# Z8536 CIO Counter/Timer and Parallel I/O Unit

## Product Specification

June 1982

**Features**

- Two independent 8-bit, double-buffered, bidirectional I/O ports plus a A-bit special-purpose I/O port. I/O ports feature programmable polarity, programmable direction (Bit mode), "pulse. catchers," and programmable open-drain outputs.
- Four handshake modes, including 3-Wire (like the IEEE-488).
- m REQUEST/WAIT signal for high-speed data transfer.
- Flexible pattern-recognition logic, programmable as a 16-vector interrupt controller.
- Three independent 16-bit counter/timers with up to four external access lines per counter/timer (count input, output, gate, and trigger), and three output duty cycles (pulsed, one-shot, and square-wave), programmable as retriggerable or nonretriggerable.
- Easy to use since all registers are read/write.

**General Description**

The Z8536 CIO Counter/Timer and Parallel I/O element is a general-purpose peripheral circuit, satisfying most counter/timer and parallel I/O needs encountered in system designs. This versatile device contains three I/O ports and three counter/timers. Many programmable options tailor its configuration to specific applications. The use of the device is simplified by making all internal registers (command, status, and data) readable and (except for status bits) writable. In addition, each register is given its own unique internal address, so that any register can be accessed in two operations. All data registers can be directly accessed in a single operation. The CIO is easily interfaced to all popular microprocessors.

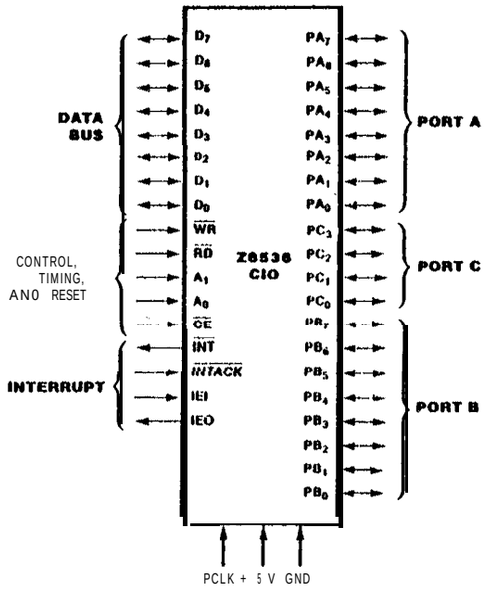



Figure 1. Pin Functions

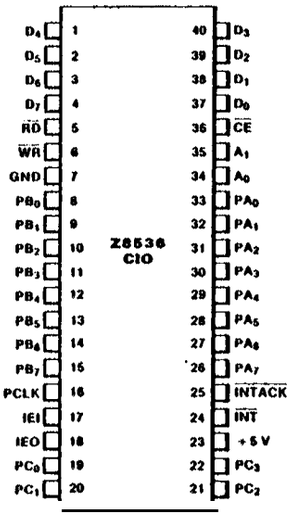



Figure 2. Pin Assignments

## Pin Description

**A0-Al.** *Address Lines* (input). These two lines are used to select the register involved in the CPU transaction: Port A's Data register, Port B's Data register, Port C's Data register, or a control register.

**CE** *Chip Enable* (input, active Low). A Low level on this input enables the CIO to be read from or written to.

**D0-D7** Data Bus (bidirectional 3-state). These eight data lines are used for transfers between the CPU and the CIO.

**IEI.** *Interrupt Enable In* (input, active High). IEI is used with IEO to form an interrupt daisy chain when there is more than one interrupt-driven device. A High IEI indicates that no other higher priority device has an interrupt under service or is requesting an interrupt.

**IEO.** *Interrupt Enable Out* (output, active High). IEO is High only if IEI is High and the CPU is not servicing an interrupt from the requesting CIO or is not requesting an interrupt (Interrupt Acknowledge cycle only). IEO is connected to the next lower priority device's IEI input and thus inhibits interrupts from lower priority devices.

**INT** *Interrupt Request* (output, open-drain, active Low). This signal is pulled Low when the CIO requests an interrupt.

**$\overline{\text{INTACK}}$.** *Interrupt Acknowledge* (input, active Low). This input indicates to the CIO that an Interrupt Acknowledge cycle is in progress $\overline{\text{INTACK}}$ must be synchronized to PCLK, and it must be stable throughout the Interrupt Acknowledge cycle.

**PA0-PA7** *Port A* I/O *lines* (bidirectional, 3-state, or open-drain). These eight I/O lines transfer information between the CIO's Port A and external devices.

**PB0-PB7** *Port B* I/O *lines* (bidirectional, 3-state, or open-drain). These eight I/O lines transfer information between the CIO's Port B and external devices. May also be used to provide external access to Counter/Timers 1 and 2.

**PC0-PC3** *Port* C I/O *lines* (bidirectional, 3-state, or open-drain). These four I/O lines are used to provide handshake, WAIT, and BEQUEST lines for Ports A and B or to provide external access to Counter/Timer 3 or access to the CIO's Port C.

**PCLK.** *Peripheral Clock* (input, TTL-compatible). This is the clock used by the internal control logic and the counter/timers in timer mode. It does not have to be the CPU clock.

**$\overline{\text{RD}}$** *Read* (input, active Low). This signal indicates that a CPU is reading from the CIO. During an Interrupt Acknowledge cycle, this signal gates the interrupt vector onto the data bus if the CIO is the highest priority device requesting an interrupt.

**$\overline{\text{WR}}$*** *Write* (input, active Low). This signal indicates a CPU write to the CIO.

*When RD and WR are detected Low at the same time (normally an illegal condition), the CIO is reset.

## Architecture

The CIO Counter/Timer and Parallel I/O element (Figure 3) consists of a CPU interface, three I/O ports (two general-purpose 8-bit ports and one special-purpose 4-bit port),

