Z8036/Z8536

Z-CIO and CIO Counter/Timer and Parallel I/O Unit

Product Specification

PS011201-0601

ZiLOG Worldwide Headquarters • 910 E. Hamilton Avenue • Campbell, CA 95008
Telephone: 408.558.8500 • Fax: 408.558.8300 • www.ZiLOG.com

This publication is subject to replacement by a later edition. To determine whether a later edition exists, or to request copies of publications, contact:

ZiLOG Worldwide Headquarters
910 E. Hamilton Avenue
Campbell, CA 95008
Telephone: 408.558.8500
Fax: 408.558.8300
www.ZiLOG.com

Document Disclaimer

Table of Contents

Table of Contents (Continued)

Table of Contents (Continued)

List of Illustrations

List of Tables

Chapter 1
General Description

1.1 INTRODUCTION

The Z8036 Z-CIO and Z8536 CIO Counter/Timer and Parallel I/O devices are general-purpose peripheral circuits that satisfy most counter/timer and parallel I/O needs encountered in system design, and are therefore helpful in real-time situations and for interrupt control. The Z8036 Z-CIO is designed for systems using the Z-BUS or any other multiplexed Address/Data bus. The Z8536 CIO is designed for CPUs using a nonmultiplexed bus, like that of the Z80 CPU. The differences between the two devices are found in the CPU interface, pin-outs, and timing.

NOTE

All material in this manual referring to "the CIO" applies to both the Z8036 and the Z8536, unless specifically designated by reference to either Z8036 or Z8536. All references to the Z-CIO refer only to the Z8036.

1.2 FEATURES

The Z-CIO and CIO devices satisfy a wide range of applications because of their extensive list of features:

- Two independent 8-bit, double-buffered, bidirectional I/O ports, plus a 4-bit special-purpose I/O port. The I/O ports feature programmable polarity, programmable direction (Bit mode), 1's catchers, and programmable open-drain outputs.
- Four handshake modes, including 3-Wire (like the IEEE-488).
- REQUEST/$\overline{\text{WAIT}}$ signal for high-speed data transfer.
- Flexible pattern-recognition logic, programmable as a 16-vector interrupt controller.
- Three independent 16-bit counter/timers, each with three output duty cycles (pulsed, one-shot, and square-wave) and up to four external access lines (count input, output, gate, and trigger). The counter/timers are programmable as retriggerable or non-retriggerable.
- All registers are read/write. In the Z8036, the registers are directly addressable; in the Z8536, the registers are accessed in two steps.

1.3 OVERVIEW

The CIO (Figure 1-1) consists of a CPU interface, three I/O ports (two general-purpose 8-bit ports and one special-purpose 4-bit port), three 16-bit counter/timers, an interrupt control logic block, and an internal control logic block. A large number of programmable options allow users to tailor the configuration to suit specific applications.

1.3.1 I/O Ports

There are three I/O ports: two general-purpose 8-bit ports (which are linkable into one 16-bit port), and one special-purpose 4-bit port.

1.3.1.1 Ports A and B

The two general-purpose 8-bit I/O ports, Ports A and B (Figure 1-2), are identical, except that Port B can be programmed to provide external access to Counter/Timers 1 and 2. Either port can be programmed to be either a handshake-driven, single- or double-buffered port (input, output, or bidirectional), or a control port with the direction of each bit individually programmable.

Both ports include pattern-recognition logic, which allows interrupt generation when a specific pattern is detected. The pattern-recognition logic can be programmed to make the port function like a priority interrupt controller. Ports A and B can also be linked to form a 16-bit I/O port with handshake.

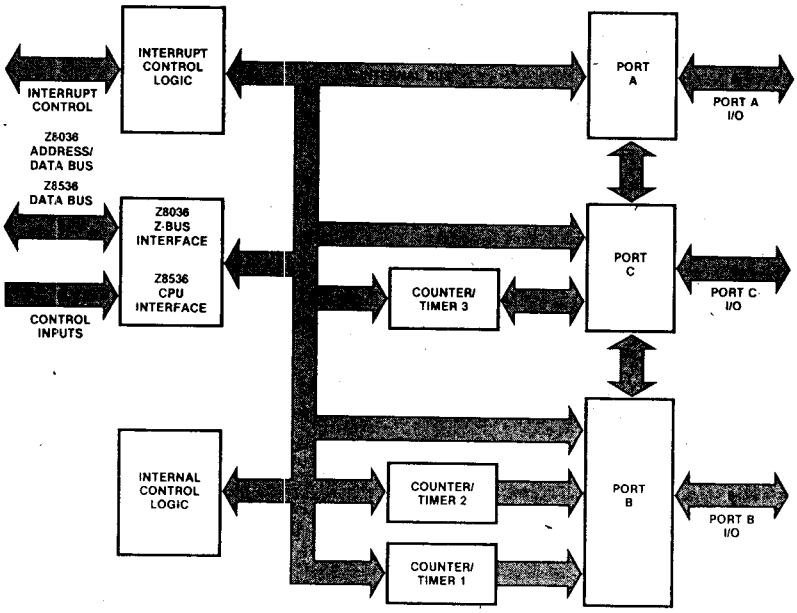


Figure 1-1. Z8036/Z8536 Z-CIO/CIO Block Diagram

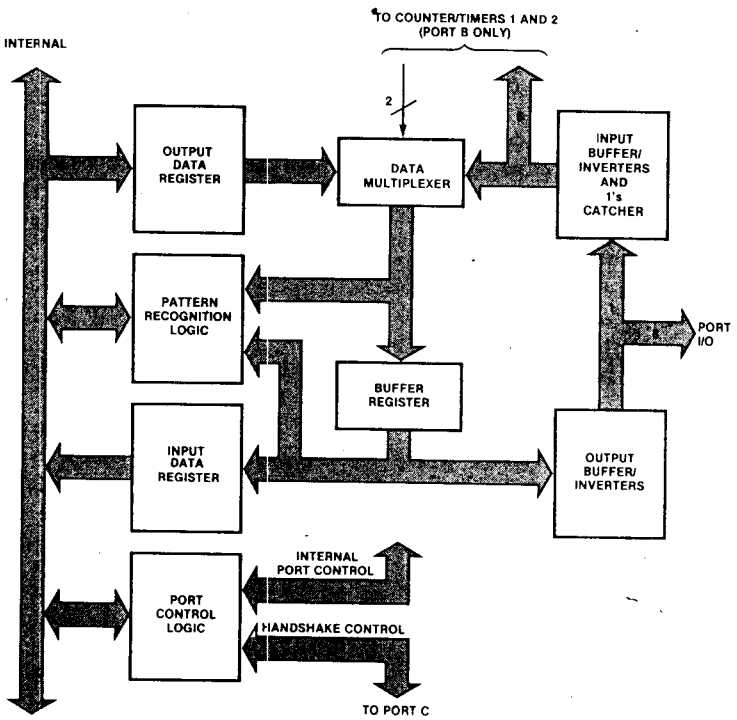


Figure 1-2. Ports A and B Block Diagram

Each port has 12 control and status registers, which control these capabilities. The data path of each port is composed of three internal registers: the Input Data register, the Output Data register, and the Buffer register. The Input Data register is accessed by writing the Port Data register; similarly, the Output Data register is accessed by reading the Port Data register. Two registers, the Mode Specification register and the Handshake Specification register, are used to define the mode of the port and to specify which type of handshake, if any, is to be used. The reference pattern for the pattern-recognition logic is specified by the contents of three registers: the Pattern Polarity register, Pattern Transition register, and Pattern Mask register. The detailed characteristics of each bit path (for example, the direction of data flow or whether a path is inverting or non-inverting) are programmed using the Data Path Polarity register, Data Direction register, and Special I/O Control register.

For each port, the primary control and status bits are grouped in a single register, the Command and Status register. After the port is configured, this is the only register that needs to 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.

1.3.1.2 Port C

The function of the special-purpose 4-bit port, Port C (Figure 1-3), depends upon the roles of Ports A and B. Port C provides the handshake lines when required by the other two ports. A REQUEST/$\overline{\text{WAIT}}$ line can also be provided by Port C so that transfers by Ports A and B can be synchronized with DMAs or CPUs. Any bits of Port C not used as handshake lines can be used as I/O lines or as external access to Counter/Timer 3.

Since Port C's function is defined primarily by Ports A and B (besides the internal Input Data and Output Data registers, which are accessed as in Ports A and B), only the three bit path registers are needed: the Data Path Polarity register, the Data Direction register, and the Special I/O Control register.

1.3.2 Counter/Timers

The three counter/timers (Figure 1-4) are all identical. Each is composed 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 C/T Command and Status registers).

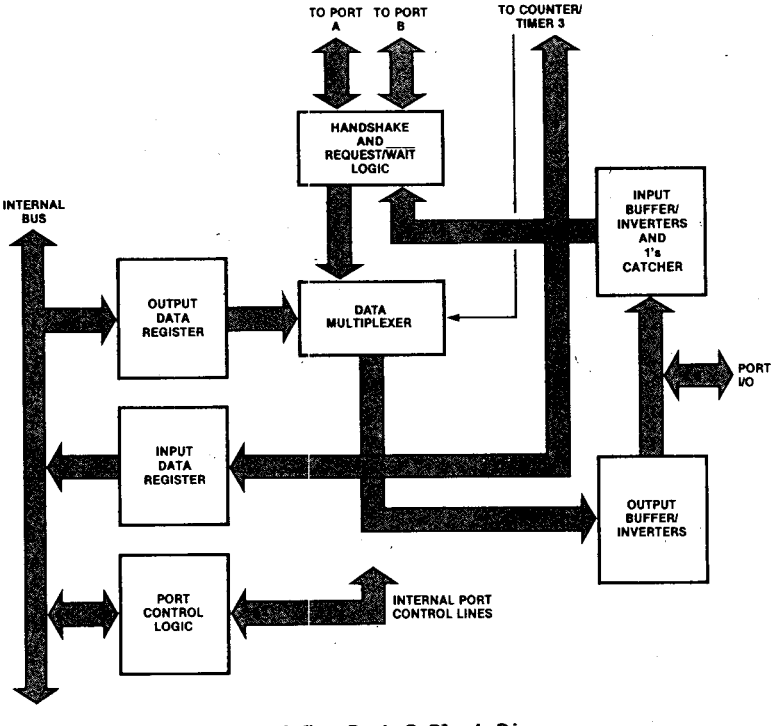


Figure 1-3. Port C Block Diagram

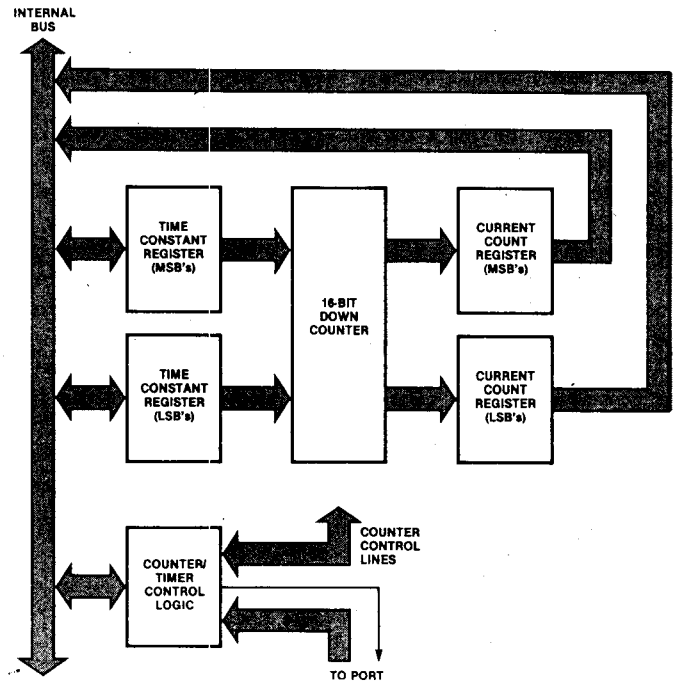


Figure 1-4. Counter/Timer Block Diagram

Up to four port pins (counter input, gate input, trigger input, and counter/timer output) can be used as dedicated external access lines for each counter/timer. Three different counter/timer output duty cycles are available: pulse, one-shot, and square-wave. The operation of the counter/timers can be programmed as either retriggerable or non-retriggerable.

1.3.3 Interrupt Control Logic

The Z8036 and Z8536 interrupt control logic provides the basis for standard Z-BUS and non-Z-BUS interrupt handling capabilities. (See Z-BUS Component Interconnect Summary, Zilog Data Book.) There are five registers (the Master Interrupt Control register, the Current Vector register, and the three Interrupt Vector registers) associated with the interrupt logic. In addition, each Port and Counter/Timer Command and Status register includes three bits associated with the interrupt logic: Interrupt Pending (IP), Interrupt Under Service (IUS), and Interrupt Enable (IE).

Chapter 2
Register Description

2.1 INTRODUCTION

This chapter provides brief descriptions of the command, status and data registers contained in the CIO. Each description includes the register address, the operation of the individual bits, and the state of the register after a reset (hardware or software).

For simplicity, the descriptions assume that the data path polarity of each bit is programmed to be non-inverting. Table 2-1 is a summary of the 48 CIO registers arranged in functional and numerical order. The binary internal addresses are the 6 bits written to an internal Pointer register as the addresses A_0-A_5. The details of the addressing schemes are described in Section 2.2 for the Z8036 and Section 2.3 for the Z8536.

For more complete discussions of the features and modes of operation of the CIO specified by these bits, refer to the appropriate chapters:

Chapter 3 Port Operation
Chapter 4 Counter/Timer Operation
Chapter 5 Interrupt Operation
Chapter 6 Initialization

Table 2-1. Z8036/Z8536 Z-CIO/CIO Register Summary

Internal Address (Binary)	Read/Write	Register Name
$A_5..A_0$		**Main Control Registers**
000000	R/W	Master Interrupt Control
000001	R/W	Master Configuration Control
000010	R/W	Port A Interrupt Vector
000011	R/W	Port B Interrupt Vector
000100	R/W	Counter/Timer Interrupt Vector
000101	R/W	Port C Data Path Polarity
000110	R/W	Port C Data Direction
000111	R/W	Port C Special I/O Control
		Most Often Accessed Registers
001000	*	Port A Command and Status
001001	*	Port B Command and Status
001010	*	Counter/Timer 1 Command and Status
001011	*	Counter/Timer 2 Command and Status
001100	*	Counter/Timer 3 Command and Status
001101	R/W	Port A Data**
001110	R/W	Port B Data**
001111	R/W	Port C Data**

Table 2-1. Z8036/Z8536 Z-CIO/CIO Register Summary--Continued

Internal Address (Binary)	Read/Write	Register Name
		Counter/Timer Related Registers
010000	R	Counter/Timer 1 Current Count MS Byte
010001	R	Counter/Timer 1 Current Count LS Byte
010010	R	Counter/Timer 2 Current Count MS Byte
010011	R	Counter/Timer 2 Current Count LS Byte
010100	R	Counter/Timer 3 Current Count MS Byte
010101	R	Counter/Timer 3 Current Count LS Byte
010110	R/W	Counter/Timer 1 Time Constant MS Byte
010111	R/W	Counter/Timer 1 Time Constant LS Byte
011000	R/W	Counter/Timer 2 Time Constant MS Byte
011001	R/W	Counter/Timer 2 Time Constant LS Byte
011010	R/W	Counter/Timer 3 Time Constant MS Byte
011011	R/W	Counter/Timer 3 Time Constant LS Byte
011100	R/W	Counter/Timer 1 Mode Specification
011101	R/W	Counter/Timer 2 Mode Specification
011110	R/W	Counter/Timer 3 Mode Specification
011111	R	Current Vector
		Port A Specification Registers
100000	R/W	Port A Mode Specification
100001	R/W	Port A Handshake Specification
100010	R/W	Port A Data Path Polarity
100011	R/W	Port A Data Direction
100100	R/W	Port A Special I/O Control
100101	R/W	Port A Pattern Polarity
100110	R/W	Port A Pattern Transition
100111	R/W	Port A Pattern Mask
		Port B Specification Registers
101000	R/W	Port B Mode Specification
101001	R/W	Port B Handshake Specification
101010	R/W	Port B Data Path Polarity
101011	R/W	Port B Data Direction
101100	R/W	Port B Special I/O Control
101101	R/W	Port B Pattern Polarity
101110	R/W	Port B Pattern Transition
101111	R/W	Port B Pattern Mask

* All bits can be read and some bits can be written.

** Also directly addressable in Z8536 using pins A_0 and A_1. (See Table 2-2 and Figures 8-1 and 8-2.)

2.2 REGISTER ADDRESSING FOR THE Z8036 (Z-CIO)

Register addressing in the Z8036 is accomplished through the use of an internal Pointer register. The Z8036 takes the contents of the multiplexed Address/Data bus and gates a subset of them into the internal Pointer register when $\overline{AS}$ is Low. The internal Pointer register identifies which register will be accessed during the subsequent part of this cycle.

The Z8036 provides two schemes for selecting the desired six of the eight address bits. The scheme to be used is determined by the Right Justify Address (RJA) bit in the Master Interrupt Control register. When RJA = 0, Address bus bits 0 and 7 are ignored, and bits 1 through 6 are decoded for the register address (A_0 derives from AD_1, A_5 derives from AD_6). When RJA = 1, address bits 0 through 5 are decoded for the register address (A_0 derives from AD_0, A_5 derives from AD_5). In the following register descriptions, only six bits are shown for addressing--they represent Address/Data bus bits 5 through 0 or 6 through 1, depending on the state of the RJA bit.

2.3 REGISTER ADDRESSING FOR THE Z8536 (CIO)

The registers in the Z8536 are accessed in a two-step sequence with pins A_0 and A_1 = 1. In the first step, a 6-bit address (the least-significant 6 bits of the Data bus) is written to an internal Pointer register. In the second step, the register identified by the Pointer register is read from or written to.

The data registers for the Z8536 Ports A, B, and C can be accessed by this sequence. The data registers can also be directly addressed by use of device pins A_0 and A_1, as shown in Table 2-2.

Table 2-2. Port Data Register Addressing for the CIO

Data Register	Address Line A_1	A_0
Port C	0	0
Port B	0	1
Port A	1	0
Control	1	1

The Z8536 contains a state machine which determines if accesses with A_0 and A_1 = 1 (see Table 2-2) are to the Pointer register or to an internal control register. (Refer to Figure 2-1 for the following discussion.) Reads in State 0 leave the state machine in State 0. Writes to the Z8536 in State 0 update the Pointer register and put the state machine into State 1. Accesses in State 1 are to the register addressed by the Pointer register, and cause the state machine to revert to State 0. State changes occur only when pin A_0 = pin A_1 = 1. Direct accesses of the data registers have no effect on state machine operation.

After any control read operation (pin A_0 = pin A_1 = 1), 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 Z8536 is in State 1 (next control access is to the register pointed to), many internal operations are suspended, Interrupt Pending (IP) cannot be set, and internal status is frozen. Therefore, to minimize interrupt latency and to allow continuous status updates, **the Z8536 should not be left in State 1.**

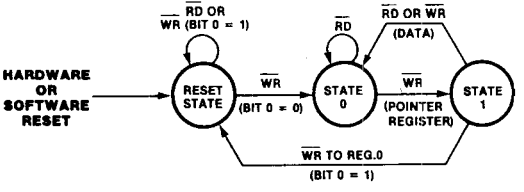


Figure 2-1. Z8536 State Machine Operation

2.4 MASTER CONTROL REGISTERS

The Master Control registers consist of the Master Interrupt Control register and Master Configuration Control register. These registers provide primary controls for the interrupt logic, port and counter/timer enable bits, port and counter/timer link bits and the RESET bit.

2.4.1 Master Interrupt Control Register

The Master Interrupt Control register contains the primary control bits for the interrupt control logic. When the device is reset all bits in all device registers are forced to 0 except RESET, which is set to 1. The RJA bit (D_1) is only

applicable to the Z8036 Z-CIO. All bits in the Master Interrupt Control register are Read/Write.

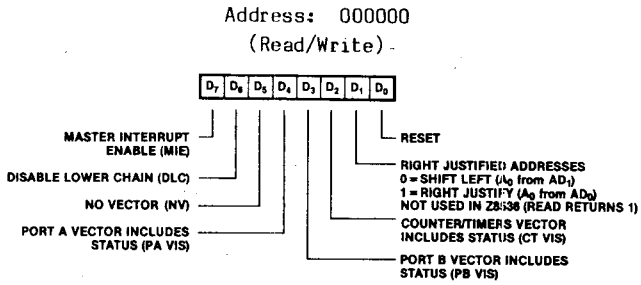


Figure 2-2. Master Interrupt Control Register.

Master Interrupt Enable--MIE (D_7). Clearing this bit to 0 inhibits the device from requesting an interrupt or responding to an Interrupt Acknowledge. Its effect is the same as pulling the Interrupt Enable In (IEI) input Low, except that the daisy-chain is left intact. A 1 in this bit allows the interrupt logic to operate normally.

The MIE bit also affects whether or not status is included when reading interrupt vectors. If MIE = 0, interrupt vector reads do not include status. If MIE = 1, vector reads always include status, independent of the state of the corresponding Vector Includes Status (VIS) bit.

Disable Lower Chain--DLC (D_6). If DLC is set to 1, the Interrupt Enable Out (IEO) output of the device is forced Low, disabling interrupts from all lower-priority devices on the daisy-chain. When DLC is 0, IEO operates normally.

No Vector--NV (D_5). When NV is set to 1, the device is inhibited from outputting an interrupt vector during an Interrupt Acknowledge cycle. This allows the vector to be provided by external hardware. It has no effect on the setting of the Interrupt Under Service (IUS) bit. If NV is written with 0, the interrupt vector is output as usual.

Port A Vector Includes Status--PA VIS (D_4). If this bit is 0 when a Port A interrupt is acknowledged, the interrupt vector that is output is the unmodified content of the Port A Interrupt Vector register. If this bit is written with a 1, the Port A base vector is modified to include status, which indicates the cause of the interrupt. Vector modification is described in Section 5.3.4. The state of this bit has no effect on the value returned when the Port A Interrupt Vector register is read. When reading the vector, the MIE bit determines if status is included in the vector, (that is, no status is included if MIE = 0).

Port B Vector Includes Status--PB VIS (D_3). This bit controls whether or not the Port B interrupt vector includes status. It operates the same way that the PA VIS bit controls the Port A interrupt vector.

Counter/Timer Vector Includes Status--CT VIS (D_2). This bit controls whether or not the base interrupt vector shared by the three counter/timers includes status. It operates the same way that the other two VIS bits (PA VIS and PB VIS) operate.

Right Justified Address--RJA (D1). (Z8036 only). When this bit is 0, the register address is shifted left one bit (see Section 2.2). Address bit A_0 is derived from Address/Data bus bit AD_1. When set to 1, the address is right justified, (for example, $A_0 = AD_0$).

The Z8536 does not use RJA--this bit is always set to 1, which causes the address to always be right-justified.

RESET--(D_0). Setting the RESET bit to 1 by a software write resets the device. The bit can also be set by a hardware reset on the Z8036 by forcing Address Strobe ($\overline{AS}$) and Data Strobe ($\overline{DS}$) Low simultaneously; or on the Z8536 by forcing Read ($\overline{RD}$) and Write ($\overline{WR}$) Low simultaneously. While RESET is 1, reads of all other registers will be 0 and writes to other registers are ignored. This bit is cleared only by writing a 0 to the RESET bit (see Sections 6.2 and 6.3).

2.4.2 Master Configuration Control Register

The Master Configuration Control register contains the control bits used to enable different sections of the device after they are initially configured, as well as the bits used to link the ports together and the timers together. All bits are cleared to 0 by resetting the device. The register is read/write.

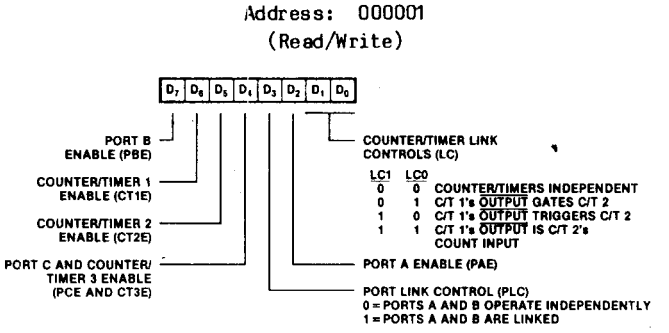


Figure 2-3. Master Configuration Control Register

Port B Enable--PBE (D_7). This bit, when set to 1, allows Port B to operate normally. When cleared to 0, it inhibits the Port B logic from issuing an interrupt request (its IP cannot be set); however, if IP was already set, clearing PBE does not clear IP. While cleared to 0, PBE inhibits READY/$\overline{WAIT}$ assertion, holds all 1's catchers in a transparent condition, and forces the Port B I/O lines into a high-impedance state. The purpose of this bit is to allow Port B to be configured initially without setting its IP erroneously or having its I/O lines go low-impedance until it is safe to do so.

Counter/Timer 1 Enable--CT1E (D_6). When cleared to 0, Counter/Timer 1 is put into an initialized state: its IP cannot be set (however, if IP was already set, clearing CT1E does not clear IP), the Count In Progress (CIP) flag is cleared, Read Counter Control (RCC) is forced to 0, and all trigger inputs are ignored. Setting CT1E to 1 allows the counter/timer to function normally.

