Chipsmall
深圳市津利帝科技有限公司
Geelead Technology Limited
Chipsmall
采购部

Z80 CMOS Microprocessors

Z84C90 KIO Serial/ Parallel Counter/Timer

Product Specification

PS011804-0612

www.zilog.com

Warning: DO NOT USE THIS PRODUCT IN LIFE SUPPORT SYSTEMS.

LIFE SUPPORT POLICY

ZILOG'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS PRIOR WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF ZILOG CORPORATION.

AS USED HEREIN

Life support devices or systems are devices which (a) are intended for surgical implant into the body, or (b) support or sustain life and whose failure to perform when properly used in accordance with instructions for use provided in the labeling can be reasonably expected to result in a significant injury to the user. A critical component is any component in a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system or to affect its safety or effectiveness.

DOCUMENT DISCLAIMER

Information in this publication concerning the devices, applications, or technology described is intended to suggest possible uses and may be superseded. ZILOG, INC. DOES NOT ASSUME LIABILITY FOR OR PROVIDE A REPRESENTATION OF ACCURACY OF THE INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED IN THIS DOCUMENT. ZILOG ALSO DOES NOT ASSUME LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT RELATED IN ANY MANNER TO USE OF INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED HEREIN OR OTHERWISE. The information contained within this document has been verified according to the general principles of electrical and mechanical engineering.

ZMOTION, Z8 Encore! XP and eZ8 are trademarks or registered trademarks of Zilog, Inc. (An IXYS Company). All other product or service names are the property of their respective owners.

Revision History

Each instance in the following revision history table reflects a change to this document from its previous version. For more details, refer to the corresponding pages or appropriate links provided in the table.

Date	Revision Level	Description	Page Number
Jun 2013	04	Corrected to remove internal discussion tags.	N/A
Jun 2012	03	Updated to include missing information covered in the DC8321-00 Databook (2Q94).	All
Sep 2002	02	Added Z84C90 KIO Peripheral: Serial/Parallel Counter/Timer Packages table, modified AC Characteristics of the Z84C90 table.	2, 38
Sep 2002	01	Original issue.	All

Table of Contents

List of Figures

List of Tables

Introduction

Zilog's Z84C90 Serial/Parallel Counter/Timer KIO is a multichannel, multipurpose I/O peripheral device designed to provide the end user with a cost-effective and powerful solution to meet an assortment of peripherals requirements. The Z84C90 KIO Peripheral combines the features of one Z84C30 CTC, one Z84C20 PIO and a Z84C4x SIO, plus an 8-bit, bit-programmable I/O port and a crystal oscillator into a single 84-pin PLCC or 100-pin LQFP package. Utilizing fifteen internal registers for data and programming information, the KIO can easily be configured to any given system environment. Although optimum performance is obtained with a Z84C00 CPU, the KIO can just as easily be used with any other CPU.

Features

The Z84C90 Serial/Parallel Counter/Timer KIO offers the following features:

- Two independent synchronous/asynchronous serial channels
- Three 8-bit parallel ports
- Four independent counter/timer channels
- On-chip clock oscillator/driver
- Software/hardware resets
- Designed in CMOS for low power operations
- Supports Z80 Family interrupt daisy chain
- Programmable interrupt priorities
- 8, 10 and 12.5 MHz bus clock frequency
- Single +5 V power supply

Table 1 lists the differing frequencies offered for the Z84C90 KIO Peripheral by package and part number.

Table 1. Z84C90 KIO Peripheral: Serial/Parallel Counter/Timer Packages

Part Number	Package	Frequency (MHz)
Z84C9008ASC	100-pin LQFP	8
Z84C9010ASC	100-pin LQFP	10
Z84C9008VEC	84-pin PLCC	8
Z84C9008VSC	84-pin PLCC	8
Z84C9010VSC	84-pin PLCC	10
Z84C9012VSC	84-pin PLCC	12

Figure 1 illustrates a block diagram of the Z84C90 KIO Peripheral.

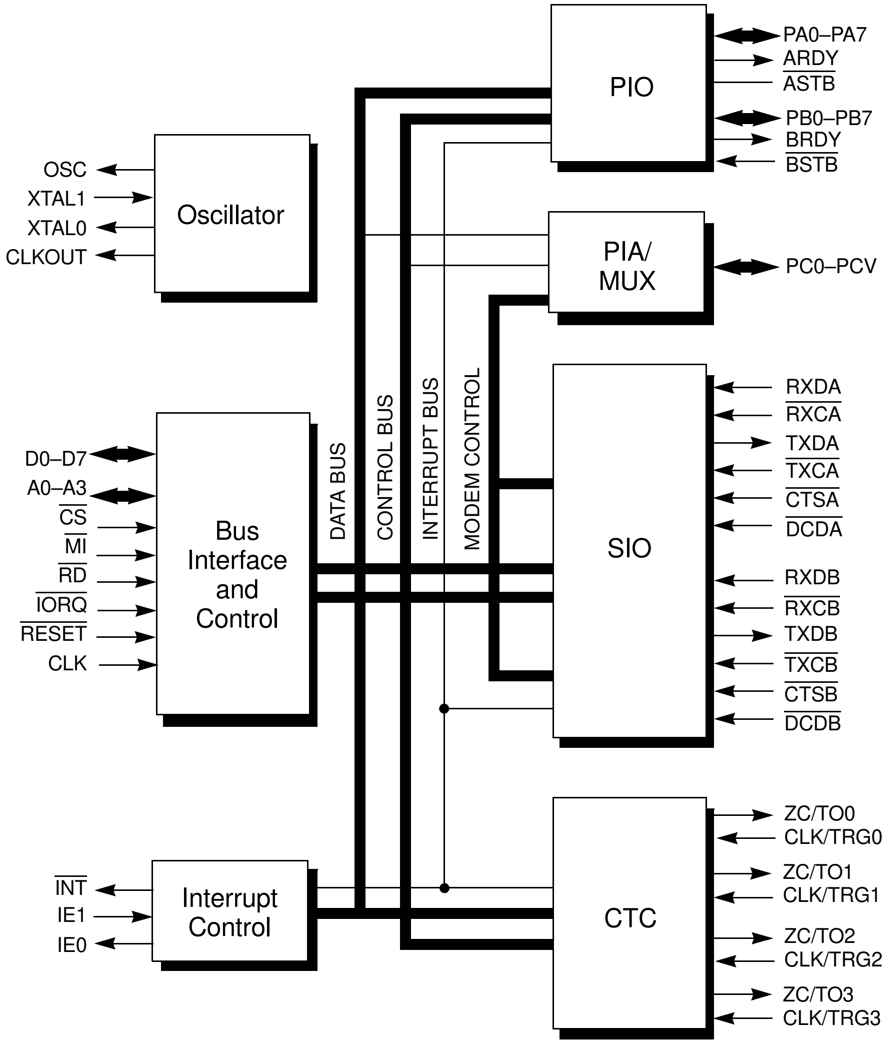


Figure 1. A Block Diagram of the Z84C90 KIO Peripheral

Block Descriptions

Z84C20 Parallel Input/Output Logic Unit. This logic unit provides both TTL- and CMOS-compatible interfaces between peripheral devices and a CPU through the use of two 8-bit parallel ports. The CPU configures the logic to interface to a wide range of peripheral devices with no external logic. Typical devices that are compatible with this interface are keyboards, printers and EPROM/PAL programmers.