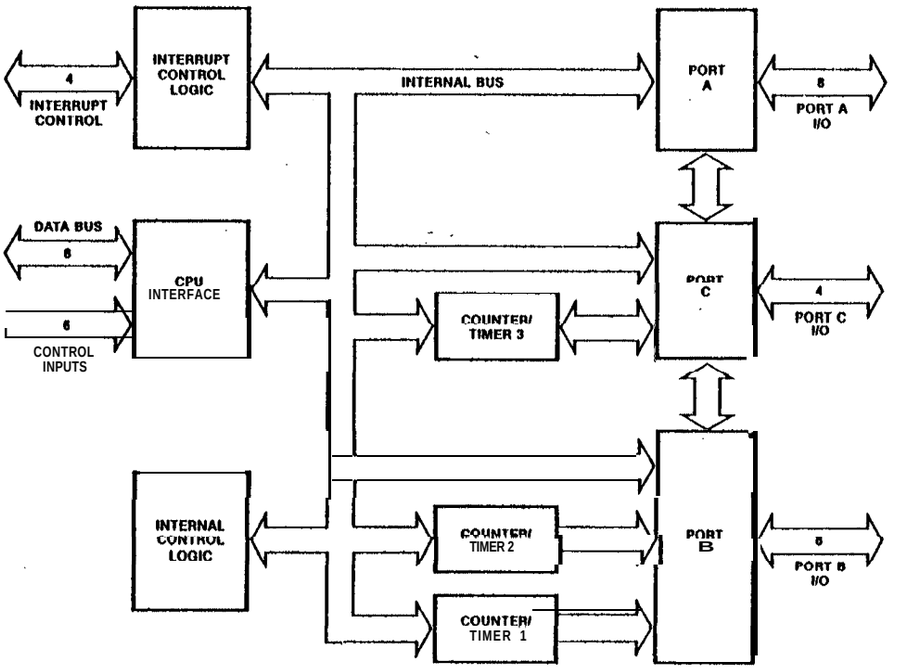



**Figure 3. CIO Block Diagram**

**Architecture**
(Continued)

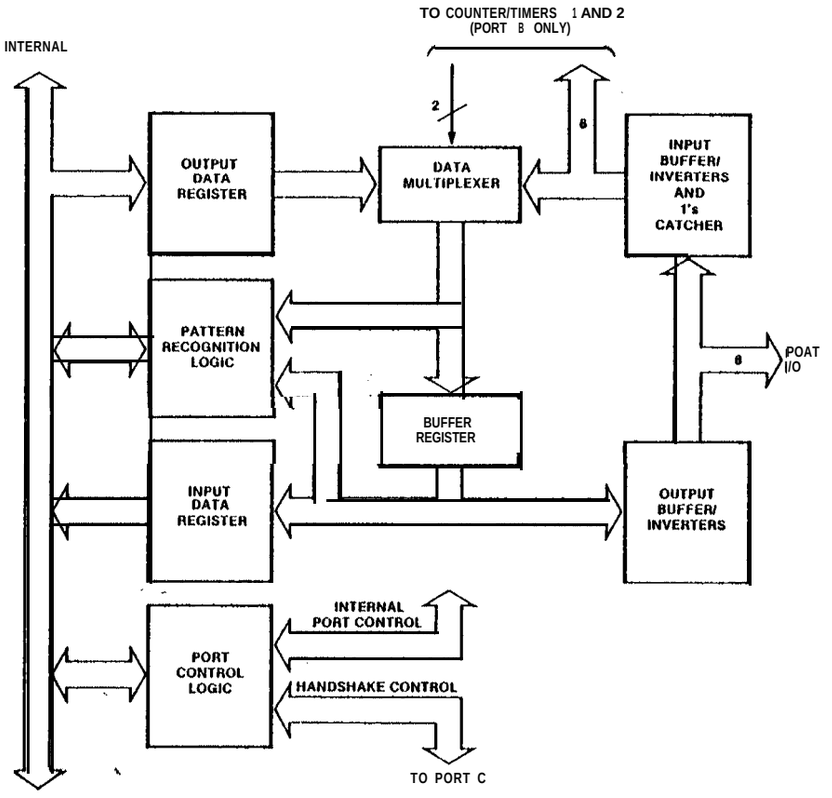



Figure 4. Ports A and B Block Diagram

three 16-bit counter/timers, an interrupt-control logic block, and the internal-control logic block. An extensive number of programmable options allow the user to tailor the configuration to best suit the specific application.

The two general-purpose 8-bit I/O ports (Figure 4) are identical, except that Port B can be specified to provide external access to Counter/Timers 1 and 2. Either port can be programmed to be a handshake-driven, double-buffered port (input, output, or bidirectional) or a control-type port with the direction of each bit individually programmable. Each port includes pattern-recognition logic, allowing interrupt generation when a specific pattern is detected. The pattern-recognition logic can be programmed so the port functions like a priority-interrupt controller. Ports A and B can also be linked to form a 16-bit I/O port.

To control these capabilities, both ports contain 12 registers. Three of these registers, the Input, Output, and Buffer registers, comprise the data path registers. Two registers, the Mode Specification and Handshake Specification registers, are used to define the mode of the port and to specify which handshake, if any, is to be used. The reference pattern for the pattern-recognition logic is defined via three registers: the Pattern Polarity, Pattern Transition, and Pattern Mask registers. The detailed characteristics of each bit path (for example, the direction of data flow or whether a path is inverting or noninverting) are programmed using the Data Path Polarity, Data Direction, and Special I/O Control registers.

The primary control and status bits are grouped in a single register, the Command and Status register, so that after the port is initially configured, only this register must be accessed frequently. To facilitate initialization, the port logic is designed so that registers associated with an unrequired capability are ignored and do not have to be programmed.

The function of the special-purpose 4-bit port, Port C (Figure 5), depends upon the roles of Ports A and B. Port C provides the required handshake lines. Any bits of Port C not used as handshake lines can be used as I/O lines or to provide external access for the third counter/timer.

Since Port C's function is defined primarily by Ports A and B, only three registers (besides the Data Input and Output registers) are needed. These registers specify the details of each bit path: the Data Path Polarity, Data Direction, and Special I/O Control registers.

**Architecture**
(Continued)

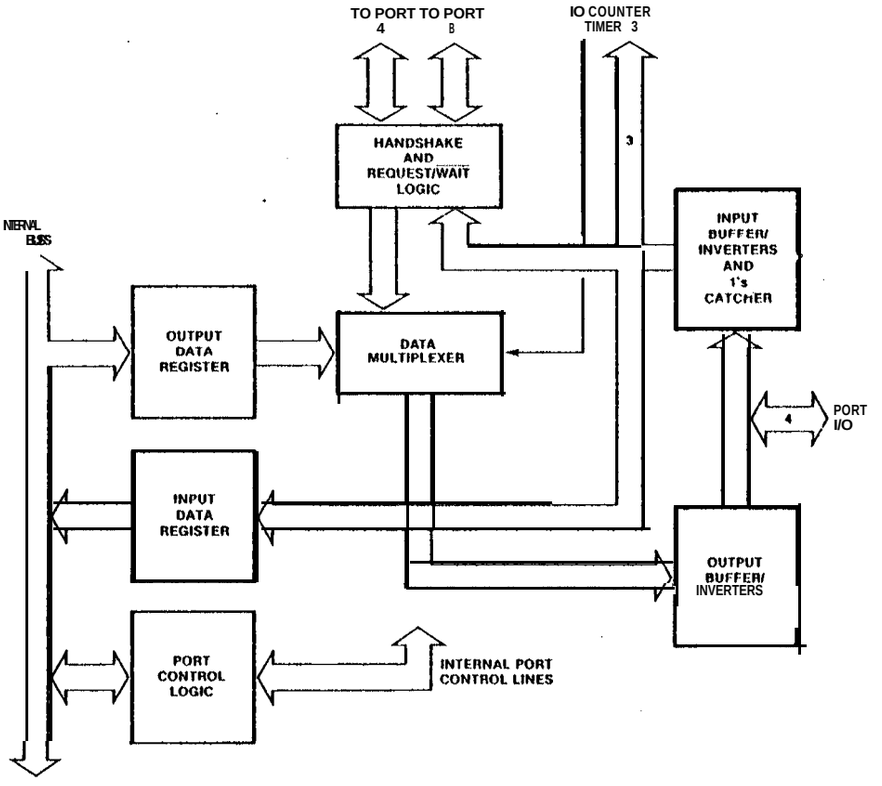



**Figure 5. Port C Block Diagram**

The three counter/timers (Figure 6) are all identical. Each is comprised of a 16-bit down-counter, a 16-bit Time Constant register (which holds the value loaded into the down-counter), a 16-bit Current Count register (used to read the contents of the down-counter), and two 8-bit registers for control and status (the Mode Specification and the Command and Status registers).

The capabilities of the counter/timer are numerous. Up to four port I/Olines can be dedicated as external access lines for each counter/timer: counter input, gate input, trigger input, and counter/timer output. Three different counter/timer output duty cycles are available: pulse, one-shot, or square-wave. The operation of the counter/timer can be programmed as either retriggerable or nonretriggerable. With these and other options, most counter/timer applications are covered.