Counter/Timer 2 Enable--CT2E (D_5). The CT2E bit performs the same function for Counter/Timer 2 that CT1E performs for Counter/Timer 1.

Port C and Counter/Timer 3 Enable--PCE and CT3E (D_4). This bit enables both Port C and Counter/Timer 3. The function is the same as D_7 (PBE) and D_6 (CT1E) for Port B and Counter/Timer 1, respectively. In addition, while this bit is cleared to 0, the handshake logic for Ports A and B is forced into an idle state and the internal Acknowledge Input ($\overline{ACKIN}$) signal is forced High. This allows the start-up of handshake operations to be precisely controlled.

Port Link Control--PLC (D_3). When PLC is set to 1, Ports A and B are linked to form a 16-bit port. In this mode, only the Port A Handshake Specification and Command and Status registers are used. Port B must be specified as a bit port and its pattern match capability must be disabled. Also, when linked, the Port B data register must be read or written before the Port A data register. A 0 in the PLC bit allows the ports to operate independently. If the ports are to be linked, this bit must be set before the ports are enabled.

Port A Enable--PAE (D_2). The Port A Enable bit performs the same function for Port A that the Port B Enable bit (D_7) performs for Port B.

Counter/Timer Link Controls--LC_1 & LC_0 (D_1 & D_0). These two bits specify if and how Counter/Timers 1 and 2 are linked. The Counter/Timers must be linked before they are enabled. The various configurations are shown in Table 2-3.

Table 2-3. Counter/Timer Link Controls

LC_1	LC_0	Configuration
0	0	Counter/Timers are independent
0	1	Counter/Timer 1's output (inverted) gates Counter/Timer 2
1	0	Counter/Timer 1's output (inverted) triggers Counter/Timer 2
1	1	Counter/Timer 1's output (inverted) is Counter/Timer 2's count input (Counter/Timer 2's External Count Enable* bit must be cleared to 0)

* (See Section 2.9.1 for description of External Count Enable bit.)

2.5 PORT SPECIFICATION REGISTERS

Each of these registers define the port operating mode, specify the type of handshake (if one is used), and contain the command and status bits used to affect data transfers of its port. There is a set of Port Specification registers for both Port A and Port B.

2.5.1 Port Mode Specification Registers

Each Port Mode Specification register contains the bits that define the operating mode of its port and specify the operation of pattern match logic of the port. A reset forces all bits to be cleared to 0. All bits are read/write.

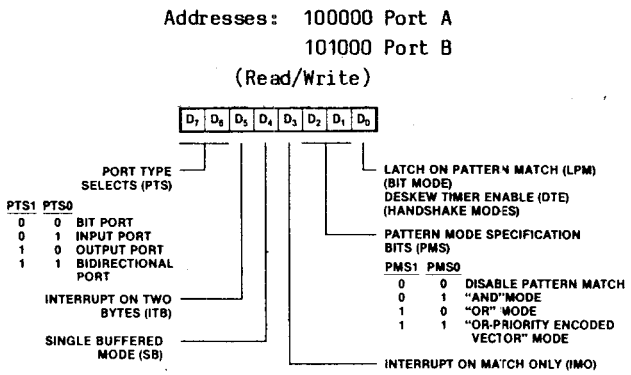


Figure 2-4. Port Mode Specification Registers

Port Type Selects--PTS_1 & PTS_0 (D_7 & D_6). The port type is specified by these two bits, as shown in Table 2-4.

Table 2-4. Port Type Selects

PTS_1	PTS_0	Port Type
0	0	Bit port (no handshake)
0	1	Input port with one of four handshakes*
1	0	Output port with one of four handshakes*
1	1	Bidirectional port with one of two handshakes**

* The four handshakes are: Interlocked, Pulsed, Strobed, or 3-wire.

** The two handshakes are: Interlocked or Strobed.

Interrupt on Two Bytes--ITB (D_5). For a port programmed with handshake, this bit indicates when an interrupt should be requested. If ITB is set to 1, IP is set when two bytes of data can be read or written. For an input port, IP is set when both the Input Data register and Buffer register are full. For an output port, IP is set when both the Output Data and Buffer register are empty. When ITB is cleared to 0, IP is set whenever a single byte of data is available to be moved (the Input Data register is full or the Output Data register is empty). This bit must always be cleared to 0 for ports specified either as bit ports, single-buffered ports (SB = 1), or bidirectional ports.

ITB also affects the operation of the Request line. When ITB = 0, the Request line will go active as soon as the device is ready for a data transfer. For input ports, the Request line will go High when the Input Data register is full. If ITB = 1, both the Buffer register and Input Data register must be full for Request to go active. For output ports with ITB = 0, the Request line will go High when the Output Data register is empty. If ITB = 1, the Request line will go High when both the Buffer register and Output Data registers are empty. In either case, the Request line will stay active as long as a byte is available to be read or written.

Single Buffer--SB (D_4). For a port programmed with handshake, this bit specifies if the port should be single- or double-buffered. When SB is cleared to 0, the port is double-buffered. When SB is set to 1 (ITB must be 0), the port is single-buffered: an input byte is loaded into both the Buffer and Input Data registers, or an output byte is loaded into both the Output Data and Buffer registers. This bit must always be cleared to 0 for bit ports.

Interrupt on Match Only--IMO (D_3). For ports with handshake (when this bit is set to 1) an interrupt will be generated only when the data moved into the Input Data register or out of the Output Data register matches the pattern specification. When cleared to 0, the port operates normally. The purpose of this bit is to allow the generation of CPU interrupts only on bytes which match the pattern specification. It is useful, for example, when the data is being moved under Direct Memory Access (DMA) control. IMO must be 0 if either SB or ITB are set to 1 or if the port is a bit port.

Pattern Mode Specification Bits--PMS_1 & PMS_0 (D_2 & D_1). These two bits define the mode of operation of the pattern match logic, as is shown in Table 2-5.

Table 2-5. Pattern Mode Specification Bits

PMS_1	PMS_0	Pattern Mode
0	0	Disable Pattern Match
0	1	AND Mode
1	0	OR Mode
1	1	OR-Priority Encoded Vector mode

The OR-Priority Encoded Vector mode must not be specified for ports configured as bit ports with the Latch on Pattern Match (LPM) bit set to 1 or for ports with handshake.

Latch On Pattern Match--LPM, or Deskew Timer Enable--DTE (D_0). The LPM/DTE bit is a dual-function bit. The LPM function is active when the port is specified in bit mode; the DTE function is active when the port is specified as an output port with handshake.

The LPM bit, when set to 1, causes the port to latch the input data present at the port when a pattern match is detected. If LPM is 0, pattern matches are still detected, but the data read back from the port follows the port pins.

The DTE bit, when set to 1, activates the deskew timer to perform delay functions as set in the Port Handshake Specification register. When cleared to 0, no delay is activated (see Section 3.4.3.2).

LPM/DTE must be cleared to 0 for input ports with handshake or for bit ports whose pattern match logic is in the OR-Priority Encoded Vector mode.

2.5.2 Port Handshake Specification Registers

Each of the Port Handshake Specification registers contain the bits that specify the type of handshake, the utilization of the REQUEST/$\overline{WAIT}$ line, and the Deskew Timer Time Constant for ports programmed with handshake. These bits are ignored if the port is a bit port. A RESET forces all bits to 0. All bits are read/write.

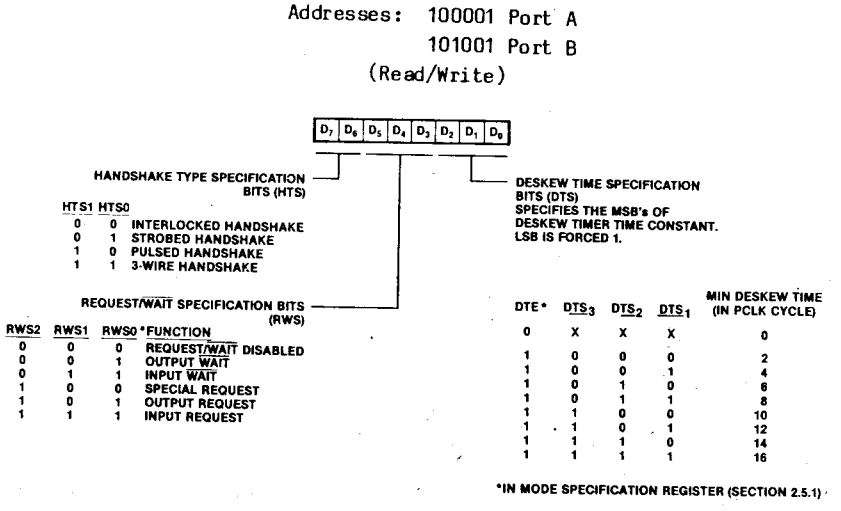


Figure 2-5. Port Handshake Specification Registers

Handshake Type Specification Bits--HTS_1 & HTS_0 (D_7 & D_6). These two bits specify the handshake type that a port with handshake will use, as is shown in Table 2-6.

Table 2-6. Handshake Type Specification Bits

HTS_1	HTS_0	Handshake Type
0	0	Interlocked Handshake
0	1	Strobed Handshake
1	0	Pulsed Handshake
1	1	3-Wire Handshake

The Pulsed Handshake and the 3-Wire Handshake must not be specified for bidirectional ports. Only one port at a time can use the Pulsed Handshake configuration. If one port uses the 3-Wire Handshake, the other port must be a bit port.

REQUEST/$\overline{WAIT}$ Specification Bits--RWS_2-RWS_0 (D_5-D_3). These three bits specify the utilization of the REQUEST/$\overline{WAIT}$ line, as is shown in Table 2-7.

Table 2-7. REQUEST/WAIT Specification Bits

RWS_2	RWS_1	RWS_0	Function
0	0	0	REQUEST/$\overline{WAIT}$ Disabled
0	0	1	Output $\overline{WAIT}$
0	1	1	Input $\overline{WAIT}$
1	0	0	Special REQUEST
1	0	1	Output REQUEST
1	1	1	Input REQUEST

If a port uses the REQUEST/$\overline{WAIT}$ capability, the other port must be programmed as a bit port, because three pins of Port C are required. (See Table 3-1.)

Deskew Time Specification Bits--DTS_3 Through DTS_1 (D_2-D_0). These three bits are the most significant bits of the Deskew Timer Time Constant. They specify the minimum amount of deskew time to be provided for output data. They define the minimum number of Peripheral Clock (PCLK) cycles of delay, 0 to 16, between the output of a new byte of data and the handshake logic indicating that new data is available ($\overline{DAV}$ falling). This logic is particularly useful in systems where large amounts of skew can exist between the data and the handshake signals or where the receiver of the data has a large set-up time requirement. The amount of deskew provided is shown in Figure 2-5.

NOTE

0 PCLK cycles deskew time is obtained by not enabling the deskew timer (DTE = 0 in the Port Mode Specification register).

2.5.3 Port Command and Status Registers

Each of these registers contain the primary command and status bits for its port. Other than the data bits themselves, these are the bits most often accessed in normal port operation. A reset forces ORE to 1 and all other bits to 0. All bits are readable and four are writeable.

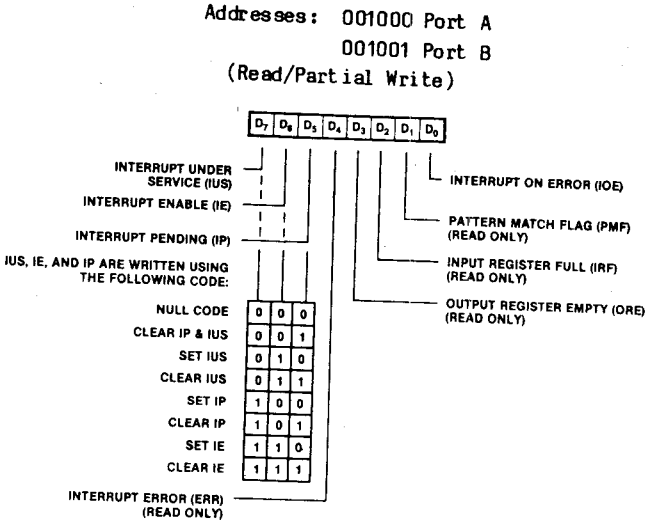


Figure 2-6. Port Command and Status Registers

Interrupt Under Service--IUS (D_7). This status bit is automatically set to 1 if its corresponding IP is the highest-priority interrupt request pending when an Interrupt Acknowledge sequence takes place. It can also be set directly by CPU command. While the IUS is set, the same and lower priority sources of interrupt are prohibited from requesting interrupts via the internal and external daisy-chains. The IUS can be cleared to 0 only by CPU command. This bit is read/write. It is changed by writing to the Command and Status register of the port using the code shown in Figure 2-6.

Interrupt Enable--IE (D_6). This bit enables or disables the port's interrupt logic. While IE is cleared to 0, the port is unable to request an interrupt or to respond to an Interrupt Acknowledge. The normal operation of IP or IUS is not affected--the IP is simply masked off from the rest of the device. A 1 in IUS still affects the interrupt daisy-chain. If IE is programmed to be 1, the interrupt logic operates normally. This bit is read/write. It is changed by writing to the Command and Status register of the port using the code shown in Figure 2-6.

Interrupt Pending--IP (D_5). IP is a status bit which, when set to 1, indicates that the port requires servicing due to a pattern match, a handshake, or an error. It is set to 1 by the port logic (or by the CPU command). If IE is also 1 and no higher-priority interrupts are under service, then the $\overline{INT}$ line is pulled Low to request an interrupt. It is cleared to 0 either automatically or by a CPU command, depending on port configuration. It is changed by writing to the Port Command and Status register using the code shown in Figure 2-6.

Interrupt Error--ERR (D_4). This status bit is automatically set to 1 along with IP when, for a bit port with pattern match enabled, a second match occurs before a previous match is acknowledged (IP is still set). If the port Interrupt On Error (IOE) bit is 0, errors are ignored and this bit is held at 0. This bit can be cleared only by clearing the corresponding IP. This bit is a read-only bit; writes to it are ignored.

Output Data Register Empty--ORE (D_3). ORE is a status bit used in conjunction with ports, specified either as output or bidirectional ports, to indicate whether or not the Output Data register is full. It is set to 1 when a byte of data is moved out of the Output Data register as part of an output handshake. The bit can only be cleared by writing to the data register. As a bit port, ORE is forced to 1 unless OR-PEV pattern match mode is specified--in which case, ORE is forced to 0. This bit is a read-only bit; writes to it are ignored. RESET empties the Output Data register, so after a RESET the ORE is set.

Input Data Register Full--IRF (D_2). IRF is a status bit used in conjunction with ports, specified either as input or bidirectional ports, to indicate whether or not the Input Data register is full. It is automatically set to 1 when a new byte of data is available to be read as the result of an input handshake. The bit can only be cleared by reading the port data register, thus "emptying" the Input Data register and forcing the bit to 0. If the port is an output port or a bit port, this bit is always forced to 0. IRF is a read-only bit; writes to it are ignored.

Pattern Match Flag--PMF (D_1). The PMF is a status bit set to 1 when a pattern match is detected. If the port is a bit port, PMF is not latched. It reflects the state of the pattern match logic just before it is read. For the Z8036, it is updated each $\overline{AS}$. For the Z8536, it is updated every second PCLK cycle while the CIO is in State 0 (See Section 2.3). For ports with handshake, the state of the PMF is updated each time a byte of data is moved into the Input Data register or out of the Output Data register. If the port pattern match logic is not enabled ($PMS_1 = PMS_0 = 0$), the PMF is forced to 0. This is a read-only bit. Writes to it are ignored.

Interrupt on Error--IOE (D_0). While IOE is cleared to 0, error conditions in bit ports using pattern-recognition logic (a second match before a previous match is acknowledged) are ignored. However, if IOE is 1, such errors will cause IP to be set and will halt normal operation of the port until the error condition is dealt with. This bit has no meaning for ports with handshake and must be cleared to 0.

2.6 BIT PATH DEFINITION REGISTERS

The Bit Path Definition registers are used to specify the details of each bit path of each port. They define:

- whether a bit path is inverting or non-inverting
- if an output is normal or open-drain
- if a bit port input has a 1's catcher inserted in its path
- which direction the data is flowing for each bit of a bit port

Each port has a set of these registers. The four most-significant bits of each register do not exist in the registers associated with Port C (writes are ignored, reads return 1s).

2.6.1 Data Path Polarity Registers

The Data Path Polarity registers each define whether the bits in its port are inverting or non-inverting on a bit-by-bit basis.

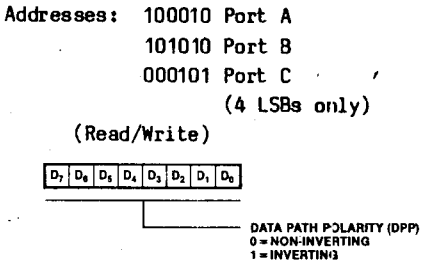


Figure 2-7. Data Path Polarity Registers

A 0 in a particular bit position of this register specifies the corresponding bit path of the port as non-inverting (that is, a High level at the port pin is 1). If a bit in this register is written with 1, the data path is programmed inverting (that is, a Low level at the pin is 1). A reset clears all bits to 0 (the port is non-inverting). The bits are read/write.

2.6.2 Data Direction Registers

Each of the Data Direction registers define the direction of data flow for the individual bits of its port if configured as a bit port. The state of this register is ignored for ports with handshake.

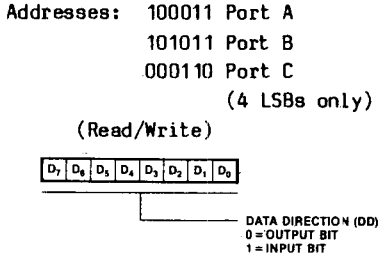


Figure 2-8. Data Direction Registers

A 0 in a bit position of this register specifies the corresponding bit of the port as an output bit, while a 1 specifies it as an input. The value programmed in this register for Ports A and B is overridden if the port is one with handshake.

An input bit specification is overridden for bits in Port C used as outputs for handshake signals or a REQUEST/$\overline{\text{WAIT}}$ line. Bits used as handshake inputs must be specified as inputs.

A reset forces all bits in these registers to 0. All bits are read/write.

2.6.3 Special I/O Control Registers

Each of the Special I/O Control registers is a dual-function register which specifies special characteristics about its port's data path. Its exact function depends on the direction of data flow defined for the path.

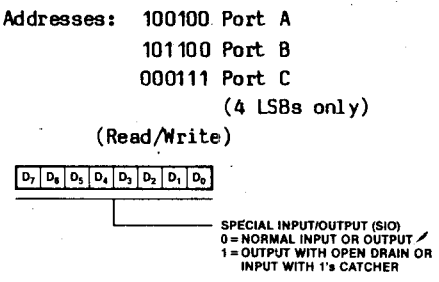


Figure 2-9. Special I/O Control Registers

If a bit is an input bit, a 1 in this register's corresponding bit position invokes a 1's catcher. A 1's catcher functions by automatically latching a 1 if its input goes to 1. It is cleared only by writing a 0 to the Input Data register. A 1's catcher is inserted into the input path after the bit's invert/non-invert logic. If the bit is programmed 0, it is a normal input bit. The 1's catcher is available only for input bit port bits.

If a bit is an output bit, a 0 in the corresponding bit position of this register specifies the output as a normal output with both a pull-up and a pull-down transistor. A 1 in this register defines the output as open-drain; no pull-up transistor is provided. The value programmed in this register applies to all output modes, independent of utilization.

A reset forces all bits to 0. All bits are read/write.

2.7 PATTERN DEFINITION REGISTERS

These registers collectively specify the match pattern for the port. As the registers must be taken together to define the pattern, they are described differently than the previous registers.

Addresses: 100101 Port A
101101 Port B
(Read/Write)

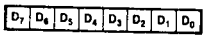


Figure 2-10. Pattern Polarity Registers

Addresses: 100110 Port A
101110 Port B
(Read/Write)

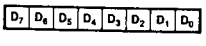


Figure 2-11. Pattern Transition Registers

Addresses: 100111 Port A
101111 Port B
(Read/Write)

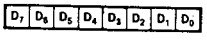


Figure 2-12. Pattern Mask Registers

A reset forces all of these registers to 0. All are read/write.

The pattern specification for each bit is defined as shown in Table 2-8.

The pattern specified by the Pattern Definition registers is a logical (not a physical) specification--this concept is important in understanding the interaction between the pattern match logic and the invert/non-invert logic. An example which shows the logical (as opposed to physical) nature of the specification is: a High level (V_{CC}) on an input pin programmed as inverting matches a 0 specification. Similarly, an output written with a 1 matches a 1 specification even if it is programmed inverting and the output pin is at a Low voltage level.

If the port is programmed as a port with handshake, or if the pattern match mode is OR-Priority Encoded Vector, the transition detection patterns should not be specified (PTn should be set to 0). If the AND mode is specified, no more than one bit should be specified to detect transitions.

2.8 PORT DATA REGISTERS

Ports A and B each have a data path that is composed of three registers: an Input Data register, an Output Data register, and a Buffer register (See Figure 1-2). Output data written to the data register is stored in the Output Data register. Reading the data register returns the contents of the Input Data register. The Buffer register is used to buffer the input and output data if the port is configured as a port with handshake. If so enabled, it is used by the bit port to latch data when a pattern match is detected.

Addresses: 001101 Port A
001110 Port B
(Read/Write)

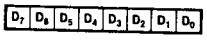


Figure 2-13. Port A and B Data Registers

Table 2-8. Pattern Specification Definition

Pattern Mask Register$_n$	Pattern Transition Register$_n$	Pattern Polarity Register$_n$	Pattern Specification
0	0	0	Bit Masked Off (X)
0	1	0	Any Transition (⇕)
1	0	0	Zero (0)
1	0	1	One (1)
1	1	0	One to Zero Transition (↓)
1	1	1	Zero to One Transition (↑)

The individual bits of the port data registers map directly onto the port I/O pins (bit 0 of the Port A Data register corresponds to the PA_0 pin, etc.).

The Port C Data register consists of two registers: an Input Data register and an Output Data register (see Figure 1-3). Output data written to the data register is stored in the Output Data register. Reading the data register returns the contents of the Input Data register. Because Port C is only four bits wide, the four least-significant bits of an 8-bit register are used for the Port C Data register. The four most-significant bits are used as a write protect mask for the four least-significant bits (bit D_7 is the write protect mask for bit D_3, etc.), as shown in Figure 2-14. Writing a 0 to the write protect mask bit enables writing to the corresponding bit in Port C. Writing a 1 inhibits writing the corresponding bit in Port C. Reading Port C always returns 1's in the upper four bits.

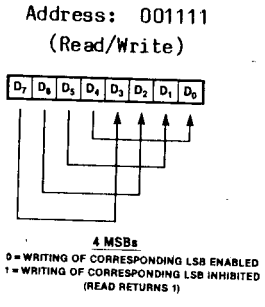


Figure 2-14. Port C Data Register

Details of the operation of these registers in the various configurations are given in Chapter 3. The data registers in the Z8536 can also be directly accessed by pin A_0 and pin A_1 (see Table 2-2).

NOTE

A reset does not effect the contents of the data registers.

2.9 COUNTER/TIMER CONTROL REGISTERS

Each counter/timer has a set of Counter/Timer Control registers, which perform several functions for the counter/timers:

- specify the mode of operation
- monitor the status
- provide control
- allow access to the down-counter so that it can be preset and read

2.9.1 Counter/Timer Mode Specification Registers

Each Counter/Timer Mode Specification register contains the bits that define its counter/timer's mode of operation and specify the external control and status lines to provide for it. A reset forces all bits to 0. All bits are read/write.

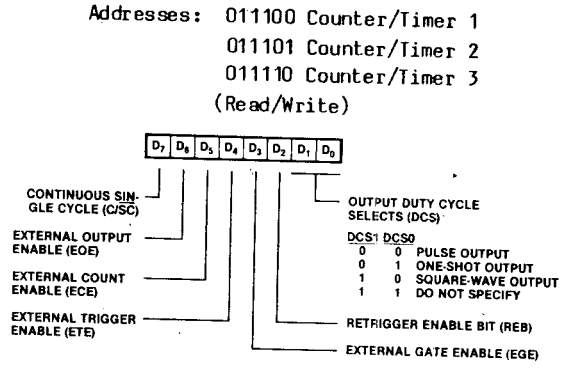


Figure 2-15. Counter/Timer Mode Specification Registers

Continuous/Single Cycle--C/$\overline{SC}$ (D_7). If C/$\overline{SC}$ is set to 1, then each time the down-counter reaches the count of 1, the time constant value is reloaded (on the next count) and the countdown sequence is repeated. If C/$\overline{SC}$ is 0 when the count of 1 is encountered (and, for square-wave outputs, if the output is a 1), the counter is allowed to count down to 0 and the countdown sequence is terminated.

External Output Enable--EOE (D_6). By programming this bit to be 1, the output of the counter/timer is provided on the I/O line of the port associated with that particular counter/timer (see Table 4-1). This bit should not be set to 1 unless the corresponding bit is available, (it is not being used as part of an input, output, or bidirectional port, or it is not being used as a handshake or REQUEST/WAIT line). The bit must be programmed to be an output bit in the Data Direction register of its port.

