adaptec, inc.

AIC-6250

High-Performance SCSI Protocol Chip

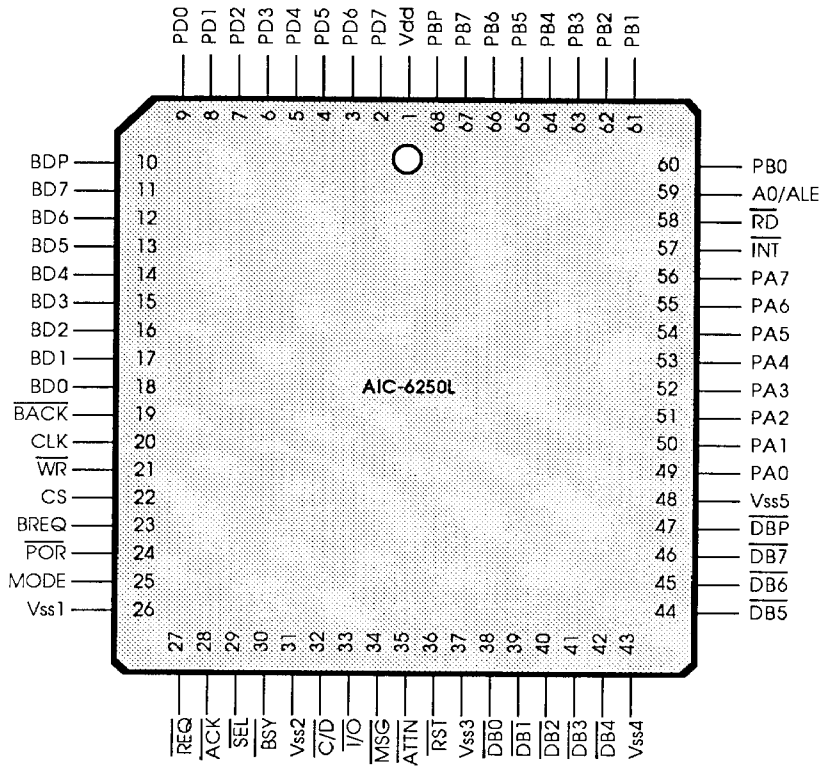
PRELIMINARY

SCSI Interface

- Asynchronous Transfers Up To 3 Mbytes/Second
- Synchronous Transfers Up To 5 Mbytes/Second With An 8-Byte Offset
- Both Initiator And Target Roles Supported
- Automatic Arbitration, Selection, And Reselection
- Automatic Response To Selection And Reselection
- Parity Generation And Checking
- Integrated 48 mA Drivers And Receivers
- Supports SCSI Bus Differential Option With External Drivers And Receivers
- 24-Bit DMA Byte Counter

Microprocessor And Buffer Interface

- Up To 20 Mbyte/Second Memory Data Transfer Rate (100 ns Buffer Cycle Time)
- 8-Bit And 16-Bit Buffer Data Bus
- Parity Generation And Checking
- All Interrupts Maskable
- Separate Buffer And Microprocessor Data Bus
- Multiplexed And Nonmultiplexed Microprocessor Addressing Modes
- Direct Access To SCSI Data From Microprocessor



CMOS 68-PIN PLCC

CONTENTS

OVERVIEW

The Small Computer Systems Interface (SCSI) defines an I/O channel (bus) used for interfacing host computers (via host adapters), controller boards and peripheral devices with embedded controllers. The SCSI specification defines electrical characteristics, mechanical specifications and the software protocol for devices attaching to the SCSI bus. For a typical SCSI bus configuration, see Figure 1.

The following is a list of key features provided in the SCSI specification:

- Up to eight devices on the bus (Each device may be an initiator or target or both)
- Software protocol (command structures and message system)
- Low-cost single-ended drivers and receivers (allowing cable lengths up to six meters)
  - — Optional differential drivers and receivers (allowing cable lengths up to 25 meters)
- A two-wire asynchronous handshake protocol
  - — An optional synchronous transfer protocol with a variable offset
- Odd-data parity

For more information on the SCSI architecture, refer to the ANSI standard X3.131–1986.

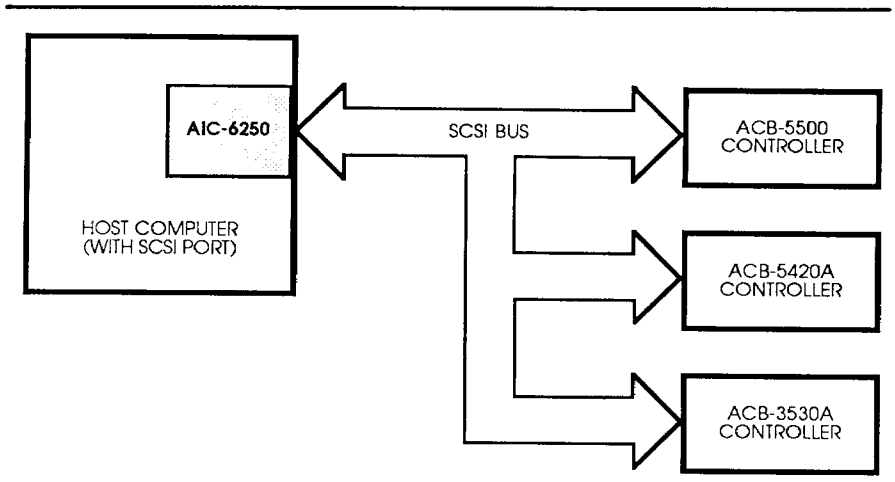


FIGURE 1. TYPICAL SCSI BUS CONFIGURATION

INTRODUCTION

The Adaptec AIC-6250 High-Performance SCSI Protocol Chip provides all the functions necessary to implement a high-performance SCSI interface on a host computer or a peripheral device. The AIC-6250 integrates an eight-byte internal FIFO, SCSI bus control, two general purpose ports and memory interface control, into one 68-pin PLCC package. The AIC-6250 also provides application flexibility by being designed to work with either the host microprocessor or a local support microprocessor.

The AIC-6250 provides 48 mA open collector drivers for the single-ended SCSI bus interface. For the differential SCSI bus, all the control signals required to control external drivers and receivers are provided.

The AIC-6250 supports both the SCSI bus initiator and target roles. The SCSI bus Initiator is the device which initiates an operation on the SCSI bus. The operation is initiated by:

1. Waiting for a bus free condition.
2. Arbitrating for control of the bus.
3. Selecting the device from which the activity is requested.
4. Sending the command descriptor block, defining the activity which is to occur.

During this sequence of events, the initiator has control of the bus only during the Arbitration and Selection phases. Once Selection is complete, the target device takes control of the bus by driving the signals which establish the appropriate phase, and initiating the information transfer through asserting and deasserting the REQ signal.

By supporting both the SCSI bus Initiator and Target roles, the AIC-6250 may be easily used in any SCSI bus configuration (see Figure 1 for a typical SCSI bus configuration). The register set of the AIC-6250 has been implemented to allow the same software subroutines to manage both initiator and target roles. This adds to the flexibility of the part by making such operations as host-to-host or device-initiated operations (copy command), easy to implement.

The AIC-6250 also supports both the asynchronous and synchronous methods of information transfer on the SCSI bus. Asynchronous transfer is a two-wire handshake protocol where the target device drives the SCSI $\overline{REQ}$ signal, and the initiator device drives the SCSI $\overline{ACK}$ signal. Synchronous transfer is similar to the asynchronous transfer, but the $\overline{REQ}$ and $\overline{ACK}$ signals may be offset by an offset count up to the size of the internal FIFO (a performance advantage is realized with a FIFO up to eight bytes). The AIC-6250 has an eight-byte FIFO, allowing a $\overline{REQ}$ to $\overline{ACK}$ offset of up to eight bytes during synchronous transfer. This offset, combined with the high speed of the AIC-6250, provides a synchronous transfer rate of up to 5 Mbytes per second. For asynchronous transfer, the AIC-6250 provides high-speed data transfer with SCSI bus data transfer rates up to 3 Mbytes per second. To increase the overall system performance, the eight-byte FIFO in the AIC-6250 is used as a speed matching buffer between the SCSI bus and the system memory when the AIC-6250 is being used in the asynchronous mode. This increases the system performance by making the SCSI bus data transfer rate independent of the host data transfer rate; thus, allowing data to be buffered in the AIC-6250 and to burst into system memory when access to system memory is given.

The maximum system performance is a function of both the speed of data transfer on the SCSI bus as well as the speed of the memory DMA Transfer. The AIC-6250 has broken the bottleneck for the memory DMA Transfer by providing a maximum DMA Transfer speed, with a 16-bit bus interface, of 20 Mbytes per second.

For lower bandwidth systems, the eight-bit only bus may be implemented. The upper eight bits are configured as a general purpose I/O port. In the eight-bit memory bus mode, the AIC-6250 has a maximum DMA speed of 10 Mbytes per second.

See Figure 2 for a block diagram of a typical host implementation using a 16-bit memory bus. See Figure 3 for a typical controller design block diagram using a local support microprocessor.

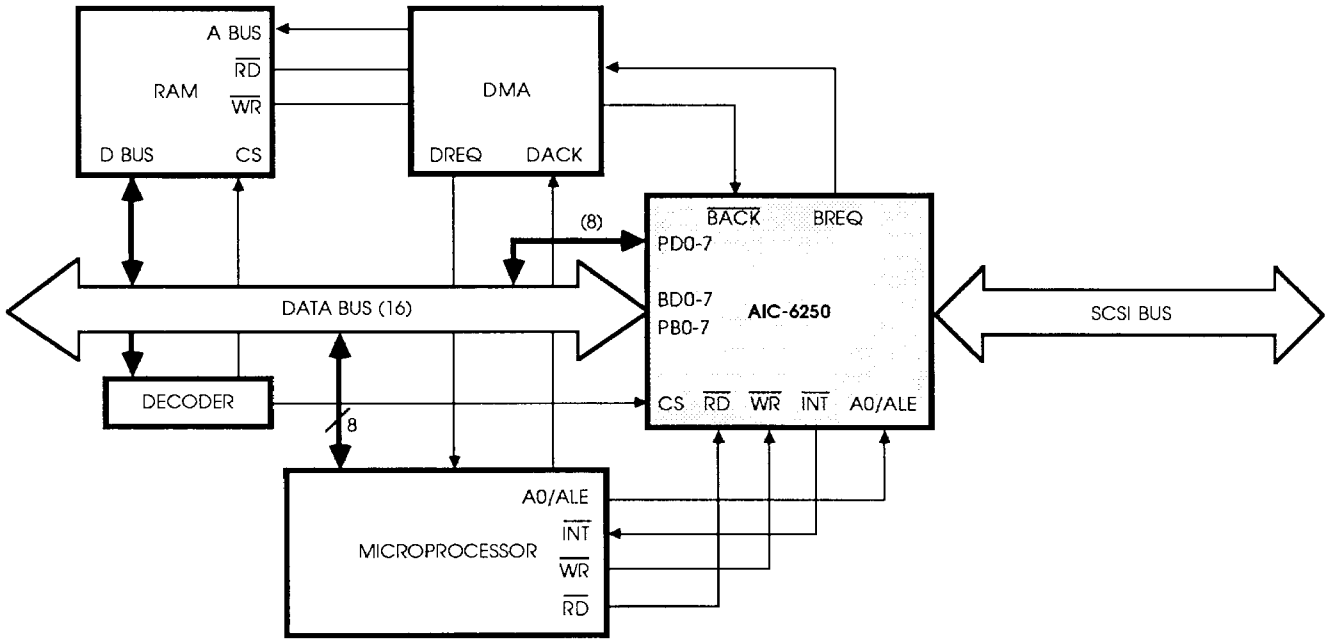


FIGURE 2. BLOCK DIAGRAM OF A TYPICAL HOST IMPLEMENTATION USING A 16-BIT MEMORY BUS

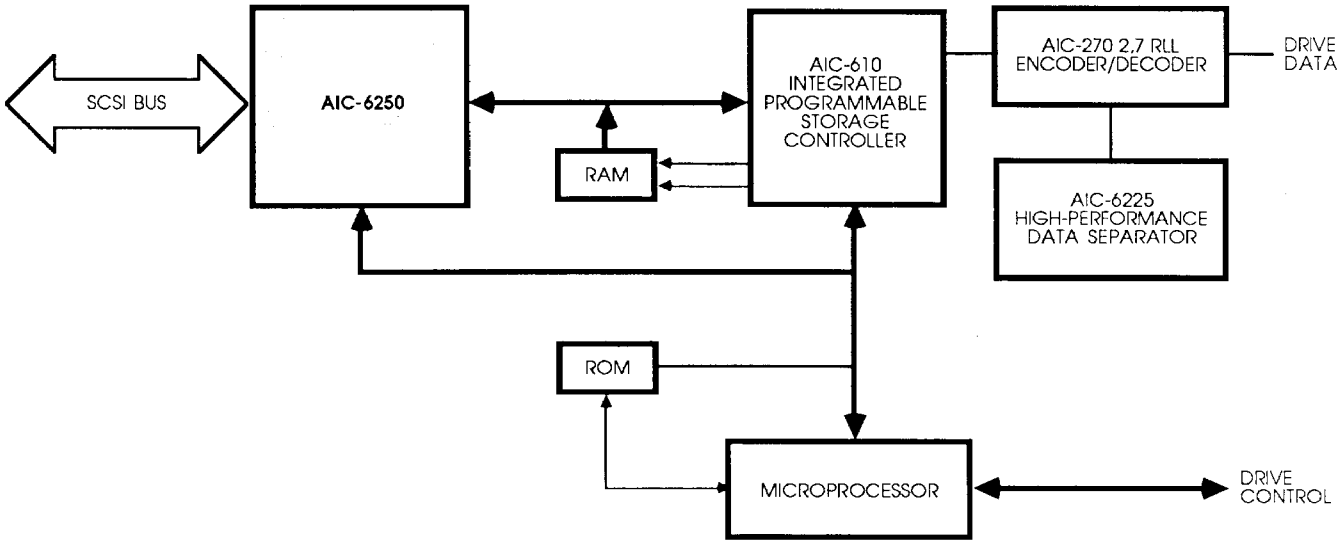


FIGURE 3. BLOCK DIAGRAM OF A TYPICAL SCSI DRIVE CONTROLLER DESIGN USING A LOCAL MICROPROCESSOR

FUNCTIONAL DESCRIPTION

Internal to the AIC-6250 are seven main functional blocks:

1. Microprocessor interface and Input/Output control
2. SCSI bus interface control and handshake
3. Parity logic
4. SCSI Select and Reselect control
5. Memory interface control and handshake
6. General purpose Input/Output ports
7. Eight-byte FIFO

Figure 4 shows the major functional blocks of the AIC-6250.

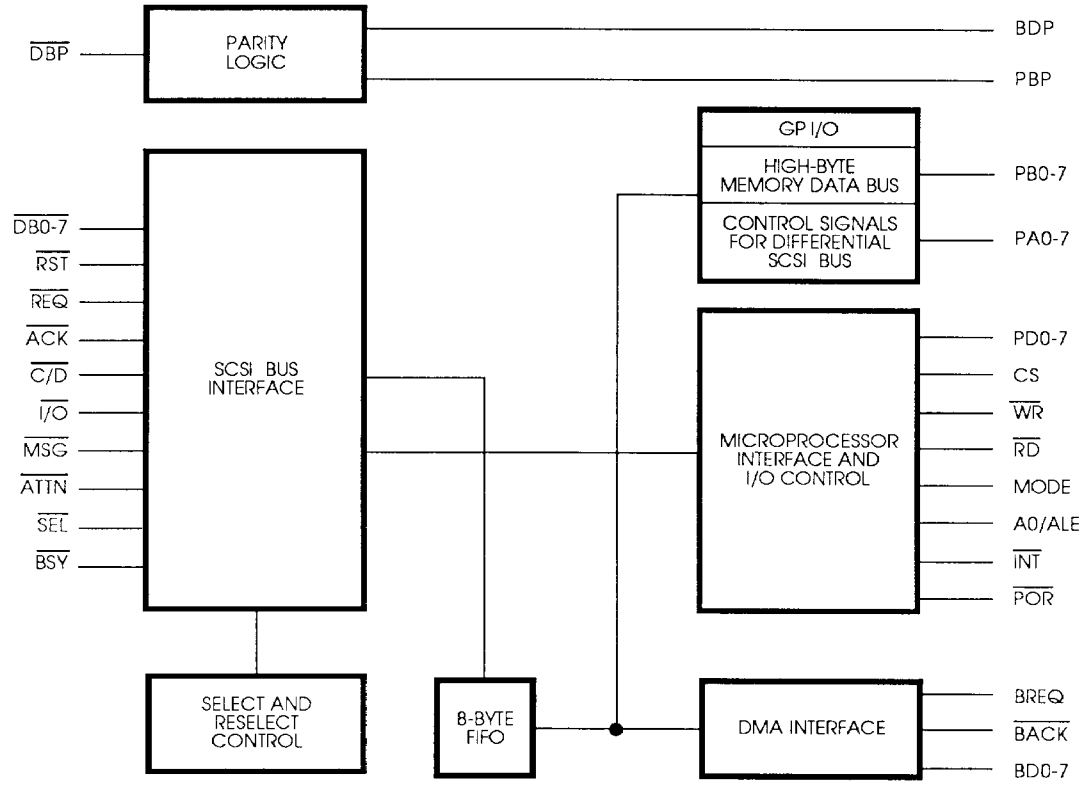


FIGURE 4. FUNCTIONAL BLOCK DIAGRAM

Microprocessor Interface And Input/Output Control

The microprocessor interface will support either an eight-bit multiplexed bus, such as that found on the Intel family of processors, or a non-multiplexed bus, such as that found in the Motorola family of processors. The mode of operation (which type of microprocessor interface is used) is selected through the MODE signal (pin 25) of the chip. See Figure 5 for an example of a multiplexed address/data bus and Figure 6 for an example of a nonmultiplexed address and data bus design using the AIC-6250.

The AIC-6250 decodes addresses from 00_H to $0F_H$. In order to prevent erroneous operation, no other addresses are decoded.

The AIC-6250 also allows the microprocessor access to SCSI data or memory data through its internal registers without any external hardware.

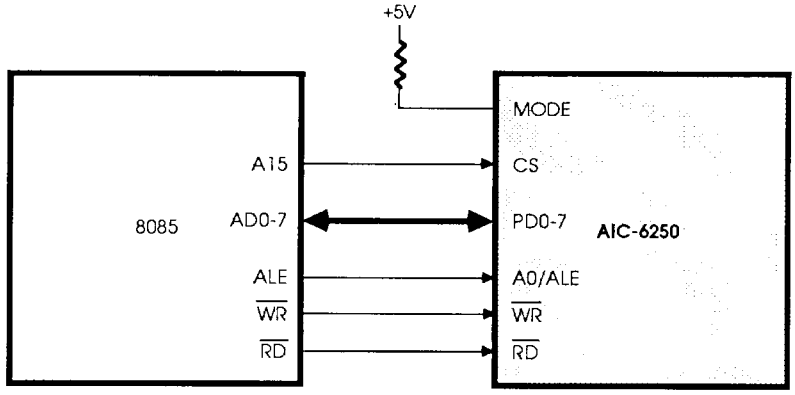


FIGURE 5. MULTIPLEXED ADDRESS/DATA BUS MODE

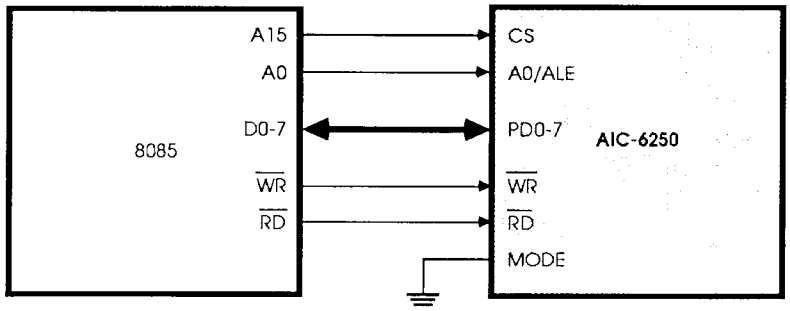


FIGURE 6. NONMULTIPLEXED (SEPARATE) ADDRESS/DATA BUS MODE

SCSI Bus Interface Control And Handshake

The SCSI bus interface control block includes the 48 mA drivers and receivers for the single-ended SCSI bus option. This eliminates the need for external drivers and receivers. The SCSI bus interface block also provides the logic to implement the target or initiator modes of SCSI bus operation. Figure 7 shows the AIC-6250 connections for the single-ended SCSI bus.

The SCSI bus interface control and handshake block also implements both asynchronous and synchronous transfer, as well as three different data transfer methods. The available data transfer methods are as follows:

1. DMA Transfer,
2. Automatic SCSI bus handshake (Automatic PIO), or,
3. Microprocessor controlled SCSI bus PIO handshake (Manual PIO).

DMA Transfer: DMA Transfer is a "hands off" method of transferring information (commands, data or messages). When the DMA Transfer method is used, both the SCSI bus and the memory handshake are performed automatically by the AIC-6250. The memory interface handshake (BREQ/BACK) is asynchronous and transfers a byte/word of data in the 8-bit/16-bit mode, respectively, per handshake. DMA transfer is the only method of data transfer which may be used with synchronous transfer. For maximum SCSI bus transfer rates in the synchronous transfer mode, the FIFO allows up to an eight-byte offset, except for initiator inbound data transfer, where the maximum offset is seven. In the asynchronous transfer mode, DMA transfer uses the FIFO as a speed matching buffer between the host bus and the SCSI bus to increase the overall system performance.

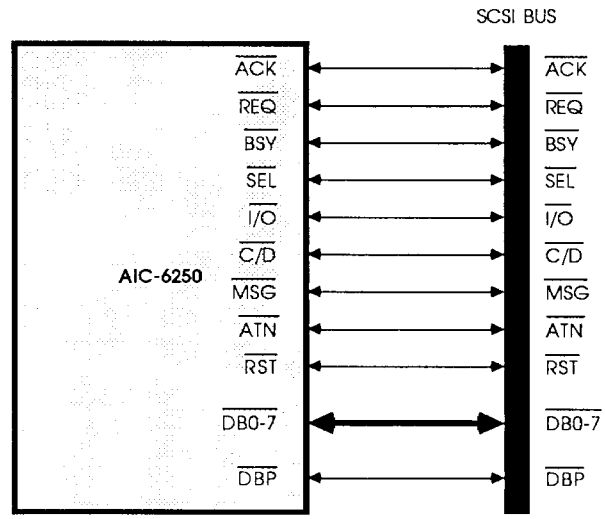


FIGURE 7. SINGLE-ENDED SCSI BUS INTERFACE OPTION

Included in the AIC-6250 is a 24-bit DMA Byte Counter, allowing data transfers of up to 16 Mbytes in length. To ease the handling of disconnection/reconnection, or error conditions, the DMA Counter is decremented for each byte that is transferred across the SCSI bus. This is independent of the number of bytes which may be in the FIFO, or the offset count, when using synchronous transfer.

