NCR CORPORATION P/N: 348-00342 REV.A

F U N C T I O N A L

D E F I N I T I O N

TITLE

SCSI INTERFACE CHIP

July 30, 1982

TABLE OF CONTENTS

1.0 INTRODUCTION

1.1 PURPOSE

The purpose of this document is to functionally define the SCSI Interface Chip. The principal sections of this document contain the functional definition information. Helpful user information is provided in the appendices. This document assumes that the reader is knowledgeable of the SCSI bus specification and terminology.

1.2 GENERAL DESCRIPTION

The SCSI Interface Chip is designed to interface a variety of microprocessors to the Small Computer System Interface (SCSI) bus. The chip is housed in a 48-pin dual-in-line package. Below is a list of important features.

- o Efficient SCSI bus utilization.
- o Significant circuitry reduction.
- o Initiator and target roles.
- o Arbitration - multiple hosts and concurrent I/O.
- o Parity generation.
- o Optional parity checking.
- o DMA or programmed I/O transfers.
- o Programmable selection and reselection timeouts.
- o Internal transfer counter.
- o Operates with either single-ended or differential drivers/receivers.

2.0 INPUT/OUTPUT SIGNALS

The SCSI Interface Chip will be packaged in a 48-pin dual-in-line package. A signal summary list, signal definitions, and electrical requirements are included in this section. Note that a slash (/) following a signal name indicates that the signal is asserted in the low voltage state (active low).

2.1 SIGNAL SUMMARY

Microprocessor Side	SCSI Side	Others
CLK	ID0/-ID2/	Vcc
RESET	DB0-DB7,DBP	GND
D0-D7	BSYIN	
INT	BSYOUT	
WR/	SELIN	
RD/	SELOUT	
CS/	ATN	
A0-A3	ACK	
DREQ	REQ	
DACK/	MSG	
	C/D	
	I/O	
	IGS	
	TGS	
	DBEN/	
	ARB	

2.2 SIGNAL DEFINITIONS

Microprocessor Side

Name	I/O	Description
CLK	I	Symmetrical square wave signal which generates internal chip timing. Maximum frequency is 10MHz. Highest performance is attained by operation at maximum frequency.

2.0 INPUT/OUTPUT SIGNALS
2.2 SIGNAL DEFINITIONS

RESET	I	Reset input. When high, this signal forces the chip into a reset state. All current operations are terminated. Internal storage elements are cleared and self-diagnostics are performed.
D0-D7	I/O	Active High Data Bus to be connected to the microprocessor data bus.
INT	O	Interrupt request to microprocessor, set high for request and cleared when the chip is reset or the Interrupt Register is read.
WR/	I	Write pulse which strobes data into the selected register.
RD/	I	Read pulse which strobes the selected register onto the microprocessor data bus.
CS/	I	Chip select input. When low, this signal enables reading from or writing to the selected register.
A0-A3	I	Address inputs to be connected to five microprocessor address bus signals. These signals select which internal register will be read or written when RD/ or WR/ occur.
DREQ	O	Data request. When high, this signal indicates that the internal data register has a byte to transfer (when inputting from SCSI) or needs a byte to transfer (when outputting to SCSI). This signal becomes active only if the DMA mode bit in the command register is on. It is cleared when DACK/ becomes active.
DACK/	I	Data Acknowledge. When low, this signal resets DREQ and selects the data register for input or output. DACK/ acts as a chip select for the data register when in DMA mode.

SCSI Bus Side

Name	I/O	Description
ID0/-ID2/	I	These signals determine the three-bit code of

		the SCSI bus ID assigned to the chip. These signals are internally pulled up. Therefore, switches to GND can be connected directly to these inputs without external pull-up resistors.
DB0-7,DBP	I/O	Active High Data Bus to be connected to the SCSI drivers and receivers.
BSYIN	I	When high, this signal indicates to the chip that the SCSI BSY signal is active.
BSYOUT	O	When high, the chip is asserting the BSY signal to the SCSI bus.
SELIN	I	When high, this signal indicates to the chip that the SCSI SEL signal is active.
SELOUT	O	When high, the chip is asserting the SEL signal to the SCSI bus.
ATN	I/O	Initiator Role: The chip asserts this sig when the microprocessor requests ATTENTION condition or a parity error is detected in a byte received from the SCSI bus. Target Role: This signal is an input which indicates the state of the ATN signal on the SCSI Bus.
ACK	I/O	Initiator Role: The chip asserts this signal in response to REQ for a byte transfer on the SCSI bus. Target Role: This signal is an input which, when active, indicates a response to the REQ signal.
REQ	I/O	Initiator Role: This signal is an input which, when active, indicates that the Target is requesting a byte transfer on the SCSI bus. Target Role: Asserted by the chip to request a byte transfer on the SCSI bus.
MSG,C/D,I/O	I/O	Initiator Role: These signals are inouts which indicate the current SCSI bus phase. Target Role: The chip drives these signa to determine the bus phase.

IGS	O	Initiator Group Select. When high, this signal indicates to the SCSI drivers and receivers that the chip is operating in the Initiator Role. Its purpose is to enable drivers for ATN and ACK and receivers for REQ, MSG, C/D, and I/O.
TGS	O	Target Group Select. When high, this signal indicates to the SCSI drivers and receivers that the chip is operating in the Target Role. Its purpose is to enable drivers for REQ, MSG, C/D, and I/O and receivers for ATN and ACK.
DBEN/	O	Data Bus Enable. When low, this signal directly enables the SCSI Data Bus drivers.
ARB	O	Arbitration Phase. When active (high), this signal enables the external circuitry to place the I.D. bit on the SCSI bus for the arbitration phase. See Appendix D for suggested implementation.

Others

Name	I/O	Description
Vcc		+5V input.
GND		Signal reference input.

2.3 ELECTRICAL CHARACTERISTICS

2.3.1 OPERATING CONDITIONS

		min.	max.	units
Supply Voltage	VDD	4.75	5.25	volts
Supply Current	IDD	TBD		ma
Ambient Free Air Temperature	TA	5	70	C

2.3.2 INPUT SIGNAL REQUIREMENTS

Parameter	Conditions	min.	max.	units
High-level Input, VIH		2.0	5.75	volts
Low-level Input, VIL		-0.7	0.8	volts
High-level Input Current, IIH	VIH=5.25v		10.0	ua
Low-level Input Current, IIL	VIL=0v		-10.0	ua

2.3.3 OUTPUT SIGNAL REQUIREMENTS

All signals except DBEN/, IGS, and TGS

Parameter	Conditions	min.	max.	units
High-level Output Voltage, VOH	VDD=4.75V @ IOH=-400uA	2.4		volts
Low-level Output Voltage, VOL	VDD=4.75V @ IOL=2.0mA		0.4	volts

DBEN/, IGS, and TGS Signals

Parameter	Conditions	min.	max.	units
High-level Output Voltage, VOH	VDD=4.75V@ IOH=-400uA	2.4		volts

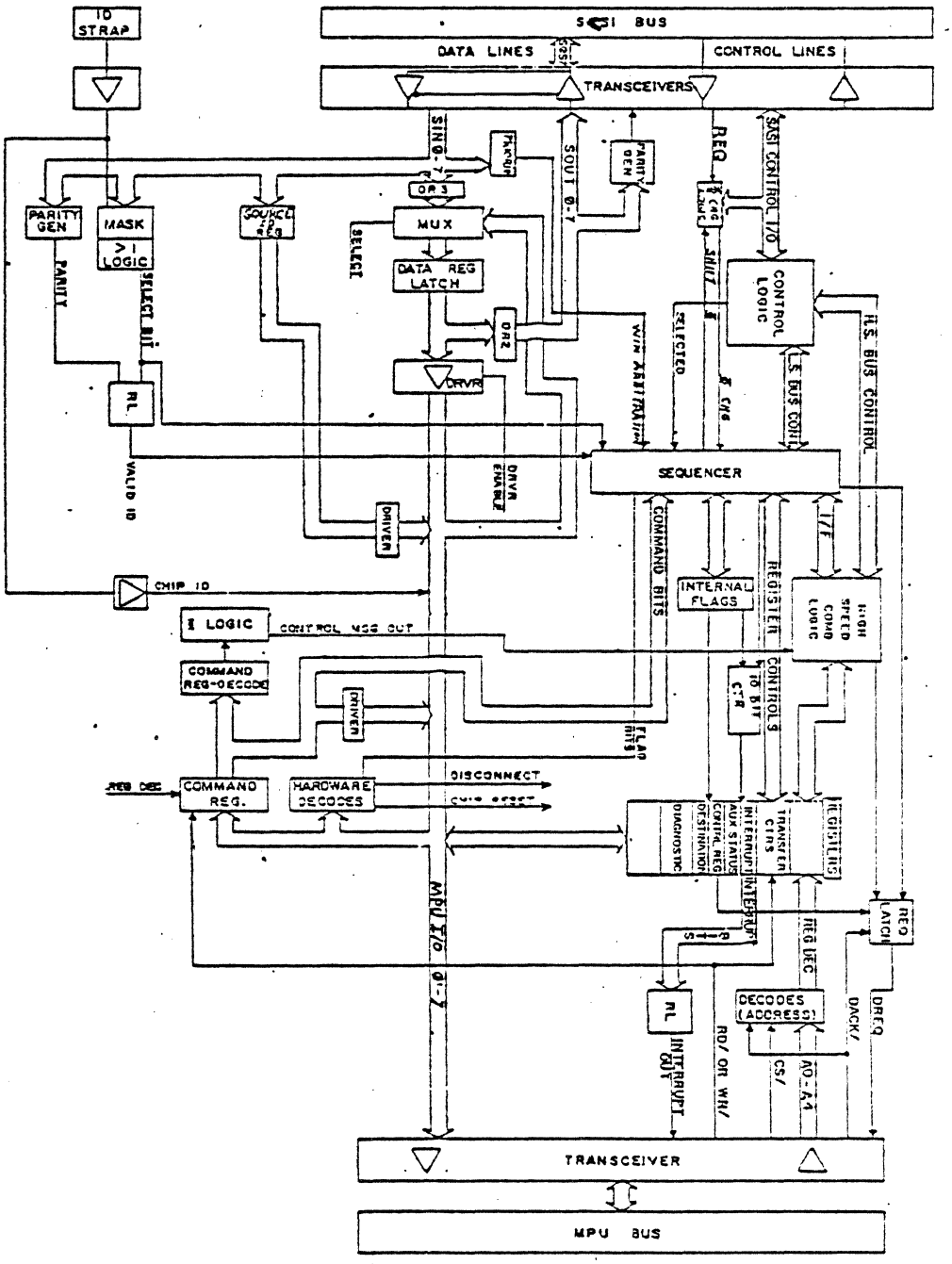
2.0 INPUT/OUTPUT SIGNALS
2.3.3 OUTPUT SIGNAL REQUIREMENTS

Low-level Output Voltage, VOL	VDD=4.75V@ IOL=4.0mA	0.4	volts

FUNCTIONAL DEFINITION
SCSI INTERFACE CHIP P/N: 348-0010342 Rev.A

3.0 BLOCK DIAGRAM

3.0 BLOCK DIAGRAM



4.0 REGISTERS

The SCSI interface chip is controlled by reading and writing registers provided inside the chip. These registers are read (written) by activating CS/ with an address on A4-A0 and then issuing a RD/ (WR/) pulse. They can be made to appear to a microprocessor as standard I/O ports or memory-mapped I/O ports depending on the external circuitry that controls CS/.

This section includes a summary that lists each register and its address. It also includes definitions for each register.

4.1 REGISTER SUMMARY

A4 A3 A2 A1 A0	R/W	REGISTER NAME
0 0 0 0 0	R/W	Data Register
0 0 0 0 1	R/W	Command Register
0 0 0 1 0	R/W	Control Register
0 0 0 1 1	R/W	Destination ID Register
0 0 1 0 0	R	Auxiliary Status
0 0 1 0 1	R	ID Register
0 0 1 1 0	R	Interrupt Register
0 0 1 1 1	R	Source ID Register
0 1 0 0 0		Not Used
0 1 0 0 1	R	Diagnostic Status
0 1 0 1 0		Not Used
0 1 0 1 1		Not Used
0 1 1 0 0	R/W	Transfer Counter (MSB)
0 1 1 0 1	R/W	Transfer Counter (2nd Byte)
0 1 1 1 0	R/W	Transfer Counter (LSB)
0 1 1 1 1		Not Used
1 0 0 0 0		Not Used
1 0 0 0 1		Not Used
1 0 0 1 0		Not Used
1 0 0 1 1		Not Used
1 0 1 0 0		Not used
1 0 1 0 1		Not Used
1 0 1 1 0		Not used
1 0 1 1 1		Not Used
1 1 0 0 0		Not Used
1 1 0 0 1		Not Used
1 1 0 1 0		Not Used
1 1 0 1 1		Not Used
1 1 1 0 0		Not Used
1 1 1 0 1		Not Used
1 1 1 1 0		Not Used
1 1 1 1 1		Not Used

NOTE: All locations labeled "Not Used" above are invalid for reading or writing. Any attempt to read or write these locations will not affect the operation of the chip. Writes will be ignored and reads may put garbage on the data bus.

4.2 REGISTER DEFINITIONS

4.2.1 DATA REGISTER

This eight bit register is used to transfer SCSI Command, Data, Status, and Message Phase bytes between the microprocessor data bus and the SCSI bus. In Non-DMA mode, the microprocessor reads from (writes to) the Data Register by activating CS/ with A4-A0 = 00000 and issuing a RD/ (WR/) pulse. A bit has been included in the Auxiliary Status Register to indicate when the Data Register is full. In DMA mode, the DMA logic reads from (writes to) the Data Register by responding to DREQ with DACK/ and issuing a RD/ (WR/) pulse. The SCSI bus reads from or writes to the Data Register when the chip is connected as an Initiator or Target and the bus is in one of the Information Transfer Phases.

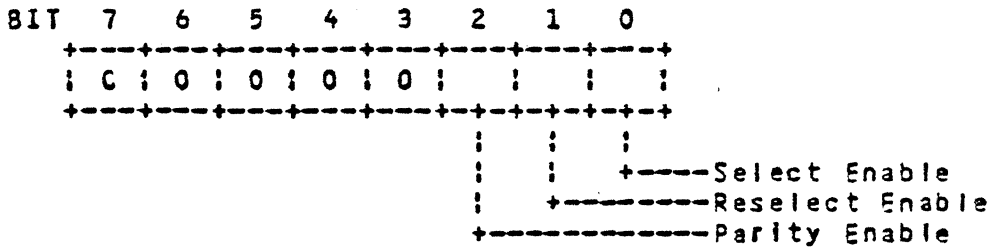
4.2.2 COMMAND REGISTER

The Command Register is an 8-bit register used to give commands to the SCSI interface chip. The microprocessor can write to (read from) the Command Register by activating CS/ with A4-A0 = 00001 and issuing a WR/ (RD/) pulse. Writing to the Command Register causes the chip to execute the command that is written. The Command Register can be read to determine what interrupting command was written last; the chip never alters the contents of the Command Register unless the chip is reset. Immediate commands are not stored.

The contents of the Command Register are defined in Section 5.0, Commands.

4.2.3 CONTROL REGISTER

This eight bit read/write register is used for enabling certain modes of operation for the SCSI interface chip. The microprocessor reads from (writes to) the Control Register by activating CS/ with A4-A0 = 00010 and issuing a RD/ (WR/) pulse.



Bit 2 Parity Enable — When the Parity Enable bit is a "1", the chip generates and checks parity on all transfers on the SCSI bus. When the Parity Enable bit is a "0", the chip generates, but does not check parity on bus transfers.

Bit 1 Reselect Enable — When this bit is a "1", the chip will respond to any attempt by a Target to reselect it. When the bit is a "0", the chip will ignore all attempts to reselect it.

Bit 0 Select Enable — When this bit is a "1" the chip will respond to attempts to select it as a Target. When it is a "0", the chip will ignore all selections.

Note: After being reset and completing self-diagnostics, the Control Register will contain all zeroes.

4.2.4 DESTINATION ID REGISTER

The Destination ID Register is an 8-bit register that is used to program the SCSI bus address of the destination device prior to issuing a SELECT or RESELECT command to the chip. Bits 0-2 specify the address, and bits 3-7 are always zeroes. The ID register can be either written or read. It is written (read) by activating CS/ with A4-A0 = 00110 and then pulsing WR/ (RD/).

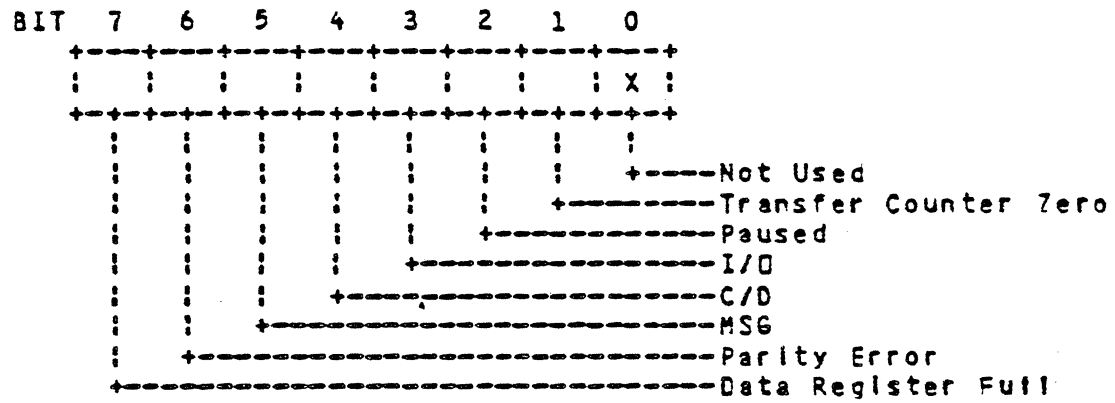
4.2.5 AUXILIARY STATUS REGISTER

The Auxiliary Status Register is an 8-bit, read-only register that contains bits that indicate the status of important operational conditions within the chip. Some of the bits in th register are needed to determine reasons for interrup

4.0 REGISTERS
4.2.5 AUXILIARY STATUS REGISTER

Therefore, the Auxiliary Status Register should always be read prior to reading the Interrupt Register when servicing interrupts. After the Interrupt Register is read, the Auxiliary Status Register bits needed to service the interrupt may change.

The Auxiliary Status Register is read by activating CS/ with A4-A0 = 00100 and then pulsing RD/. The individual bits of the Auxiliary Status Register are defined below.



Bit 7 Data Register Full

This bit indicates the status of the Data Register and must be monitored by the microprocessor during non-DMA mode commands that use the DATA Register. When the DMA Mode bit in the command register is off (0) and the command being executed is one of the SEND, RECEIVE, or TRANSFER INFO commands, data is transferred to (from) the chip by writing (reading) the Data Register. Data Register Full is set on (1) when data is written and turned off (0) when data is read. Therefore, Data Register Full should be on before taking data from the chip and off when sending data to the chip. The Data Register Full bit is always reset (to 0) at the time an interrupting type command is loaded into the Command Register in order to initialize it. Therefore, when issuing such commands, the Command Register should be loaded prior to monitoring Data Register Full and loading the Data

Register.

Bit 6 Parity Error — When on (1), this bit indicates that the chip has detected a parity error on a byte of data received over the SCSI bus. It can be set when the chip is executing one of the RECEIVE commands or the TRANSFER INFO command (when the transfer is an input). The Parity Error bit is reset whenever the Interrupt Register is read.

