HD68230
PI/T (Parallel Interface Timer)

—PRELIMINARY—

The HD68230 Parallel Interface/Timer provides versatile double buffered parallel interfaces and an operating system oriented timer to HD68000 systems. The parallel interfaces operate in unidirectional or bidirectional modes, either 8 or 16 bits wide. In the unidirectional modes, an associated data direction register determines whether the port pins are inputs or outputs. In the bidirectional modes the data direction registers are ignored and the direction is determined dynamically by the state of four handshake pins. These programmable handshake pins provide an interface flexible enough for connection to a wide variety of low, medium, or high speed peripherals or other computer systems. The PI/T ports allow use of vectored or autovectored interrupts, and also provide a DMA Request pin for connection to the HD68450 Direct Memory Access Controller or a similar circuit. The PI/T timer contains a 24-bit wide counter and a 5-bit prescaler. The timer may be clocked by the system clock (PI/T CLK pin) or by an external clock (TIN pin), and a 5-bit prescaler can be used. It can generate periodic interrupts, a square wave, or a single interrupt after a programmed time period. Also it can be used for elapsed time measurement or as a device watchdog.

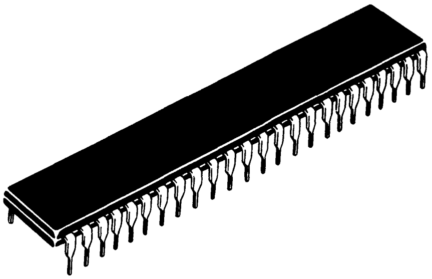
■ FEATURES

- HD68000 Bus Compatible
- Port Modes Include:
  - Bit I/O
  - Unidirectional 8-Bit and 16-Bit
  - Bidirectional 8-Bit and 16-Bit
- Selectable Handshaking Options
- 24-Bit Programmable Timer
- Software Programmable Timer Modes
- Contains Interrupt Vector Generation Logic
- Separate Port and Timer Interrupt Service Requests
- Registers are Read/Write and Directly Addressable
- Registers are Addressed for MOVEP (Move Peripheral) and DMAC Compatibility

■ TYPE OF PRODUCTS

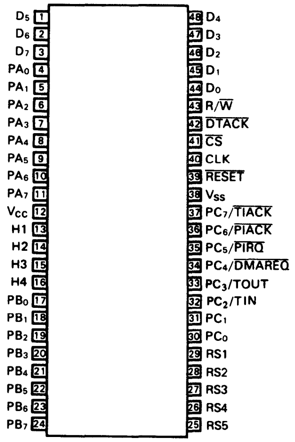
Type No.	Bus Timing
HD68230P-8	8 MHz
HD68230P-10	10 MHz

HD68230P8 HD68230P10



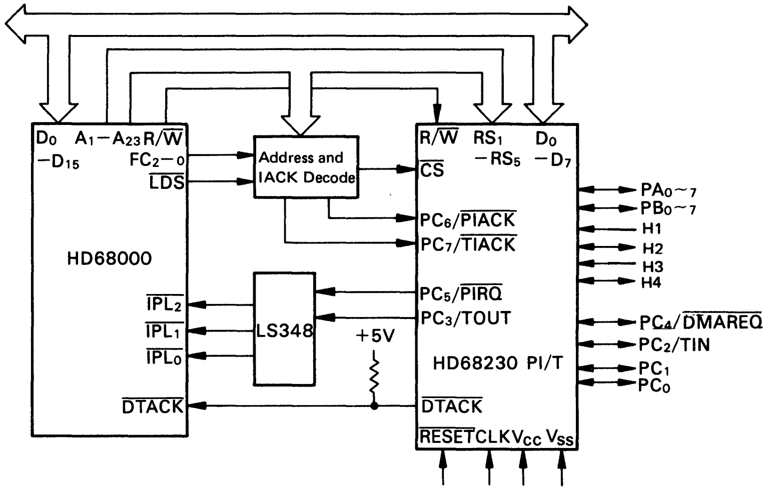
(DP-48)

■ PIN ARRANGMENT

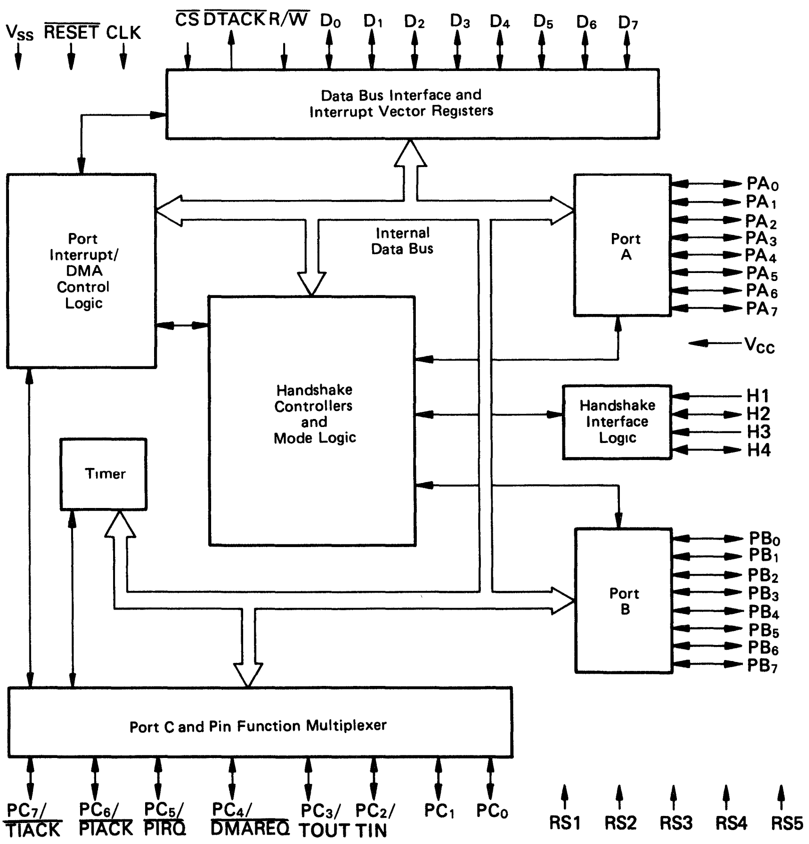


(Top View)

■ PI/T SYSTEM BLOCK DIAGRAM



■ BLOCK DIAGRAM


VSS RESET CLK
CS DTACK R/W D0 D1 D2 D3 D4 D5 D6 D7
Data Bus Interface and Interrupt Vector Registers
Port Interrupt/ DMA Control Logic
Internal Data Bus
Port A
PA0
PA1
PA2
PA3
PA4
PA5
PA6
PA7
VCC
Handshake Controllers and Mode Logic
Handshake Interface Logic
H1
H2
H3
H4
Timer
Port B
PB0
PB1
PB2
PB3
PB4
PB5
PB6
PB7
Port C and Pin Function Multiplexer
PC7/ TIACK
PC6/ PIACK
PC5/ PIRQ
PC4/ DMAREQ
PC3/ TOUT
PC2/ TIN
PC1
PC0
RS1 RS2 RS3 RS4 RS5

■ ABSOLUTE MAXIMUM RATINGS

Item	Symbol	Value	Unit
Supply Voltage	V_{CC}*	-0.3~+7.0	V
Input Voltage	V_{in}*	-0.3~+7.0	V
Operating Temperature Range	T_{opr}	0~+70	°C
Storage Temperature	T_{stg}	-55~+150	°C

* With respect to V_{SS} (SYSTEM GND)

(NOTE) 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_{SS} or V_{CC}).

■ RECOMMENDED OPERATING CONDITIONS

Item	Symbol	min	typ	max	Unit
Supply Voltage	V_{CC}*	4.75	5.0	5.25	V
Input Voltage	V_{IH}*	V_{SS} + 2.0	–	V_{CC}	V
	V_{IL}*	V_{SS} - 0.3	–	V_{SS} + 0.8	V
Operating Temperature	T_{opr}	0	25	70	°C

* With respect to V_{SS} (SYSTEM GND)

■ ELECTRICAL CHARACTERISTICS

● DC CHARACTERISTICS (V_{CC} = 5V ±5%, V_{SS} = 0V, Ta = 0 ~ +70°C unless otherwise noted.)

Item		Symbol	Test Condition	min	typ	max	Unit
Input "High" Voltage		V_{IH}		V_{SS}+2.0	–	V_{CC}	V
Input "Low" Voltage		V_{IL}		V_{SS}-0.3	–	V_{SS}+0.8	V
Input Leakage Current	H1, H3, R/$\overline{W}$, $\overline{RESET}$ CLK, RS1-RS5, $\overline{CS}$	I_{in}	Vin = 0 ~ 5.25V	–	–	10.0	µA
Three-State (Off State) Input Current	$\overline{DTACK}$, PC_0-PC_7, D_0-D_7	I_{TSI}	Vin = 0.4 ~ 2.4V	–		20	µA
	H2, H4, PA_0-PA_7, PB_0-PB_7			-0.1		-1.0	mA
Output "High" Voltage	D_0-D_7, $\overline{DTACK}$ PA_0-PA_7, PB_0-PB_7, H2, H4 PC_0-PC_7	V_{OH}	I_{OH} = -400µA I_{OH} = -150µA I_{OH} = -100µA	V_{SS}+2.4	–	–	V
Output "Low" Voltage	PC_3/TOUT, PC_5/$\overline{PIRQ}$ D_0-D_7, $\overline{DTACK}$ PA_0-PA_7, PB_0-PB_7, H2, H4 PC_0-PC_2, PC_4, PC_6, PC_7	V_{OL}	I_{OL} = 8.8mA I_{OL} = 5.3mA I_{OL} = 2.4mA I_{OL} = 2.4mA	–	–	0.5	V
Power Dissipation		P_{INT}	TA = 0°C	–	–	500	mW
Capacitance (Package Type Dependent)		C_{in}	V_{in} = 0V, Ta = 25°C	–	–	15	pF

● CLOCK TIMING

Characteristic	Symbol	8 MHz HD68230P-8		10 MHz HD68230P-10		Unit
		min	max	min	max	
Frequency of Operation	f	2.0	8.0	2.0	10.0	MHz
Cycle Time	t_{CYC}	125	500	100	500	ns
Clock Pulse Width	t_{CL}	55	250	45	250	ns
	t_{CH}	55	250	45	250	
Clock Rise and Fall Times	t_{Cr}	–	10	–	10	ns
	t_{Cf}	–	10	-	10	

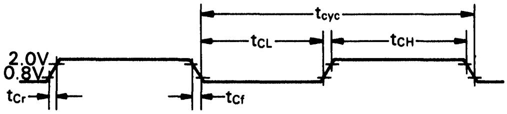


Figure 1 Input Clock Waveform

- **AC CHARACTERISTICS (V_{CC} = 5V ±5%, V_{SS} = 0V, Ta = 0 ~ +70°C unless otherwise noted.)**

No.	Characteristic	8 MHz HD68230P-8		10 MHz HD68230P-10		Unit
		min	max	min	max	
1	R/$\overline{\mathrm{W}}$ RS1-RS5 Valid to $\overline{\mathrm{CS}}$ Low (Setup Time)	0	–	0	–	ns
2(*11)	$\overline{\mathrm{CS}}$ Low to R/$\overline{\mathrm{W}}$ and RS1-RS5 Invalid (Hold Time)	100	–	65	–	ns
3(*1)	$\overline{\mathrm{CS}}$ Low to CLK Low (Setup Time)	30	–	20	–	ns
4(*2)	$\overline{\mathrm{CS}}$ Low to Data Out Valid (Delay)	–	75	–	60	ns
5	RS1-RS5 Valid to Data Out Valid (Delay)	–	140	–	100	ns
6	CLK Low to DTACK Low (Read/Write Cycle) (Delay)	0	70	0	60	ns
7(*3)	$\overline{\mathrm{DTACK}}$ Low to $\overline{\mathrm{CS}}$ High (Hold Time)	0	–	0	–	ns
8	$\overline{\mathrm{CS}}$ or $\overline{\mathrm{PIACK}}$ or $\overline{\mathrm{TIACK}}$ High to Data Out Invalid (Hold Time)	0	–	0	–	ns
9	$\overline{\mathrm{CS}}$ or $\overline{\mathrm{PIACK}}$ or $\overline{\mathrm{TIACK}}$ High to D_0-D_7 High Impedance (Delay)	–	50	–	45	ns
10	$\overline{\mathrm{CS}}$ or $\overline{\mathrm{PIACK}}$ or $\overline{\mathrm{TIACK}}$ High to $\overline{\mathrm{DTACK}}$ High (Delay)	–	50	–	30	ns
11	$\overline{\mathrm{CS}}$ or $\overline{\mathrm{PIACK}}$ or $\overline{\mathrm{TIACK}}$ High to $\overline{\mathrm{DTACK}}$ High Impedance (Delay)	–	100	–	55	ns
12	Data Invalid to $\overline{\mathrm{CS}}$ Low (Setup Time)	0	–	0	–	ns
13	$\overline{\mathrm{CS}}$ Low to Data in Invalid (Hold Time)	100	–	65	–	ns
14	Input Data Valid to H1(H3) Asserted (Setup Time)	100	–	60	–	ns
15	H1(H3) Asserted to Input Data Invalid (Hold Time)	20	–	20	–	ns
16	Handshake Input H1(H4) Pulse Width Asserted	40	–	40	–	ns
17	Handshake Input (H1-H4) Pulse Width Negated	40	–	40	–	ns
18	H1(H3) Asserted to H2(H4) Negated (Delay)	–	150	–	120	ns
19	CLK Low to H2(H4) Asserted (Delay)	–	100	–	100	ns
20(*4)	H2(H4) Asserted to H1(H3) Asserted	0	–	0	–	ns
21(*5)	CLK Low to H2(H4) Pulse Negated (Delay)	–	125	–	125	ns
22(*9,*12)	Synchronized H1(H3) to CLK Low on which $\overline{\mathrm{DMAREQ}}$ is Asserted (See Figures 18 and 19)	2.5	3.5	2.5	3.5	CLK Per
23	CLK Low $\overline{\mathrm{DMAREQ}}$ is Asserted to CLK Low on which $\overline{\mathrm{DMAREQ}}$ is Negated	3	3	3	3	CLK Per
24	CLK Low to Output Data Valid (Delay) (Modes 0, 1)	–	150	–	120	ns
25(*9,*12)	Synchronized H1(H3) to Output Data Invalid (Modes 0, 1)	1.5	2.5	1.5	2.5	CLK Per
26	H1 Negated to Output Data Valid (Modes 2, 3)	–	70	–	50	ns
27	H1 Asserted to Output Data High Impedance (Modes 2, 3)	0	70	0	70	ns
28	Read Data Valid to $\overline{\mathrm{DTACK}}$ Low (Setup Time)	0	–	0	–	ns
29	CLK Low to Data Output Valid (Interrupt Acknowledge Cycle)	–	120	–	100	ns
30(*7)	H1(H3) Asserted to CLK High (Setup Time)	50	–	40	–	ns
31	$\overline{\mathrm{PIACK}}$ or $\overline{\mathrm{TIACK}}$ Low to CLK Low (Setup Time)	50	–	40	–	ns
32(*12)	Synchronized $\overline{\mathrm{CS}}$ to CLK Low on which $\overline{\mathrm{DMAREQ}}$ is Asserted (See Figures 18 and 19)	3	3	3	3	CLK Per
33(*9,*12)	Synchronized H1(H3) to CLK Low on which H2(H4) is Asserted	3.5	4.5	3.5	4.5	CLK Per
34	CLK Low to $\overline{\mathrm{DTACK}}$ Low (Interrupt Acknowledge Cycle) (Delay)	–	75	–	60	ns
35	CLK Low to $\overline{\mathrm{DMAREQ}}$ Low (Delay)	0	120	0	100	ns
36	CLK Low to $\overline{\mathrm{DMAREQ}}$ High (Delay)	0	120	0	100	ns
–	CLK Low to $\overline{\mathrm{PIRQ}}$ Low or High Impedance	–	200	–	150	ns
–(*8)	TIN Frequency (External Clock) – Prescaler Used	0	1	0	1	Fclk(Hz)(6)
–	TIN Frequency (External Clock) – Prescaler Not used	0	1/32	0	1/32	Fclk(Hz)(6)
–	TIN Pulse Width High or Low (External Clock)	55	–	45	–	ns
–	TIN Pulse Width Low (Run/Halt Control)	1	–	1	–	CLK
–	CLK Low to TOUT High, Low, or High Impedance	0	200	0	150	ns
–	$\overline{\mathrm{CS}}$, $\overline{\mathrm{PIACK}}$, or $\overline{\mathrm{TIACK}}$ High to $\overline{\mathrm{CS}}$, $\overline{\mathrm{PIACK}}$, or $\overline{\mathrm{TIACK}}$ Low	50	–	30	–	ns

(NOTES)(*1) This specification only applies if the PI/T had completed all operations initiated by the previous bus cycle when $\overline{\text{CS}}$ was asserted. Following a normal read or write bus cycle, all operations are complete within three CLKs after the falling edge of the CLK pin on which $\overline{\text{DTACK}}$ was asserted. If $\overline{\text{CS}}$ is asserted prior to completion of these operations, the new bus cycle, and hence, $\overline{\text{DTACK}}$ is postponed.
If all operations of the previous bus cycle were complete when $\overline{\text{CS}}$ was asserted, this specification is made only to insure that $\overline{\text{DTACK}}$ is asserted with respect to the falling edge of the CLK pin as shown in the timing diagram, not to guarantee operation of the part. If the $\overline{\text{CS}}$ setup time is violated, $\overline{\text{DTACK}}$ may be asserted as shown, or may be asserted one clock cycle later.