Automatic PIO: Automatic PIO requires the microprocessor to intervene once each byte time. After completion of the information transfer, an interrupt will be generated, or the microprocessor can mask out the microprocessor interrupt and poll for the completion of transfer. This method of information transfer requires the microprocessor to also read/write the data to/from the SCSI Data Register. Automatic PIO may not be used with synchronous transfer and does not utilize the FIFO in the asynchronous mode.

Microprocessor Controlled PIO: Microprocessor controlled PIO gives the microprocessor complete control over the SCSI REQ or ACK signal (depending on whether the AIC-6250 is supporting the SCSIinitiator or target role). This method of data transfer requires the microprocessor to also read/write the data to/from the SCSI Data Register. Microprocessor controlled PIO may not be used with synchronous transfer and does not utilize the FIFO in the asynchronous mode.

When in the initiator mode, all SCSI bus transfers and transfer methods may be set up in advance of the transfer, with the exception of the microprocessor controlled PIO handshake. In addition, by predicting the next phase (using the expected phase function by setting the expected phase in the SCSI Signal Register), the AIC-6250 improves system performance by reducing the software overhead time to respond to a phase. This feature saves the microprocessor from waiting for a phase to be valid with the $\overline{REQ}$ signal set, then proceeding with the information transfer. This allows the host to immediately begin information transfer once the SCSI bus phase matches the expected phase and the $\overline{REQ}$ signal is asserted.

Parity Logic

The parity logic generates and optionally checks SCSI bus odd parity during information transfer on the SCSI bus, as well as automatically generating and optionally checking odd oarity on memory data. While parity generation and checking is automatically performed, the generation of an interrupt in the event of a parity error is optional.

When parity checking is enabled and a parity error is detected on a memory transfer, the AIC-6250 will halt the transfer immediately and interrupt the microprocessor. The microprocessor must then perform the appropriate error recovery action. In order to continue the transfer, parity error should be reset by disabling the En Parity bit. When the 16-bit memory transfer mode is being used, the AIC-6250 will generate and check parity on both the high and the low bytes of data with one parity bit for each byte.When parity checking is enabled and a parity error is detected on a SCSI bus transfer, the transfer will continue until transfer count equals zero or the transfer is terminated by the microprocessor.

The AIC-6250 generates parity on the SCSI bus also; however, flow through parity is not supported.

SCSI Select And Reselect Control

The SCSIselect and reselect control provides the logic to perform the SCSI bus arbitration scheme, implemented by most SCSI systems, as well as automatic response to selection and reselection.

The AIC-6250 will automatically look for the SCSI BUS FREE phase. When the BUS FREE phase is detected and enable selection is active, the AIC-6250 will arbitrate for control of the SCSI bus. If arbitration is won, the AIC-6250 will proceed to the SELECTION or RESELECTION phase.

The SCSI selection timeout may be accomplished by starting a timer from an interrupt which occurs when arbitration has been won and the AIC-6250 asserts the SCSI $\overline{SEL}$ signal.

The completion of the automatic arbitration and selection/reselection process is indicated by an interrupt.

Memory Interface Control And Handshake

The AIC-6250 will work well with a DMA control device to improve overall system performance by transferring data to or from the system bus at very high data rates. The DMA interface of the AIC-6250 is capable of achieving data transfer speeds of up to 20 Mbytes per second when used in the 16-bit mode (a two-byte transfer every 100 nanoseconds), as shown in Figure 9.

By interfacing the eight-byte FIFO directly with the DMA interface, the FIFO is used as a speed matching buffer when the chip is used in the asynchronous mode. This allows the data transfer rates of the system bus and the SCSI bus to be loosely coupled.

Memory data transfer may occur through one of two ways:

1. *Automatically when in the DMA transfer mode:* When transferred through the DMA mode, the data comes from the SCSI bus, through the FIFO and into memory (although the AIC-6250 does not generate any addresses for the data to be transferred to). This method of information transfer does not require any microprocessor intervention during the data transfer.

2. *The microprocessor must read/write the memory when transferring data using Automatic PIO:* When the microprocessor uses the AIC-6250's ability to read/write the memory, the microprocessor must:

- Confirm no DMA activity is occurring
- Set up HBV and LBV signals in Port A for 16-bit memory data (not required for 8-bit).
- Set transfer direction and read/write the information to/from the appropriate register
- Request a memory cycle.

Using the capability of the AIC-6250 to directly read/write memory, eliminates the need for additional hardware to allow the microprocessor to transfer directly to or from memory.

General Purpose Input/Output Port

The AIC-6250 provides two general purpose I/O ports, Port A and Port B.

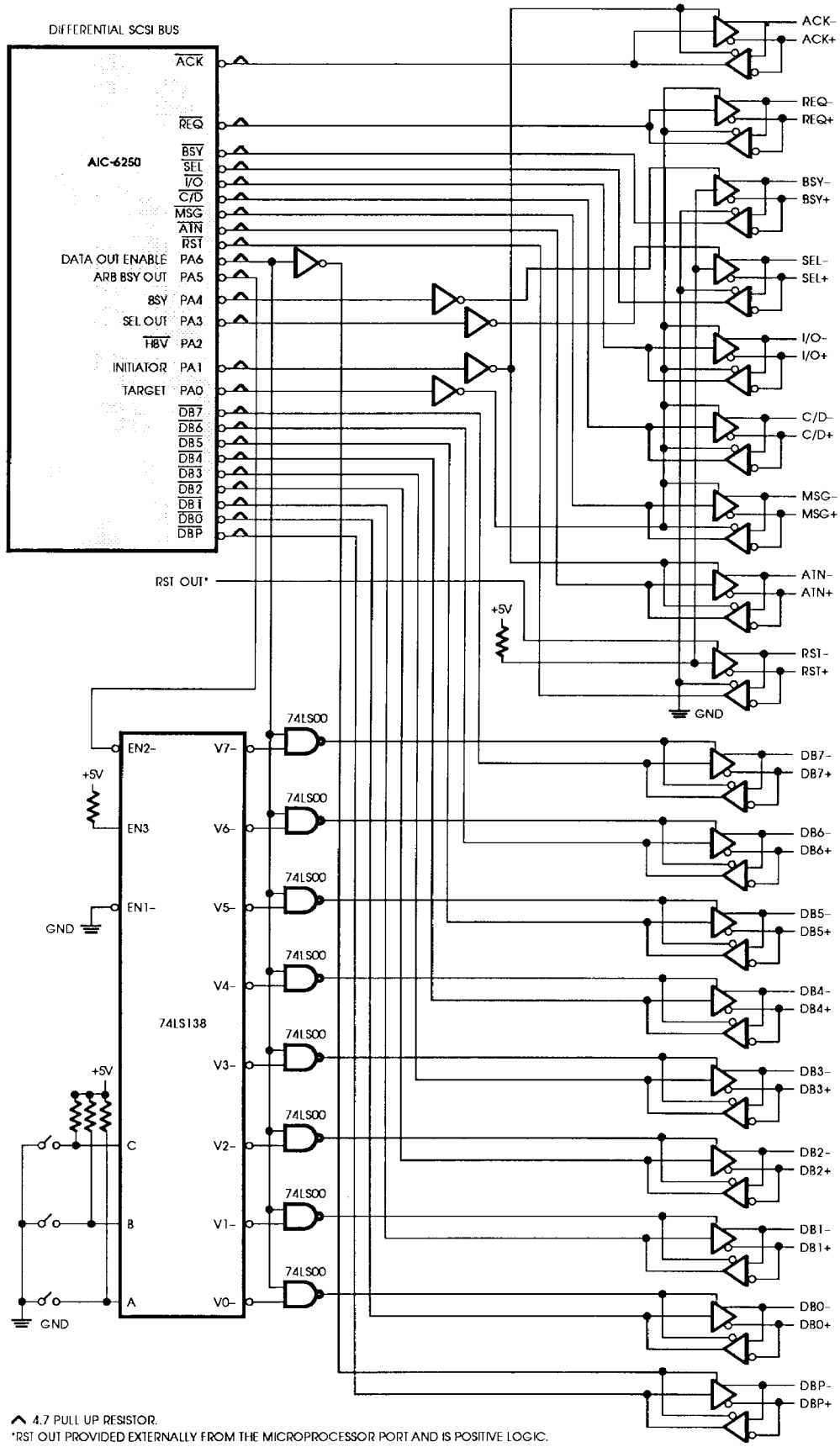
In the default state, Port A is an input or an output/input port. In this mode, all bits may be used as inputs or as outputs. When used as an output port, the values last written to the output port may be read at any time.

When the AIC-6250 is programmed to operate in the 16-bit memory interface mode, Port A, Bits 2 and 7, function as the $\overline{\text{HBV}}$ (High Byte Valid) and $\overline{\text{LBV}}$ (Low Byte Valid) signals used in conjunction with DMA odd-byte transfers. They are enabled as outputs, regardless of the mode of Port A selected.

Port A may also alternatively be used for the differential SCSI bus control as shown in Figure 8. When Port A is used as the differential SCSI bus control, all other bits (except Bit 7 and Bit 2 which perform the $\overline{\text{LBV}}$ and $\overline{\text{HBV}}$ functions) are automatically controlled by the AIC-6250 for the differential SCSI bus. In this mode, the functions of the various pins of Port A are as follows:

PA7 $\overline{\text{LBV}}$
PA6 Data Out Enable
PA5 $\overline{\text{ARB BSY}}$ Out
PA4 $\overline{\text{BSY Out}}$
PA3 $\overline{\text{SEL}}$ Out
PA2 HBV
PA1 Initiator
PA0 Target

Figure 8 shows Port A being used as the differential SCSI bus interface control logic. Since the SCSI RST Out signal is not being provided by the Port A control logic, it must be externally provided by a microprocessor port.



^ 4.7 PULL UP RESISTOR.
*RST OUT PROVIDED EXTERNALLY FROM THE MICROPROCESSOR PORT AND IS POSITIVE LOGIC.

FIGURE 8. DIFFERENTIAL SCSI BUS CONTROL WITH PORT A

Port B has two uses, as follows:

1. In the default state, Port B is an input or an output/input port. In this mode, all bits may be used as inputs or as outputs. (When used as an output port, the values last written to the output port may be read at any time.)

2. Alternatively, Port B may be used as the upper byte of the 16-bit memory bus. When Port B is used as the high byte of the memory data transfer, the AIC-6250 will not assert the BREQ signal until two bytes of data are in the FIFO unless odd start or odd end condition is present. The AIC-6250 can transfer two bytes of data to or from the system data bus once every 100 ns, providing a 20 Mbyte/second transfer rate. System performance is therefore increased through no loss of system data bus bandwidth. Figure 9 shows the AIC-6250 configured for a 16-bit memory interface.

NOTE: Pin 68 is the Parity bit for the high byte of data passed through Port B.

Eight-Byte FIFO

The eight-byte FIFO allows an eight-byte offset (for maximum Synchronous SCSI bus performance) when the AIC-6250 is used in the synchronous transfer mode except for initiator inbound transfers where the maximum offset is seven. When used in the asynchronous transfer mode, with a DMA transfer, the eight-byte FIFO works as a speed matching buffer between the SCSI bus and the memory.

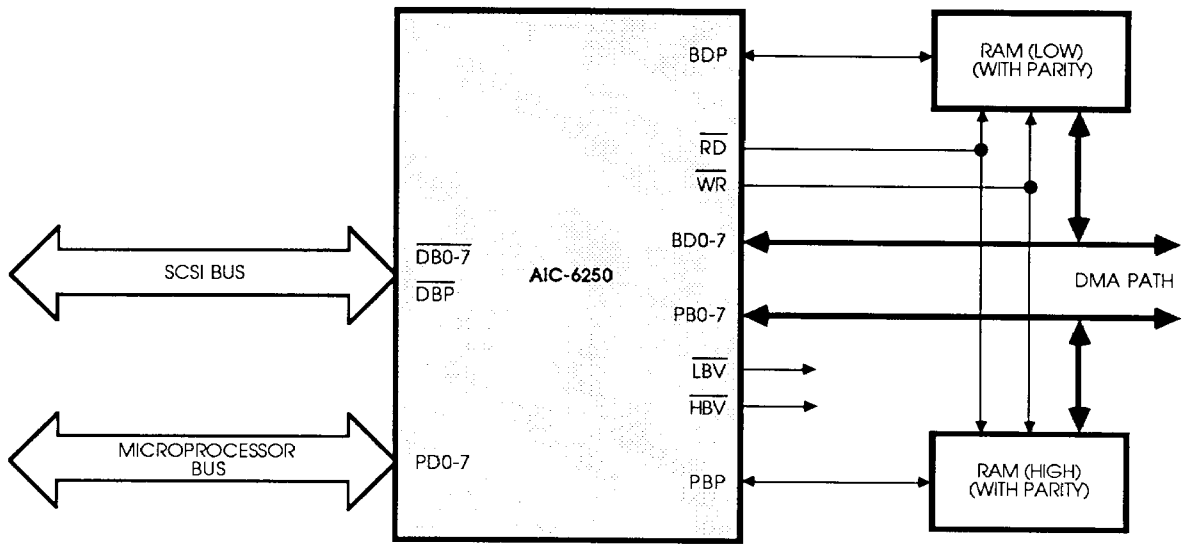


FIGURE 9. 16-BIT MEMORY INTERFACE USING PORT B

PIN DESCRIPTIONS

SYMBOL	PIN	TYPE	NAME AND FUNCTION
			BUFFER INTERFACE
BDP	10	I/O	BUFFER DATA PARITY: Buffer Data Parity is always generated, but checking is optional. Buffer Data Parity Checking is enabled by setting the Enable Memory Parity Error Interrupt bit in the Interrupt Mask Register 1 (Bit 4 of Register 06) to 1.
BD0-7	11-18	I/O	BUFFER DATA BUS BITS 0-7: Byte parallel data lines to/from the buffer. NOTE: Port B provides the upper eight bits for the 16-bit bus configuration.
BREQ	23	O	BUFFER REQUEST: This signal is used with the $\overline{\text{BACK}}$ signal to provide memory cycle requests.
$\overline{\text{BACK}}$	19	I	BUFFER ACKNOWLEDGE: Used to indicate a memory transfer cycle is active. During data transfers from memory to the SCSI bus, the rising edge of this signal is used to latch the data. During data transfers from the SCSI bus to memory, this signal is used to enable data onto the memory bus.
			MICROPROCESSOR INTERFACE
PD0-7	2-9	I/O	PROCESSOR DATA BUS BITS 0-7: Used for bidirectional data transfer between the microprocessor and the AIC-6250.
CLK	20	I	CLOCK: Used for a clock input between 5 MHz and 20 MHz with higher clock rates providing higher transfer rates. When the input clock frequency is higher than 10 MHz, the Clock Frequency Mode bit (Bit 2, Register 08) must be set to 1 to ensure correct SCSI bus timing in the Arbitration and Selection or Reselection phases. NOTE: For maximum arbitration and selection performance, the clock input should be 10 MHz or 20 MHz.
$\overline{\text{WR}}$	21	I	WRITE: With CS active, on the rising edge of this signal, data is strobed from the microprocessor bus to the specified register in the chip.
CS	22	I	CHIP SELECT: This signal allows microprocessor access to/from the chip when active.
$\overline{\text{POR}}$	24	I	POWER ON RESET: When this signal is driven low for a minimum of 50 ns, the chip is reset. The software reset latch will remain active until it is cleared by the microprocessor. To clear the reset latch, the microprocessor must set Bit 0 of Register 08 to 0. When $\overline{\text{POR}}$ is active, bidirectinal pins will be tri-stated.
MODE	25	I	MODE: This pin allows the AIC-6250 to be used easily with microprocessor's having either a multiplexed address/data bus or a nonmultiplexed address/data bus. **High:** The microprocessor data bus supports multiplexed data and addresses with A0 used as the ALE input. **Low:** The microprocessor data bus is used for data only with A0 used as address line 0. All registers must be addressed indirectly by the Address Register.

SYMBOL	PIN	TYPE	NAME AND FUNCTION
			MICROPROCESSOR INTERFACE (Continued)
$\overline{INT}$	57	O	INTERRUPT: This active low open drain output signal indicates an interrupt is pending. The interrupt condition is described in Interrupt Mask Register 0 (Register 03) and Interrupt Mask Register 1 (Register 06).
$\overline{RD}$	58	I	READ: $\overline{RD}$ and CS active cause the data from the specified register to be read onto the Processor Data Lines (PD0-7).
A0/ALE	59	I	ADDRESS 0/ADDRESS LATCH ENABLE: When MODE is low, this pin is address line 0 (A0). In this case, the address of the desired register is loaded into the Address Register during a write cycle. When MODE is high, this control signal is the ALE input.
			SCSI INTERFACE
$\overline{REQ}$	27	I/O	REQUEST: This active low signal is used with the $\overline{ACK}$ signal to perform the SCSI bus handshake protocol. This signal is an input when the chip is operating in the Initiator mode, and is an output when the chip is operating in the target mode.
$\overline{ACK}$	28	I/O	ACKNOWLEDGE: This active low signal is used with the $\overline{REQ}$ signal to perform the SCSI bus handshake protocol. This signal is an output when the chip is operating in the Initiator mode, and is an input when the chip is operating in the target mode.
$\overline{SEL}$	29	I/O	SELECT: This bidirectional active low signal carries the SCSI selection signal. The signal is used in the SCSI selection and reselection process.
$\overline{BSY}$	30	I/O	BUSY: This bidirectional active low signal carries the SCSI $\overline{BSY}$ signal. When $\overline{BSY}$ is active, the SCSI bus is being used.
$\overline{C}/D$	32	I/O	COMMAND/DATA: The $\overline{C}/D$ line is low when command information is on the SCSI bus, and is high when data is on the SCSI bus. In the Initiator mode, this signal is an input. In the target mode, this signal is an output.
$\overline{I}/O$	33	I/O	INPUT/OUTPUT: This signal is driven by the target and controls the direction of information transfer on the SCSI bus with respect to the initiator. The $\overline{I}/O$ line is low when information is input to the Initiator. The $\overline{I}/O$ line is high when information is output from the initiator. In the initiator mode, this signal is an input. In the target mode, this signal is an output.
$\overline{MSG}$	34	I/O	MESSAGE: The $\overline{MSG}$ line is low during a MESSAGE phase on the SCSI bus. In the Initiator mode, this signal is an input. In the Target mode, this signal is an output.
$\overline{ATTN}$	35	I/O	ATTENTION: The $\overline{ATTN}$ line is driven low to signal to the Target that there is a message to be transferred. In the Initiator mode, this signal is an output. In the Target mode, this signal is an input.

SYMBOL	PIN	TYPE	NAME AND FUNCTION
			SCSI INTERFACE (Continued)
$\overline{RST}$	36	I/O	SCSI BUS RESET: This bidirectional active low signal will drive $\overline{RST}$ low on the SCSI bus when the SCSI Reset Out bit is set in Control Register 1 (Bit 1, Register 08). When this pin is driven low externally, it indicates to the chip that there is another SCSI device forcing a $\overline{RST}$ condition on the SCSI bus. This condition is internally latched by the chip and may be cleared by asserting $\overline{POR}$ (pin 24) or a Chip Software Reset (Bit 0, Register 08).
$\overline{DB0\text{-}7}$	38-46	I/O	DATA BITS 0-7: These active low signals carry 8-bit bidirectional data on the SCSI bus.
$\overline{DBP}$	47	I/O	DATA BUS PARITY: This active low signal carries Odd Parity on the SCSI bus data. Parity is always generated, but Parity Checking may be disabled by setting to 0 the Enable SCSI Parity Error Interrupt in Interrupt Mask Register 1 (Bit 1, Register 06).
			PORT INTERFACE
PA0-7	49-56	I/O	PORT A BITS 0-7: This eight-bit port may be used as an eight-bit read or write port. It may also be used for differential SCSI bus control. Bit 7 and Bit 2 function as the $\overline{LBV}$ and $\overline{HBV}$ signals used for handling the 16-bit memory data bus mode.
PB0-7	60-67	I/O	PORT B BITS 0-7: This eight-bit port may be used in two ways: **General Purpose Port:** As a general purpose Input or Output port. **16-Bit Memory Data Bus:** Port B may also be used as the upper eight bits of the 16-bit memory bus. Parity may be enabled if this port is used as a memory bus.
PBP	68	I/O	PORT B PARITY: Port B Buffer Data Parity bit. Buffer Data Parity is always generated (when Port B is used as the high byte of memory transfer), but checking is optional. Buffer Data Parity Checking is enabled by setting to 1 the Enable Memory Parity Error Interrupt bit in Interrupt Mask Register 1 (Bit 4, Register 06).
			POWER
Vcc	1	PWR	+5 Volts.
Vss	26, 31, 37, 43, 48	GND	GROUND.

REGISTER DESCRIPTIONS

The AIC-6250 has 16 registers which provide the flexibility and programmability to handle both the SCSI interface and also the interface to the local buffer memory and the controlling processor. These are listed in Table 1 for quick reference and are described in detail in the following sections.

Register Conditions On Reset

The AIC-6250 essentially has two modes of reset; i.e., the Power On Reset (POR) and the SCSI Reset (SCSI RST) which could be asserted through software on the SCSI bus.

Registers 00, 01, 02, 09, 0A, 0B, 0C, and 0F are not affected by any reset condition and their content is unknown on power up. At other times, it would depend on the activity which preceded the reset action. Also, these registers are relevant only when some activity has been initiated.

Registers 04, 06, 0D and 0E are reset (to inactive zeros) by the POR only.

Registers 05 and 09 are reset (to inactive zeros) by both the POR or SCSI RST. On reset, the state of Register 05(R) would be XX110000. The other registers have multiple conditions as outlined below.

Register 03 is reset to zero by the POR; however, Bit 6 (ARB/SEL Start) would be reset by SCSI RST also.

Register 07 (R), Bits 0,1 are unaffected, while Bits 2-7 are reset by POR. Bits 2,3,5 are also reset and Bit 7 is set to '1' by the SCSI RST condition. Bit 4 will be set to '1' if EN BUS FREE DETECT INT (Reg 06, Bit 2) is set to '1'.

Register 07 (W), all bits except Bit 6, are reset by POR, while Bits 3-5 would also be reset by SCSI RST.

Register 08 (R), Bits 0-5, are reset by POR, while Bits 0-2 are reset by SCSI RST also. The state of Bits 3-5 on reset will be '0.' Bit 6 will be '1', Bit 7 will normally be '1' as determined by BACK*BREQ

Register 08 (W), Bit 5, is a 'don't care.' Bits 1-7 are reset by POR, Bit 0 is set to '1', while Bit 7 is also reset by SCSI RST.

