Zilog - Z8019533FSC Datasheet

Welcome to E-XFL.COM

Understanding Embedded - Microprocessors

Embedded microprocessors are specialized computing chips designed to perform specific tasks within an embedded system. Unlike general-purpose microprocessors found in personal computers, embedded microprocessors are tailored for dedicated functions within larger systems, offering optimized performance, efficiency, and reliability. These microprocessors are integral to the operation of countless electronic devices, providing the computational power necessary for controlling processes, handling data, and managing communications.

Applications of Embedded - Microprocessors

Embedded microprocessors are utilized across a broad spectrum of applications, making them indispensable in

Details	
Product Status	Obsolete
Core Processor	Z8S180
Number of Cores/Bus Width	1 Core, 8-Bit
Speed	33MHz
Co-Processors/DSP	-
RAM Controllers	DRAM
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	-
SATA	-
USB	-
Voltage - I/O	5.0V
Operating Temperature	0°C ~ 70°C (TA)
Security Features	-
Package / Case	100-QFP
Supplier Device Package	100-QFP
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8019533fsc

TIMING DIAGRAMS

Z8S180 MPU Timing

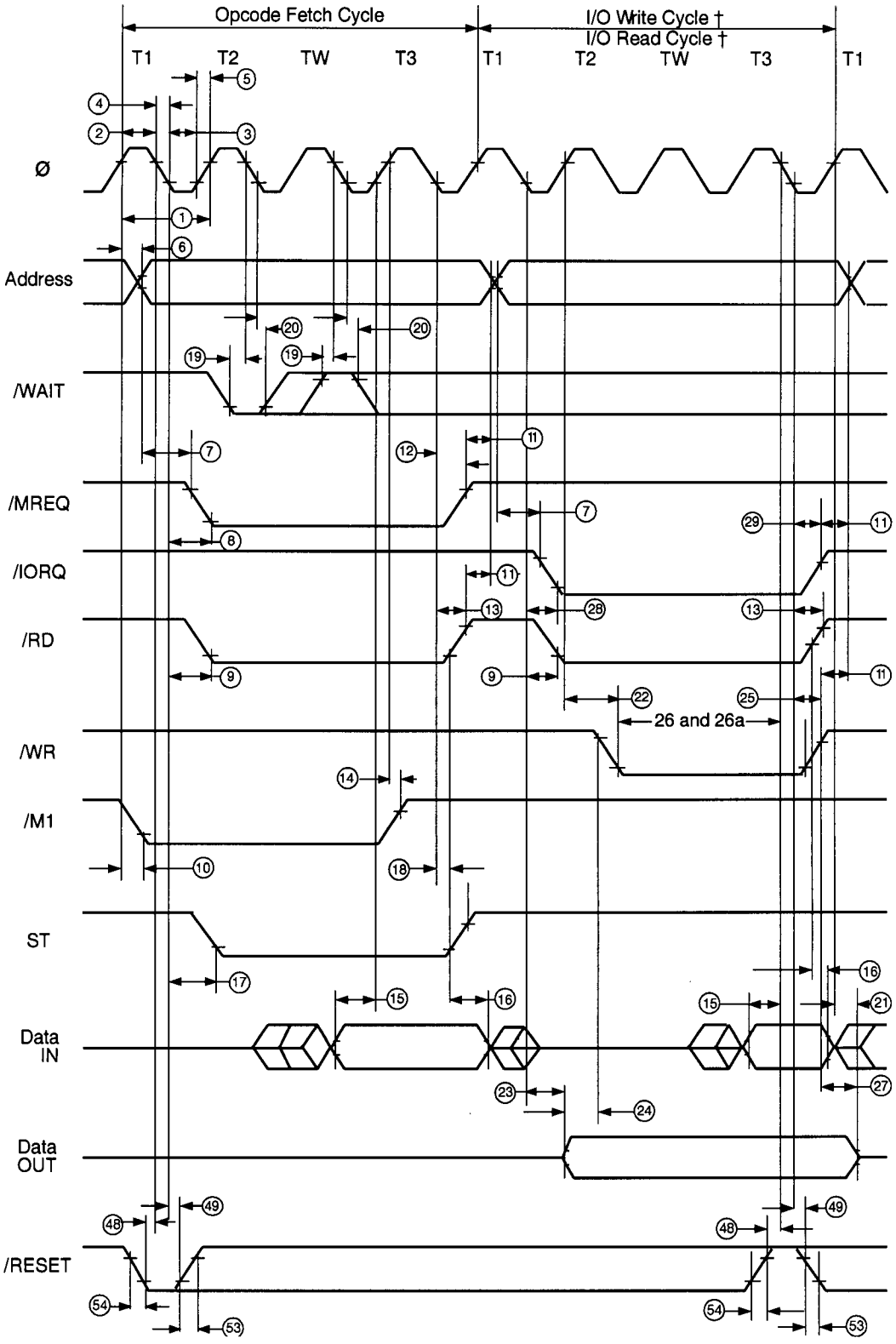


Figure 4. CPU Timing
(Opcode Fetch Cycle, Memory Read/Write Cycle
I/O Read/Write Cycle)

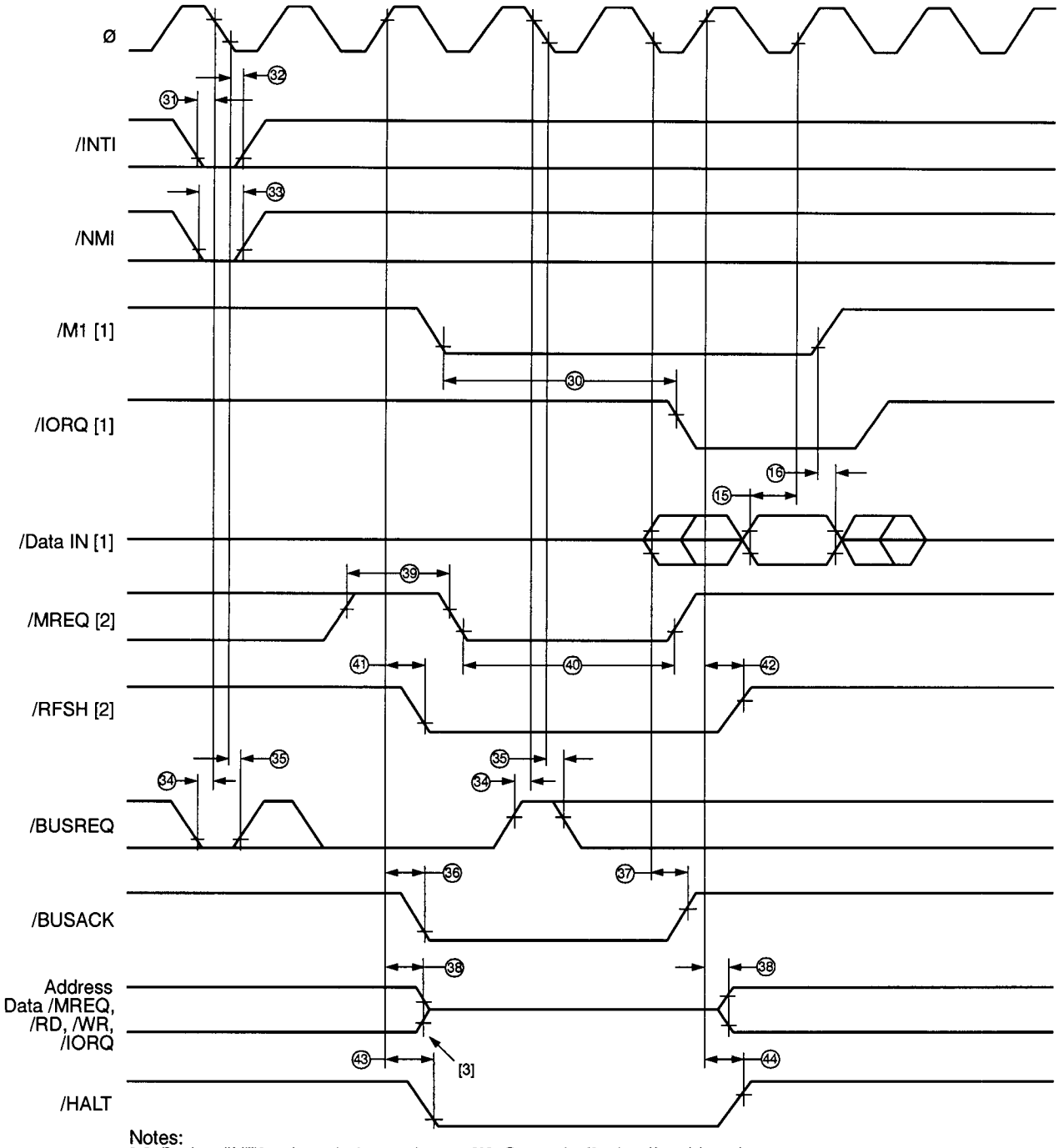


Figure 5. CPU Timing
(/INT0 Acknowledge Cycle, Refresh Cycle, BUS RELEASE mode
HALT mode, SLEEP mode, SYSTEM STOP mode)