(*2) Assuming the RS1-RS5 to Data Valid Time has also expired.

(*3) This specification imposes a lower bound on $\overline{\text{CS}}$ low time, guaranteeing that $\overline{\text{CS}}$ will be low for at least 1 CLK period.

(*4) This specification assures recognition of the asserted edge of H1(H3).

(*5) This specification applies only when a pulsed handshake option is chosen and the pulse is not shortened due to an early asserted edge of H1 (H3).

(*6) CLK refers to the actual frequency of the CLK pin, not the maximum allowable CLK frequency.

(*7) If timing number 30 is violated, H1(H3) will be recognized no later than the next rising edge of the clock.

(*8) This limit applies to the frequency of the signal at TIN compared to the frequency of the CLK signal during each clock cycle.
If any period of the waveform at TIN is smaller than the period of the CLK signal at that instant, then it is likely that the timer circuit will completely ignore one cycle of the TIN signal. Since the frequency measured by a frequency counter is the average frequency of a signal over a specific length of time, the actual frequency at any one time will vary above and below the average. These variations occur in both the TIN and CLK signals.
If these two signals are derived from different sources they will have different instantaneous frequency variations. In this case the frequency applied to the TIN pin must be distinctly less than the frequency at the CLK pin to avoid lost cycles of the TIN signal. Measurements have shown that with signals derived from different crystal oscillators applied to the TIN and CLK pins with fast rise and fall times. The TIN frequency can approach 80 to 90% of the frequency of the CLK signal without a loss of a cycle of the TIN signal.
If these two signals are derived from the same frequency source then the frequency of the signal applied to TIN can be 100% of the frequency at the CLK pin. They may be generated by different buffers from the same signal or one may be an inverted version of the other.
The TIN signal may be generated by an 'AND' function of the clock and a control signal.

(*9) This limit applies in every case. There are no exceptions to this specification.

(*10) If a bus access and peripheral access occur at the same time, add one clock to specifications 22 and 33.

(*11) See BUS INTERFACE CONNECTION section for exception.

(*12) This Limit specifies the nominal outputing in PI/T clock cycles. To obtain the output timing in nanoseconds, add or subtract the appropriate setup time and/or propagation time from the signals to the respective clock edges.

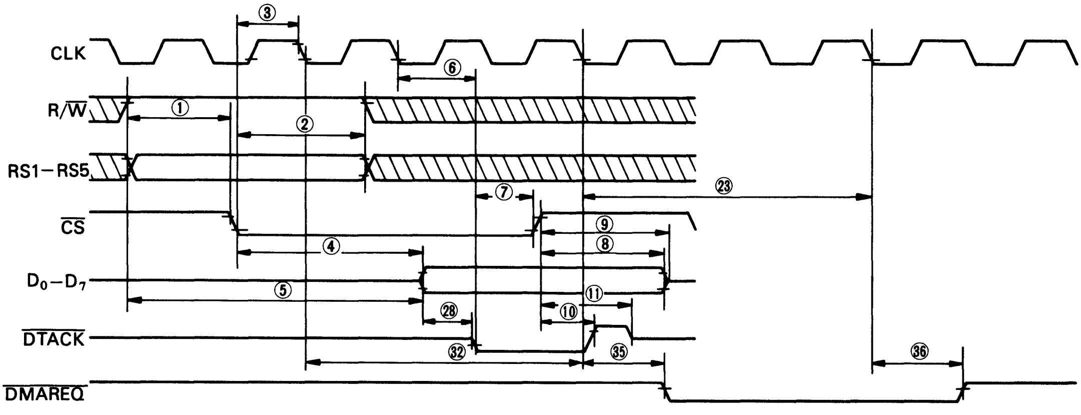


Figure 2 Bus Read Cycle Timing

NOTE: Timing measurements are referenced to and from a low voltage of 0.8 volts and a high voltage of 2.0 volts, unless otherwise noted.

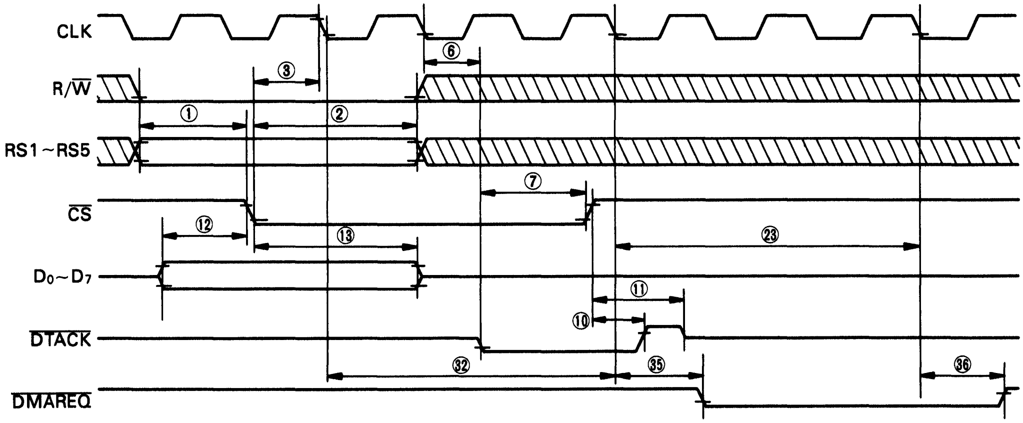


Figure 3 Bus Write Cycle Timing

NOTE: Timing measurements are referenced to and from a low voltage of 0.8 volts and a high voltage of 2.0 volts, unless otherwise noted.

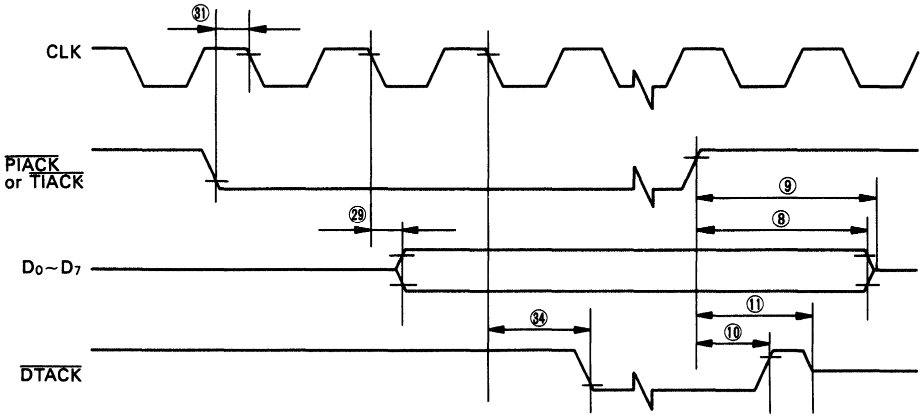


Figure 4 Interrupt Acknowledge Functional Timing Diagram

NOTE: Timing measurements are referenced to and from a low voltage of 0.8 volts and a high voltage of 2.0 volts, unless other wise noted.

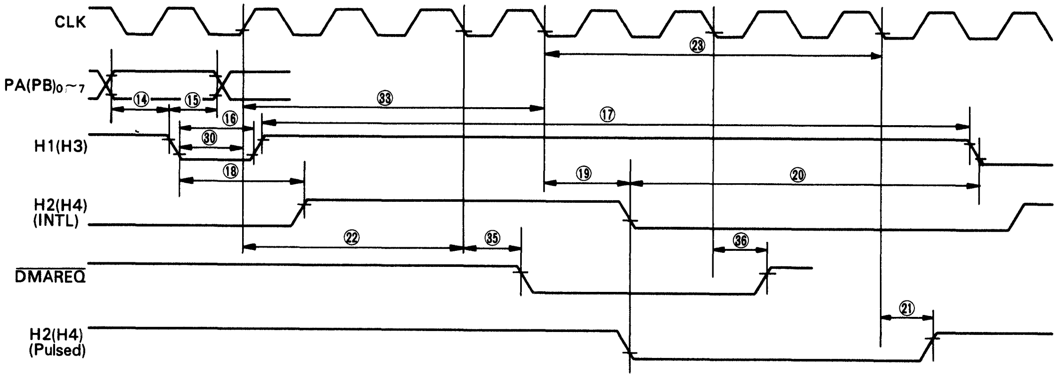


Figure 5 Peripheral Interface Input Timing

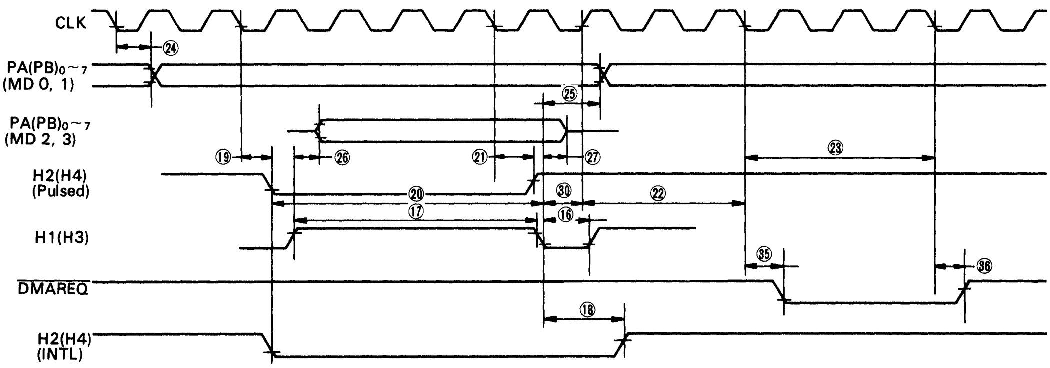


Figure 6 Peripheral Interface Output Timing

■ GENERAL DESCRIPTION

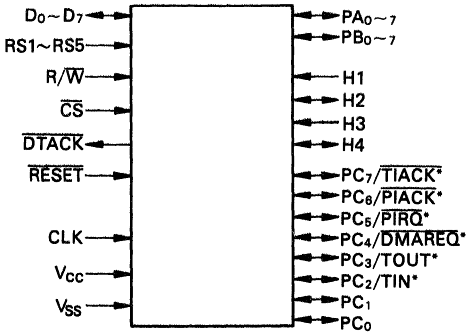
The PI/T consists of two logically independent sections: the ports and the timer. The port section consists of Port A (PA_{0-7}), Port B (PB_{0-7}), four handshake pins (H1, H2, H3, and H4), two general I/O pins, and six dual-function pins. The dual-function pins can individually operate as a third port (Port C) or an alternate function related to either Ports A and B, or the timer. The four programmable handshake pins, depending on the mode, can control data transfer to and from the ports, or can be used as interrupt generating inputs, or I/O pins.

The timer consists of a 24-bit counter, optionally clocked by a 5-bit prescaler. Three pins provide complete timer I/O: PC_2/TIN, PC_3/TOUT, and $PC_7/\overline{\text{TIACK}}$. Of course, only the ones needed for the given configuration perform the timer function, while the others remain Port C I/O.

The system bus interface provides for asynchronous transfer of data from the PI/T to a bus master over the data bus (D_0-D_7). Data transfer acknowledge ($\overline{\text{DTACK}}$), register selects (RS1-RS5), chip select, the read/write line (R/$\overline{\text{W}}$), and Port Interrupt Acknowledge ($\overline{\text{PIACK}}$) or Timer Interrupt Acknowledge ($\overline{\text{TIACK}}$) control data transfer between the PI/T and the HD68000.

■ PIN DESCRIPTION

Throughout the data sheet, signals are presented using the terms active and inactive or asserted and negated independent of whether the signal is active in the high-voltage state or low-voltage state. (The active state of each logic pin is given below.) Active low signals are denoted by a superscript bar. R/$\overline{\text{W}}$ indicates a write is active low and a read active high.


D_0~D_7
RS1~RS5
R/$\overline{\text{W}}$
$\overline{\text{CS}}$
$\overline{\text{DTACK}}$
$\overline{\text{RESET}}$
CLK
V_{CC}
V_{SS}
PA_0~7
PB_0~7
H1
H2
H3
H4
$PC_7/\overline{\text{TIACK}}$*
$PC_6/\overline{\text{PIACK}}$*
$PC_5/\overline{\text{PIRQ}}$*
$PC_4/\overline{\text{DMAREQ}}$*
PC_3/TOUT*
PC_2/TIN*
PC_1
PC_0

*Individually Programmable Dual-Function Pin

Figure 7 Input and Output Signals

Bidirectional Data Bus – (D_0-D_7)

The data bus pins D_0-D_7 form an 8-bit bidirectional data bus to/from the HD68000 or other bus master. These pins are active high.

Register Selects – (RS1-RS5)

RS1-RS5 are active high high-impedance inputs that determine which of the 25 possible registers is being addressed. They are provided by the HD68000 or other bus master.

Read/Write Input – (R/$\overline{\text{W}}$)

R/$\overline{\text{W}}$ is the high-impedance Read/Write signal from the HD68000 or bus master, indicating whether the current bus cycle is a read (high) or write (low) cycle.

Chip Select Input – ($\overline{\text{CS}}$)

$\overline{\text{CS}}$ is a high-impedance input that selects the PI/T registers for the current bus cycle. Address strobe and the data strobe (upper or lower) of the bus master, along with the appropriate address bits, must be included in the chip select equation. A low level corresponds to an asserted chip select.

Data Transfer Acknowledge Output – ($\overline{\text{DTACK}}$)

$\overline{\text{DTACK}}$ is an active low output that signals the completion of the bus cycle. During read or interrupt acknowledge cycles, $\overline{\text{DTACK}}$ is asserted by the HD68230 after data has been provided on the data bus; during write cycles it is asserted after data has been accepted at the data bus. Data transfer acknowledge is compatible with the HD68000 and with other Hitachi bus masters such as the HD68450 DMA controller. A holding resistor is required to maintain $\overline{\text{DTACK}}$ high between bus cycles.

Reset Input – ($\overline{\text{RESET}}$)

$\overline{\text{RESET}}$ is a high-impedance input used to initialize all PI/T functions. All control and data direction registers are cleared and most internal operations are disabled by the assertion of $\overline{\text{RESET}}$ (low).

Clock Input – (CLK)

The clock pin is a high-impedance TTL-compatible signal with the same specifications as the HD68000. The PI/T contains dynamic logic throughout, and hence this clock must not be gated off at any time. It is not necessary that this clock maintain any particular phase relationship with the HD68000 clock. It may be connected to an independent frequency source (faster or slower) as long as all bus specifications are met.

Port A and Port B – (PA_0-PA_7 and PB_0-PB_7)

Ports A and B are 8-bit ports that may be concatenated to form a 16-bit port in certain modes. The ports may be controlled in conjunction with the handshake pins H1-H4. For stabilization during system power-up, Ports A and B have internal pullup resisters to V_{CC}. All port pins are active high.