Bit 3-5 I/O, C/D, MSG — These bits indicate the status of the SCSI I/O, C/D, and MSG signals at the time the last REQ signal was received. They define the information phase type being requested by the target. These signals are significant when servicing interrupts and the chip is logically connected to the bus in the initiator role. The bits are coded as follows:

I/O	C/D	MSG	
0	0	0	Data Out
0	0	1	Unspecified Output Phase
0	1	0	Command
0	1	1	Message Out
1	0	0	Data In
1	0	1	Unspecified Input Phase
1	1	0	Status
1	1	1	Message In

Bit 2 Paused — When on (1), this bit indicates that the chip has aborted the command being executed in response to a PAUSE command. It is turned off when the next interrupting type command code is loaded into the Command Register.

Bit 1 Transfer Counter Zero — This bit is provided to indicate the status of the 24-bit Transfer Counter. When on (1), it indicates that the Transfer Counter is equal to zero. It is intended to facilitate interrupt servicing.

4.2.6 ID REGISTER

The ID Register is an 8-bit, read-only register that indicates the logical SCSI bus address occupied by the chip. Bits 0-2 directly reflect the logical inversion of the chip ID input signals ID0/-ID2/; the ID Register is active high whereas the ID input signals are active low. The ID Register allows the microprocessor to read the chip's SCSI bus address which would normally be strapped in hardware. Bits 3-7 of the ID Register will always be zeroes. The ID Register is read by activating CS/ with A4-A0 = 00101 and then pulsing RD/.

4.2.7 INTERRUPT REGISTER

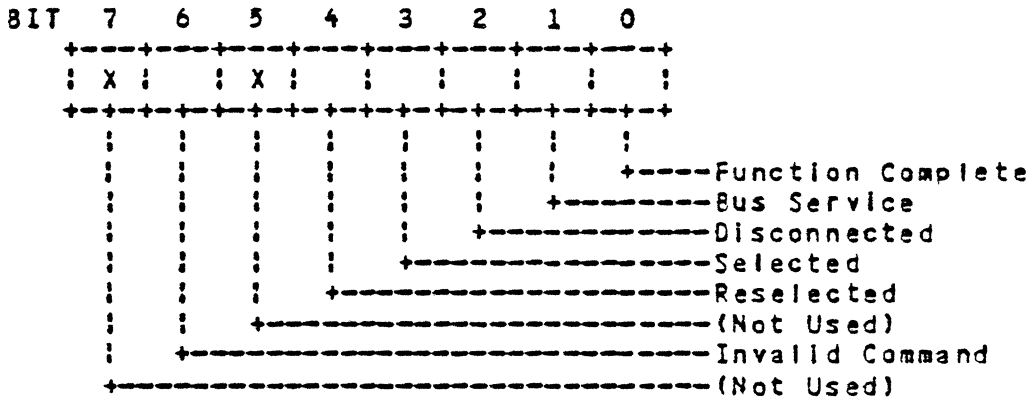
The Interrupt Register is an 8-bit register that contains the current interrupt(s). Along with the Auxiliary Status Register (Refer to Section 4.2.5), the Interrupt Register can be used by the microprocessor to determine the reason for an interrupt.

The Interrupt Register is a read-only register. It is read by activating CS/ with A4-A0 = 00011 and then pulsing RD/. When the Interrupt Register is read, it automatically resets itself (after the read is complete) and enables the chip to set a new interrupt. This also means that bits in the Auxiliary Status Register that are needed to service the interrupt may change after the Interrupt Register is read. Therefore, when an interrupt occurs, the microprocessor should always read the Auxiliary Status Register prior to reading the Interrupt Register.

Also, if a Selected or Reselected interrupt occurs after issuing a command that would normally cause an interrupt, the act of reading the Interrupt Register prevents the chip from executing that last command issued. This allows the microprocessor to service the Selected or Reselected interrupt prior to proceeding with the other operation. An example of this situation is when the microprocessor issues a command to select a target at about the same time another target reselects the chip. If the chip sees the reselection first, the microprocessor will receive an interrupt for the reselection, and the chip will ignore the select command, which would now be invalid since the chip is now logically connected on the SCSI bus to another device.

Individual interrupt conditions are defined below. Note that for all cases, an interrupt condition is on when the corresponding bit is a one (1) and off when zero (0).

4.0 REGISTERS
4.2.7 INTERRUPT REGISTER



Bit 6	Invalid Command	When on (1), this bit indicates that the last command loaded into the Command Register is not valid. Refer to Section 5.0, COMMANDS, for invalid command information.
Bit 4	Reselected*	This interrupt will be on (1) when the chip has been reselected by another SCSI device. After setting this interrupt, the chip is logically connected to the bus in the initiator role and is waiting for the target to send REQ or disconnect from the bus.
Bit 3	Selected*	This interrupt will be on (1) when the chip has been selected by another SCSI device. After setting this interrupt, the chip is logically connected to the bus in the target role and is waiting for a command to be loaded into the Command Register. * The chip will become selected (reselected) only if the ID data byte put on the SCSI bus during the selection (reselection) phase has good parity and not more than one ID bit other than the chip's ID is on.
Bit 2	Disconnected	This interrupt will be set on (1) when the chip is connected to the bus in

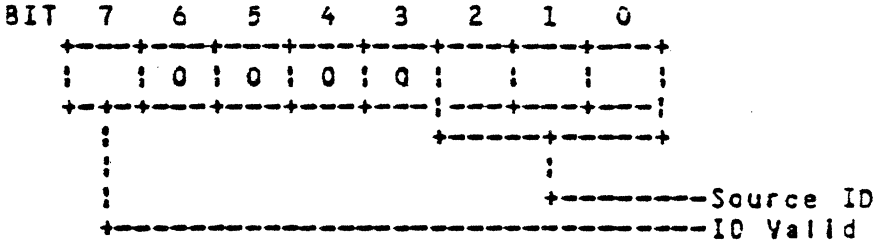
initiator role and the target disconnects or when the chip is executing a SELECT or RESELECT command and the destination device does not respond.

Bit 1 Bus Service — When the chip is logically connected to the bus in the initiator role, this bit will be set on (1) whenever the target sends a REQ which the chip cannot automatically handle. This happens when the first REQ for a connection is received or when the chip is executing a TRANSFER INFO or TRANSFER PAD command and either the transfer counter is zero or the target changes the information phase type. When the chip is logically connected in the target role, this bit will be set on (1), whenever the initiator asserts ATN. When indicating ATN, the Bus Service interrupt may occur by itself, with a Selected interrupt, or with a Function Complete interrupt.

Bit 0 Function Complete — When on (1), this bit indicates that the last interrupting command has completed. It is the normal successful completion interrupt for the SELECT commands, RESELECT, the RECEIVE commands, and the SEND commands. During any of the RECEIVE commands, it is set on, along with the Parity Error bit in the Auxiliary Register, as soon as a parity error is detected. A Bus Service interrupt may also occur simultaneously with the Function Complete if an ATN signal was activated during a SEND or a RECEIVE command.

4.2.8 SOURCE ID REGISTER

The Source ID Register is an eight bit read only register which contains the three bit encoded ID of the last device which selected or reselected the chip. Following is the format of the Source ID Register:

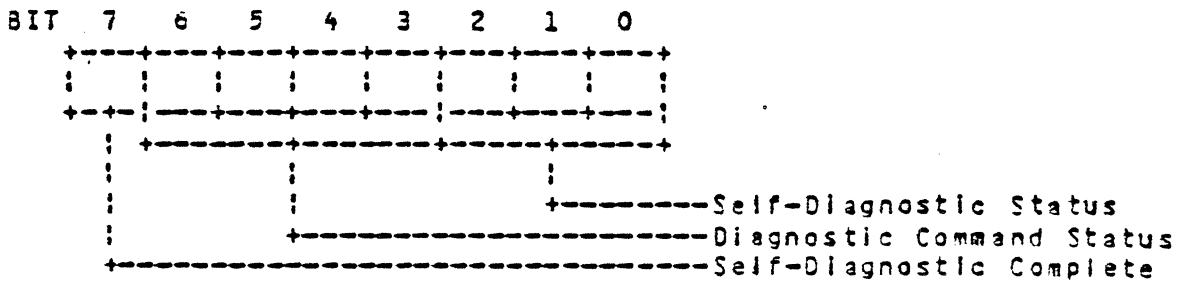


The ID Valid bit indicates that the source device placed its own ID bit on the SCSI bus during the Selection phase and therefore the chip has encoded that ID and placed it in bits 2-0. This information remains valid until the chip disconnects from the SCSI bus.

The microprocessor may read the Source ID Register by activating CS/ with A4-A0 = 00111 and issuing an RD/ pulse. Note: after reset and self diagnostics the Source ID Register contains all zeroes.

4.2.9 DIAGNOSTIC STATUS REGISTER

The Diagnostic Status Register is an 8-bit read-only register which indicates the result of self-diagnostics and the last diagnostic command issued to the chip. The format of the Diagnostic Status Register is shown below:



4.0 REGISTERS
4.2.9 DIAGNOSTIC STATUS REGISTER

Diagnostic Command Status

Bit 6543	Meaning
0000	(Not used)
0001	Turnaround Miscompare
0010	Turnaround Miscompare
0011	Turnaround Good Parity
0100	Turnaround Bad Parity
0101	(Not Used)
0110	(Not Used)
0111	(Not Used)
1000	(Not Used)
1001	(Not Used)
1010	(Not Used)
1011	(Not Used)
1100	(Not Used)
1101	(Not Used)
1110	(Not Used)
1111	(Not Used)

Self-Diagnostic Status

Bit 210	Meaning
000	Successful Completion
001	Unconditional Branch Failed
010	Data Register Full Failed
011	Initial Conditions Incorrect
100	Initial Command Bits Incorrect
101	Diagnostic Flag Failed
110	Data Turnaround Failed
111	(Not Used)

Bit 7 = 1 indicates that self-diagnostics have been completed. Note: A reset will clear bits 6-3 if possible. After a reset to the chip, the microprocessor should make sure that the Diagnostic Status Register contains the following pattern before attempting any commands: 10000000. This code indicates self-diagnostics are complete and no errors were detected. After a diagnostic command has been executed, bits 6-3 will contain the resulting status but bit 7 and bits 2-0 are not affected.

The microprocessor may read the Diagnostic Status Register by activating CS/ with A4-A0 = 01001 and issuing a RD/ pulse.

If an error is detected during self-diagnostics, the proper status is loaded into the Diagnostic Status Register and the chip halts until a Reset command or the Reset signal is asserted. Following is an explanation of the individual Self-Diagnostic Status codes:

- 000 - Successful completion. The chip executed all self-diagnostics following a reset and detected no errors.
- 001 - Unconditional Branch Failed. The chip internal sequencer attempted an unconditional branch and failed to reach the desired address.
- 010 - Data Register Full Failed. The chip attemoted to set and reset the Data Register Full status bit in the

Auxiliary Status Register and failed.

011 - Initial Conditions Incorrect. The chip detected one of its internal initial conditions in the wrong state.

100 - Initial Command Bits Incorrect. The chip tested bits 6,4,2,1 and 0 of the command register and found at least one that was not zero after reset.

101 - Diagnostic Flag Failed. The chip failed in its attempt to set and reset its internal diagnostic flag.

110 - Data Turnaround Failed. During self-diagnostics the chip attempts to flush several bytes of data through its internal data paths. It also attempts to set and reset the Parity Error bit in the Auxiliary Status Register. This status indicates that one of these operations failed.

When a diagnostic command is issued to the chip, the chip attempt to perform the function, load a status into bits 3-6, and initiate a Function Complete Interrupt.

4.2.10 TRANSFER COUNTER

The Transfer Counter is a 24-bit register/counter that is used by the chip for SEND, RECEIVE, and TRANSFER commands, that do not have single-byte specified, and for SELECT and RESELECT commands to set the timeout for no response. The Transfer Counter can be written and read. To write to (read from) the Transfer Counter, CS/ is activated with A4-A0 selecting a byte and then pulsing WR/ (RD/). The three bytes are selected by A4-A0 as shown below.

A4	A3	A2	A1	A0	SELECTED BYTE
0	1	1	0	0	Most Significant Byte
0	1	1	0	1	Middle Byte
0	1	1	1	0	Least Significant Byte

For SEND, RECEIVE, and TRANSFER commands with single-byte not specified, the Transfer Counter specifies to the chip the maximum number of bytes to be sent or received before interrupting. The Transfer Counter must be loaded prior to issuing the comman When single-byte is specified, the chip neither uses nor al

the Transfer Counter. To facilitate servicing interrupts for commands that use the Transfer Counter, a bit is provided in the Auxiliary Status Register to indicate when the Transfer Counter is zero.

For SELECT and RESELECT commands the Transfer Counter specifies the number of time intervals* that the chip will wait before automatically aborting the command due to no response (RSY) from the destination device. The Transfer Counter must be loaded prior to issuing the command. If the Transfer Counter is loaded with all zeroes, the time-out logic in the chip will be disabled, and the chip will not automatically abort the command due to no response.

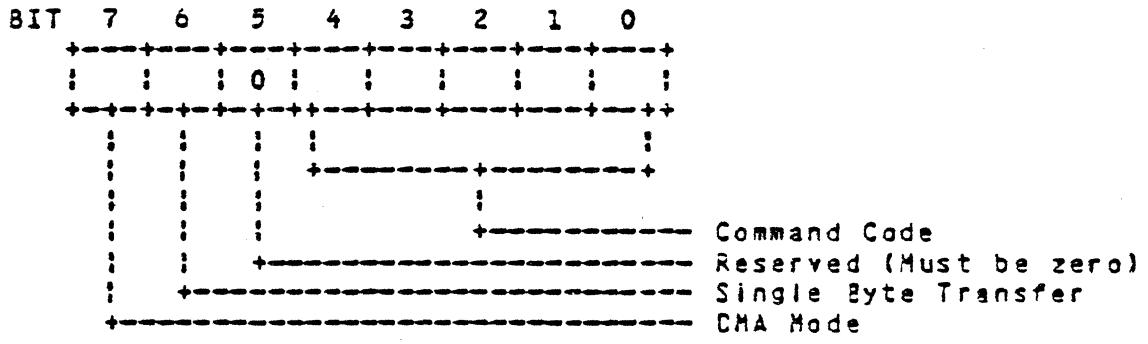
* Each time interval is equal to 1024 periods of the CLK input.

5.0 COMMANDS

This section defines command format, types, codes, and operation. Commands are given to the chip by loading the Command Register. Operation of the Command Register is explained in Section 4.2.2.

5.1 COMMAND FORMAT

The bits of the Command Register are defined as follows.



Bit7	DMA Mode	This bit is applicable only for commands that use the Data Register. When this bit is on (1), it indicates that data will be transferred to (from) the Data Register using the DMA signals DREQ and DACK/. When it is off (0), the microprocessor must monitor the state of the Data Register Full bit in the Auxiliary Status Register and, when it is in the proper state, transfer data by using the appropriate input/output command.
Bit 6	Single Byte Transfer	When on (1), this bit indicates that only one byte of data is to be transferred for this command. The Transfer Counter will not be used or altered by the chip. Therefore, for common single byte message and status transfers, the Transf Counter does not need to be loaded or

to issuing a command with this bit set. When this bit is off (0), the Transfer Counter is used by the chip to determine the length of the transfer for the command.

Bit 5 Reserved — This bit is not used and should be always programmed off (0).

Bits 0-4 Command Code — These bits specify the command to be executed by the chip. Command codes are specified in Section 5.4, Command Summary. Definitions for each command are in Section 5.5, Command Definitions.

5.2 COMMAND TYPES

There are two types of commands, immediate and interrupting. All except one of the immediate commands cause immediate results, within three clock cycles from the time the Command Register is loaded. The Pause Command is the exception and is explained in section 5.5.3. Interrupting commands do not result in immediate action. Their completion is always flagged by an interrupt.

Command codes 00000 - 00100 specify immediate commands. Command codes 00101 - 00111 are reserved for more immediate type commands. If issued, the chip will ignore these reserved codes. Command Codes 01000 - 10101 specify interrupting commands. When one of these codes is loaded into the Command Register, a second interrupting command code should not be loaded until after the interrupt has occurred for the first command. However, an immediate type command may be loaded before the interrupt for an interrupting command occurs. Command codes 10110 - 11111 are reserved for more interrupt causing commands. If these reserved codes are issued, the chip will respond with an Invalid Command interrupt.

5.3 INVALID COMMANDS

The user of the chip can be in one of three states at any particular time: disconnected, connected as an initiator, or connected as a target. Commands are valid only in specified states. If an immediate command is issued in an invalid state or a reserved immediate command code is issued, the chip will ignore the command. If an interrupting command is issued in an invalid

state or a reserved interrupting command code is issued, an Invalid Command interrupt will result.

5.4 COMMAND SUMMARY

Below is a summary that lists all commands. In the table the following abbreviations are used.

INT = Interrupting
IMM = Immediate
D = Disconnected
I = Connected as Initiator
T = Connected as Target

COMMAND CODE	COMMAND	TYPE	VALID USER STATES
00000	Chip Reset	IMM	D,I,T
00001	Disconnect	:	I,T
00010	Pause	:	D,T
00011	Set ATN	:	I
00100	Message Accepted	:	I
00101	Chip Disable	:	D,I,T
00110		:	
00111		V	
01000	Select W/ATN	INT	D
01001	Select W/O ATN	:	D
01010	Reselect	:	D
01011	Diagnostic Data Turnaround	:	D
01100	Receive Command	:	T
01101	Receive Data	:	T
01110	Receive Message Out	:	T
01111	Receive Unspecified Info Out	:	T
10000	Send Status	:	T
10001	Send Data	:	T
10010	Send Message In	:	T
10011	Send Unspecified Info In	:	T
10100	Transfer Info	:	I
10101	Transfer Pad	:	T
10110		:	
10111		:	
11000 - 11111		V	

5.5 COMMAND DEFINITIONS

5.5.1 CHIP RESET

Chip Reset immediately stops any chip operation and resets all registers, counters, etc. on the chip. It performs the same operation as the hardware "Reset" input (reference Section 2.1).

5.5.2 DISCONNECT

Upon receipt of this command, the chip immediately releases all SCSI bus signals and returns to a disconnected idle state. For the Target role, this is the normal method of disconnecting from the bus when a transfer is complete. For the Initiator role, Disconnect may be used to release the bus signals as a result of a timeout condition. In this case, the chip ignores the Target and is left in the disconnected state. When in the disconnected state, it is not valid to issue a Disconnect Command. If issued the chip will ignore the command.

5.5.3 PAUSE

Pause is an immediate command that is valid in the disconnected state or when logically connected to the bus in the target role. It is not valid when connected as an initiator.

When connected as a Target, the Pause command provides a means of halting a SEND or RECEIVE command without having to wait for the transfer to complete. When Pause is issued, it immediately sets a flag in the chip. Within one byte transfer cycle, the chip recognizes the flag, aborts the SEND or RECEIVE operation, and then sets the Paused status bit in the Auxiliary Status Register. At this time, the chip is still connected to the bus in the target role, and it is waiting for another command.

The PAUSE command stops the SEND or RECEIVE command in an orderly manner leaving the Transfer Counter in a valid state that indicates the remaining number of bytes to be transferred. Also no REQ or ACK is asserted on the bus and no data is left in the chip waiting to be transferred. An operation that is paused can be resumed, if desired, simply by reloading the original command into the Command Register. Note that after issuing a PAUSE while executing a SEND or RECEIVE, it is necessary to contin transferring data with the chip until the Paused status bit

set or an interrupt occurs.

When in the disconnected state, Pause may be issued to abort a SELECT or RESELECT command. After a SELECT or RESELECT command is issued and before an interrupt occurs, a Pause command may be issued to abort the operation. The Pause command immediately sets a flag within the chip. If the chip has not yet won arbitration, it sets the Paused auxiliary status bit and then waits in the disconnected state for another command. If the chip has won arbitration, it releases the bus by dropping the 2 ID bits with SELOUT on for a minimum of 100us, checks for no BSYIN, and then releases the bus. After this procedure, it sets the Paused auxiliary status bit and waits for another command in the disconnected state.

