
NCR 5380-53C80 SCSI

Interface Chip

Design Manual



NCR Microelectronics Division, Colorado Springs

**Copyright © 1985, by NCR Corporation
Dayton, Ohio
All Rights Reserved Printed in U.S.A.**

This document contains the latest information available at the time of publication. However, NCR reserves the right to modify the contents of this material at any time. Also, all features, functions and operations described herein may not be marketed by NCR in all parts of the world. Therefore, before using this document, consult your NCR representative or NCR office for the information that is applicable and current.

TABLE OF CONTENTS

SECTION	PAGE
1. General Description	3
2. SCSI Background	4
3. Block Diagram	5
4. Pin Description	
4.1 Microprocessor Interface Signals	6
4.2 SCSI Interface Signals	8
5. Electrical Characteristics	9
6. Internal Registers	
6.0 General	10
6.1 Data Registers	10
6.1.1 Current SCSI Data Register	10
6.1.2 Output Data Register	10
6.1.3 Input Data Register	11
6.2 Initiator Command Register	11
6.3 Mode Register	13
6.4 Target Command Register	14
6.5 Current SCSI Bus Status Register	15
6.6 Select Enable Register	15
6.7 Bus and Status Register	15
6.8 DMA Registers	16
6.8.1 Start DMA Send	16
6.8.2 Start DMA Target Receive	17
6.8.3 Start DMA Initiator Receive	17
6.9 Reset Parity/Interrupt	17
7. On-Chip SCSI Hardware Support	18
8. Interrupts	
8.1 Selection/ Reselection	19
8.2 End of Process (EOP) Interrupt	20
8.3 SCSI Bus Reset	21
8.4 Parity Error	21
8.5 Bus Phase Mismatch	22
8.6 Loss of BSY	22

9.	Reset Conditions	23
9.1	Hardware Chip Reset	23
9.2	SCSI Bus Reset Received	23
9.3	SCSI Bus Reset Issued	23
10.	Data Transfers	24
10.1	Programmed I/O Transfers	24
10.2	Normal DMA Mode	24
10.3	Block Mode DMA	24
10.4	Pseudo DMA Mode	25
10.5	Halting a DMA Operation	25
10.5.1	Using the EOP Signal	25
10.5.2	Bus Phase Mismatch Interrupt	25
10.5.3	Resetting the DMA MODE Bit	25
11.	External Timing Diagrams	
11.1	CPU Write	26
11.2	CPU Read	26
11.3	DMA Write (Non-block Mode) Target Send	27
11.4	DMA Write (Non-block Mode) Initiator Send	28
11.5	DMA Read (Non-block Mode) Target Receive	29
11.6	DMA Read (Non-block Mode) Initiator Receive	30
11.7	DMA Write (Block Mode) Target Send	31
11.8	DMA Read (Block Mode) Target Receive	32
11.9	Reset	33
11.10	Arbitration	33

APPENDICES

A1.	NCR 5380 vs. NCR 5385/86	34
A2.	Flowcharts/Software	35
A3.	65C02 Interface Diagram	50
A4.	NCR 5381 Differential Pair Option	51
A5.	NCR 53C80	54
A6.	SCSI/PLUS*	56
A7.	Register Reference Chart	58

*SCSI/PLUS is a trademark of AMPRO Computers, Inc.

SECTION 1

GENERAL DESCRIPTION

The NCR 5380 SCSI interface device is a 40 pin NMOS device designed to accommodate the Small Computer Systems Interface (SCSI) as defined by the ANSI X3T9.2 committee. The NCR 5380 operates in both the initiator and target roles and can therefore be used in host adapter, host port and formatter designs. This device supports arbitration, including reselection. Special high-current open collector output drivers, capable of sinking 48mA at 0.5V, allow for direct connection to the SCSI bus. Differential pair operation is supported using a 48 pin version of this part, designated the NCR 5381 (refer to Appendix A4).

The NCR 5380 communicates with the system microprocessor as a peripheral device. The chip is controlled by reading and writing several internal registers which may be addressed as standard or memory mapped I/O. Minimal processor intervention is required for DMA transfers because the 5380 controls the necessary handshake signals. The NCR 5380 interrupts the MPU when it detects a bus condition that requires attention. Normal and block mode DMA is provided to match many popular DMA controllers. **A CMOS version of the NCR 5380, the NCR 53C80 is available in a 44-pin PLCC or a 48-pin DIP. Refer to Appendix A5 for information on the NCR 53C80.**

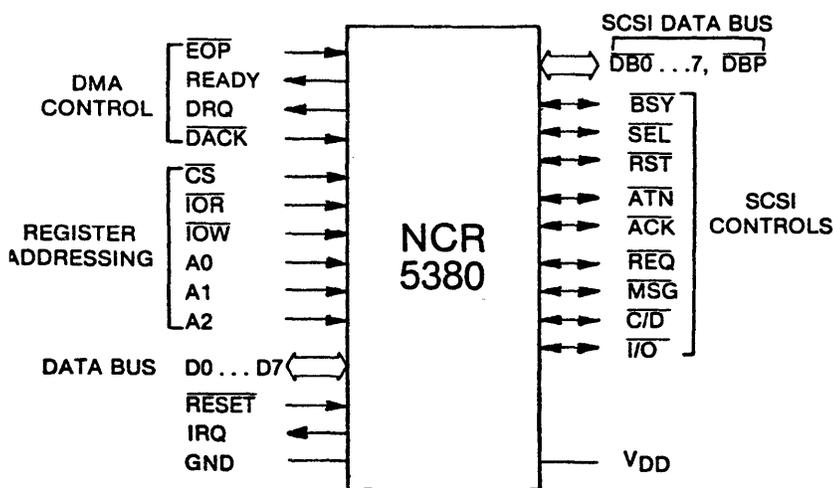
SCSI INTERFACE

- * Asynchronous, interface to 1.5 MBPS
- * Supports initiator and target roles
- * Parity generation w/optional checking
- * Supports arbitration
- * Direct control of all bus signals
- * High current outputs drive SCSI bus directly

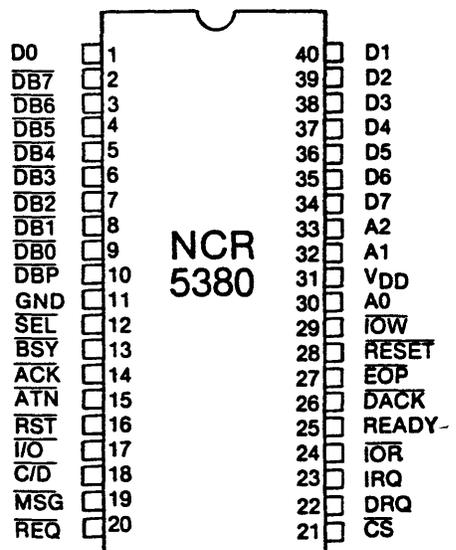
MPU INTERFACE

- * Memory or I/O mapped interface
- * DMA or programmed I/O
- * Normal or block mode DMA
- * Optional MPU interrupts

FUNCTIONAL PIN GROUPING



PINOUT



SECTION 2

SCSI BACKGROUND

SCSI (Small Computer Systems Interface) has evolved from the SASI (Shugart Associates Systems Interface) disk controller interface standard developed by Shugart Associates in the late 1970's. NCR and Shugart jointly approached the ANSC X3T9.3 subcommittee in December of 1981 and proposed that a committee be formed to develop an intelligent interface standard based on SASI. The ANSC X3T9.3 subcommittee divided into two groups so that SASI could be pursued. In February of 1982, NCR and Shugart Associates presented SASI as a working document. It was agreed that a separate group should develop the standard and the ANSC X3T9.2 subcommittee was established. This group met in April of that year and formally changed the name to the Small Computer Systems Interface (SCSI).

The proposed standard has since been forwarded from the subcommittee and is becoming a major industry standard. It is expected that other standards organizations such as ECMA (European Computer Manufactures Association) and ISO (International Standards Organization) will adopt the proposed standard as well.

NCR Microelectronics announced the NCR 5385, the first SCSI protocol controller, in April of 1983. This product family includes the NCR 5386 and the soon-to-be-announced NCR 5386S. The NCR 5380 and 5381 were designed to compliment this initial offering. Differences between the product families are described in Appendix A1.

This design manual is not an SCSI specification and assumes some prior knowledge of the SCSI proposed standard. Copies of the proposed standard may be obtained, with pre-payment of \$20, from:

X3 Secretariat, Computer and Business
Equipment Manufacturers Association
311 First Street, NW, Suite 500
Washington, D.C. 20001

Please include a self-addressed mailing label.

Other documents which may be useful are:

- NCR 5386 SCSI Protocol Controller Data Sheet (MC-704)
- NCR 5385E/86 SCSI Protocol Controller User's Guide (MC-903)
- SCSI Engineering Notebook

These documents may be obtained by contacting your local NCR Microelectronics sales representative or regional sales office.

Eastern Area

NCR Microelectronics Division
400 W. Cummings Park
Suite 2750
Woburn, MA 01801
Phone: (617) 933-0778

Central Area

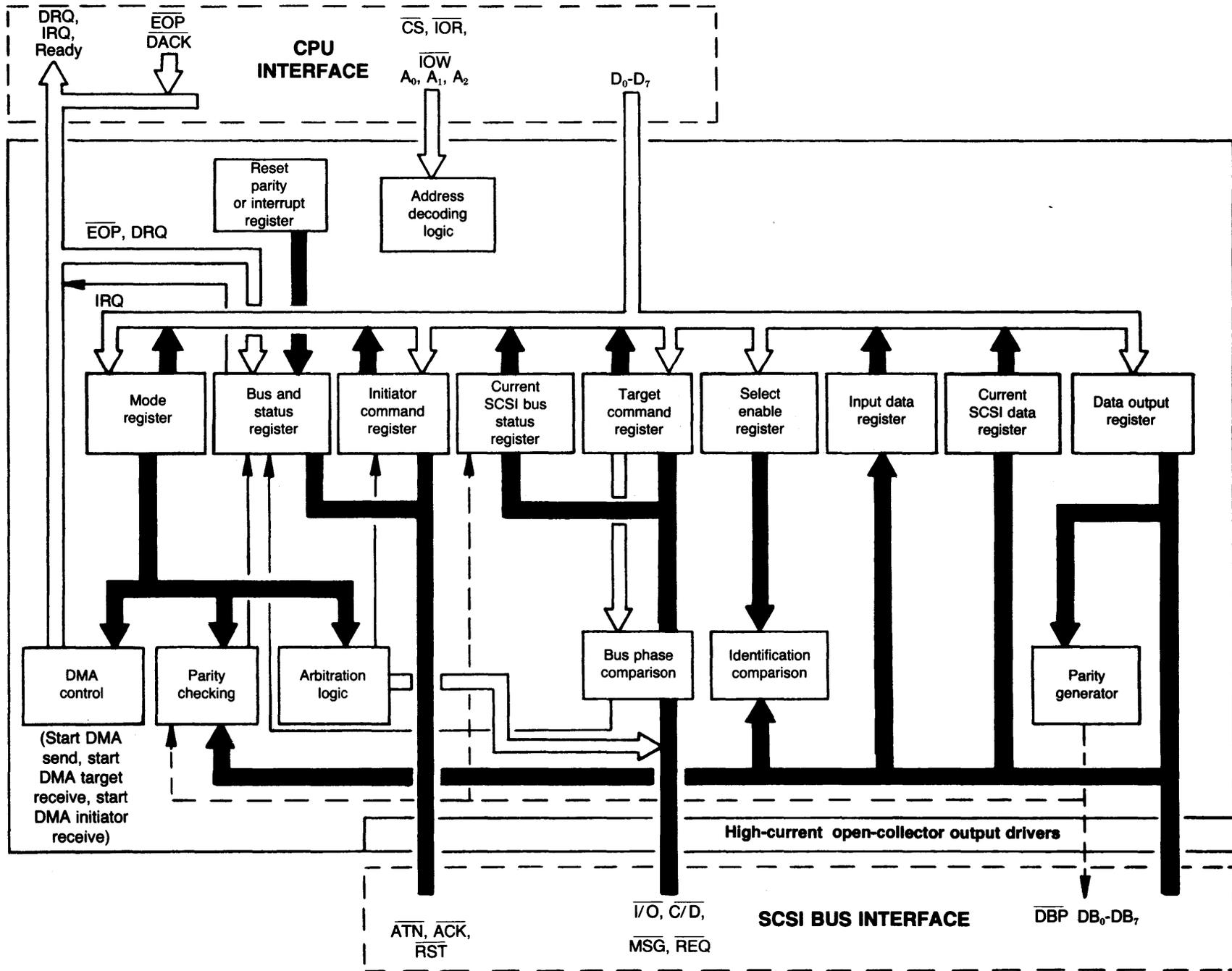
NCR Microelectronics Division
400 Chisholm Place
Suite 100
Plano, TX 75075
Phone: (214) 578-9113

Western Area

NCR Microelectronics Division
3130 De La Cruz Boulevard
Suite 209
Santa Clara, CA 95054-2410
Phone: (408) 727-6575

Southwest Area

NCR Microelectronics Division
1940 Century Park East
Los Angeles, CA 90067
Phone: (213) 556-5396



SECTION 3
BLOCK DIAGRAM

SECTION 4

PIN DESCRIPTION

4.1 Microprocessor Interface Signals

Pin Name	Pin #	Description
A0, A1, A2	30, 32, 33	<p>INPUTS</p> <p>These signals are used with \overline{CS}, \overline{IOR} or \overline{IOW} to address all internal registers.</p>
\overline{CS}	21	<p>INPUT</p> <p>Chip Select enables a read or write of the internal register selected by A0, A1 and A2. \overline{CS} is an active low signal.</p>
\overline{DACK}	26	<p>INPUT</p> <p>DMA Acknowledge resets DRQ and selects the data register for input or output data transfers. \overline{DACK} is an active low signal.</p>
DRQ	22	<p>OUTPUT</p> <p>DMA Request indicates that the data register is ready to be read or written. DRQ occurs only if DMA mode is true in the Command Register. It is cleared by \overline{DACK}.</p>
D0 . . . D7	1,40 . . . 34	<p>BI-DIRECTIONAL, TRI-STATE</p> <p>Microprocessor data bus active high</p>
\overline{EOP}	27	<p>INPUT</p> <p>The End of Process signal is used to terminate a DMA transfer. If asserted during a DMA cycle, the current byte will be transferred but no additional bytes will be requested.</p>
\overline{IOR}	24	<p>INPUT</p> <p>I/O Read is used to read an internal register selected by \overline{CS} and A0, A1 and A2. It also selects the Input Data Register when used with \overline{DACK}. \overline{IOR} is active low.</p>

Pin Name	Pin #	Description
$\overline{\text{IOW}}$	29	INPUT I/O Write is used to write an internal register selected by $\overline{\text{CS}}$ and A0, A1 and A2. It also selects the Output Data Register when used with $\overline{\text{DACK}}$. $\overline{\text{IOW}}$ is active low.
IRQ	23	OUTPUT Interrupt Request alerts a microprocessor of an error condition or an event completion.
READY	25	OUTPUT Ready can be used to control the speed of block mode DMA transfers. This signal goes active to indicate the chip is ready to send/receive data and remains false after a transfer until the last byte is sent or until the DMA Mode bit is reset.
$\overline{\text{RESET}}$	28	INPUT Reset clears all registers. It does not force the SCSI signal $\overline{\text{RST}}$ to the active state. $\overline{\text{RESET}}$ is an active low signal.

Power Signals

Pin Name	Pin #	Description
VDD	31	+5 VOLTS
GND	11	GROUND

4.2 SCSI Interface Signals

The following signals are all bi-directional, active low, open collector signals. With 48 mA sink capability, all pins interface directly with the SCSI bus.

Pin Name	Pin #	Description
$\overline{\text{ACK}}$	14	Driven by an initiator, $\overline{\text{ACK}}$ indicates an acknowledgment for a REQ/ACK data transfer handshake. In the target role, $\overline{\text{ACK}}$ is received as a response to the $\overline{\text{REQ}}$ signal.
$\overline{\text{ATN}}$	15	Driven by an initiator, $\overline{\text{ATN}}$ indicates an attention condition. This signal is received in the target role.
$\overline{\text{BSY}}$	13	This signal indicates that the SCSI bus is being used and can be driven by both the initiator and the target device.
$\overline{\text{C/D}}$	18	A signal driven by the target, $\overline{\text{C/D}}$ indicates Control or Data information is on the data bus. This signal is received by the initiator.
$\overline{\text{I/O}}$	17	$\overline{\text{I/O}}$ is a signal driven by a target which controls the direction of data movement on the SCSI bus. True indicates input to the initiator. This signal is also used to distinguish between Selection and Reselection phases.
$\overline{\text{MSG}}$	19	$\overline{\text{MSG}}$ is a signal driven by the target during the Message phase. This signal is received by the initiator.
$\overline{\text{REQ}}$	20	Driven by a target, $\overline{\text{REQ}}$ indicates a request for a REQ/ACK data transfer handshake. This signal is received by the initiator.
$\overline{\text{RST}}$	16	The $\overline{\text{RST}}$ signal indicates an SCSI bus RESET condition.
$\overline{\text{DB0}} \dots \overline{\text{DB7}}$ $\overline{\text{DBP}}$	9 ... 2 10	These eight data bits ($\overline{\text{DB0}}\text{--}\overline{\text{DB7}}$) plus a parity bit ($\overline{\text{DBP}}$) form the data bus. $\overline{\text{DB7}}$ is the most significant bit and has the highest priority during the Arbitration phase. Data parity is odd. Parity is always generated and optionally checked. Parity is not valid during arbitration.
$\overline{\text{SEL}}$	12	$\overline{\text{SEL}}$ is used by an initiator to select a target or by a target to reselect an initiator.

SECTION 5

ELECTRICAL CHARACTERISTICS

OPERATING CONDITIONS

PARAMETER	SYMBOL	MIN	MAX	UNITS
Supply Voltage	V _{DD}	4.75	5.25	Volts
Supply Current	I _{DD}		145	mA.
Ambient Temperature	T _A	0	70	°C

INPUT SIGNAL REQUIREMENTS

PARAMETER	CONDITIONS	MIN	MAX	UNITS
High-level, Input V _{IH}		2.0	5.25	Volts
Low-level, Input V _{IL}		-0.3	0.8	Volts
SCSI BUS pins 2 . . . 20				
High-level Input Current, I _{IH}	V _{IH} = 5.25 V		50	μa.
Low-level Input Current, I _{IL}	V _{IL} = 0 Volts		-50	μa.
All other pins				
High-level Input Current, I _{IH}	V _{IH} = 5.25 V		10	μa.
Low-level Input Current, I _{IL}	V _{IL} = 0 Volts		-10	μa.

OUTPUT SIGNAL REQUIREMENTS

PARAMETER	CONDITIONS	MIN	MAX	UNITS
SCSI BUS pins 2 . . . 20				
Low-level Output V _{OL}	V _{DD} = 4.75 V I _{OL} = 48.0mA.		0.5	Volts
All other pins				
High-level Output V _{OH}	V _{DD} = 4.75 V I _{OH} = -3.0mA.	2.4		Volts
Low-level Output V _{OL}	V _{DD} = 4.75 V I _{OL} = 7.0 mA.		0.5	Volts

PRELIMINARY

Notice: This is not a final specification.
Some parametric limits are subject to change.

SECTION 6

INTERNAL REGISTERS

6.0 General

The NCR 5380 SCSI Interface Device appears as a set of eight registers to the controlling CPU. By reading and writing the appropriate registers, the CPU may initiate any SCSI bus activity or may sample and assert any signal on the SCSI bus. This allows the user to implement all or portions of the SCSI protocol in software. These registers are read (written) by activating \overline{CS} with an address on A2-A0 and then issuing an \overline{IOR} (\overline{IOW}) pulse. This section describes the operation of the internal registers.