EMSCC Timing

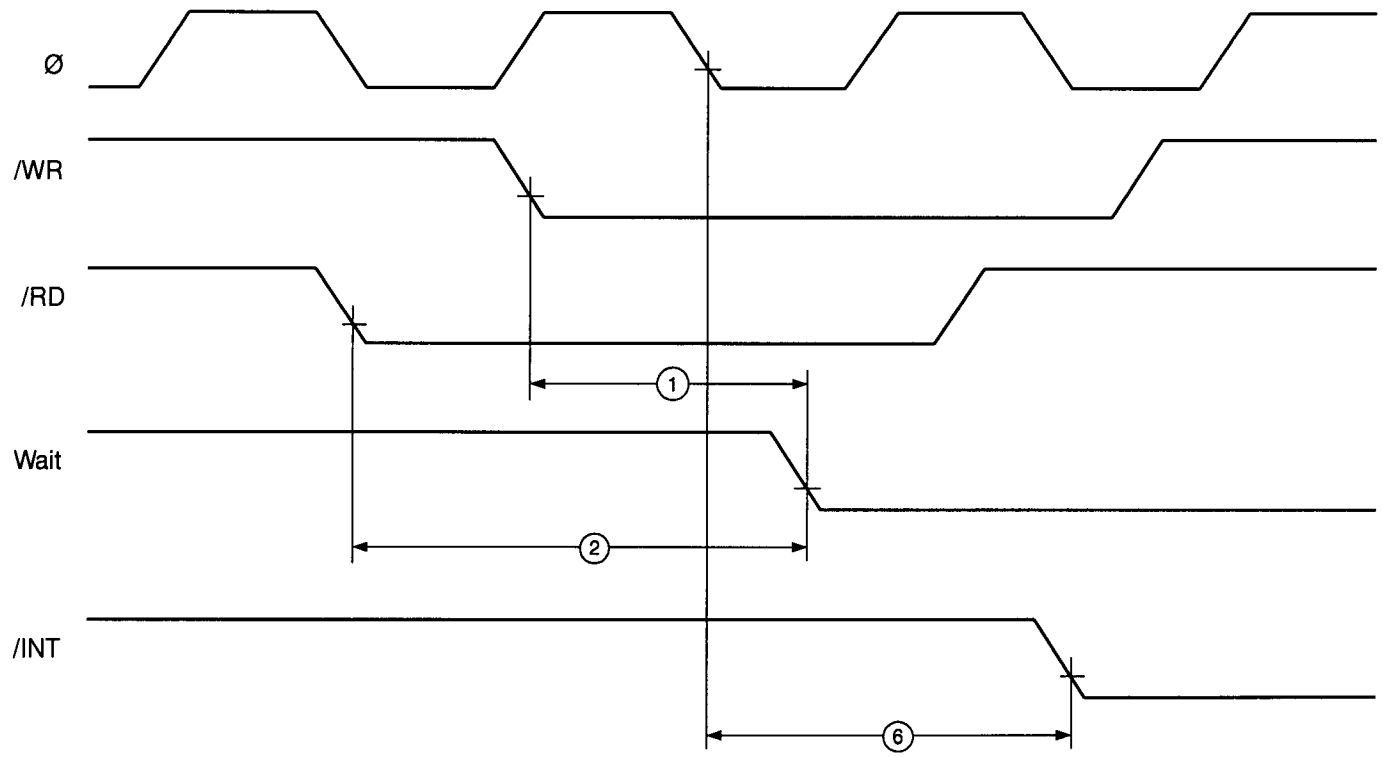


Figure 16. EMSCC AC Parameters

EMSCC Timing Parameters

No.	Symbol	Parameter	20 MHz Min	20 MHz Max	Unit
1	TdWR(W)	/WR Fall to Wait Valid Delay		50	ns
2	TdRD(W)	/RD Fall to Wait Valid Delay		50	
6	TdPC(INT)	Clock to /INT Valid Delay		160	

AC CHARACTERISTICS (Continued)

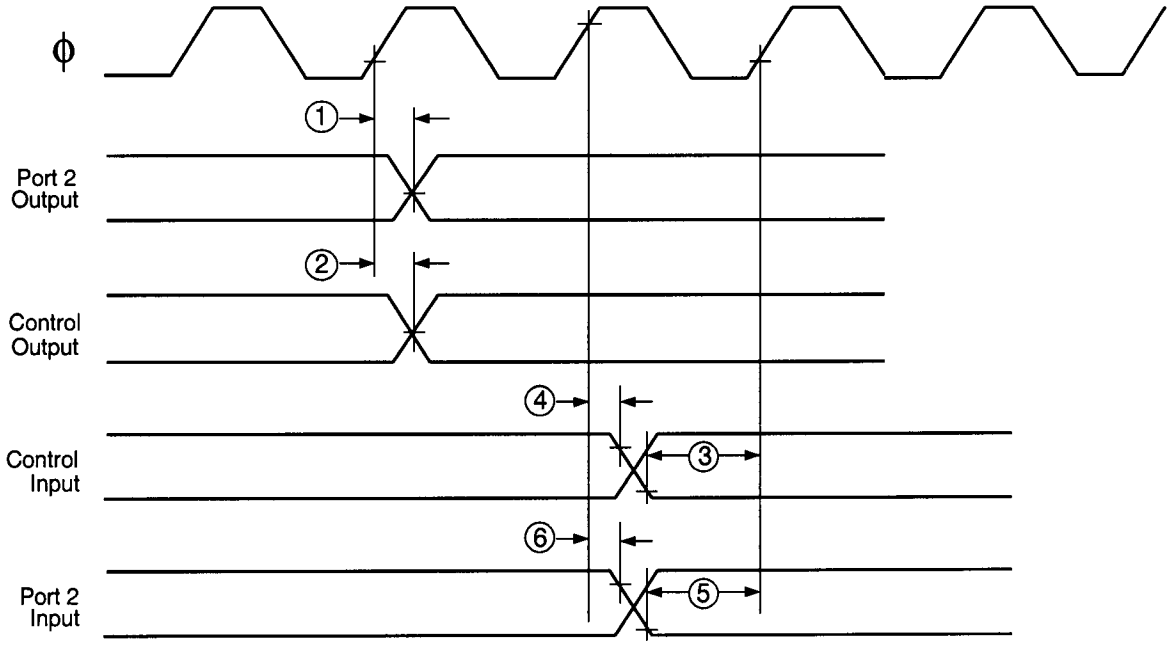


Figure 20. P1284 Bidirectional Centronics Interface Timing

P1284 Bidirectional Centronics Interface Timing

No.	Parameter	Min	Max	Units	Notes
1	CLK High to Port 2 Output		12	ns	
2	CLK High to Control Output		12	ns	(1)
3	Setup Time for Control Input to CLK High for Guaranteed Recognition	10		ns	(2)
4	Hold Time for Control Input from CLK High for Guaranteed Recognition	5		ns	(2)
5	Setup Time for Port 2 Inputs to CLK High for Guaranteed Recognition	10		ns	
6	Hold Time for Port 2 Inputs to CLK High for Guaranteed Recognition	5		ns	

Notes:

1. **Control Outputs**

Peripheral Mode	Host Mode
Busy/PtrBusy/PeriphAck	nStrobe/HostClk
nAck/PtrClk/PeriphClk	nAutoFd/HostBusy/HostAck
PError/AckDataReq/nAckReverse	nSelectIn/P1284Active
nFault/nDataAvail/nPeriphRequest	nInit/nReverseRequest
Select/Xflag	

2. **Control Inputs**

Peripheral Mode	Host Mode
Busy/PtrBusy/PeriphAck	nStrobe/HostClk
nAck/PtrClk/PeriphClk	nAutoFd/HostBusy/HostAck
PError/AckDataReq/nAckReverse	nSelectIn/P1284Active
nFault/nDataAvail/nPeriphRequest	nInit/nReverseRequest
Select/Xflag	

Z80185 MPU FUNCTIONAL DESCRIPTION (Continued)

Baud Rate Generator

The Baud Rate Generator (BRG) has two modes. The first is the same as in the Z80180. The second is a 16-bit down counter that divides the processor clock by the value in a 16-bit time constant register, and is identical to the EMSCC BRG. This allows a common baud rate of up to 512 Kbps to be selected. The BRG can also be disabled in favor of an external clock on the CKA pin.

