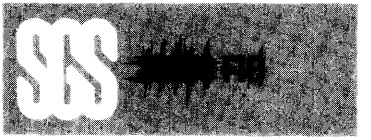
# FIFO Input/Output Interface Unit



### Features

- 128-byte FIFO buffer provides asynchronous bidirectional CPU/CPU or CPU/peripheral interface, expandable to any width in byte increments by use of multiple Z8060 FIO's
- Interlocked 2-Wire or 3-Wire Handshake logic port mode; Z-BUS or non-Z-BUS interface.
- Pattern-recognition logic stops DMA transfers and/or interrupts CPU; preset byte count can initiate variable-length DMA transfers.
- Seven sources of vectored/nonvectored interrupt which include pattern-match, byte count, empty or full buffer status; a dedicated "mailbox" register with interrupt capability provides CPU/CPU communication.
- $\overline{\text{REQUEST}}/\overline{\text{WAIT}}$ lines control high-speed data transfers.
- All functions are software controlled via directly addressable read/write registers.

### General Description

The Z8038 FIO provides an asynchronous 128-byte FIFO buffer between two CPUs or between a CPU and a peripheral device. This buffer interface expands to a 16-bit or wider data path and expands in depth to add as many Z8060 FIFOs (and an additional FIO) as are needed.

The FIO manages data transfers by assuming Z-BUS, non-Z-BUS microprocessor (a generalized microprocessor interface), Interlocked 2-Wire Handshake, and 3-Wire Handshake operating modes. These modes interface dissimilar CPUs or CPUs and peripherals running under differing speeds or protocols, allowing asynchronous data transactions and improving I/O overhead by as much as two orders of magnitude. Figures 1 and 2 show how the signals controlling these operating modes are mapped to the FIO pins.

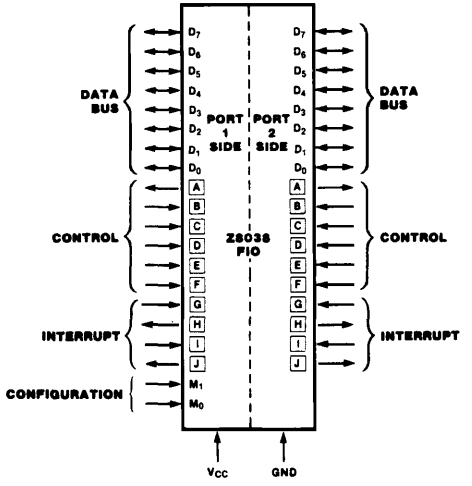



Figure 1. Logic Functions

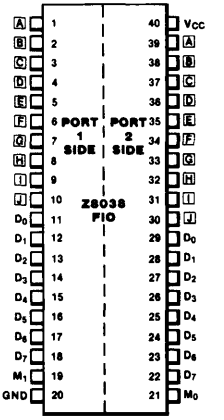



Figure 2. Pin Configuration

**General Description** (Continued)

The FIO supports the Z-BUS interrupt protocols, generating seven sources of interrupts upon any of the following events: a write to a message register, change in data direction, pattern match, status match, over/underflow error, buffer full and buffer empty status. Each interrupt source can be enabled or disabled, and can also place an interrupt vector on the port address/data lines.

The data transfer logic of the FIO has been specially designed to work with DMA (Direct Memory Access) devices for high-speed transfers. It provides for data transfers to or from memory each machine cycle, while the DMA device generates memory address and control signals. The FIO also supports the variably sized block length, improving system throughput when multiple variable length messages are transferred amongst several sources.

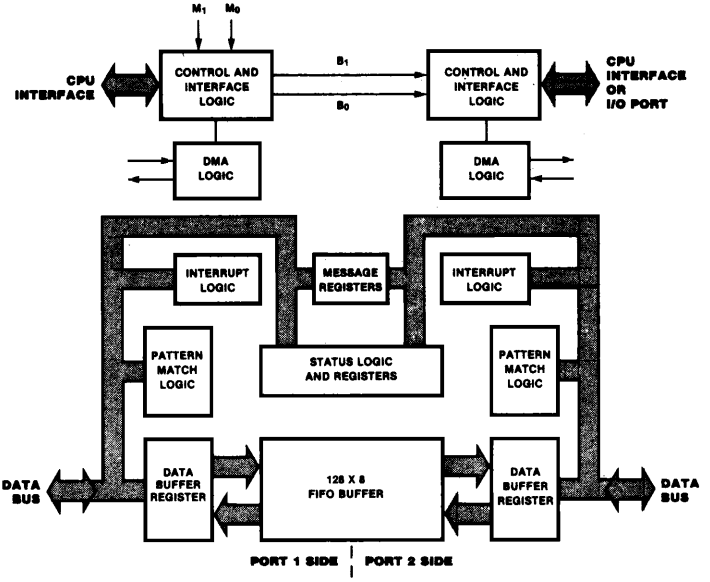



**Figure 3. FIO Block Diagram**

**Functional Description**

**Operating Modes.** Ports 1 and 2 operate in any of twelve combinations of operating modes, listed in Table 2. Port 1 functions in either the Z-BUS or non-Z-BUS microprocessor modes, while Port 2 functions in Z-BUS, non-Z-BUS, Interlocked 2-Wire Handshake, and 3-Wire Handshake modes. Table 1 describes the signals and their corresponding pins in each of these modes.

The pin diagrams of the FIO are identical, except for two pins on the Port 1 side, which select that port's operating mode. Port 2's operating mode is programmed by two bits in Port 1's Control register 0. Table 2 describes the combinations of operating modes; Table 3 describes the control signals mapped to pins A-J in the five possible operating modes.

## Functional Description (Continued)

| Control Signal Pins | Z-BUS Low Byte | Z-BUS High Byte | Non-Z-BUS | Interlocked HS Port* | 3-Wire HS Port* |
|---|---|---|---|---|---|
| A | REQ/WT | REQ/WT | REQ/WT | RFD/DAV | RFD/DAV |
| B | DMASTB | DMASTB | DACK | ACKIN | DAV/DAC |
| C | DS | DS | RD | FULL | DAC/RFD |
| D | R/W | R/W | WR | EMPTY | EMPTY |
| E | CS | CS | CE | CLEAR | CLEAR |
| F | AS | AS | C/D | DATA DIR | DATA DIR |
| G | INTACK | $A_0$ | INTACK | $IN_0$ | $IN_0$ |
| H | IEO | $A_1$ | IEO | $OUT_1$ | $OUT_1$ |
| I | IEI | $A_2$ | IEI | OE | OE |
| J | INT | $A_3$ | INT | $OUT_3$ | $OUT_3$ |

*2 side only.

**Table 1. Pin Assignments**

| Mode | $M_1$ | $M_0$ | $B_1$ | $B_0$ | Port 1 | Port 2 |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | Z-BUS Low Byte | Z-BUS Low Byte |
| 1 | 0 | 0 | 0 | 1 | Z-BUS Low Byte | Non-Z-BUS |
| 2 | 0 | 0 | 1 | 0 | Z-BUS Low Byte | 3-Wire Handshake |
| 3 | 0 | 0 | 1 | 1 | Z-BUS Low Byte | 2-Wire Handshake |
| 4 | 0 | 1 | 0 | 0 | Z-BUS High Byte | Z-BUS High Byte |
| 5 | 0 | 1 | 0 | 1 | Z-BUS High Byte | Non-Z-BUS |
| 6 | 0 | 1 | 1 | 0 | Z-BUS High Byte | 3-Wire Handshake |
| 7 | 0 | 1 | 1 | 1 | Z-BUS High Byte | 2-Wire Handshake |
| 8 | 1 | 0 | 0 | 0 | Non-Z-BUS | Z-BUS Low Byte |
| 9 | 1 | 0 | 0 | 1 | Non-Z-BUS | Non-Z-BUS |
| 10 | 1 | 0 | 1 | 0 | Non-Z-BUS | 3-Wire Handshake |
| 11 | 1 | 0 | 1 | 1 | Non-Z-BUS | 2-Wire Handshake |

**Table 2. Operating Modes**

**Functional Description** (Continued)

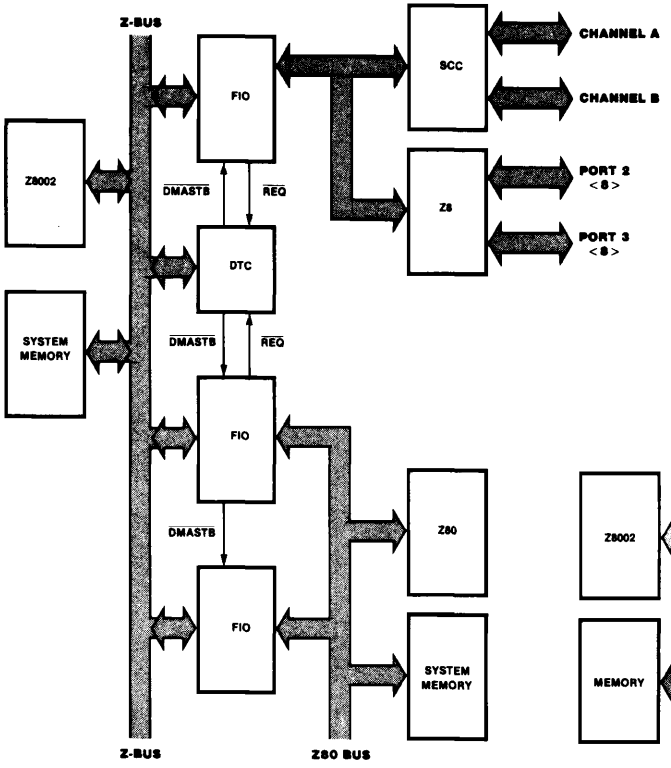



Figure 4. CPU to CPU Configuration

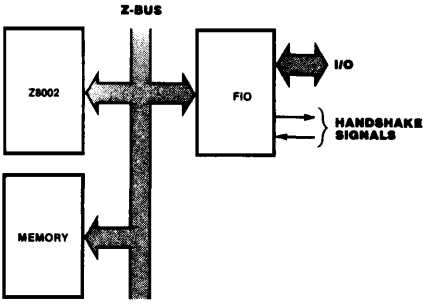



Figure 5. CPU to I/O Configuration

## Pins Common To Both Sides

| Pin Signals | Pin Names | Pin Numbers | Signal Description |
|---|---|---|---|
| $M_0$ | $M_0$ | 21 | $M_1$ and $M_0$ program Port 1 side CPU interface |
| $M_1$ | $M_1$ | 19 | |
| +5 Vdc | +5 Vdc | 40 | DC power source |
| GND | GND | 20 | DC power ground |

## Z-BUS Low Byte Mode

| Pin Signals | Pin Names | Pin Numbers Port 1 | Pin Numbers Port 2 | Signal Description |
|---|---|---|---|---|
| $AD_0$-$AD_7$ (Address/Data) | $D_0$-$D_7$ | 11-18 | 29-22 | Multiplexed bidirectional address/data lines, Z-BUS compatible. |
| $\overline{REQ}/\overline{WAIT}$ (Request/Wait) | A | 1 | 39 | Output, active Low, REQUEST (ready) line for DMA transfer; $\overline{WAIT}$ line (open-drain) output for synchronized CPU and FIO data transfers. |
| $\overline{DMASTB}$ (Direct Memory Access Strobe) | B | 2 | 38 | Input, active Low. Strobes DMA data to and from the FIFO buffer. |
| $\overline{DS}$ (Data Strobe) | C | 3 | 37 | Input, active Low. Provides timing for data transfer to or from FIO. |
| $R/\overline{W}$ (Read/Write) | D | 4 | 36 | Input; active High signals CPU read from FIO; active Low signals CPU write to FIO. |
| $\overline{CS}$ (Chip Select) | E | 5 | 35 | Input, active Low. Enables FIO. Latched on the rising edge of $\overline{AS}$. |
| $\overline{AS}$ (Address Strobe) | F | 6 | 34 | Input, active Low. Addresses, $\overline{CS}$ and $\overline{INTACK}$ sampled while $\overline{AS}$ Low. |
| $\overline{INTACK}$ (Interrupt Acknowledge) | G | 7 | 33 | Input, active Low. Acknowledges an interrupt. Latched on the rising edge of $\overline{AS}$. |
| IEO (Interrupt Enable Out) | H | 8 | 32 | Output, active High. Sends interrupt enable to lower priority device IEI pin. |
| IEI (Interrupt Enable In) | I | 9 | 31 | Input, active High. Receives interrupt enable from higher priority device IEO signal. |
| $\overline{INT}$ (Interrupt) | J | 10 | 30 | Output, open drain, active Low. Signals FIO interrupt request to CPU. |