SCSI signal names are used to describe the contents of these internal registers. Even though the bus is active low a one (1) is used to indicate signal assertion and a zero (0) is used to indicate the non-asserted or inactive state.

Address			R/W	Register Name
A2	A1	A0		
0	0	0	R	Current SCSI Data
0	0	0	W	Output Data
0	0	1	R/W	Initiator Command
0	1	0	R/W	Mode
0	1	1	R/W	Target Command
1	0	0	R	Current SCSI Bus Status
1	0	0	W	Select Enable
1	0	1	R	Bus and Status
1	0	1	W	Start DMA Send
1	1	0	R	Input Data
1	1	0	W	Start DMA Target Receive
1	1	1	R	Reset Parity/Interrupts
1	1	1	W	Start DMA Initiator Receive

Register Summary

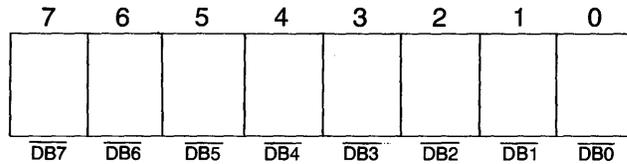
6.1 Data Registers

The data registers are used to transfer SCSI commands, data, status, and message bytes between the microprocessor data bus and the SCSI bus. The NCR 5380 does not interpret any information that passes through the data registers. The data registers consist of the transparent Current SCSI Data Register, the Output Data Register, and the Input Data Register.

6.1.1 Current SCSI Data Register— Address 0 (Read-only)

The Current SCSI Data Register is a read-only register which allows the microprocessor to read the active SCSI data bus. This is accomplished by activating \overline{CS} with an address on A2-A0 of 000 and issuing an \overline{IOR} pulse. If parity checking is enabled, the SCSI bus parity is checked at the beginning of the read cycle. This register is used during a programmed I/O data read or during arbitration to check for higher priority arbitrating devices. Parity is not guaranteed valid during arbitration.

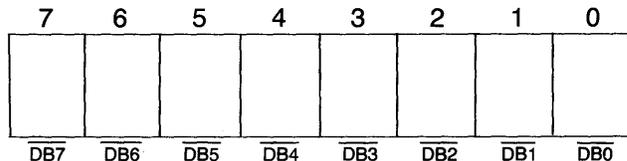
Current SCSI Data Register



6.1.2 Output Data Register— Address 0 (write-only)

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

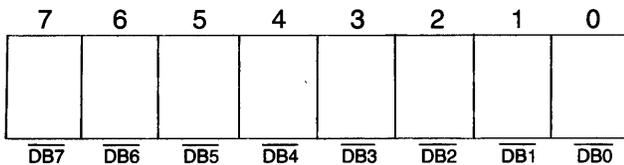
Output Data Register



6.1.3 Input Data Register— Address 6 (Read-only)

The Input Data Register is a read-only register that is used to read latched data from the SCSI bus. Data is latched either during a DMA Target receive operation when \overline{ACK} (pin 14) goes active or during a DMA Initiator receive when \overline{REQ} (pin 20) goes active. The DMA Mode bit (port 2, bit 1) must be set before data can be latched in the Input Data Register. This register may be read under DMA control using \overline{IOR} and \overline{DACK} . Parity is optionally checked when the Input Data Register is loaded.

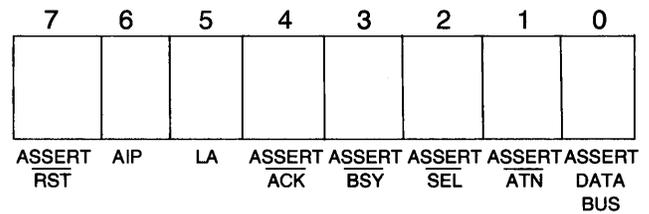
Input Data Register



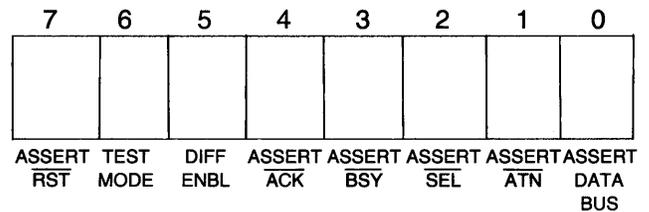
6.2 Initiator Command Register— Address 1 (Read/Write)

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

**Initiator Command Register
(Register Read)**



**Initiator Command Register
(Register Write)**



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

BIT 7—ASSERT \overline{RST}

Whenever a one (1) is written to bit 7 of the Initiator Command Register, the \overline{RST} signal (pin 16) is asserted on the SCSI bus. The \overline{RST} signal will remain asserted until this bit is reset or until an external \overline{RESET} (pin 28) occurs. After this bit is set (1), IRQ (pin 23) goes active and all internal logic and control registers are reset (except for the interrupt latch and the ASSERT \overline{RST} bit). Writing a zero (0) to bit 7 of the Initiator Command Register de-asserts the \overline{RST} signal. Reading this register simply reflects the status of this bit.

BIT 6—AIP (Arbitration in Progress—read bit)

This bit is used to determine if arbitration is in progress. For this bit to be active, the ARBITRATE bit (port 2, bit 0) must have been set previously. It indicates that a bus free condition has been detected and that the chip has asserted \overline{BSY} (pin 13) and the contents of the Output Data Register (port 0) onto the SCSI bus. AIP will remain active until the ARBITRATE bit is reset.

BIT 6—TEST MODE (write bit)

This bit may be written during a test environment to disable all output drivers, effectively removing the NCR 5380 from the circuit. Resetting this bit returns the part to normal operation.

BIT 5—LA (Lost Arbitration—read bit)

This bit, when active, indicates that the NCR 5380 detected a bus free condition, arbitrated for use of the bus by asserting \overline{BSY} (pin 13) and its ID on the data bus and lost arbitration due to \overline{SEL} (pin 12) being asserted by another bus device. For this bit to be active the ARBITRATE bit (port 2, bit 0) must be active.

BIT 5—DIFF ENBL (Differential Enable—write bit)

This bit is not used in the NCR 5380 and is only meaningful in the NCR 5381, a 48 pin device which supports external differential pair transceivers. DIFF ENBL should only be asserted if the device is physically connected as either an Initiator or as a Target. If enabled, the signal TGS (pin 14—NCR 5381) is asserted if the TARGETMODE bit (port 2, bit 6) is set (1) or the signal IGS (pin 12—NCR 5381) is asserted if the TARGETMODE bit is reset (0).

BIT 4—ASSERT \overline{ACK}

This bit is used by the bus initiator to assert \overline{ACK} (pin 14) on the SCSI bus. In order to assert \overline{ACK} the TARGETMODE bit (port 2, bit 6) must be false. Writing a zero to this bit resets \overline{ACK} on the SCSI bus. Reading this register simply reflects the status of this bit.

BIT 3—ASSERT \overline{BSY}

Writing a one (1) into this bit position asserts \overline{BSY} (pin 13) onto the SCSI bus. Conversely, a zero (0) resets the \overline{BSY} signal. Asserting \overline{BSY} indicates a successful selection or reselection and resetting this bit creates a bus disconnect condition. Reading this register simply reflects the status of this bit.

BIT 2—ASSERT \overline{SEL}

Writing a one (1) into this bit position asserts \overline{SEL} (pin 12) onto the SCSI bus. \overline{SEL} is normally asserted after arbitration has been successfully completed. \overline{SEL} may be de-asserted by resetting this bit to a zero. A read of this register simply reflects the status of this bit.

BIT 1—ASSERT \overline{ATN}

\overline{ATN} (pin 15) may be asserted on the SCSI bus by setting this bit to a one (1) if the TARGETMODE bit (port 2, bit 6) is false. \overline{ATN} is normally asserted by the initiator to request a Message Out bus phase. Note that since ASSERT \overline{SEL} and ASSERT \overline{ATN} are in the same register, a select with \overline{ATN} may be implemented with one MPU write. \overline{ATN} may be de-asserted by resetting this bit to a zero (0). A read of this register simply reflects the status of this bit.

BIT 0—ASSERT DATA BUS

The ASSERT DATA BUS bit, when set, allows the contents of the Output Data Register to be enabled as chip outputs on the signals $\overline{DB0-DB7}$. Parity is also generated and asserted on \overline{DBP} . In the NCR 5381, this bit asserts the \overline{DBEN} signal (pin 36). Resetting this bit disables the output data bus or the \overline{DBEN} signal.

When connected as an Initiator, the outputs are only enabled if the TARGETMODE bit (port 2, bit 6) is false, the received signal $\overline{I/O}$ (pin 17) is false, and the phase signals ($\overline{C/D}$, $\overline{I/O}$, and \overline{MSG}) match the contents of the ASSERT $\overline{C/D}$, ASSERT $\overline{I/O}$, and ASSERT \overline{MSG} in the Target Command Register.

This bit should also be set during DMA send operations.

6.3 Mode Register—Address 2 (Read/Write)

The Mode Register is used to control the operation of the chip. This register determines whether the NCR 5380 operates as an initiator or a target, whether DMA transfers are being used, whether parity is checked, and whether interrupts are generated on various external conditions. This register may be read to check the value of these internal control bits. The following describes the operation of these control bits.

Mode Register

7	6	5	4	3	2	1	0
BLOCK MODE DMA	TARGET MODE	ENABLE PARITY CHECK- ING	ENABLE PARITY INTER- RUPT	ENABLE EOP INTER- RUPT	MONITOR BUSY	DMA MODE	ARBI- TRATE

BIT 7—BLOCK MODE DMA

The BLOCK MODE DMA bit controls the characteristics of the DMA DRQ- \overline{DACK} handshake. When this bit is reset (0) and the DMA MODE bit is active (1), the DMA handshake uses the normal interlocked handshake and the rising edge of \overline{DACK} (pin 26) indicates the end of each byte being transferred. In block mode operation, BLOCK MODE DMA bit set (1) and DMA MODE bit set (1), the end of \overline{IOR} (pin 24) or \overline{IOW} (pin 29) signifies the end of each byte transferred and \overline{DACK} is allowed to remain active throughout the DMA operation. READY (pin 25) can then be used to request the next transfer.

BIT 6—TARGETMODE

The TARGETMODE bit allows the NCR 5380 to operate as either an SCSI bus initiator, bit reset (0), or as an SCSI bus target device, bit set (1). In order for the signals \overline{ATN} (pin 15) and \overline{ACK} (pin 14) to be asserted on the SCSI bus, the TARGETMODE bit must be reset (0). In order for the signals $\overline{C/D}$, $\overline{I/O}$, \overline{MSG} and \overline{REQ} to be asserted on the SCSI bus, the TARGETMODE bit must be set (1).

BIT 5—ENABLE PARITY CHECKING

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

BIT 4—ENABLE PARITY INTERRUPT

The ENABLE PARITY INTERRUPT bit, when set (1), will cause an interrupt (IRQ) to occur if a parity error is detected. A parity interrupt will only be generated if the ENABLE PARITY CHECKING bit (bit 5) is also enabled (1).

BIT 3—ENABLE EOP INTERRUPT

The ENABLE EOP INTERRUPT, when set (1), causes an interrupt to occur when an \overline{EOP} (End of Process) signal (pin 27) is received from the DMA controller logic.

BIT 2—MONITOR BUSY

The MONITOR BUSY bit, when true (1), causes an interrupt to be generated for an unexpected loss of \overline{BSY} (pin 13). When the interrupt is generated due to loss of \overline{BSY} , the lower 6 bits of the Initiator Command Register are reset (0) and all signals are removed from the SCSI bus.

BIT 1—DMA MODE

The DMA MODE bit is normally used to enable a DMA transfer and must be set (1) prior to writing ports 5 through 7. Ports 5 through 7 are used to start DMA transfers. The TARGETMODE bit (port 2, bit 6) must be consistent with writes to port 6 and 7 [i.e. set (1) for a write to port 6 and reset (0) for a write to port 7]. The control bit ASSERT DATA BUS (port 1, bit 0) must be true (1) for all DMA send operations. In the DMA mode, $\overline{\text{REQ}}$ (pin 20) and $\overline{\text{ACK}}$ (pin 14) are automatically controlled.

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

Note: $\overline{\text{BSY}}$ must be active in order to set the DMA Mode bit.

BIT 0—ARBITRATE

The ARBITRATE bit is set (1) to start the arbitration process. Prior to setting this bit the Output Data Register should contain the proper SCSI device ID value. Only one data bit should be active for SCSI bus arbitration. The NCR 5380 will wait for a bus free condition before entering the arbitration phase. The results of the arbitration phase may be determined by reading the status bits LA and AIP (port 1, bits 5 & 6 respectively).

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

When connected as a target device, the Target Command Register allows the MPU to control the SCSI bus information transfer phase and/or to assert $\overline{\text{REQ}}$ (pin 20) simply by writing this register. The TARGETMODE bit (port 2, bit 6) must be true (1) for bus assertion to occur. The SCSI bus phases are described in the following table.

SCSI Information Transfer Phases

Bus Phase	ASSERT	ASSERT	ASSERT
	$\overline{\text{I/O}}$	$\overline{\text{C/D}}$	$\overline{\text{MSG}}$
Data Out	0	0	0
Unspecified	0	0	1
Command	0	1	0
Message Out	0	1	1
Data In	1	0	0
Unspecified	1	0	1
Status	1	1	0
Message In	1	1	1

When connected as an Initiator with DMA Mode true, if the phase lines (I/O, C/D, and MSG) do not match the phase bits in the Target Command Register, a phase mismatch interrupt is generated when $\overline{\text{REQ}}$ (pin 20) goes active. In order to send data as an Initiator, the ASSERT $\overline{\text{I/O}}$, ASSERT $\overline{\text{C/D}}$, and ASSERT $\overline{\text{MSG}}$ bits must match the corresponding bits in the Current SCSI Bus Status Register (port 4). The ASSERT $\overline{\text{REQ}}$ bit (bit 3) has no meaning when operating as an Initiator.

Target Command Register

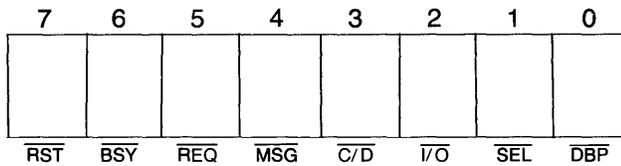
7	6	5	4	3	2	1	0
LAST BYTE SENT (53C80)	X	X	X	ASSERT REQ	ASSERT MSG	ASSERT C/D	ASSERT I/O

The NCR 53C80 uses bit 7 of this register to determine when the last byte of a DMA transfer is sent to the SCSI bus. This flag is necessary since the End of DMA bit in the Bus and Status Register only reflects when the last byte was received from the DMA.

6.5 Current SCSI Bus Status Register—Address 4 (Read-only)

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

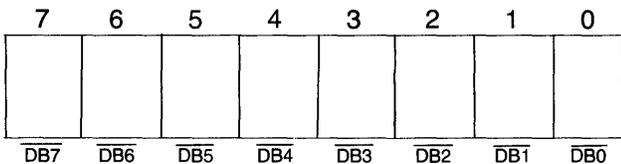
Current SCSI Bus Status Register



6.6 Select Enable Register—Address 4 (Write-only)

The Select Enable Register is a write-only register which is used as a mask to monitor a single ID during a selection attempt. The simultaneous occurrence of the correct ID bit, \overline{BSY} false, and \overline{SEL} true will cause an interrupt. This interrupt can be disabled by resetting all bits in this register. If the ENABLE PARITY CHECKING bit (port 2, bit 5) is active (1), parity will be checked during selection.

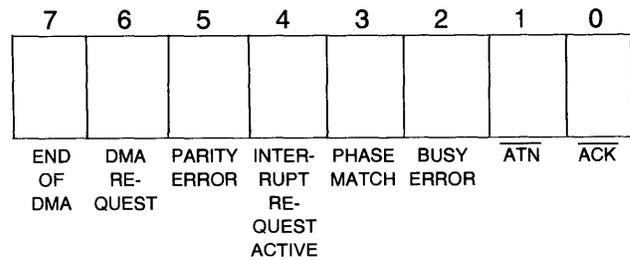
Select Enable Register



6.7 Bus and Status Register—Address 5 (Read-only)

The Bus and Status Register is a read-only register which can be used to monitor the remaining SCSI control signals not found in the Current SCSI Bus Status Register (\overline{ATN} & \overline{ACK}) as well as six other status bits. The following describes each bit of the Bus and Status Register individually.

Bus and Status Register



BIT 7—END OF DMA TRANSFER

The END OF DMA TRANSFER bit is set if \overline{EOP} (pin 27), \overline{DACK} (pin 26), and either \overline{IOR} (pin 24), or \overline{IOW} (pin 29) are simultaneously active for at least 100 nsec. Since the \overline{EOP} signal can occur during the last byte sent to the Output Data Register (port 0), the \overline{REQ} and \overline{ACK} signals should be monitored to insure that the last byte has been transferred. This bit is reset when the DMA MODE bit is reset (0) in the Mode Register (port 2).

The NCR 53C80 contains a true End of DMA Status bit (last byte sent) in bit 7 of the Target Command Register.

BIT 6—DMA REQUEST

The DMA REQUEST bit allows the MPU to sample the output pin \overline{DRQ} (pin 22). \overline{DRQ} can be cleared by asserting \overline{DACK} (pin 26) or by resetting the DMA MODE bit (bit 1) in the Mode Register (port 2). The \overline{DRQ} signal does not reset when a phase mismatch interrupt occurs.

BIT 5—PARITY ERROR

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

BIT 4—INTERRUPT REQUEST ACTIVE

This bit is set if an enabled interrupt condition occurs. It reflects the current state of the IRQ (pin 23) output and can be cleared by reading the Reset Parity/Interrupt Register (port 7).

BIT 3—PHASE MATCH

The SCSI signals \overline{MSG} , $\overline{C/D}$, and $\overline{I/O}$ (pins 19, 18, and 17) represent the current information transfer phase. The PHASE MATCH bit indicates whether the current SCSI bus phase matches the lower 3 bits of the Target Command Register. PHASE MATCH is continuously updated and is only significant when operating as a bus initiator. A Phase Match is required for data transfers to occur on the SCSI bus.

BIT 2—BUSY ERROR

The BUSY ERROR bit is active if an unexpected loss of the \overline{BSY} signal (pin 13) has occurred. This level-sensitive latch is set whenever the MONITOR BUSY bit (port 2, bit 2) is true and \overline{BSY} is false. An unexpected loss of \overline{BSY} will disable any SCSI outputs and will reset the DMA MODE bit (port 2, bit 1).

BIT 1—ATN

This bit reflects the condition of the SCSI bus control signal \overline{ATN} (pin 15). This signal is normally monitored by the target device.

BIT 0—ACK

This bit reflects the condition of the SCSI bus control signal \overline{ACK} (pin 14). This signal is normally monitored by the target device.

6.8 DMA Registers

Three write-only registers are used to initiate all DMA activity. They are Start DMA Send (port 5), Start DMA Target Receive (port 6) and Start DMA Initiator Receive (port 7). Simply writing these registers starts the DMA transfers. Data presented to the NCR 5380 on signals D0-D7 during the register write is meaningless and has no effect on the operation. Prior to writing these registers the BLOCK MODE DMA bit (bit 7), the DMA MODE bit (bit 1) and the TARGETMODE bit (bit 6) in the Mode Register (port 2) must be appropriately set. The individual registers are briefly described below.

6.8.1 Start DMA Send—Address 5 (Write-only)

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

6.8.2 Start DMA Target Receive—Address 6 (Write-only)

This register is written to initiate a DMA receive, from the SCSI bus to the DMA, for target operation only. The DMA MODE bit (bit 1) and the TARGETMODE bit (bit 6) in the Mode Register (port 2) must both be set (1) prior to writing this register.