Handshake pins (I/O depending on the Mode and Submode) – (H1-H4)

Handshake pins H1-H4 are multi-purpose pins that (depending on the operational model) may provide an interlocked handshake, a pulsed handshake, an interrupt input (independent of data transfers), or simple I/O pins. For stabilization during system power-up, H2 and H4 have internal pullup resisters to V_{CC}. Their sense (active high or low) may be programmed in the Port General Control Register bits 3-0. Independent of the mode, the instantaneous level of the handshake pins can be read from the Port Status Register.

(PC_0-PC_7 /Alternate function) – (Port C)

This port can be used as eight general purpose I/O pins (PC_0-PC_7) or any combination of six special function pins and two general purpose I/O pins (PC_0-PC_1). (Each dual function pin can be standard I/O or a special function independent of the other port C pins.) The dual function pins are defined in the following paragraphs. When used as port C pin, these pins are active high. They may be individually programmed as inputs or outputs by the Port C Data Direction Register.

The alternate functions (TIN, TOUT, and $\overline{\text{TIACK}}$) are timer I/O pins. TIN may be used as a rising-edge triggered external clock input or an external run/halt control pin (the timer is in the run state if run/halt is high and in the halt state if run/halt is low). TOUT may provided an active low timer interrupt request output or a general-purpose square-wave output, initially high. $\overline{\text{TIACK}}$ is an active low high-impedance input used for timer interrupt acknowledge.

Port A and B functions have an independent pair of active low interrupt request ($\overline{\text{PIRQ}}$) and interrupt acknowledge ($\overline{\text{PIACK}}$) pins.

The $\overline{\text{DMAREQ}}$ (Direct Memory Access Request) pin provides an active low Direct Memory Access Controller (DMAC) request pulse of 3 clock cycles, completely compatible with the HD68450 DMAC.

REGISTER MODEL

A register model that includes the corresponding Register Selects is shown in Table 1.

Table 1 Register Model

Register Select Bits													
5	4	3	2	1	7	6	5	4	3	2	1	0	
0	0	0	0	0	Port Mode Control		H34 Enable	H12 Enable	H4 Sense	H3 Sense	H2 Sense	H1 Sense	Port General Control Register
0	0	0	0	1	*	SVCRQ Select		Interrupt PFS		Port Interrupt Priority Control			Port Service Request Register
0	0	0	1	0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Port A Data Direction Register
0	0	0	1	1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Port B Data Direction Register
0	0	1	0	0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Port C Data Direction Register
0	0	1	0	1	Interrupt Vector Number						*	*	Port Interrupt Vector Register
0	0	1	1	0	Port A Submode		H2 Control			H2 Int Enable	H1 SVCRQ Enable	H1 Stat Ctrl	Port A Control Register
0	0	1	1	1	Port B Submode		H4 Control			H4 Int Enable	H3 SVCRQ Enable	H3 Stat Ctrl	Port B Control Register
0	1	0	0	0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Port A Data Register
0	1	0	0	1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Port B Data Register
0	1	0	1	0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Port A Alternate Register
0	1	0	1	1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Port B Alternate Register
0	1	1	0	0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Port C Data Register
0	1	1	0	1	H4 Level	H3 Level	H2 Level	H1 Level	H4S	H3S	H2S	H1S	Port Status Register
0	1	1	1	0	*	*	*	*	*	*	*	*	(null)
0	1	1	1	1	*	*	*	*	*	*	*	*	(null)
1	0	0	0	0	TOUT/$\overline{\text{TIACK}}$ Control			ZD Ctrl	*	Clock Control		Timer Enable	Timer Control Register
1	0	0	0	1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Timer Interrpt Vector Register
1	0	0	1	0	*	*	*	*	*	*	*	*	(null)
1	0	0	1	1	Bit 23	Bit 22	Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit 16	Counter Preload Register (High)
1	0	1	0	0	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	(Mid)
1	0	1	0	1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	(Low)
1	0	1	1	0	*	*	*	*	*	*	*	*	(null)
1	0	1	1	1	Bit 23	Bit 22	Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit 16	Count Register (High)
1	1	0	0	0	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	(Mid)
1	1	0	0	1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	(Low)
1	1	0	1	0	*	*	*	*	*	*	*	ZDS	Timer Status Register
1	1	0	1	1	*	*	*	*	*	*	*	*	(null)
1	1	1	0	0	*	*	*	*	*	*	*	*	(null)
1	1	1	0	1	*	*	*	*	*	*	*	*	(null)
1	1	1	1	0	*	*	*	*	*	*	*	*	(null)
1	1	1	1	1	*	*	*	*	*	*	*	*	(null)

*Unused, read as zero.

PROGRAMMER'S MODEL

Table 2 PI/T Register Addressing Assignments

Register	Register Select Bits					Accessible	Affected by Reset	Affected by Read Cycle
	5	4	3	2	1			
Port General Control Register (PGCR)	0	0	0	0	0	R W	Yes	No
Port Service Request Register (PSRR)	0	0	0	0	1	R W	Yes	No
Port A Data Direction Register (PADDR)	0	0	0	1	0	R W	Yes	No
Port B Data Direction Register (PBDDR)	0	0	0	1	1	R W	Yes	No
Port C Dara Direction Register (PCDDR)	0	0	1	0	0	R W	Yes	No
Port Interrupt Vector Register (PIVR)	0	0	1	0	1	R W	Yes	No
Port A Control Register (PACR)	0	0	1	1	0	R W	Yes	No
Port B Control Register (PBCR)	0	0	1	1	1	R W	Yes	No
Port A Data Register (PADR)	0	1	0	0	0	R W	No	**
Port B Data Register (PBDR)	0	1	0	0	1	R W	No	**
Port A Alternate Register (PAAR)	0	1	0	1	0	R	No	No
Port B Alternate Register (PBAR)	0	1	0	1	1	R	No	No
Port C Data Register (PCDR)	0	1	1	0	0	R W	No	No
Port Status Register (PSR)	0	1	1	0	1	R W*	Yes	No
Timer Control Register (TCR)	1	0	0	0	0	R W	Yes	No
Timer Interrupt Vector Register (TIVR)	1	0	0	0	1	R W	Yes	No
Counter Preload Register High (CPRH)	1	0	0	1	1	R W	No	No
Counter Preload Register Middle (CPRM)	1	0	1	0	0	R W	No	No
Counter Preload Register Low (CPRL)	1	0	1	0	1	R W	No	No
Count Register High (CNTRH)	1	0	1	1	1	R	No	No
Count Register Middle (CNTRM)	1	1	0	0	0	R	No	No
Count Register Low (CNTRL)	1	1	0	0	1	R	No	No
Timer Status Register (TSR)	1	1	0	1	0	RW*	Yes	No

* A write to this register may perform a special status resetting operation.
** Mode dependent.

R = Read
W = Write

The internal accessible register organization is represented in Table 2. Address space within the address map is reserved for future expansion. Throughout the PI/T data sheet the following conventions are maintained:

(1) A read from a reserved location in the map results in a read from the "null register." The null register returns all zeros for data and results in a normal bus cycle. A write to one of these locations results in a normal bus cycle but no write occurs.

(2) Unused bits of a defined register are denoted by "•" and are read as zeroes.

(3) Bits that are unused in the chosen mode/submode but are used in others, are denoted by "X", and are readable and writeable. Their content, however, is ignored in the chosen mode/submode.

(4) All registers are addressable as 8-bit quantities. To facilitate operation with the MOVEP instruction and the DMAC, addresses are ordered such that certain sets of registers may also be accessed as words (2 bytes) or long words (4 bytes).

- **Port General Control Register (PGCR)**

Port General Control Register (PGCR)

7	6	5	4	3	2	1	0
Port Mode Control		H34 Enable	H12 Enable	H4 Sense	H3 Sense	H2 Sense	H1 Sense

The Port General Control Register (PGCR) controls may of the functions that are common to the overall operation of the ports. The PGCR is composed of three major fields; bits 7 and 6 define the operational mode of Ports A and B and affect operation of the handshake pins and status bits; bits 5 and 4 allow a software controlled disabling of particular hardware associated with the handshake pins of each port; and bits 3-0 define the sense of the handshake pins. The PGCR is always readable and writeable.

All bits are reset to 0 when the $\overline{\text{RESET}}$ pin is asserted.

The Port Mode Control field should be altered only when the H12 Enable and H34 Enable bits are 0. Except when Mode 0 is desired, the Port General Control register must be written once to establish the mode, and again to enable the respective operation(s).

PGCR 7 6	Port Mode Control
0 0	Mode 0 (Unidirectional 8-Bit Mode)
0 1	Mode 1 (Unidirectional 16-Bit Mode)
1 0	Mode 2 (Bidirectional 8-Bit Mode)
1 1	Mode 3 (Bidirectional 16-Bit Mode)

PGCR 5	H34 Enable
0	Disabled
1	Enabled

PGCR 4	H12 Enable
0	Disabled
1	Enabled

PGCR 3-0	Handshake Pin Sense
0	The associated pin is at the high-voltage level when negated and at the low-voltage level when asserted.
1	The associated pin is at the low-voltage level when negated and at the high voltage level when asserted.

- **Port Service Request Register (PSRR)**

Port Service Request Register (PSRR)

7	6	5	4	3	2	1	0
*	SVCRQ Select		Interrupt PFS		Port Interrupt Priority Control		

The Port Service Request Register (PSRR) controls other functions that are common to the overall operation to the ports. It is composed of four major fields; bit 7 is unused and is always read as 0; bits 6 and 5 define whether interrupt or DMA requests are generated from activity on the H1 and H3 handshake pins; bit 4 and 3 determine whether two dual function pins operate as Port C or port interrupt request/acknowledge pins; and bits 2, 1, and 0 control the priority among all port interrupt sources. Since bits 2, 1, and 0 affect interrupt operation, it is recommended that they be changed only when the affected interrupt(s) is (are) disabled or known to remain inactive. The PSRR is always readable and writeable.

All bits are reset to 0 when the $\overline{\text{RESET}}$ pin is asserted.

PSRR 6 5	SVCRQ Select
0 X	The PC$_4$/$\overline{\text{DMAREQ}}$ pin carries the PC$_4$ function; DMA is not used.
1 0	The PC$_4$ /$\overline{\text{DMAREQ}}$ pin carries the $\overline{\text{DMAREQ}}$ function and is associated with duouble-buffered transfers controlled by H1. H1 is removed from the PI/T's interrupt structure, and thus, does not cause interrupt requests to be generated. To obtain $\overline{\text{DMAREQ}}$ pulses, Port A Control Register bit 1 (H1 SVCRQ Enable) must be a 1.
1 1	The PC$_4$ /$\overline{\text{DMAREQ}}$ pin carries the $\overline{\text{DMAREQ}}$ function and is associated with double-buffered transfers controlled by H3. H3 is removed from the PI/T's interrupt structure, and thus, does not cause interrupt requests to be generated. To obtain $\overline{\text{DMAREQ}}$ pulses, Port B Control Register bit 1 (H3 SVCRQ Enable) must be a 1.

PSRR 4 3	Interrupt Pin Function Select
0 0	The PC$_5$/$\overline{\text{PIRQ}}$ pin carries the PC5 function. The PC$_6$/$\overline{\text{PIACK}}$ pin carries the PC6 function.
0 1	The PC$_5$/$\overline{\text{PIRQ}}$ pin carries the $\overline{\text{PIRQ}}$ function. The PC$_6$/$\overline{\text{PIACK}}$ pin carries the PC6 function.
1 0	The PC$_5$/$\overline{\text{PIRQ}}$ pin carries the PC5 function. The PC$_6$/$\overline{\text{PIACK}}$ pin carries the $\overline{\text{PIACK}}$ function.
1 1	The PC$_5$/$\overline{\text{PIRQ}}$ pin carries the $\overline{\text{PIRQ}}$ function. The PC$_6$/$\overline{\text{PIACK}}$ pin carries the $\overline{\text{PIACK}}$ function.

Bits 2, 1, and 0 determine port interrupt priority. The priority is shown in descending order left to right.

PSRR	Port Interrupt Priority Control			
2 1 0	Highest	....		Lowest
0 0 0	H1S	H2S	H3S	H4S
0 0 1	H2S	H1S	H3S	H4S
0 1 0	H1S	H2S	H4S	H3S
0 1 1	H2S	H1S	H4S	H3S
1 0 0	H3S	H4S	H1S	H2S
1 0 1	H3S	H4S	H2S	H1S
1 1 0	H4S	H3S	H1S	H2S
1 1 1	H4S	H3S	H2S	H1S

- **Port A Data Direction Register (PADDR)**

The Port A Data Direction Register (PADDR) determines the direction and buffering characteristics of each of the Port A pins. One bit in the PADDR is assigned to each pin. A 0 indicates that the pin is used as an input, while a 1 indicates it is used as an output. The PADDR is always readable and writeable. This register is ignored in Mode 3.

All bits are reset to the 0 (input) state when the $\overline{\text{RESET}}$ pin is asserted.

- **Port B Data Direction Register (PBDDR)**

The PBDR is identical to the PADDR for the Port B pins and the Port B Data Register, except that this register is ignored in Modes 2 and 3.

- **Port C Data Direction Register (PCDDR)**

The Port C Data Direction Register (PCDDR) specifies whether each dual-function pin that is chosen for Port C operation is an input (0) or an output (1) pin. The PCDDR, along with bits that determine the respective pin's function, also specify the exact hardware to be accessed at the Port C Data Register address. (See the Port C Data Register description for more details.) The PCDDR is an 8-bit register that is readable and writeable at all times. Its operation is independent of the chosen PI/T mode.

These bits are cleared to 0 when the $\overline{\text{RESET}}$ pin is asserted.

- **Port Interrupt Vector Register (PIVR)**

Port Interrupt Vector Register (PIVR)

7	6	5	4	3	2	1	0
Interrupt Vector Number						*	*

The Port Interrupt Vector Register (PIVR) contains the upper order six bits of the four port interrupt vectors. The contents of this register may be read two ways; by an ordinary read cycle, or by a port interrupt acknowledge bus cycle. The exact data read depends on how the cycle was initiated and other factors. Behavior during a port interrupt acknowledge cycle is summarized above in Table 5.

From a normal read cycle ($\overline{CS}$), there is never a consequence to reading this register. Following negation of the $\overline{RESET}$ pin, but prior to writing to the PIVR, a $OF will be read. After writing to the register, the upper 6 bits may be read and the lower 2 bits are forced to 0. No prioritization computation is performed.

- **Port A Control Register (PACR)**

Port A Control Register (PACR)

7	6	5	4	3	2	1	0
Port A Submode		H2 Control			H2 Int. Enable	H1 SVCRQ Enable	H1 Stat. Ctrl.

The Port A Control Register (PACR) in conjunction with the programmed mode and the Port B submode, control the operation of Port A and the handshake pins H1 and H2. The Port A Control Register contains five fields; bits 7 and 6 specify the Port A submode; bits 5, 4, and 3 control the operation of the H2 handshake pin and H2S status bit; bit 2 determines whether an interrupt will be generated when the H2S status bit goes to 1; bit 1 determines whether a service request (interrupt request or DMA request) will occur; bit 0 controls the operation of the H1S status bit. The PACR is always readable and writeable.

All bits are cleared to 0 when the $\overline{RESET}$ pin is asserted. When the Port A submode field is relevant in a mode/submode definition, it must not be altered unless the H12 Enable bit in the Port General Control Register is 0. (See Table 4.)

The operation of H1 and H2 and their related status bits is given below, for each of the modes specified by Port General Control Register bits 7 and 6. This description is organized such that for each mode/submode all programmable options of each pin and status bit are given.

Bits 2 and 1 carry the same meaning in each mode/submode, and thus are specified only one.

PACR

2	H2 Interrupt Enable
0	The H2 interrupt is disabled.
1	The H2 interrupt is enabled.

PACR

1	H1 SVCRQ Enable
0	The H1 interrupt and DMA request are disabled.
1	The H1 interrupt and DMA request are enabled.

(1) PACR Mode 0 Port A Submode 00

PACR

5	4	3	H2 Control
0	X	X	Input pin – status only.
1	0	0	Output pin – always negated.
1	0	1	Output pin – always asserted.
1	1	0	Output pin – interlocked input handshake protocol.
1	1	1	Output pin – pulsed input handshake protocol.

