National Semiconductor

PRELIMINARY
December 1991

DP8496/DP8497
SCSI-2 Disk Data Controller

General Description

The DP8496/7 is a highly integrated, high-performance CMOS SCSI disk data controller. It is designed for use inside intelligent hard disk drives that utilize the Small Computer System Interface (SCSI) standard. It can also be used in ESDI, SMD and ST506 bridging controller applications. The DP8496/7 includes most of the data path functions needed to implement a complete hard disk controller. It includes a full featured SCSI Bus Controller, Buffer Memory Interface with pipelined pointers, fast Disk Data Controller, and a Processor Interface.

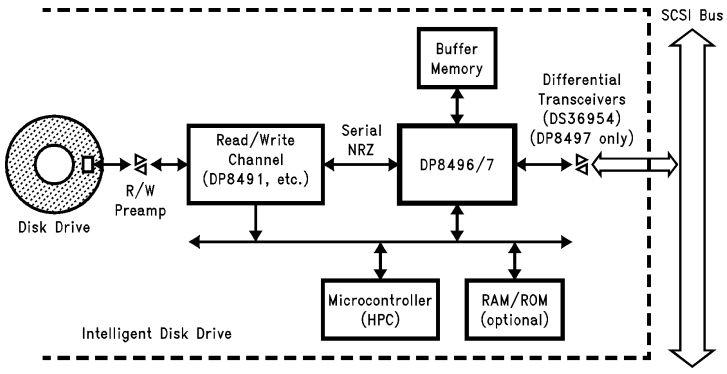
With the addition of National Semiconductor read-channel chips such as a PLL Synchronizer, and Encoder/Decoder, a pulse detector, and a head amplifier, complete data-path electronics of a SCSI drive can be implemented. A microcontroller, such as National's HPC™, may be used to manage the SCSI commands and the drive specific control signals. The high level of intelligence implemented on the DP8496/7 means lower overhead for the disk-drive's embedded microcontroller, making possible a high-performance design employing only one micro-controller.

The DP8496 provides on-chip single-ended transceivers for driving the SCSI bus. The DP8497 provides all control signals necessary for direct interfacing with differential transceivers recommended for Fast SCSI option of SCSI-2.

Features

- High disk data rates:
  - — DP8496/7-33 33 Mbit/sec
  - — DP8496/7-50 50 Mbit/sec
- Synchronous SCSI-2 transfer rates up to 10 MByte/sec with offset up to 16 (Fast option)
- Asynchronous SCSI transfer rates up to 5 MByte/sec
- Support for Fast Page Mode and Static Column Decode type DRAMs
- Attains sustained buffer bandwidths above 9 MByte/sec with byte-wide memory configuration
- Word-wide buffer memory port allows sustained bandwidths of 17 MByte/sec
- Buffer memory up to 4 MBytes DRAM or 1 MByte SRAM
- On-chip DMA with buffered pointer addresses
- Multi-phase type SCSI commands
- Parity error checking on SCSI, buffer memory, and all internal data paths
- Programmable format and sectoring modes including soft, pseudo-hard, and hard
- 32, 48 or 56-bit computer generated ECC with on-chip correction
- On-chip single-ended transceivers on DP8496. DP8497 interfaces directly with differential transceivers for Fast SCSI
- Available in 100-pin PQFP package

System Diagram



TL/F/11212–1

TRI-STATE® is a registered trademark of National Semiconductor Corporation.
HPC™ is a trademark of National Semiconductor Corporation.

Table of Contents

1.0 Overview

The DP8496/7 contains four major sections. Each section is listed below along with the major functions performed within that section.

1. Processor Interface
   Processor Interface for chip control
2. Buffer Memory Interface
   SRAM/DRAM control timing
   Memory access arbitration
   Memory access prioritization
3. Disk Data Controller
   Serializer/Deserializer (SERDES)
   Read/Write/Format Control
   CRC/ECC generation/checking/correcting
4. SCSI Bus Controller
   SCSI Data Transfer Control
   SCSI Bus Control—phase changes
   Parity generation/checking
   On-chip bus transceivers

The Processor Interface section allows the drive's processor access to all programmable features of the chip. This interface is used to initiate and control any function or operation on both disk and SCSI data. All DP8496/7 registers are accessed through this section.

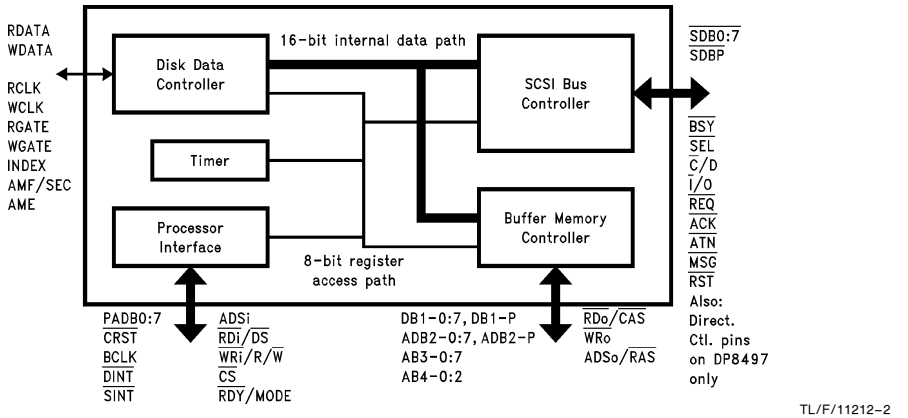
Buffer RAM is needed for all disk or SCSI data transfers. This RAM is connected to the Buffer Memory Interface section. The DP8496/7 assumes exclusive access to this buffer RAM. This enables the chip to utilize the full bandwidth of the RAM to streamline any combination of disk, SCSI or processor transfers. All transfer of data is done with on-chip DMA. Address pointers may be pipelined which will allow different groups of data to be placed in non-consecutive locations in buffer memory.

The Disk Data Controller section transfers NRZ data to the serial disk data path. Sector size, gaps, synch bytes, etc. are programmable. It can work with hard or soft sectored drives. ESDI soft sectored (pseudo-hard) AMF/AME handshaking is programmable. Fixed 32/48/56-bit ECC polynomial hardware automatically generates and checks error correction fields. Correction calculation is done on chip which will relieve the processor of the time and code space overhead of this function.

The SCSI Bus Controller section saves the user board area by integrating the 48 mA open drain drivers. The controller was designed to minimize the number of interrupts generated due to phase changes, parity errors and the like. Groups of often used phase sequences can be invoked with a single command.

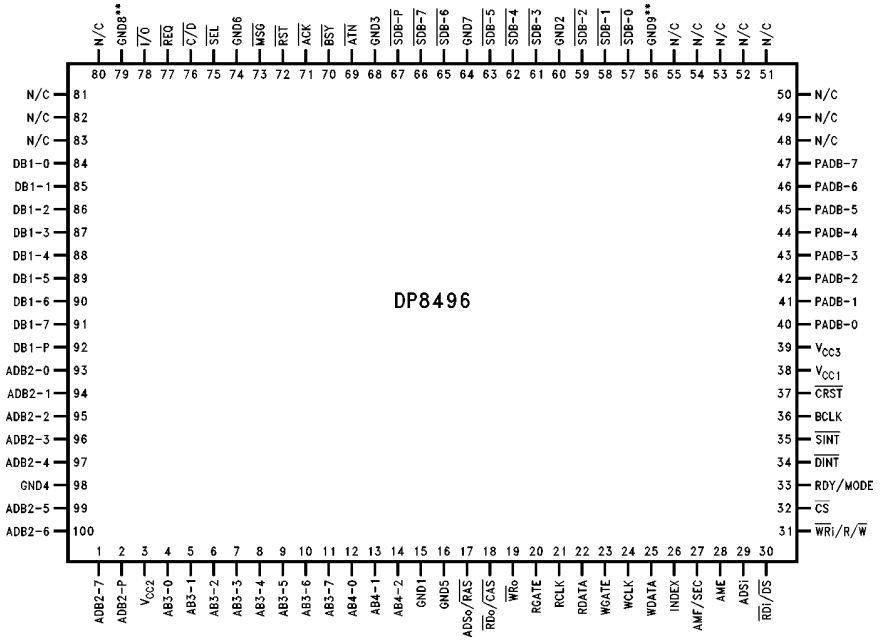
An internal timer is also available on the DP8496/7 which may be used to accurately control the execution of certain commands for the disk or SCSI sections.

Block Diagram



TL/F/11212–2

2.0 Pin Description

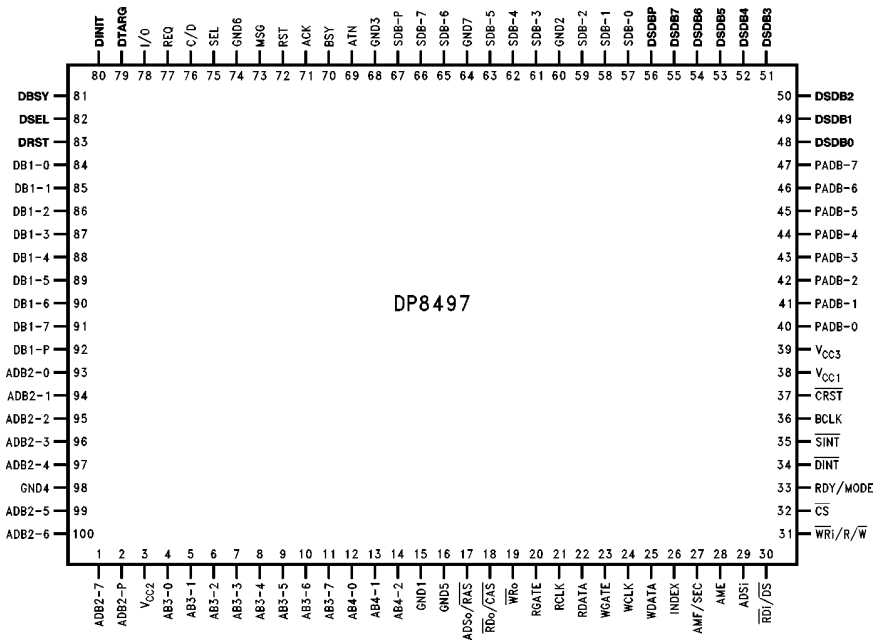


**Extra Ground Pins Only on DP8496

TL/F/11212–3

Connection Diagram

100-Pin QFP



Signals shown in **bold** are unique to the DP8497

TL/F/11212–4

Order Number DP8496VF or DP8497VF
See NS Package Number VF100B

2.0 Pin Description (Continued)

Symbol	Pin	Type	Function
DISK DATA CONTROLLER PINS			
RGATE	20	O	**READ GATE:** This active high output is asserted while the Disk Data Controller is reading data from the disk. It commands an external data separator to acquire lock and enables the RDATA pin.
RCLK	21	I	**READ CLOCK:** This input is the disk data rate clock. When RGATE is deasserted low, this pin will receive the crystal or servo derived clock. When RGATE is asserted high, this pin will receive the recovered NRZ clock from the decoder. Each rising edge of the clock at this input is used to strobe RDATA into the Disk Data Controller. The AC timing characteristics should not be violated, even during the time of transition between clock sources.
RDATA	22	I	**READ DATA:** This active high input accepts NRZ disk data from the data synchronizer/ decoder.
WGATE	23	O	**WRITE GATE:** This active high output is asserted while writing data to the disk.
WCLK	24	O	**WRITE CLOCK:** This output is derived from RCLK and is used to clock NRZ WDATA out from the DP8496/7. The rising edge indicates vaild WDATA.
WDATA	25	O	**WRITE DATA:** This active high output is the NRZ data to be written to the disk. It is synchronized to WCLK. It is held deasserted any time WGATE is deasserted.
INDEX	26	I	**INDEX:** This active high input from the disk drive signifies the start of a track. Disk Data Controller commands may be synchronized to it.
AMF/SEC	27	I	**ADDRESS MARK FOUND/SECTOR:** This active high input denotes the start of a sector in hard and pseudo-hard sectored drives. Soft sectored drives use AMF to denote the start of header and data fields.
AME	28	O	**ADDRESS MARK ENABLE:** This active high output forces the read-channel encoder to generate an Address Mark. It is also used to enable Address Mark detection for pseudo-hard sectored drives.
PROCESSOR INTERFACE PINS			
$\overline{\text{CRST}}$	37	I	**RESET:** This active low Schmitt input will reset the DP8496/7 immediately without regard to data transfers which may be in progress. Registers affected are listed in Table 3.1.
PADB0:7	40–47	I/O	**PROCESSOR/ADDRESS/DATA BUS 0–7:** These eight active high, bi-directional lines transfer information between the DP8496/7 and the processor.
ADSi	29	I	**ADDRESS STROBE IN:** This active high input latches the address from PADB0–7 on the falling edge. The latched address is used to select internal registers.
RDi or $\overline{\text{DS}}$	30	I	The function of this pin is determined by the processor mode initialized by the $\overline{\text{RDY}}$/MODE pin. **READ STROBE (IN):** NSC/Intel mode: This active low input combined with a low level on $\overline{\text{CS}}$ will assert data from the addressed register onto the PADB0–7 bus. **DATA STROBE:** Zilog/Motorola Mode: This active low input combined with a low level on $\overline{\text{CS}}$ will either assert data from the addressed register onto the PADB0–7 bus, or it will write data present on the PADB0–7 bus into the register. The data direction is determined by the W/R pin.
$\overline{\text{WR}}$i or R/$\overline{\text{W}}$	31	I	The function of this pin is determined by the processor mode initialized by the RDY/MODE pin. **WRITE STROBE (IN)**—NSC/Intel Mode: This active low input combined with a low level on $\overline{\text{CS}}$ will write data present on the PADB0–7 bus into the addressed register. **READ/WRITE**—Zilog/Motorola Mode: This pin determines the direction of data transfer on the PADB0–7 bus while the $\overline{\text{CS}}$ and $\overline{\text{DS}}$ pins are both asserted. High = Read, Low = Write.
$\overline{\text{CS}}$	32	I	**CHIP SELECT:** This active low input must be asserted to access any of the registers.

2.0 Pin Description (Continued)

Symbol	Pin	Type	Function
PROCESSOR INTERFACE PINS (Continued)			
RDY or MODE	33	I/O	The function of this pin is determined by the state of the $\overline{\text{CRST}}$ pin. **READY** ($\overline{\text{CRST}}$ Pin Deasserted): When low, this output indicates that the Buffer Memory Data register access must be extended by the processor. This pin can be connected to the processor's wait-state input. Refer to Section 4.2.8 for a more complete description of this option. This pin is only active while in the NSC/Intel mode. This pin will be driven low internally while in the Zilog/Motorola mode. **MODE** ($\overline{\text{CRST}}$ Pin Asserted): This input will determine the processor access mode of the DP8496/7. If this pin is left floating or pulled or driven high while $\overline{\text{CRST}}$ is asserted, the National/Intel mode will be enabled. If this pin is pulled or driven low while $\overline{\text{CRST}}$ is asserted, the Zilog/Motorola mode will be enabled. While $\overline{\text{CRST}}$ is asserted, an internal pull-up resister is active. See the DC specifications for characteristics.
$\overline{\text{DINT}}$	34	O	**DISK INTERRUPT:** This active low output will be asserted on any Disk Data Controller condition enabled by the Disk Interrupt Enable register. If the CI (Combine Interrupts) bit in the Setup 3 register is set to ''1'', this pin will become asserted if either a Disk Data Controller or a SCSI Bus Controller interrupt occurs.
$\overline{\text{SINT}}$	35	O	**SCSI INTERRUPT:** This active low output will be asserted on any SCSI Bus Controller condition enabled by the SCSI Interrupt Enable register.
BCLK	36	I	**BUS CLOCK:** This input is used by the Buffer Memory Interface for access timing and DMA arbitration. It is also used by the SCSI Bus Controller for timing.
SCSI BUS CONTROLLER PINS (All SCSI Signals are Active High on the DP8497)			
$\overline{\text{SDB0:7}}$, $\overline{\text{SDBP}}$	57–59 61–63 65–67	I/O	**SCSI DATA, PARITY BUS:** These nine active low, open drain, bi-directional lines should be connected directly to the SCSI data bus.
$\overline{\text{BSY}}$	70	I/O	**BUSY:** Same as above. Should be connected to SCSI control bus.
$\overline{\text{SEL}}$	75	I/O	**SELECT:** Same as above.
$\overline{\text{C/D}}$	76	I/O	**COMMAND/DATA:** Same as above.
$\overline{\text{I/O}}$	78	I/O	**INPUT/OUTPUT:** Same as above.
$\overline{\text{REQ}}$	77	I/O	**REQUEST:** Same as above.
$\overline{\text{ACK}}$	71	I/O	**ACKNOWLEDGE:** Same as above.
$\overline{\text{ATN}}$	69	I/O	**ATTENTION:** Same as above.
$\overline{\text{MSG}}$	73	I/O	**MESSAGE:** Same as above.
$\overline{\text{RST}}$	72	I/O	**SCSI RESET:** If $\overline{\text{CRST}}$ is asserted during power-up, the RST pin will not produce a glitch. See AC Specifications for input de-glitch description.

2.0 Pin Description (Continued)

Symbol	Pin	Type	Function
SCSI BUS TRANSCEIVER DIRECTION CONTROL PINS (DP8497 Only)			
DSDB0:7, DSDBP	48–56	O	**DIRECTION CONTROL FOR SCSI DATA, PARITY BUS:** Each of these signals connects to the appropriate enable pin of the differential transceivers used for a Fast SCSI implementation and controls the direction of its corresponding SCSI data bus signal. A high level on these pins enables the driver while a low level enables the receiver.
DTARG	79	O	**DIRECTION CONTROL FOR TARGET SIGNALS:** Same as above. Controls the direction of $\overline{C/D}$, $\overline{I/O}$, $\overline{REQ}$ and $\overline{MSG}$ signals.
DINIT	80	O	**DIRECTION CONTROL FOR INITIATOR SIGNALS:** Same as above. Controls the direction of $\overline{ACK}$ and $\overline{ATN}$ signals.
DBSY	81	O	**DIRECTION CONTROL FOR $\overline{BSY}$:** Same as above. Controls the direction of $\overline{BSY}$ signal.
DSEL	82	O	**DIRECTION CONTROL FOR $\overline{SEL}$:** Same as above. Controls the direction of $\overline{SEL}$ signal.
DRST	83	O	**DIRECTION CONTROL FOR $\overline{RST}$:** Same as above. Controls the direction of $\overline{RST}$ signal.
BUFFER MEMORY INTERFACE PINS			
DB1–0:7, DB1–P	84–92	I/O	**DATA BUS 1:** These nine active high, bi-directional lines transfer data between the DP8496/7 and the least significant 8-bits plus parity of buffer memory.
ADB2–0:7, ADB2–P	93–97 99–100 1–2	I/O	**ADDRESS DATA BUS 2:** These nine active high, bi-directional lines transfer data between the DP8496/7 and the most significant 8-bits plus parity of buffer memory if word mode is enabled. These pins also contain address information for static RAM. See Tables 4.2 and 4.3.
AB3–0:7	4–11	O	**ADDRESS BUS 3:** These eight active high outputs represent address information for both SRAM and DRAM. See Tables 4.2 to 4.5.
AB4–0:2	12–14	O	**ADDRESS BUS 4:** These three active high outputs represent address information for both SRAM and DRAM. See Tables 4.2 to 4.5.
$\overline{RD}$o or $\overline{CAS}$	18	O	The function of this pin is determined by the use of SRAM or DRAM. **READ STROBE (OUT):** SRAM. **COLUMN ADDRESS STROBE:** DRAM.
$\overline{WR}$o	19	O	**WRITE STROBE (OUT):** Always.
ADSo or $\overline{RAS}$	17	O	The function of this pin is determined by the use of SRAM or DRAM. **ADDRESS STROBE (OUT):** SRAM. In word mode, controls address latch for ADB2 bus. High level on this signal should be used by an external latch to accept address inputs—latching them on the falling edge. **ROW ADDRESS STROBE:** DRAM.

2.0 Pin Description (Continued)

Pin	Symbol	Function
1	ADB2–7	Buffer Memory
2	ADB2–P	Address/Data Bus 2
3	V_{CC2}	+5 V_{DC}
4	AB3–0	Buffer Memory
5	AB3–1	Address Bus 3
6	AB3–2	
7	AB3–3	
8	AB3–4	
9	AB3–5	
10	AB3–6	
11	AB3–7	
12	AB4–0	Buffer Memory
13	AB4–1	Address Bus 4
14	AB4–2	
15	GND1	GND
16	GND5	GND
17	ADSo/$\overline{RAS}$	Buffer Memory
18	$\overline{RD}$o/$\overline{CAS}$	Strobes
19	$\overline{WR}$o	
20	RGATE	Disk Read Gate
21	RCLK	Disk Read Clock
22	RDATA	Disk Read Data
23	WGATE	Disk Write Gate
24	WCLK	Disk Write Clock
25	WDATA	Disk Write Data
26	INDEX	Disk Index Input
27	AMF/SEC	Disk Address Mark Found or Sector Input
28	AME	Address Mark Enable
29	ADSi	Processor Port
30	$\overline{RD}$i/$\overline{DS}$	Strobes
31	$\overline{WR}$i/R/$\overline{W}$	
32	$\overline{CS}$	
33	RDY/MODE	Ready/Mode
34	$\overline{DINT}$	Disk Interrupt
35	$\overline{SINT}$	SCSI Interrupt
36	BCLK	Bus Clock
37	$\overline{CRST}$	Chip Reset
38	V_{CC1}	+5 V_{DC}
39	V_{CC3}	+5 V_{DC}
40	PADB-0	Processor Address/
41	PADB-1	Data Bus
42	PADB-2	
43	PADB-3	
44	PADB-4	
45	PADB-5	
46	PADB-6	
47	PADB-7	
48	DSDB0	Transceiver Direction
49	DSDB1	Control for DP8497
50	DSDB2	(N/C on DP8496)
51	DSDB3	
52	DSDB4	Direction Control (Continued)
53	DSDB5	
54	DSDB6	
55	DSDB7	
56	DSDBP(DP8497)	
	GND9 (DP8496)	GND on DP8496 Only
57	$\overline{SDB\text{-}0}$	SCSI Data Bus
58	$\overline{SDB\text{-}1}$	
59	$\overline{SDB\text{-}2}$	
60	GND2	GND
61	$\overline{SDB\text{-}3}$	SCSI Data Bus
62	$\overline{SDB\text{-}4}$	
63	$\overline{SDB\text{-}5}$	
64	GND7	GND
65	$\overline{SDB\text{-}6}$	SCSI Data Bus
66	$\overline{SDB\text{-}7}$	
67	$\overline{SDB\text{-}P}$	SCSI Parity
68	GND3	GND
69	$\overline{ATN}$	SCSI Attention
70	$\overline{BSY}$	SCSI Busy
71	$\overline{ACK}$	SCSI Acknowledge
72	$\overline{RST}$	SCSI Reset
73	$\overline{MSG}$	SCSI Message
74	GND6	GND
75	$\overline{SEL}$	SCSI Select
76	$\overline{C/D}$	SCSI Cmd/Data
77	$\overline{REQ}$	SCSI Request
78	$\overline{I/O}$	SCSI Input/Output
79	GND8 (DP8496)	Additional GND on DP8496
	DTARG ('97)	Transceiver Direction
80	DINIT	Control on DP8497 only
81	DBSY	(N/C on DP8496)
82	DSEL	
83	DRST	
84	DB1–0	Buffer Memory Data
85	DB1–1	Bus 1
86	DB1–2	
87	DB1–3	
88	DB1–4	
89	DB1–5	
90	DB1–6	
91	DB1–7	
92	DB1–P	
93	ADB2–0	Buffer Memory
94	ADB2–1	Address/Data Bus 2
95	ADB2–2	
96	ADB2–3	
97	ADB2–4	
98	GND4	GND
99	ADB2–5	Buffer Memory
100	ADB2–6	Address/Data Bus 2

3.0 Register List

DISK DATA CONTROLLER REGISTERS

Addr.	Name	Label	R/W	Pg.
00–06	ECC Shift Registers 0–6	ESRx	R	29
07	ECC Control	EC	R/W	28
08–0F	Reserved			
10	Disk Command	DCMD	W	23
11	Disk Control	DCTL	R/W	25
12	Disk Interrupt	DINT	R/W	27
13	Disk Interrupt Enable	DINTE	R/W	27
14	Disk Status	DSTAT	R	26
15	Sector Number	SN	R/W	28
16	Header Byte Count/Interlock	HBC	R/W	28
17	Sector Count	SC	R/W	28
18	ECC Byte Count (Low)	ECCL	R	29
19	ECC Byte Count (High)	ECCH	R	29
1A–1F	Reserved			

DISK FORMAT REGISTERS

Addr.	Name	Label	R/W	Pg.
20	Post Sector/Index Count	PSIG	R/W	29
21	Header Preamble Count	HPC	R/W	29
22	Preamble Pattern	PREP	R/W	30
23	Header Synch #1, #2 Counts	HSC	R/W	30
24	Header Synch #1 Pattern	HSP1	R/W	30
25	Header Synch #2 Pattern	HSP2	R/W	30
26	Header Byte 0, 1 Control	HC01	R/W	30
27	Header Byte 0 Pattern	HP0	R/W	31
28	Header Byte 1 Pattern	HP1	R/W	31
29	Header Byte 2, 3 Control	HC23	R/W	31
2A	Header Byte 2 Pattern	HP2	R/W	31
2B	Header Byte 3 Pattern	HP3	R/W	31
2C	Header Byte 4, 5 Control	HC45	R/W	31
2D	Header Byte 4 Pattern	HP4	R/W	31
2E	Header Byte 5 Pattern	HP5	R/W	31
2F	Header ECC & Postamble Count	HECC	R/W	31
30	Postamble Pattern	POSTP	R/W	31
31	Data Preamble Count	DPC	R/W	31
32	Data Synch #1, #2 Count	DSC	R/W	31
33	Data Synch #1 Pattern	DSP1	R/W	32
34	Data Synch #2 Pattern	DSP2	R/W	32
35	Sector Byte Count (Low)	SBCL	R/W	32
36	Sector Byte Count (High)	SBCH	R/W	32
37	Data Format Pattern	DFP	R/W	32
38	Data ECC & Postamble Count	DECC	R/W	32
39	Gap 3 Byte Count	GAPC	R/W	32
3A	Gap 3 & PSIG Pattern	GAPP	R/W	32
3B–3F	Reserved			

SCSI REGISTERS

Addr.	Name	Label	R/W	Pg.
40	SCSI Command	SCMD	W	39
41	SCSI Data	SDAT	R/W	53
42	SCSI Control	SCTL	R/W	53
43	SCSI Operation	SOP	R/W	53
44	Synchronous Transfer	SYNC	R/W	54
45	Identify	IDENT	R/W	55
46	Destination ID	DID	R/W	56
47	Source ID	SID	R/W	56
48	SCSI Status	SSTAT	R	56
49	SCSI Interrupt	SINT	R	57
4A	SCSI Interrupt Enable	SINTE	R/W	60
4B	SCSI Block Count	SBC	R/W	60
4C	SCSI Block Size (MSB)	SBSH	R/W	60
4D	SCSI Block Size (LSB)	SBSL	R/W	60
4E	Differential SCSI 1	SDIF1	R/W	60
4F	Differential SCSI 2	SDIF2	R/W	60
50–5F	Reserved			

SETUP AND TIMER REGISTERS

Addr.	Name	Label	R/W	Pg.
60	Setup 1	SUP1	R/W	10
61	Setup 2	SUP2	R/W	11
62	Setup 3	SUP3	R/W	12
63	Timer Prescale	TPRE	R/W	64
64	Timer Count	TCNT	R/W	64

BUFFER MEMORY REGISTERS

Addr.	Name	Label	R/W	Pg.
65	Disk Pointer (MSB)	DP2	R/W	19
66	Disk Pointer	DP1	R/W	19
67	Disk Pointer (LSB)	DP0	R/W	19
68	SCSI Pointer (MSB)	SP2	R/W	19
69	SCSI Pointer	SP1	R/W	19
6A	SCSI Pointer (LSB)	SP0	R/W	19
6B	Processor Pointer (MSB)	PP2	R/W	19
6C	Processor Pointer	PP1	R/W	19
6D	Processor Pointer (LSB)	PP0	R/W	19
6E	Buffer Memory Data	BMD	R/W	19
6F–7F	Reserved			

3.0 Register List (Continued)

3.1 RESET SUMMARY

There are four different types of resets that may be issued. Each reset has its own function and purpose. These are summarized in Table 3.1.

3.2 INITIALIZATION REGISTERS

The three basic Setup Registers, that must be initialized before proper chip operation can begin, are described in this section. All other registers are described in their respective sections in the Functional Description chapter; i.e., in the Disk Data Controller, SCSI Bus Controller, Processor Interface, Buffer Memory Interface, and Timer sections.

Setup 1 register is typically written only once after a reset to establish the physical path between the DP8496/7 and its buffer memory.

Setup 2 register contains control bits which may be modified by the processor for power-up tests, error recovery or normal operations.

Setup 3 register is typically written only once after a reset to establish pin configuration.

Setup 1 (SUP1) 60h R/W

7	6	5	4	3	2	1	0
CLK1	CLK0	SWS	DFP	DD1	DD0	RSEL	RWID

The contents of this register will be invalid after asserting the $\overline{\text{CRST}}$ pin. It must be initialized before proper chip operation.

CLK(1:0): BCLK Frequency

The Bus Clock (BCLK) is used for many functions within the DP8496/7. It is used in the SCSI Bus Controller, Buffer Memory Interface, and Timer sections. Therefore, the choice of Bus Clock frequency is very important. Table 3.2 lists the frequency range allowable for each bit combination of this field to guarantee timing specifications within the SCSI standard.

The minimum BCLK frequency allowed is 10 MHz. Choosing a clock frequency nearest the upper limit in a range results in the optimal and fastest execution fo SCSI operations.

TABLE 3.1. Summary of Resets

Reset Type	Purpose	Result
Assert RST Bit in Disk Control Register	Reset Disk Data Controller	Clears and terminates current Disk command and any pipelined command. Clears Disk Status register. Clears bits 0–5 and 7 of the Disk Interrupt register. Deasserts RGATE, WGATE, and AME pins.
Issue SCSI Reset Command	Reset SCSI Bus Controller	Clears all SCSI Bus signals. Clears and terminates current SCSI command and any pipelined command. Clears the following registers: SCSI Data, SCSI Control, SCSI Operation except the PEP (Parity Error to Processor) bit, Identify, Destination ID except the ID field, SCSI Status except the DBR (Data Buffer Ready) bit. Generates an ''Operation Complete'' interrupt.
SCSI Bus Reset	Reset All SCSI Devices on SCSI Bus	Clears all SCSI Bus signals except the RST signal. Clears and terminates current command and any pipelined command. Generates a ''SCSI Bus Reset'' interrupt.
Assert $\overline{\text{CRST}}$ Pin	Power-Up Reset or Reset All Hardware	Sets RST bit in Disk Control register (see RST bit in Disk Control register result above). Same result as issuing a SCSI Reset command (see above) except does NOT generate an interrupt. Clears the SCSI Operation register. Clears the SCSI Interrupt and SCSI Interrupt Enable registers. The contents of the Setup 1 register becomes invalid. The contents of the Setup 2 register becomes invalid except the SPP (SCSI Parity Polarity) bit which is set to a ''1'' and the SPE bit which is reset to ''0''. Resets Timer Prescale and Timer Count registers. Clears Buffer Memory Data register. (This register will not reflect the contents of buffer memory until the Processor Pointer registers are initialized.) The RDY/MODE pin functions as an input.

3.0 Register List (Continued)

TABLE 3.2. BCLK Frequency Range

BCLK Freq. Code		BCLK Range
CLK1	CLK0	
0	0	10.00 ≤ BCLK ≤ 15.00 MHz
1	0	15.00 ≤ BCLK ≤ 17.50 MHz
0	1	17.50 ≤ BCLK ≤ 20.00 MHz
1	1	20.00 ≤ BCLK ≤ 25.00 MHz

Note: Though not in binary order, this table is correct.