6.8.3 Start DMA Initiator Receive—Address 7 (Write-only)

This register is written to initiate a DMA receive, from the SCSI bus to the DMA, for initiator operation only. The DMA MODE bit (bit 1) must be true (1) and the TARGETMODE bit (bit 6) must be false (0) in the Mode Register (port 2) prior to writing this register.

6.9 Reset Parity/Interrupt—Address 7 (Read-only)

Reading this register resets the PARITY ERROR bit (bit 5), the INTERRUPT REQUEST bit (bit 4) and the BUSY ERROR bit (bit 2) in the Bus and Status Register (port 5).

SECTION 7

ON-CHIP SCSI HARDWARE SUPPORT

The NCR 5380 is easy to use because of its simple architecture. The chip allows direct control and monitoring of the SCSI bus by providing a latch for each signal. However, portions of the protocol define timings which are much too quick for traditional microprocessors to control. Therefore, hardware support has been provided for DMA transfers, bus arbitration, phase change monitoring, bus disconnection, bus reset, parity generation, parity checking, and device selection/reselection.

Arbitration is accomplished using a bus-free filter to continuously monitor \overline{BSY} . If \overline{BSY} remains inactive for at least 400 nsec then the SCSI bus is considered free

and arbitration may begin. Arbitration will begin if the bus is free, \overline{SEL} is inactive and the ARBITRATION bit (port 2, bit 0) is active. Once arbitration has begun (\overline{BSY} asserted), an arbitration delay of 2.2 μ sec must elapse before the data bus can be examined to determine if arbitration has been won. This delay must be implemented in the controlling software driver.

The NCR 5380 is a clockless device. Delays such as bus free delay, bus set delay and bus settle delay are implemented using gate delays. These delays may differ between devices because of inherent process variations, but are well within the proposed ANSI X3T9.2 specification (Revision 17).

SECTION 8 INTERRUPTS

The NCR 5380 provides an interrupt output (IRQ) to indicate a task completion or an abnormal bus occurrence. The use of interrupts is optional and may be disabled by resetting the appropriate bits in the Mode Register (port 2) or the Select Enable Register (port 4).

When an interrupt occurs, the Bus and Status Register and the Current SCSI Bus Status Register must be read to determine which condition created the interrupt. IRQ (pin 23) can be reset simply by reading the Reset Parity/Interrupt Register (port 7) or by an external chip reset ($\overline{\text{RESET}}$ active for 200 nsec).

Assuming the NCR 5380 has been properly initialized, an interrupt will be generated if the chip is selected or reselected, if an EOP signal occurs during a DMA transfer, if an SCSI bus reset occurs, if a parity error occurs during a data transfer, if a bus phase mismatch occurs, or if an SCSI bus disconnection occurs.

8.1 Selection/Reselection

The NCR 5380 can generate a select interrupt if $\overline{\text{SEL}}$ (pin 12) is true (1), its device ID is true (1) and $\overline{\text{BSY}}$ (pin 13) is false for at least a bus settle delay (400 ns). If $\overline{\text{I/O}}$ (pin 17) is active this should be considered a reselect interrupt. The correct ID bit is determined by a match in the Select Enable Register (port 4). Only a single bit match is required to generate an interrupt. This interrupt may be disabled by writing zeros into all bits of the Select Enable Register.

If parity is supported, parity should also be good during the selection phase. Therefore, if the ENABLE PARITY BIT (port 2, bit 5) is active, then the PARITY ERROR bit should be checked to insure that a proper selection has occurred. The ENABLE PARITY INTERRUPT bit need not be set for this interrupt to be generated.

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

The proper values for the Bus and Status Register (port 5) and the Current SCSI Bus Status Register (port 4) are displayed below.

Bus and Status Register

7	6	5	4	3	2	1	0
0	0	0	1	X	0	X	0
END OF DMA	DMA REQUEST	PARITY ERROR	INTER-RUPT RE-QUEST ACTIVE	PHASE MATCH	BUSY ERROR	ATN	ACK

Current SCSI Bus Status Register

7	6	5	4	3	2	1	0
0	0	0	X	X	X	1	X
RST	BSY	REQ	MSG	C/D	I/O	SEL	DBP

8.2 End of Process (EOP) Interrupt

An End of Process signal (\overline{EOP} , pin 27) which occurs during a DMA transfer (DMAMODE true) will set the END OF DMA status bit (port 5, bit 7) and will optionally generate an interrupt if ENABLE EOP INTERRUPT bit (port 2, bit 3) is true. The \overline{EOP} pulse will not be recognized (END OF DMA bit set) unless \overline{EOP} , \overline{DACK} and either \overline{IOR} or \overline{IOW} are concurrently active for at least 100 nsec. DMA transfers can still occur if \overline{EOP} was not asserted at the correct time. This interrupt can be disabled by resetting the ENABLE EOP INTERRUPT bit.

The proper values for the Bus and Status Register (port 5) and the Current SCSI Bus Status Register (port 4) for this interrupt are displayed below.

Bus and Status Register

7	6	5	4	3	2	1	0
1	0	0	1	0	0	0	X
END OF DMA	DMA REQUEST	PARITY ERROR	INTER-RUPT RE-QUEST ACTIVE	PHASE MATCH	BUSY ERROR	ATN	ACK

Current SCSI Bus Status Register

7	6	5	4	3	2	1	0
0	1	X	X	X	X	0	X
RST	BSY	REQ	MSG	C/D	I/O	SEL	DBP

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

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

If using the NCR 53C80, Last Byte Sent (bit 7 of the Target Command Register) may be sampled to determine when the last byte has been transferred.

8.3 SCSI Bus Reset

The NCR 5380 generates an interrupt when the $\overline{\text{RST}}$ signal (pin 16) transitions to true. The device releases all bus signals within a bus clear delay (800 nsec) of this transition. This interrupt also occurs after setting the ASSERT $\overline{\text{RST}}$ bit (port 1, bit 7). This interrupt cannot be disabled. (Note: The $\overline{\text{RST}}$ signal is not latched in bit 7 of the Current SCSI Bus Status Register and may not be active when this port is read. For this case, the Bus Reset interrupt may be determined by default.)

The proper values for the Bus and Status Register (port 5) and the Current SCSI Bus Status Register (port 4) are displayed below.

Bus and Status Register

7	6	5	4	3	2	1	0
0	0	0	1	X	0	0	0
END OF DMA	DMA REQUEST	PARITY ERROR	INTER-RUPT RE-REQUEST ACTIVE	PHASE MATCH	BUSY ERROR	$\overline{\text{ATN}}$	$\overline{\text{ACK}}$

Current SCSI Bus Status Register

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

8.4 Parity Error

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

The proper values for the Bus and Status Register (port 5) and the Current SCSI Bus Status Register (port 4) are displayed below.

Bus and Status Register

7	6	5	4	3	2	1	0
0	X	1	1	1	0	X	X
END OF DMA	DMA REQUEST	PARITY ERROR	INTER-RUPT RE-REQUEST ACTIVE	PHASE MATCH	BUSY ERROR	$\overline{\text{ATN}}$	$\overline{\text{ACK}}$

Current SCSI Bus Status Register

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

8.5 Bus Phase Mismatch

The SCSI phase lines are comprised of the signals $\overline{I/O}$, $\overline{C/D}$ and \overline{MSG} . These signals are compared with the corresponding bits in the Target Command Register: ASSERT $\overline{I/O}$ (bit 0), ASSERT $\overline{C/D}$ (bit 1) and ASSERT \overline{MSG} (bit 2). The comparison occurs continually and is reflected in the PHASE MATCH bit (bit 3) of the Bus and Status Register (port 5). If the DMA MODE bit (port 2, bit 1) is active and a phase mismatch occurs when \overline{REQ} (pin 20) transitions from false to true, an interrupt (IRQ) is generated.

A phase mismatch prevents the recognition of \overline{REQ} and removes the chip from the bus during an initiator send operation. ($\overline{DB0-DB7}$, \overline{DBP} will not be driven even though the ASSERT DATA BUS bit (port 1, bit 0) is active.) This interrupt is only significant when connected as an Initiator and may be disabled by resetting the DMA MODE bit. (Note: It is possible for this interrupt to occur when connected as a Target if another device is driving the phase lines to a different state.)

The proper values for the Bus and Status Register (port 5) and the Current SCSI Bus Status Register (port 4) are displayed below.

Bus and Status Register

7	6	5	4	3	2	1	0
0	0	0	1	0	0	X	0
END OF DMA	DMA REQUEST	PARITY ERROR	INTER-RUPT RE-QUEST ACTIVE	PHASE MATCH	BUSY ERROR	\overline{ATN}	\overline{ACK}

Current SCSI Bus Status Register

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

8.6 Loss of \overline{BSY}

If the MONITOR BUSY bit (bit 2) in the Mode Register (port 2) is active, an interrupt will be generated if the \overline{BSY} signal (pin 13) goes false for at least a bus settle delay (400 nsec). This interrupt may be disabled by resetting the MONITOR BUSY bit. Register values are as follows.

Bus and Status Register

7	6	5	4	3	2	1	0
0	0	0	1	X	1	0	0
END OF DMA	DMA REQUEST	PARITY ERROR	INTER-RUPT RE-QUEST ACTIVE	PHASE MATCH	BUSY ERROR	\overline{ATN}	\overline{ACK}

Current SCSI Bus Status Register

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

SECTION 9

RESET CONDITIONS

Three possible reset situations exist with the NCR 5380, as follows:

9.1 Hardware Chip Reset

When the signal RESET/ (pin 28) is active for at least 200 nsec, the NCR 5380 device is re-initialized and all internal logic and control registers are cleared. This is a chip reset only and does not create an SCSI bus reset condition.

9.2 SCSI Bus Reset (\overline{RST}) Received

When an SCSI \overline{RST} signal (pin 16) is received, an IRQ interrupt is generated and a chip reset is performed. All internal logic and registers are cleared, except for the IRQ interrupt latch and the ASSERT \overline{RST} bit (bit 7) in

the Initiator Command Register (port 1). (Note: The \overline{RST} signal may be sampled by reading the Current SCSI Bus Status Register (port 4); however, this signal is not latched and may not be present when this port is read.)

9.3 SCSI Bus Reset (\overline{RST}) Issued

If the CPU sets the ASSERT \overline{RST} bit (bit 7) in the Initiator Command Register (port 1), the \overline{RST} signal (pin 16) goes active on the SCSI bus and an internal reset is performed. Again, all internal logic and registers are cleared except for the IRQ interrupt latch and the ASSERT \overline{RST} bit (bit 7) in the Initiator Command Register (port 1). The \overline{RST} signal will continue to be active until the ASSERT \overline{RST} bit is reset or until a hardware reset occurs.

SECTION 10

DATA TRANSFERS

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

10.1 Programmed I/O Transfers

Programmed I/O is the most primitive form of data transfer. The \overline{REQ} (pin 20) and \overline{ACK} (pin 14) handshake signals are individually monitored and asserted by reading and writing the appropriate register bits. This type of transfer is normally used when transferring small blocks of data such as command blocks or message and status bytes.

An Initiator send operation would begin by setting the C/D , I/O , and MSG bits in the Target Command Register to the correct state so that a phase match exists. In addition to the phase match condition, it is necessary for the $ASSERT\ DATA\ BUS$ bit (port 1, bit 0) to be true and the received I/O signal to be false for the 5380 to send data.

For each transfer, the data is loaded into the Output Data Register (port 0). The MPU then waits for the REQ bit (port 4, bit 5) to become active. Once REQ goes active the $PHASE\ MATCH$ bit (port 5, bit 3) is checked and the $ASSERT\ ACK$ bit (port 1, bit 4) is set. The REQ bit is sampled until it becomes false and the MPU resets the $ASSERT\ ACK$ bit to complete the transfer.

10.2 Normal DMA Mode

DMA transfers are normally used for large block transfers. The SCSI chip outputs a DMA request (DRQ - pin 22) whenever it is ready for a byte transfer. External DMA logic uses this DRQ signal to generate \overline{DACK} and an \overline{IOR} or an \overline{IOW} pulse to the NCR 5380. DRQ goes inactive when \overline{DACK} is asserted and \overline{DACK} goes inactive sometime after the minimum read or write pulse width. This process is repeated for every byte. For this mode, \overline{DACK} should not be allowed to cycle unless a transfer is taking place.

Refer to Section 10.5 for information on halting a DMA transfer.

10.3 Block Mode DMA

Block Mode allows an external DMA device (Intel 8237-type DMA) to perform sequential DMA transfer without relinquishing the data bus to the CPU. Holding \overline{DACK} active prevents Intel-type CPUs from gaining access to the system bus. The handshake itself does not increase the transfer rate. Preventing the CPU from sharing the system bus increases the DMA transfer rate but also halts the CPU operation.

Block Mode DMA transfers are supported for both Initiator and Target Role operation. When using this mode of operation, DRQ is asserted to signal the beginning of the DMA transfer. In response to DRQ , \overline{DACK} is asserted and remains asserted throughout the transfer. $READY$ goes active after the IOW or IOR pulse goes inactive, effectively replacing the DRQ signal.

Care must be taken when using this mode due to the operation of $READY$. If, for example, a phase mismatch interrupt occurs, $READY$ will remain in the inactive state and INT will be active. For this condition the DMA chip must return control of the bus to the CPU so that the 5380 interrupt can be serviced. $READY$ also does not return to the active state when an EOP pulse is received. Therefore, you might want to use EOP to insure that the CPU regains bus control after the last DMA byte has been transferred. As in the non-block DMA mode, the EOP signal does not deassert ACK on the SCSI bus. To successfully complete a DMA send operation, either an additional byte of data must be written to the NCR 5380 to allow ACK to go inactive, or the CPU must reset the DMA Mode bit in the Mode Register.

Non-block mode DMA transfers end when \overline{DACK} goes false, whereas block mode transfers end when \overline{IOR} or \overline{IOW} becomes inactive after each byte. Since this is the case, DMA transfers may be started sooner in a block mode transfer.

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

The methods described in Section 10.5 “Halting A DMA Operation” apply for all DMA operations.

10.4 Pseudo DMA Mode

To avoid the tedium of monitoring and asserting the request/acknowledge handshake signals for programmed I/O transfers, the system may be designed to implement a pseudo DMA mode. This mode is implemented by programming the NCR 5380 to operate in the DMA mode, but using the MPU to emulate the DMA handshake. DRQ (pin 22) may be detected by polling the DMA REQ bit (bit 6) in the Bus and Status Register (port 5), by sampling the signal through an external port or by using it to generate an MPU interrupt. Once DRQ is detected, the MPU can perform a DMA port read or write data transfer. This MPU read/write is externally decoded to generate the appropriate \overline{DACK} and \overline{IOR} or \overline{IOW} signals.

Often, external decoding logic is necessary to generate the NCR 5380 \overline{CS} signal. This same logic may be used to generate \overline{DACK} at no extra system cost and provide an increased performance in programmed IO transfers.

10.5 Halting A DMA Operation

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

10.5.1 Using the \overline{EOP} Signal

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

10.5.2 Bus Phase Mismatch Interrupt

A bus phase mismatch interrupt may be used to halt the transfer if operating as an Initiator. Using this method frees the host from maintaining a data length

counter and frees the DMA logic from providing the \overline{EOP} signal. If performing an initiator send operation, the NCR 5380 requires \overline{DACK} to cycle before \overline{ACK} goes inactive. Since phase changes cannot occur if \overline{ACK} is active, either \overline{DACK} must be cycled after the last byte is sent or the DMA MODE bit must be reset in order to receive the phase mismatch interrupt.

10.5.3 Resetting the DMA MODE Bit

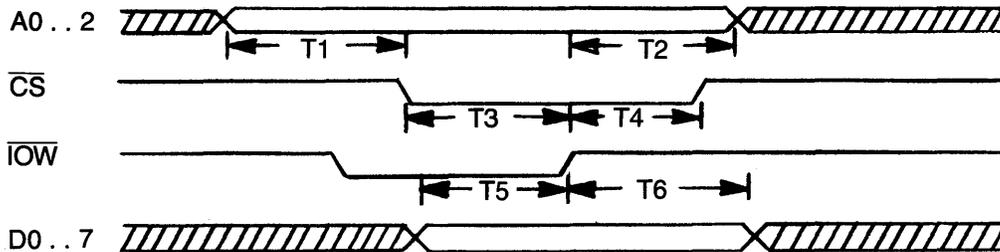
A DMA operation may be halted at any time simply by resetting the DMA MODE bit. It is recommended that the DMA MODE bit be reset after receiving an \overline{EOP} or bus phase mismatch interrupt. The DMA MODE bit must then be set before writing any of the start DMA registers for subsequent bus phases.

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

SECTION 11

EXTERNAL TIMING DIAGRAMS

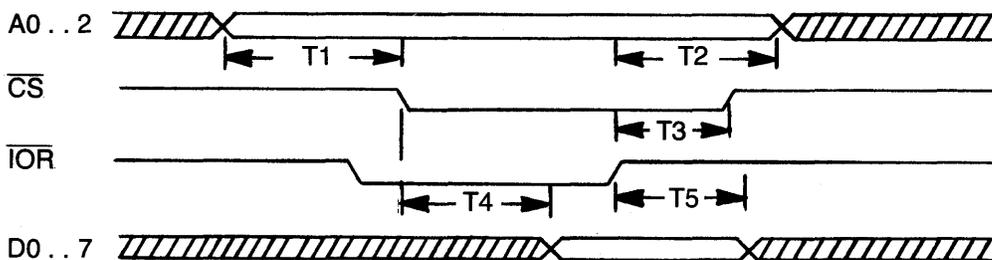
11.1 CPU WRITE



NAME	DESCRIPTION	MIN.	TYP.	MAX.	UNITS
T1	Address setup to write enable *	20			ns.
T2	Address hold from end write enable *	20			ns.
T3	Write enable width *	70			ns.
T4	Chip select hold from end of IOW	0			ns.
T5	Data setup to end of write enable *	50			ns.
T6	Data hold time from end of IOW	30			ns.

* Write enable is the occurrence of IOW and CS

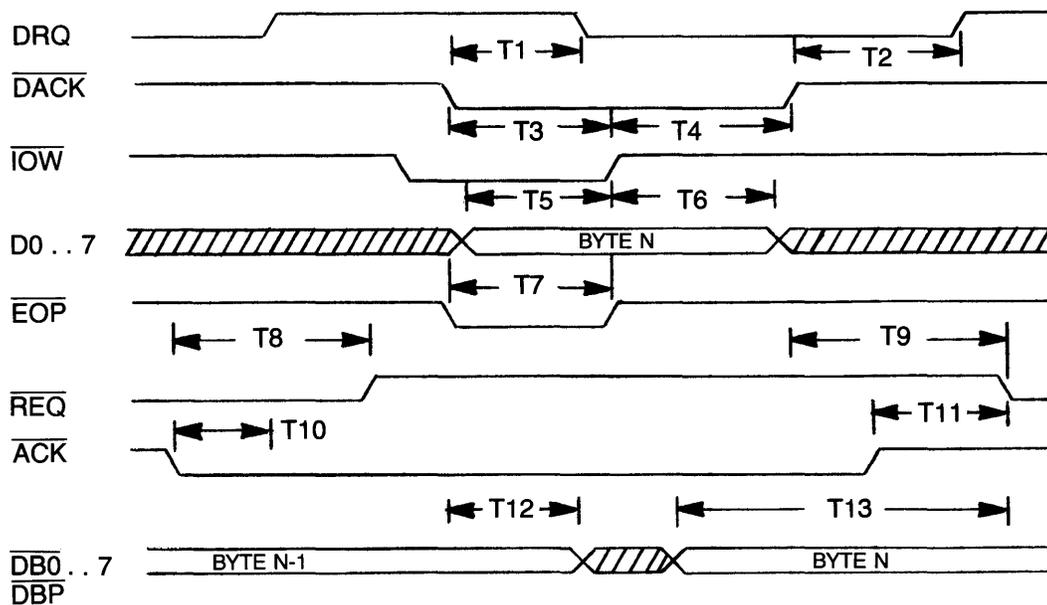
11.2 CPU READ



NAME	DESCRIPTION	MIN.	TYP.	MAX.	UNITS
T1	Address setup to read enable *	20			ns.
T2	Address hold from end read enable *	20			ns.
T3	Chip select hold from end of IOR	0			ns.
T4	Data access time from read enable *			130	ns.
T5	Data hold time from end of IOR	20			ns.