Since Pause is an immediate command, it does not cause an interrupt. As previously stated, the chip sets the Paused status bit to signify that it has been executed. It is noted, however, that if an interrupt causing event occurs before the chip sees the pause flag set, the chip will set the interrupt. In this case, the Paused status bit will not be set by the chip either before or after the interrupt. In all cases, an interrupt causing event will take precedence over Pause. For example, in the target role if ATN is on when Pause is issued, a Bus Service interrupt will occur and the Paused status bit will not be set.

If the Pause command is issued when the chip is idle, waiting for a command, the Paused status bit will be set by the chip.

5.5.4 SET ATN

The Set ATN Command causes ATN to be asserted immediately if the chip is connected as an Initiator. This command is invalid and ignored if issued when the chip is disconnected or is operating in a Target Role. The ATN signal is de-asserted in a Message Out phase when the transfer count becomes zero or one byte has been transferred (in a one-byte transfer command) during the execution of a Transfer Info command.

The chip automatically sets ATN in two cases:

1. If a Select w/ATN command is issued and Arbitration is won.
2. If a parity error is detected on an input byte during execution of a Transfer Info Command.

5.5.5 MESSAGE ACCEPTED

The Message Accepted command is an immediate command that is valid only when connected as an initiator. It is used after a Transfer Info or Pad, during which a message is input, to indicate to the chip that ACK can be deasserted for the last byte.

When an initiator receives a message, a Transfer command is used. If the transfer is an input and the information is a message (MSG = 1), the chip interrupts after receiving the last byte with a Function Complete interrupt. For this one special case, the chip also leaves ACK asserted on the bus. By interrupting and leaving ACK asserted, the chip gives the microprocessor a chance to interpret the message and set ATN prior to ACK being deasserted. This allows the chip to properly request a Message Out phase if the initiator wants to send a "Reject Message" message to the target.

Message Accepted must always be issued after a Transfer Info for a Message In phase, whether or not Set ATN is issued, to have the chip deassert ACK. If the initiator wants to reject the messa Set ATN would be issued first followed by Message Accepted. the message is not to be rejected, only Message Accepted is issued. Note, that until Message Accepted is issued, the target will not send another REQ since ACK is still asserted.

5.5.6 CHIP DISABLE

Chip Disable immediately stops all chip operations and logically disconnects it from the circuit. All outputs will be placed in a high impedance state and the chip will not respond to any commands (other than Chip Reset). The chip will also not respond to any activity on the SCSI Bus. The only way to exit this condition is to activate the Reset input signal or issue a Chip Reset command.

5.5.7 SELECT W/ATN

This command causes the chip to attempt to select a Target. It may only be used if the microprocessor is in the disconnected state. Any attempt to issue this command at another time will result in an Invalid Command interrupt. Before issuing this command the microprocessor must load the transfer counter for a timeout on the Target's response. This value is comput according to the formula below:

$$\text{Transfer Count} = \frac{\text{Desired Timeout}}{(1024)X(\text{Clock Period})}$$

where Desired Timeout and Clock Period are in the same units (e.g. seconds).

NOTE: A special case exists when the transfer counter is loaded with zero. In this case, the timeout is disabled and the chip will wait indefinitely for a response from the Target.

The microprocessor must also load the Destination ID register with the 3-bit code of the Target to be selected before issuing the Select w/ATN command.

When the chip detects the Select w/ATN command, it begins by attempting to arbitrate for control of the SCSI bus. IT, at any time during arbitration the chip becomes selected or reselected, the Select w/ATN command is aborted and forgotten and the chip will interrupt with one of the following conditions:

1. Selected
2. Selected and Bus Service
3. Reselected

If arbitration is won, the chip places the SCSI bus in the selection phase with ATN asserted and using the Destination ID register to identify the desired Target. At the same time, the chip begins a timer based on the value computed above. If the Target does not respond within the timeout period, the chip will disconnect from the bus and interrupt with Disconnected Interrupt status. (Note: the microprocessor should never monitor the Transfer Counter Zero bit in the Auxiliary Status Register to determine when a timeout has occurred.) If the Target responds within the allotted time, the chip will interrupt with Function Complete status. Control of the SCSI bus then belongs to the selected Target and after the interrupt status has been read, another interrupt may occur indicating either that the Target disconnected or is requesting a transfer.

If the timeout is disabled and the target does not respond or if arbitration is not won, the only way to abort the Select w/ATN command is to issue the Pause command. After the Pause command is issued, it is still possible that the "Function Complete" or "Disconnected" Interrupts may occur. This happens if one of the interrupts get set before the chip detects the Pause command or if the Target responds while the chip is sequencing off the SCSI

bus in a timeout condition. If the chip does not set either interrupt, it will set the 'Paused' bit in the Auxiliary Status register. If the microprocessor detects this bit after issuing the Pause command, then it is assured that the chip aborted the Select W/ATN and no connection exists.

5.5.8 SELECT W/O ATN

The Select w/o ATN command is identical to the Select W/ATN except that the ATN signal is not asserted during the Selection Phase.

5.5.9 RESELECT

This command causes the chip to attempt to reselect an Initiator. It may only be used if the microprocessor is in the disconnected state. Any attempt to issue this command at another time will result in an Invalid Command interrupt. Before issuing this command the microprocessor must load the transfer count for a timeout on the Initiator's response. This value computed according to the formula below:

$$\text{Transfer Count} = \frac{\text{Desired Timeout}}{(1024) \times (\text{Clock Period})}$$

where Desired Timeout and Clock Period are in the same units (e.g., seconds).

NOTE: A special case exists when the transfer counter is loaded with zero. In this case, the timeout is disabled and the chip will wait indefinitely for a response from the Initiator.

The microprocessor must also load the Destination ID register with the 3-bit code of the Initiator to be reselected before issuing the Reselect command.

When the chip detects the Reselect command, it begins by attempting to arbitrate for control of the SCSI bus. If at any time during arbitration, the chip becomes selected or reselected, the Reselect command is aborted and forgotten and the chip will interrupt with one of the following conditions:

1. Selected
2. Selected and Bus Service
3. Reselected

If arbitration is won, the chip places the SCSI bus in the Reselection Phase using the Destination ID register to identify the desired Initiator. At the same time, the chip begins a timer based on the value computed above. If the Initiator does not respond within the timeout period, the chip will disconnect from the bus and interrupt with Disconnected interrupt status. (Note: the microprocessor should never monitor the Transfer Counter Zero bit in the Auxiliary Status register to determine when a timeout has occurred.) If the Initiator responds within the allotted time, the chip will interrupt with Function Complete status. The chip (acting as the Target) is then in control of the SCSI bus, and waits for the Interrupt Register to be read by the microprocessor. After it has been read, the chip waits for a command from the microprocessor or ATN from the Initiator. If the ATN occurs, the chip will set the Bus Service Interrupt. This interrupt may happen immediately after a command has been issued due to internal timing. In this case, the chip waits for the Interrupt Register to be read and the command is ignored. The chip then waits for a new command.

If the timeout is disabled and the Initiator does not respond or if arbitration is not won, the only way to abort the Reselect command is to issue the Pause command. After the Pause command is issued, it is still possible that the Function Complete or Disconnect Interrupts may occur. This happens if one of the interrupts get set before the chip detects the Pause Command or if the Initiator responds while the chip is sequencing off the SCSI bus in a timeout condition. If the chip does not set either interrupt, it will set the Paused bit in the Auxiliary Status register. If the microprocessor detects this bit after issuing the Pause command, then it is assured that the chip aborted the Reselect and no connection exists.

5.5.16 DIAGNOSTIC DATA TURNAROUND

This interrupting command causes the chip to attempt to turn a data byte around through its internal data paths. When the command is loaded into the Command Register the Data Register Full bit is reset, after which the microprocessor writes one byte into the Data Register. The chip then moves the byte to another register and compares the contents with the Data Register. The byte is then moved to a third register (the SCSI output register) and good parity is generated if bit 6 of the command is off (0);

bad parity is generated if bit 6 is on (1). Finally, the chip moves the byte back to the Data Register and compares it with the contents of the second register. Based on these comparisions and parity checking, the chip stores a result into the Diagnostic Status Register and sets the Function Complete Interrupt. After reading the Interrupt Register, the microprocessor should make sure Data Register Full is on (1) and read the contents of the Data Register if the Data Register Full bit is not on, then an error has occurred. Following is a list of codes which are loaded into bits 6-3 of the Diagnostic Status Register as a result of this command.

Bit 6543	Meaning
0001	Data Miscompare (Initial)
0010	Data Miscompare (Final)
0011	Good Parity Detected
0100	Bad Parity Detected

5.5.11 RECEIVE COMMANDS

The Receive commands are interrupting commands that are valid only when connected to the bus in the target role. They are used by a target to receive command, data, and message information from an initiator.

The Receive commands transfer data; therefore, the Single Byte Transfer and DMA Mode bits in the Command Register are valid for these commands. If the Single Byte Transfer bit is off, the Transfer Counter must be loaded before a Receive command is issued to the chip. In this case, the chip uses the Transfer Counter to determine the number of bytes to receive.

When a Receive command is issued, the chip immediately resets the Data Register Full auxiliary status bit. The chip then drives the I/O, C/D, and MSG outputs for the proper information phase as follows.

Command Name	I/O	C/D	MSG
Receive Command	0	1	0
Receive Data	0	0	0
Receive Message Out	0	1	1
Receive Unspecified Output	0	0	1

The chip then proceeds to request and receive the specified number of information bytes. The DMA Mode bit in the command determines how the chip transfers these bytes from its Data Register to the microprocessor.

When a Receive command is terminated, the chip generates an interrupt. The following two events can cause termination.

1. The operation completes successfully; the Transfer Counter is zero. This event results in a Function Complete interrupt with the Parity Error auxiliary status bit off. If the initiator activated ATN during the operation, the Bus Service interrupt will also be on.

2. A parity error occurs. The last byte transferred is the byte that caused the error. This event causes a Function Complete interrupt with the Parity Error auxiliary status bit on. If the initiator activated ATN during the operation, the Bus Service interrupt will also be on.

After any of the interrupts, the chip is always left in the connected target state. The Transfer Counter indicates the number of bytes remaining to be transferred (zero if completed successfully), and the Data Register is empty (the last byte received is sent to the microprocessor). Also, ACK and REQ are inactive on the bus.

It is noted that, if a Bus Service interrupt alone occurs after issuing a Send command, the initiator activated ATN before the chip began executing the command. In this case, the command is ignored by the chip.

A Receive command may be stopped prior to an interrupt causing event by issuing a Pause command. Operation of the Pause command is explained in Section 5.5.3, Pause. In the event the initiator does not or stops responding, the chip is left in a state where it cannot respond to a Pause command. For this case, a Disconnect command can be used to abort the command and the connection. The Disconnect command is explained in Section 5.5.2.

5.5.12 SEND COMMANDS

The Send commands are interrupting commands that are valid only when connected to the bus in the target role. They are used by a target to send status, data, and message information to an initiator.

The Send commands transfer data; therefore, the Single Byte Transfer and DMA Mode bits in the Command Register are valid for these commands. If the Single Byte Transfer bit is off, the Transfer Counter must be loaded before a Send command is issued to the chip. In this case, the chip uses the Transfer Counter to determine the number of bytes to send.

When a Send command is issued, the chip immediately resets the Data Register Full auxiliary status bit. Therefore, the first byte of data for the transfer cannot be put into the Data Register until after a Send command is loaded into the Command Register.

In executing a Send command, the chip drives the I/O, C/D, and MSG outputs for the proper information phase. These lines are logically driven for each Send command as shown below.

Command Name	I/O	C/D	MSG
Send Status	1	1	0
Send Data	1	0	0
Send Message In	1	1	1
Send Unspecified Input	1	0	1

After resetting Data Register Full and driving I/O, C/D, and MSG, the chip then proceeds to monitor Data Register Full, take the data from the Data Register, and send it to the Initiator. The DMA Mode bit in the command specifies how the data is loaded into the chip.

After interrupting, the chip is left in the connected target state, and ACK and REQ are inactive on the bus. When the transfer is complete, the chip interrupts with a Function Complete interrupt. If the Initiator activated ATN during the transfer a Bus Service interrupt will also be set on by the chip.

It is noted that, if a Bus Service interrupt alone occurs after issuing a Send command, the Initiator activated ATN before the chip began executing the command. In this case, the command is ignored by the chip.

A Send command may be stopped prior to an interrupt causing event by issuing a Pause command. Operation of the Pause command is explained in Section 5.5.3, Pause. In the event the Initiator does not or stops responding, the chip is left in a state where it cannot respond to a Pause command. For this case, Disconnect command can be used to abort the command and

connection. The Disconnect command is explained in Section 5.5.2.

5.5.12 TRANSFER INFO

The Transfer Info command is an interrupting command that is valid only when connected to the bus in the initiator role. It is used by the initiator for all information transfers across the SCSI bus.

A Transfer Info command is issued by an initiator in response to a Bus Service interrupt. The Bus Service interrupt, as explained in Section 4.2.7, is received by the connected initiator upon receiving the first REQ from a target or when the chip terminates a previously issued Transfer Info or Transfer Pad command. It is not valid to issue a Transfer Info command without having a Bus Service interrupt because the target requests and controls all transfers. The chip will only permit one Transfer Info or Transfer Pad per Bus Service interrupt, and if a Transfer Info is issued before the first Bus Service interrupt occurs for a connection, the chip will issue an Invalid Command interrupt even though it is connected in the initiator role.

After an initiator receives a Bus Service interrupt and prior to issuing a Transfer Info command, the I/O, C/D, and MSG bits from the Auxiliary Status Register (read prior to reading the Interrupt Register) should be examined to determine the type of information phase and direction of transfer requested by the target. The initiator then prepares for the transfer. If the Single Byte Transfer bit is not going to be set in the command, the Transfer Counter must be loaded prior to issuing the Transfer Info command in order to specify to the chip the maximum number of bytes to be transferred.

When a Transfer Info is issued, the chip immediately resets the Data Register Full auxiliary status bit. For this reason, the first byte of data for an output operation cannot be loaded into the Data Register until after the command is loaded into the Command Register. The chip then proceeds with the transfer expecting data to be read from (input) or written to (output) its Data Register as indicated by the DMA Mode bit in the command. The chip automatically detects the direction of the transfer from the I/O bit which it stores in its Auxiliary Status Register.

The chip continues a transfer until an interrupt causing event occurs. The following four events will cause the chip to terminate and interrupt.

1. The maximum number of bytes specified have been transferred and the target activated REQ. This event results in a Bus Service interrupt. Either single byte transfer was specified or the Transfer Counter is zero as indicated by a bit in the Auxiliary Status Register. The target may or may not have changed the information phase type. The I/O, C/D, and MSG bits in the Auxiliary Status Register need to be examined at the time of the interrupt to determine this.

2. The target changes the information phase type before the maximum number of bytes are transferred. This event also causes a Bus Service interrupt. The new information phase can be determined by examining the I/O, C/D, and MSG bits in the Auxiliary Status Register. The Transfer Counter can be read to determine how many bytes remained to be transferred when the target changed the information phase type. When this interrupt occurs for an output transfer, the chip may take one more byte from the microprocessor than it transfers since it is prefetching; however, the Transfer Counter still reflects the number of bytes remaining to be transferred.

3. The target releases the bus by dropping BSY. This ev results in a Disconnected interrupt. Following this interrupt the chip is no longer in the initiator role. It is in the disconnected state.

4. The last byte of a Message Input phase has been received. This event results in a Function Complete interrupt. For this case, ACK is left active on the bus to allow the microprocessor to Set ATN for the purpose of rejecting the message. After this interrupt is received and a Set ATN is issued if desired, a Message Accepted must be issued to turn off ACK for the last byte of the Message In phase.

For input transfers (I/O = 1), the chip checks parity for each byte received if the Parity Enable bit in the Control Register is on. When checking parity and a parity error occurs, the chip activates ATN prior to deactivating ACK for the byte that causes the error. It also turns on the Parity Error auxiliary status bit. A parity error, however, does not result in an interrupt. The chip waits for one of the four events listed above before interrupting. Therefore, the Parity Error auxiliary status bit should be examined when servicing any interrupt after issuing a Transfer Info command for an input transfer.

If ATN is set on by the chip, either because of a parity error or because a SET ATN command is issued, the ATN will remain on un the end of the connection or until a control message out

transferred. Therefore, during each cycle of a Transfer Info operation for output, the chip checks for a message phase (C/D = 1, MSG = 1) and either single byte transfer specified or the Transfer Counter being zero. If these conditions exist, the chip turns off ATN prior to activating ACK for the last byte of the control message.

As previously stated, a Transfer Info normally terminates with an interrupt. If a Transfer Info command must be aborted, possibly because of a timeout violation, either a CHIP RESET or a DISCONNECT command can be used. (Refer to Sections 5.5.1 and 5.5.2, respectively.) It is noted, however, that although these commands will force the chip into a logically disconnected state, the target device is left on the bus. A SCSI bus reset, which is not a chip function, is the only way an initiator can force a target to disconnect.

5.5.14 TRANSFER PAD

The Transfer Pad command is an interrupting command that is valid only when connected to the bus in the initiator role. It is just like the Transfer Info command (Section 5.5.11) except that the data transferred between the chip and the microprocessor bus is different.

Transfer Pad can be used by an initiator to continue handshaking with a target without giving data to or taking data from the chip. The chip operates in the same manner as it does for a Transfer Info except that for output transfers it takes only one byte of data from the microprocessor and then sends that same byte repeatedly until the transfer terminates. For input transfers it accepts data from the SCSI bus but does not check parity or send it to the microprocessor. Even though data is not exchanged with the microprocessor bus, the Transfer Counter is still used by the chip so that a maximum number of pad bytes can be specified.

Protocol for using a Transfer Pad is the same as the protocol for a Transfer Info except that the DMA Mode bit has significance only for output transfers. The Transfer Pad terminates upon the same four events that cause a Transfer Info to terminate. Also similar to Transfer Info, CHIP RESET and DISCONNECT can be used to abort the command.

6.0 BUS INITIATED FUNCTIONS

6.1 SELECTION

If the Select Enable bit in the Control Register is on, the chip may be selected by another SCSI device to be a target for an I/O operation. Selection occurs in the chip only if SELOUT = 0, SELIN = 1, BSYIN = 0, I/O = 0, the chip's ID bit is asserted by the selecting device on the data bus, no more than one other ID bit (the initiator's) is asserted on the data bus, and data bus parity is good.

When all of these conditions exist, the chip is selected. It then encodes the initiator's ID and loads it into the Source ID Register. The chip also detects whether or not the initiator asserted its ID during selection and either sets or resets the Valid bit in the Source ID Register.

The chip then activates BSYOUT, waits for SELIN to turn off, and proceeds to take one of the following actions as a result of being selected:

1. If ATN is not asserted by the initiator during selection, the chip generates a Selected interrupt indicating that the chip is connected as a target.

2. If ATN is asserted, the chip simultaneously generates Selected and Bus Service interrupts indicating that the chip is connected as a target and ATN is asserted.

6.2 RESELECTION

If the Reselect Enable bit in the Control Register is on, the chip may be reselected by a SCSI target device. Reselection occurs in the chip only if SELOUT = 0, SELIN = 1, BSYIN = 0, I/O = 1, the chip's ID bit and the target's ID bit are asserted by the target on the data bus, no other ID bits are asserted, and data bus parity is good.