SWS: SRAM Wait State

This bit allows the user to extend SRAM wait states so that slower SRAMs may be used.

0: Normal wait states. Single byte burst takes 5 cycles, 4 word bursts take 12 cycles, and 6 byte bursts take 17 BCLK cycles.

1: Extended wait states. Single byte burst is extended to 6 cycles; the 4 word burst transfer is extended to 16 cycles; and the 6 byte burst transfer is extended to 23 cycles.

DFP: Disable Fast Page

1: Fast Page DRAM access mode will be disabled. This way normal DRAMs can be used. If SRAMs are being used, then SRAM ''Fast mode'' will also be disabled; thus SRAM transfers will occur in single byte/word bursts.

0: Fast Page DRAM access mode will be enabled.

DD(1:0): DRAM Size

If DRAM is used, the DP8496/7 needs to know the organization, e.g., 64k x n, 256k x n, etc. This field controls the row and column address so that the least significant address bits are always on the column address. This will guarantee proper refresh timing during Buffer Memory transfers. The proper bit settings are shown in Table 3.3.

If SRAM is used, these bits are ''don't care''.

TABLE 3.3. Organization of DRAM

DRAM Depth		Organization of DRAM
RSZ1	RSZ0	
0	0	64k x n, (64k x 1, 64k x 4 . . .)
0	1	256k x n, (256k x 1, 256k x 4 . . .)
1	0	1M x n, (1M x 1, 1M x 4 . . .)
1	1	4M x n, (4M x 1, 4M x 4 . . .)

RSEL: DRAM/SRAM Select

The DRAM/SRAM switch. This bit changes the memory access timing to control static or dynamic RAM memory. This bit must be set before any accesses are attempted to buffer memory. If DRAM is selected, the proper refresh timing to initiate $\overline{CAS}$ before $\overline{RAS}$ automatic refreshing is provided without any further intervention through the Processor Interface.

1 DRAM

0 SRAM

RWID: RAM Data Path Width

The value of this bit is used by the SDDC to determine whether to provide addresses for a byte-wide memory configuration or word-wide configuration—when disk or SCSI data accesses are made. The state of this bit has no effect on the processor address pointer, which always increments by one.

1 Byte Mode. 8-bit wide buffer memory is being used. All address pointers will increment by one.

0 Word Mode. 16-bit wide buffer memory is being used. Disk and SCSI address pointers will increment by two.

Setup 2 (SUP2) 61h R/W

7	6	5	4	3	2	1	0
AI	BPP	BPE	SPP	SPE	SID2	SID1	SID0

The contents of this register will be invalid after assertion of the $\overline{CRST}$ pin. This register must be initialized before proper chip operation.

AI: Auto Increment

1 The Processor Pointer will increment after each access, read or write, of the Buffer Memory Data register.

0 No incrementing will occur. The Processor Pointer will remain unchanged unless it is loaded with a new value.

BPP: Buffer Parity Polarity

1 Even parity is checked or generated at the buffer memory port if enabled by the BPE bit.

0 Odd parity is checked or generated at the buffer memory port if enabled by the BPE bit.

BPE: Buffer Parity Enable

1 Enables the checking or generation of parity at the buffer memory port for various types of transfers:

When Writing to Buffer: Parity is generated for the data exiting at the buffer port; except in the case of SCSI to Buffer transfer, when, the parity is generated only if SCSI Parity is not enabled (SPE = 0). If SPE = 1, then the parity accompanying SCSI Bus Data is checked upon exiting the chip at buffer port.

When Reading from Buffer: Parity is checked at the port where the data exits the chip; except in the case of Buffer to SCSI transfer with the SCSI Parity disabled: here the parity is checked on the incoming data at the buffer port. Also, in the case of processor reads of buffer memory, the parity is checked on the incoming data at the buffer port.

0 Parity is not generated or checked at the buffer memory port. Parity may still be employed at the SCSI port by using the SPE bit of this register.

3.0 Register List (Continued)

SPP: SCSI Parity Polarity

Set to ''1'' when $\overline{CRST}$ is asserted or a SCSI Reset command is issued. This is to prevent the parity bit from driving the bus immediately after a reset while in Manual Mode. (Manual Mode described in Section 4.4.3.)

1 Even parity is checked or generated on the SCSI bus if enabled by the SPE bit.

0 Odd parity is checked or generated on the SCSI bus if enabled by the SPE bit.

SPE: SCSI Parity Enable

Set to ''0'' when $\overline{CRST}$ is asserted or a SCSI Reset command is issued. Other than in Arbitration phase, parity is always provided when writing to the SCSI bus; parity is checked if buffer parity is being used (BPE = 1), or generated if buffer parity is not being used (BPE = 0). Parity verification while reading the SCSI bus is dependent upon this bit and some other variables as desribed below.

1 SCSI Parity is verified on read data conditional to A/M bit of the SCSI Operation Register (43h) and HE bit of the Synchronous Transfer Register (44h)—as specified in Table 3.4.

0 SCSI Parity is never verified.

TABLE 3.4. SCSI Parity

Auto/ Man.	Handshake Enable	When is parity checked during a read from the SCSI bus?
1	X	Target: Parity checked when ACK asserted. Initiator: Parity checked when REQ asserted. (Re)Select: Parity Checked while SCSI ID matches, SEL true, BSY false.
0	1	Target: Parity checked when ACK asserted. Initiator: Parity checked when REQ asserted.
0	0	Parity not checked.

Refer to Table 4.10 in Section 4.2.9 for more detail on parity.

SID(2:0): SCSI ID

The unique binary address which identifies the SCSI device is loaded here. This number is usually obtained by the processor after a Chip Reset from a PROM or from switches or jumpers. This is not to be confused with the LUN (Logical Unit Number) of which each SCSI device can have eight.

This field will be internally translated to the one-of-eight form required for the SCSI bus.

Setup 3 (SUP3)				62h			R/W
7	6	5	4	3	2	1	0
CI	x	x	x	REV3	REV2	REV1	REV0

Asserting the $\overline{CRST}$ pin will not effect this register. This register must be initialized before proper interrupt operation.

CI: Combine Interrupts

1 Enabled Disk and SCSI interrupts appear on the $\overline{DINT}$ pin. The $\overline{SINT}$ pin is never asserted.

0 Enabled Disk interrupts appear on the $\overline{DINT}$ pin. Enabled SCSI interrupts appear on the $\overline{SINT}$ pin.

REV(3:0): Revision Number

These four bits reflect the functional version number of the chip. If a modification is made to the DP8496/7 that may require software or hardware modifications in a system, this value will be modified as well. The Revision Number corresponding to this data sheet is 3h.

The Revision Number will not be affected by a write to this register.

4.0 Functional Description

4.1 PROCESSOR INTERFACE

The processor port of the DP8496/7 can interface with equal ease with NSC/Intel-type and Motorola/Zilog-type processors. An 8-bit multiplexed Address/Data port is provided, and through it the processor can randomly access all SDDC registers. The chip also provides two interrupt lines to allow separate disk and SCSI interrupts.

To the processor, the DP8496/7 appears as a slave peripheral at all times and can be accessed with programmed I/O in memory mapped or I/O mapped systems. All control registers are eight bits wide. Some functions are programmed through more than one register. For example, address pointers are accessed through three different registers because the address may be up to 22 bits wide. The buffer memory data is accessed through one register called the Buffer Memory Data register (6Eh). There are two interrupt registers in the DP8496/7, the SCSI Interrupt register (49h) and the Disk Interrupt register (12h). Each of these interrupt registers has an associated mask and status register.

TABLE 4.1. DP8496/DP8497 Address Map

Address	DP8496/7 Section
00–1F	Disk Data Controller
20–3F	Disk Format Registers
40–5F	SCSI Bus Controller
60–7F	Buffer Memory, Setup & Timer

All the registers in the DP8496/7 have unique addresses and most are readable and writable. They can be accessed in single instructions with a local processor. The DP8496/7 never becomes a bus master on the processor bus.

4.1.1 Access by Different Type Processors

At the time of chip reset, the user can configure the DP8496/7 to work with NSC/Intel-type or Motorola/Zilog-type processors. While the $\overline{CRST}$ pin is asserted, the RDY/MODE pin functions as a MODE input. This input will determine the processor access mode of the DP8496/7. If this pin is left floating or pulled or driven high while $\overline{CRST}$ is asserted, the National/Intel mode will be enabled. If this pin is pulled or driven low while $\overline{CRST}$ is asserted, the Zilog/Motorola mode will be enabled.

If NSC/Intel-type mode is selected, then pins $\overline{RD}$i/$\overline{DS}$ and $\overline{WR}$i/R/$\overline{W}$ act as Read Strobe and Write Strobe and they are used to read and write data from/to the addressed register. If the Motorola/Zilog-type mode is selected, then the same pins act as Data Strobe and Read/Write; where Data Strobe is used to strobe data when reading or writing, while the direction of data is determined by the Read/Write signal. In either mode, the address is strobed into the SDDC with the ADSi signal.

4.0 Functional Description (Continued)

The Processor Interface bus timing can be asynchronous to any clock on the DP8496/7.

4.2 BUFFER MEMORY INTERFACE

All data transfers to and from the disk and to and from the SCSI bus go through buffer memory. Buffer memory is attached directly to the Buffer Memory Interface. Access by the DP8496/7 to buffer memory is performed via the internal DMA controller. All access to the buffer memory is prioritized and arbitrated through the DP8496/7. Thus, concurrent disk, SCSI, and processor accesses to the buffer memory are allowed. In normal operation no other bus masters access the buffer memory.

The user has the option of configuring the memory in a byte-wide or word-wide arrangement. If byte wide SRAM or any type of DRAM is used, no components other than the memory chips are necessary on the buffer memory interface. The only external component that may be needed is an 8-bit address demultiplexing latch if word-wide SRAM is used.

Appropriate buffer memory configuration must be specified in the setup registers, Setup 1 and Setup 2, before proper chip operation can begin. Setup 1 register is used to configure the port timing to static or dynamic modes, program the optional wait states, specify the DRAM depth and data bus width, and to enable fast page mode DRAM control. Setup 2 register is used to enable and configure buffer memory parity.

4.2.1 Static RAM

In SRAM mode there are 19 address bits available to support up to 512 kBytes in byte mode, or 1 MBytes in word mode. There is a large degree of flexibility of SRAM configurations that may be interfaced to the Buffer Memory Interface. The specific pins that are used in byte mode are described in Table 4.2. The address bits 19, 20, and 21 still exist in all pointer registers but they are not output to the interface.

TABLE 4.2. SRAM Address Pins: 8-Bit Interface

	ADB2								AB4			AB3							
Pins	7	6	5	4	3	2	1	0	2	1	0	7	6	5	4	3	2	1	0
Addr Bit	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

SRAM used word-wide can transfer a little less than twice as fast as byte-wide given equal clock rates. The address increments by two in word mode making address bit 1 of the pointers the least significant address bit presented on the physical interface. The processor will still access the RAM as individual upper and lower bytes of a single word by using address bit 0 in the Processor Pointer. A0 will determine which byte is present in the Buffer Memory Data register—low byte or high byte.

The specific pins that are used in word mode are described in Table 4.3. The ADSo pin is used to strobe an external 8-bit latch, capturing A12–19 so the ADB2 bus can be used for bi-directional data during the remainder of the memory cycle.

TABLE 4.3. SRAM Address Pins: 16-Bit Interface

	ADB2 (MUXed)								AB4			AB3							
Pins	7	6	5	4	3	2	1	0	2	1	0	7	6	5	4	3	2	1	0
Addr Bit	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

4.2.2 Dynamic RAM

Since dynamic RAMs contain their own address latches, no external latches are needed in byte- or word-wide mode. Multiplexed row and column addresses are issued on the 11 physical address lines of AB3 and AB4, with the least significant address bits being issued on the $\overline{CAS}$ strobe. A full 4 MB address range is available in byte wide mode which can be apportioned on 64k, 256k, 1M or 4M by n-bit boundaries. A full 2M address range is available in word wide mode which can be apportioned on 64k, 256k, or 1M by n-bit boundaries.

The specified pins that are used in byte mode are described in Table 4.4. The specific pins that are used in word mode are described in Table 4.5. The redundant signals on AB4 may be ignored.

TABLE 4.4. DRAM Address Pins: 8-Bit Interface

Addr. Depth	Strobe	AB4			AB3							
		2	1	0	7	6	5	4	3	2	1	0
64k	RAS	21	19	17	15	14	13	12	11	10	9	8
	CAS	10	9	8	7	6	5	4	3	2	1	0
256k	RAS	21	19	17	15	14	13	12	11	10	9	16
	CAS	10	9	8	7	6	5	4	3	2	1	0
1M	RAS	21	19	17	15	14	13	12	11	10	18	16
	CAS	10	9	8	7	6	5	4	3	2	1	0
4M	RAS	21	19	17	15	14	13	12	11	20	18	16
	CAS	10	9	8	7	6	5	4	3	2	1	0

TABLE 4.5. DRAM Address Pins: 16-Bit Interface

Addr. Depth	Strobe	AB4			AB3							
		2	1	0	7	6	5	4	3	2	1	0
64k	RAS	X	20	18	16	15	14	13	12	11	10	9
	CAS	11	10	9	8	7	6	5	4	3	2	1
256k	RAS	X	20	18	16	15	14	13	12	11	10	17
	CAS	11	10	9	8	7	6	5	4	3	2	1
1M	RAS	X	20	18	16	15	14	13	12	11	19	17
	CAS	11	10	9	8	7	6	5	4	3	2	1
2M	RAS	X	20	18	16	15	14	13	12	21	19	17
	CAS	11	10	9	8	7	6	5	4	3	2	1

Note: The address signal x will always be equal to V_{DD}.

4.0 Functional Description (Continued)

The maximum capacitive loading on the Buffer Memory Interface should be taken into account when designing the buffer memory interface. The address/data pins are designed to directly drive up to 12 DRAM chips with board traces of reasonable length. By using 256k x 1 chips, 256k can easily be addressed in 8-bit mode. By using higher density DRAMs, like 1M x 4 and 1M x 1, the full addressing range of 4M in byte-wide mode, and 2M in word-wide mode, can be utilized with chip counts up to or less than 12.

No refresh address counters are provided on-chip. This necessitates the use of DRAMs with on-board counters and the ability to do a ''CAS before RAS'' type refresh.

4.2.3 Data Transfer Timing

Two basic transfer modes are possible, normal mode and the fast page mode. If the fast page mode is disabled, by setting the DFP bit to 1 in Setup 1 register (60h), then a single memory transfer will occur every 5 BCLK cycles for DRAM and every 5 or 6 cycles for SRAM, depending on the SWS bit of the SUP1 register.

If the fast page mode is selected, by setting the DFP bit to 0, then the DP8496/7 will transfer up to 6 bytes (or 4 words in word-mode) of data at a time in a ''burst'', as long as only the least significant 8 bits of address change. In the case of DRAMs then, the DP8496/7 only has to change the address and toggle the $\overline{CAS}$ line for each byte or word of the burst transferred after the first one. The number of bytes or words transferred in a ''burst'' are shown in Table 4.6 and depend on number of bytes or words present in the on-chip FIFOs, the address pointers' relation to the page boundary, and refresh, etc. Operation in this mode yields highest attainable buffer memory bandwidths, but it requires the use of fast page mode type DRAMs.

TABLE 4.6. Fast Page Mode Transfer Periods

Number of Transfers	DRAM Transfer Period	SRAM Transfer Period
6 Bytes	12*BCLK Period	17*BCLK Pd (SWS = 0) 23*BCLK Pd (SWS = 1)
4 Bytes	9*BCLK Period	
1 Byte	5*BCLK Period	
4 Words	9*BCLK Period	12*BCLK Pd (SWS = 0) 16*BCLK Pd (SWS = 1)
3 Words	8*BCLK Period	
2 Words	6*BCLK Period	
1 Word	5*BCLK Period	

Note: SWS is bit 5 in SUP1 register (60h)

The DP8496/7 issues a DRAM refresh cycle every 128 bys clock periods. Of these, 5 bus clock periods are used by the refresh cycle to complete, so 123 bus clock periods are left for actual data transfers. Thus, the minimum specified bus clock rate of 10 MHz will guarantee a refresh cycle time of less than 12.8 μs per row.

The DP8496/7 automatically manages concurrent buffer accesses by disk, SCSI and processor, while also executing DRAM refresh cycles within the requisite time. Small FIFO's have been integrated into the DP8496/7 to allow for buffer memory latency due to contention. The Disk Data Controller has a 14 byte (or 7 word) FIFO and the SCSI Bus Controller has a 32 byte (or 16 word) FIFO. The word-wide FIFOs are used if the 16-bit mode is used for buffer memory.

The on-chip arbitration circuitry uses the prioritization scheme shown in Table 4.7 to smoothly manage the data flow and insure that the FIFOs don't overflow or underflow.

TABLE 4.7. Buffer Memory Transfer Priority

Rank	Type of Transfer
1	DRAM Refresh
2	Disk Data Burst
3	SCSI Data Burst
4	Processor Access
5	Disk Data—Single Byte/Word
6	SCSI Data—Single Byte/Word
7	Idle

The DP8496/7 arbiter does not waste any cycles between consecutive transfers, even if those transfers used different pointers. Pointers are seamlessly switched so that all memory transfers use consecutive BCLK cycles.

Keeping in mind the above priority scheme, some design trade-offs must be made between the available bandwidth of the buffer memory port and rates of disk, SCSI and processor accesses that are to be supported. A sound design would assure that enough bandwidth is left over after accommodating simultaneous full-speed disk and SCSI transfers so that occasional processor access can be serviced within reasonable time. Section 4.2.6 describes how processor accesses of buffer memory are handled.

Detailed timing specifications for the buffer memory port can be found in Chapter 6 of this document. For a given bus clock frequency, calculations for gross available bandwidth can be carried out by using Table 4.6. For example, with bus clock frequency set at 20 MHz, and with 6 bytes of fast-page mode transfers taking 12 cycles, bandwidth of 10 MByte/sec is calculated for a byte-wide operation. However, after this number is adjusted for the DRAM refresh cycle overhead, maximum bandwidth is reduced to 9.6 MByte/sec. There are other practical factors that would further reduce this number to the attainable, sustained bandwidth for a particular transfer. For the case of BCLK = 20 MHz, this value turns out to be 9.3 MHz.

As a calculation based on the timing specifications in Chapter 6 would show, 100 ns DRAMs would work with the DP8496/7 operating with bus clock of 20 MHz. By using faster BCLK and DRAMs the user can achieve sustained byte-wide bandwidths of above 11 MByte/sec.

TABLE 4.8. DRAM Bandwidth Capabilities

Desired DRAM	Fastest Usable BCLK	Attainable Bandwidth	
		Byte-Wide	Word-Wide
100 ns	18 MHz	8.4 MHz	15.3 MHz
100 ns	20 MHz	9.3 MHz	17.0 MHz
80 ns	22 MHz	10.2 MHz	18.7 MHz
80 ns	24 MHz	11.2 MHz	20.4 MHz

The maximum BCLK frequency is specified in Chapter 7.

For maximum performance, however, word-wide memory architecture can be employed; and then, with 100 ns DRAMs and 20 MHz BCLK, sustained bandwidth of 17 MB/s can be attained.

4.0 Functional Description (Continued)

4.2.4 Pointers

Three 22-bit pointers represent the three possible addresses into the buffer memory. These addresses are used by the Disk Data Controller, SCSI Bus Controller, and the Processor Interface. The Disk and SCSI pointers are double buffered in what are called the Active and Holding register sets. Both of these Active pointers automatically increment on every transfer to buffer memory. The Processor pointer is only single buffered. It will increment automatically if enabled in the Setup 2 register.

4.2.5 Buffer Management

The processor will normally initialize the Holding registers with values before an operation is started. When the operation begins the DP8496/7 will copy the contents of the Holding registers into the Active registers (which are actually counters). This double buffering allows the processor to load new values into the Holding registers while the previous operation is still continuing. This feature allows for versatile management of contiguous or non-contiguous buffers.

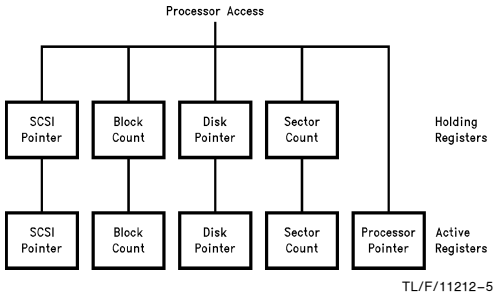


FIGURE 4.1. Holding and Active Registers

The processor is allowed read/write access to the Holding registers. It is only allowed to read the Active registers.

Normally the processor will only access the Holding registers. The Active registers are really the block counters and address pointers actually used to transfer the data. Thus only in unusual situations will the Active registers need to be read. The A/H (Active/Holding) bit in the SCSI Operation register determines which register set the processor will access.

The Processor Pointer does not have a Holding register. The Processor Pointer Active register will be accessed independent of the A/H bit.

The Disk, SCSI, and Processor Pointer registers contents have no relationship to one another, thus one pointer is not automatically kept a certain distance from the other.

Instead, the monitoring by the processor of block and sector transfers is facilitated through the use of interrupts. [Polling could also be used by reading the SCSI Interrupt (49h) and Disk Interrupt (12h) registers.]

Interrupts are programmable at three different levels (see *Figure 4.2*):

1. At the end of each block transferred.
2. After a certain number of blocks have been transferred. This is called a ''group''.
3. After all programmed blocks have been transferred, or ''command completion''.

(See the descriptions of DINTE (13h) and SINTE (4Ah) registers in Sections 4.3 and 4.4, respectively)

This provides a point of synchronization for the processor to safely monitor the progress of both disk and SCSI transfers and thus prevent overflow or underflow of the buffer memory. Overflow and underflow are analogous to pointers overtaking one another. The processor is responsible for preventing this.

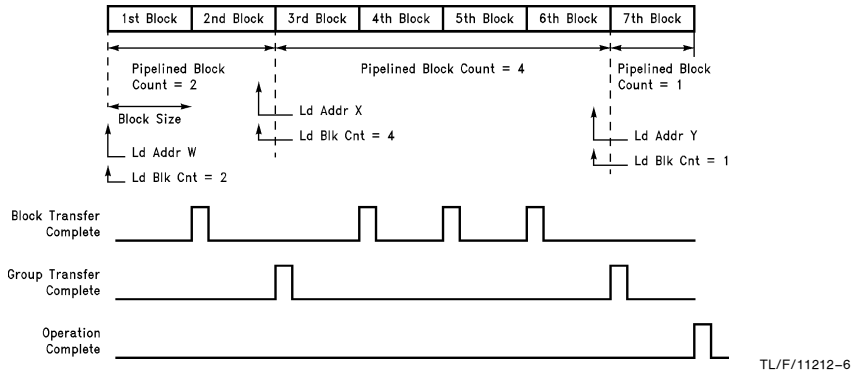


FIGURE 4.2. Flexible Interrupt Levels

4.0 Functional Description (Continued)

4.2.6 Processor Access to Buffer Memory

The processor gains access to the buffer memory by loading the Processor Pointer 0–2 registers with the address of the byte in buffer memory that needs to be accessed. The Processor Pointer 0 register should be loaded last. The data is then read from or written to the Buffer Memory Data register.

The Buffer Memory Data register can be read or written to just like any other register on the DP8496/7. The same AC specifications apply to this register. However, the data read from the Buffer Memory Data register may not be valid immediately and the data written to the Buffer Memory Data register may not be transferred to the buffer memory immediately.

The Buffer Memory Data register is simply another DMA channel trying to access the buffer memory. Since the arbitration priority of the Processor Interface is not the highest, there may be a delay before the contents of the Buffer Memory Data register are actually transferred to or from the buffer memory. The amount of the delay depends on how much extra bandwidth was allowed by the system designer in the buffer memory data path.

The Processor Pointer registers should be loaded by the processor from most-significant byte to least-significant byte. Immediately after the Processor Pointer 0 register is written to, the DP8496/7 will initiate an arbitration for a read from buffer memory at the location pointed to by the Processor Pointer registers. Once granted, the buffer memory location is read and the value is placed in the Buffer Memory Data register.

In the case of a write operation, the Buffer Memory Data register may be written to immediately after loading the Processor Pointer 0 register. If the processor writes to the buffer memory data register before the DP8496/7 completes a buffer memory read, no buffer memory read will occur.

To determine when the transfer has completed, one of the three techniques described in Section 4.2.8 must be used. Same techniques should be used if sequential reads or writes are issued.

4.2.7 16-Bit Wide Access

If the 16-bit buffer memory mode is enabled in the Setup 1 register, byte wide processor access to DP8496/7 does not change. The processor should load the Processor Pointer registers from most-significant byte to the least-significant byte. Immediately after the Processor Pointer 0 register is written to, the DP8496/7 will initiate an arbitration for a read from buffer memory at the location pointed to by the Processor Pointer registers.

This will result in the entire 16-bit word being read from buffer memory. However, the Buffer Memory Data register will only allow access to the 8 bits that were requested based on the least significant bit in the Processor Pointer 0 register. The least significant bit in the Processor Pointer 0 register determines whether the low byte (0) or high byte (1) can be accessed by the processor. By using an even processor pointer address and setting AI (Auto Increment) to ''1'', the low byte and high bytes of each word will be accessed in order.

Byte wide data written to the Buffer Memory Data register by the processor replaces the low byte or high byte depending on the least significant bit in the Processor Pointer 0 register. After the other byte has been obtained by reading the addressed word location, the 16-bit word is written back to buffer memory. This ensures that only one byte is effected with each write.

If the AI (Auto Increment) mode is enabled in the Setup 2 register, the Processor Pointer will increment by one after each read of the Buffer Memory Data register by the processor or after each transfer from the Buffer Memory Data register to the buffer memory. To complete a 16-bit processor read the processor must read the Buffer Memory Data register again. When writing, the processor must write to the Buffer Memory Data register again to change the other byte of the memory word. After this second byte is altered, the 16-bit word is again written back to buffer memory. This sequence completes a processor modification of a 16-bit buffer memory location.

If Auto Increment mode is enabled and the processor writes to the Buffer Memory Data register twice—loading the low byte and the high byte— before the DP8496/7 initiates a buffer memory read, no buffer memory read will occur. This new 16-bit value will then be written to buffer memory twice. The data is transferred twice to buffer memory to increment the processor pointer to the next word.

4.2.8 Sequential Accesses of Buffer Memory

To make sequential reads and writes by the processor more convenient, the Processor Pointer registers may be programmed to automatically increment after each access to the Buffer Memory Data register. The AI (Auto Increment) bit in the Setup 2 register determines whether the Processor Pointer will automatically increment. The Processor Pointer will increment after each read or write of the Buffer Memory Data register, then the DP8496/7 will read the next RAM location from buffer memory. Sequential writes are accomplished in the same way. Mixed reads and writes are also allowed.

There are three methods to insure safe transfer of data to/from buffer memory by the processor: Using the RDY pin, software polling, or using the Interrupt Error.

The method chosen will largely be based upon the processor facilities available. If the processor has a Ready, DTACK or some other asynchronous access control pin, the RDY output pin of the DP8496/7 can be used to automatically control access to Buffer Memory Data register. If a processor is chosen without such a facility, the software polling operation is the next best choice.

The Disk Interrupt Enable register is used to select between the different processor access techniques. Only bit 6 of this register is used for this purpose. The other bits of this register will be described in the Disk Data Controller description in Section 4.3.2. A summary of the RDY pin function and the buffer memory interrupt function is shown in Table 4.9.

Disk Interrupt Enable **13h** **R/W**

7	6	5	4	3	2	1	0
TI	BMD	I/S	GCI	SCI	HNC	HC	CCI

Ready Pin

The RDY pin may be connected to the processor's memory access control pin (or ''wait state'' pin). After $\overline{WR}$ or $\overline{RD}$ goes low, the processor will be ''wait-stated'' until the data from buffer memory becomes available. This is the easiest method of access to buffer memory since the job of regulating the speed of the processor access is done entirely in hardware by the DP8496/7. The only requirement is that the processor used with the DP8496/7 have some type of asynchronous access line.

When the processor reads the Buffer Memory Data register, the RDY pin will become deasserted if the data is not yet available. It will become asserted when the data is valid. For a write operation, the RDY pin will become deasserted if a previous write has not yet been transferred to buffer memory.

To enable this mode, the BMD (Buffer Memory Data) bit in the Disk Interrupt Enable register must be set to zero. The RDY pin will be always deasserted while in the Zilog/Motorola mode.

If non-contiguous address locations are to be accessed, the processor must poll the E/R (Error/Ready) bit in the Disk Interrupt register for a ''1'' before the Processor Pointer registers may be modified.

Software Polling

If no asynchronous access pin is available, or you choose not to use it, software polling may be used. With the BMD (Buffer Memory Data) bit in the Disk Interrupt Enable register set to ''0'', not only is the Ready pin enabled, but the R/E (Ready/Error) bit in the Disk Interrupt register will operate as ''Buffer Memory Data Ready'' and may be polled. A ''1'' in the E/R bit indicates that the Buffer Memory Data register is ready for either reading or writing, a ''0'' means that it is not ready.

A ''1'' in the E/R bit also indicates that the Processor Pointer registers may be modified by the Processor.

Disk Interrupt — 12h — R/W

7	6	5	4	3	2	1	0
TI	E/R	I/S	GCI	SCI	HNC	CCI	

Interrupt Error

By setting the BMD (Buffer Memory Data) bit in the Disk Interrupt Enable register to a ''1'', the RDY pin is disabled. Also, the E/R bit in Disk Interrupt register will operate as a true interrupt. This ''Error'' interrupt indicates that the Buffer Memory Data register has been accessed before the data is valid. The processor should simply wait a certain amount of time between each access to the Buffer Memory register. If an interrupt occurs, simply clear the interrupt, wait an additional amount of time, and try the access again.

Interrupt after a write: Indicates that the previous write has not yet been transferred to the buffer memory. The previous data is still retained and the DMA process is unaffected by the new write. The data just written will be ignored.

Interrupt after a Read: Indicates that the Buffer Memory Data register has not yet been bloaded with the data requested. The data just read is invalid.

TABLE 4.9. RDY Pin and Interrupt Summary

BMD Bit in Disk Interrupt Enable Register	RDY Pin	Function of E/R Bit in Disk Interrupt Register
1	Disabled	Buffer Memory Error
0	Enabled	Buffer Memory Ready

If non-contiguous address locations are to be accessed, the processor must verify that the Processor Pointer registers can be modified. To do this, the BMD bit in the Disk Interrupt Enable register must be set to ''0''. This enables the software polling mode. While in this mode, the Processor must poll the E/R bit in the Disk Interrupt register. When this bit is ''1'', the Processor Pointer registers may be modified. The Interrupt Error mode may then be re-enabled by setting the BMD bit in the Disk Interrupt Enable register back to ''1''.

4.2.9 Parity

Buffer memory parity is generated when writing to buffer memory and verified while reading from buffer memory only if enabled by the BPE (Buffer Parity Enable) bit in the Setup 2 register.

In general, parity (if enabled) is always checked at the point where the data exits the chip. Therefore, parity errors generated within the DP8496/7 are checked as well as external parity errors.

When the processor accesses data from buffer memory, parity information cannot be determined through the Buffer Memory Data register. If the Setup 2 register is configured to check buffer memory parity, a parity error detected while reading the Buffer Memory Data register will be reported in the SCSI Operation register (43h). The SCSI Operation register may be polled after a block of data has been read from the Buffer Memory Data register. Once a parity error is detected, it is cleared only by reading the SCSI Operation register.