* Read enable is the occurrence of IOR and CS

11.3 DMA WRITE (NON-BLOCK MODE) TARGET SEND

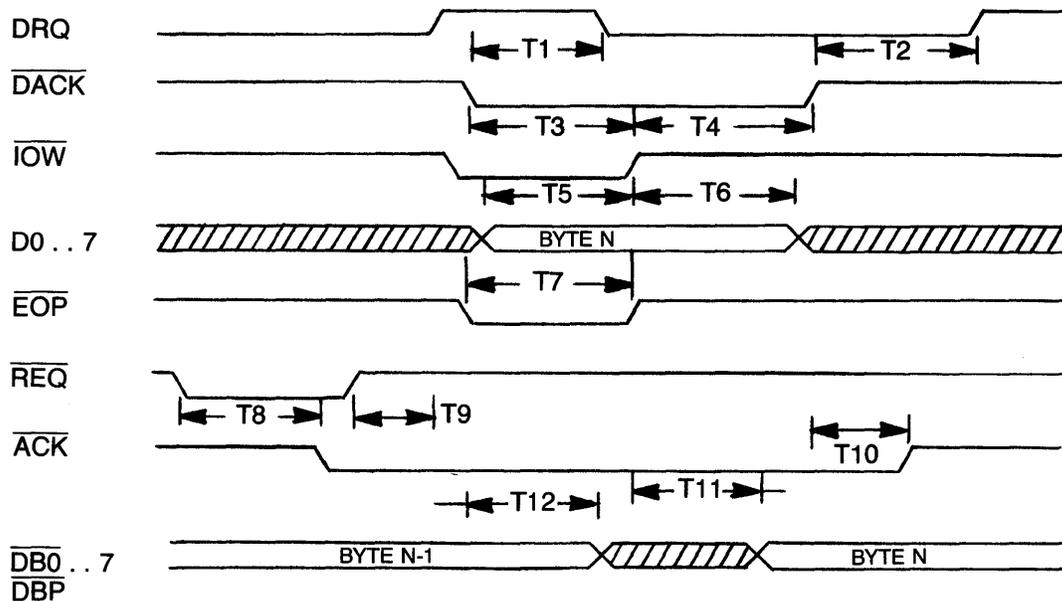


NAME	DESCRIPTION	MIN.	TYP.	MAX.	UNITS
T1	DRQ false from \overline{DACK} true			130	ns.
T2	\overline{DACK} false to DRQ true	30			ns.
T3	Write enable width *	100			ns.
T4	DACK hold from end of \overline{IOW}	0			ns.
T5	Data setup to end of write enable *	50			ns.
T6	Data hold time from end of \overline{IOW}	40			ns.
T7	Width of \overline{EOP} pulse (note 1)	100			ns.
T8	\overline{ACK} true to \overline{REQ} false	25	110	125	ns.
T9	\overline{REQ} from end of \overline{DACK} (\overline{ACK} false)	30	140	150	ns.
T10	\overline{ACK} true to DRQ true (target)	15	100	110	ns.
T11	\overline{REQ} from end of \overline{ACK} (\overline{DACK} false)	20	140	150	ns.
T12	DATA hold from write enable	15			ns.
T13	Data setup to \overline{REQ} true (target)	60			ns.

* Write enable is the occurrence of \overline{IOW} and \overline{DACK}

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

11.4 DMA WRITE (NON-BLOCK MODE) INITIATOR SEND

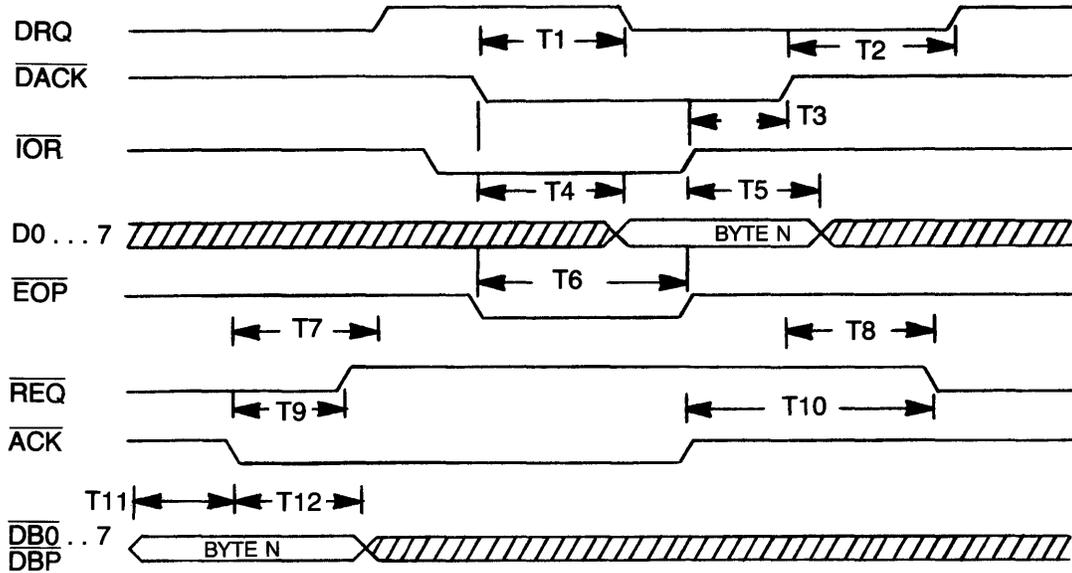


NAME	DESCRIPTION	MIN.	TYP.	MAX.	UNITS
T1	DRQ false from $\overline{\text{DACK}}$ true			130	ns.
T2	$\overline{\text{DACK}}$ false to DRQ true	30			ns.
T3	Write enable width *	100			ns.
T4	$\overline{\text{DACK}}$ hold from end of $\overline{\text{IOW}}$	0			ns.
T5	Data setup to end of write enable *	50			ns.
T6	Data hold time from end of $\overline{\text{IOW}}$	40			ns.
T7	Width of $\overline{\text{EOP}}$ pulse (note 1)	100			ns.
T8	$\overline{\text{REQ}}$ true to $\overline{\text{ACK}}$ true	20	150	160	ns.
T9	$\overline{\text{REQ}}$ false to DRQ true	20	100	110	ns.
T10	$\overline{\text{DACK}}$ false to $\overline{\text{ACK}}$ false	25	140	150	ns.
T11	$\overline{\text{IOW}}$ false to valid SCSI data			100	ns.
T12	DATA hold from write enable	15			ns.

* Write enable is the occurrence of $\overline{\text{IOW}}$ and $\overline{\text{DACK}}$

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

11.5 DMA READ (NON-BLOCK MODE) TARGET RECEIVE

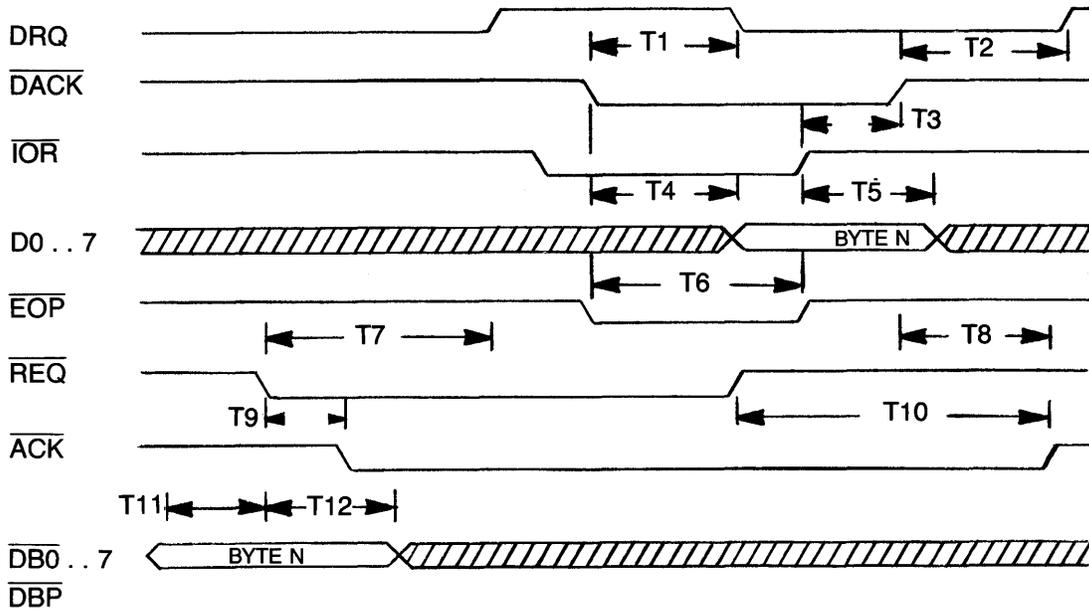


NAME	DESCRIPTION	MIN.	TYP.	MAX.	UNITS
T1	DRQ false from $\overline{\text{DACK}}$ true			130	ns.
T2	$\overline{\text{DACK}}$ false to DRQ true	30			ns.
T3	$\overline{\text{DACK}}$ hold time from end of $\overline{\text{IOR}}$	0			ns.
T4	Data access time from read enable *			115	ns.
T5	Data hold time from end of $\overline{\text{IOR}}$	20			ns.
T6	Width of $\overline{\text{EOP}}$ pulse (note 1)	100			ns.
T7	$\overline{\text{ACK}}$ true to DRQ true	15	100	110	ns.
T8	$\overline{\text{DACK}}$ false to $\overline{\text{REQ}}$ true ($\overline{\text{ACK}}$ false)	30		150	ns.
T9	$\overline{\text{ACK}}$ true to $\overline{\text{REQ}}$ false	25	110	125	ns.
T10	$\overline{\text{ACK}}$ false to $\overline{\text{REQ}}$ true ($\overline{\text{DACK}}$ false)	20	140	150	ns.
T11	DATA setup time to $\overline{\text{ACK}}$	20			ns.
T12	DATA hold time from $\overline{\text{ACK}}$	50			ns.

* Read enable is the occurrence of $\overline{\text{IOR}}$ and $\overline{\text{DACK}}$

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

11.6 DMA READ (NON-BLOCK MODE) INITIATOR RECEIVE

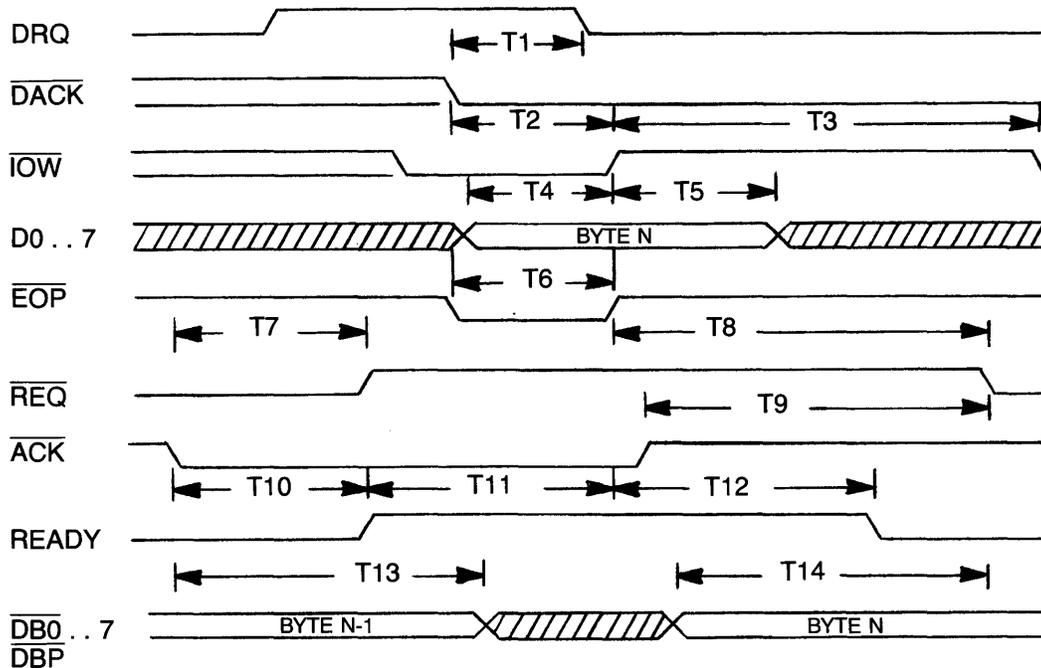


NAME	DESCRIPTION	MIN.	TYP.	MAX.	UNITS
T1	DRQ false from \overline{DACK} true			130	ns.
T2	\overline{DACK} false to DRQ true	30			ns.
T3	\overline{DACK} hold time from end of \overline{IOR}	0			ns.
T4	Data access time from read enable *			115	ns.
T5	Data hold time from end of \overline{IOR}	20			ns.
T6	Width of \overline{EOP} pulse (note 1)	100			ns.
T7	\overline{REQ} true to DRQ true	20	140	150	ns.
T8	\overline{DACK} false to \overline{ACK} false (\overline{REQ} false)	25	140	160	ns.
T9	\overline{REQ} true to \overline{ACK} true	20	150	160	ns.
T10	\overline{REQ} false to \overline{ACK} false (\overline{DACK} false)	15	120	140	ns.
T11	DATA setup time to \overline{REQ}	20			ns.
T12	DATA hold time from \overline{REQ}	50			ns.

*Read enable is the occurrence of \overline{IOR} and \overline{DACK}

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

11.7 DMA WRITE (BLOCK MODE) TARGET SEND

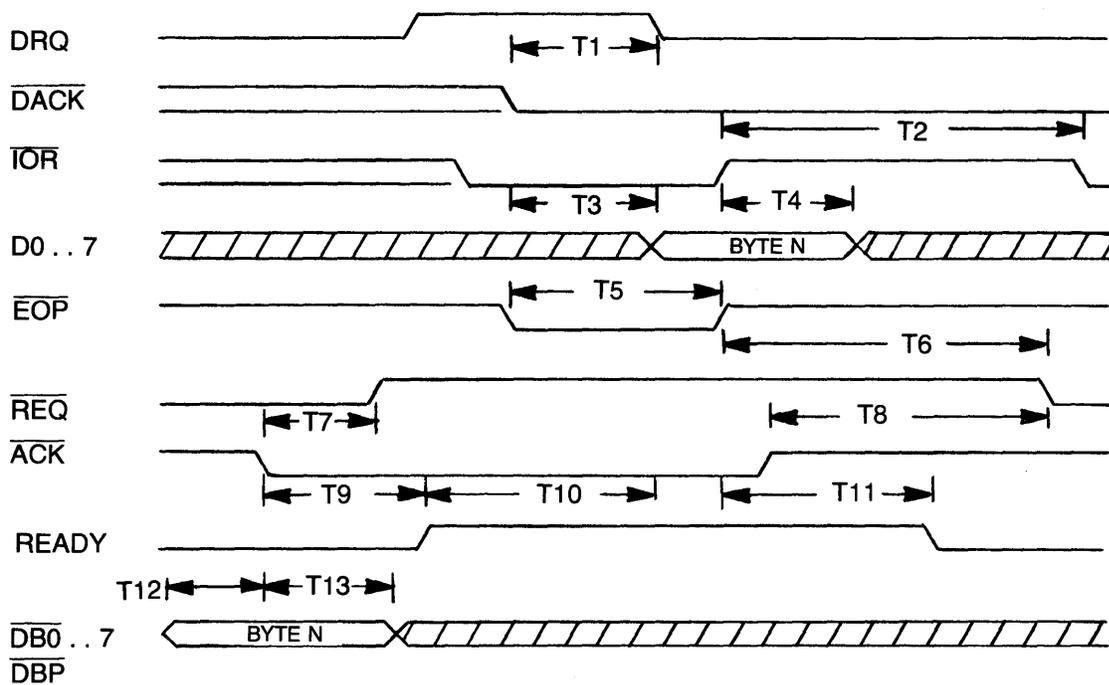


NAME	DESCRIPTION	MIN.	TYP.	MAX.	UNITS
T1	DRQ false from \overline{DACK} true			130	ns.
T2	Write enable width *	100			ns.
T3	Write recovery time	120			ns.
T4	Data setup to end of write enable *	50			ns.
T5	Data hold time from end of \overline{IOW}	40			ns.
T6	Width of \overline{EOP} pulse (note 1)	100			ns.
T7	\overline{ACK} true to \overline{REQ} false	25	110	125	ns.
T8	\overline{REQ} from end of \overline{IOW} (\overline{ACK} false)	40		180	ns.
T9	\overline{REQ} from end of \overline{ACK} (\overline{IOW} false)	20	160	170	ns.
T10	\overline{ACK} true to \overline{READY} true	20	130	140	ns.
T11	\overline{READY} true to \overline{IOW} false	70			ns.
T12	\overline{IOW} false to \overline{READY} false	20	130	140	ns.
T13	DATA hold from \overline{ACK} true	40			ns.
T14	Data setup to \overline{REQ} true	60			ns.

* Write enable is the occurrence of \overline{IOW} and \overline{DACK}

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

11.8 DMA READ (BLOCK MODE) TARGET RECEIVE

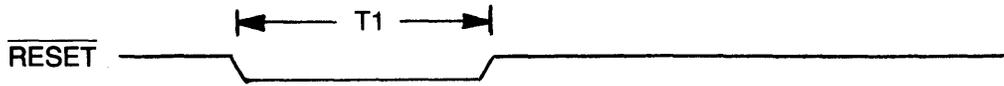


NAME	DESCRIPTION	MIN.	TYP.	MAX.	UNITS
T1	DRQ false from \overline{DACK} true			130	ns.
T2	\overline{IOR} recovery time	120			ns.
T3	Data access time from read enable *		100	110	ns.
T4	Data hold time from end of \overline{IOR}	20			ns.
T5	Width of \overline{EOP} pulse (note 1)	100			ns.
T6	\overline{IOR} false to \overline{REQ} true (\overline{ACK} false)	30	180	190	ns.
T7	\overline{ACK} true to \overline{REQ} false	25	110	125	ns.
T8	\overline{ACK} false to \overline{REQ} true (\overline{IOR} false)	20	160	170	ns.
T9	\overline{ACK} true to \overline{READY} true	20	130	140	ns.
T10	\overline{READY} true to valid data			50	ns.
T11	\overline{IOR} false to \overline{READY} false	20	125	140	ns.
T12	DATA setup time to \overline{ACK}	20			ns.
T13	DATA hold time from \overline{ACK}	50			ns.

