# Am53C80A

## SCSI Interface Controller

Advanced Micro Devices

## DISTINCTIVE CHARACTERISTICS

### SCSI Interface

- **Asynchronous interface to 4 megabytes per second**
- **Supports Initiator and Target roles**
- **Parity generation with optional checking**
- **Supports arbitration**
- **Direct control of all bus signals**
- **High-current outputs drive SCSI bus directly**

### CPU Interface

- **Memory or I/O-mapped interface**
- **DMA or programmed I/O**
- **Normal or Block mode DMA**
- **Optional CPU interrupts**

## GENERAL DESCRIPTION

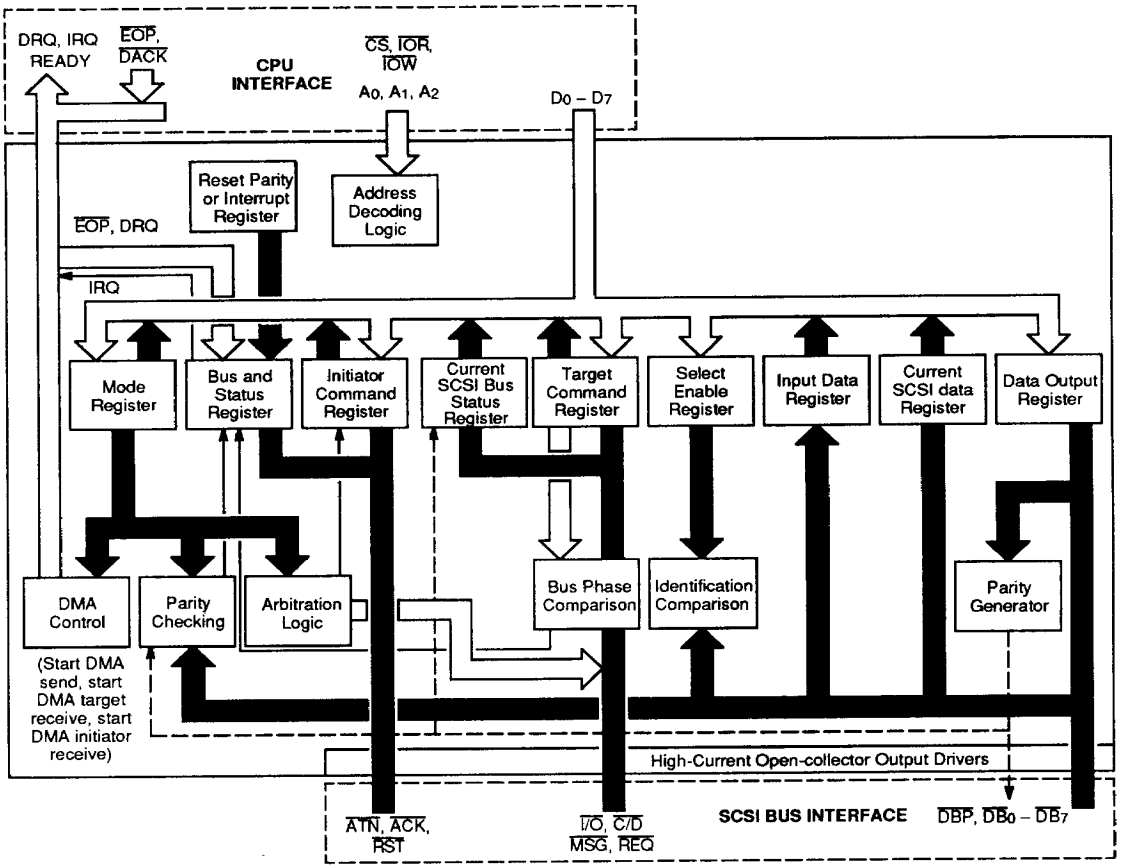
The Am53C80A Small Computer Systems Interface (SCSI) Controller is a CMOS device designed to accommodate the SCSI as defined by the ANSI X3T9.2 committee. The Am53C80A operates in both the Initiator and Target roles and can, therefore, be used in host adapter, host port, and formatter designs. This device supports Arbitration, including Reselection. Special high-current open-collector output drivers, capable of sinking 48 mA at 0.5 V, allow for direct connection to the SCSI Bus.

The Am53C80A communicates with the system microprocessor as a peripheral device. The chip is controlled by reading and writing several internal registers which may be addressed as standard or memory-mapped I/O. Minimal processor intervention is required for DMA transfers because the Am53C80A controls the necessary handshake signals. The Am53C80A interrupts the CPU when it detects a bus condition that requires attention. Normal and Block mode DMA is provided to match many popular DMA controllers.



Publication# 10665 Rev. B Amendment/0
Issue Date: November 1991

## BLOCK DIAGRAM

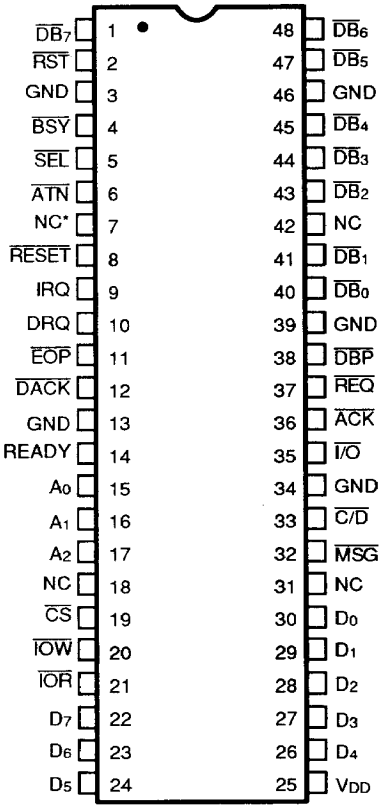



10665B-001A

## CONNECTION DIAGRAMS

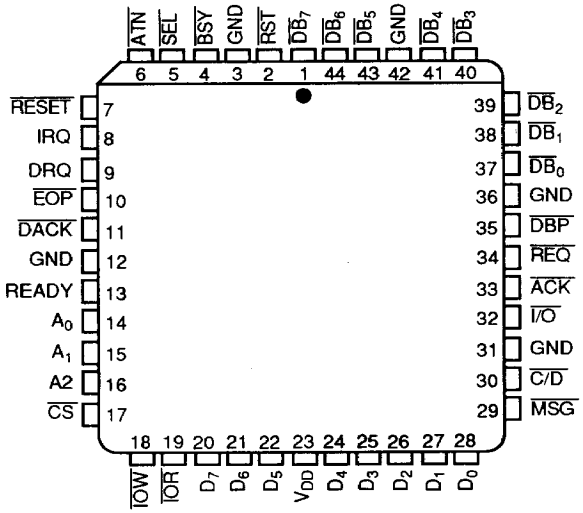
### Top View

**DIP**




10665B-002A

**PLCC**



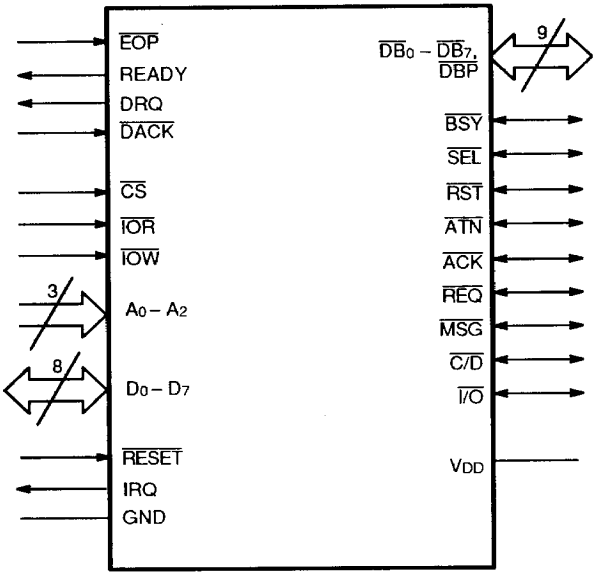

10665B-003A

**Notes:**

Pin 1 is marked for orientation.

*NC = No Connection

## LOGIC SYMBOL

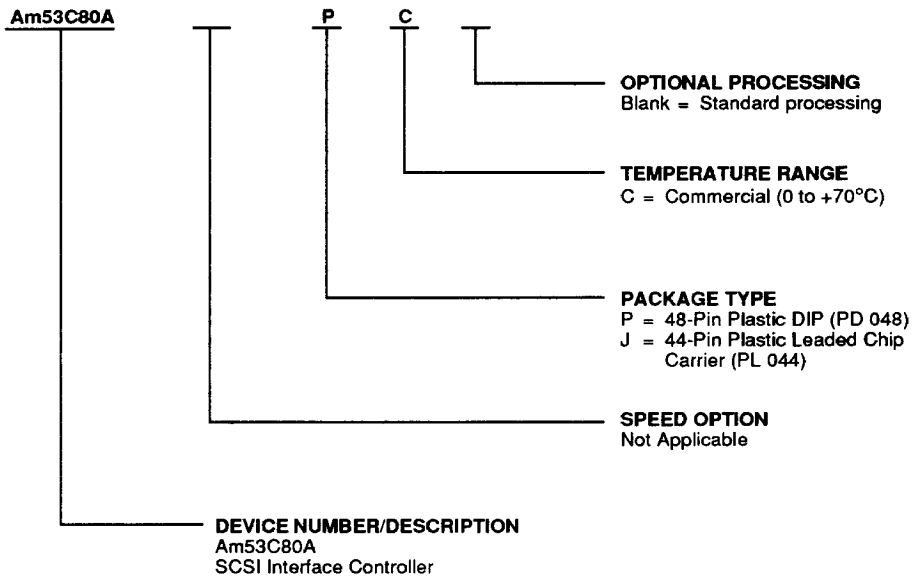



10665B-004A

## ORDERING INFORMATION

### Standard Products

AMD standard products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of:




| Valid Combinations | |
|---|---|
| Am53C80A | PC, JC |

**Valid Combinations**

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations or to check on newly released combinations.

## PIN DESCRIPTION

### Microprocessor Interface Signals

**$A_0$–$A_2$**

**Address Lines (Input)**

These signals are used with $\overline{CS}$, $\overline{IOR}$ or $\overline{IOW}$ to address all internal registers.

**$\overline{CS}$**

**Chip Select (Input, Active Low)**

$\overline{CS}$ enables a read or write of the internal register selected by $A_0$–$A_2$.

**$\overline{DACK}$**

**DMA Acknowledge (Input, Active Low)**

$\overline{DACK}$ resets DRQ and selects the data register for input or output data transfers.

**DRQ**

**DMA Request (Output)**

DRQ indicates that the data register is ready to be read or written. DRQ occurs only if DMA mode is TRUE in the Command Register. DRQ is cleared by $\overline{DACK}$.

**$D_0 - D_7$**

**Data Lines (Input/Output; Three-State, Active HIGH)**

Bidirectional microprocessor data bus lines.

**$\overline{EOP}$**

**End of Process (Input, Active Low)**

$\overline{EOP}$ is used to terminate a DMA transfer. If asserted during a DMA cycle, the current byte will be transferred, but no additional bytes will be requested.

**$\overline{IOR}$**

**I/O Read (Input, Active LOW)**

$\overline{IOR}$ is used to aread asn internal register selected by $\overline{CS}$ and $A_0 - A_2$. It also selects the Input Data Register when used with $\overline{DACK}$.

**$\overline{IOW}$**

**I/O Write (Input, Active LOW)**

$\overline{IOW}$ is used to write an internal register selected by $\overline{CS}$ and $A_0 - A_2$. It also selects the Output Data Register when used with $\overline{DACK}$.

**IRQ**

**Interrupt Request (Output)**

IRQ alerts a microprocessor of an error condition or an event completion.

**READY**

**Ready (Output)**

READY can be used to control the speed of Block mode DMA transfers. This signal goes active to indicate the chip is ready to send/receive data and remains FALSE after a transfer until the last byte is sent or until the DMA MODE bit is reset.

**$\overline{RESET}$**

**Reset (Input, Active Low)**

$\overline{RESET}$ clears all the SCSI controller registers. It does not force the SCSI $\overline{RST}$ signal to the active state.

### Power Signals

**$V_{DD}$**

**+5-Volt Power Supply**

**GND**

**Ground**

### SCSI Interface Signals

The following signals are all bidirectional, active-Low, open-collector signals. With 48-mA sink capability, all pins interface directly with the SCSI Bus.

**$\overline{ACK}$**

**Acknowledge (Input/Output; Open Collector, Active Low)**

Driven by an Initiator, $\overline{ACK}$ indicates an acknowledgment for a $\overline{REQ}$/$\overline{ACK}$ data-transfer handshake. In the Target role, $\overline{ACK}$ is received as a response to the $\overline{REQ}$ signal.

**$\overline{ATN}$**

**Attention (Input/Output; Open Collector, Active Low)**

Driven by an Initiator, $\overline{ATN}$ indicates an Attention condition. This signal is received in the Target role.

**$\overline{BSY}$**

**Busy (Input/Output; Open Collector, Active Low)**

This signal indicates that the SCSI Bus is being used and can be driven by both the Initiator and the Target device.

**$\overline{C/D}$**

**Control/Data (Input/Output; Open Collector, Active Low)**

A signal driven by the Target, $\overline{C/D}$ indicates Control or Data information is on the Data Bus. This signal is received by the Initiator.

**$\overline{I/O}$**

**Input/Output (Input/Output; Open Collector, Active Low)**

$\overline{I/O}$ is a signal driven by a Target which controls the direction of data movement on the SCSI Bus. TRUE indicates input to the intitiator. This signal is also used to distinguish between Selection and Reselection phases.