TABLE 1. REGISTER SUMMARY

DMA BYTE COUNT (R/W)

00 (L)		01 (M)		02 (H)	
7	2^7	7	2^{15}	7	2^{23}
6	2^6	6	2^{14}	6	2^{22}
5	2^5	5	2^{13}	5	2^{21}
4	2^4	4	2^{12}	4	2^{20}
3	2^3	3	2^{11}	3	2^{19}
2	2^2	2	2^{10}	2	2^{18}
1	2^1	1	2^9	1	2^{17}
0	2^0	0	2^8	0	2^{16}

03 INT MSK REG (W)

7 RESERVED
6 ARB/SEL START
5 EN AUTO ATN
4 EN ERROR INT
3 EN CMD DONE INT
2 EN SEL OUT INT
1 EN RESEL INT
0 EN SELECT INT

04 OFFSET CNTRL (W)

7 RESERVED
6 SYNC XFER RATE BIT 2
5 SYNC XFER RATE BIT 1
4 SYNC XFER RATE BIT 0
3 OFFSET BIT 3
2 OFFSET BIT 2
1 OFFSET BIT 1
0 OFFSET BIT 0

05 FIFO STATUS (R)

7 TEST SIGNAL
6 TEST SIGNAL
5 OFFSET COUNT ZERO
4 FIFO EMPTY
3 FIFO FULL
2 FIFO COUNTER BIT 2
1 FIFO COUNTER BIT 1
0 FIFO COUNTER BIT 0

05 DMA CNTRL (W)

7 RESERVED
6 RESERVED
5 RESERVED
4 RESERVED
3 RESERVED
2 ODD XFER START
1 TRANSFER DIR
0 DMA XFER EN

06 REV CNTRL (R)

7 RESERVED
6 RESERVED
5 RESERVED
4 RESERVED
3 RESERVED
2 RESERVED
1 REVISION
0 REVISION

06 INT MSK REG 1 (W)

7 RESERVED
6 EN SCSI REQ ON INT
5 EN SCSI RST INT
4 EN MEM PARITY ERR INT
3 EN PHASE MISMATCH INT
2 EN BUS FREE DETECT INT
1 EN SCSI PARITY ERR INT
0 EN PHASE CHANGE INT(INIT)
0 EN ATN ON INT (TGT)

07 STATUS REG 0 (R)

7 SCSI RST OCCURRED
6 MEMORY PARITY ERR
5 PHASE MISMATCH ERR
4 BUS FREE DETECT
3 SCSI PARITY ERR
2 SCSI REQ ON
1 SCSI PHASE CHG/ATN
0 DMA BYTE CNT ZERO

07 CONTROL REG 0 (W)

7 P MEM CYCLE REQ
6 P MEM R/W
5 TARGET MODE
4 EN PORT A INP OR OUT
3 SCSI INTERFACE MODE
2 SCSI ID 2
1 SCSI ID 1
0 SCSI ID 0

08 STATUS REG 1 (R)

7 MEM CYCLE CMPL
6 RESERVED
5 SCSI RST IN
4 ERROR
3 CMD DONE
2 SEL OUT
1 RESELECTED
0 SELECTED

08 CONTROL REG 1 (W)

7 AUTO SCSI PIO REQ
6 EN 16-BIT MEM BUS
5 RESERVED ('0' ONLY)
4 EN PORT B INP OR OUT
3 PHASE CHANGE MODE
2 CLK FREQ MODE
1 SCSI RST OUT
0 CHIP S/W RESET

09 SCSI SIGNAL REG (R)

7 SCSI C/D IN
6 SCSI I/O IN
5 SCSI MSG IN
4 SCSI ATN IN
3 SCSI SEL IN
2 SCSI BSY IN
1 SCSI REQ IN
0 SCSI ACK IN

09 SCSI SIGNAL REG (W)

7 SCSI C/D OUT
6 SCSI I/O OUT
5 SCSI MSG OUT
4 SCSI ATN OUT
3 SCSI SEL OUT
2 SCSI BSY OUT
1 SCSI ACK OUT (INIT)
1 SCSI REQ OUT (TGT)
0 RESERVED

0A SCSI ID DATA (R/W)

7 SCSI ID/DATA 7
6 SCSI ID/DATA 6
5 SCSI ID/DATA 5
4 SCSI ID/DATA 4
3 SCSI ID/DATA 3
2 SCSI ID/DATA 2
1 SCSI ID/DATA 1
0 SCSI ID/DATA 0

0B SOURCE/DEST ID (R)

7 ID 7
6 ID 6
5 ID 5
4 ID 4
3 ID 3
2 ID 2
1 ID 1
0 ID 0

0C MEMORY DATA (R/W)

7 MEM DATA 7
6 MEM DATA 6
5 MEM DATA 5
4 MEM DATA 4
3 MEM DATA 3
2 MEM DATA 2
1 MEM DATA 1
0 MEM DATA 0

0D PORT A (R/W)

7 PORT A BIT 7/$\overline{\text{LBV}}$
6 PORT A BIT 6
5 PORT A BIT 5
4 PORT A BIT 4
3 PORT A BIT 3
2 PORT A BIT 2/$\overline{\text{HBV}}$
1 PORT A BIT 1
0 PORT A BIT 0

0E PORT B (R/W)

7 PORT B BIT 7
6 PORT B BIT 6
5 PORT B BIT 5
4 PORT B BIT 4
3 PORT B BIT 3
2 PORT B BIT 2
1 PORT B BIT 1
0 PORT B BIT 0

0F SCSI LATCH DATA (R)

7 SCSI LATCHED DATA 7
6 SCSI LATCHED DATA 6
5 SCSI LATCHED DATA 5
4 SCSI LATCHED DATA 4
3 SCSI LATCHED DATA 3
2 SCSI LATCHED DATA 2
1 SCSI LATCHED DATA 1
0 SCSI LATCHED DATA 0

0F SCSI BSY RST (TGT) (W)

FOR A TARGET, A WRITE TO THIS REGISTER WILL RESET THE SCSI $\overline{\text{BSY}}$ OUT AND THE SCSI BUS WILL ENTER A BUS FREE PHASE.

Internal Register Descriptions

00 DMA BYTE COUNT, LOW (READ/WRITE)

READ: DMA Byte Count, low-order byte.
WRITE: DMA Byte Count, low-order byte.

01 DMA BYTE COUNT, MIDDLE (READ/WRITE)

READ: DMA Byte Count, middle byte.
WRITE: DMA Byte Count, middle byte.

02 DMA BYTE COUNT, HIGH (READ/WRITE)

READ: DMA Byte Count, high-order byte.
WRITE: DMA Byte Count, high-order byte.

Registers 0-2 determine the number of bytes to be transferred during SCSI data transfer (synchronous or asynchronous) when the DMA Transfer method is used. The DMA Byte Count is decremented by one after each byte of data is transferred on the SCSI bus. By decrementing this counter when a byte is transferred on the SCSI bus, there is no need to determine the number of bytes remaining in the FIFO when a transfer is abnormally terminated. The SCSI bus data transfer will be terminated when the value of Registers 0-2 is zero. The 24-bit counter allows data transfers up to 16 Mbytes without a DMA wrap.

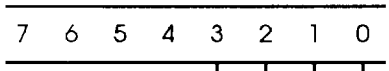
All three registers should be loaded for each DMA operation. The higher-order registers should be loaded with zeros if needed to ensure there is not a residual value from the last operation.

To speed data transfer, the AIC-6250 uses memory prefetch. During a transfer from memory to the SCSI bus, this means the DMA Address Pointer and the DMA Byte Counter in the DMA chip will be incremented when the data is transferred from memory, before the actual SCSI bus data transfer takes place. When SCSI bus data transfer is abnormally terminated, Registers 0-2 will contain the count of the bytes which have not been transferred across the SCSI bus. In this event, the correct value of the DMA controllers Address Pointer must be calculated from the original pointer and the remining byte count in the DMA Byte Count Registers, to restore the number of bytes which have yet to be transferred on the SCSI bus.

Through special logic, the AIC-6250 will stop the memory prefetch when the number of bytes in the FIFO, plus the number of bytes already transferred on the SCSI bus, sums to the total transfer length. This prevents the prefetch from requesting too many bytes from memory.

Detailed Register Descriptions

03 INTERRUPT MASK REGISTER 0 (WRITE)



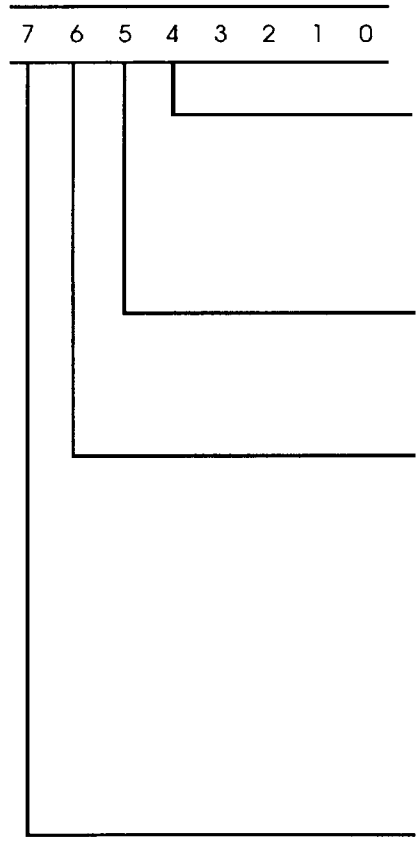
ENABLE SELECTED INTERRUPT: When this bit is set to 1, the AIC-6250 will respond automatically to the SELECTION phase if the SCSI $\overline{SEL}$ signal is asserted with the SCSI bus ID corresponding to that of the AIC-6250 and all of the following conditions are met: no Parity Error (if EN SCSI Parity ERR INT is active), no more than two ID bits on the bus, and the $\overline{I/O}$ signal is not true. After the selection is completed, the AIC-6250 will generate an interrupt indicating a successful selection. When this bit is set to 0, the AIC-6250 will not respond to a SCSI SELECTION phase. In this mode, the microprocessor may respond to the SCSI $\overline{SEL}$ signal by polling the SCSI Signal Register (Register 09) and responding to the SCSI $\overline{SEL}$ signal. This interrupt is identified by the selected bit in Status Register 1 (Bit 0, Register 08).

ENABLE RESELECTED INTERRUPT: When this bit is set to 1, the AIC-6250 will respond automatically to the RESELECTION phase if the SCSI $\overline{SEL}$ signal is asserted with the SCSI bus ID corresponding to that of the AIC-6250 and all of the following conditions are met: no Parity Error (if EN SCSI Parity ERR INT is active), no more than two ID bits on the bus, and the $\overline{I/O}$ signal is true. After the reselection sequence is completed, the AIC-6250 will generate an interrupt indicating a successful reselection. When this bit is set to 0, the AIC-6250 will not respond to a SCSI RESELECTION phase. In this mode, the microprocessor may respond to the SCSI $\overline{SEL}$ signal by polling the SCSI Signal Register (Register 09) and responding to the SCSI $\overline{SEL}$ signal. This interrupt is identified by the Reselected bit in Status Register 1 (Bit 1, Register 08).

ENABLE SELECT OUT INTERRUPT: When this bit is set to 1 before an ARBITRATION phase which is followed by a SELECTION phase, an interrupt will be generated when the SCSI $\overline{SEL}$ signal is asserted by the AIC-6250 after arbitration for the SCSI bus has been won. This interrupt may be used to start a timer for the SCSI Selection Timeout. This interrupt is identified by the Select Out bit in Status Register 1 (Bit 2, Register 08).

ENABLE COMMAND DONE INTERRUPT: When set to 1, completion of any one of the following commands will generate an interrupt: Automatic PIO, DMA Transfer or successful ARBITRATION phase followed by a SELECTION or RESELECTION. This interrupt is identified by the Command Done bit in Status Register 1 (Bit 3, Register 08).

03 INTERRUPT MASK REGISTER 0 (WRITE) Continued



ENABLE ERROR INTERRUPT: When set to 1, any error condition will generate an interrupt error condition. This includes SCSI Reset, memory Parity Error, phase mismatch, SCSI Parity Error, BUS FREE detection phase change. This interrupt is identified by the Error bit in Status Register 1 (Bit 4, Register 08).

ENABLE AUTO ATN: As an initiator, when this bit is set to '1', any of the following conditions will cause SCSI ATN to be asserted: a) SCSI parity error; b) ATN bit in SCSI register; c) SEL asserted.

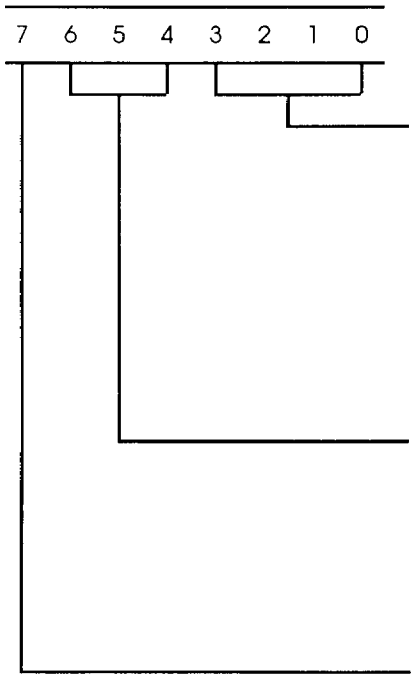
ARBITRATION/SELECTION START: When set to 1, the AIC-6250 will wait for a BUS FREE phase, then automatically proceed to an ARBITRATION phase. If arbitration is won, the AIC-6250 will automatically proceed to the SCSI SELECTION or RESELECTION phase. If arbitration is lost, the AIC-6250 will go back to waiting for a BUS FREE phase, then repeat the same sequence of events. This bit is reset to 0 by the completion of a SELECTION or RESELECTION phase or the microprocessor resets this bit to 0. If the AIC-6250 is selected or reselected by another device, then the microprocessor must reset this bit.

RESERVED.

Writing a zero to Bits 0 and 1 of this register will reset the selected or reselected interrupt status in Status Register 1 (Register 08), causing the interrupt status to be lost. Therefore, after the interrupt signal to the microprocessor is asserted, the microprocessor must first read Status Register 1 (Register 08) to get the interrupt status before writing to this register.

Both Bits 0 and 1 may be set to 1, enabling the AIC-6250 to respond to either a selection or a reselection.

04 OFFSET COUNTER (WRITE ONLY)



OFFSET BITS: These bits set the DMA transfer offset count. An offset count of zero allows asynchronous transfer only. An offset count of one to eight provides SCSI bus synchronous transfer. The maximum offset count is eight. In initiator mode for inbound data transfer, the maximum offset is seven. The offset count must always be set before using the DMA transfer method. Therefore, when performing an asynchronous DMA of information, the offset bits must be set to 0.

SYNCHRONOUS TRANSFER RATE BITS: These bits, in combination with the clock frequency, set the SCSI bus synchronous transfer rate. The table below shows the synchronous transfer rate for three different clock frequencies. The equation for determining the synchronous transfer rate for other clock frequencies is also shown below.

RESERVED.

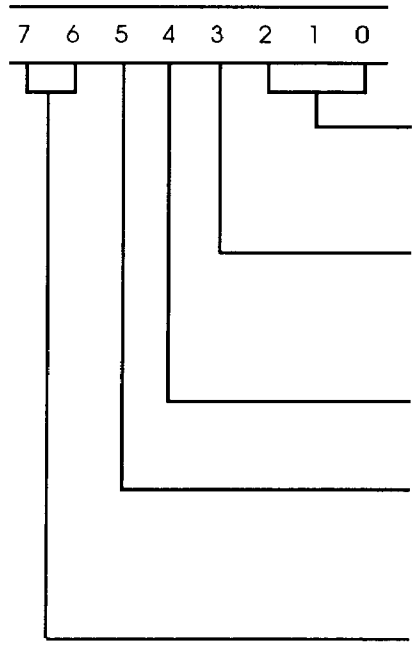
SYNCHRONOUS TRANSFER RATE	CLOCK FREQUENCY 20 MHz	10 MHz	5 MHz
000	5.00 MB/s	2.50 MB/s	1.25 MB/s
001	4.00 MB/s	2.00 MB/s	1.00 MB/s
010	3.33 MB/s	1.67 MB/s	833 KB/s
011	2.86 MB/s	1.43 MB/s	714 KB/s
100	2.50 MB/s	1.25 MB/s	625 KB/s
101	2.22 MB/s	1.11 MB/s	556 KB/s
110	2.00 MB/s	1.00 MB/s	500 KB/s
111	1.82 MB/s	909 KB/s	455 KB/s

EQUATION FOR DETERMINING THE SYNCHRONOUS TRANSFER RATE:

T = Period of Clock
Synchronous Speed = (4 * T) + (Synchronous Transfer Rate * T)

EXAMPLE:
20 MHz Clock Rate (T = 50 ns), Transfer Rate Bits = 7
Synchronous Speed = (4 * 50 ns) + (7 * 50 ns)
= 200 ns + 350 ns
= 550 ns/byte
= 1.82 Mbytes/second

05 READ: FIFO STATUS REGISTER (READ/WRITE)



FIFO COUNTER BITS (Bit 0 = Low Bit): This gives the binary count of the number of bytes remaining in the FIFO.

FIFO FULL: This bit indicates that the FIFO is full with eight bytes of data and cannot take in any more data until information is removed from the FIFO, thus creating a vacancy.

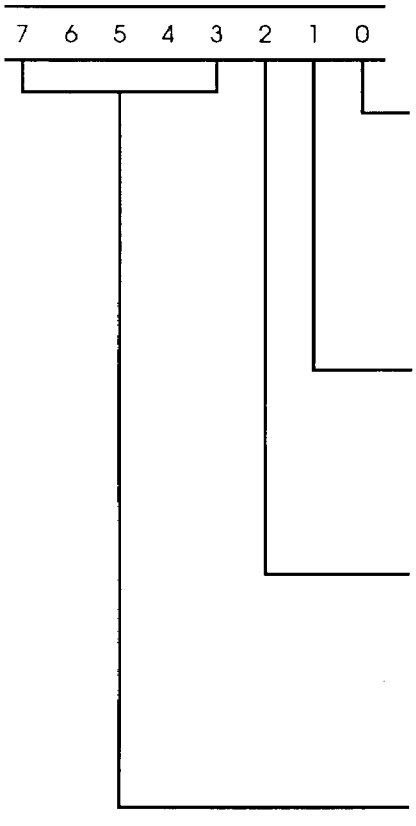
FIFO EMPTY: This bit indicates that the FIFO is empty and no data is present in the FIFO.

OFFSET/COUNT ZERO: When this bit is zero, it indicates that there are no pending SCSI ACK's. In conjuction with FIFO empty, this bit can be used to qualify a disconnection from SCSI bus.

TEST SIGNALS.

NOTE: This register is valid only when BREQ, BACK, REQ and ACK are active.

05 WRITE: DMA CONTROL REGISTER (READ/WRITE)



DMA TRANSFER ENABLE: Once set to 1, the AIC-6250 will automatically begin to transfer data. This bit will be automatically reset on completion of data trasfer. When in initiator mode and a phase match does not occur before completion of DMA transfer, this bit should be reset if DMA is not desired for the next phase. When this is done, a CMD DONE interrupt will be generated.

TRANSFER DIRECTION: This bit indicates the direction of information transfer on the SCSI bus in both the PIO and DMA Transfer modes. When set to 1, information will be transferred from the AIC-6250 to the SCSI bus. When set to 0, information will be transferred from the SCSI bus to the AIC-6250.

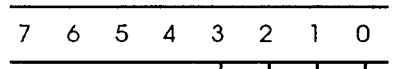
ODD XFER START: When a multibyte 16-bit memory transfer requires the first byte to be transferred to an odd address (i.e., the high-order byte on PB0-7), then this bit must be set. This is used only in 16-bit mode and should not be set when in the eight-bit mode. (Refer to "Memory Data Transfer" later in this Data Sheet for details on odd-byte handling.)

RESERVED.

The microprocessor may set Bits 0 and 1 simultaneously since the AIC-6250 will automatically delay the DMA Transfer Enable until the transfer direction is first set internal to the chip.

The microprocessor may set Bit 2 along with Bits 0 and 1 in the same instruction anytime except when in the Initiator mode for data transfers from SCSI. In such a case, this bit must be set prior to target changing the SCSI bus signals (C/D, I/O, MSG) to the DATA IN phase.

06 WRITE: INTERRUPT MASK REGISTER 1 (READ/WRITE)



ENABLE PHASE CHANGE/ATN INTERRUPT: In initiator mode, setting this bit to 1 will cause an interrupt to be generated when a phase change occurs. See Bit 3 of Control Register 1 (Register 08) for the definition of a phase change. In target mode, setting this bit to 1 will cause an interrupt to be generated when the SCSI $\overline{\text{ATN}}$ line is asserted. The interrupt generated when this event occurs is reflected by the Error bit in Status Register 1 (Bit 4, Register 08) and SCSI Phase Change/ATN bit in Status Register 0 (Bit 1, Register 07).*

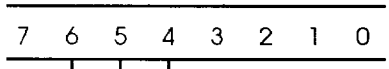
ENABLE SCSI PARITY ERROR INTERRUPT: Setting this bit to 1 enables parity error checking when information is being transferred from the SCSI bus to the AIC-6250. If a SCSI parity error is detected, an interrupt will be generated, however, the information transfer will not be halted. The interrupt generated when this event occurs is reflected by the Error bit in Status Register 1 (Bit 4, Register 08) and the SCSI Parity Error bit in Status Register 0 (Bit 3, Register 07).*

ENABLE BUS FREE DETECTION INTERRUPT: When this bit is set to 1, an interrupt will be generated when a Bus Free phase is detected. This interrupt may be used to check for a Target deasserting the SCSI $\overline{\text{BSY}}$ signal before the normal ending sequence. The interrupt generated when this event occurs is reflected by the Error bit in Status Register 1 (Bit 4, Register 08) and the Bus Free Detected bit in Status Register 0 (Bit 4, Register 07).* When this bit is not set, the AIC-6250 will not set bus free detected bit.

ENABLE PHASE MISMATCH INTERRUPT: This bit is used in the initiator mode only. When this bit is set to 1, an interrupt will be generated if the SCSI $\overline{\text{REQ}}$ signal is asserted and the SCSI phase does not match the expected phase set in the SCSI Signal Register (Register 09). The interrupt generated when this event occurs is reflected by the Error bit in Status Register 1 (Bit 4, Register 08) and the Phase Mismatch bit in Status Register 0 (Bit 5, Register 07.)*

*Denotes a secondary interrupt mask bit. The primary interrupt mask is the Enable Error Interrupt bit in Interrupt Mask Resiger 0 (Bit 4, Register 03).