* Read enable is the occurrence of \overline{IOR} and \overline{DACK}

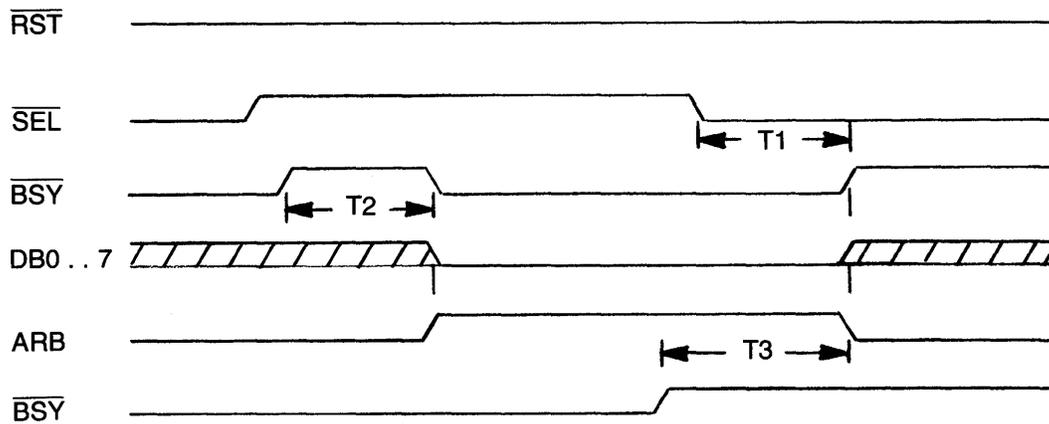
Note 1: \overline{EOP} , \overline{IOR} , and \overline{DACK} must be concurrently true for at least T5 for proper recognition of the \overline{EOP} pulse.

11.9 RESET



NAME	DESCRIPTION	MIN.	TYP.	MAX.	UNITS
T1	Minimum width of reset	200			ns.

11.10 ARBITRATION



NAME	DESCRIPTION	MIN.	TYP.	MAX.	UNITS
T1	Bus clear from SEL true			600	ns.
T2	ARBITRATE start from BSY false	1200		2200	ns.
T3	Bus clear from BSY false			1100	ns.

APPENDICES

A1. NCR 5380 vs. NCR 5385/86

The NCR 5380 was designed to provide a low-cost SCSI interface using a minimum number of parts. Much of the intelligence and some of the features included in the NCR 5385E/86 have been removed. In some instances, such as arbitration, this causes the controlling CPU to provide more of the protocol control. The NCR 5385/86 remains appropriate for many applications and will continue to be strongly supported.

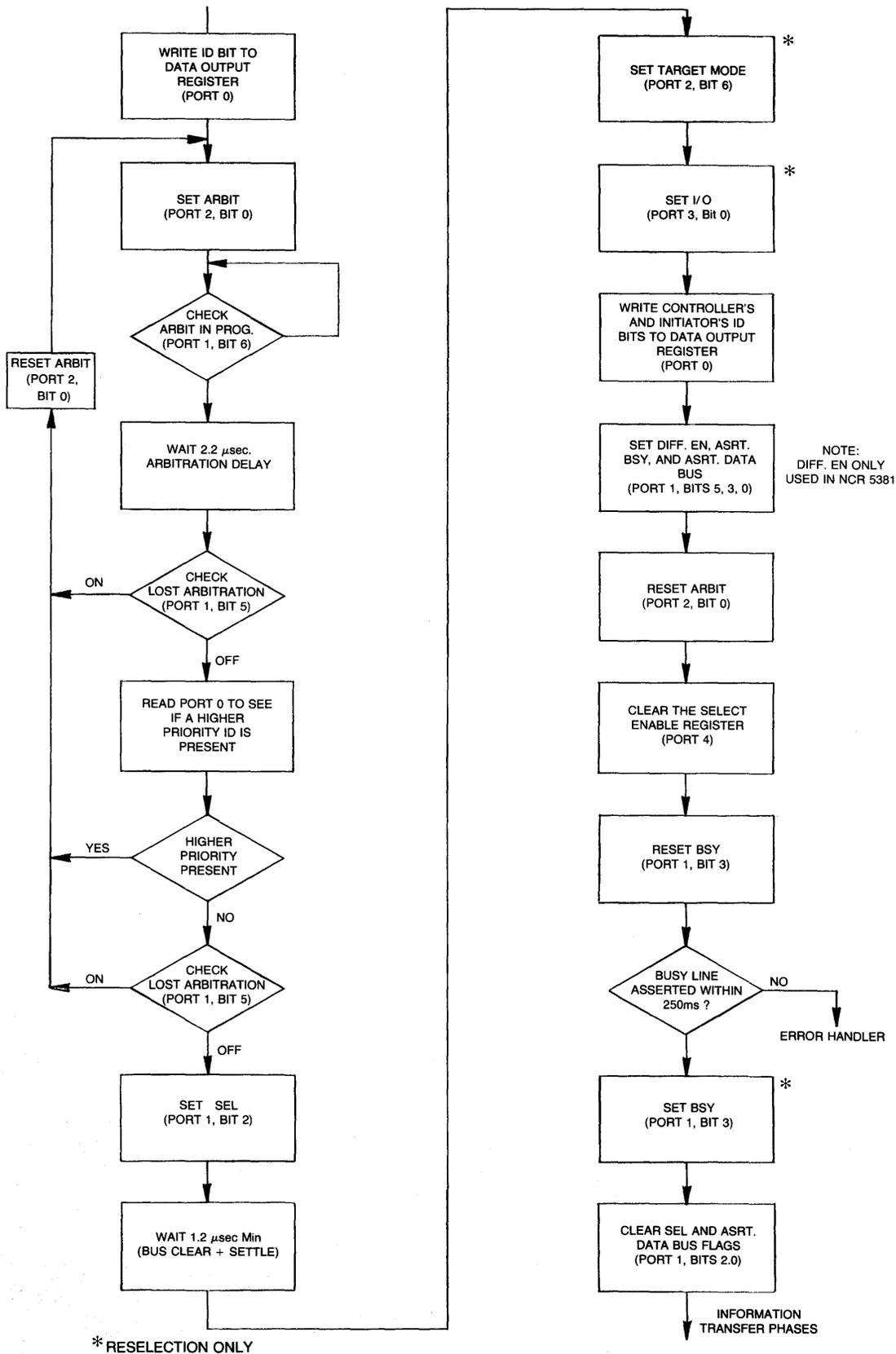
The main differences between the NCR 5380 and the NCR 5385E/86 are shown in the following table.

Functional Areas	5380	5385E/86
Arbitration	Optional, Firmware Dependant	Automatically Invoked
Maximum Transfer Rate	1.5 MBPS	2.0 MBPS
Transfer Counter	None	24 bits
Data Buffering	Single	Double
Clock Circuitry	None Req'd	5-10 MHz
Single-Ended Transceivers	On-chip	External
Differential Pair	External (NCR 5381)	External
Synchronous Mode	No Firm Plans	NCR 5386S

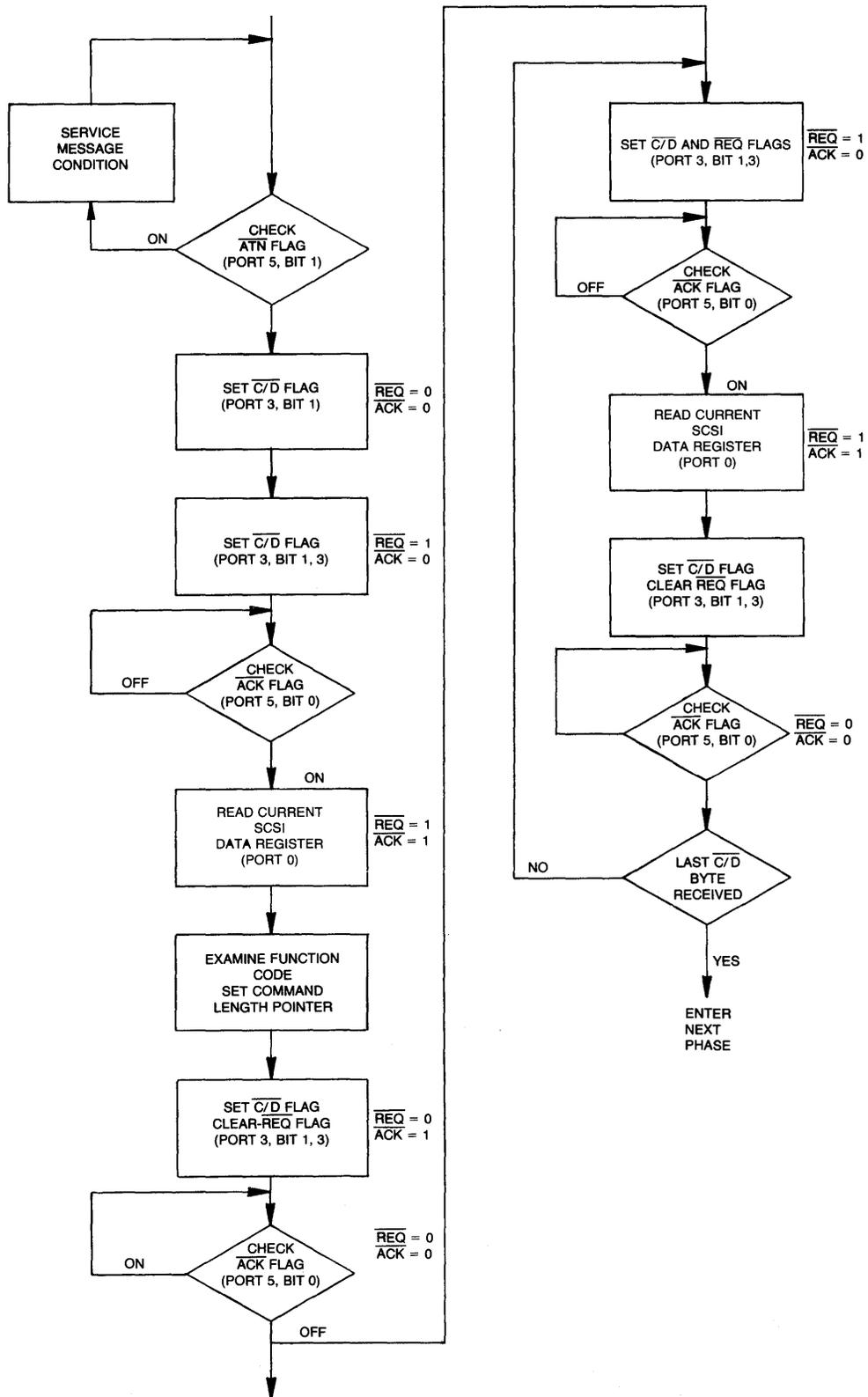
A2. FLOWCHARTS/SOFTWARE

Flowcharts and sample software drivers are provided as a guideline to facilitate your firmware development. Firmware will vary depending on the application and the level of the SCSI protocol being supported. In accordance with register definitions, a one (1) designates signal assertion and a zero (0) designates signal non-assertion, or the inactive state.

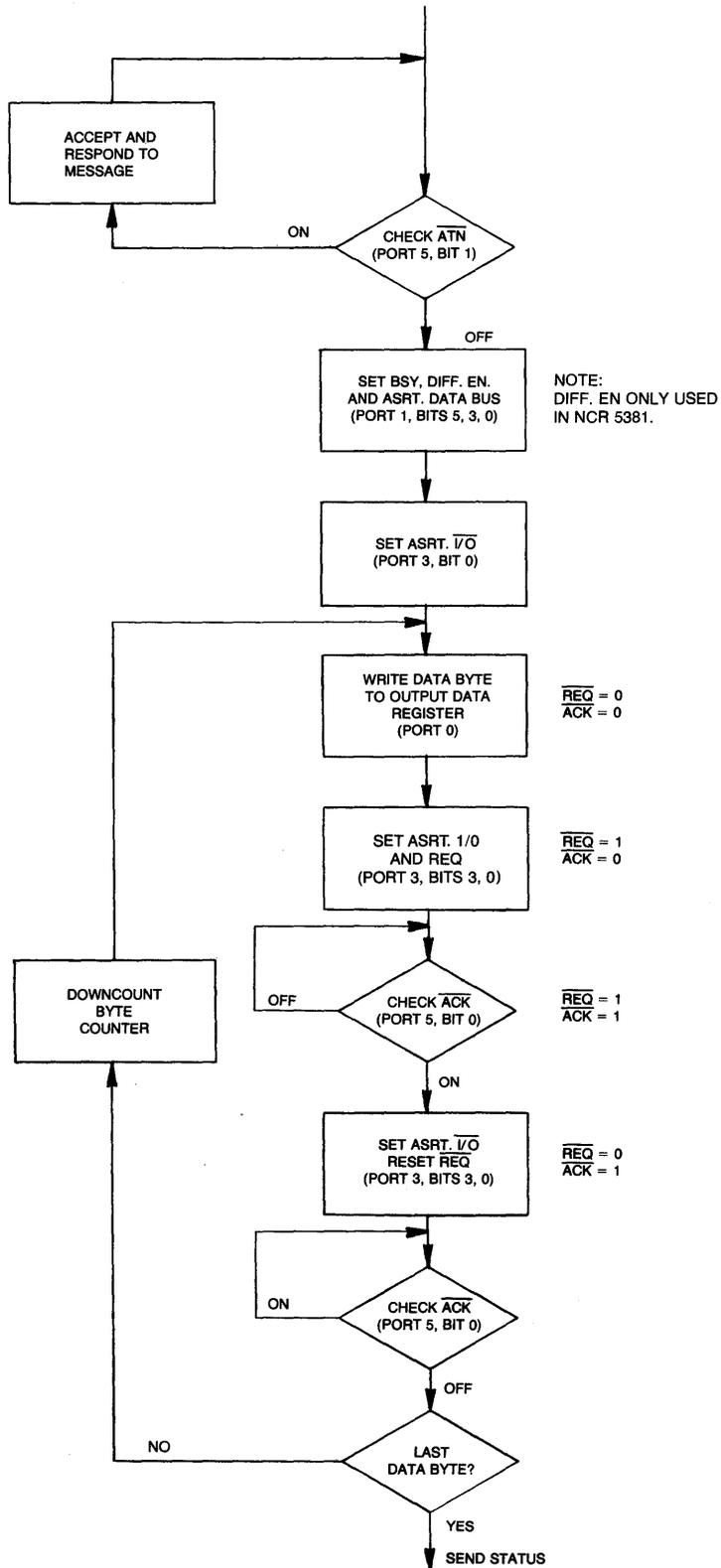
ARBITRATION AND (RE) SELECTION



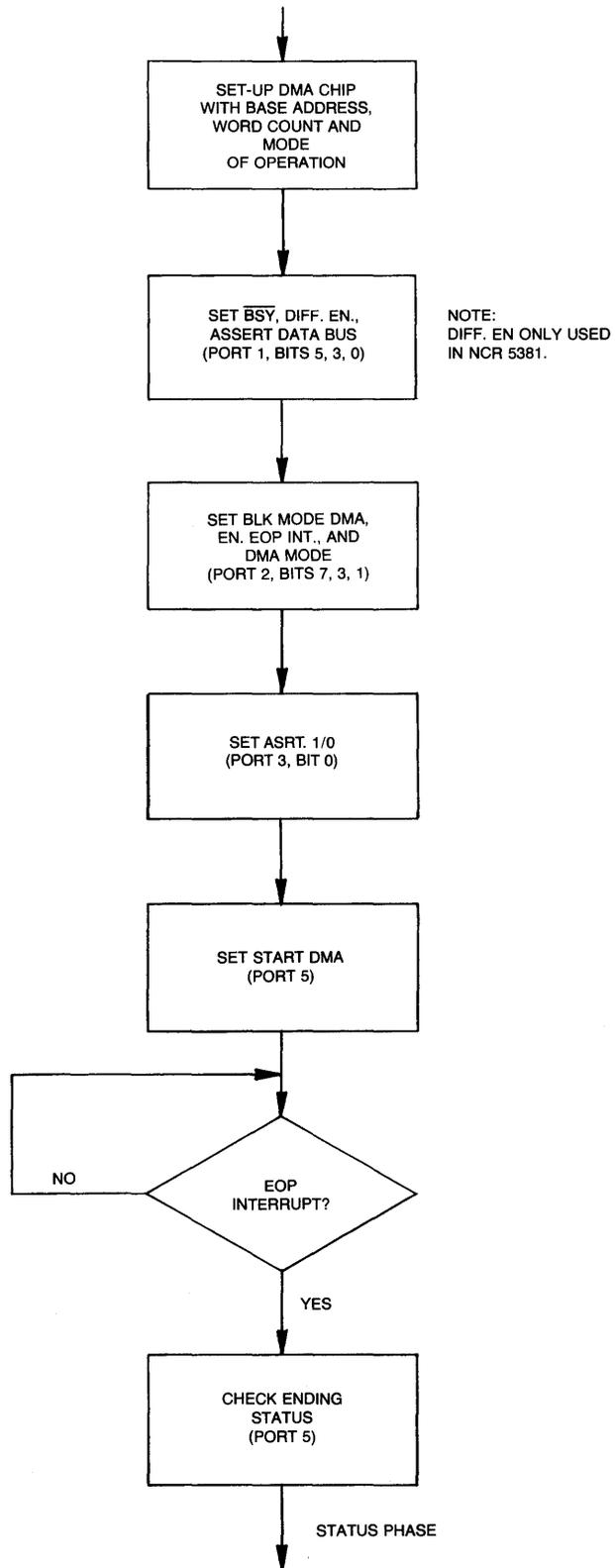
COMMAND TRANSFER PHASE (TARGET)



DATA TRANSFER TO HOST VIA PROGRAMMED I/O



DATA TRANSFER VIA DMA



LINE#	LOC	CODE	LINE
00001	0000		; NCR 5380 SCSI PROTOCOL DRIVER
00002	0000		;
00003	0000		; SUPPORTS BOTH INITIATOR AND TARGET ROLES
00004	0000		;
00005	0000		; ASSUMES THAT THE COMMAND BLOCK (CDB),
00006	0000		; DATA BLOCK(DBLK), AND THE EXPECTED PHASE
00007	0000		; TABLE HAVE BEEN SPECIFIED IN MEMORY
00008	0000		;
00009	0000		SLFAIL=\$01 ; SELECTION FAILED STATUS
00010	0000		DISCNT=\$02 ; DISCONNECTED STATUS
00011	0000		PRTYER=\$03 ; PARITY ERROR STATUS
00012	0000		BUSRST=\$04 ; SCSI BUS RESET STATUS
00013	0000		CHIPFL=\$05 ; CHIP FAILURE STATUS
00014	0000		MESSAG=\$06 ; MESSAGE IN BYTE BEING RETND
00015	0000		DIFFPH=\$07 ; UNEXPECTED PHASE REQUESTED
00016	0000		;
00017	0000		CMDCPL=\$00 ; COMMAND COMPLETE MESSAGE
00018	0000		;
00019	0000		DATA0=\$00 ; DATA OUT PHASE
00020	0000		CMD=\$08 ; COMMAND PHASE
00021	0000		STATUS=\$0C ; STATUS PHASE
00022	0000		DATAI=\$04 ; DATA IN PHASE
00023	0000		MESSO=\$18 ; MESSAGE OUT PHASE
00024	0000		MESSI=\$1C ; MESSAGE IN PHASE
00025	0000		DISCON=\$80 ; FLAG TO DISCONNECT
00026	0000		SELECT=\$40 ; FLAG TOWAIT FOR SELECTION
00027	0000		;
00028	0000		55380=\$DE00 ; 5380 ADDRESS SPACE
00029	0000		SDMA=\$DE0C ; PSEUDO DMA ADDRESS
00030	0000		IID=\$DE08 ; INIT. ID EXT. LATCH
00031	0000		SRST=\$DF00 ; NCR5380 DEVICE RESET
00032	0000		BPNTR=\$FB ; DATA BLOCK POINTER
00033	0000		;
00034	0000		;\$*=BPNTR ;
00035	0000		;
00036	0000		;\$DATAB .WORD DBLK ;
00037	0000		;
00038	0000		;
00039	0000		;\$*=\$C000 ; PROGRAM SPACE ORIGIN
00040	C000		TID ****+1 ; TARGET ID SPACE
00041	C001		ICRVAL ****+1 ; INIT. CMD REG. STORAGE
00042	C002		INITFL ****+1 ; INITIATOR FLAG
00043	C003		OCFLAG ****+1 ; OPEN COLL. FLAG
00044	C004		PTYFLG ****+1 ; PARITY FLAG
00045	C005		ATNFLG ****+1 ; ATN FLAG
00046	C006		;
00047	C006		PHSIDX ****+1 ; EXPECTED PHASE INDEX
00048	C007		XPTPHS ****+30 ; EXPECTED PHASE TABLE
00049	C025		;
00050	C025		COUNT ****+1 ; BYTE COUNT
00051	C026		XCNT ****+1 ; BYTE COUNT MULTIPLIER
00052	C027		;
00053	C027		CDB ****+12 ; CMD BLOCK STORAGE
00054	C033		DBLK ****+512 ; DATA BLOCK
00055	C233		STAT ****+2 ; STATUS BYTES