When all of these conditions exist, the chip is reselected. It then encodes the target's ID and loads it into the Source ID Register. The chip also sets the ID Valid bit in the Source ID Register.

The chip then asserts BSYOUT and waits for SELIN to be deasserted by the target. When the chip detects SELIN = 0, it deasserts BSYOUT and then generates a Reselected interrupt.

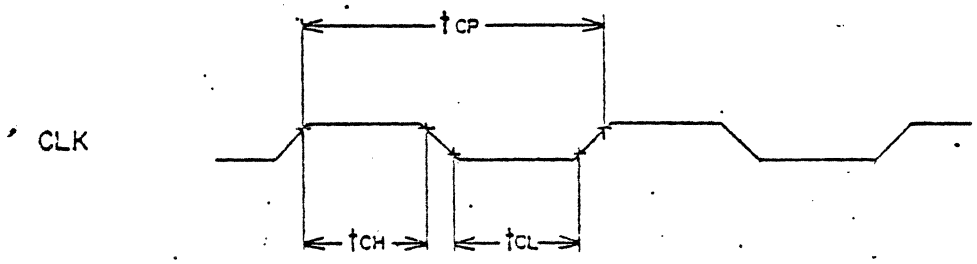
Reselection is now complete and the chip is in the connected initiator state.

7.0 EXTERNAL CHIP TIMING

Timing requirements must be met over the operating temperature (0 - 70 C) and voltage (4.75 - 5.25 V) ranges. Loading for all output signals except DBEN/ is assumed to be four low-power Schottky inputs including 50 pF capacitance. Loading for DBEN/ is assumed to be ten low-power Schottky inputs including 100pF capacitance.

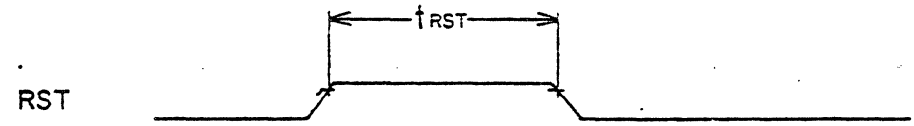
7.1 MICROPROCESSOR INTERFACE

7.1.1 CLK



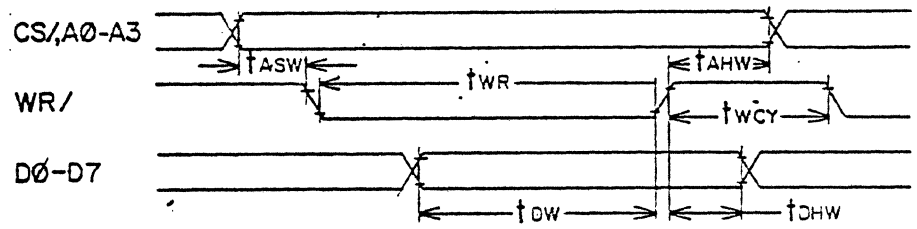
NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tCP	CLOCK PERIOD	100		200	NS
tCH	CLOCK HIGH	.45TCP		.55TCP	
tCL	CLOCK LOW	.45TCP		.55TCP	

7.1.2 RESET



NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tRST	RESET PULSE WIDTH	100			NS

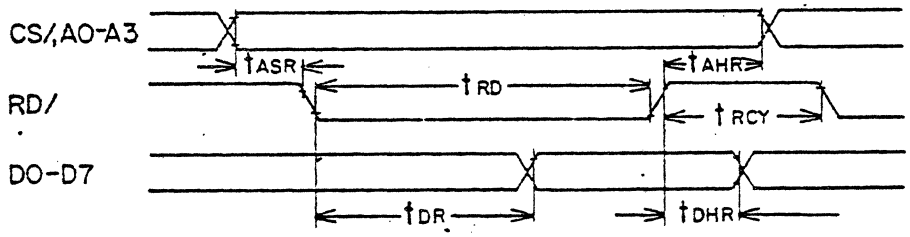
7.1.3 MPU WRITE



NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tASW	ADDRESS SET-UP TIME	0			NS
tWR	WR/ PULSE WIDTH	95			NS
tDW	DATA-TO-WR/ HIGH	50			NS
tAHW	ADDRESS HOLD TIME	0			NS
tDHW	DATA HOLD TIME	20			NS
tWCY	WR/ OFF TO WR/ OR RD/ ON	125			NS

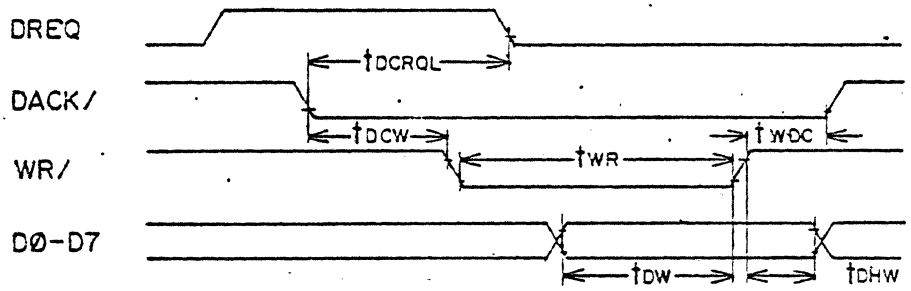
7.0 EXTERNAL CHIP TIMING
7.1.4 MPU READ

7.1.4 MPU READ



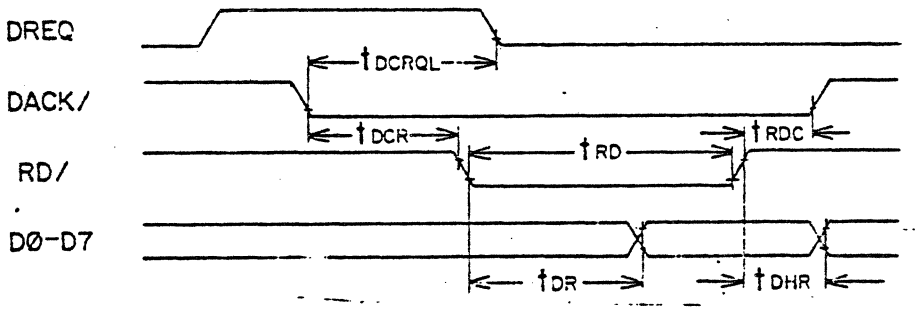
NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tASR	ADDRESS SET-UP TIME TO RD/	0			NS
tRD	RD/ PULSE WIDTH	125			NS
tDR	RD/ TO DATA			90	NS
tAHR	ADDRESS HOLD TIME	0			NS
tDHR	DATA HOLD TIME	10			NS
tRCY	RD/ OFF TO RD/ OR WR/ ON	125			NS

7.1.5 DMA WRITE



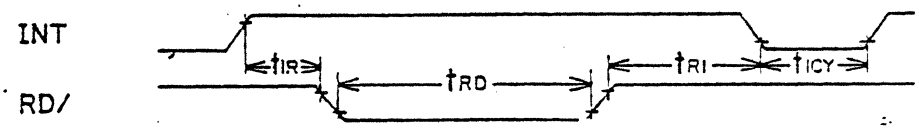
NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tDCRQL	DACK/ TO DREQ LOW	0		40	NS
tDCW	DACK/ TO WR/	0			NS
tWR	WR/ PULSE WIDTH	95			NS
tWDC	WR/ HIGH TO DACK/ HIGH	0			NS
tDHW	DATA HOLD TIME	20			NS
tDW	DATA TO WR/ HIGH	50			NS

7.1.6 DMA READ



NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tDCRQL	DACK/ TO DREQ LOW	0		40	NS
tDCR	DACK/ TO RD/	0			NS
tRD	RD/ PULSE WIDTH	95			NS
tRDC	RD/ HIGH TO DACK/ HIGH	0			NS
tDHR	DATA HOLD TIME	0			NS
tDR	RD/ TO DATA			80	NS

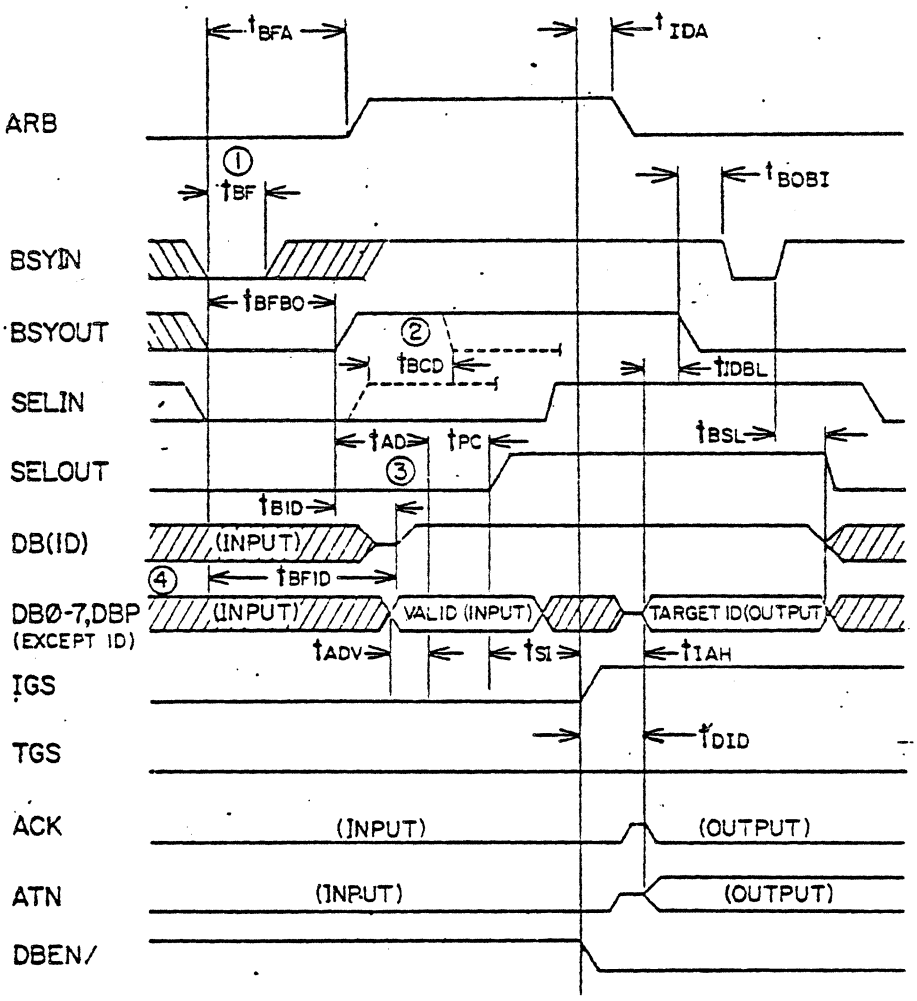
7.1.7 INTERRUPT



NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tIR	INT TO RD/	0			NS
tRD	RD/ PULSE WIDTH	95			NS
tRI	RD/ HIGH TO INT LOW			125	NS
tICY	INT OFF TO INT ON	125			NS

7.2 SCSI INTERFACE

7.2.1 SELECTION (INITIATOR)



NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tBF	BUS FREE	100			NS
tBFA	BUS FREE TO ARB HIGH	100		800	NS
tIDA	IGS & DBEN/ ACTIVE TO ARB LOW	0			NS
tBFBC	BUS FREE TO BSYOUT	100		800	NS
tBCD	BUS CLEAR DELAY			225	NS
tAD	ARBITRATION DELAY	2.0			uS
tPC	PRIORITY CHECK TO SELOUT	0			NS
tBID	BSYOUT HIGH TO ID BIT HIGH	100			NS
tBFID	BUS FREE TO ID HIGH			900	NS
tADV	ARBITRATION DATA VALID TO PRIORITY CHECK	0			NS
tSI	SELOUT TO IGS & DBEN/	950			NS
tIAH	IGS HIGH TO ATN HIGH	85			NS
tIDBL	TARGET ID & ATN HIGH TO BSYOUT LOW	100			NS
tBOBI	BSYOUT LOW TO BSYIN LOW	0		450	NS
tBSL	BSYIN HIGH TO SELOUT LOW	100			NS
tDID	DBEN/ ACTIVE TO BUS ENABLED	65			NS

NOTES:

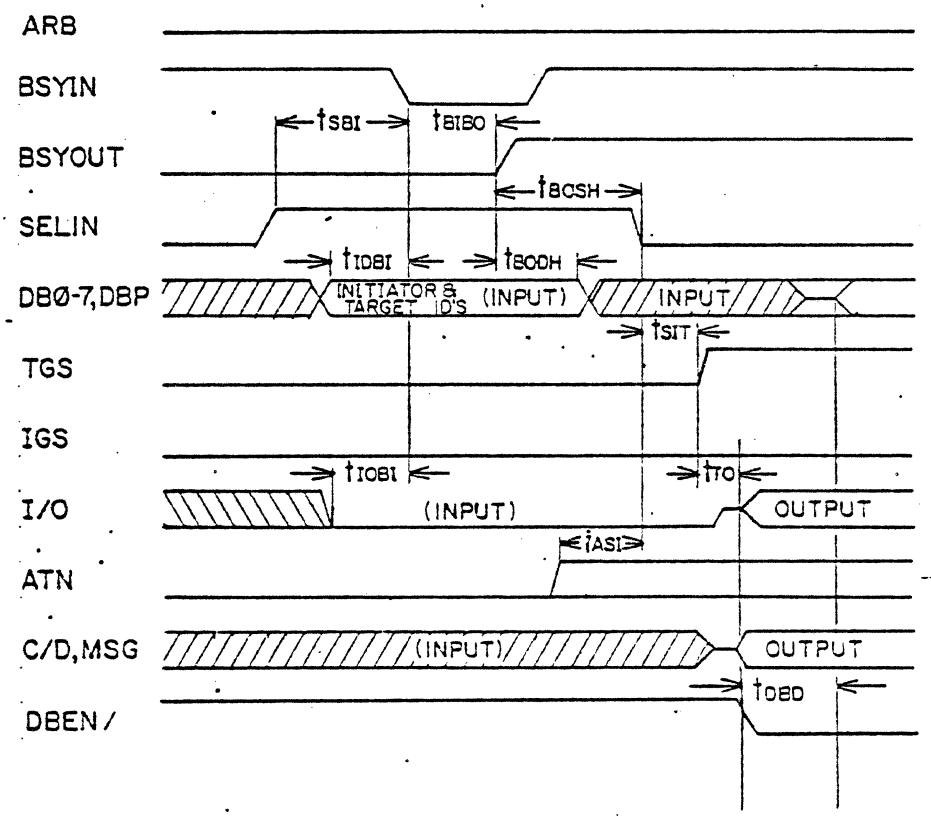
1. The chip ensures that the bus remains free (BSYIN and SELIN inactive) for tBF before attempting arbitration.

2. If SELIN becomes active at any time during arbitration, the chip must deassert BSYOUT within tBCD.

3. The chip waits tAD and then checks to see if arbitration is won (tPC). The chip then asserts SELOUT if arbitration is won.

4. One of the Data Bits is assigned as an ID bit by the IDO1-ID2/ signals. During Bus Free, the chip places all of the Data Bits (including ID) in a high impedance state. During arbitration the chip enables its ID bit and drives it high, but the remainder of the Data Bits remain in the high impedance state for reading.

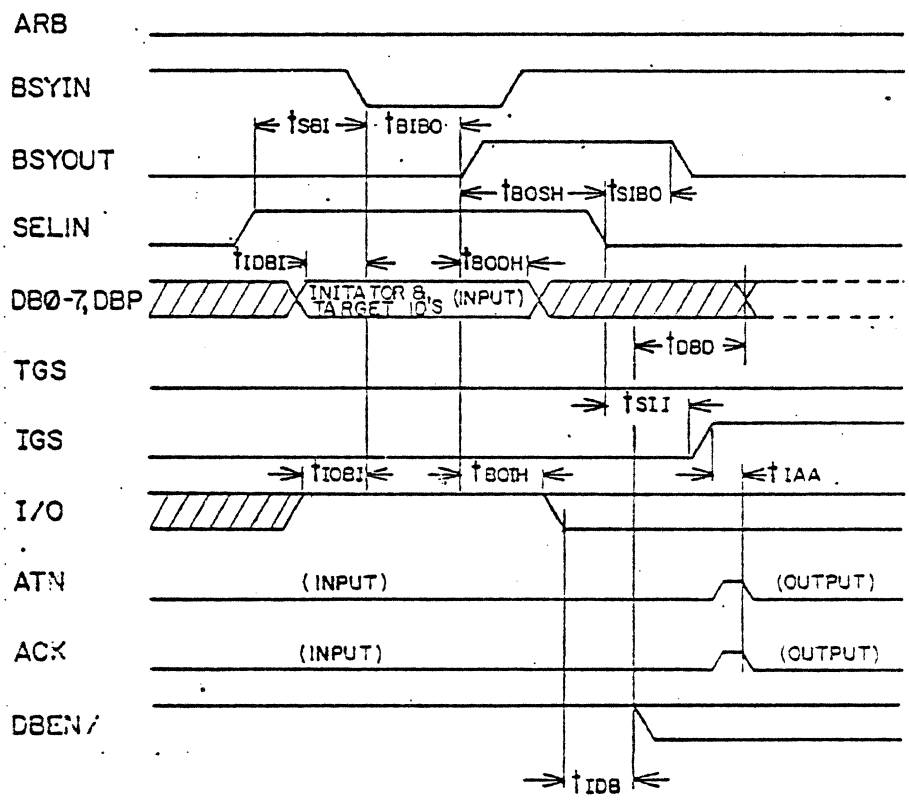
7.2.2 SELECTION (TARGET)



7.0 EXTERNAL CHIP TIMING
7.2.2 SELECTION (TARGET)

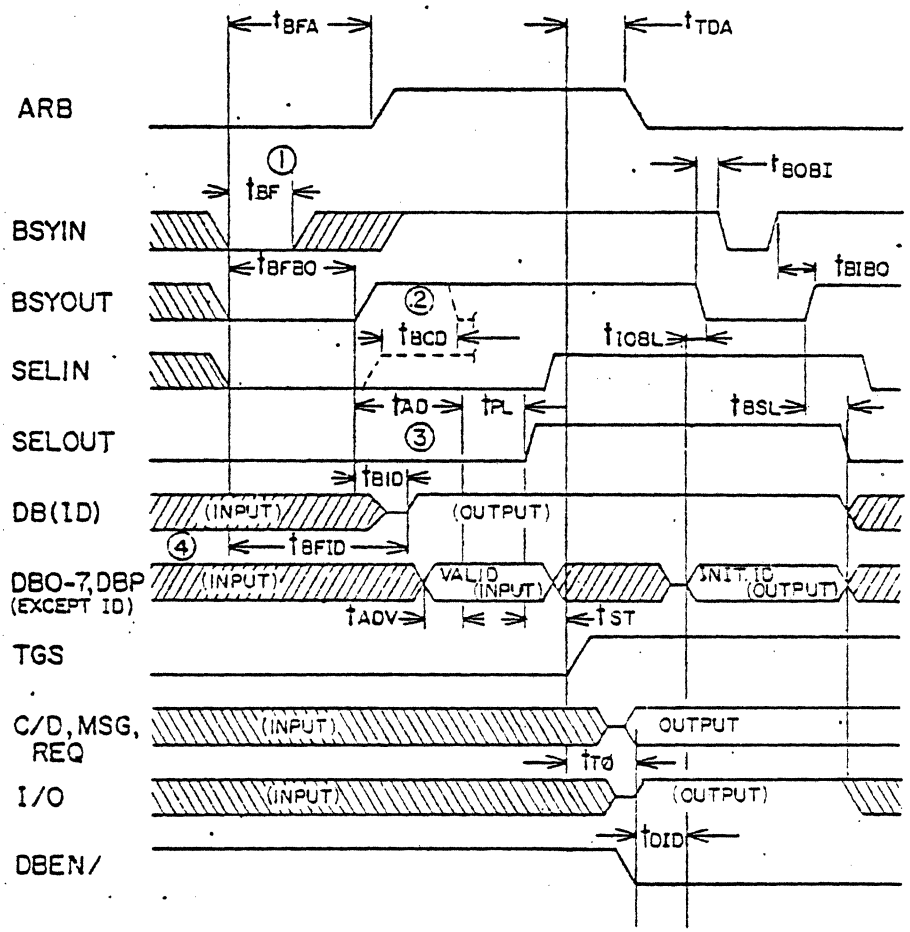
NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tSBI	SELIN HIGH TO BSYIN LOW	50			NS
tIDBI	INITIATOR & TARGET ID'S VALID TO BSYIN LOW	0			NS
tIOBI	I/O LOW TO BSYIN LOW	0			NS
tBIBC	BSYIN LOW TO BSYOUT HIGH	0		2	uS
tBOSH	BSYOUT HIGH SELIN HOLD	0			NS
tBODH	BSYOUT HIGH DATA HOLD	0			NS
tASI	ATN HIGH TO SELIN LOW	0			NS
tSIT	SELIN LOW TO TGS HIGH	0			NS
tTO	TGS HIGH TO PHASE SIGNALS DRIVEN	85			NS
tDBD	DBEN/ LOW TO DATA BUS ENABLED	85			NS