**Table 3. Signal/Pin Descriptions**

## Z-BUS High Byte Mode

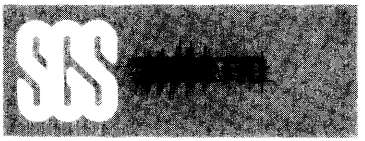
| Pin Signals | Pin Names | Pin Numbers Port 1 | Pin Numbers Port 2 | Signal Description |
|---|---|---|---|---|
| $AD_0$-$AD_7$ (Address/Data) | $D_0$-$D_7$ | 11-18 | 29-22 | Multiplexed bidirectional address/data lines, Z-BUS compatible. |
| $\overline{REQ}/\overline{WAIT}$ (Request/Wait) | A | 1 | 39 | Output, active Low, REQUEST (ready) line for DMA transfer; $\overline{WAIT}$ line (open-drain) output for synchronized CPU and FIO data transfers. |
| $\overline{DMASTB}$ (Direct Memory Access Strobe) | B | 2 | 38 | Input, active Low. Strobes DMA data to and from the FIFO buffer. |
| $\overline{DS}$ (Data Strobe) | C | 3 | 37 | Input, active Low. Provides timing for transfer of data to or from FIO. |
| $R/\overline{W}$ (Read/Write) | D | 4 | 36 | Input, active High. Signals CPU read from FIO; active Low signals CPU write to FIO. |
| $\overline{CS}$ (Chip Select) | E | 5 | 35 | Input, active Low. Enables FIO. Latched on the rising edge of $\overline{AS}$. |
| $\overline{AS}$ (Address Strobe) | F | 6 | 34 | Input, active Low. Addresses, $\overline{CS}$ and $\overline{INTACK}$ are sampled while $\overline{AS}$ is Low. |
| $A_0$ (Address Bit 0) | G | 7 | 33 | Input, active High. With $A_1$, $A_2$, and $A_3$, addresses FIO internal registers. |
| $A_1$ (Address Bit 1) | H | 8 | 32 | Input, active High. With $A_0$, $A_2$, and $A_3$, addresses FIO internal registers. |
| $A_2$ (Address Bit 2) | I | 9 | 31 | Input, active High. With $A_0$, $A_1$, and $A_3$, addresses FIO internal registers. |
| $A_3$ (Address Bit 3) | J | 10 | 30 | Input, active High. With $A_0$, $A_1$, and $A_2$, addresses FIO internal registers. |

**Table 3. Signal/Pin Descriptions** (Continued)

## Non-Z-BUS Mode

| Pin Signals | Pin Names | Pin Numbers Port 1 | Port 4 | Signal Description |
|---|---|---|---|---|
| $D_0$-$D_7$ (Data) | $D_0$-$D_7$ | 11-18 | 29-22 | Bidirectional data bus. |
| $\overline{REQ}/\overline{WT}$ (Request/Wait) | A | 1 | 39 | Output, active Low, REQUEST (ready) line for DMA transfer; $\overline{WAIT}$ line (open-drain) output for synchronized CPU and FIO data transfer. |
| $\overline{DACK}$ (DMA Acknowledge) | B | 2 | 38 | Input, active Low. DMA acknowledge. |
| $\overline{RD}$ (Read) | C | 3 | 37 | Input, active Low. Signals CPU read from FIO. |
| $\overline{WR}$ (Write) | D | 4 | 36 | Input, active Low. Signals CPU write to FIO. |
| $\overline{CE}$ (Chip Select) | E | 5 | 35 | Input, active Low. Used to select FIO. |
| C/$\overline{D}$ (Control/Data) | F | 6 | 34 | Input, active High. Identifies control byte on $D_0$-$D_7$; active Low identifies data byte on $D_0$-$D_7$. |
| $\overline{INTACK}$ (Interrupt Acknowledge) | G | 7 | 33 | Input, active Low. Acknowledges an interrupt. |
| IEO (Interrupt Enable Out) | H | 8 | 32 | Output, active High. Sends interrupt enable to lower priority device IEI pin. |
| IEI (Interrupt Enable In) | I | 9 | 31 | Input, active High. Receives interrupt enable from higher priority device IEO signal. |
| $\overline{INT}$ (Interrupt) | J | 10 | 30 | Output, open drain, active Low. Signals FIO interrupt to CPU. |

**Table 3. Signal/Pin Descriptions** (Continued)



**Port 2 – I/O Port Mode**

| Pin Signals | Pin Names | Pin Numbers | Mode | Signal Description |
|---|---|---|---|---|
| $D_0$-$D_7$ (Data) | $D_0$-$D_7$ | 29-22 | 2-Wire HS* 3-Wire HS | Bidirectional data bus. |
| RFD/$\overline{DAV}$ (Ready for Data/Data Available) | A | 39 | 2-Wire HS 3-Wire HS | Output, RFD active High. Signals peripherals that FIO is ready to receive data. $\overline{DAV}$ active Low signals that FIO is ready to send data to peripherals. |
| $\overline{ACKIN}$ (Acknowledge Input) | B | 38 | 2-Wire HS | Input, active Low. Signals FIO that output data is received by peripherals or that input data is valid. |
| $\overline{DAV}$/DAC (Data Available/Data Accepted) | B | 38 | 3-Wire HS | Input; $\overline{DAV}$ (active Low) signals that data is valid on bus. DAC (active High) signals that output data is accepted by peripherals. |
| FULL | C | 37 | 2-Wire HS | Output, open drain, active High. Signals that FIO buffer is full. |
| DAC/RFD (Data Accepted/Ready for Data) | C | 37 | 3-Wire HS | Direction controlled by internal programming. Both active High. DAC (an output) signals that FIO has received data from peripheral; RFD (an input) signals that the listeners are ready for data. |
| EMPTY | D | 36 | 2-Wire HS 3-Wire HS | Output, open drain, active High. Signals that FIFO buffer is empty. |
| $\overline{CLEAR}$ | E | 35 | 2-Wire HS 3-Wire HS | Programmable input or output, active Low. Clears all data from FIFO buffer. |
| DATA DIR (Data Direction) | F | 34 | 2-Wire HS 3-Wire HS | Programmable input or output. Active High signals data input to Port 2; Low signals data output from Port 2. |
| $IN_0$ | G | 33 | 2-Wire HS 3-Wire HS | Input line to $D_0$ of Control Register 3. |
| $OUT_1$ | H | 32 | 2-Wire HS 3-Wire HS | Output line from $D_1$ of Control Register 3. |
| $\overline{OE}$ (Output Enable) | I | 31 | 2-Wire HS 3-Wire HS | Input, active Low. When Low, enables bus drivers. When High, floats bus drivers at high impedance. |
| $OUT_3$ | J | 30 | 2-Wire HS 3-Wire HS | Output line from $D_3$ of Control register 3. |

*Handshake

**Table 3. Signal/Pin Descriptions** (Continued)

**Reset**

The FIO can be reset under either hardware or software control by one of the following methods:

- By forcing both $\overline{AS}$ and $\overline{DS}$ Low simultaneously in Z-BUS mode (normally illegal).
- By forcing $\overline{RD}$ and $\overline{WR}$ Low simultaneously in non-Z-BUS mode.
- By writing a 1 to the Reset bit in Control register 0 for software reset.

In the Reset state, all control bits are cleared to 0. Only after clearing the Reset bit (by writing a 0 to it) can the other command bits be programmed. This action is true for both sides of the FIO when programmed as a CPU interface.

For proper system control, when Port 1 is reset, Port 2 is also reset. In addition, all Port 2's outputs are floating and all inputs are ignored. To initiate the data transfer, Port 2 must be enabled by Port 1. The Port 2 CPU can determine when it is enabled by reading Control register 0, which reads "floating" data bus if not enabled and "$01_H$" if enabled.

## CPU Interfaces

The FIO is designed to work with both Z-BUS- and non-Z-BUS-type CPUs on both Port 1 and Port 2. The Z-BUS configuration interfaces CPUs with time-multiplexed address and data information on the same pins. The Z8001, Z8002, and Z8 are examples of this type of CPU. The $\overline{AS}$ (Address Strobe) pin is used to latch the address and chip select information sent out by the CPU. The R/$\overline{W}$ (Read/Write) pin and the $\overline{DS}$ (Data Strobe) pin are used for timing reads and writes from the CPU to the FIO (Figures 6 and 7).

The non-Z-BUS configuration is used for CPUs where the address and data buses are separate. Examples of this type of CPU are the Z80 and 8080. The $\overline{RD}$ (Read) and $\overline{WR}$ (Write) pins are used to time reads and writes from the CPU to the FIO (Figures 9 and 10). The C/$\overline{D}$ (Control/Data) pin is used to directly access the FIFO buffer (C/$\overline{D}$ = 0) and to access the other registers (C/$\overline{D}$ = 1). Read and write to all

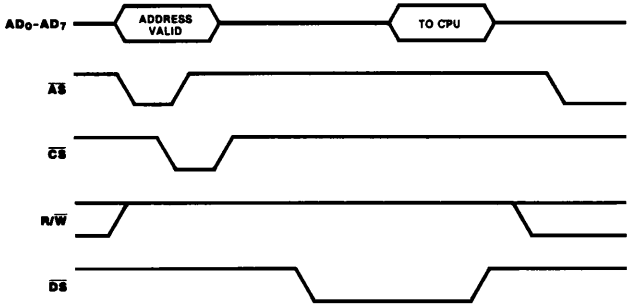



**Figure 6. Z-BUS Read Cycle Timing**

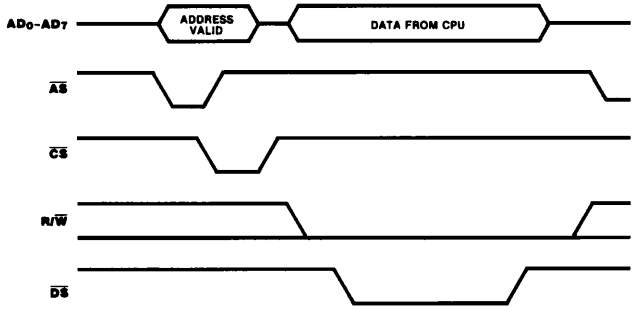



**Figure 7. Z-BUS Write Cycle Timing**

**CPU Interface** (Continued)

registers except the FIFO buffer[1] are two-step operations, described as follows (Figure 8). First, write the address ($C/\overline{D} = 1$) of the register to be accessed into the Pointer Register (State 0); second, read or write ($C/\overline{D} = 1$) to the register pointed at previously (State 1). Continuous status monitoring can be performed in State 1 by continuous Control Read operations ($C/\overline{D} = 1$).

[1]The FIFO buffer can also be accessed by this two-step operation.

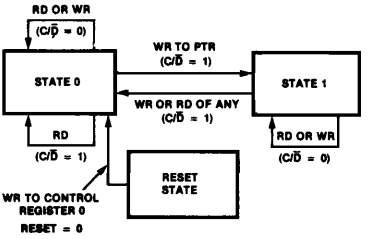



**Figure 8. Register Access in Non-Z-BUS Mode**

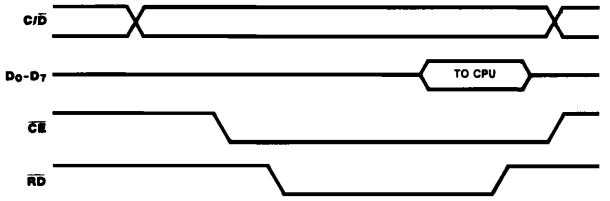



**Figure 9. Non-Z-BUS Read Cycle Timing**

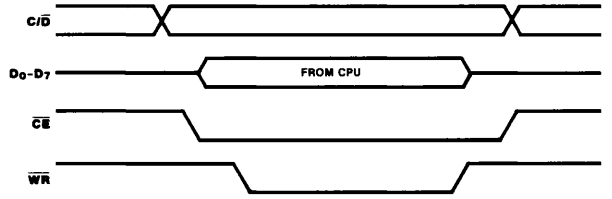



**Figure 10. Non-Z-BUS Write Cycle Timing**

**$\overline{\text{WAIT}}$ Operation**

When data is output by the CPU, the $\overline{\text{REQ}}/\overline{\text{WT}}$ ($\overline{\text{WAIT}}$) pin is active (Low) only when the FIFO buffer is full, the chip is selected, and the FIFO buffer is addressed. $\overline{\text{WAIT}}$ goes inactive when the FIFO buffer is not full.