The parallel ports (designated Port A and Port B) are byte-wide and completely compatible with the Z84C90 PIO (see Figure 2). These two ports feature several modes of operation: input, output, bidirectional or bit control. Each port features two handshake signals (RDY and $\overline{\text{STB}}$) which can be used to control data transfers. The RDY (ready) indicates that the port is ready for data transfer while $\overline{\text{STB}}$ (strobe) is an input to the port that indicates when data transfer has occurred. Each of the ports can also be programmed to interrupt the CPU upon the occurrence of specified status conditions and generate unique interrupt vectors when the CPU responds.

For more information about the operation of this portion of the logic, please refer to the Z8420/Z84C20 PIO Product Specification (PS0180).

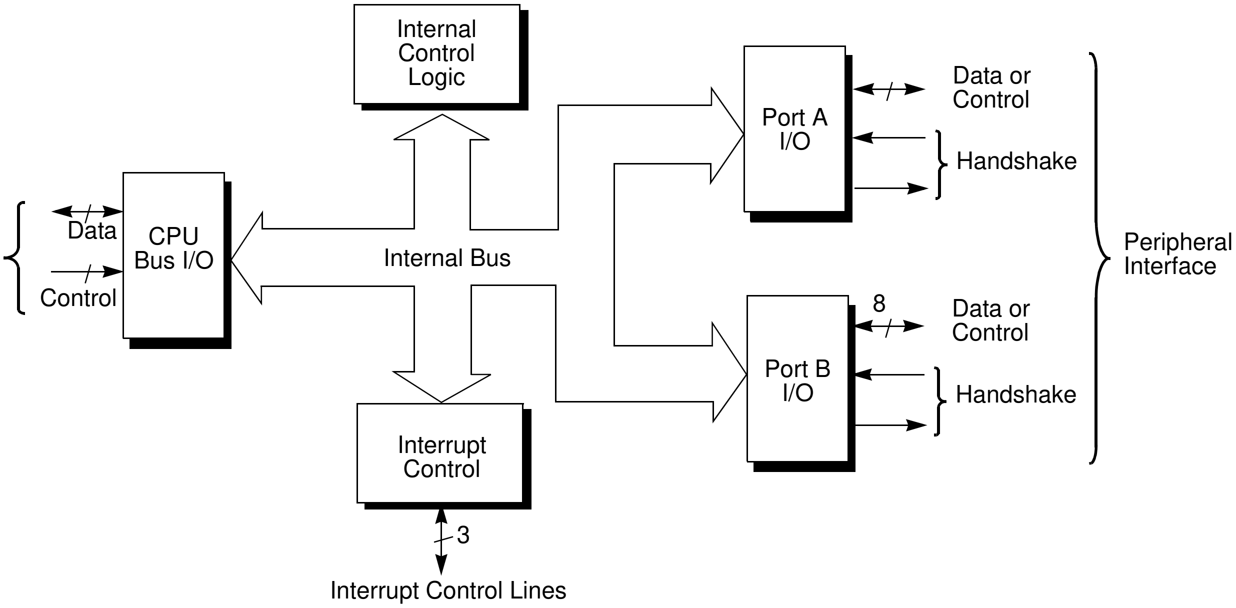


Figure 2. Z84C20 Parallel Input/Output Block Diagram

Parallel Interface Adapter (PIA) Logic Unit. This logic also offers an additional 8 bits of I/O to the user, referred to as the PIA port (see Figure 3). This port, designated as Port C, is bit-programmable for data transfers; each bit can be individually programmed as either an input or an output. Bit direction control is performed through the programming of the PIA Control Register. When programmed as outputs, the output data latches are programmed with an I/O write cycle; their state can be read with an I/O read cycle. When programmed as inputs, the state of the external pin is read with the I/O read cycle. This port does not have handshake capabilities and offers no interrupt capabilities. This port is multiplexed to provide the additional modem and CPU control signals for the serial I/O logic unit, when appropriate.

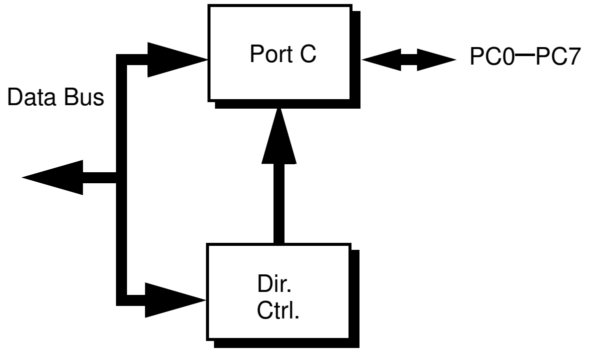


Figure 3. Parallel Interface Adapter Block Diagram

When a read from the PIA port occurs, input data will be latched when $\overline{IORQ}$, $\overline{CS}$ and $\overline{RD}$ all detected as active. The data bus will display this data as a result of the rising edge of the clock input after this occurrence. When a write to the PIA port occurs, data will be written as a result of the rising edge of the clock input after $\overline{IORQ}$ and $\overline{CS}$ have been detected as active and $\overline{RD}$ has been detected as inactive.

Counter/Timer Logic (CTC) Unit. This logic unit provides the user with four individual 8-bit counter/timer channels that are compatible with the Z84C30 CTC (see Figure 4). The counter/timers can be programmed by the CPU for a broad range of counting and timing applications. Typical applications include event counting, interrupt and interval timing and serial baud rate clock generation.

Each of the counter/timer channels, designated Channels 0 through 3, have an 8-bit prescaler (when used in timer mode) as well as its own 8-bit counter to provide a wide range of count resolution. Each of the channels also have their own clock/trigger input to quantify the counting process and an output to indicate zero crossing/time-out conditions. With only one interrupt vector programmed into this logic unit, each channel can generate a unique interrupt vector in response to the interrupt acknowledge cycle.

