


Figure 7-14. Transmitter Timing

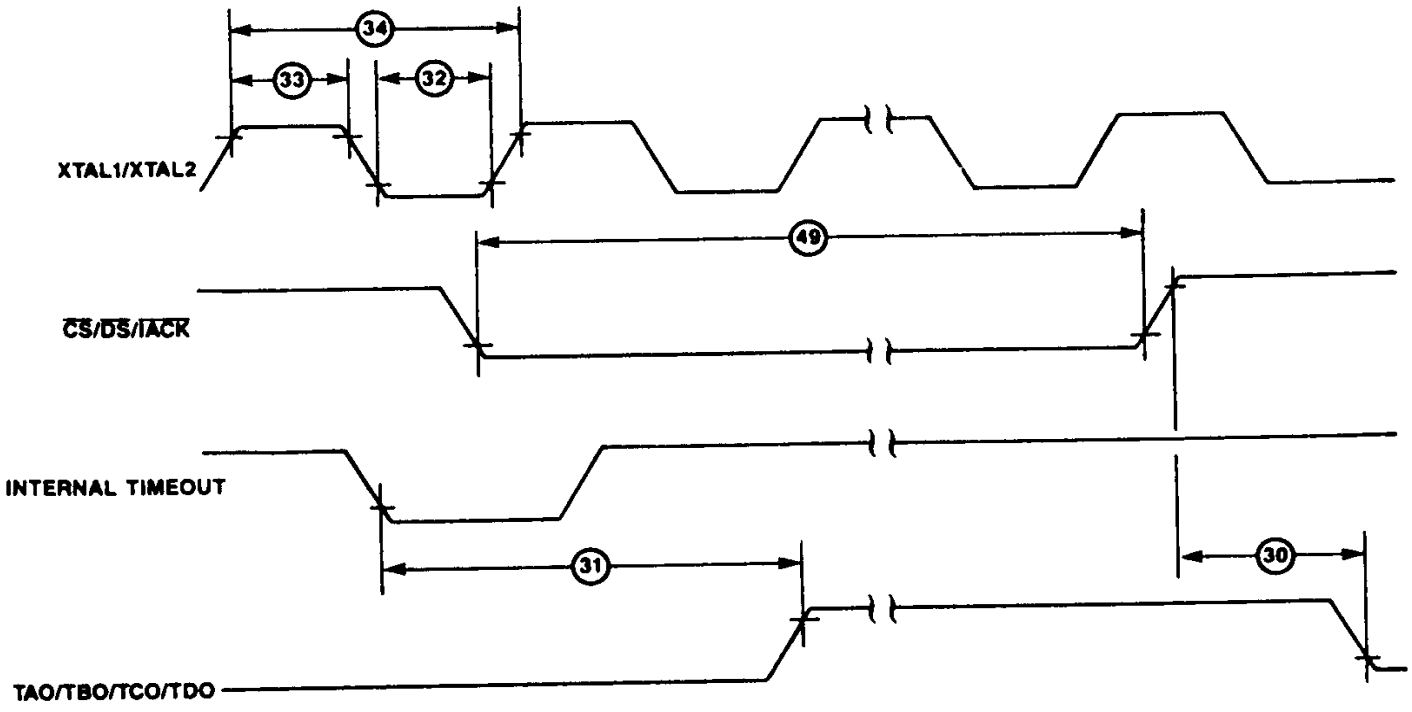


Figure 7-15. Timer Timing

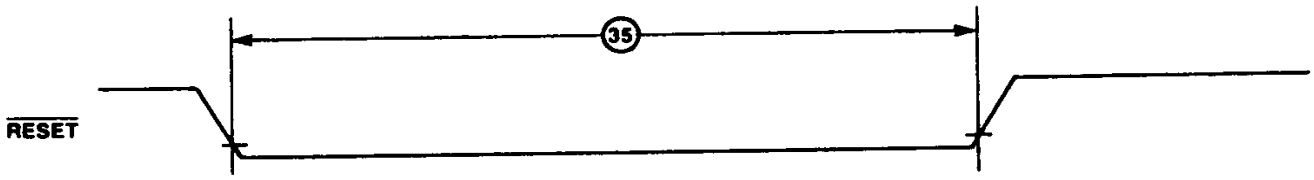


Figure 7-16. Reset Timing

7.8 TIMER AC CHARACTERISTICS

Definitions:

Error = Indicated time value − actual time value

$t_{psc} = t_{CLK} \times$ Prescale Value

Internal Timer Mode:

Single Interval Error (Free Running) (See Note 2)	± 100 ns
Cumulative Internal Error	0
Error Between Two Timer Reads	$\pm (t_{psc} - 4\ t_{CLK})$
Start Timer to Stop Timer Error	$2\ t_{CLK} + 100$ ns to $-(t_{psc} + 6\ t_{CLK} + 100$ ns)
Start Timer to Read Timer Error	0 to $-(t_{psc} + 6\ t_{CLK} + 400$ ns)
Start Timer to Interrupt Request Error (See Note 3)	$-2\ t_{CLK}$ to $-(4\ t_{CLK} + 800$ ns)

Pulse Width Measurement Mode:

Measurement Accuracy (See Note 1)	$2\ t_{CLK}$ to $-(t_{psc} + 4\ t_{CLK})$
Minimum Pulse Width	$4\ t_{CLK}$

Event Counter Mode:

Minimum Active Time of TAI and TBI	$4\ t_{CLK}$
Minimum Inactive Time of TAI and TBI	$4\ t_{CLK}$

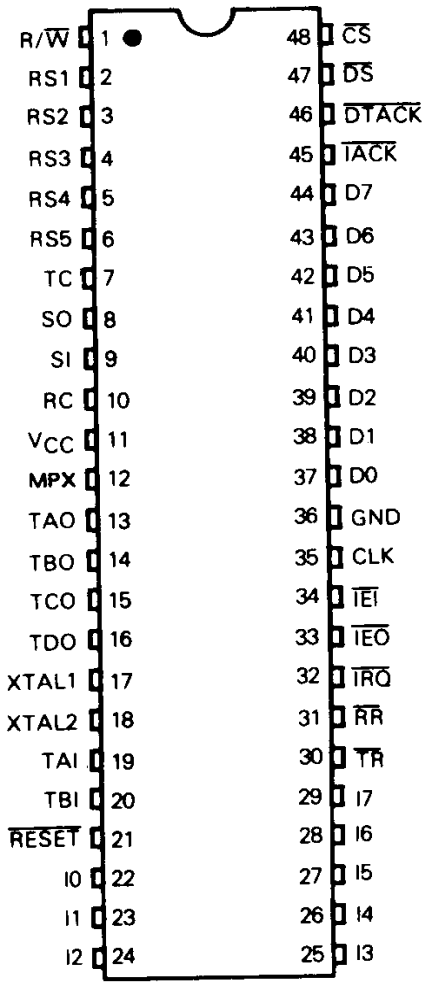
NOTES:

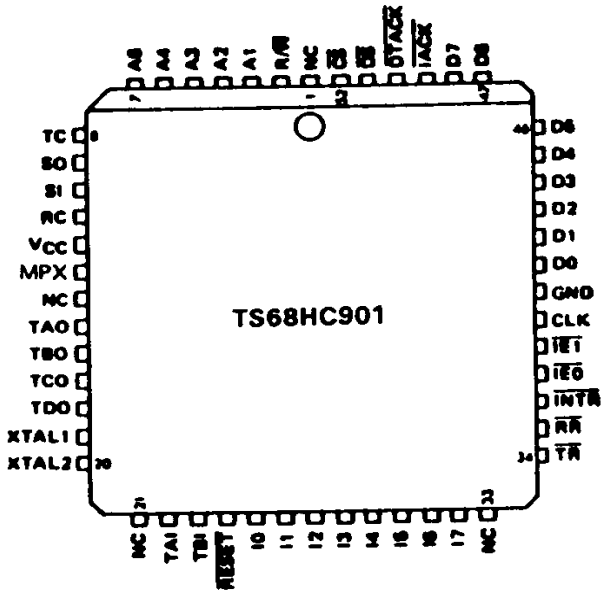
1. Error may be cumulative if repetitively performed.
2. Error with respect to t_{out} or $\overline{IRQ}$ if note 3 is true.
3. Assuming it is possible for the timer to make an interrupt request immediately.

SECTION 8
MECHANICAL DATA AND ORDERING INFORMATION

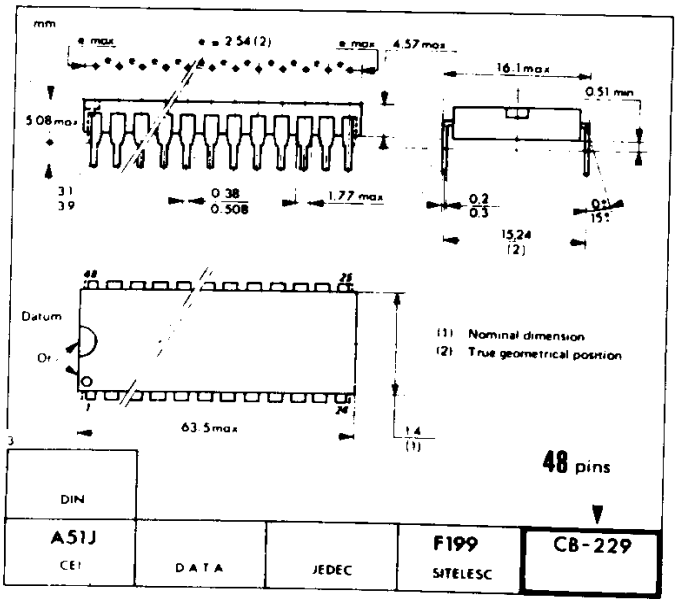
This section contains the pin assignments, package dimensions, and ordering information for the TS68HC901.