When data is input by the CPU, the $\overline{\text{REQ}}/\overline{\text{WT}}$ pin becomes active (Low) only when the FIFO buffer is empty, the chip is selected, and the FIFO buffer is addressed. $\overline{\text{WAIT}}$ goes inactive when the FIFO buffer is not empty.

**Interrupt Operation**

The FIO supports the prioritized daisy chain interrupt protocol for both Z-BUS and non-Z-BUS operating modes (for more details refer to the Z-BUS Summary).

Each side of the FIO has seven sources of interrupt. The priorities of these devices are fixed in the following order (highest to lowest): Mailbox Message, Change in Data Direction, Pattern Match, Status Match, Overflow/ Underflow Error, Buffer Full, and Buffer Empty. Each interrupt source has three bits that control how it generates the interrupt. These bits are Interrupt Pending (IP), Interrupt Enable (IE), and Interrupt Under Service (IUS).

In addition, each side of the FIO has an interrupt vector and four bits controlling the FIO interrupt logic. These bits are Vector Includes Status (VIS), Master Interrupt Enable (MIE), Disable Lower Chain (DLC), and No Vector (NV).

A typical Interrupt Acknowledge cycle for Z-BUS operation is shown in Figure 11 and for non-Z-BUS operation in Figure 12. The only difference is that in Z-BUS mode, $\overline{INTACK}$ is latched by $\overline{AS}$, and in non-Z-BUS mode $\overline{INTACK}$ is not latched.

When MIE = 1, reading the vector always includes status, independent of the state of the VIS bit. In this way, when VIS = 0, all information can be obtained with one additional read, thus conserving vector space. When MIE = 0, reading the vector register returns the unmodified base vector so that it can be verified.

In non-Z-BUS mode, IPs do not get set while in State 1. Therefore, in order to minimize interrupt latency, the FIO should be left in State 0.

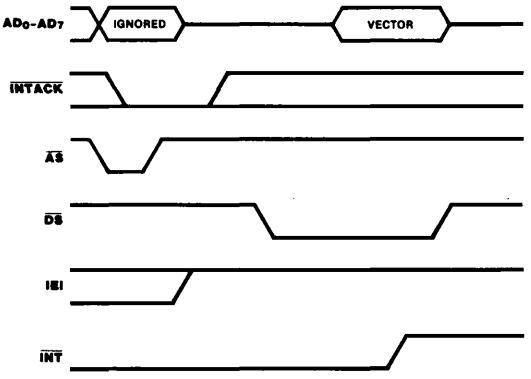



**Figure 11. Z-BUS Interrupt Acknowledge Cycle**

**Interrupt Operation** (Continued)

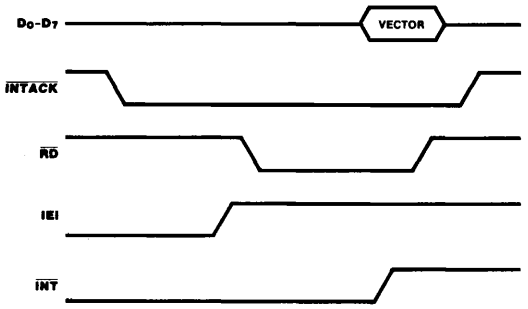



**Figure 12. Non-Z-BUS Interrupt Acknowledge Cycle**

**CPU to CPU Operation**

**DMA Operation.** The FIO is particularly well suited to work with a DMA in both Z-BUS and non-Z-BUS modes. A data transfer between the FIO and system memory can take place during every machine cycle on both sides of the FIO simultaneously.

In Z-BUS mode, the $\overline{\text{DMASTB}}$ pin (DMA Strobe) is used to read or write into the FIFO buffer. The R/$\overline{\text{W}}$ (Read/Write) and $\overline{\text{DS}}$ (Data Strobe) signals are ignored by the FIO; however, the $\overline{\text{CS}}$ (Chip Select) signal is not ignored and therefore must be kept invalid. Figures 13 and 14 show typical timing.

In Non-Z-BUS mode, the $\overline{\text{DACK}}$ pin (DMA Acknowledge) is used to tell the FIO that its DMA request is granted. After $\overline{\text{DACK}}$ goes Low, every read or write to the FIO goes into the FIFO buffer. Figures 15 and 16 show typical timing.

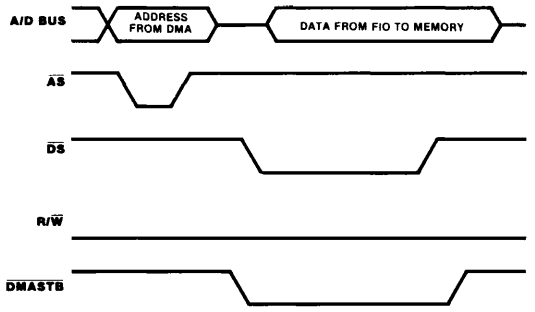



**Figure 13. Z-BUS FIO to Memory Data Transaction**

**CPU to CPU Operation** (Continued)

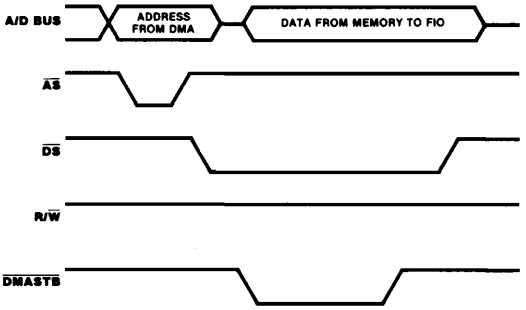



**Figure 14. Z-BUS Memory to FIO Data Transaction**

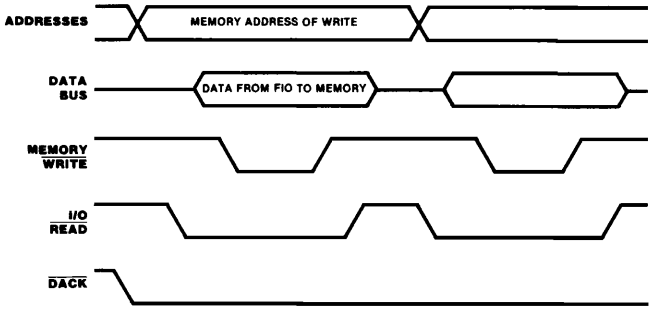



**Figure 15. Non-Z-BUS FIO to Memory Transaction**

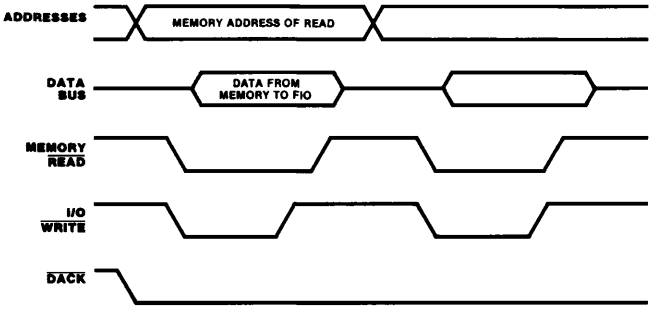


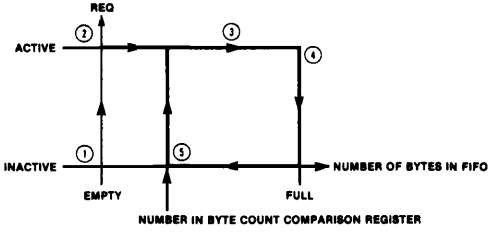

**Figure 16. Non-Z-BUS Memory to FIO Data Transaction**

**CPU to CPU Operation** (Continued)

The FIO provides a special mode to enhance its DMA transfer capability. When data is written into the FIFO buffer, the $\overline{REQ}/\overline{WT}$ (REQUEST) pin is active (Low) until the FIFO buffer is full. It then goes inactive and stays inactive until the number of bytes in the FIFO buffer is equal to the value programmed into the Byte Count Comparison register. Then the REQUEST signal goes active and the sequence starts over again (Figure 17).

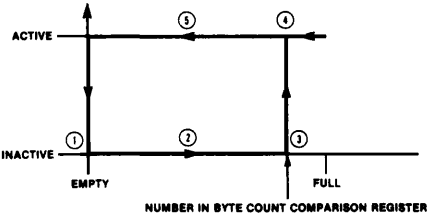
When data is read from the FIO, the $\overline{REQ}/\overline{WT}$ pin (REQUEST) is inactive until the number of bytes in the FIFO buffer is equal to the value programmed in the Byte Count Comparison register. The REQUEST signal then goes active and stays active until the FIFO buffer is empty. When empty, REQUEST goes inactive and the sequence starts over again (Figure 18).

**Message Registers.** Two CPUs can communicate through a dedicated "mailbox" register without involving the 128 × 8 bit FIFO buffer (Figure 19). This mailbox approach is useful for transferring control parameters between the interfacing devices on either side of the FIO without using the FIFO buffer. For example, when Port 1's CPU writes to the Message Out register, Port 2's message IP is set. If interrupts are enabled, Port 2's CPU is interrupted. Port 2's message IP status is readable from the Port 1 side. When Port 2's CPU reads the data from its Message In register, the Port 2 IP is cleared. Thus, Port 1's CPU can read when the message has been read and can now send another message or follow whatever protocol that is set up between the two CPU's. The same transfer can also be made from Port 2's CPU to Port 1's CPU.




NOTES:
1. FIFO empty.
2. REQUEST enabled, FIO requests DMA transfer.
3. DMA transfers data into the FIO.
4. FIFO full, REQUEST inactive.
5. The FIFO empties from the opposite port until the number of bytes in the FIFO buffer is the same as the number programmed in the Byte Count Comparison register.

**Figure 17. Byte Count Control: Write to FIO**




NOTES:
1. FIFO empty.
2. CPU/DMA fills FIFO buffer from the opposite port.
3. Number of bytes in FIFO buffer is the same as the number of bytes programmed in the Byte Count Comparison register.
4. REQUEST goes active.
5. DMA transfers data out of FIFO until it is empty.

**Figure 18. Byte Count Control: Read from FIO**

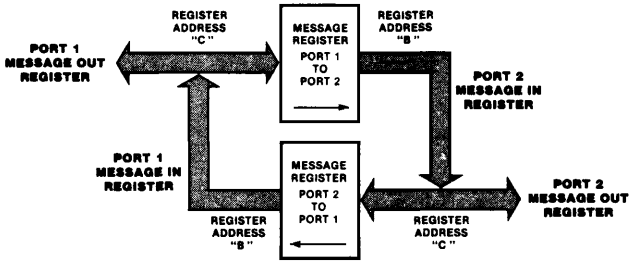
**CPU to CPU Operation** (Continued)

**CLEAR (Empty) FIFO Operation.** The $\overline{\text{CLEAR}}$ FIFO bit (active Low) clears the FIFO buffer of data. Writing a 0 to this bit empties the FIFO buffer, inactivates the REQUEST line, and disables the handshake (if programmed). The $\overline{\text{CLEAR}}$ bit does not affect any control or data register. To remove the $\overline{\text{CLEAR}}$ state, write a 1 to the $\overline{\text{CLEAR}}$ bit.

In CPU/CPU mode, under program control, only one of the ports can empty the FIFO by writing to its Control Register 3, bit 6. The Port 1 CPU must program bit 7 in Control Register 3 to determine which port controls the $\overline{\text{CLEAR}}$ FIFO operation (0 = Port 1 control; 1 = Port 2 control).

**Direction of Data Transfer Operation.** The Data Direction bit controls the direction of data transfer in the FIFO buffer. The Data Direction bit is defined as 0 = output from CPU and 1 = input to CPU. This bit reads correctly when read by either port's CPU. For example, if Port 1's CPU reads a 0 (CPU output) in its Data Direction bit, then Port 2's CPU reads a 1 (input to CPU) in its Data Direction bit.

In CPU/CPU mode, under program control, only one of the ports can control the direction of data transfer. The Port 1 CPU must program bit 5 in Control Register 3 to determine which port controls the data direction (0 = Port 1 control; 1 = Port 2 control). Figure 20 shows FIO data transfer options.




NOTE: Usable only for CPU/CPU interface.