06 WRITE: INTERRUPT MASK REGISTER 1 (READ/WRITE) Continued



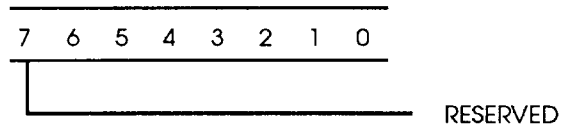
ENABLE MEMORY PARITY ERROR INTERRUPT: When this bit is set to 1, Memory Parity Error Checking is enabled. When using the 16-bit bus mode, memory Parity Checking will be enabled on both the high byte ($\overline{POR}$ B bus) and the low byte (memory data bus). When a Parity Error is detected, an interrupt will be generated and information transfer halted. The interrupt generated when this event occurs is reflected by the Error bit in Status Register 1 (Bit 4, Register 08) and the Memory Parity Error bit in Status Register 0 (Bit 6, Register 07).*

ENABLE SCSI RESET OCCURRENCE INTERRUPT: When set to 1, an interrupt will be generated when a SCSI $\overline{RST}$ condition occurs. A SCSI Reset must be of at least 500 ns in duration. The SCSI $\overline{RST}$ signal will be latched in the SCSI $\overline{RST}$ Occurred bit in Status Register 0 (Bit 7, Register 07) and may be cleared only by setting to 1 the Chip Software Reset bit in Control Register 1 (Bit 0, Register 08) or a $\overline{POR}$ (Power On Reset, pin 24). The SCSI $\overline{RST}$ signal will also be set to 1 on the SCSI Reset In bit in Status Register 1 (Bit 5, Register 08), but is not latched in this bit. This bit may be used to determine when the SCSI $\overline{RST}$ signal is deasserted and chip initialization may begin. The interrupt generated when this event occurs is reflected by the Error bit in Status Register 1 (Bit 4, Register 08) and the SCSI $\overline{RST}$ Occurred bit in Status Register 0 (Bit 7, Register 07).*

ENABLE SCSI $\overline{REQ}$ ON INTERRUPT: This bit is used in the initiator mode when the transfer method is Microprocessor Controlled PIO. When set to 1, this bit causes an interrupt to be generated when the SCSI $\overline{REQ}$ signal is asserted. The interrupt generated when this event occurs is the SCSI $\overline{REQ}$ On bit in Status Register 0 (Bit 2, Register 07).

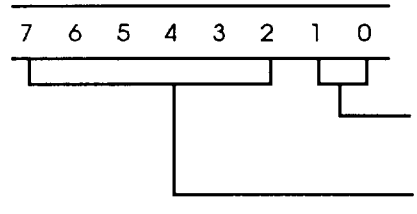
*Denotes a secondary interrupt mask bit. The primary interrupt mask is the Enable Error Interrupt bit in Interrupt Mask Resiger 0 (Bit 4, Register 03).

06 WRITE: INTERRUPT MASK REGISTER 1 (READ/WRITE) Continued

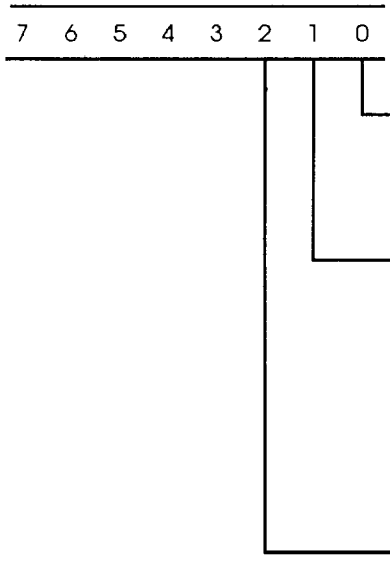


If Bits 0-6 of this register are not set to enable interrupts on the individual events, the microprocessor may poll Status Register 0 (Register 07) to detect the occurrence of the event.

06 READ: REVISION CONTROL REGISTER (READ/WRITE)



07 READ: STATUS REGISTER 0 (READ/WRITE)

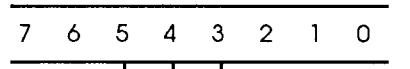


DMA BYTE COUNT ZERO: When this bit is found to be 1, the DMA Byte Count Registers (Registers 00-02) are all zero. This bit will be reset to 0 when Registers 00-02 are loaded with a new transfer count.

SCSI PHASE CHANGE/ATN: In initiator mode, when found to be 1, this bit indicates a SCSI phase change has occurred. This bit is reset when SCSI ACK signal is asserted. In target mode, when found to be 1, this bit indicates the SCSI ATN signal has been asserted. This bit is set to reflect the actual cause of the error when an interrupt is generated with the Error bit set in the Status Register 1 (Bit 4, Register 08).

SCSI $\overline{\text{REQ}}$ ON: In the initiator mode, when found to be 1, this bit indicates the SCSI $\overline{\text{REQ}}$ signal has been asserted. If the initiator wants to start transfer after the SCSI phase has been determined, or use Microprocessor Controlled PIO, this bit may be used for checking the SCSI $\overline{\text{REQ}}$ signal. In this way, if the initiator wishes to wait for $\overline{\text{REQ}}$ to be set to 1, then check the SCSI phase through the SCSI Signal Register (Register 09), this bit would indicate that the SCSI $\overline{\text{REQ}}$ signal has been asserted. The microprocessor may then perform the Phase Check and/or the Microprocessor Controlled PIO Transfer. This bit is reset to 0 when the SCSI $\overline{\text{ACK}}$ signal is asserted (Bit 3, Register 08). This bit is not used in target mode.

07 READ: STATUS REGISTER 0 (READ/WRITE) Continued

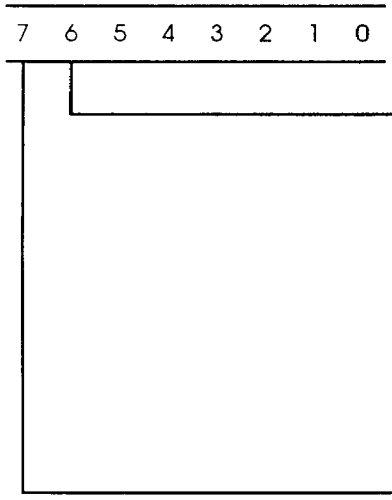


SCSI PARITY ERROR: If this bit is found to be 1 during a transfer from the SCSI bus to the AIC-6250 (inbound), a Parity Error has been detected. This bit can be reset by disabling the SCSI Parity Error Interrupt in Interrupt Mask Register 1 (Bit 1, Register 06). When a Parity Error is detected, and this interrupt is enabled, the $\overline{INT}$ signal (pin 57) is asserted, this bit is set, and the Error bit in Status Register 1 (Bit 4, Register 08) is set to 1. In the initiator mode, information transfer will continue. If AUTO ATN (Reg 03, Bit 5) is on, SCSI $\overline{ATN}$ will be asserted and the Initiator may communicate with the target through the SCSI message system. To reset parity error, the Initiator must reset to 0 the Enable SCSI Parity Error Interrupt in Interrupt Mask Register 1 (Bit 1, Register 06). In the target mode, the target is driving the SCSI bus control lines, so the target may terminate the transfer and return error status to the host. Or, the target may ask the host to retry the transfer via the SCSI message system. This bit will be reset to 0 when the Enable SCSI Parity Error Interrupt bit in Interrupt Mask Register 1 (Bit 1, Register 06) is set to 0.

BUS FREE DETECTED: When this bit is found to be 1, SCSI BUS FREE phase has been detected. This bit will be reset to 0 when the Enable Bus Free Detection Interrupt in the Interrupt Mask Register 1 (Bit 2, Register 06) is masked out by setting to 0. When this bit is found to be set to 1, the Error bit in Status Register 1 (Bit 4, Register 08) will be set as the primary interrupt.

PHASE MISMATCH: In the initiator mode, if this bit is found to be 1, it indicates the SCSI REQ signal has been asserted and the SCSI phase did not match the expected phase in the SCSI Signal Register (Register 09). This bit may also be used for the detection of early termination during data transfer. This bit will be reset to 0 when the microprocessor writes a new, expected phase to the SCSI Signal Register (Register 09) which matches the actual SCSI bus phase. When this bit is found to be 1, the Error bit in Status Register 1 (Bit 4, Register 08) will be set as the primary interrupt. In the Target mode, this bit is not used.

07 READ: STATUS REGISTER 0 (READ/WRITE) Continued

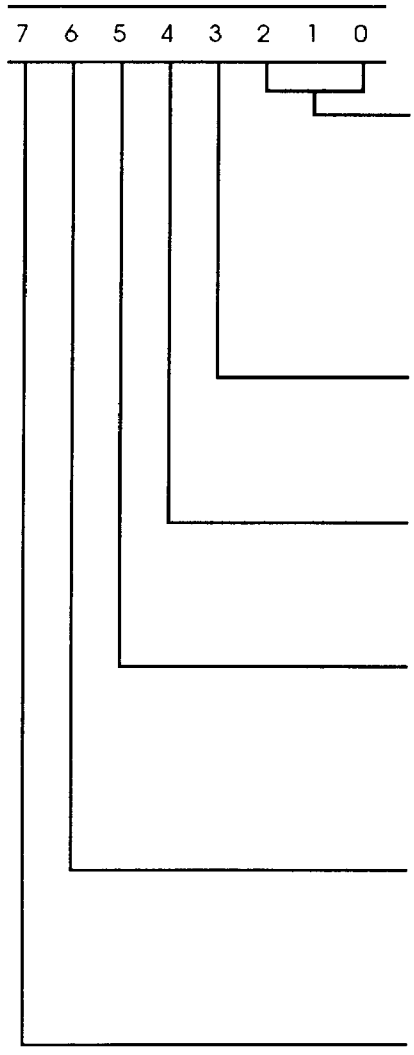


MEMORY PARITY ERROR: When this bit is found to be 1, a parity error on transfer from memory to the AIC-6250 has been detected. When a parity error is detected, information transfer is immediately halted, the $\overline{INT}$ signal (pin 57) is asserted, this bit is set, and the Error bit in Status Register 1 (Bit 4, Register 08) is set. This bit is reset, and information transfer continued, by disabling the Memory Parity Error Interrupt in the Interrupt Mask Register 1 (Bit 4, Register 06). After disabling the Memory Parity Error Interrupt, the interrupt may be re-enabled again at any time.

SCSI RST OCCURRED: When found to be 1, a SCSI Reset condition has occurred. A SCSI Reset condition is defined as the SCSI $\overline{RST}$ line being asserted for a minimum of 500 ns. When the SCSI bus Reset is detected, the Reset condition is latched in this bit, the $\overline{INT}$ signal (pin 57) is asserted, this bit is set to 1, the Error bit in Status Register 1 (Bit 4, Register 08) is set, and all SCSI signals are deasserted. To determine if the Reset signal has been deasserted, the microprocessor must read the SCSI Reset In bit in Status Register 1 (Bit 5, Register 08). This bit is reset by a Chip Software Reset in Control Register 1 (Bit 0, Register 08) or a Power On Reset, pin 24.

Bits 1 and 3-5 of this register are secondary interrupts, enabled through Interrupt Mask Register 1 (Register 06). If error interrupt is not enabled, the microprocessor may still obtain the error information by reading this register.

07 WRITE: CONTROL REGISTER 0 (READ/WRITE)



SCSI ID: These bits are the AIC-6250's decoded SCSI ID. The following is a list of the binary and equivalent decimal ID values:

000 = 0	100 = 4
001 = 1	101 = 5
010 = 2	110 = 6
011 = 3	111 = 7

SCSI INTERFACE MODE: When set to 0, the AIC-6250 is in the SCSI single-ended interface mode. When set to 1, the AIC-6250 is in the SCSI differential interface mode.

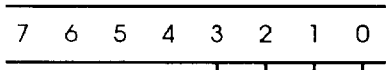
ENABLE PORT A INPUT OR OUTPUT: When this bit is set to 1, Port A is an eight-bit output port. When this bit is set to 0, Port A is an 8-bit input port.

TARGET MODE: When this bit is set to 1 before Arbitration begins, the AIC-6250 will enter the SCSI Reselection phase if arbitration is won. If this bit is not set, the AIC-6250 will proceed automatically from the ARBITRATION phase to the SELECTION phase if arbitration is won.

PROCESSOR MEMORY READ/WRITE: This bit determines the direction of transfer when the microprocessor accesses memory through the AIC-6250. When set to 1, a write cycle will occur. When set to 0, a read cycle will occur.

PROCESSOR MEMORY CYCLE REQUEST: When there is no DMA activity, this bit can be set to request a memory cycle. This bit is automatically reset after the memory cycle is completed. When DMA is in process, the microprocessor cannot access memory through the AIC-6250.

08 READ: STATUS REGISTER 1 (READ)



SELECTED: When set to 1, this bit indicates the AIC-6250 has been selected by another device and the Selection phase is complete. Masking this interrupt by writing 0 to the Enable Selected Interrupt bit of Interrupt Mask Register 0 (Bit 0, Register 03), resets this bit to 0.

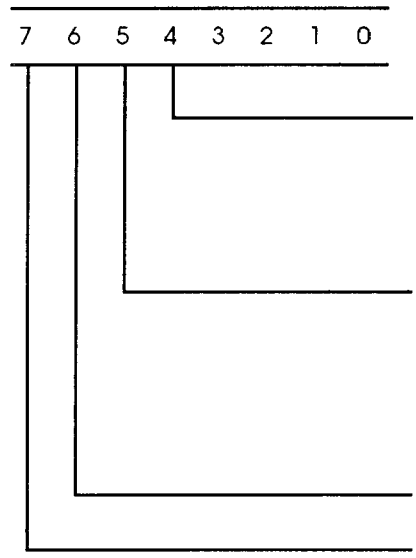
RESELECTED: When set to 1, this bit indicates the AIC-6250 has been reselected by another device and the RESELCTION phase is complete. Masking this interrupt by writing 0 to the Enable Reselected Interrupt bit of Interrupt Mask Register 0 (Bit 1, Register 03), resets this bit to 0.

SELECT OUT: When set to 1, the AIC-6250 is driving the SCSI SEL signal. This bit may be used to start the selection timeout timer. If the SCSI device to be selected does not respond within the selection timeout, there are two methods for canceling the attempted selection:

1. Setting the SCSI Reset Out bit in Control Register 1 (Bit 1, Register 08) to 1 to reset the SCSI bus, or
2. Executing the following sequence:
   a) Write 0 to the SCSI Data Register (Register 0A).
   b) Wait 200 microseconds.
   c) Check for BSY being driven on the SCSI bus by reading the SCSI BSY In bit in the SCSI Signal Register (Bit 2, Register 09).
   d) If BSY is not being driven by another device, set to 0 the Arbitration/Selection Start bit in the Interrupt Mask Register 0 (Bit 7, Register 03).
   e) If BSY is being driven by another device, reset the SCSI bus by setting to 1 the SCSI Reset Out bit in Control Register 1.

COMMAND DONE: When set to 1, this bit indicates one of the following three commands has been completed: 1) DMA Transfer, 2) Automatic PIO, or 3) Arbitration followed by a Selection or Reselection sequence. This bit will be reset to 0 when the Enable Command Done Interrupt bit (Reg 03, Bit 3) is set to zero.

08 READ: STATUS REGISTER 1 (READ) Continued



ERROR: When set to 1, this bit indicates special error handling is necessary. Detailed error status is available in Status Register 0 (Register 07). This bit is reset when the next command is started or on chip reset ($\overline{POR}$) or chip software reset.

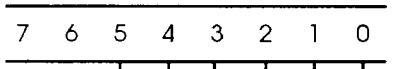
SCSI RESET IN: When set to 1, this bit indicates the SCSI $\overline{RST}$ signal is asserted. This bit is set to 1 only as long as the SCSI $\overline{RST}$ signal is asserted on the SCSI bus. When a SCSI $\overline{RST}$ occurs, this bit should be checked before chip initialization begins to verify the SCSI $\overline{RST}$ signal is still not asserted on the SCSI bus.

RESERVED: Always '1'

MEMORY CYCLE COMPLETE: When this bit is set to 1, the microprocessor memory cycle has been completed. This bit may be used for slow memory devices or multiple bus master systems where the AIC-6250 may not be able to transfer on the memory data bus immediately.

Bits 0,1 and 3 of this register must be read before writing a zero to Bits 0 and 1 of Interrupt Mask Register 1 (Register 03) since masking the interrupts will reset Bits 0 and 1 in this register and cause the interrupt status to be lost.

08 WRITE: CONTROL REGISTER 1 (WRITE)



CHIP SOFTWARE RESET: When set to 1, both the SCSI interface and the memory interface of the AIC-6250 will be reset. This bit will be automatically set after a Power On Reset (pin 24). This bit must be set to 0 to clear the reset condition. After clearing the reset condition, the microprocessor must reinitialize the AIC-6250, and all registers including Register 08.

SCSI RESET OUT: In either target or initiator mode, this bit is set to 1, the SCSI $\overline{RST}$ signal will be asserted by the AIC-6250 on the SCSI bus. When this bit is set to 0, the SCSI $\overline{RST}$ signal will be deasserted on the SCSI bus by the AIC-6250. The SCSI RST signal will continue to be driven onto the SCSI bus until this bit is set to 0.

CLOCK FREQUENCY MODE: This bit must be set to 1 when the clock frequency is greater than 10 MHz. The AIC-6250 uses the input clock to determine the timing on the SCSI bus during the ARBITRATION and the SELECTION or RESELCTION phases. When the clock frequency is greater than 10 MHz, this bit must be set to 1 to ensure correct SCSI bus timing. To achieve maximum SCSI bus performance during the ARBITRATION and the SELECTION or RESELCTION phases, the input clock rate should be at 10 MHz when this bit is set to 0, or 20 MHz with this bit set to 1.

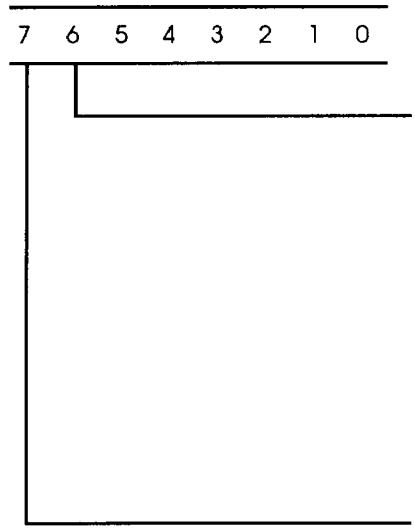
PHASE CHANGE MODE: When this bit is set to 1, any change in state (regardless of the validity of the change in state) of the SCSI $\overline{C/D}$, $\overline{I/O}$, or $\overline{MSG}$ signals, will cause the SCSI Phase Change interrupt in Status Register 0 to be generated. When this bit is set to 0, the SCSI Phase Change interrupt in Status Register 0 will be generated only if there is a change of state of the SCSI $\overline{C/D}$, $\overline{I/O}$, or $\overline{MSG}$ signals and the SCSI $\overline{REQ}$ signal is asserted (regardless of the validity of the change in state).

ENABLE PORT B INPUT OR OUTPUT: When this bit is set to 1, Port B is an input or output/input port. To utilize the 16-bit memory bus, this bit must be set to 0.*

RESERVED: Set to 0.

*For more information on Port A and Port B, see Table 2.

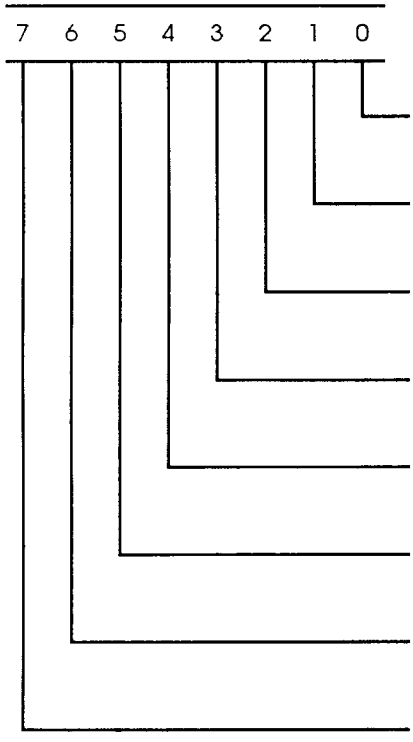
08 WRITE: CONTROL REGISTER 1 (WRITE) Continued



ENABLE 16-BIT MEMORY BUS: When set to 1, Port B is used as the high byte for the 16-bit memory bus. This bit has priority over Bit 4 of this register. Therefore, setting this bit to 1 will enable the 16-bit memory bus and prevent the Port B output port from being enabled. Also, Port A, Bits 2 and 7, become $\overline{HBV}$ and $\overline{LBV}$, respectively, and are enabled as outputs, regardless of the state of Port A Output Enable. These help in odd-byte handling on DMA Transfers to memory. For more details, refer to the section entitled "Memory Data Transfer" later in this data sheet.

AUTOMATIC SCSI PIO REQUEST: When this bit is set to 1, an automatic SCSI handshake is performed. This bit is reset after the handshake is completed. Automatic PIO requires the microprocessor to read/write data from the SCSI Data Register. Using this method of transfer, the Command Done Interrupt in Status Register 1 (Bit 3, Register 08) will be generated after each handshake is completed on the SCSI bus. If the SCSI phase does not match the phase set up in Register 09, then the microprocessor should reset this bit and start again.

09 SCSI SIGNAL REGISTER (READ ONLY)



SCSI ACK IN: When set to 1, the $\overline{ACK}$ signal is asserted on the SCSI bus.

SCSI REQ IN: When set to 1, the $\overline{REQ}$ signal is asserted on the SCSI bus.

SCSI BSY IN: When set to 1, the $\overline{BSY}$ signal is asserted on the SCSI bus.

SCSI SEL IN: When set to 1, the $\overline{SEL}$ signal is asserted on the SCSI bus.

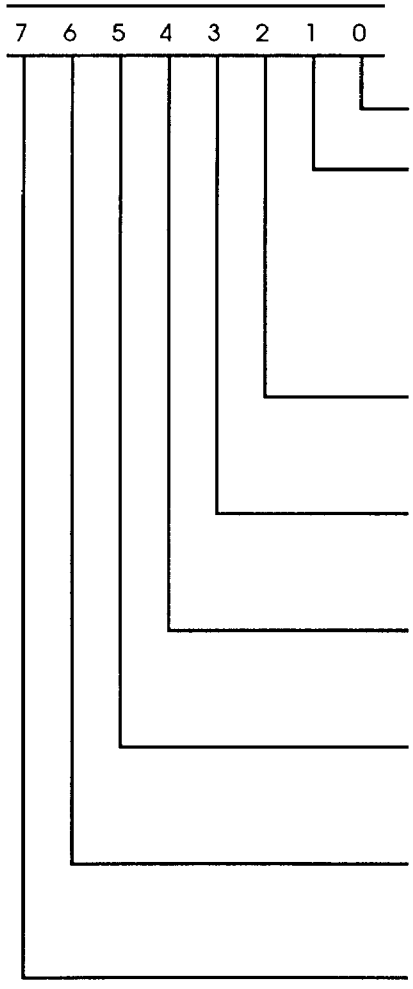
SCSI ATN IN: When set to 1, the $\overline{ATN}$ signal is asserted on the SCSI bus.