7.2.3 RESELECTION (INITIATOR)



NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tSBI	SELIN HIGH TO BSYIN LOW	50			NS
tIDBI	INITIATOR & TARGET ID'S VALID TO BSYIN LOW	0			NS
tIOBI	I/O LOW TO BSYIN LOW	0			NS
tBIBO	BSYIN LOW TO BSYOUT HIGH	0		2	uS
tBOSH	BSYOUT HIGH SELIN HOLD	0			NS
tBODH	BSYOUT HIGH DATA HOLD	0			NS
tBOIH	BSYOUT HIGH I/O HOLD	0			NS
tSIBO	SELIN LOW TO BSYOUT LOW	0			NS
tSII	SELIN LOW TO IGS HIGH	0			NS
tIAA	IGS HIGH TO ACK & ATN ENABLED	85			NS
tIDB	I/O LOW TO DBEN/LOW	0			NS
tDBD	DBEN/ LOW TO DATA BUS ENABLED	85			NS

7.2.4 RESELECTION (TARGET)



7.0 EXTERNAL CHIP TIMING
7.2.4 RESELECTION (TARGET)

NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tBF	BUS FREE	100			NS
tBFA	BUS FREE TO ARB HIGH	100		800	NS
tTDA	TGS & DBEN/ ACTIVE TO ARB LOW	0			NS
tBFBC	BUS FREE TO BSYOUT	100		800	NS
tBCD	BUS CLEAR DELAY			225	NS
tAD	ARBITRATION DELAY	2.0			uS
tPC	PRIORITY CHECK TO SELOUT	0			NS
tBID	BSYOUT HIGH TO ID BIT HIGH	100			NS
tBFID	BUS FREE TO ID HIGH			900	NS
tADV	ARBITRATION DATA VALID TO PRIORITY CHECK	0			NS
tST	SELOUT TO TGS	950			NS
tTO	TGS HIGH TO PHASE ENABLED & DBEN/ LOW	85			NS
tDID	DBEN/ LOW TO BUS ENABLED	85			NS
tIDBL	INITIATOR ID HIGH TO BSYOUT LOW	100			NS
tBOBI	BSYOUT LOW TO BSYIN LOW	0		450	NS
tBIBC	BSYIN HIGH TO BSYOUT HIGH	0		2	uS

tBSL	BSYOUT HIGH TO SELOUT LOW (I/O AND DATA CHANGE)	100			NS

NOTES:

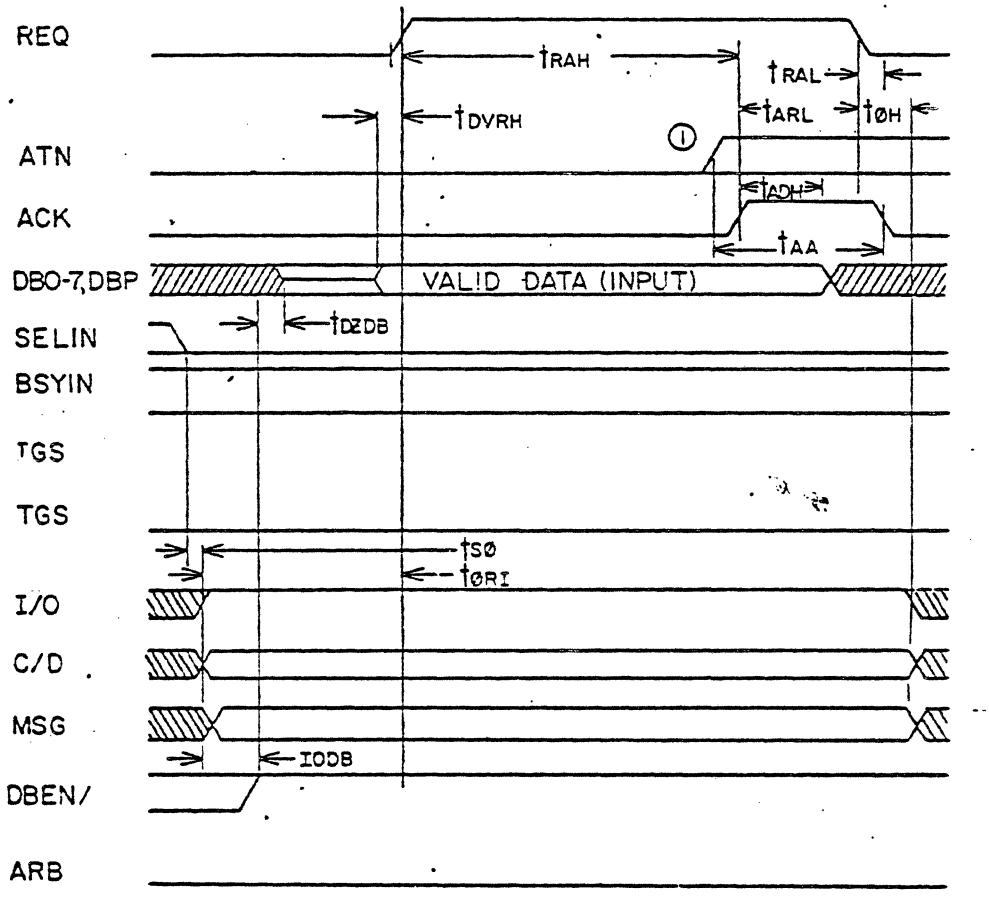
1. The chip ensures that the bus remains free (BSYIN and SELIN inactive) for t8F before attempting arbitration.

2. If SELIN becomes active at any time during arbitration, the chip must deassert BSYOUT within t8CD.

3. The chip waits tAD and then checks to see if arbitration is won (tPC). The chip then asserts SELOUT if arbitration is won.

4. One of the Data Bits is assigned as an ID bit by the ID0/-ID2/ signals. During Bus Free, the chip places all of the Data Bits (including ID) in a h' impedance state. During arbitration the c enables its ID bit and drives it high, but the remainder of the Data Bits remain in the high impedance state for reading.

7.2.5 INFORMATION TRANSFER PHASE INPUT (INITIATOR)

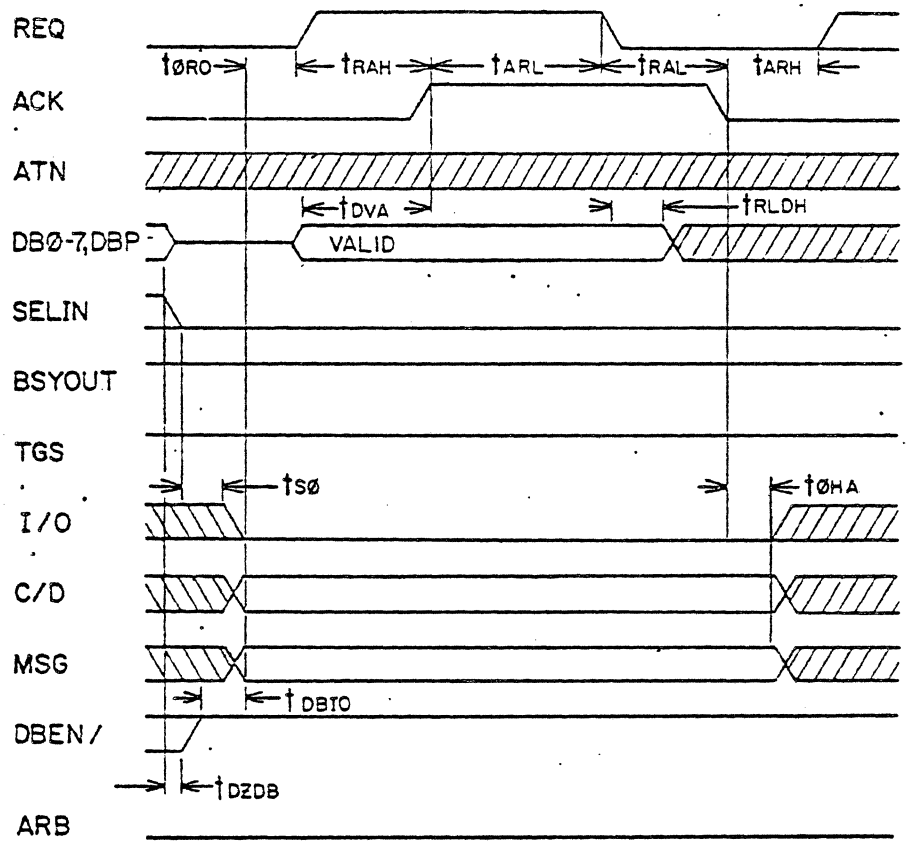


7.0 EXTERNAL CHIP TIMING
7.2.5 INFORMATION TRANSFER PHASE INPUT (INITIATOR)

NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tDVRH	DATA VALID TO REQ HIGH	0			NS
tORI	PHASE VALID TO REQ HIGH	100			NS
tRAH	REQ HIGH TO ACK HIGH	0			NS
tRAL	REQ LOW TO ACK LOW	0			NS
tAA	ATN HIGH TO ACK LOW	100			NS
tSO	SELIN LOW TO PHASE CHANGE	0			NS
tOH	PHASE HOLD FROM ACK LOW	20			NS
tADH	DATA HOLD FROM ACK HIGH	0			NS
tARL	ACK HIGH TO REQ LOW	0			NS
tIOD8	I/O HIGH TO DBEN/ HIGH			50	NS
tDZD8	DATA BUS DISABLE FROM DBEN/ HIGH			10	NS

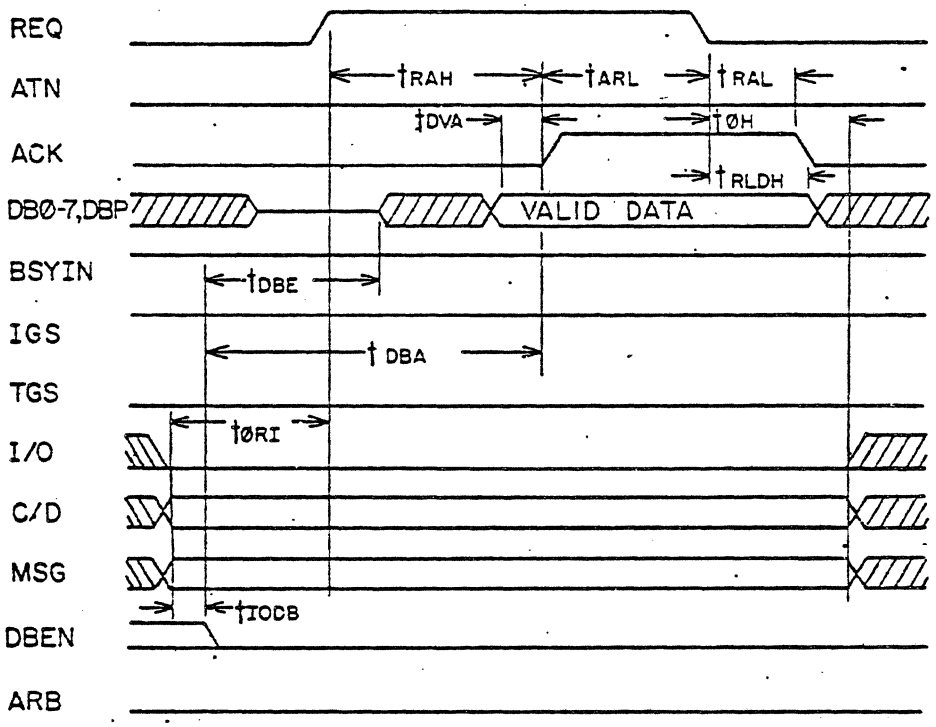
NOTE 1: If the chip detects a parity error it must assert ATN at least tAA before it deasserts ACK.

7.2.6 INFORMATION TRANSFER PHASE INPUT (TARGET)



NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tSO	SELIN LOW TO PHASE CHANGE	0			NS
tORQ	PHASE CHANGE TO REQ OUT	500			NS
tRAH	REQ HIGH TO ACK HIGH	0			NS
tARL	ACK HIGH TO REQ LOW	0			NS
tDVA	DATA VALID TO TO ACK HIGH	0			NS
tRAL	REQ LOW TO ACK LOW	0			NS
tRLDH	REQ LOW DATA DATA HOLD	0			NS
tARH	ACK LOW TO REQ HIGH	0			NS
tOHA	PHASE HOLD FROM ACK LOW	0			NS
tDBIO	DBEN/ HIGH TO I/O LOW	0			NS
tDZDB	DATA BUS DISABLE TO DBEN/ HIGH	0			NS

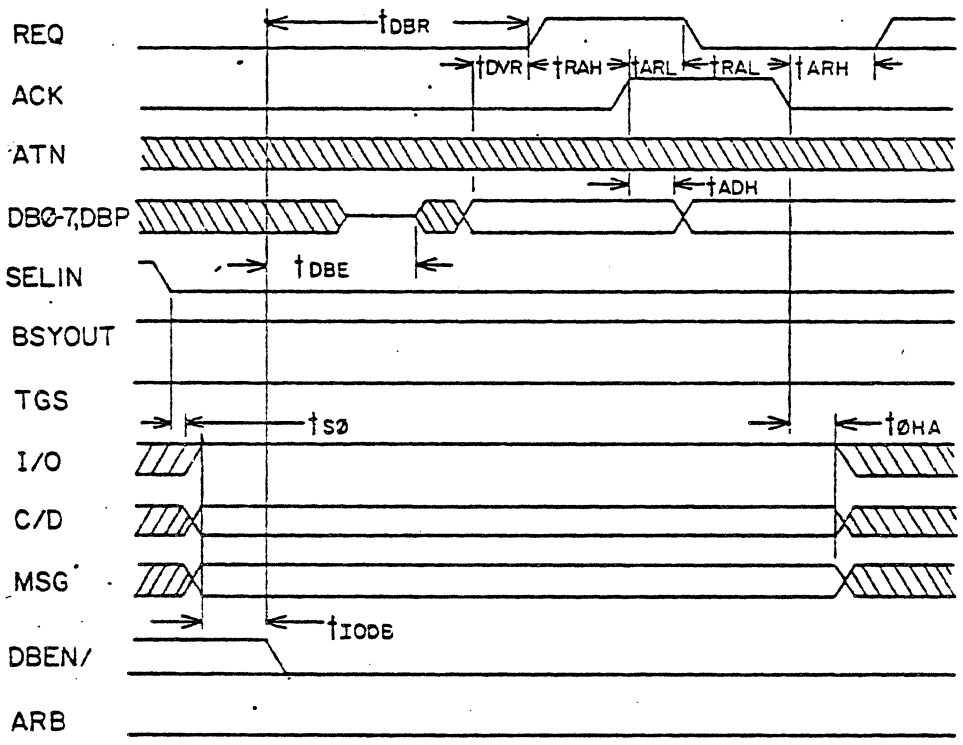
7.2.7 INFORMATION TRANSFER PHASE OUTPUT (INITIATOR)



7.0 EXTERNAL CHIP TIMING
7.2.7 INFORMATION TRANSFER PHASE OUTPUT (INITIATOR)

NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tRAH	REQ HIGH TO ACK HIGH	0			NS
tRAL	REQ LOW TO ACK LOW	0			NS
tDVA	DATA VALID TO ACK HIGH	100			NS
tRLDH	REQ LOW DATA HOLD	0			NS
tORI	PHASE VALID TO REQ	100			NS
tOH	PHASE HOLD FROM ACK LOW	20			NS
tARL	ACK HIGH TO REQ LOW	0			NS
tIODB	I/O LOW TO DBEN/	0			NS
tDBE	DBEN/ LOW TO DATA BUS ENABLE	85			NS
tDBA	DBEN/ LOW TO ACK HIGH	185			NS

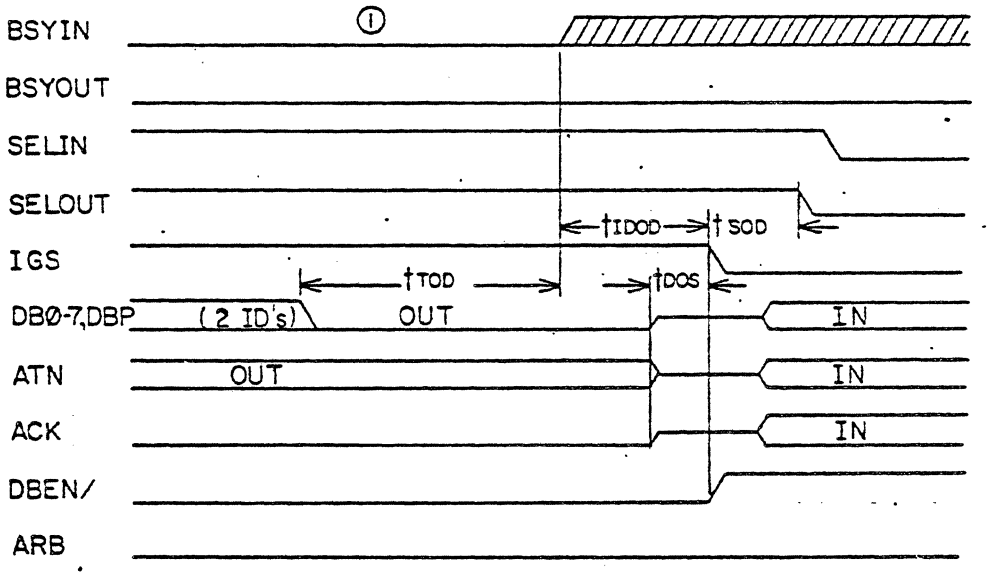
7.2.8 INFORMATION TRANSFER PHASE OUTPUT (TARGET)



7.0 EXTERNAL CHIP TIMING
7.2.8 INFORMATION TRANSFER PHASE OUTPUT (TARGET)

NAME	DESCRIPTION	MIN	TYP	MAX	UNITS
tSO	SELIN LOW TO PHASE CHANGE	0			NS
tIODB	I/O HIGH TO DBEN/ LOW	500			NS
tDBR	DBEN/ LOW TO REQ OUT	185			NS
tDVR	DATA VALID TO REQ HIGH	100			NS
tRAH	REQ HIGH TO ACK HIGH	0			NS
tARL	ACK HIGH TO REQ LOW	0			NS
tRAL	REQ LOW TO ACK LOW	0			NS
tARH	ACK LOW TO REQ HIGH	0			NS
tOHA	PHASE HOLD FROM ACK LOW	0			NS
tADH	DATA HOLD FROM ACK HIGH	0			NS
tDBE	DBEN/ LOW TO DATA BUS ENABLED	85			NS

7.2.9 BUS RELEASE FROM SELECTION (INITIATOR)

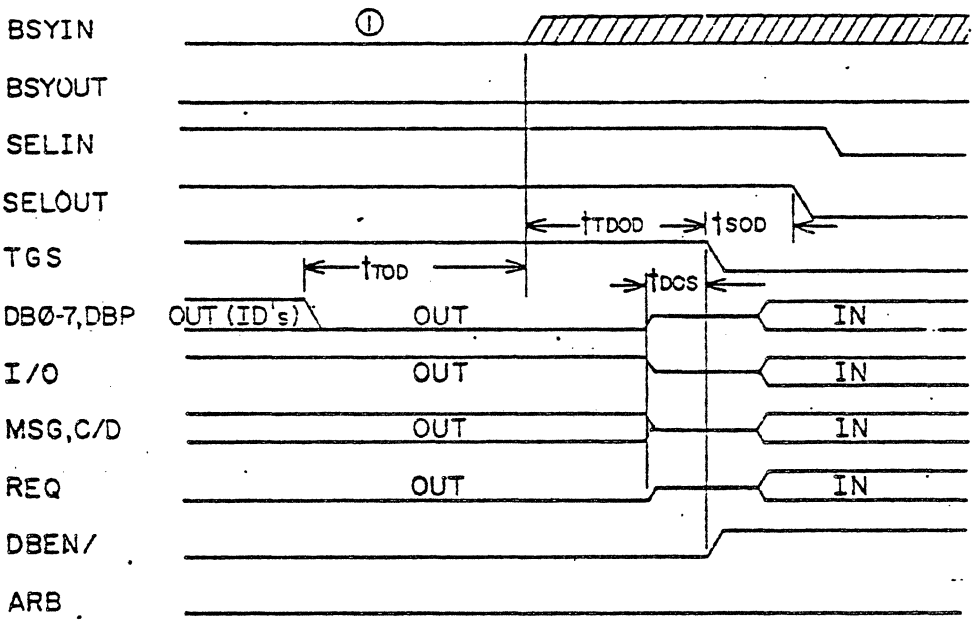


7.0 EXTERNAL CHIP TIMING
7.2.9 BUS RELEASE FROM SELECTION (INITIATOR)

NAME	SPECIFICATION	MIN	TYP	MAX	UNITS
tTOD	BUS RELEASE TIMEOUT DELAY	100			uS
tIDOD	IGS & DBEN/ TURN-OFF DELAY	0			NS
tSOD	SELOUT TURN-OFF DELAY	0			NS
tDOS	DRIVER TURN-OFF SET-UP TO IGS & DBEN/ OFF	0			NS

NOTE 1: If the chip detects BSYIN active by the end of the timeout delay, the bus release sequence shall be aborted since selection has been successful.