There are five registers (Master Interrupt Control register, three Interrupt Vector registers, and the Current Vector register) associated with the interrupt logic. In additiun, the ports' Command and Status registers and the counter/timers' Command and Status registers include bits associated with the interrupt logic. Each of these registers contains three bits for interrupt control and status: Interrupt Pending (IP), Interrupt Under Service (IUS), and Interrupt Enable (IE).

**Architecture** (Continued)

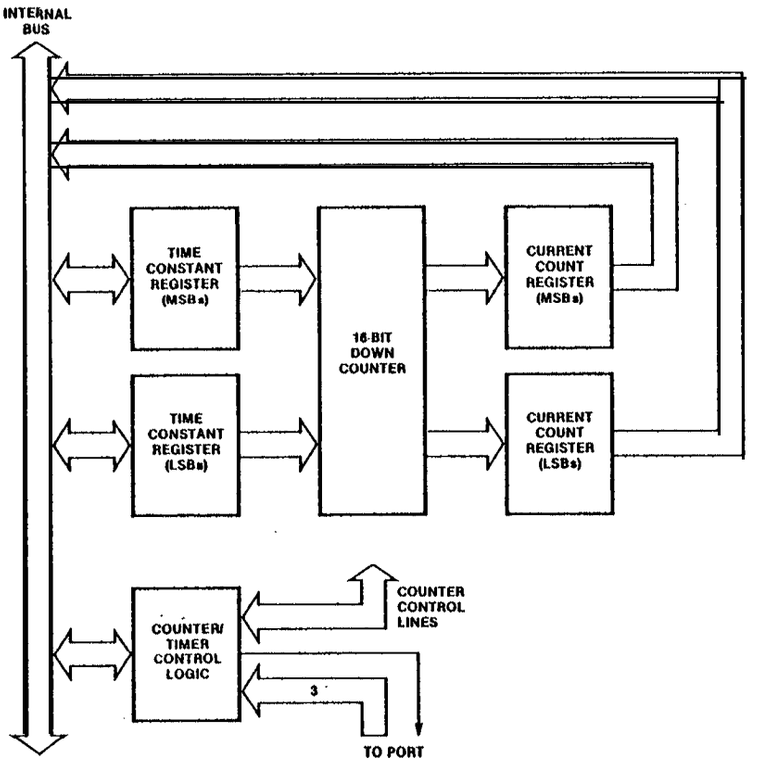



**Figure 6. Counter/Timer Block Diagram**

**Functional Description**

The following describes the functions of the ports, pattern-recognition logic, counter/timers, and interrupt logic.

**I/O Port Operations.** Of the CIO's three I/O ports, two (Ports A and B) are general-purpose, and the third (Port C) is a special-purpose 4-bit port. Ports A and B can be configured as input, output, or bidirectional ports with handshake. (Four different handshakes are available.) They can also be linked to form a single 16-bit port. If they are not used as ports with handshake, they provide 16 input or output bits with the data direction programmable on a bit-by-bit basis. Port B also provides access for Counter/Timers 1 and 2. In all configurations, Ports A and B can be programmed to recognize specific data patterns and to generate interrupts when the pattern is encountered.

The four bits of Port C provide the handshake lines for Ports A and B when required. A REQUEST/WAIT line can also be provided so that CIO transfers can be synchronized with DMAs or CPUs. Any Port C bits not used for handshake or REQUEST/WAIT can be used as input or output bits (individually data-direction programmable) or external access lines for Counter/Timer 3. Port C does not contain any pattern-recognition logic. It is, however, capable of bit-addressable writes. With this feature, any combination of bits can be set and/or cleared while the other bits remain undisturbed without first reading the register.

*Bit Port Operations.* In bit port operations, the port's Data Direction register specifies the direction of data flow for each bit. A 1 specifies an input bit, and a 0 specifies an output bit. If bits are used as I/O bits for a counter/timer, they should be set as input or output, as required.

The Data Path Polarity register provides the capability of inverting the data path. A 1 specifies inverting, and a 0 specifies noninverting. All discussions of the port operations assume that the path is noninverting.

The value returned when reading an input bit reflects the state of the input just prior to the read. A l's catcher can be inserted into the input data path by programming a 1 to the corresponding bit position of the port's Special I/O Control register. When a 1 is detected at the l's catcher input, its output is set to 1 until it is cleared. The I's catcher is cleared by writing a 0 to the bit. In all other cases, attempted writes to input bits are ignored.

When Ports A and B include output bits, reading the Data register returns the value being output. Reads of Port C return the state of the pin. Outputs can be specified as open-drain by writing a 1 to the corresponding bit of the port's Special I/O Control register. Port C has the additional feature of bit-addressable writes. When writing to Port C, the four most significant bits are used as a write protect mask for the least significant bits (O-4, l-5, 2-6, and 3-7). If the write protect bit is written with a 1, the state of the corresponding output bit is not changed.

**Functional Description** (Continued)

*Ports with Handshake Operation.* Ports A and B can be specified as 8-bit input, output, or bidirectional ports with handshake. The CIO provides four different handshakes for its ports: Interlocked, Strobed, Pulsed, and 3-Wire. When specified as a port with handshake, the transfer of data into and out of the port and interrupt generation is under control of the handshake logic. Port C provides the handshake lines as shown in Table 1. Any Port C lines not used for handshake can be used as simple I/O lines or as access lines for Counter/Timer 3.

When Ports A and B are configured as ports with handshake, they are double-buffered. This allows for more relaxed interrupt service routine response time. A second byte can be input to or output from the port before the interrupt for the first byte is serviced. Normally, the Interrupt Pending (IP) bit is set and an interrupt is generated when data is shifted into the Input register (input port) or out of the Output register (output port). For input and output ports, the IP is automatically cleared when the data is read or written. In bidirectional ports, IP is cleared only by command. When the Interrupt on Two Bytes (ITB) control bit is set to 1, interrupts are generated only when two bytes of data are available to be read or written. This allows a maximum of 16 bits of information to be transferred on each interrupt. With ITB set, the IP is not automatically cleared until the second byte of data is read or written.

When the Single Buffer (SB) bit is set to 1, the port acts as if it is only single-buffered. This is useful if the handshake line must be stopped on a byte-by-byte basis.

Ports A and B can be linked to form a 16-bit port by programming a 1 in the Port Link Control (PLC) bit. In this mode, only Port A's Handshake Specification and Command and Status registers are used. Port B must be specified as a bit port. When linked, only Port A has pattern-match capability. Port B's pattern-match capability must be disabled. Also, when the ports are linked, Port B's Data register must be read or written before Port A's.

When a port is specified as a port with handshake, the type of port it is (input, output, or bidirectional) determines the direction of data flow. The data direction for the bi-directional port is determined by a bit in Port C (Table 1). In all cases, the contents of the Data Direction register are ignored. The contents of the Special I/O Control register apply only to output bits (3-state or open-drain). Inputs may not have l's catchers; therefore, those bits in the Special I/O Control register are ignored. Port C lines used for handshake should be pro grammed as inputs. The handshake specification overrides Port C's Data Direction register for bits that must be outputs. The contents of Port C's Data Path Polarity register still apply.

**Interlocked Handshake.** In the Interlocked Handshake mode, the action of the CIO must be acknowledged by the external device before the next action can take place. Figure 7 shows timing for Interlocked Handshake. An output port does not indicate that new data is available until the external device indicates it is ready for the data. Similarly, an input port does not indicate that it is ready for new data until the data source indicates that the previous byte of the data is no longer available, thereby acknowledging the input port's acceptance of the last byte. This allows the CIO to interface directly to the port of a Z8 microcomputer, a UPC, an FIO, an FIFO, or to another CIO port with no external logic.