LINE#	LOC	CODE	LINE		
00056	C235	27 C0	CDBS	.WORD	CDB, DBLK, STAT
00056	C237	33 C0			
00056	C239	33 C2			
00057	C23B				
00058	C23B				; INITIALIZATION
00059	C23B				
00060	C23B	A9 00	START	LDA #00	; ZERO ACCUM
00061	C23D	8D 06 C0		STA PHSIDX	; INITIALIZE PHASE INDEX
00062	C240	AD 00 DF		LDA SRST	; RESET 5380NUMBER
00063	C243	A9 04		LDA #%00000100	; ENABLE MONITOR BSY INT.
00064	C245	AE 04 C0		LDX PTYFLG	; LOAD PARITY FLAG
00065	C248	F0 02		BEG NOPTY	; IF ZERO, NO PARITY
00066	C24A	09 30		ORA #%00110000	; OR IN CHECK PARITY BITS
00067	C24C	8D 02 DE	NOPTY	STA S5380+2	; STORE IN MODE REGISTER
00068	C24F				
00069	C24F	AE 06 C0		LDX PHSIDX	; LOAD VALUE OF PHASE INDEX
00070	C252	A9 40		LDA #SELECT	; GET VALUE OF SEL CMD
00071	C254	DD 07 C0		CMP XTPHS, X	; COMPARE W/CURRENT PHASE
00072	C257	D0 03		BNE INIT	; IF NOT = BEGIN ARBITRATION
00073	C259	4C 8D C3		JMP TARSEL	; ELSE, WAIT FOR TARGET SELECT
00074	C25C				
00075	C25C				; BEGIN SCSI BUS ARBITRATION
00076	C25C				
00077	C25C	AD 02 DE	INIT	LDA S5380+2	; READ MODE REG.
00078	C25F	29 FE		AND #%11111110	; MASK ARB BIT
00079	C261	8D 02 DE		STA S5380+2	; RESET ARBITRATION BIT
00080	C264	AD 08 DE	ARB	LDA IID	; BEGIN ARBITRATION
00081	C267	8D 00 DE		STA S5380+0	; LOAD ID INTO ODR
00082	C26A	AD 02 DE		LDA S5380+2	; READ MODE REG.
00083	C26D	09 01		ORA #%00000001	; SET ARBITRATION BIT
00084	C26F	8D 82 53		STA S5380+2	; STORE IN MODE REG.
00085	C272				
00086	C272				; HAS BUS GONE FREE?
00087	C272				
00088	C272	2C 01 DE	NFREE	BIT S5380+1	; BUS FREE?
00089	C275	50 FB		BVC NFREE	; NO LOOP UNTIL FREE
00090	C277				
00091	C277	EA		NOP	; YES, WAIT AN ARB DELAY (2.2USE)
00092	C278	AD 01 DE		LDA S5380+1	; LOAD INIT CMD REG.
00093	C27B	29 20		AND #%00100000	; MASK ALL BUT LA BIT
00094	C27D	D0 DD		BNE INIT	; IF LOST ARB, RESTART
00095	C27F				
00096	C27F				; CHECK FOR HIGHER PRIORITY ID?
00097	C27F				
00098	C27F	AD 00 DE		LDA S5380+0	; LOAD CURRENT DATA REG.
00099	C282	38		SEC	; SET CARRY BIT
00100	C283	ED 08 DE		SBC IID	; SUB YOUR ID FROM DATA REG.
00101	C286	F0 08		BEG WIN	; IF EQUAL TO ZERO, WIN ARB
00102	C288	38		SEC	; NOT=, SOMEONE ELSE IS ARB-ING
00103	C289	ED 08 DE		SBC IID	; SUBTRACT YOUR ID AGAIN
00104	C28C	30 02		BMI WIN	; IF NEG, YOUR ID WAS HIGHER
00105	C28E	D0 CC		BNE INIT	; OTHERWISE, RESTART
00106	C290				
00107	C290				; RECHECK LOST ARBITRATION
00108	C290				

LINE#	LOC	CODE	LINE		
00109	C290	AD 01 DE	WIN	LDA S5380+1	;LOAD INIT. CMD REG.
00110	C293	29 20		AND %00100000	;MASK ALL BUT LA BIT
00111	C295	D0 C5		BNE INIT	;IF LOST ARB, RESTART
00112	C297				
00113	C297	A9 0C		LDA %00001100	;LOAD VALUE TO SET SEL SIGNAL
00114	C299	AE 05 C0		LDX ATNFLG	;LOAD ATN FLAG
00115	C29C	F0 02		BEQ WOATN	; IF ZERO, SEL W/D ATN
00116	C29E	09 02		ORA %00000010	; OR IN ATN BIT
00117	C2A0	8D 01 DE	WOATN	STA S5380+1	;TURN ON SEL LINE
00118	C2A3				
00119	C2A3			; WAIT 1.2 USEC	
00120	C2A3				
00121	C2A3	EA		NOP	
00122	C2A4	AD 02 C0		LDA INITFL	;LOAD IN A SOFTWARE FLAG
00123	C2A7	D0 0D		BNE SEL	;IF FLAG SET, PERFORM INIT. SEL
00124	C2A9				
00125	C2A9			; ELSE, TARGET RESELECTION	
00126	C2A9				
00127	C2A9				
00128	C2A9	AD 02 DE		LDA S5380+2	;READ MODE REG.
00129	C2AC	09 40		ORA %01000000	;ENABLE TARGET MODE
00130	C2AE	8D 02 DE		STA S5380+2	;SET TARGET MODE
00131	C2B1	A9 01		LDA #\$01	;ENABLE ASSERT I/O
00132	C2B3	8D 03 DE		STA S5380+3	;SET ASSERT I/O
00133	C2B6				
00134	C2B6	AD 08 DE	SEL	LDA IID	; LOAD INITIATOR ID
00135	C2B9	0D 00 C0		ORA TID	; OR IN TARGET ID
00136	C2BC	8D 00 DE		STA S5380+0	;LOAD INT & TAR ID'S INTO ODR
00137	C2BF				
00138	C2BF			; TEST FOR DIFFERENTIAL PAIR	
00139	C2BF				
00140	C2BF	A9 05		LDA %00000101	;SEL & DATA BUS BITS
00141	C2C1	AE 03 C0		LDX OCFLAG	;LOAD IN A SOFTWARE FLAG
00142	C2C4	D0 02		BNE OPNCOL	;IF FLAG SET, OPEN COLLECTOR
00143	C2C6				
00144	C2C6			; DIFFERENTIAL PAIR	
00145	C2C6				
00146	C2C6	09 20		ORA %00100000	; OR IN DIFF. ENBL BITS
00147	C2C8	8D 01 DE	OPNCOL	STA S5380+1	;SET SEL, DATA BUS, & (DIFF. PAI
00148	C2CB	8D 01 C0		STA ICRVAL	; RETAIN VALUE OF INIT CMD REG.
00149	C2CE				
00150	C2CE			; RESET ARBITRATION BIT	
00151	C2CE				
00152	C2CE	AD 02 DE		LDA S5380+2	;READ MODE REGISTER
00153	C2D1	29 FE		AND %11111110	;MASK ARB BIT
00154	C2D3	8D 02 DE		STA S5380+2	;RESET ARB BIT
00155	C2D6				
00156	C2D6			; DISABLE THE SEL EN REGISTER TO AVOID A SEL INT.	
00157	C2D6				
00158	C2D6	A9 00		LDA #\$00	; ZERO ACCUM.
00159	C2D8	8D 04 DE		STA S5380+4	; ZERO SELECT ENABLE REG.
00160	C2DB				
00161	C2DB			; RELEASE BUSY	
00162	C2DB				
00163	C2DB	AD 01 C0		LDA ICRVAL	; GET INIT CMD REG VALUE

```

LINE# LOC CODE LINE
00164 C2DE 29 F7 AND #%11110111 ; MASK OUT BSY BIT
00165 C2E0 8D 01 DE STA S5380+1 ; RESET BSY
00166 C2E3 8D 01 C0 STA ICRVAL ; RETAIN ICR VALUE
00167 C2E6 ;
00168 C2E6 ; NOW WAIT 400NSEC AND BEGIN LOOKING FOR BSY
00169 C2E6 ;
00170 C2E6 A0 60 LDY ##60 ; LOAD UP X REG FOR COUNTER
00171 C2E8 A2 FF RELD LDX ##FF ; LOAD UP Y REG FOR COUNTER
00172 C2EA 2C 04 DE STIM BIT S5380+4 ; SAMPLE BSY BIT
00173 C2ED 70 18 BVS SLECT ; IF BSY ACTIVE, SELECTED
00174 C2EF ;
00175 C2EF ; WAIT 250 MSEC
00176 C2EF ;
00177 C2EF CA DEX ; DELAY
00178 C2F0 D0 F8 BNE STIM ; IF NOT ZERO LOOP
00179 C2F2 88 DEY
00180 C2F3 D0 F3 BNE RELD ; IF Y NOT ZERO RELOAD X
00181 C2F5 ;
00182 C2F5 ; SELECTION TIMEOUT
00183 C2F5 ;
00184 C2F5 A9 00 LDA ##00 ; TAR. DID NOT RESPOND TO SEL
00185 C2F7 8D 00 DE STA S5380+0 ; RESET ID BITS
00186 C2FA A2 20 LDX ##20 ; LOAD 200 USEC COUNTER
00187 C2FC 2C 04 DE CHK BIT S5380+4 ; CHECK BSY AGAIN
00188 C2FF 70 06 BVS SLECT ; IF SET SELECTION OK
00189 C301 CA DEX ;
00190 C302 D0 F8 BNE CHK ;
00191 C304 ;
00192 C304 ; SELECTION FAILED
00193 C304 ;
00194 C304 A9 01 LDA #SLFAIL ; LOAD STATUS IN ACCUM.
00195 C306 60 RTS ; RETURN TO CALLING PRGM
00196 C307 ;
00197 C307 ; SUCCESSFUL (RE)SELECTION
00198 C307 ;
00199 C307 AD 01 C0 SLECT LDA ICRVAL ; GET VALUE OF INIT CMD REG.
00200 C30A AE 02 C0 LDX INITFL ; GET INIT FLAG
00201 C30D D0 08 BNE IF ; IF INITIATOR JUMP
00202 C30F 09 08 ORA #%00001000 ; AND SET BSY IF TARGET.
00203 C311 8D 01 DE STA S5380+1 ; WRITE TO ICR
00204 C314 8D 01 C0 STA ICRVAL ; UPDATE PRESENT ICR VALUE
00205 C317 A9 28 IF LDA #%00101000 ; MASK TO RESET SEL & DATA BUS
00206 C319 2D 01 C0 AND ICRVAL ; AND WITH ICR VALUE
00207 C31C 8D 01 DE STA S5380+1 ; RESET SEL & DATA BUS
00208 C31F 8D 01 C0 STA ICRVAL ; UPDATE NEW ICR VALUE
00209 C322 ;
00210 C322 ; BEGIN TRANSFERS
00211 C322 ;
00212 C322 CA DEX ; DEC INITIATOR FLAG
00213 C323 F0 03 BEQ PDMA ; IF ZERO, INITIATOR ROLE
00214 C325 4C 2D C4 JMP RES ; ELSE, TARGET ROLE
00215 C328 ;
00216 C328 ; INITIATOR ROLE
00217 C328 ;
00218 C328 ; USE PSEUDO DMA MODE

```

LINE#	LOC	CODE	LINE
00219	C328		;
00220	C328	A9 00	PDMA LDA #DATA0 ; LOAD TCR W/DATA OUT PHASE
00221	C32A	8D 03 DE	STA S5380+3 ;
00222	C32D	AD 02 DE	NXT LDA S5380+2 ; GET MODE REGISTER
00223	C330	09 02	ORA #%00000010;OR IN DMA MODE BIT
00224	C332	8D 02 DE	STA S5380+2 ; SET DMA MODE BIT
00225	C335	AE 06 C0	LDX PHSIDX ; LOAD X W/PHASE INDEX
00226	C338	BD 08 C0	LDA XPTPHS+1,X; GET PHASE COUNT
00227	C33B	8D 25 C0	STA COUNT ; STORE IN PHASE COUNT BYTE
00228	C33E	BD 09 C0	LDA XPTPHS+2,X; GET COUNT MULTIPLIER
00229	C341	8D 26 C0	STA XCNT ; STORE IN MULTIPLIER
00230	C344		;
00231	C344		; WAIT FOR PHASE MISMATCH INT.
00232	C344		;
00233	C344	AD 05 DE	WAIT LDA S5380+5 ; SAMPLE BUS&STATUS REG.
00234	C347	29 10	AND #%00010000 ; LOOK FOR INT. REQ.
00235	C349	F0 F9	BEQ WAIT ; IF NOT SET, WAIT
00236	C34B		;
00237	C34B		; IRQ IS ACTIVE
00238	C34B		;
00239	C34B	AD 02 DE	LDA S5380+2 ; GET MODE REG.
00240	C34E	29 FD	AND #%11111101 ; RESET DMA MASK
00241	C350	8D 02 DE	STA S5380+2 ; RESET DMA MODE BIT
00242	C353	AD 05 DE	LDA S5380+5 ; GET BUS & STATUS REG
00243	C356	AD 05 DE	LSR ; SHIFT RIGHT 3 TIMES
00244	C359	AD 05 DE	LSR ;
00245	C35C	AD 05 DE	LSR ;
00246	C35F	B0 15	BCS EBUSY ; LOSS OF BUSY ERROR
00247	C361	B0 15 DE	LSR ; SHIFT
00248	C364	90 76	BCC PHSMM ; IF CARRY CLEAR, MISMATCH
00249	C366	90 76 DE	LSR ; SHIFT TWICE
00250	C369	90 76 DE	LSR ;
00251	C36C	B0 0B	BCS EPRTY ; IF SET, PARITY ERROR
00252	C36E	2C 04 DE	BIT S5380+4 ; GET CURRENT SCSI BUS STATUS
00253	C371	30 14	BMI BRST ; IF BIT 7 SET, BUS RESET OCCURED
00254	C373	4C 8A C3	JMP FAIL ; SHOULD NOT GET HERE
00255	C376		;
00256	C376		; RETURN ERROR STATUS TO CALLING PROGRAM
00257	C376		;
00258	C376	A9 02	EBUSY LDA #DISCNT ; SET DISCONNECT FLAG
00259	C378	60	RTS
00260	C379	AD 01 C0	EPRTY LDA ICRVAL ; GET INIT. CMD REG. VALUE
00261	C37C	09 02	ORA #%00000010 ; TURN ON ATN SIGNAL
00262	C37E	8D 01 DE	STA S5380+1 ; SET ATN
00263	C381	8D 01 C0	STA ICRVAL
00264	C384	A9 03	LDA #PRTYER ; SET PARITY ERROR
00265	C386	60	RTS
00266	C387	A9 04	BRST LDA #BUSRST ; SET BUS RESET ERROR
00267	C389	60	RTS
00268	C38A	A9 05	FAIL LDA #CHIPFL ; SET CHIP FAIL ERROR
00269	C38C	60	RTS ; RETURN TO CALLING PRGM
00270	C38D		;
00271	C38D		; WAIT FOR TARGET SELECTION
00272	C38D		;
00273	C38D	AD 02 DE	TARSEL LDA S5380+2 ; GET MODE REG.

LINE#	LOC	CODE	LINE		
00274	C390	09 40		DRA #X01000000	; SET TARGET MODE MASK
00275	C392	8D 02 DE		STA S5380+2	; SET TARGET MODE BIT
00276	C395	AD 08 DE		LDA IID	; GET TARGET ID
00277	C398	8D 04 DE		STA S5380+4	; STORE IN SELECT ENABLE REG.
00278	C39B	AD 05 DE	LOOK	LDA S5380+5	; SAMPLE BUS&STATUS REG.
00279	C39E	29 10		AND #X00010000	; LOOK FOR INT REQ
00280	C3A0	F0 F9		BEQ LOOK	; KEEP WAITING
00281	C3A2			:	
00282	C3A2			:	
00283	C3A2			; CHECK FOR MORE THAN TWO ID'S ACTIVE	
00284	C3A2			:	
00285	C3A2	AD 00 DE		LDA S5380+0	; READ SCSI DATA BUS
00286	C3A5	A2 09		LDX #X09	; SHIFT COUNT
00287	C3A7	A0 00		LDY #X00	; INITIALIZE BIT COUNT
00288	C3A9	A0 00 DE	UP	LSR	; SHIFT BIT INTO CARRY BIT
00289	C3AC	CA		DEX	; DECR. SHIFT COUNT
00290	C3AD	F0 05		BEQ OUT	; IF ZERO, DONE COUNTING
00291	C3AF	90 F8		BCC UP	; IF CARRY NOT SET, DO NEXT
00292	C3B1	C8		INY	; IF CARRY SET BUMP BIT CNT
00293	C3B2	B0 F5		BCS UP	; GET NEXT BIT
00294	C3B4	38	OUT	SEC	; SET CARRY BIT
00295	C3B5	98		TYA	; PUT Y IN ACCUM
00296	C3B6	E9 03		SBC #X03	; SUBTRACT 3 FROM BIT COUNT
00297	C3B8	30 06		BMI CI	; IF MINUS, OK
00298	C3BA	AD 07 DE		LDA S5380+7	; NOT MINUS, RESET IRQ.
00299	C3BD	4C 9B C3		JMP LOOK	; WAIT FOR GOOD SELECTION
00300	C3C0			:	
00301	C3C0			; CHECK INTERRUPT	
00302	C3C0			:	
00303	C3C0	AD 05 DE	CI	LDA S5380+5	; SAMPLE AGAIN
00304	C3C3	29 20		AND #X00100000	; MASK PARITY BIT
00305	C3C5	F0 B2		BEQ EPTY	; PARITY SELECTION ERROR
00306	C3C7	AD 04 DE		LDA S5380+4	; GET CURRENT SCSI BUS ST.
00307	C3CA	29 02		AND #X00000010	; CHECK SEL
00308	C3CC	F0 BC		BEQ FAIL	; IF NOT SET, FAILURE
00309	C3CE	A9 08		LDA #X00001000	; SET BSY MASK
00310	C3D0	8D 01 DE		STA S5380+1	; SET BSY SEL COMPLETE
00311	C3D3	8D 01 C0		STA ICRVAL	; RETAIN ICR VALUE
00312	C3D6	AD 07 DE		LDA S5380+7	; RESET INTERRUPT
00313	C3D9	4C 2D C4		JMP RES	;
00314	C3DC			:	
00315	C3DC			; PHASE MISMATCH CONDITION	
00316	C3DC			:	
00317	C3DC	AE 06 C0	PHSMM	LDX PHSIDX	; LOAD X WITH PHASE POINTER
00318	C3DF	AD 04 DE		LDA S5380+4	; LOAD CURRENT SCSI BUS STATUS
00319	C3E2	29 1C		AND #X00011100	; MASK ALL BUT PHASE BITS
00320	C3E4	DD 07 C0		CMP XPTPHS, X	; COMPARE TO XPTED PHASE
00321	C3E7	F0 03		BEQ PHSMTH	; YES, PHASE MATCHES
00322	C3E9	4C B4 C4		JMP DP	; ELSE, DIFFERENT PHASE
00323	C3EC			:	
00324	C3EC			; PHASE MATCHES EXPECTED PHASE	
00325	C3EC			:	
00326	C3EC	4C B4 C4	PHSMTH	LSR	; SHIFT TO TCR REG. FORMAT
00327	C3EF	4C B4 C4		LSR	;
00328	C3F2	8D 03 DE		STA S5380+3	; STORE IN TCR