**Figure 19. Message Register Operation**

## CPU to CPU Operation (Continued)

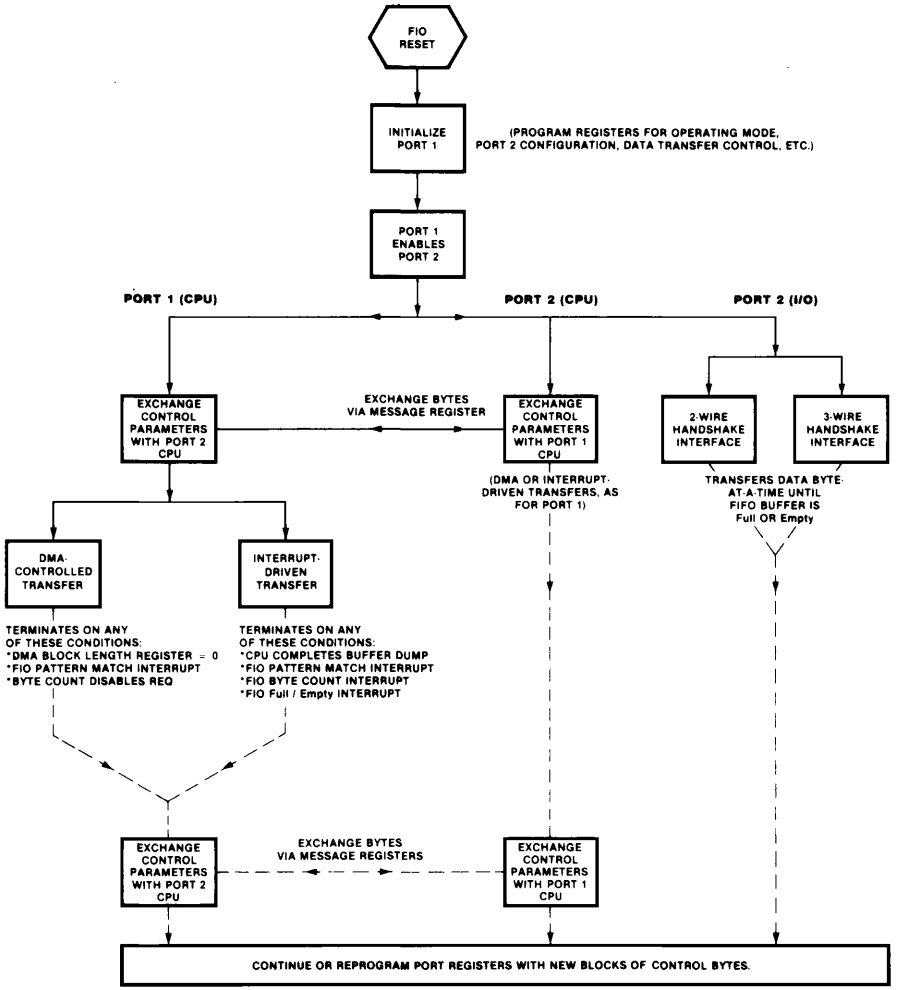



**Figure 20. FIO Data Transfert Options**

### CPU to I/O Operation

When Port 2 is programmed in the Interlocked 2-Wire Handshake mode or the 3-Wire Handshake mode, and Port A is programmed in Z-BUS or non-Z-BUS Microprocessor mode, the FIO interfaces a CPU and a peripheral device. In the Interlocked 2-Wire Handshake mode, RFD/$\overline{\text{DAV}}$ and $\overline{\text{ACKIN}}$ strobe data to and from Port 2. In the 3-Wire Handshake mode, RFD/$\overline{\text{DAV}}$, $\overline{\text{DAV}}$/DAC, and DAC/RFD signals control data flow.

**Interlocked 2-Wire Handshake.** In the Interlocked Handshake, the action of the FIO must be acknowledged by the other half of the handshake before the next action can take place. In output mode, Port 2 does not indicate that new data is available until the external device indicates it is ready for the data. Similarly, in input mode, Port 2 does not indicate that it is ready for new data until the data source indicates that the previous byte of the data is no longer available, thereby acknowledging Port 2's acceptance of the last byte. This allows the FIO to directly interface to a Z8's port, a CIO's port, a UPC's port, another FIO port, or another FIFO Z8060, with no external logic (Figures 21 and 22).

**3-Wire Handshake.** The 3-Wire Handshake is designed for applications 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 that an input port is ready for new data or that it has accepted the present data. In the 3-Wire Handshake, the rising edge of the RFD status line indicates that the port is ready for data, and the rising edge of the DAC status line indicates that the data has been accepted. With 3-Wire Handshake, the lines of many input ports can be bussed together with open-drain drivers and the output port knows when all of the ports are ready and have accepted the data. This handshake is the same handshake used in the IEEE-488 Instruments. Since the port's direction can be changed under software control, bidirectional IEEE-488-type transfers can be performed. Figures 23 and 24 show the timings associated with 3-Wire Handshake communications.

**CLEAR FIFO Operation.** In CPU-to-I/O operation, the CLEAR FIFO operation can be performed by the CPU side (Port 1) under software control as previously explained. The CLEAR FIFO operation can also be performed under hardware control by defining the $\overline{\text{CLEAR}}$ pin of Port 2 as an input (Control Register 3, bit 7 = 1).

For cascading purposes, the $\overline{\text{CLEAR}}$ pin can also be defined as an output (Control Register 3, bit 7 = 0), which reflects the current state of the $\overline{\text{CLEAR}}$ FIFO bit. It can then empty other FIOs or initialize other devices in the system.

**Data Direction Control.** In CPU-to-I/O mode, the direction of data transfer can be controlled by the CPU side (Port 1) under software control as previously explained. The data direction can also be determined by hardware control by defining the Data Direction pin of Port 2 as an input (Control Register 3, bit 5 = 1).

For cascading purposes, the Data Direction pin can also be defined as an output (Control Register 3, bit 5 = 0) pin which reflects the current state of the Data Direction bit. It can then be used to control the direction of data transfer for other FIOs or for external logic.

On the Port 2 side, when data direction is 0, Port 2 is in Output Handshake mode. When data direction is 1, Port 2 is in Input Handshake mode.

**CPU to I/O Operation** (Continued)

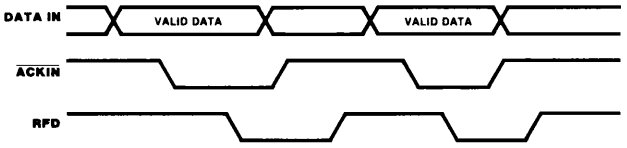



**Figure 21. Interlocked Handshake Timing (Imput) Port 2 Side Only**

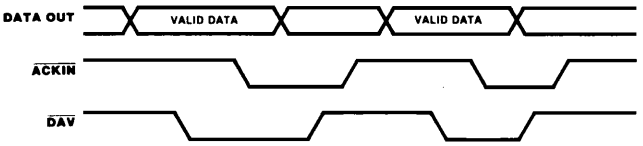



**Figure 22. Interlocked Handshake Timing (Output) Port 2 Side Only**


**Figure 23. Input (Acceptor) Timing IEEE-488 Port: Port 2 Side Only**


**Figure 24. Output (Source) Timing IEEE-488 HS Port: Port 2 Side Only**

## Programming

The programming of the FIO is greatly simplified by the efficient grouping of the various operation modes in the control registers. Since all of the control registers are read/write, the need for maintaining their image in system memory is eliminated. Also, the read/write feature of the registers aids in system debugging.

Each side of the FIO has 16 registers. All 16 registers are used by the Port 1 side; Control register 2 is not used on the Port 2 side. All registers are addressable $0_H$ through $F_H$.

In the Z-BUS Low Byte mode, the FIO allows two methods for register addressing under control of the Right Justify Address (RJA) bit in Control register 0. When RJA = 0, address bus bits 1-4 are used for register addressing and bits 5, 6, and 7 are ignored (Table 4). When RJA = 1, bits 0-3 are used for the register addresses, and bits 4-7 are ignored.

**Control Registers.** These four registers specify FIO operation. The Port 2 side control registers operate only if the Port 2 device is a CPU. The Port 2 CPU can control interface operations, including data direction, only when enabled by the setting of bit 0 in the Port 1 side of Control Register 2. A 1 in bit 1 of the same register enables the handshake logic.

**Interrupt Status Registers.** These four registers control and monitor the priority interrupt functions for the FIO.

**Interrupt Vector Register.** This register stores the interrupt service routine address. This vector is placed on $D_0$-$D_7$ when IUS is set by the Interrupt Acknowledge signal from the CPU. When bit 4 (Vector Includes Status) is set in Control Register 0, the reason for the interrupt

| Non Z-BUS | $D_7$-$D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ | |
|---|---|---|---|---|---|---|
| Z-BUS High | | $A_3$ | $A_2$ | $A_1$ | $A_0$ | |
| Z-BUS Low RJA=0 | $AD_7$-$AD_5$ | $AD_4$ | $AD_3$ | $AD_2$ | $AD_1$ | $AD_0$ |
| Z-BUS Low RJA=1 | $AD_7$-$AD_4$ | $AD_3$ | $AD_2$ | $AD_1$ | $AD_0$ | |
| **Description** | | | | | | |
| Control Register 0 | x | 0 | 0 | 0 | 0 | x |
| Control Register 1 | x | 0 | 0 | 0 | 1 | x |
| Interrupt Status Register 0 | x | 0 | 0 | 1 | 0 | x |
| Interrupt Status Register 1 | x | 0 | 0 | 1 | 1 | x |
| Interrupt Status Register 2 | x | 0 | 1 | 0 | 0 | x |
| Interrupt Status Register 3 | x | 0 | 1 | 0 | 1 | x |
| Interrupt Vector Register | x | 0 | 1 | 1 | 0 | x |
| Byte Count Register | x | 0 | 1 | 1 | 1 | x |
| Byte Count Comparison Register | x | 1 | 0 | 0 | 0 | x |
| Control Register 2* | x | 1 | 0 | 0 | 1 | x |
| Control Register 3 | x | 1 | 0 | 1 | 0 | x |
| Message Out Register | x | 1 | 0 | 1 | 1 | x |
| Message In Register | x | 1 | 1 | 0 | 0 | x |
| Pattern Match Register | x | 1 | 1 | 0 | 1 | x |
| Pattern Mask Register | x | 1 | 1 | 1 | 0 | x |
| Data Buffer Register | x | 1 | 1 | 1 | 1 | x |

x = Don't Care
*Register is only on Port 1 side

**Table 4. FIO Register Address Summary**

**Programming** Continued)

is encoded within the vector address in bits 1, 2, and 3. If bit 5 is set in Control register 0, no vector is output by the FIO during an Interrupt Acknowledge cycle. However, IUS is set as usual.

**Byte Count Compare Register.** This register contains a value compared with the byte count in the Byte Count register. If the Byte Count Compare interrupt is enabled, an interrupt will occur upon compare.

**Message Out Register.** Either CPU can place a message in its Message Out register. If the opposite side Message register interrupt is enabled, the receiving side CPU will receive an interrupt request, advising that a message is present in its Message In register. Bit 5 in Control Register 1 on the initiating side is set when a message is written. It is cleared when the message is read by the receiving CPU.

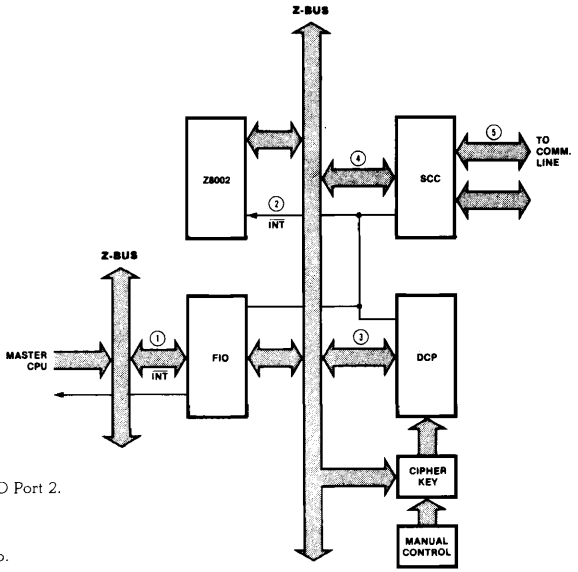
**Message In Register.** This register receives a message placed in the Message Out register by the opposite side CPU.

**Pattern Match Register.** This register contains a bit pattern matched against the byte in the Data Buffer register. When these patterns match, a Pattern Match interrupt will be generated, if previously enabled.

**Pattern Mask Register.** The Pattern Mask register may be programmed with a bit pattern mask that limits comparable bits in the Pattern Match register to non-masked bits (1 = mask).