External Count Enable--ECE (D_5). When ECE is set to 1, the counter/timer is put into the counter mode. The I/O line of the port associated with the counter/timer (Table 4-1) is used as an external counter input. On each rising edge of the count input (when the data path is specified non-inverting), the down-counter is decremented. The bit must be available and it must be specified to be an input. (Even if the port bit is programmed as an output bit, the port pin [if enabled] is used as the counter/timer input, allowing the CPU to write this input directly.)

External Trigger Enable--ETE (D_4). When ETE is set to 1, the I/O line of the port associated with the counter/timer (see Table 4-1) is used as a trigger input to the counter/timer. A rising edge (when the data path is specified non-inverting) on this line will cause the down-counter to be loaded. To guarantee that the counter/timer will be triggered on a particular rising edge of the clocking signal (PCLK/2 or counter input), the trigger rising edge must satisfy a setup time to the preceding falling edge of the clocking signal. As in the external count input, the bit of the port must be available for use by the counter/timer, and must be programmed as an input bit. (Even if the port bit is programmed as an output bit, the port pin is used as the counter/timer input [if enabled], allowing the CPU to write this input directly.)

External Gate Enable--EGE (D_3). By setting EGE to 1, the I/O line of the port associated with the counter/timer (see Table 4-1) is used as an external gate input to the counter/timer. If the external gate input is a 0 (assuming the data path is programmed non-inverting), the countdown sequence is suspended; forcing it to a 1 enables the countdown sequence to continue. To guarantee the enabling or disabling of the counter/timer for a particular rising edge of the clocking signal (PCLK/2 or counter input), the gate input must satisfy a setup time to the preceeding falling edge of the clocking signal. Like external trigger input, the bit must be available and it must be programmed to be an input. (Even if the port bit is programmed as an output bit, the port pin is used as the counter/timer input if enabled. This allows the CPU to write this input directly.)

Retrigger Enable Bit--REB (D_2). If REB is set to 0, triggers (internal or external) which occur during a countdown sequence are ignored. If REB is 1, each trigger causes the time constant value to be reloaded and a new countdown sequence to be initiated. When a counter/timer is programmed in square-wave mode, a retrigger will cause the Time Constant value to be reloaded and the new countdown will start on the first half of the square-wave cycle.

Output Duty Cycle Selects--DCS_1 & DCS_0 (D_1 & D_0). These two bits select the output duty cycle according to the information indicated in Table 2-9.

Table 2-9. Output Duty Cycle Selects

DCS_1	DCS_0	Output Duty Cycle
0	0	Pulse Output
0	1	One-Shot Output
1	0	Square Wave Output
1	1	- DO NOT USE -

(See Section 4.2.5 for a description of each output duty cycle type.)

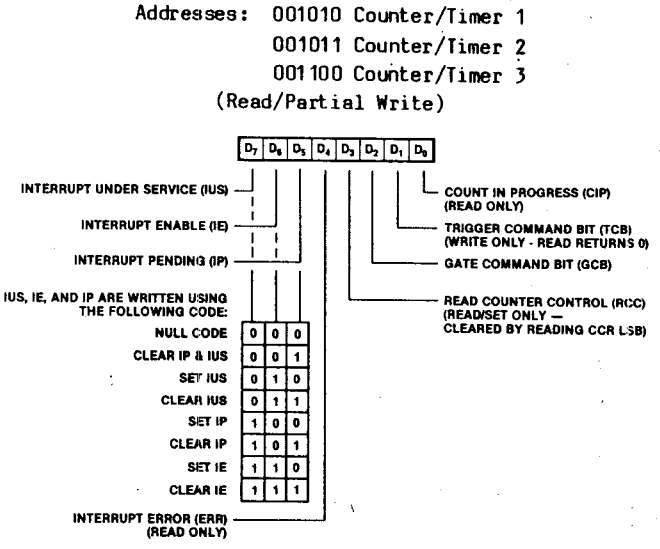


Figure 2-16. Counter/Timer Command and Status Registers

2.9.2 Counter/Timer Command and Status Registers

Each Counter/Timer Command and Status register contains the primary command and status bits for its counter/timer and (in most cases) will be the register most often accessed. A reset forces all bits to 0. The detailed bit descriptions will discuss whether or not a bit can be read or written.

Interrupt Under Service--IUS (D_7). The operation is the same as the port IUS bit.

This status bit is automatically set to 1 if its corresponding IP is the highest-priority interrupt request pending when an Interrupt Acknowledge sequence takes place. It can also be set directly by CPU command. As long as it is set, the same and lower-priority sources of interrupt are inhibited from requesting interrupts via the internal and external daisy-chains. It can be cleared only by CPU command. This bit is read/write. It is changed by writing to the Counter/Timer Command and Status register of the port using the code shown in Figure 2-16.

Interrupt Enable--IE (D_6). The operation is the same as the port IE bit.

This bit enables or disables the counter/timer's interrupt logic. When IE is cleared to 0, the counter/timer is unable to request an interrupt or to respond to an Interrupt Acknowledge. It does not affect the normal operation of IP or IUS, but simply masks IP off from the rest of the device. A 1 in IUS still affects the interrupt daisy-chain. If IE is programmed to be 1, the interrupt logic operates normally. This bit is read/write. It is changed by writing to the Counter/Timer Command and Status register of the port using the code shown in Figure 2-16.

Interrupt Pending--IP (D_5). The operation is similar to the port IP bit.

IP is a status bit which, when set to 1, indicates that the counter/timer requires servicing. It is automatically set to 1 each time the counter/timer reaches its terminal count (or by the CPU command). If IE is also 1 and no higher-priority interrupts are under service, then the $\overline{INT}$ line is pulled Low to request an interrupt. This bit is read/write. It is changed by writing to the Counter/Timer Command and Status register using the code shown in Figure 2-16.

Interrupt Error--ERR (D_4). This status bit is set along with IP to indicate that an error has occurred. An error occurs for a counter/timer whenever terminal count is reached and IP is still set from a previous terminal count. ERR can be cleared only by having software clear the IP it corresponds to. ERR is a read-only bit.

Read Counter Control--RCC (D_3). RCC is a command bit that enables the counter/timer to be read reliably while it is in a countdown sequence. Writing a 1 to RCC causes the contents of the Counter/Timer Current Count register (CCR), which normally follows the down-counter, to be frozen

until the least-significant byte of the CCR is read. Reading the RCC bit indicates when the CCR is frozen. RCC can only be set directly and cannot be set unless the Counter/Timer is enabled in the Master Configuration Control register (CT1E, CT2E, or CT3E). RCC can be cleared automatically by reading the least-significant byte of the CCR or by disabling the counter/timer via the corresponding enable bit.

Gate Command Bit--GCB (D_2). GCB is a command bit that can be used to halt a countdown sequence. By writing GCB with a 0, the countdown sequence is halted. Returning GCB to 1 allows the sequence to resume where it left off. The state of the GCB bit does not affect the operation of the trigger inputs. GCB is a read/write bit.

Trigger Command Bit--TCB (D_1). Writing a 1 to the TCB triggers the counter/timer. It causes the down-counter to be loaded with the time constant value and a countdown sequence to be initiated. It can also retrigger the counter/timer if the Retrigger Enable bit (REB) is set to 1. TCB is a write-only bit. When read, it always returns 0. In this way, erroneous trigger comands are not issued when bit set or clear operations are performed on the other bits in this register.

Count In Progress--CIP (D_0). CIP is a status bit that indicates if a countdown sequence is in progress. It is automatically set to 1 when the counter/timer is triggered and the down-counter is loaded with the time constant value. It is automatically reset to 0 when the down-counter reaches a count of 0. The state of the gate inputs (internal and external) has no effect on this bit. CIP is read-only.

2.9.3 Counter/Timer Time Constant Registers

Each of the Time Constant registers is 16-bits and holds the value loaded into the down-counter of its counter/timer when a trigger is detected. It is accessed by the CPU as two consecutive 8-bit registers (bit 7 of the most-significant byte is bit 15 of the Time Constant register). These registers can be read and written at any time. However, care must be taken when writing them so that a trigger does not occur while the time constant value is changing. A reset does not effect the Time Constant register.

Addresses: 010110 Counter/Timer 1's MSB
010111 Counter/Timer 1'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)

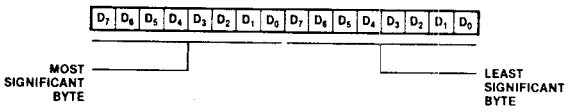


Figure 2-17. Counter/Timer Time Constant Registers

2.9.4 Counter/Timer Current Count Registers

Each of the Counter/Timer Current Count registers (CCR) is a 16-bit register used to read the contents of its counter/timer down-counter. The CCR follows the down-counter until the RCC bit in the Counter/Timer Command and Status register is written with a 1. The value present when the write occurs is held until the least-significant byte is read. Then, the CCR follows the down-counter again. The countdown sequence is not affected. The CCR is accessed as two consecutive 8-bit registers (bit 7 of the most-significant byte is bit 15 of the Time Constant register). They can be read at anytime, whether or not the

Addresses: 010000 Counter/Timer 1's MSB
010001 Counter/Timer 1'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)

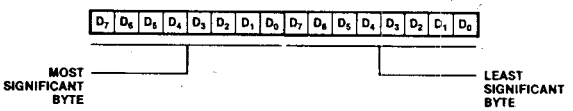


Figure 2-18. Counter/Timer Current Count Registers

value is frozen. Writes to the CCR are ignored. A reset forces the CCR to follow the down-counter (neither are forced to a specific value).

2.10 INTERRUPT RELATED REGISTERS

These registers contain the interrupt vectors output during Interrupt Acknowledge sequences. Three vector registers are provided: one for Port A, one for Port B, and one shared by the three counter/timers. Another register is provided, which facilitates using this device in a polled environment.

2.10.1 Interrupt Vector Registers

Each of the Interrupt Vector registers holds the interrupt vector returned when the source of interrupt associated with its port is acknowledged. The interrupt vector value is user-defined by writing the desired 8-bit identification code to this register when initializing the CIO. A modified version of the value written to the Interrupt Vector register can be returned if the vector is programmed to include status. This does not affect the value written to the Interrupt Vector register.

Addresses: 000010 Part A
000011 Part B
000100 Counter/Timers
(Read/Write)

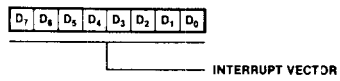


Figure 2-19. Interrupt Vector Register

The Interrupt Vector register is a read/write register. When read, the value returned always includes the status if MIE = 1 (whether or not the associated Vector Includes Status bit is 1). If MIE = 0, the unmodified vector is returned independent of the state of the VIS bit. A reset does not affect the Interrupt Vector register. The status bit outputs are as shown in Table 2-10.

Table 2-10. Interrupt Vector Register Status Bits

Port Vector Status			
OR-Priority Encoded Vector Mode:			
D_3	D_2	D_1	
x	x	x	Number of highest-priority bit with a match
All Other Modes:			
D_3	D_2	D_1	
ORE	IRF	PMF	Normal
0	0	0	Error
Counter/Timer Status			
D_2	D_1		
0	0		Counter/Timer 3
0	1		Counter/Timer 2
1	0		Counter/Timer 1
1	1		Error*

*The error status indicates that the highest-priority counter/timer with an interrupt pending also has its ERR flag set. The CPU must poll the Command and Status registers to determine which counter/timer has its ERR flag set.

2.10.2 Current Vector Register

When the Current Vector register is read, it returns the interrupt vector that would have been output by the device during an Interrupt Acknowledge cycle if its IEI input had been High. The vector returned corresponds to the highest priority IP independent of the IUS. The order of priority (highest to lowest) is: Counter/Timer 3, Port A, Counter/Timer 2, Port B, Counter/Timer 1. If no enabled interrupts are pending, a pattern of all 1s is output. This is useful in a polled environment or when CPU does not read vectors. This register is a read-only register. Since a reset disables all interrupts, reading the Current Vector register after a reset will return all 1s.

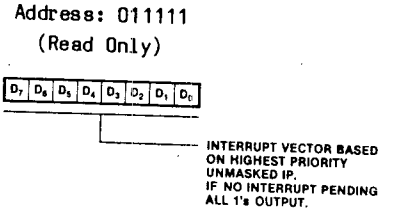


Figure 2-20. Current Vector Register

Chapter 3
I/O Port Operation

3.1 OVERVIEW

There are three I/O ports provided by the CIO device. Ports A and B are 8-bit general-purpose ports; Port C is a 4-bit special-purpose port. There are two port configurations: bit port and port with handshake. All three ports can be programmed as bit ports; only Ports A and B can function as handshake ports.

In general, bit ports are used to provide status input lines and control output lines. When the I/O ports are configured as bit ports, data can be moved in either direction on an individual, pin-by-pin basis. There are up to twenty pins available for this kind of data handling by the three ports.

By configuring Ports A and B as ports with handshake (input, output, or bidirectional), the data can be moved in either direction on a byte-by-byte (parallel 8-bit or 16-bit) basis. Four different handshakes are available: Interlocked, Strobed, Pulsed, or 3-Wire.

Port C is a 4-bit wide, special-purpose port that provides the handshake control lines for Ports A and B, when required. A REQUEST/$\overline{\text{WAIT}}$ line can also be provided to synchronize Port A and B data transfers with DMAs or CPUs. Any Port C bits not used as handshake lines can be used as I/O lines.

Another I/O Port function is to provide external access for the control of three independent counter/timers and distribution of their outputs. Port B provides access for Counter/Timers 1 and 2. Port C provides access to Counter/Timer 3.

Pattern-recognition capability is provided in Ports A and B. In general, it is possible to test data for specified patterns and to generate interrupt requests based on the match obtained.

3.2 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, 0-to-1 transition, 1-to-0 transition, or any transition. Individual bits can be masked off. Three modes of pattern-recognition operation are supported: AND, OR, and OR-Priority Encoded Vector (OR-PEV). A pattern match is defined as the simultaneous satisfaction of all nonmasked bit specifications in the AND mode or the satisfaction of any nonmasked bit specifications in either the OR or OR-PEV modes.

The pattern specified in the Pattern Definition register assumes that the data path is programmed to be non-inverting. 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/non-invert logic.

The operation of the pattern-recognition logic in the various port modes will be described in detail in the following sections.

3.3 BIT PORT OPERATION

Bit ports are used to provide the CPU with input lines to monitor status, and with output lines to provide control. There are up to twenty bits available for this type of data handling provided by the three ports of the CIO: eight each by Ports A and B and four by Port C.

Writing the data register of a bit port updates the value being output by all output bits in the port. Reading the data register of the bit port returns the state of all bits, outputs as well as inputs.

3.3.1 Bit Port Simple Operation

The port's Data Direction register specifies the direction of data flow for each bit of a bit port. A 1 specifies an input bit; a 0 specifies an output bit.

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

The value returned when reading an input bit reflects the state of the input just prior to the read. A 1'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 1's catcher input, its output is automatically set to 1 until it is cleared by software. The 1's catcher is cleared by writing a 0 to the corresponding bit in the data register. In all other cases, attempted writes to input bits are ignored. The 1's catcher is level-sensitive. If the input is still a 1 when it is cleared, the output will again be set to a 1. Also, the input to the 1's catcher follows the invert/non-invert logic. If the bit is programmed inverting, a low voltage level at the pin will cause the 1's catcher output to go to a 1.

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 pins. 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 (0-4, 1-5, 2-6, and 3-7). With this feature, any combination of bits can be set or cleared (while other bits remain undisturbed), without first reading the register.

3.3.2 Bit Port Pattern-Recognition Operation

Ports A and B contain pattern-recognition logic, which enables the port to detect a user-specified pattern and to generate an interrupt request when the pattern is detected. Pattern-recognition may be performed on all bits, including those used as I/O for the counter/timers. For input bits, the input to the pattern-recognition logic reflects the value on the pins (through the invert/non-invert logic) in all cases except for inputs with 1's catchers. In this case, the output of the 1's catcher is used. For output bits, this is the value being output before the invert/non-invert logic is used. When operating in the AND or OR mode, the transition from a no-match to a match state causes the interrupt. In the OR mode, if a second match occurs before the first match goes away, it does not cause a second interrupt. Bit ports specified in the OR-PEV mode generate interrupts as long as a match state exists. A transition from a no-match to a match state is not required. Since a match condition only lasts a short time when transition patterns are specified, care must be taken--no more than one bit should be programmed with a transition match specification in a port operating in the AND mode.

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 = 1), the state of all the port inputs at the time the match was detected is latched in the Buffer register and held until IP is cleared. In all cases, the Pattern Match Flag (PMF) in the port's Command and Status register indicates the state of the port at the time the PMF is read. Only Transparent mode (LPM = 0) is supported when OR-PEV is specified. In all modes, the port's IP bit is set and an interrupt generated (if enabled) when the pattern match is detected. The IP can only be cleared by a command to the Port Command and Status register.

If a second 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 first IP is cleared, the IP is automatically set to 1 along with the Interrupt Error (ERR) flag. Matches occurring while ERR is set are ignored. ERR is automatically cleared when the corresponding IP is cleared by software.

When a pattern-match is present in the OR-Priority Encoded Vector mode, IP is set to 1. The IP can-

not 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 bit and bit 0 is the lowest-priority bit. 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 Interrupt Acknowledge cycle is initiated, the vector is frozen until the corresponding Interrupt Under Service (IUS) is cleared. If an input that causes interrupts changes before the interrupt is serviced, the 1's catcher can be used to hold the value. Bits should not be specified with transition detection, because the match will no longer be valid at the time of the Interrupt Acknowledge. If no match is present at the time of the Acknowledge, the vector will indicate the lowest-priority bit (Bit 0).

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 programmer error detection is performed in this mode and the Interrupt on Error bit should be 0.

One application of the OR-PEV pattern match mode is to use the CIO as a Programmable Interrupt Controller (PIC). This facilitates using non-Z-BUS peripherals with a Z-BUS CPU. (See Chapter 5 for further discussion.)

3.4 HANDSHAKE PORT 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 or out of the port and interrupt generation is under the control of the handshake logic. Port C provides the handshake lines, as shown in Table 3-1.

When Ports A and B are configured as ports with handshake, they are single- or double-buffered according to the setting of the Single Buffered Mode (SB) bit of their respective Port Mode Specification registers.

The double-buffered mode (SB = 0) 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. The Single-Buffered mode (SB = 1) is useful if the handshake line must be stopped on a byte-by-byte basis.

Normally, the Interrupt Pending (IP) bit is set and an interrupt is generated when data is moved into the Input Data register (input port) or out of the Output Data register (output port). For input and output ports, the IP is normally cleared automatically when the data is read or written. In bidirectional ports, IP is cleared only by software command to the port Command and Status register. 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 16 bits (two bytes) 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.

Pattern recognition logic is also available for use with the port with handshake. Each time a byte is moved into the Input Data register or out of the Output Data register, the pattern match flag indicates whether a match has occurred.

Ports A and B can be linked to form a 16-bit port by programming a 1 in the Port Link Control (PLC) bit of the Master Configuration Control register. In this mode, only the Port A Handshake Specification and Command and Status registers are used, and 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, the Port B Data register must be read or written before the Port A Data register.

When a port is specified as a port with handshake, its mode (input, output, or bidirectional) determines the direction of data flow. The data direction for the bidirectional port is determined by a bit in Port C (See Table 3-1). In all cases, the contents of the port's Data Direction register are ignored. The contents of the Special I/O Control register apply only to output bits (normal or open-drain). Input ports with handshake do not have 1's catchers; therefore, those bits in the Special I/O Control register are ignored. Port C lines used for handshake can all be programmed as inputs because the handshake specification overrides the Port C Data Direction register for bits that must be outputs. All other Port C options are available (polarity, 1's catcher, open-drain outputs, etc.).

Table 3-1. Port C Pin Utilization

Port A/B Configuration	Port C Bits: Pin C_3	Pin C_2	Pin C_1	Pin C_0
Ports A & 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 $\overline{DAV}$	$\overline{ACKIN}$	REQUEST/$\overline{WAIT}$ or Bit I/O	Bit I/O
Port B = Input or Output port (Interlocked, Strobed, or Pulsed Handshake)*	REQUEST/$\overline{WAIT}$ or Bit I/O	Bit I/O	RFD or $\overline{DAV}$	$\overline{ACKIN}$
Port A or B = Input port (3-Wire Handshake)	RFD (Output)	$\overline{DAV}$ (Input)	REQUEST/$\overline{WAIT}$ or Bit I/O	DAC (Output)
Port A or B = Output port (3-Wire Handshake)	$\overline{DAV}$ (Output)	DAC (Input)	REQUEST/$\overline{WAIT}$ or Bit I/O	RFD (Input)
Port A or B = Bidirectional port (Interlocked or Strobed Handshake)	RFD or $\overline{DAV}$	$\overline{ACKIN}$	REQUEST/$\overline{WAIT}$ or Bit I/O	IN/$\overline{OUT}$

* Both Ports A & B can be specified input or output with Interlocked, Strobed, or Pulsed Handshake at the same time if neither uses REQUEST/$\overline{WAIT}$. However, only one port can use the Pulsed Handshake at a time.

3.4.1 Four Handshake Modes

There are four handshake modes: Interlocked, Strobed, Pulsed, and 3-Wire.

3.4.1.1 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. 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 data is no longer available, thereby acknowledging the input port's acceptance of the last byte. This handshake allows the CIO to interface directly to the port of a Z8 microcomputer, a UPC, an FIO, an FIFO, or to another CIO port, etc., with no external logic.

3.4.1.2 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 ($\overline{ACKIN}$) strobes data into or out of the port. In contrast to the Interlocked Handshake, the signal indicating that the port is ready for another data transfer operates independently of the $\overline{ACKIN}$ input. The external logic must ensure that data does not transfer at too fast or too slow a rate.

3.4.1.3 Pulsed Handshake

The Pulsed Handshake mode is designed to interface to mechanical-type devices which 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 (Counter/Timer 3) is linked to the

handshake logic. If the port is specified in the input mode, the timer is inserted in the $\overline{ACKIN}$ path. The external $\overline{ACKIN}$ input triggers the timer, and its output (an internal delayed $\overline{ACKIN}$) is used as the Interlock Handshake's normal Acknowledge input. If the port is an output port, the timer is placed in the Data Available ($\overline{DAV}$) output path. The timer is triggered when the normal Interlocked Handshake $\overline{DAV}$ output goes Low and the timer output is used as the actual $\overline{DAV}$ output. The counter/timer maintains all of its normal capabilities. This handshake is not available to bidirectional ports.

3.4.1.4 3-Wire Handshake

The 3-Wire Handshake mode is designed for situations 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, the rising edge of one status line indicates that the port is ready for data (RFD), and the rising edge of another status line indicates that the data has been accepted (DAC). With the 3-Wire Handshake, the output lines on many input ports can be bused together (wire-AND) with open-drain drivers; the output port knows when all the ports have accepted the data and are ready. This is the same handshake 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. However, because the direction of the port can be changed under software control, bidirectional IEEE-488-type transfers can be performed.

3.4.2 Input Port With Handshake

An input port handles data movement from the CIO port pins to the CPU. This allows 8-bit data (or 16-bit if the Ports A and B are linked) to be read from external devices. (See Figure 3-1.)

Only one of the three Bit Path Definition registers affects input port operations: the data path is modified as specified by the Data Path Polarity register (Section 2.5.1). Both the Data Direction and Special I/O registers are ignored.

Since the port mode of operation is independent of the handshakes, the port operation modes will be examined first, independent of the handshake types. This will then be followed by an examination of the four handshake types (Interlocked, Strobed, Pulsed, and 3-Wire) in the input port context.

3.4.2.1 Basic Modes of Operation

There are three independent modes of operation that, taken together, characterize a particular input port configuration. These modes of operation are:

- double- or single-buffered
- interrupted on one or two bytes
- pattern match logic used or not used

Double-Buffered (SB = 0).

When the input port is specified as double-buffered (SB = 0) in the Port Mode Specification register, input data is latched in the Buffer register by the handshake logic. The falling edge of Acknowledge Input ($\overline{ACKIN}$) latches the incoming data and the Ready For Data (RFD) output signal goes Low, indicating that the Buffer register is full. When the Input Data register is empty (IRF = 0), the data is moved out of the Buffer register (RFD may now go High depending on the particular handshake) and into the Input Data register causing it to be "filled" (IRF = 1).

The Interrupt on Two Bytes (ITB) command bit of the Port Mode Specification register determines when IP is set and when an interrupt request is generated. When programmed to interrupt on every byte (ITB = 0), Interrupt Pending (IP) of the Port Command and Status register is set (along with Input Data register Full (IRF) of the Port Command and Status register) when the data is shifted into the Input Data register. Reading the Port Data register "empties" the Input Data register and automatically clears the IP, hence, IRF = 0 and IP = 0. While IP can be cleared (IP = 0) by software command (by writing bits D_7-D_5 of the Port Command and Status register), the Input Data register is not "emptied" (cleared) until the data is read.