PACR

0	H1 Status Control
X	Not Used.

(2) PACR Mode 0 Port A Submode 01

PACR

5	4	3	H2 Control
0	X	X	Input pin – status only.
1	0	0	Output pin – always negated.
1	0	1	Output pin – always asserted.
1	1	0	Output pin – interlocked output handshake protocol.
1	1	1	Output pin – pulsed output handshake protocol.

PACR

0	H1 Status Control
0	The H1S status bit is 1 when either the Port A initial or final output latch can accept new data. It is 0 when both latches are full and cannot accept new data.
1	The H1S status bit is 1 when both of the Port A output latches are empty. It is 0 when at least one latch is full.

(3) PACR Mode 0 Port A Submode 1X

PACR

5	4	3	H2 Control
0	X	X	Input pin – status only.
1	X	0	Output pin – always negated.
1	X	1	Output pin – always asserted.

PACR

0	H1 Status Control
X	Not used.

(4) PACR Mode 1 Port A Submode XX Port B Submode X0

PACR

5	4	3	H2 Control
0	X	X	Input pin – status only.
1	X	0	Output pin – always negated.
1	X	1	Output pin – always asserted.

PACR

0	H1 Status Control
X	Not used.

(5) PACR Mode 1 Port A Submode XX Port B Submode X1

PACR

5	4	3	H2 Control
0	X	X	Input pin – status only.
1	X	0	Output pin – always negated.
1	X	1	Output pin – always asserted.

PACR

0	H1 Status Control
X	Not used.

(6) PACR Mode 2

PACR

5	4	3	H2 Control
X	X	0	Output pin – interlocked output handshake protocol.
X	X	1	Output pin – pulsed output handshake protocol.

PACR

0	H1 Status Control
0	The H1S status bit is 1 when either the Port B initial or final output latch can accept new data. It is 0 when both latches are full and cannot accept new data.
1	The H1S status bit is 1 when both of the Port B output latches are empty. It is 0 when at least one latch is full.

(7) PACR Mode 3

PACR

5	4	3	H2 Control
X	X	0	Output pin – interlocked output handshake protocol.
X	X	1	Output pin – pulsed output handshake protocol.

PACR

0	H1 Status Control
0	The H1S status bit is 1 when either the initial or final output latch of Port A and B can accept new data. It is 0 when both latches are full and cannot accept new data.
1	The H1S status bit is 1 when both the initial and final output latches of Ports A and B are empty. It is 0 when either the initial or final latch of Ports A and B is full.

- **Port B Control Register (PBCR)**

Port B Control Register (PBCR)

7	6	5	4	3	2	1	0
Port 8 Submode		H4 Control			H4 Int. Enable	H3 SVCRQ Enable	H3 Stat. Ctrl.

The Port B Control Register (PBCR) specifies the operation of Port B and the handshake pins H3 and H4. The Port B control register contains five fields; bits 7 and 6 specify the Port B submode; bis 5, 4, and 3 control the operation of the H4 handshake pin and H4S status bit; bit 2 determines whether an interrupt will be generated when the H4S status bit goes to 1; bit 1 determines whether a service request (interrupt request or DMA request) will occur; bit 0 controls the operation of the H3S status bit. The PACR is always readable and writeable. There is never a consequence to reading the register.

All bits are cleared to 0 when the $\overline{\text{RESET}}$ pin is asserted.

When the Port B submode field is relevant in a mode/submode definition, it must not be altered unless the H34 Enable bit in the Port General Control Register is 0. (See Table 4.)

The operation of H3 and H4 and their related status bits is given below, for each of the modes specified by Port General Control Register bis 7 and 6. This description is organized such that for each mode/submode all programmable options of each pin and status bit are given.

Bits 2 and 1 carry the same meaning in each mode/submode, and thus are specified only once.

PBCR

2	H4 Interrupt Enable
0	The H4 interrupt is disabled.
1	The H4 interrupt is enabled.

PBCR

1	H3 SVCRQ Enable
0	The H3 interrupt and DMA request are disabled.
1	The H3 interrupt and DMA request are enabled.

(1) PBCR Mode 0 Port B Submode 00

PBCR

5	4	3	H4 Control
0	X	X	Input pin – status only.
1	0	0	Output pin – always negated.
1	0	1	Output pin – always asserted.
1	1	0	Output pin – interlocked input handshake protocol.
1	1	1	Output pin – pulse input handshake protocol.

PBCR

0	H3 Status Control
X	Not used.

(2) PBCR Mode 0 Port B Submode 01

PBCR

5	4	3	H4 Control
0	X	X	Input pin – status only.
1	0	0	Output pin – always negated.
1	0	1	Output pin – always asserted.
1	1	0	Output pin – interlocked output handshake protocol.
1	1	1	Output pin – pulse output handshake protocol.

PBCR

0	H3 Status Control
0	The H3S status bit is 1 when either the Port B initial or final output latch can accept new data. It is 0 when both latches are full and cannot accept new data.
1	The H3S status bit is 1 when both of the Port B output latches are empty. It is 0 when at least one latch is full.

(3) PBCR Mode 0 Port B Submode 1X

PBCR

5	4	3	H4 Control
0	X	X	Input pin – status only.
1	X	0	Output pin – always negated.
1	X	1	Output pin – always asserted.

PBCR

0	H3 Status Control
X	Not used .

(4) PBCR Mode 1 Port B Submode X0

PBCR

5	4	3	H4 Control
0	X	X	Input pin – status only.
1	0	0	Output pin – always negated.
1	0	1	Output pin – always asserted.
1	1	0	Output pin – interlocked input handshake protocol.
1	1	1	Output pin – pulsed input handshake protocol.

PBCR

0	H3 Status Control
X	Not used.

(5) PBCR Mode 1 Port B Submode X1

PBCR

5	4	3	H4 Control
0	X	X	Input pin – status only.
1	0	0	Output pin – always negated.
1	0	1	Output pin – always asserted.
1	1	0	Output pin – interlocked output handshake protocol.
1	1	1	Output pin – pulsed output handshake protocol.

PBCR

0	H3 Status Control
0	The H3S status bit is 1 when either the initial or final output latch of Port A and B can accept new data. It is 0 when both latches are full and cannot accept new data.
1	The H3S status bit is 1 when both the initial and final output latches of Port A and B are empty. It is 0 when neither the initial or final latch of Port A and B is full.

(6) PBCR Mode 2

PBCR

5	4	3	H4 Control
X	X	0	Output pin – interlocked input handshake protocol.
X	X	1	Output pin – pulsed input handshake protocol.

PBCR

0	H3 Status Control
X	Not used.

(7) PBCR Mode 3

PBCR

5	4	3	H4 Control
X	X	0	Output pin – interlocked input handshake protocol.
X	X	1	Output pin – pulsed input handshake protocol.

PBCR

0	H3 Status Control
X	Not used.

- **Port A Data Register (PADR)**

The Port A Data Register (PADR) is an address for moving data to and from the Port A pins. The Port A Data Direction Register determines whether each pin is an input (0) or an output (1), and is used in configuring the actual data paths. This is mode dependent and is described with the modes above.

This register is readable and writeable at all times. Depending on the chosen mode/submode, reading or writing may affect the double-buffered handshake mechanism. The Port A Data Register is not affected by the assertion of the $\overline{\text{RESET}}$ pin.

- **Port B Data Register (PBDR)**

The Port B Data Register (PBDR) is an address for moving data to and from the Port B pins. The Port B Data Direction Register determines whether each pin is an input (0) or an output (1), and is used in configuring the actual data paths. This is mode dependent and is described with the modes, above.

This register is readable and writeable at all times. Depending on the chosen mode/submode, reading or writing may affect the double-buffered handshake mechanism. The Port B Data Register is not affected by the assertion of the $\overline{\text{RESET}}$ pin.

- **Port A Alternate Register (PAAR)**

The Port A Alternate Register (PAAR) is an alternate address for reading the Port A pins. It is a read-only address and no other PI/T condition is affected. In all modes and the instantaneous pin level is read and no input latching is performed except at the data bus interface (see Bus Interface Connection.) Writes to this address are answered with $\overline{\text{DTACK}}$, but the data is ignored.

- **Port B Alternate Register (PBAR)**

The Port B Alternate Register (PBAR) is an alternate address for reading the Port B pins. It is a read-only address and no other PI/T condition is affected. In all modes the instantaneous pin level is read and no input laching is performed except at the data bus interface (see Bus Interface Connection). Writes to this address are answered with $\overline{\text{DTACK}}$, but the data is ignored.

- **Port C Data Register (PCDR)**

The Port C Data Register (PCDR) is an address for moving data to and from each of the eight Port C/alternate-function pins. The exact hardware accessed is determined by the type of bus cycle (read or write) and individual conditions affecting each pin. These conditions are (1) whether the pin is used for the Port C or alternate function, and (2) whether the Port C Data Direction Register indicates the input or output direction. The Port C Data Register is single buffered for output pins and not buffered for input pins. These conditions are summarized in Table 3.

The Port C Data Register is not affected by the assertion of the $\overline{\text{RESET}}$ pin.

The operation of the PCDR is independent of the chosen PI/T mode.

Table 3 PCDR Hardware Accesses

Read Port C Data Register			
Port C function PCDDR = 0	Port C function PCDDR = 1	Alternate function PCDDR = 0	Alternate function PCDDR = 1
pin	Port C output register	pin	Port C output register
Write Port C Data Register			
Port C function PCDDR = 0	Port C function PCDDR = 1	Alternate function PCDDR = 0	Alternate function PCDDR = 1
Port C output register, buffer disabled	Port C output register, buffer enabled	Port C output register	Port C output register

Note that two additional useful benefits result from this structure. First, it is possible to directly read the state of a dual-function pin while used for the non-Port C function. Second, it is possible to generate program controlled transitions on alternate-function pins by switching back to the Port C function, and writing to the PCDR.

This register is readable and writeable at all times.

- **Port Status Register (PSR)**

Port Status Register (PSR)

7	6	5	4	3	2	1	0
H4 Level	H3 Level	H2 Level	H1 Level	H4S	H3S	H2S	H1S

The Port Status Register (PSR) contains information about handshake pin activity. Bits 7-4 show the instantaneous level of the respective handshake pin, and is independent of the handshake pin sense bits in the Port General Control Register. Bit 3-0 are the respective, status bits referred to throughout this data sheet. Their interpretation depends on the programmed mode/submode of the PI/T. For Bits 3-0 a 1 is the active or asserted state.

- **Timer Control Register (TCR)**

Timer Control Register (TCR)

7	6	5	4	3	2	1	0
TOUT/TIACK Control			Z.D. Ctrl.	*	Clock Control		Timer Enable

The Timer Control Register (TCR) determines all operations of the timer. Bits 7-5 configure the PC$_3$/TOUT and PC$_7$/TIACK pins for Port C, square wave, vectored interrupt, or autovectored interrupt operation; bit 4 specifies whether the counter receives data from the Counter Preload Register or continues counting when zero detect is reached; bit 3 is unused and is read as 0; bits 2 and 1 configure the path from the CLK and TIN pins to the counter controller; bit 0 enables the timer. This register is readable and writeable at all times.

All bits are cleared to 0 when the RESET pin is asserted.

TCR
7 6 5 TOUT:TIACK Control

0 X X The dual-function pins PC$_3$/TOUT and PC$_7$/TIACK carry the Port C function.

0 1 X The dual-function pin PC$_3$/TOUT cames the TOUT function. In the run state it is used as a square wave output and is toggled on zero detect. The TOUT pin is high while in the halt state. The dual-function pin PC$_7$/TIACK carries the PC$_7$ function.

1 0 0 The dual-function pin PC$_3$/TOUT carries the TOUT function. In the run or halt state it is used as a timer interrupt request output. The timer interrupt is disabled; thus, the pin is always three-stated. The dual-function pin PC$_7$/TIACK carries the TIACK function; however, since interrupt request is negated, the PI/T produces no response, i.e., no data or DTACK, to an asserted TIACK. Refer to Time Interrupt Cycle section for details. This combination and the 101 state below support vectored timer interrupts.

1 0 1 The dual-function pin PC$_3$/TOUT carries the TOUT function and is used as a timer interrupt request output. The timer interrupt is enabled; thus, the pin is low when the timer ZDS status bit is 1. The dual function pin PC$_7$/TIACK carries the TIACK function and is used as a timer interrupt acknowledge input. Refer to the Timer Interrupt Acknowledge Cycle section for details. This combination and the 100 state above support vectored timer interrupts.

1 1 0 The dual-function pin PC$_3$/TOUT carries the TOUT function. In the run or halt state it is used as a timer interrupt request output. The timer interrupt is disabled; thus, the pin is always three-stated. The dual-function pin PC$_7$/TIACK carries the PC$_7$ function.

1 1 1 The dual-function pin PC$_3$/TOUT carries the TOUT function and is used as a timer interrupt request output. The timer interrupt is enabled; thus, then pin is low when the timer ZDS status bits is 1. The dual-function pin PC$_7$/TIACK carries the PC$_7$ function and autovectored interrupts are supported.

TCR
4 Zero Detect Control

0 The counter is loaded from the Counter Preload Register on the first clock to the 24-bit counter after zero detect, and resumes counting.

1 The counter rolls over on zero detect, then continues counting.

Bit 3 is unused and is always read as 0.

TCR
2 1 Clock Control

0 0 The PC2/TIN input pin carries the Port C function and the CLK pin and prescaler are used. The prescaler is decremented on the falling transition of the CLK pin; the 24-bit counter is decremented or loaded from the Counter Preload Register when the prescaler rolls over from \$00 to \$1F. The Timer Enable bit determines whether the timer is in the run or halt state.

0 1 The PC$_2$/TIN pin serves as a timer input and the CLK pin and prescaler are used. The prescaler is decremented on the falling transition of the CLK pin; the 24-bit counter is decremented or loaded from the Counter Preload Registers when the prescaler rolls over from \$00 to \$1F. The timer is in the run state when the Timer Enable bit is 1 and the TIN pin is high; otherwise the timer is in the halt state.

1 0 The PC$_2$/TIN pin serves as a timer input and the prescaler is used. The prescaler is decremented following the rising transition of the TIN pin after syncing with the internal clock. The 24-bit counter is decremented or loaded from the counter preload registers when the prescaler rolls over from \$00 to \$1F. The Timer Enable bit determines whether the timer is in the run or halt state.

1 1 The PC$_2$/TIN pin serves as a timer input and the prescaler is unused. The 24-bit counter is decremented or loaded from the Counter Preload Registers following the rising edge of the TIN pin after syncing with the internal clock. The Timer Enable bit determines whether the timer is in the run or halt state.

TCR
0 Timer Enable

0 Disabled.

1 Enabled.

- **Timer Interrupt Vector Register (TIVR)**

The timer interrupt vector register contains the 8-bit vector supplied when the timer interrupt acknowledge pin TIACK is asserted. The register is readable and writeable at all times, and the same value is always obtained from a normal read cycle and a timer interrupt acknowledge bus cycle (TIACK). When the RESET pin is asserted the value of \$0F is automatically loaded into the register. Refer to Timer Interrupt Acknowledge Cycle section for more details.

- **Counter Preload Register H,M,L (CPRH–L)**

Counter Preload Register H,M,L (CPRH-L)

7	6	5	4	3	2	1	0	
Bit 23	Bit 22	Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit 16	CPRH
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	CPRM
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	CPRL

The Counter Preload Registers (CPRH-L) are a group of three 8-bit registers used for storing data to be transferred to the counter. Each of the registers is individually addressable, or the group may be accessed with the MOVEP. L or the MOVEP, W instructions. The address one less than the address of CPRH is the null register, and is reserved so that zeros are read in the upper 8 bits of the destination data register when a MOVEP.L is used. Data written to this address is ignored.

The registers are readable and writeable at all times. A read cycle proceeds independently of any transfer to the counter, which may be occuring simultaneously.

To insure proper operation of the PI/T Timer, a value of $000000 may not be stored in the Counter Preload Registers for use with the counter.

The $\overline{\text{RESET}}$ pin does not affect the contents of these registers.

- **Count Register H,M,L (CNTRH–L)**

Count Register H,M,L (CNTRH-L)

7	6	5	4	3	2	1	0	
Bit 23	Bit 22	Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit 16	CNTRH
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	CNTRM
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	CNTRL

The count registers (CNTRH-L) are a group of three 8-bit addresses at which the counter can be read. The contents of the counter are not latched during a read bus cycle; thus, the data read at these addresses is not guaranteed if the timer is in the run state. (Bits 2, 1, and 0 of the Timer Control Register specify the state.) Write operations to these addresses result in a normal bus cycle but the data is ignored.