4.0 Functional Description (Continued)

TABLE 4.10. Operation of Parity Generation and Checking

Control Bits (SUP2)				Direction of Transfer	Action on Port			Error Report
SPEN	DBPEN	SPP	DBPP		Disk	Buffer	SCSI	
0	1	x	x	SCSI to Buffer	—	Generate	—	—
1	x	0	0	SCSI to Buffer	—	Check	Check	1
1	x	1	1	SCSI to Buffer	—	Check	Check	1
1	x	0	1	SCSI to Buffer	—	Inv. + Check	Check	1
1	x	1	0	SCSI to Buffer	—	Inv. + Check	Check	1
x	0	x	x	Buffer to SCSI	—	—	Generate	—
0	1	0	0	Buffer to SCSI	—	Check	—	2
0	1	1	1	Buffer to SCSI	—	Check	—	2
0	1	0	1	Buffer to SCSI	—	Inv. + Check	—	2
0	1	1	0	Buffer to SCSI	—	Inv. + Check	—	2
1	1	0	0	Buffer to SCSI	—	—	Check	2
1	1	1	1	Buffer to SCSI	—	—	Check	2
1	1	0	1	Buffer to SCSI	—	Invert	Check	2
1	1	1	0	Buffer to SCSI	—	Invert	Check	2
x	0	x	x	Not between SCSI & Buffer	—	—	—	—
x	1	x	x	Disk to Buffer	—	Generate	—	—
x	1	x	0	Buffer to Disk	Check	Invert	—	3
x	1	x	1	Buffer to Disk	Check	—	—	3
x	1	x	x	Processor to Buffer	—	Generate	—	—
x	1	x	x	Buffer to Processor	—	Check	—	4

Error Reporting:

1. SCSI Bus Error in CIC field of SINT register (49h) and SCSI Parity Error in SSTAT register (48h).
2. Buffer Parity Error in CIC field of SINT register and Buffer Memory Error in SSTAT register.
3. Error in DINT register (12h) and Disk Parity error in EC field of DSTAT register (14h).
4. PEP bit in the SOP register (43h).

4.0 Functional Description (Continued)

4.2.10 Register Description

Disk Pointer—MSB (DP2) 65h R/W

7	6	5	4	3	2	1	0
x	x	Address					

Disk Pointer (DP1) 66h R/W

7	6	5	4	3	2	1	0
Address							

Disk Pointer—LSB (DP0) 67h R/W

7	6	5	4	3	2	1	0
Address							

The 22-bit address value programmed in these registers is used as pointer into the buffer memory for all data transfers between the disk port and buffer. The DP8496/7 uses these registers as ''holding registers'' in that the contents of these registers are loaded into a set of ''active registers'' upon start of operation. Thus the holding registers may be accessed even while an operation is continuing. The value in the active registers is incremented each time an access to buffer memory is made by the disk controller. The active register set may be read only by the processor by using the same register address, but by first setting the A/H bit in the SCSI Operation register.

SPSI Pointer—MSB (SP2) 68h R/W

7	6	5	4	3	2	1	0
x	x	Address					

SCSI Pointer (SP1) 69h R/W

7	6	5	4	3	2	1	0
Address							

SCSI Pointer—LSB (SP0) 6Ah R/W

7	6	5	4	3	2	1	0
Address							

The 22-bit address value programmed in these registers is used as pointer into the buffer memory for all data transfers between the SCSI port and buffer. The DP8496/7 uses these registers as ''holding registers'' in that the contents of these registers are loaded into a set of ''active registers'' upon start of operation. Thus the holding registers may be accessed even while an operation is continuing. The value in the active registers is incremented each time an access to buffer memory is made by the disk controller. The active register set may be read only by the processor by using the same register address, but by first setting the A/H bit in the SCSI Operation register.

Processor Pointer—MSB (PP2) 6Bh R/W

7	6	5	4	3	2	1	0
x	x	Address					

Processor Pointer (PP1) 6Ch R/W

7	6	5	4	3	2	1	0
Address							

Processor Pointer—LSB (PP0) 6Dh R/W

7	6	5	4	3	2	1	0
Address							

The 22-bit address value programmed in these registers is used as pointer into the buffer memory for all data transfers between processor and buffer. The value in these registers is incremented automatically each time an access to buffer memory is completed—if the AI bit is Setup 2 register is set. The user should write into the PP0 register last—as that initiates a read operation. In case of a write operation, the BMD register should next be written.

Buffer Memory Data (BMD) 6Eh R/W

7	6	5	4	3	2	1	0
Data							

This register provides the processor a window into the buffer memory for reading or writing at the address pointed to by the PP2, PP1, and PP0 registers. While the writing of pointer LSB into the PP0 register initiates a read operation, the act of writing to this register initiates a write operation—terminating the read operation in effect.

For 16-bit wide memory operations, the processor has to read or write this register twice to access both the low-order and high-order bytes. Bit 0 and PP0 register is used to determine if the byte in BMD register is low-order (PP0:0 = 0) or high-order (PP0:0 = 1). However, the processor does not have to wait for one byte to be transferred before accessing the other byte of the same word since the DP8496/7 will read or write from the memory the full word.

4.0 Functional Description (Continued)

4.3 DISK DATA CONTROLLER

The Disk Data Controller is concerned only with the data aspects of the disk drive. Control signals for head movement, head selection, track zero detection, etc. are left to the processor or through an external I/O port. Because of this, the DP8496/7 can easily be used in a wide variety of SCSI disk systems.

Track format is specified through the internal pattern and count registers loaded by the host processor at initialization. A unique format can be developed for each application taking into account speed tolerances, PLL lock-on time and write splice length, etc.

An additional field before the Header and Data Preambles (Header Gap and Data Gap) controls RGATE vs. WGATE assertion times. This assures that the PLL never sees a write splice between the header and data field. Byte count programmability of these fields ensures that minimum track area for the write splice.

The data field length (sector size) can range from 16 kBytes to 64 kBytes with a two byte resolution.

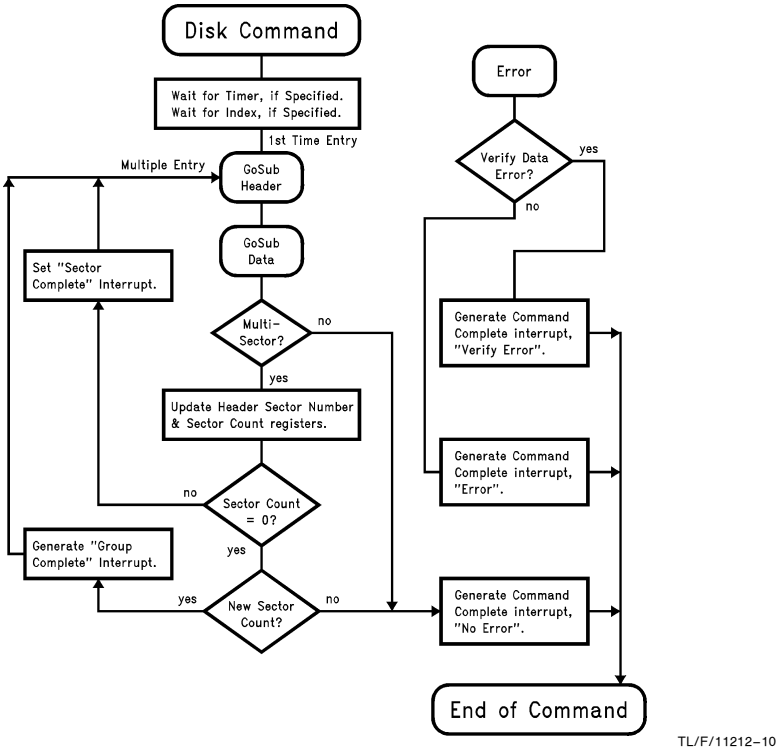
Finally, both the ID and Data portions of the sector can be protected with CRC or ECC ranging in length from 16 bits (CRC) to 32, 48 or 56 bits (ECC). All the polynomials are fixed computer generated codes except the 16-bit CRC which is a CCITT standard. Very little processor overhead is needed for disk data error correction due to on-chip logic.

Additional features include a Sector or Index pulse interrupt option. This makes processor monitoring of disk rotational speed easier and also location of sectors based on time since last index pulse for soft sectored drives or count of sector interrupts for hard sectored drives. Also programmable control over AME and AMF/Sector pins supports pseudo-hard sectored (ESDI soft sectored) as well as hard and soft sectored drives.

All registers are buffered such that individual bits will not change while $\overline{RD}$ strobe is active except where specified.

4.3.1 Command Operation

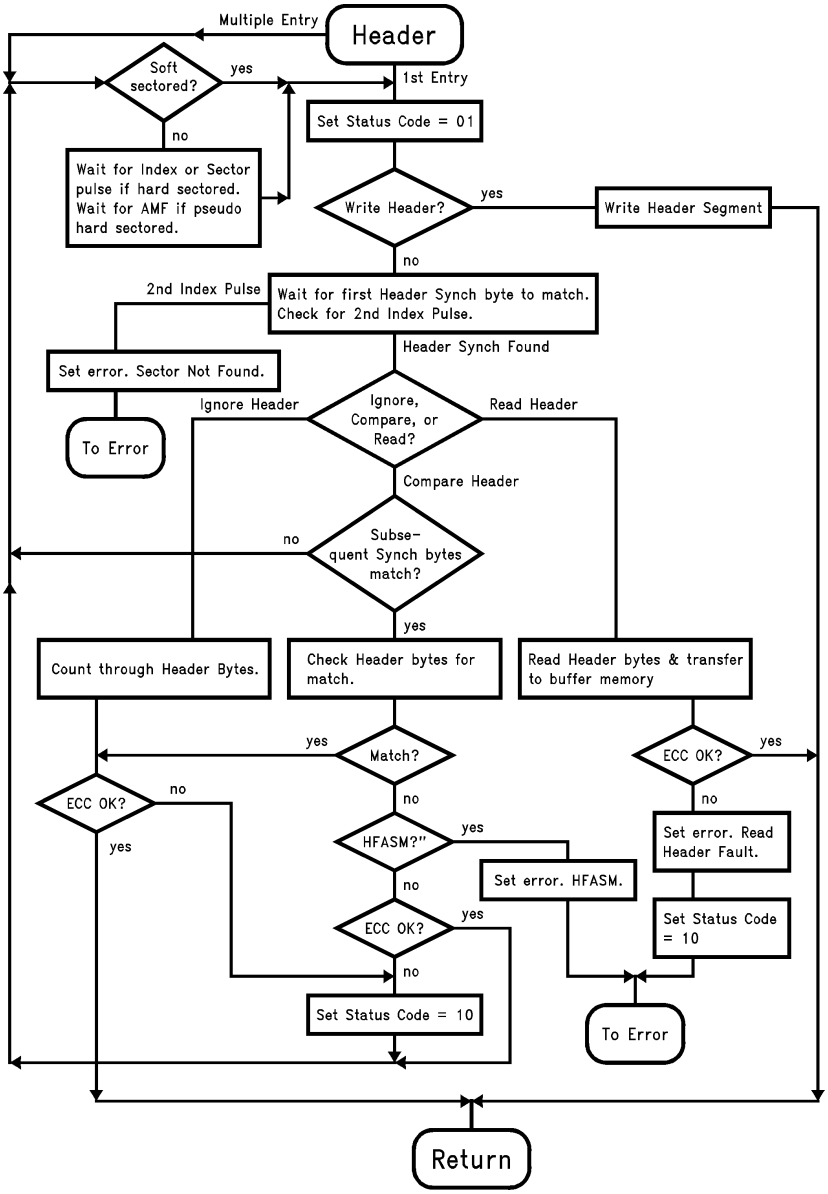
User issues disk commands by writing to the Disk Command register (DCMD, 10h) after format registers and other disk control registers have been specified. Most disk commands follow the flowcharts shown in *Figures 4.3*, *4.4*, and *4.5*. The command will begin after the starting condition specified by the SCT (Start Control) bit in the Disk Command register and the DCOZ (Drive Command On Zero) bit in the Timer Prescale register. Each command is really broken into two sub-commands: a Header Segment command and a Data Segment command. If a single sector command is issued (determined by the MSO (Multi-Sector Operation) bit in the Drive Control register) the command will terminate at the end of the specified sector. Multi-sector commands will update the Sector Number register and the Sector Count register after processing each sector and will continue until the Sector Count register reaches zero.



TL/F/11212–10

FIGURE 4.3. Disk Command Flowchart

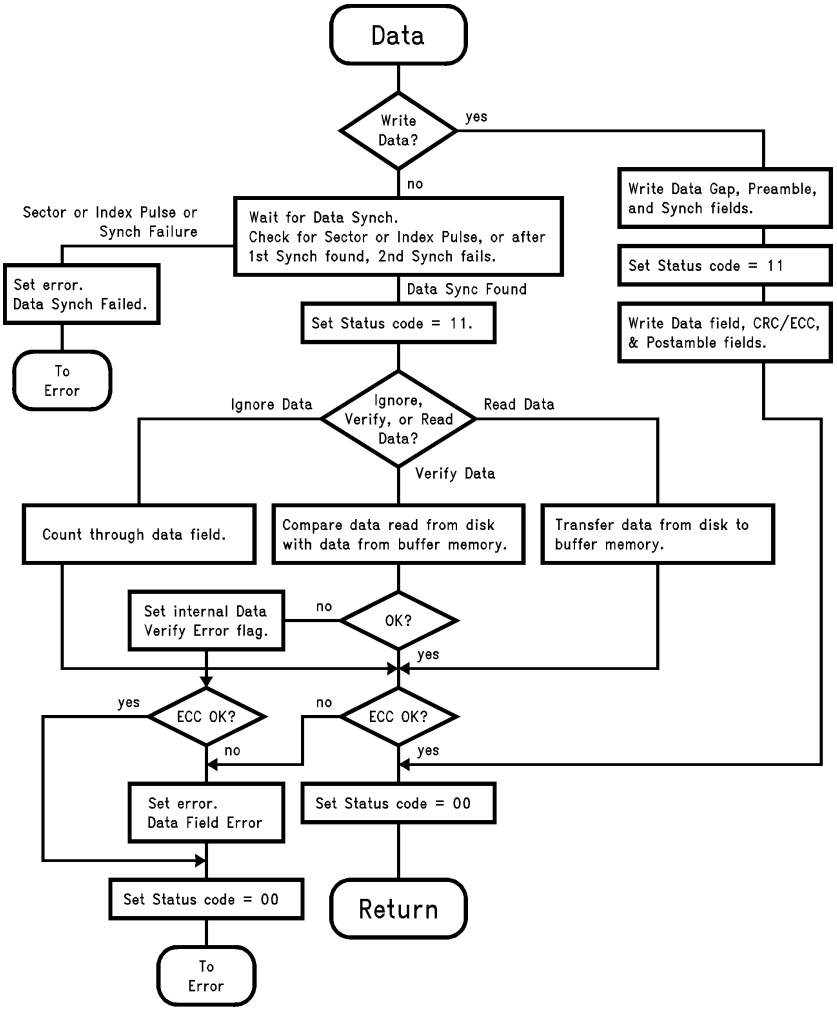
4.0 Functional Description (Continued)



TL/F/11212–11

Note: HFASM = Header Failed Although Sector Matched as enabled in the Header Byte Control registers.

FIGURE 4.4. Header Segment Flowchart



TL/F/11212–12

FIGURE 4.5. Data Segment Flowchart

4.3.2 Disk Command and Control Registers

Disk Command(DCMD)			10h				W Only
7	6	5	4	3	2	1	0
MSO	SCT	X	Disk Command Opcode				

Writing to this register will initiate a disk command. Normally, other disk format and control registers are configured before writing to this register, as described in *Figure 4.6* in the section describing Disck Operations (4.3.4). After all other registers are loaded, writing to this register will initiate the operation.

Disk commands may be pipelined. Just before a new command begins, the Disk Command register is downloaded internally. A new command may then be written to the Disk Command register. See Section 4.3.4 for additional comments on pipelining.

MSO: Multi-Sector Operation

1 Multi-sector operation the Sector Number register and the Sector Count register.

0 Single sector operation.

SCT: Start Control

The Disk Command will not start until the conditions specified in Table 4.11 are satisfied. Note, however, that the DCOZ bit in the TIMER PRESCALE register overrides the SCT bit. When the DCOZ bit is set the command will start when the time counts down to zero—regardless of the state of SCT bit.

TABLE 4.11. Start Control

SCT	Drive Type	Starting Condition
1	Soft	Immediate
	Pseudo-Hard	After AMF
	Hard	After Index or Sector
0	Soft	After Index
	Pseudo-Hard	After AMF after First Index
	Hard	After Index

Disk Command Opcode

This field contains the opcode of the actual command that the Disk Data Controller will perform.

Table 4.12 shows a list of valid command opcodes. No other opcodes may be used.

No Operation

NOP corresponds to a sector operation. When the operation begins (depending upon the specified starting condition) the disk sequencer will blindly count through the Post Index Gap, Data CRC/ECC, and Data Postamble fields and then terminate with a Disk Operation Complete interrupt. RGATE and WGATE are never asserted but signals associated with the specified starting condition must be present in addition to RCLK. In soft sectored mode, the NOP will begin immediately if the SCT (Start Control) bit is set to one.

Multi-sector NOPs are permitted in which case the Sector Count active register will be decremented and the Sector Number register incremented for each sector operation.

TABLE 4.12. Disk Command List

Disk Comand Opcode (Hex)	Operation
0	No Operation
4	Ignore Header/Ignore Data
5	Ignore Header/Verify Data
6	Ignore Header/Write Data
7	Ignore Header/Read Data
8	Compare Header/Ignore Data(*)
9	Compare Header/Verify Data
A	Compare Header/Write Data
B	Compare Header/Read Data
0C	Write Header/Ignore Data
0D	Write Header/Write Data
10	Format Type A
11	Format Type B
12	Format Type C
13	Format Type D
14	Read Header/Ignore Data
15	Read Header/Read Data
18	Write Unformatted
19	Read Unformatted
1C	Start Correction Cycle

*The Compare Header/Ignore Data command transfers all the headers to buffer memory.

Ignore Header/Ignore Data

This command will optionally check the Header CRC/ECC or Data CRC/ECC fields, or both without comparing the header or reading the data. The CRC/ECC verification is enabled in the ECC Control register. This command will search for a Header Synch field, count through the other header fields, and optionally check the Header CRC/ECC. It will then search for the Data Synch field, count through the data field, and optionally check the Data CRC/ECC.

Ignore Header/Verify Data

This command will search for a Header Synch field, count through the other header fields, and optionally check the Header CRC/ECC. It will then perform a byte for byte comparison with the data field and buffer memory and optionally check the Data CRC/ECC.

Ignore Header/Write Data

This command will search for a Header Synch field, count through the other header fields, and optionally check the Header CRC/ECC. It will then write data from the buffer memory to the data field.

Ignore Header/Read Data

This command will search for a Header Synch field, count through the other header fields, and optionally check the Header CRC/ECC. It will then read the data field, transferring it to the buffer memory, and optionally check the Data CRC/ECC.

4.0 Functional Description (Continued)

Compare Header/Ignore Data

This command is unique. As the Header Segments read from the disk are being scanned for a comparison to the Header Byte Control and Pattern registers, the header bytes are all transferred to buffer memory. The header bytes from each sector read will be placed in the buffer memory after the previous sector's header bytes. If the number of header bytes from a sector is odd, an extra dummy byte will be transferred to buffer memory at the end of each sector.

Only the Header Bytes are transferred. No Synch or CRC/ECC data will be transferred.

This command is normally used in the single sector mode (MSO = 0). This command will terminate under the same conditions as any other single sector command: a header match is found, an error occurs, or 2 index pulses are received without a header match.

Compare Header/Verify Data

Headers are compared as defined in the Header Byte Control and Pattern registers. Header CRC/ECC is optionally checked. When a matching header is found, a byte for byte comparison between the data field and buffer memory will be performed. Data CRC/ECC will be optionally checked. Buffer memory parity is not checked during this operation.

Compare Header/Write Data

This is the ''normal'' write data command. Headers are compared as defined in the Header Byte Control and Pattern registers. Header CRC/ECC is optionally checked. When a matching header is found, data will be written from the buffer memory to the data field.

Compare Header/Read Data

This is the ''normal'' read data command. Headers are compared as defined in the Header Byte Control and Pattern registers. Header CRC/ECC is optionally checked. When a matching header is found, the data field will be read from the disk and transferred to buffer memory. Data CRC/ECC will be optionally checked.

Write Header/Ignore Data

This operation will work properly only with hard sectored drives. The Post Sector/Index Gap will be counted through or written (based on the WPSIG (Write Post/Sector Index Gap) bit in the Disk Control register). The Header Gap, Header Preamble, Synch fields, Header fields, CRC/ECC, and Header Postamble will all be written. The data field CRC/ECC will be optionally checked.

Write Header/Write Data

This operation will work properly only with hard sectored drives. All Header fields and Data fields will be written to the disk if they are enabled *EXCEPT Gap3* . Gap3 will not be written with this operation.

Format Type A

Internal header bytes, internal data pattern.

This is the most simple type of format. All fields in the header and data segments are generated from the internal Disk Data Controller registers. This mode is usually used for an interleave of one. The multi-sector operation capability of the DP8496/7 increments the Sector Number register which is written in each sector.

The ECC/CRC for the ID Segment and the Data Segment will be automatically generated during the format operation. This is true for all four types of the Format command (A, B, C, and D).

Format Type B

Internal header bytes, data from buffer memory

All header fields are generated from the internal Disk Data Controller registers. The data field of the Data Segment is obtained from buffer memory. This allows formatting with real data in the data field. This mode is usually used for an interleave of one. The multi-sector operation capability of the DP8496/7 increments the Sector Number register which is written in each sector.

Format Type C

Header from buffer memory, internal data pattern.

The header bytes in the Header Segment are obtained from buffer memory. All data fields are from the internal registers. The header bytes should be arranged contiguously in buffer memory. If the number of header bytes in each sector is odd, an extra dummy byte must be appended to the end of each sector's header in buffer memory. This approach is ideal for sector interleaving of greater than one and offers the minimum of processor intervention during the formatting process.

Format Type D

Header from buffer memory, data from buffer memory.

The header bytes in the Header Segment and the data field in the Data Segment are obtained from buffer memory. The buffer memory should be arranged with the header bytes for one sector followed by its data field. This should be repeated for each sector to be formatted. For the case where word-wide memory configuration is employed, if the number of header bytes in each sector is odd, an extra dummy byte must be appended to the end of each sector's header in buffer memory.

Read Header/Ignore Data

This is a ''normal'' read header command. After the first Header Sync is found, the Header Bytes are transferred to buffer memory. If the number of Header Bytes read is odd, an extra dummy byte will be transferred to buffer memory after each sector. Header CRC/ECC is optionally checked. The data field CRC/ECC may also be optionally checked.

Read Header/Read Data

Header and data bytes are transferred to buffer memory. After the first Header Synch is found, the Header Bytes are transferred to buffer memory. If the number of Header Bytes read is odd, an extra dummy byte will be transferred to buffer memory after each sector. Header CRC/ECC is optionally checked.

Write Unformatted

This command will write a Data Segment only (although the Data Segment registers may be programmed to appear as a Header Segment). No header fields will be read, counted through, or written. After the specified starting condition, the Post Sector/Index Gap (if enabled), Data Preamble, Data Synch, Data Field, CRC/ECC (if enabled) and Data Postamble are written to the disk.

This command may be used for many special applications such as:

1. Writing data fields that are split by servo information.
2. Writing to non-hard sectored Header Segments.
3. Headerless sectoring.

This command could be useful when used with the LOI (Load On Index) bit in the Timer Prescaler register. Write Unformatted can be ''dropped'' accurately on the track using the time from Index as a count.

Read Unformatted

This command will read a Data Segment only (although the Data Segment registers may be programmed to appear as a Header Segment). No header fields will be read, counted through, or written. After the specified starting condition, a Data Synch will be searched for. After Data Synch is found, the data field will be read and transferred to buffer memory. Data CRC/ECC will be optionally checked.

This command could be useful when used with the LOI (Load On Index) bit in the Timer Prescale register. Read Unformatted can be ''dropped'' accurately on the track using the time from Index as a count. It is also useful for reading split data fields due to embedded servo.

Start Correction Cycle

This command will start the internal ECC correction cycle. When the correction cycle has completed, A Disk Completion Interrupt will be generated. If the correction cycle was successful, no error will be indicated. If unsuccessful, a Disk Complete Interrupt will occur with an error indication. The Disk Status register will report the Correction Failed condition. This operation, like all other operations, may be started any time another disk operation is not in progress. Refer to Section 4.3.4 for a discussion of the procedure for correcting the data.

Disk Control (DCTL) 11h R/W

7	6	5	4	3	2	1	0
IR	I/S	HSS	DT	WPSIG	WGF	PLL	RES

IR: Interlock Required

1 The Interlock register (Header Byte Counter register) must be written to after the Disk Data Controller has completed a header operation and before the beginning of the Data Postamble field. If the Interlock register is not written to in time, the command will terminate with the LI (Late Interlock) bit in the Disk Status register set to ''1''. This allows the safe updating of header formatting registers during a Format operation. Normally used with the H/NC (Header Complete or Next Command) bit in the Disk Interrupt Enable register set to ''1'' to enable Header Complete interrupts. See Section 4.3.4 for a description of this technique.

0 Normal operation. No interlock function.

I/S: Index/Sector/AMF Interrupt

This bit controls the generation of an interrupt at each index, sector, or AMF. This interrupt may be masked by the I/S (Index/Sector) bit in the Disk Interrupt Enable register.

1 An interrupt will be generated as determined by Table 4.17. In general this will generate an interrupt at each Index pulse for soft sectored drives, each Index or AMF signal for pseudo-hard sectored drives, or at each Index or Sector pulse for hard sectored drives.

0 An interrupt will be generated by each Index pulse.

HSS: Hard or Soft Sectored

Works in conjunction with the DT (Drive Type) bit to control the AME and the AMF/Sector pins. In general, this bit should be set for hard or pseudo-hard sectored drives and cleared for soft sectored drives. Refer to Table 4.17 for the specific use of this bit.

DT: Drive Type

Works in conjunction with the HSS (Hard/Soft Sectored) bit to control the AME and the AMF/Sector pins. In general, this bit should be set for pseudo-hard sectored drives. Refer to Table 4.17 for the specific use of this bit.

WPSIG: Write Post Sector/Index Gap

This bit is only used in Write Header or Format operations.

1 WGATE is asserted during the Post Sector/Index Gap when a header is written.

0 WGATE is not asserted during the Post Sector/Index Gap.

WGF: Write Gate Format

This bit will allow a single revolution format operation with an RLL code. In some applications, if WGATE pulses low for 2 bit times, an external ENDEC will generate a valid preamble pattern. This bit will only effect the operation of WGATE during the Data Gap field.

1 WGATE is deasserted for 2 bit times at the beginning of the Data Gap field of each sector written.

0 WGATE remains asserted throughout the Data Gap field.

PLL: PLL Recovery Time

This bit is only used in soft sectored mode. It is used to control the amount of PLL relock time given to an external synchronizer. When used with the National's DP8459 or DP8469 the two byte choice gives the shortest necessary preamble lengths and takes advantage of the zero phase start of these PLLs. When using other PLLs, the six byte choice should be used. See Section 4.3.4.

1 RGATE is deasserted for six byte times following a synch mis-compare in Soft Sector mode.

0 RGATE is deasserted for two byte times following a synch mis-compare in Soft Sector mode.

RES: Reset

1 Reset Active: Setting this bit initiates a Disk Data Controller reset operation. Any disk command which may be in progress will be terminated after transferring up to four additional bytes. See Table 3.1 for a summary of registers affected by this reset.

This bit will also be set by asserting the $\overline{\text{CRST}}$ pin. Once this bit is set (even by $\overline{\text{CRST}}$), the only method to clear it is by writing a zero to this bit.

0 Normal Operation: Clearing this bit terminates the reset condition. The processor must wait a minimum of 17 RCLK cycles after reset has begun before clearing reset.

Disk Status (DSTAT) 14h R Only

7	6	5	4	3	2	1	0
LI	DL	SOR	EC			SC	

These codes allow processor monitoring of the progress of disk commands and a check on completion. Some of the information here is redundant to information provided in the Disk Interrupt register.

LI: Late Interlock

Will only occur if the IR (Interlock Required) bit in the Disk Control register is set to ''1''. The processor has failed to write to the Interlock register (Header Byte Count register) before the end of the data field of the present sector. The current command will terminate at the end of the data field. This status register will be updated immediately.

This error will only occur due to system design errors.

This error could occur in combination with the Data Field Error, Sector Overrun Error, or the Data Lost Error.

DL: Data Lost

This error will occur during a disk transfer if the FIFO overflows or underflows as data is transferred to or from buffer memory. The data transferred to or from the disk during this condition is invalid. The status register will be updated immediately. The current command will terminate after the current sector is completely written to the disk. The command will terminate immediately for a non-write operation.

This error will only occur due to system design errors. For example, when not enough memory bandwidth has been allocated for the disk data rate. Since disk transfers have the highest priority, this should be a rare occurrence indeed!

This error could occur in combination with the Sector Overrun Error, Late Interlock Error, Data Field Error, Header Fault Error, or the Disk Parity Error.

SOR: Sector Overrun

This error will occur when a sector or index pulse is detected while reading or writing in the middle of a sector. Specifically, for soft sectored drives, if RGATE is active, the first header synch byte has already been found, and a sector or index pulse is received, this error will be set. For hard sectored drives, the same conditions apply except no synch bytes need to be found.

Also, if WGATE is active, this error will occur when a sector or index pulse is received, except in the case of a soft sectored Format command while GAP 3 is being written.

This error could occur in combination with the Data Lost Error, Late Interlock Error, Data Field Error, Header Fault Error, or the Disk Parity Error.

EC: Error Code

000 **No Encoded Error.**

001 **Header Failed Although Sector Matched:** At least one of the header bytes marked with the SM (Sector Matched) bit in the corresponding Header Control register(s) matched correctly, but other header bytes were in error. This error will occur at the end of the header field who's sector number matched.

010 **Sector Not Found:** When the desired header cannot be found before two consecutive index pulses in any non-write Header operation. This error will occur at the second index pulse.

011 **Data Synch Failed:** If a sector or index pulse occurs while the DP8496/7 is waiting to byte align on the first data synch field (Synch #1, or Synch #2 if Synch #1 is disabled). This error will occur at the sector or index pulse.

This error code is also set if the DP8496/7 byte aligns to the first synch byte of the data field but does not match to subsequent bytes (more bytes of Synch #1 or Synch #2). This error will occur at the synch byte that does not match.

100 **Correction Failed:** Only possible after a Start Correction Cycle command. This indicates that the error cannot be corrected by the DP8496/7.

101 **Data Field Error::** Occurs when a CRC/ECC error is detected during an Ignore, Verify, or Read Data Field operation if the Data CRC/ECC verification is enabled in the ECC Control register.

110 **Read Header Fault:** Occurs only when a CRC/ECC error is detected during a Read Header operation if Header CRC/ECC verification is enabled by the ECC Control register. CRC/ECC faults during an Ignore or Compare Header operations are reported in the Status Code field as a Header Fault. This error occurs at the end of the header field.

111 **Disk Parity Error:** Occurs if a parity error is detected in the buffer memory during any disk write operation if buffer parity is enabled in the Setup 2 register. The data will be written to the disk normally as if there were no error. This status register will be updated immediately. The current command will terminate after the current sector is completely written to the disk.

SC: Status Codes

00 **Idle:** The Disk Data Controller is not executing a command, or has started executing a command but is waiting for the starting condition.

01 **Header In Progress:** The sequencer in the Disk Data Controller is in the Header section of the flowchart in *Figure 4.4* . The disk sequencer is searching for a match on a compare header operation, reading a header in a read header operation, or writing a header in a write header or format operation.

If a ''Sector Not Found'' or ''Header Failed Although Sector Matched'' error occurs, this Status Code will remain frozen until the next Disk Command or Reset.

10 **Header Fault:** While the Disk Data Controller is searching for the requested header during any header operation, all headers read will be scanned for CRC/ECC errors if enabled. This Header Fault code (10) will be set if a CRC/ECC error is detected in any header field encountered. However, if the header being sought is found and has no CRC/ECC error, the Header Fault code is changed to Data Operation In Progress (11). If the header being sought is not found and a Header Fault occurred, the Header Fault code (10) will remain frozen until the next Disk Command or Reset. This code could provide useful diagnostic information if a Sector Not Found error occurs.

11 **Data Operation In Progress:** The sequencer in the Disk Data Controller is in the Data section of the flowchart in *Figure 4.5* . In compare header operation, the header has been found. In a read header operation, a header has been read. In a write header operation a header has been written. This state is entered at the end of the Data Synch field which coincides with Header Complete Interrupt if enabled. This is at the point at which the Sector Byte Count registers are down-loaded.

