# NCR 53C94, 53C95, 53C96
## Advanced SCSI Controller

## General Description

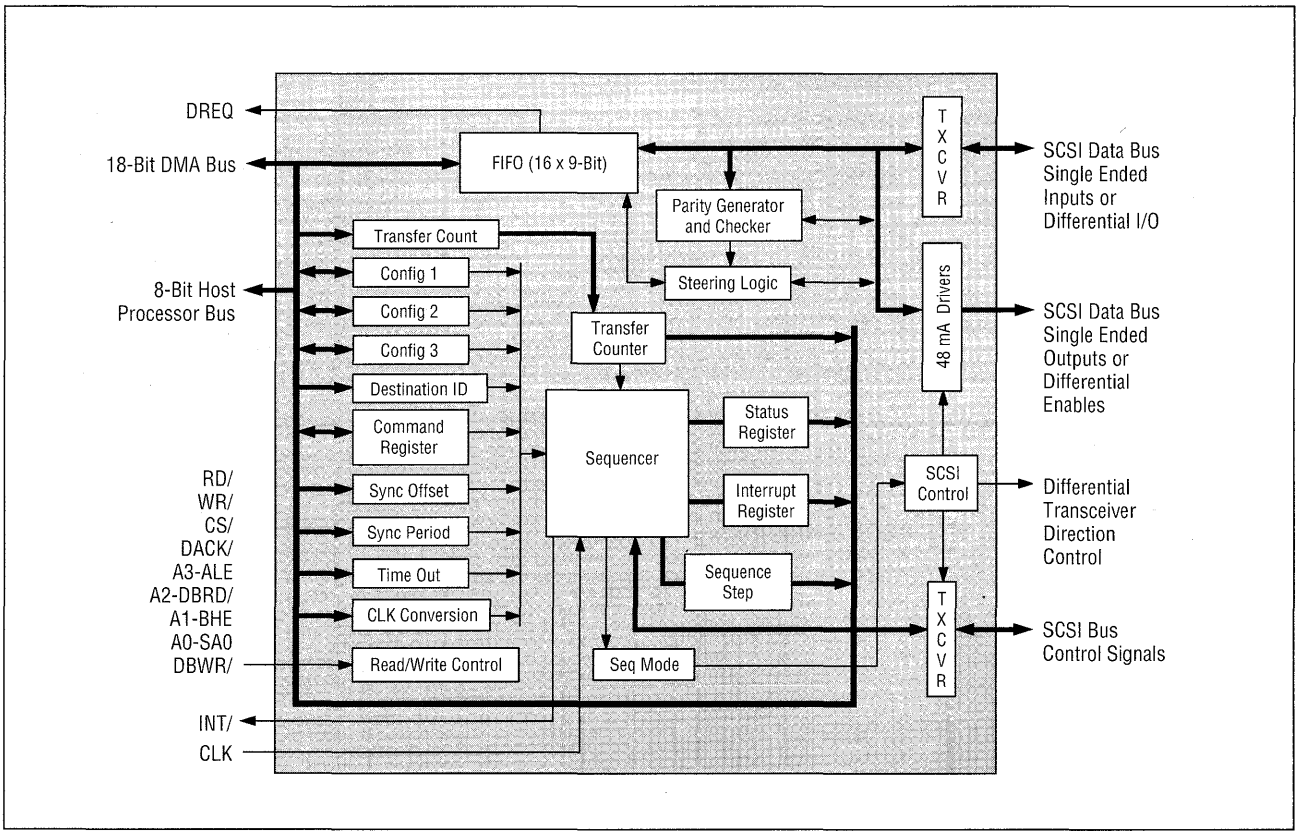
The 53C94, 53C95 and 53C96 are high performance CMOS devices conforming to the ANSI standard, X3.131-1986, for Small Computer Systems Interface (SCSI). They are a super-set of the 53C90 with additional commands, registers and an 18-bit DMA interface (two parity bits). The microprocessor bus width is 8-bits, and may be configured to be separate from the 18-bit DMA bus (called dual bus) or share the lower half (called single bus). The 53C94 has on-chip 48 mA drivers for single ended transmission, while the 53C95 provides control signals for external differential tranceivers. The 53C96 offers both single ended and differential mode in a single 100 pin Quad Flat Pack (QFP) package.

The 53C90 Family reduces protocol overhead by performing common SCSI algorithms, or sequences, in response to a single command. The 53C94 and 53C95 will operate at sustained data tansfer rates up to 5 MB/S in synchronous mode and 5 MB/S in asynchronous mode. Refer to *Data Transfer Rate.*

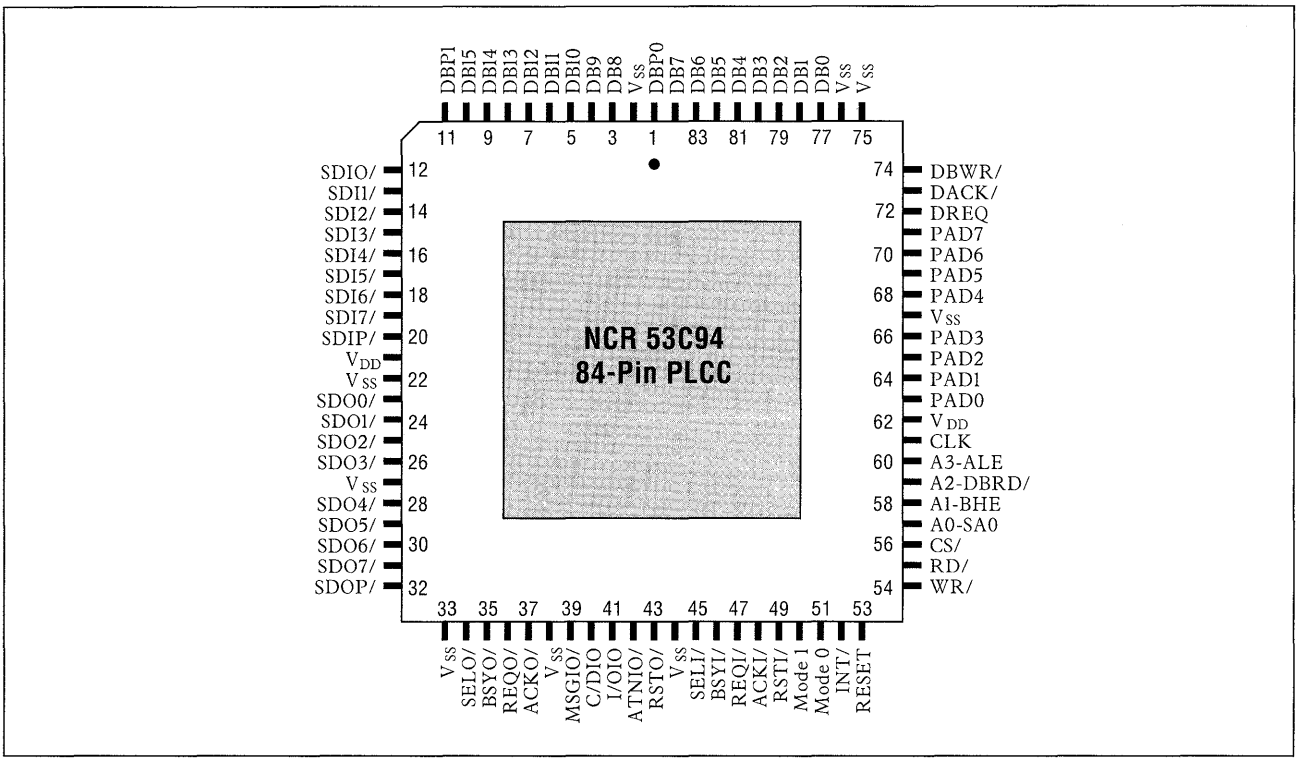
## Features

- ANSI X3.131-1986 compatible
- On-chip 48 mA drivers
- Control logic for differential tranceivers
- Parity generation, optional checking
- Parity pass-thru
- Software compatible with 53C90
- SCSI-2 tagged-queuing
- 18-bit DMA interface
- Dual bus or single bus processor and DMA
- Multiplexed or nonmultiplexed address/data
- Burst mode
- 20 MB/S DMA interface
- Up to 5 MB/S asynchronous SCSI
- Up to 5 MB/S synchronous SCSI
- Low power CMOS
- 84-pin PLCC and 100-pin QFP

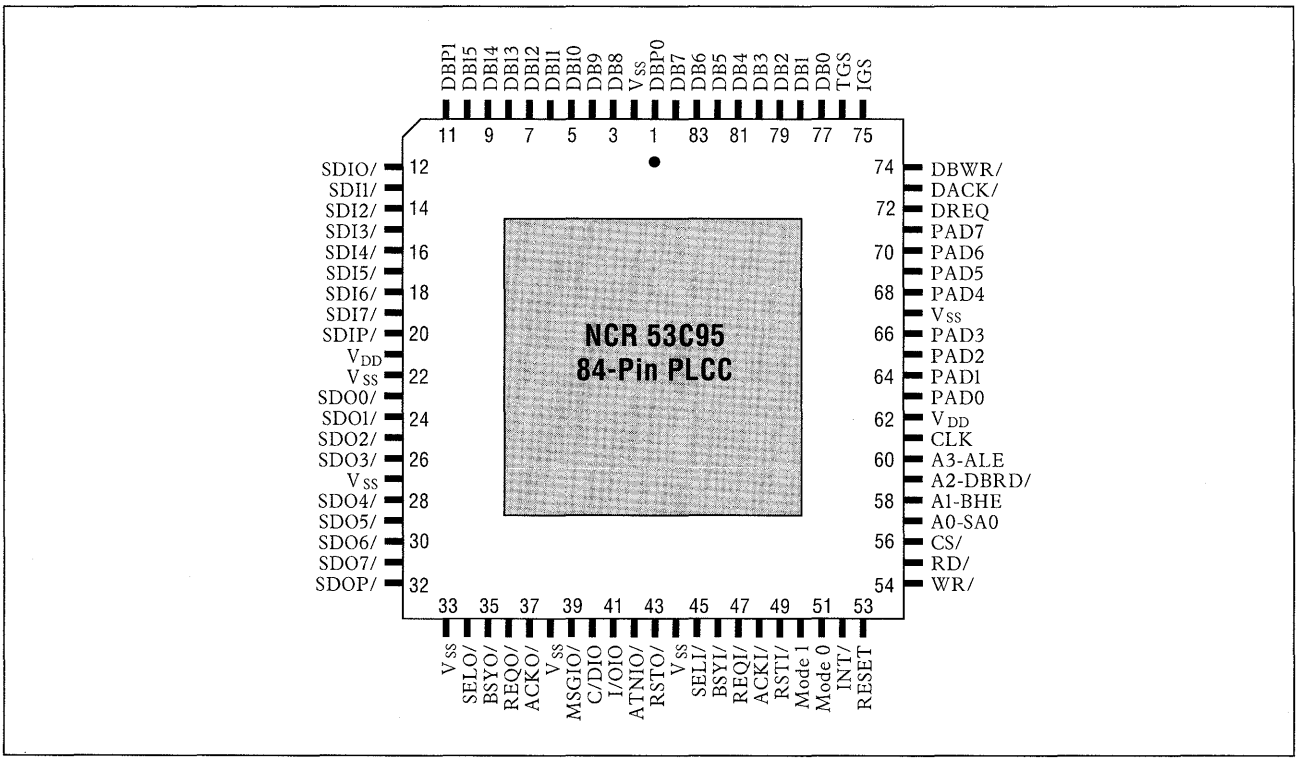
## Figure 1. Functional Block Diagram



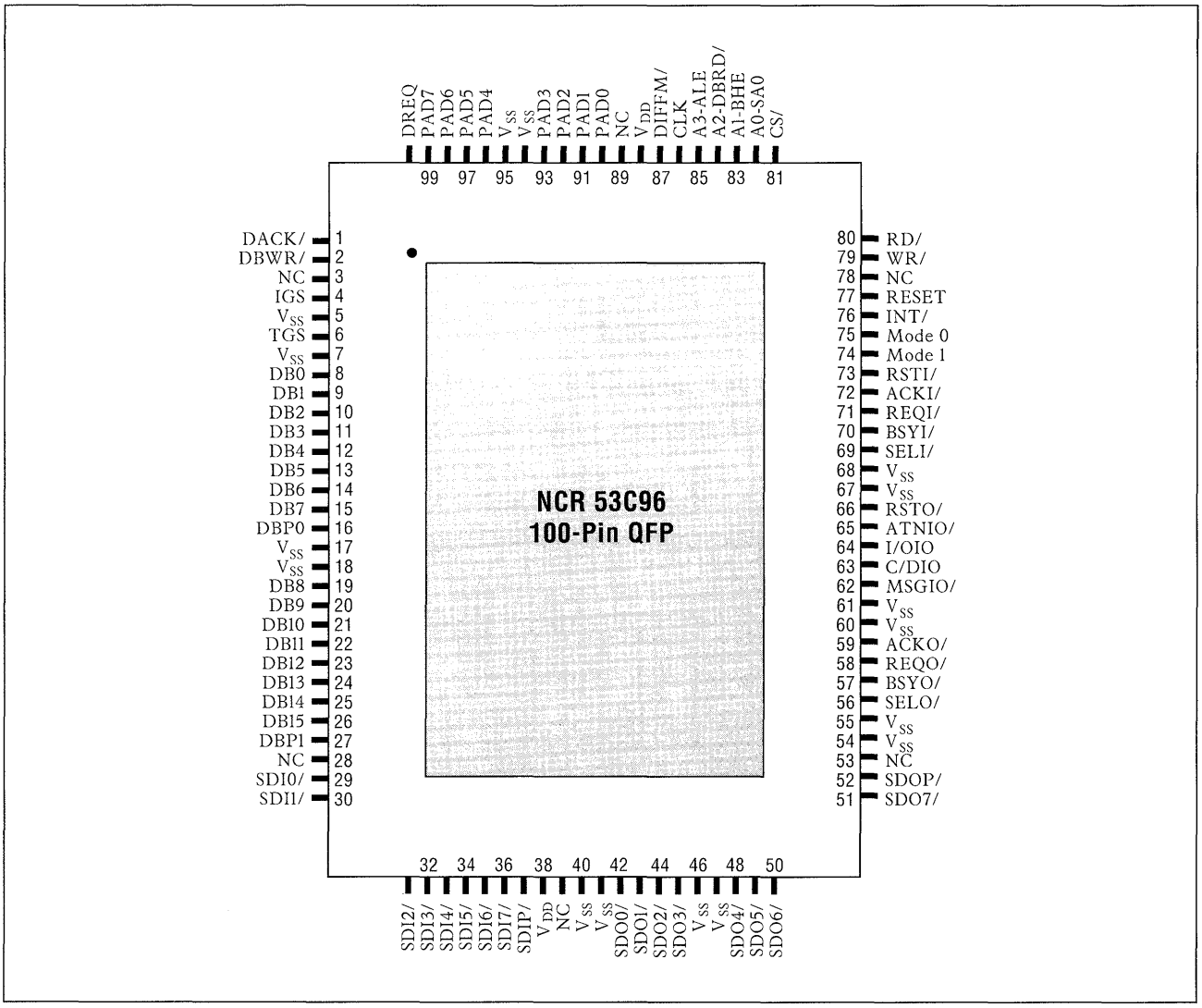
## Figure 2. NCR 53C94 Pin Configuration




## Figure 3. NCR 53C95 Pin Configuration



## Figure 4. NCR 53C96 Pin Configuration




## Table 1. Power and Ground Pins

| PLCC Pin Number | QFP Pin Number | Signal | Description |
|---|---|---|---|
| 21, 62 | 38, 88 | $V_{DD}$ | +5 V power input |
| 2, 22, 27, 33, 38, 44, 67 | 5, 7, 17, 18, 40, 41, 46, 47, 54, 55, 60, 61, 94, 95 | $V_{SS}$ | Ground. NCR recommends a ground plane be used. |

## Table 2. Microprocessor and DMA Interface Pins

| Signal | Type | Description |
|---|---|---|
| PAD7-0 | B | Bi-directional active-high processor address-data bus with internal pull-ups. When the mode pins are strapped for dual-bus operation, these pins allow the processor to access the chip's internal registers at the same time the DMA bus is active. In multiplexed mode, address and data share this bus. In nonmultiplexed mode, these pins are for data only. In single bus mode these pins are not used. Refer to *Appendix A* for connection diagrams. |
| DB15-0 | B | Bi-directional active-high data bus with internal pull-ups. When the mode pins are strapped for dual-bus operation, these pins are the 16-bit DMA data bus. In single bus mode, the processor accesses internal registers on the lower 8 bits, while the DMA accesses the FIFO using all 16 bits. In byte control mode, BHE and SA0 allow DMA data to be transferred on the lower half, or the upper half, or the entire 16-bit DB bus. |
| DBP1<br>DBP0 | B<br>B | Odd parity for DB15-8<br>Odd parity for DB7-0 |
| A3-ALE,<br>A2-DBRD/ | I | In nonmultiplexed mode, these inputs are address bits 3 and 2. In multiplexed mode, they become ALE and DBRD/. The address on the PAD bus will be internally latched when ALE switches from high to low. DBRD/ is the read signal for the DB bus. |
| A1-BHE,<br>A0-SA0 | I | In nonmultiplexed mode, these pins are address inputs 1 and 0. In multiplexed mode with byte control, these pins are defined as BHE and SA0. Byte control mode is available in mode 2 only (Refer to *Configuration Pins* and *Host Bus Configuration*) when bit 5 in the Config 2 register is set. These pins are not used in multiplexed nonbyte control mode.<br>**BHE** **SA0** **Bytes Transferred On**<br>0 0 DB7-0 and DBP0<br>0 1 None<br>1 0 DB15-0 and DBP1 and DBP0<br>1 1 DB15-8 and DBP1 |
| DBWR/ | I | Active-low DMA write signal which strobes DB15-0 data into the FIFO when DACK/ is true. In single bus mode, DBWR/ must be tied to WR/. |
| CS/ | I | Active-low chip select. This input enables 8-bit access to internal registers during read or write. CS/ uses the address inputs to access any register, including the FIFO while DACK/ accesses only the FIFO. CS/ and DACK/ must never be active simultaneously in single bus mode, but may both be true in dual bus mode provided that CS/ is not accessing the FIFO. |
| RD/ | I | Active-low register read signal. This input allows internal registers to drive the data bus when either CS/ or DACK/ is also true. Refer to *Appendix A.* |
| WR/ | I | Active-low register write signal. This input causes the ASC to write data into its internal registers when CS/ is also true. |
| INT/ | O | Active-low, open-drain interrupt signal to the microprocessor. It is latched on the rising edge of CLK. It may be cleared by reading the interrupt register or by a host hardware reset or a software reset chip (but not by a SCSI reset). This output cannot be masked by the user. |

## Table 2. Microprocessor and DMA Interface Pins (Cont'd)

| Signal | Type | Description |
|---|---|---|
| DREQ | O | Tri-state active-high DMA request signal to the DMA controller. DREQ will remain true as long as either the FIFO contains at least one word (or one byte if 8-bit mode) to send to memory during DMA read, or has room for one more word (or byte if 8-bit mode) in the FIFO during DMA write. |
| DACK/ | I | Active-low DMA acknowledge from the DMA controller. DACK/ accesses the FIFO only, while CS/ accesses any register including the FIFO. CS/ and DACK/ must never be true simultaneously in single bus mode. |
| RESET | I | Active-high chip reset. Reset must be asserted for two CLK periods, minimum, after the voltage on the power pins has reached $V_{DD}$ min. |
| CLK | I | Square wave clock input which generates internal chip timing. The maximum frequency is 25 MHz. The minimum frequency for asynchronous SCSI is 10 MHz. The minimum frequency for synchronous SCSI is 12 MHz. The synchronous transmission period is equal to the CLK period multiplied by the value in the synchronous transfer period register. The asynchronous transmission rate is indirectly affected by the CLK period. Refer to *Data Transfer Rate.* |

## Table 3. SCSI Pins

| Signal | Type | Description |
|---|---|---|
| SDO7/-0/, SDOP/ | O | 48 mA, open drain SCSI data/parity output bus. In single-ended mode (DIFFM grounded) these pins are active-low SCSI data signals. In differential mode (DIFFM pulled high) these outputs are used to control the direction of external differential tranceivers, with high meaning output to the SCSI bus, low meaning input from the SCSI bus. |
| SDI7/-0/, SDIP | B | Schmitt trigger, active-low SCSI data parity input bus. In single-ended mode (DIFFM = 0) these inputs are SCSI data bus inputs. In differential mode (DIFFM = 1) these pins are bi-directional data and parity signals for external tranceivers. |
| BSYO/, SELO/, RSTO/ | O | 48 mA open-drain SCSI outputs. In single-ended mode, these signals are active-low. In differential mode, they are active-high. The reset SCSI bus command will cause the ASC to drive RSTO/ true for 25-40 µs, depending on CLK frequency and clock conversion factor. Refer to *Miscellaneous Commands.* |
| REQO/ | O | 48 mA, open-drain, SCSI output. Asserted only in target mode. |
| ACKO/ | O | 48 mA, open-drain, SCSI output. Driven by the ASC in initiator mode only. |
| ATNIO/ | B | 48 mA open-drain output, Schmitt trigger input. In initiator mode, it is an output, and ATN will be automatically asserted when the ASC detects an incoming parity error, or may be asserted by certain ASC commands. In target mode, this signal is an input. Hysterisis is nominally 400 mV centered at 1.4 V. |
| MSGIO/, C/DIO, I/OIO, | B | Bi-directional SCSI phase signals. They are 48 mA outputs in target mode, and Schmitt trigger inputs in initiator mode. The hysterisis is nominally 400 mV and centered at 1.4 V. |
| BSYI, SELI/, RSTI/, ACKI/, REQI/ | I | Schmitt trigger, active-low SCSI input signals. The hysterisis is nominally 400 mV and centered at 1.4 V. |

## Table 4. Transceiver Control Pins

| Signal | Type | Description |
|---|---|---|
| IGS | O | Active-high initiator group select. This output is high whenever the ASC is in initiator mode. It is used in differential mode to enable the initiator signals (ACKO/, ATNO/). When high, the ASC drives these signals. |
| TGS | O | Active-high target group select. This output is high whenever the ASC is in target mode. It is used in differential mode to enable the target signals (REQO/, MSGI0/, C/DIO, I/OIO). |

## Table 5. Configuration Pins

| Mode 1 | Mode 0 | Register Address | Register Data | DMA Width | Configuration |
|---|---|---|---|---|---|
| 0 | 0 | A3-A0 | DB | 8 | Single bus, 8-bit processor, 8-bit DMA |
| 0 | 1 | A3-A0 | DB | 16 | Single bus, 8-bit processor, 16-bit DMA |
| 1 | 0 | PAD 3-0 | PAD | 16 | Dual bus, multiplexed, byte control |
| 1 | 1 | A3-A0 | PAD | 16 | Dual bus, 8-bit processor, 16-bit DMA |
| | | These input pins configure the PAD bus, DB bus and the address/byte control bus (A3-ALE, A2-DBRD, A1 -BHE, A0-SAO). Refer to *Appendix A* for configuration diagrams. | | | |
| DIFFM/ | | Differential mode enable. When this pin is high, the ASC operates in single-ended mode, with separate SCSI data input and output buses. When this pin is grounded, the ASC operates in differential mode, with bi-directional SCSI data on the SDI pins and active-high transceiver enables on the SDO pins. This pin is available only on the 53C96. | | | |

## Functional Description

The ASC has a command set that allows it to perform common SCSI sequences at hardware speed without host intervention. Its on-chip FIFO may be accessed simultaneously by the SCSI bus and either the microprocessor or the host DMA controller. All command, data, status and message bytes pass through the FIFO on their way to or from the SCSI bus. Most ASC commands have two versions: DMA and non-DMA. When DMA instructions are used, data will pass between memory and the SCSI bus with the FIFO acting as temporary storage when the DMA channel is temporarily shut down by a higher priority event, such as DRAM refresh.

The FIFO also helps speed execution during non-DMA transfers. For example, in initiator role, the microprocessor will load the CDB (Command Descriptor Block) and optionally, one or three message bytes into the FIFO, issue one of several selection commands then wait for an interrupt. The ASC will wait for bus free, arbitrate for the bus again and again until it acquires it, send the message bytes, followed by the CDB, then generate an interrupt. Meanwhile, a multi-tasking host may continue with other tasks.

In target role, the microprocessor will enable selection, then wait for an interrupt. Eventually, an initiator will select the ASC. It will then automatically step through the selection and command phases before generating an interrupt. When the interrupt occurs, the entire CDB will be in the FIFO along with any message bytes sent by the initiator. Combination commands, such as these, are identified with the *sequence* suffix in the table of ASC commands.

After selection phase has been successfully completed, the ASC may transfer bytes in any of the SCSI information phases whether operating in initiator or target role. The ASC supports disconnect/reselect in both initiator and target roles, making high performance multi-threaded systems easy to implement.

The ASC may transfer data phase bytes across the bus synchronously, at speeds up to 5 MB/S, or asynchronously at speeds up to 6 MB/S. Refer to *Data Transfer Rate*. The difference between the two is transparent to the user except that the synchronous offset and the synchronous transfer period registers must be programmed prior to synchronous data transfer. The default, after hardware or software reset is asynchronous transmission.

Data phase bytes will usually be transferred using DMA. The microprocessor will program an external DMA controller, program the ASC transfer count, issue an ASC data transfer command (there are several), then wait for an interrupt. The DMA controller and the ASC will transfer all the data without microprocessor intervention.

To end the SCSI transaction, the ASC target will place a status byte and a message byte in the FIFO, then issue a single command (there are two to chose from) which will cause the ASC to first assert status phase, send the first byte, assert message in phase, send the second byte, disconnect from the SCSI bus (after the initiator releases ACK [Acknowledge]) and interrupt the microprocessor.