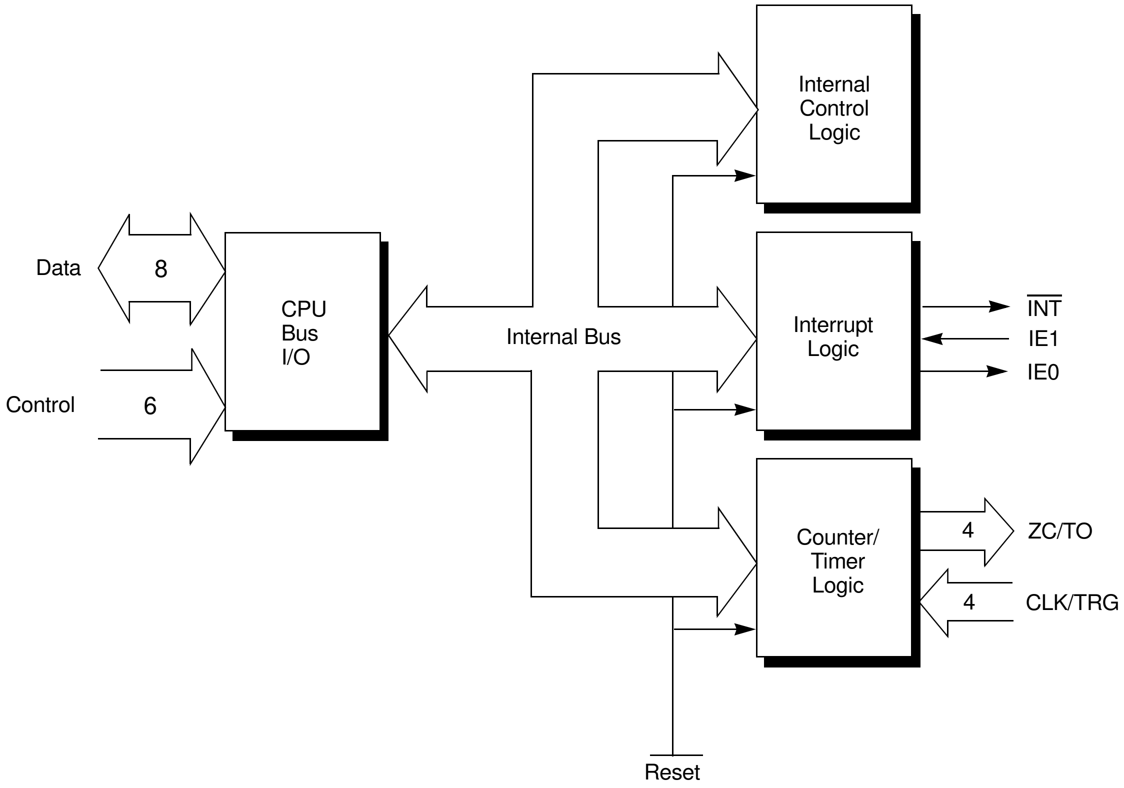


Figure 4. Counter/Timer Block Diagram

Serial I/O Logic Unit. This logic unit provides the user with two separate serial I/O channels that are completely compatible with the Z84C4x SIO (see Figure 5). Their basic functions as serial-to-parallel and parallel-to-serial converters can be programmed by a CPU for a broad range of serial communications applications. Each channel, designated Channel A and Channel B, is capable of supporting all common asynchronous and synchronous protocols (Monosync, Bisync and SDLC/HDLC), byte- or bit-oriented.

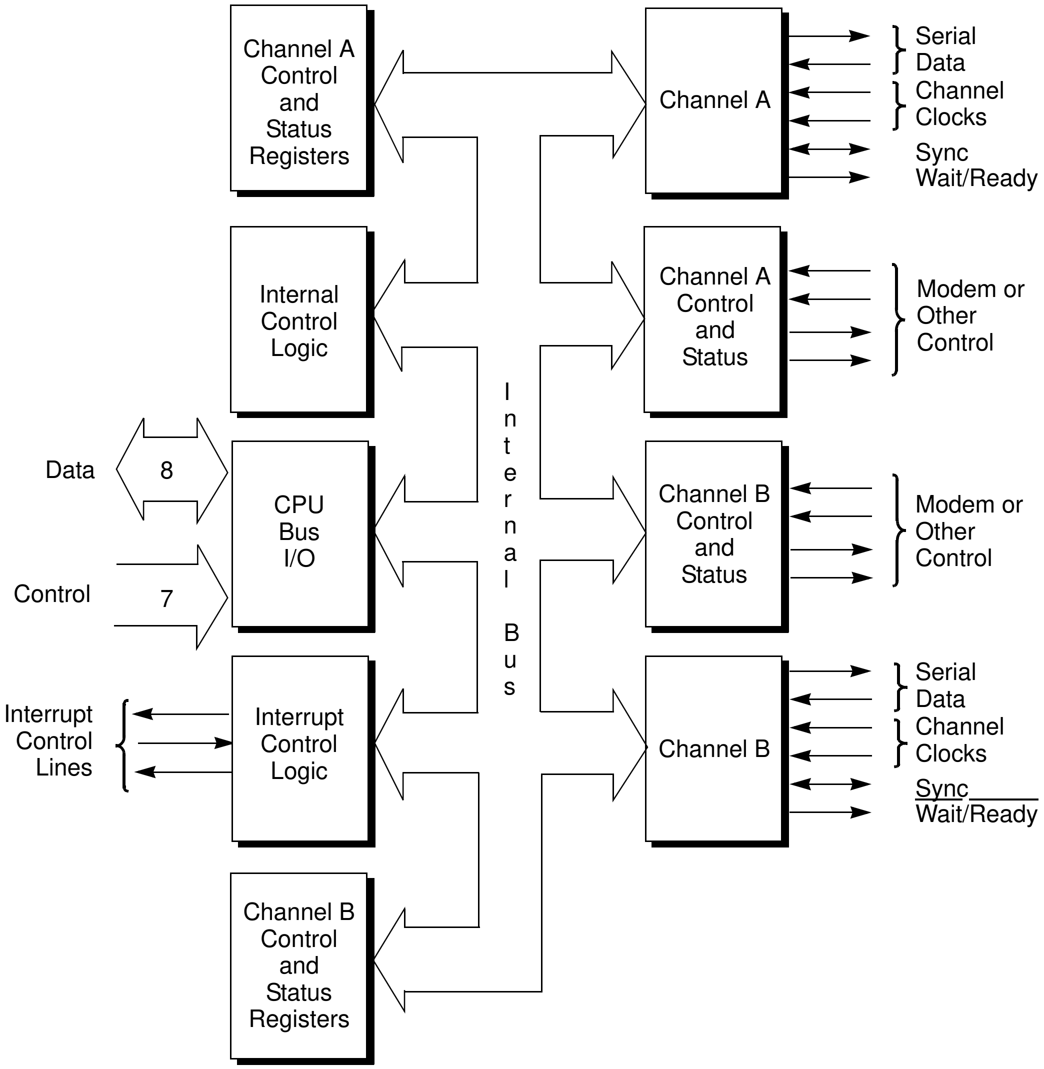


Figure 5. SIO Block Diagram

In the default state of the KIO, each serial channel supports full duplex communication with separate transmit and receive data lines, two modem control signals ($\overline{\text{CTS}}$ and $\overline{\text{DCD}}$) and separate transmit and receive clock inputs. Optionally, additional modem and CPU/DMA control signals can be obtained through the PIA port.