**Data Buffer Register.** This register contains the data to be read from or written to the FIFO buffer.

**Byte Count Register.** This is a read-only register, containing the byte count for the FIFO buffer. The byte count is derived by subtracting the number of bytes read from the buffer from the number of bytes written into the buffer. The count is "frozen" for an accurate reading by setting bit 6 (Freeze Status register) in Control Register 1. This bit is cleared when the Byte Count register read is completed.

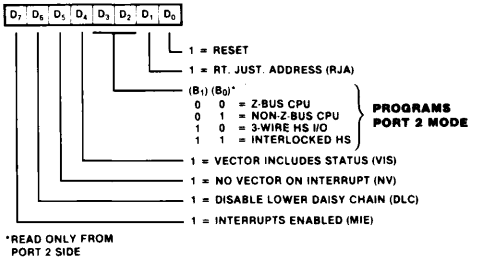



NOTES:
1. Data from master CPU — FIO Port 2.
2. FIO Port 1 — DCP.
3. DCP — RAM.
4. RAM — SCC.
5. SCC — data comm. line loop.

**Figure 25. Typical Application: Node Controller**

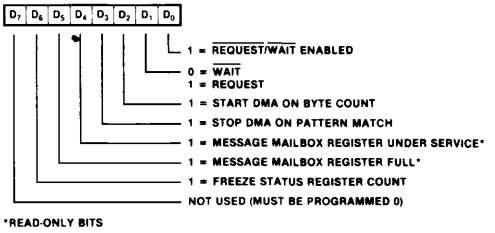
## Registers

### Control Register 0
Address: 0000
(Read/Write)



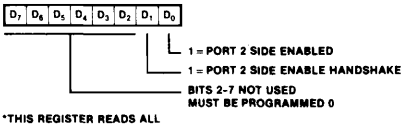

*READ ONLY FROM PORT 2 SIDE

### Control Register 1
Address: 0001
(Read/Write)



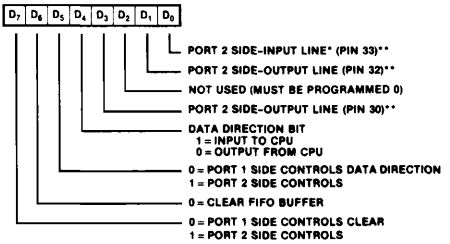

*READ-ONLY BITS

### Control Register 2*
Address: 1001
(Read/Write)




*THIS REGISTER READS ALL 0'S FROM PORT 2 SIDE

### Control Register 3
Address: 1010
(Read/Write)




*READ-ONLY BITS
**ONLY WHEN PORT 2 IS AN I/O PORT

**Figure 26. Control Registers**

### Interrupt Status Register 0
Address: 0010
(Read/Write)

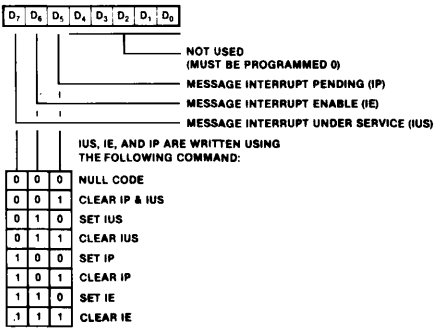
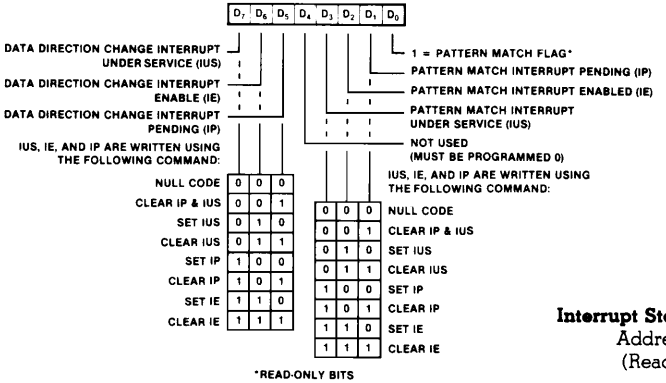


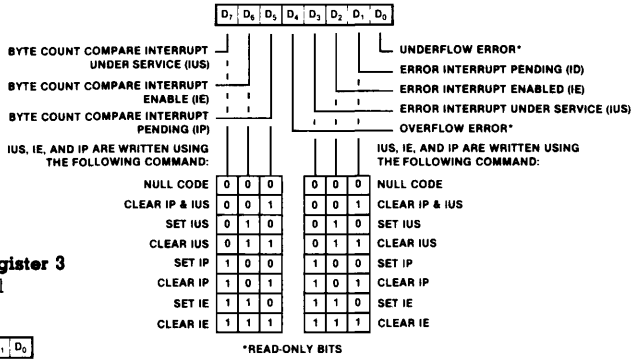

**Figure 27. Interrupt Status Registers**

**Registers** (Continued)

**Interrupt Status Register 1**
Address: 0011
(Read/Write)




**Interrupt Status Register 2**
Address: 0100
(Read/Write)




**Interrupt Status Register 3**
Address: 0101
(Read/Write)

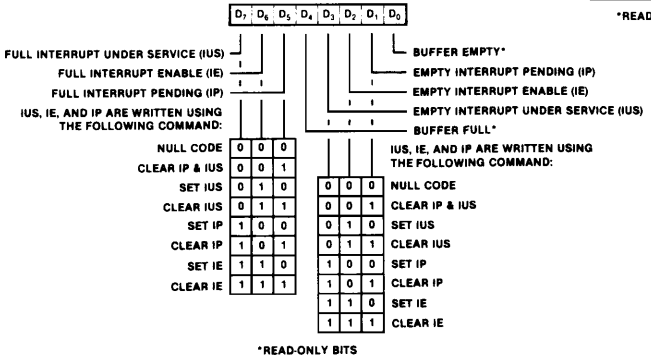



**Figure 27. Interrupt Status Registers** (Continued)

**Registers** (Continued)

**Byte Count Register**
Address: 0111

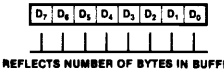



**Figure 28. Byte Count Register**

**Interrupt Vector Register**
Address: 0110
(Read/Write)

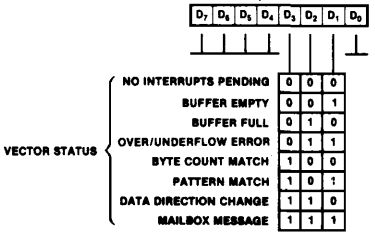



**Figure 29. Interrupt Vector Register**

**Pattern Match Register**
Address: 1011
(Read/Write)

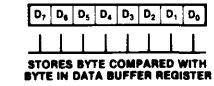



**Figure 30. Pattern Match Register**

**Pattern Mask Register**
Address: 1110
(Read/Write)

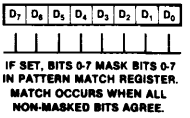



**Figure 31. Pattern Mask Register**

**Data Buffer Register**
Address: 1111
(Read/Write)

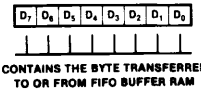



**Figure 32. Data Buffer Registe**

**Byte Count Comparison Register**
Address: 1000
(Read/Write)

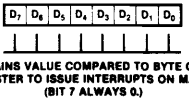



**Figure 33. Byte Count Comparison Register**

**Message Out Register**
Address: 1011
(Read/Write)

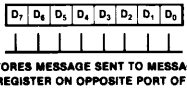



**Figure 34. Message Out Register**

**Message In Register**
Address: 1100
(Read Only)

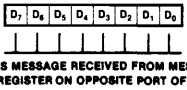



**Figure 35. Message In Register**

### Absolute Maximum Ratings

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

Operating Ambient Temperature . . . . . 0°C to +70°C

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

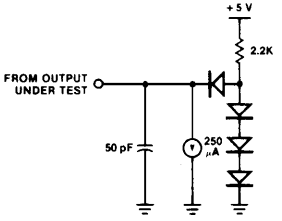



**Figure 36. Standard Test Load**

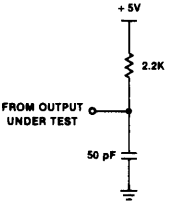



**Figure 37. Open-Drain Test Load**

### 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 μA |
| $V_{OL}$ | 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 | +10.0 | μA | 0.4 ≤ $V_{IN}$ ≤ +2.4V |
| $I_{OL}$ | Output Leakage | −10.0 | +10.0 | μA | 0.4 ≤ $V_{OUT}$ ≤ +2.4V |
| $I_{LM}$ | Mode Pins Input Leakage (Pins 19 and 21) | −100 | +10.0 | μA | 0 < $V_{IN}$ < $V_{CC}$ |
| $I_{CC}$ | $V_{CC}$ Supply Current | | 200 | mA | |

$V_{CC}$ = 5 V ± 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 | |

## Inputs

| | | | | |
|---|---|---|---|---|
| tr | Any Input Rise Time | 100 | ns | |
| tf | Any Input Fall Time | 100 | ns | |

f = 1 MHz, over specified temperature range.

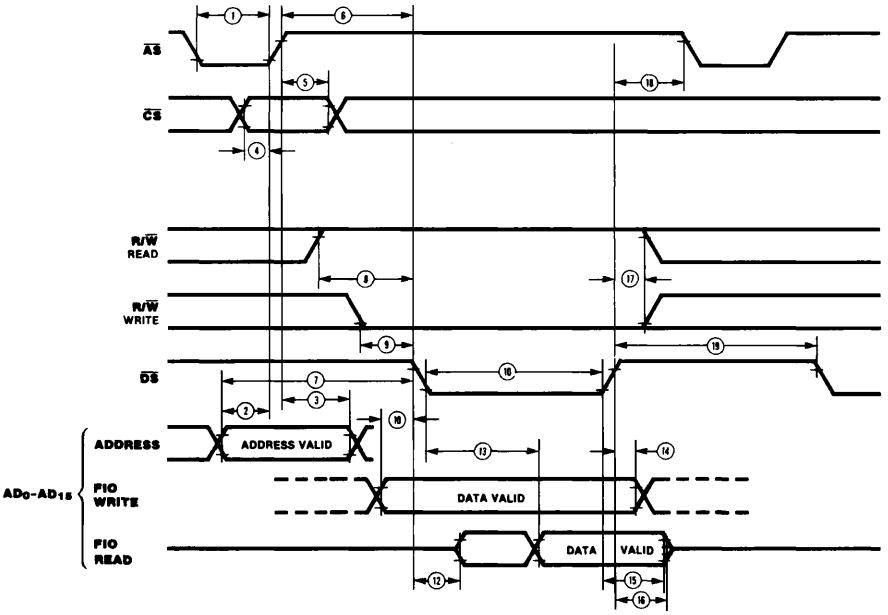
## Z-BUS CPU Inteface Timing

| No. | Symbol | Parameter | 4 MHz Min | 4 MHz Max | 6 MHz Min | 6 MHz Max | Notes*† |
|---|---|---|---|---|---|---|---|
| 1 | TwAS | $\overline{AS}$ Low Width | 70 | | 50 | | 1 |
| 2 | TsA(AS) | Address to $\overline{AS}$ ↑ Setup Time | 30 | | 10 | | 1 |
| 3 | ThA(AS) | Address to $\overline{AS}$ ↑ Hold Time | 50 | | 30 | | 1 |
| 4 | TsCSO(AS) | $\overline{CS}$ to $\overline{AS}$ ↑ Setup Time | 0 | | 0 | | 1 |
| 5 | ThCSO(AS) | $\overline{CS}$ to $\overline{AS}$ ↑ Hold Time | 60 | | 40 | | 1 |
| 6 | TdAS(DS) | $\overline{AS}$ ↑ to $\overline{DS}$ ↑ Delay | 60 | | 40 | | 1 |
| 7 | TsA(DS) | Address to $\overline{DS}$ ↓ (with $\overline{AS}$ ↑ to $\overline{DS}$ ↓ = 60 ns) | 120 | | 100 | | |
| 8 | TsRWR(DS) | R/$\overline{W}$ (Read) to $\overline{DS}$ ↓ Setup Time | 100 | | 80 | | |
| 9 | TsRWW(DS) | R/$\overline{W}$ (Write) to $\overline{DS}$ ↓ Setup Time | 0 | | 0 | | |
| 10 | TwDS | $\overline{DS}$ Low Width | 390 | | 250 | | |
| 11 | TsDW(DSf) | Write Data to $\overline{DS}$ ↓ Setup Time | 30 | | 20 | | |
| 12 | TdDS(DRV) | $\overline{DS}$ (Read) ↓ to Address Data Bus Driven | 0 | | 0 | | |
| 13 | TdDSf(DR) | $\overline{DS}$ ↓ to Read Data Valid Delay | | 250 | | 180 | |
| 14 | ThDW(DS) | Write Data to $\overline{DS}$ ↑ Hold Time | 30 | | 20 | | |
| 15 | TdDSr(DR) | $\overline{DS}$ ↑ to Read Data Not Valid Delay | 0 | | 0 | | |
| 16 | TdDS(DRz) | $\overline{DS}$ ↑ to Read Data Float Delay | | 70 | | 45 | 2 |
| 17 | ThRW(DS) | R/$\overline{W}$ to $\overline{DS}$ ↑ Hold Time | 55 | | 40 | | |
| 18 | TdDS(AS) | $\overline{DS}$ ↑ to $\overline{AS}$ ↓ Delay | 50 | | 25 | | |
| 19 | Trc | Valid Access Recovery Time | 1000 | | 650 | | 3 |

NOTES:
1. Parameter does not apply to Interrupt Acknowledge transactions.
2. Float delay is measured to the time when the output has changed 0.5 V from steady state with minimum as load and load and maximum dc load.
3. This is the delay from $\overline{DS}$ of one CIO access to $\overline{DS}$ of another FIO access (either read or write).