The end of a SCSI transaction is similar for an ASC initiator except that it receives two bytes into its FIFO. The initiator prevents the target from disconnecting by holding ACK asserted on the bus while the microprocessor examines the status and message bytes. If both bytes are good, the message accepted command is used to instruct the ASC to release ACK, which allows the target to disconnect which causes the initiator to interrupt its host and report the disconnect. If the status and message bytes are not good, the host should first issue the set ATN (Attention) command before issuing the message accepted command. This instructs the ASC to assert ATN before releasing ACK, which should cause the target to request message out phase rather than disconnect.

### Bus Initiated Sequences

- Selection
- Reselection
- SCSI bus reset

Selection or reselection sequences occur in the disconnected state when the ASC is selected or reselected by another initiator or target, if the enable selection or reselection command had previously been received by the ASC.

In addition to responding to bus initiated events, the ASC may initiate a bus event by using one of several selection or reselection commands. If one of these commands

starts executing, *it will clear enable selection or reselection* after arbitration has been won. Normally the microprocessor will have 250 ms (ANSI recommended selection time-out period) after the chip disconnects from the bus to re-enable bus initiated events. If the time-out is exceeded, an initiator or target which is attempting to connect to the ASC, may time-out and abort.

If, on the other hand, the bus initiated event occurs before the command starts executing, the FIFO and command register will be cleared, and any further writes by the microprocessor will be ignored until the interrupt register is read. Since a selection or reselection command requires that something be placed in the FIFO, these bytes will be lost, as will any command written to the command register. The interrupt handler that services a selection or reselection command will have to examine the bits in the interrupt register to determine if the ASC selected another device, or if it was selected by another device. The former case will cause a function complete interrupt, the latter case will cause a selection or reselection interrupt.

## Bus Initiated Selection

When the ASC has been selected as a target, the following data will be in its FIFO:

- Bus ID
- Identify message
- Command Descriptor Block (CDB)

The bus ID will always be present and will always be one byte. It is an un-encoded version of the state of the bus during selection phase. Any SCSI data bits that were true during selection phase will be set. The target ID (our ID) must always be set. In arbitrating systems, the initiator ID must also be set. The initiator ID is optional in non-arbitrating systems.

The identify message will always be placed in the FIFO and will always be one byte in SCSI-1 systems but may be one or three bytes in SCSI-2 systems. If the ASC is selected with ATN false, it will store a null byte (00) in the FIFO behind the bus ID, then begin requesting command phase bytes. A detected parity error will cause the ASC to interrupt and stop.

If the ASC is selected with ATN true, and the SCSI-2 bit is not set, it will request one message byte, place it in the FIFO behind the bus ID. Then it requests command phase bytes unless the message byte is not a valid identify message (bit 7 not-zero), or a parity error is detected, which will cause the ASC to interrupt and stop. The sequence step register should then be examined.

If the ASC is selected with ATN true and the SCSI-2 bit set, the ASC will examine both the message byte and the ATN signal to determine how many bytes to request. If the first byte is a valid identify message and if ATN goes false after receiving the first byte the ASC will only request one message byte. If the first byte is a valid identify message byte and ATN is still true, it will request two more message bytes. After requesting the message bytes, the ASC requests command phase bytes unless one of the following situations occurs:

1) The first byte was not a valid identify message
2) A parity error was detected
3) ATN went false between the second and third bytes
4) ATN remained true but the SCSI-2 bit was false which causes the ACS to interrupt and stop

If one of the above situations occurs, examine the sequence step register.

The CDB will always begin at the third or fifth byte in the FIFO, assuming selection completed normally. The CDB may be 6, 10 or 12 bytes long. Thus, in SCSI-2, the entire FIFO may be filled if a tagged-queue twelve byte command is used.

## Bus Initiated Reselection

The ASC will allow itself to be reselected as an initiator by a target if it has previously received the enable selection/reselection command. If the sequence completes normally, the following information will be in the FIFO:

- Bus ID
- Identify message

The bus ID is the same as the selection case, described above. The identify message will always be present and always be one byte.

## Bus Initiated Reset

A bus initiated reset will be recognized by the ASC at any time. When SCSI RST pulses true, the ASC will disconnect from the bus and reset its internal sequencer. If bit 1 in Config 1 register is not set, the ASC will generate a SCSI reset detected interrupt.

## Parity Checking and Generation

The ASC has four bits which control parity generation and checking. If parity checking is disabled, the ASC does not check for parity errors. In this document, the word *detected* in conjunction with *parity error* should be understood to imply that parity checking has previously been enabled. In target role, detected parity errors will set the parity error status bit and clear the command register. In initiator role, detected parity errors will set the parity error bit and assert ATN (Attention) prior to releasing ACK (Acknowledge). Parity errors occurring on the first few bytes after a phase change to synchronous data in are handled slightly differently in initiator mode. Refer to *Initiator Commands*.

If parity test mode is enabled, the DPB0 is a duplicate of DB7 and DBP1 is a duplicate of DB15. This is true both for data flowing from the FIFO to the SDB (SCSI Data Bus) or data flowing from the FIFO to the DB (Host Data Bus).

The 53C94, 95, and 96 have two parity pins (DBP0, DBP1) that may always be used by the DMA, and may be used by the host processor if configured for mode zero or mode 1. In mode two and mode three, the processor connects to the FIFO on an 8-bit bus only. In these modes, the internal parity generator creates parity to send to the SCSI bus.

When the DBP pins are enabled, parity may pass between SCSI and host buses without change or may be generated by the ASC from the data byte. Whether generated internally or externally, the parity bit is always loaded into the FIFO along with the data byte. From there on, it moves through the FIFO along with the byte. The FIFO may be accessed by three buses: SCSI bus, microprocessor bus or host DMA bus.

When checking parity, the ASC checks "at the edge of the board." Parity errors are flagged as data comes into the FIFO from the SCSI bus, or as it leaves the FIFO on its way out to the SCSI bus. Note that, a detected parity error will set the parity error status bit, and cause ATN to be asserted if detected during an initiator information in phase, but will not cause an interrupt.

## Table 6. Parity Control

| Control Bit | Data Direction | Bit Set | Bit Not Set |
|---|---|---|---|
| Parity checking<br>Config 1, bit 4 | SCSI to FIFO | Enable parity checking and error reporting.<br>SDPB loaded into FIFO. | Disable parity checking and error reporting.<br>Parity generator to FIFO. |
| Test parity<br>Config 1, bit 5 | FIFO to SCSI | SDBP is replica of SDB7 | FIFO to SDBP |
| | FIFO to memory | DBP0 is replica of DB7<br>DBP1 is replica of DB15 | FIFO to SDBP |
| DMA parity<br>Config 2, bit 0 | DACK/ to FIFO | DBP to FIFO | Parity generator to FIFO |
| | FIFO to SCSI | Enable parity checking and error reporting | Disable parity checking and error reporting |
| Register parity<br>Config 2, bit 1 | CS/ to FIFO | DBP to FIFO | Parity generator to FIFO |
| | FIFO to SCSI | Enable parity checking and error reporting | Disable parity checking and error reporting |

## Host Bus Configuration

The DMA and microprocessor buses may be configured in one of four ways:

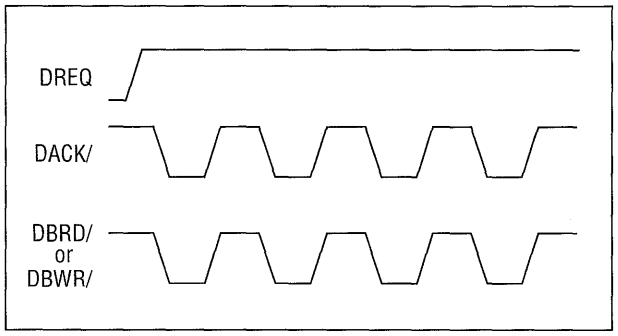
| Mode | Description |
|---|---|
| Zero | Single bus; 8-bit DMA, 8-bit processor bus |
| One | Single bus; 16-bit DMA, 8-bit processor bus |
| Two | Dual bus; 16-bit DMA bus with byte control and 8-bit multiplexed processor address/data bus |
| Three | Dual bus; 16-bit DMA bus and 8-bit processor bus |

The operating mode is selected by the mode 1 and mode 0 strapping pins. The four operating modes are labeled mode zero through mode three. These names are derived from the binary encoded state of the mode configuration pins with the mode 1 pin being the most significant bit. Refer to *Appendix A* for configuration diagrams. In both single bus modes, the DMA and the microprocessor share the same data bus. Therefore, CS/ and DACK/ must never be true at the same time when operating in single bus mode. Conversely, both dual bus modes have separate data buses for DMA and microprocessor, which may be active simultaneously provided CS/ is not accessing the FIFO.

## Threshold

The threshold is the number of bytes in the FIFO that triggers DREQ (DMA Request). For DMA read, DREQ will be asserted when the FIFO contains at least the threshold number of bytes. For DMA write, the FIFO must be able to accept this number of bytes. For 8-bit DMA operation, the normal threshold is one byte. For 16-bit operation, the normal threshold is two bytes (one word). In normal operation, DREQ will remain true as long as the threshold is exceeded. In normal mode, DREQ goes false when the current DMA acknowledge (DMA read or DMA write) causes the number of bytes in the FIFO to drop below the threshold.

**Figure 5. Normal DMA Mode**



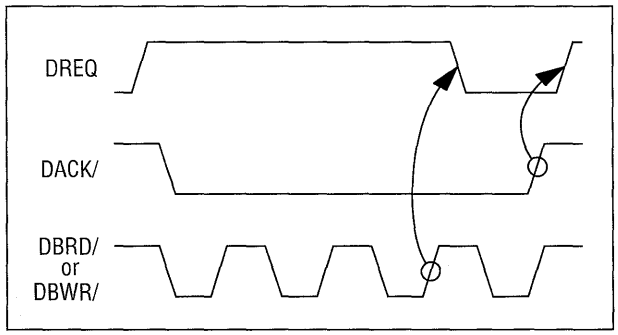

*Normal DMA Mode* illustrates the case where the threshold is always exceeded. This is typical of a DMA interface that is slower than the SCSI device to which the system is connected.

The threshold 8-bit in Config 3 changes the threshold to eight bytes for both 8-bit and 16-bit DMA operation. Refer to *DMA Burst Mode*, and *Configuration 3 Register*.

## DMA Burst Mode

Burst mode is a method of improving host bus efficiency. It is enabled by setting both the threshold 8 and the alternate DMA mode bits in Config 3. Threshold 8 causes the ASC to delay assertion of DMA request until it can transfer eight bytes (four words). The alternate DMA mode causes the ASC to deassert DREQ after the third word transfer (or the seventh byte transfer if configured for mode zero) causing an 8237 DMA controller to relinquish the bus after exactly four transfers.

**Figure 6. DMA Burst Mode**




This regular surrendering of the DMA channel has two benefits for two common DMA interface problems. For DMA controllers that do not recognize higher priority requests until the current device finishes, the ASC can

periodically force DMA arbitration, allowing DRAM refresh, for example, to occur during an SCSI operation. For DMA controllers that are much faster than the SCSI peripheral to which the system is connected, bus efficiency is improved by ensuring that the ASC has data to transfer while the DMA controller is controlling the bus.

## SCSI Input and Output Pins

The ASC SCSI data bus has a set of inputs and a set of outputs. This allows the ASC to be used in either single-ended mode or differential mode. In single-ended mode, the inputs are usually connected to the outputs on the circuit board. In differential mode, the SDI (SCSI Data Input) pins become bi-directional data pins, while the SDO (SCSI Data Output) pins become enable signals for external differential transceivers. Separate enables are required because, during arbitration, one data bus signal becomes an output while the other seven must be inputs, and during selection, two data bus signals become outputs while the other six must be inputs. Two signals, TGS and IGS, control the direction of the external transceivers, allowing the ASC to dynamically switch between initiator and target roles.

## Data Transfer Rate

Performance claims for the ASC are based on it being directly connected to the SCSI bus with no external transceivers. In differential mode, external transceivers are required and will slow asynchronous transmission by the propagation delay of the chosen transceiver but will not slow synchronous transmission.

The synchronous data transmission period is equal to the CLK input frequency multiplied by the encoded value in the synchronous period register. Sustained synchronous transfer rates of 5 MB/S are attainable across the commercial voltage and temperature range.

The asynchronous transmission rate will vary with cable length and the CLK period. The ASC can reach sustained transfer rates of 6 MB/S on short (one foot) cables using typical devices operating at or near nominal voltage and temperature. The typical transfer rate on a six meter cable is 4 MB/S using two typical ASCs talking to each other. The worst case asynchronous transmission rate, over voltage, temperature, and process variations is 3 MB/S on a maximum length (single-ended) cable and 4 MB/S on a one foot cable.

The asynchronous transmission rate is only slightly affected by the CLK frequency when sending data. The ASC will drive the data bus for a minimum of one CLK period (plus any additional time required to meet the ANSI required 55 ns setup time) before asserting REQ or ACK. The CLK frequency does not affect the asynchronous transfer rate when receiving data.

## Misaligned Boundary

The ASC provides all the necessary hooks to start a block of 16-bit DMA data at an even or odd address. If the hardware supports byte control, and if the byte control enable bit in Config 2 is set, then the hardware will handle the "misaligned boundary" condition automatically without the programmer having to worry about it. The ASC supports byte control only in mode two. If the hardware does not support byte control, then the firmware must occasionally make certain adjustments.

First, an explanation of the misaligned boundary condition. The memory in 16-bit systems is structured such that words occupy two bytes and begin on an even address. An even address has its least significant bit equal to zero.

**Words Aligned on Word Boundaries**

| |
|---|
| Word Address 4 |
| Word Address 2 |
| Word Address 0 |

Even addreses are also known as word addresses, or word boundaries. When a word starts on a word boundary, the low byte of the word will be in an even byte address, while the high byte of the word will be in an odd byte address.

**Two Bytes per Word**

| | |
|---|---|
| High byte of word | Low byte of word |
| Byte address 3 | Byte address 2 |
| Byte address 1 | Byte address 0 |

When the word is written to an odd address, the low byte of the word resides in the upper half of a system word address. Its high byte resides in the lower half of the next system word address.

### Misaligned Boundary

| | |
|---|---|
| | |
| | High byte of word |
| Low byte of word | |

To handle misaligned boundaries in software, the microprocessor will move the first byte and the DMA controller will move the rest. What the microprocessor does for the first byte depends on which direction data is flowing and whether the SCSI bus bit is operating synchronously or asynchronously.

For both synchronous and asynchronous data transfer when data is flowing out to the SCSI bus from the FIFO, the microprocessor may *preload* the FIFO. To preload the FIFO, the microprocessor simply places the first byte in the FIFO before issuing the DMA command (any of the initiator or target transfer commands).

For asynchronous data coming into the FIFO from the SCSI bus, the microprocessor must read the first byte out of the FIFO and write it to the odd memory address. The remaining bytes can be paired as words beginning on an even address, so the normal DMA transfer command will cause the rest of the block to transfer correctly. (Refer to *Config 3, Bit 2* to handle a single byte left over at the end of the transfer.)

For target synchronous data out phase (data flowing into FIFO from SCSI bus) the microprocessor must preload the FIFO with the lower half of the destination word. The microprocessor will read this byte from its own memory, write it to the FIFO, then issue the DMA receive data command. When the first 16-bit word is moved (via DMA) from the FIFO to memory, the low byte will be overwritten with a copy of itself. The high byte of this first word will be the first byte received over the SCSI bus. Subsequent bytes will be aligned as words and transferred 16-bits at a time.

For initiator synchronous data in (data flowing into FIFO from SCSI bus) the reserve FIFO byte option must be used. Config 2, bit 7 enables this feature. The FIFO must be preloaded by writing to a special register, register 0F (hex), the FIFO bottom register.

The reserve FIFO byte enable bit must be set before the phase changes to synchronous data in. After the interrupt, the microprocessor must read the byte residing in the low byte of own memory, place it in the special FIFO bottom register, then issue the DMA transfer info command. When the first 16-bit word is moved (via DMA) from the FIFO to memory, the low byte will be overwritten with a copy of itself. The high byte of this first word will be the first byte received over the SCSI bus. Subsequent bytes will be aligned as words and transferred 16 bits at a time.

## Table 7. Register Set

| Address (hex) | Read | Write |
|---|---|---|
| 0 | Transfer counter LSB | Transfer count LSB |
| 1 | Transfer counter MSB | Transfer count MSB |
| 2 | FIFO | FIFO |
| 3 | Command | Command |
| 4 | Status | Destination bus ID |
| 5 | Interrupt | Select/reselect timeout |
| 6 | Sequence step | Synchronous period |
| 7 | FIFO flags/sequence step | Synchronous offset |
| 8 | Configuration 1 | Configuration 1 |
| 9 | NCR reserved | Clock conversion factor |
| A | NCR reserved | Test mode |
| B | Configuration 2 | Configuration 2 |
| C | Configuration 3 | Configuration 3 |
| F | NCR reserved | Reserve FIFO byte (Config 2) |

## Register Set

Some ASC registers have different meanings during reads than writes. When CS/ is true, the register being accessed is determined by either RD/ or WR/ together with the address pins A0-3. The FIFO may be accessed using either CS/ or DACK/ together with RD/ or WR/. Address pins A0-3 are ignored when DACK/ is active, but must be driven when CS/ is active.

### Transfer Count (Write address 0, 1)

These two registers together form a 16-bit transfer count for DMA operations. It specifies the number of bytes that are to be transferred over the SCSI bus. Values written to these two registers will be stored internally and loaded into the transfer counter by any DMA command. These values remain unchanged while the transfer counter decrements. Thus, successive blocks of equal size may be transferred without reprogramming the count. They may be reprogrammed any time after the previous DMA operation has started, whether it has finished or not. Zero specifies a maximum length count (65536). These registers are not changed by any reset, their states are unpredictable after power-up.

### Transfer Counter (Read address 0, 1)

A read from these two addresses will return the value currently in the counter. DMA commands use the counter to terminate a transfer. Any DMA command will load count into the counter. A DMA NOP (80 hex) will load the counter while the non-DMA NOP (00) will not.

During SCSI data phase, the transfer counter decrements on the leading edge of:

| Target | Decremented by |
|---|---|
| Data in phase | DACK/ |
| Data out phase | REQO/ |

| Initiator | Decremented by |
|---|---|
| Synchronous data in | DACK/ |
| Asynchronous data in | ACKO/ |
| Data out | DACK/ |

Note that DACK/ can decrement the counter even if RD/ or WR/ do not go true. False DACK/s can cause the counter to get out of sync with the data stream, leading to subtle errors that are difficult to trace. When false DACK/s are expected to interfere with a temporarily suspended DMA operation, the DREQ hi-Z bit in Config 2 should be set.

The counter counts bytes. It decrements by one when transferring a single byte, or by two when transferring a word.

With two exceptions, non-DMA commands do not use the counter. During bus initiated selection and during target receive command sequence, the ASC decodes the group code field of the CDB (Command Descriptor Block), loads the counter with the number of bytes in the CDB, then decrements once for every byte received.

## FIFO Register (Read/write address 02)

The FIFO is a 16 by 9-bit First-In-First-Out buffer between the SCSI bus and memory. Read the *Functional Description* to understand it's use during SCSI transactions.

The SCSI bus may transfer 8 or 9-bit bytes to the FIFO, depending on the parity control bit settings (refer to *Table 6*). The microprocessor may transfer 8 or 9-bit bytes to the FIFO using CS/ and RD/ or WR/, and the address bits. An external DMA controller may transfer 8 or 9-bit bytes or 16 or 18-bit words (depending on chip configuration, byte control inputs and the parity control bits – refer to *Tables 2, 5, and 6*) to the FIFO using DACK/ and RD/ or WR/. When accessed by CS/, the address bits must be valid. When accessed by DACK/, the address bits are ignored.

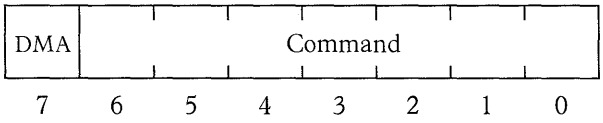
The bottom FIFO element and the FIFO flags are initialized to zero during hardware reset, software reset chip and the beginning of bus initiated selection or reselection. The contents of the rest of the FIFO are not changed by any reset, but when the flags are zero, successive FIFO reads will access the bottom register.

## Command Register (Read/write address 03)

The command register is a two deep, 8-bit read/write register used to give commands to the ASC. Up to two commands may be stacked in the command register. The second command may be written before the ASC completes (or even starts) the first. Reset chip, reset SCSI bus and target stop DMA execute immediately, all others wait for the previous command to complete. The last executed (or executing) command will remain in the command register and may be read by the microprocessor. Reading the command register has no effect on its contents. The internal sequencer maintains a working copy of the bottom of the command FIFO. The following conditions will cause the working copy to be cleared, so that the next command will fall through into the sequencer:

1) Hardware reset
2) Software reset
3) SCSI bus reset
4) SCSI bus disconnect
5) Bus initiated selection or reselection
6) Select command
7) Reconnect command if ATN is set
8) Select or reselect time-out
9) Target terminate command
10) Parity error detected in target mode
11) Assertion of ATN in target mode
12) Any phase change in initiator mode
13) Illegal command

**Figure 7. Command Register (Read/write address 03)**




If two commands are placed in the command register, two interrupts may result. If the first interrupt is not serviced before the second finishes, the second interrupt is stacked behind the first. When the interrupt register is read by the host to service the first interrupt, the contents of the status register, sequence step register, and interrupt register will change to describe the second interrupt. When using stacked commands, the phase latch bit (Config 2, bit 6) should be set.

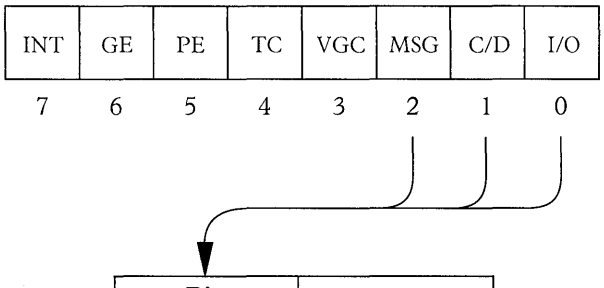
**Bit 7 (Enable DMA)**
When bit 7 is set, the command is a DMA instruction. When it is not set, the command is a non-DMA instruction. DMA instructions will load the internal byte counter with the value in the transfer count register, without changing the count register, then transfer data until that count decrements to zero. If the transfer terminates prematurely, the bits in the status, sequence step, and interrupt registers will indicate why.

**Bits 6-0 (Command code)**
The ASC commands are shown in *Table 19*. Bits 6, 5 and 4 specify a mode group. Commands from the miscellaneous group may be issued at any time (except target stop DMA). Commands from the disconnected, target or initiator groups will only be accepted by the ASC if it is in the same mode as the command when it falls to the bottom of the command FIFO. Otherwise, an illegal command interrupt will be generated. For example, after hardware or software reset, the ASC will be in the disconnected state. A command from either the target group or the initiator group will cause an illegal command interrupt. An enable selection/reselection command by itself will not change modes. However, if another SCSI device then selects the ASC, it will be in the target state; if another device reselects the ASC, it will then be in the initiator state. Similarly, any select command will place the ASC in initiator mode, while the reselect sequence command will place the ASC in target mode.

## Status Register (Read address 04)

The status register contains important flags that indicate certain events have occurred. Bits 7-3 are latched until the interrupt register is read. The phase bits are not normally latched. They may be latched (for stacked commands) by setting Config 2, bit 6.

**Figure 8. Status Register (Read address 04)**




| Bits 2 | 1 | 0 | SCSI Phase |
|---|---|---|---|
| 0 | 0 | 0 | Data out |
| 0 | 0 | 1 | Data in |
| 0 | 1 | 0 | Command |
| 0 | 1 | 1 | Status |
| 1 | 0 | 0 | ANSI reserved |
| 1 | 0 | 1 | ANSI reserved |
| 1 | 1 | 0 | Message out |
| 1 | 1 | 1 | Message in |

**Bit 7 (Interrupt)**
This bit is set whenever the ASC drives the INT output true. It may be polled. It is buffered from the actual output so that in wired-OR (shared interrupt) designs, this bit will indicate whether the ASC is attempting to interrupt the microprocessor. This bit is reserved by NCR in the 53C90. Hardware reset or software reset chip or a read from the interrupt register will release an active INT signal and also clear this bit.

**Bit 6 (Gross error)**
This bit is set when one of the following occurs:

1) The top of the FIFO is overwritten
2) The top of the command register has been overwritten
3) Direction of DMA transfer is opposite to the direction of the SCSI transfer
4) An unexpected phase change in initiator role during synchronous data phase

Gross error does not cause an interrupt, it may be detected only while servicing another interrupt. The bit is cleared by reading the interrupt register if the interrupt output is asserted. It will also be cleared by hardware reset, or software reset chip (but not SCSI reset).

**Bit 5 (Parity error)**
This bit will be set if parity checking is enabled in the Config 1 register and the ASC detects a SCSI parity error on incoming command, data, status or message bytes. Detected parity errors will not cause an interrupt, they are merely reported along with other interrupt causing events. If a parity error is detected during an initiator information in phase, ATN is automatically asserted on the SCSI bus.

This bit will be cleared by reading the interrupt register if the interrupt output is asserted. Hardware reset or software reset chip will clear this bit (but not SCSI reset).

**Bit 4 (Terminal count)**
This bit is set when the transfer counter decrements to zero. It resets when the transfer count is loaded. Since a DMA NOP (80 hex) command will load the transfer counter, it will also clear this bit. Note that a non-DMA NOP (00) will not load the counter and will not clear this bit. Reading the interrupt register will not clear this bit. Hardware reset or software reset chip will clear it (but not SCSI reset).