For more information about the operation of this portion of the logic, please refer to the Z8420/Z84C20 PIO Product Specification (PS0180).

Clock Oscillator/Driver Logic Unit. A clock oscillator/driver is available that will allow the user to eliminate circuitry with in a new design, or that can be used as another oscillator within the system. This logic will accept either a crystal ceramic resonator or TTL-compatible clock input and generate a MOS-compatible clock output and also an oscillator reference output. Zilog recommends a fundamental parallel resonant crystal; see Figure 6. The preferred value of the two capacitors C1 and C2 is 33pF each.

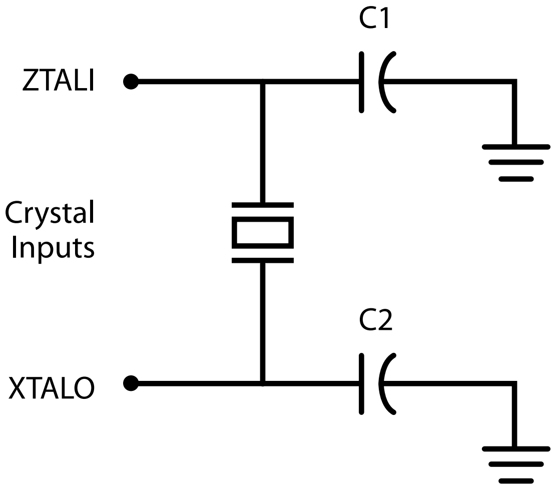


Figure 6. Crystal Connection

Command Logic Unit. This logic unit provides for much more than just controlling the interface between the KIO and the CPU. The main function provided by this unit is to allow the user to configure the internal interrupt daisy chain of the KIO into a preferred sequence of peripherals to interrupt. Any one of the three devices (SIO, CTC, PIO) can be the highest priority, while another can be second priority and the remaining device the third. The user can even configure the daisy chain such that no internal peripherals are involved in the chain. Programming of the daisy chain configuration is performed by programming the Command Register with the appropriate 3-bit pattern in addresses D0–D2, with D3 set to 1.

A second function of this logic unit is to provide software-controllable hardware resets to each of the individual devices. As a result, an individual peripheral is allowed to be reset without having to reset the entire KIO. Requiring bit D3 to be set to a 1 to program the daisy chain configuration allows the user to reset the individual devices without changing the daisy chain. The software reset commands for the individual devices still remain available to the user.

A third function of the Command Register allows the user to obtain use of the additional control signals of the SIO logic instead of the PIA port by programming bit D7 of the Command Register with a 1.

Pin Signals

Figure 7 shows the pin-outs for the 84-pin PLCC Z84C90 KIO Peripheral package; Figure 8 shows the 100-pin LQFP pin-outs.