* All timing references assume 2.0 V for a logic "1" and 0.8 V for a logic "0". All timings are preliminary and subject to change.
† Units in nanoseconds (ns).

**Z-BUS CPU Inteface Timing** (Continued)



## Z-BUS CPU Interrupt Acknowledge Timing

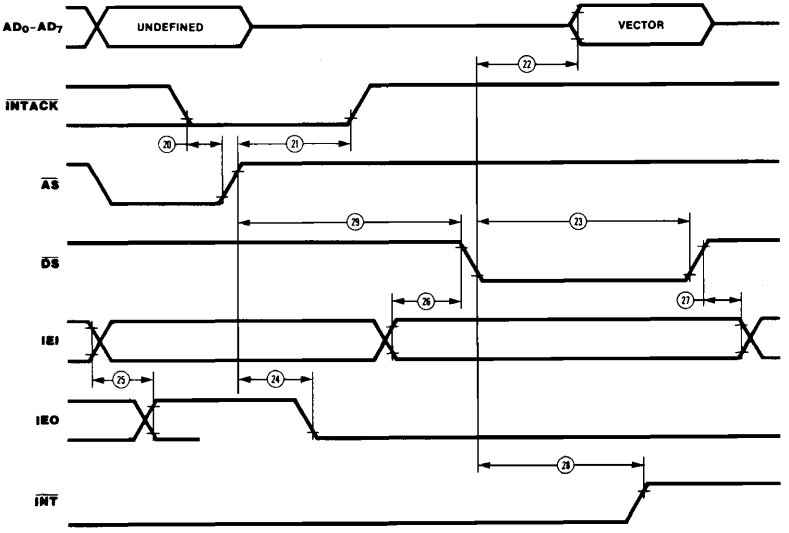
| No. | Symbol | Parameter | 4 MHz Min | 4 MHz Max | 6 MHz Min | 6 MHz Max | Notes*† |
|---|---|---|---|---|---|---|---|
| 20 | TsIA(AS) | $\overline{INTACK}$ to $\overline{AS}$ ↑ Setup Time | 0 | | | 0 | |
| 21 | ThIA(AS) | $\overline{INTACK}$ to $\overline{AS}$ ↑ Hold Time | 250 | | 250 | | |
| 22 | TsDSA(DR) | $\overline{DS}$ (Acknowledge) ↓ to Read Data Valid Delay | | 250 | | 180 | |
| 23 | TwDSA | $\overline{DS}$ (Acknowledge) Low Width | 390 | | 250 | | |
| 24 | TdAS(IEO) | $\overline{AS}$ ↑ to IEO ↓ Delay ($\overline{INTACK}$ Cycle) | | 350 | | 250 | 4 |
| 25 | TdIEI(IEO) | IEI to IEO Delay | | 150 | | 100 | 4 |
| 26 | TsIEI(DSA) | IEI to $\overline{DS}$ (Acknowledge) ↓ Setup Time | 100 | | 70 | | |
| 27 | ThIEI(DSA) | IEI to $\overline{DS}$ (Acknowledge) ↑ Hold Time | 50 | | 30 | | 4 |
| 28 | TdDS(INT) | $\overline{DS}$ ($\overline{INTACK}$ Cycle) to $\overline{INT}$ Delay | | 900 | | 800 | |
| 29 | TdDCST | Interrupt Daisy Chain Settle Time | | | | | 4 |

NOTES:

4. The parameters for the devices in any particular daisy chain must meet the following constraint: The delay from $\overline{AS}$ to $\overline{DS}$ must be greater than the sum of TdAS(IEO) for the highest priority peripheral, TsIEI(DSA) for the lowest priority peripheral and TdIEI(IEO) for each peripheral, separating them in the chain.

* Timings are preliminary and subject to change.

† Units in nanoseconds (ns).

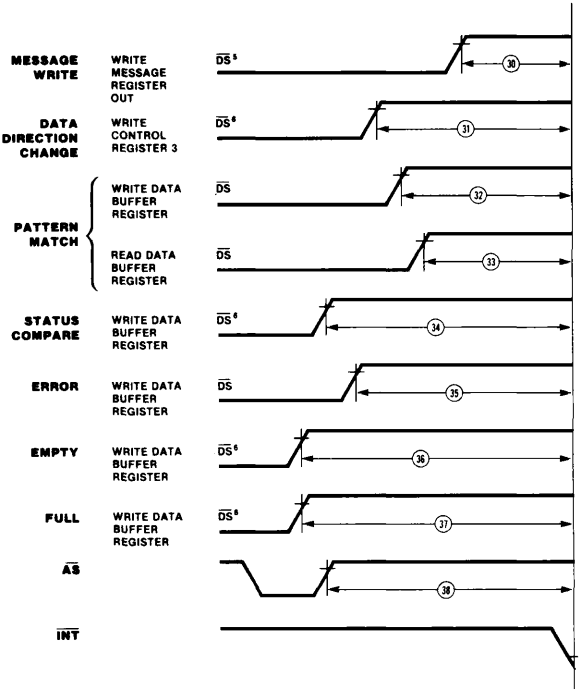


## Z-BUS Interrupt Timing

| No. | Symbol | Parameter | 4 MHz Min | 4 MHz Max | 6 MHz Min | 6 MHz Max | Notes*† |
|---|---|---|---|---|---|---|---|
| 30 | TdMW(INT) | Message Write to $\overline{INT}$ Delay | | 1 | | 1 | 5 |
| 31 | TdDC(INT) | Data Direction Change to $\overline{INT}$ Delay | | 1 | | 1 | 6 |
| 32 | TdPMW(INT) | Pattern Match to $\overline{INT}$ Delay (Write Case) | | 1 | | 1 | |
| 33 | TdPMR(INT) | Pattern Match (Read Case) to $\overline{INT}$ Delay | | 1 | | 1 | |
| 34 | TdSC(INT) | Status Compare to $\overline{INT}$ Delay | | 1 | | 1 | 6 |
| 35 | TdER(INT) | Error to $\overline{INT}$ Delay | | 1 | | 1 | |
| 36 | TdEM(INT) | Empty to $\overline{INT}$ Delay | | 1 | | 1 | 6 |
| 37 | TdFL(INT) | Full to $\overline{INT}$ Delay | | 1 | | 1 | 6 |
| 38 | TdAS(INT) | $\overline{AS}$ to $\overline{INT}$ Delay | | | | | |

NOTES:
5. Write is from the other side of FIO.
6. Write can be from either side, depending on programming of FIO.

* Timings are preliminary and subject to change.
† Units equal to $\overline{AS}$ Cycles + ns.

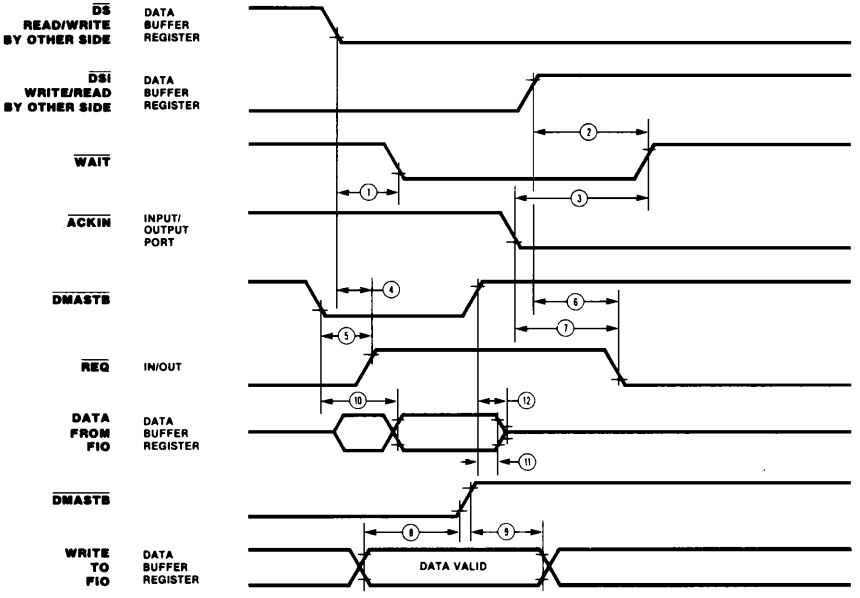


## Z-BUS Request/Wait Timing

| No. | Symbol | Parameter | 4 MHz Min | 4 MHz Max | 6 MHz Min | 6 MHz Max | Notes*† |
|---|---|---|---|---|---|---|---|
| 1 | TdDS(WAIT) | $\overline{AS}$ ↑ to $\overline{WAIT}$ ↓ Delay | | 190 | | 160 | |
| 2 | TdDS1(WAIT) | $\overline{DS1}$ ↑ to $\overline{WAIT}$ ↑ Delay | | 1000 | | 1000 | |
| 3 | TdACK(WAIT) | $\overline{ACKIN}$ ↓ to $\overline{WAIT}$ ↑ Delay | | 1000 | | 1000 | 1 |
| 4 | TdDS(REQ) | $\overline{DS}$ ↓ to $\overline{REQ}$ ↑ Delay | | 350 | | 300 | |
| 5 | TdDMA(REQ) | $\overline{DMASTB}$ ↓ to $\overline{REQ}$ ↑ Delay | | 350 | | 300 | |
| 6 | TdDS1(REQ) | $\overline{DS1}$ ↑ to $\overline{REQ}$ ↓ Delay | | 1000 | | 1000 | |
| 7 | TdACK(REQ) | $\overline{ACKIN}$ ↓ to $\overline{REQ}$ ↓ Delay | | 1000 | | 1000 | |
| 8 | TdSU(DMA) | Data Setup Time to $\overline{DMASTB}$ | 200 | | 150 | | |
| 9 | TdH(DMA) | Data Hold Time to $\overline{DMASTB}$ | 30 | | 20 | | |
| 10 | TdDMA(DR) | $\overline{DMASTB}$ ↓ to Valid Data | | 150 | | 100 | |
| 11 | TdDMA(DRH) | $\overline{DMASTB}$ ↑ to Data Not Valid | 0 | | 0 | | |
| 12 | TdDMA(DR2) | $\overline{DMASTB}$ ↑ to Data Bus Float | | 70 | | 45 | |

NOTES:
1. The delay is from $\overline{DAV}$ for 3-Wire Input Handshake. The delay is from DAC for 3-Wire Handshake.

* Timings are preliminary and subject to change.
† Units in nanoseconds (ns).

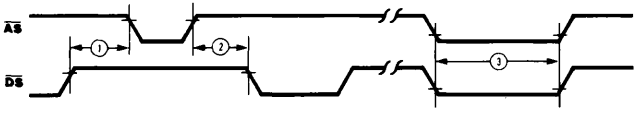


## Z-BUS Reset Timing

| No. | Symbol | Parameter | 4 MHz Min | 4 MHz Max | 6 MHz Min | 6 MHz Max | Notes*† |
|---|---|---|---|---|---|---|---|
| 1 | TdDSQ(AS) | Delay from $\overline{DS}$ ↑ to $\overline{AS}$ ↓ for No Reset | 40 | | 20 | | |
| 2 | TdASQ(DS) | Delay for $\overline{AS}$ ↑ to $\overline{DS}$ ↓ for No Reset | 50 | | 30 | | |
| 3 | Tw(AS + DS) | Minimum Width of $\overline{AS}$ and $\overline{DS}$ Both Low for Reset. | 500 | | 350 | | 1 |

NOTES:
1. Internal circuitry allows for the reset provided by the Z8 (DS held Low while $\overline{AS}$ pulses) to be sufficient.