The Receiver and Transmitter will subsequently divide the output of the BRG (or the signal from the CKA pin) by 1, 16 or 64, under the control of the DR bit in the CNTLB register, and the X1 bit in the ASCI Extension Control Register. To compute baud rate, use the following formulas.

If ss2,1,0 = 111, baud rate = f_{CKA} / Clock mode

else if BRG mode baud rate = f_{PHI} / (2 * (TC+2) * Clock mode)

else baud rate = f_{PHI} / ((10 + 20*PS) * 2^ss * Clock mode)

Where:
BRG mode is bit 3 of the ASEXT register
PS is bit 5 of the CNTLB register

TC is the 16-bit value in the ASCI Time Constant registers The TC value for a given baud rate is:

TC = (f_{PHI} / (2 * baud rate * Clock mode)) - 2

Clock mode depends on bit 4 in ASEXT and bit 3 in CNTLB:

X1	DR		Clock Mode
0	0	=	16
0	1	=	64
1	0	=	1
1	1	=	Reserved, do not use.

2^ss depends on the three LS bits of the CNTLB register:

ss2	ss1	ss0	2^ss
0	0	0	= 1
0	0	1	= 2
0	1	0	= 4
0	1	1	= 8
1	0	0	= 16
1	0	1	= 32
1	1	0	= 64
1	1	1	= External Clock from CKA0 (see above).

The ASCIs require a 50 percent duty cycle when CKA is used as an input. Minimum High and Low times on CKA0 are typical of most CMOS devices.

RDRF is set, and if enabled an Rx Interrupt or DMA Request is generated, when the receiver transfers a character from the Rx Shift Register to the Rx FIFO. The FIFO merely provides margin against overruns. When there's more than one character in the FIFO, and software or a DMA channel reads a character, RDRF either remains set or is cleared and then immediately set again. For example, if a receive interrupt service routine doesn't read all the characters in the RxFIFO, RDRF and the interrupt request remain asserted.

The Rx DMA request is disabled when any of the error flags PE or FE or OVRN are set, so that software can identify with which character the problem is associated.

If Bit 7, RDRF Interrupt Inhibit, is set to 1 (see Figures 32 and 33), the ASCI does not request a Receive interrupt when its RDRF flag is 1. Set this bit when programming a DMA channel to handle the receive data from an ASCI. The other causes for an ASCI Receive interrupt (PE, FE, OVRN, and for ASCI0, DCD) continue to request Rx interrupt if the RIE bit is 1. (The Rx DMA request is inhibited if PE or FE or OVRN is set, so that software can tell where an error occurred.) When this bit is 0, as it is after a Reset, RDRF will cause an ASCI interrupt if RIE is 1.

Programmable Reload Timer (PRT)

This logic consists of two separate channels, each containing a 16-bit counter (timer) and count reload register. The time base for the counters is derived from the system clock (divided by 20) before reaching the counter. PRT channel 1 provides an optional output to allow for waveform generation.

The TOUT output of PRT1 is available on a multiplexed pin.

Clocked Serial I/O (CSIO)

The pins for this function are multiplexed with the RTS, CTS, and clock pins for ASCI0. **Note:** It is possible to use both ASCI0 and the CSIO at the same time. If bit 4 of the System Configuration Register is set to 1, the CKS clock signal will internally drive the clock for ASCI0 instead of the system clock.

Z8S180 POWER-DOWN MODES

The following is a detailed description of the enhancements to the Z8S180 from the standard Z80180 in the areas of STANDBY, IDLE, and STANDBY-QUICK RECOVERY modes.

Add-On Features

There are five different power-down modes. SLEEP and SYSTEM STOP are inherited from the Z80180. In SLEEP mode, the CPU is in a stopped state while the on-chipI/Os are still operating. In I/O STOP mode, the on-chip I/Os are in a stopped state while leaving the CPU running. In SYSTEM STOP mode, both the CPU and the on-chip I/Os are in the stopped state to reduce current consumption. The Z8S180 has added two additional power-down modes, STANDBY and IDLE, to reduce current consumption even further. The differences in these power-down modes are summarized in Table 2.

Table 2. Power Down Modes

Power-Down Modes	CPU Core	On-Chip I/O	OSC.	CLKOUT	Recovery Source	Recovery Time (Minimum)
SLEEP	Stop	Running	Running	Running	RESET, Interrupts	1.5 Clock
I/O STOP	Running	Stop	Running	Running	By Programming	-
SYSTEM STOP	Stop	Stop	Running	Running	RESET, Interrupts	1.5 Clock
IDLE†	Stop	Stop	Running	Stop	RESET, Interrupts, BUSREQ	8 +1.5 Clock
STANDBY†	Stop	Stop	Stop	Stop	RESET, Interrupts, BUSREQ	2^{17} +1.5 Clock (Normal Recovery) 2^{6} +1.5 Clock (Quick Recovery)

Note: † IDLE and STANDBY modes are only offered in the Z8S180. Note that the minimum recovery time can be achieved if INTERRUPT is used as the Recovery Source.

STANDBY Mode

The Z8S180 is designed to save power. Two low-power programmable power-down modes have been added: STANDBY mode and IDLE mode. The STANDBY/IDLE mode is selected by multiplexing D6 and D3 of the CPU Control Register (CCR, I/O Address = 1FH).

To enter STANDBY mode:

1. Set D6 and D3 to 1 and 0, respectively.
2. Set the I/O STOP bit (D5 of ICR, I/O Address = 3FH) to 1.
3. Execute the SLEEP instruction.

When the device is in STANDBY mode, it behaves similar to the SYSTEM STOP mode as it exists on the Z80180, except that the STANDBY mode stops the external oscillator, internal clocks and reduces power consumption to 50 µA (typical).

Since the clock oscillator has been stopped, a restart ofthe oscillator requires a period of time for stabilization. An 18-bit counter has been added in the Z8S180 to allow for oscillator stabilization. When the part receives an external IRQ or BUSREQ during STANDBY mode, the oscillator is restarted and the timer counts down 2^{17} counts before acknowledgment is sent to the interrupt source.

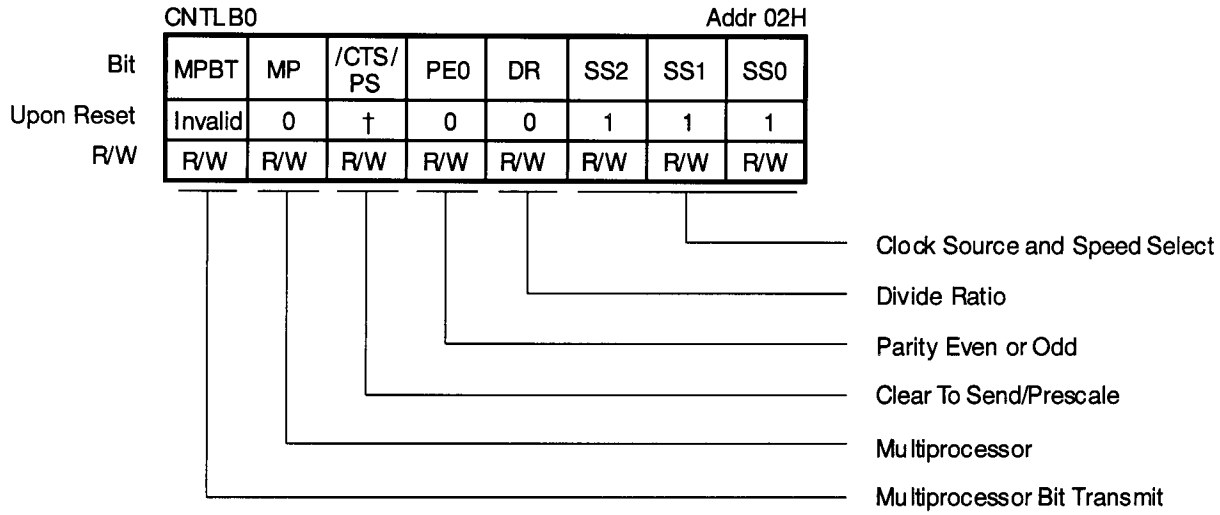
The recovery source needs to remain asserted for the duration of the 2^{17} count, otherwise standby will be resumed.

The following is a description of how the device exits STANDBY for different interrupts and modes of operation.

STANDBY Mode Exit with /RESET

The /RESET input needs to be asserted for a duration long enough for the crystal oscillator to stabilize, and then exit from the STANDBY mode. When /RESET is de-asserted, it goes through the normal reset timing to start instruction execution at address (logical and physical) 0000H.