A 4-bit deskew timer can be inserted in the Data Available ($\overline{DAV}$) output for output ports. As data is transferred to the Buffer register, the deskew timer is triggered. After the number of PCLK cycles specified by the deskew timer time constant plus one, $\overline{DAV}$ is allowed to go Low. The deskew timer therefore guarantees that the output data is valid for a specified minimum amount of time before $\overline{DAV}$

| Port A/B Configuration | $PC_3$ | $PC_2$ | $PC_1$ | $PC_0$ |
|---|---|---|---|---|
| Ports A and B: Bit Ports | Bit I/O | Bit I/O | Bit I/O | Bit I/O |
| Port A: Input or Output Port (Interlocked, Strobed or Pulsed Handshake)* | RFD or DAV | ACKIN | REQUEST/WAIT or Bit I/O | Bit I/O |
| Port B: Input or Output Port (Interlocked, Strobed or Pulsed Handshake)* | REQUEST/WAIT or Bit I/O | Bit I/O | RFD or DAV | ACKIN |
| Port A or B: Input Port (3-Wire Handshake) | RFD (Output) | DAV (Input) | REQUEST/WAIT or Bit I/O | DAC (Output) |
| Port A or B: Output Port (3-Wire Handshake) | DAV (Output) | DAC (Output) | REQUEST/WAIT or Bit I/O | RFD (Input) |
| Port A or B: Bidirectional Port (Interlocked or Strobed Handshake) | RFD or DAV | ACKIN | REQUEST/WAIT or Bit I/O | IN/OUT |

*Both Ports A and B can be specified input or output with Interlocked, Strobed, or Pulsed Handshake at the same time if neither uses REOUEST/WAIT.

**Table 1. Port C Bit Utilization**

**Functional Description** (Continued)

goes Low. Deskew timers are available for output ports independent of the type of handshake employed.

Strobed Handshake. In the Strobed Handshake mode, data is "strobed" into or out of the port by the external logic. The falling edge of the Acknowledge Input (ACKIN) strobes data into or out of the port. Figure 7 shows timing for the Strobed Handshake. In contrast to the Interlocked handshake, the signal indicating the port is ready for another data transfer operates independently of the ACKIN input. It is up to the external logic to ensure that data overflows or underflows do not occur.

3-Wire Handshake. The 3-Wire Handshake is designed for the situation in which one output port is communicating with many input ports simultaneously. It is essentially the same as the Interlocked Handshake, except that two signals are used to indicate if an input port is ready for new data or if it has accepted the present data. In the 3-Wire Handshake (Figure 8), the rising edge of one status line indicates that the port is ready for data, and the rising edge of another status line indicates'that the data has been accepted. With the 3-Wire Handshake, the output lines of many input ports can be bussed together with open-drain drivers; the output port knows when all the ports have accepted the data and are ready. This is the same handshake as is used on the IEEE-488 bus. Because this handshake requires three lines, only one port (either A or B) can be a 3-Wire Handshake port at a time. The 3-Wire Handshake is not available in the bidirectional mode. Because the port's direction can be changed under software control, however, bidirectional IEEE-488-type transfers can be performed.

Pulsed Handshake. The Pulsed Handshake (Figure 9) is designed to interface to mechanical-type devices that require data to be held for long periods of time and need relatively wide pulses to gate the data into or out of the device. The logic is the same as the Interlocked Handshake mode, except that an internal counter/timer is linked to the handshake logic. If the port is specified in the input mode, the timer is inserted in the ACKIN path. The external ACKIN input triggers the timer and its output is used as the Interlocked Handshake's normal acknowledge input. If the port is an output port, the timer is placed in the Data Available (DAV) output path. The timer is triggered when the normal Interlocked Handshake DAV output goes Low and the timer output is used as the actual DAV output. The counter/timer maintains all of its normal capabilities. This handshake is not available to bidirectional ports.

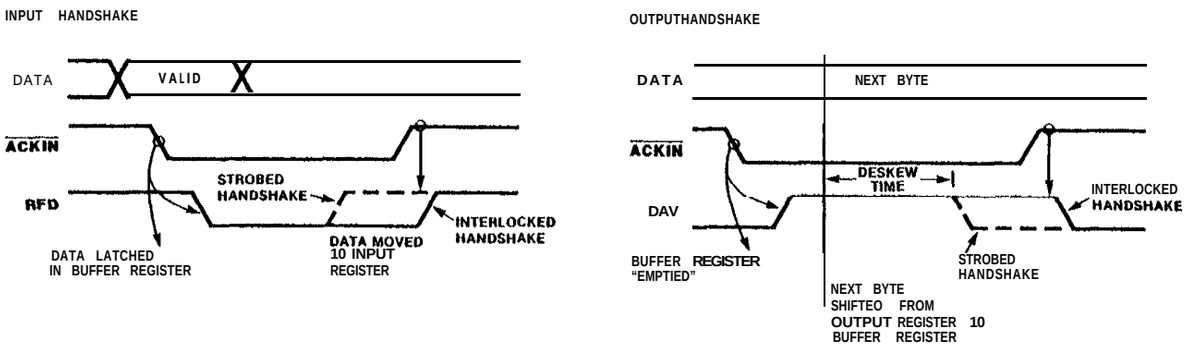



**Figure 7. Interlocked and Strobed Handshakes**

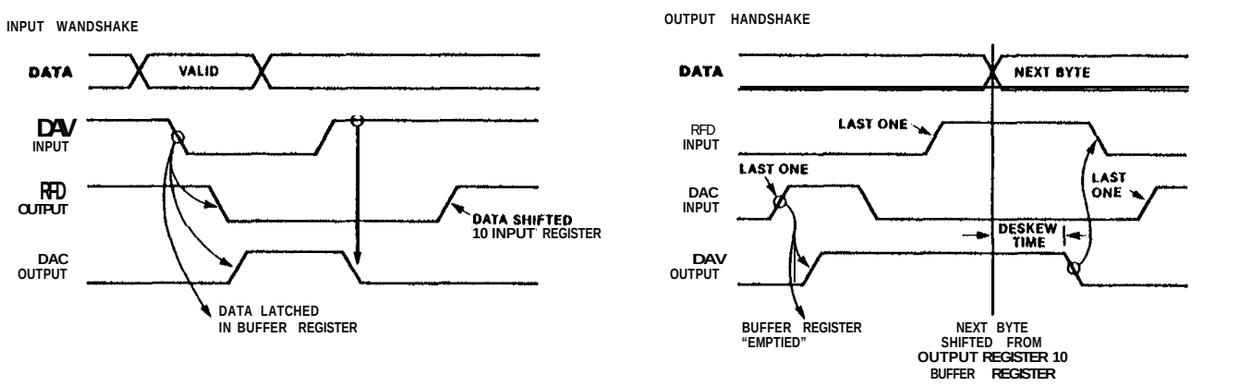



**Figure 8. 3-Wire Handshake**

**Functional Description** (Continued)

**REQUEST/WAIT Line Operation.** Port C can be programmed to provide a status signal output in addition to the normal handshake lines for either Port A or B when used as a port with handshake. The additional signal is either a REQUEST or WAIT signal. The REQUEST signal indicates when a port is ready to perform a data transfer via the CPU interface. It is intended for use with a DMA-type device. The WAIT signal provides synchronization for transfers with a CPU. Three bits in the Port Handshake Specification register provide controls for the REQUEST/WAIT logic. Because the extra Port C line is used, only one port can be specified as a port with a handshake and a REQUEST/WAIT line. The other port must be a bit port.