OR

Correction Cycle In Progress: Set when the Start Correction Cycle command is written to the Disk Command register and remains until the Disk Complete Interrupt.

Disk Interrupt (DINT) 12h R/W

7	6	5	4	3	2	1	0
TI	E/R	I/S	GCI	SCI	H/NC	CCI	

This register indicates interrupts that have occurred. There are two types of interrupts, checkpoint and completion.

Checkpoint interrupts (bits 2–7) are separated into separate bits and may be asserted simultaneously. These interrupts may be cleared only by writing a ''1'' into the corresponding bit location. These bits may be cleared individually or simultaneously.

Completion interrupts are encoded into bits 0–1. Only one completion interrupt may be read at time. Completion interrupts may be cleared only by writing the same two bit pattern back to the Disk Interrupt register. A completion interrupt may be cleared simultaneously with other checkpoint interrupts.

Interrupts will be reported in the Disk Interrupt register always, independent of the Disk Interrupt Enable register.

If enabled interrupts remain after writing to the Disk Interrupt register, the $\overline{\text{DINT}}$ pin will deassert momentarily to retrigger an edge sensitive interrupt input to the processor.

This register is latched to prevent the contents from changing while reading the register.

TI: Timer Interrupt

This bit is set when the Timer Count register has reached zero. The Timer description can be found in Section 4.5.

E/R: Buffer Memory Data Error/Ready

If the BMD (Buffer Memory Data) bit in the Disk Interrupt Enable register is set to ''0'', the E/R bit indicates that the Buffer Memory Data register is ready to be read or written. Also, the Processor Pointer registers may be modified.

If the BMD bit in the Disk Interrupt Enable register is set to ''1'', this bit indicates that an interrupt error has occurred. The processor has attempted to read or write to the Buffer Memory Data register too quickly. The read or write was unsuccessful. See Section 4.2.8.

I/S: Index or Sector

This interrupt is generated according to Table 4.17 if the I/S (Index/Sector) bit in the Disk Control register is set to one. If the I/S it is set to zero, this interrupt will be generated at each Index pulse.

GCI: Group Complete Interrupt

This indicates that the Sector Count register can be reloaded. This occurs at the end of the data field of the last sector in a multi-sector command. This interrupt will only be set if the Sector Count register has been written to since the last down-load of the Sector Count register.

SCI: Sector Complete Interrupt

This interrupt is set after the last byte of a sector has been read and the CRC/ECC was good. This interrupt will not be set at the end of the last sector of a multi-sector operation.

H/NC: Header Complete or Next Command

This is a dual function interrupt defined by HC bit in the Disk Interrupt Enable register.

When HC bit in the Disk Interrupt Enable register is ''1'' (Header Complete), an interrupt will be generated at the end of the ID Segment. This is also where the Sector Byte Count registers are down-loaded. This will indicate when a new sector byte count or other format register can be loaded. This feature could be used to alter the length of sectors on the same track. See Section 4.3.4.

When HC in the Disk Interrupt Enable register is ''0'' (Next Command), an interrupt will be generated, as soon as a pending disk command is down-loaded. This will indicate when a new command may be written to the Disk Command register for disk command pipelining. See Section 4.3.4.

CCI: Command Completion Interrupt

One of these three codes will be obtained upon completing a disk command. They are cleared by writing the exact code pattern back to this register. For instance, a ''11'' will not clear a ''01'' coded interrupt.

00 **No Interrupt.**

01 **No Error:** Disk command complete. Set upon completion of any buffer memory activity and reads or writes to the drive. Also set upon the successful completion of a correcton cycle.

10 **Error:** Any error which aborts a disk command other than a Verify command. The specific error condition is coded in the Disk Status register.

11 **Verify Data Error:** If a data mis-compare occurs during a Verify Data command, this code will result. If this error occurs during a multi-sector operation, the rest of multi-sector operation will not complete. Other errors may occur at the same time as this error and will be reported appropriately in the Disk Status register.

Disk Interrupt Enable (DINTE) 13h R/W

7	6	5	4	3	2	1	0
TI	BMD	I/S	GCI	SCI	H/NC	HC	CCI

TI, I/S, GCI, SCI, H/NC

Setting these bits to a ''1'' will individually enable the corresponding interrupt as described in the Disk Interrupt register. When set to a zero, the interrupt functions are still active in the Disk Interrupt register, but will not assert the interrupt pin. Interrupts should be cleared from the Disk Interrupt register whether or not they are enabled in the Disk Interrupt Enable register.

BMD: Buffer Memory Data

1 Enables the Buffer Memory Data register Error interrupt function. Interrupt will occur when the processor has read or written the Buffer Memory Data register too quickly.

0 Enables Buffer Memory Data Ready function (no interrupt). The processor can poll the Disk Interrupt register to see if enough time has passed to read or write to the Buffer Memory Data register.

HC: Header Complete

1 Defines the H/NC interrupt as ''Header Complete''.

0 Defines the H/NC interrupt as ''Next Command''.

CCI: Command Complete Interrupt

1 Enables all disk Command Complete Interrupts (CCI field in the Disk Interrupt register).

0 Disables all disk Command Complete Interrupts.

Sector Number (SN) 15h R/W

7	6	5	4	3	2	1	0
Pattern							

The contents of this register will replace any header byte where the SN (Sector Number Substitution) bit is set to ''1'' in the corresponding Header Byte Control register. This register will be incremented after each sector operation is completed during a multi-sector command. This register should be initialized with the desired starting sector number for a multi-sector command.

This allows sequential logical sectors to be accessed during multi-sector commands without processor intervention.

Header Byte Count/Interlock (HBC) 16h R/W

7	6	5	4	3	2	1	0
x	x	x	x	x	Count		

Count is the number of header bytes in the Header Segment. The Count field must be loaded by the processor only for a Format C or Format D command. This count will allow the Disk Data Controller to request the proper number of header bytes from the buffer memory. Allowable values for count are 3 to 6. An even number of header bytes will always be transferred to and from buffer memory even though only the specified count will be used in the header.

This register is also used in interlock mode to signal completion to the Disk Data Controller if registers were updated by the processor between sectors. If the processor fails to write to this register before the end of the sector when the interlock mode is enabled, a Late Interlock error will be generated and the disk operation will be aborted.

Sector Count (SC) 17h R/W

7	6	5	4	3	2	1	0
Count							

Sets the number of sectors to operate on in a multi-sector operation.

Sector Count is really two registers called Active and Holding. The holding register is down-loaded by the Disk Data Controller to the active register at the same time a new command is down-loaded from the Disk Command register (just before the new command begins).

The Sector Count register is also down-loaded following the last sector operation which decrements the active Sector Count to zero. This down-load will only occur if a new value has been written to the Sector Count register after a previous down-load and before the zero detection. If this down-load occurs, the current disk command will continue with this new Sector Count.

The processor may access either active or holding register through this address as controlled by the A/H (Active/Holding) bit in the SCSI Operation register. Normally, the processor should only access the holding register.

ECC Control (EC) 07h R/W

7	6	5	4	3	2	1	0
PRE	DE	HE	Span				

PRE: Preset Control

0 A zeros preset is used for all CRC/ECC calculations.

1 A ones preset is used for all CRC/ECC calculations. This is the normal setting for most applications.

DE: Data Error Check Enable

This bit controls the verification of the CRC/ECC in the Data Segment during a Read, Ignore, or Verify Data operation.

1 Data field CRC/ECC is verified.

0 Data field CRC/ECC is not verified.

HE: Header Error Check Enable

This bit controls the verification of the CRC/ECC in the Header Segment during an Ignore, Compare or Read Header operation.

1 Header field CRC/ECC is verified.

0 Header field CRC/ECC is not verified.

SPAN

Span bits 0–4 are set for the longest burst error that is to be corrected. Allowable values are 3 to 28, depending on polynomial. Values under 3 will default to 3. Refer to Table 4.18 in Section 4.3.4 for recommended values for SPAN. This value is only used during the Start Correction Cycle command.

TABLE 4.13. ECC Byte Ordering

When Read	CRC/ECC Selected	Syndrom Bytes (Most Significant to Least Significant) (Individual Bits Are in **Reverse** Order) Most Significant Byte						Least Significant Byte
No Correction Cycle	16-Bit	ESR6	ESR0					
	32-Bit	ESR6	ESR5	ESR1	ESR0			
	48-Bit	ESR6	ESR5	ESR4	ESR2	ESR1	ESR0	
	56-Bit	ESR6	ESR5	ESR4	ESR3	ESR2	ESR1	ESR0
		Correction Mask (Read in Order) (Individual Bits Are in **Correct** Order) **1st**	**2nd**	**3rd**	**4th**	**5th**		
After Successful Correction Cycle	32-Bit	ESR1	ESR5	ESR6				
	48-Bit	ESR1	ESR2	ESR4	ESR5			
	56-Bit	ESR1	ESR2	ESR3	ESR4	ESR5		

Note: ESRx = ECC Shift Register x.

ECC Shift Register (ESRx) 00–06h R Only

7	6	5	4	3	2	1	0
Pattern							

The shift register will yield different types of information depending on when it is read. If read after a successful disk data read or write operation, they contain zeros.

Reading the shift register after an CRC/ECC error produces the syndrome bytes for that particular error. This syndrome may be stored and compared with syndromes from multiple reads to verify a hard error condition.

An error correction command may be issued to attempt to correct the data read from the disk. When the ECC Shift Register is read after a successful correction cycle it contains a correction mask which can be used to correct the error contained in Buffer Memory. See Section 4.3 for a discussion of the procedure for error correction.

ECC Byte Count—Low (ECCL) 18h R Only

7	6	5	4	3	2	1	0
Byte Count (Low)							

ECC Byte Count—High (ECCH) 19h R Only

7	6	5	4	3	2	1	0
Byte Count (High)							

These two registers contain an offset byte count after a correction cycle, which when added to the address of the start of the offending sector, identifies the location of the error. These registers only contain valid information after the end of a correction cycle.

4.3.3 Format Registers

The disk format is defined by using the format pattern and control registers. Generally these registers are set up in pairs: a pattern register and a control register. In each pair, the pattern register is loaded with an appropriate 8-bit pattern that will be written to the disk during a Format or Write operation, or will be used during a Read or Compare operation for byte alignment or a comparison in locating a sector. The control register will generally determine how many times the pattern is repeated, or the count. Refer to Table 4.14 and Table 4.15 to relate each pattern and control register to the actual disk format.

These registers should be loaded during the initialization process for the particular format chosen. Reset does not effect these registers.

Be sure to observe the restrictions on count fields being set to zero as specified in the individual register descriptions. There are some count fields which may not be zero. All fields listed in Table 4.14 and 4.15 are programmable.

RGATE and WGATE are asserted during certain fields. This is shown in *Figure 4.9* and *Figure 4.10* at the end of the section on Disk Operations (4.3.4).

Post Sector/Index Count (PSIG) 20h R/W

7	6	5	4	3	2	1	0
Count							

Sets the number of bytes in the Post Sector/Index field. The pattern used is from the Gap 3 Pattern register. WGATE will be asserted during this field according to the WPSIG (Write Post Sector/Index Gap) bit in the Disk Control register. This field can be used to skip sectored or wedge servo fields. RGATE is never asserted during this field for hard and pseudo-hard sectored drives. The Post Sector/Index field is not used if the DCOZ (Drive Command On Zero) bit is set in the Timer Prescale register).

For soft-sectored drives, PSIG is only used before the first sector. For other sectors, PSIG is assumed to have a length of one. Range is 1–FF.

Header Preamble Count (HPC) 21h R/W

7	6	5	4	3	2	1	0
Header Gap				Header Preamble			

Header Gap

Sets the number of bytes in the Header Gap field. The pattern used is from the Preamble Pattern register. Used to separate the time between RGATE and WGATE assertions. WGATE is always asserted during this field for a write operation. RGATE is never asserted during this field for hard and pseudo-hard sectored drives. Range is 1–7.

TABLE 4.14. Register Addresses for Format of ID Segment (Hex)

	PSIG	Hdr Gap	Hdr Prmbl	Snyc1	Snyc2	Hdr0	Hdr1	Hdr2	Hdr3	Hdr4	Hdr5	CRC/ ECC	Hdr Post
Pattern	3A	22	22	24	25	27	28	2A	2B	2D	2E	auto	30
Control	20	21	21	23	23	26	26	29	29	2C	2C	2F	2F
Range	1–FF	1–7	0–1F	0–7	0–1F	0–1	0–1	0–1	0–1	0–1	0–1	0–7	3–1F

TABLE 4.15. Register Addresses for Format of Data Segment (Hex)

	Data Gap	Data Prmbl	Sync1	Sync2	Data Field	CRC/ ECC	Data Post	Gap3
Pattern	22	22	33	34	37	auto	30	3A
Control	31	31	32	32	35, 36	38	38	39
Range	1–7	0–1F	0–7	0–1F	10–FFFE (Must Be Even)	0–7	1–1F	1–FF

Header Preamble

Sets the number of bytes in the Header Preamble for the PLL to lock to. The pattern used is from the Preamble Pattern register. Range is 0–1F.

Preamble Pattern (PREP) 22h R/W

7	6	5	4	3	2	1	0
Pattern							

Pattern used for the Header Gap, Header Preamble, Data Gap, and Data Preamble fields. This pattern must be all zeros for a soft-sectored drive.

Header Sync #1, #2 Count (HSC) 23h R/W

7	6	5	4	3	2	1	0
Synch #1			Synch #2				

Sets the number of bytes in the Synch #1 and Synch #2 fields. The range of Synch #1 is 0–7. The range of Synch #2 is 0–1F. *Both Synch #1 and Synch #2 cannot be zero.* At least one must be non-zero.

For soft sectored drives, AME is generated while writing the Synch #1 field to the disk. AMF is expected while reading Synch #1 from the disk.

Header Synch #1 Pattern (HSP1) 24h R/W

7	6	5	4	3	2	1	0
Pattern							

Pattern used for the Synch #1 field. Can be used as an Address Mark in soft sectored formats.

Header Synch #2 Pattern (HSP2) 25h R/W

7	6	5	4	3	2	1	0
Pattern							

Pattern used for the Synch #2 field.

Header Byte 0,1 Control (HC01) 26h R/W

7	6	5	4	3	2	1	0
NC0	SM0	SN0	HB0	NC1	SM1	SN1	HB1

This register does not perform any pattern repetition, nor does it define a field size. It is provided to control the function of each corresponding header byte. There is one Header Byte Control register for each pair of the six Header Byte Pattern registers. The upper four bits correspond to one pattern register. The lower four bits correspond to another pattern register.

NC: Not Compared

1 This header byte will always match true for a Compare Header operation.

0 This header byte will be compared normally without modification of the outcome.

SM: Header Failed Although Sector Matched Enable

1 The Header Failed Although Sector Matched function is enabled for this header byte. While searching for a matching Header Segment, if this field matches while other header bytes don't, an error is reported in the Disk Status register. Typically, this would be the sector number field. See Section 4.3.4 for additional information.

0 The Header Failed Although Sector Matched function is not enabled for this header byte.

SN: Sector Number Substitution

1 The contents of the Sector Number register are substituted for this Header Byte pattern during a Write Header operation and compared during a Compare Header operation. This is normally used in multi-sector commands.

0 The contents of the Header Byte Pattern register is written to the disk for a Write Header operation and compared during a Compare Header operation.

HB: Header Byte Active

1 This header byte contains valid data and will be used in the header operation.

These seven locations (00–06h) contain the CRC/ECC shift register contents. The register ordering is determined by the CRC/ECC polynomial selected and whether a correction cycle has taken place. This ordering is shown in Table 4.13.

0 This header byte is not included in the header byte field and will not be used in the header operation. The SN, SM, and NC bits for this header byte must also be set to zero. *Only the upper 4 bits or the lower 4 bits may be set to all zero for each Header Byte Control register, not both*. This implies that the minimum number of Header Bytes is three, and the maximum is six.

Header Byte 0 Pattern (HP0) 27h R/W

7	6	5	4	3	2	1	0
Pattern							

Pattern used for Header Byte 0.

Header Byte 1 Pattern (HP1) 28h R/W

7	6	5	4	3	2	1	0
Pattern							

Pattern used for Header Byte 1.

Header Byte 2,3 Control (HC23) 29h R/W

7	6	5	4	3	2	1	0
NC2	SM2	SN2	HB2	NC3	SM3	SN3	HB3

Same as Header Byte 0,1 Control but for Header Bytes 2 and 3.

Header Byte 2 Pattern (HP2) 2Ah R/W

7	6	5	4	3	2	1	0
Pattern							

Pattern used for Header Byte 2.

Header Byte 3 Pattern (HP3) 2Bh R/W

7	6	5	4	3	2	1	0
Pattern							

Pattern used for Header Byte 3.

Header Byte 4,5 Control (HC45) 2Ch R/W

7	6	5	4	3	2	1	0
NC4	SM4	SN4	HB4	NC5	SM5	SN5	HB5

Same as Header Byte 0,1 Control but for Header Bytes 4 and 5.

Header Byte 4 Pattern (HP4) 2Dh R/W

7	6	5	4	3	2	1	0
Pattern							

Pattern used for Header Byte 4.

Header Byte 5 Pattern (HP5) 2Eh R/W

7	6	5	4	3	2	1	0
Pattern							

Pattern used for Header Byte 5.

Header ECC and Postamble Count (HECC) 2Fh R/W

7	6	5	4	3	2	1	0
CRC/ECC				Postamble Count			

CRC/ECC

000: No CRC/ECC

010: 16-Bit CRC

100: 32-Bit ECC

110: 48-Bit ECC

111: 56-Bit ECC

Postamble Count

Number of bytes in the Header Postamble. Range is 3–1F.

Postamble Pattern (POSTP) 30h R/W

7	6	5	4	3	2	1	0
Pattern							

Pattern used for Header Postamble field and Data Postamble field.

Data Preamble Count (DPC) 31h R/W

7	6	5	4	3	2	1	0
Data Gap				Data Preamble			

Data Gap

Sets the number of bytes in the Data Gap field. The pattern used is from the Preamble Pattern register. Used to separate the time between RGATE and WGATE assertions. WGATE is always asserted during this field for a write operation. RGATE is never asserted during this field for hard and pseudo-hard sectored drives. Range is 1–7.

Data Preamble

Sets the number of bytes in the Data Preamble for the PLL to lock to. The pattern used is from the Preamble Pattern register. Range is 0–1F.

Data Synch #1, #2 Count (DSC) 32h R/W

7	6	5	4	3	2	1	0
Synch #1				Synch #2			

Sets the number of bytes in the Synch #1 and Synch #2 fields. The range of Synch #1 is 0–7. The range of Synch #2 is 0–1F. *Both Synch #1 and Synch #2 cannot be zero.* At least one must be non-zero.

For soft sectored drives, AME is generated while writing the Synch #1 field to the disk. AMF is expected while reading Synch #1 from the disk.

4.0 Functional Description (Continued)

Data Synch #1 Pattern (DSP1) 33h R/W

7	6	5	4	3	2	1	0
Pattern							

Pattern used for the Synch #1 field. Can be used as an Address Mark in soft sectored formats.

Data Synch #2 Pattern (DSP2) 34h R/W

7	6	5	4	3	2	1	0
Pattern							

Pattern used for the Synch #2 field.

Sector Byte Count—Low (SBCL) 35h R/W

7	6	5	4	3	2	1	0
Count							

Sector Byte Count—High (SBCH) 36h R/W

7	6	5	4	3	2	1	0
Count							

These two registers set the number of data bytes in a sector. The range of the combined value for the Byte Count is 10h–FFFEh. This must be an even number.

Data Format Pattern (DFP) 37h R/W

7	6	5	4	3	2	1	0
Pattern							

Pattern used for the data field during a Format Type A or C. For Format Types B and C, the data field will be transferred from buffer memory.

Data ECC and Postamble Count (DECC) 38h R/W

7	6	5	4	3	2	1	0
CRC/ECC			Postamble Count				

CRC/ECC

000: No CRC/ECC

010: 16-Bit CRC

100: 32-Bit ECC

110: 48-Bit ECC

111: 56-Bit ECC

Postamble Count

Number of bytes in the Data Postamble. Range is 1–1F.

Gap 3 Byte Count (GAPC) 39h R/W

7	6	5	4	3	2	1	0
Count							

Number of bytes in Gap 3. In hard sectored mode, this count must be programmed small enough to time-out before the next Index or Sector pulse.

During a soft-sectored Format operation, Gap 3 functions normally except after the last sector. After the last sector, the Gap 3 pattern is repeated until the Index pulse is received. The Gap 3 Byte Count is ignored after the last sector.

Gap 3 Pattern/PSIG (GAPP) 3Ah R/W

7	6	5	4	3	2	1	0
Pattern							

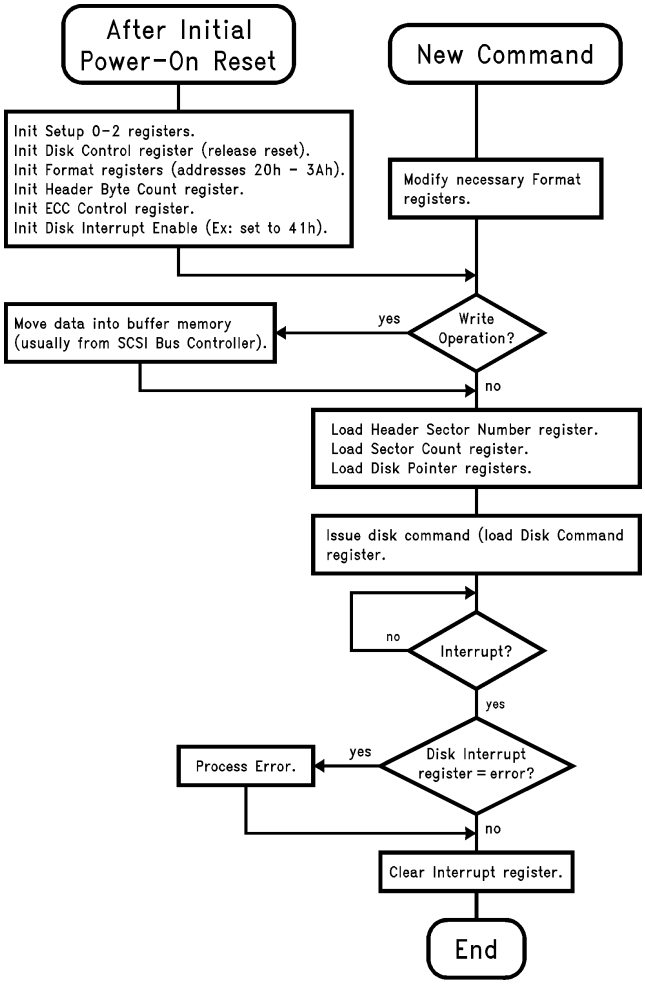
Pattern used in Gap 3 field and Post Sector/Index Gap field.

4.0 Functional Description (Continued)

4.3.4 Disk Operations

The DP8496/7 has been designed to offer a great deal of flexibility in many areas. The disk formats that may be used are quite varied. The procedures that the processor may use to access information on the disk are varied as well.

The flowchart in *Figure 4.6* describes how to implement simple commands such as reading data from the disk, writing data to the disk, and formatting the disk. This flowchart has been kept quite simple. More advanced techniques may be used to streamline performance, such as pipelining commands. Some of these techniques are described later.



TL/F/11212–13

FIGURE 4.6. Software Flowchart for Simple Disk Operations

4.0 Functional Description (Continued)

Extending the Current Disk Command

Disk commands with the MSO (Multi-Sector Operation) bit set to ''1'' in the Disk Command register may be extended to transfer more sectors than originally programmed. If a new value is written to the Sector Count register while a disk command is executing, the new value will be downloaded from the holding register to the active register when the active Sector Count reaches zero. A new Disk Pointer will also be downloaded at the same time if a new value has been written to the Disk Pointer registers.

A new, pipelined Sector Count may be written to the Sector Count register any time before the active Sector Count reaches zero. The Sector Count register should not be updated again (after the first pipelined value) until the previous pipelined value has been downloaded. The ''Group Complete'' interrupt indicates when a pipelined Sector Count and Disk Pointer is downloaded. After this interrupt is received, the Sector Count register and Disk Pointer registers may be updated again.

A disk command may be extended multiple times if desired. Simply wait for the ''Group Complete'' interrupt between each update of the Sector Count register.

Table 4.16 summarizes these interrupts.

Pipelining Disk Commands

Disk commands may be pipelined. If a new disk command is written to the Disk Command register while an operation is executing, the new disk command will be buffered and downloaded internally when the current command has completed.

To assist in determining when a new disk command may be written to the Disk Command register for pipelining, an interrupt may be used. If the H/NC (Header Complete/Next Command) bit in the Disk Interrupt Enable register is set to ''0'', the ''Next Command'' interrupt is enabled. When this interrupt occurs, a new disk command may be written to the Disk command register. This new disk command will be downloaded and executed after the current disk command has completed all of its operations. If the current disk command is multi-sectored, the new disk command will not be downloaded until all the sector operations have completed.

The Sector Count register will also be downloaded at the same time a disk command is downloaded. If a new value is not written to this register, the value from the previous download will be used. A new Disk Pointer will also be downloaded at the same time as the new disk command—but only if the Disk Pointer has been written to since the last disk command download.

Disk commands may be pipelined multiple times if desired. Simply wait for the ''Next Command'' interrupt between each update of the Disk Command register.

Special Formats

Some applications require tracks that are formatted with sectors of varying lengths, or sectors with different synch bytes, or other unique format variations. This requires modifying the header format registers ''on the fly'' during a format operation. It is important to modify these registers only while they are not being used by the Disk Data Controller.

The processor can use the Interlock mode and the Header Complete interrupt to allow modification of the disk format registers for the ID Segment (addresses 21h, 23h–2Fh) or the Sector Byte Count registers (35h, 36h) during the format operation. The Header Complete interrupt is generated after the Sector Byte Count registers have been downloaded at the beginning of the data field. The processor has until the Data Postamble field to make any necessary changes to the desired registers and then write to the Interlock (Header Byte Count) register, telling the DP8496/7 that changes are complete. If Interlock is not received in time, a Late Interlock interrupt is generated.

TABLE 4.16. Disk Buffer Management Interrupts

Command Type	Sector Count Reached Zero?	New Disk Command Been Loaded?	New Sector Count Been Loaded?	Interrupt Type	Comments
Multi-Sector (MSO = 1)	No	X	X	SCI	Interim sector completed of a multi-sector operation
	Yes	No	Yes	GCI	The last sector of the previous Sector Count has been transferred to/from the disk. Command is extended with new Sector Count.
	Yes	Yes	X	CCI	Previous disk command has completed. New pipelined command is starting.
	Yes	No	No	CCI	Previous disk command has completed. No new command.
Single Sector (MSO = 0)	X	Yes	X	CCI	Single sector command has completed. New pipelined command is starting.
	X	No	X	CCI	Single sector command has completed.

Note: MSO (Multi-Sector Operation) bit is in the Disk Command register. SCI = Sector Complete Interrupt, GCI = Group Complete Interrupt, CCI = Command Complete Interrupt.

4.0 Functional Description (Continued)

The altered format registers will apply to the next sector to be operated on, regardless of whether the drive command is a multi-sector type or not.

A new drive command may also be loaded along with updating the Format registers. This would facilitate a Read Header/Read Data command (to locate track position) followed by a Compare Header/Read Data command (multi-sector) to read the rest of the track. See the previous sub-section for a description of how commands are pipelined.

Changing disk format registers other than the ones listed above can have critical timing restrictions. In general, if the field to be changed for the next sector has been, or is being written to disk for the current sector, it is safe to change the count and pattern registers for that field.

The ''Header Complete'' interrupt is enabled by setting the H/NC (Header Complete/Next Command) bit in the Disk Interrupt Enable register to a ''1''.

Single Sector Formatting

Format operations normally start with an index pulse and end with the next index pulse, thus formatting one track. Individual sectors can be formatted for hard sectored drives only, via the Write Header/Write Data command. A Compare Header/Read Data may be issued before the Write Header/Write Data command to locate the sector preceding the sector to be formatted.

AMF/AME Operation

Address mark found (input) and address mark enable (output) pins are usually used with soft sectored drives to synchronize the actual disk with DP8496/7 data operations. On true soft sectored drives (ST506/412 type), AME is generated for each byte of Synch #1 generated during a format or write operation. On a compare or read operation, AMF is expected for each Synch #1 byte read. In MFM systems this is commonly generated by a missing clock from an A1h byte.

In the pseudo-hard sectored mode AME is used to perform two different functions. During a format, AME is asserted during the Header Gap field. This, along with the fact that WGATE was already asserted instructs the drive to write an Address Mark (the Header Gap Count should be loaded with a 3 according to the ANSI draft proposed ESDI specification). During a read or write data operation AME is asserted after the specified starting condition for a soft sectored drive (immediately, wait for index, wait for timer, etc.). The Disk Data Controller will then wait for AMF to be asserted by the drive before executing the command. AME will be asserted again before each sector operation, and the Disk Data Controller will wait for the AMF before continuing.

Table 4.17 shows the proper state for the HSS and DT bits (Hard/Soft Sectored and Drive Type) in the Disk Control register based on the drive used.

Disk Data Error Handling

The DP8496/7 uses a fixed 32, 48 or 56-bit ECC polynomial for detection and correction of errors. Correction is performed entirely on-chip with one of the fixed polynomials. The processor needs only to EX-OR the correction mask bytes contained on-chip after a successful correction cycle with the bytes in error in buffer memory.

It is not recommended to use the maximum correction span listed in Table 4.18. The chance for a mis-correction is too great. The recommended correction span produces the best trade-off between correction span and probability of mis-correction. The correction span is set in the ECC Control register.

TABLE 4.17. Drive Types

Drive Type	Operation	HSS	DT	PSIG Written	AME Asserted during . . .	AMF/Sector Pulse Expected	If I/S* Set to ''1'', Int. Generated:
Soft Sectored ST506 Type Format	Format	0	0	First Sector of Track	Hdr. Synch #1, Data Synch #1	X	Index
	Write	0	0	No	Data Synch #1	Hdr. Synch #1	
	Read	0	0	No	Never	Hdr. Synch #1, Data Synch #1	
Pseudo Hard Sectored (ESDI Soft Sectored) Using Gap Type Address Mark	Format	0	1	First Sector of Track	Hdr. Gap	X	Index
	Write	1	1	No	ESDI Handshake	AMF for ESDI Handshake	Index or AMF
	Read	1	1	No	ESDI Handshake	AMF for ESDI Handshake	
Hard Sectored Not Using Gap Type Address Mark	Format	1	0	All Sectors	Never	Sector	Index or Sector
	Format	1	1	All Sectors	Post Sector/Index Gap	Sector	
	Write	1	0	No	Never	Sector Pulse	
	Read	1	0	No	Never	Sector Pulse	

Note: I/S (Index/Sector) bit is in Disk Control register. If the I/S bit is set to ''0'', an interrupt is generated at each Index pulse only.

TABLE 4.18. ECC Correction Span

Polynomial	Maximum	Recommended
16-Bit CRC	None	None
32-Bit ECC	11 Bits	5 Bits
48-Bit ECC	15 Bits	11 Bits
56-Bit ECC	22 Bits	17 Bits

CRC may be used for either Header or Data fields, or both. This is set in the Data or Header ECC count registers. The CRC-CCITT polynomial used by the DP8496/7 is given below:

$$P(x) = x^{16} + x^{12} + x^5 + 1$$

The 32-bit code is a public domain, computer generated code. This is listed below:

$$P(x) = x^{32} + x^{28} + x^{26} + x^{19} + x^{17} + x^{10} + x^6 + x^2 + 1$$

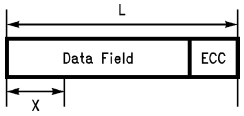
The 48 and 56-bit polynomials must be licensed by National Semiconductor to users of the DP8496/7 for exclusive use in products containing the DP8496/7. The codes generated on disks by these polynomials may be distributed freely, as when used in floppy or removable hard disk media. There is no charge for this license.

Operation during Correction

The DP8496/7 can be set to correct an error any time after an error has been detected and before another command has been issued. A correction cycle will not take place unless a Start Correction Cycle command has been issued.