SCSI MSG IN: When set to 1, the $\overline{MSG}$ signal is asserted on the SCSI bus.

SCSI I/O IN: When set to 1, the $\overline{I/O}$ signal is asserted on the SCSI bus.

SCSI C/D IN: When set to 1, the $\overline{C/D}$ signal is asserted on the SCSI bus.

09 SCSI SIGNAL REGISTER (WRITE ONLY)



RESERVED.

SCSI ACK OUT: In initiator mode, setting this bit will cause the ACK signal to be asserted by the AIC-6250.*

SCSI REQ OUT: In target mode, setting this bit will cause the REQ signal to be asserted by the AIC-6250.*

SCSI BSY OUT: When this bit is set, the SCSI BSY signal will be asserted by the AIC-6250, when in either target or initiator mode.

SCSI SEL OUT: When this bit is set, the SCSI SEL signal will be asserted by the AIC-6250, when in either target or initiator mode.

SCSI ATN OUT: In initiator mode only, when this bit is set, the SCSI ATN signal will be asserted by the AIC-6250. In target mode, this bit is not valid.

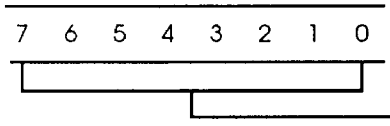
SCSI MSG OUT: In target mode, when this bit is set, the SCSI MSG signal will be asserted by the AIC-6250. In initiator mode, the expected phase is set.

SCSI I/O OUT: In target mode, when this bit is set, the SCSI I/O signal will be asserted by the AIC-6250. In initiator mode, the expected phase is set.

SCSI C/D OUT: In target mode, when this bit is set, the SCSI C/D signal will be asserted by the AIC-6250. In initiator mode, the expected phase is set.

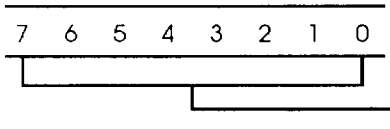
*Using the same bit to set both of these signals prevents the microprocessor from mistakenly setting both the REQ and the ACK signal on the SCSI bus.

0A SCSI ID/DATA (READ/WRITE)



The microprocessor must store both source and destination ID's in this register before the SCSI Arbitration and SELECTION/RESELCTION phases. After the Arbitration and Selection or Reselection sequence is complete, this register may be used to read or write data from or to the SCSI bus. Using the Microprocessor Controlled PIO method to transfer information, this register would be used in conjunction with the SCSI Signal Register (Register 09) and, for an Initiator, the SCSI $\overline{REQ}$ On interrupt in Status Register 0 (Bit 2, Register 07) to transfer data directly from or to the microprocessor to or from the SCSI bus. When transferring data from the AIC-6250 to the SCSI bus, the microprocessor may write data to this register where the information will be stored until it is gated onto the SCSI bus. When transferring data from the SCSI bus to the microprocessor, the microprocessor may read this register, enabling the information which is currently present on the SCSI bus to be gated to the microprocessor. The SCSI data read from this register is not latched by a SCSI $\overline{REQ}$ or $\overline{ACK}$ signal.

0B SOURCE AND DESTINATION ID'S (READ ONLY)



After the SELECTION or RESELCTION phase is complete, this register contains both the source and the destination ID's. The microprocessor should read this register after a SELECTION (target) or RESELCTION (initiator) phase to determine the source ID.

0C MEMORY DATA (READ/WRITE)

This register allows the microprocessor to directly read or write memory data through the AIC-6250.

WRITE: A write to this register stores the microprocessor data in the register. The microprocessor then sets the Processor Memory Cycle Request bit in Control Register 0 (Bit 7, Register 07) to 1 to request a memory cycle, and sets the Processor Memory Write/Read bit in Control Register 0 (Bit 6, Register 07) to 1 to indicate direction.

READ: To read memory data, the microprocessor sets the Processor Memory Cycle Request bit in Control Register 0 (Bit 7, Register 07) to 1 to request a memory cycle, and sets the Processor Memory Write/Read bit in Control Register 0 (Bit 6, Register 07) to 0 to request a memory read cycle. The microprocessor then reads the data from this register.

When using the 16-bit bus mode, the upper byte of data is transferred through Port B (Register 0E).

0D PORT A (READ/WRITE)

READ: Reading this register will read the data which is present on the Port A bus regardless of how Port A is set up. If Port A is set up as an input port, the data off the Port A bus will be read. If Port A is set up as an output/input port, the last data to be put out onto the Port A bus will be read in.

WRITE: The following are example usages of Port A.

16-Bit Buffer Interface: When the Control Register 1 (Register 08), Bit 6 (Enable 16-bit memory bus), is set, indicating the selection of the 16-bit buffer interface, Port A, Bits 2 and 7, become $\overline{HBV}$ (High Byte Valid) and $\overline{LBV}$ (Low Byte Valid), respectively, and are enabled regardless of the state of Port A output enable. For more details on odd-byte handling in which these signals play an important role, refer to the section entitled "Memory Data Transfer" later in this data sheet.

Single-Ended SCSI Interface (General Purpose Mode): All eight bits can be used as input ports or output ports. Writing this register will store the microprocessor data in the Port A register. If Port A is used as an output/input port, the data will be gated to the Port A chip interface.

Differential SCSI Interface: In the Differential mode, Bits 0-7 (except Bits 2 and 7 in the eight-bit buffer interface mode, which are "dont't care", and $\overline{LBV}$ and $\overline{HBV}$ in the 16-bit buffer interface mode) are used as the control signals for the external drivers and receivers. When in the differential mode, the microprocessor does not write or read this register. All signals are controlled based on bits being set and reset in the other registers of the AIC-6250.

The bit definitions are provided below. *All signals are negative true signals and require a pull-up resistor of 4.7K to Vcc.*

0 ENABLE TARGET
1 ENABLE INITIATOR
2 $\overline{HBV}$
3 SCSI SEL OUT
4 SCSI BSY OUT
5 ARBITRATION BSY OUT
6 ENABLE SCSI DATA OUT
7 $\overline{LBV}$

See Table 2 for a summary of how to initialize the AIC-6250 for the different usage options for Port A and Port B.

0E PORT B (READ/WRITE)

The following are the two usages for Port B.

WRITE:

1. **General Purpose Port:** A microprocessor write to this register will store the microprocessor data in the Port B register. If Port B is set up as an output port, the data will be gated to the Port B interface.

2. **16-Bit Memory Data:** A microprocessor write to this register will store the microprocessor data in the Port B register. The data will be enabled to the AIC-6250 Port B interface during the memory write cycle. During a microprocessor read memory operation, after the memory read cycle is completed, this register will contain the high byte of memory data.

When transferring data from memory to the SCSI bus, the AIC-6250 will transfer the first byte from the Buffer Data (BD) bus to the SCSI bus, then the byte from Port B to the SCSI bus.

When transferring data from the SCSI bus to memory, the AIC-6250 will transfer the first byte from the SCSI bus to memory through the Buffer Data (BD) bus, then the second byte from the SCSI bus to memory through Port B.

See Table 2 for a summary of how to initialize the AIC-6250 for the different usage options for Port A and Port B.

READ:

Reading this register will read the data which is present on the Port B bus regardless of how Port B is set up. If Port B is set up as an input port, the data off the Port B bus will be read. If Port B is set up as an output/input port, the last data to be put out onto the Port B bus will be read in.

0F REGISTER (READ/WRITE)

READ—SCSI LATCH DATA: Reading this register will read the data which is latched during the SCSI REQ/ACK cycle. In the Target mode, the data is clocked when the ACK signal is asserted. In the Initiator mode, the data is clocked when the REQ signal is asserted.

NOTE: The initiator should read this register before the beginning of the next handshake in order to avoid data over-write.

WRITE—SCSI BSY RESET (TARGET): In the target mode, writing any value to this register will reset the SCSI BSY Out signal and the SCSI bus will enter the BUS FREE phase.

FUNCTIONAL DESCRIPTION AND OPERATION

This section outlines the functional details and the programming techniques required to control the AIC-6250 SCSI Protocol Chip.

The major function of the AIC-6250 is to control the handshake (between the SCSI bus and the memory bus) necessary to achieve the data transfer. It provides for both initiator and target modes of operation; synchronous data transfer capability up to 5 Mbytes/second with eight-byte offset; asynchronous data transfer capability up to 3 Mbytes/second; and automatic arbitration, selection, reselection, and parity generation with optional checking. It has internal 48 mA drivers and hysteresis receivers for the single-ended SCSI bus and also provides the necessary control signals for the SCSI differential drivers, using external drivers and receivers.

On the microprocessor and buffer data interface, it provides for optional checking on the buffer data bus for added reliability; an 8/16-bit buffer data bus resulting in very high bandwidths; separate buffer and microprocessor data bus to increase throughput; direct access to the SCSI data from the microprocessor; 8/16-bit direct access between buffer data bus and the microprocessor data bus; and general purpose I/O ports.

Microprocessor Interface

The AIC-6250 interfaces to the microprocessor through a separate eight-bit data bus and provides the necessary control signals to do so. Essentially, the task of the microprocessor is to set up the registers of the AIC-6250 to initiate the particular task desired. Once initiated, the AIC-6250 indicates the completion of the task by interrupting the microprocessor. At this juncture, the status of the task issued could be determined by exploring the concerned registers of the chip. The AIC-6250 can be configured to work with either an eight-bit multiplexed address/data bus, or an eight-bit demultiplexed address/data bus. This is accomplished with the help of the Mode pin on the chip.

When Mode is tied high, the AIC-6250 can be interfaced to a microprocessor supporting a multiplexed address/data bus. The trailing edge of ALE is used to latch the address on the bus to an internal address register accomplishing the data read/write. When Mode is tied low, the AIC-6250's ALE/A0 line is used as address line 0. So, when A0 is driven low, the data on the bus is written to the internal address register and the corresponding data register is accessible when A0 is driven high. Refer to Figures 5 and 6 for the interfacing details.

To speed register access operations, the internal address register is automatically incremented after each data register access if the address value is between 00 and 07. For Registers 08 through 0F, the address register value will remain unchanged after access to data registers. This technique allows a more efficient register addressing scheme, as illustrated in the following two examples.

Example 1: The microprocessor writes a zero to the address register. The transfer count may then be written to Registers 00, 01, and 02 without writing the address register again.

Example 2: The microprocessor may write a 9 (1001) to the address register to address the SCSI Signal Register. Since this address is greater than seven, the microprocessor may then read and/or write the SCSI Signal Register multiple times and the address register will not be incremented.

Once set up to initiate an operation, the AIC-6250 uses an interrupt line to communicate with the microprocessor on the status of the operation. The microprocessor can then go in and read the various registers of the chip to determine the status of the operation. There are two status registers to provide status information to the microprocessor. These provide two levels of interrupt vectors, hence, providing additional interrupt status. The first level interrupt vectors are found in Status Register 1 (Register 08) and include the following vectors:

- Selected (Bit 0)
- Reselected (Bit 1)
- Select Out (Bit 2)
- Command Done (Bit 3)
- Error (Bit 4)

The second level interrupts are found in Status Register 0 (Register 07) and include the following vectors:

- SCSI Phase Change/ATTN (Bit 1), set with the Error vector
- SCSI REQ On (Bit 2)
- SCSI Parity Error (Bit 3), set with the Error vector
- Bus Free Detected (Bit 4), set with the Error vector
- Wrong Phase (Bit 5), set with the Error vector
- Memory Parity Error (Bit 6), set with the Error vector
- SCSI Reset Occurred (Bit 7), set with the Error vector

The six error interrupts are OR'ed together and indicate that a first level error interrupt also exists and should be checked in Status Register 1. The first level interrupt, which is associated with a second level interrupt, will be either Error (Bit 4) or Command Done (Bit 3). Except for Command Done, SCSI Parity Error, Memory Parity Error, Bus Free, Selected, Reseleted, all interrupts may be selectively disabled. All interrupts may also be read back through the Interrupt Status Register, independent of whether the interrupt has been enabled. In this way, the microprocessor achieves the task with minimum intervention. The listed interrupts must be enabled to be read through the Status Register.

Power On Reset And Chip Initialization

When the AIC-6250 is powered on, the chip is reset by asserting the Power On Reset line (pin 24) or the chip could be reset by setting the Chip Software Reset bit in the Control Register (Bit 0, Register 08). The $\overline{POR}$ line needs to be asserted for a minimum of 50 ns for the chip to be reset properly. While the $\overline{POR}$ line is asserted, all outputs, except BREQ, will be tristated. Clearing the reset condition is achieved by deasserting the $\overline{POR}$ line and resetting the Chip Software Reset bit in the Control Register. After clearing the reset condition, the microprocessor must initialize the AIC-6250. It should be kept in mind that whenever the chip is reset, then the registers of the chip have to be reinitialized, including all of the bits in Register 08 after the Software Reset Bit is off.

Having cleared the reset condition, the AIC-6250 must be initialized to set up certain critical parameters as follows: 1) the Clock Frequency mode (Bit 2, Register 08) must be set appropriately, based on the input clock frequency as this is used to determine timing on the SCSI bus; 2) enable Port B to be either an I/O port or to be the upper eight bits in case of a 16-bit data bus (Bits 4-6, Register 08); 3)Enable Port A as an I/O port. (Set up the Target mode bit in Control Register 0 in case the AIC-6250 must function as one. When this bit is set before arbitration begins, the AIC-6250 enters the SCSI RESELCTION phase if arbitration is won. If this bit is 0, the AIC-6250 will automatically proceed from the ARBITRATION phase to the SELECTION Phase, if arbitration is won. This enables the AIC-6250 to switch between an Initiator and a Target function very easily; 4) The SCSI Interface mode must be set to indicate single-ended or differential SCSI bus (Bit 3, Register 07); 5) Set the Offset bits, depending on whether an asynchronous or synchronous data transfer is desired (Bits 0-3, Register 04) and the synchronous transfer rate desired (Bits 4-6, Register 04). Having set up these parameters, the method of SCSI data transfer must be decided (DMA, Automatic PIO, Manual PIO) and the appropriate registers must be set up before the arbitration and selection sequence for the SCSI bus can begin.

SCSI Interface

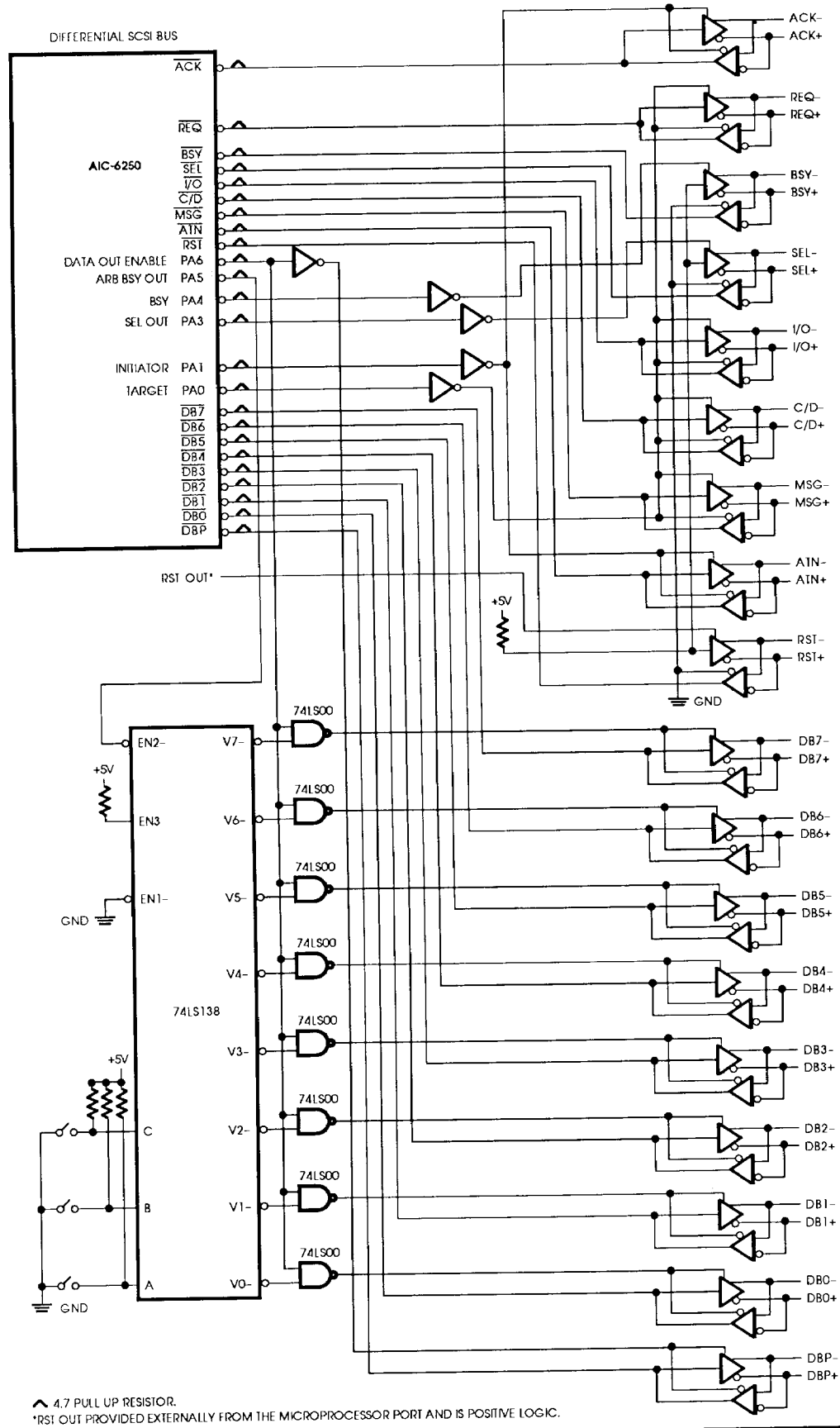
The AIC-6250 handles all the tasks associated with the SCSI bus. The following sections describe the function of the chip with the various aspects of the SCSI bus.

SINGLE-ENDED INTERFACE OPTION

The AIC-6250 includes the 48 mA drivers and hysterisis receivers for the single-ended SCSI bus option. This eliminates the need for external drivers and receivers. This method allows for SCSI bus lengths not to exceed 6 meters. The AIC-6250 connections for the single-ended SCSI bus are shown in Figure 7.

DIFFERENTIAL SCSI INTERFACE CONTROL

The AIC-6250 provides the logic to control external drivers and receivers for the SCSI differential interface. This is accomplished by using Port A of the GP I/O ports, while maintaining the ability to transfer data on the SCSI bus using the asynchronous or synchronous option, as well as being able to implement the 16-bit memory bus by using Port B as the high byte for memory transfer. When Port A is used as the differential SCSI bus control, Bits 0-7 are automatically configured by the AIC-6250 for the differential SCSI bus drivers and receivers control signals (with the exception of Bits 2 and 7). When implementing the SCSI bus differential interface option, each pin is a negative true signal and must have a pull-up resistor. Figure 10 shows Port A being used as the differential SCSI bus interface control logic, using industry standard drivers and receivers. This option allows SCSI cable lengths of up to 25 meters and is particularly recommended for situations where the following of FCC regulations is of importance. It should be noted that the RST Out signal must be generated by the microprocessor as it is not provided on Port A control signals.



^ 4.7 PULL UP RESISTOR.
*RST OUT PROVIDED EXTERNALLY FROM THE MICROPROCESSOR PORT AND IS POSITIVE LOGIC.

FIGURE 10. DIFFERENTIAL SCSI INTERFACE ELECTRICAL CONNECTIONS USING PORT A FOR THE CONTROL SIGNALS

SCSI ARBITRATION AND SELECTION OR RESELECTION

The AIC-6250 automatically looks for the SCSI BUS FREE phase. When a BUS FREE phase is detected (SCSI SEL, BSY, and RST are all deasserted for 400 ns) and if the Arbitration/Selection start bit is set to 1 (between 800 ns to 1.6 s, depending on the clock rate), the AIC-6250 will drive BSY onto the SCSI bus and begin arbitration for the bus. If arbitration is won, the AIC-6250 will automatically proceed to the SCSI SELECTIONn or the SCSI RESELECTION phase, depending on whether the chip is set up for the Initiator or the Target mode. The SCSI bus arbitration and selection or reselection process is independent of whether the chip is operating in the single-ended or the differential mode and is also independent of the information transfer method which is used. This section describes the SCSI arbitration and selection or reselection process for both an initiator and a target device.

Initiator

SCSI Selection: Before beginning the ARBITRATION phase, the following three actions must be initiated:

1. Set up the designation and source ID's in the SCSI ID/Data Register (Register 0A).
2. Set the EN AUTO ATN Out signal and Start Arbitration/Selection bit in the SCSI Signal Register to 1 (Bit 5, 6 Register 03) if the MESSAGE phase following the SELECTION phase is desired, and reset target mode bit in Register 07 (W).

After the AIC-6250 receives this instruction, and if the SCSI bus is free, the AIC-6250 will wait 16 clock cycles,* assert SCSI BSY, then enable its own ID onto the SCSI bus. After 56 more clock cycles,* the AIC-6250 will examine the bus. Depending on the state of the SCSI bus, the AIC-6250 will take one of two possible actions:

1. If, during this time, another device has asserted the SCSI SEL signal, or the AIC-6250 determines that there is a higher priority ID present on the SCSI bus, then the SCSI BSY signal will be deasserted. The AIC-6250 will then wait for the next BUS FREE phase and begin the process over.
2. If the AIC-6250 has the highest ID during the ARBITRATION phase, it will then assert the SCSI SEL signal 24 clock cycles later and enable the source and destination ID's, from the SCSI Data Register, onto the SCSI bus, while removing the SCSI BSY two clock cycles* later. Simultaneously, the ATN signal will be asserted on the SCSI bus, if EN AUTO ATN is set (Reg 03, Bit 5). When the SCSI BSY signal is asserted by the other SCSI device (Target), the AIC-6250 will deassert the SCSI SEL signal one clock cycle* later, the AIC-6250 will then assert the INT signal (pin 57) and set to 1 the Command Done bit in Status Register 1 (Bit 3, Register 08).