Operation of the REQUEST line is modified by the state of the port's Interrupt on Two Bytes (ITB) control bit. When ITB is 0, the REQUEST line goes active as soon as the CIO is ready for a data transfer. If ITB is 1, REQUEST does not go active until two bytes can be transferred. REQUEST stays active as long as a byte is available to be read or written.

The SPECIAL REQUEST function is reserved for use with bidirectional ports only. In this case, the REQUEST line indicates the status of the register not being used in the data path at -- that time. If the IN/OUT line is High, the REQUEST line is High when the Output register is empty. If IN/OUT is Low, the REQUEST line is High when the Input register is full.

*Pattern-Recognition Logic Operation.* Both Ports A and B can be programmed to generate interrupts when a specific pattern is recognized at the port. The pattern-recognition logic is independent of the port application, thereby allowing the port to recognize patterns in all of its configurations. The pattern can be independently specified for each bit as 1, 0, rising edge, falling edge, or any transition. Individual bits may be masked off. A pattern-match is defined as the simultaneous satisfaction of all nonmasked bit specifications in the AND mode or the satisfaction of any non-Masked bit specifications in either of the OR or OR-Priority Encoded Vector modes.

**Figure 9. Pulsed Handshake**

The pattern specified in the Pattern Definition register assumes that the data path is programmed to be noninverting. If an input bit in the data path is programmed to be inverting, the pattern detected is the opposite of the one specified. Output bits used in the pattern-match logic are internally sampled before the invert/noninvert logic.

**Bit Port Pattern-Recognition Operations.** During bit port operations, pattern-recognition may be performed on all bits, including those used as I/O for the counter/timers. The input to the pattern-recognition logic follows the value at the pins (through the invert/noninvert logic) in all cases except for simple inputs with l's catchers. In this case, the output of the l's catcher is used. When operating in the AND or OR mode, it is the transition from a no-match to a match state that causes the interrupt. In the "OR" mode, if a second match occurs before the first match goes away, it does not cause an interrupt. Since a match condition only lasts a short time when edges are specified, care must be taken to avoid losing a match condition. Bit ports specified in the OR-Priority Encoded Vector mode generate interrupts as long as any match state exists. A transition from a no-match to a match state is not required.

The pattern-recognition logic of bit ports operates in two basic modes: transparent and latched. When the Latch on Pattern Match (LPM) bit is set to 0 (Transparent mode), the interrupt indicates that a specified pattern has occurred, but a read of the Data register does not necessarily indicate the state of the port at the time the interrupt was generated. In the Latched mode (LPM= l), the state of all the port inputs at the time the interrupt was generated is latched in the input register and held until IP is cleared. In all cases, the PMF indicates the state of the port at the time it is read.

If a match occurs while IP is already set, an error condition exists. If the Interrupt On Error bit (IOE) is 0, the match is ignored. However, if IOE is 1 after the firs! IP is cleared, it is automatically set to 1 along with the Interrupt Error (ERR) flag. Matches occurring while ERR is set are ignored. ERR is cleared when the corresponding IP is cleared.

When a pattern-match is present in the OR-Priority Encoded Vector mode, IP is set to 1. The IP cannot be cleared until a match is no longer present. If the interrupt vector is allowed to include status, the vector returned during Interrupt Acknowledge indicates the highest priority bit matching its specification at the time of the Acknowledge cycle. Bit 7 is the highest priority and bit 0 is the lowest. The bit initially causing the interrupt may not be the one indicated by the vector if a higher priority bit matches before the Acknowledge. Once the

**Functional Description** (Continued)

Acknowledge cycle is initiated, the vector is frozen until the corresponding IP is cleared. Where inputs that cause interrupts might change before the interrupt is serviced, the l's catcher can be used to hold the value. Because a no-match to match transition is not required, the source of the interrupt must be cleared before IP is cleared or else a second interrupt is generated. No error detection is performed in this mode, and the Interrupt On Error bit should be set to 0.

**Ports with Handshake Pattern-Recognition Operation.** In this mode, the handshake logic normally controls the setting of IP and, therefore, the generation of interrupt requests. The pattern-match logic controls the Pattern-Match Flag (PMF). The data is compared with the match pattern when it is shifted from the Buffer register to the Input register (input port) or when it is shifted from the Output register to the Buffer register (output port). The pattern match logic can override the handshake logic in certain situations. If the port is programmed to interrupt when two bytes of data are available to be read or written, but the first byte matches the specified pattern, the pattern-recognition logic sets IP and generates an interrupt. While PMF is set, IP cannot be cleared by reading or writing the data registers. IP must be cleared by command. The input register is not emptied while IP is set, nor is the output register filled until IP is cleared.

If the Interrupt on Match Only (IMO) bit is set, IP is set only when the data matches the pattern. This is useful in DMA-type application when interrupts are required only after a block of data is transferred.

**Counter/Timer Operation.** The three independent 16-bit counter/timers consist of a presettable 16-bit down counter, a 16-bit Time Constant register, a 16-bit Current Counter register, an 8-bit Mode Specification register, an 8-bit Command and Status register, and the associated control logic that links these registers.

| Function | $C/T_1$ | $C/T_2$ | $C/T_3$ |
|---|---|---|---|
| Counter/Timer Output | PB 4 | PB 0 | PC 0 |
| Counter/Input | PB 5 | PB 1 | PC 1 |
| Trigger Input | PB 6 | PB 2 | PC 2 |
| Gate Input | PB 7 | PB 3 | PC 3 |

**Table 2. Counter/Timer External Access**

The flexibility of the counter/timers is enhanced by the provision of up to four lines per counter/timer (counter input, gate input, trigger input, and counter/timer output) for direct external control and status. Counter/Timer l's external I/O lines are provided by the four most significant bits of Port B. Counter/Timer 2's are provided by the four least significant bits of Port B. Counter/Timer 3's external I/O lines are provided by the four bits of Port C. The utilization of these lines (Table 2) is programmable on a bit-by-bit basis via the Counter/Timer Mode Specification registers.

When external counter/timer I/O lines are to be used, the associated port lines must be vacant and programmed in the proper data direction. Lines used for counter/timer I/O have the same characteristics as simple input lines. They can be specified as inverting or noninverting; they can be read and used with the pattern-recognition logic. They can also include the l's catcher input.

Counter/Timers 1 and 2 can be linked inter-Nally in three different ways. Counter/Timer l's output (inverted) can be used as Counter/Timer 2's trigger, gate, or counter input. When linked, the counter/timers have the same capabilities as when used separately. The only restriction is that when Counter/Timer 1 drives Counter/Timer 2's count input, Counter/Timer 2 must be programmed with its external count input disabled.

There are three duty cycles available for the timer/counter output: pulse, one-shot, and square-wave. Figure 10 shows the counter/timer waveforms. When the Pulse mode

Figure 10. Counter/Timer Waveforms

**Functional Description** (Continued)

is specified, the output goes High for one clock cycle, beginning when the down-counter leaves the count of 1. In the One-Shot mode, the output goes High when the counter/timer is triggered and goes Low when the down-counter reaches 0. When the square-wave output duty cycle is specified, the counter/timer goes through two full sequences for each cycle. The initial trigger causes the down-counter to be loaded and the normal countdown sequence to begin. If a 1 count is detected on the down-counter's clocking edge, the output goes High and the time constant value is reloaded. On the clocking edge, when both the down-counter and the output are l's, the output is pulled back Low.