Each of the registers is individually addressable, or the group may be accessed with the MOVEP.L or the MOVEP.W instructions. The address one less than the address of CNTRH is the null register, and is reserved so that zeros are read in the upper 8 bits of the destination data register when a MOVEP.L is used. Data written to this address is ignored.

- **Timer Status Register (TSR)**

Timer Status Register (TSR)

7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	ZDS

The Timer Status Register (TSR) contains one bit from which the zero detect status can be determined. The ZDS status bit (bit 0) is an edge-sensitive flip-flop that is set to 1 when the 24-bit counter decrements from $000001 to $000000. The ZDS status bit is cleared to 0 following the direct clear operation (similar to that of the ports), or when the timer is halted. Note also that when the $\overline{\text{RESET}}$ pin is asserted the timer is disabled, and thus enters the halt state.

This register is always readable without consequence. A write access performs a direct clear operation if bit 0 in the written data is 1. Following that, the ZDS bit is 0.

This register is constructed with a reset dominant S-R flip-flop so that all cleaning conditions prevail over the possible zero detect condition.

Bits 7-1 are unused and are read as 0.

PORT CONTROL STRUCTURE

The primary focus of most applications will be on Ports A and B, the handshake pins, the port interrupt pins, and the DMA request pin. They are controlled in the following way: the Port General Control Register contains a 2-bit field that specifies a set of four operation modes. These govern the overall operation of the ports and determine their interrelationships. Some modes require additional information from each port's control register to further define its operation. In each port control register, there is a 2-bit submode field that serves this purpose. Each port mode/submode combination specifies a set of programmable characteristics that fully define the behavior of that port and two of the handshake pins. This structure is summarized in Table 4 and Figure 8.

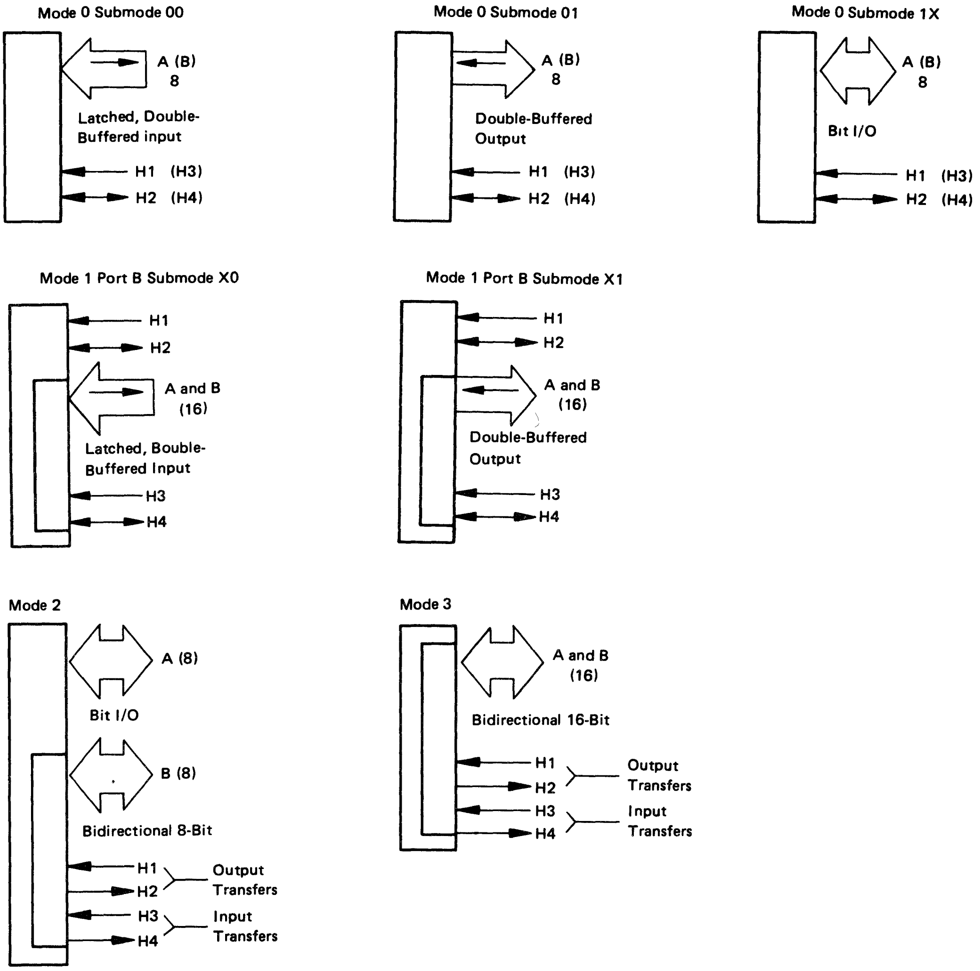


Figure 8 Port Mode Layout

PORT GENERAL INFORMATION AND CONVENTIONS

The following paragraphs introduce concepts that are generally applicable to the PI/T ports independent of the chosen mode and submode. For this reason, no particular port or handshake pins are mentioned; the notation H1 (H3) indicates that, depending on the chosen mode and sub-mode, the statement given may be true for either the H1 or H3 handshake pin.

• Unidirectional vs Bidirectional

Figure 8 shows the configuration of Ports A and B and each of the handshake pins in each port mode and submode. In Modes 0 and 1, a data direction register is associated with each of the ports. These registers contain one bit for each port pin to determine whether that pin is an input or an output. Modes 0 and 1 are, thus, called unidirectional modes because each pin assumes a constant direction, changeable only by a reset condition or a programming change. These modes allow double-buffered data transfers in one direction. This direction, determined by the mode and submode definition, is known as the primary direction. Data transfers in the primary direction are controlled by the handshake pins. Data transfers not in the primary direction are generally unrelated, and single or unbuffered data paths exist.

In Modes 2 and 3 there is no concept of primary direction as in Modes 0 and 1. Except for Port A in Mode 2 (Bit I/O), the data direction registers have no effect. These modes are bidirectional, in that the direction of each transfer (always 8 or 16 bits, double-buffered) is determined dynamically by the state of the handshake pins. Thus, for example, data may be transferred out of the ports, followed very shortly by a transfer into the same port pins. Transfers to and from the ports are independent and may occur in any sequence. Since the instantaneous direction is always determined by the external system, a small amount of arbitration logic may be required.

• Control of Double-Buffered Data Paths

Generally speaking, the PI/T is a double-buffered device. In the primary direction, double-buffering allows orderly transfers by using the handshake pins in any of several programmable protocols. (When Bit I/O is used, double-buffering is not available and the handshake pins are used as outputs or status/interrupt inputs.)

Use of double-buffering is most beneficial in situations where a peripheral device and the computer system are capable of transferring data at roughly the same speed. Double-buffering allows the fetch operation of the data transmitter to be overlapped with the store operation of the data receiver. Thus, throughput measured in bytes or words-per-second may be greatly enhanced. If there is a large mismatch in transfer capability between the computer and the peripheral, little or no benefit is obtained. In these cases there is no penalty in using double-buffering.

• Double-Buffered Input Transfers

In all modes, the PI/T supports double-buffered input transfers. Data that meets the port setup and hold times is latched on the asserted edge of H1(H3). H1(H3) is edge-sensitive, and may assume any duty-cycle as long as both high and low minimum times are observed. The PI/T contains a Port Status Register whose H1S(H3S) status bit is set anytime any input data is present in the double-buffered latches that has not been read by the bus master. The action of H2(H4) is programmable; it may indicate whether there is room for more data in the PI/T latches or it may serve other purposes. The

Table 4 Port Mode Control Summary

Mode 0 (Unidirectional 8-Bit mode)	Mode 1 (Unidirectional 16-Bit mode)	Mode 2 (Bidirectional 8-Bit mode)	Mode 3 (Bidirectional 16-Bit mode)
Port A	Port A – Double-Buffered Data (Most significant)	Port A – Bit I/O (with no handshaking pins)	Port A – Double-Buffered Data (Most significant)
Submode 00 – Double-Buffered Input H1 – Latches input data H2 – Status/interrupt generating input, general-purpose output, or operation with H1 in the interlocked or pulsed input handshake protocols Submode 01 – Double-Buffered Output H1 – Indicates data received by peripheral H2 – Status/interrupt generating input, general-purpose output, or operation with H1 in thè interlocked or pulsed output handshake protocols Submode 1X – Bit I/O H1 – Status/interrupt generating input H2 – Status/interrupt generating input or general-purpose output	Submode XX (not used) H1 – Status/interrupt generating input H2 – Status/interrupt generating input or general-purpose output	Submode XX (not used)	Submode XX (not used)
Port B	Port B – Double-Buffered Data (Least significant)	Port B – Bidirectional 8-Bit Data (Double-Buffered)	Port B – Double-Buffered Data (Least significant)
H3 and H4 – Identical to Port A, H1 and H2	Submode X0 – Unidirectional 16-Bit Input H3 – Latches input data H4 – Status/interrupt generating input, general-purpose output, or operation with H3 in the interlocked or pulsed input handshake protocols Submode X1 – Unidirectional 16-Bit Output H3 – Indicates data received by peripheral H4 – Status/interrupt generating input, general-purpose output , or operation with H3 in the interlocked or pulsed output handshake protocols	Submode XX (not used) H1 – indicates output data received by peripheral H2 – Operation with H1 in the interlocked or pulsed output handshake protocols H3 – Latches input data H4 – Operation with H3 in the interlocked or pulsed input handshake protocols	Submode XX (not used) H1 – Indicates output data received by peripheral H2 – Operation with H1 in the interlocked or pulsed output handshake protocols H3 – Latches input data H4 – Operation with H3 in the interlocked or pulsed input handshake protocols

following options are available, depending on the mode.

(1) H2(H4) may be an edge-sensitive input that is independent of H1(H3) and the transfer of port data. On the asserted edge of H2(H4), the H2S(H4S) status bit is set. It is cleared by the direct method (refer to Direct Method of Resetting Status), the $\overline{\text{RESET}}$ pin being asserted, or when the H12 Enable (H34 Enable) bit of the Port General Control Register is 0.

(2) H2(H4) may be a general purpose output pin that is always negated. The H2S(H4S) status bit is always 0.

(3) H2(H4) may be a general purpose output pin that is always asserted. The H2S(H4S) status bit is always 0.

(4) H2(H4) may be an output pin in the interlocked input handshake protocol. It is asserted when the port input latches are ready to accept new data. It is negated asynchronously following the asserted edge of the H1(H3) input.

As soon as the input latches become ready, H2(H4) is again asserted. When the input double-buffered latches are full, H2(H4) remains negated until data is removed. Thus, anytime the H2(H4) output is asserted, new input data may be entered by asserting H1(H3). At other times transitions on H1(H3) are ignored. The H2S(H4S) status bit is always 0. When H12 Enable (H34 Enable) is 0, H2(H4) is held negated.

(5) H2(H4) may be an output pin in the pulsed input handshake protocol. It is asserted exactly as in the interlocked input protocol, but never remains asserted longer than 4 clock cycles. Typically, a four clock cycle pulse is generated. But in the case that a subsequent H1(H3) asserted edge occurs before termination of the pulse, H2(H4) is negated asynchronously. Thus, anytime after the leading edge of the H2(H4) pulse, new data may be entered in the PI/T double-buffered input latches. The H2S(H4S) status bit is always 0. When H12 Enable (H34 Enable) is 0, H2(H4) is held negated.

A sample timing diagram is shown in Figure 9. The H2(H4) interlocked and pulsed input handshake protocols are shown. The $\overline{\text{DMAREQ}}$ pin is also shown assuming it is enabled. All handshake pin sense bits are assumed to be 0 (refer to Port General Control Register); thus, the pins are in the low state when asserted. Due to the great similarity between modes, this timing diagram is applicable to all double-buffered input transfers.

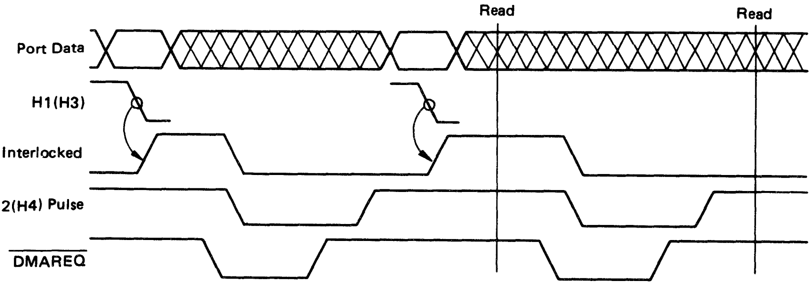


Figure 9 Double-buffered Input Transfers

- **Double-Buffered Output Transfers**

The PI/T supports double-buffered output transfers in all modes. Data, written by the bus master to the PI/T, is stored in the port's output latch. The peripheral accepts the data by asserting H1(H3), which causes the next data to be moved to the port's output latch as soon as it is available. The function of H2(H4) is programmable; it may indicate whether new data has been moved to the output latch or it may serve other purposes. The H1S(H3S) status bit may be programmed for two interpretations. Normally the status bit is a 1 when there is at least one latch in the double-buffered data path that can accept new data. After writing one byte/word of data to the ports, an interrupt service routine could check this bit to determine if it could store another byte/word; thus, filling both latches. When the bus master is finished, it is often useful to be able to check whether all of the data has been transferred to the peripheral. The H1S(H3S) Status Control bit of the Port A and B Control Registers provide this flexibility. The programmable options of the H2(H4) pin are given below, depending on the mode.

(1) H2(H4) may be an edge-sensitive input pin independent of H1(H3) and the transfer of port data. On the asserted edge of H2(H4), the H2S(H4S) status bit is set. It is reset by the direct method (refer to Direct Method of Resetting Status), the $\overline{\text{RESET}}$ pin being asserted, or when the H12 Enable (H34 Enable) bit of the Port General Control Register is 0.

(2) H2(H4) may be a general-purpose output pin that is always negated. The H2S(H4S) status bit is always 0.

(3) H2(H4) may be a general-purpose output pin that is always asserted. The H2S(H4S) status bit is always 0.

(4) H2(H4) may be an output pin in the interlocked output handshake protocol. H2(H4) is asserted two clock cycles after data is transferred to the double-buffered output latches. The data remains stable and H2(H4) remains asserted until the next asserted edge of the H1(H3) input. At that time, H2(H4) is asynchronously negated. As soon as the next data is available, it is transferred to the output latches. When H2(H4) is negated, asserted transitions on H1(H3) have no effect on the data paths. As is explained later, however, in Modes 2 and 3 they do control the three-state output buffers of the bidirectional port(s). The H2S(H4S) status bit is always 0. When H12 Enable (H34 Enable) is 0, H2(H4) is held negated.

(5) H2(H4) may be an output pin in the pulsed output handshake protocol. It is asserted exactly as in the interlocked output protocol above, but never remains asserted longer than four clock cycles. Typically, a four clock pulse is

generated. But in the case that a subsequent H1(H3) asserted edge occurs before termination of the pulse, H2(H4) is negated asynchronously shortening the pulse. The H2S(H4S) status bit is always 0. When H12 Enable (H34 Enable) is 0 H2(H4) is held negated.

A sample timing diagram is shown in Figure 10. The H2(H4) interlocked and pulsed output handshake protocols are shown. The $\overline{\text{DMAREQ}}$ pin is also shown assuming it is enabled. All handshake pin sense bits are assumed to be 0; thus, the pins are in the low state when asserted. Due to the great similarity between modes, this timing diagram is applicable to all double-buffered output transfer.

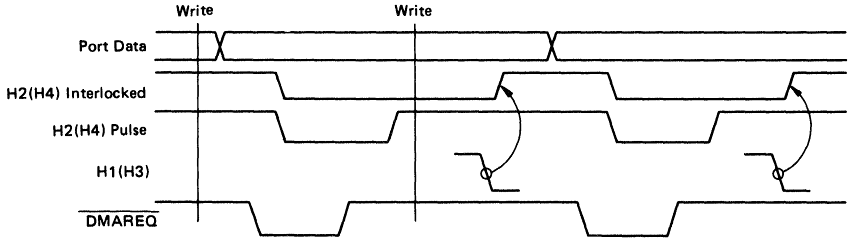


Figure 10 Double-buffered Output Transfers

- **Requesting Bus Master Service**

The PI/T has several means of indicating a need for service by a bus master. First, the processor may poll the Port Status Register. It contains a status bit for each handshake pin, plus a level bit that always reflects the instantaneous state of that handshake pin. A status bit is 1 when the PI/T needs servicing, i.e., generally when the bus master needs to read or write data to the ports, or when a handshake pin used as a simple status input has been asserted. The interpretation of these bits is dependent on the chosen mode and submode.