* Timings are preliminary and subject to change.
† Units in nanoseconds (ns).



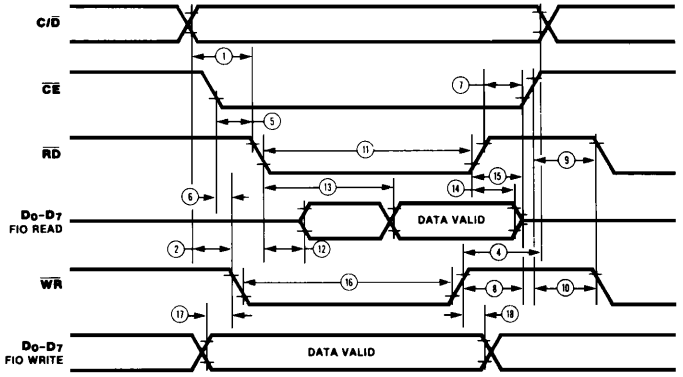

## Non-Z-BUS CPU Interface Timing

| No. | Symbol | Parameter | 4 MHz Min | 4 MHz Max | 6 MHz Min | 6 MHz Max | Notes*† |
|---|---|---|---|---|---|---|---|
| 1 | TsA(RD) | Address Setup to $\overline{RD}$ ↓ | 80 | | 80 | | 1 |
| 2 | TsA(WR) | Address Setup to $\overline{WR}$ ↓ | 80 | | 80 | | |
| 3 | ThA(RD) | Address Hold Time to $\overline{RD}$ ↑ | 0 | | 0 | | 1 |
| 4 | ThA(WR) | Address Hold Time to $\overline{WR}$ ↑ | 0 | | 0 | | |
| 5 | TsCEI(RD) | $\overline{CE}$ Low Setup Time to $\overline{RD}$ | 0 | | 0 | | 1 |
| 6 | TsCEI(WR) | $\overline{CE}$ Low Setup Time to $\overline{WR}$ | 0 | | 0 | | |
| 7 | ThCEI(RD) | $\overline{CE}$ Low Hold Time to $\overline{RD}$ | 0 | | 0 | | 1 |
| 8 | ThCEI(WR) | $\overline{CE}$ Low Hold Time to $\overline{WR}$ | 0 | | 0 | | |
| 9 | TsCEh(RD) | $\overline{CE}$ High Setup Time to $\overline{RD}$ | 100 | | 70 | | 1 |
| 10 | TsCEh(WR) | $\overline{CE}$ High Setup Time to $\overline{WR}$ | 100 | | 70 | | |
| 11 | TwRD1 | $\overline{RD}$ Low Width | 390 | | 250 | | |
| 12 | TdRD(DRA) | $\overline{RD}$ ↓ to Read Data Active Delay | 0 | | 0 | | |
| 13 | TdRDf(DR) | $\overline{RD}$ ↓ to Valid Data Delay | | 250 | | 180 | |
| 14 | TdRDr(DR) | $\overline{RD}$ ↑ to Read Data Not Valid Delay | 0 | | 0 | | |
| 15 | TdRD(DRz) | $\overline{RD}$ ↑ to Data Bus Float | | 70 | | 45 | 2 |
| 16 | TwWR1 | $\overline{WR}$ Low Width | 390 | | 250 | | |
| 17 | TsDW(WR) | Data Setup Time to $\overline{WR}$ | 0 | | 0 | | |
| 18 | ThDW(WR) | Data Hold Time to $\overline{WR}$ | 30 | | 20 | | |
| 19 | Trc | Valid Access Recovery Time | 1000 | | 650 | | 3 |

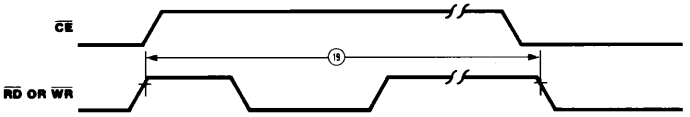
NOTES:
1. Parameter does not apply to Interrupt Acknowledge transactions.
2. Float delay is measured to the time the output has changed 0.5 V from steady state with minimum ac load and maximum dc load.
3. This is the delay from $\overline{RD}$ ↑ to $\overline{WR}$ ↑ of one FIO access to $\overline{RD}$ ↓ or $\overline{WR}$ ↓ of another FIO access.

* Timings are preliminary and subject to change.
† Units in nanoseconds (ns).

## Non-Z-BUS CPU Interface Timing(Continued)




**Non-Z-BUS CPU Interface Timing**




**Non-Z-BUS Interface Timing**

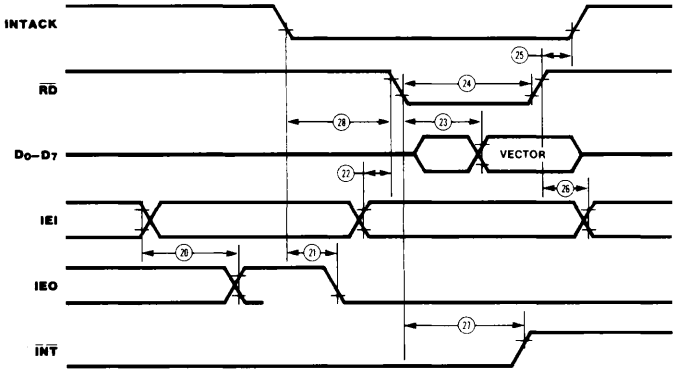
## Non-Z-BUS Interrupt Acknowledge Timing

| No. | Symbol | Parameter | 4 MHz Min | 4 MHz Max | 6 MHz Min | 6 MHz Max | Notes†* |
|---|---|---|---|---|---|---|---|
| 20 | TdIEI(IEO) | IEI to IEO Delay | | 150 | | 100 | 4 |
| 21 | TdI(IEO) | $\overline{INTACK}$ ↓ to IEO ↓ Delay | | 350 | | 250 | 4 |
| 22 | TsIEI(RDA) | IEI Setup Time to $\overline{RD}$ (Acknowledge) | 100 | | 70 | | 4 |
| 23 | TdRD(DR) | $\overline{RD}$ ↓ to Vector Valid Delay | | 250 | | 180 | |
| 24 | TwRDl(IA) | Read Low Width (Interrupt Acknowledge) | 390 | | 250 | | |
| 25 | ThIA(RD) | $\overline{INTACK}$ ↑ to $\overline{RD}$ ↑ Hold Time | 30 | | 20 | | |
| 26 | ThIEI(RD) | IEI Hold Time to $\overline{RD}$ ↑ | 20 | | 10 | | |
| 27 | TdRD(INT) | $\overline{RD}$ ↑ to $\overline{INT}$ ↑ Delay | | 900 | | 800 | |
| 28 | TdDCST | Interrupt Daisy Chain Settle Time | 350 | | 250 | | 4 |

NOTES:
4. The parameter for the devices in any particular daisy chain must meet the following constraint: The delay from $\overline{INTACK}$ ↓ to $\overline{RD}$ ↓ must be greater than the sum of TdINA(IEO) for the highest priority peripheral, TsIEI(RD) for the lowest priority peripheral, and TdIEI(IEO) for each peripheral separating them in the chain.

† Units in nanoseconds (ns).

* Timings are preliminary and subject to change.

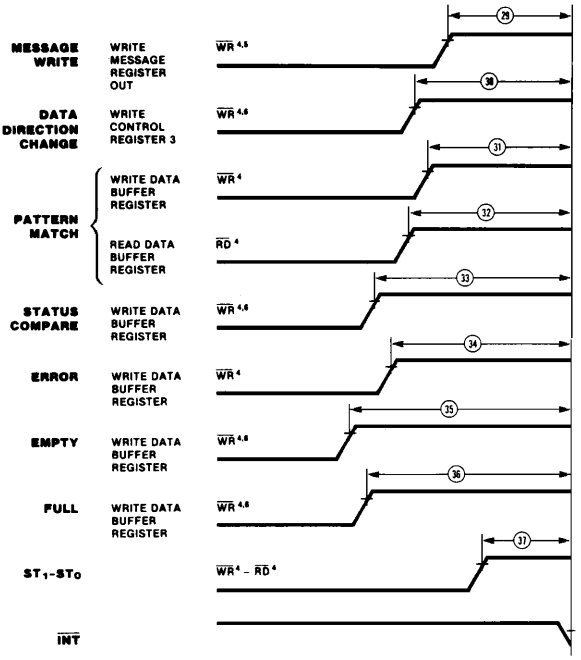


## Non-Z-BUS Interrupt Timing

| No. | Symbol | Parameter | 4 MHz Min | 4 MHz Max | 6 MHz Min | 6 MHz Max | Notes*† |
|---|---|---|---|---|---|---|---|
| 29 | TdMW(INT) | Message Write to $\overline{INT}$ Delay | | | | | 5,6 |
| 30 | TdDC(INT) | Data Direction Change to $\overline{INT}$ Delay | | | | | 5,7 |
| 31 | TdPMW(INT) | Pattern Match (Write Case) to $\overline{INT}$ Delay | | | | | 5 |
| 32 | TdPMR(INT) | Pattern Match (Read Case) to $\overline{INT}$ Delay | | | | | 5 |
| 33 | TdSC(INT) | Status Compare to $\overline{INT}$ Delay | | | | | 5,7 |
| 34 | TdER(INT) | Error to $\overline{INT}$ Delay | | | | | 5,7 |
| 35 | TdEM(INT) | Empty to $\overline{INT}$ Delay | | | | | 5,7 |
| 36 | TdFL(INT) | Full to $\overline{INT}$ Delay | | | | | 5,7 |
| 37 | TdSO(INT) | State 0 to $\overline{INT}$ Delay | | | | | |

NOTES:
5. Delay number is valid for State 0 only.
6. Write is from other side of FIO.
7. Write can be from either side, depending on programming of FIO.

* Timings are preliminary and subject to change.
† Units in nanoseconds (ns).

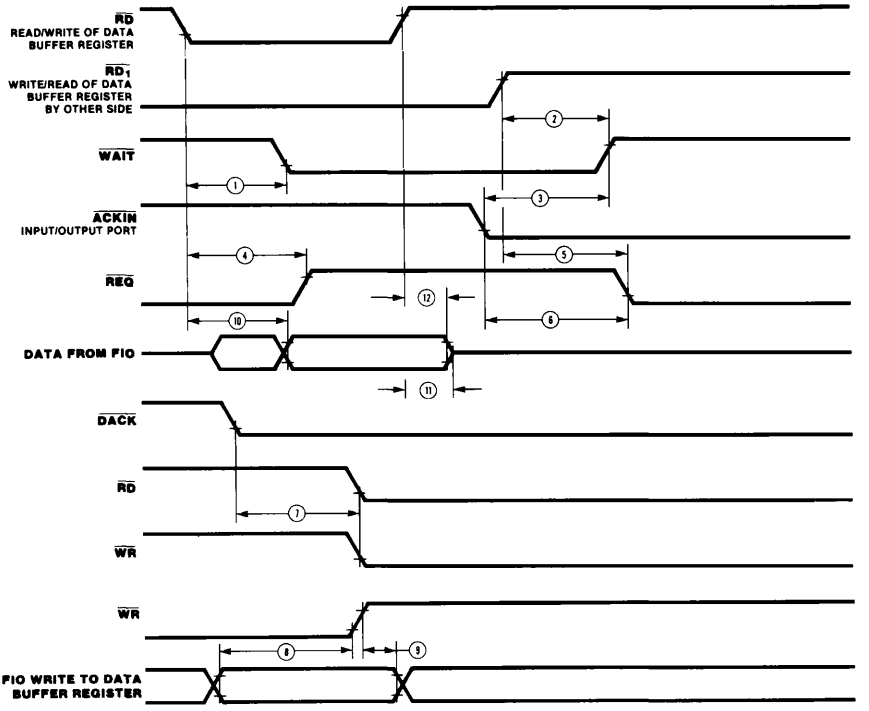


## Non-Z-BUS Request/Wait Timing

| No. | Symbol | Parameter | 4 MHz Min | 4 MHz Max | 6 MHz Min | 6 MHz Max | Notes*† |
|---|---|---|---|---|---|---|---|
| 1 | TdRD(WT) | $\overline{CE}$ ↓ to $\overline{WAIT}$ Active | | 200 | | 170 | |
| 2 | TdRD1(WT) | $\overline{RD}$1 ↑ or $\overline{WR}$1 ↑ to $\overline{WAIT}$ Inactive | | 1000 | | 1000 | |
| 3 | TdACK(WT) | $\overline{ACKIN}$ ↓ to $\overline{WAIT}$ Inactive | | 1000 | | 1000 | 1 |
| 4 | TdRD(REQ) | $\overline{RD}$ ↓ or $\overline{WR}$ ↓ to $\overline{REQ}$ Inactive | | 350 | | 300 | |
| 5 | TdRD1(REQ) | $\overline{RD}$1 ↑ or $\overline{WR}$1 ↑ to $\overline{REQ}$ Active | | 1000 | | 1000 | |
| 6 | TdACK(REQ) | $\overline{ACKIN}$ ↓ to $\overline{REQ}$ Active | | 1000 | | 1000 | |
| 7 | TdDAC(RD) | $\overline{DACK}$ ↓ to $\overline{RD}$ ↓ or $\overline{WR}$ ↓ | 100 | | 80 | | |
| 8 | TSU(WR) | Data Setup Time to $\overline{WR}$ | 200 | | | | |
| 9 | Th(WR) | Data Hold Time to $\overline{WR}$ | 30 | | | 20 | |
| 10 | TdDMA | $\overline{RD}$ ↓ to Valid Data | | 150 | | 100 | 2 |
| 11 | TdDMA(DRH) | $\overline{RD}$ ↑ to Data Not Valid | 0 | | 0 | | 2 |
| 12 | TdDMA(DRZ) | $\overline{RD}$ ↑ to Data Bus Float | | 70 | | 45 | 2 |

NOTES:
1. The delay is from $\overline{DAV}$ ↓ for 3-Wire Input Handshake. The delay is from $\overline{DAC}$ ↑ for 3-Wire Input Handshake.
2. Only when $\overline{DACK}$ is active.