The clocking is resumed within the Z8S180 and at the system clock output after /RESET is asserted when the crystal oscillator is restarted, but not yet stabilized.



† /CTS - Depending on the condition of /CTS pin.
PS - Cleared to 0.

General Divide Ratio	PS = 0 (Divide Ratio = 10)		PS = 1 (Divide Ratio = 30)	
SS, 2, 1, 0	**DR = 0 (x16)**	**DR = 1 (x64)**	**DR = 0 (x16)**	**DR = 1 (x64)**
000	Ø ÷ 160	Ø ÷ 640	Ø ÷ 480	Ø ÷ 1920
001	Ø ÷ 320	Ø ÷ 1280	Ø ÷ 960	Ø ÷ 3840
010	Ø ÷ 640	Ø ÷ 2560	Ø ÷ 1920	Ø ÷ 7680
011	Ø ÷ 1280	Ø ÷ 5120	Ø ÷ 3840	Ø ÷ 15360
100	Ø ÷ 2560	Ø ÷ 10240	Ø ÷ 7680	Ø ÷ 30720
101	Ø ÷ 5120	Ø ÷ 20480	Ø ÷ 15360	Ø ÷ 61440
110	Ø ÷ 10240	Ø ÷ 40960	Ø ÷ 30720	Ø ÷ 122880
111	External Clock (Frequency < Ø)			

Figure 24. ASCI Control Register B (Ch. 0)

TIMER DATA REGISTERS

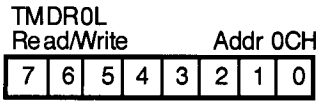
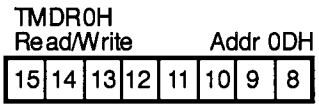


Figure 36. Timer 0 Data Register L



When Read, read Data Register L before reading Data Register H.

Figure 38. Timer 0 Data Register H

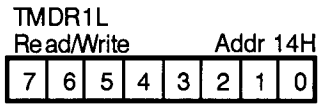
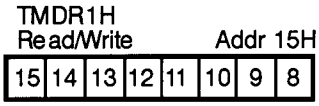


Figure 37. Timer 1 Data Register L



When Read, read Data Register L before reading Data Register H.

Figure 39. Timer 1 Data Register H

TIMER RELOAD REGISTERS

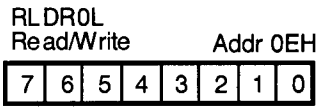


Figure 40. Timer 0 Reload Register L

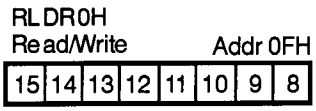


Figure 42. Timer 0 Reload Register H

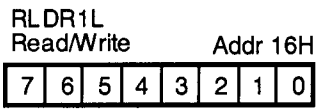


Figure 41. Timer 1 Reload Register L

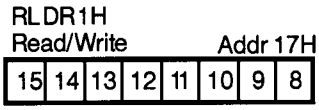


Figure 43. Timer 1 Reload Register H

DMA REGISTER DESCRIPTION

Bit 7. This bit should be set to 1 only when both DMA channels are set to take their requests from the same device. If this bit is 1 (it resets to 0), the channel end output of DMA channel 0 sets a flip-flop, so that thereafter the device's request is visible to channel 1, but is not visible to channel 0. The channel end output of channel 1 clears the FF, so that thereafter, the device's request is visible to channel 0, but not visible to channel 1.

Bit 6. When both DMA channels are programmed to take their requests from the same device, this bit (FF mentioned in the previous paragraph) controls which channel the device's request is presented to: 0 = DMA 0, 1 = channel 1. When bit 7 is 1, this bit is automatically toggled by the channel end output of the channels, as described above.

Bits 5-4. Reserved and should be programmed as 0.

Bits 3. This bit controls the direction and use of the TOUT/DREQ pin. When it's 0, TOUT/DREQ is the DREQ input; when it's 1, TOUT/DREQ is an output that can carry the TOUT signal from PRT1, if PRT1 is so programmed.

Bits 2-0. With "DIM1", bit 1 of DCNTL, these bits control which request is presented to DMA channel 1, as follows:

DIM1	IAR18-16	Request Routed to DMA Channel 1
0	000	ext TOUT/DREQ
0	001	ASCI0 Tx
0	010	ASCI1 Tx
0	011	EMSCC out
0	10X	Reserved, do not program.
0	1X0	Reserved, do not program.
0	111	PIA27-20 out
1	000	ext TOUT/DREQ
1	001	ASCI0 Rx
1	010	ASCI1 Rx or TOUT//DREQ pin
1	011	EMSCC in
1	10X	Reserved, do not program.
1	1X0	Reserved, do not program.
1	111	PIA27-20 in