Second, the PI/T may be placed in the processor's interrupt structure. As mentioned previously, the PI/T contains Port A and B Control Registers that configure the handshake pins. Other bits in these registers enable an interrupt associated with each handshake pin. This interrupt is made available through the PC5/$\overline{\text{PIRQ}}$ pin, if the $\overline{\text{PIRQ}}$ function is selected. Three additional conditions are required for $\overline{\text{PIRQ}}$ to be asserted: (1) the handshake pin status bit set, (2) the corresponding interrupt (service request) enable bit is set, (3) and DMA requests are not associated with that data transfer (H1 and H3 only). The conditions from each of the four handshake pins and corresponding status bits are ORed to determine $\overline{\text{PIRQ}}$.

The third method of requesting service is via the PC4/$\overline{\text{DMAREQ}}$ pin. This pin can be associated with double-buffered transfers in each mode. If it is used as a DMA controller request, it can initiate requests to keep the PI/T's input/output double-buffering empty/full as much as possible. It will not overrun the DMA controller. The pin is compatible with the HD68450 Direct Memory Access Controller (DMAC).

- **Vectored, Prioritized Port Interrupts**

Use of HD68000-compatible vectored interrupts with the PI/T requires the $\overline{\text{PIRQ}}$ and $\overline{\text{PIACK}}$ pins. When $\overline{\text{PIACK}}$ is asserted, the PI/T places an 8-bit vector on the data pins D_0-D_7. Under normal conditions, this vector corresponds to highest priority, enabled, active port interrupt source with which the $\overline{\text{DMAREQ}}$ pin is not currently associated. The most-significant six bits are provided by the Port Interrupt Vector Register (PIVR), with the lower two bits supplied by prioritization logic according to conditions present when $\overline{\text{PIACK}}$ is asserted. It is important to note that the only affect on the PI/T caused by interrupt acknowledge cycles is that the vector is placed on the data bus. Specifically, no registers, data, status, or other internal states of the PI/T are affected by the cycle.

Several conditions may be present when the $\overline{\text{PIACK}}$ input is asserted to the PI/T. These conditions affect the PI/T's response and the termination of the bus cycle. If the PI/T has no interrupt function selected, or is not asserting $\overline{\text{PIRQ}}$, the PI/T will make no response to $\overline{\text{PIACK}}$ ($\overline{\text{DTACK}}$ will not be asserted). If the PI/T is asserting $\overline{\text{PIRQ}}$ when $\overline{\text{PIACK}}$ is received, the PI/T will output the contents of the Port Interrupt Vector Register and the prioritization bits. If the PIVR has not been initialized, $0F will be read from this register. These conditions are summarized in Table 5.

Table 5 Response to Port Interrupt Acknowledge

Conditions	$\overline{\text{PIRQ}}$ negated OR interrupt request function not selected	$\overline{\text{PIRQ}}$ asserted
PIVR has not been initialized since $\overline{\text{RESET}}$	No response from PI/T. No $\overline{\text{DTACK}}$.	PI/T provides $0F, the Uninitialized Vector.*
PIVR has been initialized since $\overline{\text{RESET}}$	No response from PI/T. No $\overline{\text{DTACK}}$.	PI/T provides PIVR contents with prioritization bits.

*The uninitialized vector is the value returned from an interrupt vector register before it has been initialized.

The vector table entries for the PI/T appear as a contiguous block of four vector numbers whose common upper six bits are programmed in the PIVR. The following table pairs each interrupt source with the 2-bit value provided by the prioriza-tion logic, when interrupt acknowledge is asserted.

H1 source	00
H2 source	01
H3 source	10
H4 source	11

- **Autovectored Port Interrupts**

Autovectored interrupts use only the $\overline{\text{PIRQ}}$ pin. The operation of the PI/T with vectored and autovectored interrupts is identical except that no vectors are supplied and the PC6/$\overline{\text{PIACK}}$ pin can be used as a Port C pin.

- **Direct Method of Resetting Status**

In certain modes one or more handshake pins can be used as edge-sensitive inputs for sole purpose of setting bits in the Port Status Register. These bits consist of simple flip-flops. They are set (to 1) by the occurrence of the asserted edge of the handshake pin input. Resetting a handshake status bit can be done by writing an 8-bit mask to the Port Status Register. This is called the direct method of resetting. To reset a status bit that is resettable by the direct method, the mask must contain a 1 in the bit position of the Port Status Register corresponding to the desired bit. Other positions must contain 0's. For status bits that are not resettable by the direct method in the chosen mode, the data written to the port status register has no effect. For status bits that are resettable by the direct method in the chosen mode, a 0 in the mask has no effect.

- **Handshake Pin Sense Control**

The PI/T contains exclusive-OR gates to control the sense of each of the handshake pins, whether used as inputs or outputs. Four bits in the Port General Control Register may be programmed to determine whether the pins are asserted in the low or high voltage state. As with other control registers, these bits are reset to 0 when the $\overline{\text{RESET}}$ pin is asserted, defaulting the asserted level to be low.

- **Enabling Ports A and B**

Certain functions involved with double-buffered data transfers, the handshake pins, and the status bits, may be disabled by the external system or by the programmer during initialization. The Port General Control Register contains two bits, H12 Enable and H34 Enable, which control these functions. These bits are cleared to the 0 state when the $\overline{\text{RESET}}$ pin is asserted, and the functions are disabled. The functions are the following.

(1) Independent of other actions by the bus master or peripheral (via the handshake pins), the PI/T's disabled handshake controller is held to the "empty" state, i.e., no data is present in the double-buffered data path.

(2) When any handshake pin is used to set a simple status flip-flop, unrelated to double-buffered transfers, these flip-flops are held reset to 0. (See Table 4.)

(3) When H2(H4) is used in an interlocked or pulsed handshake with H1(H3), H2(H4) is held negated, regardless of the chosen mode, submode, and primary direction. Thus, for double-buffered input transfers, the programmer may signal a peripheral when the PI/T is ready to begin transfers by setting the associated handshake enable bit to 1.

- **The Port A and B Alternate Registers**

In addition to the Port A and B Data Registers, the PI/T contains Port A and B Alternate Registers. These registers are read-only, and simply provide the instantaneous level of each port pin. They have no effect on the operation of the handshake pins, double-buffered transfers, status bits, or any other aspect of the PI/T, and they are mode/submode independent.

■ PORT MODES

This section contains information that distinguishes the various port modes and submodes.

- **Mode 0 – Unidirectional 8-Bit Mode**

In Mode 0, Ports A and B operate independently. Each may be configured in any of its three possible submodes:

Submode 00 – Double-Buffered Input
Submode 01 – Double-Buffered Output
Submode 1X – Bit I/O

Handshake pins H1 and H2 are associated with Port A and configured by programming the Port A Control Register. (The H12 Enable bit of the Port General Control Register, enables Port A transfers.) Handshake pins H3 and H4 are associated with Port B and configured by programming the Port B Control Register. (The H34 Enable bit of the Port General Control Register enables Port B transfers.) The Port A and B Data Direction Registers operate in all three submodes. Along with the submode, they affect the data read and written at the associated data register according to Table 6. They also enable the output buffer associated with each port pin. The $\overline{\text{DMAREQ}}$ pin may be associated with either (not both) Port A or Port B, but does not function if the Bit I/O submode is programmed for the chosen port.

Table 6 Mode 0 Port Data Paths

Mode	Read Port A/B Data Register		Write Port A/B Data Register	
	DDR = 0	DDR = 1	DDR = X	
0 Submode 00	FIL. D.B.	FOL Note 3	FOL. S.B.	Note 1
0 Submode 01	Pin	FOL Note 3	IOL/FOL. D.B.	Note 2
0 Submode 1X	Pin	FOL Note 3	FOL. S.B.	Note 1

Abbreviations:

IOL	Initial Output Latch	S.B.	Single Buffered
FOL	Final Output Latch	D.B.	Double Buffered
FIL	Final Input Latch	DDR	Data Direction Register

Note 1: Data is latched in the output data registers (final output latch) and will be single buffered at the pin if the DDR is 1. The output buffers will be turned off if the DDR is 0.

Note 2: Data is latched in the double-buffered output data registers. The data in the final output latch will appear on the port pin if the DDR is a 1.

Note 3: The output drivers that connect the final output latch to the pins are turned on.

(1) Port A or B Submode 00 (8-Bit Double-Buffered Input)

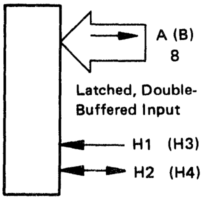


Figure 11 Mode 0 Submode 00

In Mode 0, double-buffered input transfers of up to 8-bits are available by programming Submode 00 in the desired port's control register. The operation of H2 and H4 may be selected by programming the Port A and Port B Control Registers, respectively. All five double-buffered input handshake options, previously mentioned in the Port General Information and Conventions section, are available.

For pins used as outputs, the data path consists of single latch driving the output buffer. Data written to the port's data register does not affect the operation of any handshake pin, status bit, or any other aspect of the PI/T. Output pins may be used independently of the input transfer. However, read bus cycles to the data register do remove data from the port. Therefore, care should be taken to avoid processor instructions that perform unwanted read cycles.

Refer to PARALLEL PORTS Double-Buffered Input Transfers for a sample timing diagram. (Figure 9)

(2) Port A or B Submode 01 (8-Bit Double-Buffered Output)

In Mode 0, double-buffered output transfers of up to 8 bits are available by programming submode 01 in the desired port's control register. The operation of H2 and H4 may be selected by programming the Port A and Port B Control Registers, respectively. All five double-buffered output handshake options, previously mentioned in the Port General Information and Conventions section, are available.

For pins used as inputs, data written to the associated data register is double-buffered and passed to the initial or final output latch, as usual, but the output buffer is disabled.

Refer to PARALLEL PORTS Double-Buffered Output Transfers for a sample timing diagram (Figure 10)

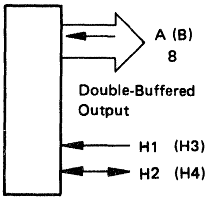


Figure 12 Mode 0 Submode 01

(3) Port A or B Submode 1X (Bit I/O)

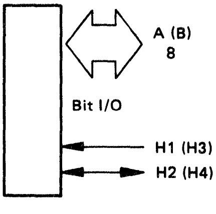


Figure 13 Mode 0 Submode 1X

In Mode 0, simple Bit I/O is available by programming Submode 1X in the desired port's control register. This submode is intended for applications in which several independent devices must be controlled or monitored. Data written to the associated data register is single-buffered. If the data direction register bit for that pin is a 1 (output), the output buffer is enabled. If ıt is 0 (input), data written is still latched, but is not available at the pin. Data read from the data register is the instantaneous value of the pin or what was written to the data register, depending on the contents of the data direction register. H1(H3) is an edge-sensitive status input pin only and it controls no data-related function. The H1S(H3S) status bit is set following the asserted edge of the input waveform. It is reset by the direct method, the $\overline{\text{RESET}}$ pin being asserted, or when the H12 Enable (H34 Enable) bit is 0.

H2(H4) can be programmed as a simple status input (identical to H1(H3)), or as an asserted or negated output. The interlocked or pulsed handshake configurations are not available.

- **Mode 1 – Unidirectional 16-Bit Mode**

In Mode 1, Ports A and B are concatenated to form a single 16-bit port. The Port B Submode field controls the configuration of both ports. The possible submodes are:

Port B Submode X0 – Double-Buffered Input
Port B Submode X1 – Double-Buffered Output

Handshake pins H3 and H4, configured by programming the Port B Control Register, are associated with the 16-bit double-buffered transfer. These 16-bit transfers, are enabled by the H34 Enable bit of the Port General Control Register. Handshake pins H1 and H2 may be used as simple status inputs not related to the 16-bit data transfer or H2 may be an output. Enabling of the H1 and H2 handshake pins is done by the H12 Enable bit of the Port General Control Register. The Port A and B Data Direction Registers operate in each sub-mode. Along with the submode, they affect the data read and written at the data register according to Table 7. They also enable the output buffer associated with each port pin. The $\overline{\text{DMAREQ}}$ pin may be associated only with H3.

Mode 1 can provide convenient, high-speed 16-bit transfers. The Port A and B data registers are addressed for compatibility with the HD68000 Move Peripheral (MOVEP) instruction and with the HD68450 DMAC. To take advantage of this, Port A should contain the most-significant byte of data and always be read or written by the bus master first. The interlocked and pulsed handshake protocols are keyed to accesses to the Port B Data Register in Mode 1. If it is accessed last, the 16-bit double-buffered transfers proceed smoothly.

Table 7 Mode 1 Port Data Paths

Mode	Read Port A/B Register		Write Port A/B Register	
	DDR = 0	DDR = 1	DDR = 0	DDR = 1
1, Port B Submode X0	FIL. D.B.	FOL Note 3	FOL, S.B. Note 2	FOL, S.B. Note 2
1, Port B Submode X1	Pin	FOL Note 3	IOL/FOL. D.B., Note 1	IOL/FOL. D.B., Note 1

Note 1: Data written to Port A goes to a temporary latch. When the Port B data register is later written, Port A data is transferred to IOL/FOL.

Note 2: Data is latched in the output data registers (final output latch) and will be single buffered at the pin if the DDR is 1. The output buffers will be turned off if the DDR is 0.

Note 3: The output drivers that connect the final output latch to the pins are turned on.

Abbreviations:

IOL –	Initial Output Latch	S.B. –	Single Buffered
FOL –	Final Output Latch	D.B. –	Double Buffered
FIL –	Final Input Latch	DDR –	Data Direction Register

(1) Port B Submode X0 (16-Bit Double-Buffered Input)

In Mode 1 Port B Submode X0, double-buffered input transfers of up to 16 bits may be obtained. The level of all 16 pins is asynchronously latched with the asserted edge of H3. The processor may check H3S status bit to determine if new data is present. The $\overline{\text{DMAREQ}}$ pin may be used to signal a DMA controller to empty the input buffers. Regardless of the bus master, Port A data should be read first. (Actually, Port A data need not be read at all.) Port B data should be read last. The operation of the internal handshake controller, the H3S bit, and $\overline{\text{DMAREQ}}$ are keyed to the reading of the Port B data register. (The HD68450 DMAC can be programmed to perform the exact transfers needed for compatibility with the PI/T.) H4 may be programmed for all five of the handshake options mentioned in the Port General Information and Conventions section.

For pins used as outputs, the data path consists of a single latch driving the output buffer. Data written to the port's data register does not affect the operation of any handshake pin, status bit, or any other aspect of the PI/T. Thus, output pins may be used independently of the input transfer. However, read bus cycles to the Port B Data Register do remove data, so care should be taken to avoid unwanted read cycles.

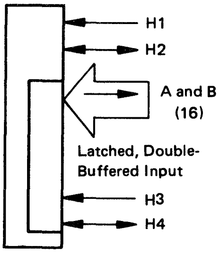


Figure 14 Mode 1 Port B Submode X0

(2) Port B Submode X1 (16-Bit Double-Buffered Output)

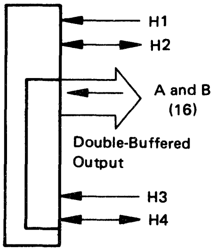


Figure 15 Mode 1 Port B Submode X1

Refer to PARALLEL PORTS Double-Buffered Input Transfers for a sample timing diagram (Figure 9).

In Mode 1 Port B Submode X1, double-buffered output transfers of up to 16 bits may be obtained. Data is written by the bus master (processor or DMA controller) in two bytes. The first byte (most-significant) is written to the Port A Data Register. It is stored in a temporary latch until the next byte is written to the Port B Data Register. Then all 16 bits are transferred to the final output latches of Ports A and B. Both options for interpretation of the H3S status bit, mentioned in Port General Information and Comments section, are available and apply to the 16-bit port as a whole. The $\overline{\text{DMAREQ}}$ pin may be used to signal a DMA controller to transfer another word to the port output latches. (The HD68450 DMAC can be programmed to perform the exact transfers needed for compatibility with the PI/T.) H4 may be programmed for all five of the handshake options mentioned in Port General Information and Comments section.