### $\overline{MSG}$

**Message (Input/Output; Open Collector, Active Low)**

$\overline{MSG}$ is a signal driven by the Target during the Message phase. This signal is received by the Initiator.

### $\overline{REQ}$

**Request (Input/Output; Open Collector, Active Low)**

Driven by a Target, $\overline{REQ}$ indicates a request for a $\overline{REQ}$/$\overline{ACK}$ data-transfer handshake. This signal is received by the Initiator.

### $\overline{RST}$

**SCSI Bus RESET (Input/Output; Open Collector, Active Low)**

The $\overline{RST}$ Signal indicates an SCSI Bus RESET condition. An internal 30 μA pull up transistor is built-in to prevent spurious interrupts.

### $\overline{DB_0}$–$\overline{DB_7}$, $\overline{DBP}$

**Data Bits, Parity Bit (Input/Output; Open Collector, Active Low)**

These eight data bits ($\overline{DB_0}$–$\overline{DB_7}$), plus a parity bit ($\overline{DBP}$) form the Data Bus. $\overline{DB_7}$ is the most significant bit (MSB) and has the highest priority during the Arbitration phase. Data parity is odd. Parity is always generated and optionally checked. Parity is not valid during Arbitration.

### $\overline{SEL}$

**Select (Input/Output; Open Collector, Active Low)**

$\overline{SEL}$ is used by an Initiator to select a Target, or by a Target to reselect an Initiator.

## FUNCTIONAL DESCRIPTION

### General

The Am53C80A Small Computer Systems Interface (SCSI) device appears as a set of eight registers to the controlling CPU. By reading and writing the appropriate registers, the CPU may initiate any SCSI Bus activity or may sample and assert any signal on the SCSI Bus. This allows the user to implement all or portions of the SCSI protocol in software. These registers are read (written) by activating $\overline{CS}$ with an address on $A_0$–$A_2$ and then issuing an $\overline{IOR}$ ($\overline{IOW}$) pulse. This section describes the operation of the internal registers.

**Table 1. Register Summary**

| Address | | | | |
|---|---|---|---|---|
| $A_2$ | $A_1$ | $A_0$ | R/W | Register Name |
| 0 | 0 | 0 | R | Current SCSI Data |
| 0 | 0 | 0 | W | Output Data |
| 0 | 0 | 1 | R/W | Initiator Command |
| 0 | 1 | 0 | R/W | Mode |
| 0 | 1 | 1 | R/W | Target Command |
| 1 | 0 | 0 | R | Current SCSI Bus Status |
| 1 | 0 | 0 | W | Select Enable |
| 1 | 0 | 1 | R | Bus and Status |
| 1 | 0 | 1 | W | Start DMA Send |
| 1 | 1 | 0 | R | Input Data |
| 1 | 1 | 0 | W | Start DMA Target Receive |
| 1 | 1 | 1 | R | Reset Parity/Interrupts |
| 1 | 1 | 1 | W | Start DMA Initiator Receive |

### Data Registers

The data registers are used to transfer SCSI commands, data, status, and message bytes between the microprocessor Data Bus and the SCSI Bus. The Am53C80A does not interpret any information that passes through the data registers. The data registers consist of the transparent Current SCSI Data Register, the Output Data Register, and the Input Data Register.

#### Current SCSI Data Register—Address 0 (Read Only)

The Current SCSI Data Register is a read-only register that allows the microprocessor to read the active SCSI Data Bus. This is accomplished by activating $\overline{CS}$ with an address on $A_2$–$A_0$ of 000 and issuing an $\overline{IOR}$ pulse. If parity checking is enabled, the SCSI Bus parity is checked at the beginning of the read cycle. This register is used during a programmed I/O data read or during Arbitration to check for higher priority arbitrating devices. Parity is not guaranteed valid during Arbitration.

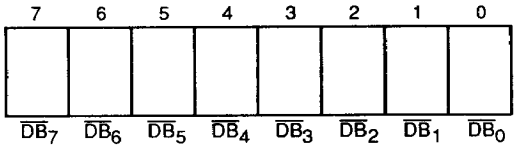



**Figure 1. Current SCSI Data Register**

#### Output Data Register—Address 0 (Write Only)

The Output Data Register is a write-only register that is used to send data to the SCSI Bus. This is accomplished by either using a normal CPU write, or under DMA control, by using $\overline{IOW}$ and $\overline{DACK}$. This register is also used to assert the proper ID bits or the SCSI Bus during the Arbitration and Selection phases.

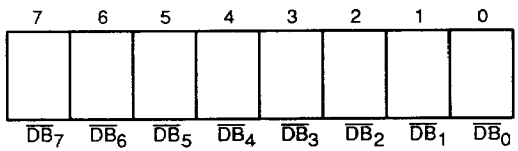



**Figure 2. Output Data Register**

#### Input Data Register—Address 6 (Read Only)

The Input Data Register is a read-only register that is used to read latched data from the SCSI Bus. Data is latched either during a DMA Target receive operation when $\overline{ACK}$ goes active or during a DMA Initiator receive when $\overline{REQ}$ goes active. The DMA Mode bit (port 2, bit 1) must be set before data can be latched in the Input Data Register. This register may be read under DMA control using $\overline{IOR}$ and $\overline{DACK}$. Parity is optionally checked when the Input Data Register is loaded.

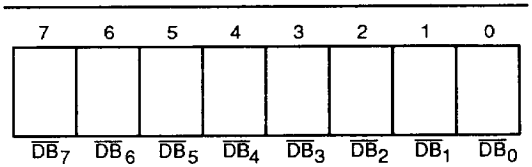



**Figure 3. Input Data Register**

### Initiator Command Register—Address 1 (Read/Write)

The Initiator Command Register is a read/write register that is used to assert certain SCSI Bus signals, to monitor those signals, and to monitor the progress of bus arbitration. Many of these bits are significant only when being used as Initiators; however, most can be used during Target role operation.

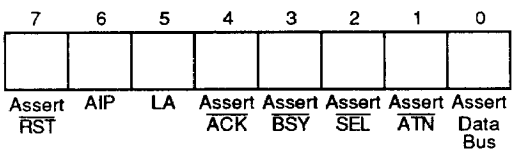



**Figure 4-1.**
**Initiator Command Register—Register Read**

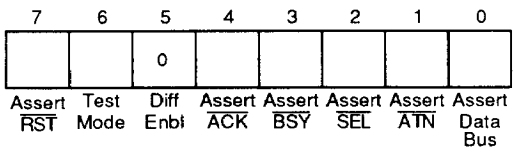



**Figure 4-2.**
**Initiator Command Register—Register Write**

The following describes the operation of all bits in the Initiator Command Register.

#### Bit 7—Assert RST

Whenever a one is written to bit 7 of the Initiator Command Register, the RST signal is asserted on the SCSI Bus. The RST signal will remain asserted until this bit is reset or until an external RESET occurs. After this bit is set (1), IRQ goes active and all internal logic and control registers are reset (except for the interrupt latch and the Assert RST bit). Writing a zero to bit 7 of the Initiator Command Register de-asserts the RST signal. Reading this register simply reflects the status of this bit.

#### Bit 6—AIP (Arbitration in Progress) (Read Bit)

This bit is used to determine if Arbitration is in progress. For this bit to be active, the Arbitrate bit (port 2, bit 0) must have been set previously. It indicates that a Bus-Free condition has been detected and that the chip has asserted BSY and the contents of the Output Data Register (port 0) onto the SCSI Bus. AIP will remain active until the Arbitrate bit is reset.

#### Bit 6—Test Mode (Write Bit)

This bit may be written during a test environment to disable all output drivers, effectively removing the Am53C80A from the circuit. Resetting this bit returns the part to normal operation.

#### Bit 5—LA (Lost Arbitration) (Read Bit)

This bit, when active, indicates that the Am53C80A detected a Bus-Free condition, arbitrated for use of the bus by asserting BSY and its ID on the Data Bus, and lost Arbitration due to SEL being asserted by another bus device. For this bit to be active, the Arbitrate bit (port 2, bit 0) must be active.

#### Bit 5—Diff Enbl (Differential Enable) (Write Bit)

This bit should be written with a zero for proper operation.

#### Bit 4—Assert ACK

This bit is used by the bus initiator to assert ACK on the SCSI Bus. In order to assert ACK, the Targetmode bit (port 2, bit 6) must be False. Writing a zero to this bit resets ACK on the SCSI Bus. Reading this register simply reflects the status of this bit.

#### Bit 3—Assert BSY

Writing a one (1) into this bit position asserts BSY onto the SCSI Bus. Conversely, a zero resets the BSY signal. Asserting BSY indicates a successful selection or reselection and resetting this bit creates a Bus-Disconnect condition. Reading this register simply reflects the status of this bit.

#### Bit 2—Assert SEL

Writing a one (1) into this bit position asserts SEL onto the SCSI Bus. SEL is normally asserted after Arbitration has been successfully completed. SEL may be de-asserted by resetting this bit to a zero. A read of this register simply reflects the status of this bit.

#### Bit 1—Assert ATN

ATN may be asserted on the SCSI Bus by setting this bit to a one (1) if the Targetmode bit (port 2, bit 6) is False. ATN is normally asserted by the initiator to request a Message Out bus phase. Note that since Assert SEL and Assert ATN are in the same register, a select with ATN may be implemented with one CPU write. ATN may be de-asserted by resetting this bit to zero. A read of this register simply reflects the status of this bit.

#### Bit 0—Assert Data Bus

The Assert Data Bus bit, when set, allows the contents of the Output Data Register to be enabled as chip outputs on the signals $\overline{DB_0}$–$\overline{DB_7}$. Parity is also generated and asserted on DBP.

When connected as an initiator, the outputs are only enabled if the Targetmode bit (port 2, bit 6) is False, the received signal I/O is False, and the phase signals (C/D, I/O, and MSG) match the contents of the Assert C/D, Assert I/O, and Assert MSG in the Target Command Register.

This bit should also be set during DMA send operations.

### Mode Register—Address 2 (Read/Write)

The Mode Register is used to control the operation of the chip. This register determines whether the Am53C80A operates as an Initiator or a Target, whether DMA transfers are being used, whether parity is checked, and whether interrupts are generated on

various external conditions. This register may be read to check the value of these internal control bits. Figure 5 describes the operation of these control bits.

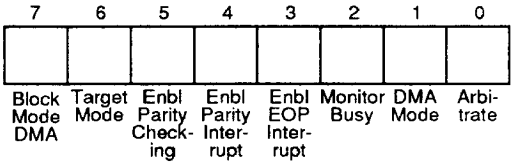



**Figure 5. Mode Register**

### Bit 7—Block Mode DMA

The Block Mode DMA bit controls the characteristics of the DMA DRQ-DACK handshake. When this bit is reset (0) and the DMA Mode bit is active (1), the DMA handshake uses the normal interlocked handshake, and the rising edge of DACK indicates the end of each byte being transferred. In block mode operations, Block Mode DMA bit set (1), and DMA Mode bit set (1), the end of IOR or IOW signifies the end of each byte transferred and DACK is allowed to remain active throughout the DMA operation. Ready can then be used to request the next transfer.

### Bit 6—Targetmode

The Targetmode bit allows the Am53C80A to operate as either an SCSI Bus Initiator, bit reset (0), or as an SCSI Bus Target device, bit set (1). In order for the signals ATN and ACK to be asserted on the SCSI Bus, the Targetmode bit must be reset (0). In order for the signals C/D, I/O, MSG, and REQ to be asserted on the SCSI Bus, the Targetmode bit must be set (1).

### Bit 5—Enable Parity Checking

The Enable Parity Checking bit determines whether parity errors will be ignored or saved in the parity error latch. If this bit is reset (0), parity will be ignored. Conversely, if this bit is set (1), parity errors will be saved.

### Bit 4—Enable Parity Interrupt

The Enable Parity Interrupt bit, when set (1), will cause an interrupt (IRQ) to occur if a parity error is detected. A parity interrupt will only be generated if the Enable Parity Checking bit (bit 5) is also enabled (1).

### Bit 3—Enable EOP Interrupt

The Enable EOP Interrupt bit, when set (1), causes an interrupt to occur when the EOP (End of Process) signal is received from the DMA controller logic.

### Bit 2—Monitor Busy

The Monitor Busy bit, when True (1), causes an interrupt to be generated for an unexpected loss of BSY. When the interrupt is generated due to loss of BSY, the lower six bits of the Initiator Command Register are reset (0) and all signals are removed from the SCSI Bus.

### Bit 1—DMA Mode

The DMA Mode bit is normally used to enable a DMA transfer and must be set (1) prior to writing ports 5 through 7. Ports 5 through 7 are used to start DMA transfers. The Targetmode bit (port 2, bit 6) must be consistent with writes to port 6 and 7 [i.e., set (1) for a write to port 6 and reset (0) for a write to port 7]. The control bit Assert Data Bus (port 1, bit 0) must be True (1) for all DMA send operations. In the DMA mode, REQ and ACK are automatically controlled.

The DMA Mode bit is not reset upon the receipt of an EOP signal. Any DMA transfer may be stopped by writing a zero into this bit location; however, care must be taken not to cause CS and DACK to be active simultaneously.

### Bit 0—Arbitrate

The Arbitrate bit is set (1) to start the Arbitration process. Prior to setting this bit, the Output Data Register should contain the proper SCSI device ID value. Only one data bit should be active for SCSI Bus Arbitration. The Am53C80A will wait for a Bus-Free condition before entering the Arbitration phase. The results of the Arbitration phase may be determined by reading the status bits LA and AIP (port 1, bits 5 and 6, respectively).