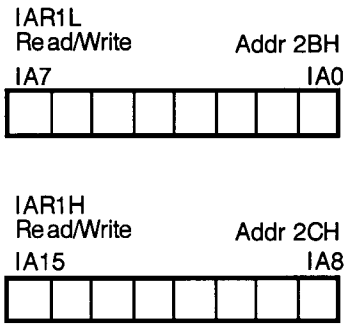


Figure 52. DMA I/O Address Registers

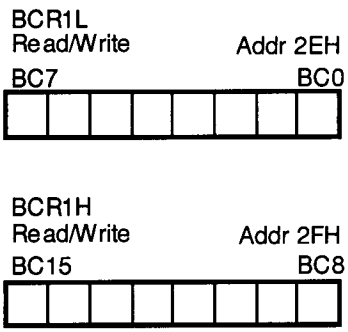


Figure 53. DMA 1 Byte Count Registers

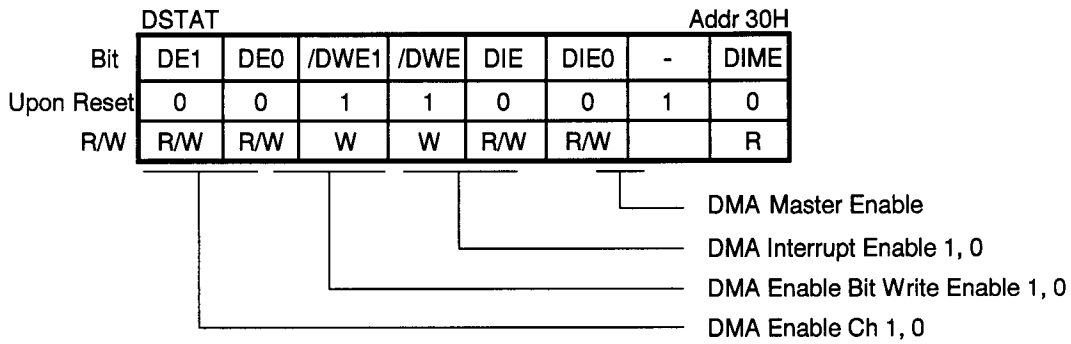


Figure 54. DMA Status Register

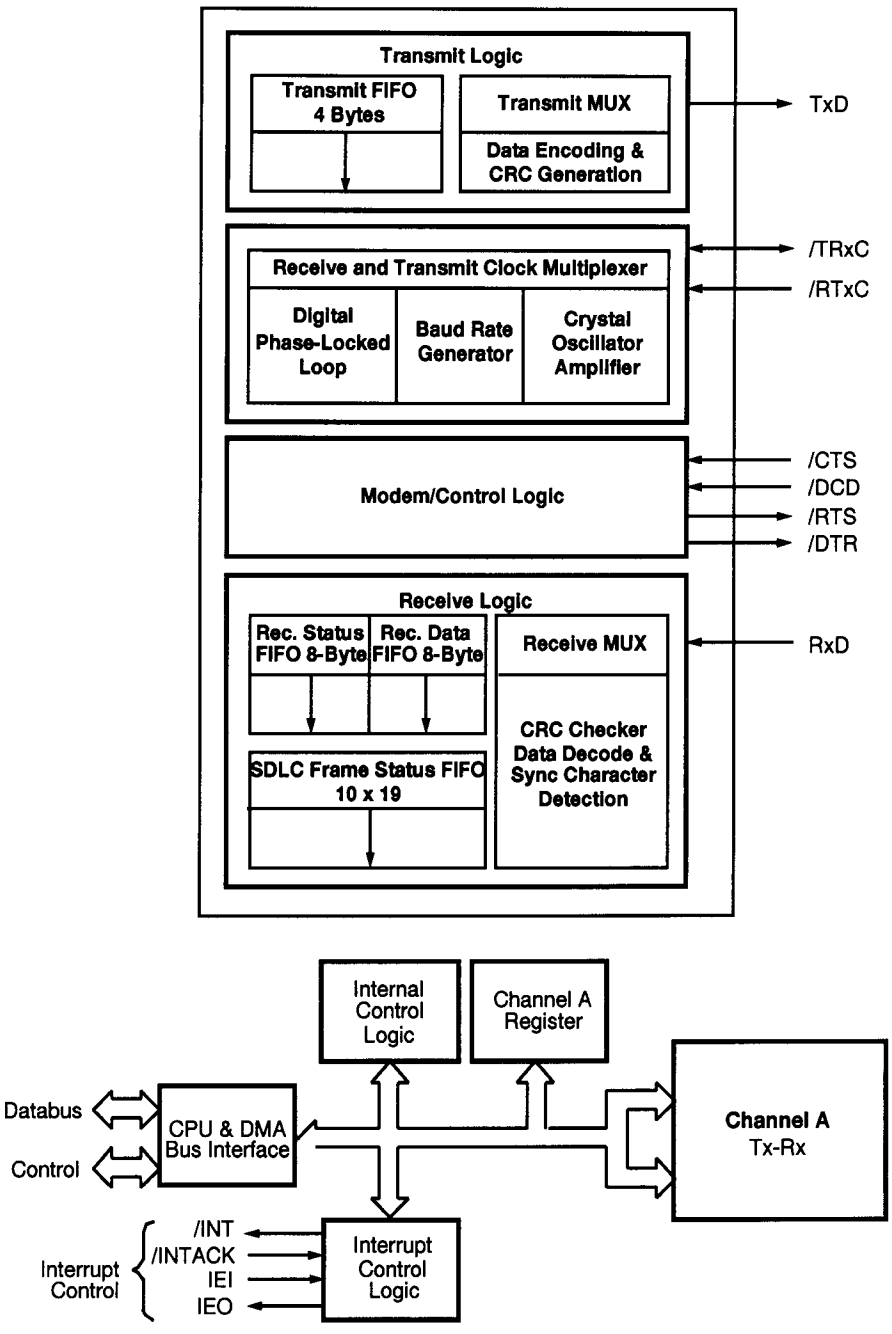


Figure 66. EMSCC Block Diagram

EMSCC REGISTERS

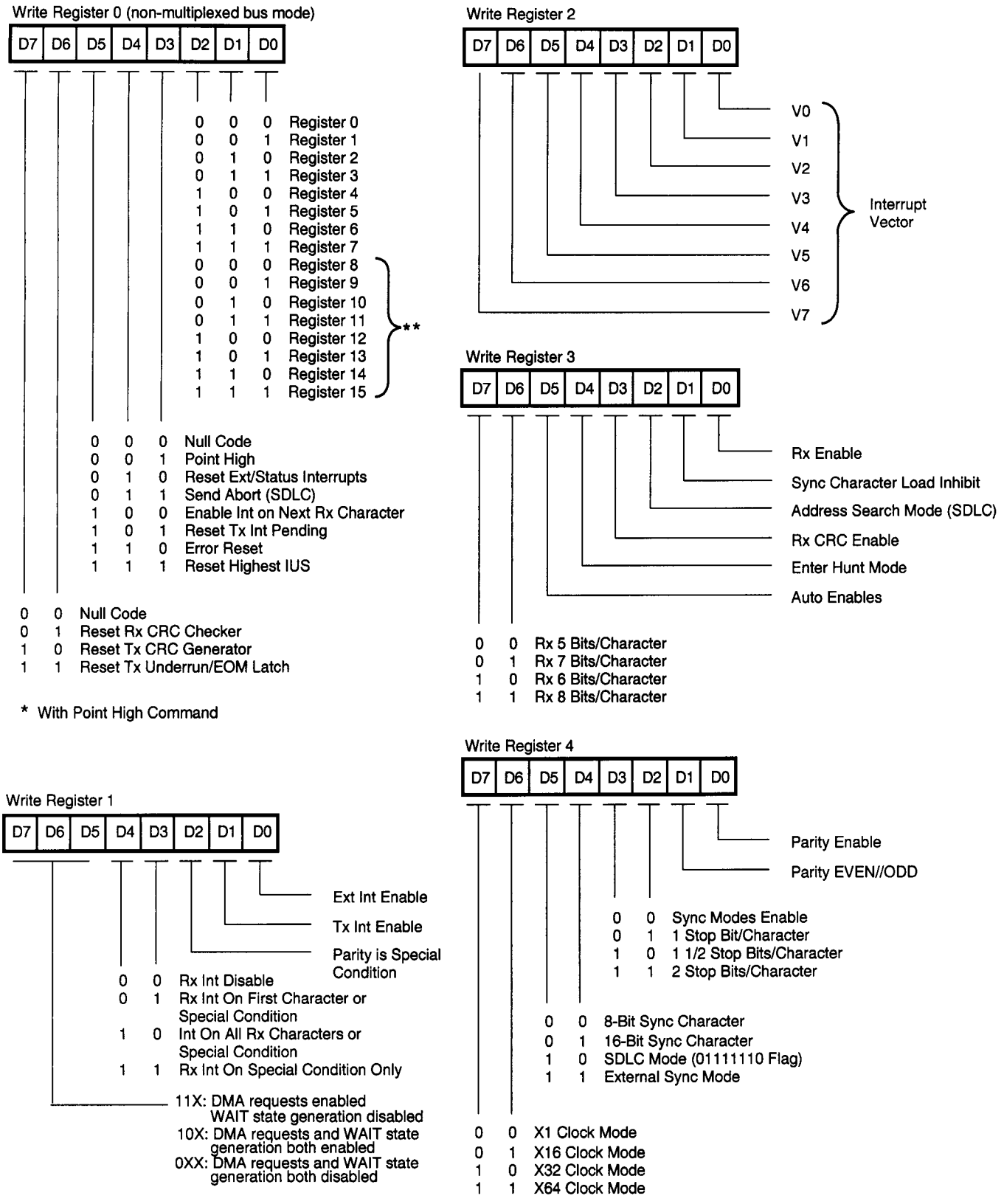


Figure 68. Write Register Bit Functions

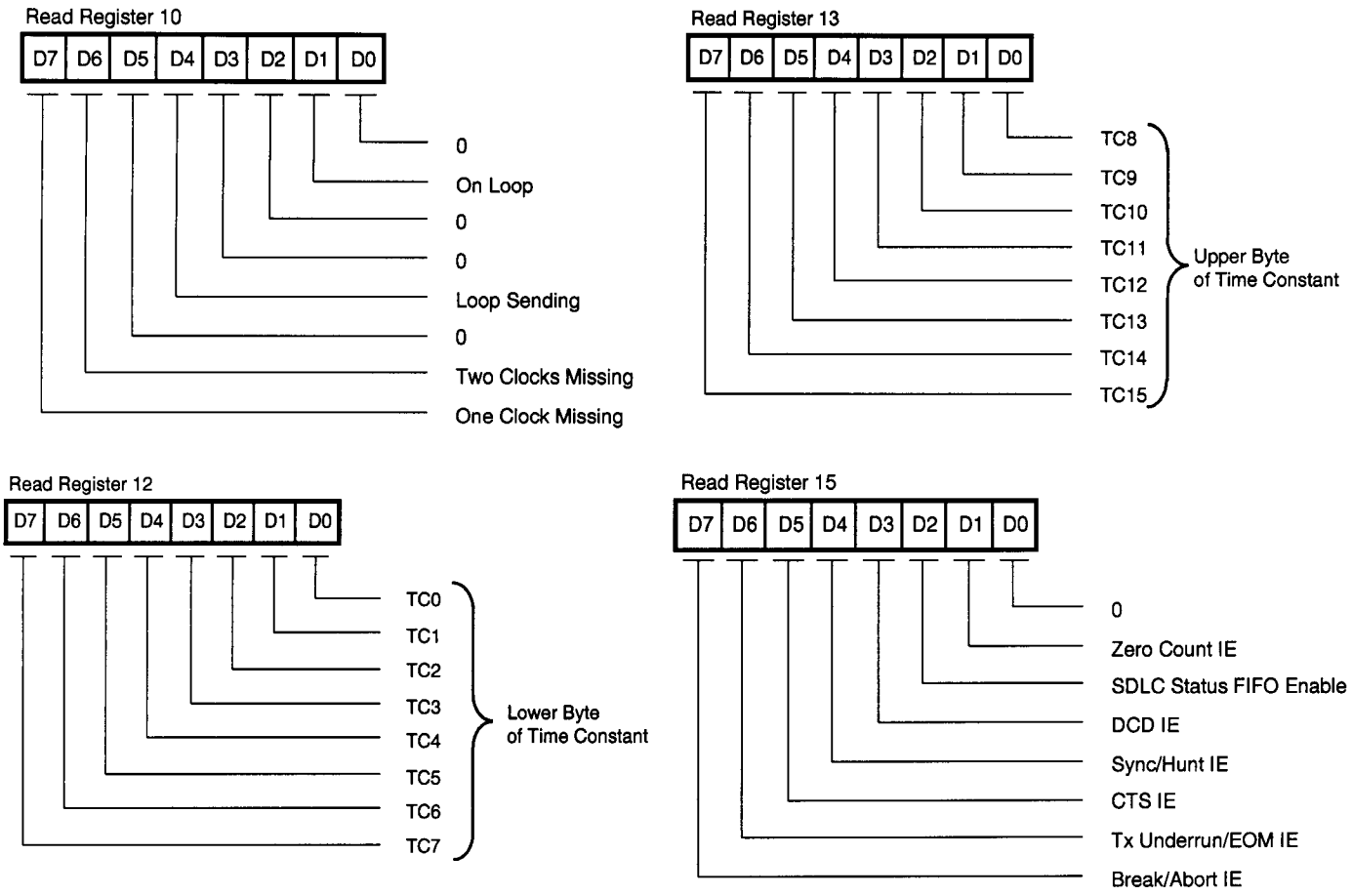


Figure 73. Read Register Bit Functions (Continued)

Z80185 BIDIRECTIONAL CENTRONICS P1284 CONTROLLER (Continued)

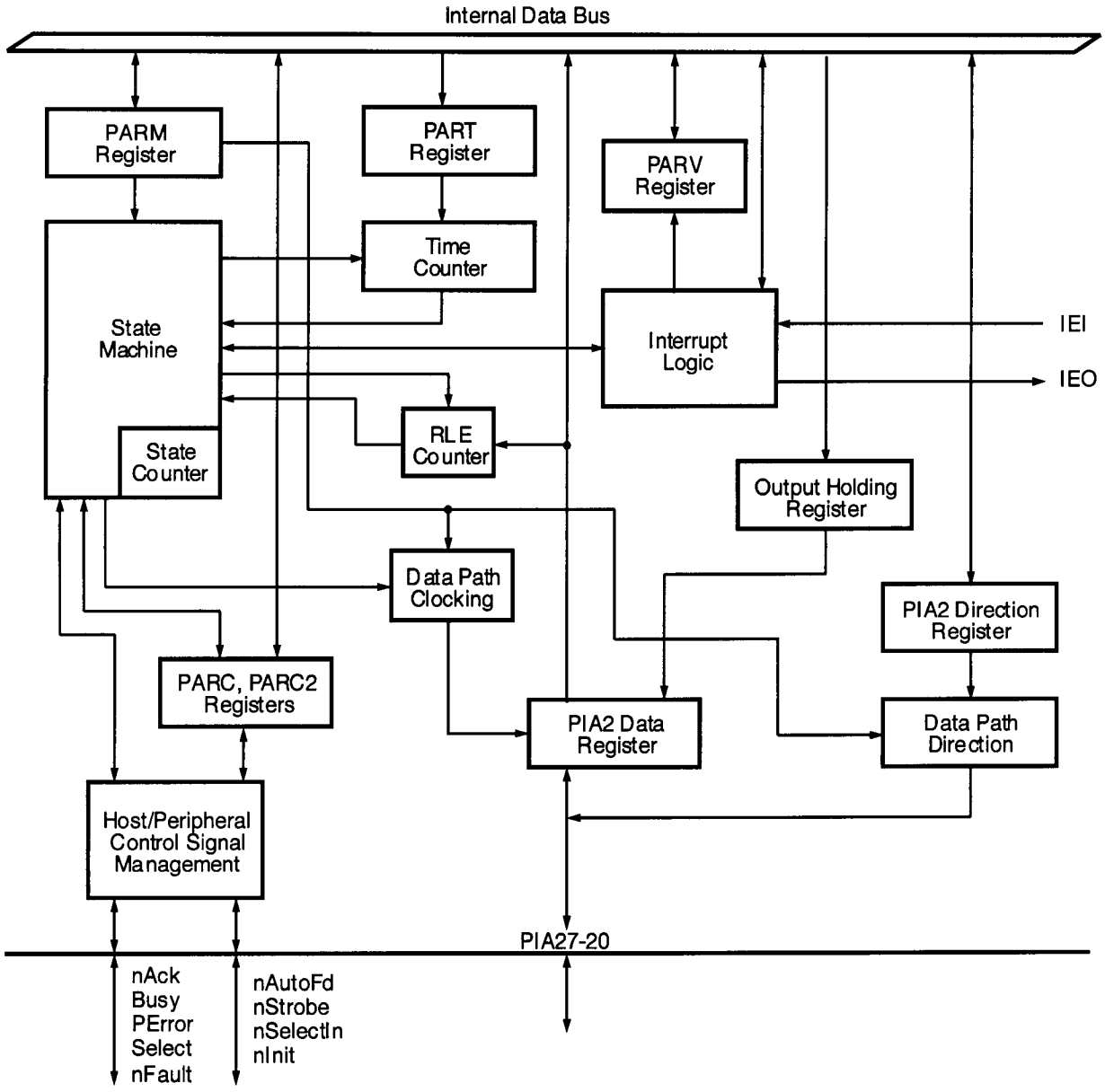


Figure 81. Bidirectional Centronics P1284 Controller Functional Block Description

Z80185 BIDIRECTIONAL CENTRONICS P1284 CONTROLLER (Continued)

Peripheral Nibble Mode

1. Software shouldn't set this mode until there is reverse data available to send. In other words, it should implement the P1284 "reverse idle mode" via software in Peripheral Compatibility/Negotiation mode. After software has driven nDataAvail (nFault), AckDataReq (PError), and Xflag (Select) all Low to signify that data is available, then driven PtrClk (nAck) High after 500 ns, and if requested programmed a DMA channel to provide data to send, when it sees HostBusy (nAutoFd) Low to request data, software should set this mode.

Setting this mode sets DREQ and Idle, but these settings do not request an interrupt. The PIA27-20 pins remain configured for data input but are not used. Instead, four of the five control outputs are driven with the LS and MS four bits of the Input/Output Register, as shown in Table 2, while PtrClk (nAck) serves as a handshake/clock output. On entering this mode the hardware begins routing bits 3-0 of the IOR to these lines.

2. If software, or a DMA channel, writes a byte to the Output Holding Register when Idle is set, the controller immediately transfers the byte to the IOR and clears Idle, and negates DREQ only momentarily to request another byte from software or the DMA channel.