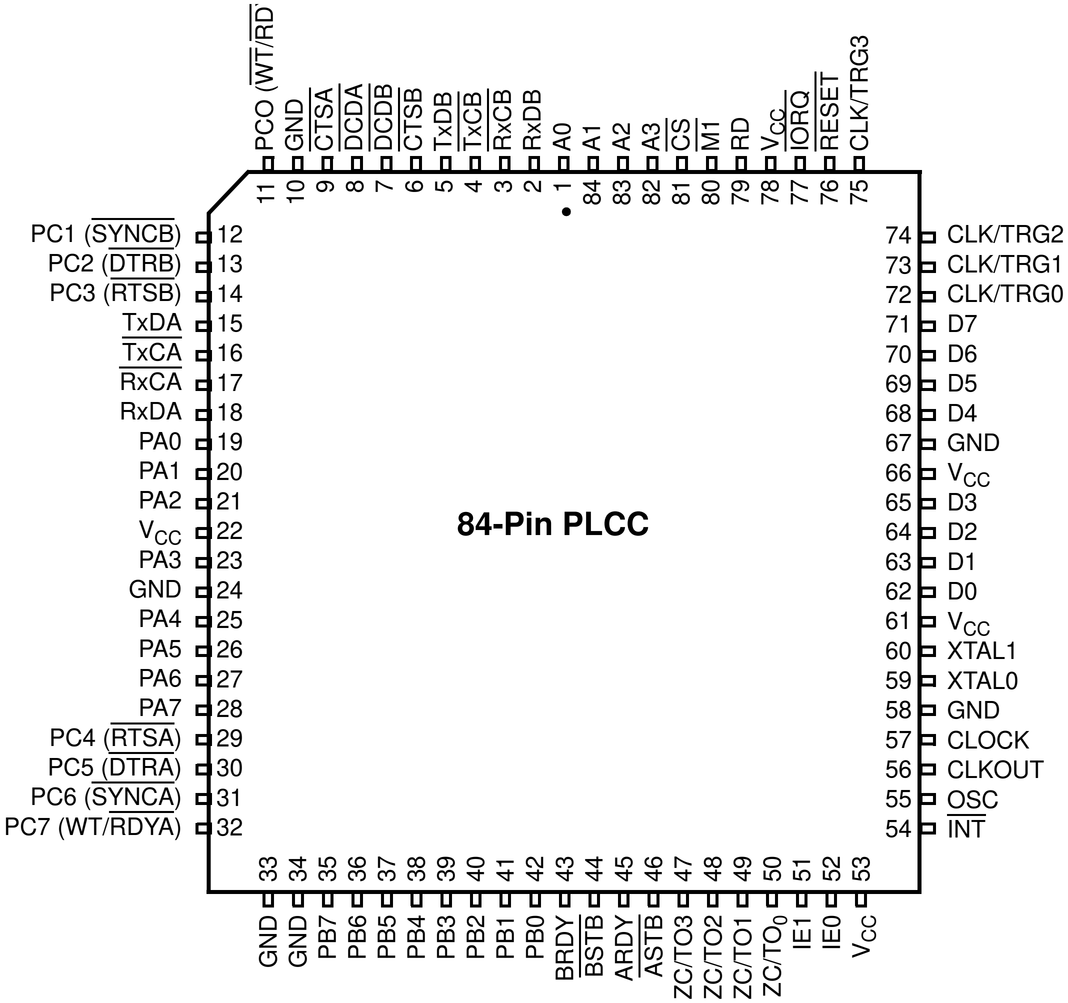


Figure 7. Z84C90 84-Pin PLCC Configuration

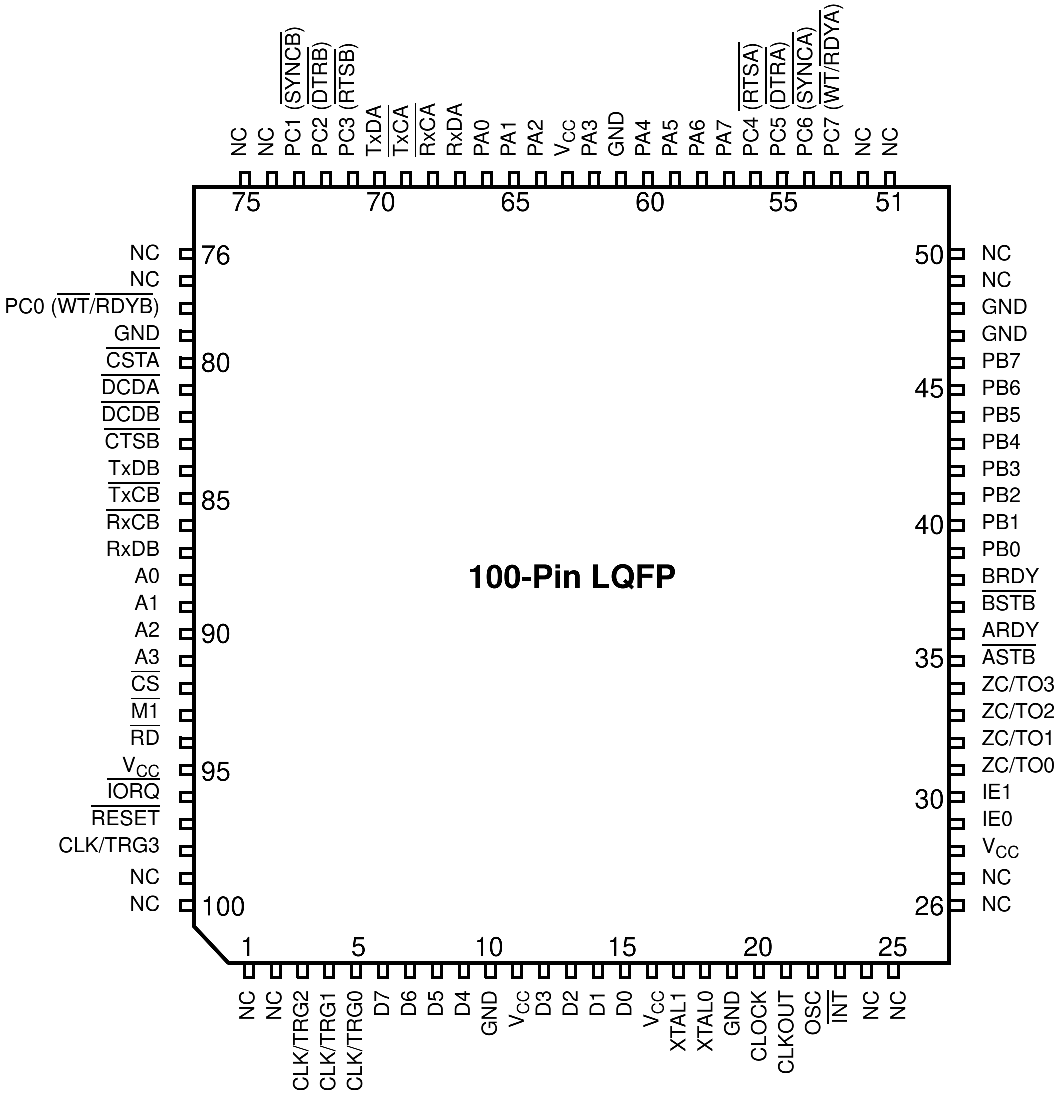


Figure 8. 100-Pin LQFP Configuration

Pin Descriptions

A0–A3. Address bus (inputs). Used to select the port/register for each bus cycle.

ARDY, BRDY. Port Ready (outputs, active High). These signals indicate that the port is ready for a data transfer. In Mode 0, the signal indicates that the port has data available to the peripheral device. In Mode 1, the signal indicates that the port is ready to accept data from the peripheral device. In Mode 2, ARDY indicates that Port A has data available for the peripheral device, but that the data is not be placed onto PA0–PA7 until the $\overline{\text{ASTB}}$ signal is Active. BRDY indicates that Port A is able to accept data from a peripheral device.

Note: Port B does not support Mode 2 operation and can only be used in Mode 3 when Port A is programmed for Mode 2. BRDY is not associated with Port B when it is operating in Mode 3.

$\overline{\text{ASTB}}$, $\overline{\text{BSTB}}$. Port Strobe (inputs, active Low). These signals indicate that the peripheral device has performed a transfer. In Mode 0, the signal indicates that the peripheral device has accepted the data present on the port pins. In Mode 1, the signal causes the data on the port pins to be latched onto Port A. In Mode 2, $\overline{\text{ASTB}}$ Low causes the data in the output data latch of Port A to be placed onto the Port A pins. $\overline{\text{BSTB}}$ Low causes the data present on the Port A pins to be latched into the Port A input data latch. The end of the current transaction is noted by the rising edge of these signals.

Note: Port B does not support Mode 2 operation, and can only be used in Mode 3 when Port A is programmed for Mode 2. $\overline{\text{BSTB}}$ is not associated with Port B when it is operating in Mode 3.

CLK/TRG0–CLK/TRG3. External Clock/Timer Trigger (inputs, user-selectable active High or Low). These four pins correspond to the four counter/timer channels of the KIO. In Counter mode, each active edge causes the downcounter to decrement. In Timer mode, an active edge starts the timer.

CLKOUT. Clock Out (output). This output is a divide-by-two of the oscillator (XTAL) input.

CLOCK. System Clock (input). This clock must be the same as (or a derivative of) the CPU clock. If the CLKOUT is to be used as the system clock, then these two pins must be connected together.

CS. Chip Select (input, active Low). Used to activate the internal register decoding mechanism and allow the KIO to perform a data transfer to/from the CPU.