By the time an error is reported by the DP8496/7, the data read from the disk will have been transferred to buffer memory. To correct an error, a Start Correction command must be issued by the processor. During the execution of this command, the buffer memory will not be accessed by the Disk Data Controller, which leaves the buffer memory free for other non-disk operations. At the end of the correction command, the processor must use the information provided by the DP8496/7 to manually correct the data residing in buffer memory.

The time it takes the Start Correction command to complete is determined by the error's location in the sector. The nearer to the start of the sector, the longer the DP8496/7 takes to locate the error. This time can be determined using the formula shown in *Figure 4.7* . It should be noted that this is the internal correction time only. More time is required for the processor to perform additional operations.



TL/F/11212–14

$$\text{Correction time} \approx \frac{L - X}{f}$$

L = Length of data and ECC fields (bits)
X = Distance from start of sector to first bit in error (bits)
f = Read clock frequency (Hz)

FIGURE 4.7. Time for ECC Correction

The proper procedure to perform error correction is listed below:

1. Clear Disk Interrupt register.
2. Set the Sector Byte Count registers to the sum of the original sector length plus the length of the ECC polynomial (4, 6, or 7 bytes).
3. Issue the Start Correction Cycle command.
4. After the command has finished (interrupt generated), check the Disk Interrupt register. If an error is indicated, the Correction Failed bit in the Disk Status register will also be set. The ECC error is not correctable by the DP8496/7.
5. If no error is reported, the error is correctable. The address of the first byte in buffer memory that must be corrected is given by the formula below:

   [Address of Start of Sector] + [Contents of ECC Byte Count Register] − 1

   If the contents of the ECC Byte Count register is greater than or equal to the sector length, the error is in the ECC itself. The data in buffer memory is already correct and should not be modified.
6. Use the Processor Pointer and Buffer Memory Data register to read the invalid bytes from buffer memory. Use the data in the ECC Shift Registers specified in Table 4.13. EX-OR the first ECC Shift register byte with the first buffer memory byte in error. EX-OR the second ECC Shift register byte with the second buffer memory byte in error, etc.
7. Write the corrected data back to buffer memory with the Processor Pointer and the Buffer Memory Data register.

Header Diagnostic Operations

The Header Failed Although Sector Matched function is enabled independently for each header byte by the SM (Header Failed Although Sector Matched) bit in the Header Control registers. If a Compare Header/Ignore Data operation is performed and the total header did not compare but any byte with the SM bit enabled did compare correctly, an ''Error'' completion interrupt will be generated. The ''Header Failed Although Sector Matched'' status will be reported in the Disk Status register. The processor could then read the header bytes from buffer memory.

To find the last header bytes read from the disk in the buffer memory, the following steps may be taken:

1. Set the A/H (Active/Holding) bit in the SCSI Operation register to ''1''.
2. The first byte of the failing header information is located in buffer memory at the address pointed to by:

   [Contents of Disk Pointer Register] − [Number of Bytes in Each Header]
3. Set the A/H bit in the SCSI Operation register back to ''0''.

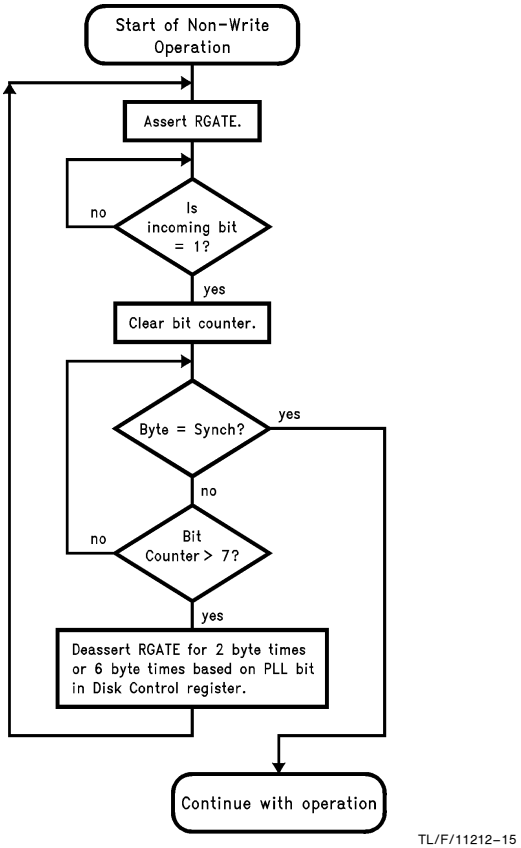


FIGURE 4.8. RGATE Operation during Search for Synch

RGATE Timing for Soft Sectors

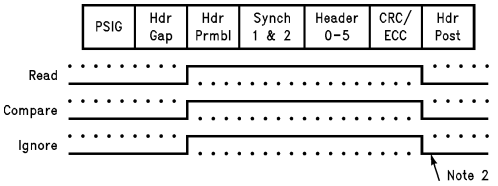
Ideally RGATE should transition from deasserted to asserted only while the drive's head is positioned over a preamble field. Proper programming of the gap lengths in the Format registers strive for this condition.

However, for soft sectored drives, it is impossible to avoid asserting RGATE over non-preamble fields as shown in *Figure 4.9* .

For soft sectored drives, the DP8496/7, employs a RGATE algorithm to allow an external PLL to re-lock to the proper frequency while searching for a preamble and address mark. This algorithm is shown in *Figure 4.8*. While reading a preamble (or what the DP8496/7 thinks is a preamble), the first non-zero bit clears a bit counter. If the data read for this bit and the next 7 bits do not match the proper Synch register, RGATE is deasserted. If a Synch match is made, the read operation continues to its conclusion.

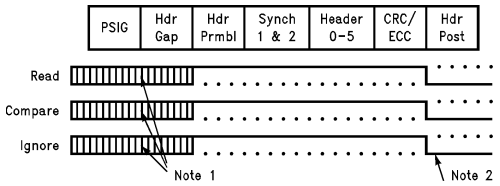
If RGATE is deasserted, it will remain deasserted for either two or six byte times. This time is determined by the PLL (PLL Recovery Time) bit in the Disk Control register.

RGATE during Header Segment (Hard and Pseudo-Hard Sectored)



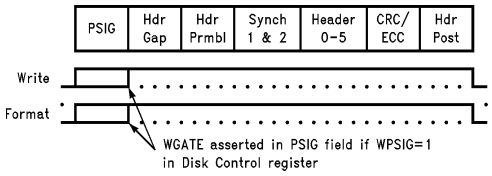
TL/F/11212–16

RGATE during Header Segment (Soft Sectored)



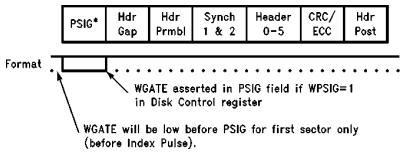
TL/F/11212–17

WGATE during Header Segment (Hard Sectored)



TL/F/11212–18

WGATE during Header Segment (Soft and Pseudo-Hard Sectored)



TL/F/11212–19

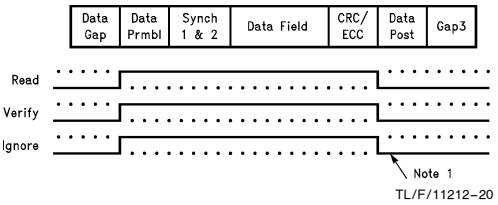
Note 1: While searching for the Header Preamble, RGATE will transition from asserted to deasserted following the algorithm shown in *Figure 4.8*.

Note 2: RGATE will become deasserted one byte time after the end of CRC/ECC field.

Note *: PSIG* field for soft sectored drives only appears before the first sector. For other sectors, PSIG* will have a field length of one byte.

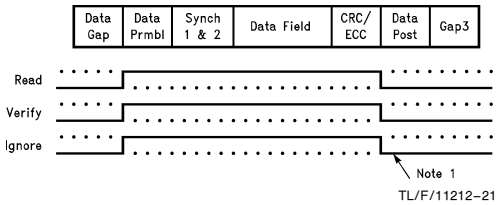
FIGURE 4.9. RGATE and WGATE Timing during Header Segment

RGATE during Data Segment (Hard and Pseudo-Hard Sectored)



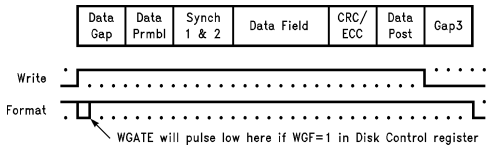
TL/F/11212–20

RGATE during Data Segment (Soft Sectored)



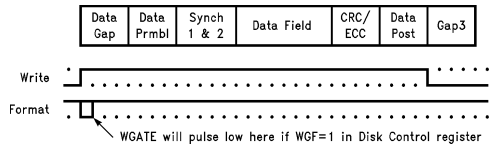
TL/F/11212–21

WGATE during Data Segment (Hard Sectored)



TL/F/11212–22

WGATE during Data Segment (Soft and Pseudo-Hard Sectored)



TL/F/11212–23

Note 1: RGATE will become deasserted one byte time after the end of CRC/ECC field. Also, if an additional sector is to be read, RGATE will become re-asserted after a Header Gap time plus two bytes after this point.

FIGURE 4.10. RGATE and WGATE Timing during Data Segment

4.0 Functional Description (Continued)

4.4 SCSI INTERFACE

Primary among the objectives for the SCSI interface is to simplify the microprocessor's interaction with the DP8496/7. The most complex portion of SCSI related code is usually dedicated to interrupt handling, whether vectored or polled. In the DP8496/7, the multi-phase commands and combination commands help to reduce the number of interrupts and to pre-define the reason for those interrupts which do occur. An interrupt mask register enables the software to control interrupt flow. Also, a physically separate pin dedicated for SCSI interrupts ($\overline{SINT}$) can shorten the processor response time. The chip is also capable of operating in either the target mode, for which it is optimized, or the initiator mode.

The SCSI Port can be controlled by the processor by utilizing the DP8496/7 in one of two different modes: Automatic or Manual. Automatic mode enables the SCSI Command register and the state machines which make up the SCSI protocol engine to interpret and perform multi-phase or combination commands.

Manual mode disables the SCSI Command register and SCSI protocol engine. The processor must then manipulate both the SCSI control and data bus through programmed I/O. The SCSI Control register and the SCSI Data register are the windows into the SCSI bus which allow this manipulation.

The DP8496/7 may be switched between the Automatic and Manual modes of operation. There are a few restrictions, however, and these are detailed in the SCSI Operation register description in Section 4.4.2.

The DP8496 provides 48 mA, single-ended transceivers on all SCSI bus signal pins; so the bus lines can be driven directly. The DP8497 version is designed to interface with differential transceivers external to the chip for Fast SCSI applications. Therefore, on the DP8497 all SCSI outputs are the standard, 2 mA type. Fourteen additional pins on the DP8497 provide direction control for the external differential transceivers. In the manual mode of operation, the user controls the transceiver direction by programming two registers, SDIF1 and SDIF2 (4Eh and 4Fh). Also note that on the DP8497 all SCSI signals are active high.

4.4.1 Simple SCSI Operations

The flowchart in *Figure 4.11* describes how to implement simple SCSI operations such as reading data from the disk and writing data to the disk. This flowchart has been kept quite simple. The next section describes the various registers that the user must program in order to configure and control the SCSI Controller's operation. More advanced SCSI operations are then described in Section 4.4.3.

4.4.2 SCSI Registers

SCSI Command (SCMD) 40h W Only

7	6	5	4	3	2	1	0
SCSI Command Opcode							

While in Automatic mode, writing an opcode to this register will initiate a SCSI command. Normally other SCSI registers are configured before writing to this register. After the proper registers are loded, writing to this register will initiate the operation. Invalid opcodes should NEVER be issued. After any invalid opcode is written to this register, the DP8496/7 should be reset with the $\overline{CRST}$ pin.

In general SCSI commands cannot be pipelined. The combination commands eliminate the need for most pipelining. New commands should be written to the SCSI Command register only after the previous command has completed its operation as indicated by its completion interrupt. See Section 4.4.3 for a description of certain commands that may be pipelined.

SCSI commands which transfer information blocks may be extended, however. If a new value is written to the SCSI Block Count register, the command will continue by downloading the new holding register value to the active SCSI Block Count register when the current active contents reaches zero. This is described in Section 4.4.3.

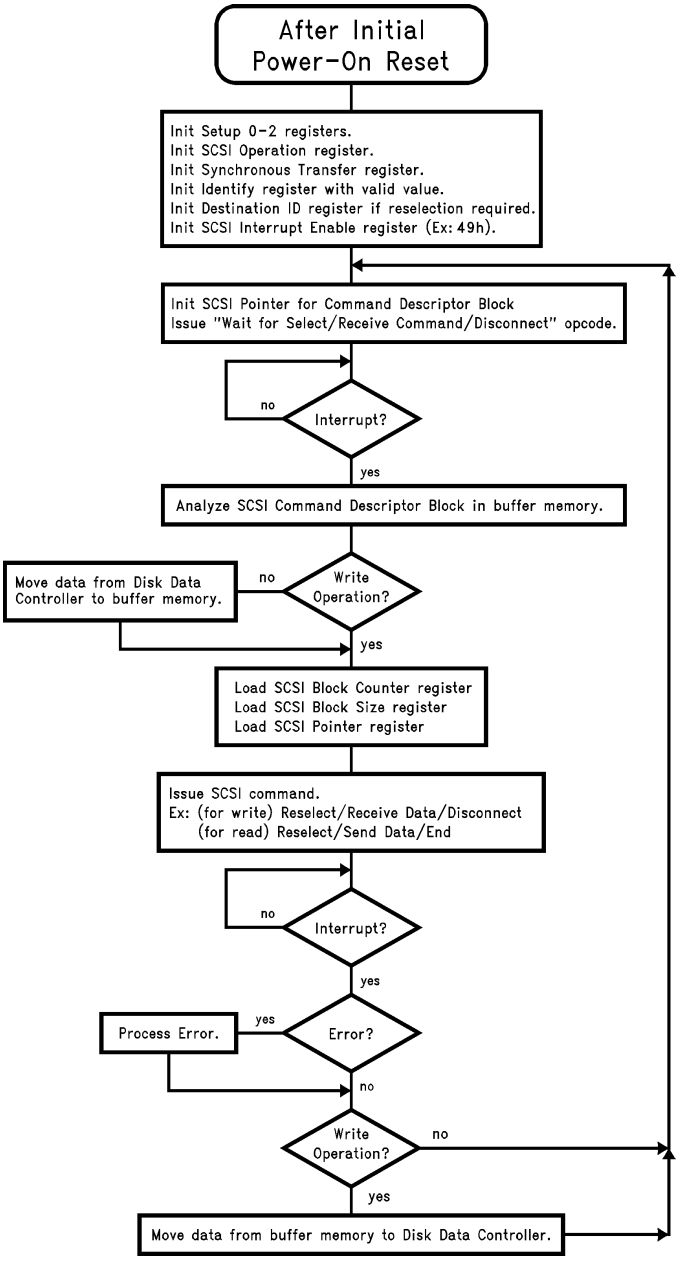


FIGURE 4.11. Software Flowchart for Simple SCSI Operation

4.0 Functional Description (Continued)

The valid SCSI Commands are listed in Table 4.19. Flowcharts of the multi-phase commands are shown in *Figures 4.12* to *4.19* at the end of this description of the SCMD register.

TABLE 4.19. SCSI Command Opcodes

Op-Code	Command	Mode	Int?
00	Clear Pending Command	I/T	N
2B	Reset	I/T	Y
32	Disconnect	I/T	N
30	Message Accepted	I	N
10	(Re)Select	I/T	Y
08	Abort (Re)Select	I/T	Y
3E	Receive Command	T	Y
3F	Receive Data	T	Y
3A	Receive Message	T	Y
3B	Receive Unspecified	T	Y
0A	Send Status	T	Y
0B	Send Data	T	Y
0E	Send Message	T	Y
0F	Send Unspecified	T	Y
3C	Transfer Info	I	Y
38	Transfer Pad	I	Y
33	Send Disconnect	T	Y
31	Send End	T	Y
1F	Reselect-Receive Data	T	Y
1B	Reselect-Send Data	T	Y
1C	Reselect-Receive Data-Disconnect	T	Y
18	Reselect-Send Data-Disconnect	T	Y
1D	Reselect-Receive Data-End	T	Y
19	Reselect-Send Data-End	T	Y
35	Wait for Sel.-Busy-End	T	Y
36	Wait for Sel.-Receive Command	T	Y
37	Wait for Sel.-Receive Command-Disconnect	T	Y
90	(Re)Select Self Test	I/T	Y
B2	Disconnect Self Test	I/T	N
8B	Send Data Loopback	T	Y
BF	Receive Data Loopback	T	Y
BC	Transfer Info Loopback	I	Y

Mode: I = Initiator, T = Target

INT = Interrupt generated at command termination? (Y/N)

Clear Pending Command (Opcode = 00)

This command will clear the SCSI Command register of any outstanding or pending commands. This may be used in the following situations.

1. After a ''Wait for Select'' command is written to the SCSI Command register, a Clear Pending Command may be issued to prevent the ''Wait for Select'' command from executing. See Section 4.4.3 for a description of the proper sequence to clear the ''Wait for Select''.
2. If a new command is written to the SCSI Command register, and you want to prevent it from executing you may issue the Clear Pending Command. The following condition may create this situation:
   A. If a command is pipelined, the Clear Pending Command may be issued to clear the pipelined command before it is executed. See 4.4.3 for pipelining restrictions.
   B. If a command is pipelined, and the currently executing command terminates with an error, the pipelined command will not execute until the interrupt is cleared. The Clear Pending Command may be issued to clear the pipelined command before it is executed.
   C. If an asynchronous event generates a completion interrupt (SCSI Bus Reset or (Re)Select) just before a command is written to the SCSI Command register, the command will be pipelined and will not execute until the interrupt is cleared. The Clear Pending Command may be issued to clear the pipelined command before it is executed.
3. Before modifying the T/I (Target/Initiator) bit in the SCSI Operation register, the Clear Pending Command should be issued. This will prevent an ''Invalid Command'' interrupt from being issued if the last command written to the SCSI Command register is invalid for the new mode set with the T/I bit. This operation is not required if you are sure that the last command is not invalid for the new mode.
4. The Clear Pending Command is used to clear the self test mode. The Clear Pending Command should be issued after issuing the Disconnect Self Test command.

Reset (Opcode = 2B)

This command will terminate and clear the current command (if any) and will clear any pipelined command. Refer to Table 3.1 for other registers that are affected by this command. This command may be issued at any time.

At the completion of this command, the ''Operation Complete'' interrupt will be generated. This interrupt may be buffered after another interrupt as described in the SCSI Interrupt register.

Disconnect (Opcode = 32h)

This command will immediately release the SCSI lines to the Bus Free phase. This command must be issued only if there are no pending commands (Status Code in SCSI Status register = 0111).

This command will not generate an interrupt because it is immediate, without the possibility of an error.

Message Accepted (Opcode = 30h)

Valid only as an initiator. This command should be used after a Transfer Info or Transfer Pad command while in the Message In phase. At the termination of the Transfer Info or Transfer Pad command, the ACK signal will normally be deasserted. However, if these commands are issued while in the Message In phase, the ACK signal will remain asserted at the termination of the command. The Message Accepted command can be used to deassert the ACK signal in this case.

If the message is to be accepted, issue the Message Accepted command which will deassert the ACK signal. If the message is to be rejected, the ATN signal should be asserted before the Message Accepted command is issued. The ATN signal may be asserted by setting the ATN bit in the SCSI Control register.

(Re)Select (Opcode = 10h)

This command initiates a rather lengthy series of events commencing with detecting the Bus Free phase. An Arbitration phase is executed next if enabled in the SCSI Operation register. This phase asserts the SCSI ID (initialized in the Setup 2 register) on the SCSI bus and detects if any device of higher priority is arbitrating with it. The DP8496/7 will cycle through this sequence until it wins arbitration. After winning arbitration, the chip Selects the Target or Reselects the Initiator whose ID is contained in the Destination ID register. It will wait for a response from that device indefinitely with SEL and ID's asserted and BSY deasserted. When the selected device responds by asserting BSY, the command will terminate with a completion interrupt.

None of the above sequences are protected by time-outs, because it is much more efficient to bracket the entire selection process with a single, processor controlled timer.

The choice between Selection and Reselection is determined by the T/I (Target/Initiator) bit in the SCSI Operation register. While in the Target mode, this command will execute a Reselection. While in the Initiator mode, a Selection will be executed.

While in the Initiator mode, the ATN signal may be controlled during selection by the SWA (Select With Attention) bit in the SCSI Operation register.

Abort (Re)Selection (Opcode = 08h)

This command is used to terminate a pending (Re)select combination command. This command will function correctly only if the initial (re)selection has not yet begun, or while the (re)selection is in progress.

If this command is issued before or during arbitration, the SCSI bus is immediately released. This will result in an ''Operation Complete'' interrupt.

If issued after this point but before the desired SCSI device has responded with the BSY signal, the SCSI Data Bus will be released. If the BSY signal is not detected after a certain delay, the SEL signal will be deasserted and the command will terminate with an ''Operation Complete'' interrupt. If the BSY signal is detected, a ''Last Command Ignored'' interrupt will be generated and the DP8496/7 will continue as if the Abort (Re)Selection command had not been issued at all (except for the ''Last Command Ignored'' interrupt). This follows option 2 of paragraph 5.1.3.5 of the X3.131-1986 specification.

If this command is issued after the desired SCSI device has already responded with the BSY signal, no interrupt will be generated for this command. The DP8496/7 will continue as if the Abort (Re)Selection command had not been issued at all.

To verify the result of this command, the SCSI Status register should be read at least 20 BCLK periods after this command is issued. If the Status Code is less than 7 (0000–0111), the Abort (Re)Select command was issued in time. Simply wait for the completion interrupt.

If the Status Code is 7 or greater (0111–1111), the Abort (Re)Select command was issued too late. The Selection has already completed, and there may be no completion interrupt generated by the Abort (Re)Selection command. However, a completion interrupt will be generated when the original (Re)Select or (Re)Select combination command terminates.

Receive Command (Opcode = 3Eh)
Receive Data (Opcode = 3Fh)
Receive Message (Opcode = 3Ah)
Receive Unspecified (Opcode = 3Bh)

Send Status (Opcode = 0Ah)
Send Data (Opcode = 0Bh)
Send Message (Opcode = 0Eh)
Send Unspecified (Opcode = 0F)

The Send and Receive commands listed above are used to transfer information on the SCSI bus. The information transferred will be to or from the buffer memory.

These commands are only valid if the DP8496/7 is configured as a Target in the Automatic mode. The DP8496/7 must already be connected (BSY asserted). The function of these commands are all similar except for the I/O, C/D and MSG lines. The state of these control lines are listed in Table 4.20.

The Receive Data command and the Send Data command will automatically use synchronous transfers on the SCSI bus if enabled in the Synchronous Transfer register.

A SCSI bus parity error or the assertion of the ATN signal will terminate these commands. The termination will occur immediately or at the end of the current phase based on the SE (Stop Enable) bit in the SCSI Operation register. Refer to the SE bit description for other conditions which will terminate these commands.

The number of bytes transferred over the SCSI bus is determined by the product of the SCSI Block Size register and the SCSI Block Count register.

After command termination, the SCSI bus control signals will be left in the state used by the command.

TABLE 4.20. SCSI Control Signals

Command	Bus Phase	C/D	I/O	MSG
Receive Command	Command	1	0	0
Receive Data	Data Out	0	0	0
Receive Message	Message Out	1	0	1
Receive Unspecified	Undefined	0	0	1
Send Status	Status	1	1	0
Send Data	Data In	0	1	0
Send Message	Message In	1	1	1
Send Unspecified	Undefined	0	1	1

Note: 1 = Asserted, 0 = Deasserted

Transfer Info (Opcode = 3Ch)

This command is used to transfer information over the SCSI bus, similar to the Receive and Send commands. However, this command is used only in the Initiator mode. The information transferred will be to or from the buffer memory depending on the state of the I/O signal.

The transfer type (Receive, Send, Command, Data, Message) is determined by the control signals present on the SCSI bus listed in Table 4.20.

The correct transfer mode (synchronous or asynchronous) must be set in the Synchronous Transfer register prior to issuing this command. The transfer mode is not automatically selected by the bus phase. If a non-data phase is being used, be sure to set for asynchronous transfers.

In any transfer type other than Mesasge In, the command completes with ACK deasserted. In the case of a Mesage In transfer (I/O = 1, C/D = 1, MSB = 1), ACK is left asserted upon the last byte transferred. When a new Transfer Info or Transfer Pad command is issued, the ACK signal will become deasserted. Since the ACK signal remains asserted after the Message In phase, the target is prevented from starting a synchronous data transfer before a new Transfer Info command has been issued. This will prevent a FIFO overflow. This handling of ACK will also allow the software to accept or reject a message.

If the Target chooses to terminate the command, perhaps by changing the phase lines, before the normal termination, an ''Uncompleted Command'' interrupt will be generated.

The number of bytes transferred over the SCSI bus is determined by the product of the SCSI Block Size register and the SCSI Block Count register.

Transfer Pad (Op-Code = 38h)

This command is identical to the Transfer Info command, except the data transferred on the SCSI bus will not be read or written to the buffer memory.

For Send type transfers, the last byte transferred to the Target in Automatic mode or a byte loaded into the SCSI Data register is repeatedly sent. For Receive type transfers, the DP8496/7 will blindly receive the bytes. The data will not be transferred to buffer memory. Parity is checked if enabled in the Setup 2 register.

The number of bytes transferred over the SCSI bus is determined by the product of the SCSI Block Size register and the SCSI Block Count register.

This command may be useful if the Target requests more information than the Initiator has to give it.

Send Disconnect (Opcode = 33h)

The DP8496/7 will change to the Message In phase and transfer a Save Data Pointers message followed by a Disconnect message. The bus will then be released to Bus Free phase. This command can be used to temporarily interrupt a data transfer and prepare the initiator for a subsequent reconnection.

If the ATN signal is asserted, this command will terminate immediately with an ''Uncompleted Command'' interrupt.

Send End (Opcode = 31h)

While connected as a Target, the Send End command will change to Status phase, send good status, change to Message In phase, and transfer a Command Complete message. The bus will then be released to Bus Free phase. This command would be used to end a complete SCSI Command after the Data Phase.

If the ATN signal is asserted, this command will terminate immediately with an ''Uncompleted Command'' interrupt.

Reselect/Receive Data (Opcode = 1Fh)

This combination command is identical to issuing the Reselect command (10h), followed by a single byte Message In phase using the Identify register contents, followed by the Receive Data Command (3Fh).

The Message In phase is unique. Instead of transferring data from buffer memory, a single byte is transferred from the identify register. The Identify register must be initialized before this command is issued.

Time-out on Reselection can be monitored by checking the Status Code in the SCSI Status register.

Reselect/Send Data (Opcode = 1Bh)

This combination command is identical to issuing the Reselect command (10h), followed by a single byte Message In phase using the Identify register contents, followed by the Send Data command (0Bh).

Reselect/Receive Data/Disconnect (Opcode = 1Ch)

This combination command is identical to issuing the Reselect command (10h), followed by a single byte Message In phase using the Identify register contents, followed by the Receive Data command (3Fh), followed by the Send Disconnect command (33h).

4.0 Functional Description (Continued)

All the normal interrupts will be generated during the data transfer phase of the command (Block Transfer Complete, Group Transfer Complete, etc.). In addition, a ''Group Complete'' interrupt will always be generated just before the Disconnect operation begins. This will allow the processor to process the information read from the SCSI bus while the disconnect operation is taking place.

Reselect/Send Data/Disconnect (Opcode = 18h)

This combination command is identical to issuing the Reselect command (10h), followed by a single byte Message In phase using the Identify register contents, followed by the Send Data command (0Bh), followed by the Send Disconnect command (33h).

A ''Group Complete'' interrupt will always be generated just before the Disconnect operation begins. This will allow the processor to set up the SCSI Bus Controller for a new command while the disconnect operation is taking place.

Reselect/Receive Data/End (Opcode = 1Dh)

This combination command is identical to issuing the Reselect command (10h), followed by a single byte Message In phase using the Identify register contents, followed by the Receive Data command (3Fh), followed by the Send End command (31h).

A ''Group Complete'' interrupt will always be generated just before the Disconnect operation begins. This will allow the processor to process the information read from the SCSI bus while the disconnect operation is taking place.

Reselect/Send Data/End (Opcode = 19h)

This combination command is identical to issuing the Reselect command (10h), followed by a single byte Message In phase using the Identify register contents, followed by the Send Data command (0Bh), followed by the Send End command (31h).

A ''Group Complete'' interrupt will always be generated just before the Disconnect operation begins. This will allow the processor to set up the SCSI Bus Controller for a new command while the disconnect operation is taking place.

Wait-for-Select/Receive Command (Opcode = 36h)

This command should normally be issued any time there is no other SCSI bus data transfers in progress. This will allow new SCSI commands to be received by the DP8496/7 (configured as a target).

The DP8496/7 will wait until it recognizes itself being selected with BSY deasserted, SEL asserted, and its own ID present on the SCSI data bus. If the ATN signal was also asserted, a Message Out phase will be generated with an Identify Message. The Identify byte will be loaded into the Identify register, not buffer memory. If the ATN signal was not asserted the Command Phase will be entered immediately, without a Message Out phase.

If a Message Out phase is executing and the ATN signal remains asserted when the ACK signal is deasserted, the command is terminated with an ''Uncompleted Command'' interrupt. This indicates that the Initiator is sending an extended or multi-byte message.

After the first byte of the Command Phase has been read from the SCSI bus, the DP8496/7 will interpret the Group Number in bits 5, 6 and 7. This determines the total number of bytes in the SCSI Command Descriptor Block as shown in Table 4.21. The correct number of bytes are then read from the SCSI bus. The entire SCSI Command Descriptor Block will be transferred to buffer memory.

TABLE 4.21. Group Numbers

Group Number 7 6 5	# of Bytes in Command
0 0 0	6
0 0 1	10
0 1 0	10
1 0 1	12

Note: Any other group combinations are unknown length and will cause an ''Uncompleted Command'' interrupt.

A SCSI bus parity error or the assertion of the ATN signal will terminate this command. The termination will occur immediately or at the end of the current phase based on the SE (Stop Enable) bit in the SCSI Operation register. Also, during a Message Out phase, if the ATN signal goes low, the command will terminate based on the SE bit.

The command will terminate after the last Command Descriptor byte has been read. An ''Operation Complete'' interrupt will be generated. The SCSI Control signals will remain in the Command Phase.

The Clear Pending Command can be issued to terminate this command if the DP8496/7 has not yet been selected.

If the DP8496/7 is reselected as an Initiator (before it has been selected as a Target), the chip will respond properly as an Initiator on the SCSI bus as if there were no active command. The Wait-for-Select/Receive command will then terminate with a ''(Re)Selected'' interrupt and the ''Last Command Ignored'' bit set also.

Wait-for-Select/Receive Command/Disconnect (Op-Code = 37h)

This command is identical to the previous command, except with the addition of an optional Disconnect function at the end.

If the SCSI Command Descriptor Block that is read from the SCSI bus is a read disk command, the DP8496/7 will enter the ''Message In'' phase, send a ''Disconnect'' message and terminate with an ''Operation Complete'' interrupt. A read command is determined by the first byte of the Command Descriptor Block that was transferred during the Command Phase. If this byte is either a 08h, 28h, 48h, or A8h, then this is interpreted as a read command.

If it is not a read command, the command will terminate after the last Command Descriptor byte has been read. An ''Uncompleted Command'' interrupt will be generated. In this case the SCSI Control signals may not remain in the Command Phase.

Wait-for-Select/Send Busy (Opcode = 35h)

If no command can be accepted by the Target, this command will turn away all Selections in a clean manner with minimal processor overhead. This command operates the

4.0 Functional Description (Continued)

same as the Wait-for-Select/Receive command, except after receiving the Command Descriptor Block, the DP8496/7 will send a busy status, Command Complete message, and will disconnect. It will then generate an ''Operation Complete'' interrupt.

The Command Descriptor Block read from the SCSI bus will not be transferred to buffer memory.