The Continuous/Single Cycle (C/SC) bit in the Mode Specification register controls operation of the down-counter when it reaches terminal count. If C/SC is 0 when a terminal count is reached, the countdown sequence stops. If the C/SC bit is 1 each time the countdown counter reaches 1, the next cycle causes the time constant value to be reloaded. The time constant value may be changed by the CPU, and on reload, the new time constant value is loaded.

Counter/timer operations require loading the time constant value in the Time Constant register and initiating the countdown sequence by loading the down-counter with the time constant value. The Time Constant register is accessed as two 8-bit registers. The registers are readable as well as writable, and the access order is irrelevant. A 0 in the Time Constant register specifies a time constant of 65,536. The down-counter is loaded in one of three ways: by writing a 1 to the Trigger Command Bit (TCB) of the Command and Status register, on the rising edge of the external trigger input, or, for Counter/Timer 2 only, on the rising edge of Counter/Timer l's internal output if the counters are linked via the trigger input. The TCB is write-only, and read always returns 0.

Once the down-counter is loaded, the countdown sequence continues toward terminal count as long as all the counter/timers' hardware and software gate inputs are High. If any of the gate inputs goes Low (0), the countdown halts. It resumes when all gate inputs are 1 again.

The reaction to triggers occurring during a countdown sequence is determined by the state of the Retrigger Enable Bit (REB) in the Mode Specification register. If REB is 0, retriggers are ignored and the countdown continues normally. If REB is 1, each trigger causes the down-counter to be reloaded and the countdown sequence starts over again. If the output is programmed in the Square-Wave mode, retrigger causes the sequence to start over from the initial load of the time constant.

The rate at which the down-counter counts is determined by the mode of the counter/timer. In the Timer mode (the External Count Enable [ECE] bit is 0), the down-counter is clocked internally by a signal that is half the frequency of the PCLK input to the chip. In the Counter mode (ECE is l), the down-counter is decremented on the rising edge of the counter/timer's counter input .

Each time the counter reaches terminal count, its Interrupt Pending (IP) bit is set to 1, and if interrupts are enabled (IE = 1), an interrupt is generated. If a terminal count occurs while IP is already set, an internal error flag is set. As soon as IP is cleared, it is forced to 1 along with the Interrupt Error (ERR) flag. Errors that occur after the internal flag is set are ignored.

The state of the down-counter can be determined in two ways: by reading the contents of the down-counter via the Current Count register or by testing the Count In Progress (CIP) status bit in the Command and Status register. The CIP status bit is set when the down-counter is loaded; it is reset when the down-counter reaches 0. The Current Count register is a 16-bit register, accessible as two 8-bit registers, which mirrors the contents of the down-counter. This register can be read anytime. However, reading the register is asynchronous to the counter's counting, and the value returned is valid only if the counter is stopped. The down-counter can be reliably read "on the fly" by the first writing of a 1 to the Read Counter Control (RCC) bit in the counter/timer's Command and Status register. This freezes the value in the Current Count register until a read of the least significant byte is performed.

**Interrupt Logic Operation.** The CIO has five potential sources of interrupts: the three counter/timers and Ports A and B. The priorities of these sources are fixed in the following order: Counter/Timer 3, Port A, Counter/Timer 2, Port B, and Counter/Timer 1. Since the counter/timers all have equal capabilities and Ports A and B have equal capabilities, there is no adverse impact from the relative priorities.

The CIO interrupt priority, relative to other components within the system, is determined by an interrupt daisy chain. Two pins, Inter-rupt Enable In (IEI) and Interrupt Enable Out (IEO), provide the input and output necessary to implement the daisy chain. When IEI is pulled Low by a higher priority device, the CIO cannot request an interrupt of the CPU. The following discussion assumes that the IEI line is High.

Each source of interrupt in the CIO contains three bits for the control and status of the interrupt logic: an Interrupt Pending (IP) status bit, an Interrupt Under Service (IUS)

**Functional Description** (Continued)

status bit, and an Interrupt Enable (IE) control bit. IP is set when an event requiring CPU intervention occurs. The setting of IP results in forcing the Interrupt (INT) output Low, if the associated IE is 1.

The IUS status bit is set as a result of the Interrupt Acknowledge cycle by the CPU and is set only if its IP is of highest priority at the time the Interrupt Acknowledge commences. It can also be set directly by the CPU. Its primary function is to control the interrupt daisy chain. When set, it disables lower priority sources in the daisy chain, so that lower priority interrupt sources do not request servicing while higher priority devices are being serviced.

The IE bit provides the CPU with a means of masking off individual sources of interrupts. When IE is set to 1, interrupt is generated normally. When IE is set to 0, the IP bit is set when an event occurs that would normally require service; however, the INT output is not forced Low.

The Master Interrupt Enable (MIE) bit allows all sources of interrupts within the CIO to be disabled without having to individually set each IE to 0. If MIE is set to 0, all IPs are masked off and no interrupt can be requested or acknowledged. The Disable Lower Chain (DLC) bit is included to allow the CPU to modify the system daisy chain. When the DLC bit is set to 1, the CIO's IEO is forced Low, independent of the state of the CIO or its IEI input, and all lower priority devices' interrupts are disabled.

As part of the Interrupt Acknowledge cycle, the CIO is capable of responding with an S-bit interrupt vector that specifies the source of the interrupt. The CIO contains three vector registers: one for Port A, one for Port B, and one shared by the three counter/timers. The vector output is inhibited by setting the No Vector (NV) control bit to 1. The vector output can be modified to include status information to. pinpoint more precisely the cause of interrupt. Whether the vector includes status or not is controlled by a Vector Includes Status (VIS) control bit. Each base vector has its own VIS bit and is controlled independently. When MIE = 1, reading the base vector register always includes status, independent of the state of the VIS bit. In this way, all the information obtained by the vector, including status, can be obtained with one additional instruction when VIS is set to 0. When MIE = 0, reading the vector register returns the unmodified base vector so that it can be verified. Another register, the Current Vector register, allows use of the CIO in a polled environment. When read, the data returned is the same as the interrupt vector that would be output in an acknowledge, based on the highest priority IP set. If no unmasked IPs are set, the value FFH is returned. The Current Vector register is read-only.

**Programming**

The data registers within the CIO are directly accessed by address lines A0 and Al (Table 3). All other internal registers are accessed by the following two-step sequence, with the address lines specifying a control operation. First, write the address of the target register to an internal 6-bit Pointer Register; then read from or write to the target register. The Data registers can also be accessed by this method.

An internal state machine determines if accesses with A0 and Al equalling 1 are to the Pointer Register or to an internal control register (Figure 11). Following any control read operation, the state machine is in State 0 (the next control access is to the Pointer Register). This can be used to force the state machine into a known state. Control reads in State 0 return the contents of the last register pointed to. Therefore, a register can be read continuously without writing to the Pointer. While the CIO is in State 1 (next control access is to the register pointed to), many internal operations are suspended-no 1Ps are set and internal status is frozen. Therefore, to minimize interrupt latency and to allow continuous status updates, the CIO should not be left in State 1.