$\overline{\text{CTSA}}$, $\overline{\text{CTSB}}$. Clear to Send (inputs, active Low). These signals are modem control signals for the serial channels. When programmed for Auto Enable, a Low on these pins enables their respective transmitters. If not programmed as Auto Enable, these pins may be used as general-purpose input signals.

D0–D7. Data Bus (bidirectional, active High, tristated). Used for data exchanges between the CPU and the KIO for programming and data transfer. The KIO also monitors the data bus for Return from Interrupt (RETI) instructions to maintain its Interrupt Under Service (IUS) status.

$\overline{\text{DCDA}}$, $\overline{\text{DCDB}}$. Data Carrier Detect (inputs, active Low). These signals are modem control signals for the serial channels. When programmed for Auto Enable, a Low on these pins enables their respective receivers. If not programmed as Auto Enable, these pins may be used as general-purpose input signals.

$\overline{\text{DTRA}}$, $\overline{\text{DTRB}}$. Data Terminal Ready (outputs, active Low). These signals are modem control signals for the serial channels. They follow the state programmed into their respective serial channels, and are multiplexed with Port C, bits 5 and 2, respectively.

IEI. Interrupt Enable In (input, active High). This signal is used with Interrupt Enable Out (IEO) to form a priority daisy chain when there is more than one interrupt-driven device. A High on this line indicates that no higher-priority device is requesting an interrupt.

IEO. Interrupt Enable Out (output, active High). This signal is used with Interrupt Enable In (IEI) to form a priority daisy chain when there is more than one interrupt-driven device. A High on this line indicates that this device is requesting an interrupt, and that no higher-priority device, is not requesting an interrupt. A Low blocks any lower-priority devices from requesting an interrupt.

IORQ. Input/Output Request (input, active Low). $\overline{\text{IORQ}}$ is used with $\overline{\text{RD}}$, A0–A3, and $\overline{\text{CS}}$ to transfer data between the KIO and the CPU. When $\overline{\text{IORQ}}$, $\overline{\text{RD}}$, and $\overline{\text{CS}}$ are active Low, the device selected by A0–A3 transfers data to the CPU. When $\overline{\text{IORQ}}$ and $\overline{\text{CS}}$ are active Low, but $\overline{\text{RD}}$ is active High, the device selected by A0–A3 is written into by the CPU. When $\overline{\text{IORQ}}$ and $\overline{\text{M1}}$ are both active Low, the KIO may respond with an interrupt vector from its highest-priority interrupting device.

M1. Machine Cycle 1 (input, active Low). When $\overline{\text{M1}}$ and $\overline{\text{RD}}$ are Low, the Z80 CPU fetches an instruction from memory; the KIO decodes this cycle to determine if the RETI instruction sequence is being executed. When $\overline{\text{M1}}$ and $\overline{\text{IORQ}}$ are both active, the KIO decodes the cycle to be an interrupt acknowledge, and may respond with a vector from its highest-priority interrupting device.

OSC. Oscillator (output). This output is a reference clock for the oscillator.

PA0–PA7. Port A Bus (bidirectional, tristated). One of the 8-bit ports of the PIO. PA0 is the least-significant bit of the bus.

PB0–PB7. Port B Bus (bidirectional, tristated). One of the 8-bit ports of the PIO. PB_0 is the least-significant bit of the bus. This port can also supply 1.5mA at 1.5V to drive Darlington transistors.

PC0–PC7. Port C Bus (bidirectional, tristated). PC_0 is the least-significant bit of the bus. These pins are multiplexed between the 8-bit PIA and additional modem control signals for the serial channels.

RD. Read (input, active Low). When $\overline{\text{RD}}$ is active, a memory or I/O read operation is in progress. $\overline{\text{RD}}$ is used with A0–A3, $\overline{\text{CS}}$ and $\overline{\text{IORQ}}$ to transfer data between the KIO and CPU.

RESET. Reset (input, active Low). A Low on this pin forces the KIO into a Reset condition. This signal must be active for a minimum of three clock cycles. When the KIO is reset, the following events occur:

- The PIO ports are in Mode 1 operation
- Handshakes are inactive and interrupts are disabled
- The PIA port is in Input mode and active
- CTC channel counting is terminated and interrupts are disabled
- SIO channels are disabled
- Marking with interrupts is disabled

All control registers must be rewritten after a hardware reset.

$\overline{\text{RTSA}}$, $\overline{\text{RTSB}}$. Request to Send (outputs, active Low). These signals are modem control signals for their serial channels. They follow the inverse state programmed into their respective serial channels, and are multiplexed with Port C, bits 4 and 3, respectively.

$\overline{\text{RxCA}}$, $\overline{\text{RxCB}}$. Receive Clock (inputs, active Low). These clocks are used to assemble the data in the receiver shift register for their serial channels. Data is sampled on the rising edge of the clock.

RxDA, RxDB. Receive Data (inputs, active High). These pins are the input data pins to the receive shift register for their serial channels.

$\overline{\text{SYNCA}}$, $\overline{\text{SYNCB}}$. Synchronization (bidirectional, active Low). In the Asynchronous mode of operation, these pins act much like the $\overline{\text{CTS}}$ and $\overline{\text{DCD}}$ pins. Transitions affect the Sync/

Hunt status bit for their respective serial channels, but serve no other purpose. These pins are multiplexed with Port C, bits 6 and 1, respectively.

$\overline{\text{TxCA}}$, $\overline{\text{TxCB}}$. Transmit Clock (inputs, active Low). These clocks are used to transmit data from the transmit shift register for their serial channels. Data is transmitted on the falling edge of the clock.

TxDA, TxDB. Transmit Data (outputs, active High). These pins are the output data pins from the transmitter for their serial channels.

$\overline{\text{WT}}$/$\overline{\text{RDYA}}$, $\overline{\text{WT}}$/$\overline{\text{RDYB}}$. Wait/Ready (outputs, open-drain when programmed as Wait; tristated when programmed as Ready). These pins may be programmed as Ready lines for a DMA controller or Wait lines for interfacing to a CPU. As a Ready line, these pins indicate (when active Low) that the transmitter or the receiver requests a transfer between the serial channel and the DMA. As a Wait line, these pins dictate (when Low) that the CPU must wait until the transmitter or receiver can complete the requested transaction. These pins are multiplexed with Port C, bit 7 and 0, respectively.

XTALI. Crystal/Clock Connection. (input).

XTALO. Crystal Connection. (output).

ZC/TO0–ZC/TO3. Zero count/Time-Out (outputs, active High). These four pins are outputs from the four counter/timer channels of the KIO. Each pin pulses High when its corresponding downcounter reaches 0.

Register Address Decoding for the KIO

Address lines A0–A3 determine which one of the 16 control registers is being accessed. Table 2 shows the address decoding of each of the KIO control registers; also see Figure 9 on page 15.