## Target Command Register—Address 3 (Read/Write)

When connected as a target device, the Target Command Register allows the CPU to control the SCSI Bus Information Transfer phase and/or to assert REQ simply by writing this register. The Targetmode bit (port 2, bit 6) must be True (1) for bus assertion to occur. The SCSI Bus phases are described in Table 2.

**Table 2. SCSI Information Transfer Phases**

| Bus Phase | Assert I/O | Assert C/D | Assert MSG |
|---|---|---|---|
| Data Out | 0 | 0 | 0 |
| Unspecified | 0 | 0 | 1 |
| Command | 0 | 1 | 0 |
| Message Out | 0 | 1 | 1 |
| Data In | 1 | 0 | 0 |
| Unspecified | 1 | 0 | 1 |
| Status | 1 | 1 | 0 |
| Message In | 1 | 1 | 1 |

When connected as an Initiator with DMA Mode True, if the phase lines (I/O, C/D, and MSG) do not match the phase bits in the Target Command Register, a phase-mismatch interrupt is generated when REQ goes active. In order to send data as an Initiator, the Assert I/O, Assert C/D, and Assert MSG bits must match the corresponding bits in the Current SCSI Bus Status Register (port 4). The Assert REQ bit (bit 3) has no meaning when operating as an Initiator.

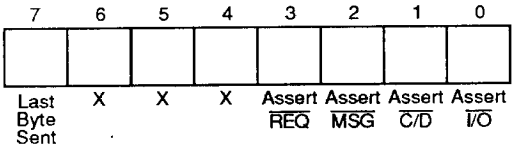



**Figure 6. Target Command Register**

The Am53C80A uses bit 7 of this register to determine when the last byte of a DMA transfer is sent to the SCSI Bus. This flag is necessary since the end of DMA bit in the Bus and Status Register only reflects when the last byte was received from the DMA.

## Current SCSI Bus Status Register—Address 4 (Read Only)

The Current SCSI Bus Status Register is a read-only register that is used to monitor seven SCSI Bus control signals plus the Data Bus parity bit. For example, an Initiator device can use this register to determine the current bus phase and to poll REQ for pending data transfers. This register may also be used to determine why a particular interrupt occurred. Figure 7 describes the Current SCSI Bus Status Register.

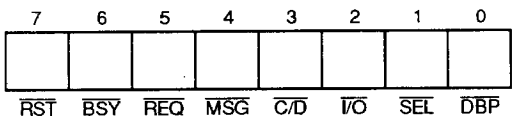



**Figure 7. Current SCSI Bus Status Register**

## Select Enable Register—Address 4 (Write Only)

The Select Enable Register is a write-only register which is used as a mask to monitor a signal ID during a selection attempt. The simultaneous occurrence of the correct ID bit, BSY False, and SEL True will cause an interrupt. This interrupt can be disabled by resetting all bits in this register. If the Enable Parity Checking bit (port 2, bit 5) is active (1), parity will be checked during selection.

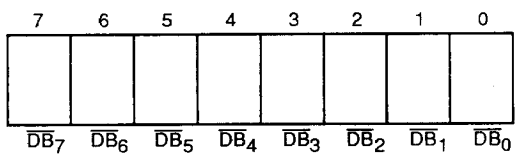



**Figure 8. Select Enable Register**

## Bus and Status Register—Address 5 (Read Only)

The Bus and Status Register is a read-only register that can be used to monitor the remaining SCSI control signals not found in the Current SCSI Bus Status Register (ATN and ACK), as well as six other status bits. Individual descriptions of each bit of the Bus and Status Register follow.

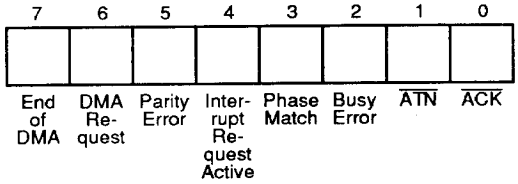



**Figure 9. Bus and Status Register**

### Bit 7—End of DMA Transfer

The End of DMA Transfer bit is set if EOP, DACK, and either IOR or IOW are simultaneously active for at least 100 ns. Since the EOP signal can occur during the last byte sent to the Output Data Register (port 0), the REQ and ACK signals should be monitored to ensure that the last byte has been transferred. This bit is reset when the DMA Mode bit is reset (0) in the Mode Register (port 2).

### Bit 6—DMA Request

The DMA Request bit allows the CPU to sample the output pin DRQ. DRQ can be cleared by asserting DACK or by resetting the DMA Mode bit (bit 1) in the Mode Register (port 2). The DRQ signal does not reset when a phase-mismatch interrupt occurs.

### Bit 5—Parity Error

This bit is set if a parity error occurs during a data receive or a device selection. The Parity Error bit can only be set (1) if the Enable Parity Check bit (port 2, bit 5) is active (1). This bit may be cleared by reading the Reset Parity/Interrupt Register (port 7).

### Bit 4—Interrupt Request Active

This bit is set if an enabled interrupt condition occurs. It reflects the current state of the IRQ output and can be cleared by reading the Reset Parity/Interrupt Register (port 7).

### Bit 3—Phase Match

The SCSI signals, MSG, C/D, and I/O, represent the current Information Transfer phase. The Phase Match bit indicates whether the current SCSI Bus phase matches the lower 3 bits of the Target Command Register. Phase Match is continuously updated and is only

significant when operating as a Bus Initiator. A phase match is required for data transfers to occur on the SCSI Bus.

#### Bit 2—Busy Error

The Busy Error bit is active if an unexpected loss of the $\overline{BSY}$ signal has occurred. This latch is set whenever the Monitor Busy bit (port 2, bit 2) is True and $\overline{BSY}$ is False. An unexpected loss of $\overline{BSY}$ will disable any SCSI outputs and will reset the DMA Mode bit (port 2, bit 1).

#### Bit 1—ATN

This bit reflects the condition of the SCSI Bus control signal $\overline{ATN}$. This signal is normally monitored by the Target device.

#### Bit 0—ACK

This bit reflects the condition of the SCSI Bus control signal $\overline{ACK}$. This signal is normally monitored by the Target device.

## DMA Registers

Three write-only registers are used to initiate all DMA activity. They are Start DMA Send (port 5), Start DMA Target Receive (port 6), and Start DMA Initiator Receive (port 7). Simply writing these registers starts the DMA transfers. Data presented to the Am53C80A on signals $D_0$–$D_7$ during the register write is meaningless and has no effect on the operation. Prior to writing these registers, the Block Mode DMA bit (bit 7), the DMA Mode bit (bit 1) and the Targetmode bit (bit 6) in the Mode Register (port 2) must be appropriately set. The individual registers are briefly described as follows.

#### Start DMA Send—Address 5 (Write Only)

This register is written to initiate a DMA send, from the DMA to the SCSI Bus, for either Initiator or Target role operations. The DMA Mode bit (port 2, bit 1) must be set prior to writing this register.

#### Start DMA Target Receive—Address 6 (Write Only)

This register is written to initiate a DMA receive—from the SCSI Bus to the DMA—for Target operation only. The DMA Mode bit (bit 1) and the Targetmode bit (bit 6) in the Mode Register (port 2) must both be set (1) prior to writing this register.

#### Start DMA Initiator Receive—Address 7 (Write Only)

This register is written to initiate a DMA receive—from the SCSI Bus to the DMA, for Initiator operation only. The DMA Mode bit (bit 6) must be False (0) in the Mode Register (port 2) prior to writing this register.

#### Reset Parity/Interrupt—Address 7 (Read Only)

Reading this register resets the Parity Error bit (bit 5), the Interrupt Request bit (bit 4), and the Busy Error bit (bit 2) in the Bus and Status Register (port 5).

## On-Chip SCSI Hardware Support

The Am53C80A is easy to use because of its simple architecture. The chip allows direct control and monitoring of the SCSI Bus by providing a latch for each signal. However, portions of the protocol define timings that are much too quick for traditional microprocessors to control. Therefore, hardware support has been provided for DMA transfers, bus arbitration, phase-change monitoring, bus disconnection, bus reset, parity generation, parity checking, and device selection/reselection.

Arbitration is accomplished using a Bus-Free filter to continuously monitor $\overline{BSY}$. If $\overline{BSY}$ remains inactive for at least 400 ns, then the SCSI Bus is considered free and Arbitration may begin. Arbitration will begin if the bus is free, $\overline{SEL}$ is inactive, and the Arbitration bit (port 2, bit 0) is active. Once arbitration has begun ($\overline{BSY}$ asserted), an arbitration delay of 2.2 µs must elapse before the Data Bus can be examined to determine if Arbitration has been won. This delay must be implemented in the controlling software driver.

The Am53C80A has no clock. Delays such as bus-free delay, bus-set delay, and bus-settle delay are implemented using gate delays. These delays may differ between devices because of inherent process variations, but are well within the proposed ANSI X3T9.2 specification.

## Interrupts

The Am53C80A provides an interrupt output (IRQ) to indicate a task completion or an abnormal bus occurrence. The use of interrupts is optional and may be disabled by resetting the appropriate bits in the Mode Register (port 2) or the Select Enable Register (port 4).

When an interrupt occurs, the Bus and Status Register and the Current SCSI Bus Status Register must be read to determine which condition created the interrupt. IRQ can be reset simply by reading the Reset Parity/Interrupt Register (port 7) or by an external chip reset ($\overline{RESET}$ active for 200 ns).

Assuming the Am53C80A has been properly initialized, an interrupt will be generated if the chip is selected or reselected, if an $\overline{EOP}$ signal occurs during a DMA transfer, if an SCSI Bus reset occurs, if a parity error occurs during a data transfer, if a bus phase mismatch occurs, or if an SCSI Bus disconnection occurs.

#### Selection/Reselection

The Am53C80A can generate a select interrupt if $\overline{SEL}$ is True (1), its device ID is True (1), and $\overline{BSY}$ is False for at least a bus-settle delay (400 ns). If $\overline{I/O}$ is active, this should be considered a reselect interrupt. The correct ID bit is determined by a match in the Select Enable Register (port 4). Only a single bit match is required to generate an interrupt. This interrupt may be disabled by writing zeros into all bits of the Select Enable Register.

If parity is supported, parity should also be good during the selection phase. Therefore, if the Enable Parity bit (port 2, bit 5) is active, then the Parity Error bit should be checked to ensure that a proper selection has occurred. The Enable Parity Interrupt bit need not be set for this interrupt to be generated.

The proposed SCSI specification also requires that no more than two device IDs be active during the selection process. To ensure this, the Current SCSI Data Register (port 0) should be read.

The proper values for the Bus and Status Register (port 5) and the Current SCSI Bus Status Register (port 4) are displayed in Figures 10 and 11, respectively.

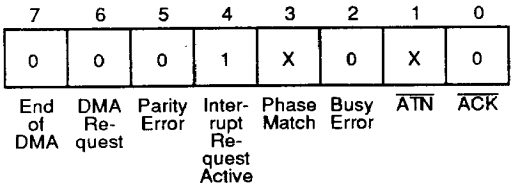



**Figure 10. Bus and Status Register**

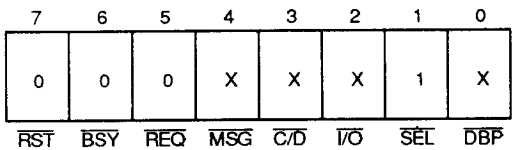



**Figure 11. Current SCSI Bus Status Register**

### End of Process (EOP) Interrupt

An End of Process ($\overline{EOP}$) signal which occurs during a DMA transfer (DMA Mode True) will set the End of DMA Status bit (port 5, bit 7) and will optionally generate an interrupt if the Enable EOP Interrupt bit (port 2, bit 3) is True. The $\overline{EOP}$ pulse will not be recognized (End Of DMA bit set) unless $\overline{EOP}$, $\overline{DACK}$, and either $\overline{IOR}$ or $\overline{IOW}$ are concurrently active for at least 100 ns. DMA transfers can still occur if $\overline{EOP}$ was not asserted at the correct time. This interrupt can be disabled by resetting the Enable EOP Interrupt bit.

The proper values for the Bus and Status Register (port 5) and the Current SCSI Bus Status Register (port 4) for this interrupt are shown in Figures 12 and 13, respectively.

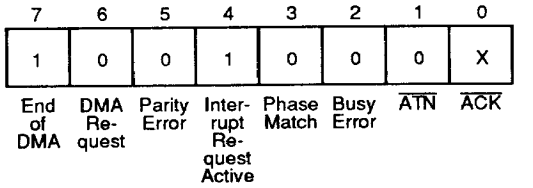



**Figure 12. Bus and Status Register**

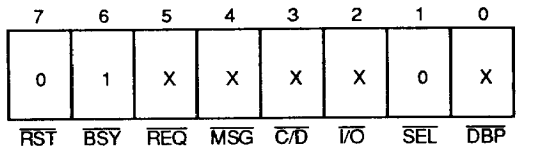



**Figure 13. Current SCSI Bus Status Register**

The End of DMA bit is used to determine when a block transfer is complete. Receive operations are complete when there is no data left in the chip and no additional handshakes occurring. The only exception to this is receiving data as an Initiator when the Target opts to send additional data for the same phase. In this case, $\overline{REQ}$ goes active and the new data is present in the Input Data Register. Since a phase-mismatch interrupt will not occur, $\overline{REQ}$ and $\overline{ACK}$ need to be sampled to determine that the Target is attempting to send more data.