3. After data has been valid on the four control outputs for 500 ns (as controlled by the PART register), the controller drives the PtrClk (nAck) line Low. Then it waits for the host to drive the HostBusy (nAutoFd) line back to High, after which it drives PtrClk (nAck) back to High, switches the four control lines to bits 7-4 of the IOR, and begins waiting for the host to drive HostBusy (nAutoFd) back to Low. When bits 7-4 have been valid for 500 ns and the host has driven HostBusy (nAutoFd) Low, the controller drives PtrClk (nAck) Low again and begins waiting for the host to drive HostBusy (nAutoFd) High. When HostBusy (nAutoFd) has been driven High, the controller returns the four control outputs to the state set by software in PARC. At this point, if software or a DMA channel has not yet written another byte to the Output Holding Register (thus clearing DREQ), the controller sets Idle and waits for software to do so. If/when software or a DMA channel has written a new byte to the OHR, the controller transfers the byte to the IOR, sets DREQ, and clears Idle if it had been set. Then, when the control outputs have been valid for 500 ns, the controller drives PtrClk (nAck) to High. It then waits for the host to drive HostBusy (nAutoFd) back to Low, at which time it switches the four control lines back to bits 3-0 of the IOR and returns to the event sequence at the start of this paragraph.

If there is no more data to send, when the controller sets Idle, software should modify PARC to make nDataAvail (nFault) and AckDataReq (PError) High, and then change the mode to Peripheral Compatible/Negotiation. Then (after 500 ns) software should set PtrClk (nAck) back to High in PARC and enter Reverse Idle state.

Status interrupts in Peripheral Nibble mode include rising and falling edges on P1284Active (nSelectIn) and nInit. The controller sets the IllOp (Illegal Operation) bit if P1284Active (nSelectIn) goes Low in this mode, before it drives nAck High for the status states on the four control lines, or after the host drives HostBusy Low thereafter, in which case software should immediately enter Peripheral Compatibility/Negotiation mode. If P1284Active goes Low, but IllOp stays 0, indicating that the Host negated P1284Active in a legitimate manner, software should enter Peripheral Inactive mode for the duration of the "return to Compatibility mode", and then enter Peripheral Compatibility/Negotiation mode.