The CIO is reset by forcing RD and WR Low simultaneously (normally an illegal condition) or by writing a 1 to the Reset bit. Reset disables all functions except a read from or write to the Reset bit; writes to all other bits are ignored, and all reads return 01h. In this state, all control bits are forced to 0 and may be programmed only after clearing the Reset bit (by writing a 0 to it).

| A1 | A0 | Register |
|---|---|---|
| 0 | 0 | Port C's Data Register |
| 0 | 1 | Port B's Data Register |
| 1 | 0 | Port A's Data Register |
| 1 | 1 | Control Registers |

**Table 3. Register Selection**

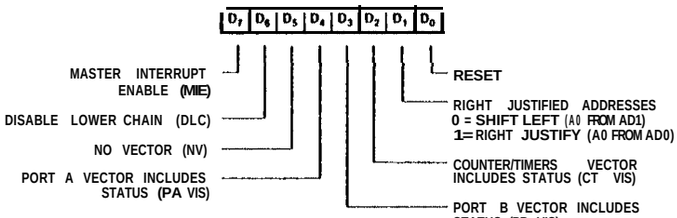
NOTE: State changes occur only when A0 = A1 = 1. No other accesses have effect.

**Figure 11. State Machine Operation**

## Registers

### Master Interrupt Control Register

Address: 000000
(Read/Write)




### Master Configuration Control Register

Address: 000001
(Read/Write)

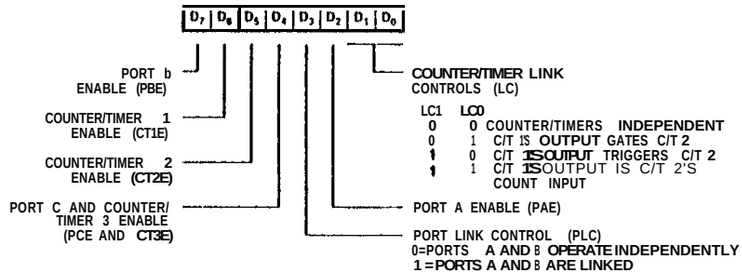
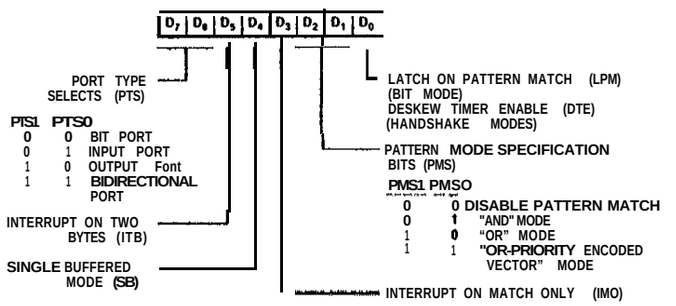



**Figure 12. Master Control Registers**

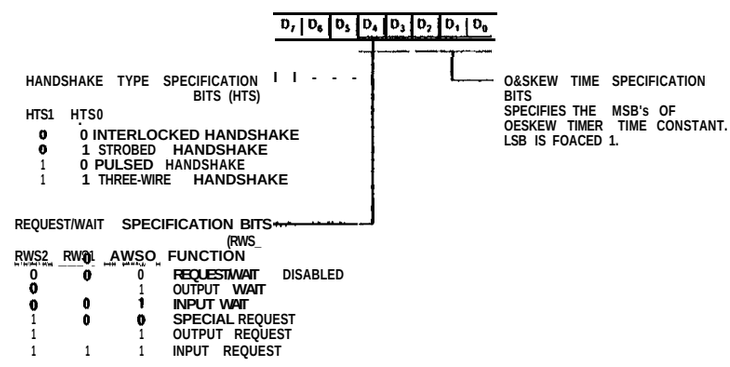
### Port Mode Specification Registers

Addresses: 100000 Port A
101000 Port B
(Read/Write)




### Port Handshake Specification Registers

Addresses: 100001 Port A
101001 Port B
(Read/Write)




### Port Command and Status Registers

Addresses: 001000 Port A
001001 Port B
(Read/Partial Write)

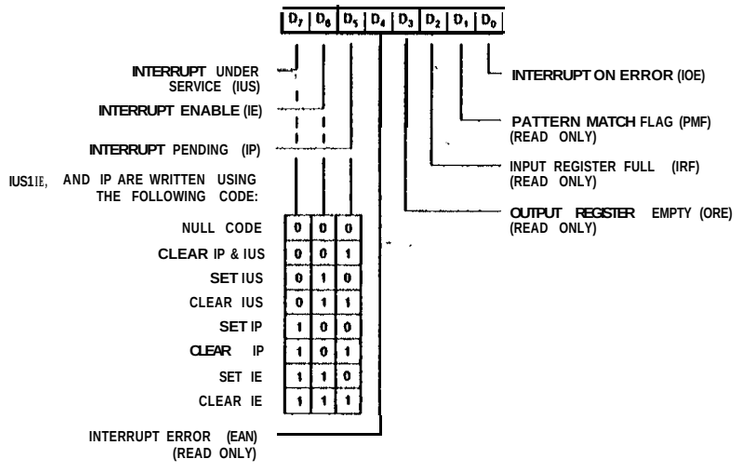
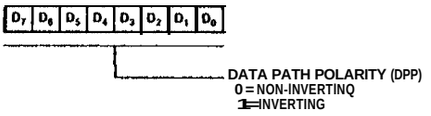


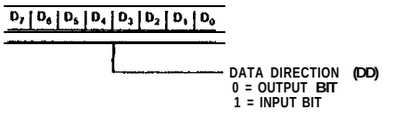

**Figure 13. Port Specifications Registers**

**Registers** (Continued)

Data Path Polarity Registers
Addresses: 100010 Port A
101010 Port B
000101 Port C (4 LSBs only)
(Read/Write)




Data Direction Registers
Addresses: 100011 Port A
101011 Port B
000110 Port C (4 LSBs only)
(Read/Write)




Special I/O Control Registers
Addresses: 100100 Port A
101100 Port B
000111 Port C (4 LSBs only)
(Read/Write)

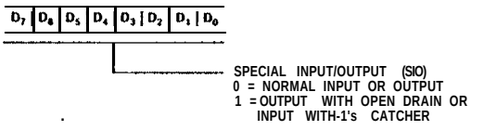
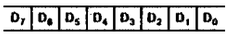



Figure 14. Bit Path Definition Registers

Port Data Registers
Addresses: 001101 Port A*
001110 Port B* ,
(Read/Write)




*These registers can be addressed directly.

Port C Data Register
Address: 001111'
(Read/Write)

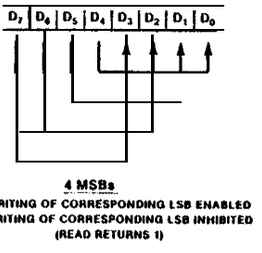



Figure 15. Port Data Registers

Pattern Polarity Registers (PP)
Addresses: 100101 Port A
101101 Port B
(Read/Write)

Pattern Transition Registers (PT)
Addresses: 100110 Port A
101110 Port B
(Read/Write)

Pattern Mask Registers (PM)
Addresses: 100111 Port A
101111 Port B
(Read/Write)