For send operations, the End of DMA bit is set when the DMA finishes its transfer, but the SCSI transfer may still be in progress. If connected as a Target, $\overline{REQ}$ and $\overline{ACK}$ should be sampled until both are False. If connected as an Initiator, a phase change interrupt can be used to signal the completion of the previous phase. It is possible for the Target to request additional data for the same phase. In this case, a phase change will not occur and both $\overline{REQ}$ and $\overline{ACK}$ must be sampled to determine when the last byte was transferred.

### SCSI Bus Reset

The Am53C80A generates an interrupt when the $\overline{RST}$ signal transitions to True. The device releases all bus signals within a bus-clear delay (800 ns) of this transition. This interrupt also occurs after setting the Assert $\overline{RST}$ bit (port 1, bit 7). This interrupt cannot be disabled. (Note: $\overline{RST}$ is not latched in bit 7 of the Current SCSI Bus Status Register and may not be active when this port is read. For this case, the Bus Reset interrupt may be determined by default.)

The proper values for the Bus and Status Register (port 5) and the Current SCSI Bus Status Register (port 4) are displayed in Figures 14 and 15, respectively.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | X | 0 | 1 | X | 0 | X | X |
| End of DMA | DMA Request | Parity Error | Interrupt Request Active | Phase Match | Busy Error | $\overline{ATN}$ | $\overline{ACK}$ |

**Figure 14. Bus and Status Register**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| X | X | X | X | X | X | X | X |
| $\overline{RST}$ | $\overline{BSY}$ | $\overline{REQ}$ | $\overline{MSG}$ | $\overline{C/D}$ | $\overline{I/O}$ | $\overline{SEL}$ | $\overline{DBP}$ |

**Figure 15. Current SCSI Bus Status Register**

### Parity Error

An interrupt is generated for a received parity error if the Enable Parity Check (bit 5) and the Enable Parity Interrupt (bit 4) bits are set (1) in the Mode Register (port 2). Parity is checked during a read of the Current SCSI Data Register (port 0) and during a DMA receive operation. A parity error can be detected without generating an interrupt by disabling the Enable Parity Interrupt bit and checking the Parity Error flag (port 5, bit 5).

The proper values for the Bus and Status Register (port 5) and the Current SCSI Bus Status Register (port 4) are displayed in Figures 16 and 17, respectively.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | X | 1 | 1 | 1 | 0 | X | X |
| End of DMA | DMA Request | Parity Error | Interrupt Request Active | Phase Match | Busy Error | $\overline{ATN}$ | $\overline{ACK}$ |

**Figure 16. Bus and Status Register**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | 1 | X | X | X | X | 0 | X |
| $\overline{RST}$ | $\overline{BSY}$ | $\overline{REQ}$ | $\overline{MSG}$ | $\overline{C/D}$ | $\overline{I/O}$ | $\overline{SEL}$ | $\overline{DBP}$ |

**Figure 17. Current SCSI Bus Status Register**

### Bus Phase Mismatch

The SCSI phase lines are composed of the signals $\overline{I/O}$, $\overline{C/D}$, and $\overline{MSG}$. These signals are compared with the corresponding bits in the Target Command Register: Assert $\overline{I/O}$ (bit 0), Assert $\overline{C/D}$ (bit 1), and Assert $\overline{MSG}$ (bit 2). The comparison occurs continually and is reflected in the Phase Match bit (bit 3) of the Bus and Status Register (port 5). If the DMA Mode bit (port 2, bit 1) is active and a phase mismatch occurs when $\overline{REQ}$ changes from False to True, an interrupt (IRQ) is generated.

A phase mismatch prevents the recognition of $\overline{REQ}$ and removes the chip from the bus during an initiator send operation [$\overline{DB_0}$–$\overline{DB_7}$ and $\overline{DBP}$ will not be driven even though the Assert Data Bus bit (port 1, bit 0) is active]. This interrupt is only significant when connected as an initiator and may be disabled by resetting the DMA Mode bit. (Note: It is possible for this interrupt to occur when connected as a Target if another device is driving the phase lines to a different state.)

The proper values for the Bus and Status Register (port 5) and the Current SCSI Bus Status Register (port 4) are displayed in Figures 18 and 19, respectively.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 0 | X | 0 |
| End of DMA | DMA Request | Parity Error | Interrupt Request Active | Phase Match | Busy Error | $\overline{ATN}$ | $\overline{ACK}$ |

**Figure 18. Bus and Status Register**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | 1 | X | X | X | X | 0 | X |
| $\overline{RST}$ | $\overline{BSY}$ | $\overline{REQ}$ | $\overline{MSG}$ | $\overline{C/D}$ | $\overline{I/O}$ | $\overline{SEL}$ | $\overline{DBP}$ |

**Figure 19. Current SCSI Bus Status Register**

### Loss of $\overline{BSY}$

If the Monitor Busy bit (bit 2) in the Mode Register (port 2) is active, an interrupt will be generated if the $\overline{BSY}$ signal goes False for at least a bus-settle delay (400 ns). This interrupt may be disabled by resetting the Monitor Busy bit. Register values are displayed in Figures 20 and 21.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | X | 1 | 0 | 0 |
| End of DMA | DMA Request | Parity Error | Interrupt Request Active | Phase Match | Busy Error | $\overline{ATN}$ | $\overline{ACK}$ |

**Figure 20. Bus and Status Register**

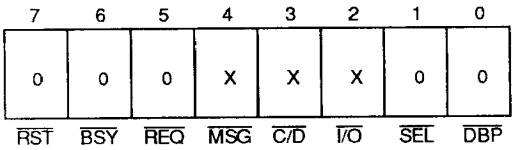



**Figure 21. Current SCSI Bus Status Register**

## Reset Conditions

Three possible reset situations exist with the Am53C80A.

### Hardware Chip Reset

When the signal $\overline{RST}$ is active for at least 200 ns, the Am53C80A device is re-initialized and all internal logic and control registers of SCSI are cleared. This is a chip reset only and does not create an SCSI Bus-Reset condition.

### SCSI Bus Reset ($\overline{RST}$) Received

When an SCSI $\overline{RST}$ signal is received, an IRQ interrupt is generated and a SCSI chip reset is performed. All internal logic and registers are cleared, except for the IRQ interrupt latch and the Assert $\overline{RST}$ bit (bit 7) in the Initiator Command Register (port 1). (Note: The $\overline{RST}$ signal may be sampled by reading the Current SCSI Bus Status Register (port 4); however, this signal is not latched and may not be present when this port is read.)

### SCSI Bus Reset ($\overline{RST}$) Issued

If the CPU sets the Assert $\overline{RST}$ bit (bit 7) in the Initiator Command Register (port 1), the $\overline{RST}$ signal goes active on the SCSI Bus and an internal reset is performed. Again, all internal logic and registers are cleared except for the IRQ interrupt latch and the Assert $\overline{RST}$ bit (bit 7) in the Initiator Command Register (port 1). The $\overline{RST}$ signal will continue to be active until the Assert $\overline{RST}$ bit is reset or until a hardware reset occurs.

## Data Transfers

Data may be transferred between SCSI Bus devices in one of four modes: (1) Programmed I/O, (2) Normal DMA, (3) Block Mode DMA, or (4) Pseudo DMA. The following sections describe these modes in detail. (Note: For all data transfer operations, $\overline{DACK}$ and $\overline{CS}$ should never be active simultaneously.)

### Programmed I/O Transfers

Programmed I/O is the most primitive form of data transfer. The $\overline{REQ}$ and $\overline{ACK}$ handshake signals are individually monitored and asserted by reading and writing the appropriate register bits. This type of transfer is normally used when transferring small blocks of data, such as command blocks or message and status bytes.

An Initiator send operation would begin by setting the $\overline{C/D}$, $\overline{I/O}$, and $\overline{MSG}$ bits in the Target Command Register to the correct state so that a phase match exists. In addition to the phase match condition, it is necessary for the Assert Data Bus bit (port 1, bit 0) to be True and the received I/O signal to be False for the Am53C80A to send data.

For each transfer, the data is loaded into the Output Data Register (port 0). The CPU then waits for the $\overline{REQ}$ bit (port 4, bit 5) to become active. Once $\overline{REQ}$ goes active, the Phase Match bit (port 5, bit 3) is checked and the Assert $\overline{ACK}$ bit (port 1, bit 4) is set. The $\overline{REQ}$ bit is sampled until it becomes False and the CPU resets the Assert $\overline{ACK}$ bit to complete the transfer.

### Normal DMA Mode

DMA transfers are normally used for large block transfers. The SCSI chip outputs a DMA request (DRQ) whenever it is ready for a byte transfer. External DMA logic uses this DRQ signal to generate $\overline{DACK}$ and an $\overline{IOR}$ or an $\overline{IOW}$ pulse to the Am53C80A. DRQ goes inactive when $\overline{DACK}$ is asserted, and $\overline{DACK}$ goes inactive some time after the minimum read or write pulse width. This process is repeated for every byte. For this mode, $\overline{DACK}$ should not be allowed to cycle unless a transfer is taking place.

### Block Mode DMA

Some popular DMA controllers such as the Am9517A provide a Block Mode DMA transfer. This type of transfer allows the DMA controller to transfer blocks of data without relinquishing the use of the Data Bus to the CPU after each byte is transferred; thus, faster transfer rates are achieved by eliminating the repetitive access and release of the CPU Bus.

If the Block Mode DMA bit (port 2, bit 7) is active, the Am53C80A will begin the transfer by asserting DRQ. The DMA controller then asserts $\overline{DACK}$ for the remainder of the block transfer. DRQ goes inactive for the duration of the transfer.

Non-Block Mode DMA transfers end when $\overline{DACK}$ goes False, whereas Block mode transfers end when $\overline{IOR}$ or $\overline{IOW}$ becomes inactive. Since this is the case, DMA transfers may be started sooner in a Block Mode transfer.

To obtain optimum performance in Block Mode operation, the DMA logic may optionally use the normal DMA mode interlocking handshake. READY is still available to throttle the DMA transfer, but DRQ is 30 to 40 ns faster than READY and may be used to start the cycle sooner.

The methods described under "Halting a DMA Operation" apply for all DMA operations.

### Pseudo DMA Mode

To avoid the tedium of monitoring and asserting the request/acknowledge handshake signals for programmed I/O transfers, the system may be designed to implement a pseudo DMA mode. This mode is implemented by programming the Am53C80A to operate in the DMA mode, but using the CPU to emulate the DMA handshake.

DRQ may be detected by polling the DMA Request bit (bit 6) in the Bus and Status Register (port 5), by sampling the signal through an external port, or by using it to generate a CPU interrupt. Once DRQ is detected, the CPU can perform a read or write data transfer. This CPU read/write is externally decoded to generate the appropriate $\overline{DACK}$ and $\overline{IOR}$ or $\overline{IOW}$ signals.

Often, external decoding logic is necessary to generate the Am53C80A $\overline{CS}$ signal. This same logic may be used to generate $\overline{DACK}$ at no extra system cost and provide an increased performance in programmed I/O transfers.

### Halting a DMA Operation

The $\overline{EOP}$ signal is not the only way to halt a DMA transfer. A bus phase mismatch or a reset of the DMA Mode bit (port 2, bit 1) can also terminate a DMA cycle for the current bus phase.

### Using the $\overline{EOP}$ Signal

If $\overline{EOP}$ is used, it should be asserted for at least 100 ns while $\overline{DACK}$ and $\overline{IOR}$ or $\overline{IOW}$ are simultaneously active. Note, however, that if $\overline{IOR}$ or $\overline{IOW}$ is not active an interrupt will be generated, but the DMA activity will continue. The $\overline{EOP}$ signal does not reset the DMA Mode bit. Since the $\overline{EOP}$ signal can occur during the last byte sent to the Output Data Register (port 0), the $\overline{REQ}$ and $\overline{ACK}$ signals should be monitored to ensure that the last byte has transferred.

### Bus Phase Mismatch Interrupt

A bus phase mismatch interrupt may be used to halt the transfer if operating as an Initiator. Using this method frees the host from maintaining a data length counter and frees the DMA logic from providing the $\overline{EOP}$ signal. If performing an Initiator send operation, the Am53C80A requires $\overline{DACK}$ to cycle before $\overline{ACK}$ goes inactive. Since phase changes cannot occur if $\overline{ACK}$ is active, either $\overline{DACK}$ must be cycled after the last byte is sent or the DMA Mode bit must be reset in order to receive the phase mismatch interrupt.

### Resetting the DMA Mode Bit

A DMA operation may be halted at any time simply by resetting the DMA Mode bit. It is recommended that the DMA Mode bit be reset after receiving an $\overline{EOP}$ or bus phase-mismatch interrupt. The DMA Mode bit must then be set before writing any of the start DMA registers for subsequent bus phases.

If resetting the DMA Mode bit is used instead of $\overline{EOP}$ for Target role operation, then care must be taken to reset this bit at the proper time. If receiving data as a Target device, the DMA Mode bit must be reset once the last DRQ is received and before $\overline{DACK}$ is asserted to prevent an additional $\overline{REQ}$ from occurring. Resetting this bit causes DRQ to go inactive. However, the last byte received remains in the Input Data Register and may be obtained either by performing a normal CPU read or by cycling $\overline{DACK}$ and $\overline{IOR}$. In most cases $\overline{EOP}$ is easier to use when operating as a Target device.

## Flowcharts

Flowcharts are provided (see Figures 22 through 25) as a guideline to facilitate your firmware development. Firmware will vary depending on the application and the level of the SCSI protocol being supported.