When programmed to interrupt on two bytes (ITB = 1), IP is not set until both the Input Data and Buffer registers are full (the Buffer register becomes full while the Input Data register is full). IP is automatically cleared (IP = 0) when the second byte of data is read by the CPU as

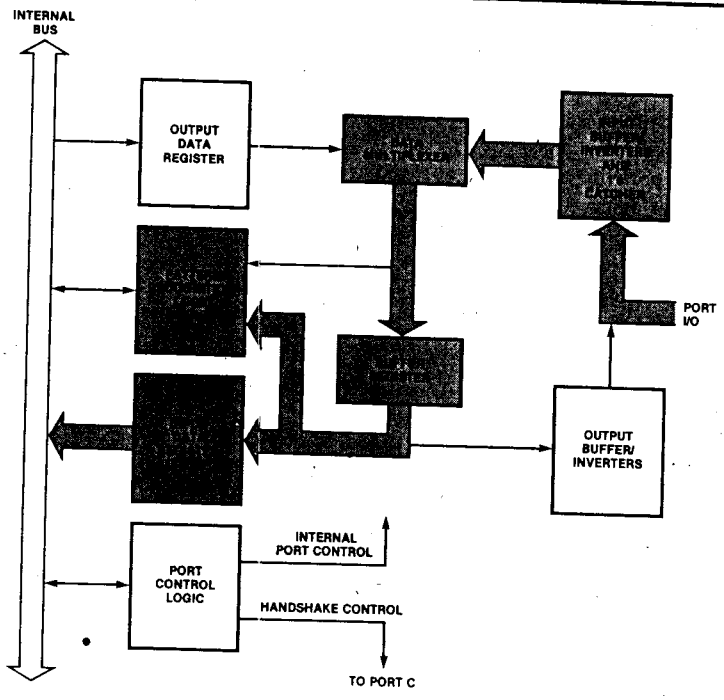


Figure 3-1. Input Port Data Path

follows: the first data read empties the Input Data register, allowing the Buffer register data to be moved into the Input Data register. The second read empties the Input Data register again and automatically clears the IP (IP = 0).

NOTE

When ITB = 1, the Input Data register should not be read unless IP = 1 even if IRF = 1. Otherwise, the data may--at that instant--be read just as a byte is being latched in the Buffer register and IP is being set. Then an interrupt may be generated, but only one byte can be read, because the first byte has already been read.

NOTE

The IP can be cleared on command. This suggests the following possible sequence for providing byte-by-byte control of the RFD output: reading data and then clearing IP. This allows an interrupt on the next byte that is moved into the Buffer register.

Single-Buffered (SB = 1)

When the input port is specified as single-buffered (SB = 1), input data is latched in the Buffer register as in double-buffered mode. However, when the data is moved from the Buffer register to the Input Data register, the Buffer register is not emptied; consequently, RFD stays Low. Reading the Port Data register empties both the Input Data and Buffer registers. As in double-buffered mode, IP is set (IP = 1) when the data is moved into the Input Data register and the IP is automatically cleared when the data is read.

NOTE

ITB = 1 does not make sense in single-buffered operation. Thus, when SB = 1, ITB must = 0.

With Pattern Match Added (PMS_1, $PMS_0 \neq 0$)

The port's built-in pattern match logic can be used to test the incoming data as it is moved into or through the Input Data register. The available pattern match modes operate independently of handshake type and are specified by the Pattern Mode

Specification bits, (PMS_1 and PMS_0). In the input port operation, the AND and the OR modes of Pattern Match operation are available for use, but the OR-PEV is not. This pattern availability for AND and OR logic is a consequence of the data pattern being tested as the data is moved into the Input Data register, eliminating access to the transition information. Because of this, transition patterns cannot be used.

The Pattern Match Flag (PMF) of the Port Command and Status register is automatically set to 1 or cleared to 0 as the data is moved from the Buffer register to the Input Data register. When a byte of data is moved from the Buffer register to the Input Data register, IRF is set to 1, indicating that another byte is available to be read. If the data matches the specified pattern, the PMF is set along with IRF; otherwise, PMF is cleared. If the interrupt vector includes status, it indicates that a match has been detected. Each time data is transferred to the Input Data register, the PMF is updated.

When ITB = 0, IP is set normally when data is moved into the Input Data register. However, if PMF = 1, reading the Input Data register does not automatically clear the IP; the IP can only be cleared by writing to the Port Command and Status register. Also, reading the Input Data register does not "empty" it; the Input Data register can be "emptied" (and IRF cleared to 0) only if it is read and the IP is cleared (in any order).

When ITB = 1, the pattern match logic can override the ITB logic. If the byte moved into the Input Data register matches the specified pattern, the IP will be set immediately. IP is cleared and the Input Data registers are "emptied" in the same manner as when ITB = 0.

In this mode of operation (ITB = 1 and Pattern Match is enabled) care must be taken, because an IP can mean that either one or two bytes are available to be read (depending upon whether the match occurred on the first or second input byte). There is also the possibility that, after a data register read and a clear IP, a second byte matches. There are three conditions which can cause an interrupt:

Condition 1: Two bytes have been received--neither match the pattern

Condition 2: Two bytes have been received--the second byte matches the pattern

Condition 3: One byte has been received--it matches the pattern

Given the above information, the following set of operations will determine the cause of the interrupt and properly process it.

Poll PMF

If PMF is set (a match occurred) — The cause is condition 3
- Read data
- Clear IP
- Return

If PMF is not set
- Read data (reads the first byte)
- Poll PMF (to test the second byte)

    If PMF is set (a match occured) — The cause is condition 2
    - Read data
    - Clear IP
    - Return

    If PMF not set — The cause is condition 1
    - Read data
    - Return

In summary, then, careful interrupt testing and handling is required if ITB = 1 and the pattern match logic is enabled.

Interrupt on Match Only (IMO = 1) Specified

When the Interrupt on Match Only (IMO) bit of the Port Mode Specification register is set to 1, an interrupt will be generated only when the data moved into the Input Data register matches the pattern specification. For input ports, the IMO capability is especially useful when data transfer is under the control of an external device (for example, a DMA controller). In this way the bulk of data transfers can be accomplished without interrupts (that is, without involvement of the CPU), by having an interrupt generated only when the match pattern in encountered.

NOTE

IMO must be 0 if either ITB or SB = 1, or if the port is a bit port.

3.4.2.2 Handshake Types

The operation of the Port A and B input handshakes is explained in this section by describing in detail the sequence of operations performed by an input port programmed with the Interlocked Handshake. Any differences encountered when using the other handshakes will then be described. Table 3-1 identifies the handshake lines furnished by Port C bits for Ports A and B.

Interlocked Input Handshake

As noted in Section 3.3, the Interlocked Input Handshake requires the input port to not indicate that it is ready for data until the data source indicates that the previous byte of data is no longer available, thereby acknowledging that the input port has accepted the previous byte. A primary benefit of Interlocked Handshake port configuration is that it allows the CIO to communicate directly with a variety of other devices without the need for intervening external logic. Devices such as another Z-CIO/CIO, an FIO, an FIFO, a Z8 Port, etc., can be directly connected and serviced. Figure 3-2 shows two interconnected CIOs: output port's $\overline{DAV}$ output connects to input port's $\overline{ACKIN}$ input and input port's RFD output connects to output port's $\overline{ACKIN}$ input.

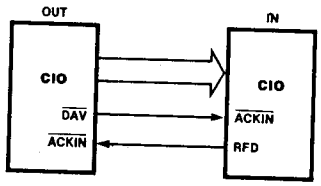


Figure 3-2. Two Interconnected CIOs Using Interlocked Handshake

In Interlocked Handshake mode (Figure 3-3), on the falling edge of the Acknowledge Input ($\overline{ACKIN}$), the data on the port input lines is latched in the Buffer register. This fills the Buffer register and the Ready for Data (RFD) output is pulled Low. If the Input Data register is empty, the data is moved to it ("emptying" the Buffer register) and the Input Data register Full (IRF) flag is automatically set to 1. When the Buffer register becomes empty (and if $\overline{ACKIN}$ is High), the RFD line will return High only if the $\overline{ACKIN}$ input is High. This achieves the interlock.

The following example provides a step-by-step analysis of a double-buffered input port using Interlocked Handshake. (This description uses Figure 3-3 as reference).

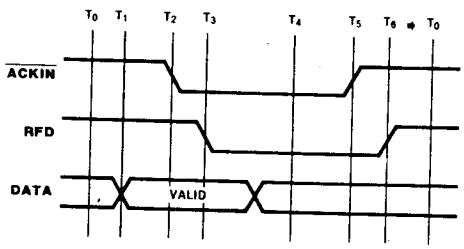


Figure 3-3. Interlocked Input Handshake Timing Diagram

T_0. $\overline{ACKIN}$ and RFD are both High. $\overline{ACKIN}$ High indicates that the data is not valid; RFD High indicates that the Buffer register is empty and ready for data.

T_1. Data on port pins becomes valid.

T_2. $\overline{ACKIN}$ goes Low, indicating that the data is valid, and causing it to be latched into the Buffer register.

T_3. RFD goes Low, indicating that the Buffer register is full and the port is not ready for more data.

T_4. The data is transferred into the Input Data register, the Input Data register Full (IRF) flag goes High and the Buffer register is emptied. The port is now ready for the next byte of data. RFD could go High if $\overline{ACKIN}$ is High; but because $\overline{ACKIN}$ is Low, RFD stays Low.

T_5. $\overline{ACKIN}$ goes High.

T_6 **becomes** T_0. RFD goes High, concluding the handshake process; the cycle is ready to repeat.

Strobed Input Handshake

The Strobed Handshake (Figure 3-4) operates in the same way as the Interlocked Handshake, except that the rising edge of the RFD output is independent of $\overline{ACKIN}$ going High. As soon as the Buffer register is emptied, RFD goes High, even if $\overline{ACKIN}$ is still Low. In all other respects, the two handshakes are the same. The falling edge of the $\overline{ACKIN}$ input "strobes" the data into the port.

The following example provides a step-by-step analysis of an input port configured as double-buffered and using Strobed Handshake. (This description uses Figure 3-4 as reference.)

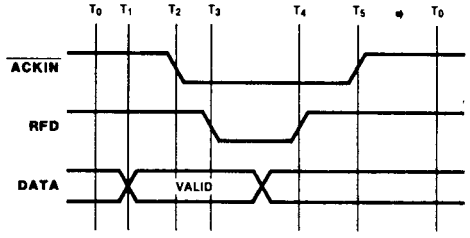


Figure 3-4. Strobed Input Handshake Timing Diagram

T_0. $\overline{ACKIN}$ and RFD are both High. $\overline{ACKIN}$ High indicates that the data is not valid; RFD High indicates that the Buffer register is empty and ready for data.

T_1. Data on port pins becomes valid.

T_2. $\overline{ACKIN}$ goes Low, indicating that the data is valid, and causing it to be latched in the Buffer register.

T_3. RFD goes Low, indicating that the Buffer register is full and the port is not ready for more data.

T_4. The data is moved into the Input Data register, the Input Data register Full (IRF) flag goes High, the Buffer register is emptied, and RFD goes High.

T_5 **becomes** T_0. $\overline{ACKIN}$ is High; the cycle is ready to repeat.

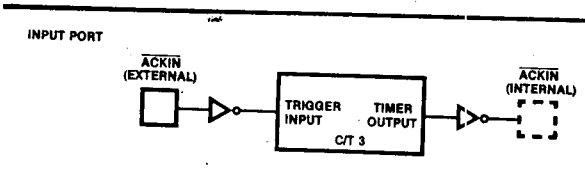


Figure 3-5. Pulsed Input Handshake Counter/Timer Insertion

Pulsed Input Handshake

The Pulsed Handshake operates exactly like the Interlocked Handshake with Counter/Timer 3 inserted in the $\overline{\text{ACKIN}}$ input path (see Figure 3-5). The counter/timer is triggered on the falling edge of $\overline{\text{ACKIN}}$. The output of the timer is inverted and used as the Acknowledge input for the handshake. The falling edge of the internal $\overline{\text{ACKIN}}$ latches the data, and RFD cannot go High until the internal $\overline{\text{ACKIN}}$ goes High. Because all of the programmable capabilities of Counter/Timer 3 are available, many different operations are possible. However, since only Counter/Timer 3 is used, only one port can have Pulsed Handshake at a time.

If the counter/timer output duty cycle is programmed in the pulse mode, TC cycles (where TC is the value programmed in the counter/timer Time Constant register) after the $\overline{\text{ACKIN}}$ falling edge is detected, the Internal Acknowledge falls, latching the data in the Buffer register; the internal $\overline{\text{ACKIN}}$ rises a cycle later. If the counter is programmed with the one-shot duty cycle, then as soon as the external falling edge on the $\overline{\text{ACKIN}}$ input is detected, the Internal Acknowledge falls; it rises TC cycles later. When the counter/timer duty cycle is selected to be square-wave, the Internal Acknowledge goes Low TC clock cycles after $\overline{\text{ACKIN}}$ falls and stays Low for TC cycles. See Figure 3-6 for timing diagrams of the three different duty cycles (pulsed, one-shot, and square-wave) available with Counter/Timer 3.

Because the handshake is interlocked with the internal $\overline{\text{ACKIN}}$ (not the $\overline{\text{ACKIN}}$ port pin), the RFD output is held Low as long as the Internal Acknowledge is Low. This can be used to guarantee a minimum RFD Low time without CPU intervention. Many simple interfaces can be made with the Pulsed Handshake by connecting the RFD output (inverted) to the $\overline{\text{ACKIN}}$ input.

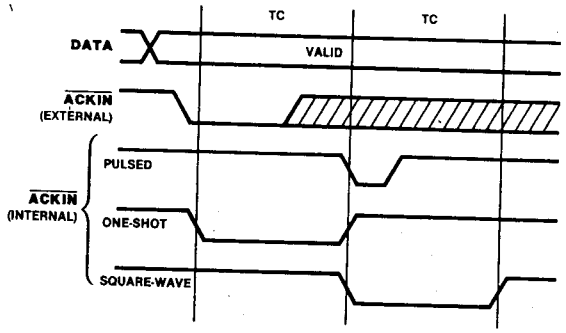


Figure 3-6. Pulsed Input Handshake Counter/Timer Duty Cycles

3-Wire Input Handshake

The 3-Wire Handshake (Figure 3-7) is the same as the Interlocked Handshake, except that the role of the Ready for Data (RFD) output is replaced by two signals: RFD and Data Accepted (DAC). The name of the $\overline{\text{ACKIN}}$ input is changed to Data Available ($\overline{\text{DAV}}$) to be consistent with the IEEE-488 specification, but its function is the same. The RFD output goes High when the Buffer register is empty and the $\overline{\text{DAV}}$ input is High. When $\overline{\text{DAV}}$ falls, the input data is latched, RFD goes Low (the Buffer register is full), and DAC goes High, indicating that the data was received. When the $\overline{\text{DAV}}$ input goes High, DAC is forced Low. When the Buffer register is emptied, RFD goes High, indicating that the port is ready for the next byte. Like the Interlocked Handshake, RFD will not go High until the $\overline{\text{DAV}}$ input goes High. The operation of the interrupt logic is the same as for the Interlocked Handshake.

One Talker--Many Listeners

The 3-Wire Handshake is useful in the situation in which there is one source of data (the "Talker") and many receivers ("Listeners") for the data.

Each Listener has one input line called Data Available ($\overline{\text{DAV}}$) and two output lines called Ready For Data (RFD) and Data Accepted (DAC). The RFD and DAC lines of all Listeners are connected together in a wire-AND.

(A wired-AND requires all inputs [Listener's signals] to go High before the output [signal to the Talker] goes High.) The RFD signal to the Talker

tells the Talker when the last Listener is ready to receive data (final Listener's RFD goes High). The Talker then tells the Listeners that data is now available by bringing $\overline{DAV}$ Low.

Each Listener, working at its individual pace, signals when data reception is done by letting its DAC line go High. The wired-AND DAC signal will tell the Talker when the last Listener has accepted the current data (the last Listener's DAC finally goes High).

The Talker then tells the Listeners that the data is no longer valid ($\overline{DAV}$ goes High). Each Listener puts DAC Low and, when ready for new data, puts RFD high.

The wired-AND RFD and the Low DAC tells the Talker that all Listeners are ready for new data, and the cycle is ready to begin again.

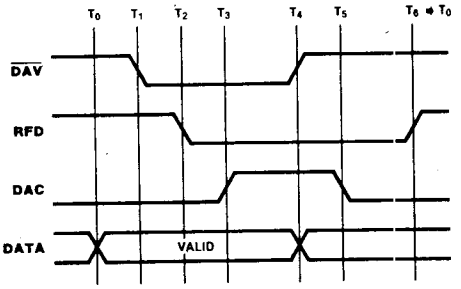


Figure 3-7. 3-Wire Input Handshake Timing Diagram

This procedure is in agreement with the IEEE-488 3-Wire Handshake.

The following example provides a step-by-step analysis of an input port configured as double-buffered and using the 3-Wire Handshake. (This description uses Figure 3-7 as reference).

T_0. $\overline{DAV}$ and RFD are both High. $\overline{DAV}$ High indicates that the data is not valid.

T_1. $\overline{DAV}$ goes Low, indicating that the data is ready to be read and that the input ports are ready for data (the interlock requires RFD to be High before the output port can lower $\overline{DAV}$. The data is latched into the input port Buffer register.

T_2. RFD goes Low, indicating that the Buffer register is full and the port is not ready for more data.

T_3. The individual input ports allow DAC to go High. The wire-ANDed DAC goes High, indicating that the data was received by all input ports.

T_4. $\overline{DAV}$ goes High as the start of the completion of data handling handshake, acknowledging that the data was received.

T_5. DAC goes Low as the data transfer is completed.

T_6 becomes T_0. When their Buffer registers are empty, the individul input ports allow RFD to go High. The wire-ANDed RFD goes High, indicating that the port is ready for the next byte; the cycle is ready to repeat.

3.4.3 Output Port With Handshake

Output ports handle data movement from the CPU to the CIO port pins. This allows the writing of 8-bit (or 16-bit if Ports A and B are linked) data to external devices. (See Figure 3-8.)

There are two Bit Path Definition registers that can affect output port operation: The data path

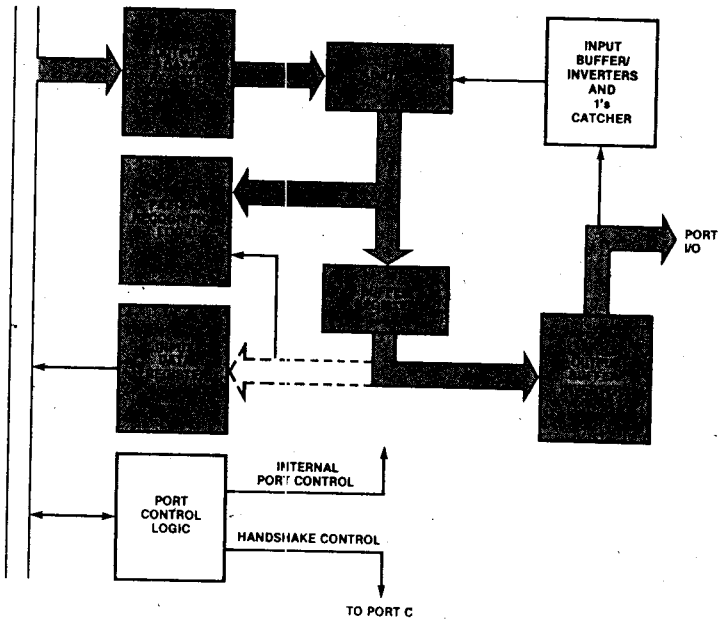


Figure 3-8. Output Port Data Path

is modified as specified by the Data Path Polarity register (see Section 2.6.1); and the Special I/O Control register allows selection of either normal or open-drain outputs (see Section 2.6.3).

When a port is programmed in the output mode, its Input Data Register Full (IRF) flag of the Port Command and Status register is automatically held at 0.

Because the port operation is independent of the handshake, the port operation modes will be examined in this section independent of handshake types. This will be followed by an examination of the four handshake types (Interlocked, Strobed, Pulsed, and 3-Wire) in the output port context.

3.4.3.1 Basic Modes of Operation

There are three independent modes of operation that, taken together, characterize a particular output port configuration. These modes of operation are:

- double- or single-buffered
- interrupt on one or two bytes
- using or not using pattern match logic

Double-Buffered (SB = 0)

The CPU writes data to the Output Data register. The data is moved to the Buffer register if it is empty. When the output port is specified as double-buffered (SB = 0) in the Port Mode Specification register, the data move to the Buffer register "empties" the Output Data register, setting the Output Data Register Empty (ORE) flag; the CPU can then write another byte into the Output Data register. The falling edge of $\overline{\text{ACKIN}}$ indicates that the data has been taken, and empties the Buffer register. Reading the Input Data register will return the current value in the Buffer register.

The $\overline{\text{DAV}}$ output tells receivers that the output port data is available and valid when this signal is Low.

The Interrupt on Two Bytes (ITB) control bit of the Port Mode Specification register determines when IP is set and when an interrupt should be requested.

While programmed to interrupt on every byte (ITB = 0), Interrupt Pending (IP) of the Port

Command and Status register is automatically set to 1 along with Output Data Register Empty (ORE) of the Port Command and Status register when the data is moved out of the Output Data register and into the Buffer register. Writing to the port data register "fills" the Output Data register and automatically clears the IP (hence, ORE = 0 and IP = 0). IP can be cleared (IP = 0) by software command; that is, by writing bits D_7-D_5 of the Port Command and Status register. However, the Output Data register is not "filled" until the data is written. Since IP is set only when data is moved out of the Output Data register, then if the port is enabled (PAE or PBE is set to 1) without writing any data, IP will not be set even though the Output Data register is empty.

While programmed to interrupt on two bytes (ITB = 1), IP is not automatically set to 1 until both the Output Data and Buffer registers are empty; that is, the Buffer register becomes empty while the Output Data register is empty. IP is automatically cleared (IP = 0) when the second byte of data is written by the CPU to the data register. The first data write fills the Output Data register and allows the data to be moved into the Buffer register. The second data write fills the now empty Output Data register and automatically clears the IP. As when ITB = 0, when the port is initially enabled, the IP will not be set nor will an interrupt request be generated until a byte of data is written to it. Data can be written to it after it is initially configured and before it is enabled.

When ITB = 1, the Output Data register should not be written unless IP = 1, even if ORE = 1. Otherwise, the data may be written just as the Buffer register is going empty and as IP is being set. In this case, an interrupt may be requested for a two-byte write, when only one byte is needed, because the first byte has already been written but not output.

NOTE

The IP can be cleared on command. This suggests the following possible sequence for providing byte-by-byte control of the DAV output: writing one byte of data and then clearing IP by command allows an interrupt when the Buffer register is "emptied" by the handshake logic.

Single-Buffered (SB = 1)

When the output port is specified as single-buffered (SB = 1), output data is moved from the Output Data register to the Buffer register as in the double-buffered case. However, when the data is moved into the Buffer register, the Output Data register is not emptied. A copy of the data is maintained there. In this mode, the handshake logic "empties" both registers: when $\overline{ACKIN}$ falls, the Output Data and Buffer registers are emptied, and both ORE and IP are automatically set to 1. Writing the port data register fills the Output Data register and clears IP (ORE = IP = 0).

NOTE

Unlike the double-buffered operation, ITB = 1 does not make sense in single-buffered operation. Thus, when SB = 1, ITB must = 0.

With Pattern Match Added (PMS_1 = PMS_0 ≠ 0)

The port's built-in pattern match logic can be used to test the data as it is coming into or through the data register. The available pattern match modes operate independently of handshake type and are specified by the Pattern Mode Specification bits (PMS_1 and PMS_0). In the output port operation, the AND and the OR modes are available for use, but the OR-PEV is not. This pattern availability for AND and OR logic is a consequence of the pattern being tested as the data is moved into the Input Data register, eliminating access to the transition information. Because of this, transition patterns cannot be used.

The Pattern Match Flag (PMF) of the Port Command and Status register is set to 1 or is cleared to 0 as the data is moved from the Output Data register to the Buffer register.

When ITB = 0, IP is set when data is moved out of the Output Data register. If PMF = 1, writing the data register does not automatically clear the IP--the IP can only be cleared by writing to the Port Command and Status Register. Also, writing the Output Data register does not "fill" it; the Output Data register can be "filled" (and ORE cleared to 0) only if it is written and the IP is cleared, (in any order). Data that is written

before IP is cleared, however, is latched in the Output Data register.

While ITB = 1, the pattern match logic can override the ITB logic. If the byte moved into the Buffer register matches the specified pattern, the IP goes to 1 immediately. IP is cleared and the Output Data register is filled in the same manner as when ITB = 0.

In the Pattern Match mode of operation, where ITB = 1 and Pattern Match is enabled, care must be taken, because an IP = 1 can mean that either one or two bytes may be written (depending upon whether a match occurred and if the match occured on the first or second output byte). There is also the possibility that, if IP is cleared by writing to Port Command and Status register after a single data register write, no further interrupts will be generated if the byte matched.

A normal assumption when ITB = 1 is that both the Output Data and Buffer registers are empty when an interrupt occurs, and two consecutive writes will fill both registers and clear IP. However, if the first byte written matches the pattern, the second write will not "fill" the Output Data register or clear IP because PMF was set to 1 when the first byte was moved into the Buffer register. If the second byte matches, IP and PMF will be set when it is moved into the Buffer register and an interrupt request will be generated--however, only one byte can be written since only the Output Data register is empty.