SCSI Reselection: After a Target has disconnected, a Reselection must take place to reconnect. Before responding to a Reselection sequence, the Enable Reselected Interrupt bit in the Interrupt Mask Register 0 (Bit 1, Register 03) should be set to 1. After this, the AIC-6250 will watch the SCSI bus for a SCSI RESELECTION phase. If the following conditions are true for at least 12 clock cycles (600 ns), the AIC-6250 will then assert the SCSI BSY signal:

1. The SCSI SEL signal is asserted with the ID that matches the AIC-6250's ID.
2. No more than two ID bits are asserted.
3. The Enable Reselected bit in the Interrupt Mask Register 0 (Bit 1, Register 03) is set to 1.
4. If SCSI Parity Checking is enabled, there is no SCSI Parity Error.
5. The SCSI I/O signal is asserted.
6. The SCSI RST signal is not asserted.

After the Initiator deasserts the SCSI SEL signal, the INT signal (pin 57) will be set and the Selected bit in the Status Register 1 (Bit 0, Register 08) will be set to 1. The Enable Arbitration/Selection Start bit in the Interrupt Mask Register 0 (Bit 0, Register 03) must be reset at this time.

*All clock cycles are referred to assuming the high-frequency mode, set by the Clock Frequency mode bit in Control Register 1 (Bit 2, Register 08). If the low-frequency mode is being used, the number of clock cycles must be divided by two.

Target

SCSI Selection: When the AIC-6250 is being used as a Target (Register 07, Bit 5,), then, before responding to a Selection sequence, the Enable Selected Interrupt bit in the Interrupt Mask Register 0 (Register 03, Bit 0) must be set to 1. Once this is done, the AIC-6250 will watch the SCSI bus for a SCSI SELECTION phase. If all the following conditions are found to be true for at least 12 clock cycles (600 ns), the AIC-6250 will assert the SCSI BSY signal:

1. The SCSI SEL signal is asserted with the ID that matches the AIC-6250's ID.
2. No more than two ID bits are asserted.
3. If SCSI Parity Checking is enabled, there is no SCSI Parity Error.
4. The SCSI I/O signal is not asserted.
5. The SCSI RST signal is not asserted.

After the Initiator deasserts the SCSI SEL signal, the AIC-6250 will keep the SCSI BSY signal asserted. The INT signal (pin 57) will be generated and the Selected bit in Status Register 1 (Register 08, Bit 0) will be set to 1. If the Enable Arbitration/Selection Start bit in the Interrupt Mask Register 0 is set to 1 (Register 03, Bit 0), then it must be reset by the microprocessor at this time.

SCSI Reselection: When the AIC-6250 is functioning as a target, a reselection operation is necessary in order for it to reconnect. Before beginning the ARBITRATION phase, the following four actions must be initiated:

1. Set up the destination and source ID's in the SCSI ID/Data Register (Register 0A).
2. Set to 1 the Enable Target bit in Control Register 0 (, Register 07, Bit 5).
3. Assert the SCSI I/O signal in the SCSI Signal Register (Register 09, Bit 6).
4. Set the Start Arbitration/Selection bit in Interrupt Mask Register 0 (Register 03, Bit 7,).

To start arbitration after the AIC-6250 receives this instruction, and if the SCSI bus is free, the AIC-6250 will wait 16 clock cycles,* assert SCSI BSY, then enable its own ID onto the SCSI bus. After 56 more clock cycles,* the AIC-6250 will examine the bus. Depending on the state of the SCSI bus, the AIC-6250 will take one of two possible actions:

1. If during this time another device has asserted the SCSI SEL signal, or the AIC-6250 determines that there is a higher priority ID present on the SCSI data bus, the SCSI BSY signal will be deasserted. The AIC-6250 will then wait for the next BUS FREE phase and begin the process over.

2. If the AIC-6250 has the highest ID during the ARBITRATION phase, it will then assert the SCSI SEL signal and enable the source and destination ID's, from the SCSI Data Register, onto the SCSI bus. If the SCSI BSY signal is asserted by the other SCSI device, the AIC-6250 will assert SCSI BSY and deassert the SCSI SEL signal one clock cycle* later and the AIC-6250 will assert the INT signal (pin 57) and set to 1 the Command Done bit in Status Register 1 (Register 08, Bit 3).

*All clock cycles are referred to assuming the high-frequency mode, set by the Clock Frequency mode bit in Control Register 1 (Bit 2, Register 08). If the low-frequency mode is being used, the number of clock cycles must be divided by two.

SCSI SELECTION TIMEOUT

In the event that a device does not respond to the SELECTION or RESELECTION phase within 250 ms (the timeout value recommended in the SCSI specification) or any other value the microprocessor determines as the optimal timeout value, the microprocessor must initiate the following:

1. Write a zero to the SCSI ID/Data Register (Register 0A).
2. Wait for another 200 μS.
3. Check the SCSI bus through the SCSI Signal Register (Register 09) to determine if any other device has asserted the SCSI $\overline{BSY}$ signal.

If the SCSI $\overline{BSY}$ signal has been asserted, the SCSI bus must be reset by setting to 1 the SCSI RST Out bit in Control Register 1 (Bit 1, Register 08).

If the SCSI $\overline{BSY}$ signal has not been asserted, the microprocessor may drop the SCSI $\overline{SEL}$ signal, being driven by the AIC-6250, by setting to 0 the Start Arbitration/Selection bit in Interrupt Mask Register 0 (Register 03, Bit 7). Setting this bit to 0 will cause the AIC-6250 to stop driving the $\overline{BSY}$ and $\overline{SEL}$ signals onto the bus, thereby allowing the SCSI bus to go back to a BUS FREE phase after the selection timeout.

The selection timeout must be done using an external counter/clock and, when the preset time has elapsed, the microprocessor will take the necessary steps to stop the arbitration as outlined above.

SCSI Data Transfer

In the asynchronous and synchronous transfer protocols, the AIC-6250 can handle three modes of information transfer on the SCSI bus. The method of data transfer is a software implementation decision left to the designer. The three modes of transfer are DMA transfer, Automatic PIO, and Microprocessor Controlled PIO.

1. *DMA Transfer:* In the DMA transfer method, the AIC-6250 may transfer data on the SCSI bus through either an asynchronous protocol or a synchronous protocol. With DMA transfer, the FIFO internal to the AIC-6250 is utilized in both the asynchronous mode and the synchronous transfer mode; and the handshake for transfer, to or from host memory, occurs automatically. While transferring data from the SCSI bus to the memory, the FIFO holds data bytes less than or equal to the synchronous offset. If for some reason the device disconnects, then the FIFO could be left with some data. This data still continues to be transferred to the memory by the BREQ/BACK handshake, despite the fact that a phase mismatch interrupt has occurred. In the other direction, memory to SCSI, the remaining data in the FIFO is lost; but the exact number of bytes transferred across the SCSI bus is known, so the Address Pointer for the memory could be set back to the correct value. DMA is the most efficient technique supported by the AIC-6250.

2. *Automatic PIO:* In the Automatic PIO method of transferring data, the microprocessor is responsible for data transfer to or from the AIC-6250, while the AIC-6250 will control the SCSI bus handshake. Automatic PIO will generate an interrupt after each byte is transferred across the SCSI bus. This can be used only for the asynchronous SCSI protocol.

3. *Microprocessor Controlled PIO:* Microprocessor Controlled PIO is a fully-programmable transfer mechanism. The microprocessor is responsible for the data transfer to or from the AIC-6250, and has control over the SCSI $\overline{\text{ACK}}$ (initiator) or the SCSI $\overline{\text{REQ}}$ (target) signal. This can be used only for the asynchronous SCSI protocol.

When operating in the initiator mode, any information transfer (except Microprocessor Controlled PIO) may be set up prior to the first SCSI $\overline{\text{REQ}}$. This is accomplished by first setting the expected SCSI phase in Register 09. When the expected SCSI phase matches the actual SCSI phase, the AIC-6250 will begin the information transfer. If the actual SCSI phase does not match the expected SCSI phase, the AIC-6250 will not begin the information transfer and will set the Error bit in Status Register 1 (Register 08, Bit 4) to 1.

DMA TRANSFER— ASYNCHRONOUS SCSI

There are two possible methods for the AIC-6250 transferring data from/to the memory to/from the SCSI bus: target and initiator. The following are the necessary operations to be performed on the AIC-6250 in order to achieve a proper information transfer.

Target

In asynchronous SCSI transfer, as a target device, the assertion of the SCSI REQ signal cannot occur until after the deassertion of the SCSI ACK signal. When transferring information via the DMA transfer method, the microprocessor must first set up the following registers in the AIC-6250:

1. Set up the Enable 16-Bit Memory Bus bit in Control Register 1 (Register 08, Bit 6).
2. Set up the FIFO Offset bits to 00, implying = asynchronous SCSI transfers desired (Register 04, Bits 0, 1, 2, 3).
3. Set up the SCSI phase in the SCSI Signal Register (Register 09).
4. Load the transfer byte count in the DMA Byte Count Registers (Registers 00-02).
5. Set the Transfer Direction bit and DMA Transfer Enable bit in the DMA Control Register, based on the direction of the data transfer desired (Register 05, Bit 0).

Depending on the direction of information transfer, the AIC-6250 will follow one of the two following sequence of events.

Data Transfer From The SCSI Bus To Memory (Read From SCSI): In this case, the AIC-6250 will assert the SCSI REQ signal as long as the following conditions are met.

The FIFO is not full: If the FIFO is full, the AIC-6250 will wait until a transfer out of the FIFO has occurred before asserting the REQ signal on the SCSI bus to initiate the transfer of another byte to the FIFO.

The transfer counter is not equal to zero/The SCSI ACK signal is deasserted: When the SCSI ACK is asserted, the SCSI REQ will be deasserted, and data transfer occurs. After the transfer counter reaches zero and the handshake for the last byte of data is complete, the AIC-6250 will wait until the FIFO is empty; then assert the INT signal (pin 57); and set to 1 the Command Done bit in Status Register 1 (Bit 3, Register 08) if Command Done interrupt is enabled.

Data Transfer From Memory To The SCSI Bus (Write To SCSI): In this case, the AIC-6250 will prefetch the data from memory to the internal FIFO. The AIC-6250 will assert the SCSI REQ signal when the following conditions are met.

The FIFO is not empty: If the FIFO is empty, the AIC-6250 will wait until a memory transfer into the FIFO has occurred before asserting the REQ signal on the SCSI bus to initiate the transfer of another byte.

If Memory Parity Checking is enabled and there is no Parity Error: If Memory Parity Checking is enabled and there is a Parity Error, the AIC-6250 will assert the INT signal (pin 57) and set the error bit in Status Register 1 and the Memory Parity Error bit in Status Register 0 (Bit 6, Register 08, and Bit 3, Register 07, respectively) to 1. The information transfer will not continue after the FIFO is empty unless the microprocessor disables Parity Checking and allows the transfer to continue.

The transfer counter is not equal to zero/The SCSI ACK signal is deasserted: After the AIC-6250 receives the SCSI ACK signal, the SCSI REQ signal will be deasserted and the next handshake begun. After the transfer counter reaches zero and the handshake for the last byte of data is complete, the AIC-6250 will assert the INT signal (pin 57) and set to 1 the Command Done bit in Status Register 1 (Bit 3, Register 08).

Initiator

In asynchronous SCSI transfer, as an initiator device, the assertion of the SCSI $\overline{ACK}$ signal cannot occur until after the assertion of the SCSI $\overline{REQ}$ signal. When transferring information via the DMA transfer method, the microprocessor must first set up the following registers:

1. Set up the Enable 16-Bit Memory Bus bit in Control Register 1 (Bit 6, Register 08).
2. Set the FIFO Offset bits to 00, implying = asynchronous SCSI transfers desired (Bits 0-3, Register 04).
3. Set up the expected SCSI phase in the SCSI Signal Register (Register 09).
4. Load the transfer byte count in the DMA Byte Count Registers (Registers 00-02).
5. Set the Transfer Direction bit and DMA Transfer Enable bit in the DMA Control Register, based on the direction of the data transfer desired (Bit 0,1 Register 05).

Depending on the direction of information transfer, the AIC-6250 will follow one of the two following sequence of events.

Data Transfer From The SCSI Bus To Memory (Read From SCSI): In this case, the AIC-6250 will wait until the SCSI $\overline{REQ}$ signal is asserted and all of the following conditions are met:

- The SCSI phase matches the expected phase.
- The SCSI $\overline{REQ}$ signal is asserted.
- The transfer byte count in the DMA Byte Count Registers is not equal to zero.
- The FIFO is not full.

When all of the above conditions are met, the memory BREQ signal will be asserted to continue the transfer from the FIFO to the memory data bus and the SCSI $\overline{ACK}$ signal will be asserted to bring another byte from the SCSI bus into the FIFO. Upon deassertion of the SCSI $\overline{REQ}$, the AIC-6250 will deassert the SCSI $\overline{ACK}$ signal immediately. After the deassertion of the last SCSI $\overline{ACK}$ and all data has been transferred into memory, the AIC-6250 will assert the $\overline{INT}$ signal (pin 57) and set to 1 the Command Done bit in Status Register 1 (Bit 3, Register 08). If SCSI parity error is detected, transfer will continue, parity error will be set, and an interrupt will be generated.

Data Transfer From Memory To The SCSI Bus (Write to SCSI): In this case, the AIC-6250 will prefetch the data from memory to the internal FIFO. The AIC-6250 will assert the SCSI $\overline{ACK}$ signal when all of the following conditions are met:

- The SCSI phase matches the expected phase.
- The SCSI $\overline{REQ}$ signal is asserted.
- Memory Parity Checking is enabled and there is no parity error.
- The FIFO is not empty.
- The transfer byte count in the DMA Byte Count Registers is not equal to zero.

After transferring the last byte of data on the SCSI bus and upon deassertion of the SCSI $\overline{REQ}$ signal, the AIC-6250 will immediately deassert the SCSI $\overline{ACK}$ signal, assert the $\overline{INT}$ signal (pin 57), and set to 1 the Command Done bit in Status Register 1 (Bit 3, Register 08).

DMA TRANSFER— SYNCHRONOUS SCSI

SCSI bus synchronous transfer may be implemented using the DMA transfer method only. It should be noted that synchronous data transfer is optional and may be used in the data phase if previously agreed to by the initiator and target through the message system. The following is a description of the registers which must be set up and the conditions which must be met to perform the synchronous transfer for both a target and an initiator function with the AIC-6250.

Target

In synchronous SCSI transfer, with the AIC-6250 as a target device, the assertion of the SCSI REQ signal is decided by an offset count. The REQ/ACK offset specifies the maximum number of REQ pulses that can be sent by the target in advance of the number of ACK pulses received from the initiator, establishing a pacing mechanism. However, a requirement for successful completion of the data phase is that the number of ACK and REQ pulses be equal. Before beginning Synchronous information transfer, the microprocessor must first set up the following registers:

1. Set up the expected SCSI phase in the SCSI Signal Register (Register 09).
2. Set the Offset bits to the desired offset (Bits 1 to 8, Register 04).
3. Load the transfer byte count in the DMA Byte Count Registers (Registers 00-02).
4. Set the Enable 16-Bit Memory Bus in Control Register 1 (Bit 6, Register 08).
5. Set the transfer Direction bit and DMA Transfer Enable bit in the DMA Control Register (Bits 0 and 1, Register 05).

Depending on the direction of data transfer, the AIC-6250 will follow one of the two following sequence of events.

Data Transfer From The SCSI Bus To Memory (Read From SCSI): In this case, the AIC-6250 will keep asserting the SCSI REQ signal, according to the transfer rate, until the maximum offset is reached. For each SCSI REQ signal that is sent out, the transfer byte count will be decremented by one. The SCSI bus data transfer will continue as long as the FIFO is not full, and the offset count does not reach the maximum value allowed. When the FIFO becomes full, or the offset count reaches the maximum allowed, the SCSI REQ signal will not be asserted again until this condition has been corrected. (The correction will occur automatically; i.e., the FIFO has some space, or the offset count falls below the limit). Then, the REQ signal will be reasserted. If a SCSI Parity Error occurs, the AIC-6250 will assert the INT signal (pin 57); and set the Error bit in Status Register 1 and the SCSI Parity Error bit in Status Register 0 (Bit 4, Register 08, and Bit 3, Register 07, respectively) to 1. For a transfer to be successful, eventually the number of SCSI ACK's must equal the number of SCSI REQ's issued. On the memory (buffer) side, the handshake will continue as long as the FIFO is not empty. If the FIFO becomes empty, the memory handshake will be temporarily suspended until more data is put in the FIFO. If the FIFO becomes full, the AIC-6250 will wait until a memory bus transfer out of the FIFO has occurred before asserting the REQ signal on the SCSI bus to initiate the transfer of another byte. When the Transfer Counter (Registers 00-02) reaches zero, the SCSI REQ signal will no longer be asserted. After all the data is received, and transferred into memory (the FIFO is empty), and the SCSI ACK signal is deasserted, the AIC-6250 will assert the INT signal (pin 57) and set to 1 the Command Done bit in Status Register 1 (Bit 3, Register 08), indicating completion of the command.

Data Transfer From Memory To The SCSI Bus (Write to SCSI): In this case, the AIC-6250 will assert the SCSI REQ signal, according to the transfer rate, until the maximum offset is reached. For each SCSI REQ signal that is sent out, the transfer byte count will be decremented by one. The SCSI bus data transfer will continue as long as the FIFO is not empty and the offset count does not reach the maximum value allowed. When the FIFO becomes empty, or the offset count reaches the maximum allowed, the SCSI REQ signal will not be asserted until this condition has been corrected. (The correction will occur automatically.) Then, the REQ signal will continue to be asserted. The memory handshake will continue as long as the FIFO is not full and a memory bus Parity Error does not occur. If the FIFO becomes full, the memory handshake will be temporarily suspended until a transfer out of the FIFO has occurred before asserting the BREQ signal on the memory bus to initiate the transfer of another byte. If Memory Parity Checking is enabled and there is a Parity Error, the AIC-6250 will assert the INT signal (pin 57) and set the Error bit in Status Register 1 and the Memory Parity Error bit in Status Register 0 (Bit 6, Register 08, and Bit 3, Register 07, respectively) to 1. The information transfer will not continue unless the microprocessor disables Parity Checking and allows the transfer to continue. When the Transfer Counter (Registers 00-02) reaches zero, and the offset of SCSI ACK and REQ equals zero, the SCSI ACK signal is deasserted and the AIC-6250 will assert the INT signal (pin 57) and set to 1 the Command Done bit in Status Register 1 (Bit 3, Register 08).

Initiator

In synchronous SCSI Transfers, using the AIC-6250 as an initiator device, the assertion of the SCSI ACK signal is decided by an offset count. The Initiator can predict the next phase and start DMA Transfer (before the Target sends its first SCSI REQ) or wait for the first SCSI REQ to be asserted, and then set up the phase and start the DMA transfer. Whichever method is used, the microprocessor must set up the registers, as given below, before the synchronous information transfer may begin:

1. Set the expected SCSI phase in the SCSI Signal Register (Register 09).
2. Set the Offset bits to the desired offset (1-8 for synchronous) and the Synchronous Transfer Rate bits in the Offset Counter Register (Bits 0-3 and 4-7, respectively, of Register 04).
3. Load the transfer byte count in the DMA Byte Count Registers (Registers 00-02).
4. Set the Enable 16-Bit Memory Bus in Control Register 1 (Bit 6, Register 08).
5. Set the Transfer Direction bit and DMA Transfer Enable bit in the DMA Control Register (Bit 0, Register 05).

NOTE: If the next phase is DATA IN, then the current phase must be successfully completed. Otherwise the AIC-6250 will not take data from the SCSI bus. Successful completion does not include wrong phase termination.

Depending on the direction of data transfer, the AIC-6250 will follow one of the two following sequence of events.

Data Transfer From The SCSI Bus To Memory (Read From SCSI): In the synchronous SCSI mode, as an Initiator, the AIC-6250 will wait until the SCSI REQ signal is asserted, and all of the following conditions are met, before initiating data transfer to the memory:

- The SCSI phase matches the expected phase.
- The offset count is equal to, or greater than, one.

When all of these conditions are met, the AIC-6250 will assert the SCSI ACK signal only if the data have been transferred into memory according to the transfer rate until the offset reaches zero. For each SCSI ACK signal that is sent out, the transfer byte count will be decremented by one. The SCSI bus data transfer will continue as long as the FIFO is not full, and the offset count does not reach the maximum value. When the FIFO becomes full, or the offset count reaches zero, the SCSI ACK signal will not be asserted until the condition has been corrected (the correction will occur automatically). Then, the SCSI ACK signal will continue to be asserted. If a SCSI Parity Error occurs, data trasnfer will continue, the AIC-6250 will assert the INT signal (pin 57), and will set the Error bit in Status Register 1 and the SCSI Parity Error bit in Status Register 0 (Bit 4, Register 08, and Bit 3, Register 07, respectively) to 1. If the FIFO is empty, the memory handshake will be temporarily suspended until the FIFO is not empty. If the FIFO becomes full, the AIC-6250 will wait until a memory bus transfer out of the FIFO has occurred before asserting the ACK signal on the SCSI bus to initiate the transfer of another byte. When the DMA byte count (Registers 00-02) reaches zero, the SCSI ACK signal will no longer be asserted. After all the data is received, the offset count is equal to zero, and all the data is transferred into memory (the FIFO is empty), the AIC-6250 will deassert the SCSI ACK signal and assert the INT signal (pin 57) and set to 1 the Command Done bit in Status Register 1 (Bit 3, Register 08).

Data Transfer From Memory To The SCSI Bus (Write To SCSI): The AIC-6250 will wait until the SCSI REQ signal is asserted and all of the following conditions are met:

- The SCSI phase matches the expected phase.
- The offset count is greater than, or equal to, one.
- If Memory Parity Checking is enabled, there is no Parity Error.
- The FIFO is not empty.

After all of the above conditions are met, the AIC-6250 will assert the SCSI ACK signal, according to the transfer rate, until the offset reaches zero or FIFO is empty. For each SCSI ACK signal that is sent out, the transfer byte count will be decremented by one. If the memory transfer rate is slower than the SCSI transfer rate, a FIFO empty condition may occur. In this case, the AIC-6250 will not assert the SCSI ACK signal until there is data in the FIFO. The SCSI bus data transfer will continue as long as the FIFO is not empty and the offset count does not reach zero. When the FIFO becomes empty, or the offset count reaches zero, the SCSI ACK signal will not be asserted until these conditions are corrected. Then, the ACK signal will continue to be asserted. The memory handshake will continue as long as the FIFO is not full and a memory bus Parity Error does not occur. If the FIFO becomes full, the memory handshake will be temporarily suspended (until a transfer out of the FIFO has occurred) before asserting the BREQ signal on the memory bus to initiate the transfer of another byte. If Memory Parity Checking is enabled and there is a Parity Error, the AIC-6250 will assert the INT signal (pin 57) and set to 1 the Error bit in Status Register 1 and the Memory Parity Error bit in Status Register 0 (Bit 6, Register 08, and Bit 3, Register 07, respectively). The information transfer will not continue unless the microprocessor disables Parity Checking and allows the transfer to continue. After the transfer count and the offset count reach zero, the SCSI ACK signal is deasserted and the AIC-6250 will assert the INT signal (pin 57) and set to 1 the Command Done bit in Status Register 1 (Bit 3, Register 08).