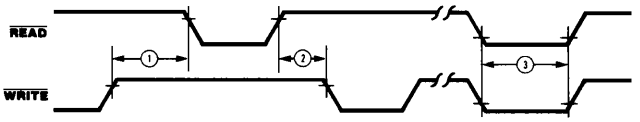
* Timings are preliminary and subject to change.
† Units in nanoseconds (ns).



## Non-Z-BUS Reset Timing

| No. | Symbol | Parameter | 4 MHz Min | 4 MHz Max | 6 MHz Min | 6 MHz Max | Notes*† |
|---|---|---|---|---|---|---|---|
| 1 | TdWR(RD) | Delay from $\overline{WR}$ ↑ to $\overline{RD}$ ↓ | 100 | | 70 | | |
| 2 | TdRD(WR) | Delay from $\overline{RD}$ ↑ to $\overline{WR}$ ↓ | 100 | | 70 | | |
| 3 | TwRD + WR | Width of $\overline{RD}$ and $\overline{WR}$, both Low for Reset | 500 | | 350 | | |

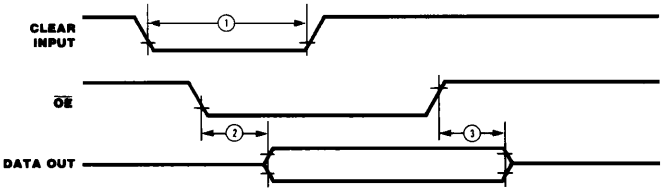
NOTES:
* Timings are preliminary and subject to change.
† Units in nanoseconds (ns).




## Port 2 Side Operation

| No. | Symbol | Parameter | 4 MHz Min | 4 MHz Max | 6 MHz Min | 6 MHz Max | Notes*† |
|---|---|---|---|---|---|---|---|
| 1 | TwCLR | Width of Clear to Reset FIFO | 700 | | 700 | | |
| 2 | TdOE(DO) | $\overline{OE}$ ↓ to Data Bus Driven | 0 | | 0 | | |
| 3 | TdOE(DRZ) | $\overline{OE}$ ↑ to Data Bus Float | | | | | |

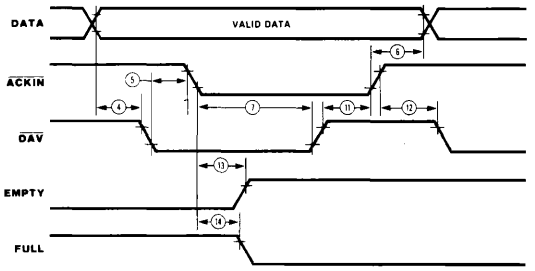
NOTES:
* Timings are preliminary and subject to change.
† Units in nanoseconds (ns).



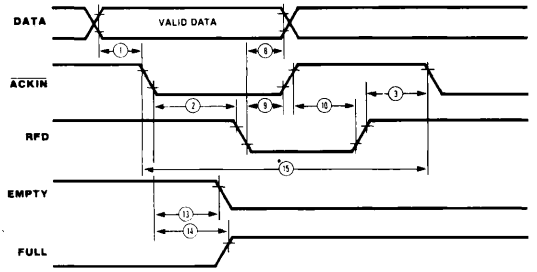
## FIO 2-Wire Handshake Timing

| No. | Symbol | Parameter | 4 MHz Min | 4 MHz Max | 6 MHz Min | 6 MHz Max | Notes*† |
|---|---|---|---|---|---|---|---|
| 1 | TsDI(ACK) | Data Input to $\overline{ACKIN}$ ↓ to Setup Time | 50 | | 50 | | |
| 2 | TdACKf(RFD) | $\overline{ACKIN}$ ↓ to RFD ↓ Delay | 0 | 500 | 0 | 500 | |
| 3 | TdRFDr(ACK) | RFD ↑ to $\overline{ACKIN}$ ↓ Delay | 0 | | 0 | | |
| 4 | TsDO(DAV) | Data Out to $\overline{DAV}$ ↓ Setup Time | 50 | | 25 | | |
| 5 | TdDAVf(ACK) | $\overline{DAV}$ ↓ to $\overline{ACKIN}$ ↓ Delay | 0 | | 0 | | |
| 6 | ThDO(ACK) | Data Out to $\overline{ACKIN}$ Hold Time | 50 | | 50 | | |
| 7 | TdACK(DAV) | $\overline{ACKIN}$ ↓ to $\overline{DAV}$ ↑ Delay | 0 | 500 | 0 | 500 | |
| 8 | ThDI(RFD) | Data Input to RFD ↓ Hold Time | 0 | | 0 | | |
| 9 | TdRFDf(ACK) | RFD ↓ to $\overline{ACKIN}$ ↑ Delay | 0 | | 0 | | |
| 10 | TdACKr(RFD) | $\overline{ACKIN}$ ↑ ($\overline{DAV}$ ↑) to RFD ↑ Delay—Interlocked and 3-Wire Handshake | 0 | 400 | 0 | 400 | |
| 11 | TdDAVr(ACK) | $\overline{DAV}$ ↑ to $\overline{ACKIN}$ ↑ (RFD ↑) | 0 | | 0 | | |
| 12 | TdACKr(DAV) | $\overline{ACKIN}$ ↑ to $\overline{DAV}$ ↓ | 0 | 800 | 0 | 800 | |
| 13 | TdACKf(Empty) | $\overline{ACKIN}$ ↓ to Empty | 0 | | 0 | | |
| 14 | TdACKf(Full) | $\overline{ACKIN}$ ↓ to Full | 0 | | 0 | | |
| 15 | TcACK | $\overline{ACKIN}$ Cycle Time | 1 | | 1 | | 1 |

NOTES:
* Timings are preliminary and subject to change.
† Units in nanoseconds (ns), except as noted.
1. Units in microseconds.




**2-Wire Handshake (Port 2 Side Only) Output**

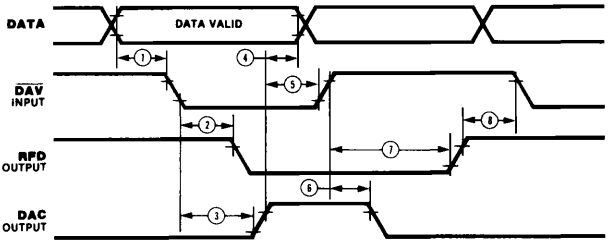



**2-Wire Handshake (Port 2 Side Only) Input**

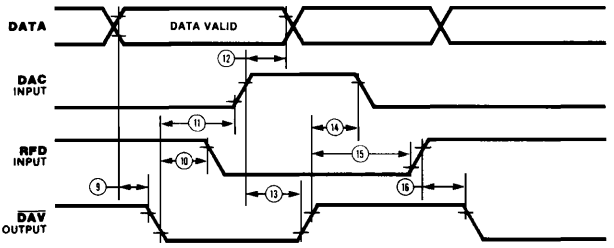
## 3-Wire Handshake Timing

| No. | Symbol | Parameter | 4 MHz Min | 4 MHz Max | 6 MHz Min | 6 MHz Max | Notes*† |
|---|---|---|---|---|---|---|---|
| 1 | TsDI(DAV) | Data Input to $\overline{DAV}$ ↓ Setup Time | 50 | | 50 | | |
| 2 | TdDAVIf(RFD) | $\overline{DAV}$ ↓ to RFD ↓ Delay | 0 | 500 | 0 | 500 | |
| 3 | TdDAVf(DAC) | $\overline{DAV}$ ↓ to DAC ↑ Delay | 0 | 500 | 0 | 500 | |
| 4 | ThDI(DAC) | Data In to DAC ↑ Hold Time | 0 | | 0 | | |
| 5 | TdDACIr(DAV) | DAC ↑ to $\overline{DAV}$ ↑ Delay | 0 | | 0 | | |
| 6 | TdDAVIr(DAC) | $\overline{DAV}$ ↑ to DAC ↓ Delay | 0 | 500 | 0 | 500 | |
| 7 | TdDAVIr(RFD) | $\overline{DAV}$ ↑ to RFD ↑ Delay | 0 | 500 | 0 | 500 | |
| 8 | TdRFDI(DAV) | RFD ↑ to $\overline{DAV}$ ↓ Delay | 0 | | 0 | | |
| 9 | TsDO(DAC) | Data Out to $\overline{DAV}$ ↓ | | | | | |
| 10 | TdDAVOf(RFD) | $\overline{DAV}$ ↓ to RFD ↓ Delay | 0 | | 0 | | |
| 11 | TdDAVOf(DAC) | $\overline{DAV}$ ↓ to DAC ↑ Delay | 0 | | 0 | | |
| 12 | ThDO(DAC) | Data Out to DAC ↑ Hold Time | | | | | |
| 13 | TdDACOr(DAV) | DAC ↑ to $\overline{DAV}$ ↑ Delay | | 400 | | 400 | |
| 14 | TdDAVOr(DAC) | $\overline{DAV}$ ↑ to DAC ↓ Delay | 0 | | 0 | | |
| 15 | TdDAVOr(RFD) | $\overline{DAV}$ ↑ to RFD ↑ Delay | 0 | | 0 | | |
| 16 | TdRFDO(DAV) | RFD ↑ to $\overline{DAV}$ ↓ Delay | 0 | 800 | 0 | 800 | |

NOTES:
* Timings are preliminary and subject to change.
† Units in nanoseconds (ns).




**3-Wire Handshake Input**



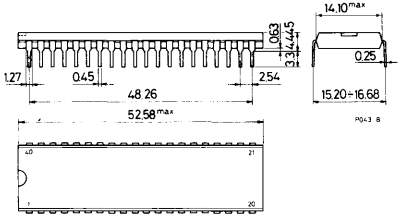

**3-Wire Handshake Output**

## Ordering Information

| Type | Package | Temp | Clock | Description |
|---|---|---|---|---|
| Z8038 B1 | Plastic | 0/+70°C | 4MHz | Z8038 FIFO Buffer, Z-BUS Compatible |
| B6 | Plastic | -40/+85°C | | |
| D1 | Ceramic | 0/+70°C | | |
| D6 | Ceramic | -40/+85°C | | |
| Z8038A B1 | Plastic | 0/+70°C | 6MHz | |
| B6 | Plastic | -40/+85°C | | |
| D1 | Ceramic | 0/+70°C | | |
| D6 | Ceramic | -40/+85°C | | |
| Z8538 B1 | Plastic | 0/+70°C | 4MHz | Z8038 FIFO Buffer Universal Bus Compatible |
| B6 | Plastic | -40/+85°C | | |
| D1 | Ceramic | 0/+70°C | | |
| D6 | Ceramic | -40/+85°C | | |
| Z8538A B1 | Plastic | 0/+70°C | 6MHz | |
| B6 | | -40/+85°C | | |
| D1 | Ceramic | 0/+70°C | | |
| D6 | Ceramic | -40/+85°C | | |

## Packages (dimensions in mm)

Plastic




Ceramic