**Bit 3 (Valid group code)**
The name of this bit has changed from transfer complete in the 53C90 to valid group code in the 53C94, 95 and 96, but its function remains the same.

When the ASC is selected, it decodes the group code field in the first byte of the CDB (Command Descriptor Block). If the group code matches one defined in ANSI X3.131-1986, this bit will be set. An undefined group code (designated reserved by the ANSI committee) leaves it not set. If the SCSI-2 bit is set in the Config 2 register, group 2 commands will be recognized as ten-byte commands and the bit will be set. If the SCSI-2 bit is cleared, group 2 commands will be treated as reserved commands. Groups 3 and 4 are always treated as reserved commands. A reserved group command will cause the ASC to request 6 command bytes. The ASC recognizes group 6 as six-byte vendor unique commands and group 7 as ten-byte vendor unique commands. The valid group code bit will be cleared by reading the interrupt register if the interrupt output is asserted. It will also be cleared by hardware reset or software reset chip (but not by SCSI reset).

**Bits 2-0 (Phase bits)**
These bits indicate the phase on the SCSI bus. They may be latched or unlatched, depending on Config 3, bit 6.

When not latched, they indicate the phase at the time the status register was read. Per the ANSI definition of the phase signals, these bits must be stable during any status register read that follows an ASC generated interrupt.

The phase bits may be latched to permit stacking ASC commands. When the latch is enabled, the SCSI phase is latched upon command completion. The see through latch is re-opened when the interrupt register is read.

## Destination ID (Write address 04)

The least significant 3 bits of this register specify the encoded destination bus ID for a selection or reselection command. These bits are binary encoded, with 111 representing device ID 7 which appears as 80 (hex) on the SCSI bus. The most significant 5 bits are reserved by NCR. The destination ID is not changed by any reset, the states of these bits are unpredictable after power-up.

## Interrupt Register (Read address 05)

This 8-bit register is used in conjunction with the status register and sequence step register to determine the cause of an interrupt. Reading this register when the interrupt output is true will clear all three registers. The entire interrupt register will be cleared (0) by hardware reset or software reset chip (but not SCSI reset).

**Figure 9. Interrupt Register (Read address 05)**

| SCSI RST | Ill CMD | Dis | BS | FC | Re SEL | SEL ATN | SEL |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

**Bit 7 (SCSI reset detected)**
This bit is set if the SCSI reset reporting bit in the Config 1 register is not set and the chip detects a reset on the SCSI bus.

**Bit 6 (Illegal command)**
This bit is set when an unused code is placed in the command register or when the command is from a mode group different than the mode the ASC is currently in. Refer to the *Command Register* definition.

When the SCSI reset reporting bit in the Config 1 register is not set, this bit is set. The chip detects a reset on the SCSI bus.

**Bit 5 (Disconnect)**
In initiator mode, this bit is set when the target disconnects or a selection or reselection time-out occurs. When the ASC is in target mode, this bit is set if a terminate sequence or command complete sequence command causes the ASC to disconnect from the bus.

**Bit 4 (Bus service)**
This bit indicates that another device is requesting service. In target mode, it is set whenever the initiator asserts ATN (Attention).

In initiator mode, it is set whenever the target is requesting an information transfer phase.

**Bit 3 (Function complete)**
This bit will be set after any target mode command has completed. In initiator mode, it is set after a target has been selected (before transferring any command phase bytes), after command complete finishes, or after a transfer info command when the target is requesting message in phase.

**Bit 2 (Reselected)**
This bit is set during reselection phase to indicate that the ASC has been reselected as an initiator.

**Bit 1 (Selected with ATN)**
This bit is set during selection phase to indicate that the ASC has been selected as a target and that ATN (Attention) was asserted on the SCSI bus.

**Bit 0 (Selected)**
This bit is set during selection phase to indicate that the ASC has been selected as a target and that ATN (Attention) was false during selection.

## Time-Out (Write address 05)

This 8-bit write-only register specifies the amount of time to wait for a response during selection or reselection. (The ASC has no way to time-out if it never wins arbitration, it will keep trying indefinitely until it wins.) The time-out register is normally loaded to specify a time-out period of 250 ms to comply with the ANSI standard. The register value (RV) may be calculated from:

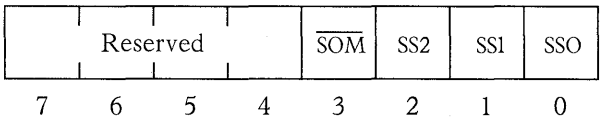
$$RV = \frac{\text{(time-out period) (CLK frequency)}}{\text{8192 (clock conversion factor)}}$$

For example, at 25 MHz, the register value that gives a 250 ms time-out period is 153 decimal or 99 hexadecimal. The clock conversion factor is defined in the description of write address 9. The time-out register remains unchanged by any reset, the states of these bits are unpredictable after power-up.

## Sequence Step (Read address 06)

The lower 3 bits of this register are used to indicate how far the internal sequencer was able to proceed in executing combination commands. This counter will be incremented at certain points in various algorithms to aid in error recovery if the previous command does not complete normally.

**Figure 10. Sequence Step Register (Read address 06)**




**Bit 3 (Synchronous offset max)**
When this active-low bit is not set, the synchronous offset counter has reached its maximum value.

**Bits 2-0 (Sequence step)**
The sequence step counter is set to zero at the beginning of certain commands. The counter is then incremented at specific points in the various algorithms to aid in error recovery. The possible states are described in *Tables 8 though 18.*

## Table 8. Initiator Select without ATN

| Sequence Step | Interrupt Register | Interpretation |
|---|---|---|
| 2 1 0 | 7 6 5 4 3 2 1 0 | |
| 0 0 0 | 0 0 1 0 0 0 0 0 | Arbitration complete; selection time-out; disconnected |
| 0 1 0 | 0 0 0 1 1 0 0 0 | Arbitration and selection complete; stopped because target did not assert command phase |
| 0 1 1 | 0 0 0 1 1 0 0 0 | Stopped during command transfer because target prematurely changed phase |
| 1 0 0 | 0 0 0 1 1 0 0 0 | Select sequence complete |

## Table 9. Initiator Select with ATN

| Sequence Step | Interrupt Register | Interpretation |
|---|---|---|
| 2 1 0 | 7 6 5 4 3 2 1 0 | |
| 0 0 0 | 0 0 1 0 0 0 0 0 | Arbitration complete; selection time-out; disconnected |
| 0 0 0 | 0 0 0 1 1 0 0 0 | Arbitration and selection complete; stopped because target did not assert message out phase; ATN still driven by ASC |
| 0 1 0 | 0 0 0 1 1 0 0 0 | Message out complete; sent one message byte with ATN true, then released ATN; stopped because target did not assert command phase after message byte was sent |
| 0 1 1 | 0 0 0 1 1 0 0 0 | Stopped during command transfer due to premature phase change; Some CDB bytes may not have been sent; check FIFO flags |
| 1 0 0 | 0 0 0 1 1 0 0 0 | Selection with ATN sequence complete |

## Table 10. Initiator Select with ATN3

| Sequence Step | Interrupt Register | Interpretation |
|---|---|---|
| 2 1 0 | 7 6 5 4 3 2 1 0 | |
| 0 0 0 | 0 0 1 0 0 0 0 0 | Arbitration complete; selection time-out; disconnected |
| 0 0 0 | 0 0 0 1 1 0 0 0 | Arbitration and selection complete; stopped because target did not assert message out phase; ATN still driven by ASC |
| 0 1 0 | 0 0 0 1 1 0 0 0 | Sent 1, 2 or 3 message bytes; stopped because target prematurely changed from message out phase or did not assert command phase after third message byte; ATN released only if third message byte was sent |
| 0 1 1 | 0 0 0 1 1 0 0 0 | Stopped during command transfer due to premature phase change; some CDB bytes may not have been sent; check FIFO flags |
| 1 0 0 | 0 0 0 1 1 0 0 0 | Selection with ATN3 sequence complete |

## Table 11. Initiator Select with ATN and Stop

| Sequence Step | Interrupt Register | Interpretation |
|---|---|---|
| 2 1 0 | 7 6 5 4 3 2 1 0 | |
| 0 0 0 | 0 0 1 0 0 0 0 0 | Arbitration complete; selection time-out; disconnected |
| 0 0 0 | 0 0 0 1 1 0 0 0 | Arbitration and selecton complete; stopped because target did not assert message out phase; ATN still asserted by ASC |
| 0 0 1 | 0 0 0 1 1 0 0 0 | Message out complete; sent one message byte; ATN on |

## Table 12. Target Selected without ATN

| Sequence Step | Interrupt Register | Interpretation |
|---|---|---|
| 2 1 0 | 7 6 5 4 3 2 1 0 | |
| 0 0 0 | 0 0 0 0 0 0 0 1 | Selected, loaded bus ID into FIFO, loaded null-byte message into FIFO |
| 0 0 1 | 0 0 0 0 0 0 0 1 | Stopped in command phase due to parity error; some command descriptor block bytes may not have been received; check FIFO flags |
| 0 0 1 | 0 0 0 1 0 0 0 1 | Same as above, initiator asserted ATN in command phase |
| 0 1 0 | 0 0 0 0 0 0 0 1 | Selected, received entire command descriptor block; check valid group status bit |
| 0 1 0 | 0 0 0 1 0 0 0 1 | Same as above, initiator asserted ATN in command phase |

## Table 13. Target Selected with ATN and SCSI-2 Bit Not Set

| Sequence Step | Interrupt Register | Interpretation |
|---|---|---|
| 2 1 0 | 7 6 5 4 3 2 1 0 | |
| 0 0 0 | 0 0 0 0 0 0 1 0 | Selected with ATN, stored bus ID and one message byte; stopped due either to parity error or invalid ID message |
| 0 0 0 | 0 0 0 1 0 0 1 0 | Selected with ATN, stored bus ID and one message byte; stopped because ATN remained true after 1st message byte |
| 0 0 1 | 0 0 0 0 0 0 1 0 | Stopped in command phase due to parity error; some CDB bytes not received; check valid group code bit and FIFO flags |
| 0 0 1 | 0 0 0 1 0 0 1 0 | Stopped in command phase; parity error and ATN true |
| 0 1 0 | 0 0 0 0 0 0 1 0 | Selection complete; received one message byte and the entire command descriptor block |
| 0 1 0 | 0 0 0 1 0 0 1 0 | Same as above, initiator asserted ATN during command phase |

## Table 14. Target Selected with ATN and SCSI-2 Bit Set

| Sequence Step | Interrupt Register | Interpretation |
|---|---|---|
| 2 1 0 | 7 6 5 4 3 2 1 0 | |
| 0 0 0 | 0 0 0 0 0 0 1 0 | Selected with ATN, stored bus ID and one message byte; stopped due either to parity error or invalid ID message |
| 1 0 0 | 0 0 0 0 0 0 1 0 | Parity error during second or third message byte |
| 1 0 0 | 0 0 0 1 0 0 1 0 | ATN remained true after third message byte |
| 1 0 1 | 0 0 0 0 0 0 1 0 | Received 3 message bytes then stopped in command phase due to parity error; some CDB bytes not received; check valid group code bit and FIFO flags |
| 1 0 1 | 0 0 0 1 0 0 1 0 | Stopped in command phase; parity error and ATN true |
| 1 1 0 | 0 0 0 0 0 0 1 0 | Selection complete; received 3 message bytes and the entire command descriptor block |

## Table 15. Target Receive Command Sequence

| Sequence Step | Interrupt Register | Interpretation |
|---|---|---|
| 2 1 0 | 7 6 5 4 3 2 1 0 | |
| 0 0 1 | 0 0 0 0 1 0 0 0 | Stopped during command transfer due to parity error; check FIFO flags |
| 0 0 1 | 0 0 0 1 1 0 0 0 | Stopped during command transfer due to parity error; ATN asserted by initiator |
| 0 1 0 | 0 0 0 0 1 0 0 0 | Received entire command descriptor block |
| 0 1 0 | 0 0 0 1 1 0 0 0 | Received entire CDB, initiator asserted ATN |

## Table 16. Target Disconnect Sequence

| Sequence Step | Interrupt Register | Interpretation |
|---|---|---|
| 2 1 0 | 7 6 5 4 3 2 1 0 | |
| 0 0 0 | 0 0 0 1 1 0 0 0 | Sent one message byte; stopped because initiator set ATN |
| 0 0 1 | 0 0 0 1 1 0 0 0 | Sent two message bytes; stopped because initiator set ATN |
| 0 1 0 | 0 0 1 0 1 0 0 0 | Disconnect sequence complete; disconnected, bus is free |

## Table 17. Target Terminate Sequence

| Sequence Step | Interrupt Register | Interpretation |
|---|---|---|
| 2 1 0 | 7 6 5 4 3 2 1 0 | |
| 0 0 0 | 0 0 0 1 1 0 0 0 | Sent status byte; stopped because initiator set ATN |
| 0 0 1 | 0 0 0 1 1 0 0 0 | Sent status and message bytes; stopped because initiator set ATN |
| 0 1 0 | 0 0 1 0 1 0 0 0 | Terminate sequence complete; disconnected, bus is free |

## Table 18. Target Command Complete Sequence

| Sequence Step | Interrupt Register | Interpretation |
|---|---|---|
| 2 1 0 | 7 6 5 4 3 2 1 0 | |
| 0 0 0 | 0 0 0 1 1 0 0 0 | Sent status byte; stopped because initiator set ATN |
| 0 0 1 | 0 0 0 1 1 0 0 0 | Sent status and message bytes; stopped because initiator set ATN |
| 0 1 0 | 0 0 0 0 1 0 0 0 | Command complete sequence complete |

## Synchronous Transfer Period (Write address 06)

The lower five bits of this register specify the minimum time between leading edges of successive REQ (Request) or ACK (Acknowledge) pulses. Synchronous data will be transmitted or received at the rate of one byte every N Clocks (CLK). N is related to the register value as shown below.

| Register Value | Clocks per Byte |
|---|---|
| 0 0 1 0 0 | 5 |
| 0 0 1 0 1 | 5 |
| 0 0 1 1 0 | 6 |
| 0 0 1 1 1 | 7 |
| • | • |
| • | • |
| • | • |
| 1 1 1 1 1 | 31 |
| 0 0 0 0 0 | 32 |
| 0 0 0 0 1 | 33 |
| 0 0 0 1 0 | 34 |
| 0 0 0 1 1 | 35 |

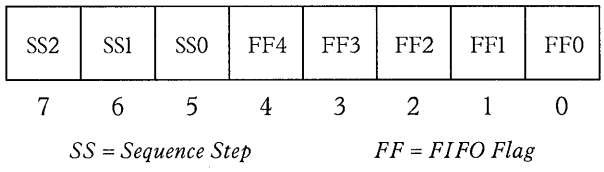
Missing entries in the table above follow the binary code. The upper three bits are reserved by NCR. This register defaults to 5 after hardware RESET or software reset chip (but not SCSI reset).

## FIFO Flags (Read address 07)

The least significant 5 bits of this register indicate how many bytes are currently in the FIFO. The value is binary encoded. The flags should not be polled while transferring data because they will not be stable while the SCSI interface is changing the contents of the FIFO.

The upper three bits are duplicates of the sequence step register bits in normal mode. If test mode is enabled, bit 5 is set to indicate that the offset counter is not zero. Not zero means that synchronous data may continue to be transferred. Zero means that the synchronous offset count has expired and the ASC will not transfer any more data until it receives an acknowledge.

**Figure 11. FIFO Flags and Sequence Step Register (Read address 07 – Normal mode)**




## Synchronous Offset (Write address 07)

The least significant four bits of this register specify whether the ASC will transfer data phase bytes synchronously or asynchronously. *Zero specifies asynchronous transfer.* Any other value specifies the synchronous offset; the number of data phase bytes that may be sent synchronously without an acknowledge (either REQ (Request) or ACK ([Acknowledge]), depending on whether the ASC is in initiator or target mode).

When transmitting to the SCSI bus, the ASC will stop sending bytes when it reaches this offset, and thereafter send one byte for every acknowledge it receives from the other SCSI device.

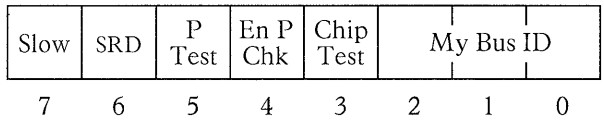
When receiving from the SCSI bus, the ASC will send an acknowledge every time a byte is removed from its FIFO on the DMA (or microprocessor) interface. The maximum offset of fifteen allows a receiving ASC to store data in its FIFO while the external DMA controller gains control of the memory bus.

The synchronous offset is cleared (0) by hardware reset or software reset chip (but not SCSI reset).

## Configuration 1 Register (Config 1) (Read/write 08)

This 8-bit read/write register specifies various operating conditions for the ASC. Any bit pattern written to this register may be read back and should be identical.

**Figure 12. Configuration 1 Register (Config1) (Read/write address 08)**



**Bit 7 (Slow cable mode)**
Slow cable mode will seldom be necessary. It compensates for excessive capacitive loading on the SCSI data signals by inserting an extra CLK period between data being asserted on the bus and REQ or ACK being driven true. This bit is cleared (0) by hardware reset or software reset chip (but not SCSI reset).

**Bit 6 (SCSI reset reporting interrupt disable)**
This bit disables the reporting of a SCSI reset. If the SCSI reset signal goes true when this bit is set, the ASC will disconnect from the SCSI bus and remain idle in the disconnected state without interrupting the host. If the bit is not set the ASC will respond to the SCSI reset by first interrupting the host. This bit is cleared by hardware reset or software reset chip (but not SCSI reset).

**Bit 5 (Parity test mode)**
When bit 5 is set, the parity bit will equal bit 7 when unloading the FIFO to either the SCSI bus or the microprocessor bus. This allows parity errors to be created so that hardware and software may be tested. This bit must not be set during normal operation. Refer to *Parity Checking and Generation.* This bit is cleared (0) by hardware reset or software reset chip, (but not SCSI reset).

**Bit 4 (Enable parity checking)**
When this bit is set, the ASC will check parity on incoming SCSI bytes during any information transfer phase except when receiving bad bytes. Detected parity errors will cause a bit to be set in the status register but will not cause an interrupt. In Initiator role, bad parity will also set ATN (Attention) on the SCSI bus. When this bit is not set, parity will not be checked, the bit in the status register will not be set, and ATN will not be asserted. Refer to *Parity Checking and Generation.* This bit is cleared (0) by hardware reset or software reset chip (but not SCSI reset).

**Bit 3 (Chip test mode enable)**
When this bit is set, the chip is placed in special test mode which enables the test register at address 0A (hex). Once it has been set, the chip must be reset (hard or soft but not SCSI) before normal operation can begin. This bit should not be set during normal operation. This bit is cleared (0) by hardware reset or software reset chip (but not SCSI reset).

**Bit 2-0 (My bus ID)**
This bit field is the bus ID of this device. It is the ID to which the ASC responds during bus initiated selection or reselection, and the ID that the ASC uses to arbitrate for the bus. The name of this field has changed from bus ID on the C90, but its function remains the same. This three-bit field is binary encoded. It is not changed by any reset, after power-up the states of these bits are unpredictable.

## Clock Conversion (Write address 09)

This register must be set according to the CLK (Clock) input frequency. All timings longer than 400 ns depend on this register correctly agreeing with the CLK frequency. The least significant three bits are binary encoded, and should be set to one of the four values below.

| CLK Frequency (MHz) | Clock Conversion Factor |
|---|---|
| 10 | 2 |
| 10.01 to 15 | 3 |
| 15.01 to 20 | 4 |
| 20.01 to 25 | 5 |

This register must never be loaded with 1. Hardware reset or software reset chip will set the clock conversion register to 2. SCSI reset will not affect it. The upper 5 bits of this register are reserved by NCR.

## Test Register (Write address 0A)

This register is enabled by setting the special test mode bit in Config 1 at address 08. After test mode has been entered, a hardware reset or software reset chip must occur before normal operation can begin.

**Figure 13. Test Register (Write address 0A)**


**Bit 2 (All outputs to high impedance)**
When this bit is set, all bi-directional and all output pins go to high impedance and will not significantly load a TTL or compatible device.

**Bit 1 (Initiator mode)**
When this bit is set, the ASC is artificially forced into Initiator mode. Any initiator command will be accepted by the ASC. For example, a set ATN command will cause ATN to be driven on the SCSI bus even if the ASC is disconnected.

**Bit 0 (Target mode)**
When this bit is set, the ASC is artificially forced into Target mode. Any target command will be accepted by the ASC. For example, a DMA command will load or unload the FIFO and set the SCSI phase, data and REQ signals even if arbitration and selection have not occurred.

## Configuration 2 Register (Config 2) (Read/write address 0B)

The 53C94, 95, and 96 have a second configuration register that did not exist in the original C90. After hardware reset or software reset chip, the bits in this register are all cleared, which makes the chip compatible with 53C90 software. Any bit pattern written to this register may be read back and should be identical.

**Figure 14. Configuration 2 Register (Config 2) (Read/write address 0B)**

| RFB | EPL | EBC | DREQ Hi Z | SCSI 2 | BPA | RPE | DPE |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

**Bit 7 (Reserve FIFO byte)**
This bit allows 16-bit DMA to begin on misaligned word boundaries for initiator synchronous data in. It must be set before the phase changes to synchronous data in.

Synchronous data in requires DMA to move data through the FIFO – the microprocessor must not access the FIFO. When this bit is set, a single byte is reserved in the bottom of the FIFO when the phase changes to synchronous data in. This reserved byte will become the low byte of the first 16-bit word that the ASC will transfer to memory using DMA, the first byte received across the SCSI bus will become the high byte of the first word.

While servicing the interrupt for a phase change to synchronous data in on a misaligned boundary, the microprocessor should copy the byte at start address 1 from its own memory to ASC register 0F (hex) then issue the transfer info command. When the ASC writes its first word to memory (via DMA) it will overwrite the low byte (which is not part of the current SCSI data block) with the value placed in register 0F. Thus the low byte of the first word will be overwritten with a copy of itself, and the high byte will be the first byte received over the SCSI bus. The remaining bytes will be aligned on word boundaries and will be transfered 16 bits at a time.

This bit has no effect for phases other than initiator synchronous data in. The ASC "knows" whether the transfer is synchronous or not by the value in the synchronous transfer register.

This bit is cleared by a hardware reset, or software reset chip, or a write to register 0F (hex) when the interrupt output it true and the SCSI phase is synchronous data in. This bit is not affected by a SCSI bus reset.

**Bit 6 (Enable phase latch)**
When this bit is not set, the phase bits, reported in the status register are live indicators of the state of the SCSI phase lines. When this bit is set, the phase is latched at each command completion. This permits simpler software routines for stacked commands. Reading the interrupt register re-opens the phase latch. This bit is left not set by hardware reset or software reset chip, but is not affected by SCSI reset.

**Bit 5 (Enable byte control)**
When the mode strapping pins are set to mode 2 (mode 1 = 1 and mode 0 = 0) this bit will enable byte control on the DMA interface. In mode 2, the byte control inputs BHE and A0 instruct the ASC to transfer data on the low byte or the high byte, or both bytes of a 16-bit word. When this bit is not set, the byte control inputs are ignored. Hardware reset or software reset chip will leave this bit not set, while SCSI reset will not affect it.

**Bit 4 (DREQ high impedance)**
When this bit is set, the DREQ output (DMA request) goes to high impedance and will not significantly load a TTL compatible device. This is useful when several devices share the DMA request line (known as wired-OR). When this bit is set, the ASC will ignore any activity on the DACK/ (DMA acknowledge) input.

When this bit is cleared, the DREQ output will be driven to TTL high or low voltages. When this bit is cleared, DACK/ is enabled to decrement the transfer counter and load or unload the FIFO, depending or WR/ or RD/. DACK/ should not pulse true without RD/ or WR/ because the transfer counter may decrement without transferring any data. Refer to *Transfer Counter Register.*

**Bit 3 (SCSI-2)**
Allows the ASC to support two new features adopted in SCSI-2: the 3-byte message exchange for *Tagged-Queuing* and *Group 2* commands.

*Tagged-Queing*
When this bit is set, and the ASC is selected with ATN (Attention) it will request either one or three message bytes depending on whether ATN remains true or goes false. If ATN is still true after the first byte has been received, the ASC may request two more message bytes before switching to command phase. If ATN goes false, it will request only one message byte then switch to command phase. When the bit is not set it will request a single message byte (as a target) when selected with ATN, and abort the selection sequence (as an initiator) if the target does not switch to command phase after one message byte has been transferred. Refer to *Bus Initiated Selection.*

*Group 2 Commands*
When the SCSI-2 bit is set, group 2 commands are recognized as 10-byte commands. Receiving a group 2 command with this bit set will set the valid group code bit in the status register. If the SCSI-2 bit is not set the ASC will treat group 2 commands as reserved commands, it will request only 6 bytes in command phase and will not set the valid group code status bit.

**Bit 2 (Target bad parity abort)**
When this bit is set, the ASC will abort a receive command or receive data sequence when the ASC detects a parity error.

**Bit 1 (Register parity enable)**
When this bit is set, parity from the host DBP pins will be loaded into the FIFO when CS/ and WR/ are both true. When this bit is not set the ASC generates parity from the host data bus when CS/ and WR/ are both true and places it in the FIFO along with the data from which was generated.

When the ASC is moving data from the FIFO to the SCSI bus, it will flag outgoing parity errors if either this bit or the DMA parity enable bit are set.

**Bit 0 (DMA Parity enable)**
When this bit is set, parity from the host DBP pins will be loaded into the FIFO when DACK/ and WR/ are both true. When this bit is not set, the ASC generates parity from the host data bus when DACK/ and WR/ are both true and places it in the FIFO along with the data from which is was generated.