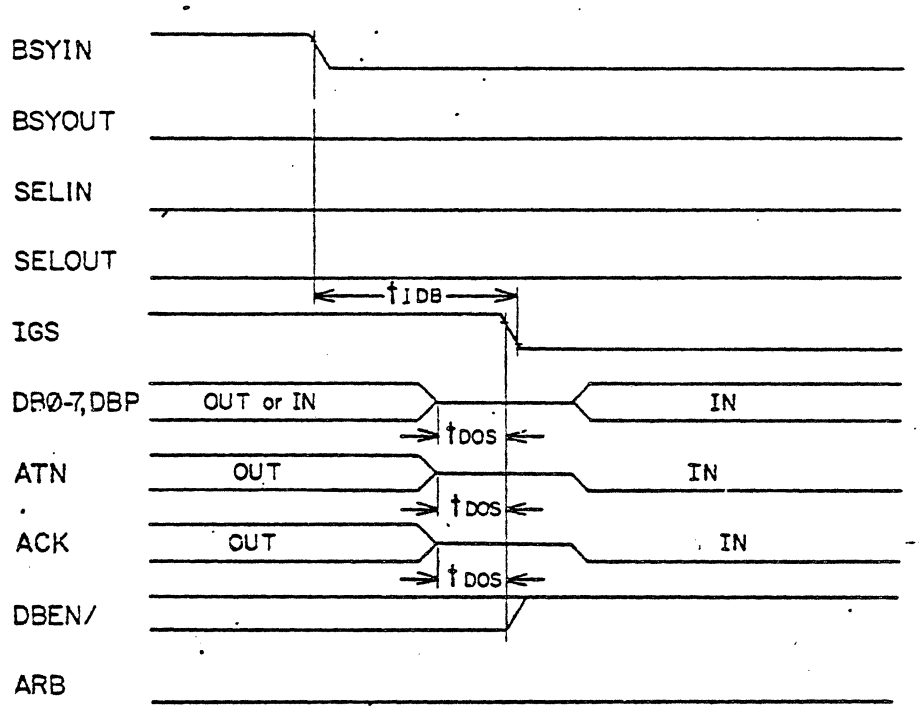
7.2.10 BUS RELEASE FROM RESELECTION (TARGET)



7.0 EXTERNAL CHIP TIMING
7.2.10 BUS RELEASE FROM RESELECTION (TARGET)

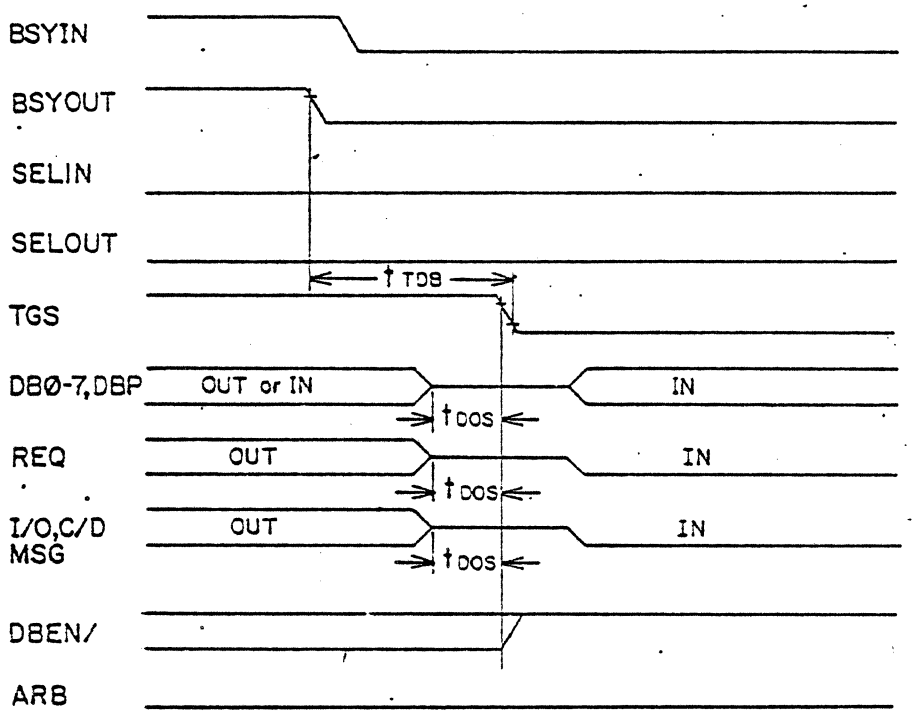
NAME	SPECIFICATION	MIN	TYP	MAX	UNITS
tTOD	BUS RELEASE TIMEOUT DELAY	100			uS
tDOD	TGS & DBEN/ TURN-OFF DELAY	0			NS
tSOD	SELOUT TURN-OFF DELAY	0			NS
tDOS	DRIVER TURN-OFF SET-UP TO TGS OFF	0			NS

7.2.11 BUS RELEASE FROM INFORMATION PHASE (INITIATOR)



NAME	SPECIFICATION	MIN	TYP	MAX	UNITS
tIDB	IGS & DBEN/ TURN-OFF DELAY FROM BSYIN OFF			225	NS
tDOS	DRIVER TURN-OFF SET-UP TO IGS OFF	0			NS

7.2.12 BUS RELEASE FROM INFORMATION PHASE (TARGET)



NAME	SPECIFICATION	MIN	TYP	MAX	UNITS
tTDB	TGS & DBEN/ TURN-OFF DELAY FROM BSYIN OFF			225	NS
tDOS	DRIVER TURN-OFF SET-UP TO TGS OFF	0			NS

8.0 INTERNAL HARDWARE FLAGS AND CONTROLS

The following definitions are included to provide clarity to the Chip Functional Flowchart. These definitions are used by the internal control circuitry of the chip and are not accessible by external logic unless noted.

8.1 FLAGS

8.1.1 COMMAND PENDING LATCH

Indicates that an interrupting command has been loaded and requires action by the chip controller. It is reset when the chip begins to execute the command and whenever an Interrupt is generated.

8.1.2 SELECTED LATCH

This is set when the chip has been selected as a Target. It is used by the chip controller to later set the "SELECTED" interrupt for use by the MPU.

8.1.3 RESELECTED LATCH

This latch is set when the chip has been reselected as an Initiator. It is used by the chip controller to later set the "RESELECTED" interrupt for use by the MPU.

8.1.4 PHASE CHANGE

This latch is used when the chip is in the Initiator role. It is set during the Information Transfer Phase if the type of information to be transferred changes.

Ex. Data Phase to Message Phase.

It is used by the Chip controller to set the "BUS SERVICE INTERRUPT" which indicates to the MPU that help is needed to

handle the next information transfer.

8.1.5 INTERRUPT SERVICE COMPLETE

This latch is an indication to the chip controller that an interrupt has been issued to the MPU but has not been serviced.

No more interrupts will be issued until the present ones are serviced. This simplifies interrupt handling for the MPU.

8.1.6 TRANSFER PAD BIT

This bit is set if a "Transfer Pad" command is received. It allows the same sequencing logic to be used for the "Transfer Information" command as is used for "Transfer Pad" command. Depending on the state of the Transfer Pad bit, the sequencing logic chooses the proper path to execute the command received.

8.1.7 BUS CONNECTED LATCH

Having completed the Selection or Reselection process successfully, the "BUS connected" latch is set. The latch will remain set until the "BUS FREE" condition causes it to reset.

The "DISCONNECT" interrupt will be issued if the chip was in the Initiator Role when the reset occurred.

8.1.8 MSG OUT

This signal is a decode of the I/O, C/D and MSG. It is used by the chip when acting as an initiator. If active, the chip controller will reset the ATN signal on the SCSI BUS as the last message byte is being transferred.

8.2 HARDWARE COMMAND DECODES

8.2.1 DISCONNECT

The "DISCONNECT" command causes the immediate disconnection of the SCSI BUS and return of the chip to the disconnected id loop.

8.2.2 CHIP RESET

This command is hardware decoded and causes the chip to immediately disconnect from the SCSI bus. All storage elements are initialized. The chip then executes a level 0 diagnostic program and reports the results in the Diagnostic Status register. Upon successful completion of this command the "SELF DIAGNOSTIC COMPLETE" Status is set.

This procedure will be the same for power up.

8.2.3 ATN FF

This latch will be set immediately after receiving the "SET ATN" command. In turn the ATN LINE on the SCSI BUS will be driven active if the chip is connected as an initiator.

8.2.4 PAUSE FF

This latch is set when the Pause command is received. The chip sequences will then return to an IDLE state and set the PAUSE STATUS bit.

The latch + PAUSE Status bit are reset when another command is received.

8.2.5 CHIP DISABLE FF

This latch will be set immediately upon receipt of a Chip Disable command. It causes all chip activity to cease and outputs to be disabled. The latch is reset when the Reset input signal activated or the Chip Reset Command is received.

9.0 CHIP FUNCTIONAL FLOWCHARTS

Functional flowcharts that detail the internal operation of the SCSI Interface Chip are provided on an accompanying computer printout.

A1.0 USER NOTES AND GUIDELINES

A1.0 USER NOTES AND GUIDELINES

This appendix contains important notes and guidelines concerning the use of the SCSI interface chip.

1. Always read the Auxiliary Status Register before reading the Interrupt Register when servicing an interrupt.

2. The act of reading the Interrupt Register turns off the interrupt signal (INT), resets the interrupt register, and resets an internal flag that indicates an interrupting command has been loaded into the chip. Therefore, if an interrupt causing bus event occurs at about the same time an interrupting command is loaded into the Command Register and the chip generates an interrupt for the bus event, the command in the Command Register will not be executed by the chip.

3. The act of loading an interrupting command resets the Data Register Full status bit in the Auxiliary Status Register. Therefore, when a command is issued that requires data to be put into the Data Register, the first byte of data cannot be loaded until a minimum of one full clock cycle after the Command Register is loaded. This allows the chip sufficient time to complete resetting the Data Register Full bit.

4. When the chip is disconnected and Select Enable = 1 (Reselect Enable = 1), the user may receive an interrupt due to selection (reselection) at any time, even after issuing a command to the chip to select or reselect a device. For example, after issuing a SELECT command, a higher priority device wins the bus and selects (reselects) the chip. In this case, the chip may generate a Selected (Reselected) interrupt instead of a Function Complete interrupt for the SELECT command. The user must wait until the chip is disconnected before issuing the SELECT command again.

5. When a PAUSE command is issued and an interrupting condition occurs before the Pause flag is detected, the chip will always give priority to the interrupt instead of setting the Paused status bit.

6. If a DISCONNECT command is issued when connected as an

A1.0 USER NOTES AND GUIDELINES

intitiator, the target is left hanging on the bus. A bus reset may be required to free the bus.

7. The bus reset signal (RST) is not driven by the chip. It must be handled by external curcuitry. The chip does have a reset input (RESET) which can be tied to the bus reset, possibly gated by external circuitry.

8. SCSI message protocol is totally transparent to the chip.

9. SCSI I/O command protocol is completely transparent to the chip.

B1.0 INITIALIZATION AFTER POWER-UP OR RESET

B1.0 INITIALIZATION AFTER POWER-UP OR RESET

Below are the steps that a user of the SCSI Interface Chip would typically perform after the chip is powered-up or reset. It is assumed that no errors occur.

1. Loop on reading the Diagnostic Status Register until the Self-Diagnostic Complete bit is on.

2. Check the Diagnostic Command Status and Self-Diagnostic Status bits of the Diagnostic Status Register for all zeroes (no errors).

3. Load the Control Register with the desired information (Parity Enable, Reselect Enable, Select Enable).

It is noted that immediately following step 3 the chip is in the disconnected state. If Reselect Enable or Select Enable are turned on, an interrupt can occur prior to issuing any commands to the chip.

82.0 INITIATOR ROLE

Below are the steps that the user of the SCSI Interface Chip would follow in performing a complete I/O function as an initiator. For this typical example, it is assumed that the initiator and target both can handle messages and are able to disconnect and reconnect during the function. To simplify this example, it is further assumed that no errors or exceptions occur during the entire operation.

INITIATOR SELECTS TARGET

1. Load the Destination ID Register with the target's ID.
2. Load the Transfer Counter to program the selection timeout. Write to each of the three 8-bit registers.
3. Load the Command Register with a SELECT W/ATN command.
4. Wait for an interrupt.
5. Read the Auxiliary Status Register.
6. Read the Interrupt Register.
7. Check for a Function Complete interrupt that indicates the SELECT W/ATN was successful.

(The user is now in the connected initiator state).

INITIATOR SENDS ID MESSAGE

8. Wait for an interrupt.
9. Read the Auxiliary Status Register.
10. Read the Interrupt Register.
11. Check the interrupt. A Bus Service interrupt should have occurred indicating that the target has activated REQ f an information transfer.

B2.0 INITIATOR ROLE

12. Check the I/O, C/D, and MSG bits read from the Auxiliary Status Register. The target should be requesting a Message Out phase to receive the "Identify" message.

13. Load the Command Register with a Transfer Info command. Since the "Identify" message is a single byte, program the Single Byte Transfer bit on and the DMA Mode bit off.

14. Read the Auxiliary Status Register.

15. Check the Data Register Full bit.

16. Repeat (14) and (15) until Data Register Full is off.

17. Write the "Identify" message into the Data Register.

18. Wait for an interrupt.

19. Read the Auxiliary Status Register.

20. Read the Interrupt Register.

21. Check the interrupt. Another Bus Service interrupt should have occurred indicating that the target again has activated REQ for another information transfer.

INITIATOR RECEIVES DISCONNECT MESSAGE

(NOTE: The target is not required to send a Disconnect message and disconnect at this point. It may request the command before disconnecting or not disconnect at all. If the target does not issue the Disconnect message here, proceed to step 65.)

22. Check the I/O, C/D, and MSG bits read from the Auxiliary Status Register. The target should be requesting a Message In phase.

23. Load the Command Register with a Transfer Info command. The Single Byte Transfer bit should be on and the DMA Mode bit off.

24. Read the Auxiliary Status Register.

25. Check the Data Register Full bit.

B2.0 INITIATOR ROLE

26. Repeat (24) and (25) until Data Register Full is on.

27. Read the Data Register.

28. Check the message. The target should have sent a "Disconnect" message indicating that he will reconnect later to complete the I/O function.

29. Wait for an interrupt. (It is noted that this interrupt may occur any time after the Transfer Info command was loaded, step (23). The user must handle the possible occurrence of this interrupt during steps (24)-(28). One way the user can handle this is to set an interrupt flag in the interrupt service routine, mask any more interrupts, and then complete steps (24)-(28). If the interrupt is a Disconnect, it would need immediate service.)

30. Read the Auxiliary Status Register.

31. Read the Interrupt Register.

32. Check the interrupt. A Function Complete should have occurred indicating that the last byte of the message has been received. The chip has left ACK on so that ATN can be set on if the message needs to be rejected.

33. Load the Command Register with a Message Accepted command.

INITIATOR WAITS FOR DISCONNECTION

34. Wait for an interrupt.

35. Read the Auxiliary Status Register.

36. Read the Interrupt Register.

37. Check the interrupt. The target should have disconnected causing a Disconnected interrupt.

(The user is now in the disconnected state and may start or handle I/O functions for any other logical unit. For this I/O function to continue the user must wait until reselected by the target while in the disconnected state. This is where step (38) continues the flow.)

B2.0 INITIATOR ROLE

INITIATOR IS RESELECTED

38. Wait for an interrupt.

39. Read Auxiliary Status Register.

40. Read Interrupt Register.

41. Check the interrupt. Assuming the target has now reselected the user to continue the I/O function, a Reselected interrupt should have occurred.

(The user is now in the connected initiator state.)

INITIATOR RECEIVES ID MESSAGE

42. Wait for an interrupt.

43. Read the Auxiliary Status Register.

44. Read the Interrupt Register.

45. Check the interrupt. A Bus Service interrupt should have occurred indicating that the target has activated REQ for an information transfer.

46. Check the I/O, C/D, and MSG bits read from the Auxiliary Status Register. The target should be requesting a Message In phase to identify the I/O.

47. Load the Command Register with a Transfer Info Command, Single Byte Transfer = 1 and DMA Mode = 0.

48. Read the Auxiliary Status Register.

49. Check the Data Register Full bit.

50. Repeat (48) and (49) until Data Register Full is on.

51. Read the Data Register.

52. Check the message. The target should have sent an "Identify" message which contains the logical unit number

for the I/O.

53. Read the Source ID Register.

54. Check the contents of the Source ID Register to determine which device did the reselection.

55. Having identified the device and logical unit number, the user can now retrieve the command, data, and status pointers for this I/O and store them in a working pointer area. Note that the pointers and working pointers are kept by the user outside of the SCSI Interface Chip.