Host Byte Mode

1. When in Host Negotiation mode the software has presented the value hex 01 or 05 on PIA27-20, it has been acknowledged by the peripheral, and the peripheral has driven nDataAvail (nFault) and AckDataReq (PError) to Low to indicate data availability and then driven PtrClk (nAck) back to High, software should set this mode. This sets PIA27-20 as inputs regardless of the contents of register E2, and clears the Idle flag. The controller then waits 500 ns (as controlled by the PART register) before proceeding.

2. For each byte, the controller drives HostBusy (nAutoFd) Low to indicate readiness for a byte from the peripheral. Then it waits for PtrClk (nAck) to go Low, at which time it captures the state of PIA27-20 into the Input/Output Register; sets the DREQ bit to request software, or the DMA channel to take the byte, and drives HostBusy (nAutoFd) High and HostClk (nStrobe) Low. When software, or the DMA channel, has taken the byte (thus clearing DREQ) and the peripheral has driven PtrClk (nAck) back High, and at least 500 ns after driving HostClk (nStrobe) Low, the controller drives HostClk (nStrobe) back to High, and samples nDataAvail (nFault). If it is still Low, the controller returns to the event sequence at the start of this paragraph, otherwise it sets the Idle flag.

Peripheral ECP Reverse Mode

1. In this mode, as long as nReverseRequest (nInit) is Low, and P1284Active (nSelectIn) is High, the controller drives the contents of the Input/Output Register onto PIA27-20, regardless of the contents of the E2 register. On entry to this mode, the controller sets Idle, and sets DREQ to request data from software, or a DMA channel.

2. If software, or a DMA channel, writes data to the Output Holding Register while the Input/Output Register is empty, the controller immediately transfers the byte to the IOR, clears Idle, and negates DREQ only momentarily, to request another byte.

3. In this mode, an alternate address for the Output Holding Register allows software to send a "channel address" or an RLE count value. Such bytes are not typically written by a DMA channel. Writing to this alternate address loads the OHR and clears DREQ, the same as writing to the primary address, but clears a ninth bit set when software, or a DMA channel, writes to the primary address. A similar ninth bit is associated with the IOR, and drives the PeriphAck (Busy) line in this mode.

4. As each nine bits arrive in the IOR, and thus out onto PIA27-20 and PeriphAck (Busy), the controller waits one PHI clock, and then drives PeriphClk (nAck) Low. It then waits for the host to drive HostAck (nAutoFd) High, after which it drives PeriphClk (nAck) back to High. The controller then waits for the host to drive HostAck (nAutoFd) back to Low. When this has happened, if software, or the DMA channel, has written a new byte to the Output Holding Register, and thus cleared DREQ, the controller transfers the byte to the IOR, sets DREQ again, and returns to the start of the event sequence in this paragraph. Otherwise, it returns to the event sequence at the start of paragraph #2. If software, or the DMA channel, doesn't provide new data within the time indicated by the PART register, the controller sets the Idle bit.

5. While this mode is in effect, software should monitor whether the host drives nReverseRequest (nInit) High. If it detects this, it should set the mode back to Peripheral ECP Forward, wait 500 ns and then drive nAckReverse (PError) back to High, before proceeding as described for Peripheral ECP Forward mode above.

6. Status interrupts in Peripheral ECP Reverse mode include rising and falling edges on P1284Active (nSelectIn) and nReverseRequest (nInit). Since there are no "legal terminations" during the time this mode is set, the controller sets IllOp for any falling edge on P1284Active (nSelectIn) in this mode.

Table 6. Data Bus Direction (Z185 Bus Master)

I/O Memory Transactions

	I/O Write To On-Chip Peripherals	I/O Read From On-Chip Peripherals	I/O Write to Off-Chip Peripherals	I/O Read from Off-Chip Peripherials	Write to Memory	Read From On-Chip ROM	Read From Off-Chip Memory	Refresh	Z80185 Idle Mode
Z80185 DATA BUS (ROME=0)	Out	Z	Out	In	Out	Z	In	Z	Z
Z80185 Data Bus (ROME=1)	Out	Out	Out	In	Out	Out	In	Z	Z

Interrupt Acknowledge Transaction

	Intack for On-Chip Peripheral	Intack for Off-Chip Peripheral
Z80185 DATA BUS (ROME=0)	Z	In
Z80185 Data Bus (ROME=1)	Out	In

Table 7. Data Bus Direction (Z185 *Is Not* Bus Master)

I/O Memory Transactions

	I/O Write To On-Chip Peripherals	I/O Read From On-Chip Peripherals	I/O Write from Off-Chip Peripherals	I/O Read to Off-Chip Peripherials	Write to Memory	Read From On-Chip ROM	Read From Off-Chip Memory	Refresh	Ext. Bus Master is Idle
Z80185 DATA BUS (ROME=0)	In	Out	Z	Z	Z	Out	In	Z	Z
Z80185 Data Bus (ROME=1)	In	Out	Z	Z	Z	Out	In	Z	Z

Interrupt Acknowledge Transaction

	Intack for On-Chip Peripheral	Intack for Off-Chip Peripheral
Z80185 DATA BUS (ROME=0)	Out	In
Z80185 Data Bus (ROME=1)	Out	In

Notes: "Out" means that the Z185 data bus direction is in output mode; "In" means input mode, and "Z" means High impedance. ROME stands for ROM Emulator mode and is the status of the D2 bit in the System Configuration Register.

Wait State Generation (WSG)

The Memory Wait Insertion field of the DCNTL register applies to all accesses to memory, and allows insertion of 0-3 wait states. In the Z80185, the WSG Chip Select Register allows individual wait state control for the various types and areas of memory.

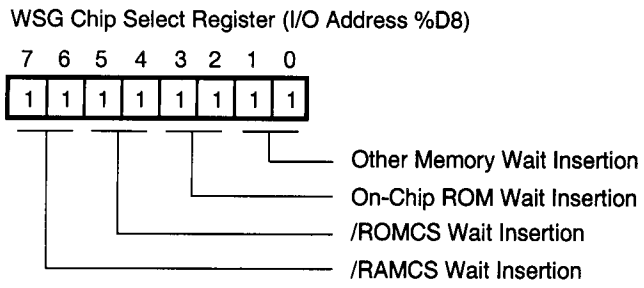


Figure 86. WSG Chip Select Register
(I/O Address %D8)

Bits 7-6. This field controls how many wait states are inserted for accesses to external memory in which /RAMCS is asserted: 00 = none, 01 = 1, 10 = 2, 11 = 4 wait states.

Bits 5-4. This field controls how many wait states are inserted for accesses to external memory in which /ROMCS is asserted, and is encoded like bits 7-6.

Bits 3-2. This field controls how many wait states are inserted for accesses to on-chip ROM, and is encoded like bits 7-6. **Note:** On-chip ROM should be fast enough to support no-wait-state operation at the maximum specified clock rate, but this field is included as a "hedge" against difficulties in this area, as well as to provide timing compatibility in unusual circumstances.

Bits 1-0. This field controls how many wait states are inserted for accesses to external memory in which neither /RAMCS nor /ROMCS is asserted, and is encoded the same as bits 7-6.

All fields in this register Reset to 11. The 4-wait-state feature is included to allow the use of commodity DRAMs with a clock rate at, or near, the maximum.

Note that this facility, and the one in the DCNTL register, both logically OR into the WAIT signal, to allow this register full control of wait states. Bits 7-6 of DCNTL should be programmed to 00.

Watch-Dog Control Registers

Two registers control WDT operations. These are WDT Master Register (WDTMR; I/O Address F0h) and the WDT Command Register (WDTCR; I/O Address F1h). WDT logic has a "double key" structure to prevent accidental disabling of the WDT.

Enabling the WDT. The WDT is enabled by reset, and setting the WDT Enable Bit (WDTMR7) to 1.

Disabling the WDT. The WDT is disabled by clearing WDT Enable bit (WDTR7) to 0 followed by writing "B1h" to the WDT Command Register (WDTCR; I/O Address F1h).

Clearing the WDT. The WDT can be cleared by writing "4Eh" into the WDTCR.

Watch-Dog Timer Master Register (WDTMR;I/O address F0h). This register controls the activities of the Watch-Dog Timer.

Bit D7. *Watch-Dog Timer Enable* (WDTE). The WDT can be enabled by setting this bit to 1. To disable WDT, write 0 to this bit, followed by writing "B1h" to the WDT Command Register. Upon Power-On Reset, this bit is set to 1 and the WDT is enabled.