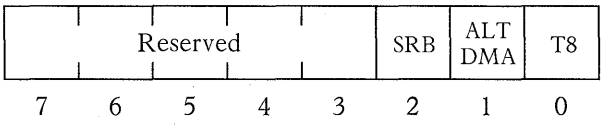
When the ASC is moving data from the FIFO to the SCSI bus, it will flag outgoing parity errors if either this bit or the register parity enable bit are set.

## Configuration 3 Register (Config 3) (Read/write address 0C)

The 53C94, 95, and 96 have a third configuration register that did not exist in the original C90.

After hardware reset or software reset chip, the bits in this register are all cleared, which makes the chip compatible with 53C90 software. Any bit pattern written to this register may be read back and should be identical.

**Figure 15. Configuration 3 Register (Config 3) (Read/write address 0C)**




**Bit 2 (Save residual byte)**
The residual byte is the modulo 2 remainder at the end of a 16-bit DMA data stream. If byte control is used (hardware configured for mode 2 and bit 5 set in Config 2) this feature should not be used.

When this bit is set, DREQ (DMA Request) will not be asserted for the last residual byte at the end of a transfer, if such residue exists. The microprocessor should remove (or insert) the residual byte from (into) the FIFO. If this bit is not set and the transfer ends with a single byte left over, DREQ will be asserted and a subsequent 16-bit DMA transfer will contain the last byte on the lower half of the bus.

For DMA read, the upper eight bits will be set to ones. This bit has no effect in 8-bit DMA mode or during any SCSI phase except data in or data out. It is left not set by hardware reset or software reset chip, but is not affected by SCSI reset.

**Bit 1 (Alternate DMA mode)**
Setting this bit modifies the DMA interface to take advantage of the demand mode on an 8237 DMA controller when the threshold 8-bit is also set. Refer to *DMA Burst Mode.* This bit should not be set unless threshold 8 is also set. All but the last DMA burst will be four words (or eight bytes if 8-bit DMA) long. The last burst may be one to four words (or one to eight bytes), depending the modulo 8 remainder left in the transfer counter.

When this bit is set, DMA data is strobed into or out from the ASC when:

- *DMA Write*

  For multiple DMA writes, DACK/ remains asserted while DBWR/ toggles for each write.

- *DMA Read*

  1) In modes 0 and 1 (Refer to *Host Bus Configuration*) during multiple DMA transfers, DACK/ remains asserted while RD/ toggles for each DMA transfer. If the alternate DMA bit is not set, the ASC outputs data when DACK/ is true, RD/ need not be true.
  2) In mode 2, during multiple DMA transfers, DACK/ remains asserted while DBRD/ toggles for each transfer. The ASC outputs data when both DACK/ and DBRD/ are true.
  3) In mode 3, DACK/ must toggle for each DMA read. The ASC outputs data when DACK/ is true, RD/ need not be true.

**Bit 0 (Threshold 8)**
Setting this bit causes the ASC to delay assertion of DREQ (DMA Request) until it can transfer eight bytes (four words). This higher threshold applies only to SCSI data phases. The threshold for all other phases is one byte for 8-bit DMA mode, or one word for 16-bit DMA mode.

*When threshold 8 is set, the maximum synchronous offset is limited to 7.* DREQ will go true when:

- *DMA Write*

  DREQ is true whenever the top eight bytes of the FIFO are empty.

- *DMA Read*

  DREQ will go true when:

  1) End of transfer
     a) Target mode: DREQ is set when the transfer counter is zero or ATN is set.
     b) Initiator synchronous data in: DREQ is true when the transfer counter is less than eight.
     c) Initiator mode, not synchronous data in: DREQ is true when the transfer counter is zero, or after any phase change.
  2) Not end of transfer
     a) Initiator synchronous data in: DREQ is true if the transfer counter is greater than seven and the bottom eight bytes of the FIFO are full.
     b) Not initiator synchronous data in: DREQ is true whenever the bottom eight bytes of the FIFO are full.

## FIFO Bottom (Write address 0F)

This register is used only during initiator synchronous data in to align 16-bit DMA transfers to word boundaries. When Config 2, bit 7 is set and the phase changes to initiator synchronous data in, the ASC reserves a byte in the bottom of the FIFO. If the microprocessor writes a byte to this register after the interrupt (for synchronous data in) the byte will become the low byte of the first word transfered out from the FIFO to the external DMA controller. The first byte received across the SCSI bus will become the high byte of the first 16-bit word transfered to memory.

## Table 19. ASC Command Set

| Command Register | | | | | | | | Command Mnemonic | Interrupt |
|---|---|---|---|---|---|---|---|---|---|
| **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** | **Miscellaneous Group** | |
| X | 0 | 0 | 0 | 0 | 0 | 0 | 0 | NOP | No |
| X | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Flush FIFO | No |
| X | 0 | 0 | 0 | 0 | 0 | 1 | 0 | Reset chip | No |
| X | 0 | 0 | 0 | 0 | 0 | 1 | 1 | Reset SCSI bus | No * |
| | | | | | | | | **Disconnected State Group** | |
| X | 1 | 0 | 0 | 0 | 0 | 0 | 0 | Reselect sequence | Yes |
| X | 1 | 0 | 0 | 0 | 0 | 0 | 1 | Select without ATN sequence | Yes |
| X | 1 | 0 | 0 | 0 | 0 | 1 | 0 | Select with ATN sequence | Yes |
| X | 1 | 0 | 0 | 0 | 0 | 1 | 1 | Select with ATN and stop sequence | Yes |
| X | 1 | 0 | 0 | 0 | 1 | 0 | 0 | Enable selection/reselection | No |
| X | 1 | 0 | 0 | 0 | 1 | 0 | 1 | Disable selection/reselection | Yes |
| X | 1 | 0 | 0 | 0 | 1 | 1 | 0 | Select ATN3 | Yes |
| | | | | | | | | **Target State Group** | |
| X | 0 | 1 | 0 | 0 | 0 | 0 | 0 | Send message | Yes |
| X | 0 | 1 | 0 | 0 | 0 | 0 | 1 | Send status | Yes |
| X | 0 | 1 | 0 | 0 | 0 | 1 | 0 | Send data | Yes |
| X | 0 | 1 | 0 | 0 | 0 | 1 | 1 | Disconnect sequence | Yes |
| X | 0 | 1 | 0 | 0 | 1 | 0 | 0 | Terminate sequence | Yes |
| X | 0 | 1 | 0 | 0 | 1 | 0 | 1 | Target command complete sequence | Yes |
| X | 0 | 1 | 0 | 0 | 1 | 1 | 1 | Disconnect | No |
| X | 0 | 1 | 0 | 1 | 0 | 0 | 0 | Receive message | Yes |
| X | 0 | 1 | 0 | 1 | 0 | 0 | 1 | Receive command sequence | Yes |
| X | 0 | 1 | 0 | 1 | 0 | 1 | 0 | Receive data | Yes |
| X | 0 | 1 | 0 | 1 | 0 | 1 | 1 | Receive command sequence | Yes |
| X | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Target abort DMA | No ** |
| | | | | | | | | **Initiator State Group** | |
| X | 0 | 0 | 1 | 0 | 0 | 0 | 0 | Transfer information | Yes |
| X | 0 | 0 | 1 | 0 | 0 | 0 | 1 | Initiator command complete sequence | Yes |
| X | 0 | 0 | 1 | 0 | 0 | 1 | 0 | Message accepted | Yes |
| X | 0 | 0 | 1 | 1 | 0 | 0 | 0 | Transfer pad | Yes |
| X | 0 | 0 | 1 | 1 | 0 | 1 | 0 | Set ATN | No |
| X | 0 | 0 | 1 | 1 | 0 | 1 | 1 | Reset ATN | No |

* *The command itself does not cause an interrupt, however, external connection of the RSTO/ pin to RSTI/ pin causes an interrupt if the SCSI reset reporting is not disabled in the configuration register.*

** *The command itself does not cause an interrupt, however, it may allow a stalled command to finish and generate an interrupt.*

## Command Set

From the programmers point of view, DMA commands will move data between memory and the SCSI bus, non-DMA commands will move data between the FIFO and the SCSI bus. Non-DMA commands require the microprocessor to move data between the FIFO and memory. DMA commands require an external DMA controller to move data between the FIFO and memory. A command with bit 7 set is a DMA command. A command with bit 7 not set is a non-DMA command. DMA commands will load the transfer counter with whatever value is in the transfer count register, so the value must be correct before issuing the command.

**Table 20. Miscellaneous Commands**

| DMA | Non-DMA | Mnemonic |
|---|---|---|
| 80 | 00 | No Operation (NOP) |
| 81 | 01 | Flush FIFO |
| 82 | 02 | Reset chip |
| 83 | 03 | Reset SCSI bus |

## Miscellaneous Commands

### NOP

No-Operation (NOP). The 53C94 and 53C95, 53C96 require this command only after hardware reset or software reset chip. A DMA NOP (80 hex) may be used to load the transfer counter with the value in transfer count register. No interrupt is generated from this command.

### Flush FIFO

The flush FIFO command initializes the FIFO to the empty condition by resetting the FIFO flags and setting the bottom byte of the FIFO to zero.

### Reset Chip

This command resets all functions in the chip and returns it to a disconnected state. The command has the same effect as a hardware reset, with the exception that reset chip cannot change between single-ended mode or differential mode.

### Reset SCSI Bus

This command will assert the RSTO (SCSI Reset Output) signal for T2 µs, where

T2 = 130 (CLK period) (CCF)

CCF = Clock Conversion Factor. Refer to *Write Register 9*. and CLK is the clock input to the ASC. This command does not cause an interrupt, however, since RSTI will be externally connected to RSTO, an interrupt will be generated unless it is disabled in the Config 1 register.

### Target Stop DMA

This command allows the microprocessor to stop a DMA data transfer command. The ASC must be in target state when this command falls to the bottom of the command FIFO or an illegal command interrupt will be generated. Target stop DMA may only be used when all of the following are true.

1) Either a target sends data or a target receives data command are currently executing.

2) The DMA controller has stopped.

3) The ASC is in *steady state*, ie:
   a) Send data – the FIFO is empty.
   b) Receive asynchronous data – the FIFO is full or the transfer counter is zero.
   c) Receive sync data – the transfer counter is zero or the FIFO flags are 15.

Upon receiving this command, the ASC will reset the DMA interface (release DREQ) then terminate the current command. It will not generate its interrupt until the rest of the completion criteria are met.

1) Send asynchronous data – completes immediately.

2) Send synchronous data – completes when the offset counter is zero.

3) Receive asynchronous data – completes immediately. There will be data in the FIFO which should be removed by the microprocessor.

4) Receive synchronous data – completes when all outstanding SCSI ACKs have been received. The offset counter is separate from the transfer counter. There will be data in the FIFO which should be removed by the microprocessor.

**Table 21. Disconnected State Commands**

| DMA | Non-DMA | Mnemonic |
|---|---|---|
| C0 | 40 | Reselect sequence |
| C1 | 41 | Select without ATN sequence |
| C2 | 42 | Select with ATN sequence |
| C3 | 43 | Select with ATN and stop sequence |
| C4 | 44 | Enable selection or reselection |
| C5 | 45 | Disable selection and reselection |
| C6 | 46 | Select with ATN3 |

## Disconnected State Commands

If any of the disconnected state commands are received by the ASC when it is not in the disconnected state, the command will be ignored, the command register will be cleared, and the ASC will generate an illegal command interrupt.

### Reselect Sequence

This command will cause the ASC target to arbitrate for the bus then enter reselection phase when it wins arbitration. The identify message, required by SCSI protocol, must either be placed in the FIFO by the microprocessor before issuing the command or must be transferred by DMA which involves setting the transfer count to one and setting up the external DMA controller. In either case, the time-out and destination ID registers must have previously been programmed. The sequence will terminate early if a reselect time-out occurs. If it terminates normally, a function complete interrupt will occur.

### Select without ATN Sequence

This command will cause the ASC initiator to arbitrate for the bus, enter selection phase when it wins and send the CDB (Command Descriptor Block). The 6, 10 or 12-byte CDB must have either been placed in the FIFO previously by the microprocessor or must be transferred by DMA which involves setting the transfer count to 6, 10 or 12 and programming the external DMA controller. In either case, the time-out and destination ID registers must have previously been programmed. This command terminates early if a reselection time-out occurs, or the target does not assert command phase or the target removes command phase too early. If it terminates normally, a function complete/bus service interrupt will be generated.

### Select with ATN Sequence

This command will cause the ASC initiator to arbitrate for the bus, select a device with ATN true then send one message phase byte followed by 6, 10 or 12 command phase bytes. The message and command bytes must have either been placed in the FIFO by the microprocessor or must be transferred by DMA which involves setting the transfer count to 7, 11 or 13 and programming the external DMA controller. In either case, the time-out and destination ID registers must have previously been programmed. This command terminates early if a select time-out occurs, target does not assert message phase followed by command phase, or target removes command phase early. If it completes normally, a function complete and bus service interrupt will be generated.

### Select with ATN and Stop

This command should be used in place of the one above when multiple message phase bytes are to be sent (for example, the synchronous negotiation). The command will select a target with ATN asserted, send one message phase byte that has previously been stored in the FIFO, and generate a bus service interrupt and stop. After the interrupt, the FIFO may be filled with other message bytes. A transfer info command will then transfer bytes with ATN true until the FIFO empties. If a DMA transfer info command is used, ATN will remain true until the transfer counter decrements to zero.

### Enable Selection/Reselection

After receiving this command, the ASC will respond to bus initiated selection or reselection. A command that causes the ASC to select or reselect will cancel this command. The command must be re-issued within 250 ms after the ASC disconnects to preserve ANSI recommended timings. If DMA is enabled, incoming information will be placed in memory. If DMA is not enabled, incoming information will remain in the FIFO.

### Disable Selection/Reselection

This command disables an earlier enable selection/reselection command. If bus initiated selection or reselection had not yet begun when this command is received by the ASC, it will generate a function complete interrupt. If bus initiated selection or reselection had already begun, this command (and every other command) will be ignored. Refer to *Bus Initiated Selection* and *Bus Initiated Reselection.*

### Select with ATN3 Sequence

This command is similar to the select with ATN command, but sends three message bytes instead of one. It will cause the ASC initiator to arbitrate for the bus, select a device with ATN true, send three message phase bytes, deassert ATN, then send 9, 13 or 15 command phase bytes. The message and command bytes must have either been placed in the FIFO by the microprocessor or must be transferred by DMA. This involves setting the transfer count to 7, 11 or 13 and programming the external DMA controller. In either case, the time-out and destination ID registers must have previously been programmed. This command terminates early if a selection time-out occurs, the target does not assert message phase followed by command phase, or the target removes command phase early. If it completes normally, a function complete and bus service interrupt will be generated.

**Table 22. Initiator Commands**

| DMA | Non-DMA | Mnemonic |
|---|---|---|
| 90 | 10 | Transfer information |
| 91 | 11 | Initiator command complete sequence |
| - | 12 | Message accepted |
| 98 | 18 | Transfer pad |
| - | 1A | Set ATN (Attention) |
| - | 1B | Reset ATN |

## Initiator Commands

If the ASC is not in initiator state when it receives any of these commands, the command will be ignored, an illegal command interrupt will be generated and the command register will be cleared. Refer to *Command Register*.

If BSY goes false while the ASC is connected as an initiator, it will generate a disconnected interrupt. The interrupt output will occur 1.5 to 3.5 CLK cycles after BSY goes false.

When the ASC receives the last byte of a message in phase, it will leave ACK (Acknowledge) asserted on the bus to prevent the target from sending any more bytes until the initiator decides to accept or reject the message. For non DMA commands, the last byte means that the FIFO is empty. For DMA commands, the transfer counter signals the last byte.

If parity checking is enabled and the ASC detects a parity error while in initiator mode, it will automatically assert ATN prior to deasserting ACK for the byte which has the error. The one exception is after a phase change to synchronous data in, and is described as follows.

If the synchronous offset register is non-zero (synchronous) and the phase changes to data in, the DMA interface is immediately disabled and the reporting of a parity error during data in phase is delayed. The phase change to data in will: latch the FIFO flags to indicate how many bytes were in the FIFO (these bytes will be lost), clear the FIFO, load the FIFO with the first data in byte, generate an interrupt, and continue to load the FIFO with incoming data in bytes as long as the target sends them, but not more than the specified offset. To continue receiving data in bytes, the microprocessor would normally issue the transfer information command to re-enable the DMA interface. If parity checking is enabled, and a parity error occurred on a previous input phase (message in or status) then the parity error flag will be set in the status register and ATN will be set on the SCSI bus. If a parity error occurred during the data in phase, the parity bit will not be set nor will ATN be asserted until after the ASC receives the subsequent transfer information command.

## Transfer Information

This command can be used to send or receive any information phase bytes, but is most often used for data transfer. For synchronous transfer, DMA must be used. The ASC will continue to transfer information until one of the following terminating events occurs:

- Transfer is complete. This successful completion will create a bus service interrupt. For a DMA transfer info, the transfer is complete when the transfer count decrements to zero and the FIFO is empty and the target asserts REQ for the next byte. For non-DMA transfer info in which the ASC is sending bytes to the SCSI bus, transfer is complete when the FIFO empties and the target asserts REQ for the next byte. For non DMA transfer info in which the ASC is receiving bytes from the SCSI bus, transfer is complete after one byte is received and the target asserts REQ for the next byte. Thus non DMA transfer info commands will generate an interrupt for every byte received.

  If the phase is message out, the ASC removes ATN prior to asserting ACK for the last byte of the message. For non-DMA, the FIFO flags indicate the last byte. For DMA, the transfer counter indicates the last byte.

- Target changes phase. The ASC clears the command register and generates a bus service interrupt, after the target asserts REQ for the next byte.

- Target releases BSY (Busy). The ASC generates a disconnected interrupt.

- The ASC receives the last byte of a message in phase. (For non-DMA every byte is assumed to be the last byte. For DMA, the transfer counter signals the last byte). The ASC leaves ACK asserted and generates a function complete interrupt.

  All message in and status phase transfers are handled one byte at a time. If DMA is enabled, the next byte will not be received until the current byte has been written to buffer memory and the FIFO is empty. If DMA is not enabled, each byte will create an interrupt.

## Initiator Command Complete Sequence

This command will cause the ASC to receive a status byte followed by a message byte. It terminates early if the target does not assert message in phase, or if the target disconnects. After receiving the message byte, the ASC leaves ACK asserted on the bus to allow the initiator to assert ATN if the message is unacceptable.

## Message Accepted

This command releases the ACK signal on the SCSI bus. Any of the commands that receive bytes during message phase will leave ACK asserted after receiving the last message byte. To accept the message, issue this command. To reject the message, set ATN then issue this command.

## Transfer Pad

Transfer pad is usually an error recovery technique. It is useful when a target requests more bytes than an initiator has to send, or when an initiator must receive and discard a number of bytes from a target.

When transmitting to the SCSI bus, transfer pad will fill the FIFO with null bytes and send them to the SCSI bus. When receiving from the SCSI bus, transfer pad will receive bytes, place them on the top of the FIFO and discard them from the bottom of the FIFO.

When sending pad bytes to the SCSI bus, DMA must be enabled. No DMA requests are actually made, but the ASC uses the transfer counter to end the transfer.

The command terminates under the same conditions as the transfer info command, except that the ASC does not leave ACK asserted on the last byte of a message in phase. If the command terminates before the transfer counter reaches zero (due to phase change or disconnect) the FIFO may contain pad bytes.

## Set ATN

This command asserts attention on the SCSI bus. No interrupt is generated from this command. ATN stays asserted until the last byte of a message out phase.

DMA commands use the transfer counter to indicate the last byte. For non-DMA commands, the last byte means that the FIFO is empty. For DMA transfers, the last byte means that the transfer counter is zero. ATN will also be released if the target prematurely disconnects.

## Reset ATN

This command causes ATN to be released. It does not cause an interrupt.

*This command must not be used when connected to a device supporting the Common Command Set (CCS).* The ASC obeys CCS protocol by releasing ATN on the last byte of a message out phase. The reset ATN command is provided for older devices which do not respond properly to the ATN condition.

**Table 23. Target Commands**

| DMA | Non-DMA | Mnemonic |
|---|---|---|
| A0 | 20 | Send message |
| A1 | 21 | Send status |
| A2 | 22 | Send data |
| A3 | 23 | Disconnect sequence |
| A4 | 24 | Terminate sequence |
| A5 | 25 | Target command complete sequence |
| A7 | 27 | Disconnect |
| A8 | 28 | Receive message sequence |
| A9 | 29 | Receive command |
| AA | 2A | Receive data |
| AB | 2B | Receive command sequence |
| 84 | 04 | Target abort DMA |

## Target Commands

If the ASC receives any of these commands when it is not in target state, it will ignore the command, clear the command register, and generate an illegal command interrupt. Refer to *Command Register*.

Normal completion of these commands will cause a function complete interrupt. If ATN is asserted, the bus service bit will be set in the status register. If the ASC was idle when ATN was asserted, a bus service interrupt will be generated, the function complete bit will be zero, and the command register will be cleared.

## Send Message

This command will cause the ASC to assert message in phase and send bytes until the FIFO is empty and the transfer counter is zero (if DMA).

## Send Status

This command will cause the ASC to assert status phase and send bytes until the FIFO is empty and the transfer counter is zero (if DMA).

## Send Data

This command will cause the ASC to assert data in phase and send bytes until the FIFO is empty and the transfer counter is zero (if DMA).

## Disconnect Sequence

This command will cause the ASC to assert message in phase, send two bytes, then disconnect from the SCSI bus. Normally, the first byte will be a save data pointers message and the second will be a disconnect message. These bytes must be loaded into the FIFO by the microprocessor, or may be loaded by DMA. If ATN is asserted by the initiator, the bus service and function complete bits will be set, an interrupt will be generated, but the ASC will not disconnect.

### Terminate Sequence

This command will cause the ASC to first assert status phase, send one byte; then assert message in phase and send one more byte. These bytes must be loaded into the FIFO by the microprocessor, or may be loaded by DMA. If ATN is asserted by the initiator, the bus service and function complete bits will be set, an interrupt will be generated, but the ASC will not disconnect.

### Target Command Complete Sequence

This command is similar to terminate sequence, but is used for linked commands. It will cause the ASC to first assert status phase, send one byte, then assert message in phase and send one more byte. The message byte will normally be a command complete message. If ATN is asserted by the initiator, the bus service and function complete bits will be set, an interrupt will be generated, but the ASC will not disconnect.

### Disconnect

This command causes the ASC to release all SCSI bus signals except RSTO (once triggered, RSTO is driven true for 25 µs or so, depending on CLK frequency and clock conversion factor). The ASC returns to the disconnected state without generating an interrupt.

### Receive Message Sequence

This command will cause the the ASC to assert message out phase and receive one byte, then generate an interrupt.

### Receive Command

This command will cause the ASC to assert command phase and receive bytes from the initiator. For non-DMA receive command, only one byte per interrupt may be received. DMA receive command will interrupt after the transfer counter decrements to zero.

### Receive Data

This command will cause the ASC to assert data out phase and receive bytes from the initiator. For non-DMA receive data, only one byte per interrupt may be received. DMA receive data will interrupt after the transfer counter decrements to zero.

### Receive Command Sequence

This command will cause the ASC to assert command phase and receive a number of bytes, which will vary according to the the group code field of the first byte. If the SCSI-2 bit is set in the Config 2 register, group 2 commands will be recognized as 10-byte commands. If the SCSI-2 bit is cleared, group 2 commands will be recognized as reserved commands. Groups 3 and 4 are always reserved. The ASC will request six bytes for reserved commands, six bytes for group 6 vendor unique commands, and 10 bytes for group 7 vendor unique commands.