This command will remain active until overwritten with another command. This command is unique in that it will re-enable itself after each successful ''Operation Complete'' interrupt. If the target needs to monitor selection attempts, the desired registers (Identify register) must be read by the processor before the next selection occurs.

Any other interrupt besides ''Operation Complete'' will pause this command. The Clear Pending Command should be issued before this interrupt is cleared to prevent the Wait-for-Select/Send Busy command from re-enabling itself again after the error is cleared.

Self Test Commands

The five self test commands operate exactly the same as the corresponding non-self test commands. However, the SCSI bus is held TRI-STATE® and the inputs are ignored. The SCSI outputs are still active internal to the chip and the SCSI inputs are emulated for the correct response. The other pins of the DP8496/7 (non-SCSI) operate normally.

Once the first self test command is issued, the DP8496/7 enters its self test mode. The Clear Pending Command must be issued to clear the self test mode. Be sure to issue the Disconnect Self Test command before leaving the self test mode to ensure that all the internal SCSI bus signals are deasserted.

(Re)Select Self Test (Opcode = 90h)

Operates the same as the (Re)Select command but with all necessary input handshaking internally generated on chip. This should be the first self test command issued.

Disconnect Self Test (Opcode = B2h)

Operates the same as the Disconnect command. This command should be issued before the self test mode is cleared.

Send Data Loopback (Opcode = 8Bh)

Operates the same as the Send Data command in the Target mode. The data is read from buffer memory and parity is checked if enabled. The data asserted internally on the SCSI bus may be monitored by reading the SCSI Data register. The asynchronous mode must be used.

Receive Data Loopback (Opcode = BFh)

Operates the same as the Receive Data command in the Target mode. The processor should initialize the SCSI Data register with a value before this command is issued. The contents of the SCSI Data register plus parity will be asserted on the internal SCSI data bus. The SCSI parity bit is calculated when a byte is written to the SCSI Data register by the processor. This value will be read and transferred to buffer memory. SCSI parity is checked if enabled. The asynchronous mode must be used.

Transfer Info Loopback (Opcode = BCh)

Operates the same as the Transfer Info command in the Initiator mode. The I/O bit in the SCSI Control register should be set high or low to indicate the direction of the transfer. The asynchronous mode must be used.

If receiving data from the SCSI bus, the processor should initialize the SCSI Data register with a value before this command is issued. The contents of the SCSI Data register plus parity will be asserted on the internal SCSI data bus. The SCSI parity bit is calculated when a byte is written to the SCSI Data register by the processor. This value will be read and transferred to buffer memory. SCSI parity is checked if enabled.

If sending data to the SCSI bus, the data is read from buffer memory and parity is checked if enabled. The data asserted internally on the SCSI bus may be monitored by reading the SCSI Data register.

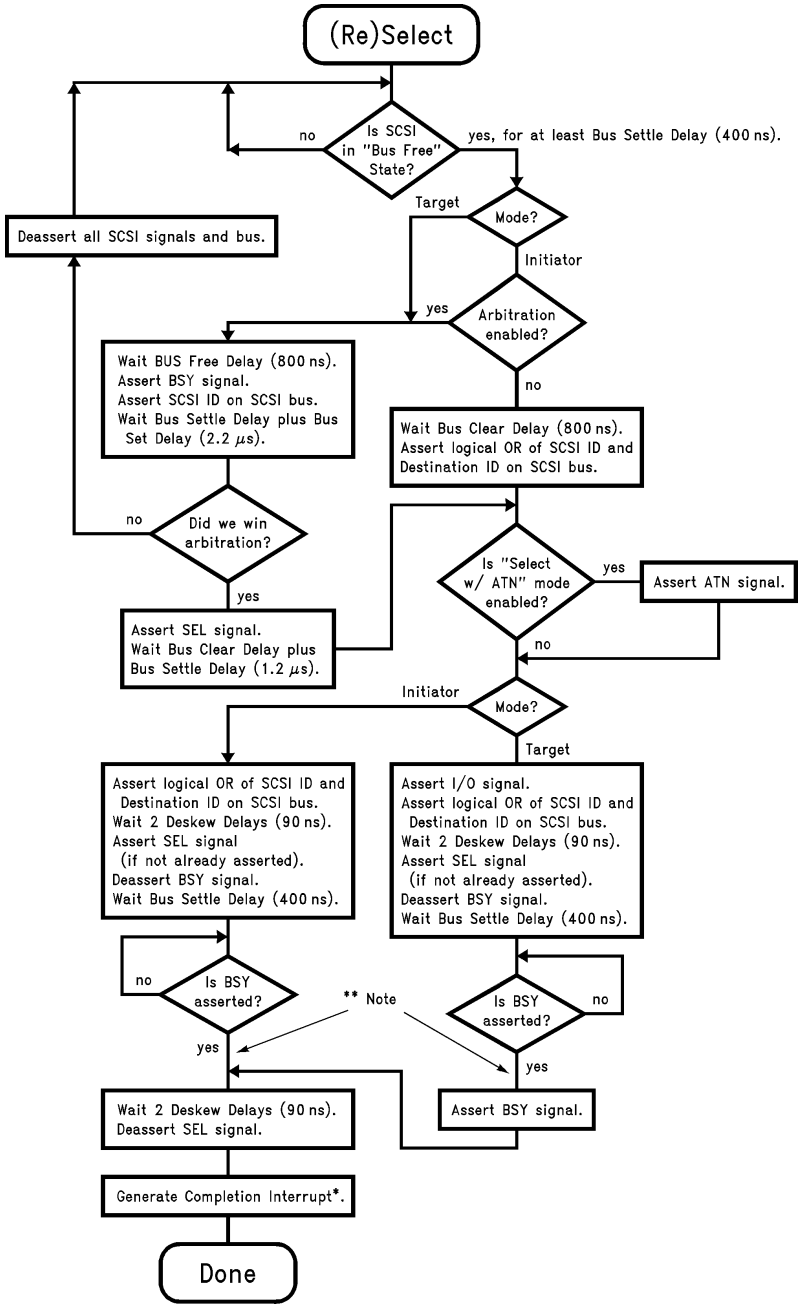
Forcing Parity Errors in Self Test Mode

In order to force a SCSI bus parity error in either a Receive Data Loopback or Transfer Info Loopback (with the I/O bit set to ''1'' in the SCSI Control register), the following sequence may be used.

1. Set the SPE (SCSI Parity Enable) bit in the Setup 2 register to a ''1''.
2. Issue (Re)Select Self Test.
3. Write a byte into SCSI Data register.
4. Switch SCSI Parity Polarity (the SPP bit in the Setup 2 register)
5. Issue a receive type loopback command.
6. Should obtain parity error in the SCSI Status register.

Command Flowcharts

The following flowcharts describe in detail the operation of the multiphase commands.

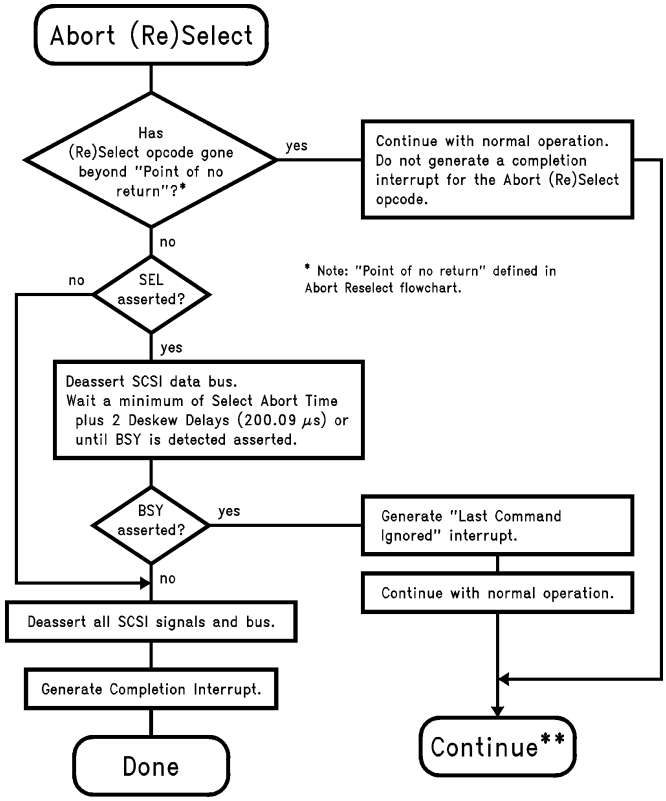


TL/F/11212–25

*If this routine called as a subroutine, the Completion Interrupt is not generated here.

**After reaching this point, the "point of no return", the Abort (Re)Select opcode will not function.

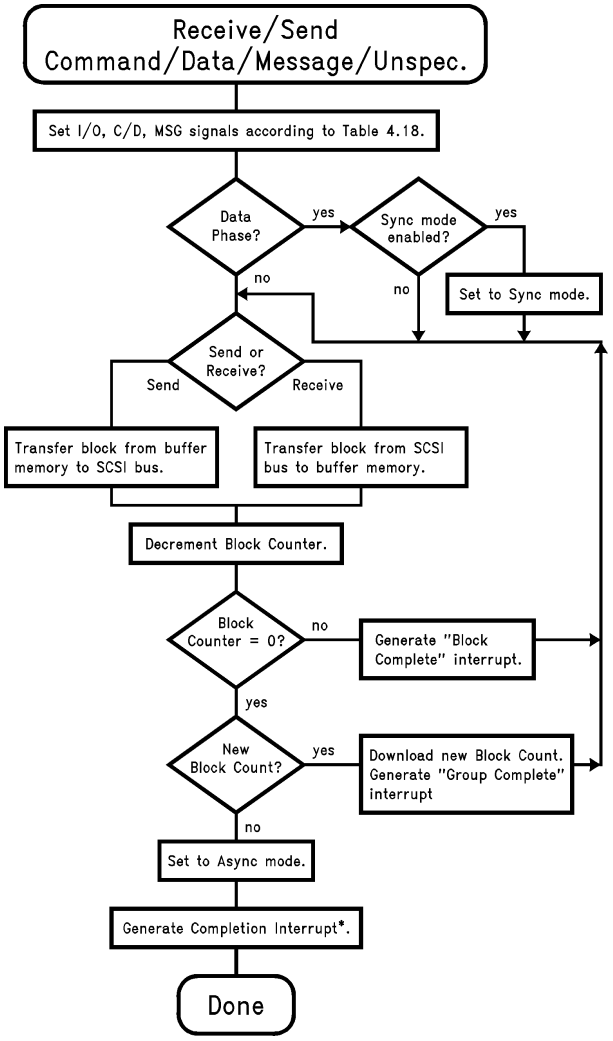
FIGURE 4.12. (Re)Select Flowchart



TL/F/11212–26

**Continue with whatever operation was occurring when Abort (Re)Select opcode was issued.

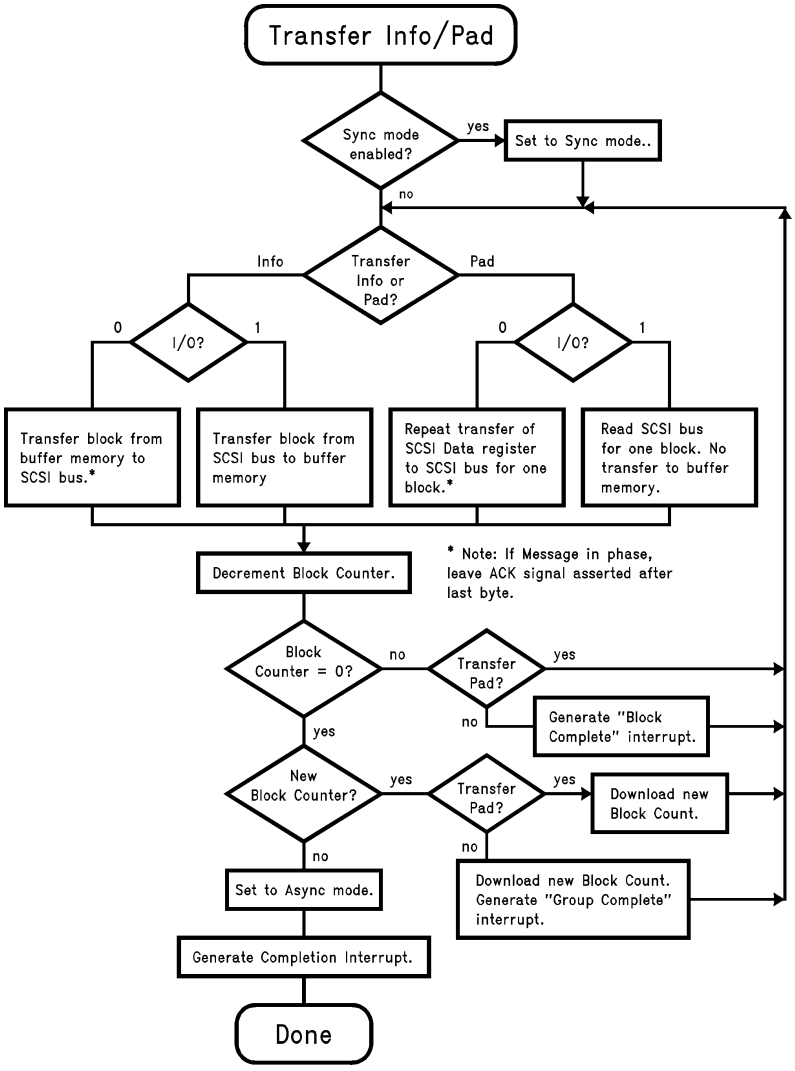
FIGURE 4.13. Abort (Re)Select Flowchart



TL/F/11212–27

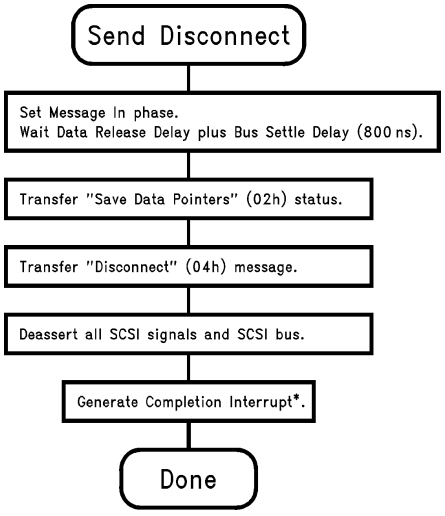
*If this routine called as a subroutine, the Completion Interrupt is not generated here.

FIGURE 4.14. Target Information Transfer Flowchart



TL/F/11212–28

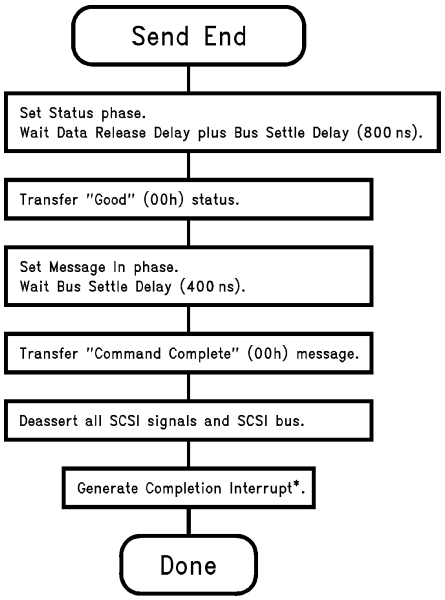
FIGURE 4.15. Initiator Information Transfer Flowchart



TL/F/11212–29

*If this routine called as a subroutine, the Completion Interrupt is not generated here.

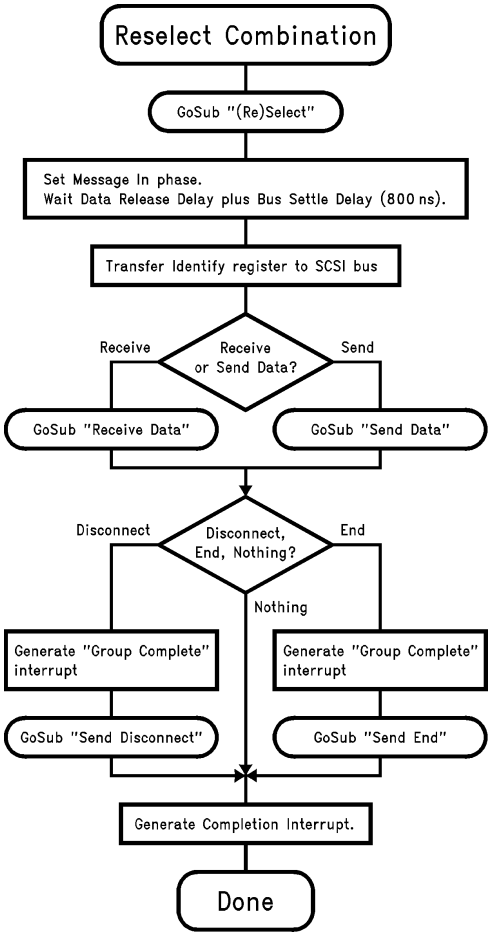
FIGURE 4.16. Send Disconnect Flowchart



TL/F/11212–30

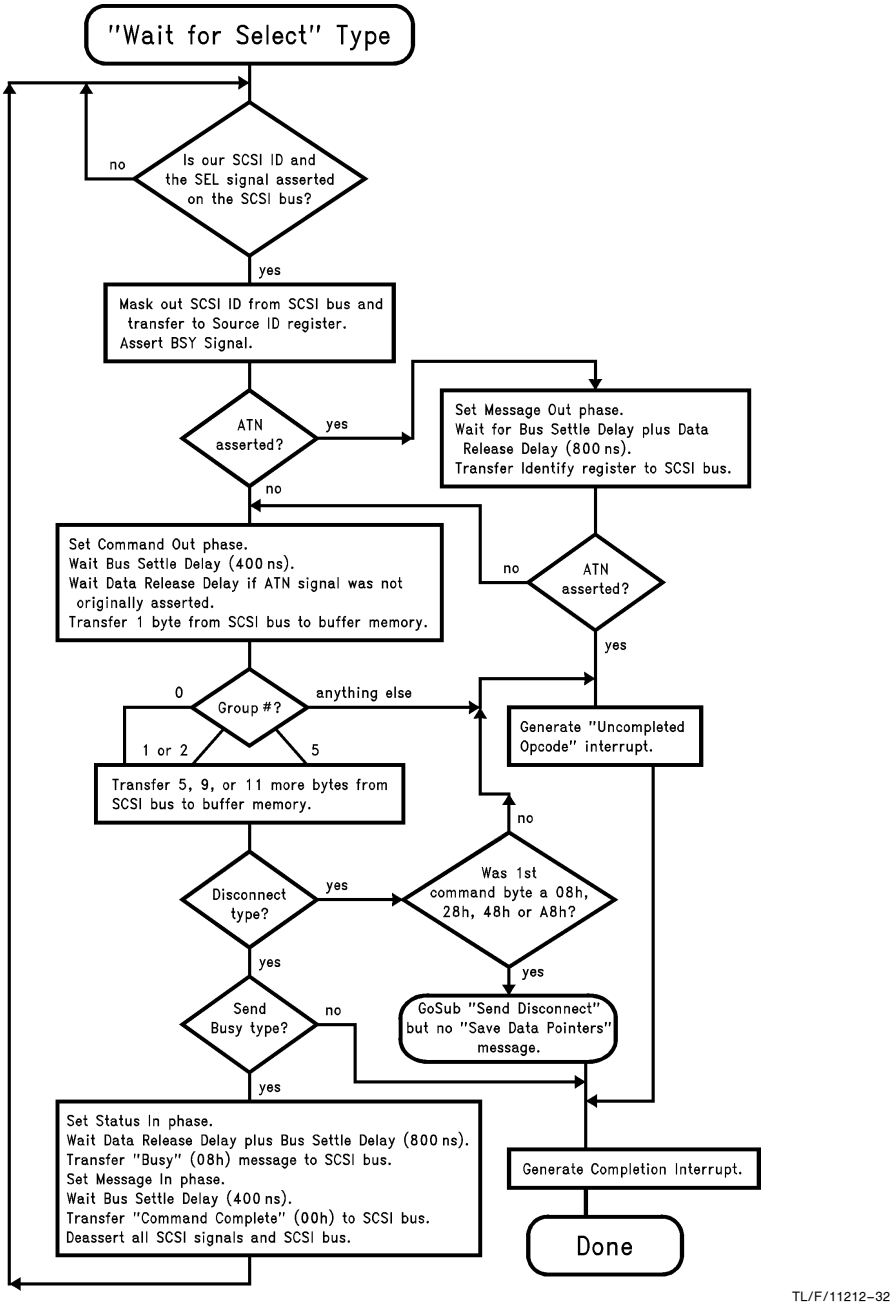
*If this routine called as a subroutine, the Completion Interrupt is not generated here.

FIGURE 4.17. Send End Flowchart



TL/F/11212–31

FIGURE 4.18. Reselect Combination Flowchart



TL/F/11212–32

FIGURE 4.19. "Wait for Select" Type Flowchart

SCSI Data (SDAT) 41h R/W

7	6	5	4	3	2	1	0
Data							

This register allows access to the SCSI Data Bus by the processor while in Manual Mode. See Section 4.4.3 for details about Manual Mode data transfers. These bits are asserted only if a ''1'' is written. On the single-ended SCSI bus that is open-drain, writing a ''0'' turns off the driver and allows the bus to float.

This register should only be used while in the Manual Mode. There are two exceptions:

1. Just before switching from Automatic to Manual Mode. See Section 4.4.3.
2. Just before a Transfer Pad command. See the Transfer Pad command description.

SCSI Control (SCTL) 42h R/W

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

The SCSI Control register is really an image of eight out of the nine SCSI control lines. The direction and R/W attributes of these bits are determined by the Target or Initiator mode and whether in Automatic or Manual operation. A summary of bit attributes are given in Table 4.22. When they are read, ATN, C/D, I/O, and MSG bits reflect the logical OR of what is on the SCSI bus and what the chip is attempting to drive. These two values may not be the same under some abnormal conditions. For example, on the DP8497, incorrect use of the Differential SCSI register in manual mode or an external circuit fault may cause such a conflict.

The SCSI RST signal does not appear in this register but is driven by the DP8496/7 from the SCSI Operation register. When RST is driven by another SCSI device it will generate a ''SCSI Reset'' interrupt.

Note the ATN line is writable in Automatic mode when configured as an Initiator. However, the SWA (Select With Attention) bit in the SCSI Operation register should be used to assert the ATN signal properly during the Selection phase. The ATN bit is only intended to be written during the Transfer Info or Transfer Pad commands.

These bits are asserted only if a ''1'' is written. Since the SCSI bus is open-drain, writing a ''0'' turns off the driver and allows the bus to float.

TABLE 4.22. Bit Attributes in SCSI Control Register

Sig.	Target			Initiator		
	Dir.	Auto	Man.	Dir.	Auto	Man.
BSY	I/O	R	R/W	I/O	R	R/W
SEL	I/O	R	R/W	I/O	R	R/W
ATN	I/O	R	R	O	R/W	R/W
C/$\overline{D}$	O	R/W	R/W	I	R	R
I/$\overline{O}$	O	R/W	R/W	I	R	R
MSG	O	R/W	R/W	I	R	R
REQ	O	R	R/W	I	R	R
ACK	I	R	R	O	R	R/W

SCSI Operation (SOP) 43h R/W

7	6	5	4	3	2	1	0
T/I	SWA	A/M	SE	A/H	APE	PEP	RST

This register must be loaded before SCSI Block Count and SCSI Block Size registers.

T/I: Target/Initiator

This bit must not be changed while an information transfer phase is in progress. It can be changed after an unexpected Selection or Reselection, but in these cases it must be changed before a transfer command is issued. It also can be changed while the SCSI bus is idle.

Caution: If the last command written to the SCSI Command register is legal in the current mode only (Target or Initiator), a Clear Pending Command should be issued before the (T/I bit is modified. Even if the last command has completed and the interrupt has been processed, the Clear Pending Command should be issued. If this is not done, an ''Invalid Command'' interrupt will be generated after the T/I bit is changed.

1 Target Mode. All commands and modes relevant to the Target functions are enabled. ((Re)Select command becomes Reselect)

0 Initiator Mode. All commands and modes relevant to the Initiator function are enabled. ((Re)Select command becomes Select)

SWA: Select With Attention

1 When in the Initiator mode and Selecting, the ATN signal will be asserted as per SCSI specifications.

0 When in the Initiator mode and Selecting, the ATN line will not be asserted.

When executing Transfer Info/Pad commands, ATN can be controlled through the SCSI Control register. In addition, when executing Transfer Info/Pad commands while executing any Information In phase, ATN will be asserted on a SCSI bus parity error if enabled in the Setup 2 register.

A/M: Automatic/Manual Mode

1 Enables the Automatic mode of operation, enabling the SCSI sequencer. SCSI commands may be issued to the SCSI Command register. The chip will respond properly to (Re)Selections.

0 Manual Mode completely disables the SCSI sequencer and clears any curent SCSI command. No SCSI commands may be issued in this mode, but it has no effect on the Disk Data Controller. In this mode the processor has complete control over the SCSI bus and any sequence, legal or illegal, can be performed.

Automatic to Manual Restrictions: Before switching from Automatic mode to Manual mode, the T/I (Target/Initiator) bit in this same register should be set to the desired value. Also the HE (Handshake Enable) bit in the Synchronous Transfer register should be programmed. If the DP8496/7 is not currently connected to the SCSI bus, the HE bit should be set to ''0''. This guarantees that all 18 SCSI signals will remain glitchless during the A/M bit transition. If the HE bit is set to ''1'', only the 9 SCSI Bus Control signals are guaranteed to remain glitchless.

The SCSI Data register should be programmed with the desired value to be driven on the SCSI bus. If the bus should not be driven immediately, a zero should be written to the SCSI Data register and the SPP (SCSI Parity Polarity) bit in the Setup 2 register should be set to ''1'' for even parity to prevent the SCSI parity bit being asserted on the SCSI bus.

If the DP8496/7 is currently connected, there should be no SCSI bus activity during the A/M bit transition.

Immediately after switching to Manual mode a command complete interrupt will be generated.

Manual to Automatic Restrictions: Before switching from Manual mode to Automatic mode, a Clear Pending Command should be written to the SCSI Command register. The DP8496/7 must be disconnected.

SE: Stop Enable

1 For the following conditions the current command will be terminated immediately. There will be a ''SCSI Bus Error'' interrupt for a SCSI parity error, or an ''Uncompleted Command'' with ''Attention Detected'' for the ATN signal.

1. In Initiator or Target mode, a SCSI parity error is detected while receiving information.
2. While in Target mode, the ATN signal is detected going high during any SCSI transfer,

If in Target mode, during a Message Out phase, and the ATN signal becomes deasserted before the leading edge of the REQ signal for the last byte, the command will terminate immediately with an ''Uncompleted Command'' interrupt.

0 Command will terminate immediately as desribed above only for the following situations while in Target mode:

1. SCSI parity error during Command Phase.
2. SCSI parity error during Message Out Phase.
3. ATN going high during Status Phase.
4. ATN going high during Message In Phase.

In all other situations the command will terminate at the end of the current phase instead of immediately. The reported interrupts will be the same as above.

If in target mode, during a Message Out phase, and the ATN signal becomes deasserted before the leading edge of the REQ signal for the last byte, the command will terminate at the end of the Message Out phase with an ''Uncompleted Command'' interrupt.

A/H: Active/Holding

1 The processor has access to the Active set of Disk and SCSI pointers and block counts. **Caution:** the contents of the Active registers may change as they are being read if data transfers are currently taking place. See Section 4.2.6 for more information.

0 The processor has access to the Holding set of registers. This is the normal mode of operation.

APE: Arbitration Phase Enable

This bit is only significant while in Automatic mode as an Initiator. This bit should only be modified when there are no active or pending commands and while disconnected.

1 The Select command will contain an Arbitration Phase between detection of bus free and the Selection Phase.

0 The Select command will not contain an Arbitration Phase.

In Target mode, an Arbitration Phase will always be executed for Reselected and Reselect combination commands.

The APE bit is also used to clear an ''Invalid Command'' condition. Clearing an ''Invalid Command'' interrupt is a two-step process. First, the APE should be modified twice (set, then reset, or reset, then set). Second, clear the interrupt by writing the three-bit pattern for ''Invalid Command'' (100) back to the SCSI Interrupt register.

PEP: Parity Error to Processor

If the Setup 2 register is configured to check buffer memory parity, a parity error detected while the processor reads the Buffer Memory Data register will set this bit. This bit may be read after a block of data has been read from the Buffer Memory Data register. Once this bit is set, it can only be cleared by reading this register. It will not be cleared by the SCSI Reset command.

RST: SCSI Bus Reset

1 The SCSI RST signal will be held asserted as long as this bit is set. This is valid in any mode; Initiator, Target, Automatic or Manual. Be sure to keep this asserted for the minimum time required by the SCSI specification (25 μs).

0 Normal operating mode. The SCSI RST signal will not be asserted by the DP8496/7.

Reading this bit will simply reflect the last value written to this bit, not the current state of the SCSI RST signal. A ''SCSI Reset'' interrupt will be generated if the SCSI RST signal is asserted. This interrupt is generated whether the RST signal was asserted by the DP8496/7, or by any other source.

Synchronous Transfer (SYNC) 44h **R/W**

7	6	5	4	3	2	1	0
HE	Transfer Period			Offset			

Synchronous transfers are only legal in the Data Phase while in Automatic mode.

Adequate memory bandwidth should be provided for full speed disk and SCSI transfers. If the bandwidth is inadequate and the DP8496/7 is receiving SCSI data, overruns may occur (this would be reported in the SCSI Status register). If the DP8496/7 is sending SCSI data, the data may not be sent at full speed.

HE: Handshake Enable

This bit is valid only in the Manual mode and further defines the way the SCSI Data register operates.

1 This mode is used to transfer single bytes over the SCSI bus using the DBR (Data Buffer Ready) bit in the SCSI Status register. The processor should poll the DBR bit for the proper state depending on the data direction. The processor can then read or write to the SCSI Data register. In this mode, the REQ and ACK signals handshake properly for every byte. As an application, message and status phases could be implemented this way. HE should be set to ''1'' only after the SCSI Operation register and SCSI Control register are loaded correctly.

0 The SCSI Data register is simply a transparent latch between the SCSI bus and the processor. The REQ and ACK signals are not automatically asserted REQ and ACK (and other SCSI control signals) may be modified by the processor through the SCSI Control register.

Parity is not verified when reading data from the SCSI bus. Parity is always generated when writing to the SCSI bus.

For information transfers, it is not recommended to use HE = 0, because parity is not checked.

Transfer Period

This field determines the time between adjacent Synchronous Data transfers. It is also used to choose between Synchronous and Asynchronous Transfer modes.

A value of zero will give Asynchronous Transfers with fast handshaking between REQ and ACK. A value of one will give Asynchronous Transfers an extra BCLK period per REQ and ACK handshake. (It is also possible for the user to set the setup time delay between REQ and ACK and the SCSI data signals by setting the SCK1 and SCK0 bits of the Differential SCSI 2 register.)

A value of two to seven enables Synchronous Transfers with a Transfer Period (TP) equal to BCLK period multiplied by the value programmed (2–7).

While in the Target Mode, if a non-Data Phase is executing, Asynchronous transfers will be used, regardless of the value in this field. This is done because Synchronous transfers are only allowed during the Data In Phase and the Data Out Phase.

Caution: While in Initiator Mode, the DP8496/7 will NOT automatically switch between Synchronous and Asynchronous modes. This must be switched by the processor.

Table 4.23 summarizes the transfer mode used based on this field and the phase in progress.

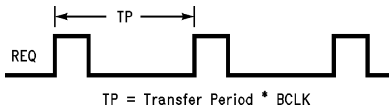
The value in this field, along with the BCLK frequency, also determines the DP8496/7s adherence to other synchronous SCSI timing specifications—REQ assertion period (90 ns, 30 ns for Fast), REQ negation period (90 ns, 30 ns for Fast), DATA setup time (55 ns, 25 ns for Fast), and Data Hold Time (100 ns, 35 ns for Fast). The equations for these times are specified in the A.C. Specification section of this datasheet—in subsections 6.29 thru 6.32. Maximum BCLK is specified in subsection 6.1.