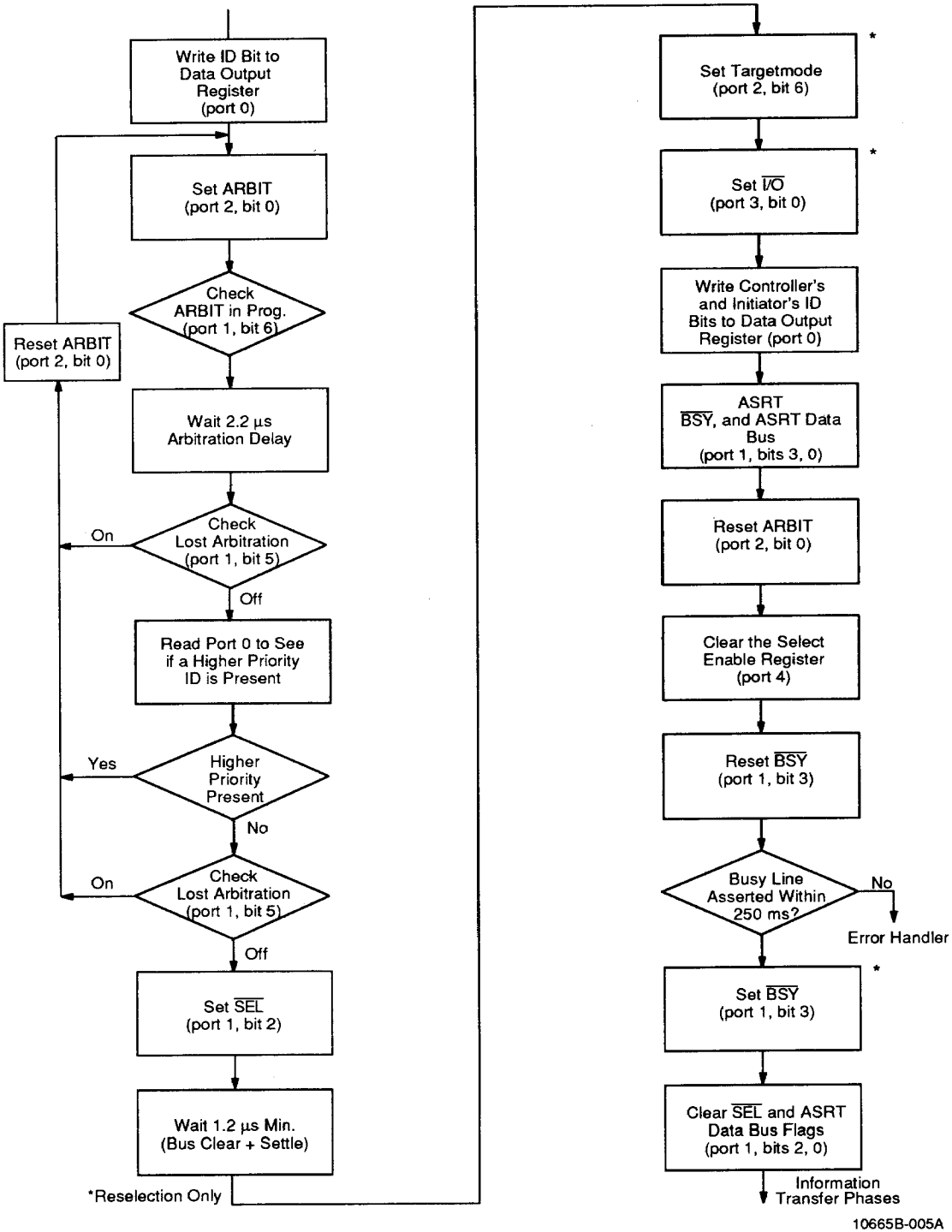



**Figure 22. Arbitration and (Re) Selection**

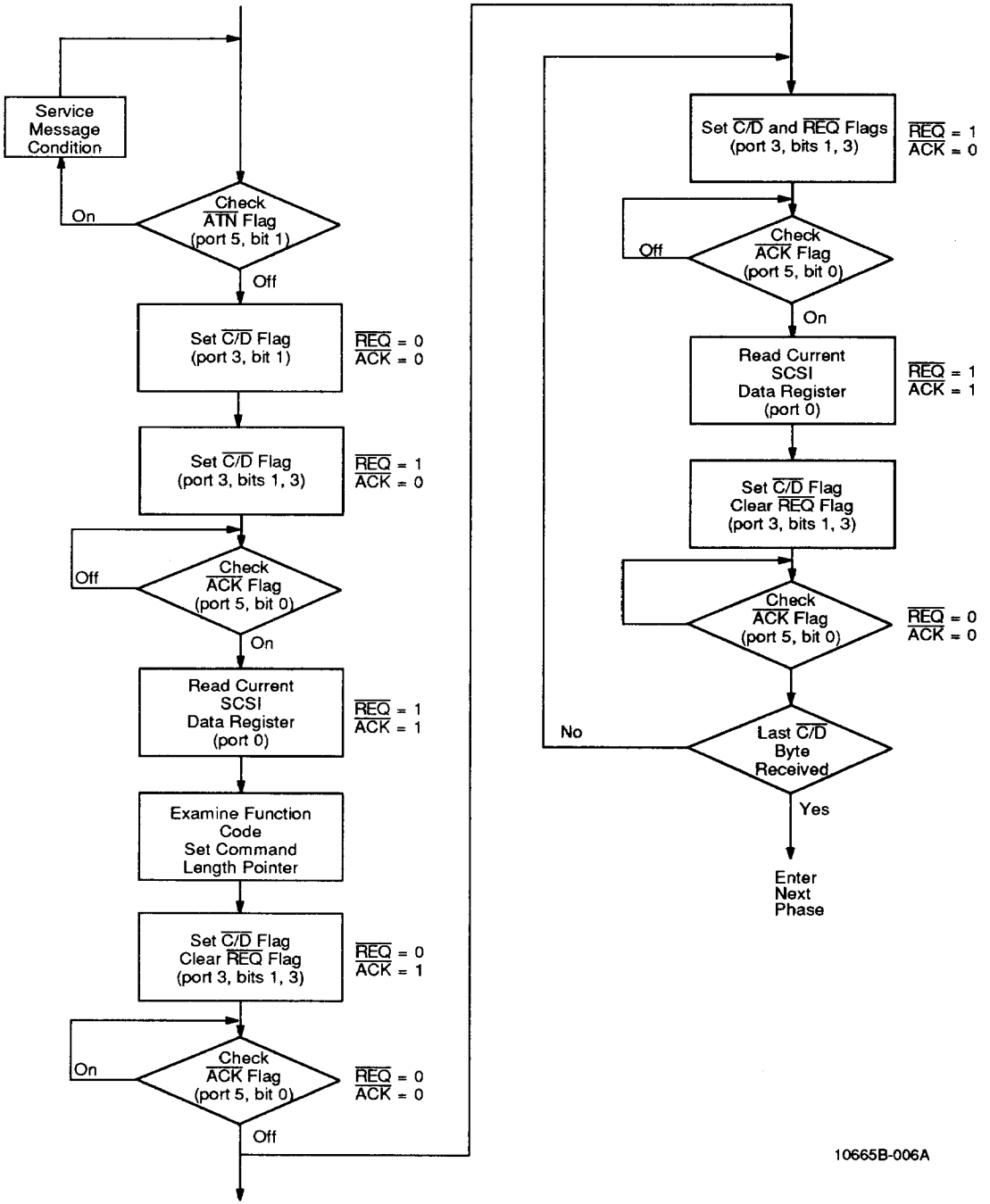



**Figure 23. Command Transfer Phase (Target)**

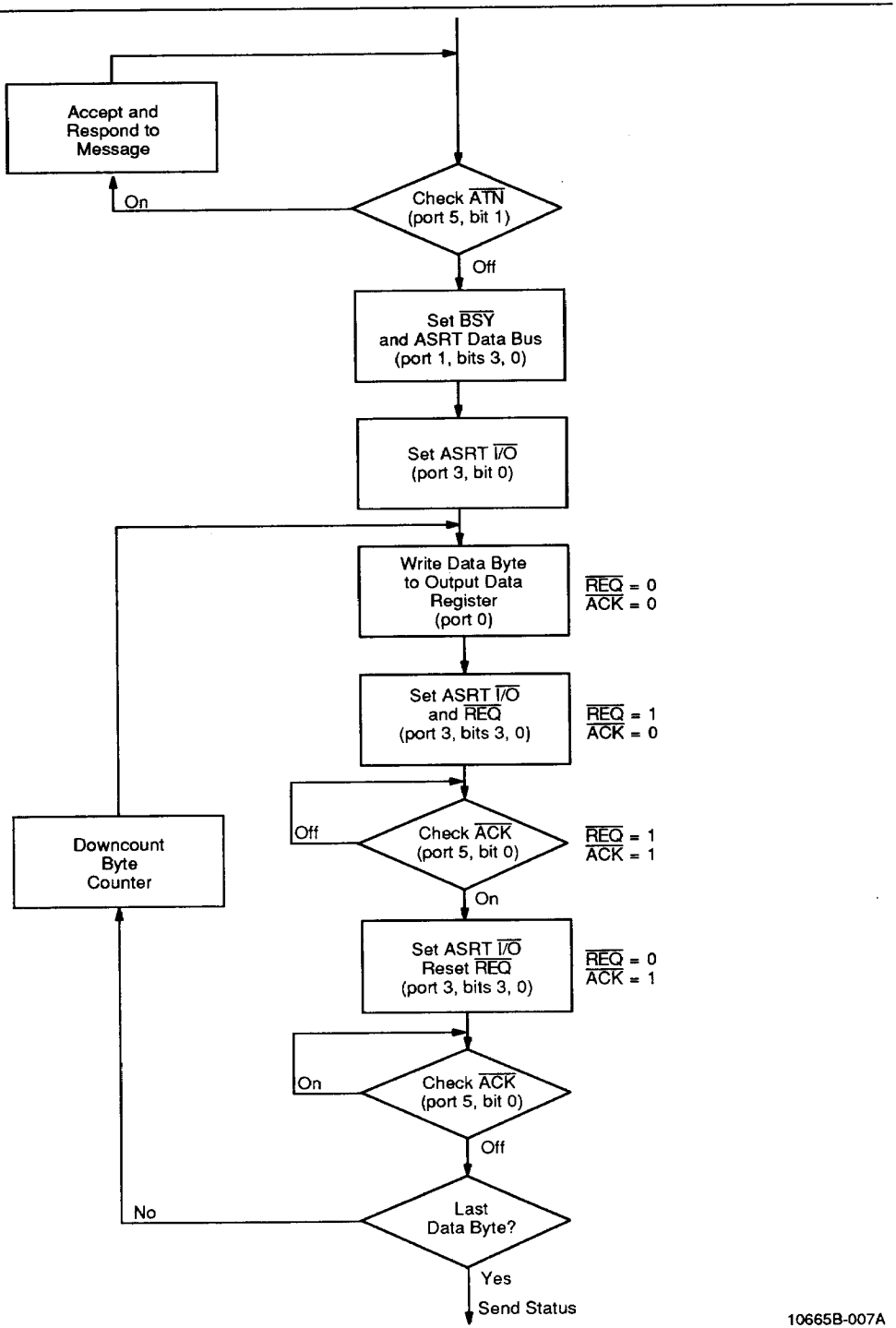



**Figure 24. Data Transfer to Host via Programmed I/O**

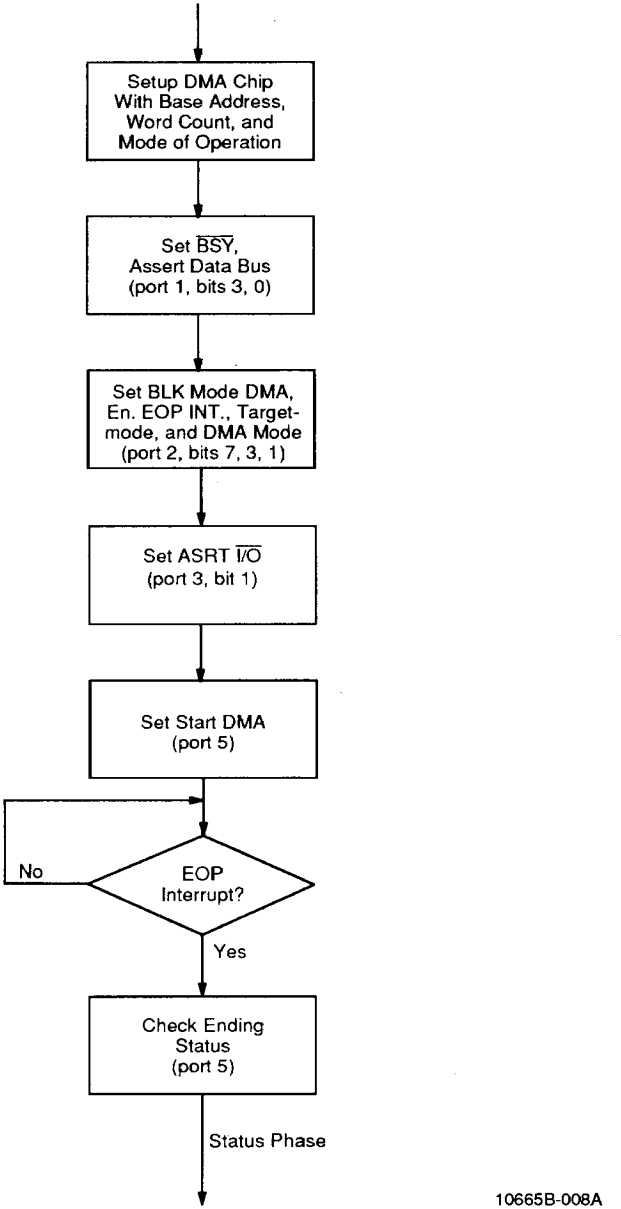



**Figure 25. Data Transfer via DMA**

**Read** **Write**

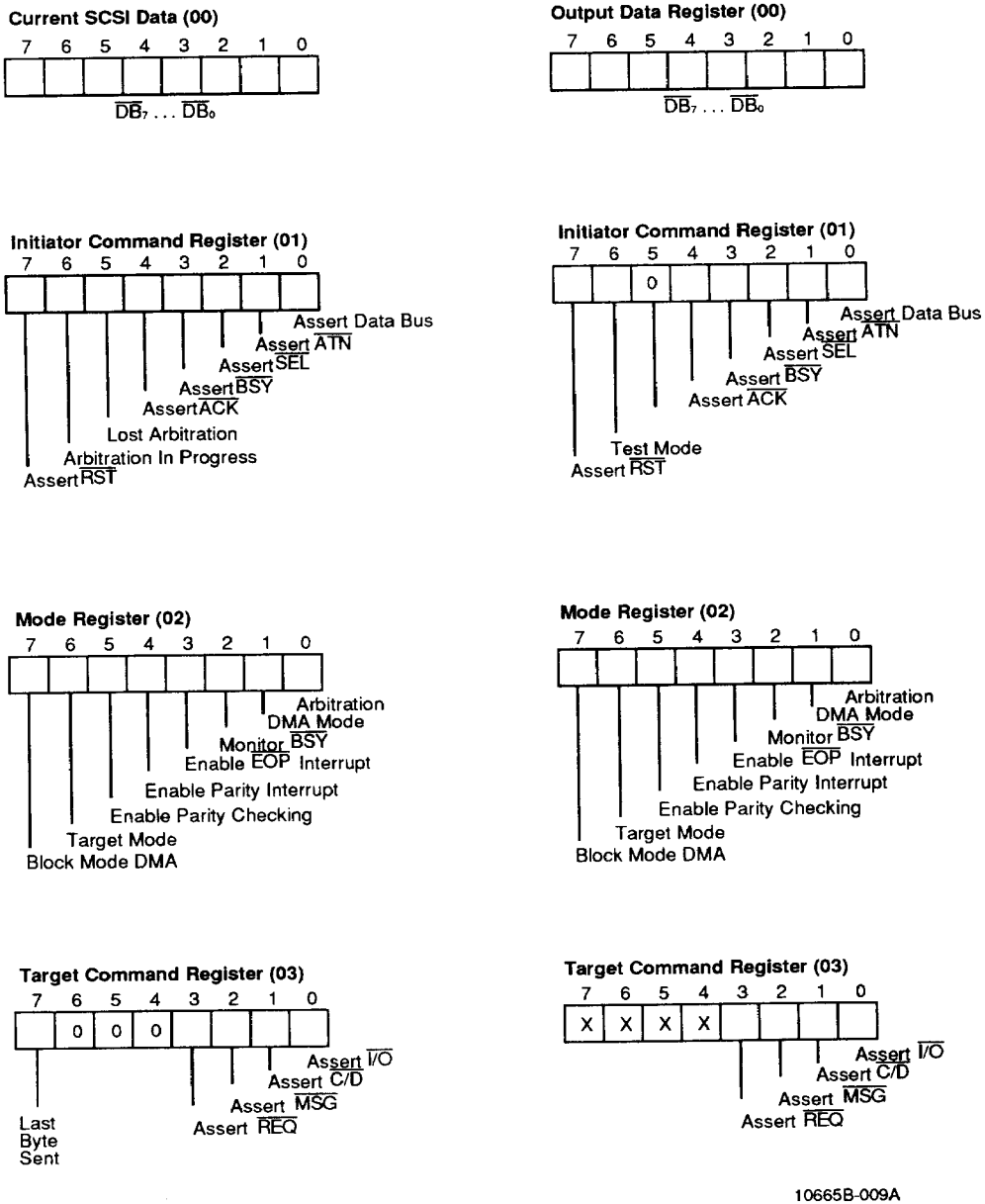
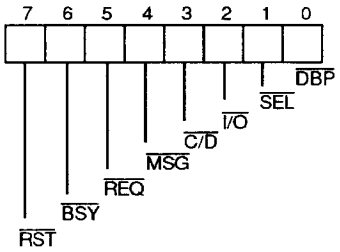


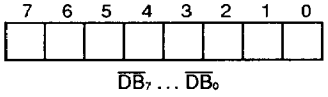

**Figure 26. Register Reference Chart**

**Read** **Write**

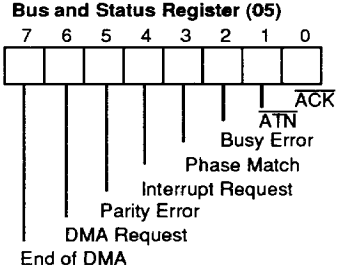
**Current SCSI Bus Status (04)**



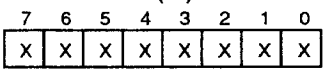

**Select Enable Register (04)**



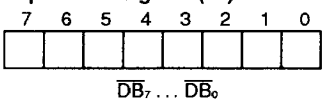

**Bus and Status Register (05)**



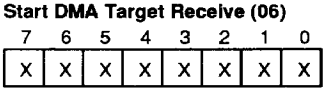

**Start DMA Send (05)**



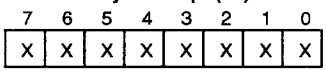

**Input Data Register (06)**



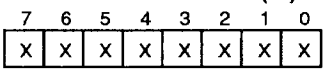

**Start DMA Target Receive (06)**




**Reset Parity/Interrupt (07)**




**Start DMA Initiator Receive (07)**




10665B-010A

**Note:**
X = Don't Care

**Figure 26. Register Reference Chart (continued)**

## ABSOLUTE MAXIMUM RATINGS

| | |
|---|---|
| Storage Temperature | −65 to +150°C |
| Supply Voltage on Any Pin with Respect to Ground | −0.5 to +7.0 V |
| Power Dissipation | 100 mW |

*Stresses above those listed under Absolute Maximum Ratings may cause permanent device failure. Functionality at or above these limits is not implied. Exposure to absolute maximum ratings for extended periods may affect device reliability.*

## OPERATING RANGES

**Commercial (C) Devices**

| | |
|---|---|
| Ambient Temperature ($T_A$) | 0 to +70°C |
| Supply Voltage ($V_{CC}$) | +4.75 to +5.25 V |

*Operating ranges define those limits between which the functionality of the device is guaranteed.*

## DC CHARACTERISTICS over operating ranges

| Parameter Description | Test Conditions | Min. | Max. | Unit |
|---|---|---|---|---|
| **Input Signal Requirements** | | | | |
| HIGH-Level, Input $V_{IH}$ | | 2.0 | 5.25 | V |
| LOW-Level, Input $V_{IL}$ | | −0.3 | 0.8 | V |