## Notes:

## Notes:

# DC Electrical Characteristics

## Absolute Maximum Stress Ratings

| Parameter | Symbol | Pins | Test Conditions | Min | Max | Unit |
|---|---|---|---|---|---|---|
| Storage temperature | $T_{STG}$ | - | - | -55 | 150 | °C |
| Supply voltage | $V_{DD}$ | - | - | -0.5 | 7.0 | V |
| Input voltage | $V_{IN}$ | - | - | $V_{SS}$-0.5 | $V_{DD}$+ 0.5 | V |
| Latch-up current | $I_{LP}$ | - | $-2V < V_{PIN} < +8V$ | ±100 | - | mA |
| Electrostatic discharge | ESD | - | Human body model | - | - | - |
| | | SCSI pins | 100 pF at 1.5K ohms | ±6000 | - | V |
| | | Other pins | 100 pF at 1.5K ohms | ±3000 | - | V |

*Conditions that exceed the absolute maximum stress limits may destroy the device.*

*Conditions that exceed the operating limits may cause the device to function incorrectly.*

## Operating Conditions

| Parameter | Symbol | Pins | Test Conditions | Min | Max | Unit |
|---|---|---|---|---|---|---|
| Supply voltage | $V_{DD}$ | - | - | 4.75 | 5.25 | V |
| Supply current | $I_{DD}$ | - | Static* | - | 10 | mA |
| Supply current | $I_{DD}$ | - | Dynamic | - | 50 | mA |
| Ambient temperature | $T_A$ | - | - | 0 | 70 | °C |

* *Static means: all inputs at $V_{SS}$, all outputs floating, and all bi-directional pins configured as inputs.*

## Inputs

| Parameter | Symbol | Pins | Test Conditions | Min | Max | Unit |
|---|---|---|---|---|---|---|
| Input high voltage | $V_{IH}$ | - | - | 2.0 | $V_{DD}$ + 0.5 | V |
| Input low voltage | $V_{IL}$ | - | - | $V_{SS}$ -0.5 | 0.8 | V |
| Input leakage current | $I_{IN}$ | Non-SCSI | $0 < V_{IN} < V_{DD}$<br>$4.75 \le V_{DD} \le 5.25$ | -10 | 10 | µA |
| Hysteresis | $V_H$ | BSYI/, SELI/, REQI/, RSTI/ | - | 200 | - | mV |
| Input low leakage | $I_{IL}$ | SCSI | $V_{IN}$= 0.5; $0 \le V_{DD} \le 5.5$ | -10 | 0.0 | µA |
| Input high leakage | $I_{IH}$ | SCSI | $V_{IN}$= 2.7; $0 \le V_{DD} \le 5.5$ | 0.0 | 10 | µA |
| Capacitance | $C_{IN}$ | | - | - | 10 | pF |

## Outputs

| Parameter | Symbol | Pins | Test Conditions | Min | Max | Unit |
|---|---|---|---|---|---|---|
| Output high voltage | $V_{OH}$ | DREQ, IGS, TGS | $I_{OH}$= -2 mA | 2.4 | $V_{DD}$ | V |
| Output low voltage | $V_{OL}$ | DREQ, IGS, TGS, INT/ | $I_{OL}$= 4 mA | $V_{SS}$ | 0.4 | V |
| Output low voltage | $V_{OL}$ | RSTO/, SELO, ACKO/, REQO/, SDOP/, BSYO/, SDO7-0 | $I_{OL}$= 48 mA | $V_{SS}$ | 0.5 | V |
| Hi Z state leakage | $I_{OZ}$ | - | $0 < V_{OUT} < V_{DD}$ | -10 | 10 | µA |
| Fall Time | $T_F$ | SCSI pins | SCSI termination | 4 | - | ns |
| Capacitance | $C_{OUT}$ | - | - | - | 10 | pF |

## Bi-Directional Pins

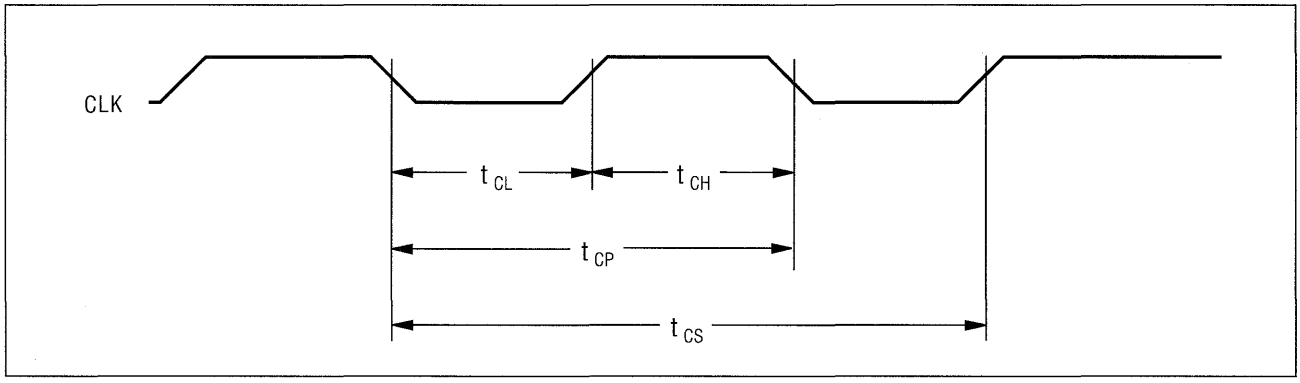
| Parameter | Symbol | Pins | Test Conditions | Min | Max | Unit |
|---|---|---|---|---|---|---|
| Input high voltage | $V_{IH}$ | - | - | 2.0 | $V_{DD}$+ 0.5 | V |
| Input low voltage | $V_{IL}$ | - | - | $V_{SS}$-0.5 | 0.8 | V |
| Output high voltage | $V_{OH}$ | SDI7-0, DBI5-0, DBP1-0, PAD7-0 | $I_{OH}$= -2 mA | 2.4 | $V_{DD}$ | V |
| Output low voltage | $V_{OL}$ | SDI7-0, DBI5-0, DBP1-0, PAD7-0 | $I_{OL}$= 4 mA | $V_{SS}$ | 0.4 | V |
| Output low voltage | $V_{OL}$ | ATNIO/, MSGIO/, C/DIO, I/OIO | $I_{OL}$= 48 mA | $V_{SS}$ | 0.5 | V |
| Hysteresis | $V_H$ | ATNIO/, MSGIO/, C/DIO, I/OIO | - | 200 | - | mV |
| Input low leakage | $I_{IL}$ | SDI7-0, ATNIO/, MSGIO/, C/DIO, I/OIO | $0 < V_{IN} < V_{DD}$ | -10 | 10 | µA |
| Input current, low | $I_{IL}$ | DBI5-0, DBP1-0, PAD7-0 | $V_{IN}$= $V_{IL}$ | -400 | 10 | µA |
| Input current, high | $I_{IH}$ | DBI5-0, DBP1-0, PAD7-0 | $V_{IN}$= $V_{IH}$ | -10 | 10 | µA |
| Hi Z pull-up current | $I_{PU}$ | DBI5-0, DBP1-0, PAD7-0 | - | 100 | 400 | µA |
| Capacitance | $C_{IO}$ | | - | - | 10 | pF |

## AC Electrical Characteristics

The AC characteristics described herein apply over the operating voltage and temperature range, $4.75\ V \leq V_{DD} \leq 5.25\ V$ and $0°C \leq T_A \leq 70°C$. Output timing is based on simulation under worst case conditions (4.75 V, 70°C) and worst case processing using the following termination. All timings in this specification are taken from the 10% and 90% points with respect to the specified $V_{OL}$ and $V_{OH}$ of the waveforms.

| Pin | Termination |
|---|---|
| REQ, TGS, IGS, SDIP/, SDI7/-O/, PAD7-0 | 50 pF |
| INT/ | 50 pF, 1K pull-up |
| DB15-0, DBP0, DBP1 | 80 pF |
| SDOP/, SDO7/-0/, RSTO/, SELO/, BSYO/, ATNIO/, MSGIO/, C/DIO, I/OIO, REQO/, ACKO/ | 200 pF, 110 pullup, 165 pulldown |

## Clock Input

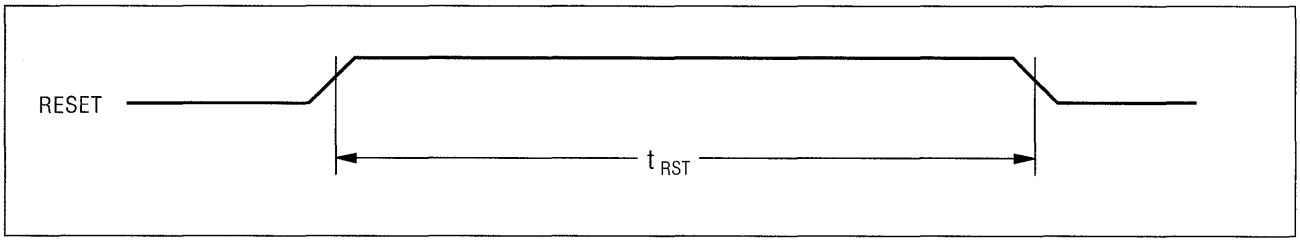



| Parameter | Symbol | Minimum | Maximum | Units | Notes |
|---|---|---|---|---|---|
| Clock frequency, asynchronous SCSI | $t_{CPA}$ | 10 | 25 | MHz | - |
| Clock frequency, synchronous SCSI | $t_{CPS}$ | 12 | 25 | MHz | - |
| Clock high time | $t_{CH}$ | 14.58 | - | ns | 1 |
| Clock low time | $t_{CL}$ | 14.58 | - | ns | 1 |
| Clock period | $t_{CP}$ | 40 | 100 | ns | - |
| Synchronization latency = $t_{CP} + t_{CL}$ | $t_{CS}$ | - | - | - | - |

*1) For synchronous SCSI data transmission, CLK must also meet the following:*

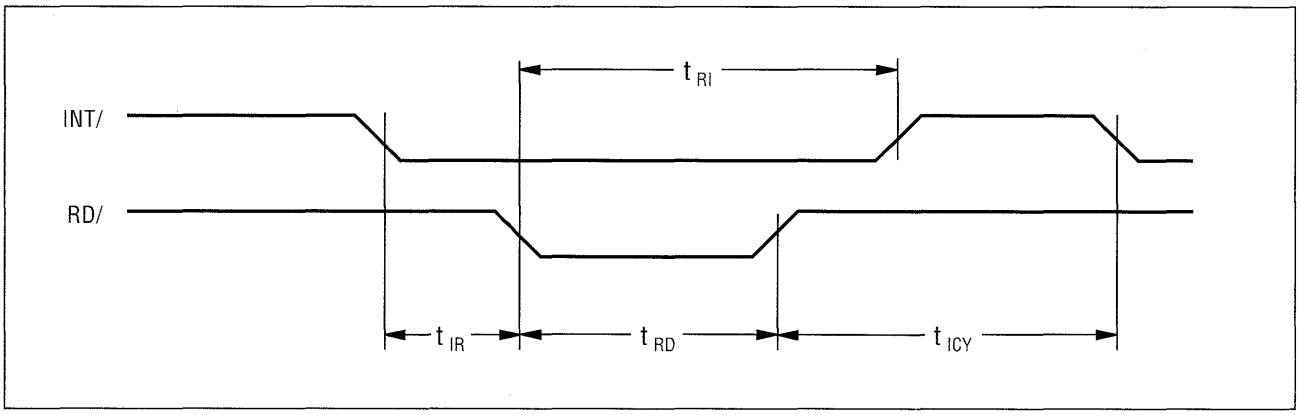
$$2t_{CP} + t_{CL} \geq 97.92\ ns \quad \textit{and} \quad 2t_{CP} + t_{CH} \geq 97.92\ ns$$

## Reset Input




| Parameter | Symbol | Minimum | Maximum | Units | Notes |
|---|---|---|---|---|---|
| RESET pulse width | $t_{RST}$ | 500 | - | ns | - |

## Interrupt Output

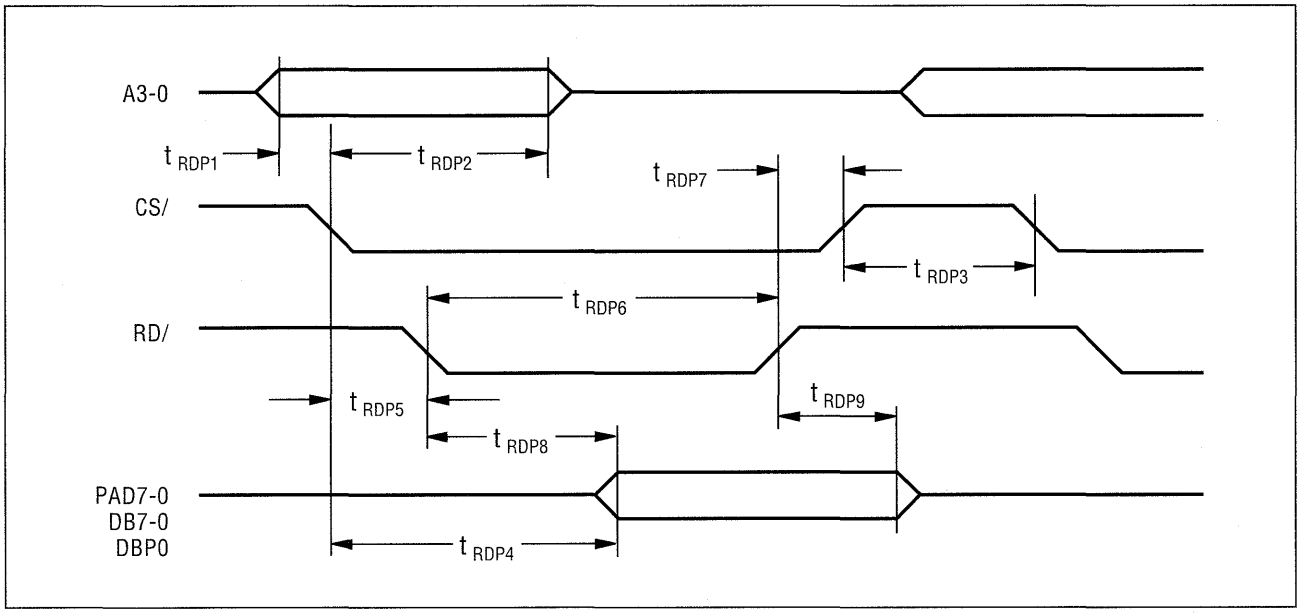



| Parameter | Symbol | Minimum | Maximum | Units | Notes |
|---|---|---|---|---|---|
| INT/ low to interrupt register read | $t_{IR}$ | 0 | - | ns | 2 |
| RD/ pulse width | $t_{RD}$ | 50 | - | ns | 1 |
| RD/ low to INT/ high | $t_{RI}$ | 0 | 100 | ns | - |
| RD/ high to INT/ low | $t_{ICY}$ | $t_{CS} - t_{RI}$ | - | ns | - |

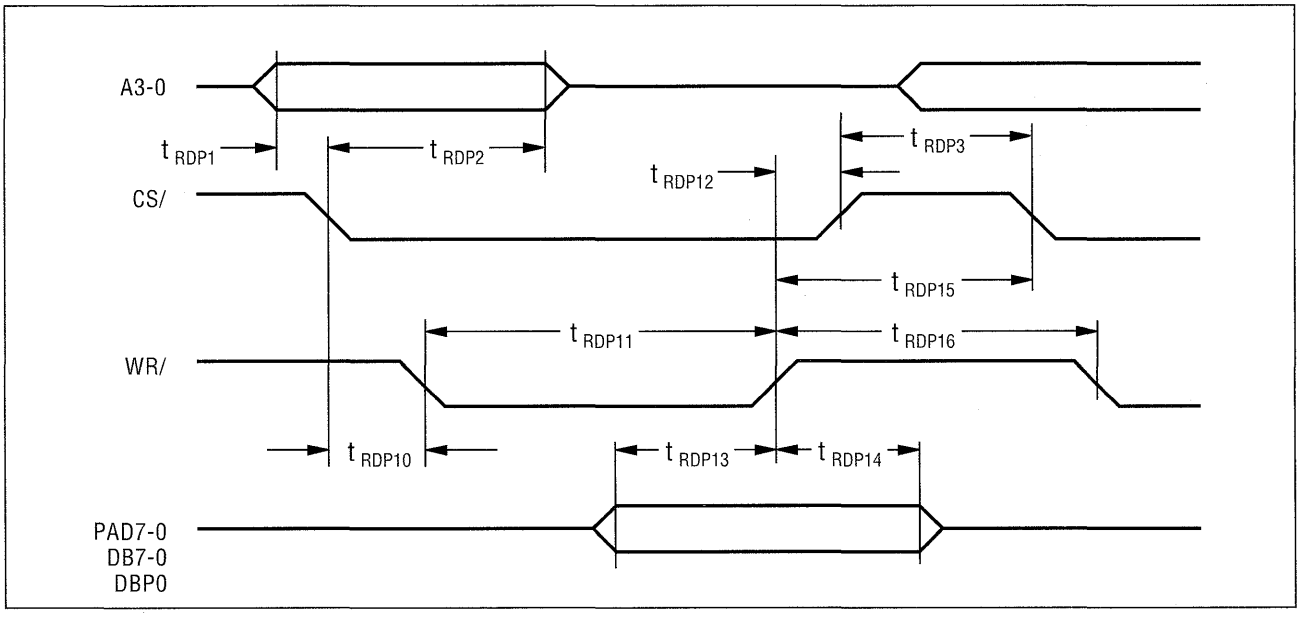
*1) Refer to the register read specifications for the timing requirements of CS/, RD/, and address for reading the interrupt register.*

*2) The interrupt register should not be read when INT/ is false.*

## Register Read, DB and Nonmultiplexed PAD Bus




## Register Write, DB and Nonmultiplexed PAD Bus



## Register Interface, DB Bus and Nonmultiplexed PAD

| Parameter | Symbol | Minimum | Maximum | Units | Notes |
|---|---|---|---|---|---|
| Address setup to CS/ low | $t_{RDP1}$ | 0 | - | ns | - |
| Address hold from CS/ low | $t_{RDP2}$ | 50 | - | ns | - |
| CS/ high to CS/ low | $t_{RDP3}$ | 40 | - | ns | - |
| CS/ low to read data valid | $t_{RDP4}$ | - | 90 | ns | 3 |
| CS/ setup to RD/ low | $t_{RDP5}$ | 0 | - | ns | 4 |
| RD/ pulse width | $t_{RDP6}$ | 50 | - | ns | - |
| RD/ high to CS/ high | $t_{RDP7}$ | 0 | - | ns | 4 |
| RD/ low to data valid | $t_{RDP8}$ | - | 50 | ns | 5 |
| Read data output disable | $t_{RDP9}$ | 2 | 40 | ns | - |
| CS/ setup to WT/ low | $t_{RDP10}$ | 0 | - | ns | 6 |
| WR/ pulse width | $t_{RDP11}$ | 40 | - | ns | - |
| WR/ high to CS/ high | $t_{RDP12}$ | 0 | - | ns | 6 |
| Data setup to WR/ high | $t_{RDP13}$ | 15 | - | ns | - |
| Data hold after WR/ high | $t_{RDP14}$ | 0 | - | ns | - |
| WR/ high to CS/ low | $t_{RDP15}$ | 60 | - | ns | - |
| WR/ high to WT/ low | $t_{RDP16}$ | 60 | - | ns | - |

*1) CS/ must make a high to low transition to capture a new register address.*

*2) For single bus mode (mode zero or mode one) DACK/ must be inactive during all register accesses. WR/ must be tied to DBWR/.*

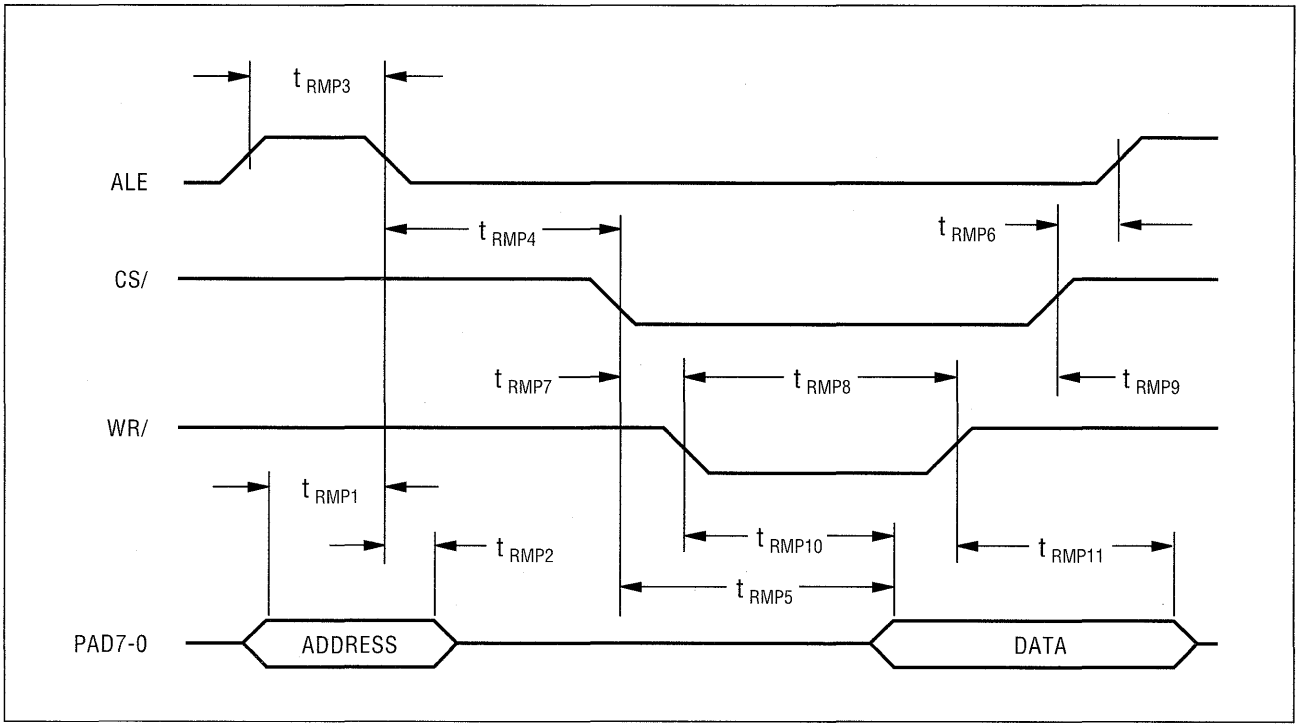
*3) $t_{RDP8}$ must also be satisfied.*

*4) RD/ edges may precede or follow CS/ edges. Recommended values are $t_{RDP5} > t_{RDP4} - t_{RDP8}$ and $t_{RDP7} > t_{RDP9}$. If RD/ is held low, the time from CS/ low to stable data is $t_{RDP4}$ and the output disable time from CS/ high is $t_{RDP9}$.*

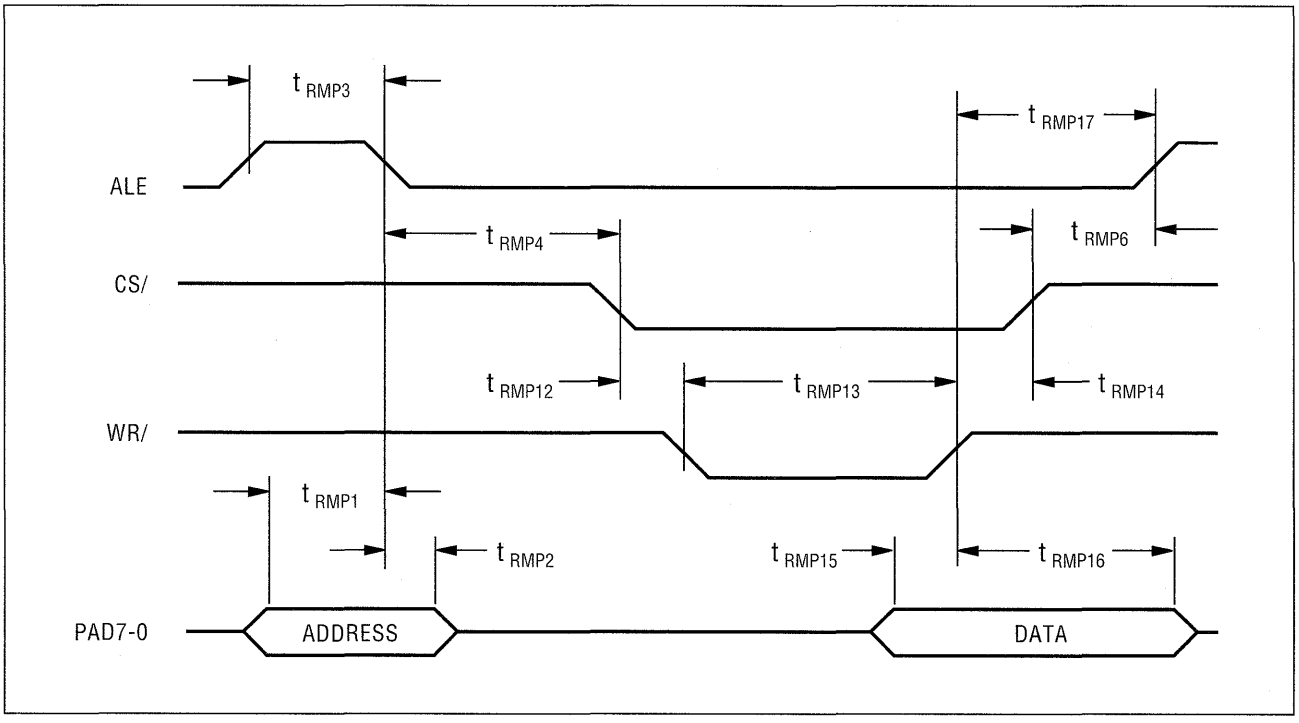
*5) $t_{RDP4}$ must also be satisfied.*

*6) WR/ edges may precede or follow CS/ edges. Recommended values are $t_{RDP10} \geq 0$ and $t_{RDP12} \geq 0$. If WR/ is held low, the data setup to CS/ high is 25 ns minimum; data hold from CS/ high is 60 ns minimum; $t_{RDP3}$ is 60 ns minimum.*

## Register Read, Multiplexed PAD




## Register Write, Multiplexed PAD



## Register Interface, DB Bus and Multiplexed PAD

| Parameter | Symbol | Minimum | Maximum | Units | Notes |
|---|---|---|---|---|---|
| Address setup to ALE low | $t_{RMP1}$ | 10 | - | ns | - |
| Address hold from ALE low | $t_{RMP2}$ | 10 | - | ns | - |
| ALE pulse width | $t_{RMP3}$ | 20 | - | ns | - |
| ALE low CS/ low | $t_{RMP4}$ | 10 | - | ns | - |
| CS/ low to data | $t_{RMP5}$ | - | 90 | ns | 3 |
| CS/ high to ALE high | $t_{RMP6}$ | 50 | - | ns | - |
| CS/ setup to RD/ low | $t_{RMP7}$ | 0 | - | ns | 4 |
| RD/ pulse width | $t_{RMP8}$ | 50 | - | ns | - |
| RD/ high to CS/ high | $t_{RMP9}$ | 0 | - | ns | 4 |
| RD/ low to data valid | $t_{RMP10}$ | - | 50 | ns | 5 |
| Data release time | $t_{RMP11}$ | 2 | 40 | ns | - |
| CS/ setup to WR/ low | $t_{RMP12}$ | 0 | - | ns | 6 |
| WR/ pulse width | $t_{RMP13}$ | 40 | - | ns | - |
| WR/ high to CS/ high | $t_{RMP14}$ | 0 | - | ns | 6 |
| Data setup to WR/ high | $t_{RMP15}$ | 15 | - | ns | - |
| Data hold from WR/ high | $t_{RMP16}$ | 0 | - | ns | - |
| WR/ high to ALE high | $t_{RMP17}$ | 50 | - | ns | - |