TABLE 4.23. Transfer Mode and Period

Trns. Prd.	Current SCSI Phase of DP8496/DP8497	
	Target, Data In Target, Data Out Initiator, All Phases	Target, All Phases EXCEPT Data In and Data Out
0	Async	Async
1	Slow Async	Slow Async
2	Sync, TP = 2*BCLK	Async
3	Sync, TP = 3*BCLK	Async
4	Sync, TP = 4*BCLK	Async
5	Sync, TP = 5*BCLK	Async
6	Sync, TP = 6*BCLK	Async
7	Sync, TP = 7*BCLK	Async

Notes: TP = Transfer Period.

See AC Timing for Async vs. Slow Async.



TL/F/11212–33

FIGURE 4.20. Synchronous Transfer Period

Offset

The offset value specifies the number of outstanding bytes allowed before handshaking is inhibited. A value of zero results in an offset of 16.

If the Block Count register is greater than one, or if the Block Count is pipelined, the offset value must not be greater than the Block Size registers. Otherwise, data may be written to buffer memory incorrectly. If the Block Count is one, and there is no pipelining, then there is no restriction on the value of the offset.

Identify (IDENT) 45h R/W

7	6	5	4	3	2	1	0
I	DISC	TAR	Reserved	LUN			

The Identify register is used only with Reselect combination and ''Wait for Select'' type commands. This is actually a simple 8-bit register with no fixed structure. All of the bits may be read and written by the processor. However, in most applications the bit definitions should correspond with the ANSI SCSI specification.

The Reselect combination commands use the contents of the Identify register during the Message In phase following Reselection. The processor must load the Identify register before a Reselect combination command is issued.

The ''Wait for Select'' commands modify the Identify register. This byte is received from the Message Out phase after a connection. The data here is valid as soon as the message phase is completed, although the processor is not notified with an interrupt until the entire command is complete. The Identify register is overwritten upon the next ''Wait for Select'' operation or when the processor loads the Identify register preceding a Reselect combination command.

Caution: This register must be initialized with a valid value after a reset before any command is written to the SCSI Command register. Otherwise an ''Uncompleted Command'' interrupt will be generated. A valid value is any value with bit 7 set and bits 3 and 4 cleared.

I: Identify Message

Since this register is used only for the Identify Message, this bit must always be set to ''1''. If it is not set to ''1'', no new command can be executed.

This bit will always be a ''1'' when read following a successful ''Wait for Select'' operation which included a Message Out phase. Otherwise, an ''Uncompleted Command'' interrupt would be generated.

DISC: Disconnect Privilege

When this register is used for the Message In phase during a Reselect combination command, this bit has no meaning.

Following a ''Wait for Select'' command which included a Message Out phase, this bit indicates the ability of the Initiator to support disconnection and reconnection. A ''1'' indicates this support. A ''0'' indicates no support.

TAR: Logical Unit Target

When this register is used for the Message In phase during a Reselect combination command, this bit has no meaning.

Following a ''Wait for Select'' command which included a Message Out phase, this bit specifies where the Identify Message is directed to. A ''0'' indicates a logical unit (see the LUN field). A ''1'' indicates a target routine. See the SCSI-2 specification.

Reserved

Since these bits are reserved, it is recommended that these bits be set to ''0'' before a Reselect combination command is issued.

These bits will always be a ''0'' when read following a successful ''Wait for Select'' operation which included a Message Out phase. Otherwise, an ''Uncompleted Command'' interrupt would be generated.

LUN: Logical Unit Number

Before a Reselect combination operation, this field should be set with the Logical Unit Number from which the data is being read from or being written to.

Following a ''Wait for Select'' command which included a Message Out phase, this field indicates the Logical Unit Number associated with the command to follow. They should match the LUN bits contained in the Command Descriptor Block.

Destination ID (DID) 46h R/W

7	6	5	4	3	2	1	0
Phase Code				MCH	Destination ID		

Phase Code

These four bits set up the SCSI bus phase for which the DP8496/7 will match or not match to generate the Phase Compare interrupt. Comparison is qualified with the active edge of the REQ signal in both match and mismatch cases.

The Phase Code bit patterns are exactly the same as the Status Code patterns defined in Table 4.25 in the SCSI Status register.

MCH: Match

1 Looks for a match with the indicated Phase Code. If found, a ''Phase Compare'' interrupt will be generated.

0 Looks for a mismatch of the current SCSI bus phase with the indicated Phase Code. This code should be loaded after the Target's phase is stable (REQ is asserted) if the DP8496/7 is in Initiator mode. If the phase changes, a ''Phase Compare'' interrupt will be generated.

Destination ID

This ID is OR'ed with SCSI ID from the Setup 2 register. This OR'ed value is asserted on the SCSI bus during Selection or Reselection phases. Parity is also generated.

Source ID (SID) 47h R Only

7	6	5	4	3	2	1	0
VID	x	x	x	x	Source ID		

While in Target mode during the Selection Phase, or while in Initiator mode during the Reselection Phase, the Initiator's ID and the Target ID are both present on the SCSI bus. Our Target ID (SCSI ID in Setup 2 register) is masked off, and the resulting Initiator ID can be read from Source ID field. The VID (Valid ID) bit will be ''1''.

If there is no Initiator ID on the SCSI bus during Selection, the VID bit will be ''0''. If the DP8496/7 is not properly Selected (ex. three or more bits set), the DP8496/7 will not respond and the VID bit will be ''0''.

SCSI Status (SSTAT) 48h R Only

7	6	5	4	3	2	1	0
DBR	BMR	Error Code		Status Code			

DBR: Data Buffer Ready

In Manual mode with the HE (Handshake Enable) bit set to ''1'', DBR is used to indicate when to send or receive the next byte.

1 The SCSI Data register is full. If receiving, the SCSI Data register should be read. If transmitting, the processor should wait until DBR = 0.

0 The SCSI Data register is empty. If receiving, the processor should wait for another byte to arrive. If transmitting, the SCSI Data register should be written.

The DBR bit is reset whenever an error is cleared in the SCSI Interrupt register. An error is defined as a Completion Interrupt Code value between 2h and 7h in the SCSI Interrupt register.

BME: Buffer Memory Parity Error

A parity error was detected while writing data to the SCSI port from buffer memory. This indicates a parity error in buffer memory. This will terminate any SCSI command at the end of the current transfer phase.

The ''Buffer Memory Parity Error'' completion interrupt will be generated.

This bit is cleared to zero when a new command is downloaded and executed.

4.0 Functional Description (Continued)

Error Code

The error code is used in conjunction with the Status Codes to determine the present state of the SCSI interface. In the case of a SCSI Data Overrun error occurring at the same time as a SCSI parity error, the more serious error, SCSI Data Overrun, will be encoded. In the case where the SE bit of the SCSI Operation register is set and therefore a SCSI parity error causes the current command to terminate immediately, a SCSI data overrun may be caused. In this case also the error code for SCSI Data Overrun will be reported, effectively masking the original SCSI parity error from being reported.

TABLE 4.24. Error Codes

Error Code 5 4	Error
0 0	No Error
0 1	SCSI Parity Error
1 1	SCSI Data Overrun

The Error Code field is cleared to zero when a new command is downloaded and executed.

SCSI Parity Error (0 1)

This code is present in any Selection or Information transfer phase upon detection of a parity error from the SCSI bus in either Auto or Manual mode.

SCSI Data Overrun (1 1)

This code is set if the DP8496/7 receives too many bytes to transfer in the available bandwidth to buffer memory. Data loss has occurred! If this error occurs in any transfer mode, it indicates a lack of available buffer memory bandwidth. If it occurs during an Asynchronous transfer, some AC timing specification has been violated—such as too fast a REQ/ACK period for a given BCLK. There must be at least 2 BCLK periods per REQ or ACK period.

Also set if 2 REQ pulses are received before a Transfer Info or Transfer Pad command in Initiator mode is issued by the processor. This indicates that the Target device is in Synchronous Transfer mode and the DP8496/7 is not set up to respond in this mode.

During synchronous transfers, if the number of outstanding REQ or ACK pulses received is greater than the programmed offset (offset overflow), this error will be set.

Also set if there are any abnormal handshakes between REQ and ACK. Examples may be an ACK which deasserts before REQ or any non-match of REQ/ACK pulses following a Synchronous mode transfer.

Status Code

This field indicates present bus condition and in the case of bus free phase, how the DP8496/7 arrived in the phase. All bits are latched on the leading edge of the $\overline{RD}$ strobe and will not change during the read cycle.

Note that the state of the ATN signal can be read at any time in any mode through the SCSI Control register.

The Status Code is not valid while in the Manual mode. The Status Code is also not valid after an ''Invalid Command'' interrupt has been generated.

Important: In Initiator mode, information transfer phases are updated on the leading edge of REQ. If, during a Transfer Info or Transfer Pad command, the phase changes from that detected on the first REQ, the command will be aborted with an ''Uncompleted Command' interrupt if there is no parity error.

TABLE 4.25. Status Codes

Status Code 3 2 1 0	Comment
0 0 0 0	Bus Free or Idle due to Reset command, $\overline{CRST}$ pin, SCSI bus reset, normal disconnection for initiator, or Abort (Re)Selection command.
0 0 0 1	Bus Free due to command complete (except after Abort (Re)Select command or Reset command). Target mode only.
0 0 1 0	A DP8496/7 initiated Arbitration phase is pending or in progress.
0 0 1 1	A DP8496/7 initiated (Re)Selection phase is in progress.
0 1 0 0	Selected. The DP8496/7 has been Selected as a Target.
0 1 0 1	Reselected. The DP8496/7 has been Reselected as an Initiator and the ''Reselected'' interrupt completion code has not yet been cleared.
0 1 1 0	Unexpected bus free in Initiator mode. Command has been aborted.
0 1 1 1	Connected, but no pending command
1 0 0 0	Data Out Phase
1 0 0 1	Unspecified Info Out Phase
1 0 1 0	Command Phase
1 0 1 1	Message Out Phase
1 1 0 0	Data In Phase
1 1 0 1	Unspecified Info In Phase
1 1 1 0	Status Phase
1 1 1 1	Message In Phase

While in the Initiator mode, the Status Code will be frozen at the time that an error completion code is generated (not code ''001''). When the completion interrupt is cleared, the Status Code will return to normal operation. A completion code of ''001'' will not freeze the Status Code.

SCSI Interrupt (SINT) 49h R/W

7	6	5	4	3	2 1 0
PCMP	ATN	GTC	BTC	LCI	Completion Interrupt Code

This register indicates interrupts that have occurred. There are two types of interrupts, checkpoint and completion.

Checkpoint interrupts (bits 3–7) are separated into separate bits and may be asserted simultaneously. These interrupts may be cleared only by writing a ''1'' into the corresponding bit location. These bits may be cleared individually or simultaneously.

Completion interrupts are encoded into bits 0–2. Only one completion interrupt may be read at a time. Completion interrupts may be buffered behind each other if more than one completion interrupt has occurred. See Table 4.27 for a description of buffered interrupts. Completion interrupts may be cleared only by writing the same three bit pattern back to the SCSI Interrupt register. This will clear that interrupt and allow a new or buffered completion interrupt to occur. A completion interrupt may be cleared simultaneously with other checkpoint interrupts.

Interrupts will be reported in the SCSI Interrupt register always, independent of the SCSI Interrupt Enable register.

If enabled interrupts remain after writing to the SCSI Interrupt register, the $\overline{SINT}$ pin will deassert momentarily to retrigger an edge sensitive interrupt input to the processor.

This register is latched to prevent the contents from changing while reading the register.

PCMP: Phase Compare

Set when the Phase Code selected in the Destination ID register either matches (MCH = 1) or differs (MCH = 0) from the current phase of the SCSI bus. The MCH (Match) bit is also found in the Destination ID register.

Application Hint: One use of this interrupt is to inform the processor of the start of data transfer (a Match of Data In or Out phase) and thus the point at which a new SCSI pointer and block count can be loaded on a combination type command. If in Initiator mode, the mismatch mode could be set up to notify the processor if the phase changes from the current phase.

ATN: Attention Detected

1 Set on assertion of the ATN signal when DP8496/7 is configured as a Target in Automatic mode during an information transfer phase. Also set on assertion of the ATN line during selection phase unless executing a ''Wait for Select'' type command.

The current command will terminate immediately or at the end of the current phase depending on the state of the SE (Stop Enable) bit in the SCSI Operation register. The SCSI Control register can be used to monitor when the ATN signal goes away.

0 ATN signal not detected.

GTC: Group Transfer Complete Interrupt

This interrupt indicates that the SCSI Block Count register may be reloaded. This occurs at the end of a block transfer where the block counter has reached zero, and a new block count has been downloaded. This interrupt will only be set if the SCSI Block Count register has been written to since the last download of the SCSI Block Count.

This bit is not asserted during a Transfer Pad command.

In addition, while executing a Reselect Combination command that ends with a Disconnect or Send End option, the Group Complete Interrupt will be asserted after the last block of data has been transferred (before sending any Status or Message information).

BTC: Block Transfer Complete Interrupt

This interrupt indicates the Complete transfer of one block, as defined by the SCSI Block Size registers reaching zero. This bit will not be set when the GTC (Group Transfer Complete) bit or any completion interrupt code is asserted. It is also not asserted for a Transfer Pad command in initiator mode.

LCI: Last Command Ignored

Can occur on the following conditions:

1. When a ''Wait for Select'' type command is overwritten with another command, the new command will not be executed if the DP8496/7 has started to respond to the anticipated Selection from another SCSI device.
2. A (Re)Select or Reselect combination command may be ignored if the DP8496/7 has been or is in the process of being (Re)Selected by another SCSI device. This will also produce a ''(Re)Selected'' completion interrupt.
3. If the Abort (Re)Select command is issued and the DP8496/7 has already started a (Re)Selection phase and received a response from the other device, the (Re)Selection phase will not be aborted and the LCI bit will be set.
4. Any ''Wait for Select'' type command is ignored if the DP8496/7 has been or is in the process of being reselected by another SCSI device. This will also produce a ''(Re)Selected'' completion interrupt.
5. A new command is loaded before a ''(Re)Selected'' interrupt is cleared.

Completion Interrupt Code

Table 4.26 lists the possible completion interrupts. While in Manual mode, only codes ''000'', ''010'', and ''111'' may be generated.

If an error completion code is generated, even if it is buffered, no new operations will be performed. When the error completion code is cleared, new SCSI commands may be executed. See the end of this register description for an explantion of buffered interrupts.

TABLE 4.26. Completion Interrupt Codes

Comp. Int. Code 2	1	0	Cause
0	0	0	No Interrupt
0	0	1	No Error
0	1	0	SCSI Bus Error*
0	1	1	Buffer Memory Parity Error
1	0	0	Invalid Command
1	0	1	Uncompleted Command
1	1	0	(Re)Selected
1	1	1	SCSI Bus Reset Received or Generated

*SCSI Bus Error can either be parity or data lost. Read SCSI Status register to determine which.

No Error (001)

This is the normal completion interrupt of SCSI commands. This is presented upon the successful completion of a SCSI command. The phase of the SCSI bus depends on the command executed and can be found by reading the SCSI Status register.

SCSI Bus Error (010)

While in Automatic mode, this interrupt indicates one of two possible conditions. Either the DP8496/7 observed a SCSI parity error, or data was lost in the synchronous transfer mode due to lack of bandwidth to the buffer memory. The exact error can be determined by reading the Error Code field in the SCSI Status register.

4.0 Functional Description (Continued)

This interrupt will be presented after the command has completed.

While in Manual mode, this error can only indicate a SCSI parity error.

Buffer Memory Parity Error (011)

A parity error was detected while writing data to the SCSI port from buffer memory. This indicates a parity error in buffer memory. This will terminate any SCSI command at the end of the current transfer phase.

The BME (Buffer Memory Parity Error) bit in the SCSI Status register should be set to ''1'' also.

Invalid Command (100)

An invalid command for the present mode of the chip was received in the SCSI Command register. It is not guaranteed that all invalid commands will be detected. You should NEVER issue an invalid command.

Clearing an ''Invalid Command'' interrupt is a two-step process. First, the APE (Arbitration Phase Enable) bit in the SCSI Operation register should be modified twice (set, then reset, or reset, then set). Second, clear the interrupt by writing the three-bit pattern (100) back to the SCSI Interrupt register.

Uncompleted Command (101)

In Target mode, this interrupt occurs when an Initiator sends something unexpected. For example, the ATN signal remains asserted after the first mesage byte in combination commands, or something other than a SCSI read command is received in a Wait for Select/Disconnect command.

When this completion interrupt is generated, the SCSI Status register will freeze in the state in which the SCSI bus was in when the ''Uncompleted Command'' interrupt was generated. While in Initiator mode, the actual phase may now differ.

When the ''Uncompleted Command'' interrupt is cleared, the SCSI Status register will return to normal operation and will reflect the current status of the SCSI bus.

The unexpected events include:

- While executing a ''Wait for Select/Disconnect'' command, if a non-read SCSI Command Descriptor Block is received.
- While executing a ''Wait for Select'' type command, if a message byte other than Identify is received or if a reserved bit is not zero.
- While executing a ''Wait for Select'' type command, if the first byte of the Command Descriptor Block does not map to a Group 0, 1, 2, or 5 type command.
- While executing any command, if the ATN signal remains asserted after the trailing edge of ACK on the last byte of the Message Out phase.
- While executing a Message Out phase, if the ATN signal is deasserted before the leading edge of the REQ for the last byte.
- While executing any information transfer phase, if the ATN signal is asserted.
- While executing a Reselect combination command, if the ATN signal is asserted during the Reselection phase, if the SE (Stop Enable) bit in the SCSI Operation register is set to ''1''.
- While in a Message Out phase, if the ATN signal becomes deasserted after REQ is false and while ACK is true.
- While in a Message Out phase, if the ATN signal becomes deasserted after both REQ and ACK are false.

In the Initiator mode, this interrupt will occur for an unexpected bus free condition. It will also occur if the bus phase changes after the first REQ is received. The DP8496/7 records the SCSI bus phase when the Target asserts REQ. Any change of the bus phase with REQ asserted will immediately terminate the current command with this interrupt.

(Re)Selected (110)

This interrupt could be generated without any SCSI command being issued. The moment Automatic mode is enabled, the DP8496/7 is susceptible to being (re)selected. It is also possible to (re)select the DP8496/7 after a SCSI command has been issued by the processor. If the DP8496/7 loses Arbitration and the winning device chooses to select the DP8496/7, the chip will allow this and respond properly.

This interrupt will usually be unexpected and tell the processor that the DP8496/7 has responded to a Selection. As a Target, the ATN line must be checked after receiving this interrupt to determine if a Message Out phase is required. As an Initiator, this interrupt would indicate a Reselection from a Target.

If a ''Wait for Select'' type command is currently executing and waiting to be Selected, it will terminate with an ''Operation Complete'' interrupt, not a ''(Re)Selected'' interrupt.

SCI Bus Reset (111)

This interrupt could be generated without any SCSI command being issued. A reset signal greater than one BCLK period in length was received on the SCSI bus. The DP8496/7 is now in Bus Free phase and any command pending or executing has been terminated.

Buffered Interrupts

The 3-bit Interrupt Code field can only present one type of Interrupt at one time. There may be occasions where more than one interrupt has occurred. If this is true, then the first interrupt that occurred will be reported by the Interrupt Code field. The second interrupt will be buffered internally. When the first completion code is cleared, the buffered completion code will be reported in the Interrupt Code field. The $\overline{\text{SINT}}$ pin will pulse inactive for a short period of time between interrupts.

The exception to this is the ''No Error'' completion interrupt. If this interrupt is generated, it need not be cleared before another completion interrupt is generated. A new completion interrupt will overwrite any pending ''No Error'' completion interrupt. This is summarized in Table 4.27.

TABLE 4.27. Interrupt Buffering

Interrupt Code being generated from current command	Interrupt Code from previous command residing in: SCSI Interrupt register	Interrupt Code from previous command residing in: Buffered internally	Comments
non-error or error	none	none	When a command completes, the Interrupt Code is normally sent directly to the SCSI Interrupt register.
non-error	old non-error	none	If a non-error completion interrupt is pending, a new NON-ERROR interrupt will overwrite the old non-error interrupt.
error	old non-error	none	If a non-error completion interrupt is pending, a new ERROR interrupt will be buffered. The old non-error interrupt will remain in the SCSI Interrupt register until cleared.
error	old error	none	If an error completion interrupt is pending, a new ERROR interrupt will be buffered. The old error interrupt will remain in the SCSI Interrupt register until cleared.
non-error	old error	none	A completion interrupt will never be generated after an error interrupt is generated until the old error interrupt is cleared.

TL/F/11212–34

SCSI Interrupt Enable (SINTE) 4Ah R/W

7	6	5	4	3	2	1	0
PCMP	ATN	GTC	BTC	LCI	x	x	CI

This register simply enables selected interrupts to physically assert the $\overline{SINT}$ pin. If an interrupt is disabled via this register, all the effects of the interrupt remain the same except it will not affect the $\overline{SINT}$ pin.

Bits 3–7 enable individual checkpoint type interrupts. A ''1'' allows the individual interrupt to assert the interrupt pin. A ''0'' prevents it.

The CI (Completion Interrupts) bit allows all completion type interrupts to assert the interrupt pin. The completion interrupts cannot be enabled individually.

Interrupts should be cleared from the SCSI Interrupt register whether or not they are enabled in the SCSI Interrupt Enable register.

SCSI Block Count (SBC) 4Bh R/W

7	6	5	4	3	2	1	0
Count							

The value programmed in this register sets the number of blocks to be transferred. A value of 00h means 256 blocks. This value, in conjunction with the block size value in SBSH and SBSL registers, determines the total number of bytes to be transferred.

SCSI Block Size—High (SBSH) 4Ch R/W

7	6	5	4	3	2	1	0
Size—High Order Byte							

SCSI Block Size—Low (SBSL) 4Dh R/W

7	6	5	4	3	2	1	0
Size—Low Order Byte							

The size of the user's SCSI blocks is programmed in these two registers. This value, in conjunction with the block count value in the SBC register, determines the total number of bytes to be transferred.

Differential SCSI 1 (SDIF1) 4Eh R/W

7	6	5	4	3	2	1	0
DCSDB7	DCSDB6	DCSDB5	DCSDB4	DCSDB3	DCSDB2	DCSDB1	DCSDB0

Differential SCSI 1 (SDIF1) 4Fh R/W

7	6	5	4	3	2	1	0
SCK1	SCK0	DCBSY	DCSEL	DCTARG	DCINIT	DCRST	DCSDBP

(Other than the SCK1 and SCK0 bits, all other bits are meaningful only on the DP8497.)

DCSDB(7:0), DCSDBP

The polarity of each of these bits sets the polarity of the DSDB(7:0) and DSDBP pins of the DP8497 and also the direction of the on-chip transceiver at the associated signal pin; thus providing differential transceiver direction control to the user in manual mode operation. A ''1'' in any of these bits, and therefore a high level on the corresponding direction control pin, means the transceivers are in ''drive'' mode; while a ''0'' signifies the ''receive'' mode.

In effect only during manual mode.

DCBSY

Same as above. Affects the DBSY pin. In effect only during manual mode.

DCSEL

Same as above. Affects the SEL pin. In effect only during manual mode.

DCTARG

Same as above. Affects the DTARG pin which controls the direction of target signals: $\overline{C/D}$, $\overline{I/O}$, $\overline{REQ}$, $\overline{MSG}$. In effect only during manual mode.

DCINIT

Same as above. Affects the DINIT pin which controls the direction of initiator signals: $\overline{ACK}$, $\overline{ATN}$. In effect only during manual mode.

DCRST

Same as above. Affects the DRST pin. In effect only during manual mode.

SCK(1:0)

By programming these two bits the user can adjust the set-up time delay of $\overline{REQ}$ and $\overline{ACK}$ signals with respect to the SCSI Data signals. Delay values of approximately 1, 1.5, or 2 Bus Clock periods are possible, depending on the values of these bits and the CLK(1:0) bits of the Setup 1 register. Default values for these bits are 00.

TABLE 4.28. SCSI Strobe Time Settings

		SCSI Clock Bits (1:0)			
		00	**01**	**11**	**10**
BCLK Freq (1:0)	00	1.0	1.0	2.0	1.5
	01	1.5	1.0	2.0	1.5
	10	1.5	1.0	2.0	1.5
	11	1.5	1.0	2.0	1.5

Where:

1.0 is defined as bcp − k; k is a characterization constant
1.5 is defined as bcp + bc − k; bc is either bcl or bch
2.0 is defined as 2*bcp − k; k is a characterization constant
bcp, bch and bcl refer to the BCLK period, high pulse width, and low pulse width, respectively.

4.4.3. SCSI Operations

Extending the Current SCSI Command

SCSI commands may be programmed to transfer more blocks than originaly programmed. If a new value is written to the Block Count holding register while a SCSI command is executing, the new value will be downloaded from the holding register to the active register when the active Block Count reaches zero. This new Block Count must be written to the Block Count holding register any time before the active Block Count reaches zero. The Block Count register should not be updated again (after the first pipelined value) until the previous pipelined value has been downloaded. The ''Group Complete'' interrupt indicates when the pipelined Block Count is downloaded.

The SCSI Pointer can also be modified when extending the current executing SCSI command—allowing for noncontiguous buffer memory transfers. If the current executing SCSI command is a single phase information transfer command, the SCSI Pointer can be modified after 10 BCLK periods have passed since the SCSI command was issued. If the current executing SCSI command is a reselect combination command, the SCSI Pointer can be modified after detecting the SCSI data transfer phase. The SCSI pointer should not be updated again (after the first pipelined value) until the previous pipelined value has been downloaded. If the SCSI pointer have been modified, the ''Group Complete'' interrupt indicates when the pipelined SCSI pointer is downloaded.

If the SCSI pointer has not been modified when extending the current executing SCSI command, a SCSI pointer download will not occur and contiguous buffer memory transfers will result. Although ''Group Complete'' interrupts will still occur at the completion of each transfer.

The ''Group Complete'' interrupt indicates when the pipelined Block Count and the SCSI Pointer (if modified) are downloaded. After this interrupt is received, the Block Count register and SCSI Pointer registers may be updated again.

A SCSI command may be extended multiple times if desired. Simply wait for the ''Group Complete'' interrupt between each update of the Block Count register.

Table 4.29 summarizes these interrupts.

Pipelining SCSI Commands

There are several combinations of SCSI commands that can be pipelined:

Receive Data followed by Send Disconnect
Receive Data followed by Send End
Send Data followed by Send Disconnect
Send Data followed by Send End
Reselect-Receive Data followed by Send Disconnect
Reselect-Receive Data followed by Send End
Reselect-Send Data followed by Send Disconnect
Reselect-Send Data followed by Send End

Except for the Clear Pending command only these command pairs can be pipelined. The Send End, Send Disconnect and Clear Pending commands are the only commands allowed to be pipelined. No additional commands can be written to the SCSI Command register until the pipelined command has completed its operation.

The Clear Pending Command can be issued to clear the pipelined command if the first command is still executing and the pipelined command has not yet begun.

In addition, changing the SCSI pointer or the Block Count register does not affect the pipelined Send End or Send Disconnect command. If the Block Count register is modified after the Send End or Send Disconnect command has been pipelined the current executing SCSI command may possibly be extended.

A pipelined command should not be issued before 10 BCLK periods have passed since the first command was issued.

It is important to understand how the DP8496/7 determines completion interrupts when pipelining commands. It is possible to completely miss a completion interrupt if errors do not occur for either of the pipelined commands. Refer to Table 4.27 for a description of when completion interrupts are generated and how they are buffered.

TABLE 4.29. SCSI Bufer Management Interrupts

Block Count Reached Zero?	New SCSI Command Been Loaded?	New Block Count Been Loaded	Interrupt Type	Comments
No	X	X	BTC	Single block transfer complete.
Yes	X	Yes	GTC	The last block of the Group has been transferred to/from the SCSI Bus. Command is extended with new Block Count.
Yes	Yes	No	CCI	Previous SCSI command has completed. New pipelined command is starting.
Yes	No	No	CCI	Previous SCSI command has completed. No new command.

Note: BTC = Single Complete Interrupt, GTC = Group Complete Interrupt, CCI = Command Complete Interrupt.

Automatic Mode

The Automatic Mode transfers data between the SCSI bus and external buffer memory using internal DMA. The Automatic Mode is enabled by setting the A/M (Automatic/Manual) bit in the SCSI Operation register.

If the Synchronous Transfer register is set for synchronous transfers, the transfer mode will automatically change between synchronous and asynchronous depending on the information transfer phase, if the DP8496/7 is configured as a Target. If configured as an Initiator, the Synchronous Transfer register must be setup correctly for the phase prior to issuing the Transfer Info command.

Transfers of odd byte blocks are allowed in both Asynchronous and Synchronous modes. However, the SCSI FIFO will be flushed after each block transferred if the block size is odd in word mode. Therefore, the maximum data transfer rate will be achieved only for transfers of even byte counts while in word mode. While in byte mode, odd and even transfers are equally fast.

In both synchronous and asynchronous transfer modes, successive requests or acknowledges cannot occur faster than two bus clock cycles apart. If this condition occurs, a ''SCSI Bus Error'' interrupt will be generated and a ''SCSI Data Overrun'' error code will be set in the SCSI Status register.

Manual Mode Information Transfer

Manual Mode is enabled by clearing the A/M (Automatic/Manual) bit in the SCSI Operation register. This is the default mode after a CRST or a SCSI Reset Command. While in this mode all the commands in the SCSI Command register become unavailable. This mode is intended to accommodate lower level processor controlled transfers which may be unique to an application. In other words, the SCSI protocol becomes completely processor controlled.

A typical sequence may be to issue a (Re)Select command in the Automatic Mode, since this is not easy or fast in the Manual Mode, then switch to Manual Mode for further, perhaps unique, transfers or phase sequences.

Data can then be transferred in two different ways while in Manual Mode. By setting the HE (Handshake Enable) bit in the Synchronous Transfer register to a ''0'', the SCSI data bus appears as a simple transceiver without a latch. This mode is intended for unique applications which require processor control over the Arbitration and Selection phases. Data written to the SCSI Data register will be asserted immediately on the SCSI bus and data read from the SCSI Data register will be an exact and immediate reflection of the SCSI bus.

By setting the HE bit to a ''1'', the SCSI Data register becomes latched and the processor can easily transfer single bytes of data. The DBR (Data Buffer Ready) bit in the SCSI Status register should be polled to pace the data transfer. If sending data out of the DP8496/7, the DBR bit should be polled for a ''0''. This indicates the SCSI Data register is empty and ready for another byte to be sent. If receiving data into the DP8496/7, the DBR bit should be polled for a ''1''. This indicates the SCSI Data register is full and a byte is ready to be read by the processor.

A Few Cautions (while in Manual Mode):

If the DP8496/7 is a target receiving information from the SCSI Bus then the processor should switch the I/O line before reading the last byte. This is to prevent the DP8496/7 from issuing another REQ and causing a DMA overrun condition on the Initiator side.

If in Manual Mode and not active on the SCSI bus, several conditions must be met to prevent the DP8496/7 from driving the SCSI bus or causing unexpected error conditions within itself. The HE (Handshake Enable) bit in the Synchronous Transfer register must be 0, the SCSI Data register must be 00, the SPP (SCSI Parity Polarity) bit in the Setup 2 register must be 1 (even parity). These conditions are all set up upon assertion of the $\overline{\text{CRST}}$ pin or a SCSI Reset command. Be sure that these conditions are set up before switching the A/M bit from Automatic to Manual Mode.

If connected as a Target in Manual Mode, REQ is asserted as soon as the HE (Handshake Enable) bit in the Synchronous Transfer register is switched from ''0'' to ''1''.

Note that no DMA transfers occur between the SCSI bus and buffer memory while in Manual Mode.

While in Manual Mode, there is no automatic protection against generating illegal SCSI operations. Care should be taken to ensure that other devices on the SCSI bus are not adversely affected by custom sequences.