| HIGH-Level Input Current, $I_{IH}$, on: | | | | |
| SCSI Bus Pin except $\overline{RST}$ | $V_{IH}$ = 5.25 V, $V_{IL}$ = 0 | | 50 | µA |
| All Other Pins | | | 10 | |
| LOW-Level Input Current, $I_{IL}$, on: | | | | |
| SCSI Bus Pins except $\overline{RST}$ | $V_{IH}$ = 5.25 V, $V_{IL}$ = 0 | | −50 | µA |
| All Other Pins | | | −10 | |
| **Output Signal Requirements** | | | | |
| HIGH-Level Output on All Pins | $V_{DD}$ = 4.75 V, $I_{OH}$ = −3.0 mA | 2.4 | | V |
| LOW-Level Output on: | | | | |
| SCSI Bus Pins | $V_{DD}$ = 4.75 V, $I_{OL}$ = 48.0 mA | | 0.5 | V |
| All Other Pins | $V_{DD}$ = 4.75 V, $I_{OL}$ = 7.0 mA | | 0.5 | V |

## KEY TO SWITCHING WAVEFORMS

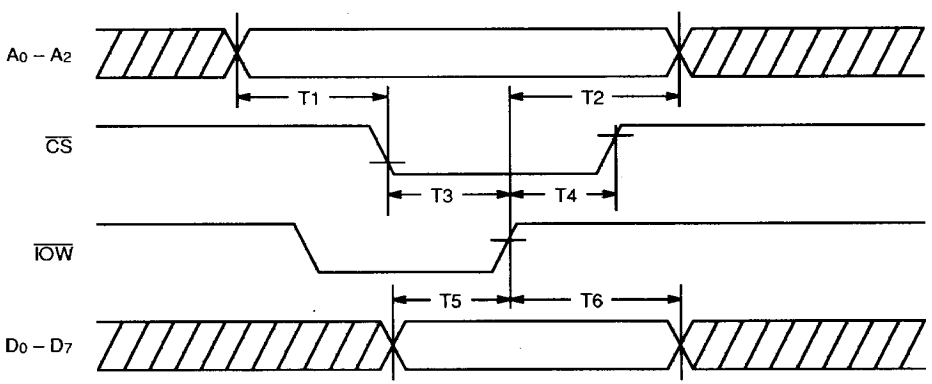
| WAVEFORM | INPUTS | OUTPUTS |
|---|---|---|
| | Must be Steady | Will be Steady |
| | May Change from H to L | Will be Changing from H to L |
| | May Change from L to H | Will be Changing from L to H |
| | Don't Care, Any Change Permitted | Changing, State Unknown |
| | Does Not Apply | Center Line is High-Impedance "Off" State |

KS000010

## SWITCHING CHARACTERISTICS/WAVEFORMS

| Name | Parameter Description | Min. | Max. | Unit |
|---|---|---|---|---|
| T1 | Address Setup to Write Enable* | 10 | | ns |
| T2 | Address Hold from End Write Enable* | 0 | | ns |
| T3 | Write Enable Width* | 10 | | ns |
| T4 | Chip Select Hold from End of $\overline{IOW}$ | 0 | | ns |
| T5 | Data Setup to End of Write Enable* | 10 | | ns |
| T6 | Data Hold Time from End of $\overline{IOW}$ | 10 | | ns |

*Write Enable is the occurence of $\overline{IOW}$ and $\overline{CS}$.



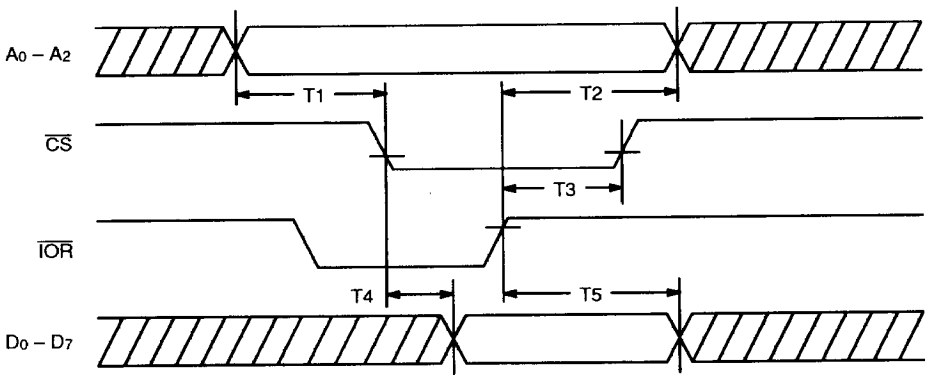

10665B-011B

**CPU Write Cycle**

## SWITCHING CHARACTERISTICS/WAVEFORMS

| Name | Parameter Description | Min. | Max. | Unit |
|---|---|---|---|---|
| T1 | Address Setup to Read Enable* | 10 | | ns |
| T2 | Address Hold from End Read Enable* | 0 | | ns |
| T3 | Chip Select Hold from End of $\overline{IOR}$ | 0 | | ns |
| T4 | Data Access Time from Read Enable* | | 40 | ns |
| T5 | Data Hold Time from End of $\overline{IOR}$ | 20 | | ns |

*Read Enable is the occurence of $\overline{IOR}$ and $\overline{CS}$.