Table 2. KIO Registers

Address	A3	A2	A1	A0
Register 0: PIO Port A Data	0	0	0	0
Register 1: PIO Port A Command	0	0	0	1
Register 2: PIO Port B Data	0	0	1	0
Register 3: PIO Port B Command	0	0	1	1
Register 4: CTC Channel 0	0	1	0	0

Note: Additionally, $\overline{\text{IORQ}}$ and $\overline{\text{CS}}$ must be Low. Registers are written to or read from by the CPU, applying a 1 or a 0 respectively on the $\overline{\text{RD}}$ pin.

Table 2. KIO Registers (Continued)

Address	A3	A2	A1	A0
Register 5: CTC Channel 1	0	1	0	1
Register 6: CTC Channel 2	0	1	1	0
Register 7: CTC Channel 3	0	1	1	1
Register 8: SIO Port A Data	1	0	0	0
Register 9: SIO Port A Command/Status	1	0	0	1
Register 10: SIO Channel B Data	1	0	1	0
Register 11: SIO Channel B Command/Status	1	0	1	1
Register 12: PIA Port C Data	1	1	0	0
Register 13: PIA Port C Command	1	1	0	1
Register 14: KIO Command	1	1	1	0
Register 15: KIO Command B	1	1	1	1

Note: Additionally, $\overline{\text{IORQ}}$ and $\overline{\text{CS}}$ must be Low. Registers are written to or read from by the CPU, applying a 1 or a 0 respectively on the $\overline{\text{RD}}$ pin.

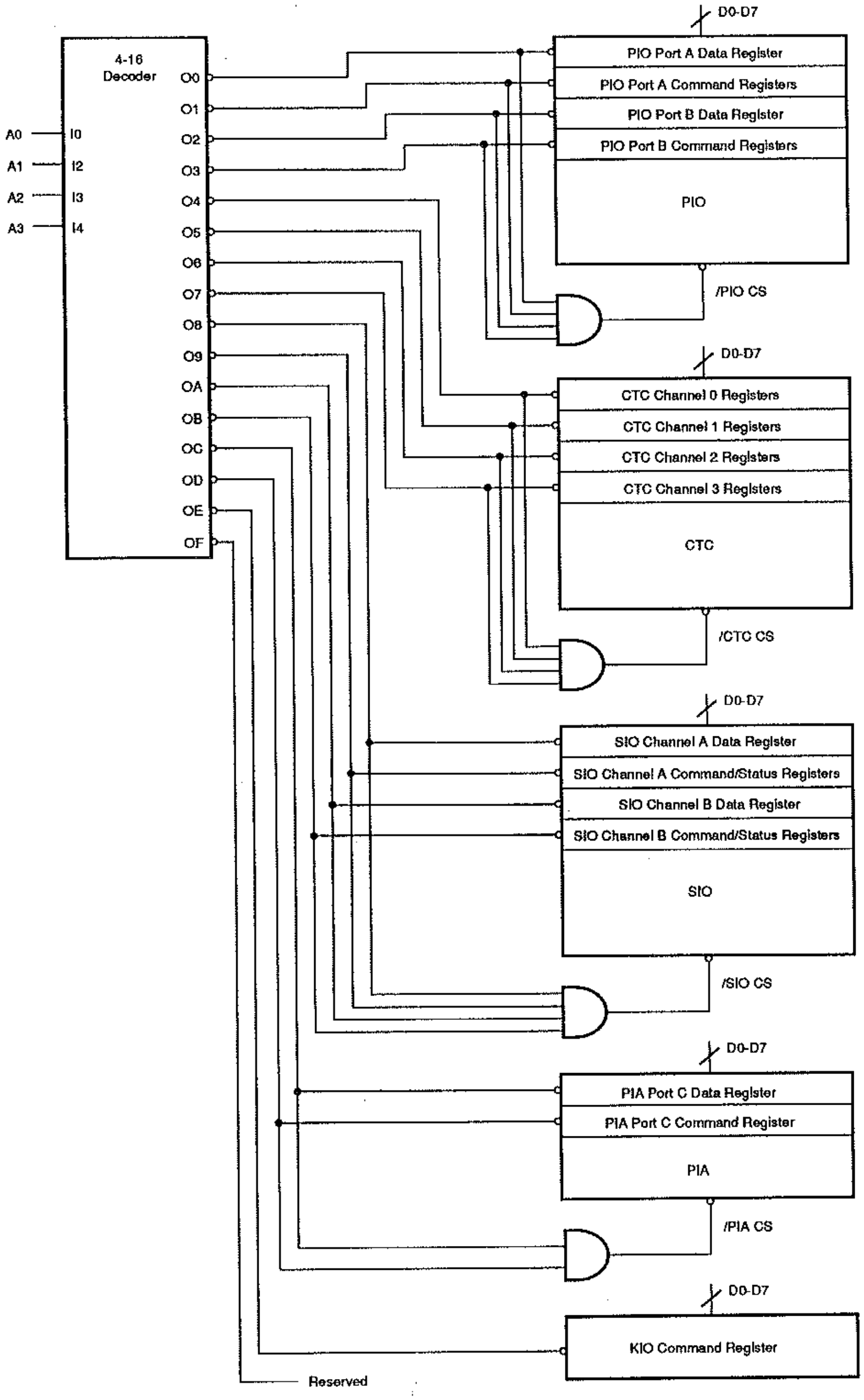


Figure 9. KIO Register Addressing

Register Programming

This section describes how the bits within each of the Z84C90 KIO's command registers change their respective command register functions, as well as the effects of such changes.

PIO Registers

The PIO registers described in this section apply to channels A and B (additionally, see the Register Address Decoding for the KIO section on page 13). For more information about these PIO registers, please consult the Z80 CPU Peripherals User Manual (UM0081).

Interrupt Vector Word. When Bit 0 of the command register is cleared to 0, the command register functions as the Interrupt Vector Word Register. The PIO logic unit is designed to work with the Z80 CPU in Interrupt Mode 2. This word must be programmed if interrupts are to be used; bit D0 must be 0. See Figure 10.

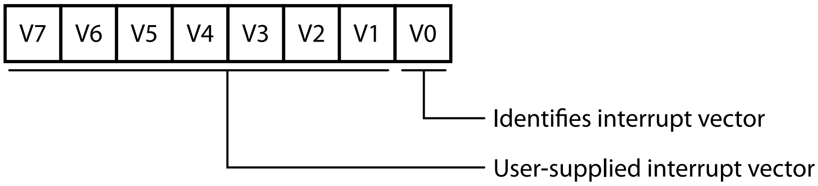


Figure 10. PIO Interrupt Vector Word Register

Mode Control Word. When bits B2 to B0 are set to 1, the command register functions as the Mode Control Word Register. Selects the port operating mode. This word is required and can be written at any time. See Figure 11.