*1) If DMA is active, the FIFO register must not be accessed.*

*2) ALE must pulse to capture a new register address.*

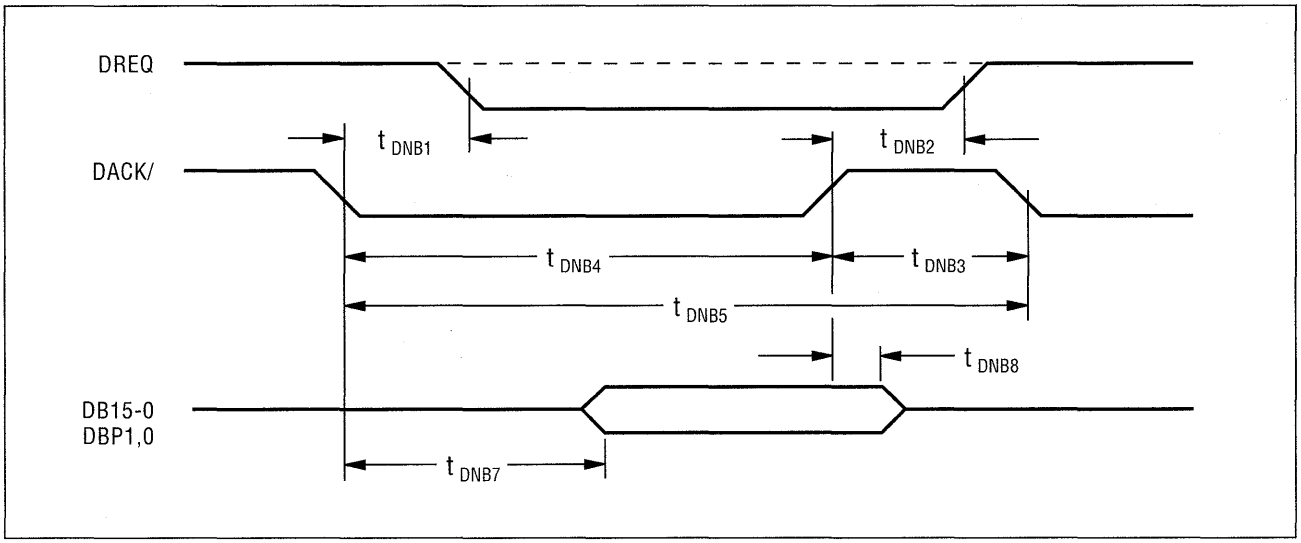
*3) $t_{RMP10}$ must also be satisfied.*

*4) RD/ edges may precede or follow CS/ edges. Recommended values are $t_{RMP7} > t_{RMP5} - t_{RMP10}$ and $t_{RMP9} > t_{RMP11}$. If RD/ is held low, the time from CS/ low to stable data is $t_{RMP5}$ and the data release time from CS/ high is $t_{RMP11}$.*

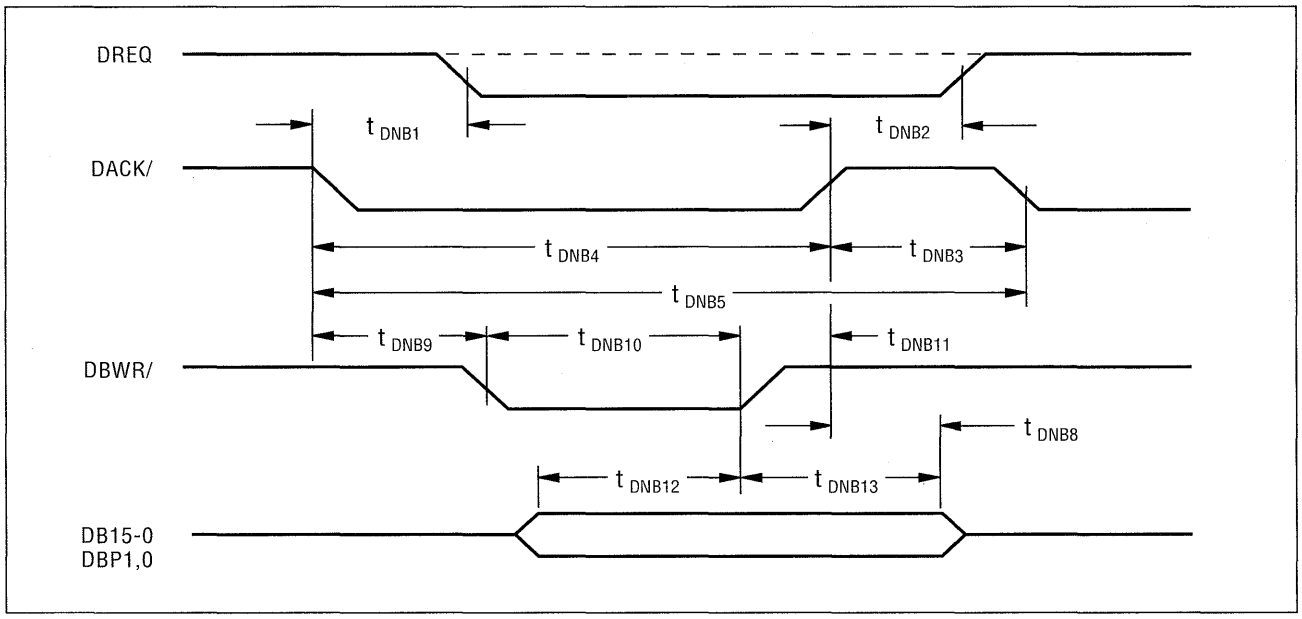
*5) $t_{RMP5}$ must also be satisfied.*

*6) WR/ edges may precede or follow CS/ edges. Recommended values are $t_{RMP12} \geq 0$ and $t_{RMP14} \geq 0$. If WR/ is held low, data setup to CS/ high is 25 ns and data hold from CS/ high is 50 ns minimum.*

## DMA Read, without Byte Control




## DMA Write, without Byte Control



## DMA Interface Nonbyte Control

| Parameter | Symbol | Minimum | Maximum | Units | Notes |
|---|---|---|---|---|---|
| DACK/ low to DREQ low | $t_{DNB1}$ | - | 38 | ns | 5 |
| DACK/ high to DREQ high | $t_{DNB2}$ | - | 40 | ns | - |
| DACK/ high to DACK/ low | $t_{DNB3}$ | 12 | - | ns | 4 |
| DACK/ pulse width | $t_{DNB4}$ | 60 | - | ns | - |
| DACK/ period (low to low) | $t_{DNB5}$ | 100 | - | ns | - |
| DACK/ period (high to high) | $t_{DNB6}$ | $t_{CS}+50-t_{DNB3}$ | - | ns | 6 |
| DACK/ low to data valid | $t_{DNB7}$ | $2t_{CP}$ | 41 | ns | - |
| Data release time | $t_{DNB8}$ | 2 | 40 | ns | - |
| DACK/ low to DBWR/ low | $t_{DNB9}$ | 0 | - | ns | 4 |
| DBWR/ pulse width | $t_{DNB10}$ | 50 | - | ns | - |
| DBWR/ high to DACK/ high | $t_{DNB11}$ | 0 | - | ns | 4 |
| Data setup to DBWR/ | $t_{DNB12}$ | 15 | - | ns | - |
| Data hold to DBWR/ | $t_{DNB13}$ | 0 | - | ns | - |
| DBWR/ high to DBWR/ low | $t_{DNB14}$ | 40 | - | ns | - |

*1) Alternate DMA is disabled.*

*2) For single bus mode (mode zero or mode one) CS/ must be inactive while DACK is active.*

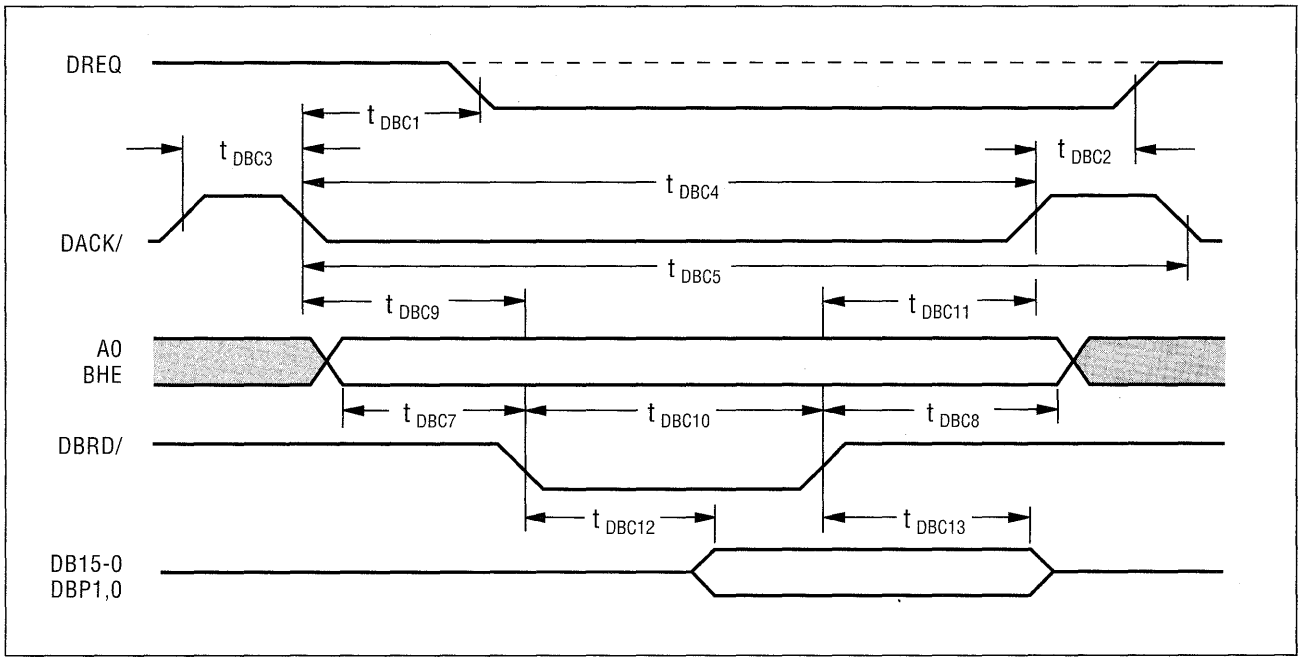
*3) ACK/ must toggle once for each access.*

*4) DBWR/ edges may precede or follow DACK/ edges. Recommended values are: $t_{DNB9} \geq 0$ and $t_{DNB11} \geq 0$. If DBWR/ is held low, the data setup to DACK/ high 15 ns minimum; data hold from DACK/ high is 15 ns mimimum; and $t_{DNB3}$ is 40 ns mimimum.*

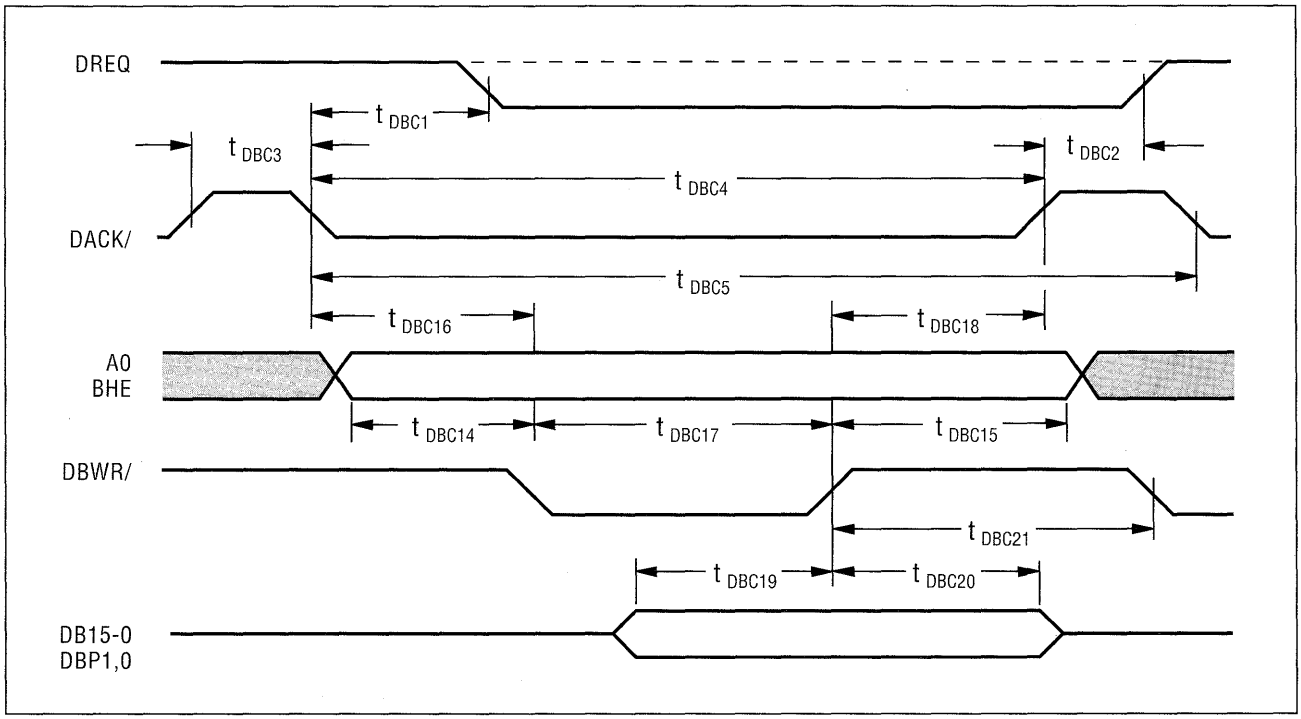
*5) DREQ may stay high if the FIFO has room to accept another word (or byte if byte mode) during DMA write, or send another word (or byte if byte mode) during DMA read. If the current DMA acknolowdge cycle fills the FIFO (write) or empties the FIFO (read), then DREQ will go bw.*

*6) Minimum high to high DACK/ period for synchronous SCSI transfer is: $t_{CS}+50-t_{DNB3}$ for asynchronous SCSI and $2\,t_{CP}$ for synchronous SCSI.*

## DMA Read, with Byte Control




## DMA Write, with Byte Control



## DMA Interface with Byte Control

| Parameter | Symbol | Minimum | Maximum | Units | Notes |
|---|---|---|---|---|---|
| DACK/ low to DREQ low | $t_{DBC1}$ | - | 38 | ns | 7 |
| DACK/ high to DREQ high | $t_{DBC2}$ | - | 40 | ns | 7 |
| DACK/ high to DACK/ low | $t_{DBC3}$ | 12 | - | ns | 2 |
| DACK/ pulse width | $t_{DBC4}$ | 60 | - | ns | - |
| DACK/ period (low to low) | $t_{DBC5}$ | 100 | - | ns | - |
| DACK/ period (high to high) | $t_{DBC6}$ | - | - | ns | - |
| BHE, SA0 setup to DBRD/ low | $t_{DBC7}$ | 20 | - | ns | - |
| BHE, SA0 hold from DBRD/ high | $t_{DBC8}$ | 20 | - | ns | - |
| DACK low to DBRD/ low | $t_{DBC9}$ | 0 | - | ns | - |
| DBRD/ pulse width | $t_{DBC10}$ | 60 | - | ns | 4 |
| DBRD/ high to DACK high | $t_{DBC11}$ | 0 | - | ns | 5 |
| DBRD/ to data valid | $t_{DBC12}$ | - | 51 | ns | - |
| Data release time | $t_{DBC13}$ | 2 | 40 | ns | - |
| BHE, SA0 to DBWR/ low | $t_{DBC14}$ | 20 | - | ns | - |
| BHE, SA0 hold from DBWR/ high | $t_{DBC15}$ | 20 | - | ns | - |
| DACK/ low to DBWR/ low | $t_{DBC16}$ | 0 | - | ns | - |
| DBWR/ pulse width | $t_{DBC17}$ | 50 | - | ns | - |
| DBWR/ high to DACK high | $t_{DBC18}$ | 0 | - | ns | 6 |
| Data setup to DBWR/ high | $t_{DBC19}$ | 15 | - | ns | - |
| Data hold from DBWR/ high | $t_{DBC20}$ | 0 | - | ns | - |
| DBWR/ high to DBWR/ low | $t_{DBC21}$ | 40 | - | ns | - |

*1) Alternate DMA is disabled.*

*2) For single bus mode, (mode zero or mode one) CS/ must be inactive.*

*3) DACK/ must toggle once for each access.*

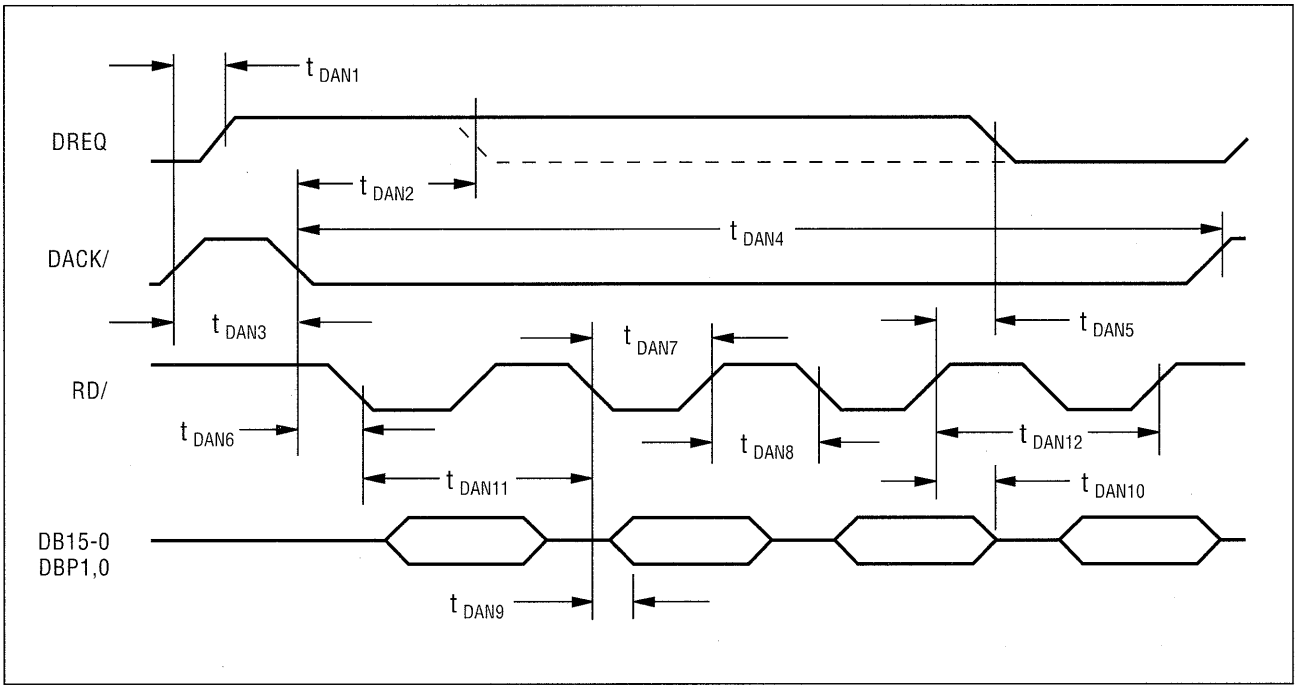
*4) DBRD/ must meet the same pulse width requirements as DACK/.*

*5) DBRD/ trailing edge may precede or follow DACK/ trailing edge. The recommended value is: $t_{DVC11} \geq 0$. If DBWR/ is held low, the time from DACK/ low to stable data is 41 ns max.*

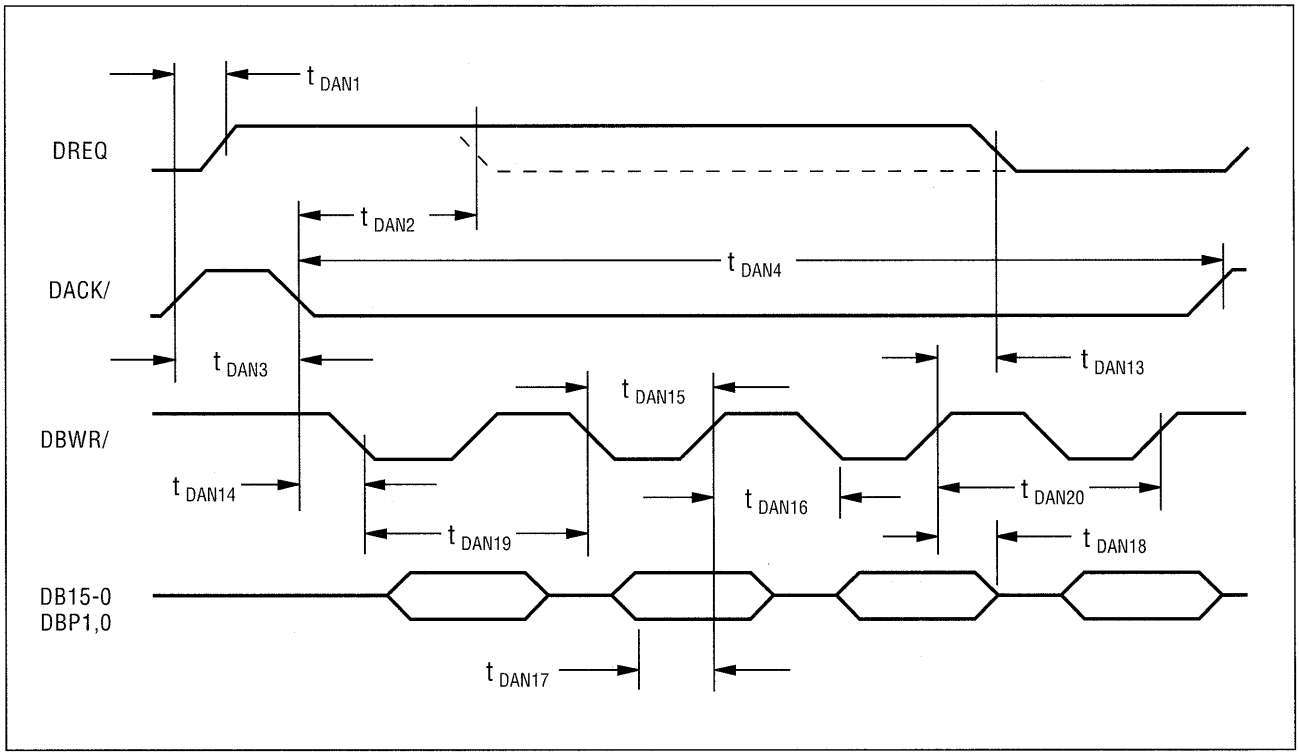
*6) DBWR/ trailing edge may precede or follow DACK/ trailing edge. The recommended value is: $t_{DBC18} \geq 0$. If DBWR/ is held past DACK/, the data setup to DACK/ high is 15 ns minimum, and $t_{DBC2}$ is 40 ns.*

*7) DREQ may stay high if the FIFO has room to accept more data during DMA write, or send more data during DMA read. If the current DMA acknowledge cycle fills the FIFO (write) or empties the FIFO (read), then DREQ will go low.*