10665B-012A

**CPU Read Cycle**

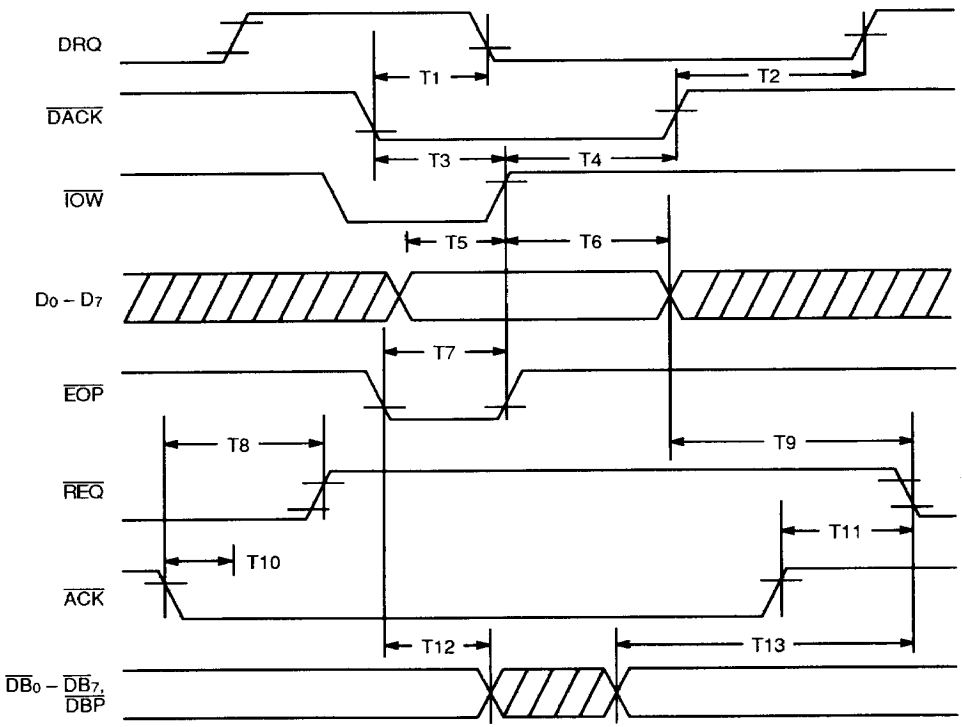
## SWITCHING CHARACTERISTICS/WAVEFORMS

| Name | Parameter Description | Min. | Max. | Unit |
|---|---|---|---|---|
| T1 | DRQ FALSE from $\overline{DACK}$ TRUE | | 35 | ns |
| T2 | $\overline{DACK}$ FALSE to DRQ TRUE | 30 | | ns |
| T3 | Write Enable Width* | 40 | | ns |
| T4 | $\overline{DACK}$ Hold from End of $\overline{IOW}$ | 0 | | ns |
| T5 | Data Setup to End of Write Enable* | 5 | | ns |
| T6 | Data Hold Time from End of $\overline{IOW}$ | 5 | | ns |
| T7 | Width of $\overline{EOP}$ Pulse (Note 1) | 40 | | ns |
| T8 | $\overline{ACK}$ TRUE to $\overline{REQ}$ FALSE | | 85 | ns |
| T9 | $\overline{REQ}$ from End of $\overline{DACK}$ ($\overline{ACK}$ FALSE) | | 40 | ns |
| T10 | $\overline{ACK}$ TRUE to DRQ TRUE (Target) | | 90 | ns |
| T11 | $\overline{REQ}$ from End of $\overline{ACK}$ ($\overline{DACK}$ FALSE) | | 30 | ns |
| T12 | Data Hold from Write Enable | 0 | | ns |
| T13 | Data Setup to $\overline{REQ}$ TRUE (Target) | 40 | | ns |

*Write Enable is the occurence of $\overline{IOW}$ and $\overline{DACK}$.

**Note:**

1. $\overline{EOP}$, $\overline{IOW}$, and $\overline{DACK}$ must be concurrently TRUE for at least T7 for proper recognition of the $\overline{EOP}$ pulse.




10665B-013B

**DMA Write (Non-Block Mode) Target Send Cycle**

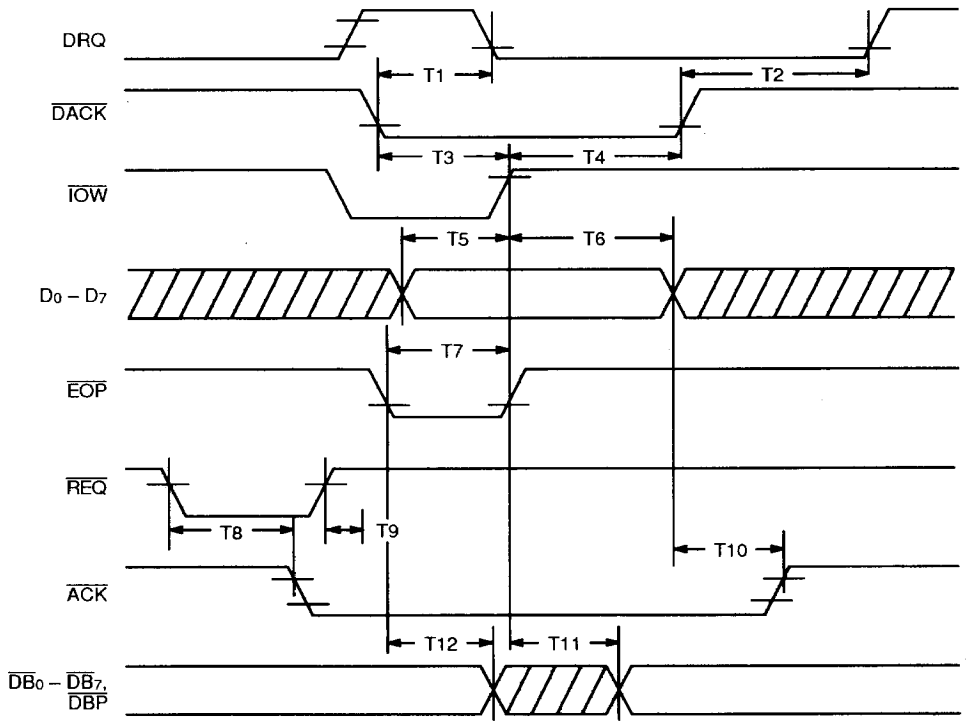
## SWITCHING CHARACTERISTICS/WAVEFORMS

| Name | Parameter Description | Min. | Max. | Unit |
|---|---|---|---|---|
| T1 | DRQ FALSE from $\overline{DACK}$ TRUE | | 35 | ns |
| T2 | $\overline{DACK}$ FALSE to DRQ TRUE | 30 | | ns |
| T3 | Write Enable Width* | 40 | | ns |
| T4 | $\overline{DACK}$ Hold from End of $\overline{IOW}$ | 0 | | ns |
| T5 | Data Setup to End of Write Enable* | 10 | | ns |
| T6 | Data Hold Time from End of $\overline{IOW}$ | 10 | | ns |
| T7 | Width of $\overline{EOP}$ Pulse (Note 1) | 40 | | ns |
| T8 | $\overline{REQ}$ TRUE to $\overline{ACK}$ TRUE | | 85 | ns |
| T9 | $\overline{REQ}$ FALSE to DRQ TRUE | | 45 | ns |
| T10 | $\overline{DACK}$ FALSE to $\overline{ACK}$ FALSE | | 45 | ns |
| T11 | $\overline{IOW}$ FALSE to Valid SCSI Data | | 65 | ns |
| T12 | Data Hold from Write Enable | 0 | | ns |

*Write Enable is the occurence of $\overline{IOW}$ and $\overline{DACK}$.

**Note:**

1. $\overline{EOP}$, $\overline{IOW}$, and $\overline{DACK}$ must be concurrently TRUE for at least T7 for proper recognition of the $\overline{EOP}$ pulse.




10665B-014B

**DMA Write (Non-Block Mode) Initiator Send Cycle**

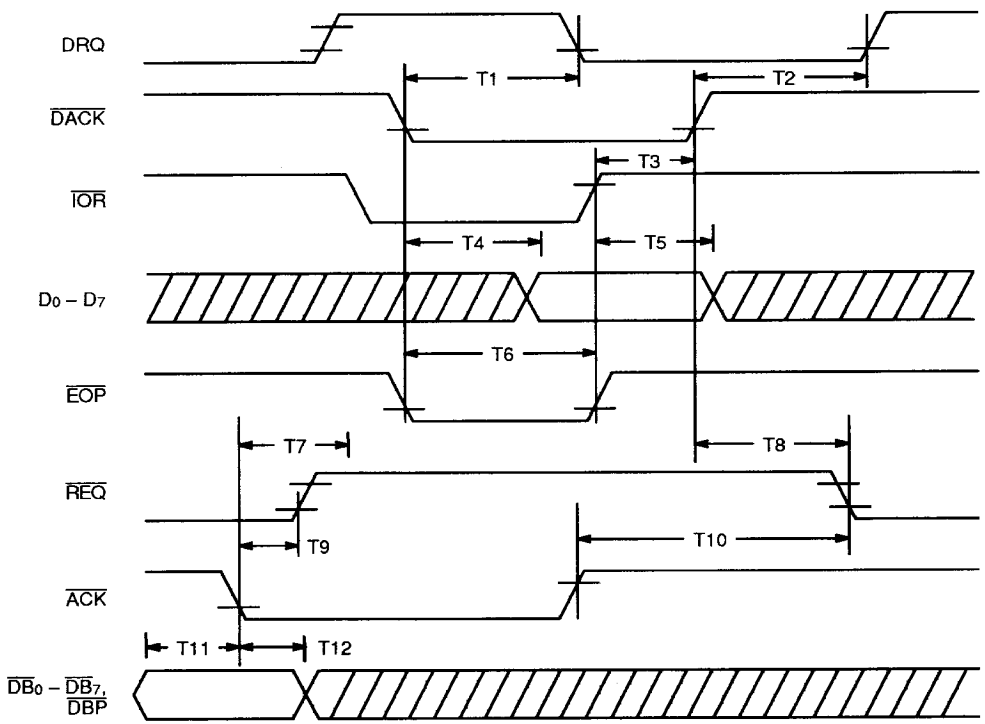
## SWITCHING CHARACTERISTICS/WAVEFORMS

| Name | Parameter Description | Min. | Max. | Unit |
|---|---|---|---|---|
| T1 | DRQ FALSE from $\overline{DACK}$ TRUE | | 35 | ns |
| T2 | $\overline{DACK}$ FALSE to DRQ TRUE | 30 | | ns |
| T3 | $\overline{DACK}$ Hold Time from End of $\overline{IOR}$ | 0 | | ns |
| T4 | Data Access Time from Read Enable* | | 20 | ns |
| T5 | Data Hold Time from End of $\overline{IOR}$ | 0 | | ns |
| T6 | Width of $\overline{EOP}$ Pulse (Note 1) | 40 | | ns |
| T7 | $\overline{ACK}$ TRUE to DRQ TRUE | | 90 | ns |
| T8 | $\overline{DACK}$ FALSE to $\overline{REQ}$ TRUE ($\overline{ACK}$ FALSE) | | 40 | ns |
| T9 | $\overline{ACK}$ TRUE to $\overline{REQ}$ FALSE | | 90 | ns |
| T10 | $\overline{ACK}$ FALSE to $\overline{REQ}$ TRUE ($\overline{DACK}$ FALSE) | | 35 | ns |
| T11 | Data Setup Time to $\overline{ACK}$ | 10 | | ns |
| T12 | Data Hold Time from $\overline{ACK}$ | 65 | | ns |

*Read Enable is the occurence of $\overline{IOR}$ and $\overline{DACK}$.

**Note:**

1. $\overline{EOP}$, $\overline{IOR}$, and $\overline{DACK}$ must be concurrently TRUE for at least T6 for proper recognition of the $\overline{EOP}$ pulse.




10665B-015B

**DMA Read (Non-Block Mode)Target Receive Cycle**

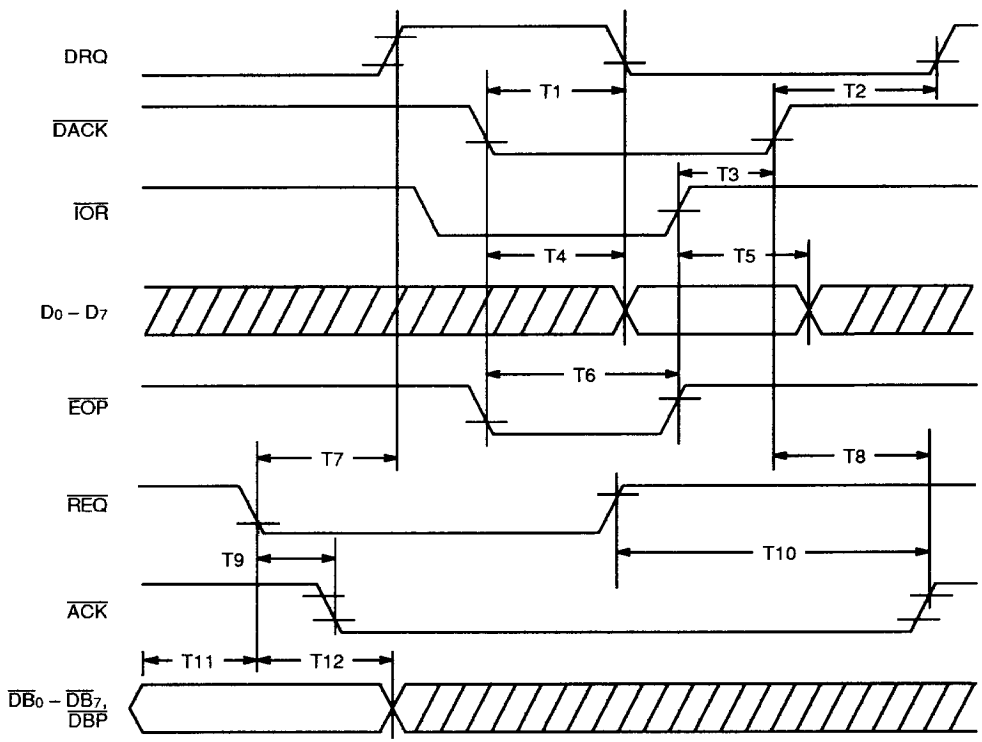
## SWITCHING CHARACTERISTICS/WAVEFORMS

| Name | Parameter Description | Min. | Max. | Unit |
|---|---|---|---|---|
| T1 | DRQ FALSE from $\overline{DACK}$ TRUE | | 35 | ns |
| T2 | $\overline{DACK}$ FALSE to DRQ TRUE | 30 | | ns |
| T3 | $\overline{DACK}$ Hold Time from End of $\overline{IOR}$ | 0 | | ns |
| T4 | Data Access Time from Read Enable* | | 20 | ns |
| T5 | Data Hold Time from End of $\overline{IOR}$ | 0 | | ns |
| T6 | Width of $\overline{EOP}$ Pulse (Note 1) | 40 | | ns |
| T7 | $\overline{REQ}$ TRUE to DRQ TRUE | | 95 | ns |
| T8 | $\overline{DACK}$ FALSE to $\overline{ACK}$ FALSE ($\overline{REQ}$ FALSE) | | 40 | ns |
| T9 | $\overline{REQ}$ TRUE to $\overline{ACK}$ TRUE | | 85 | ns |
| T10 | $\overline{REQ}$ FALSE to $\overline{ACK}$ FALSE ($\overline{DACK}$ FALSE) | | 35 | ns |
| T11 | Data Setup Time to $\overline{REQ}$ | 10 | | ns |
| T12 | Data Hold Time from $\overline{REQ}$ | 65 | | ns |

*Read Enable is the occurence of $\overline{IOR}$ and $\overline{DACK}$.