PIN ASSIGNMENTS

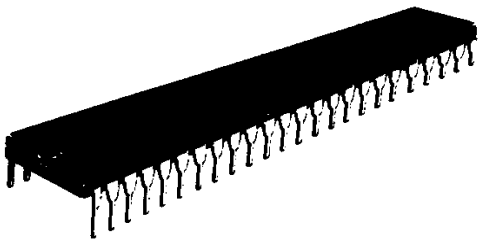




PHYSICAL DIMENSIONS

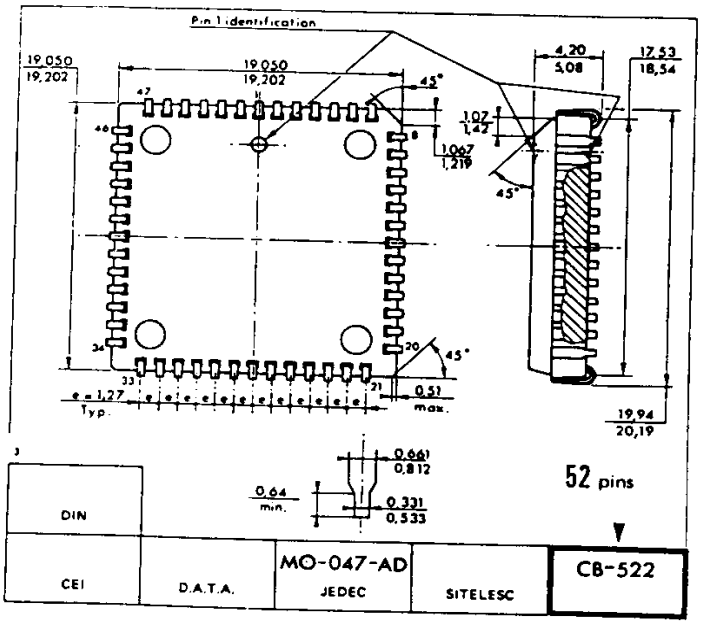


CB-229

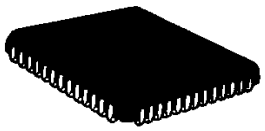


P SUFFIX
PLASTIC PACKAGE

ALSO AVAILABLE
C SUFFIX
CERAMIC PACKAGE



CB-522



FN SUFFIX
PLCC 52

ORDERING INFORMATION

STANDARD VERSIONS

Package Type	Frequency (MHz)	Temperature Range	Part Number
Ceramic DIL C Suffix	4.0 5.0 8.0	0°C to + 70°C 0°C to + 70°C 0°C to + 70°C	TS68HC901CC4 TS68HC901CC5 TS68HC901CC8
Plastic DIL P Suffix	4.0 5.0 8.0	0°C to + 70°C 0°C to + 70°C 0°C to + 70°C	TS68HC901CP4 TS68HC901CP5 TS68HC901CP8
PLCC FN Suffix	4.0 5.0 8.0	0°C to + 70°C 0°C to + 70°C 0°C to + 70°C	TS68HC901CFN4 TS68HC901CFN5 TS68HC901CFN8

Hi-REL VERSIONS

In order to fit more closely to customer specific requirements, THOMSON SEMICONDUCTEURS is proposing different screening levels for its Hi-REL ranges.

G/B screening : Available only from THOMSON SEMICONDUCTEURS, this quality level, very close to the MIL-STD-883, is a cost effective alternative for customers who want to buy Hi-REL devices (low guaranteed AQL). The G/B level is in full accordance with the NFC96883 class G.

B/B screening : Full accordance with the MIL-STD-883 Rev. C, class B (US), the CECC 90,000 class B (european) and with the NFC96883 class B (French).

Details on screening procedures for these levels of selection are available on request (please contact our sales representatives).

Package Type	Frequency (MHz)	Temperature Range	Part Number
Ceramic DIL C Suffix	4.0 4.0 4.0 4.0 4.0 4.0	— 40°C to + 85°C — 55°C to + 125°C — 40°C to + 85°C — 40°C to + 85°C — 55°C to + 125°C — 55°C to + 125°C	TS68HC901VC4 TS68HC901MC4 TS68HC901VCG/B4 TS68HC901VCB/B4 TS68HC901MCG/B4 TS68HC901MCB/B4