For pins used as inputs, data written to either data register is double-buffered and passed to the initial or final output latch, as usual, but the output buffer is disabled.

Refer to PARALLEL PORTS Double-Buffered Input/Output Transfer for a sample timing diagram (Figure 10).

- **Mode 2 – Bidirectional 8-Bit Mode**

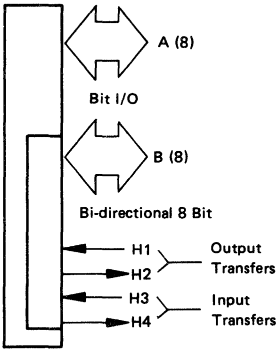


Figure 16 Mode 2

In Mode 2, Port A is used for simple bit I/O with no associated handshake pins. Port B is used for bidirectional 8-bit double-buffered transfers. H1 and H2, enabled by the H12 Enable bit in the Port General Control Register, control output transfers, while H3 and H4, enabled by the Port General Control Register bit H34 Enable, control input transfers. The instantaneous direction of the data is determined by the H1 handshake pin. The Port B Data Direction Register is not used. The Port A and Port B submode fields do not affect PI/T operation in Mode 2.

(1) Double-Buffered I/O (Port B)

The only aspect of bidirectional double-buffered transfers that differs from the uni-directional modes lies in controlling the Port B output buffers. They are controlled by the level of H1. When H1 is negated, the Port B output buffers (all 8) are enabled and the pins drive the bidirectional bus. Generally, H1 is negated in response to an asserted H2, which indicates that new output data is present in the double-buffered latches. Following acceptance of the data, the peripheral asserts H1, disabling the Port B output buffers. Other than controlling the output buffer, H1 is edge-sensitive as in other modes. Input transfers proceed identically to the double-buffered input protocol described in the Port General Information and Conventions Section. In Mode 2, only the interlocked and pulsed

Table 8 Mode 2 Port B Data Paths

Mode	Read Port B Data Register	Write Port B Data Register
2	FIL, D.B.	IOL/FOL, D.B.

Abbreviations:

IOL – Initial Output Latch
FOL – Final Output Latch
D.B. – Double Buffered
FIL – Final Input Latch

handshake pin options are available on H2 and H4. The $\overline{\text{DMAREQ}}$ pin may be associated with either input transfers (H3) or output transfers (H1), but not both. Refer to Table 8 for a summary of the Port B Data Register responses in Mode 2.

(2) Bit I/O (Port A)

Mode 2, Port A performs simple bit I/O with no associated handshake pins. This configuration is intended for applications in which several independent devices must be controlled or monitored. Data written to the Port A data register is single-buffered. If the Port A Data Direction Register bit for that pin is 1 (output), the output buffer is enabled. If it is 0, data written is still latched but not available at the pin. Data read from the data register is either the instantaneous value of the pin or what was written to the data register, depending on the contents of the Port A Data Direction Register. This is summarized in Table 9.

Table 9 Mode 2 Port A Data Paths

Mode	Read Port A Data Register		Write Port A Data Register	
	DDR = 0	DDR = 1	DDR = 0	DDR = 1
2	Pin	FOL	FOL	FOL, S.B.

Abbreviations:

S.B. – Single Buffered
FOL – Final Output Latch
DDR – Data Direction Register

- **Mode 3 – Bidirectional 16-Bit Double Buffered I/O**

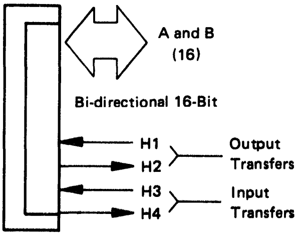


Figure 17 Mode 3

In Mode 3, Ports A and B are used for bidirectional 16-bit double-buffered transfers. H1 and H2 control output transfers, while H3 and H4 control input transfers. (H1 and H2 are enabled by the H12 Enable bit while H3 and H4 are enabled by the H34 Enable bit of the Port General Control Register.) The instantaneous direction of the data is determined by the H1 handshake pin, and thus, the data direction registers are not used. The Port A and Port B submode fields do not affect PI/T operation in Mode 3.

The only aspect of bidirectional double-buffered transfers that differs from the unidirectional modes lies in controlling the Port A and B output buffers. They are controlled by the level of H1. When H1 is negated, the output buffers (all 16) are enabled and the pins drive the bidirectional bus. Generally, H1 is negated in response to an asserted H2, which indicates that new output data is present in the double-buffered latches. Following acceptance of the data, the peripheral asserts H1, disabling the output buffers. Other than controlling the output buffers, H1 is edge-sensitive as in other modes. Input transfers proceed identically to the double-buffered input protocol described in the Port General Information and Conventions section. Port A and B data is latched with the asserted edge of H3. In Mode 3, only the interlocked and pulsed handshake pin options are available to H2 and H4. The $\overline{\text{DMAREQ}}$ pin may be associated with either input transfers (H3) or output transfers (H1), but not both. H2 indicates when new data is available in the Port B (and implicitly Port A) output latches, but unless the buffer is enabled by H1, the data is not driving the pins.

Mode 3 can provide convenient high-speed 16-bit transfers. The Port A and B Data Registers are addressed for compatibility with the HD68000's Move Peripheral (MOVEP) instruction and with the HD68450 DMAC. To take advantage of this, Port A should contain the most-significant data and always be read or written by the bus master first. The interlocked and pulsed handshake protocols are keyed to accesses to the Port B Data Register in Mode 3. If it is accessed last, the 16-bit double-buffered transfer proceed smoothly. Refer to Table 10 for a summary of the Port A and B data paths in Mode 3.

■ DMA REQUEST OPERATION

The Direct Memory Access Request ($\overline{\text{DMAREQ}}$) pulse can be associated with output or input transfers to keep the initial and final output latches full or initial and final input latches empty respectively. Figure 18 and 19 show all the possible paths in generating DMA requests.

Table 10 Mode 3 Port A and B Data Paths

Mode	Read Port A and B Data Register	Write Port A and B Data Register
3	FIL, D.B.	IOL/FOL. D.B., Note 1

Note 1: Data written to Port A goes to a temporary latch. When the Port B data register is later written. Port A data is transferred to IOL/FOL.

Abbreviations:

IOL –	Initial Output Latch	S.B. –	Single Buffered
FOL –	Final Output Latch	D.B. –	Double Buffered
FIL –	Final Input Latch		

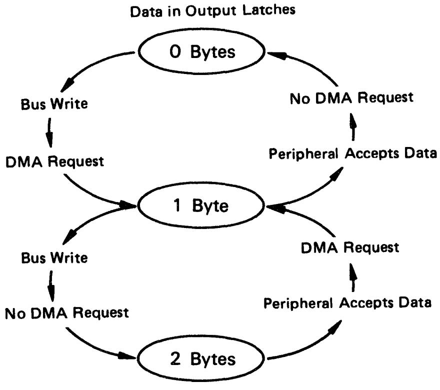


Figure 18 $\overline{\text{DMAREQ}}$ Associated with Output Transfers

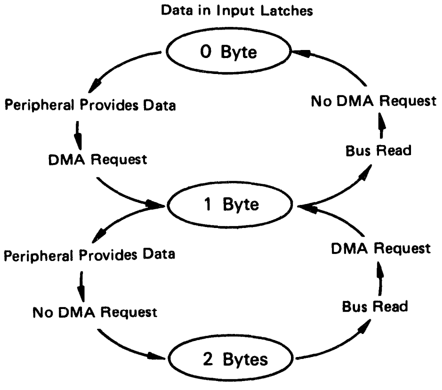


Figure 19 $\overline{\text{DMAREQ}}$ Associated with Input Transfers

■ TIMER

The HD68230 timer can provide several facilities needed by HD68000 operating systems. It can generate periodic interrupts, a square wave, or a single interrupt after a programmed time period. Also, it can be used for elapsed time measurement or as a device watchdog. This section describes the programmable options available, capabilities, and restrictions that apply to the timer.

The PI/T timer contains a 24-bit synchronous down counter that is loaded from three 8-bit Counter Preload Registers. The 24-bit counter may be clocked by the output of a 5-bit (divide-by-32) prescaler or by an external timer input TIN. If the prescaler is used, it may be clocked by the system clock (CLK pin) or by the TIN external input. The counter signals the occurrence of an event primarily through zero detection. (A zero is when the counter of the 24-bit timer is equal to zero.) This sets the zero detect status (ZDS) bit in the Timer Status Register. It may be checked by the processor or may be used to generate a timer interrupt. The ZDS bit is reset by writing a 1 to the Timer Status Register in that bit position.

The general operation of the timer is flexible and easily programmable. The timer is fully configured and controlled by programming the 8-bit Timer Control Register. It controls:
(1) the choice between the Port C operation and the timer operation of three timer pins,
(2) whether the counter is loaded from the Counter Preload Register or rolls over when zero detect is reach,
(3) the clock input,
(4) whether the prescaler is used, and
(5) whether the timer is enabled.

• RUN/HALT Definition

The overall operation of the timer is described in terms of the run or halt states. The control of the current state is determined by programming the Timer Control Register. When in the halt state, all of the following occur.
(1) The prior contents of the counter is not altered and is reliably readable via the Count Registers.
(2) The prescaler is forced to $1F whether or not it is used.
(3) The ZDS status bit is forced to 0, regardless of the possible zero contents of the 24-bit counter.

The run state is characterized by:
(1) The counter is clocked by the source programmed in the Timer Control Register.
(2) The counter is not reliably readable.
(3) The prescaler is allowed to decrement if programmed for use.
(4) The ZDS status bit is set when the 24-bit counter transitions from $000001 to $000000.

- **Timer Rules**

This section provides a set of rules that allow easy application of the timer.

(1) When the RESET pin is asserted, all bits of the Timer Control Register go to 0, configuring the dual function pins as Port C inputs.
(2) The contents of the Counter Preload Registers and counter are not affected by the RESET pin.
(3) The Count Registers provide a direct read data path from each portion of the 24-bit counter, but data written to their addresses is ignored. (This results in a normal bus cycle.) These registers are readable at any time, but their contents are never latched. Unreliable data may be read when the timer is in the run state.
(4) The Counter Preload Registers are readable and writable at any time and this occurs independently of any timer operation. No protection mechanisms are provided against ill-timed writes.
(5) The input frequency to the 24-bit counter from the TIN pin or prescaler output, must be between 0 and the input frequency at CLK pin divided by 32 regardless of the configuration chosen.
(6) For configurations in which the prescaler is used (with the CLK pin or TIN pin as an input), the contents of the Counter Preload Register (CPR) is transferred to the counter the first time that the prescaler passes from $00 to $1F (rolls over) after entering the run state. Thereafter, the counter decrements or is loaded from the Counter Preload Register when the prescaler rolls over.
(7) For configurations in which the prescaler is not used, the contents of the Counter Preload Registers are transferred to the counter on the first asserted edge of the TIN input after entering the run state. On subsequent asserted edges the counter decrements or is loaded from the Counter Preload Registers.
(8) The lowest value allowed in the Counter Preload Register for use with the counter is $000001.

- **Timer Interrupt Acknowledge Cycles**

Several conditions may be present when the timer interrupt acknowledge pin (TIACK) is asserted. These conditions affect the PI/T's response and the termination of the bus cycle. (see Table 11)

Table 11 Response to Timer Interrupt Acknowledge

PC3/TOUT Function	Response to Asserted TIACK
PC3 – Port C Pin	No response. No DTACK.
TOUT – Square Wave	No response. No DTACK.
TOUT – Negated Timer Interrupt Request	No response. No DTACK.
TOUT – Asserted Timer Interrupt Request	Timer Interrupt Vector Contents. DTACK Asserted.

- **TIMER APPLICATIONS SUMMARY**

This section outlines programming of the Timer Control Register for several typical examples.

(1) Periodic Interrupt Generator

7	6	5	4	3	2	1	0
TOUT/TIACK Control			Z.D. Ctrl.	*	Clock Control		Timer Enable
1	X	1	0	0	00 or 1X		changed

In this configuration the timer generates a periodic interrupt. The TOUT pin is connected to the system's interrupt request circuitry and the TIACK pin may be used as an interrupt acknowledge input to the timer. The TIN pin may be used as a clock input.

The processor loads the Counter Preload Registers and Timer Control Register, and then enables the timer. When the 24-bit counter passes from $000001 to $000000 the ZDS status bit is set and the TOUT (interrupt request) pin is asserted. At the next clock to the 24-bit counter it is again loaded with the contents of the CPR's, and thereafter decrements. In normal operation, the processor must direct clear the status bit to negate the interrupt request. (Figure 20)

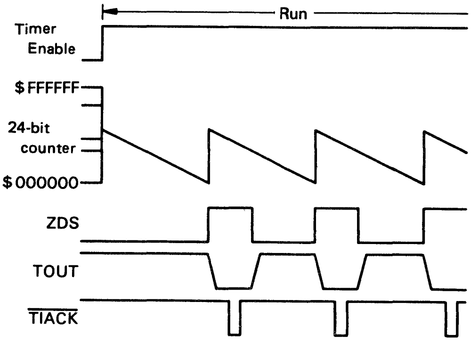


Figure 20 Periodic Interrupt Generator

(2) Square Wave Generator

Square Wave Generator

7	6	5	4	3	2	1	0
TOUT/TIACK Control			Z.D. Ctrl.	*	Clock Control		Timer Enable
0	1	X	0	0	00 or 1X		changed

In this configuration the timer produces a square wave at the TOUT pin. The TOUT pin is connected to the user's circuitry and the TIACK pin is not used. The TIN pin may be used as a clock input.

The processor loads the Counter Preload Registers and Timer Control Register, and then enables the timer. When the 24-bit counter passes from $000001 to $000000 the ZDS status bit is set and the TOUT (square wave output) pin is toggled. At the next clock to the 24-bit counter it is again loaded with the contents of the CPRs, and thereafter decrements. In this application there is no need for the processor to direct clear the ZDS status bit; however, it is possible for the processor to sync itself with the square wave by clearing the ZDS status bit, then polling it. The processor may also read the TOUT level at the Port C address.

Note that the PC_3/TOUT pin functions as PC3 following the negation of $\overline{\text{RESET}}$. If used in the square wave configuration a pullup resistor may be required to keep a known level prior to programming. Prior to enabling the timer, TOUT is high. (Figure 21)

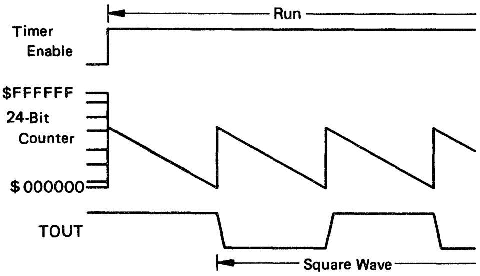


Figure 21 Square Wave Generator

(3) Interrupt After Timeout

Interrupt After Timeout

7	6	5	4	3	2	1	0
TOUT/$\overline{\text{TIACK}}$ Control			Z.D. Ctrl.	*	Clock Control		Timer Enable
1	X	1	1	0	00 or 1X		changed

In this configuration the timer generates an interrupt after a programmed time period has expired. The TOUT pin is connected to the system's interrupt request circuitry and the $\overline{\text{TIACK}}$ pin may be an interrupt acknowledge input to the timer. The TIN pin may be used as a clock input.

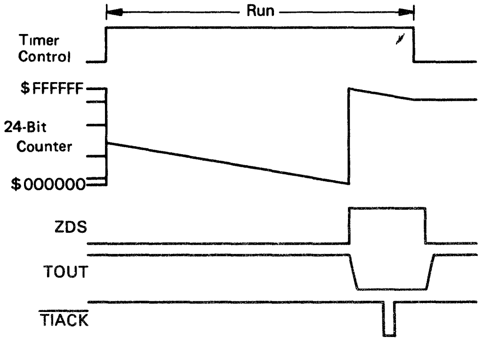


Figure 22 Interrupt After Timeout

This configuration is similar to the periodic interrupt generator except that the Zero Detect Control bit is set. This forces the counter roll over after Zero Detect is reached, rather than reloading from the CPRs. When the processor takes the interrupt it can halt the timer and read the counter. This allows the processor to measure the delay time from Zero Detect (interrupt request) to entering the service routine. Accurate knowledge of the interrupt latency may be useful in some applications. (Figure 22)

- **Elapsed Time Measurement**

Elapsed time measurement takes several forms; two are described below.