When ITB = 1 and both PMS_1 and PMS_0 are not 0, there are three conditions which can cause an interrupt:

Condition 1: 2 bytes were written--neither match the pattern

Condition 2: 2 bytes were written--the second byte matches the pattern

Condition 3: 1 byte was written--it matches the pattern

Given the above information, the following set of operations will determine the cause of the interrupt and properly process it.

```
Poll PMF (this could be Status In Vector)

   If PMF = 0 (both registers are empty)        The cause is condition 1
   • Write 1st byte
   • Poll PMF

        If PMF = 0 (1st byte doesn't match)
        • Write 2nd byte (IP automatically cleared)
        • Return

        If PMF = 1 (1st byte matches)           The cause is condition 3
        • Clear IP
        • Write 2nd byte (if any)
        • Return

   If PMF = 1 (2nd byte matches)                The cause is condition 2
   • Clear IP
   • Write a byte (if any)
   • Return
```

In summary, careful interrupt handling is required if both the ITB and pattern match logic are enabled.

Interrupt on Match Only (IMO = 1) Specified

When the Interrupt on Match Only bit of the Port Mode Specification register is set to 1, an interrupt request will be generated only when the data moved out of the Output Data register into the Buffer register matches the pattern specification. For output ports, the IMO capability is especially useful when data transfer is under the control of an external device (for example, a DMA controller). In this way the bulk of the data transfers can be accomplished without interrupts (that is, without involvement of the CPU) by having an interrupt generated only when the match pattern is encountered. The CIO, through the IMO, allows a long string of bytes to be output without interrupts. This is accomplished by simply waiting on a pattern which is inserted at the end of a string to signal the end of the transmission by way of a single CPU interrupt request. This way, many bytes can be moved with only one interrupt request being generated.

NOTE

IMO must be 0 if either ITB or SB = 1, or if the port is a bit port.

3.4.3.2 Handshake Types

The operation of Port A and B output handshakes will be explained by describing in detail the sequence of operations performed by an output port programmed with Interlocked Handshake. Any differences encountered when using the other handshakes will then be described. See Table 3-1 for identification of the handshake lines furnished by Port C bits for Ports A and B.

Deskew Timer Description

Because external devices may require that the data be valid for a certain minimum amount of time prior to the $\overline{DAV}$ signal being pulled Low (to indicate data available), the CIO provides a separate deskew timer for each port. As data is transferred to the Buffer register, the deskew timer is triggered (if the timer is enabled (DTE = 1). After the number of PCLK cycles (up to 16) specified by the deskew timer Time Constant (see Figure 2-5), $\overline{DAV}$ is allowed to go Low (the interlock may keep $\overline{DAV}$ High). The deskew timer does not extend the time from $\overline{ACKIN}$ rising to $\overline{DAV}$ falling. The deskew timer therefore guarantees that the output data is valid for a specified minimum amount of time before $\overline{DAV}$ goes Low.

Deskew timers are available for output ports independent of the type of handshake employed. Each port has a separate 4-bit deskew timer. Thus, the CIO can provide the proper timing to interface to those external devices that require a large data valid to $\overline{DAV}$ falling setup time. For example, the IEEE-488 specification requires data to be valid for 2 microseconds before $\overline{DAV}$ can go Low.

Interlocked Output Handshake

The Interlocked Output Handshake requires that the output port does not indicate that it has data available ($\overline{DAV}$ goes Low) until the receiving port indicates that the previous byte of data has been accepted (indicated by $\overline{ACKIN}$ being High).

Interlocked Handshake port configuration allows the CIO to communicate directly with a variety of other devices without the need for external logic. Devices such as another CIO, FIO, FIFO, Z8 Port, etc., can be interfaced directly. Figure 3-2 shows two interconnected CIOs: the output port's $\overline{DAV}$ output connects to the input port's $\overline{ACKIN}$ input and the input port's RFD output connects to the output port's $\overline{ACKIN}$ input.

In Interlocked Output Handshake mode (Figure 3-9), the CPU writes data into the Output Data register. If the Buffer register is empty, the data is moved to it and on to the port output pins ("emptying" the Output Data register). The Output Data register Empty (ORE) bit in the port Command and Status register is then set to 1 and Data Available ($\overline{DAV}$) handshake line goes Low (if $\overline{ACKIN}$ is High). The CPU can now write another byte into the Output Data register, setting ORE to 0. When

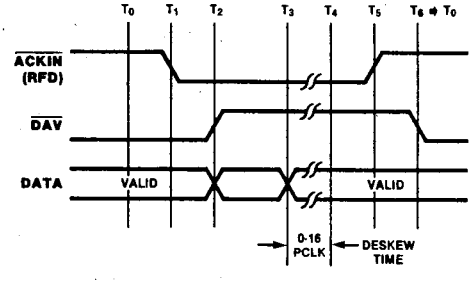


Figure 3-9. Interlocked Output Handshake Timing Diagram

the receiving port has accepted the data it lowers $\overline{ACKIN}$; the Buffer register is "emptied" and $\overline{DAV}$ goes High, indicating that data is no longer available. The data in the Output Data register is moved to the Buffer register and out of the port. At this time the next byte of data is available. However, the $\overline{DAV}$ signal cannot go Low until the $\overline{ACKIN}$ input is High, indicating that the receiving port is ready to accept another byte of data. This achieves the interlock.

The following example is a step-by-step analysis of a double-buffered output port using Interlocked Handshake. (This description uses Figure 3-9 as reference).

T_0. The Buffer register is full, data is valid on the pins, $\overline{DAV}$ is Low, and $\overline{ACKIN}$ is High.

T_1. $\overline{ACKIN}$ goes Low, indicating that the receiver has taken the data and is not ready for more data. This empties the Buffer register.

T_2. $\overline{DAV}$ goes High, indicating that the Buffer register is empty and the data is no longer valid.

T_3. The next byte to be output is moved into the Buffer register from the Output Data register. Data is valid at the port pins. The Deskew Timer is triggered if enabled.

T_4. The Deskew Timer times out, but $\overline{DAV}$ remains High because $\overline{ACKIN}$ is still Low (this is the Interlock).

T_5. $\overline{ACKIN}$ goes High, indicating that the input port is ready for data.

T_6 becomes T_0. $\overline{DAV}$ goes Low, indicating that the data is valid and has been valid for Deskew time programmed; the cycle is ready to repeat.

Strobed Output Handshake

The Strobed Handshake (Figure 3-10) operates in the same way as the Interlocked Handshake, except that $\overline{DAV}$ goes Low independent of $\overline{ACKIN}$. That is, the output port can indicate that new data is available by $\overline{DAV}$ going Low (independent of $\overline{ACKIN}$). In all other respects, the two handshakes are the same. The falling edge of the $\overline{DAV}$ output indicates that the data is ready to be read (just like in Interlocked Handshake). The deskew timers can be used as described in output port with Interlocked Handshake above.

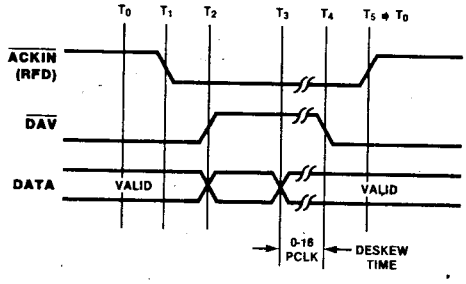


Figure 3-10. Strobed Output Handshake Timing Diagram

The following example is a step-by-step analysis of a double-buffered output port using Strobed Handshake. (This description uses Figure 3-10 as reference).

T_0. The Buffer register is full, data is valid on the pins, $\overline{DAV}$ is Low, and $\overline{ACKIN}$ is High.

T_1. $\overline{ACKIN}$ goes Low, indicating that the receiver has taken the data and is not ready for more data. This empties the Buffer register.

T_2. $\overline{DAV}$ goes High, indicating that the Buffer register is empty and the data is no longer valid.

T_3. The next byte to be output is moved into the Buffer register from the Output Data register. Data is valid at the port pins. The deskew timer is triggered if enabled.

T_4. The deskew timer times out, and $\overline{DAV}$ goes Low, independent of $\overline{ACKIN}$ (there is no $\overline{ACKIN}$ Interlock).

T_S becomes T_0. $\overline{ACKIN}$ goes High, indicating that the input port is ready for data; the cycle is ready to repeat.

Pulsed Output Handshake

The Pulsed Handshake operates exactly like the Interlocked Handshake with Counter/Timer 3 inserted internally in the $\overline{DAV}$ output path (see Figure 3-11). The timer is triggered on the falling edge of an internal $\overline{DAV}$ signal. The output of the timer is inverted and used as the Data Available output for the handshake. The interlock is between the $\overline{ACKIN}$ input and the internal $\overline{DAV}$ signal (the internal $\overline{DAV}$ cannot go Low until $\overline{ACKIN}$ is High). Because all the capabilities of Counter/Timer 3 are available for use, many operations are possible depending on how the timer is programmed. However, since only Counter/Timer 3 is used, only one port can have Pulsed Handshake at a time. The deskew timers can be used to delay the internal $\overline{DAV}$ signal as described before.

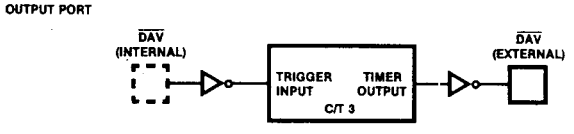


Figure 3-11. Pulsed Output Handshake Counter/Timer Insertion

If Counter/Timer 3's output duty cycle is programmed in the pulse mode, then TC cycles (where TC is the value programmed in the counter/timer Time Constant register) after the internal $\overline{DAV}$ falling edge is detected, the $\overline{DAV}$ output falls and the external $\overline{DAV}$ rises a cycle later.

If the counter is programmed with the one-shot duty cycle, then the $\overline{DAV}$ output falls as soon as the external falling edge on the internal $\overline{DAV}$ signal is detected; it rises TC cycles later. When the counter duty cycle is selected to be square-wave, the $\overline{DAV}$ output goes Low TC cycles after internal $\overline{DAV}$ falls and it stays Low for TC cycles. The duty cycle selected for Counter/Timer 3 determines which $\overline{DAV}$ outputs are available (see the timing diagrams in Figure 3-12).

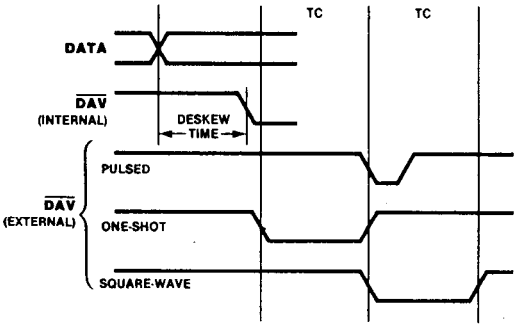


Figure 3-12. Pulsed Output Handshake Counter/Timer Duty Cycles

Many simple interfaces can be made with the Pulsed Handshake by linking the $\overline{DAV}$ output to the $\overline{ACKIN}$ input. For example, if the duty cycle selected for Counter/Timer 3 is square-wave, the port will provide valid data with a setup of TC clock cycles to the external $\overline{DAV}$ falling edge (even longer if the deskew timer is enabled). Also the $\overline{DAV}$'s Low time will be TC clock cycles. This could be used for interfacing to a device which requires a large data setup time to a strobe and a minimum time between characters.

3-Wire Output Handshake

The 3-Wire Handshake (Figure 3-13) is the same as the Interlocked Handshake, except that the role of the $\overline{ACKIN}$ input is replaced by two signals: Ready for Data (RFD) and Data Accepted (DAC). This nomenclature is consistent with the IEEE-488 specification.

When the output port Buffer register is full, its data is available to send. However, the 3-Wire Interlock requires that the receiver(s) first signal that it is ready for data by having DAC Low and raising RFD High (the interlock).

If the deskew timer is enabled, the deskew countdown starts with data moved into the Buffer register. On deskew timeout, the $\overline{DAV}$ signal goes Low; the data has been valid for the whole Deskew count. If the deskew timer is not enabled, the output port then immediately lowers $\overline{DAV}$, signaling that the data is available.

The input port puts RFD Low and, after accepting the data, DAC goes High. The rising edge of DAC "empties" the Buffer register (like $\overline{ACKIN}$ falling in the Interlocked and Strobed Handshakes). The output port then raises $\overline{DAV}$ High which causes the input port's DAC to return Low. New data can now move into the Buffer register in preparation for a new cycle. The new cycle begins when the input port(s) signals that it is ready for data by RFD going High (the interlock).

An output port using 3-Wire can be used as the source of data in a communication network that has one source (Talker) and many receivers (Listeners). The following is a description of the operation of such a network from the Talker point of view.

In the 3-Wire Handshake (Figure 3-13), separate signals are used to indicate when the Listeners are ready to receive new data (RFD), and when the data has been accepted (DAC). (In the Interlocked Handshake, the $\overline{ACKIN}$ input indicates both [High = Ready for Data, Low = Data Accepted]). Since two signals are used, many Listeners can be connected simultaneously to a single Talker by wire-ANDing their RFD and DAC outputs. In this way RFD will rise only when all Listeners are ready, and DAC will rise only after all Listeners have received the data.

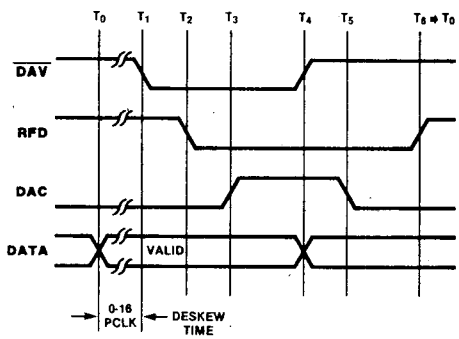


Figure 3-13. 3-Wire Output Handshake Timing Diagram

When $\overline{DAV}$ falls, the Listeners can receive the data at their own individual pace, indicating that they have received the data by bringing RFD Low, and letting their own DAC go High. Since DAC is a wired-AND, the DAC signal to the Talker will indicate when the last Listener has accepted the current data.

The Talker then tells the Listeners that the data is no longer valid ($\overline{DAV}$ goes High). The Listeners then respond by pulling DAC Low and, when they are ready, letting RFD go High. When the last Listener's RFD goes High, the wired-AND tells the Talker that all Listeners are again ready for new data, and the cycle is ready to begin again.

This procedure is in agreement with the IEEE-488 3-Wire Handshake. See the 3-Wire Handshake timing example (Section 3.4.2.2) for a step-by-step analysis of a port configured as double-buffered which uses 3-Wire Handshake.

3.4.4 Bidirectional Port Operation

The bidirectional port is both an input and an output port. That is, it handles data movement in either direction between the CPU and the CIO pins. The direction of data flow is controlled by a single line, the IN/$\overline{OUT}$ line, provided by Port C (see Table 3-1). If the line of Port C used as the direction control is programmed to be an input bit, the port is a "slave" bidirectional port. The direction of data flow is controlled by an external "master," and the IN/$\overline{OUT}$ input controls the port. (If the bit is programmed to be an output bit, the port is a "master" bidirectional port.) In this configuration, the CPU controls the direction of data flow and the IN/$\overline{OUT}$ line is a status line that indicates the direction of data flow. The operation of data transfer is the same for both "master" and "slave" ports--only the mechanism of changing direction is different. The details of the differences will be described in the following paragraphs.

The bidirectional port uses two lines in addition to the IN/$\overline{OUT}$ line for handshaking. One line, the RFD/$\overline{DAV}$ line, has multiple functions depending on the data direction. When acting as an input port, this line is the Ready For Data (RFD) output, like the simple input port. In the output mode, this line is the Data Available ($\overline{DAV}$) output, like the simple output port. The other line is the Acknowledge Input ($\overline{ACKIN}$). Since three of Port C's four lines are required for handshake and direction control lines, only one port may be bidirectional at any one time.

In the bidirectional mode, a port's Input register and Output register are both operational and independent. Even when the IN/$\overline{OUT}$ line is Low (output mode), reading the Port's data register gives the

contents of Input Data register, not the value being output at the time (like a simple output port gives). Reading the port's data register empties the Input register; writing the data register fills the Output register, regardless of the direction of the port at the time of the read or write.

Both double- and single-buffered operations function in the bidirectional mode like simple input and output ports, depending on the direction of data flow.

Since the direction of data flow can be changed at almost any time and because a single Buffer register is multiplexed between the input and output path, any data in the Buffer register is somewhat vulnerable. The user must take precautions that data is not lost.

Only two handshakes are supported by a port in the bidirectional mode: the Interlocked Handshake and the Strobed Handshake. The Pulsed Handshake and the 3-Wire Handshake must not be specified. Like the simple input and output ports, the only difference between the Interlocked and the Strobed handshakes is whether or not the $\overline{ACKIN}$ input influences the operation of the RFD output (input mode) or $\overline{DAV}$ output (output mode). The operation of the bidirectional port will be explained by describing in detail the input operation, input-to-output transition, output operation, and output-to-input transition for a port using the Interlocked Handshake. Differences encountered using the Strobed Handshake will be described as they occur.

NOTE

When in the bidirectional operation condition, the Interrupt on Two Bytes (ITB) bit of the Port Mode Specification register must be set to 0 (zero).

3.4.4.1 Input Operation

The Bidirectional port operating in the input direction operates like a simple input port, except for how IP is cleared and how the Input register is emptied.

On the falling edge of the Acknowledge Input ($\overline{ACKIN}$) the input data is latched in the Buffer register and RFD/$\overline{DAV}$ output is pulled Low, indicating that the data has been accepted. If the Input Data register is empty, the data is transferred to the Input Data register, the Input Register Full (IRF) flag is set to 1, the Buffer register is emptied (if not single-buffered), and IP is set to 1. When the Buffer register becomes empty, the RFD/$\overline{DAV}$ returns High, if $\overline{ACKIN}$ is High. If the Strobed Handshake is specified, it goes High independent of $\overline{ACKIN}$'s state.

In bidirectional input operation, the IP cannot be cleared automatically. It must be cleared by writing to the Port Command and Status Register. Also, the Input register (and Buffer register, if single-buffered) will remain "full" until the Input Data register is read and IP is cleared (in any order).

3.4.4.2 Input to Output Direction Change

The direction can safely be changed at any time except for the period between the falling edge of $\overline{ACKIN}$ and the time that RFD/$\overline{DAV}$ falls, indicating the reception of input data. As long as any input data is in the Buffer register or any input interrupts are pending, the port remains in the input mode internally. As the data is shifted into the Input Data register, IP is set; the port remains an input port until that IP is cleared. While the internal state of the port differs from the IN/$\overline{OUT}$ line, the RFD/$\overline{DAV}$ line is forced High, indicating that no output data is available. If the IN/$\overline{OUT}$ line is returned High during this time, RFD/$\overline{DAV}$ may go Low to indicate that the port is not ready for input data and that a false RFD/$\overline{DAV}$ falling edge may be generated. This problem can be alleviated if the input-to-output transition is not made while the RFD/$\overline{DAV}$ line is Low. When all input interrupts have been recognized, (IP's cleared), the port becomes an output port and operates normally. Until this time, all port data lines remain input (the output drivers will not go active).

If the port becomes an output port and the Output Data register is empty, IP is automatically set just as if the Output register had become empty as a result of an output handshake.

3.4.4.3 Output Operations

A bidirectional port, operating in the output direction, functions like a simple output port

except for the way in which IP is cleared and the Output register is "filled".

The output operation starts by writing a byte to the port's data register. If IP = 0, this fills the Output Data register and clears the Output Register Empty (ORE) flag to 0. If the Buffer register is empty, the data is moved to the Buffer register (and, therefore, to output on the port pins), and, if specified, the optional deskew timer is triggered. If the port is double-buffered (SB = 0), the Output Data register is "emptied" and both ORE and IP are set to 1. If the port is single-buffered (SB = 1), the Output Data register is not "emptied" when the data is moved into the Buffer register. If $\overline{ACKIN}$ is High, the RFD/$\overline{DAV}$ output is pulled Low either immediately, or after the deskew timer times out. If the Strobed Handshake is specified, RFD/$\overline{DAV}$ goes Low independent of the state of $\overline{ACKIN}$. When $\overline{ACKIN}$ goes Low, indicating that the data has been received, the Buffer register is emptied, and the RFD/$\overline{DAV}$ line is forced High. If the port is single-buffered, this also "empties" the Output Data register and causes both IP and ORE to be set to 1.

In bidirectional output operation, IP cannot be cleared automatically. It can only be cleared by writing to the Port's Command and Status register. Also, the Output Data register remains "empty" until both the Output Data register is written and IP is cleared (in any order).

3.4.4.4 Output to Input Direction Change

The change in direction from output to input can safely take place at any time, except for the time between the falling edge of $\overline{ACKIN}$ and the rising edge of RFD/$\overline{DAV}$ (which indicates that the receipt of the data is recognized). As soon as the direction change to input is recognized, the port becomes an input port. The output drivers are forced to high impedence and the Buffer register is emptied (if full) to prepare it for an input data byte. Any data that is in the Buffer register is lost. If the direction change could occur at any time (and possibly destroy Buffer register data), the port should be specified to be single-buffered. In this case, a copy of the data to be output is contained in the Output Data register, and the data is not lost. When the Buffer register is cleared, the port is ready to receive new data immediately and RFD/$\overline{DAV}$ is output High. Although the port is ready to receive data in the Buffer register, it will not be moved to the Input Data register until IP is cleared. This guarantees that no interrupts are missed on direction change.

It is recommended that a "master" bidirectional port only change directions to input when the RFD/$\overline{DAV}$ output is High. This can be done by programming the port to be single-buffered and then changing directions in response to an output handshake interrupt. At this time, the previous output handshake is completed and RFD/$\overline{DAV}$ is High.

If the Input Data register is full when the transition to an input port takes place, an interrupt sequence is initiated just as if the register had become full as a result of an input handshake. (This is delayed until any output IP is cleared by software.)

3.4.4.5 Pattern Match

The operation of the pattern match logic (if enabled) in the bidirectional mode is essentially the same as combining the input and output modes. Each time IP is set, the PMF is updated, based on whether or not the byte transferred into the Input Data register or out of the Output Data register matches the pattern. IP is set and cleared like a normal bidirectional port.

NOTE

The ITB bit must be 0 during bidirectional operation. If the port is programmed to be single-buffered (SB - 1), the Interrupt on Match Only (IMO) bit must not be 1.

3.4.5 REQUEST/$\overline{WAIT}$ Line Operation

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

3.4.5.1 REQUEST Line Operation

Operation of the REQUEST line is dependent on the state of the port's Interrupt on Two Bytes (ITB) control bit. When ITB = 0, the REQUEST line goes active as soon as the CIO is ready for a data transfer. If the port is used for the input, the REQUEST line goes High when the Input Data register is full. If the port is used for output, the REQUEST line goes High when the Output Data register is empty. If ITB = 1, REQUEST goes active only if two bytes can be moved. For input, both Input Data and Buffer registers are full, and for output, both Buffer and Output Data registers are empty. REQUEST stays active as long as a byte is available to be read or written. However, if the port is single-buffered or if the Pattern Match Flag is set, REQUEST goes Low when the data is read or written.

In the bidirectional mode, ITB must be 0. Therefore, the REQUEST line reflects the state of the Input or Output Data register, depending on the type of REQUEST line specified.

The DMA-type transfer is facilitated by the use of the Interrupt on Match Only (IMO) control bit in the Port Mode Specification register. In most DMA transfers, the peripheral does not generate an interrupt request. However, the Interrupt on Match Only (IMO) capability of the CIO allows it to interrupt the CPU when there is a specified byte match, such as an end of transfer flag, etc. Except for the specified pattern match, the CPU is not involved in the transfer; the data movement is consequently accomplished much faster. This REQUEST line/IMO pattern match operation can function in all port operating modes.

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/$\overline{\text{OUT}}$ line is High (input port), the REQUEST line is High when the Output Data register is empty. If IN/$\overline{\text{OUT}}$ is Low (output port), the REQUEST line goes High when the Input Data register is full. In this mode, both the Input and the Output Data registers can be monitored. The RFD/$\overline{\text{DAV}}$ indicates the state of the register in the data path, and the REQUEST line monitors the register not in the data path.

This line can be used to indicate when to change the CIO direction. For example, if the CIO is a "slave" bidirectional port in the input direction (IN/$\overline{\text{OUT}}$ = High), then when the SPECIAL REQUEST line goes Low (indicating that the Output Data register has a byte to be transferred), the master port can use this as the signal to turn the port around so that the byte can be output.

3.4.5.2 $\overline{\text{WAIT}}$ Line Operation

The REQUEST/$\overline{\text{WAIT}}$ line configured in the $\overline{\text{WAIT}}$ mode is useful to synchronize CPU-to-CIO transactions. For example, when the CPU wants the data transfer to be performed as rapidly as possible, it does not want to wait for an interrupt service routine to tell it that the CIO is ready to receive or supply a byte of data. Rather, the CPU attempts a read or write to the CIO. If $\overline{\text{WAIT}}$ is enabled, and the CIO is not ready for a read or write to it, $\overline{\text{WAIT}}$ is pulled Low and the CPU is forced to wait until the CIO is ready (the Input register becomes full or the Output register becomes empty), signified by the $\overline{\text{WAIT}}$ line going High. The CPU is now released from the $\overline{\text{WAIT}}$ pause and can complete the data transaction.