56. Wait for an interrupt. (Note that this interrupt may occur any time after step (47). If it occurs during steps (48)-(55), the user can set an interrupt flag in the interrupt service routine, mask any more interrupts, and then complete steps (48)-(55). A Disconnect interrupt would need service immediately.)

57. Read the Auxiliary Status Register.

58. Read the Interrupt Register.

59. Check the interrupt. A Function Complete should have occurred indicating that the last byte of the message has been received. The chip has left ACK on so that ATN can be set on if the message needs to be rejected.

60. Load the Command Register with a Message Accepted command.

INITIATOR TRANSFERS COMMAND, DATA, OR STATUS

61. Wait for an interrupt.

62. Read the Auxiliary Status Register.

63. Read the Interrupt Register.

64. Check the interrupt. A Bus Service interrupt should have occurred indicating that the target has activated REQ for another information phase.

65. Check the I/O, C/D, and MSG bits read from the Auxiliary Status Register. The target should be requesting Command, Data, or Status phase.

B2.0 INITIATOR ROLE

66. Prepare circuitry external to the chip for the requested transfer by using the appropriate working pointer.

67. Load the Transfer Counter for the maximum number of bytes to be transferred. Write to each of the three 8-bit registers. (This step would be omitted for a single byte transfer.)

68. Load the Command Register with a Transfer Info command. For command or data transfers normally Single Byte Transfer = 0 and DMA Mode = 1. For status these bits might be 1 and 0, respectively.

69. If Single Byte Transfer = 0, go to step (74).

70. Read the Auxiliary Status Register.

71. Check the Data Register Full bit.

72. Repeat (70) and (71) until the Data Register Full bit is off.

73. Write the Data Register (with the status byte).

74. Wait for an interrupt.

75. Read the Auxiliary Status Register.

76. Read the Interrupt Register.

77. Check the interrupt. A Bus Service interrupt should have occurred indicating that the target has activated REQ for a different information phase.

INITIATOR UPDATES WORKING POINTER FOR LAST TRANSFER

78. IF the last transfer was a single byte, go to step (82).

79. Check the Transfer Counter Zero bit in the Auxiliary Status Register.

80. If Transfer Counter Zero = 1, go to step (82).

81. Read the Transfer Counter.

82. Update the working pointer for the last information

phase. (Note that the stored pointer is not updated now. Stored pointers are updated only when a "Save State" or "Command Complete" message is received.)

INITIATOR CHECKS NEW PHASE TYPE

83. Check the I/O, C/D, and MSG bits read from the Auxiliary Status Register.

84. If the target is requesting a Command, Data, or Status phase, go back to step (66).

INITIATOR RECEIVES MESSAGE

85. Otherwise, the target should be requesting a Message In phase. Load the Command Register with a Transfer Info, Single Byte Transfer = 1 and DMA Mode = 0.

86. Read the Auxiliary Status Register.

87. Check the Data Register Full bit.

88. Repeat (86) and (87) until Data Register Full is on.

89. Read the Data Register, which contains the message.

90. If the message is a "Command Complete", go to step (113).

91. If the message is a "Disconnect", go to step (103).

INITIATOR HANDLES SAVE STATE MESSAGE

92. Otherwise, for normal operation, the message should be a "Save State". In this case, the user should save the state of the working pointers by moving them to the stored pointer area.

93. Wait for an interrupt. (Note this interrupt may occur any time after step (85). If it occurs before step (93), the user can set an interrupt flag in the interrupt service routine, mask any more interrupts, and then proceed to complete the steps prior to (93). If a Disconnect interrupt occurs before Data Register Full goes following (85), it would need to be serviced immediatel

B2.0 INITIATOR ROLE

94. Read the Auxiliary Status Register.

95. Read the Interrupt Register.

96. Check the interrupt. A Function Complete should have occurred indicating that the last byte of the message has been received. The chip has left ACK on so that ATN can be set on if the message needs to be rejected.

97. Load the Command Register with a Message Accepted command.

98. Wait for an interrupt.

99. Read the Auxiliary Status Register.

100. Read the Interrupt Register.

101. Check the interrupt. A Bus Service interrupt should have occurred indicating that the target has activated REQ for another information phase.

102. Go to step (83).

INITIATOR HANDLES DISCONNECT MESSAGE

103. Wait for an interrupt. (Note that this interrupt may occur any time after step (85). If it occurs before step (103), the user can set an interrupt flag in the interrupt service routine, mask any more interrupts, and then proceed to complete the steps prior to (103). If a Disconnect interrupt occurs before Data Register Full goes on following (85), it would need to be serviced immediately.)

104. Read the Auxiliary Status Register.

105. Read the Interrupt Register.

106. Check the interrupt. A Function Complete should have occurred indicating that the last byte of the message has been received. The chip has left ACK on so that ATN can be set on if the message needs to be rejected.

107. Load the Command Register with a Message Accepted command.

INITIATOR WAITS FOR DISCONNECTION

108. Wait for an interrupt.

109. Read the Auxiliary Status Register.

110. Read the Interrupt Register.

111. Check the interrupt. After sending the "Disconnect" message, the target should have disconnected resulting in a Disconnected interrupt.

112. Go to step (38). (The note prior to (38) applies.)

INITIATOR HANDLES COMMAND COMPLETE MESSAGE

113. Save the state of the working pointers by moving them to the stored pointer area.

114. Wait for an interrupt. (Note this interrupt may occur any time after step (85). If it occurs before (114), the user can set an interrupt flag in the interrupt service routi- mask any more interrupts, and then complete the steps pr to (114). If a Disconnect interrupt occurs before D- a Register Full goes on following (85), it would need to be serviced immediately.

115. Read the Auxiliary Status Register.

116. Read the Interrupt Register.

117. Check the interrupt. A Function Complete should have occurred indicating that the last byte of the message has been received. The chip has left ACK on so that ATN can be set on if the message needs to be rejected.

118. Load the Command Register with a Message Accepted command.

INITIATOR WAITS FOR DISCONNECTION

119. Wait for an interrupt.

120. Read the Auxiliary Status Register.

121. Read the Interrupt Register.

122. Check the interrupt. After sending the "Command Comple-

message, the target should have disconnected resulting in a Disconnected interrupt.

(The I/O function is now complete. The user is back in the disconnected state.)

NOTE: Steps 14-16 and 70-72 can be omitted if the microprocessor guarantees that one full clock cycle elapses between loading the Command Register and loading the Data Register. This should be the normal case.

B3.0 TARGET ROLE

Below are the steps that the user of the SCSI Interface Chip would follow in performing a complete I/O function as a target. For this typical example, it is assumed that the target and initiator both can handle messages and are able to disconnect. It is also assumed that no errors or exceptions occur during the entire operation. The sequence of steps begins after the chip has been selected as a target.

TARGET IS SELECTED

1. Wait for an interrupt.
2. Read the Auxiliary Status Register.
3. Read the Interrupt Register.
4. Check the interrupt. Selected and Bus Service interrupts should have occurred indicating that the chip has been selected as a target and the initiator has asserted the ATN signal.

   (The user is now in the connected target state.)

TARGET RECEIVES ID MESSAGE

5. Load the Command Register with a RECEIVE MESSAGE OUT command, Single Byte TRANSFER = 1, DMA MODE = 0.
6. Read the Auxiliary Status Register.
7. Check the Data Register Full bit.
8. Repeat (6) and (7) until Data Register Full is on.
9. Read the Data Register.

B3.0 TARGET ROLE

10. Check the message. The initiator should have sent an "Identify" message which indicates whether the initiator can disconnect and contains the logical unit number for the I/O.

11. Read the Source ID Register.

12. Check the ID Valid bit. It will be on for an initiator which can disconnect. (The user now has the initiator ID and the logical unit number which uniquely defines an I/O. The user may record this information and disconnect.

13. Wait for an interrupt. (Note that this interrupt may occur any time after step (9). If it occurs during steps (10) - (12), the user may set an interrupt flag in the interrupt service routine, mask any more interrupts, and then complete steps (10) - (12).)

14. Read the Auxiliary Status Register.

15. Read the Interrupt Register.

16. Check the interrupt. A Function Complete should have occurred indicating that the ID message has completed.

    (The user is back in the connected target state. If it is desired not to disconnect at this point, the user should go to step (43).)

TARGET SENDS DISCONNECT MESSAGE AND DISCONNECTS

17. Load the Command Register with a SEND MESSAGE IN with Single Byte Transfer = 1 and DMA Mode = 0.

16. Read the Auxiliary Status Register.

19. Check the Data Register Full bit.

20. Repeat (18) and (19) until Data Register Full is off.

21. Write the "Disconnect" message into the Data Register.

22. Wait for an interrupt.

23. Read the Auxiliary Status Register.

24. Read the Interrupt Register.

25. Check the interrupt. A Function Complete interrupt should have occurred indicating that the message was sent successfully.

26. Load the Command Register with a DISCONNECT command.

(The DISCONNECT command immediately breaks the connection. The user is now back in the disconnected state and when ready to continue the I/O operation, starts again at step (27).)

TARGET RESELECTS INITIATOR

27. Load the Destination ID Register with the initiator's ID.

28. Load the Transfer Counter to program the reselection timeout. Write to each of the three 8-bit registers.

29. Load the Command Register with a RESELECT command.

30. Wait for an interrupt.

31. Read the Auxiliary Status Register.

32. Read the Interrupt Register.

33. Check the interrupt. A Function Complete interrupt should have occurred indicating that the RESELECT was successful.

(The user is now in the connected target state.)

TARGET SENDS ID MESSAGE

34. Load the Command Register with a SEND MESSAGE IN command, Single Byte Transfer = 1 and DMA Mode = 0.

35. Read the Auxiliary Register.

36. Check the Data Register Full bit.

37. Repeat (35) and (36) until Data Register Full is off.

38. Write the "Identify" message into the Data Register.

B3.0 TARGET ROLE

39. Wait for an interrupt.

40. Read the Auxiliary Status Register.

41. Read the Interrupt Register.

42. Check the interrupt. A Function Complete interrupt should have occurred indicating that the message was sent successfully.

TARGET RECEIVES COMMAND OR TRANSFERS DATA

43. Load the Transfer Counter for a command or data transfer.

44. Load the Command Register with a RECEIVE COMMAND, RECEIVE DATA, or a SEND DATA command with Single Byte Transfer = 0 and DMA Mode = 1.

45. Wait for an interrupt.

46. Read the Auxiliary Status Register.

47. Read the Interrupt Register.

48. Check the interrupt. A Function Complete interrupt should have occurred indicating that the transfer was successful.

49. To do a data transfer, go back to (43).

50. If the I/O function is complete go to step (62) to send status and the "Command Complete" message. Otherwise, proceed to "Save State" and disconnect with the intent of reconnecting later.

TARGET SENDS SAVE STATE AND DISCONNECT MESSAGES AND DISCONNECTS

51. Load the Command Register with a SEND MESSAGE IN with Single Byte Transfer = 1 and DMA Mode = 0/.

52. Read the Auxiliary Status Register.

53. Check the Data Register Full bit.

54. Repeat (52) and (53) until Data Register Full is off.

B3.0 TARGET ROLE

55. Write the "Save State" message into the Data Register.

56. Wait for an interrupt.

57. Read the Auxiliary Status Register.

58. Read the Interrupt Register.

59. Check the interrupt. A Function Complete interrupt should have occurred indicating that the message was sent successfully.

60. Repeat steps (51)-(59) for a "Disconnect" message.

61. Load the Command Register with a DISCONNECT command.

(The DISCONNECT command immediately breaks the connection. The user is now back in the disconnected state and when ready to continue the I/O operation, starts again at step (27).)

TARGET SENDS STATUS BYTE

62. Load the Command Register with a Send Status command with Single Byte Transfer = 1 and DMA Mode = 0.

63. Read the Auxiliary Status Register.

64. Check the Data Register Full bit.

65. Repeat (63) and (64) until Data Register Full is off.

66. Write the status byte into the Data Register.

67. Wait for an interrupt.

68. Read the Auxiliary Status Register.

69. Read the Interrupt Register.

70. Check the interrupt. A Function Complete interrupt should have occurred indicating that the status byte was sent successfully.

TARGET SENDS COMMAND COMPLETE MESSAGE

B3.0 TARGET ROLE

71. Load the Command Register with a SEND MESSAGE IN with Single Byte Transfer = 1 and DMA Mode = 0.

72. Read the Auxiliary Status Register.

73. Check the Data Register Full bit.

74. Repeat (72) and (73) until Data Register Full is off.

75. Write the "Command Complete" message into the Data Register.

76. Wait for an interrupt.

77. Read the Auxiliary Status Register.

78. Read the Interrupt Register.

79. Check the interrupt. A Function Complete interrupt should have occurred indicating that the message was sent successfully.

80. Load the Command Register with a DISCONNECT command.

(The I/O function is now complete. The user is back in the disconnected state.)

NOTE: Steps 18-20, 35-37, 52-54, 63-65, and 72-74 can be omitted if the microprocessor guarantees that one full clock cycle elapses between loading the Command Register and loading the Data Register. This should be the normal case.

C1.0 USER INTERRUPT SERVICE ROUTINES

C1.0 USER INTERRUPT SERVICE ROUTINES

This section defines all of the possible interrupt conditions that can occur and provides the user with a suggested response.

Below is a list of the interrupting condition bits and auxiliary status bits. It is noted that when interrupted the user should first read and save the Auxiliary Status Register and then the Interrupt Register. The process of reading the Interrupt Register clears the interrupt and allows the chip to interrupt again.

INTERRUPTS		AUXILIARY STATUS	
0	Function Complete	0	(Not used)
1	Bus Service	1	Transfer Counter Zero
2	Disconnected	2	Paused
3	Selected	3	I/O
4	Reselected	4	C/D
5	(Not Used)	5	MSG
6	Invalid Command	6	Parity Error
7	(Not Used)	7	Data Register Full

At any point in time prior to an interrupt, the user must know if he is disconnected, connected as a target, or connected as an initiator. This state determines which commands that are valid for him to execute and it determines what his interrupt service routine should be. Below are suggested interrupt service routines for each state.

C2.0 INTERRUPT SUMMARY

C2.0 INTERRUPT SUMMARY

USER STATE	INTERRUPTS (7-0)	AUXILIARY STATUS (7-0)	EVENT
Disconnected	0000 1000	XXXX XXXX	Selected as target, ATN off.
Disconnected	0000 1010	XXXX XXXX	Selected as target, ATN on.
Disconnected	0001 0000	XXXX XXXX	Reselected as initiator.
Disconnected	0000 0001	XXXX XXXX	SELECT W/ATN, SELECT W/O ATN, or RESELECT, command completed successfully.
Disconnected	0000 0100	XXXX XXXX	No response from destination while executing a SELECT or RESELECT command.
Disconnected	0100 0000	XXXX XXXX	Invalid command issued.
Disconnected	All Others	XXXX XXXX	Hardware Error - should not occur.
Connected as Target	0000 0010	XXXX XXXX	ATN received.
Connected as Target	0000 0001	X0XX XXXX	SEND or RECEIVE command completed successfully.
Connected as Target	0000 0011	X0XX XXXX	SEND or RECEIVE command completed; ATN was turned on during the transfer.
Connected as Target	0000 0001	X1XX XXXX	RECEIVE command terminated due to a bus parity error.
Connected as Target	0000 0011	X1XX XXXX	RECEIVE command terminated due to a bus parity error and ATN is on.
Connected as Target	0100 0000	XXXX XXXX	Invalid command issued.
Connected as Target	All Others	XXXX XXXX	Hardware Error - should not occur.
Connected as Initiator	0000 0010	XXXX XXXX	Request from Target needs to be serviced.
Connected as Initiator	0000 0100	X0XX XXXX	Target disconnected from bus.
Connected as Initiator	0000 0001	XXXX XXXX	TRANSFER for Message In has

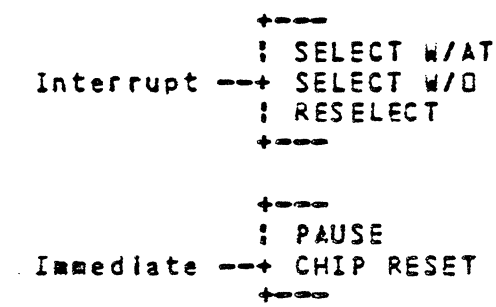
C2.0 INTERRUPT SUMMARY

			completed.
Connected as Initiator	0000 0100	X1XX XXXX	Target disconnected from bus and did not respond to ATN due to a parity error.
Connected as Initiator	0000 0010	X1XX XXXX	Request from Target needs to be serviced. A parity error was previously detected and ATN turned on.
Connected as Initiator	0100 0000	XXXX XXXX	Invalid command issued.
Connected as Initiator	All Others	XXXX XXXX	Hardware Error - should not occur.

C3.0 USER DISCONNECTED ISR

C3.0 USER_DISCONNECTED_ISR

In this state the user is currently logically disconnected from the SCSI bus or was disconnected at the time the last command to the chip was issued. Valid commands that may have been issued in this state are listed below.



Below are descriptions of all interrupts that may occur in this state.

INTERRUPTS (7-0): 0000 1000

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:
User has been selected as a target. ATN was not turned on by the initiator; therefore, he is not capable of using messages (except "Command Complete") and cannot disconnect prior to completing the function.

SUGGESTED RESPONSE:
User should set up the Transfer Counter, issue a RECEIVE COMMAND, and proceed with the function thru status phase and "Command Complete" message.

INTERRUPTS (7-0): 0000 1010

AUXILIARY STATUS (7-0): XXXX XXXX

C3.0 USER DISCONNECTED ISR

REASON FOR INTERRUPT:
User has been selected as a target. ATN was turned on by the initiator; therefore, he is capable of using messages.

SUGGESTED RESPONSE:
User should read the Source ID Register. If the ID Valid bit is not on, the initiator cannot be disconnected until the function is completed. The user should then proceed by setting the Transfer Counter and issuing a RECEIVE MESSAGE OUT. If the ID Valid bit is on, the user should still issue a RECEIVE MESSAGE OUT and use the ID message, if sent, to determine whether the initiator can disconnect. In either case the user can proceed with the function.

INTERRUPTS (7-0): 0001 0000

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:
User has been reselected as an initiator. It is impl that the target has disconnect and control mess capability.

SUGGESTED RESPONSE:
User must wait for another interrupt, either Bus Service or Disconnected. Normally, a Bus Service interrupt for a Message In phase for sending an "ID" message would be expected.

INTERRUPTS (7-0): 0000 0001

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:
The user command has been completed successfully. SELECT W/ATN or SELECT W/O ATN implies that the user has been successfully connected to a target and is now acting as an initiator. RESELECT implies that he has been reconnected to an initiator and is now acting as a target.

SUGGESTED RESPONSE:
User proceeds with his intended command sequence. After either SELECT command he waits for a Bus Service or Disconnected interrupt. After a RESELECT, he should issue SEND MESSAGE IN in order to transmit an "ID" message.

C3.0 USER DISCONNECTED ISR

INTERRUPTS (7-0): 0000 0100

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:
While executing a SELECT or RESELECT command, no response (BSY) was received from the destination device within the specified timeout. The operation was aborted.

SUGGESTED RESPONSE:
User should retry a limited number of times.

INTERRUPTS (7-0): 0100 0000

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:
The user issued a command that is not valid in the disconnected state.

SUGGESTED RESPONSE:
If the command is valid, retry or issue CHIP RESET and retry.

INTERRUPTS (7-0): All Others

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:
Chip malfunction.

SUGGESTED RESPONSE:
Issue chip reset and retry operation.