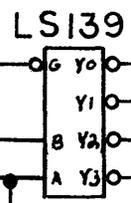
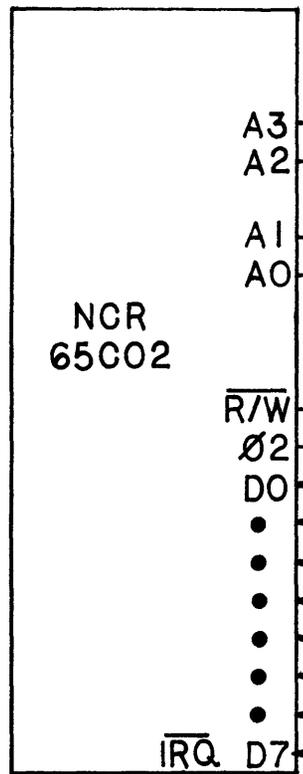
LINE#	LOC	CODE	LINE
00329	C3F5	AD 07 DE	LDA S5380+7 ; RESET INTERRUPT
00330	C3F8	A9 18	LDA #MESSO ; LOAD MESSOUT VALUE
00331	C3FA	DD 07 C0	CMP XTPHS,X ; WAS PHASE MATCH MESS. OUT
00332	C3FD	D0 0B	BNE GMR ; IF NOT MESS. OUT, CONTINUE
00333	C3FF		;
00334	C3FF		; MESSAGE OUT, RESET ATN
00335	C3FF		;
00336	C3FF	AD 01 C0	LDA ICRVAL ; GET INITIATOR CMD. REG
00337	C402	29 FD	AND #%11111101 ; MASK OFF ATN
00338	C404	8D 01 DE	STA S5380+1 ; TURN OFF ATN
00339	C407	8D 01 C0	STA ICRVAL ; UPDATE ICR VALUE
00340	C40A	AD 02 DE	GMR LDA S5380+2 ; GET MODE REG
00341	C40D	09 02	ORA #%00000010 ; SET DMA MODE BIT
00342	C40F	8D 02 DE	STA S5380+2 ; STORE IN TCR
00343	C412	AD 03 DE	LDA S5380+3 ; GET PHASE AGAIN
00344	C415	29 01	AND #%00000001 ; SET I/O MASK
00345	C417	F0 0B	BEQ IDMAO ; IF ZERO, DMA OUTPUT
00346	C419		;
00347	C419		; INITIATOR DMA INPUT
00348	C419		;
00349	C419	8D 07 DE	IDMAI STA S5380+7 ; START INIT. RCV.
00350	C41C	A0 00	LDY #00 ; INITIALIZE Y
00351	C41E	20 FE C4	JSR DMAIN ; PERFORM DMA INPUT
00352	C421	4C 2D C3	JMP NXT ; PREPARE FOR NEXT PHASE
00353	C424		;
00354	C424		; INITIATOR DMA OUTPUT
00355	C424		;
00356	C424	20 27 C5	IDMAO JSR DMAOUT ; PERFORM DMA OUTPUT
00357	C427	8D 0C DE	STA SDMA ; EXTRA WRITE FOR ACK TO GO OFF
00358	C42A	4C 2D C3	JMP NXT ; PREPARE FOR NEXT PHASE
00359	C42D		;
00360	C42D		; TARGET OPERATION
00361	C42D		;
00362	C42D	AD 02 DE	RES LDA S5380+2 ; GET MODE REGISTER
00363	C430	09 02	ORA #%00000010;OR IN DMA MODE BIT
00364	C432	8D 02 DE	STA S5380+2 ; SET DMA MODE BIT
00365	C435	AE 06 C0	LDX PHSIDX ; LOAD X W/PHASE INDEX
00366	C438	8D 08 C0	LDA XTPHS+1,X ; GET PHASE COUNT
00367	C43B	8D 25 C0	STA COUNT ; STORE IN PHASE COUNT BYTE
00368	C43E	8D 09 C0	LDA XTPHS+2,X ; GET COUNT MULTIPLIER
00369	C441	8D 26 C0	STA XCNT ; STORE IN MULTIPLIER
00370	C444		;
00371	C444	AD 05 DE	LDA S5380+5 ; GET BUS & STATUS REG.
00372	C447	29 02	AND #%00000010 ; MASK ATN BIT
00373	C449	D0 4D	BNE MESSOT ; ATN ACTIVE DO MESS OUT PHASE
00374	C44B	AE 06 C0	LDX PHSIDX ; GET CURRENT PHASE INDEX
00375	C44E	A9 80	LDA #DISCON ; GET DISCONNECTED VALUE
00376	C450	DD 07 C0	CMP XTPHS,X ; COMPARE W/PHASE VALUE
00377	C453	F0 3B	BEQ DISCTD ; IF =, TIME TO DISCONNECT
00378	C455	8D 07 C0	LDA XTPHS,X ; GET PHASE
00379	C458	8D 07 C0	LSR ; SHIFT TO TCR FORMAT
00380	C45B	8D 07 C0	LSR ;
00381	C45E	8D 03 DE	STA S5380+3 ; STORE IN TARGET COMMAND REG.
00382	C461	29 01	AND #%00000001 ; SAVE I/O BIT
00383	C463	F0 1D	BEQ TDMAO ; IF ZERO, DMA OUTPUT

LINE#	LOC	CODE	LINE
00384	C465		;
00385	C465		; TARGET DMA INPUT
00386	C465		;
00387	C465	8D 06 DE	TDMAI STA S5380+6 ; START DMA TARGET RCV
00388	C468	A0 01	LDY #01 ; SET Y TO ONE, SO NO EXTRA REQ
00389	C46A	20 FE C4	JSR DMAIN ; PERFORM DMA INPUT
00390	C46D		;
00391	C46D		; HANDLE LAST BYTE TO PREVENT EXTRA REQ
00392	C46D		;
00393	C46D	2C 05 DE	LSTDRQ BIT S5380+5 ; LOOK FOR DRQ
00394	C470	50 FB	BVC LSTDRQ ; LOOP TILL ON
00395	C472	AD 02 DE	LDA S5380+2 ; GET MODE REG.
00396	C475	29 FD	AND #%11111101 ; MASK DMA MODE BIT
00397	C477	8D 02 DE	STA S5380+2 ; RESET DMA MODE BIT
00398	C47A	AD 0C DE	LDA SDMA ; GET LAST BYTE FROM CHIP
00399	C47D	91 FB	STA (BPNT),Y ; STORE LAST BYTE
00400	C47F	4C 2D C4	JMP RES ; DO NEXT PHASE
00401	C482		;
00402	C482		; TARGET DMA OUTPUT
00403	C482		;
00404	C482	20 27 C5	TDMAO JSR DMAOUT ; PERFORM DMA OUTPUT
00405	C485	AD 02 DE	LDA S5380+2 ; GET DMA MODE
00406	C488	29 FD	AND #%11111101 ; MASK DMA MODE BIT
00407	C48A	8D 02 DE	STA S5380+2 ; RESET DMA MODE BIT
00408	C48D	4C 2D C4	JMP RES ; DO NEXT PHASE
00409	C490		;
00410	C490		; TARGET DISCONNECT
00411	C490		;
00412	C490	A9 00	DISCTD LDA #00 ; LOAD ACCUM W/ ZERO
00413	C492	8D 01 DE	STA S5380+1 ; RESET BSY & OTHER SIGNALS
00414	C495	A9 02	LDA #DISCNT ; DISCONNECTED STATUS
00415	C497	60	RTS ; RETURN TO CALLING PRGM
00416	C498		;
00417	C498		; MESSOUT PHASE (TARGET)
00418	C498		;
00419	C498	A9 18	MESSOT LDA #MESSO ; GET VALUE OF MESSAGE OUT
00420	C49A	A9 18 DE	LSR ; SHIFT TO TCR FORMAT
00421	C49D	A9 18 DE	LSR ;
00422	C4A0	8D 03 DE	STA S5380+3 ; MESSOUT PHASE
00423	C4A3	A9 01	LDA #1 ; LOAD MULTIPLIER/COUNTER VALUE
00424	C4A5	CA	DEX ; MOVE POINTER
00425	C4A6	9D 07 C0	STA XTPHS,X ; STORE MULTIPLIER
00426	C4A9	CA	DEX ; MOVE POINTER TO COUNT VALUE
00427	C4AA	9D 07 C0	STA XTPHS,X ; STORE COUNT
00428	C4AD	CA	DEX ; MOVE TO PHASE
00429	C4AE	8E 06 C0	STX PHSIDX ; UPDATE MOVED PHASE INDEX
00430	C4B1	4C 82 C4	JMP TDMAO ; DO DMA OUT
00431	C4B4		;
00432	C4B4		; DIFFERENT PHASE
00433	C4B4		;
00434	C4B4	A9 1C	DP LDA #MESSI ; LOAD VALUE OF MESSAGE IN PHASE
00435	C4B6	DD 07 C0	CMP XTPHS,X ; IS THIS A MESSAGE IN PHASE
00436	C4B9	F0 03	BEQ MESSIN ; IF=, READ MESSAGE
00437	C4BB	A9 07	LDA #DIFFPH ; LOAD DIFFERENT PHASE ST.
00438	C4BD	60	RTS ; RETN W/UNEXPECTED PHASE STATUS

LINE#	LOC	CODE	LINE
00439	C4BE		;
00440	C4BE		; MESSAGE IN PHASE
00441	C4BE		;
00442	C4BE	60 07 C0	MESSIN LSR ; SHIFT TO TCR FORMAT
00443	C4C1	60 07 C0	LSR ;
00444	C4C4	8D 03 DE	STA S5380+3 ; LOAD TCR
00445	C4C7	AD 07 DE	LDA S5380+7 ; RESET INT.
00446	C4CA	AD 04 DE	POLL LDA S5380+4 ; READ CURRENT BUS STATUS
00447	C4CD	29 20	AND #%00100000 ; LOOK FOR REQ.
00448	C4CF	F0 F9	BEQ POLL ; IF ZERO, NO REQ.
00449	C4D1	AD 01 C0	LDA ICRVAL ; GET CURRENT ICR VALUE
00450	C4D4	09 10	ORA #%00010000 ; OR IN ASSERT ACK
00451	C4D6	8D 01 DE	STA S5380+1 ; ASSERT ACK
00452	C4D9	8D 01 C0	STA ICRVAL ; UPDATE ICR
00453	C4DC	AD 04 DE	STILON LDA S5380+4 ; READ CURRENT BUS STATUS
00454	C4DF	29 20	AND #%00100000 ; LOOK FOR NOT REQ
00455	C4E1	D0 F9	BNE STILON ; IF NOT ZERO, STILL ON
00456	C4E3	A9 00	LDA #CMD CPL ; LOAD COMMAND COMPLETE
00457	C4E5		;
00458	C4E5		; LEAVE ACK ACTIVE SO MESSAGE CAN BE REJECTED
00459	C4E5		;
00460	C4E5	CD 06 DE	CMP S5380+6 ; COMPARE W/MESSAGE
00461	C4E8	D0 0E	BNE DIFMES ; IF NOT CMD COMPLTE, DIFF.
00462	C4EA	AD 01 C0	LDA ICRVAL ; GET ICR VAL
00463	C4ED	29 EF	AND #%11101111 ; MASK ACK BIT
00464	C4EF	8D 01 DE	STA S5380+1 ; RESET ACK
00465	C4F2	8D 01 C0	STA ICRVAL ; UPDATE ICR
00466	C4F5	4C 2D C3	JMP NXT ; GO TO NEXT PHASE
00467	C4F8		;
00468	C4F8		; NOT MESSAGE COMPLETE, RETURN FOR EVALUATION
00469	C4F8		;
00470	C4F8	AE 06 DE	DIFMES LDX S5380+6 ; GET MESSAGE VALUE
00471	C4FB	A9 06	LDA #MESSAG ; LOAD MESSAGE RETN STATUS
00472	C4FD	60	RTS ; RETURN FOR MESSAGE EVALUATION
00473	C4FE		;
00474	C4FE		;
00475	C4FE		; DMA INPUT
00476	C4FE		;
00477	C4FE	A9 00	DMAIN LDA #00 ; ZERO ACCUM.
00478	C500	AA	TAX ; ZERO X
00479	C501		;
00480	C501		; RESET ASSERT DATA BUS
00481	C501		;
00482	C501	AD 01 C0	LDA ICRVAL ; GET ICR VALUE
00483	C504	29 FE	AND #%11111110 ; MASK ASSERT DATA BUS
00484	C506	8D 01 DE	STA S5380+1 ; RESET ASSERT DATA BUS BIT
00485	C509	8D 01 C0	STA ICRVAL ; UPDATE ICR
00486	C50C		;
00487	C50C		; WAIT FOR DRQ
00488	C50C		;
00489	C50C	2C 05 DE	REPT1 BIT S5380+5 ; TEST FOR DRQ
00490	C50F	50 FB	BVC REPT1 ; IF NOT THERE, LOOP
00491	C511		;
00492	C511	AD 0C DE	GO1 LDA SDMA ; READ DMA PORT
00493	C514	91 FB	STA (BPNTR),Y ; STORE DATA IN BUFFER

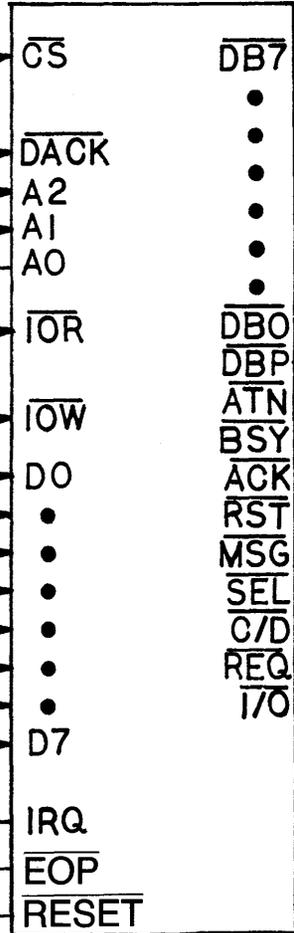
LINE#	LOC	CODE	LINE		
00494	C516	C8		INY	; INCR. POINTER
00495	C517	CC 25 C0		CPY COUNT	; DONE?
00496	C51A	D0 F0		BNE REPT1	; IF NOT ZERO, REPEAT
00497	C51C	E8		INX	; ZERO, CHECK MULTIPLIER
00498	C51D	EC 26 C0		CPX XCNT	; COMPARE X WITH MULTIPLIER
00499	C520	F0 32		BEQ NXTPHS	; IF EQUAL, COUNT DONE
00500	C522	E6 FC	BUMP	INC BPNTR+1	; GREATER THAN 256 BYTES BUMP MSB
00501	C524	4C 11 C5		JMP GO1	; GET MORE BYTES
00502	C527			;	
00503	C527			; DMA OUTPUT	
00504	C527			;	
00505	C527	A9 01	DMAOUT	LDA #%00000001	; SET MASK
00506	C529	0D 01 C0		ORA ICRVAL	; OR WITH ICR VALUE
00507	C52C	8D 01 DE		STA S5380+1	; SET ASSERT DATA BUS BIT
00508	C52F	8D 01 C0		STA ICRVAL	; UPDATE ICR VALUE
00509	C532	8D 05 DE		STA S5380+5	; START DMA INIT SEND
00510	C535			;	
00511	C535			; LOOK FOR DMA REQ (DRQ)	
00512	C535			;	
00513	C535	A9 00		LDA #00	; ZERO ACCUM
00514	C537	AB		TAY	; ZERO Y
00515	C538	AA		TAX	; ZERO X
00516	C539	2C 05 DE	REPT	BIT S5380+5	; SAMPLE DRQ
00517	C53C	50 FB		BVC REPT	; IF NOT SET, REPEAT
00518	C53E	B1 FB	GO	LDA (BPNTR), Y	; GET BYTE FROM BLOCK
00519	C540	8D 0C DE		STA SDMA	; WRITE BYTE TO CHIP
00520	C543	C8		INY	; INC Y POINTER
00521	C544	CC 25 C0		CPY COUNT	; COMPARE WITH BYTE CNT
00522	C547	D0 F0		BNE REPT	; IF Y NOT EQ. SEND MORE
00523	C549	E8		INX	; IF EQUAL INCR. X
00524	C54A	EC 26 C0		CPX XCNT	; COMPARE W/ MULTIPLIER
00525	C54D	F0 05		BEQ NXTPHS	; IF EQ, NEXT PHASE
00526	C54F	E6 FC		INC BPNTR+1	; MORE THAN 256 BUMP MSB
00527	C551	4C 3E C5		JMP GO	; SEND MORE DATA
00528	C554			;	
00529	C554			; NEXT PHASE	
00530	C554			;	
00531	C554	EE 06 C0	NXTPHS	INC PHSIDX	; PHASE POINTER INDEX +3
00532	C557	EE 06 C0		INC PHSIDX	;
00533	C55A	EE 06 C0		INC PHSIDX	;
00534	C55D	60		RTS	; RETN TO INIT OR TRGT OPER.

FROM ADDRESS DECODE LOGIC

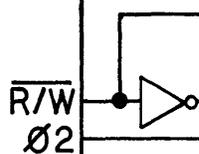
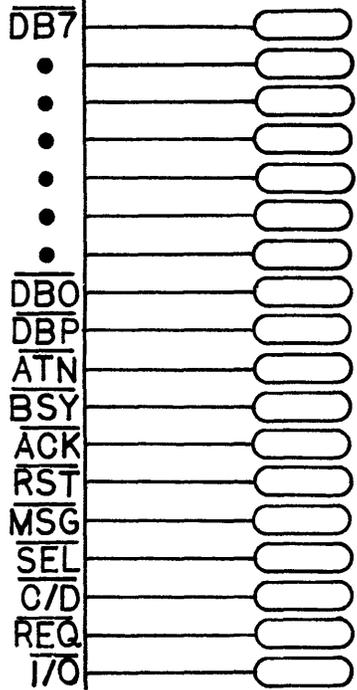


(XXX0H - XXX7H)
(XXXCH - XXXFH)

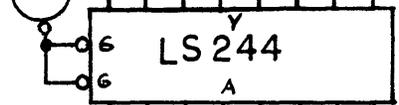
NCR 5380



SCSI BUS

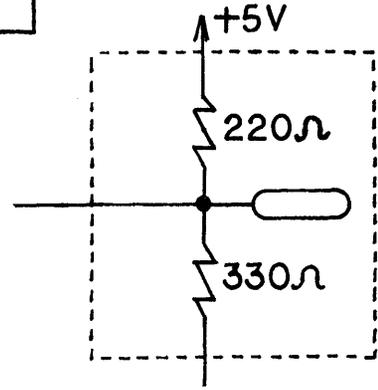


(XXX8H - XXXBH)



SCSI DEVICE ID STRAP

SCSI SIGNAL TERMINATION



A.3 65C02 Interface Diagram

A4. NCR 5381—Differential Pair Option

The NCR 5381 is a 48 pin version of the NCR 5380 device, designed to support external differential pair transceivers. These external transceivers are controlled with the additional signals provided in the higher pinout package. The NCR 5381 may still operate as a single-ended device if the SINGLEND signal (pin 2) is active. In single-ended operation, the signals provided for differential support remain functional.