For an input port, $\overline{\text{WAIT}}$ is pulled Low when an attempt is made to read the Input Data register and the port is empty. For an output port, $\overline{\text{WAIT}}$ is pulled Low when an attempt is made to write to the Output Data register that is still full; the data integrity is maintained in the case of the output port--the data is not overwritten. Action is merely suspended until the write can take place.

In the Z8036, $\overline{\text{WAIT}}$ falling is caused by $\overline{\text{DS}}$ falling. In the Z8536, $\overline{\text{WAIT}}$ is synchronized with either $\overline{\text{RD}}$ or $\overline{\text{WD}}$ falling. $\overline{\text{WAIT}}$, however, may be required to be valid at the CPU prior to this. A practical way to use $\overline{\text{WAIT}}$ with the CIO is to have external logic generate a $\overline{\text{WAIT}}$ cycle automatically, which can then be extended as needed by the CIO.

The release of $\overline{\text{WAIT}}$ in both the Z8036 and the Z8536 is synchronized with the PCLK input. Thus for the Z8036 REQUEST/$\overline{\text{WAIT}}$ function, a PCLK input is required. Also, while the Z8036 is asserting $\overline{\text{WAIT}}$, internal status is updated using PCLK (not $\overline{\text{AS}}$ as it normally does), because $\overline{\text{AS}}$ stops as long as the CPU is $\overline{\text{WAIT}}$ed.

3.4.6 Linked Port Operation

Ports A and B can be linked to form a 16-bit port by programming a 1 in the Port Link Control (PLC) bit of the Master Configuration Control register. In this mode, only Port A's Handshake Specification 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. The PLC bit must be set to 1 before the ports are enabled.

Chapter 4
Counter/Timer Operation

4.1 COUNTER/TIMER ARCHITECTURE

The three independent 16-bit counter/timers each 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.

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 1's external I/O lines are provided by the four most-significant bits of Port B. Counter/Timer 2's external I/O lines 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 4-1) 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 non-inverting, and can be read and used with the pattern-recognition logic. They can also include the 1's catcher input.

4.2 COUNTER/TIMER SEQUENCE OF EVENTS

The following discussion assumes that the inputs and outputs are programmed non-inverting.

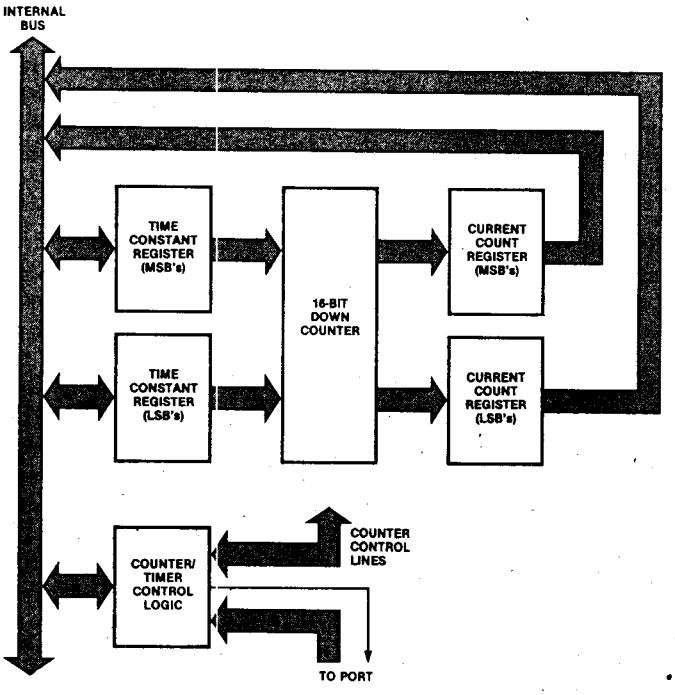


Figure 4-1. Counter/Timer Block Diagram

Table 4-1. Counter/Timer External Access

Function	Counter/ Timer 1	Counter/ Timer 2	Counter/ Timer 3
Counter/Timer Output	Port B 4	Port B 0	Port C 0
Counter Input	Port B 5	Port B 1	Port C 1
Trigger Input	Port B 6	Port B 2	Port C 2
Gate Input	Port B 7	Port B 3	Port C 3

4.2.1 Initializing the Counter/Timer

Before starting a counter/timer sequence:

First, the Counter/Timer Mode Specification register and the Counter/Timer Command and Status register of the desired counter/timer must be initialized. Initialization requires several things to be specified, for example, the external lines to be used, the output duty cycle, and whether the cycle is continuous or single-cycle.

Second, the Time Constant must be specified by writing the desired value to the Time Constant register. The Time Constant register is accessed as two 8-bit registers. The registers are readable as well as writeable, and can be accessed in any order. A 0 in the Time Constant register specifies a Time Constant of 65,536.

Third, if external access is going to be provided, the port to be used must be programmed as a bit port and the necessary bits must be programmed in the proper direction (see Section 3.3).

Finally, the Counter/Timer Enable bit in the Master Configuration Control register is set. This initialization sequence can best be understood by examining the function of the various enable bits. This bit, while cleared to 0, prevents spurious counter/timer operation:

- IPs can not be set.
- Counter/Timers can not be triggered.
- The Read Current Count bit that freezes the value in the Current Count register will be held cleared to 0.
- The Counter/Timer output is forced to 0.

Clearing an enable bit will not clear an existing IP that is set--it will only inhibit the IP from being set again. Clearing the enable bit will clear the Read Counter Control bit, causing the Current Count register to follow the down-counter.

4.2.2 Starting the Counter/Timer

The countdown sequence is initiated when the counter/timer is triggered and the down-counter is loaded with the contents of the Time Constant register. The down-counter is normally loaded on the rising edge of the external trigger input, or by writing a 1 to the Trigger Command Bit (TCB) of the Command and Status register. But, for Counter/Timer 2 only, triggering can occur on the falling edge of Counter/Timer 1's internal output if the counters are linked via the trigger input. Also, Counter/Timer 3 can be triggered by the handshake logic when it is used with the Pulsed Handshake (see Section 3.4.1.3).

The trigger functions as the logical OR of all the potential triggers (see Figure 4-2). Since the trigger function is an OR function, and since it is rising-edge sensitive, any input remaining in its active state will mask off other trigger sources as it stays High.

NOTE

In order to ensure the loading of a Trigger Constant on a particular rising edge of the clocking signal, sufficient setup time must be allowed--the trigger must occur prior to the immediately preceding falling edge of the clocking signal. (The clocking signal equals the count input if in Counter mode or PCLK/2 if in Timer mode.)

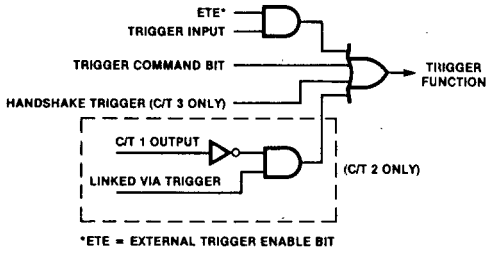


Figure 4-2. Trigger OR-Function Diagram

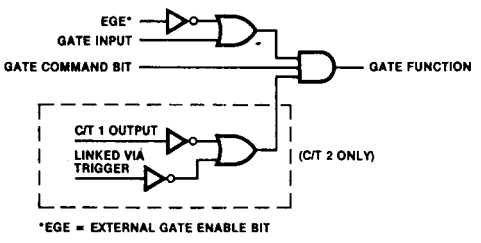


Figure 4-3. Gate AND-Function Diagram

4.2.3 Countdown Sequence

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 1), the down-counter is decremented on the rising edge of the counter/timer's counter input.

Once the down-counter is loaded, the countdown sequence continues toward terminal count as long as all of the counter/timers' hardware and software gate inputs are High. The gate inputs are: the Gate Command bit of the Counter/Timer Command and Status register, and the external gate input if enabled in the External Gate Enable bit of the counter/timer Mode Specification register. Also, for Counter/Timer 2 use only, the counter/timer output (inverted) can be used as a gate if linked via the gate in the Counter/Timer Link Controls bits of the Master Configuration Control register. If any of the gate inputs go Low (0), the countdown halts. It resumes when all gate inputs are 1 again. The gate function does not affect the trigger function.

The gate functions as the logical AND of all the potential gates (see Figure 4-3).

NOTE

In order to ensure the enabling or disabling of the counter/timer on a particular rising edge of the clocking signal, sufficient setup time must be allowed. The gate signal must be valid prior to the immediately preceding falling edge of the clocking signal.

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, a retrigger causes the sequence to start over from the initial load of the time constant.

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 can be guaranteed as valid only if the counter is stopped. The down-counter can be read reliably while it is counting by first writing 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. A read of RCC indicates if the CCR is holding a value, or if it is following the down-counter.

4.2.4 Ending Condition

The Continuous/$\overline{\text{Single Cycle}}$ (C/$\overline{\text{SC}}$) bit in the Mode Specification register controls operation of the down-counter when it reaches terminal count (the count following the count of 1). If C/$\overline{\text{SC}}$ is 0

when a terminal count is reached, the countdown sequence stops. If the C/$\overline{SC}$ bit is 1 each time the count-down 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. **This must be done with care.**

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 request 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 a 1 along with the Interrupt Error (ERR) flag. Errors that occur after the internal flag is set are ignored. ERR is cleared to 0 when the corresponding IP is cleared.

4.2.5 Counter/Timer Output

There are three duty cycles available for the timer/counter output: pulse, one-shot, and square-wave. Figure 4-4 shows the counter/timer timing diagrams. When the Pulse mode is specified, the output goes High for one 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 count-down sequence to begin. When 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 1's, the output is forced Low.

4.2.6 Linked Sequence

Counter/Timers 1 and 2 can be linked internally in three different ways. Counter/Timer 1'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. However, when they are linked, they should be linked before they are enabled. The only restriction is that when Counter/Timer 1 drives Counter/Timers 2's count input, Counter/Timer 2 must be programmed with its external count input disabled (ECE = 0).

The initialization procedure, then, is the same as for individual counter/timers, except that the linking bits need to be appropriately set.

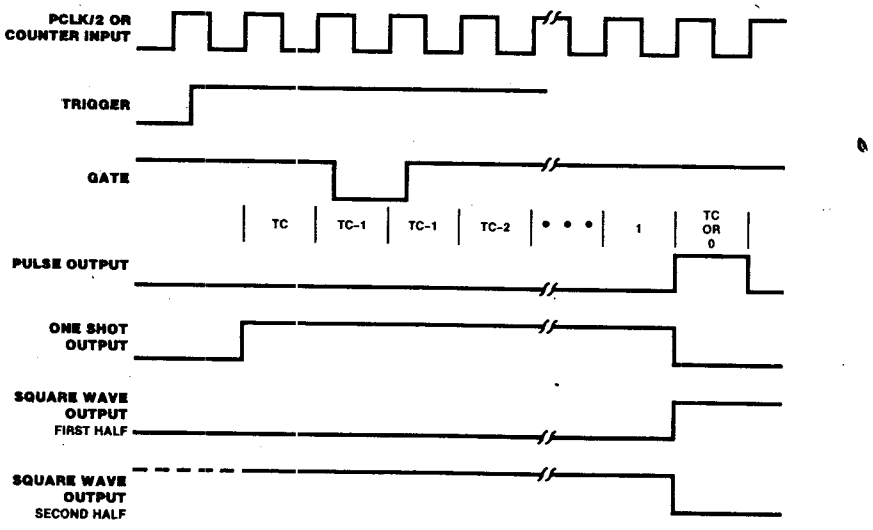


Figure 4-4. Counter/Timer Timing Diagram

Chapter 5
Interrupt Operation

5.1 OVERVIEW

Interrupts are generated whenever CPU intervention is required by a peripheral device. Three examples of interrupt request sources in the CIO are: a pattern match occurring in a bit port; another byte becoming available in an input port with handshake; and a counter/timer reaching its terminal count.

The operation of the Z8036 is compatible with the Z-BUS specification (see Zilog Data Book) and all members of the Z8000 family. The operation of the Z8536 is similar to the Z-BUS specification, which, with a minimum amount of external logic, can be interfaced to a Z80 and its family of peripherals.

Interrupt operation is affected by three things: the external device pins, internal register bit settings, and Interrupt Acknowledge transaction including vector response.

5.2 PRIORITY HANDLING AND THE CIO

The CIO is designed to provide interrupt priority resolution in situations where there may be competing interrupt requestors.

Interrupt priority resolution can be accomplished by using either a separate interrupt controller device or a daisy-chain. The interrupt controller device handles priority resolution and interrupt request generation for all the peripherals in a system. In contrast, the daisy-chain structure uses serial daisy-chain pin connections to establish the priority of the interrupt devices, thus distributing the decision-making among these peripheral devices. The interrupt priority of a device is determined by its position in the daisy-chain.

One of the features of the CIO is that it allows both of the above interrupt schemes to be used in the same CPU environment. The CIO contains all the logic necessary for it to be used in both the Z-BUS and the Z80 BUS daisy-chain interrupt structures. This logic generates interrupts, resolves interrupt priority, inhibits preemption by lower-priority interrupts, and identifies the source of interrupt. The OR-PEV mode pattern recognition logic of the ports (see Section 3.2) allows the CIO to act as an interrupt controller, facilitating the use of devices which do not have the necessary interrupt logic. (See Figure 5-1.)

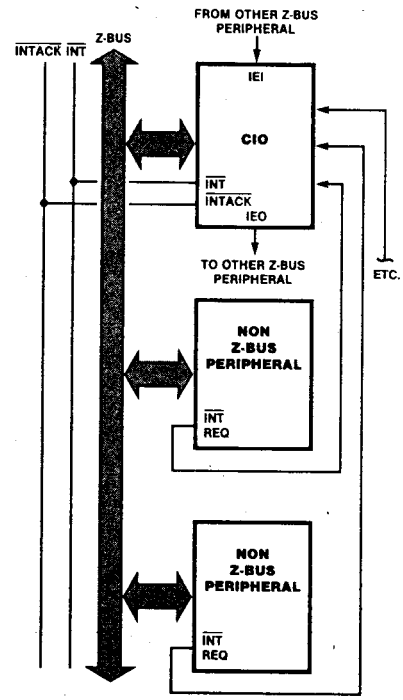


Figure 5-1. The CIO as an Interrupt Controller

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 (highest to lowest): 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 fixed priorities.

The CIO interrupt priority, relative to other components within the system, is determined by the interrupt daisy-chain. Two pins, Interrupt 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 IEO output is connected the IEI input of the next lower-priority device on the daisy-chain. IEO is forced Low to inhibit interrupts from all lower-priority devices. The following discussion assumes that the IEI input is High.

5.3 THE FOUR INTERRUPT LOGIC FUNCTIONS

The CIO has the logic necessary to: generate interrupts, resolve priority when there is more than one interrupt requestor, inhibit preemptive interrupts by the lower-priority requestors, and clearly identify the exact source of interrupt.

5.3.1 Generating the Interrupt Request

Each source interrupt in the CIO contains three bits for the control and status of the interrupt logic: an Interrupt Pending (IP) bit, an Interrupt Under Service (IUS) bit, and an Interrupt Enable (IE) bit. IP is automatically set when an event requiring CPU intervention occurs. (Chapters 3 and 4 describe in detail how a particular IP is normally set and cleared.) The setting of IP results in an Interrupt Request ($\overline{INT}$) output pulled Low if all other conditions are met. (IP can also be set by a command. This is useful when debugging interrupt handler software.)

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

The IUS status bit is set by the CPU as a result of the Interrupt Acknowledge cycle if, at the time of the Interrupt Acknowledge cycle, the corresponding IP is the highest-priority unmasked IP. (The details of setting and clearing IUS are described in Section 5.3.3.) When IUS is 1, it indicates that the corresponding IP has been recognized by the CPU and is being serviced. As long as IUS is set the corresponding IP is masked off and the IEO output is forced Low.

The Master Interrupt Enable (MIE) bit allows all sources of interrupts within the CIO to be disabled without having to individually clear each IE to 0. If MIE is set to 0, all IPs are masked off and no interrupt can be requested or acknowledged.

The IEI input is also involved in the control of interrupt generation. If IEI is Low, it indicates that a higher-priority interrupt is being serviced (that is, a higher-priority IUS is set to 1). An interrupt request can be made only when IEI is High.

In summary, for a device with one source of interrupt, $\overline{INT}$ is pulled Low and an interrupt request is generated only when IP and IE are 1 and IUS is 0, MIE is 1, and IEI input is 1. For a device with many sources of interrupt (for example, the CIO), there is an internal IEI-IEO daisy-chain that determines the internal interrupt priority. The internal IEI for the particular interrupt source must also be a 1.

5.3.2 Priority Resolution

The CPU responds to an interrupt request by generating an Interrupt Acknowledge cycle to determine the source of the interrupt. The first part of the cycle (from Interrupt Acknowledge [$\overline{INTACK}$] falling, until Data Strobe [$\overline{DS}$] [in the Z8036] or Read [$\overline{RD}$] [in the Z8536] goes Low) is used to determine which requestor has the highest priority. More than one device may have all the conditions satisfied for pulling $\overline{INT}$ Low. As soon as the Interrupt Acknowledge begins, all devices that have an unmasked (IE = MIE = 1, IUS = 0) IP set will pull their IEO Low. The Low will ripple down the daisy-chain, disabling all lower devices by forcing their IEIs Low. When the daisy-chain settles, only one source of interrupt will have an unmasked IP with its IEI High--this is the interrupt source being acknowledged.

During the Interrupt Acknowledge cycle, IPs cannot be set, so the daisy-chain has an opportunity to stabilize. To satisfy this restriction in the Z8036, IPs are set only when Address Strobe ($\overline{AS}$) goes Low. In the Z8536, IPs are set by PCLK during State 0. (This is why the $\overline{ACKIN}$ input must be synchronous with PCLK in the Z8536.)

5.3.3 Inhibiting Preemption by Lower-Priority Sources

When $\overline{DS}$ (Z8036) or $\overline{RD}$ (Z8536) falls during an Interrupt Acknowledge cycle, the IUS corresponding to the highest unmasked IP is automatically set to 1. As long as IUS is set, IEO is held Low, prohibiting interrupt requests from lower-priority interrupt sources. This guarantees that an interrupt service routine will not be interrupted to service a lower-priority interrupt. IUS can be reset to 0 only by writing to the corresponding Counter/Timer or Port Command and Status register. It is not cleared automatically. IUS can be cleared before interrupt servicing is complete if lower-priority interrupts wish to be recognized. However, IP must be cleared or a second interrupt request will be generated.

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 interrupts from all lower-priority devices are disabled.

Daisy-chain operation is handled differently between the Z8000 peripherals and the Z80 peripherals--however, they are compatible. (Refer to Interfacing 8500 Peripherals to the Z80, Microcomputer Applications Reference Book, document #00-2145-01). The CIO forces IEO Low when IEI is Low, or when an IUS is 1 (except during an Interrupt Acknowledge cycle with an unmasked IP = 1 when IEO is also forced Low).

The Z80 peripherals (CTC, PIO, DMA, and SIO) normally force IEO Low if IEI is Low, or if either IP or IUS is set. However, they use the Z80 Return from Interrupt Instruction (RETI ED_H - $4D_H$) to automatically clear the highest IUS set. To implement this when an ED is decoded as the first byte of an instruction fetch, Z80 peripherals inhibit IP from affecting the daisy-chain.

Although the daisy-chains are different, during critical times (during an Interrupt Acknowledge or when a RETI instruction is executed), they are the same and are therefore compatible.

5.3.4 Identification of the Highest-Priority Interrupt Request; The Use of Vectors.

As part of the Interrupt Acknowledge cycle, the CIO is capable of responding with an 8-bit interrupt vector that specifies the highest-priority interrupt requestor (see Table 5-1). The vector is output when $\overline{DS}$ (Z8036) or $\overline{RD}$ (Z8536) goes Low and IUS is set. The identification vector is a key item of the Z8000 Family interrupt handling logic. It speeds the information passing and can, if desired, include additional status information identifying the cause of the interrupt as well as the source identification.

The CIO contains three vector registers: one for Port A, one for Port B, and one shared by the three counter/timers. Unique identification information can be placed by the user in the Interrupt Vector register for each interrupt source needed during initialization. The vector output can be modified to include status information to pinpoint the cause of interrupt. A Vector Includes Status (VIS) control bit controls whether or not the vector includes status.

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. All the information obtained by the vector, including status, can thus be obtained with one additional instruction when VIS is set to 0. When MIE = 0,

Table 5-1. Interrupt Vector Encoding if Vector Includes Status

Port Vector Status

OR-Priority Encoded Vector Mode:

D_3	D_2	D_1	
x	x	x	Number of highest-priority bit with a match

All Other Modes:

D_3	D_2	D_1	
ORE	IRF	PMF	Normal
0	0	0	Error

Counter/Timer Status

D_2	D_1	
0	0	Counter/Timer 3
0	1	Counter/Timer 2
1	0	Counter/Timer 1
1	1	Error

reading the vector register returns the unmodified base vector so that it can be verified.

Another register, the Current Vector register, facilitates the use of the CIO in a polled environment. When read, the data returned is the same as the interrupt vector that is output in an Interrupt Acknowledge, based on the highest-priority IP set. If no unmasked IPs are set, the value FF_H is returned. The Current Vector register provides a simple way to poll all IPs in a single read.

The No Vector (NV) control bit of the Master Interrupt Control register, when set to 1, inhibits the outputting of an interrupt vector during an $\overline{INTACK}$ cycle. The NV bit does not affect the setting of the IUS operation. The only thing that the NV does is prevent the vector from being output on the bus.

5.4 Z-BUS INTERRUPT OPERATION

Figure 7-5 displays Interrupt Acknowledge timing. The Z8036 generates an interrupt request by lowering the $\overline{INT}$ line only if:

- Such interrupt requests are enabled (IE is 1, MEI is 1).
- It has an interrupt pending (IP = 1).
- It does not have an interrupt under service (IUS is 0).
- No higher-priority interrupt is being serviced (IEI is 1).

Figure 5-2 shows a typical Z-BUS interrupt arbitration setting.

IEO is not pulled down by the Z8036 at this time; IEO continues to follow IEI until an Interrupt Acknowledge transaction occurs.

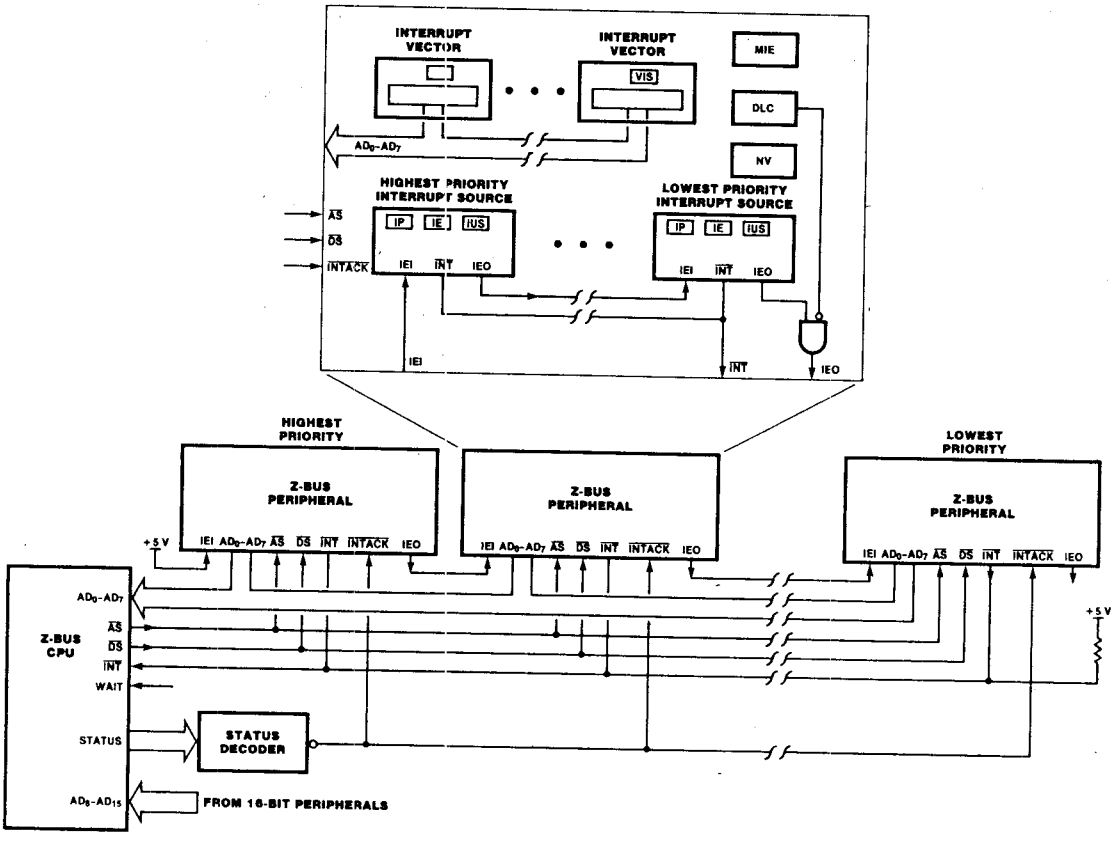


Figure 5-2. Z-BUS Interrupt Arbitration

Some time after $\overline{INT}$ has been pulled Low, the CPU initiates an Interrupt Acknowledge transaction. Between the rising edge of $\overline{AS}$ and the falling edge of $\overline{DS}$, the IEI/IEO daisy-chain settles. Any Z-BUS peripheral with one of its interrupts pending (IP is 1) or one of its interrupts under service (IUS is 1) holds its IEO line Low; all others make IEO follow IEI.