## Burst Mode DMA Read without Byte Control




## Burst Mode DMA Write without Byte Control



## Burst Mode DMA Interface, Nonbyte Control

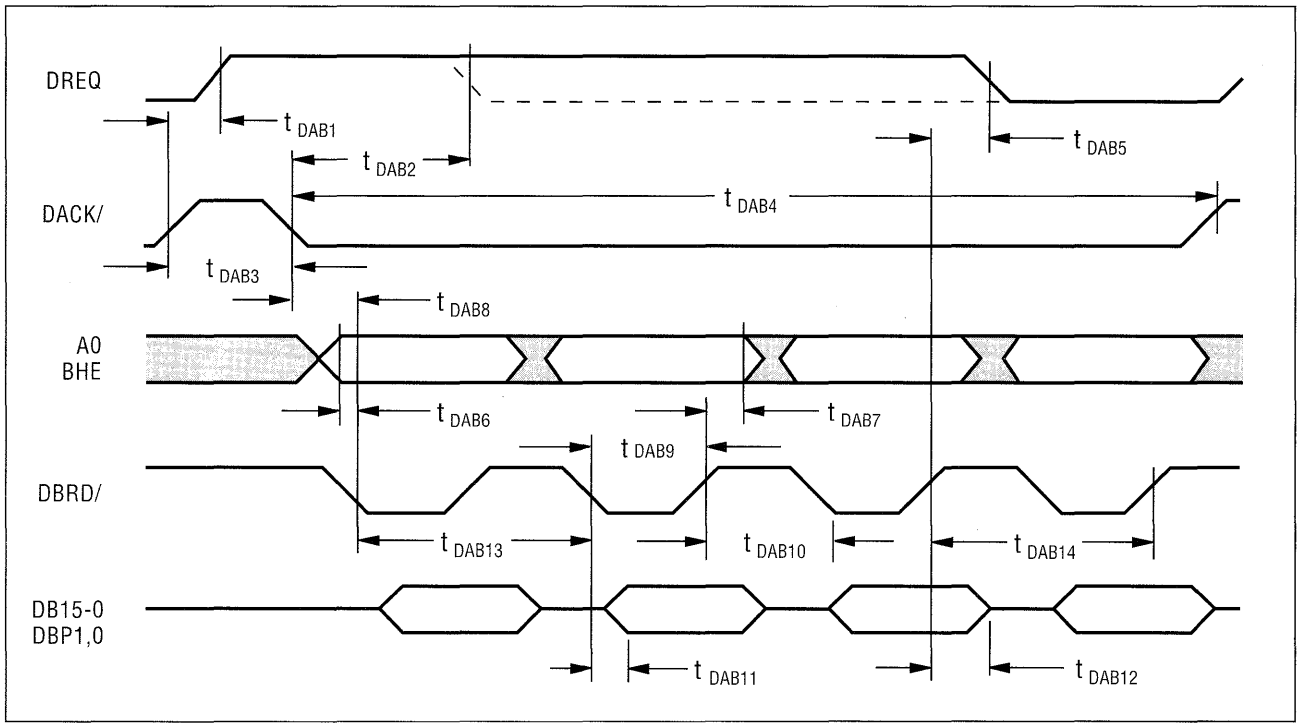
| Parameter | Symbol | Minimum | Maximum | Units | Notes |
|---|---|---|---|---|---|
| DACK/ low to DREQ high | $t_{DAN1}$ | - | 40 | ns | 3 |
| DACK/ low to DREQ low | $t_{DAN2}$ | - | 45 | ns | 1 |
| DACK/ high to DACK/ low | $t_{DAN3}$ | 60 | - | ns | - |
| DACK/ pulse width | $t_{DAN4}$ | 100 | - | ns | - |
| RD/ high to DREQ low | $t_{DAN5}$ | - | 140 | ns | 2 |
| DACK/ low to RD/ low | $t_{DAN6}$ | 0 | - | ns | - |
| RD/ pulse width | $t_{DAN7}$ | 70 | - | ns | - |
| RD/ high to RD/ low | $t_{DAN8}$ | 60 | - | ns | - |
| RD/ low to data valid | $t_{DAN9}$ | 70 | - | ns | - |
| Data release time | $t_{DAN10}$ | 2 | 50 | ns | - |
| RD/ low to RD/ low | $t_{DAN11}$ | 130 | - | ns | - |
| RD/ high to RD/ high | $t_{DAN12}$ | $t_{CS}+50$ | - | ns | - |
| DBWR/ high to DREQ low | $t_{DAN13}$ | - | 140 | ns | 2 |
| DACK/ low to DBWR/ low | $t_{DAN14}$ | 0 | - | ns | - |
| DBWR/ pulse width | $t_{DAN15}$ | 100 | - | ns | - |
| DBWR/ high to DBWR/ low | $t_{DAN16}$ | 60 | - | ns | - |
| Data setup to DBWR/ high | $t_{DAN17}$ | 15 | - | ns | - |
| Data hold from DBWR/ high | $t_{DAN18}$ | 0 | - | ns | - |
| DBWR/ low to DBWR/ low | $t_{DAN19}$ | 160 | - | ns | - |
| DBWR/ high to DBWR/ high | $t_{DAN20}$ | $t_{CS}+50$ | - | ns | - |

*1) Single DMA transfer only.*

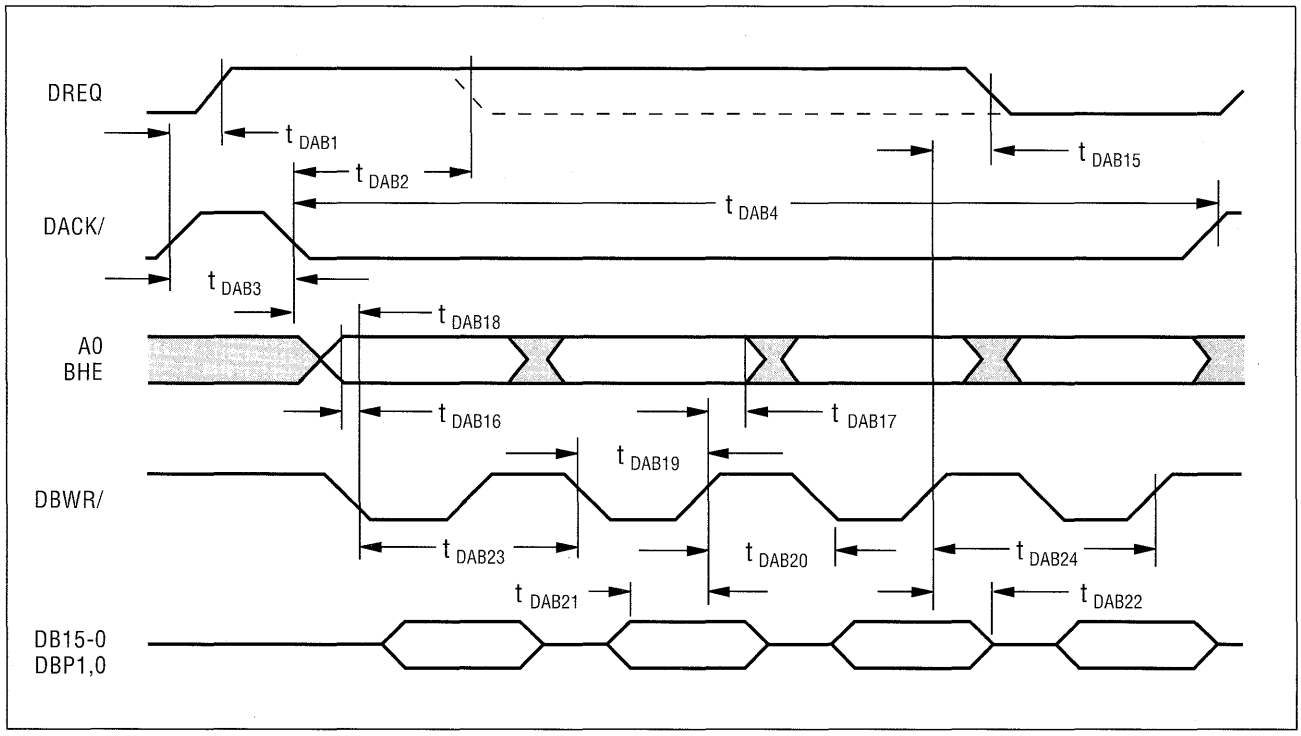
*2) Multiple DMA transfers only.*

*3) Assertion pending. If the FIFO is empty during DMA read, or full during DMA write, then assertion will not be pending..*

## Burst Mode DMA Read with Byte Control




## Burst Mode DMA Write with Byte Control



## Burst Mode DMA Interface, with Byte Control

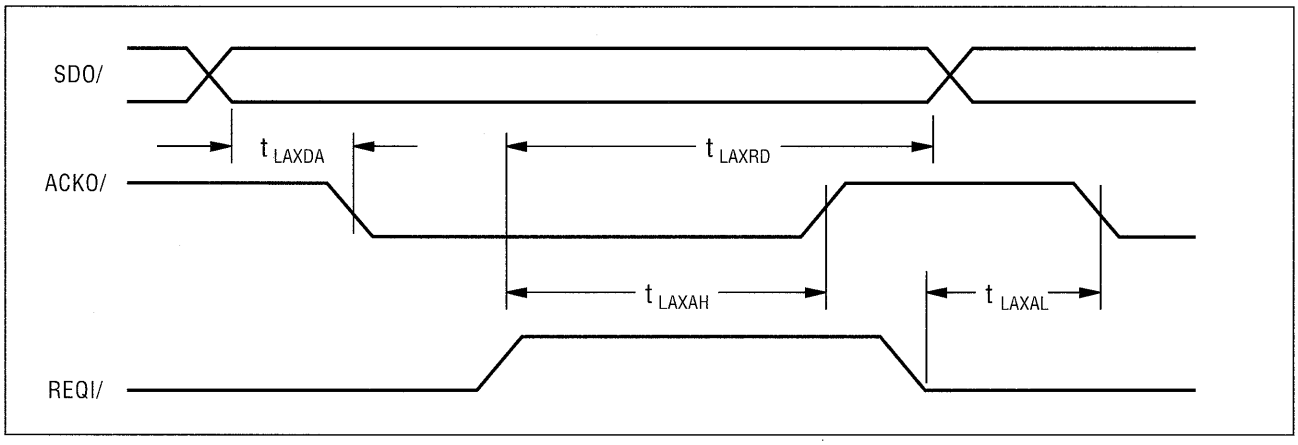
| Parameter | Symbol | Minimum | Maximum | Units | Notes |
|---|---|---|---|---|---|
| DACK/ high to DREQ high | $t_{DAB1}$ | - | 50 | ns | 3 |
| DACK/ low to DREQ low | $t_{DAB2}$ | - | 45 | ns | 1 |
| DACK/ high to DACK/ low | $t_{DAB3}$ | 60 | - | ns | - |
| DACK/ pulse width | $t_{DAB4}$ | 100 | - | ns | - |
| RD/ high to DREQ low | $t_{DAB5}$ | - | 140 | ns | 2 |
| BHE, A0 setup to DBRD/ low | $t_{DAB6}$ | 20 | - | ns | - |
| BHE, A0 hold after DBRD/ high | $t_{DAB7}$ | 20 | - | ns | - |
| DACK/ low to DBRD/ low | $t_{DAB8}$ | 0 | - | ns | - |
| DBRD/ pulse width | $t_{DAB9}$ | 70 | - | ns | - |
| DBRD/ high to RD/ low | $t_{DAB10}$ | 60 | - | ns | - |
| DBRD/ low to data valid | $t_{DAB11}$ | 70 | - | ns | - |
| Data release time | $t_{DAB12}$ | 2 | 50 | ns | - |
| DBRD/ low to DBRD/ low | $t_{DAB13}$ | 130 | - | ns | - |
| DBRD/ high to DBRD/ high | $t_{DAB14}$ | $t_{CS}$+50 | - | ns | - |
| DBWR/ high to DREQ low | $t_{DAB15}$ | - | 140 | ns | 2 |
| BHE, A0 setup to DBRD/ low | $t_{DAB16}$ | 20 | - | ns | - |
| BHE, A0 hold after DBWR high | $t_{DAB17}$ | 20 | - | ns | - |
| DACK/ low to DBWR/ low | $t_{DAB18}$ | 0 | - | ns | - |
| DBWR/ pulse width | $t_{DAB19}$ | 100 | - | ns | - |
| DBWR/ high to DBWR/ low | $t_{DAB20}$ | 60 | - | ns | - |
| Data setup to DBWR/ high | $t_{DAB21}$ | 15 | - | ns | - |
| Data hold after DBWR/ high | $t_{DAB22}$ | 0 | - | ns | - |
| DBWR/ low to DBWR/ low | $t_{DAB23}$ | 160 | - | ns | - |
| DBWR/ high to DBWR/ high | $t_{DAB24}$ | $t_{CS}$+50 | - | ns | - |

*1) Single DMA transfer only.*

*2) Multiple DMA transfers only.*

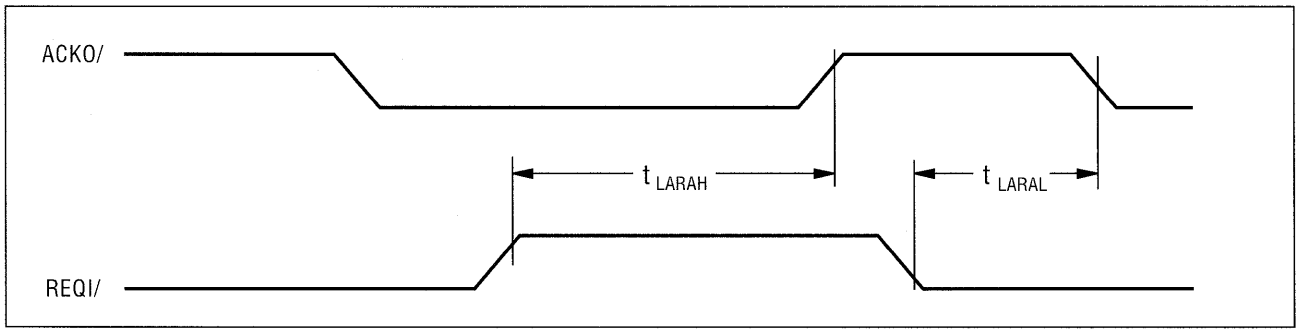
*3) Assertion pending. If the FIFO is empty during DMA read, or full during DMA write, then assertion will not be pending..*

## Initiator Asynchronous Send



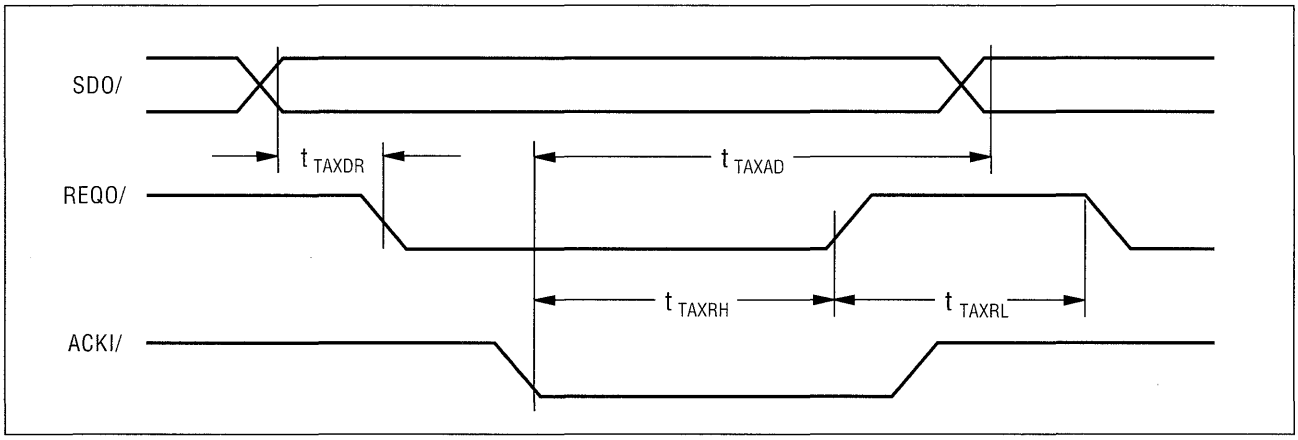

| Parameter | Symbol | Min | Typical | Max | Units |
|---|---|---|---|---|---|
| Data to ACKO/ low | $t_{LAXDA}$ | 55 | - | - | ns |
| REQI/ high to ACKO/ high | $t_{LAXAH}$ | - | 25 | 46 | ns |
| REQI/ high to data (FIFO bottom full) | $t_{LAXRD}$ | - | 40 | 80 | ns |
| REQI/ low to ACKO/ low (data already setup) | $t_{LAXAL}$ | - | 25 | 55 | ns |

## Initiator Asynchronous Receive



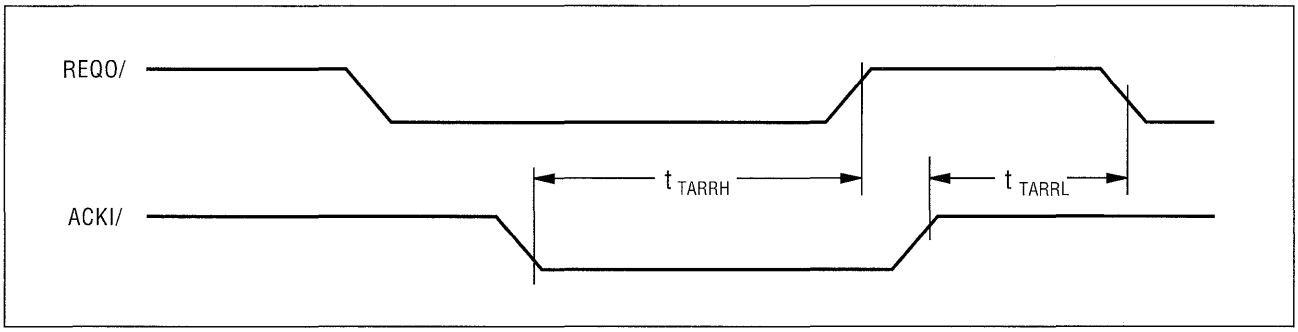

| Parameter | Symbol | Min | Typical | Max | Units |
|---|---|---|---|---|---|
| REQI/ high to ACKO/ high | $t_{LARAH}$ | - | 25 | 43 | ns |
| REQI/ low to ACKO/ low (FIFO not full) | $t_{LARAL}$ | - | 25 | 47 | ns |

## Target Asynchronous Send



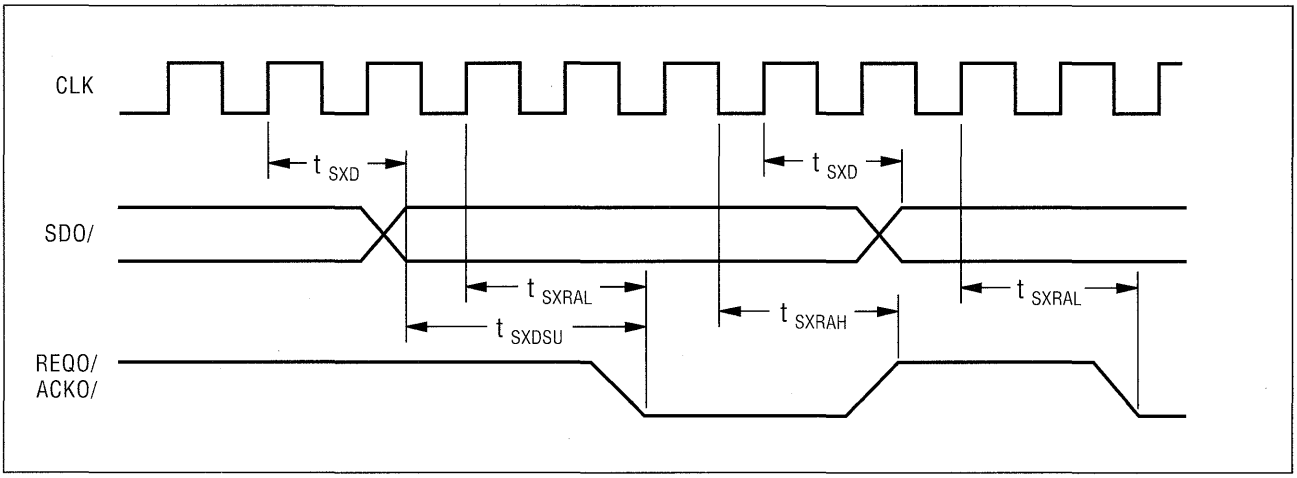

| Parameter | Symbol | Min | Typical | Max | Units |
|---|---|---|---|---|---|
| Data to REQO/ low | $t_{TAXDR}$ | 55 | - | - | ns |
| ACKI/ low to REQO/ high | $t_{TAXRH}$ | - | 25 | 43 | ns |
| ACKI/ low to data (FIFO bottom full) | $t_{TAXAD}$ | - | 40 | 78 | ns |
| ACKI/ high to REQO/ low (Data already setup) | $t_{TAXRL}$ | - | 25 | 45 | ns |

## Target Asynchronous Receive




| Parameter | Symbol | Min | Typical | Max | Units |
|---|---|---|---|---|---|
| ACKI/ low to REQO/ high | $t_{TARRH}$ | - | 25 | 43 | ns |
| ACKI/ high to REQO/ low (FIFO not full) | $t_{TARRL}$ | - | 25 | 45 | ns |

## Target and Initiator Synchronous Transmit




| Parameter | Symbol | Min | Typical | Max | Units |
|---|---|---|---|---|---|
| Data from CLK high | $t_{SXD}$ | 20 | 40 | 90 | ns |
| REQO/ or ACKO/ low from CLK high | $t_{SXRAL}$ | 15 | - | 68 | ns |
| REQO/ or ACKO/ high from CLK low | $t_{SXRAH}$ | 17 | - | 70 | ns |
| Data setup to ACKO/ or REQO/ low | $t_{SXDSU}$ | 55 | 35 | - | ns |

| # | ASC Read Registers | ASC Write Registers |
|---|---|---|
| 0 | Transfer counter low (LSB) | Transfer count low (LSB) |
| 1 | Transfer counter high (MSB) | Transfer count high (MSB) |
| 2 | FIFO | FIFO |
| 3 | Command | Command |
| 4 | Status | Destination ID |
| 5 | Interrupt | (Re) Selection timout |
| 6 | Sequence step | Synchronous period |
| 7 | FIFO flags | Synchronous offset |
| 8 | Configuration 1 | Configuration 1 |
| 9 | NCR reserved | Clock conversion factor |
| A | NCR reserved | Test mode |
| B | Configuration 2 | Configuration 2 |
| C | Configuration 3 | Configuration 3 |
| F | NCR reserved | Reserve FIFO byte |

### Status Register (Read address 04)

0 I/O
1 C/D
2 MSG
3 Xfr complete
4 Xfr count 0
5 Parity error
6 Gross error
7 Interrupt

### Interrupt Register (Read address 05)

0 Selected
1 Selected w/ATN
2 Reselected
3 Function complete
4 Bus service
5 Disconnect
6 Illegal command
7 SCSI reset detected

### Sequence Step (Read address 06)

0 Sequence step 0
1 Sequence step 1
2 Sequence step 2

### Config 1 Register (Read/write address 08)

0 Bus ID 0
1 Bus ID 1
2 Bus ID 2
3 Chip test mode enable
4 Enable parity checking
5 Parity test mode
6 SCSI reset reporting interrupt disable
7 Slow cable mode

### Test Register (Write address 0A)

0 Target mode
1 Initiator mode
2 Tri-state mode

### Config 2 Register (Read/write address 0B)

0 DMA parity enable
1 Register parity enable
2 Target bad parity abort
3 SCSI 2
4 DREQ hi-Z
5 Enable byte control
6 Enable phase latch
7 Reserve FIFO byte

### Config 3 (Read/write address 0C)

0 Threshold 8
1 Alternate DMA mode
2 Save residual byte

### Write Register 0F

Reserve FIFO byte
(Refer to *Config 2 Bit7*)

### Miscellaneous Commands

00 80 No Operation (NOP)
01 81 Flush FIFO
02 82 Reset chip
03 83 Reset SCSI bus

### Disconnected Commands

40 C0 Reconnect sequence
41 C1 Select without ATN sequence
42 C2 Select with ATN sequence
43 C3 Sel with ATN and stop sequence
44 C4 Enable selection/reselection
45 C5 Disable selection/reselection
46 C6 Select ATN 3

### Initiator Commands

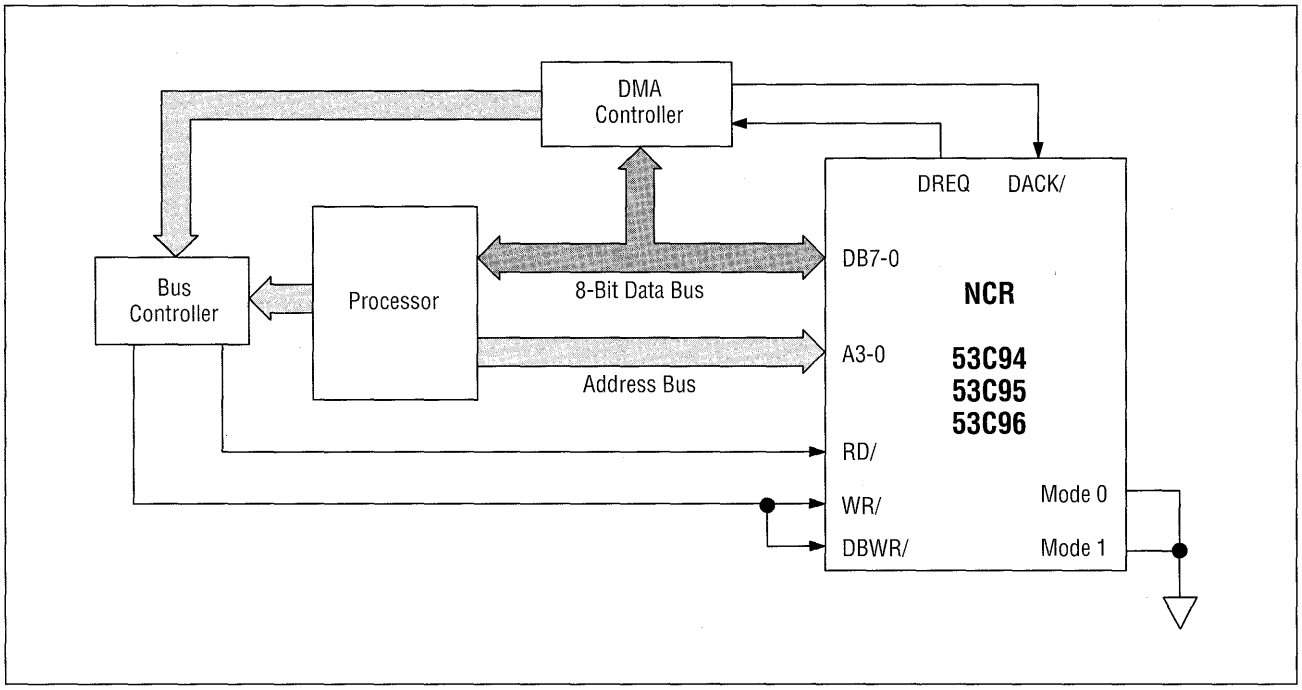
10 90 Transfer information
11 91 Initiator command complete sequence
12 92 Message accepted
18 98 Transfer pad
1A 9A Set ATN (Attention)