Bit D6-D5. *WDT Periodic field* (WDTP). This 2-bit field determines the desired time period. Upon Power-on reset, this field sets to "11".

00 - Period is (TcC * 2^{16})

01 - Period is (TcC * 2^{18})

10 - Period is (TcC * 2^{20})

11 - Period is (TcC * 2^{22})

Bit D4. If this bit is 1 and the WDT times out, the Z80185 drives the /Reset pin Low to reset external logic. If this bit is 0, a WDT timer only resets the Z80185 internally.

Bit D3-D0. *Reserved.* These three bits are reserved and should always be programmed as 0011. Reading these bits returns 0011.

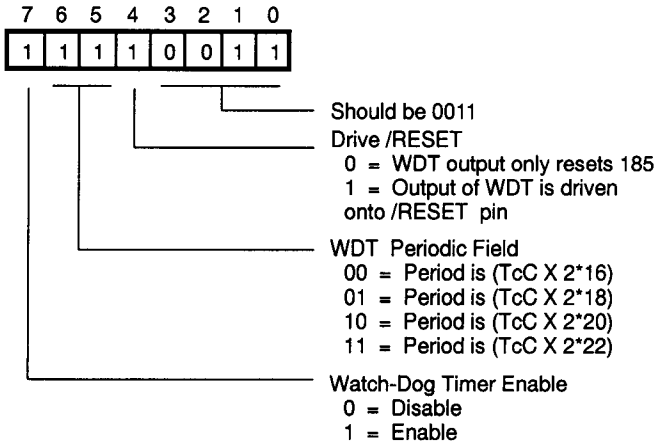


Figure 92. Watch-Dog Timer Master Register
(I/O Address %F0)

Watch-Dog Timer Command Register (WDTCR; I/O Address F1h). This register is Write Only (Figure 93).

Write B1h after clearing WDTE to "0" - Disable WDTWrite 4Eh - Clear WDT

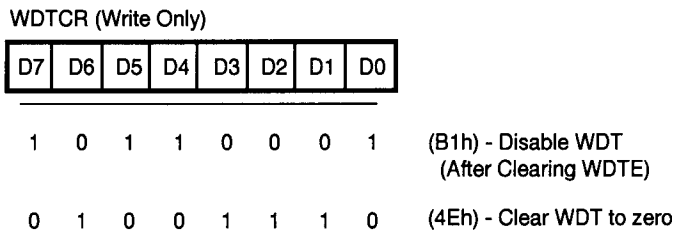


Figure 93. Watch-Dog Timer Command Register

Z80185 CTC, AND MISCELLANEOUS REGISTERS (Continued)

Parallel Ports

The Z80185 has two 8-bit bidirectional ports. Each bit is individually programmable for input or output. Each port includes two registers: the Port Direction Control Register and the Port Data Register. The second port also includes an Alternate Address that is used with the Bidirectional Centronics feature.

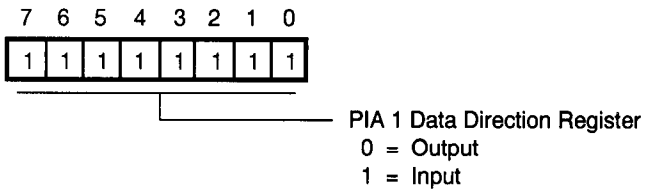


Figure 94. PIA 1 Data Direction Register (I/O Address %E0)

The data direction register determines which of the PIA16-10 pins are inputs and outputs. When a bit is set to 1, the corresponding bit in the PIA 1 Data Register is an input. If the bit is 0, then the corresponding pin is an output. These bits must be set appropriately if these pins are used for CTC inputs and outputs.

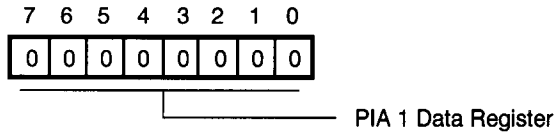


Figure 95. PIA 1 Data Register (I/O Address %E1)

When the processor writes to the PIA 1 Data Register, the data is stored in the internal buffer. Any bits that are output are then driven on to the pins.

When the processor reads the PIA 1 Data Register, the data on the external pins is returned.

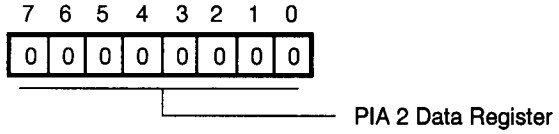


Figure 96. PIA 2 Data Direction Register (I/O Address %E2)

The data direction register determines which of the PIA27-20 pins are inputs and outputs. When a bit is set to a 1, the corresponding pin is an input. If the bit is 0, then the corresponding bit is an output. These settings can be overridden by the Bidirectional Centronics Controller.

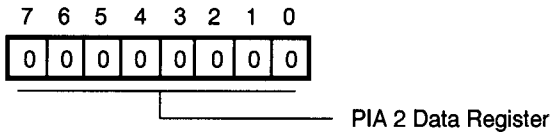


Figure 97. PIA 2 Data Register (I/O Address %E3)

When the processor writes to the PIA 2 Data Register, the data is stored in the internal buffer. Any bits that are output are then driven on to the pins. In certain modes of the Bidirectional Centronics Controller, an intermediate register called the Output Holding Register is activated, and the transfer of data from the OHR to the pins is under the control of the controller.

When the processor reads the PIA 2 Data Register, the data on the external pins is returned. In certain modes of the Bidirectional Centronics Controller, reading from this address reads the data stored in the port register from PIA27-20 under the control of the controller.

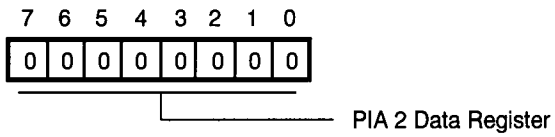


Figure 98. PIA 2 Data Alternate Address (RW) (I/O Address %EE)

Reading and writing this register is exactly the same as reading and writing address E3 as described above, except that in certain modes of the Bidirectional Centronics Controller, writing to this address sets a "ninth bit" in the opposite sense from writing address E3, and this drives one of the control outputs with the opposite polarity.

ELECTRICAL CHARACTERISTICS

The following classification table describes pins in terms of input and output classes. V_{DD} = 5V ±10%, unless otherwise noted.

Pin Input/Output Classification

Class "O" output:	Full time / totem pole
	V_{OL} 0.4V max at I_{OL} = 2.0 mA
	$V_{OH} = V_{DD} - 1.2V$ min at I_{OL} = 200 µA
	Slew rate 0.33 V/ns min at C_{LOAD} = 50 pF
	C_{OUT} = 15 pF max (output or I/O)
Class "3" output:	As "O" except tri-state.
Class "H" output:	As "O" except $V_{OH} = V_{DD}$-0.6V min at I_{OH} = 200 µA
Class "D" output:	Open Drain
	V_{OL} 0.4V max at I_{OL} = 12 mA
	C_{OUT} = 15 pF max (output or I/O)
Class "T" output:	Tri-State
	As Class "O" at V_{DD} = 3.3V ±10%
	V_{OL} 0.4V max at I_{OL} = 12 mA, V_{DD} = 5V ±10%
	V_{OH} 2.4V min at I_{OL} = 12 mA, V_{DD} = 5V ±10%
	Output impedance 45 ohms max
	Slew rate 0.05 - 0.40 V/ns (C_{LOAD} not stated by IEEE)
	C_{OUT} =15 pF max (output or I/O)
Class "I" input":	V_{IL} 0.8V max at V_{DD} = 5V ±10%
	V_{IL} 0.6V max at V_{DD} = 3.3V ±10%
	V_{IH} 2.0V min
	Ii ±10 µA max, Vi = 0 to 5V (includes leakage if I/O)
	C_{IN} = 5 pF max (if input only, see output type if I/O)
	Inputs of this type include Weak Latch circuits.
Class "R" input:	V_{IL} 0.6V max
	V_{IH} V_{DD}-0.6 min at V_{DD} = 5V ±10%
	V_{IH} V_{DD}-0.3 min at V_{DD} = 3.3V ±10%
	Ii ±10 µA max, Vi = 0 to 5V
	C_{IN} = 5 pF max
Class "S" input:	V_{IL} 0.8V max at V_{DD} = 5V ±10%
	V_{IL} 0.6V max at V_{DD} = 3.3V ±10%
	V_{IH} 2.0V min
	Hysteresis 0.2V min
	Ii ±20 µA max, Vi=0.8 to 2V (includes leakage if I/O)
	Inputs of this type include Weak Latch circuits.