When $\overline{DS}$ falls, only the highest-priority interrupt source with a pending interrupt (IP is 1) has its IEI input High, its IE bit set to 1, and its IUS bit set to 0. This is the interrupt source being acknowledged, and at this time it sets its IUS bit to 1. If its NV bit is 0, the Z8036 identifies itself by placing its interrupt vector from the corresponding interrupt vector register onto address/data lines AD_0-AD_7. If NV is 1, the Z8036 AD_0-AD_7 lines remain floating, allowing external logic to supply a vector. (All Z-BUS interrupts require a vector to identify the requesting device.)

If the corresponding Z8036 VIS is 1, the vector also contains status information, (see Table 5-1) which further identifies the source of the interrupt within the Z8036. IF VIS is 0, the vector held in the interrupt vector register is output without status included (base vector).

5.5 NON-Z-BUS INTERRUPT OPERATION

Figure 8-5 shows the non-Z-BUS Interrupt Acknowledge cycle timing. Figure 5-3 displays a Z80 system using Z8500 and Z80 devices. (Section 8.5 describes generation of $\overline{INTACK}$ from Z80 signals.) Figure 5-4 shows the external logic required for interfacing Z80 to Z8500 peripherals.

For the Z8536, the IP bit is not set while the device is in State 1 (Refer to Section 2.3 for a description of state conditions.) Therefore, to minimize interrupt latency, **the Z8536 should not be left in State 1.**

The Z8536 generates an interrupt request by lowering the $\overline{INT}$ line only if:

- Such interrupt requests are enabled (IE is 1, MIE is 1).
- It has an interrupt pending (IP = 1).
- It does not have an interrupt under service (IUS is 0).
- No higher-priority interrupt is being serviced (IEI is 1).

IEO is not pulled down by the Z8536 at this time; IEO continues to follow IEI until an Interrupt Acknowledge transaction occurs.

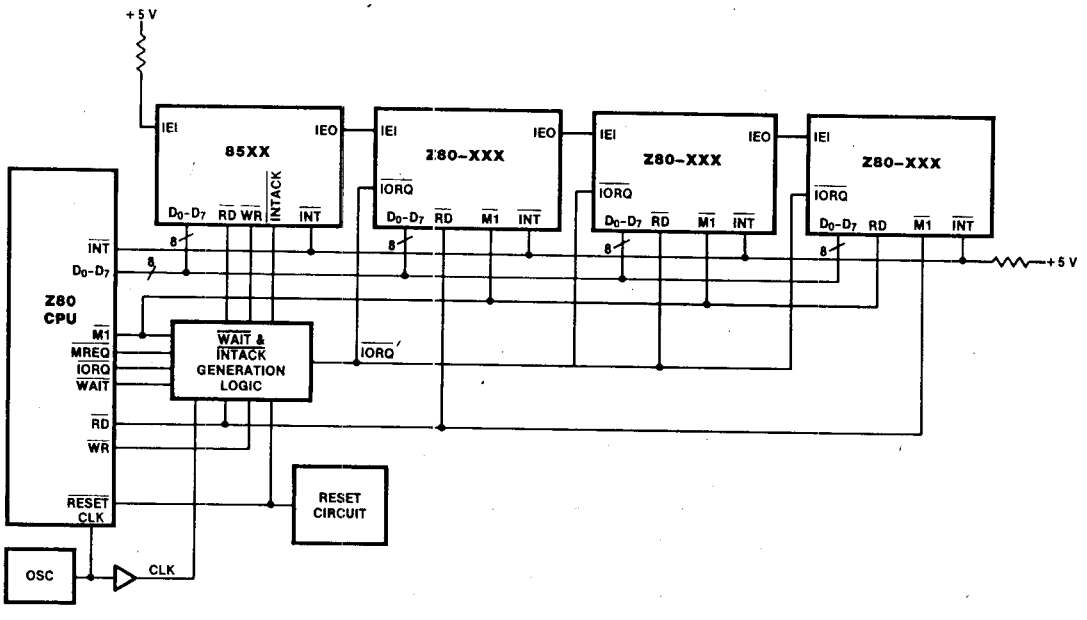


Figure 5-3. Non Z-Bus Interrupt Arbitration

Some time after $\overline{INT}$ has been pulled Low, the CPU initiates an Interrupt Acknowledge transaction. Between the falling edge of $\overline{INTACK}$ and the falling edge of $\overline{RD}$, the IEI/IEO daisy-chain settles. Any peripheral with one of its interrupts pending (IP is 1) or one of its interrupts under service (IUS is 1) holds its IEO line Low; all other conditions make IEO follow IEI.

When $\overline{RD}$ falls, only the highest-priority interrupt source with a pending interrupt (IP is 1) has its IEI input High, its IE bit set to 1, and its IUS bit set to 0. This is the interrupt source being acknowledged, and at this point it sets its IUS bit to 1. If its NV bit is 0, the Z8536 identifies itself by placing its interrupt vector from the corresponding interrupt vector register on data lines D_0-D_7. If NV is 1, the Z8536's D_0-D_7 lines remain floating, allowing external logic to supply a vector.

If the Z8536 VIS is 1, the vector also contains status information (see Table 5-1) which further identifies the source of the interrupt within the Z8536. If VIS is 0, the vector held in the interrupt vector is output without status included (base vector). The bit codes are in Section 2.9.1.

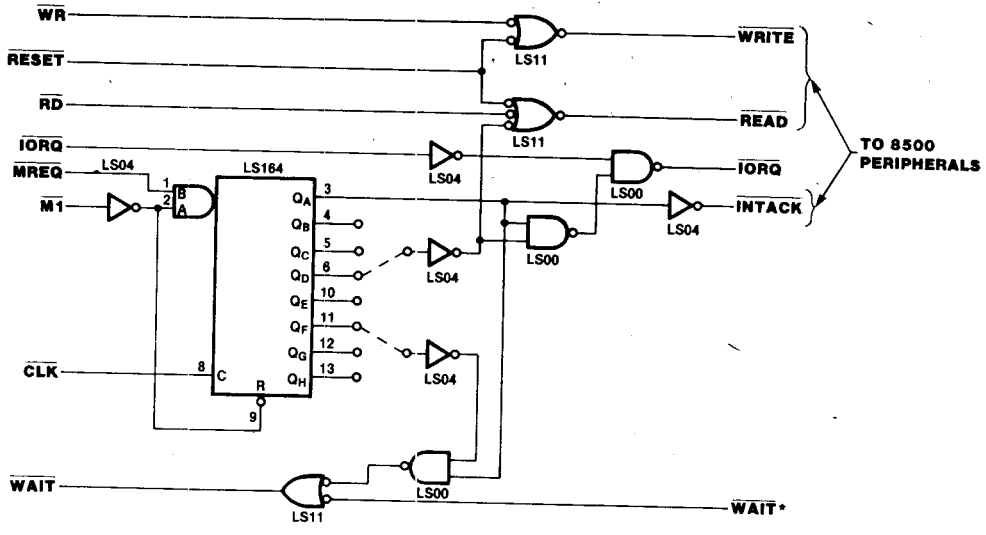


Figure 5-4. $\overline{WAIT}$ and $\overline{INTACK}$ Generation Logic

Chapter 6
Z-CIO/CIO Initialization

6.1 INTRODUCTION

This chapter discusses reset, initialization, and programming for both the Z8036 Z-CIO and the Z8536 CIO.

The normal sequence for initializing this device is simple and straightforward:

1. Reset the device
2. Clear RESET
3. Program the various functions
4. Set the appropriate Enable bits

NOTE

Reset operations are substantially different in the Z8036 and the Z8536 and are therefore discussed separately.

6.2 Z8036 (Z-CIO) RESET

The Z8036 is reset by forcing $\overline{AS}$ and $\overline{DS}$ Low simultaneously (normally an illegal operation), or by writing a 1 to the Reset bit (D_0) in the Master Interrupt Control register. After the Z8036 is reset, a Read or Write to the Reset bit is the only responsive command available--writes to all other bits (in all other registers) are ignored and all reads return 0s. In this state, all control bits are forced to 0 (see Chapter 2 for details), all port I/O lines are high-impedance, the interrupt pin is not asserted, and the Interrupt Enable Out (IEO) pin follows the Interrupt Enable In (IEI) pin. Only after clearing the Reset bit (by writing a 0 to it) can the other command bits be programmed.

6.3 Z8536 (CIO) RESET

The Z8536 is reset by forcing $\overline{RD}$ and $\overline{WR}$ Low simultaneously (normally an illegal condition), or by writing a 1 to the Reset bit (D_0) in the Master Interrupt Control register. RESET disables all functions except a read or write to the Reset bit. In the reset state, the pointer always points to the Master Interrupt Control register (see the state diagram shown in Figure 6-1). Writes to all other bits are ignored, and all reads return 01_H. In this state, all control bits are forced to 0 (see Chapter 2 for details), all port I/O lines are high-impedance, the interrupt pin is not asserted, and the IEO pin follows the IEI pin. The other command bits can be programmed only after clearing the Reset bit by writing a 0 to it.

Even if the state of the Z8536 is not known, the following sequence will reset it and put it in State 0.

```
IN   A, (CIOCTL)  ;INSURES STATE 0 OR RESET STATE
LD   A, 0
OUT  (CIOCTL), A  ;WRITE POINTER OR CLEAR RESET
IN   A, (CIOCTL)  ;STATE 0
LD   A,0          ;REG 0--MASTER INTERRUPT CONTROL
OUT  (CIOCTL), A  ;WRITE POINTER
LD   A,1
OUT  (CIOCTL), A  ;WRITE RESET
LD   A,0
OUT  (CIOCTL), A  ;CLEAR RESET
```

6.4 ENABLE BITS OPERATION

As the different functions of the device are being initially programmed, it is possible for erroneous interrupt requests to be generated, or for an illegal combination of modes to be temporarily specified. To alleviate this problem without imposing severe restrictions on the sequence of events required to initialize the device, five internal enable control bits are provided: Port A Enable, Port B Enable, Counter/Timer 1 Enable, Counter/Timer 2 Enable, and one enable shared by Counter/Timer 3 and Port C. While these bits are

cleared to 0, the corresponding logic sections are in an initialization mode. All of the registers can be read and written, but the normal operation of the sections is inhibited. The Port A and Port B Enables, when cleared to 0, force their respective I/O lines into a high-impedance state, hold the 1's catchers in a reset condition, inhibit REQUEST/$\overline{\text{WAIT}}$ generation, and prevent the setting of their Interrupt Pending (IP) bits (the states of IP and Interrupt Under Service (IUS) are not affected). Additionally, output data can be written (the first data output is valid when the output drivers go active), but the data direction for these bits must be properly specified before the data is written. The Port C Enable operates in the same way, and, until set to 1, the handshake logic for Ports A and B is forced into an idle state. The Counter/Timer Enables, when set to 0, terminate any countdown sequence in progress, inhibit the counter/timer from being triggered, and force the counter output to 0. While the enable is 0, the Read Counter Control (RCC) bit in the Counter/Timer Command and Status register is forced to 0. Independent enable bits are provided for the different sections of the device so that the individual sections can be reconfigured without disturbing the status of the unchanged sections. By using these enable bits, the device can be initialized in any sequence as long as the desired configuration for a section is specified before its enable bit is set to 1. When ports or counter/timers are to be linked, the bits which specify linking must be programmed before the functions are enabled. In this case two writes are required to the Master Configuration Control register.

6.5 PROGRAMMING

Programming the CIO entails loading control registers with bits to implement the desired operation. As discussed above, individual enable bits are provided for the various major blocks so that erroneous operations do not occur while the port is being initialized. Before the ports are enabled: IPs cannot be set, REQUEST and $\overline{\text{WAIT}}$ cannot be asserted, and all outputs remain high-impedance; the handshake lines are ignored until Port C is enabled; and the counter/timers cannot be triggered until their enable bits are set.

6.5.1 Programming the Z8036

Programming the Z8036 is simple, because every register is directly addressable--a key advantage of the multiplex Address/Data bus.

The Z8036 allows two schemes for register addressing. Both schemes use only six of the eight bits of the Address/Data bus. The scheme used is determined by the Right Justify Address (RJA) bit in the Master Interrupt Control register. When RJA equals 0, Address bus bits 0 and 7 are ignored, and bits 1 through 6 are decoded for the register address (A_0 is derived from AD_1). When RJA equals 1, bits 0 through 5 are decoded for the register address (A_0 is derived from AD_0).

6.5.2 Programming the Z8536

The Data Registers of Ports A, B, and C are directly addressed by pins A_0 and A_1, as shown in Table 6-1.

Table 6-1. Z8536 Data Register Addressing

A_1	A_0	Register
0	0	Port C Data Register
0	1	Port B Data Register
1	0	Port A Data Register
1	1	Control Registers

All other internal registers are accessed by the following two-step sequence (with pins $A_0 = A_1 = 1$). 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.

In the Z8536, an internal state machine determines if access (with pins $A_0 = A_1 = 1$) is to the Pointer register or to an internal control register (See Figure 6-1). Following any control read operation the state machine is in State 0, and 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 Z8536 is in State 1, the next control access is to the register pointed to, which returns the state machine to State 0. Note that when in State 1, many internal operations are suspended--no IPs are set and internal status is frozen. Therefore to minimize interrupt latency and to allow continuous status updates, **the Z8536 should NOT be left in State 1.**

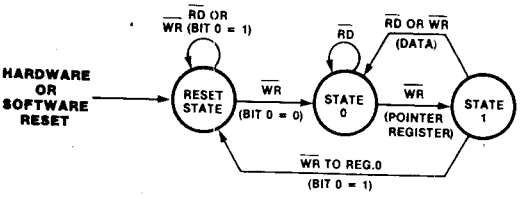


Figure 6-1. Z8536 State Machine Operation

Chapter 7
Z8036 (Z-CIO) Interfacing

7.1 INTRODUCTION

This section provides information on pin functions and assignments and functional timing diagrams for the Z8036 Z-CIO.

7.2 FEATURES

The following features of the Z8036 are not obvious without reference to the ac timing diagrams in the Z8036 Product Specification, document #00-2014-A0.

- The Address Strobe ($\overline{AS}$) input functions as the clock of the Z8036. If the $\overline{AS}$ stops, then data does not get clocked in or through the device, IPs are not set, etc. Care should be taken in the design of the system to ensure that $\overline{AS}$ to the Z8036 is not blocked.
- The assertion of REQUEST is synchronous with PCLK.
- The release of $\overline{WAIT}$ is sychronous with PCLK.
- PCLK is only used with the counter/timers (in timer mode), the deskew timers, and the REQUEST/$\overline{WAIT}$ logic. If these functions are not used, the PCLK input can be held Low.

7.3 PIN FUNCTIONS AND ASSIGNMENTS

The Z8036 is configured for Z-BUS interface controls and timing. The pin functions and assignments are shown in Figures 7-1 and 7-2. Section 7.4 is a description of the pin functions for the Z8036.

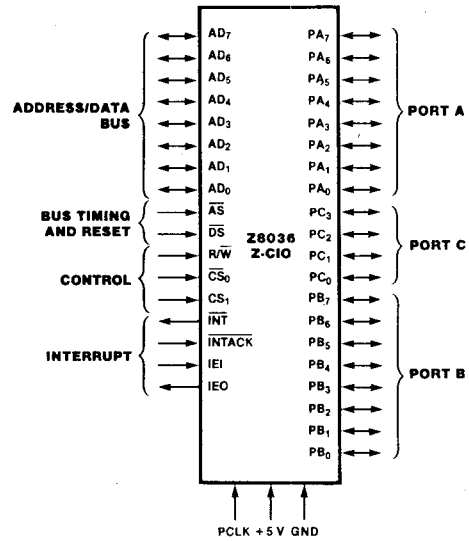


Figure 7-1. Z8036 (Z-CIO) Pin Functions

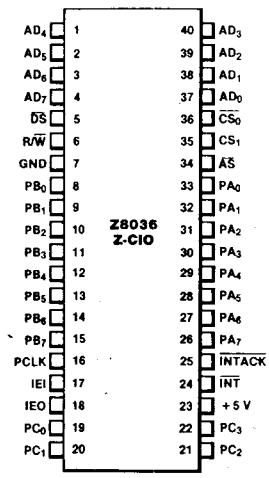


Figure 7-2. Z8036 (Z-CIO) Pin Assignments

7.4 PIN DESCRIPTIONS

AD_0-AD_7. Z-BUS Address/Data lines (bidirectional/3-state). These multiplexed Address/Data

lines are used for transfers between the CPU and Z-CIO.

$\overline{AS}$*. Address Strobe (input, active Low). Addresses, $\overline{INTACK}$, and $\overline{CS_0}$ are sampled while $\overline{AS}$ is Low.

$\overline{CS_0}$ and CS_1. Chip Select 0 (input, active Low) and Chip Select 1 (input, active High). $\overline{CS_0}$ and CS_1 must be Low and High, respectively, in order to select a device. $\overline{CS_0}$ is latched by $\overline{AS}$.

$\overline{DS}$. Data Strobe (input, active Low). $\overline{DS}$ provides timing for the transfer of data into or out of the Z8036.

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 either:

(1) the CPU is not servicing an interrupt from the CIO, or

(2) during an Interrupt Acknowledge Cycle, the CIO is not requesting an interrupt.

IEO is connected to the next lower-priority device's IEI input and thus inhibits interrupts from lower-priority devices.

$\overline{INT}$. Interrupt Request (output, open-drain, active Low). This signal is pulled Low when the Z8036 requests an interrupt.

$\overline{INTACK}$. Interrupt Acknowledge (input, active Low). This signal indicates to the Z8036 that an Interrupt Acknowledge cycle is in progress. $\overline{INTACK}$ is sampled while $\overline{AS}$ is Low.

PA_0-PA_7. Port A I/O lines (bidirectional, 3-state, or open-drain). These eight I/O lines transfer information between the Z8036's Port A and external devices.

PB_0-PB_7. Port B I/O lines (bidirectional, 3-state, or open-drain). These eight I/O lines transfer information between the Z8036's Port B and external devices. They may also be used to provide external access to Counter/Timers 1 and 2.

PC_0-PC_3. Port C I/O lines (bidirectional, 3-state, or open-drain). These four I/O lines are used to provide handshake, $\overline{WAIT}$, and REQUEST lines for Ports A and B; to provide external access to Counter/Timer 3; or to access Port C of the Z8036.

PCLK. (input, TTL-compatible). This is a peripheral clock that may be, but is not necessarily, the CPU clock. It is used with timers and REQUEST/$\overline{WAIT}$ logic.

R/$\overline{W}$. Read/Write (input). R/$\overline{W}$ indicates that the CPU is reading from (High) or writing to (Low) the Z8036.

* When AS and DS are detected Low at the same time (normally an illegal condition), the Z-CIO is reset.

7.5 Z8036 (Z-CIO) READ CYCLE TIMING

The CPU places an address on the Address/Data bus. The most-significant bits and status information are combined and decoded by external logic to provide two Chip Selects ($\overline{CS_0}$ and CS_1). Six bits of the least-significant byte of the address are latched within the Z8036 and used to specify a Z8036 register. The data from the register specified is strobed onto the Address/Data bus when the CPU issues a $\overline{DS}$. If the register indicated by the address does not exist, the Z8036 remains high-impedance.

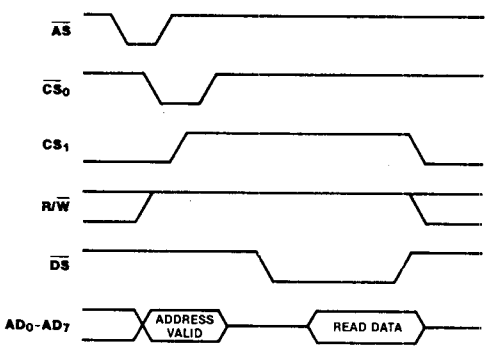


Figure 7-3. Z8036 (Z-CIO) Read Cycle Timing

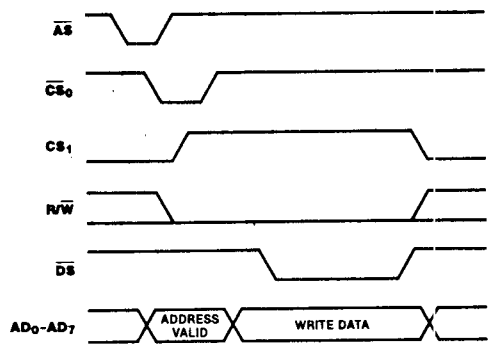


Figure 7-4. Z8036 (Z-CIO) Write Cycle Timing

7.6 Z8036 (Z-CIO) WRITE CYCLE TIMING

The CPU places an address on the Address/Data bus. The most-significant bits and status information are combined and decoded by external logic to provide two Chip Selects ($\overline{CS}_0$ and CS_1). Six bits of the least-significant byte of the address are latched within the Z8036 and used to specify a Z8036 register. The CPU places the data on the Address/Data bus and strobes it into the Z8036 register by issuing a $\overline{DS}$.

7.7 Z8036 (Z-CIO) INTERRUPT ACKNOWLEDGE TIMING

When one of the IP bits in the Z8036 goes High and interrupts are enabled, the Z8036 pulls its $\overline{INT}$ output line Low, requesting an interrupt. The CPU responds with an Interrupt Acknowledge cycle. When $\overline{INTACK}$ goes Low with IP set, the Z8036 pulls its IEO Low, disabling all lower-priority devices on the daisy-chain. The CPU reads the Z8036 interrupt vector by issuing a Low $\overline{DS}$, thereby strobing the interrupt vector onto the Address/Data bus. The IUS that corresponds to the IP is also set, which causes IEO to remain Low.

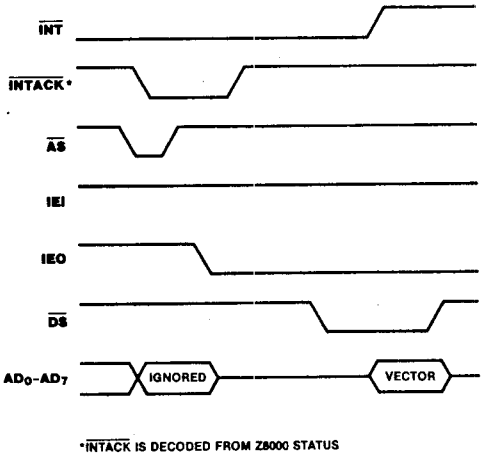


Figure 7-5. Z8036 (Z-CIO) Interrupt Acknowledge Timing

Chapter 8
Z8536 (CIO) Interfacing

8.1 INTRODUCTION

This section provides information on pin functions and assignments and functional timing diagrams for the Z8536.

8.2 FEATURES

The following features of the Z8536 are not obvious without reference to the ac timing diagrams in the Z8536 Product Specification, document #00-2021-A0.

- The state machine conventions relating to programming and register addresssing (see Section 6.5.2) must be followed.
- PCLK can be asynchronous with respect to the CPU--it does not have to be the same as the CPU. However, a minimum of three PCLK cycles must occur between two successive accesses of the Z8536 (that is, between the end of the first access and the beginning of the second access).
- The $\overline{\text{INTACK}}$ input is synchronous, that is, $\overline{\text{INTACK}}$ and PCLK have a relationship that must be maintained.
- The assertion of REQUEST is synchronous with PCLK.
- The release of $\overline{\text{WAIT}}$ is synchronous with PCLK.

8.3 PIN FUNCTIONS AND ASSIGNMENTS

The Z8536 is configured for general microcomputer interface controls and timing. The pin functions and assignments are shown in Figures 8-1 and 8-2. Section 8.4 is a description of the pin functions for the Z8536 CIO.

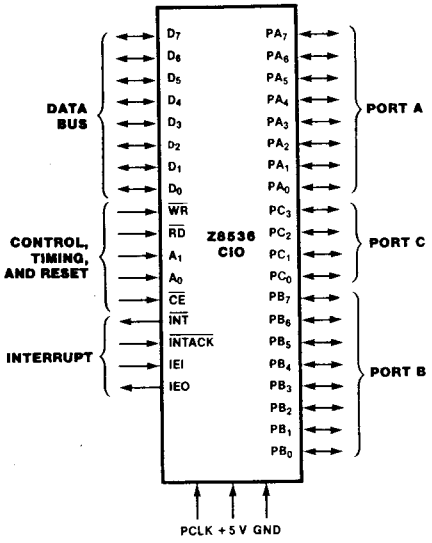


Figure 8-1. Z8536 (CIO) Pin Functions

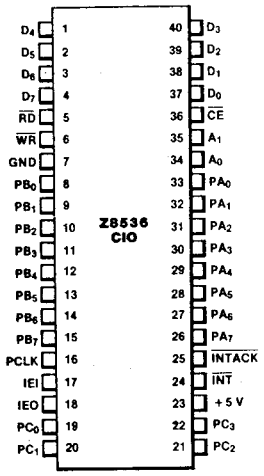


Figure 8-2. Z8536 (CIO) Pin Assignments

8.4 PIN DESCRIPTIONS

A_0-A_1. 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.