AUTOMATIC PIO MODE OF DATA TRANSFER

The Automatic PIO method of data transfer can be used only for asynchronous SCSI protocols. This is different from the DMA Transfer method, in that the DMA Byte Count Registers (00-02) are not used and the internal FIFO does not play a role at all. The Automatic PIO Handshake must be enabled for each byte transferred and the microprocessor must read/write the information to/from the SCSI data register.

Target

When using Automatic PIO with the AIC-6250 as a Target device, it follows the sequence of automatically asserting the SCSI REQ signal, waiting for the ACK signal to be deasserted and then deasserting the REQ signal. Once the ACK signal has been deasserted, the Command Done interrupt will be set, indicating the byte has been transferred on the SCSI bus. Before initiating the Automatic PIO Transfer, the microprocessor must first set up the SCSI phase in the SCSI Signal Register (Register 09). Depending on the direction of information transfer, the AIC-6250 will follow one of the two following sequence of events.

Data Transfer From The SCSI Bus To The AIC-6250 (Read From SCSI): The microprocessor must set up the following registers before the data transfer may take place:

1. Set up SCSI phase.
2. Set to 1 the Automatic SCSI PIO Request bit in Control Register 1 (Bit 7, Register 08).

When the Automatic SCSI PIO Request bit is enabled, the AIC-6250 will assert the SCSI REQ and set to 0 the Automatic SCSI PIO Request bit. When the information transfer is complete, the AIC-6250 will assert the INT signal (pin 57) and set to 1 the Command Done bit in Status Register 1 (Bit 3, Register 08). If an error occurred during the transfer, the INT signal (pin 57) will be set to 1, and the Error bit in Status Register 1 (Bit 4, Register 08) will be set. In this case, the microprocessor must read Status Register 0 (Register 07) for additional error information. At this point, the microprocessor can read the SCSI Latch Data Register (Register OF) if required, thereby completing the information transfer.

Initiator

Data Transfer From The AIC-6250 To The SCSI Bus (Write To SCSI): The microprocessor must set up the following registers before the data transfer may take place:

1. Set up SCSI phase.
2. Write the data to be transferred to the SCSI ID/Data Register (Register 0A).
3. Set to 1 the Automatic SCSI PIO Request bit in Control Register 1 (Bit 7, Register 08).

When the Automatic SCSI PIO Request bit is enabled, the AIC-6250 will assert the SCSI $\overline{REQ}$ and reset to 0 the Automatic SCSI PIO Request bit. When the information transfer is complete, the AIC-6250 will assert the $\overline{INT}$ signal (pin 57) and set to 1 the Command Done bit in Status Register 1 (Bit 3, Register 08). If an error occurred during the transfer, the $\overline{INT}$ signal (pin 57) will be set to 1, and the Error bit in Status Register 1 (Bit 4, Register 08) will be set to 1. In this case, the microprocessor must read Status Register 0 (Register 07) for additional error information.

When the Automatic PIO Handshake is enabled, the AIC-6250 will wait until the SCSI $\overline{REQ}$ signal is asserted. After the SCSI $\overline{REQ}$ signal is asserted, the AIC-6250 will check to see if the SCSI phase matches the expected phase and, then, the SCSI $\overline{ACK}$ signal will be asserted. Automatic PIO can be set up before $\overline{REQ}$; this is the same as DMA Transfer. After the SCSI Handshake is completed, a Command Done interrupt signal (Bit 3, Register 08) will be set. If the SCSI phase does not match the expected phase, the Error interrupt signal (Bit 4, Register 08) will be set, qualified by the actual cause in the Status Register 0 (Register 07). Before beginning the Automatic PIO Transfer, the microprocessor must first set up the expected SCSI phase in the SCSI Signal Register (Register 09). Depending on the direction of information transfer, the AIC-6250 will follow one of the two following sequence of events.

Data Transfer From The SCSI Bus To The AIC-6250 (Read From SCSI): The microprocessor must set up the following registers before the data transfer may take place.

1. Set up the expected SCSI phase.
2. Wait for the SCSI $\overline{REQ}$ signal to be asserted. SCSI $\overline{REQ}$ may be detected through an interrupt by enabling the SCSI REQ On interrupt in Status Register 0 (Bit 2, Register 07).
3. Read SCSI Data Register 0A provided a phase match has occurred.
4. Set to 1 the Automatic SCSI PIO Request bit in Control Register 1 (Bit 7, Register 08), initiating the operation.

When the SCSI $\overline{REQ}/\overline{ACK}$ handshake is complete, the AIC-6250 will assert the $\overline{INT}$ signal (pin 57) and set to 1 the Command Done bit in Status Register 1 (Bit 3, Register 08). If an error occured during the transfer, the $\overline{INT}$ signal (pin 57) will be set to 1, and the Error bit in Status Register 1 (Bit 4, Register 08) will be set to 1. In this case, the microprocessor must read Status Register 0 (Register 07) for additional error information.

Data Transfer From The AIC-6250 To The SCSI Bus (Write To SCSI): The microprocessor must set up the following events before the data transfer may take place.

1. Set up the expected SCSI phase.
2. Write the data to be transferred to the SCSI ID/Data Register (Register 0A).
3. Set to 1 the Automatic SCSI PIO Request bit in Control Register 1 (Bit 7, Register 08).

When the Automatic SCSI PIO Request bit is enabled, the AIC-6250 will wait until the SCSI $\overline{REQ}$ signal is asserted and, then, check to see if the SCSI phase matches the expected phase. If the SCSI phase matches the expected SCSI phase, the SCSI $\overline{ACK}$ signal will be asserted and the Automatic SCSI PIO Request bit is reset to 0. When the information transfer is complete, the AIC-6250 will assert the $\overline{INT}$ signal (pin 57) and set to 1 the Command Done bit in Status Register 1 (Bit 3, Register 08). If an error occurred during the transfer, the $\overline{INT}$ signal (pin 57) will be set to 1, but the Error bit in Status Register 1 (Bit 4, Register 08) will be set. In this case, the microprocessor must read Status Register 0 (Register 07) for additional error information.

MICROPROCESSOR CONTROLLED PIO

Microprocessor Controlled PIO is the mode of data transfer which requires the user to manipulate all the SCSI bus signals. It lets the microprocessor control the SCSI bus signal assertions and deassertions through the SCSI Signal Register (Register 09) to accomplish the SCSI handshake. Microprocessor Controlled PIO differs from Automatic PIO in that the Enable Automatic PIO bit does not need to be set. The microprocessor has complete control over the SCSI bus signals and information transfer may not be set up prior to the first SCSI $\overline{REQ}$. Also, a Command Done interrupt is not generated. When transferring data with the Microprocessor Controlled PIO method, the microprocessor has complete control of all of the events which must occur to complete the information transfer.

Target

When the AIC-6250 is used as a target device with this mode of data transfer, the microprocessor must set the SCSI phase in the SCSI Signal Register (Register 09) independent of the transfer direction. Depending on the direction of information transfer, one of the two following sequence of events occurs.

Data Transfer From The SCSI Bus To The AIC-6250 (Read From SCSI): To perform the transfer, the microprocessor must set up the following registers:

1. Set the SCSI phase.
2. Set the SCSI Out bit in the SCSI Signal Register (Bit 1, Register 09) to assert $\overline{REQ}$ on the SCSI bus.
3. Read the SCSI Signal Register waiting for the SCSI $\overline{ACK}$ In bit (Bit 0, Register 09) to be set to 1.
4. Read data from the SCSI ID/ Data Register (Register 0A).
5. Set the SCSI REQ Out bit in the SCSI Signal Register (Bit 1, Register 09) to 0.
6. Read the SCSI Signal Register (Register 09) waiting for the SCSI ACK In bit (Bit 1) to be set to 0.

Data Transfer From The AIC-6250 To The SCSI Bus (Write To SCSI): To perform the transfer, the microprocessor must execute the following events:

1. Set the SCSI phase.
2. Write data to the SCSI ID/Data Register (Register 0A).
3. Set the SCSI REQ Out bit in the SCSI Signal Register (Bit 1, Register 09) to 1 to assert $\overline{REQ}$ on the SCSI bus.
4. Read the SCSI Signal Register waiting for the SCSI ACK In bit (Bit 0, Register 09) to be set to 1.
5. Set the SCSI REQ Out bit in the SCSI Signal Register (Bit 1, Register 09) to 1.
6. Read the SCSI Signal Register (Register 09) waiting for the SCSI ACK In bit (Bit 1) to be set to 0.

Initiator

When used as an initiator device, independent of the transfer direction, the microprocessor must first read the SCSI phase from the SCSI Signal Register (Register 09). When the SCSI phase matches the expected phase, one of the following two sequence of events needs to be followed.

Data Transfer From The SCSI Bus To The AIC-6250 (Read From SCSI): To perform the transfer, the microprocessor must do the following:

1. Read the SCSI Signal Register (Register 09), waiting for the SCSI REQ In bit (Bit 1) to be set to 1, and check SCSI phase.
2. Read data from the SCSI ID/ Data Register (Register 0A).
3. Set the SCSI ACK Out bit in the SCSI Signal Register (Bit 1, Register 09) to 1.
4. Read the SCSI Signal Register (Register 09) waiting for the SCSI REQ In bit (Bit 1) to be set to 0.
5. Set the SCSI ACK Out bit in the SCSI Signal Register (Bit 1, Register 09) to 0.

Data Transfer From AIC-6250 To The SCSI Bus (Write To SCSI): To perform the transfer, the microprocessor must do the following:

1. Read the SCSI Signal Register (Register 09), waiting for the SCSI REQ In bit (Bit 1) to be set to 1, and check SCSI phase.
2. Write data to the SCSI ID/Data Register (Register 0A).
3. Set the SCSI ACK Out bit in the SCSI Signal Register (Bit 1, Register 09) to 1.
4. Read the SCSI Signal Register (Register 09) waiting for the SCSI REQ In bit (Bit 1) to be set to 0.
5. Set the SCSI ACK Out bit in the SCSI Signal Register (Bit 1, Register 09) to 0.

Memory Data Transfer

The AIC-6250 may be configured for an eight-bit or a 16-bit memory data transfer. The memory data transfer occurs through an asynchronous, two-wire handshake (BREQ/$\overline{\text{BACK}}$). When the AIC-6250 requires a memory data transfer, BREQ will be set to 1 and the AIC-6250 will wait for $\overline{\text{BACK}}$ to be set to 0. After $\overline{\text{BACK}}$ is set to 0, the AIC-6250 will reset BREQ to 0 and wait for $\overline{\text{BACK}}$ to be reset to 1 before starting the next memory cycle.

Memory data transfer may occur through one of three ways:

1. *DMA Transfer:* When DMA Transfer is selected, the AIC-6250 automatically performs the handshake on the SCSI bus and the memory bus. When a write from memory to the SCSI bus is occurring, the AIC-6250 will prefetch data from memory and place it into the FIFO prior to the SCSI bus transfer.

2. *Microprocessor read/write:* The microprocessor can perform a read/write memory cycle through the AIC-6250 by requesting a memory PIO cycle.

3. *The microprocessor reading and writing memory through external hardware.*

DMA TRANSFER

When data is being transferred from memory to the SCSI bus (write to SCSI), the AIC-6250 will continue the memory data handshake as long as the FIFO is not full. If the FIFO is full, it waits until data is removed onto the SCSI bus; then resumes the handshake. This handshake (BREQ/$\overline{\text{BACK}}$) is asynchronous and transfers a byte, or a word, at a time. If, for some reason, a phase change occurs and the device requests disconnection, then the DMA Byte Count (Registers 00-02) reflects the bytes of data transferred across the SCSI bus. In order to take care of the data left in the FIFO, the DMA Address Pointer for the memory must be moved back by the difference between it and the number of bytes transferred across the SCSI bus.

When the DMA byte count reaches 8, prefetch will be interrupted. Memory transfer will resume on a byte/word basis when the FIFO has been emptied until the DMA count reaches zero.

When data is being transferred from the SCSI bus to memory (read from SCSI), the AIC-6250 will continue the memory data handshake as long as the FIFO is not empty. If an error condition occurs; i.e., a disconnection request, giving a phase mismatch error, then the AIC-6250 holds off the SCSI ACK and continues the BREQ/BACK Handshake to flush out the data before giving the ACK and interrupting the processor.

Eight-Bit DMA Transfer

In the 8-bit memory bus mode, the AIC-6250 will initiate the handshake at a maximum rate of once every 100 ns (10 Mbytes per second). When data is being transferred from the AIC-6250 to memory, BREQ will be generated as soon as there is a byte in the FIFO. When data is being transferred from memory to the AIC-6250, BREQ will be generated during an outbound DMA Transfer as long as the FIFO is not full. The AIC-6250 will terminate the memory data Handshake when the DMA Byte Count Register is 0 (Registers 00-02).

16-Bit DMA Transfer

When transferring data from memory to the SCSI bus, the AIC-6250 will transfer two bytes from memory to the AIC-6250, at the same time, through the BREQ/BACK handshake. These two bytes will then be transferred onto the SCSI bus with the byte from the buffer data bus being transferred to the SCSI bus first, and the byte from the Port B bus being transferred to the SCSI bus next.

When transferring data from the SCSI bus to memory, the AIC-6250 will transfer the first byte from the SCSI bus to the buffer data bus and the second byte from the SCSI bus to the Port B bus. The data on the buffer data bus and the data on the Port B bus will then be transferred to memory at the same time through the BREQ/BACK Handshake.

In the 16-bit memory bus mode, the AIC-6250 will initiate the handshake at a maximum rate of once every 100 ns (20 Mbytes per second). When data is being transferred from the AIC-6250 to memory, BREQ will be generated as soon as there are two bytes in the FIFO. When data is being transferred from memory to the AIC-6250, BREQ will be generated during an outbound DMA Transfer as long as the FIFO has at least a two byte space available.

Odd-Byte Handling In 16-Bit DMA Transfer: When a multibyte 16-bit transfer requires the first byte to be transferred to or from an odd memory address (Port B bus), Bit 2 of the DMA Control Register (Register 05) must be set. Enabling the 16-bit memory bus mode (Bit 6, Register 08), configures Port A, Bits 2 and 7, as output signals (HBV, High Byte Valid, and LBV, Low Byte Valid, respectively), regardless of the state of Port A output enable. The Odd-Byte Start bit (Register 05) can be set along with the DMA Enable bit in the same instruction when in the target mode for data transfers to/from SCSI and in the Initiator mode for data to SCSI only. In the Initiator mode for data transfers from SCSI, this bit must be set prior to target changing the SCSI bus signals (C/D, I/O, and MSG) to the desired phase; i.e., MESSAGE IN, DATA IN, and STATUS IN phases.

The LBV and HBV signals are active low in nature and are active on both transfer directions. While transferring 16-bit data between the FIFO and the buffer memory, if the transfer length is odd and beginning on an even address, then, when the last byte is transferred, HBV will be inactive and LBV will be active, and the data is present on the BD0-7 bus only. However, if an odd-byte transfer occurs at the beginning of a 16-bit DMA cycle, then LBV will be inactive and HBV will be active, and the data will be present on the PB0-7 bus only. Meanwhile, if a 16-bit transfer takes place between the FIFO and the buffer, then both these lines will be active.

The odd-byte start operation is only applicable to the DMA mode of data transfer; i.e., when Register 05, Bit 0, is set, implying the DMA Transfer of Command, Data, Status, or Message. Each time a DMA odd address transfer is to be done, this bit needs to be set as this bit is internally reset when the first memory cycle of the transfer is completed. The odd-byte handling feature results in a constraint.

In initiator mode, when two consecutive DMA operations are required and the second operation is a Synchronous data transfer with Odd-Byte Start, then it is impossible for the micro-processor to set the Odd-Byte Start bit in Regiter 05 between the two DMA operations. Therefore, the last byte of the first transfer operation (Command or Message) must be PIO and the ODD START bit should be set up prior to the completion of the last byte transfer.

For DMA synchronous SCSI data transfer if DMA ends at an odd address, the next information transfer phase has to be Automatic SCSI PIO or Microprocessor Controlled PIO.

MICROPROCESSOR CONTROLLED MEMORY READ/WRITE

The microprocessor may access the buffer memory through the Memory Data Register (Register 0C) in the AIC-6250. Before requesting memory accesses, the microprocessor must confirm that there is no DMA in progress.

Depending on whether the information transfer is a read or write of memory, one of the following sequence of events must be followed. If multiple bytes (or words) are to be accessed, the following sequence of events may be repeated as many times as necessary.

Memory Write

8-bit Mode

1. Write data into the Memory Data Register (Register 0C).
2. Set the Processor Memory Cycle Request bit in Control Register 0 (Bit 7, Register 07) to 1.
3. Set the Processor Memory Write/Read bit in Control Register 0 (Bit 6, Register 07) to 1 to start a memory write cycle.

16-bit Mode

1. If both bytes are valid, set $\overline{HBV}$ and $\overline{LBV}$, write data to Port B and Memory Data Register. If only one byte of data is valid, set $\overline{LBV}$ or $\overline{HBV}$ and write data to Port B or Memory Data Register respectively.
2. Set the Processor Memory Cycle Request bit and Processor Memory Write/Read bit in Control Register 0 (Bit 6, Registers 07) to 1 to start a memory write cycle.

At this point, the AIC-6250 will automatically perform the BREQ/$\overline{BACK}$ handshake to transfer the memory data. In 16-bit mode after the cycle is completed, set both $\overline{LBV}$ and $\overline{HBV}$ to zero.

Memory Read

1. If a 16-bit memory read, then set $\overline{HBV}$ and $\overline{LBV}$ as needed depending on the validity of the high and low bytes of data.
2. Set the Processor Memory Cycle Request bit and Processor Memory Write/Read bit in Control Register 0 (Bit 6,7, Register 07) to 1 to start a memory read cycle.

At this point, the AIC-6250 will automatically perform the BREQ/BACK handshake to transfer the memory data. Completion of the memory transfer is indicated by the Memory Cycle Complete bit in Status Register 1 (Bit 7, Register 08) being set to a 1. Then, read data from the Memory Data Register (Register 0C). For 16-bit transfers, read data from the appropriate register.

General Purpose I/O (GP I/O) Ports

There are two general purpose ports available on the AIC-6250 which also have specific functions.

PORT A

Port A has two functions:

1) in the default state, Port A is an input or an output/input port. After power up, Port A will be in a high-impedance mode.
2) if the AIC-6250 is programmed to support the differential SCSI interface, then Port A provides the control signals to drive the external drivers and receivers. Figure 10 shows the differential ended SCSI interface and the Port A assignments. In this mode, the Port A output signals are active low logic and each pin must have a pull-up resistor of 4.7K to Vcc. Refer to Table 2 for the set-up options for Port A and Port B.

It should be noted that, when the 16-bit memory mode is selected, Port A, Bits 2 and 7, will function as the $\overline{LBV}$ and $\overline{HBV}$ signals. When in the eight-bit memory mode, Port A functions as a GP I/O port.

NOTE: Port A outputs are inverted compared to REG OD writes.

PORT B

Port B has two functions which are as follows:

1. In the default state, Port B is an input or an output/input port. In this mode, all bits may be used as an input or as an output/input. (When used as an output port, the values last written to the output port may be read at any time.)

2. The second use for Port B is as the upper byte of the 16-bit memory bus. When Port B is used as the high byte of the memory data transfer, the AIC-6250 will not assert the BREQ signal until two bytes of data are in the FIFO. The AIC-6250 can transfer two bytes of data to/from the microprocessor data bus once every 100 ns, providing a 20 Mbyte/second transfer rate. System performance is, therefore, increased through no loss of system microprocessor data bus bandwidth.

NOTE: Pin 68 is the Parity bit for the high byte of data passed through Port B.

TABLE 2. SET UP OPTIONS FOR PORT A AND PORT B

ENABLE PORT B OUTPUT	ENABLE 16-BIT MEMORY	ENABLE PORT A OUTPUT	DIFFERENTIAL END SCSI	OUTCOME
B4, R08	B6, R08	B4, R07	B3, R07	
0	0	0	0	Both Port A and Port B are input ports.
0	0	1	0	Port A is an output port. Port B is an input port.
0	0	0	1	Port B is an input port. Port A (Bits 0-7) is used for the SCSI Differential interface control (except Bits 2 and 7).
1	0	0	0	Port A is an input port. Port B is an output port.
1	0	1	0	Both Port A and Port B are output ports.
1	0	0	1	Port B is an output port. Port A (Bits 0-7) is used for the SCSI Differential interface control (except Bits 2 and 7).
0	1	0	0	Port A is an input port (except Bits 2 and 7 which are $\overline{LBV}$ and $\overline{HBV}$). Port B is used for the high byte of the memory bus.
0	1	1	0	Port A is an output port (except Bits 2 and 7 which are $\overline{LBV}$ and $\overline{HBV}$). Port B is used for the high byte of the memory bus.
0	1	0	1	Port B is used for the high byte of the memory bus. Port A (Bits 0-7) is the SCSI Differential interface control (except Bits 2 and 7 which are $\overline{LBV}$ and $\overline{HBV}$).