**Note:**

1. $\overline{EOP}$, $\overline{IOR}$, and $\overline{DACK}$ must be concurrently TRUE for at least T6 for proper recognition of the $\overline{EOP}$ pulse.




10665B-016B

**DMA Read (Non-Block Mode) Initiator Receive Cycle**

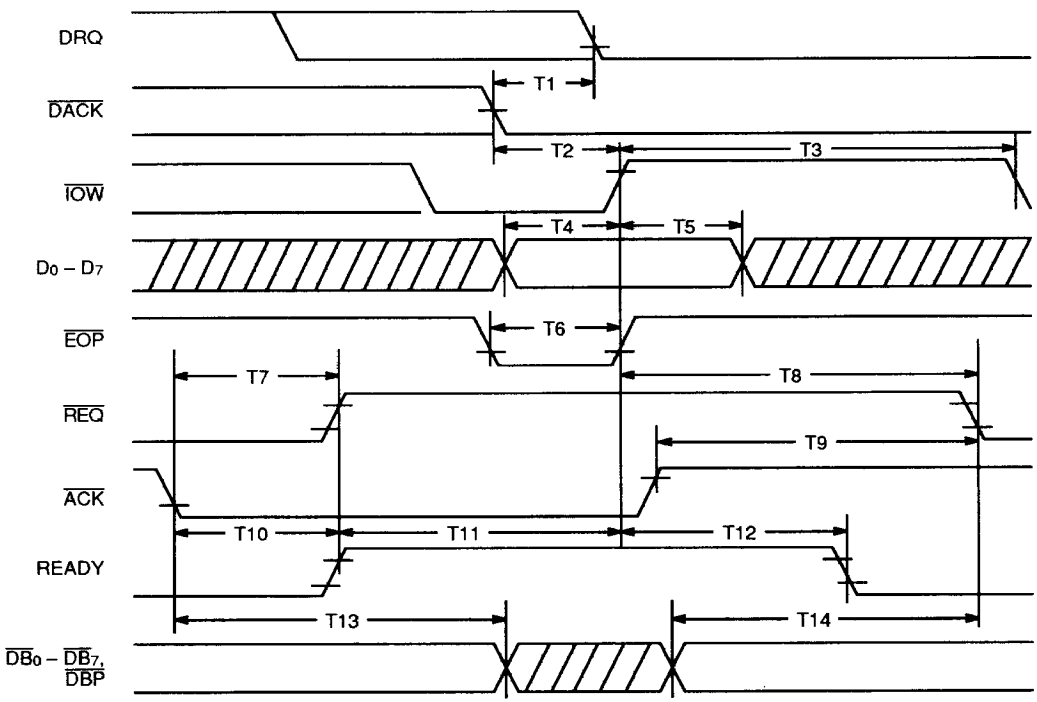
## SWITCHING CHARACTERISTICS/WAVEFORMS

| Name | Parameter Description | Min. | Max. | Unit |
|---|---|---|---|---|
| T1 | DRQ FALSE from $\overline{DACK}$ TRUE | | 35 | ns |
| T2 | Write Enable Width* | 40 | | ns |
| T3 | Write Recovery Time | 80 | | ns |
| T4 | Data Setup to End of Write Enable* | 5 | | ns |
| T5 | Data Hold Time from End of $\overline{IOW}$ | 5 | | ns |
| T6 | Width of $\overline{EOP}$ Pulse (Note 1) | 40 | | ns |
| T7 | $\overline{ACK}$ TRUE to $\overline{REQ}$ FALSE | | 85 | ns |
| T8 | $\overline{REQ}$ from End of $\overline{IOW}$ ($\overline{ACK}$ FALSE) | | 50 | ns |
| T9 | $\overline{REQ}$ from End of $\overline{ACK}$ ($\overline{IOW}$ FALSE) | | 40 | ns |
| T10 | $\overline{ACK}$ TRUE to READY TRUE | | 100 | ns |
| T11 | READY TRUE to $\overline{IOW}$ FALSE | 45 | | ns |
| T12 | $\overline{IOW}$ FALSE to READY FALSE | 20 | 60 | ns |
| T13 | Data Hold to $\overline{ACK}$ TRUE | 0 | | ns |
| T14 | Data Setup to $\overline{REQ}$ TRUE | 40 | | ns |

*Write Enable is the occurence of $\overline{IOW}$ and $\overline{DACK}$.

**Note:**

1. $\overline{EOP}$, $\overline{IOW}$, and $\overline{DACK}$ must be concurrently TRUE for at least T6 for proper recognition of the $\overline{EOP}$ pulse.




10665B-017B

**DMA Write (Block Mode) Target Send Cycle**

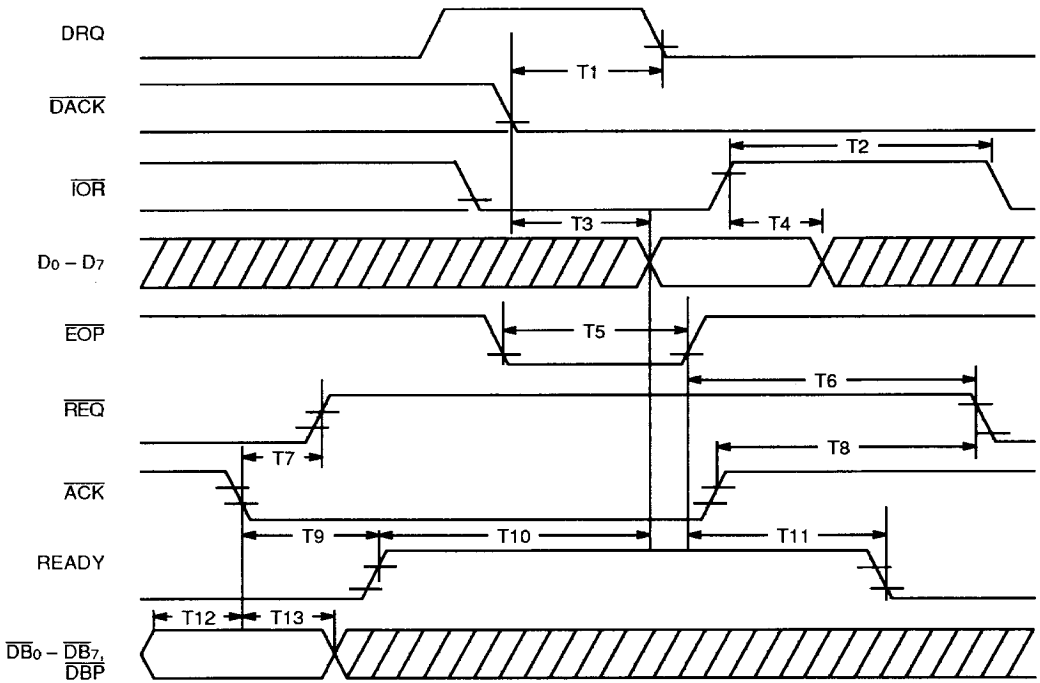
## SWITCHING CHARACTERISTICS/WAVEFORMS

| Name | Parameter Description | Min. | Max. | Unit |
|---|---|---|---|---|
| T1 | DRQ FALSE from $\overline{DACK}$ TRUE | | 35 | ns |
| T2 | $\overline{IOR}$ Recovery Time | 80 | | ns |
| T3 | Data Access Time from Read Enable* | | 20 | ns |
| T4 | Data Hold Time from End of $\overline{IOR}$ | 0 | | ns |
| T5 | Width of $\overline{EOP}$ Pulse (Note 1) | 40 | | ns |
| T6 | $\overline{IOR}$ FALSE to $\overline{REQ}$ TRUE ($\overline{ACK}$ FALSE) | | 45 | ns |
| T7 | $\overline{ACK}$ TRUE to $\overline{REQ}$ FALSE | | 90 | ns |
| T8 | $\overline{ACK}$ FALSE to $\overline{REQ}$ TRUE ($\overline{IOR}$ FALSE) | | 35 | ns |
| T9 | $\overline{ACK}$ TRUE to READY TRUE | | 100 | ns |
| T10 | READY TRUE to Valid Data | | 50 | ns |
| T11 | $\overline{IOR}$ FALSE to READY FALSE | | 40 | ns |
| T12 | Data Setup Time to $\overline{ACK}$ | 10 | | ns |
| T13 | Data Hold Time from $\overline{ACK}$ | 65 | | ns |

*Read Enable is the occurence of $\overline{IOR}$ and $\overline{DACK}$.

**Note:**

1. $\overline{EOP}$, $\overline{IOR}$, and $\overline{DACK}$ must be concurrently TRUE for at least T5 for proper recognition of the $\overline{EOP}$ pulse.

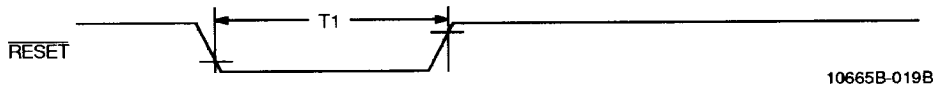



10665B-018B

**DMA Read (Block Mode) Target Receive Cycle**

## SWITCHING CHARACTERISTICS/WAVEFORMS

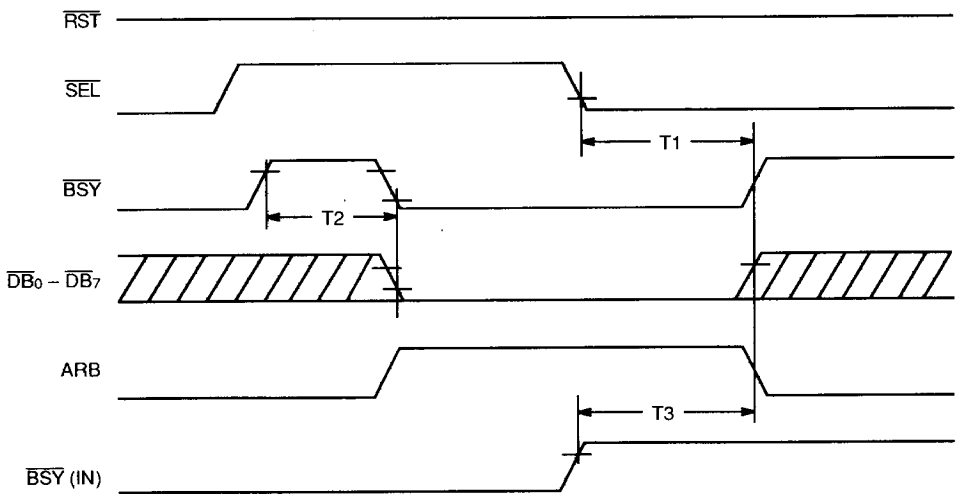
| Name | Parameter Description | Min. | Max. | Unit |
|---|---|---|---|---|
| T1 | Minimum Width of Reset | 100 | | ns |




10665B-019B

**Reset**

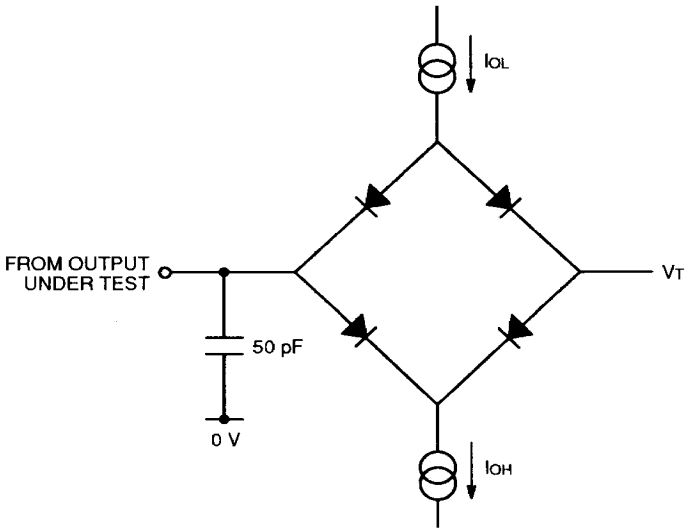
| Name | Parameter Description | Min. | Max. | Unit |
|---|---|---|---|---|
| T1 | Bus Clear from $\overline{SEL}$ TRUE | | 600 | ns |
| T2 | Arbitrate Start from $\overline{BSY}$ FALSE | 1200 | 2400 | ns |
| T3 | Bus Clear from $\overline{BSY}$ FALSE | | 1100 | ns |




10665B-020A

**Arbitration**

## SWITCHING TEST CIRCUIT




10665B-021A

## SWITCHING TEST WAVEFORM


10665B-022A