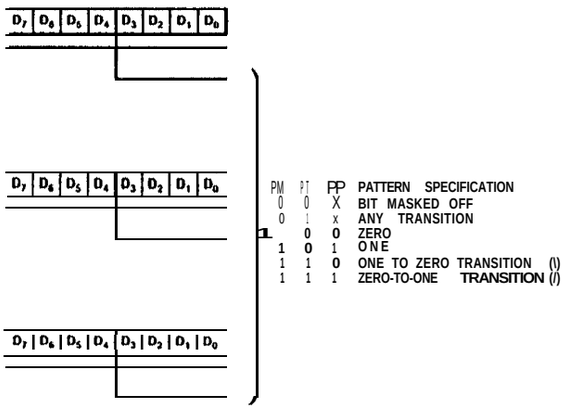
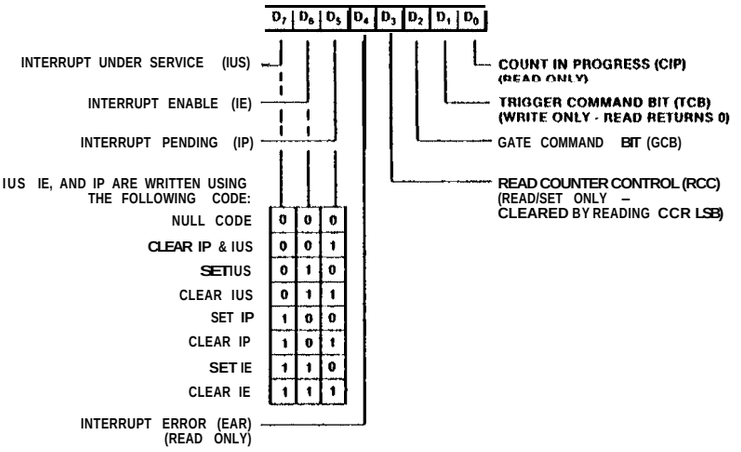



Figure 16. Pattern Definition Registers

**Registers**
(Continued)

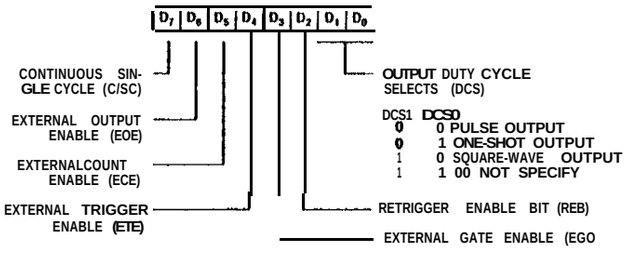
### Counter/Timer Command and Status Registers

Addresses: 001010 Counter/Timer 1
001011 Counter/Timer 2
00 1100 Counter/Timer 3
(Read/Partial Write)



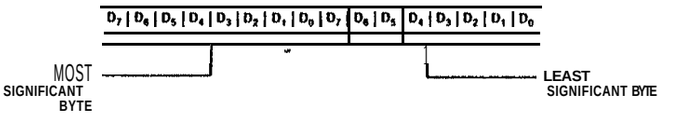

### Counter/Timer Mode Specification Registers

Addresses: 011100 Counter/Timer 1
011101 Counter/Timer 2
011110 Counter/Timer 3
(Read/Write)




### Counter/Timer Current Count Registers

Addresses: 010000 Counter/Timer l's MSB
010001 Counter/Timer l's LSB
010010 Counter/Timer 2's MSB
010011 Counter/Timer 2's LSB
010100 Counter/Timer 3's MSB
010101 Counter/Timer 3's LSB
(Read Only)




### Counter/Timer Time Constant Registers

Addresses: 0 10110 Counter/Timer l's MSB
010111 Counter/Timer l's LSB
011000 Counter/Timer 2's MSB
011001 Counter/Timer 2's LSB
011010 Counter/Timer 3's MSB
011011 Counter/Timer 3's LSB
(Read/Write)

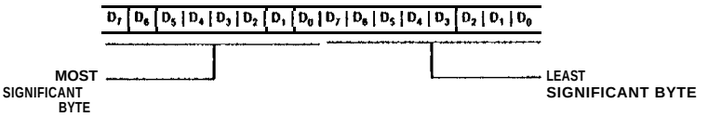



**Figure 17. Counter/Timer Register**

## Registers (Continued)

**Interrupt Vector Register**
Addresses: 000010 Port A
000011 Port B
000100 Counter/Timers
(Read/Write)


PORT VECTOR STATUS

PRIORITY ENCODED VECTOR MODE:

NUMBER OF HIGHEST PRIORITY BIT WITH A MATCH

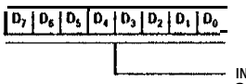
ALL OTHER MODES:

| D3 ORE | D2 IRF | D1 PMF | NORMAL |
|---|---|---|---|
| 0 | 0 | 0 | ERROR |

COUNTER/TIMER STATUS

| D2 | D1 | |
|---|---|---|
| 0 | 0 | C/T3 |
| 0 | 1 | C/T2 |
| 1 | 0 | C/T1 |
| 1 | 1 | ERROR |

**Current Vector Register**
Address: 011111
(Read only)




**Figaro 18. Interrupt Vector Registers**

## Register Address Summary

**Main Control Registers**

| Address | Register Name |
|---|---|
| 000000 | Master Interrupt Control |
| 00000 1 | Master Configuration Control |
| 000010 | Port A's Interrupt Vector |
| 0000 11 | Port B's Interrupt Vector |
| 000100 | Counter/Timer's Interrupt Vector |
| 000101 | Port C's Data Path Polarity |
| 000110 | Port C's Data Direction |
| 000111 | Port C's Special l/O Control |

**Most Often Accessed Registers**

| Address | Register Name |
|---|---|
| 001000 | Port A's Command and Status |
| 001001 | Port B's Command and Status |
| 001010 | Counter/Timer l's Command and Status |
| 001011 | Counter/Timer 2's Command and Status |
| 001100 | Counter/Timer 3's Command and Status |
| 001101 | Port A's Data (can be accessed directly) |
| 001110 | Port B's Data (can be accessed directly) |
| 001111 | Port C's Data (can be accessed directly) |

**Counter/Timer Related Registers**

| Address | Register Name |
|---|---|
| 010000 | Counter/Timer l's Current Count-MSBs |
| 010001 | Counter/Timer l's Current Count-LSBs |
| 010010 | Counter/Timer 2's Current Count-MSBs |
| 010011 | Counter/Timer 2's Current Count-LSBs |
| 010100 | Counter/Timer 3's Current Count-MSBs |
| 010101 | Counter/Timer 3's Current Count-LSBs |
| 010110 | Counter/Timer l's Time Constant-MSBs |
| 010111 | Counter/Timer l's Time Constant-LSBs |
| 011000 | Counter/Timer 2's Time Constant.MSBs |
| 011001 | Counter/Timer 2's Time Constant-LSBs |
| 011010 | Counter/Timer 3's Time Constant-MSBs |
| 011011 | Counter/Timer 3's Time Constant-LSBs |
| 011100 | Counter/Timer l's Mode Specification |
| 011101 | Counter/Timer 2's Mode Specification |
| 011110 | Counter/Timer 3's Mode Specification |
| 011111 | Current Vector |

**Port A Specification Registers**

| Address | Register Name |
|---|---|
| 100000 | Port A's Mode Specification |
| 100001 | Port A's Handshake Specification |
| 100010 | Port A's D&a Path Polarity |
| 100011 | Port A's Data Direction |
| 100100 | Port A's Special I/O Control |
| 100101 | Port A's Pattern Polarity |
| 100110 | Port A's Pattern Transition |
| 100111 | Port A's Pattern Mask |

**Port B Specification Registers**

| Address | Register Name |
|---|---|
| 101000 | Port B's Mode Specification |
| 101001 | Port B's Handshake Specification |
| 101010 | Port B's Data Path Polarity |
| 101011 | Port B's Data Direction |
| 101 loo | Port B's Special I/O Control |
| 101101 | Port B's Pattern Polarity |
| 101110 | Port B's Pattern Transition |
| 101111 | Port B's Pattern Mask |