$\overline{CE}$. Chip Enable (input, active Low). A Low level on this input enables the Z8536 to be read from or written to.

D_0-D_7. 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 either:

(1) the CPU is not serving an interrupt from the CIO, or

(2) during an Interrupt Acknowledge cycle, the CIO is not requesting an interrupt.

IEO is connected to the next lower-priority device's IEI input and thus inhibits interrupts from lower-priority devices.

$\overline{INT}$. Interrupt Request (output, open-drain, active Low). This signal is pulled Low when the Z8536 requests an interrupt.

$\overline{INTACK}$. Interrupt Acknowledge (input, active Low). This input indicates to the Z8536 that an Interrupt Acknowledge cycle is in progress. $\overline{INTACK}$ must be synchronized to PCLK, and it must be stable throughout the Interrupt Acknowledge cycle.

PA_0-PA_7. 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.

PB_0-PB_7. Port B I/O lines (bidirectional, 3-state, or open-drain). These eight I/O lines transfer information between the Z8536's Port B and external devices. They may also be used to provide external access to Counter/Timers 1 and 2.

PC_0-PC_3. Port C I/O lines (bidirectional, 3-state, or open-drain). These four I/O lines are used to provide handshake, $\overline{WAIT}$, and REQUEST lines for Ports A and B; external access to Counter/Timer 3; or access to the Z8536'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.)

$\overline{RD}$*. 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 Z8536 is the highest-priority device requesting an interrupt.

$\overline{WR}$*. Write (input, active Low). This signal indicates a CPU write to the Z8536.

* When $\overline{RD}$ and $\overline{WR}$ are detected Low at the same time (normally an illegal condition), the Z8536 is reset.

8.5 Z8536 (CIO) READ CYCLE TIMING

At the beginning of a read cycle, the CPU places an address on the Address bus. Bits A_0 and A_1 specify a Z8536 register; the remaining address bits and status information are combined and decoded to generate a Chip Enable ($\overline{CE}$) signal that selects the Z8536. When Read ($\overline{RD}$) goes Low, data from the specified register is gated onto the Data bus.

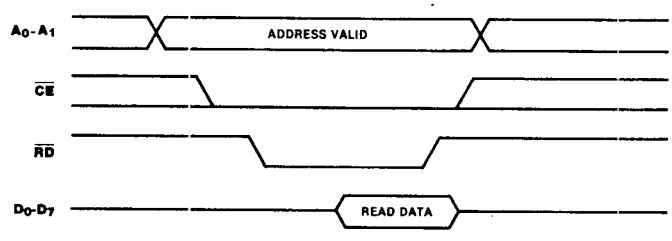


Figure 8-3. Z8536 (CIO) Read Cycle Timing

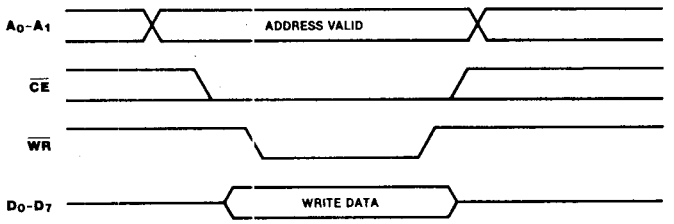


Figure 8-4. Z8536 (CIO) Write Cycle Timing

8.6 Z8536 (CIO) WRITE CYCLE TIMING

At the beginning of a write cycle, the CPU places an address on the Data bus. Bits A_0 and A_1 specify a Z8536 register; the remaining address bits and status information are combined and decoded to generate a Chip Enable ($\overline{CE}$) signal that selects the Z8536. When $\overline{WR}$ goes Low, data placed on the bus by the CPU is strobed into the specified Z8536 register.

8.7 Z8536 (CIO) INTERRUPT ACKNOWLEDGE TIMING

The Z8536 pulls its Interrupt Request ($\overline{INT}$) line Low, requesting interrupt service from the CPU, if an Interrupt Pending (IP) bit is set and interrupts are enabled. The CPU responds with an Interrupt Pending (IP) bit is set and interrupts are enabled. The CPU responds with an Interrupt Acknowledge cycle. When Interrupt Acknowledge ($\overline{INTACK}$) goes active and the IP is set, the Z8536 forces Interrupt Enable Out (IEO) Low, disabling all lower priority devices in the interrupt daisy-chain. If the CIO is the highest priority device requesting service (IEI is High), it places its interrupt vector on the Data bus and sets the Interrupt Under Service (IUS) bit when Read ($\overline{RD}$) goes Low.

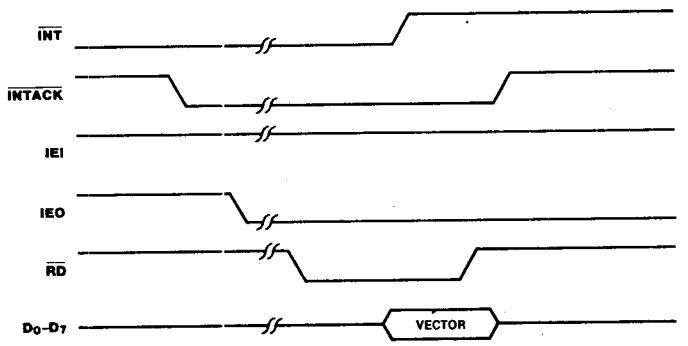


Figure 8-5. Z8536 (CIO) Interrupt Acknowledge Timing

Appendix A

CIO MNEMONICS

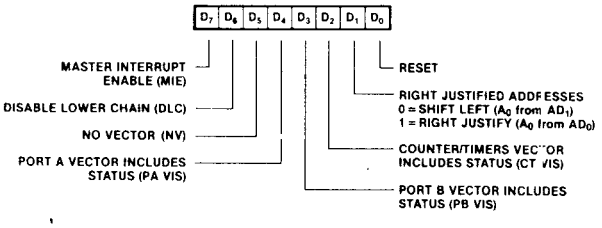
$\overline{\text{ACKIN}}$	Acknowledge Input
$\overline{\text{AS}}$	Address Strobe
C/$\overline{\text{SC}}$	Continuous/$\overline{\text{Single Cycle}}$
CIO	Counter/Timer, Parallel Input/Output Unit
CIP	Count In Progress
CInE	Counter/Timer n Enable
CT3E and PCE	Counter/Timer 3 and Port C Enable
CT VIS	Counter/Timer Vector Includes Status
DAC	Data Accepted
$\overline{\text{DAV}}$	Data Available
DCS	Duty Cycle Selects
DD	Data Direction register
DLC	Disable Lower Chain
DMA	Direct Memory Access
DPP	Data Path Polarity register
$\overline{\text{DS}}$	Data Strobe
DTE	Deskew Timer Enable
DTE/LPM	Deskew Timer Enable/Latch on Pattern Match
ECE	External Count Enable
EGE	External Gate Enable
EOE	External Output Enable
ERR	Interrupt Error
ETE	External Trigger Enable
GCB	Gate Command Bit
HTS	Handshake Type Specification bits
IE	Interrupt Enable
IEI	Interrupt Enable In
IEO	Interrupt Enable Out
IMO	Interrupt on Match Only
$\overline{\text{INT}}$	Interrupt
$\overline{\text{INTACK}}$	Interrupt Acknowledge
I/O	Input/Output
IP	Interrupt Pending
IRF	Input Register Full
ITB	Interrupt on Two Bytes
IUS	Interrupt Under Service
LC	Counter/Timer Link Controls
LPM	Latch on Pattern Match
LSB	least-significant bit
MIE	Master Interrupt Enable
MSB	most-significant bit
NV	No Vector
OR-PEV	OR-Priority Encoded Vector
ORE	Output Register Empty
PA	Port A
PA VIS	Port A Vector Includes Status
PAE	Port A Enable
PB	Port B
PB VIS	Port B Vector Includes Status
PBE	Port B Enable
PC	Port C
PCE	Port C Enable
PCLK	Peripheral Clock
PLK	Port Link Control
PM	Pattern Mask registers
PMF	Pattern Match Flag
PMS	Pattern Mode Specification bits
PP	Pattern Polarity registers
PT	Pattern Transition registers
PTS	Port Type Selects
RCC	Read Counter Control
REB	Retrigger Enable Bit
RFD	Ready For Data
RJA	Right Justified Address
R/$\overline{\text{W}}$	Read/Write
RWS	REQUEST/$\overline{\text{WAIT}}$ Specification bits
SB	Single-Buffered
SIO	Special Input/Output
TCB	Trigger Command Bit
VIS	Vector Includes Status
1's Catcher	One's Catcher
3-Wire	3-Wire Handshake (IEEE-488)

Appendix B

Registers

Master Interrupt Control Register

Address: 000000
(Read/Write)



Master Configuration Control Register

Address: 000001
(Read/Write)

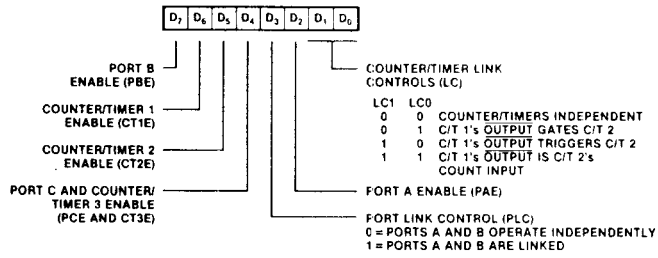
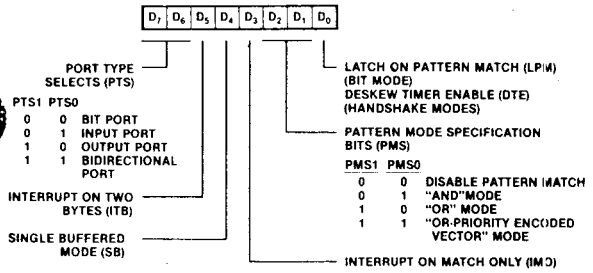


Figure B-1. Master Control Registers

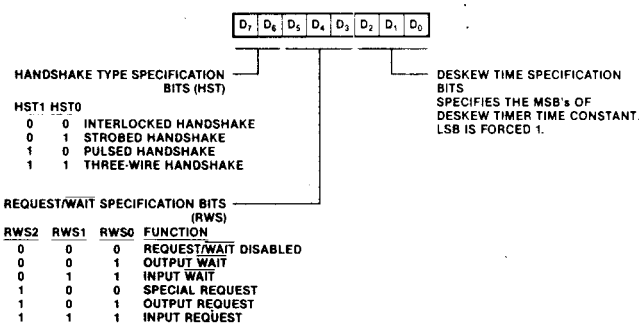
Port Mode Specification Registers

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)

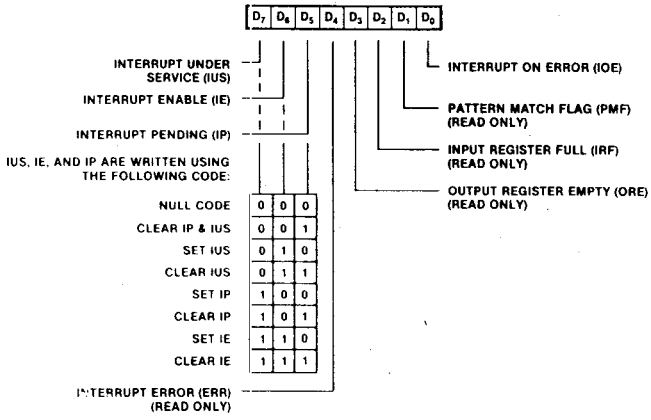
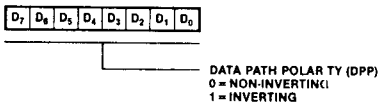


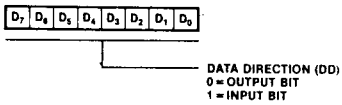
Figure B-2. Port Specification Registers

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
0C0111 Port C (4 LSBs only)
(Read/Write)

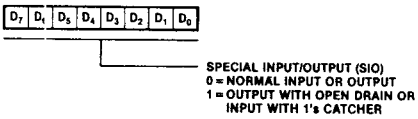
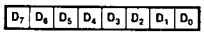


Figure B-3. Bit Path Definition Registers

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



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

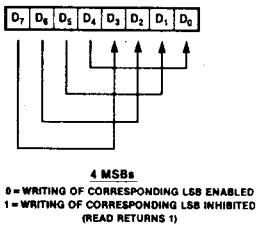
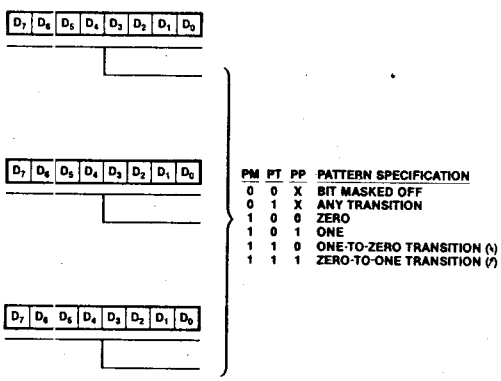


Figure B-4. Port Data Registers

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)

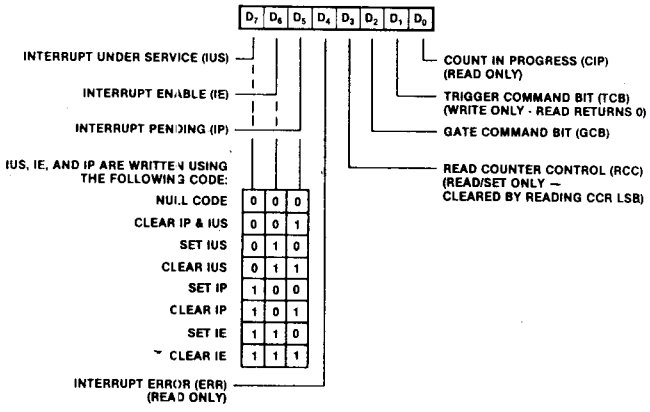


PM	PT	PP	PATTERN SPECIFICATION
0	0	X	BIT MASKED OFF
0	1	X	ANY TRANSITION
1	0	0	ZERO
1	0	1	ONE
1	1	0	ONE-TO-ZERO TRANSITION (\)
1	1	1	ZERO-TO-ONE TRANSITION (/)

Figure B-5. Pattern Definition Registers

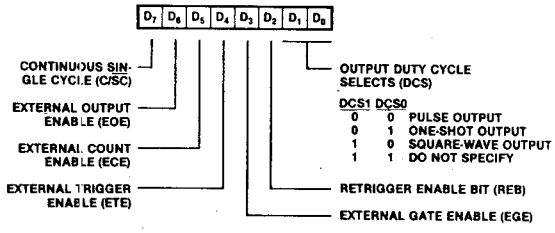
Counter/Timer Command and Status Registers

Addresses: 001010 Counter/Timer 1
001011 Counter/Timer 2
001100 Counter/Timer 3
(Read/Partial Write)



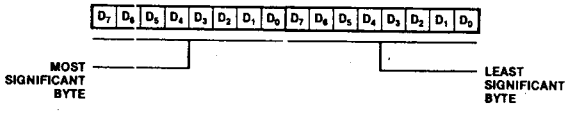
Counter/Timer Mode Specification Registers

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



Counter/Timer Current Count Registers

Addresses: 010000 Counter/Timer 1's MSB
010001 Counter/Timer 1'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: 010110 Counter/Timer 1's MSB
010111 Counter/Timer 1'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)

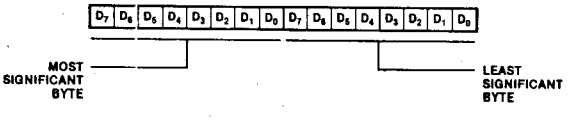
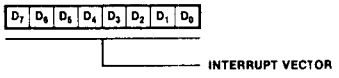


Figure B-6. Counter/Timer Registers

Registers (Continued)

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



PORT VECTOR STATUS

PRIORITY ENCODED VECTOR MODE:

D3	D2	D1	
x	x	x	NUMBER OF HIGHEST PRIORITY BIT WITH A MATCH

ALL OTHER MODES:

D3	D2	D1	
ORE	IRF	PMF	NORMAL
0	0	0	ERROR

COUNTER/TIMER STATUS

D2	D1	
0	0	C/T 3
0	1	C/T 2
1	0	C/T 1
1	1	ERROR

Current Vector Register
Address: 011111
(Read Only)

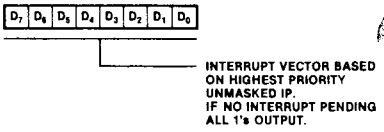


Figure B-7. Interrupt Vector Registers

Register Address Summary

Main Control Registers

Address	Register Name
000000	Master Interrupt Control
000001	Master Configuration Control
000010	Port A's Interrupt Vector
000011	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 I/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 1'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
001110	Port B's Data
001111	Port C's Data

Counter/Timer Related Registers

Address	Register Name
010000	Counter/Timer 1's Current Count-MSBs
010001	Counter/Timer 1'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 1's Time Constant-MSBs
010111	Counter/Timer 1'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 1'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 Data 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
101100	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

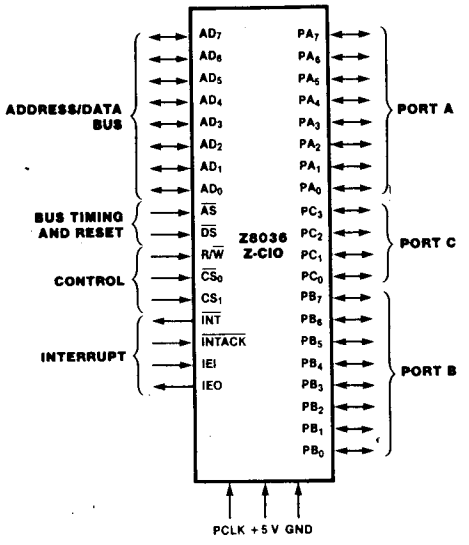


Figure 1. Pin Functions

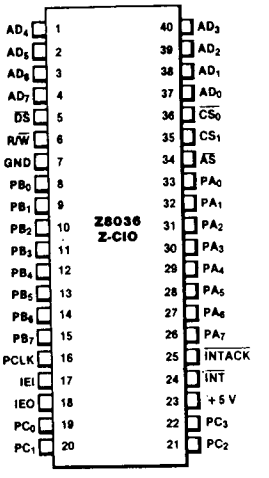


Figure 2. Pin Assignments

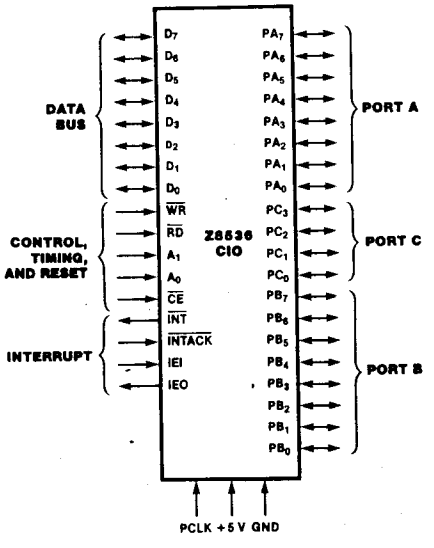


Figure 1. Pin Functions

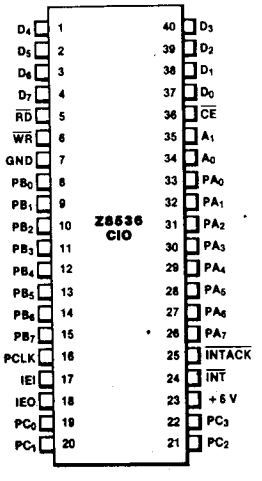


Figure 2. Pin Assignments

Absolute Maximum Ratings

Voltages on all inputs and outputs with respect to GND . . . . . -0.3 V to +7.0 V

Operating Ambient Temperature . . . . . As Specified in Ordering Information

Storage Temperature . . . . . -65°C to +150°C

Stresses greater than those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; operation of the device at any condition above those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Standard Test Conditions

The characteristics below apply for the following standard test conditions, unless otherwise noted. All voltages are referenced to GND. Positive current flows into the referenced pin. Standard conditions are as follows:

- $+4.75\ V \leq V_{CC} \leq +5.25\ V$
- GND = 0 V
- T_A as specified in Ordering Information

All ac parameters assume a load capacitance of 50 pF max.

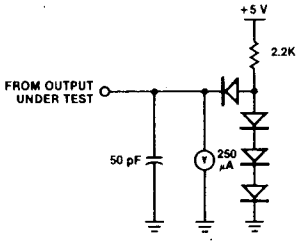


Figure 21. Standard Test Load

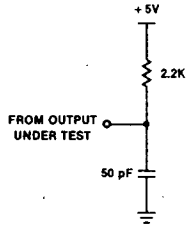


Figure 22. Open-Drain Test Load

DC Characteristics

Symbol	Parameter	Min	Max	Unit	Condition
V_{IH}	Input High Voltage	2.0	$V_{CC}+0.3$	V	
V_{IL}	Input Low Voltage	-0.3	0.8	V	
V_{OH}	Output High Voltage	2.4		V	$I_{OH} = -250\ \mu A$
V_{OL}	Output Low Voltage		0.4	V	$I_{OL} = +2.0$ mA
			0.5	V	$I_{OL} = +3.2$ mA
I_{IL}	Input Leakage		±10.0	µA	$0.4 \leq V_{IN} \leq +2.4$ V
I_{OL}	Output Leakage		±10.0	µA	$0.4 \leq V_{OUT} \leq +2.4$ V
I_{CC}	V_{CC} Supply Current		250	mA	

$V_{CC} = 5\ V \pm 5\%$ unless otherwise specified, over specified temperature range.

Capacitance

Symbol	Parameter	Min	Max	Unit	Test Condition
C_{IN}	Input Capacitance		10	pF	Unmeasured Pins Returned to Ground
C_{OUT}	Output Capacitance		15	pF	
$C_{I/O}$	Bidirectional Capacitance		20	pF	

f = 1 MHz, over specified temperature range.

Ordering Information

Product Number	Package/ Temperature	Speed	Description
Z8536	CE, CM, CMB, CS, DE, DS, PE, PS	4.0 MHz	CIO (40-pin)
Z8536A	CE, CM, CMB, CS, DE, DS, PE, PS	6.0 MHz	Same as above
Z8036	CE, CS, DE, DS, PE, PS	4.0 MHz	Z-CIO (40-pin)
Z8036	CM, CMB	6.0 MHz	Same as above
Z8036A	CE, CM, CMB, CS, DE, DS, PE, PS	6.0 MHz	Same as above

NOTES: C = Ceramic, D = Cerdip, P = Plastic; E = -40°C to +85°C, M = -55°C to +125°C, MB = -55°C with MIL-STD-883 with Class B processing, S = 0°C to +70°C.

Index

-R-

-S-

-T-

-W-

-Z-

LifeElectronics
Живое партнерство

ООО «ЛайфЭлектроникс» **“LifeElectronics” LLC**

ИНН 7805602321 КПП 780501001 Р/С 40702810122510004610 ФАКБ "АБСОЛЮТ БАНК" (ЗАО) в г.Санкт-Петербурге К/С 30101810900000000703 БИК 044030703

Компания «Life Electronics» занимается поставками электронных компонентов импортного и отечественного производства от производителей и со складов крупных дистрибьюторов Европы, Америки и Азии.

С конца 2013 года компания активно расширяет линейку поставок компонентов по направлению коаксиальный кабель, кварцевые генераторы и конденсаторы (керамические, пленочные, электролитические), за счёт заключения дистрибьюторских договоров

Мы предлагаем:

- *Конкурентоспособные цены и скидки постоянным клиентам.*
- *Специальные условия для постоянных клиентов.*
- *Подбор аналогов.*
- *Поставку компонентов в любых объемах, удовлетворяющих вашим потребностям.*

- *Приемлемые сроки поставки, возможна ускоренная поставка.*
- *Доставку товара в любую точку России и стран СНГ.*
- *Комплексную поставку.*
- *Работу по проектам и поставку образцов.*
- *Формирование склада под заказчика.*

- *Сертификаты соответствия на поставляемую продукцию (по желанию клиента).*
- *Тестирование поставляемой продукции.*
- *Поставку компонентов, требующих военную и космическую приемку.*
- *Входной контроль качества.*
- *Наличие сертификата ISO.*

В составе нашей компании организован Конструкторский отдел, призванный помогать разработчикам, и инженерам.

Конструкторский отдел помогает осуществить:

- *Регистрацию проекта у производителя компонентов.*
- *Техническую поддержку проекта.*
- *Защиту от снятия компонента с производства.*
- *Оценку стоимости проекта по компонентам.*
- *Изготовление тестовой платы монтаж и пусконаладочные работы.*

Тел: +7 (812) 336 43 04 (многоканальный)
Email: org@lifeelectronics.ru

www.lifeelectronics.ru