### Target Commands

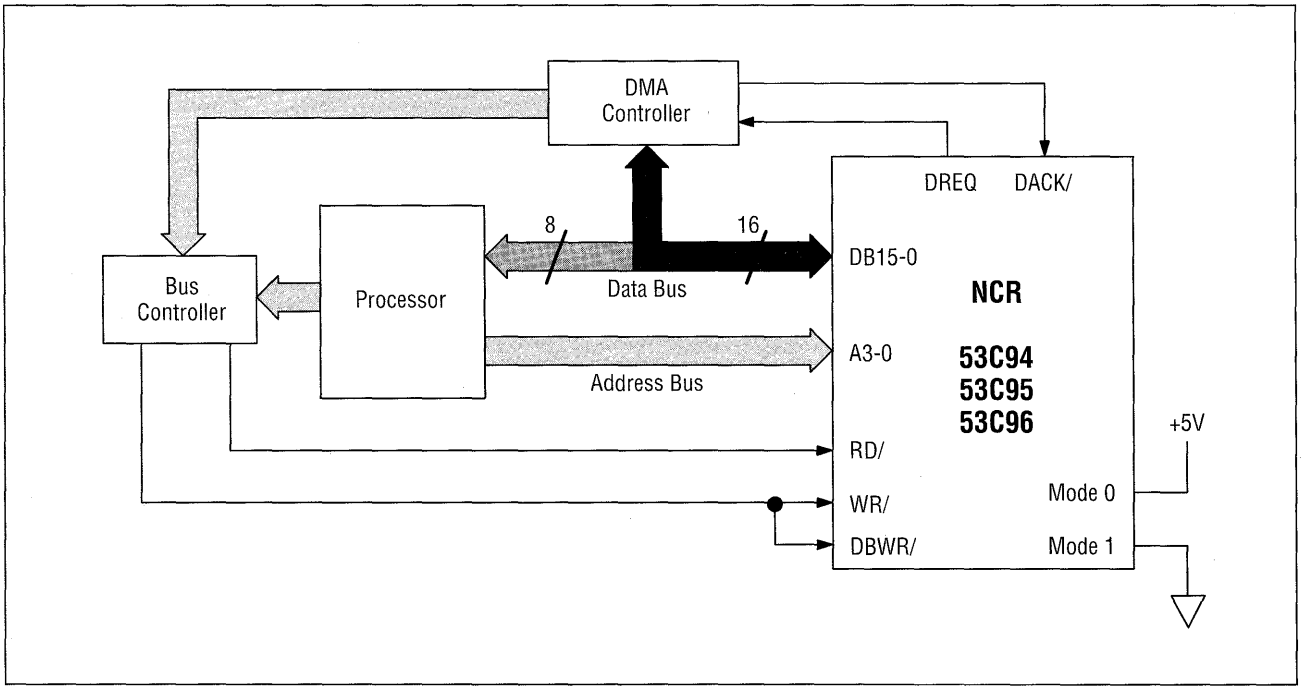
20 A0 Send message
21 A1 Send status
22 A2 Send data
23 A4 Disconnect sequence
24 A5 Terminate sequence
25 A6 Target command complete sequence
27 A7 Disconnect
28 A8 Receive message
29 A9 Receive command sequence
2A AA Receive data
2B AB Receive command sequence
06 86 Target abort DMA

# Appendix A – Bus Configurations

## Mode Zero

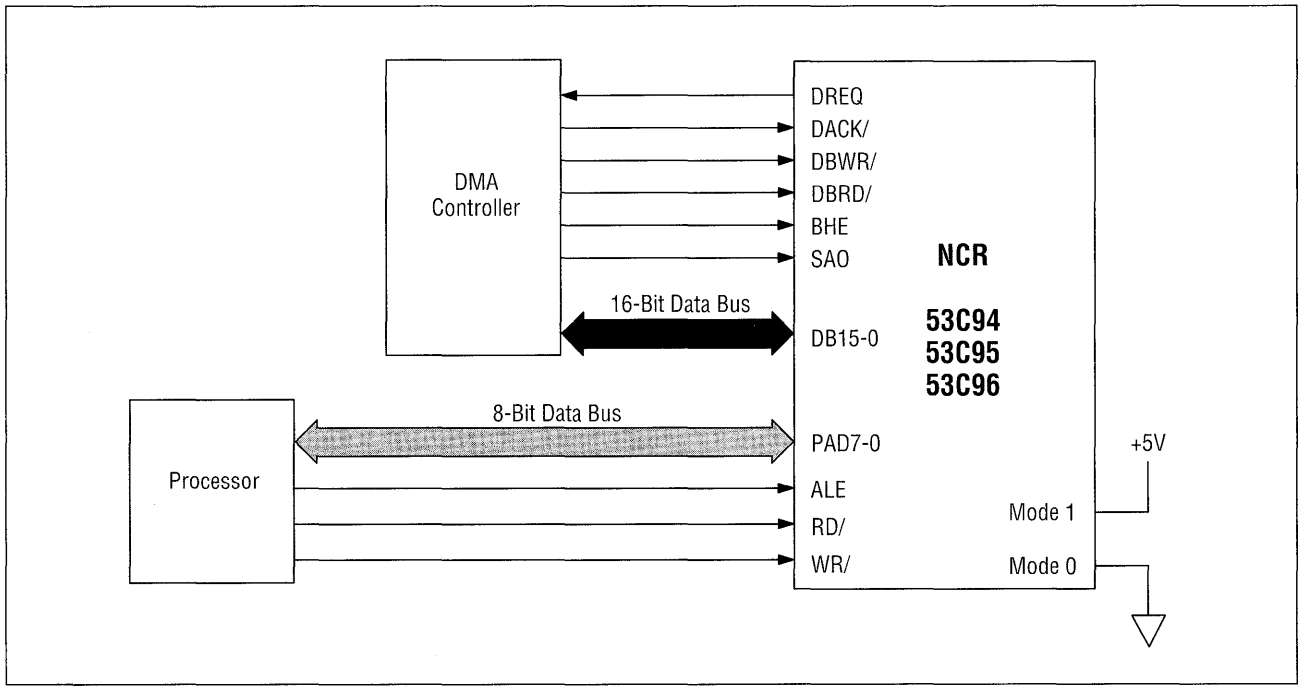



## Mode One

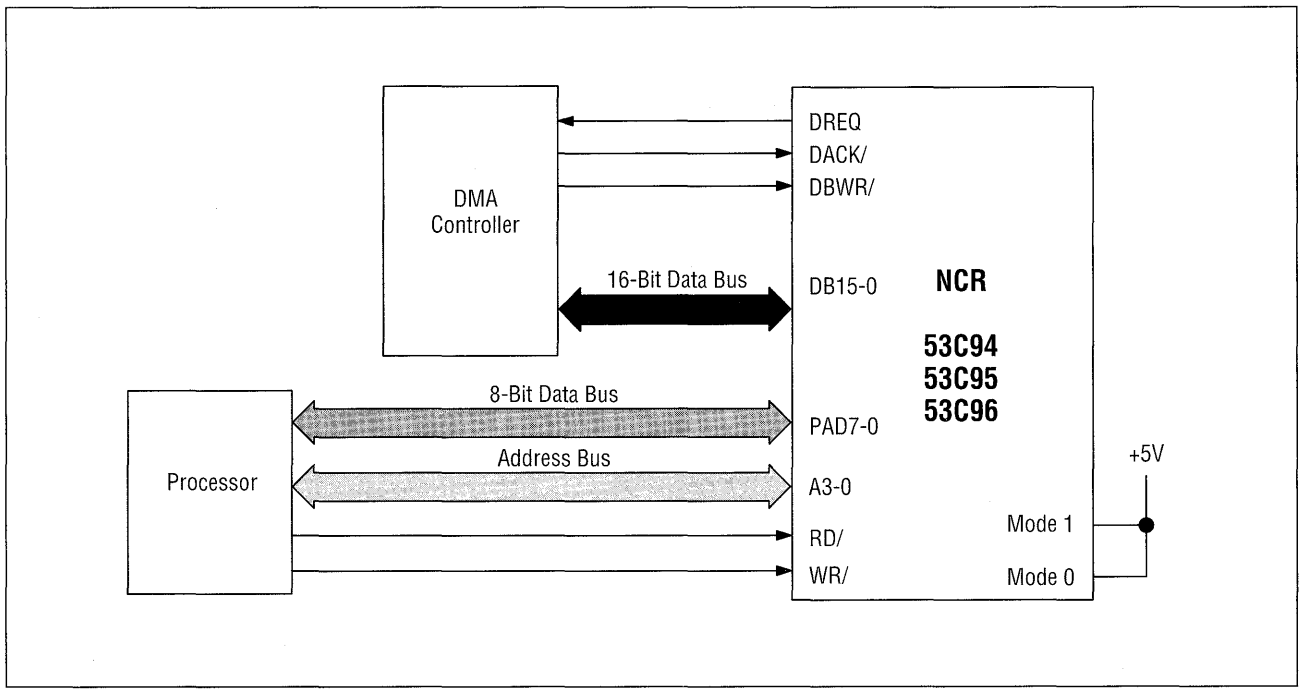


# Appendix A – Bus Configurations (Cont'd)

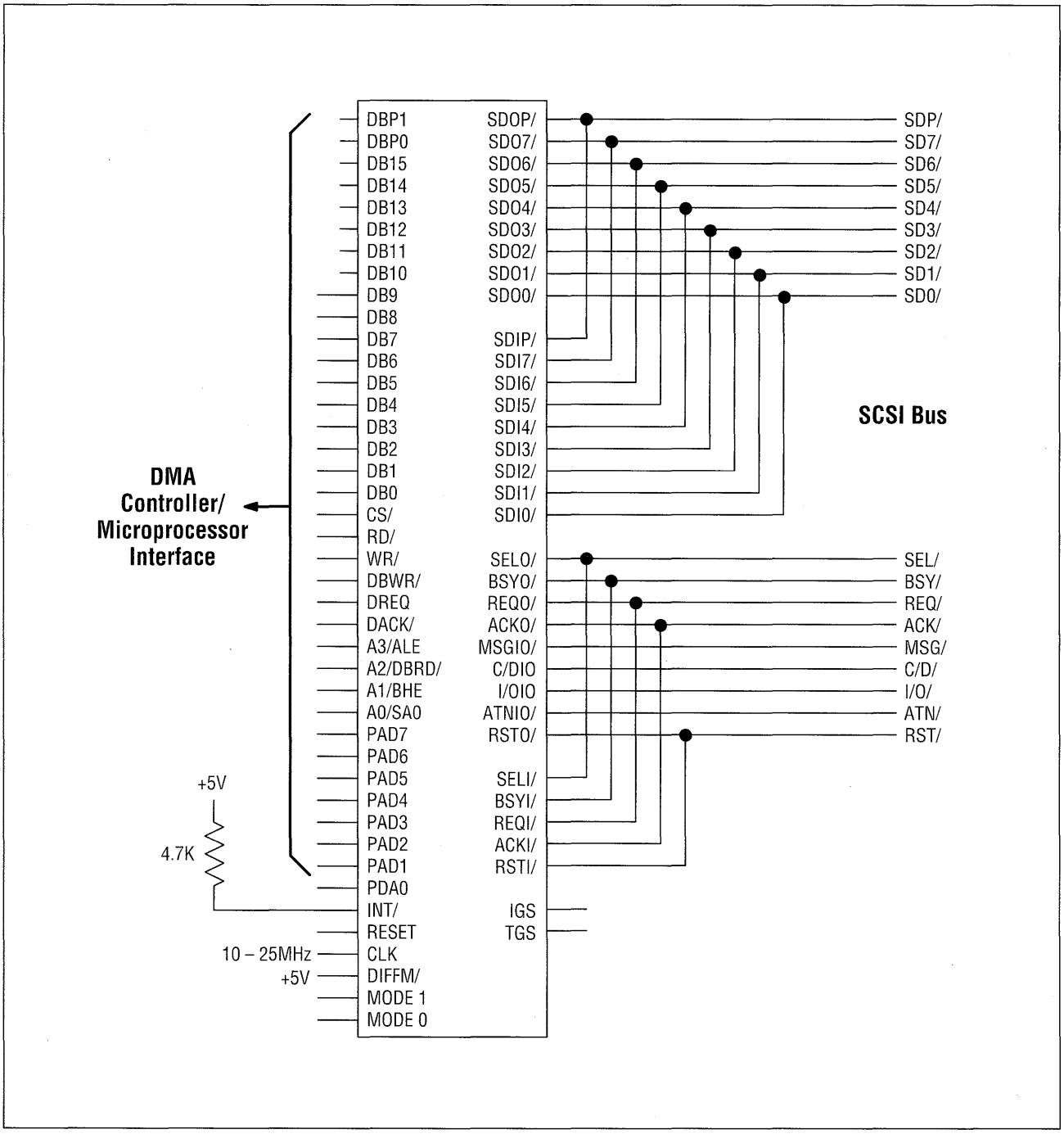
## Mode Two



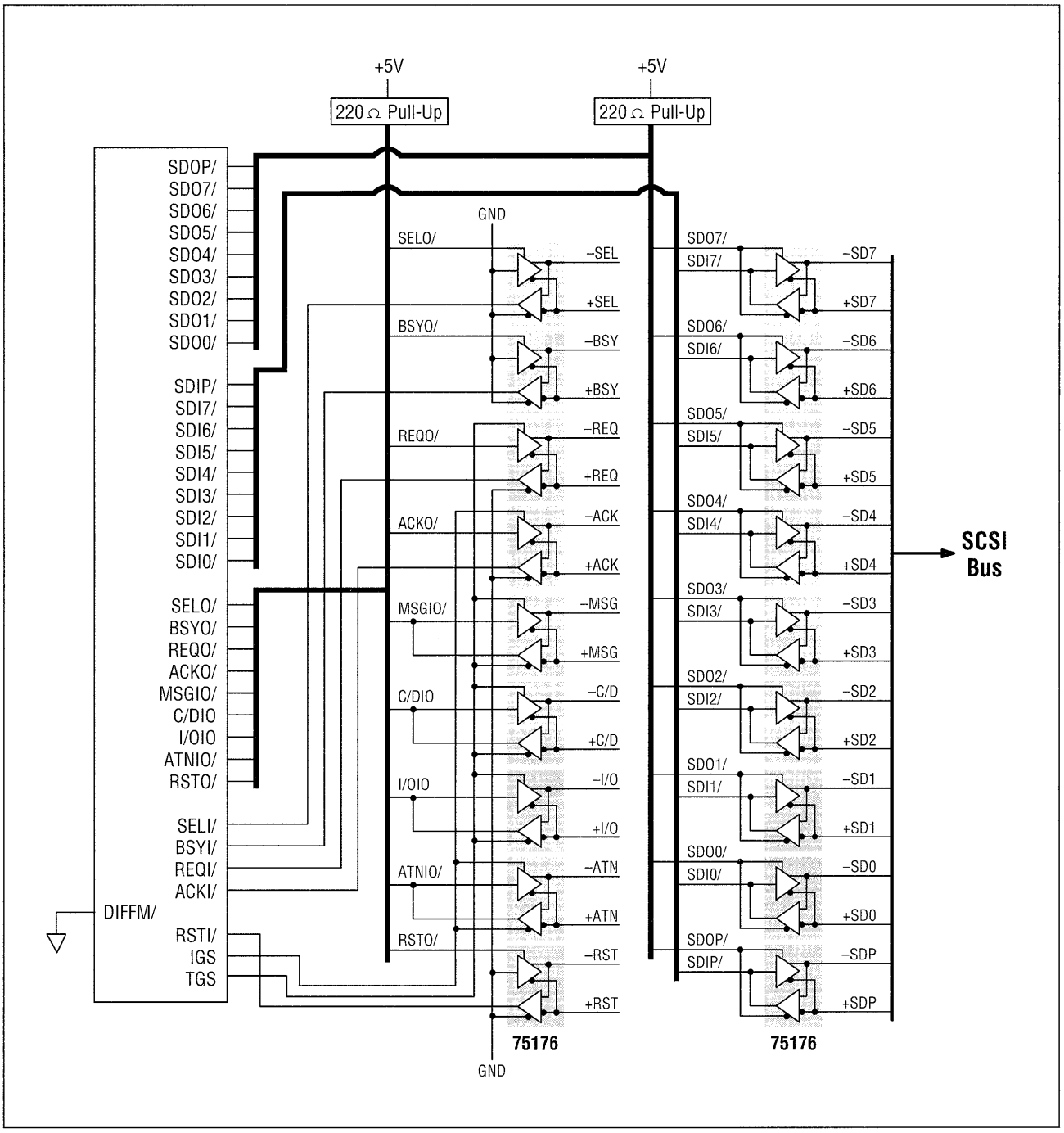

## Mode Three



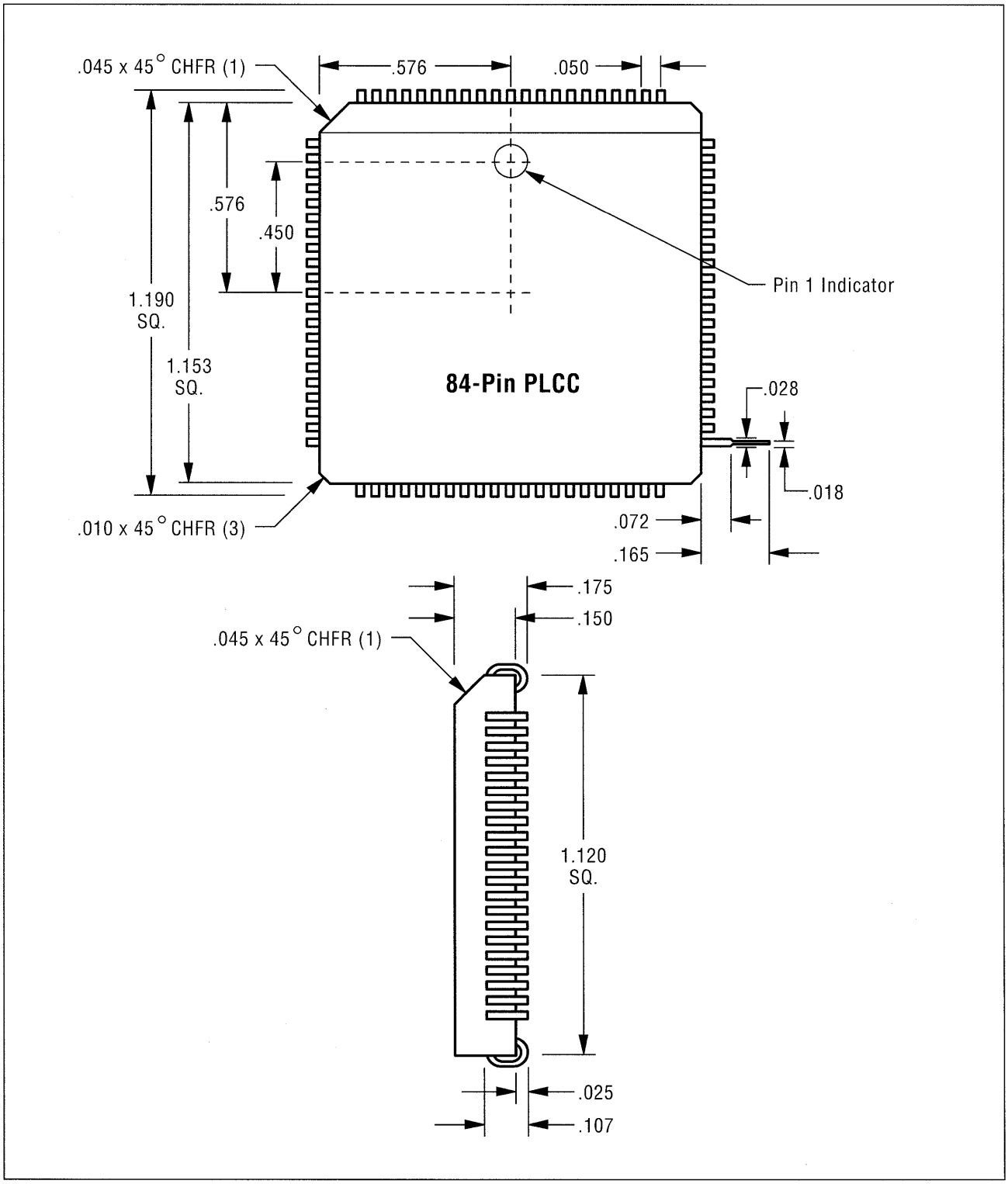
## Appendix B – 53C94 and 53C96 Single-Ended SCSI Bus Interface



## Appendix B – 53C95 and 53C96 Differential SCSI Bus Interface

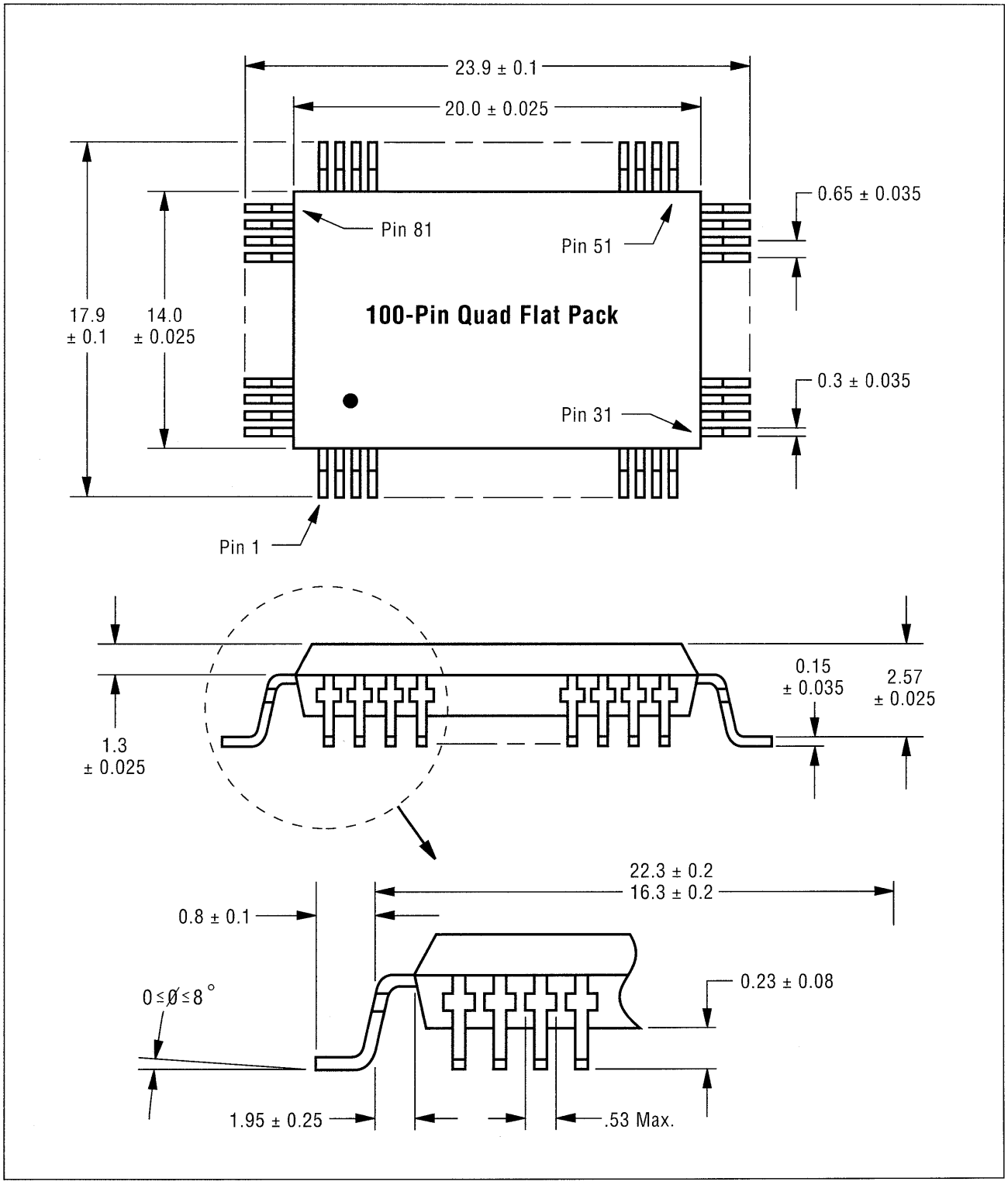


## Appendix C – Mechanical Drawings




***Note:*** *All dimensions are in inches*

## Appendix C – Mechanical Drawings (Cont'd)




***Note:*** *All dimensions are in inches*

# NCR Microelectronic Products Division – Sales Locations

**For literature on any NCR Microelectronics product or service call the NCR hotline toll-free:**

**1-800-334-5454**

**NCR Microelectronic Products Division**
**Worldwide Sales Headquarters**
3130 De La Cruz Boulevard, Suite 209
Santa Clara, CA 95054
(408) 980-6200

## Division Plants

**NCR Microelectronic Products Division**
2001 Danfield Court
Fort Collins, CO 80525
(303) 226-9500

*Commercial ASIC Products*
*Customer Owned Tooling*
*Communications Products*
*Memory Products*
*Software Products*

**NCR Microelectronic Products Division**
1635 Aeroplaza Drive
Colorado Springs, CO 80916
(719) 596-5611
1-800-525-2252

*High Reliability ASIC*
*Military Products*
*Automotive Products*
*Logic Products*
*SCSI Products*
*Internal ASIC*

## North American Sales Offices

**Northwest Sales**
3130 De La Cruz Boulevard, Suite 209
Santa Clara, CA 95054
(408) 727-6575

**Southwest Sales**
3300 Irvine Avenue, Suite 255
Newport Beach, CA 92660
(714) 474-7095

**North Central Sales**
8000 Townline Avenue, Suite 209
Bloomington, MN 55438
(612) 941-7075

**South Central Sales**
400 Chisholm Place, Suite 100
Plano, TX 75075
(214) 578-9113

**Northeast Sales**
500 West Cummings Parkway, Suite 4000
Woburn, MA 01801
(617) 933-0778

**Southeast Sales**
1051 Cambridge Square, Suite C
Alpharetta, GA 30201
(404) 740-9151

## International Sales Offices

**European Sales Headquarters**
Gustav-Heinemann-Ring 133
8000 Munchen 83
West Germany
49 89 632202

**Asia/Pacific Sales Headquarters**
2501 Vicwood Plaza
199 Des Voeux Road
Central Hong Kong
852 859 6888



SCSIP-53C94/95/96 0290