The use of the DIFFERENTIAL ENABLE bit (bit 5) in the Initiator Command Register reflects the only

software difference between the two parts. When active, this bit is used to assert the signals IGS (pin 18) or TGS (pin 14) depending on the status of the TARGETMODE bit (port 2, bit 6). (IGS is active if TARGETMODE is false and TGS is active if TARGETMODE is true.) As in the NCR 5385/86, IGS is used to enable the external drivers for the signals \overline{ACK} (pin 17) and \overline{ATN} (pin 18) and TGS is used to enable the external drivers for the signals $\overline{I/O}$ (pin 20), $\overline{C/D}$ (pin 21), \overline{MSG} (pin 22) and \overline{REQ} (pin 23).

The signal differences between the NCR 5380 and the NCR 5381 are as follows:

Pin	SIGNAL NAME	DESCRIPTION
2	SINGLEND	This signal, when active (1) selects the single-ended mode of operation. When inactive, the NCR 5381 operates in the differential pair mode.

These signals will change from input/output pins to input only pins if the SINGLEND signal is false.

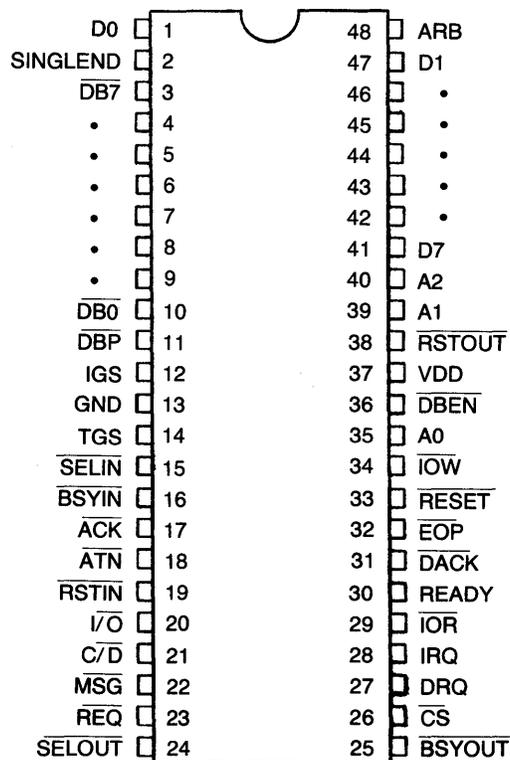
	SINGLEND = 1	SINGLEND = 0
PIN #16	\overline{BSY}	\overline{BSYIN}
PIN #15	\overline{SEL}	\overline{SELIN}
PIN #19	\overline{RST}	\overline{RSTIN}

36

\overline{DBEN}

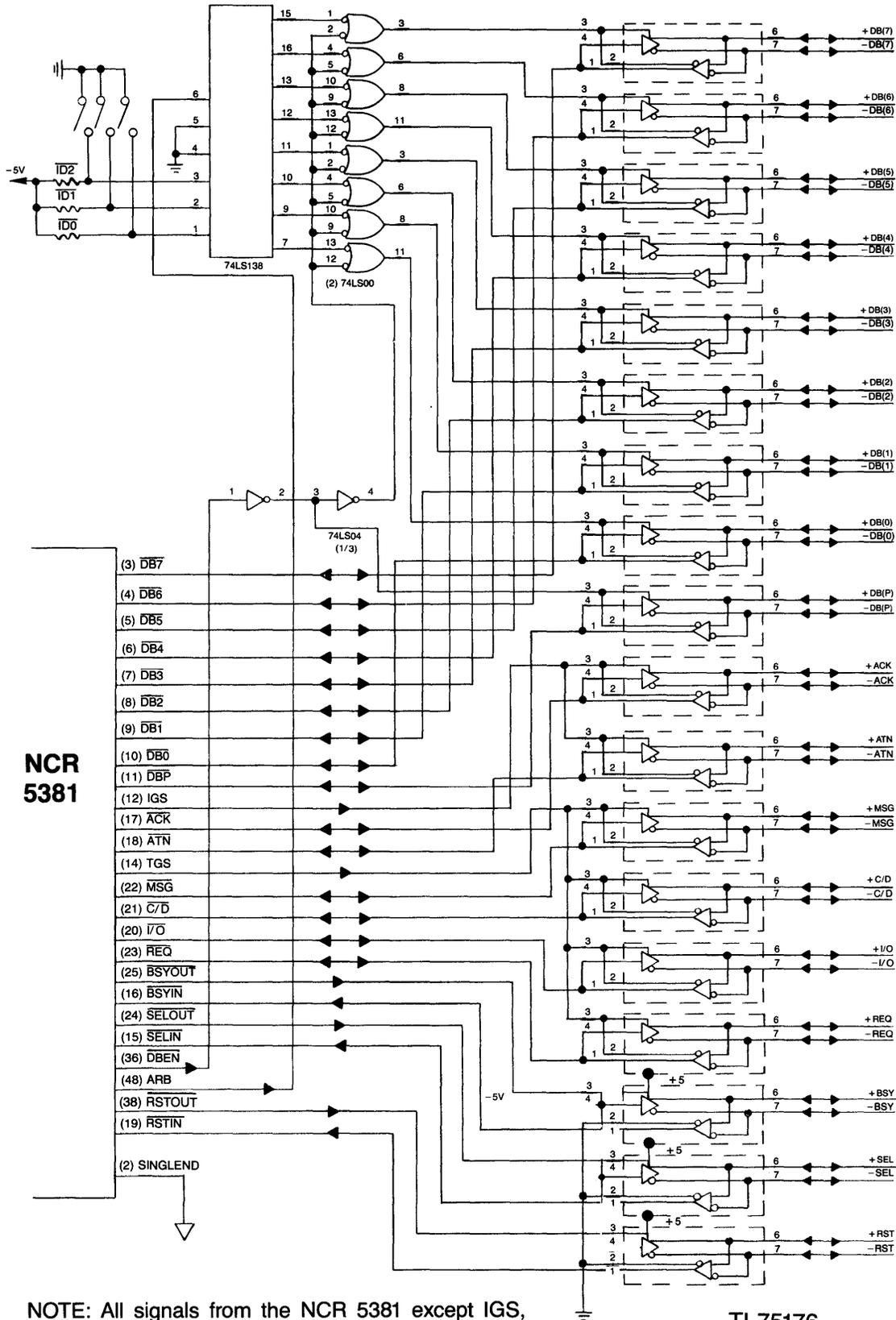
This signal is asserted whenever the ASSERT DATA BUS bit (port 1, bit 0) and the TARGETMODE bit are set (1). It is also asserted when ASSERT DATA BUS and PHASE MATCH are true and both TARGETMODE and $\overline{I/O}$ are false. This signal is used to enable the external transceivers to drive the data bus.

Pin	SIGNAL NAME	DESCRIPTION
14	TGS	This signal is active when the TARGETMODE bit and the DIFFERENTIAL ENABLE bit are true. It is used to enable the external transceivers to drive $\overline{I/O}$, $\overline{C/D}$, \overline{MSG} , and \overline{REQ} .
12	IGS	This signal is active when the TARGETMODE bit is false and the DIFFERENTIAL ENABLE bit is true. It is used to enable the external transceivers to drive \overline{ACK} and \overline{ATN} .
48	ARB	The NCR 5380 chip asserts this signal when the ARBITRATION bit is set and the device has detected a bus free condition. It is used to assert the proper device ID on the bus during the arbitration phase.
25	\overline{BSYOUT}	This signal is active whenever \overline{BSY} is asserted. This signal will be inactive at all other times.
24	\overline{SELOUT}	This signal is active if the ASSERT \overline{SEL} bit is true. Conversely, this signal is inactive if the ASSERT \overline{SEL} bit is reset.
38	\overline{RSTOUT}	This signal is active if the ASSERT \overline{RST} bit is true. Conversely, this signal is inactive if the ASSERT \overline{RST} bit is reset.



NCR 5381 PINOUT

NCR 5381 Suggested Differential Pair Interface.



NOTE: All signals from the NCR 5381 except IGS, TGS, DBEN, and ARB have open collector outputs and require pull-ups.

TI 75176
OR
NATIONAL
3695

A5. NCR 53C80 PRODUCT INFORMATION

GENERAL:

The NCR 53C80 CMOS SCSI interface device is functionally equivalent to the NMOS NCR 5380 device. However, this chip does not offer pin compatibility.

CMOS and NMOS are entirely different processes with inherently different device characteristics. All CMOS devices are designed to prohibit a condition commonly referred to as 'latch-up'. When the high current 48 ma drivers in the NCR 53C80 switch, noise level is sufficiently high, the output drivers become more prone to 'latch-up'. Therefore, for the CMOS design, pin compatibility has been foresaken for reliability. Four additional ground lines have been added to increase design stability. Having additional ground signals also allows the output buffers to switch more quickly, creating the added benefit of a faster design. This does not imply that the NMOS device is marginal in any respect.

DIFFERENCES:

To make the device easier to design with, the NCR 53C80 contains some minor modifications. An explanation of each is listed below.

Spurious RST Interrupt

If the NCR 5380 is not terminated on the SCSI bus, the floating condition on the input of the RST signal can generate spurious interrupts.

The NCR 53C80 contains an internal 30 uamp pull-up on the RST signal to prevent an interrupt due to an un-terminated bus.

True End Of DMA Status For Send Operations

If sending data to the NCR 5380 and EOP is asserted on the last byte, the END OF DMA status bit indicates only that the last byte has been received from the DMA device. There is no indication that this byte has been transferred to the SCSI bus.

The NCR 53C80 uses bit 7 of the Target Command register to indicate that the last byte of a DMA transfer has been sent to the SCSI bus.

Improved REQ/ACK Transition Times

The NCR 53C80 has improved the response times of the REQ/ACK handshake signals. This has an overall impact on achievable transfer rates. This performance improvement has been attained by increasing the number of ground lines, using faster CMOS cells, and giving cell placement priority to the REQ/ACK signal paths.

Prevents The Possibility Of An Additional ACK From Occurring

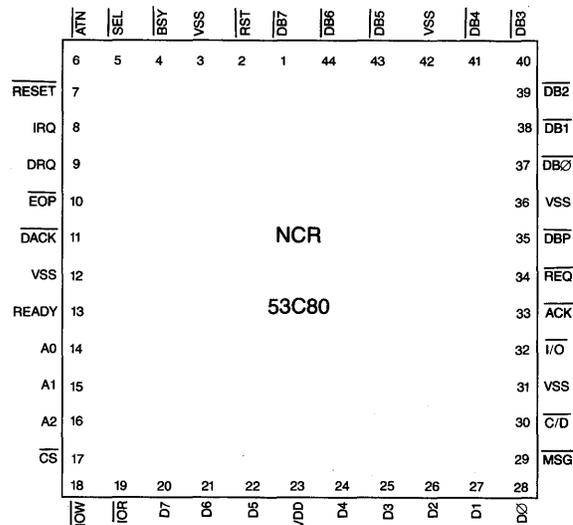
The NCR 5380, upon receipt of an EOP signal, sets the END OF DMA status bit and prevents additional DMA requests (DRQ). It does not reset the DMA Mode bit. If receiving data as an Initiator and the Target continues to request data for the same bus phase, after receiving an EOP pulse, the NCR 5380 will assert ACK without issuing a DMA request (DRQ).

The NCR 53C80 prevents ACK from being asserted until the device is instructed to continue by writing the Start DMA Initiator Receive register.

PERFORMANCE:

Even though early material on the NCR 53C80 has demonstrated better performance than the NCR 5380, device characterization was not completed before this section was printed.

PINOUT:



DB7/	1		48	DB6/
RST/	2		47	DB5/
GND	3		46	GND
BSY/	4		45	DB4/
SEL/	5		44	DB3/
ATN/	6		43	DB2/
NC	7		42	NC
RESET/	8		41	DB1/
IRQ	9		40	DB0/
DRQ	10		39	GND
EOP/	11	NCR	38	DBP/
DACK/	12	53C80	37	REQ
GND	13		36	ACK
READY	14		35	I/O/
AO	15		34	GND
A1	16		33	C/D/
A2	17		32	MSG/
NC	18		31	NC
CS/	19		30	DO
IOW/	20		29	D1
IOR/	21		28	D2
D7	22		27	D3
D6	23		26	D4
D5	24		25	VDD

A6. SCSI/PLUS *

AMPRO Computers, Inc. is proposing a general enhancement to the SCSI specification which allows the bus to operate as either a single or multi-master high speed parallel bus, capable of accessing up to 64

modules. This new bus structure is referred to as SCSI/PLUS. The table below describes the types of devices that may now be added due to the enhanced SCSI specification.

SCSI/PLUS DEVICE TYPES	EXAMPLES
Operating System Processors and Co-processors	UNIX MS-DOS CP/M FORTH Lisp Prolog
Communication Servers	Modems Arcnet Ethernet SDLC Mainframe links
Display Controllers	Graphics Text Touch
System Resources	Printer Spooler Time-of-day clock Speech I/O Protocol Converter DBMS Processor Array Processor
Real World Interfaces	A/D D/A AC & DC Control

Examples of SCSI/PLUS Devices

* SCSI/PLUS is a trademark of AMPRO Computers, Inc.

SCSI/PLUS provides three functional additions to the SCSI specification which allow the bus to operate as either a loosely coupled distributed system bus or a low-cost single master I/O bus. As proposed, SCSI/PLUS is a superset of the original specification, and its operation will not interfere with any existing SCSI implementation.

To allow for more complex system configurations, SCSI/PLUS provides Binary Arbitration and Binary Selection phases. The data bus represents a binary address and accommodates 64 physical bus devices, compared to eight in the current specification. In addition, four logical units may be associated with each bus device for a total of 256 logical bus devices. As in the SCSI specification, the arbitration phase is optional.

The addition of a master/slave mode to the specification provides for a cost-effective single-master/multi-slave configuration. This mode allows the design of SCSI/PLUS Targets which have no on-board intelligence. An optional interrupt protocol allows these "dumb" targets to asynchronously notify the bus master that they desire service.

To encourage board-level interchangeability, a recommended board size and interface connector is defined. The preferred board size is the single-wide Eurocard format with the double-wide card used as an option. The proposed interface connector is the DIN 41612—Type C connector. By using this form-factor and connector specification, bussed backplane or ribbon cable systems may be implemented.

The SCSI/PLUS architectural concept has inherent advantages over traditional microprocessor backplane architectures. SCSI/PLUS is CPU-independent, provides flexibility of form factor, operates across a ribbon cable bus, and allows both high-performance multi-master and low-cost single-master operation.

The NCR 5380 is an ideal part for designing an interface to connect to SCSI/PLUS. Its simplicity provides the flexibility needed to support the defined protocol modifications, and its popularity with SCSI users guarantees plug compatibility with existing host adapters.

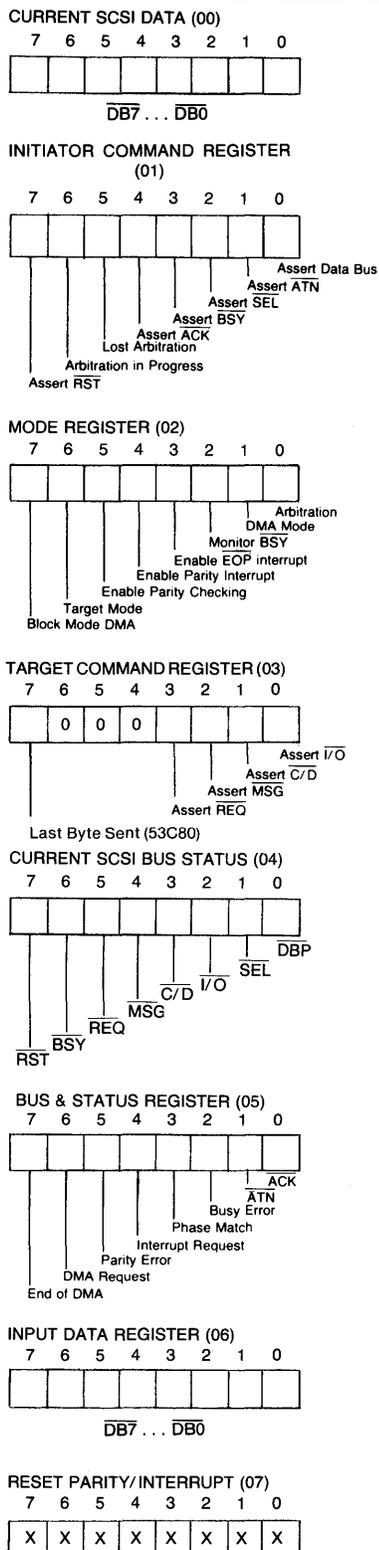
The NCR 5380 uses the Output Data Register to assert the proper device ID onto the SCSI bus during the Arbitration and Selection phases. Since the user is not restricted by the number of bits he is allowed to assert on the SCSI data bus, the Binary Arbitration and Binary Selection phases can be easily supported. In a Target role the Select Enable Register may be used to generate an interrupt if any bit in this register matches the binary address on the SCSI bus. Here again the NCR 5380 does not restrict this implementation.

The ability to support the master/slave operation requires independent control over the SCSI control signals by the bus slave devices and recognition by the bus master of the newly defined bus phase. The NCR 5380 provides independent signal control during Target operation and can be configured to generate an interrupt when a bus phase mismatch occurs if operating as an Initiator.

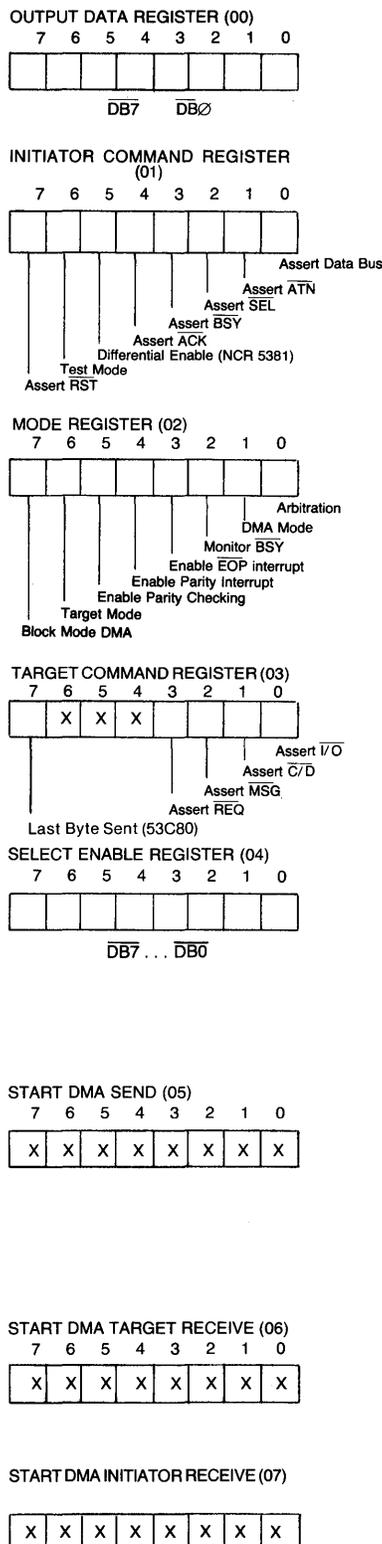
As in normal SCSI implementations, the use of on-chip bus transceivers significantly reduces parts count and provides for a highly reliable, cost effective SCSI/PLUS design. An additional advantage of on-chip MOS transceivers is the low leakage current. The NCR 5380 maximum leakage current of 50 uA meets the SCSI/PLUS bus load requirements. Up to 64 devices may occupy SCSI/PLUS bus positions if low-leakage integrated circuits such as the NCR 5380 are used.

A7. REGISTER REFERENCE CHART

READ



WRITE



NOTE: X = DON'T CARE



NCR Microelectronics Division
1635 Aeroplaza Drive
Logic Products Marketing
Colorado Springs, CO 80916
(303) 596-5612 or (800) 525-2252
TELEX 45-2457