(1) System Clock

System Clock

7	6	5	4	3	2	1	0
TOUT/$\overline{\text{TIACK}}$ Control			Z.D. Ctrl.	*	Clock Control		Timer Enable
0	0	X	1	0	0	0	changed

This configuration allows time interval measurement by software. No timer pins are used.

The processor loads the Counter Preload Registers (generally with all 1s) and Timer Control Register, and then enables the timer. The counter decrements until the ending event takes place. When it is desired to read the time interval, the processor must halt the timer, then read the counter.

For applications in which the interval could have exceeded that programmable in this timer, interrupts can be counted to provide the equivalent of additional timer bits. At the end, the timer can be halted and read.

(2) External Clock

External Clock

7	6	5	4	3	2	1	0
TOUT/$\overline{\text{TIACK}}$ Control			Z.D. Ctrl.	*	Clock Control		Timer Enable
0	0	X	1	0	1	X	changed

This configuration allows measurement (counting) of the number of input pulses occurring in an interval in which the counter is enabled. The TIN input pin provides the input pulses. Generally the TOUT and $\overline{\text{TIACK}}$ pins are not used.

This configuration is identical to the Elapsed Time Measurement/System Clock configuration except that the TIN pin is used to provide the input frequency. It can be connected to a simple oscillator, and the same methods could be used. Alternately, it could be gated off and on externally and the number of cycles occurring while in the run state can be counted. However, minimum pulse width high and low specifications must be met. (Figure 23)

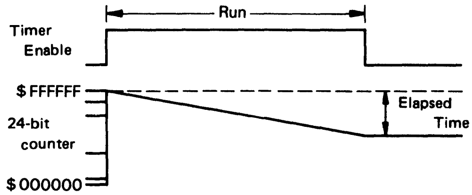


Figure 23 Elapsed Time Measurement

- **Device Watchdog**

Device Watching

7	6	5	4	3	2	1	0
TOUT/$\overline{\text{TIACK}}$ Control			Z.D. Ctrl.	*	Clock Control		Timer Enable
1	X	1	1	0	0	1	changed

This configuration provides the watchdog function needed in many systems. The TIN pin is the timer input whose period at the high (1) level is to be checked. Once allowed by the processor, the TIN input pin controls the run/halt mode. The TOUT pin is connected to external circuitry requiring notification when the TIN pin has been asserted longer than the programmed time. The $\overline{\text{TIACK}}$ pin (interrupt acknowledge) is only needed if the TOUT pin is connected to interrupt circuitry.

The processor loads the Counter Preload Register and Timer Control Register, and then enables the timer. When the TIN input is asserted (1, high) the timer transfers the contents of the Counter Preload Register to the counter and begins counting. If the TIN input is negated before Zero Detect is reached, the TOUT output and the ZDS status bit remain negated. If Zero Detect is reached while the TIN input is still asserted the ZDS status bit is set and the TOUT output is asserted. (The counter rolls over and keeps on counting.)

In either case, when the TIN input is negated the ZDS status bit is 0, the TOUT output is negated, the counting stops, and the prescaler is forced to all 1s. (Figure 24)

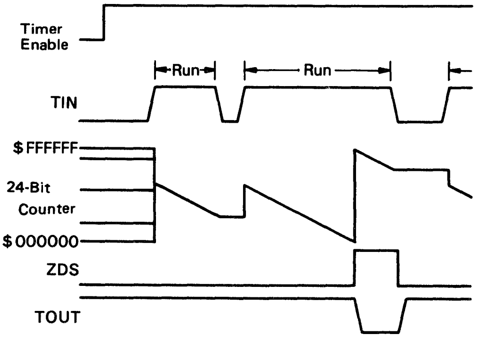


Figure 24 Device Watchdog

BUS INTERFACE CONNECTION

The PI/T has an asynchronous bus interface, primarily designed for use with the HD68000 microprocessor. With care, however, it can be connected to synchronous microprocessor buses. This section completely describes the PI/T's bus interface, and is intended for the asynchronous bus designer unless otherwise mentioned.

In an asynchronous system the PI/T CLK may operate at a significantly different frequency, either higher or lower, than the bus master and other system components, as long as all bus specifications are met. The HD68230 CLK pin has the same specifications as the HD68000 CLK, and must not be gated off at any time.

The following signals generate normal read and write cycles to the PI/T: $\overline{\text{CS}}$ (Chip Select), R/$\overline{\text{W}}$ (Read/Write). RS1-RS5 (five Register Select bits), D_0-D_7 (the 8-bit bidirectional data bus), and $\overline{\text{DTACK}}$ (Data Transfer Acknowledge). To generate interrupt acknowledge cycles PC_5/$\overline{\text{PIACK}}$ or PC_7/$\overline{\text{TIACK}}$ is used instead of CS, and the Register Select pins are ignored. No combination of the following pins may be asserted simultaneously: $\overline{\text{CS}}$, $\overline{\text{PIACK}}$, or $\overline{\text{TIACK}}$.

- **Read Cycles Via Chip Select**

This category includes all register reads, except port or timer interrupt acknowledge cycles. When $\overline{\text{CS}}$ is asserted, the Register Select and R/$\overline{\text{W}}$ inputs are latched internally. They must meet small setup and hold time requirements with respect to the asserted edge of $\overline{\text{CS}}$. (See the AC ELECTRICAL CHARACTERISTICS table.) The PI/T is not protected against aborted (shortened) bus cycles generated by an Address Error or Bus Error exception in which it is addressed.

Certain operations triggered by normal read (or write) bus cycles are not complete within the time allotted to the bus cycle. On example is transfers to/from the double-buffered latches that occur as a result of the bus cycle. If the bus master's CLK is significantly faster than the PI/T's the possibility exists that, following the bus cycle, $\overline{\text{CS}}$ can be negated then re-asserted before completion of these internal operations. In this situation the PI/T does not recognize the re-assertion of $\overline{\text{CS}}$ until these operations are complete. Only at that time does it begin the internal sequencing necessary to react to the asserted $\overline{\text{CS}}$. Since $\overline{\text{CS}}$ also controls the $\overline{\text{DTACK}}$ response, this "bus cycle recovery time" can be related to the CLK edge on which $\overline{\text{DTACK}}$ is asserted for that cycle. The PI/T will recognize the subsequent assertion of $\overline{\text{CS}}$ three (3) CLK periods after the CLK edge on which $\overline{\text{DTACK}}$ was previously asserted.

The Register Select and R/$\overline{\text{W}}$ inputs pass through an internal latch that is transparent when the PI/T can recognize a new $\overline{\text{CS}}$ pulse (see above paragraph). Since the internal data bus of the PI/T is continuously enabled for read transfers, the read access time (to the data bus buffers) begins when the Register Selects are stabilized internally. Also, when the PI/T is ready to begin a new bus cycle, the assertion of $\overline{\text{CS}}$ enables the data bus buffers within a short propagation delay. This does not contribute to the overall read access time unless $\overline{\text{CS}}$ is asserted significantly after the Register Select and R/$\overline{\text{W}}$ inputs are stabilized (as may occur with synchronous bus microprocessors).

. In addition to Chip Select's previously mentioned duties, it controls the assertion of $\overline{\text{DTACK}}$ and latching of read data at the data bus interface. Except for controlling input latches and enabling the data bus buffers, all of these functions occur only after $\overline{\text{CS}}$ has been recognized internally and synchronized with the internal clock. Chip Select is recognized on the falling edge of the CLK if the setup time is met, $\overline{\text{DTACK}}$ is asserted (low) on the next falling edge of the CLK. Read data is latched at the PI/T's data bus interface at the same time $\overline{DTACK}$ is asserted. It is stable as long as Chip Select remains asserted independent of other external conditions.

From the above discussion it is clear that if the $\overline{CS}$ setup time prior to the falling edge of the CLK is met, the PI/T can consistently respond to a new read or write bus cycle every four (4) CLK cycles. This fact is especially useful in designing the PI/T's clock in synchronous bus systems not using $\overline{DTACK}$. (An extra CLK period is required in interrupt acknowledge cycles, see Read Cycles via Interrupt Acknowledge.)

In asynchronous bus systems in which the PI/T's CLK differs from that of the bus master, generally there is no way to guarantee that the $\overline{CS}$ setup time with respect to the PI/T CLK is met. Thus, the only way to determine that the PI/T recognized the assertion of $\overline{CS}$ is to wait for the assertion of $\overline{DTACK}$. In this situation, all latched bus inputs to the PI/T must be held stable until $\overline{DTACK}$ is asserted. These include Register Select, R/$\overline{W}$, and write data inputs (see below).

System specifications impose a maximum delay from the trailing (negated) edge of Chip Select to the negated edge of $\overline{DTACK}$. As system speeds increase this becomes more difficult to meet with a simple pullup resistor tied to the $\overline{DTACK}$ line. Therefore, the PI/T provides an internal active pullup device to reduce the rise time, and a level-sensitive circuit that later turns this device off. $\overline{DTACK}$ is negated asynchronously as fast as possible following the rising edge of Chip Select, then three-stated to avoid interference with the next bus cycle.

The system designer must take care that $\overline{DTACK}$ is negated and three-stated quickly enough after each bus cycle to avoid interference with the next one. With the HD68000 this necessitates a relatively fast external path from the data strobe to $\overline{CS}$ going negated.

- **Write Cycles**

In many ways write cycles are similar to normal read cycles (see above). On write cycles, data at the D_0-D_7 pins must meet the same setup specifications as the Register Select and R/$\overline{W}$ lines. Like these signals, write data is latched on the asserted edge of $\overline{CS}$, and must meet small setup and hold time requirements with respect to that edge. The same bus cycle recovery conditions exist as for normal read cycles. No other differences exist.

- **Read Cycles Via Interrupt Acknowledge**

Special internal operations take place on PI/T interrupt acknowledge cycles. The Port Interrupt Vector Register or the Timer Interrupt Vector Register are implicitly addressed by the assertion of $PC_6/\overline{PIACK}$ or $PC_7/\overline{TIACK}$, respectively. The signals are first synchronized with the falling edge of the CLK. One clock period after they are recognized the data bus buffers are enabled and the vector is driven onto the bus. $\overline{DTACK}$ is asserted after another clock period to allow the vector some setup time prior to $\overline{DTACK}$.$\overline{DTACK}$ is negated, then three-stated as with normal read or write cycle, when $\overline{PIACK}$ or $\overline{TIACK}$ is negated.

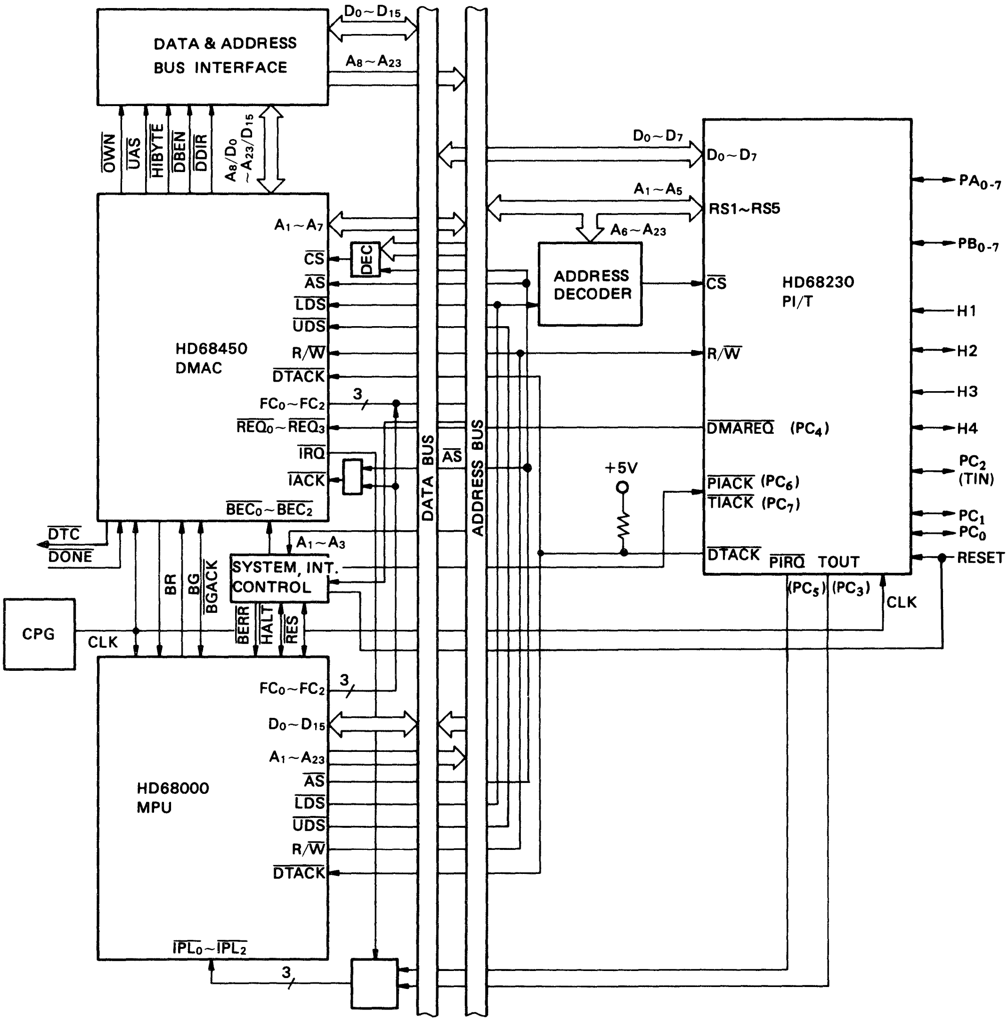


Figure 25 HD68000 Interface-Example

RESTRICTION ON HD68230 USAGE

The parallel interface/timer (PI/T) exhibits an anomaly during interrupt acknowledge ($\overline{\text{IACK}}$) cycles for certain configurations of the part. If the PI/T is configured to have only one interrupt source (either the port or the timer), and pins 36 (PC_6/$\overline{\text{PIACK}}$) and 37 (PC_6/$\overline{\text{TIACK}}$) are both low during $\overline{\text{IACK}}$ cycles, an incorrect vector number will be placed on the data bus and the interrupt vector register corresponding to the IACK cycles will be changed.

Specifically, if:

the PI/T is programmed to generate a vectored timer interrupt (i.e., pin 33 is programmed as TOUT and pin 37 as $\overline{\text{TIACK}}$), and pin 36 is programmed to be a general-purpose input or output that is low during $\overline{\text{TIACK}}$ cycles.

or if

the PI/T is programmed to generate a vectored port interrupt (i.e., pin 35 is programmed as $\overline{\text{PIRQ}}$ and pin 36 as PIACK), and pin 37 is programmed to be a general-purpose input or output that is low during $\overline{\text{PIACK}}$ cycles,

then,

during $\overline{\text{IACK}}$ cycles, the PI/T will misinterpret the low signals present on pins 36 and 37 as simultaneously asserted $\overline{\text{PIACK}}$ and $\overline{\text{TIACK}}$ signals, which is an illegal condition.

There is both a hardware solution and a software solution for this anomaly:

Hardware: Insure that whichever of the two pins not programmed as an $\overline{\text{IACK}}$ input will be high during $\overline{\text{IACK}}$ cycles. For example, if pin 37 is used as $\overline{\text{TIACK}}$ and pin 36 is programmed as a PC_6 input, force pin 36 high whenever pin 37 is low. This can be accomplished with either a pullup resistor or external logic.

Software: If only timer interrupts are to be used, initialize the PIVR with \$FC and select a vector number for the TIVR that has the two least significant bits clear (i.e., binary xxxxxx00). If only port interrupts are to be used, initialize the TIVR with \$FF and select any vector number for the PIVR.

Note that this anomaly will not arise if the PI/T interrupts are autovectored (since no $\overline{\text{IACK}}$ signal will be required) or if the PI/T is programmed to accept both port and timer interrupt acknowledges (since external $\overline{\text{IACK}}$ logic will insure that pins 36 and 37 are never low simultaneously).