C4.0 USER CONNECTED AS TARGET ISR

C4.0 USER CONNECTED AS TARGET ISR

In this state the user is logically connected on the SCSI bus in the Target role. Commands that are valid to issue in this state are listed below:

```
              +--
              ! RECEIVE COMMAND
              ! RECEIVE DATA
              ! RECEIVE MESSAGE OUT
              ! RECEIVE UNSPECIFIED OUTPUT
Interrupt --+ ! SEND STATUS
              ! SEND DATA
              ! SEND MESSAGE IN
              ! SEND UNSPECIFIED INPUT
              +--

              +--
              ! PAUSE
Immediate --+ ! DISCONNECT
              ! CHIP RESET
              +--
```

In order to service and interrupt in this state the user should know if he has a command pending that will result in an interrupt and what the command is.

Below is a descritpion of the interrupts that may occur in this state.

INTERRUPTS (7-0): 0000 0010

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:
ATN has been received from the initiator. If this interrupt is received after issuing an interrupting command, the command was not and will not be executed by the chip.

SUGGESTED RESPONSE:
User should issue a RECEIVE MESSAGE OUT to determine why t initiator set ATN. If a command was aborted, it will n

C4.0 USER CONNECTED AS TARGET ISR

to be reissued.

INTERRUPTS (7-0): 0000 0001

AUXILIARY STATUS (7-0): XOXX XXXX

REASON FOR INTERRUPT:
A SEND or RECEIVE command has completed successfully.

SUGGESTED RESPONSE:
User proceeds with function by issuing any other valid command.

INTERRUPTS (7-0): 0000 0011

AUXILIARY STATUS (7-0): XOXX XXXX

REASON FOR INTERRUPT:
A SEND or RECEIVE command has completed. ATN was turned on by the initiator during the transfer.

SUGGESTED RESPONSE:
User should issue a RECEIVE MESSAGE OUT to determine why the user set ATN.

INTERRUPTS (7-0): 0000 0001

AUXILIARY STATUS (7-0): X1XX XXXX

REASON FOR INTERRUPT:
A RECEIVE command terminated due to a bus parity error. (ATN is not on.)

SUGGESTED RESPONSE:
If the error occurred during a RECEIVE MESSAGE OUT, the user should issue a SEND MESSAGE IN, "Message Parity Error" followed by a RECEIVE MESSAGE OUT in order to retry the message. If the error occurred during another RECEIVE command, the user should issue a SEND MESSAGE IN, "Restore State", and retry the entire transmission. In either case, the number of retries should be limited.

INTERRUPTS (7-0): 0000 0011

C4.0 USER CONNECTED AS TARGET ISR

AUXILIARY STATUS (7-0): X1XX XXXX

REASON FOR INTERRUPT:
A RECEIVE command terminated due to a bus parity error, and the initiator is asserting ATN.

SUGGESTED RESPONSE:
Similar to previous interrupt, except if error did not occur on a message then the ATN should be serviced first by issuing a RECEIVE MESSAGE OUT.

INTERRUPTS (7-0): 0100 0000

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:
The user issued a command that is not valid in the connected target state.

SUGGESTED RESPONSE:
If the command is valid, retry or issue CHIP RESET and re , the entire operation.

INTERRUPTS (7-0): All Others

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:
Chip malfunction.

SUGGESTED RESPONSE:
Issue chip reset and retry operation.

C5.0 USER CONNECTED AS INITIATOR ISR

C5.0 USER CONNECTED AS INITIATOR ISR

In this state the user is logically connected on the SCSI bus in the Initiator role. Commands that are valid to issue in this state are listed below:

```
             +--
             ! TRANSFER INFO
Interrupt --+ TRANSFER PAD
             +--

             +--
             ! MESSAGE ACCEPTED
             ! SET ATN
Immediate --+ DISCONNECT
             ! CHIP RESET
             +--
```

In order to service an interrupt the user should know if he has a command pending that will result in an interrupt and what the command is. He should also remember what the information phase was at the time he issued the last TRANSFER command.

Below is a description of the interrupts that may occur in this state.

INTERRUPTS (7-0): 0000 0010

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:

A REQ has been received from a target that the chip cannot service automatically. This may occur prior to issuing a TRANSFER command, when a REQ is received after TC=0 during a TRANSFER command, or when an information phase change is detected by the chip during a TRANSFER command.

SUGGESTED RESPONSE:

Determine if an information phase change has occurred by comparing I/O, C/D, and MSG in the Auxiliary Status with the past information phase. If the phase type changed, read the Transfer counter and update working pointers for the old

C5.0 USER CONNECTED AS INITIATOR ISR

phase and then proceed to set up for the new transfer. If the phase did not change, then the Transfer Counter Zero bit should be set and a buffer overflow has occurred.

INTERRUPTS (7-0): 0000 0100

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:
The target disconnected from the bus. The disconnection may or may not be unexpected depending on the previous sequence of events.

SUGGESTED RESPONSE:
The user does housekeeping to complete his initiator role.

INTERRUPTS (7-0): 0000 0001

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:
A TRANSFER command for a Message In phase has completed. The chip has left ACK active on the bus.

SUGGESTED RESPONSE:
The user should examine the message. If the user wants to reject the message, a SET ATN followed by a MESSAGE ACCEPTED should be issued. If the user wants to accept the message, only the MESSAGE ACCEPTED command should be issued.

INTERRUPTS (7-0): 0000 0100

AUXILIARY STATUS (7-0): X1XX XXXX

REASON FOR INTERRUPT:
The target disconnected from the bus at a time when ATN was on due to a parity error.

SUGGESTED RESPONSE:
The user should consider this I/O to be bad since the target never sent a Message Out to find out about the parity error. The I/O should probably be aborted.

C5.0 USER CONNECTED AS INITIATOR ISR

INTERRUPTS (7-0): 0000 0010

AUXILIARY STATUS (7-0): X1XX XXXX

REASON FOR INTERRUPT:
A REQ from the target cannot be serviced automatically by the chip. Also, a parity error occurred during the last TRANSFER INFO command. The interrupt occurs not at the time of the parity error, but when TC=0 or when the target changes information phases. The chip automatically sets ATN at the time of the parity error.

SUGGESTED RESPONSE:
The user should determine if a phase change occurred by using I/O, C/D, and MSG. If the phase changed and the new phase is a Message Out, the user sends either a "Message Parity Error" or a "Retry" message depending on whether the last phase was a message phase. If the phase changed and the new phase is not a Message Out, the user should service the new phase and issue a TRANSFER command. (The chip will keep ATN on until a Message Out is sent with TC=0.) If the phase did not change and the TC=0, then a buffer overflow occurred in addition to the parity error.

INTERRUPTS (7-0): 0100 0000

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:
The user issued a command that is not valid in the connected initiator state.

SUGGESTED RESPONSE:
If the command is valid, retry or issue CHIP RESET and then retry the entire operation.

INTERRUPTS (7-0): All Others

AUXILIARY STATUS (7-0): XXXX XXXX

REASON FOR INTERRUPT:
Chip malfunction.

SUGGESTED RESPONSE:
Issue chip reset and retry operation.

D1.0 MICROPROCESSOR INTERFACE

The signals used to interface the SCSI chip to the microprocessor are listed in Section 2.1. An example of how these signals are interfaced is shown in Figure D1. The SCSI chip is attached to the microprocessor's Address, Data, and Control Bus as shown. For high transfer rates, the DMA CONTROL LOGIC block should be included so that the SCSI interface will be allowed to "steal" memory cycles. This is achieved through the use of two handshake signals, DREQ and DACK/.

If high speed is not required, the DMA CONTROL LOGIC block may be omitted and DACK/ must be externally pulled up. In this case, all data transfers will be accomplished by means of CS/ and the address lines.

Maximum data transfer rate would be accomplished if the CONTROL LOGIC were replaced with a dedicated data buffer and associated control logic.

D2.0 SCSI INTERFACE

D2.0 SCSI INTERFACE

The suggested interface between the chip and the SCSI differential transceivers is shown in Figure D2. Figure D3 shows the suggested interface between the SCSI chip and single-ended drivers and receivers. In each case, the 3 to 8 decoder enables the ID bit onto the SCSI bus when BSYOUT is active but TGS and IGS are inactive (Arbitration phase). At the same time, all other data bit receivers are enabled for reading and the SCSI chip drives the selected ID data bit high.

The Data Bus drivers are enabled together when DBEN/ is low. At all other times, the Data Bus receivers are enabled.

The ACK and ATN drivers are enabled when IGS is active (Initiator Role). The MSG, C/D, I/O, and REQ drivers are enabled when TGS is active (Target Role).

D2.0 SCSI INTERFACE

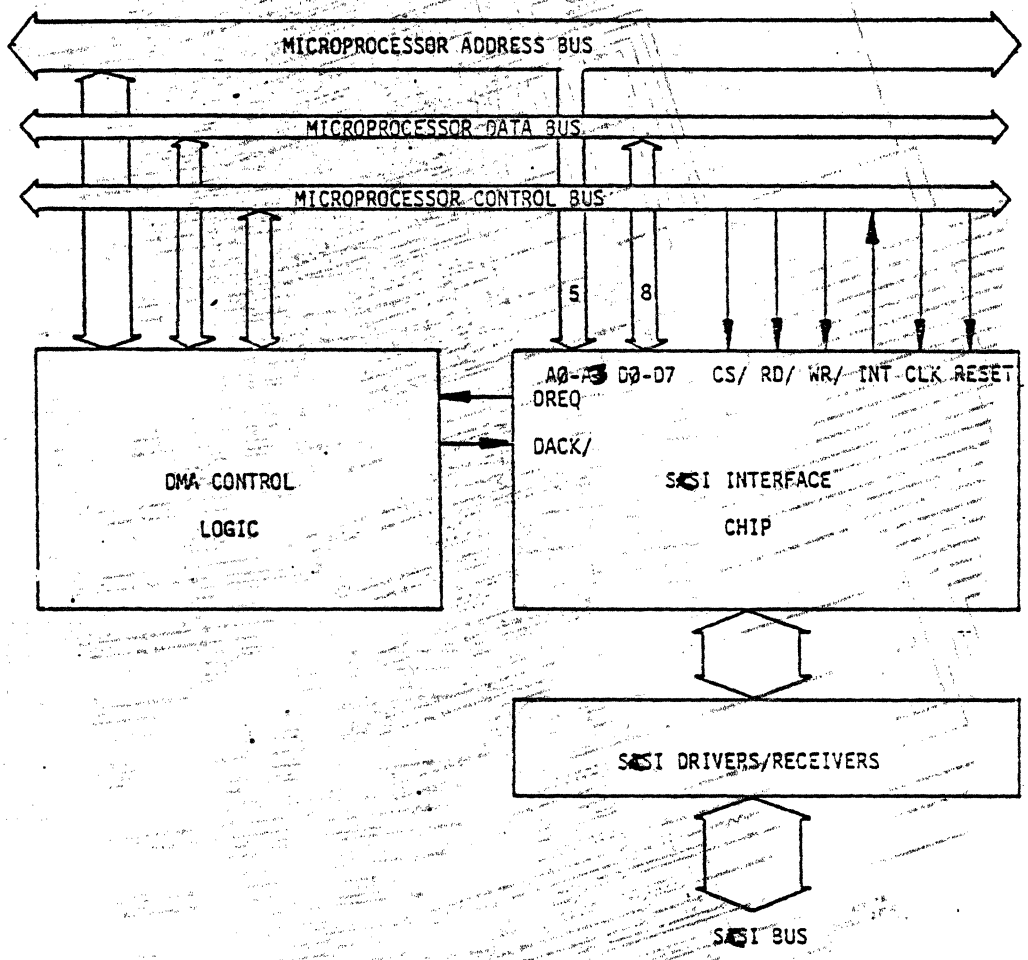


Figure D1. Suggested Microprocessor Interface

FUNCTIONAL DEFINITION
SCSI INTERFACE CHIP P/N: 348-0010342 Rev.A
APPENDIX D - SUGGESTED HARDWARE INTERFACE

D2.0 SCSI INTERFACE

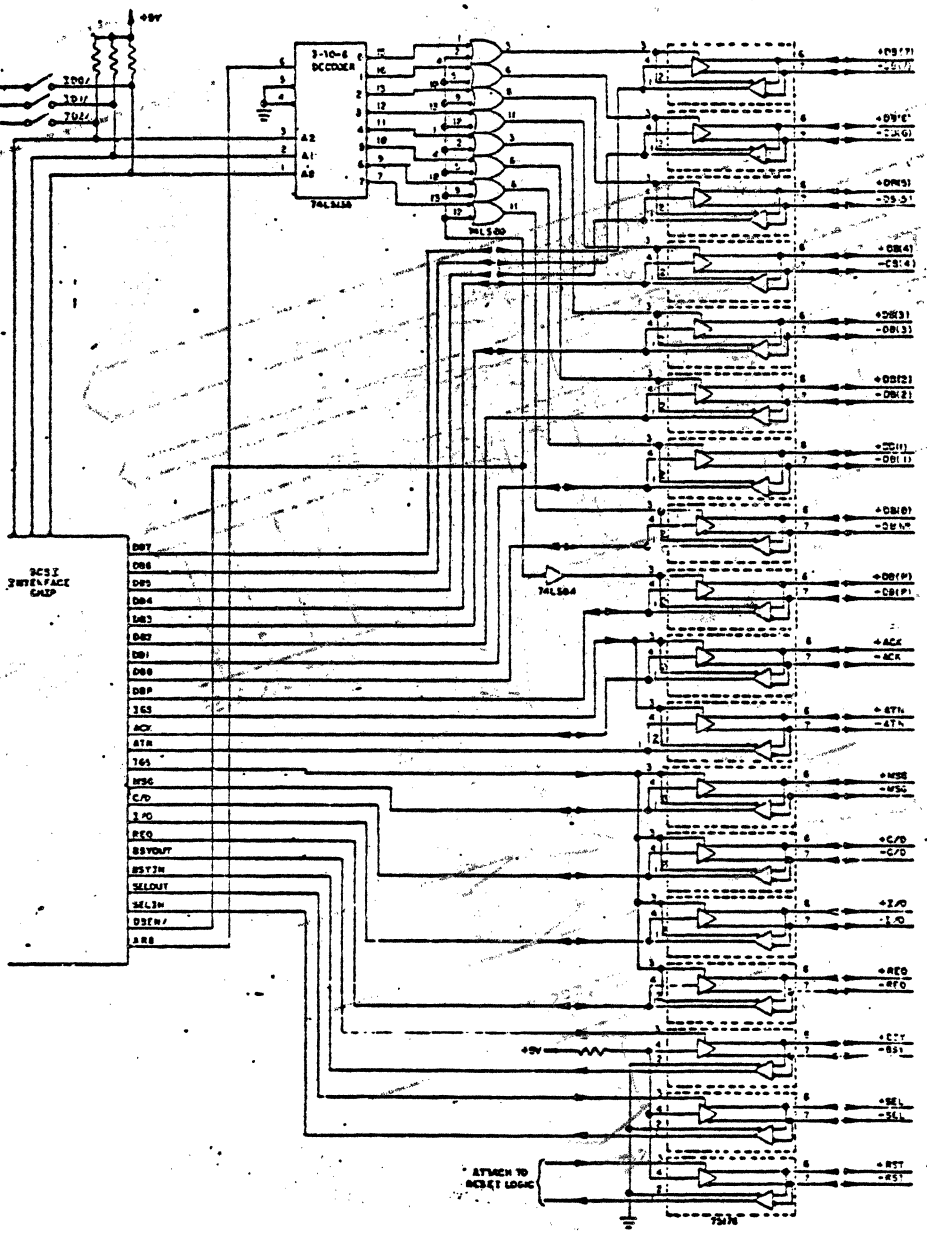


Figure D2. Suggested Interface to SCSI Differential Transceivers

FUNCTIONAL DEFINITION
SCSI INTERFACE CHIP P/N: 348-0010342 Rev.A
APPENDIX D - SUGGESTED HARDWARE INTERFACE

D2.0 SCSI INTERFACE

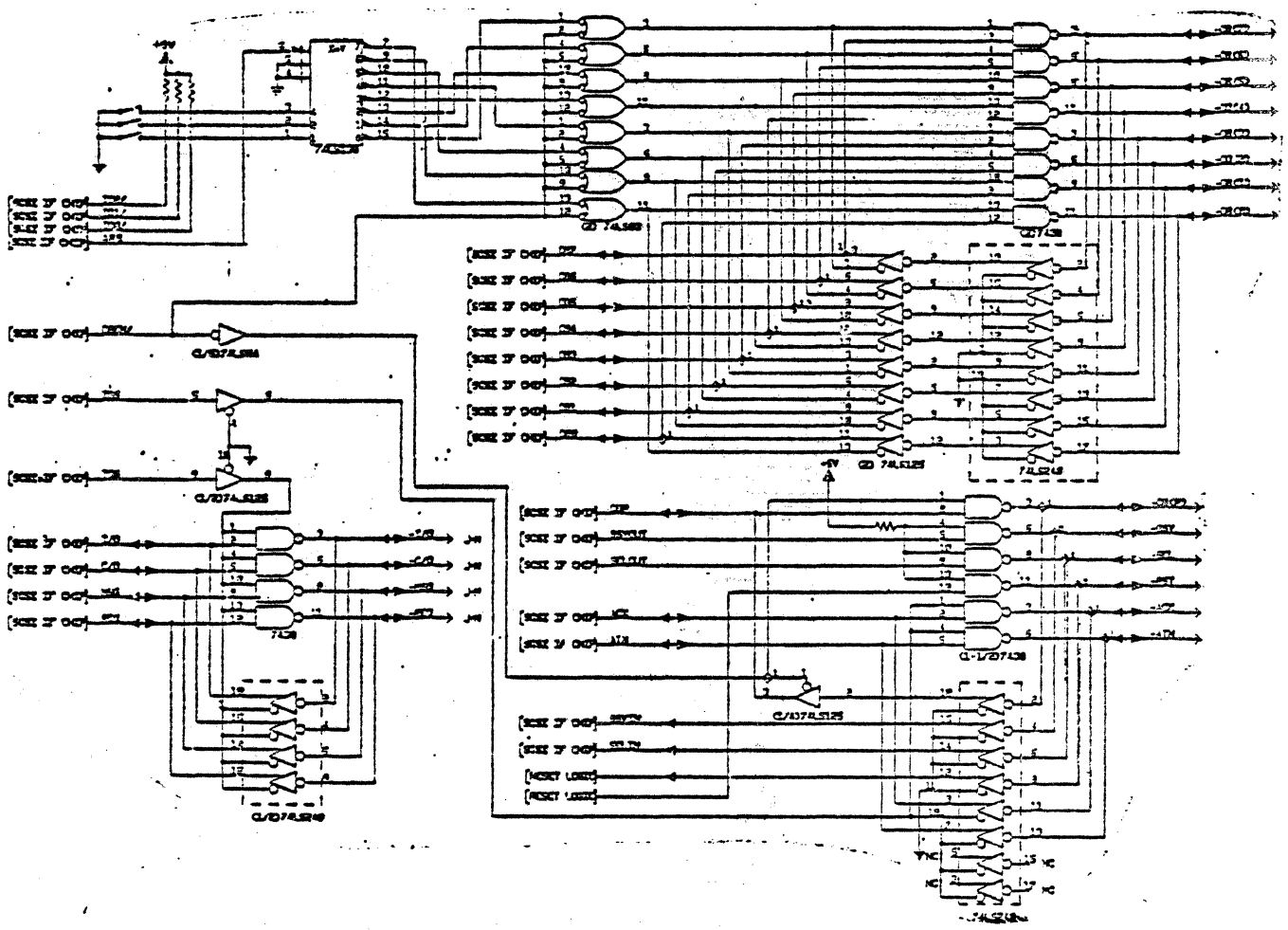


Figure D3. Suggested Interface to SCSI

Single-Ended Drivers/Receivers