AC/DC TIMING PARAMETERS

Absolute Maximum Ratings

Ambient Temperature Under Bias ..... 0°C to 70°C
Storage Temperature ..... -65°C to 150°C
Voltage on Any Pin with Respect to Ground ..... -0.5 to 7 Volts
Power Dissipation ..... 0.5 Watts

NOTE: Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

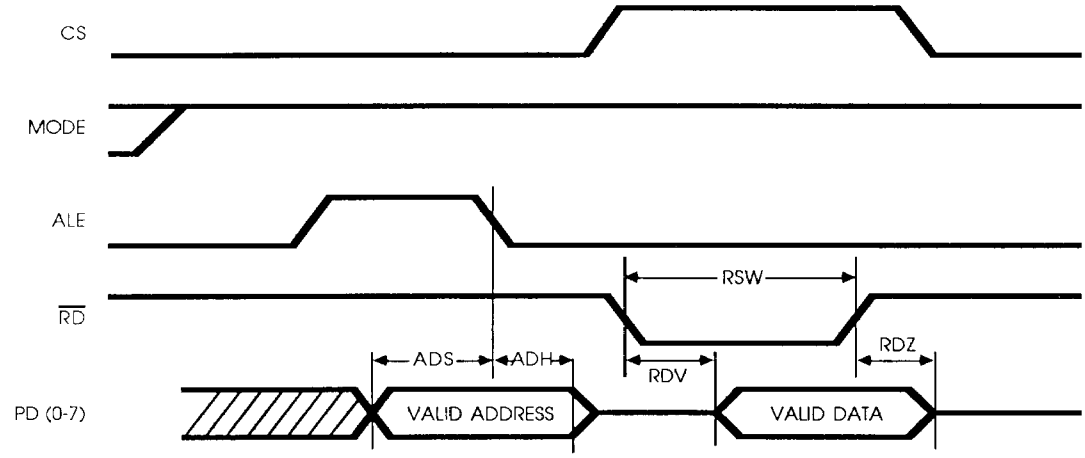
Recommended Operating Conditions

(Conditions: V_{CC} = 5.0V ± 5%, V_{SS} = 0V, 0 C < T $_a$ < 70 C)

SYMBOL	PARAMETER	MIN	MAX	UNITS	CONDITIONS
I_{DDQ}	Quiescent Current		100	µA	$V_{IN} = V_{DD}$
I_{DD}	Operating Current		70	mA	
V_{IH1}	Input High Voltage	2.0		V	
V_{IL1}	Input Low Voltage (DB0-7, DBP, RST, C/D, I/O, MSG, REQ, ACK, ATTN, SEL and BSY)	-0.25	0.8	V	
I_{oz}	Output leakage for tri-state (Off State) (DB0-7, DBP, RST, C/D, I/O, MSG, REQ ACK, ATTN, SEL, BSY, BD0-7, BDP, PD0-7 PA0-7, PB0-7 and PBP)	-40	40	µA	$V_{IN} = V_{DD}$ to V_{SS}
V_H	Hysteresis (DB0-7, DBP, RST, C/D, I/O, MSG, REQ, ACK, ATTN, SEL and BSY)		200	mV	
V_{IL2}	Input Low Voltage (BD0-7, BDP, PD0-7, PA0-7, PB0-7, PBP, CS, WR, RD, MODE, A0/ALE, CLK, BACK and POR)		0.8	V	
I_{IL}	Input Leakage (CS, WR, RD, MODE, A0/ALE, CLK, BACK and POR)	-2	2	µA	$V_{IN} = V_{DD}$ to V_{SS}
V_{OH1}	Output High Voltage (DB0-7, DBP, RST, C/D, I/O, MSG, REQ, ACK, ATTN, SEL and BSY)				Open Drain
V_{OL1}	Output Low Voltage (DB0-7, DBP, RST, C/D, I/O, MSG, REQ, ACK, ATTN, SEL and BSY)		0.5	V	@ I_{OL} = 48 mA
V_{OH2}	Output High Voltage (BD0-7, BDP, PD0-7, PA0-7, PB0-7, PBP and BREQ)	2.4		V	@ I_{OL} = -400 µA
V_{OL2}	Output Low Voltage (BD0-7, BDP, PD0-7, PA0-7, PB0-7, PBP and BREQ)		0.4	V	@ I_{OL} = 2 mA
V_{OL3}	Output Low Voltage (INT)		0.4	V	@ I_{OL} = 4 mA

ALE Register Read

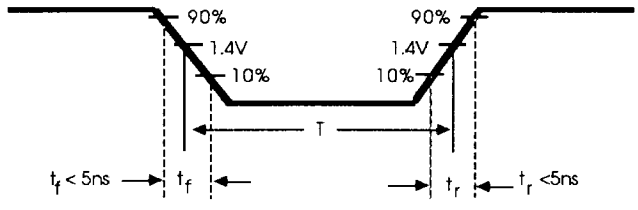
Multiplexed Address/Data Bus Mode



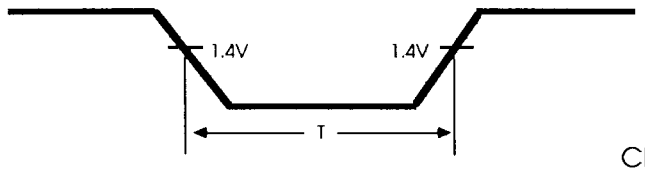
SYMBOL	PARAMETER	MIN	MAX	UNITS
RSW	Read Strobe Width	100		ns
RDV	Read Data Valid		80	ns
ADS	Address Set Up to ALE ↓	50		ns
RDZ	Read Strobe to Data Tri-State	10		ns
ADH	Address Hold from ALE ↓	15		ns

A.C. Test Conditions (Conditions: V_{cc} = 5.0V± 5%, V_{ss} = 0V, 0°C < T_a < 70°C)

Input Timing Conditions:



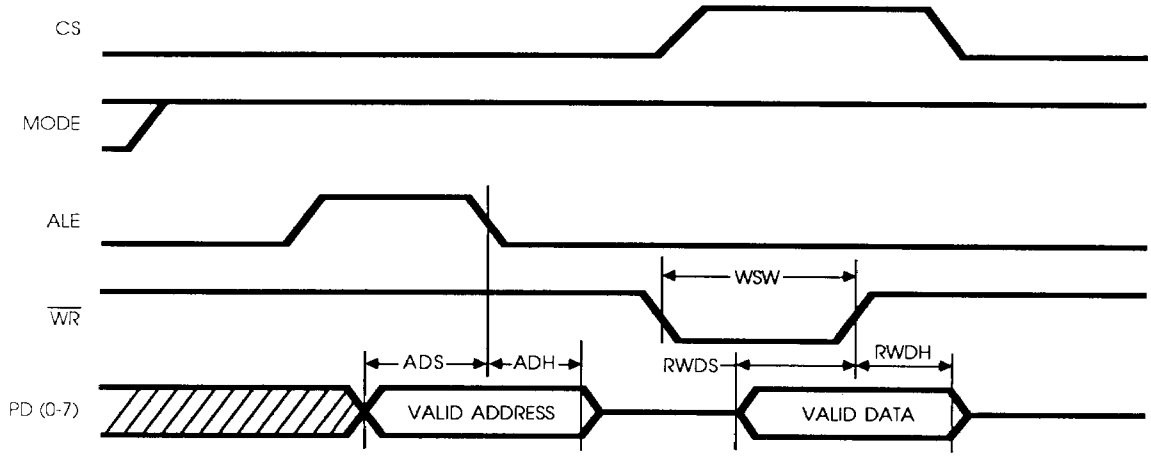
Output Timing Conditions:



CLOAD = 5opf

ALE Register Write

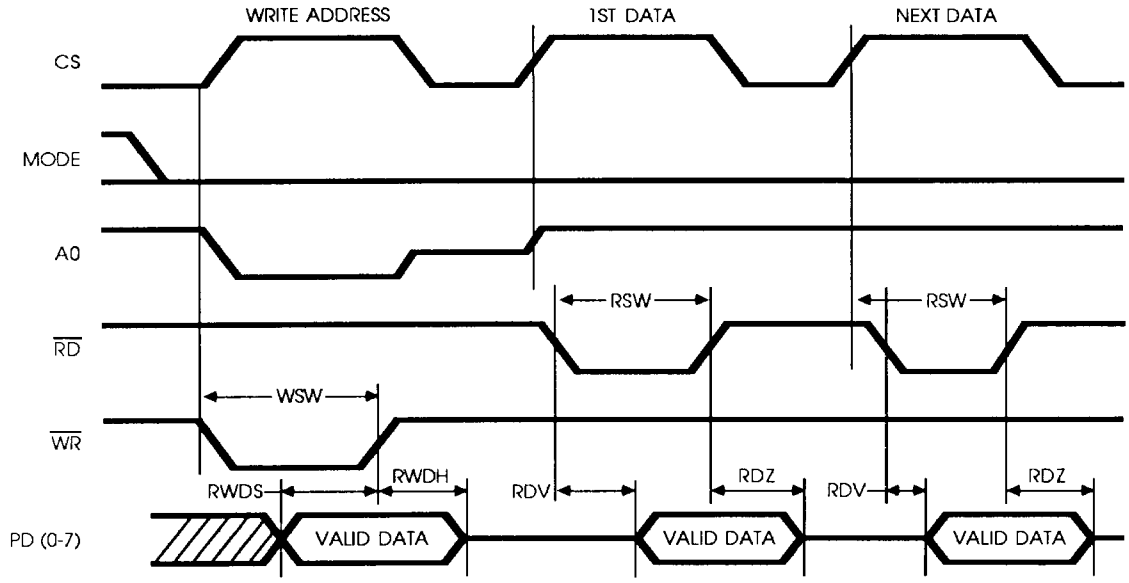
Multiplexed Address/Data Bus



SYMBOL	PARAMETER	MIN	MAX	UNITS
WSW	Write Strobe Width	100		ns
RWDS	Register Write Data Set Up	100		ns
RWDH	Register Write Data Hold	10		ns
ADS	Address Set Up To ALE ↓	50		ns
ADH	Address Hold from ALE ↓	15		ns

A0 Register Read

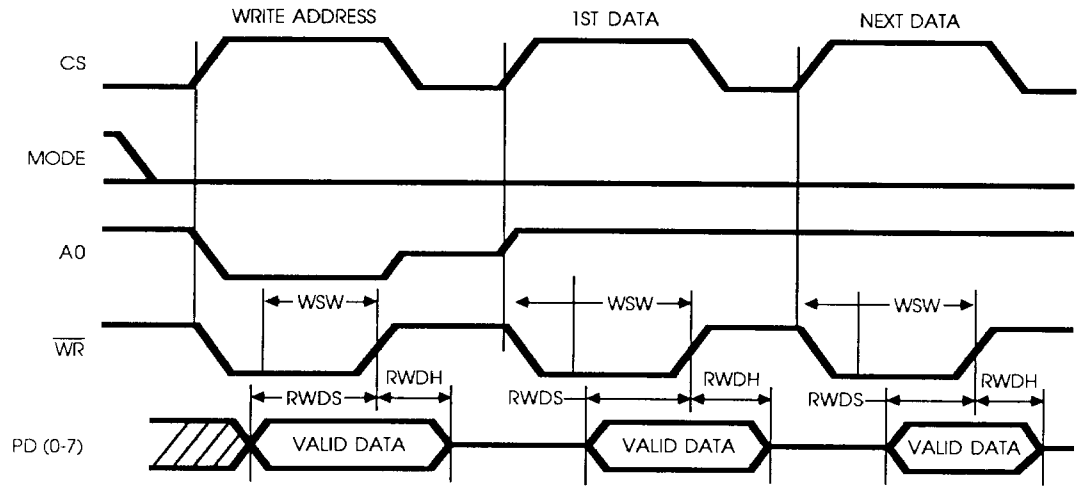
Nonmultiplexed Address/Data Bus



SYMBOL	PARAMETER	MIN	MAX	UNITS
WSW	Write Strobe Width	100		ns
RSW	Read Strobe Width	100		ns
RWDS	Register Write Data Set Up	100		ns
RDV	Read Data Valid		80	ns
RWDH	Register Write Data Hold	10		ns
RDZ	Read Strobe to Data Tri-State	10		ns

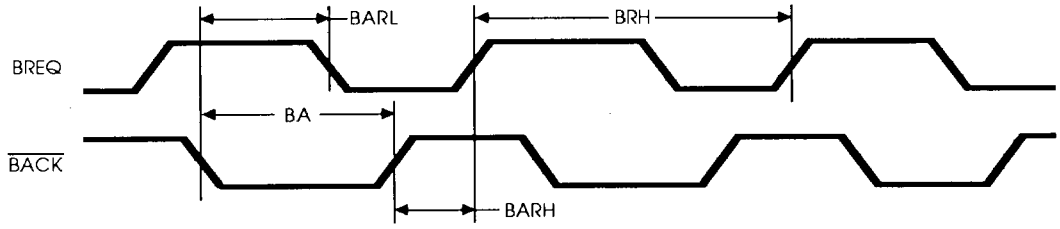
A0 Register Write

Nonmultiplexed Address/Data Bus



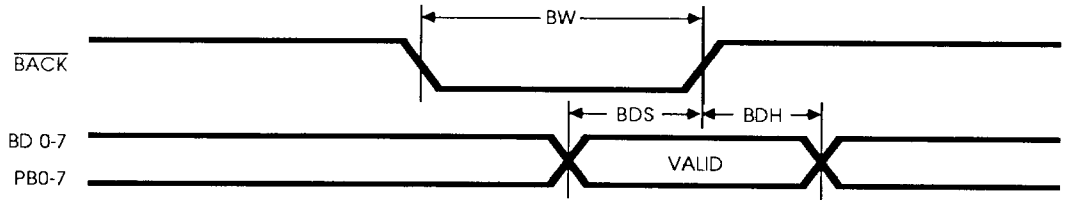
SYMBOL	PARAMETER	MIN	MAX	UNITS
WSW	Write Strobe Width	100		ns
RWDS	Register Write Data Set Up	100		ns
RWDH	Register Write Data Hold	10		ns

Buffer Read/Write Timing



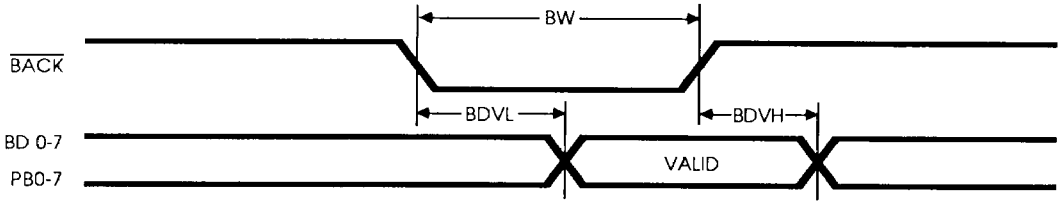
SYMBOL	PARAMETER	MIN	MAX	UNITS
BARL	$\overline{BACK}$ ↓ to BREQ ↓		20	ns
BARH	$\overline{BACK}$ ↑ to BREQ ↑		18	ns
BRH	BREQ ↑ to Next BREQ ↑	2T		
BA	$\overline{BACK}$ Pulse Width	50		ns

Data Transfer From Memory To The AIC-6250



SYMBOL	PARAMETER	MIN	MAX	UNITS
BDS	Buffer Data Set Up Time	10		ns
BDH	Buffer Data Hold Time	10		ns
BW	$\overline{\text{BACK}}$ Pulse Width	50		ns

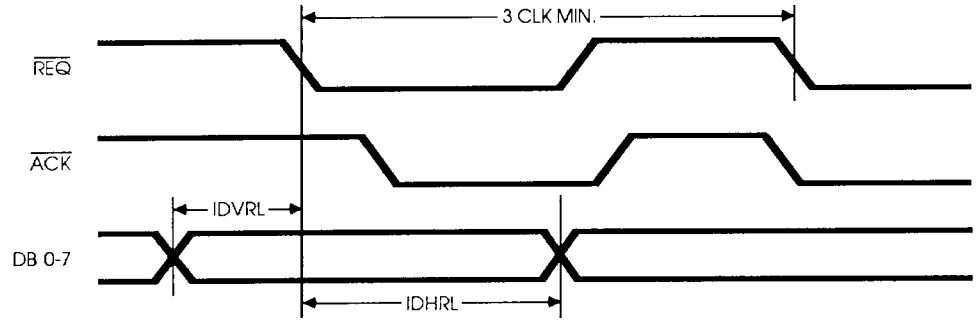
Data Transfer From The AIC-6250 To Memory



SYMBOL	PARAMETER	MIN	MAX	UNITS
BW	$\overline{\text{BACK}}$ Pulse Width	50		ns
BDVL	Buffer Data Valid from $\overline{\text{BACK}}$ ↓		40	ns
BDVH	Buffer Data Valid from $\overline{\text{BACK}}$ ↑		20	ns

SCSI Bus Timing (Initiator)

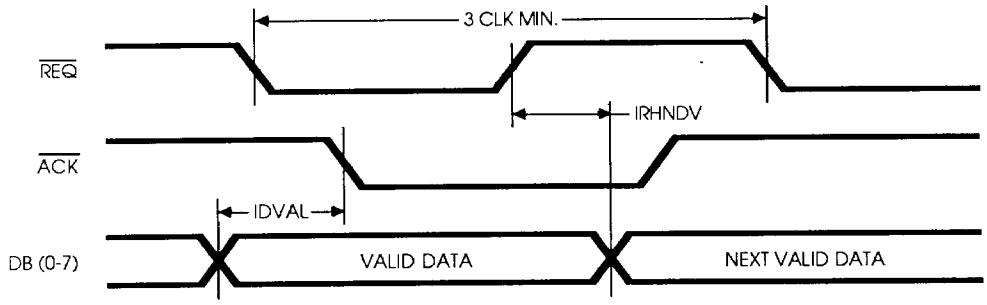
Data From SCSI



SYMBOL	PARAMETER	MIN	MAX	UNITS
IDVRL	Data Valid to REQ ↓	50		ns
IDHRL	Data Hold from REQ ↓	50		ns

SCSI Bus Timing (Initiator)

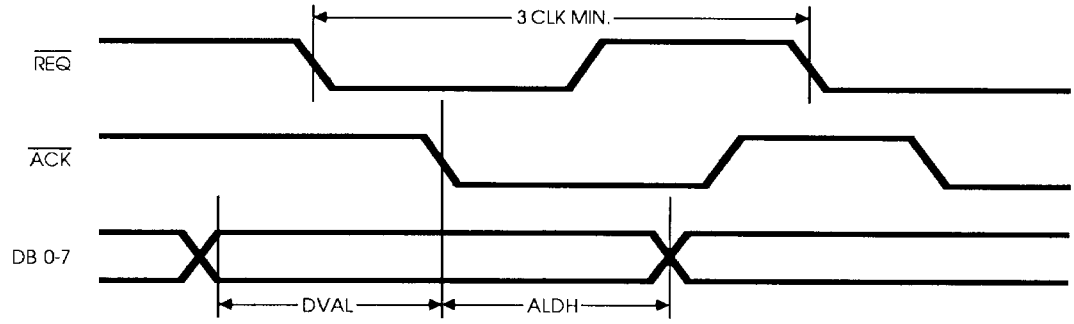
Data To SCSI



SYMBOL	PARAMETER	MIN	MAX	UNITS
IDVAL	Data Valid to ACK ↓	50		ns
iRHNDV	REQ ↑ to Next Data Valid	0		ns

SCSI Bus Timing (Target)

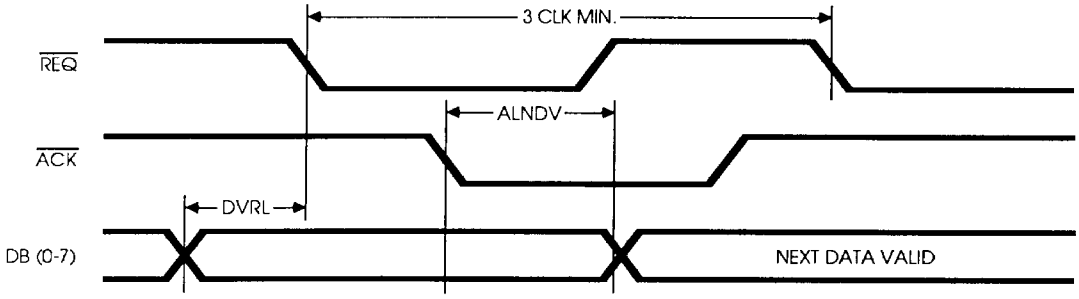
Data From SCSI



SYMBOL	PARAMETER	MIN	MAX	UNITS
ALDH	Data Hold from $\overline{ACK}$ ↓	50		ns
DVAL	Data Valid to $\overline{ACK}$ ↓	50		ns

SCSI Bus Timing (Target)

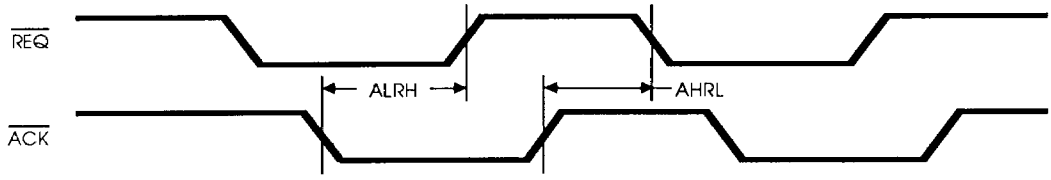
Data To SCSI



SYMBOL	PARAMETER	MIN	MAX	UNITS
DVRL	Data Valid to $\overline{REQ}$ ↓	50		ns
ALNDV	$\overline{ACK}$ ↓ to Next Data Valid	0		ns

SCSI $\overline{\text{REQ}}$ To $\overline{\text{ACK}}$ Timing

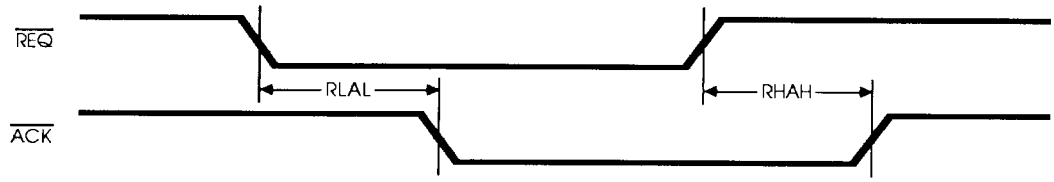
Target



SYMBOL	PARAMETER	MIN	MAX	UNITS
ALRH	$\overline{\text{ACK}}\downarrow$ to $\overline{\text{REQ}}\uparrow$	100 + T	100 + 2T	ns
AHRL	$\overline{\text{ACK}}\uparrow$ to $\overline{\text{REQ}}\downarrow$		40	ns

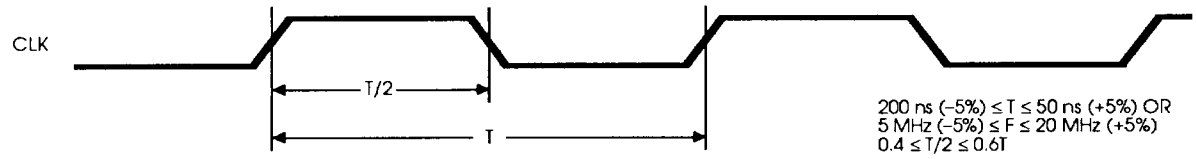
SCSI $\overline{\text{ACK}}$ To $\overline{\text{REQ}}$ Timing

Initiator



SYMBOL	PARAMETER	MIN	MAX	UNITS
RLAL	$\overline{\text{REQ}}\downarrow$ to $\overline{\text{ACK}}\downarrow$	100 + T	100 + 2T	ns
RHAH	$\overline{\text{REQ}}\uparrow$ to $\overline{\text{ACK}}\uparrow$		40	ns

Input Clock Timing



PACKAGING INFORMATION