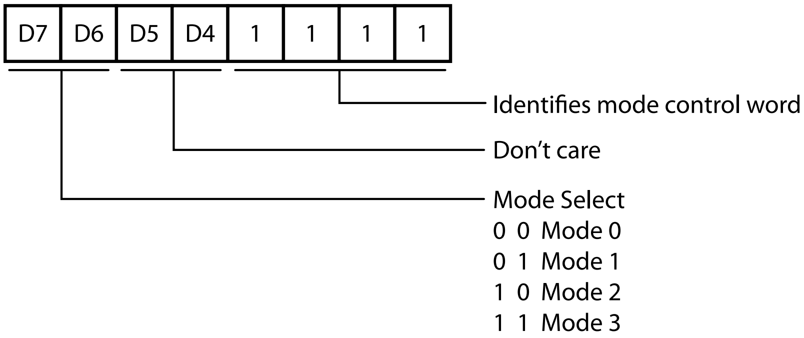


Figure 11. PIO Mode Control Word Register

I/O Register Control Word. When Mode 3 is selected, the Mode Control Word data must be followed by the loading of the I/O Register Control Word data. This word configures the I/O Register, which defines which port lines are inputs or outputs. A 1 indicates input, while a 0 indicates output. this word is required with in Mode 3. See Figure 12.

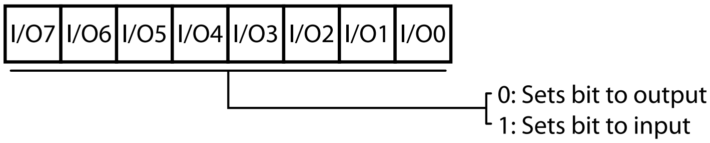
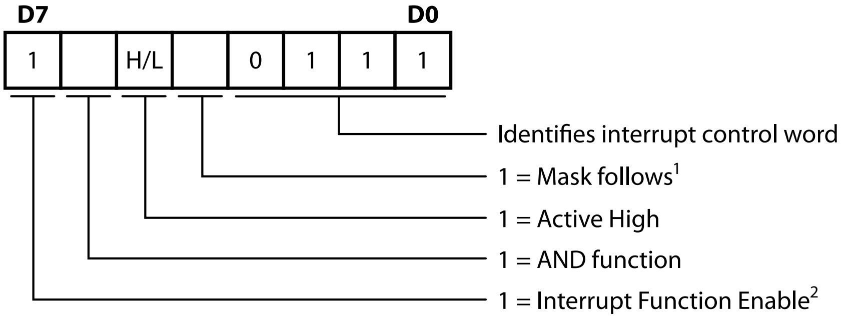


Figure 12. PIO I/O Register Control Word

PIO Interrupt Control Word. When bits D3 to D0 are loaded with 0111, the command register functions as the PIO Interrupt Control Word Register. In Mode 3 operation, handshake signals are not used. Interrupts are generated as a logic function of the input signal levels. The Interrupt Control Word sets the logic conditions and the logic levels required for generating an interrupt. Two logic conditions or functions are available: AND (if all input bits change to the active level, an interrupt is triggered) and OR (if any one of the input bits change to the active logic level, an interrupt is triggered). The user can also program which input bits are to be considered as part of this logic function. bit D6 sets the logic function, bit D5 sets the logic level and bit D4 specifies the mask control word data to follow. See Figure 13.


D7
D0
1
H/L
0
1
1
1
Identifies interrupt control word
1 = Mask follows1
1 = Active High
1 = AND function
1 = Interrupt Function Enable2

Notes:

1. Regardless of the operating mode, setting bit D4 = 1 causes any pending interrupts to be cleared.
2. The port interrupt is not enabled until the interrupt function enable is followed by an active $\overline{\text{M1}}$.

Figure 13. PIO Interrupt Control Word

Mask Control Word. This words sets the Mask Control Register, thus allowing any unused bits to be masked off. If any bits are to be masked, bit D4 of the Interrupt Control

Word must be set. When bit D4 of the Interrupt Control Word is set, the next word loaded into the command register must be the Mask Control Word. To mask an input bit, the corresponding Mask Control Word bit must be a 1. See Figure 14.

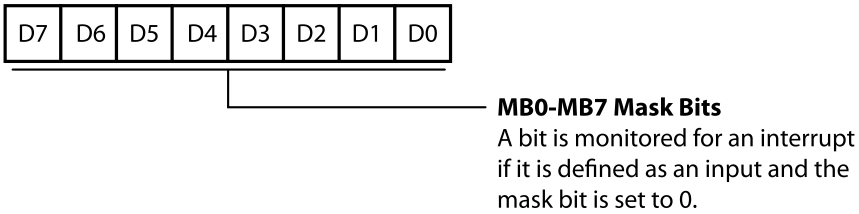

D7
D6
D5
D4
D3
D2
D1
D0
MB0-MB7 Mask Bits
A bit is monitored for an interrupt if it is defined as an input and the mask bit is set to 0.

Figure 14. PIO Mask Control Word

Interrupt Disable Word. When bits B3 to B0 are loaded with 0011, the command register functions as the Interrupt Disable Word Register. This word can be used to enable or disable a port's interrupts without change the remainder of the port's interrupt conditions. See Figure 15.

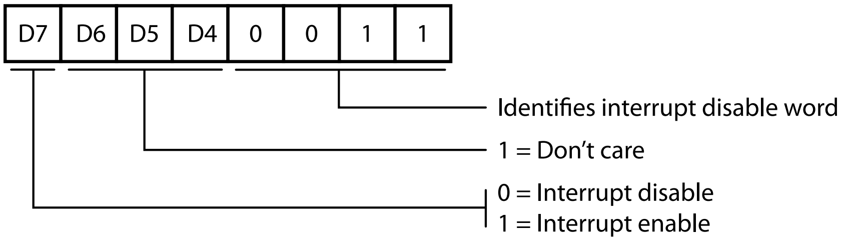


Figure 15. PIO Interrupt Disable Word

CTC Registers

The CTC registers apply to channels 0, 1, 2 and 3 (additionally, see the Register Address Decoding for the KIO section on page 13). For more information about these CTC registers, please consult the Z80 CPU Peripherals User Manual (UM0081).

Channel Control Word. This word sets the operating modes and parameters as described in the following paragraphs. Bit D0 of the CTC Register must be a 1 to indicate a Control Word; otherwise, it is an Interrupt Vector Word. See Figure 16.