Differential Transceiver Control in Manual Mode

On the DP8497, fourteen pins are dedicated to the task of controlling the direction of off-chip differential transceivers required for Fast SCSI type applications. Control of these

signals in manual mode can be accomplished by writing to the Differential SCSI 1 and 2, SDIF1 and SDIF2 (4Eh and 4Fh), registers.

SCSI Bus Reset

Normally, if the RST signal on the SCSI bus becomes asserted, the DP8496/7 will deassert all the SCSI bus signals. A ''SCSI Bus Reset Received'' interrupt will be generated. If a SCSI command is executing, it will be terminated immediately with a ''SCSI Bus Reset Received'' interrupt. This interrupt may be buffered after other pending interrupts as defined in Table 4.27.

The RST signal on the SCSI bus will be ignored by the DP8496/7 for only one condition. This condition is while in the Manual Mode with the HE (Handshake Enable) bit in the Synchronous Transfer register is set to ''0''. In this situation the DP8496/7 will not be affected by the RST signal. This is a dangerous mode to be in on a normally operating SCSI bus! The intended use of this mode is to allow manipulation of SCSI bus signals while SCSI Bus Reset shuts down all other devices on the SCSI bus.

Target and Initiator Modes

The DP8496/7 has commands in the Automatic Mode for both the Initiator and Target roles. However, since the chip will primarily be used in the Target Mode, there are combination commands which optimize performance for this role. Single- and multi-phase commands are available for the Initiator role, for example: Select, Transfer Info and Transfer Pad. Single- and multi-phase commands are also available for the Target role. But, combination commands are also available in the Target mode which can minimize processor overhead. Manual mode can be used with either Target or Initiator modes.

SCSI Parity

Odd or even parity generation and checking on SCSI transfers is offered. The user enables parity and selects parity polarity by setting the SPE and SPP bits of the Setup 2 register. Refer to Setup 2 register (61h) description in Section 3.2 and Table 4.10 in Section 4.2.9 for more details.

When in Initiator mode, the DP8496/7 will assert ATN whenever it detects a parity error on the received data, if SCSI parity is enabled.

Combination Commands

Combination commands bring together most of the commonly used sequences for Target operation. Their primary purpose is to reduce the number of interrupts and thus the processor overhead required in SCSI transactions.

Most combination commands may be issued before or after a SCSI connection. However, two commands (Send Disconnect and Send End) can only be issued after a SCSI connection has been established.

The other combination commands are of two classes: those which wait to be selected and those which reselect. Both of these classes can be issued before or after a SCSI connection has been established. For example, an Initiator may have selected the Target before any ''Wait for Select'' command was issued by the local processor. This ''Wait for Select'' command can still be issued after the selection is complete and the specified transfer phases will be accomplished. Likewise, the Target may have used the standalone Reselect command to re-establish a connection with some Initiator and now wish an Identify message and data transfer sequence. Simply issue the appropriate Reselect combination command and the Reselection sequence will automatically be skipped. The command will start by sending the Idenfity message.

The ''Wait-for-Select'' type commands are terminated either by completing their execution, being overwritten by another command, or aborted by an unexpected condition.

However, if a ''Wait for Select'' command is overwritten by another command after it has started responding to a selection, but before an interrupt has been issued, a ''Last Command Ignored'' interrupt will be generated through the SCSI Interrupt register.

It is important to consider the SCSI Pointer while overwriting a ''Wait for Select'' with another command such as a Reselect combination command. Typically two different pointer locations will be necessary; one for the Command Descriptor Block of the ''Wait for select'' command and one for the data block(s) of the Reselect command. The software must make sure there is no chance the ''Wait for Select'' command is executing before changing the SCSI Pointer. The following procedure will ensure proper SCSI Pointer modification.

1. Issue a Clear Pending Command to overwrite the ''Wait for Select'' command.
2. Read the SCSI Status register and check for the Selection in Process code.
3. If the SCSI Bus is free (Status Code = 0000 or 0001), then the ''Wait for Select'' command has been successfully aborted. You are free to update the SCSI Pointer.
4. If any other Status Code, wait for an interrupt to indicate the completion of the ''Wait for Select'' command (or perhaps an unexpected reselection). Then, update the SCSI Pointer.

Unexpected (Re)Selection

Even if no command is currently executing, the DP8496/7 will still respond to a (Re)Selection from an Initiator or Target broadcasting the proper ID while in Automatic Mode. This will generate a ''(Re)Selected'' interrupt.

Asserting $\overline{\text{CRST}}$ or issuing a SCSI Reset command will set the Manual Mode to avoid a Selection response before the system is initialized.

A (re)selection is usually asynchronous to normal operations. Therefore, new commands may be written to the SCSI Command register at critical times related to a (re)selection. Here is a summary of responses by the DP8496/7 based on the relationship between writing to the SCSI Command register and a (re)Selection attempt. This assumes that the DP8496/7 is not connected and there is no command currently executing.

1. A new command is written just before (re)selection. Assuming the DP8496/7 losses arbitration and the winning initiator (or Target) selects the DP8496/7, a ''(Re)Selected'' interrupt will be generated and a ''Last Command Ignored'' status will be reported.
2. A new comand is written during a (re)selection. In this case the DP8496/7 will generate a ''(Re)Selected'' interrupt and ''Last Command Ignored'' interrupt.
3. A new command is written after a (re)selection. Since the (re)selection is complete, a ''(Re)Selected'' interrupt will be generated and a ''Last Command Ignored'' status will be reported.

4.0 Functional Description (Continued)

4.5 TIMER

Since there are time-outs of various lengths used throughout SCSI bus transactions and time periods that need to be measured for other external and internal system timing, a general purpose Timer is put on-chip. It can be programmed to generate interrupts at a constant rate. In addition to Selection/Reselection time-outs and monitoring for a ''hung'' bus, this Timer can be reset by the index pulse and used to check disk rotational speed, seek time-outs, head position and many other events. Execution of disk commands can also be delayed until a time-out for positioning of certain operations on disk data.

The Timer interrupt is observable in the Disk Interrupt register and is maskable in the Disk Interrupt Enable register. Two registers control the timer, the Timer Prescale register and the Timer Count register. Timer Prescale not only controls the divisor, but three set/reset functions. Timer Count is simply the number of periods in the interval and it reads back the current count. Timer Count is loaded last and starts the timer.

TABLE 4.30. Timer Periods

BCLK	Min Period	Max Period
10 MHz	400 ns	3.36 sec.
14 MHz	286 ns	2.40 sec.
18 MHz	222 ns	1.86 sec.
22 MHz	182 ns	1.53 sec.

4.5.1 Timer Register Descriptions

Timer Prescale (TPRE) 63h R/W

7	6	5	4	3	2	1	0
LOI	ROZ	DCOZ	x	Prescale Code			

This register is reset by the $\overline{CRST}$ pin and the reset commands.

LOI: Load On Index

1 Restarts counting sequence when leading edge of index pulse is received. This works independently of enabling the Index interrupt. This bit reloads the timer count and leaves the prescale value unchanged. The next interrupt generated will be [1/BCLK * PRESCALE * TIMER COUNT] seconds away. No interrupt will be generated until the Index pulse has been received and the count value has elapsed.

0 The index pulse has no effect on the Timer.

ROZ: Reload On Zero

1 When the Timer Count reaches zero, it is automatically reloaded with the initial value loaded by the processor. An interrupt is generated, if enabled, when the Timer Count register reaches zero. This mode can be used as a ''rate generator'' giving an interrupt at a periodic rate.

0 No reloading of the Timer Count register occurs. An interrupt occurs, if enabled, when the Timer Count register reaches zero. The Timer Interrupt will not occur again until a new value is loaded into Timer Counter.

DCOZ: Drive Command On Zero

1 Drive commands loaded to Disk Command register will not be started until the Timer Count register counts down to zero. The Timer Count register should be loaded after the Disk Command register with a value calculated to position the command at the track position desired. Typically the LOI (Load On Index) bit will be set and command would be a Read or Write Unformatted.

0 Drive commands are executed normally.

Prescale Code

TABLE 4.31. Prescale Code

Prescale Code 3 2 1 0	Prescale Code (Hex)	Prescale Value
0 0 0 0	0	4
0 0 0 1	1	8
0 0 1 0	2	16
0 0 1 1	3	32
0 1 0 0	4	64
0 1 0 1	5	128
0 1 1 0	6	256
0 1 1 1	7	512
1 0 0 0	8	1024
1 0 0 1	9	2048
1 0 1 0	A	4096
1 0 1 1	B	8192
1 1 0 0	C	16k
1 1 0 1	D	32k
1 1 1 0	E	64k
1 1 1 1	F	128k

LOI and ROZ Bits

Load On Index and Restart On Zero bits can be used in any combination. When both the LOI and ROZ bits are set, the LOI function takes priority. If the Index pulse is received, a reload is done regardless of the count. The interesting case is when the Timer Count register reaches zero before the next index pulse. Multiple interrupts can be generated during the disk revolution with each revolution synchronized to the index pulse. A useful setup might be to program the prescale and count to interrupt for each sector on a track.

Timer Count (TCNT) 64h R/W

7	6	5	4	3	2	1	0
Count							

The action of loading the register starts the counting process. Timer Count is loaded by the processor and specifies the number of [1/BCLK * PRESCALE] periods to count. It decrements to zero and creates an interrupt if enabled. The Timer Count is readable at any time and contains the current count as it is decrementing.

5.0 Application Information: Buffer Memory Interfacing in a Fast SCSI Implementation

Today's hard disk systems require high throughput in each stage from the magnetic media to the CPU. Additionally, users wish to have wide access to a great many different drive types and manufacturers. This was the motivation for the SCSI interface. To achieve high throughput, SCSI-2 defines a differential cable transmission scheme and a Fast option that allows transfers at 10 MBytes/sec. At the same time faster serial data rate from the disk drive is required in order to utilize this increased bus bandwidth. Thus disk data rates of 33 Mbits/sec, (or 4.2 MBytes/sec) are becoming common. Into this fray of information exchange is the added requirements of error correction and drive control, hence the need for a processor to also have access to the data.

This example application, shown in *Figure 5.1,* illustrates how the DP8497 can be used to implement a drive design that achieves 10 MBytes/sec SCSI transfer rate and a 33 Mbit/sec disk data rate while using only low-cost 100 ns variety DRAMs. The key feature of the DP8497 that makes this possible is its word-wide buffer memory port. This example application requires the buffer memory port to provide enough bandwidth to support 10 MBytes/sec (SCSI) + 4.2 MBytes/sec (Disk) + Processor Accesses + DRAM Refreshes under the maximum load. The DP8496/7 automatically handles refresh if DRAMs are used. The refresh occurs once every 128 BCLK cycles and takes 5 BCLK cycles to complete. Hence a refresh cycle starts after 123 BCLK cycles. Under the maximum load, the FIFOs will transfer 4 words in 9 BLCK cycles. This gives over 13 transfers per refresh. Using a word length of 2 bytes, this means that 2 × 13 × 4 or 104 bytes will be transferred in 123 BLCK cycles. If BCLK is running at 20 MHz, then there is at least 16.91 MBytes/sec available for disk, SCSI and processor accesses. With the above constraints this allows the processor to have access to a bandwidth of 2.7 MBytes/sec under the worst case.

Standard 100 ns fast page mode DRAMs have a cycle time of 180 ns for a random access. Page mode DRAMs, the most common type, allow a faster access to data that shares the same ''page'' as the previous access. Each of these ''same page'' accesses takes 55 ns. Thus with a burst of 4 words under a high load condition, there is one random access (usually) followed by 3 fast accesses to the same page. The minimum time in which these accesses could be achieved would be 180 ns + 3 × 55 ns or 345 ns. The time for a burst read of 4 words is 9 BCLKs; thus at a BCLK of 20 MHz, the total time taken is 450 ns. This is more than enough time for the DRAM. With this ability it is possible to have a sustained memory bandwidth of 16.9 MBytes/sec with 100 ns Page mode DRAMs. However, the random and page access times, cycle time, and setup and hold times should be checked against the buffer memory timing spec for the particular type of DRAMs used.

Since page mode DRAMs have a limited page size, there is often a need to change pages. This action requires a new row address to be presented to the DRAMs. If a burst will require the address to change its page, then the burst is truncated to fit up to the end of the page, and a new burst transfer occurs at the beginning of the page boundary. The worst case is when there is only one word that can be transferred at the end of the page. Thus when the burst should have transferred 4 words in 9 BCLK cycles that same four word transfer now takes 5 (1 word) + 8 (three words) or 13 BCLK cycles. The system will only cross a page boundary at the most once every page size. In the DP8496/7 this is 256 addresses. If transfers are never aligned to page boundaries, the following sequence of events will occur for a 256 word transfer:

3 blocks of 54 words each, terminated with a refresh.

1 block of 54 words, terminated with a refresh and a page boundary.

1 block of 40 words, to complete the 256 word transfer.

Total of 256 words transferred.

This requires the following number of corresponding cycles:

3 blocks of 128 BCLK cycles.

1 block of 132 cycles, because of the page boundary.

1 block of 90 cycles for the remaining 40 words.

Total of 606 BCLK cycles taken.

Therefore the corresponding transfer rate becomes 16.898 MBytes/sec. Note that the reduction in bandwidth due to page effects is only 0.7%. *Figure 5.3* illustrates the net, attainable bandwidth as a function of BCLK frequency.

The 512 KByte buffer memory shown in the schematic makes possible large cache buffers for the purpose of reducing effective access times. The 256k × 18 DRAM chip was chosen to minimize the number of chips, while still maintaining the word-wide bandwidth. In this way, only one DRAM chip is used. However, cheaper x4 DRAM chips may be used at the cost of increased board space. The DP8496/7s support other widths of DRAMs such as 256k x 4s and 1 Meg x 1s. The SDDC can support the load of up to 12 DRAMs. The example also uses parity to help ensure that the data from the host computer is not errantly changed on its way to the magnetic media, or visa versa.

The Rest of the Circuit:

To start, the connection between the SCSI cable and the differential transceivers requires termination. The termination resistors are not shown in the schematic; however, they should be the same as defined in the SCSI standard. This is shown in *Figure 5.2*.

The DB8497 has the differential transceiver enable controls on chip. These connect directly to the drive enable pins on the DS36954As. In this way, complete differential SCSI can be achieved by only adding the transceiver chips and termination resistors.

The μController used is the HPC46003. This 16-bit processor is capable of achieving high instruction rates and uses a very compact coding scheme. Other versions of the μController have internal ROM that may be mask programmed. All versions have at least 256 bytes of internal RAM.

This circuit has an external 32 KBytes of ROM for prototyping. An additional 32 KBytes of RAM is provided to allow the HPC to download programs through its UART, again for prototyping, and also for additional storage. Since the μController has been forced into 8-bit access, only a single latch is required to hold the address for the RAM and ROM.

5.0 Application Information: Buffer Memory Interfacing in a Fast SCSI Implementation (Continued)

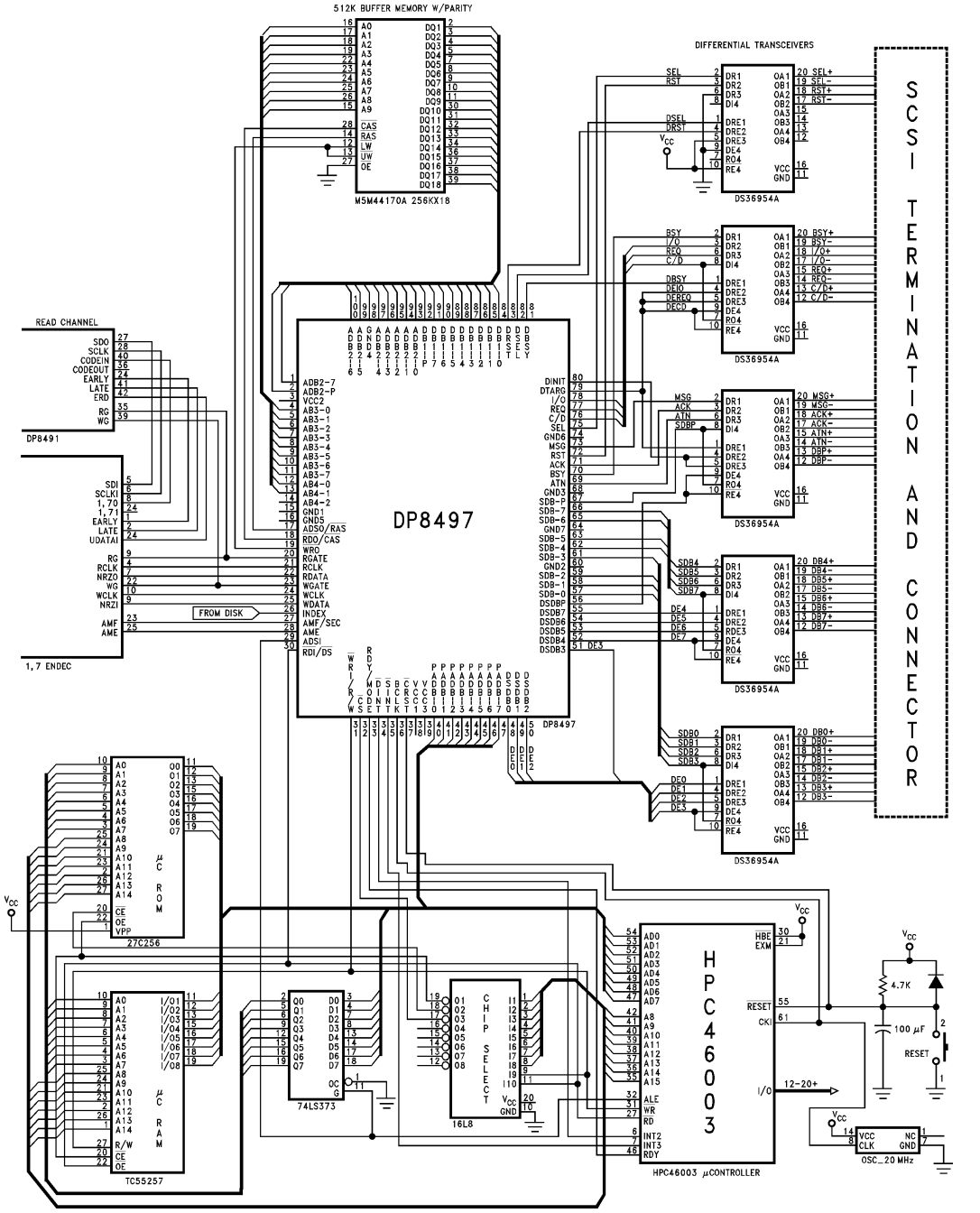


FIGURE 5.1. Typical Application Diagram

5.0 Application Information: Buffer Memory Interfacing in a Fast SCSI Implementation (Continued)

There is also a PAL programmed to provide the chip select signals. Additionally, the HPC46003 has many input and output pins available for drive control.

The disk connections are straight forward. The disk controller accepts the unencoded NRZ from any encoder/decoder, here the 1,7 ENDEC is shown. The Read channel is the last block before the head's preamp. This schematic shows only those read-channel connections that apply to the disk controller portion of the disk.

The '96B/97 have an advanced SCSI controller. This means the on-board μController does not need extremely sophisticated firmware for controlling the SCSI interface. They also have high level disk control commands. The μController can just issue simple commands with no further need for intervention.

To set up the DP8496/DP8497 a few registers need to be initialized. The first setup register is SUP1 at 60h. The bits in this application are defined as:

Bit #	Name	Description
7, 6	CLK (1:0)	These two bits are set to 01 for a 20 MHz bus clock.
5	SWS	This bit is for SRAM wait states, it does not apply to systems using DRAMs for their buffer memory. Here it is set to 0 as a default.
4	DFP	This bit when set disables the use of fast page mode DRAMs. Here it is set to 0 because fast page mode is necessary to acheive the high bandwidth.
3, 2	DD (1:0)	These bits select the DRAM size. Since 256k x 16's are used, the values for these bits should be 01.
1	RSEL	This is the RAM select bit. Here it is 1 to select DRAMs.
0	RWID	RAM Data Path Width. Here it is 0 for word mode.

Therefore the value to load at 60h in the DP8496/DP8497 is 01000110 or 46h.

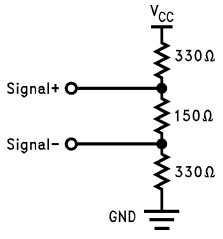
The setup register SUP2 is concerned with how the processor accesses are performed and the enabling and type of parity for the buffer and SCSI interfaces. Here it is loaded as follows:

Bit #	Name	Description
7	AI	Auto Increment. Depends on how the processor wants to access the buffer memory. Since it depends on the software, it is set to 0 here for example.
6	BPP	Buffer Parity Polarity. Even parity is chosen arbitrarily, so this bit is set to 1.
5	BPE	Buffer Parity Enable. This is a 1 to enable the buffer memory parity checks.
4	SPP	SCSI Parity Polarity. Here Odd parity is chosen, so the bit is set to 0.
3	SPE	SCSI Parity Enable. This is set to 1 to check the parity on the SCSI bus when the conditions are met.
2–0	SID	SCSI ID. This is set in this example to be 110 (6) to make it the next highest priority after the initiator. This is a good priority for a boot drive. But typically, the μC's firmware will set this value based on switches or jumpers set by the user.

Thus the value to be written to the SUP2 register at 61h is 01101110 or 7Eh.

The third setup register SUP3 has only one bit to set for this configuration. The bit is the CI bit, the Combine Interrupts bit. It is 0 to make the disk related interrupts appear on the DINT pin and the SCSI related interrupts to appear on the SINT pin. Therefore a value of 00 should be written to the SUP3 register at 62h.

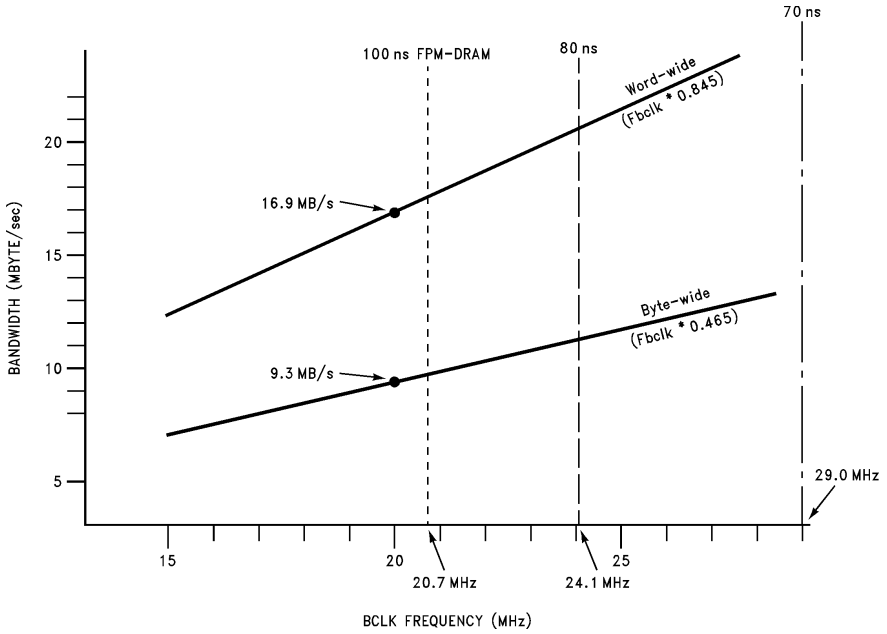
The DP8496/DP8497 devices are extremely efficient in providing high speed disk control and a SCSI interface. The disk control and interface functions in this application of a 512 KByte buffer, Fast 10 MBytes/sec differential SCSI, and a high performance μController can be made using only 12 chips.



TL/F/11212–8

FIGURE 5.2

5.0 Application Information: Buffer Memory Interfacing in a Fast SCSI Implementation (Continued)



TL/F/11212–73

FIGURE 5.3. Net, Attainable Bandwidth of DP8496/DP8497

6.0 D.C. Specifications

Absolute Maximum Ratings (Notes 1 and 2)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Supply Voltage (V_{CC})	−0.5V to +7.0V
DC Input Voltage (V_{IN})	−0.5V to V_{CC} + 0.5V
DC Output Voltage (V_{OUT})	−0.5V to V_{CC} + 0.5V
Storage Temperature (T_{STG}) Range	−65°C to +165°C
Package Power Dissipation (P_D)	1.5W
Lead Temperature (T_L) (Soldering, 10 Seconds)	260°C

Operating Conditions

		Min	Max	Units
Supply Voltage (V_{CC})		4.5	5.5	V
Operating Temperature (T_A)		0	+70	°C
ESD Tolerance:	C_{ZAP} = 100 pF, R_{ZAP} = 1.5 kΩ (Note 4)	2000		V

DC Electrical Characteristics V_{CC} = 5V ±10%, T_A = 0°C to 70°C unless otherwise specified. (Note 3)

Symbol	Parameter	Conditions	Min	Max	Units
V_{IH}	High Level Input Voltage		2.0		V
V_{IL}	Low Level Input Voltage			0.8	V
I_{IH}	Input Current	V_{IN} = V_{CC} or GND		±20	μA
I_{OZ}	Output TRI-STATE Leakage Current	V_{OUT} = V_{CC} or GND		±20	μA
I_{CC}	Average V_{CC} Supply Curent	V_{IN} = V_{IH} or V_{IL}, I_{OUT} = 0 μA, BCLK = 20 MHz, RCLK = 33 MHz		50	mA
BUFFER MEMORY PINS (DB1, ADB2, AB3, AB4, ADSo, RDo, WRo)					
V_{OH}	High Level Output Voltage	V_{IN} = V_{IH} or V_{IL}, $\|I_{OUT}\|$ = 20 μA	V_{CC} − 0.1		V
		V_{IN} = V_{IH} or V_{IL}, $\|I_{OUT}\|$ = 8.0 mA	3.5		V
V_{OL}	Low Level Output Voltage (SCSI)	V_{IN} = V_{IH} or V_{IL}, $\|I_{OUT}\|$ = 20 μA		0.1	V
		V_{IN} = V_{IH} or V_{IL}, $\|I_{OUT}\|$ = 8.0 mA		0.4	V
SCSI BUS PINS (SCSI Data and SCSI Control Pins on DP8496 Only)					
I_{LKG}	Output High Leakage Current	V_{OUT} = V_{CC} or GND		±20	μA
V_{OL}	Low Level Output Voltage (SCSI)	V_{IN} = V_{IH} or V_{IL}, $\|I_{OUT}\|$ = 20 μA		0.1	V
		V_{IN} = V_{IH} or V_{IL}, $\|I_{OUT}\|$ = 48.0 mA		0.5	V
RDY/MODE PIN IN MODE CONFIGURATION					
I_{IL}	Input Low Current	V_{IN} = GND		−500	μA
ALL OTHER PINS (Including SCSI Data, Control, and Transceiver Direction Control Pins on DP8497)					
V_{OH}	High Level Output Voltage	V_{IN} = V_{IH} or V_{IL}, $\|I_{OUT}\|$ = 20 μA	V_{CC} − 0.1		V
		V_{IN} = V_{IH} or V_{IL}, $\|I_{OUT}\|$ = 2.0 mA	3.5		V
V_{OL}	Low Level Output Voltage (SCSI)	V_{IN} = V_{IH} or V_{IL}, $\|I_{OUT}\|$ = 20 μA		0.1	V
		V_{IN} = V_{IH} or V_{IL}, $\|I_{OUT}\|$ = 2.0 mA		0.4	V

SCSI INPUT PINS CHARACTERISTICS (SCSI Data and SCSI Control Pins)

Symbol	Parameter	Conditions	Typ	Units
V_{HYS}	Input Hysteresis		0.2	V
V_{TH}	Nominal Switching Threshold		1.4	V

Note 1: Absolute Maximum Ratings are those values beyond which damage to the device may occur.

Note 2: Unless otherwise specified, all voltages are referenced to ground.

Note 3: These DC Electrical Characteristics are measured staticly, and not under dynamic conditions.

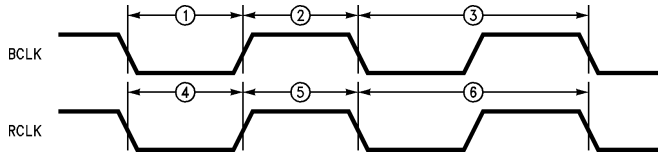
Note 4: Value based on test complying with NSC SOP-5-028 human body model ESD testing using the ETS-910 tester.

7.0 A.C. Specifications

Refer to Section 8.0 for AC Timing Test Conditions

The timing values and formulae specified in this preliminary document are based on the chip's design values; and therefore are intended to assist the user in making initial design choices. Final AC Specifications will be determined after a thorough characterization of the product has been performend.

7.1 CLOCK TIMING



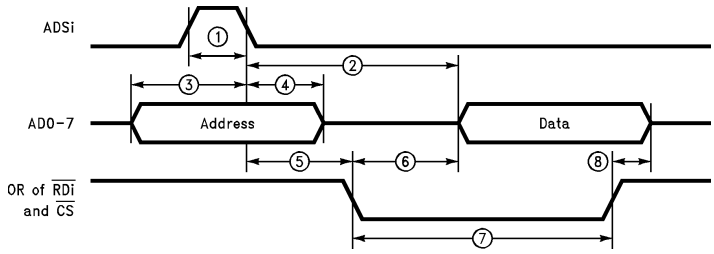
TL/F/11212–35

ID #	Symbol	Parameter	DP8496/7-33	DP8496/7-50	Max	Units
			Min	Min		
1.1	bcl	BCLK Low (Note 1)	18	18	60	ns
1.2	bch	BCLK High (Note 1)	18	18	60	ns
1.3	bcp	BCLK Period (Note 1)	40	40	100	ns
1.4	rcl	RCLK Low	13	9	600	ns
1.5	rch	RCLK High	13	9	600	ns
1.6	rcp	RCLK Period	30	20	1000	ns

Note 1: BCLK frequency should not be less than 10 MHz to ensure proper DRAM refresh and to guarantee proper SCSI bus timing. Please refer to Table 4.21A for Synchronous Information Transfers.

Note 2: BCLK can be as low as 2 MHz to save power while the chip is idle, however, the DRAM contents will be invalid.

7.2 REGISTER READ (NSC/Intel Mode)

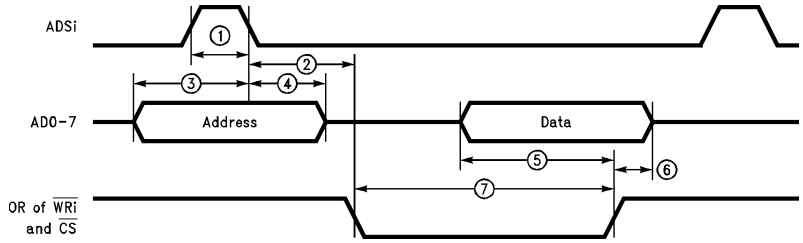


TL/F/11212–36

ID #	Symbol	Parameter	Min	Max	Units
2.1	aswi	Address Strobe Width	20		ns
2.2	asdv	Address Strobe to Data Valid		120	ns
2.3	asas	Address Setup to Address Strobe	9		ns
2.4	ahas	Address Hold from Address Strobe	10		ns
2.5	asrd	Address Strobe to RD Strobe	20		ns
2.6	dvcs	Data Valid from RD		50	ns
2.7	rdwi	RD Strobe Width	50		ns
2.8	dhcs	Data Hold from RD	20	30	ns

7.0 A.C. Specifications (Continued)

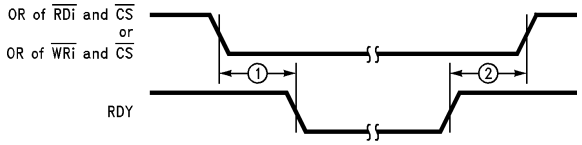
7.3 REGISTER WRITE (NSC/Intel Mode)



TL/F/11212–37

ID #	Symbol	Parameter	Min	Max	Units
3.1	aswi	Address Strobe Width	20		ns
3.2	aswr	Address Strobe to Write Strobe	20		ns
3.3	asas	Address Setup to Address Strobe	9		ns
3.4	ahas	Address Hold from Address Strobe	10		ns
3.5	dvwr	Data Valid to WR Strobe	100		ns
3.6	dhwr	Data Hold from WR Strobe	10		ns
3.7	wrwi	WR Strobe Width	100		ns

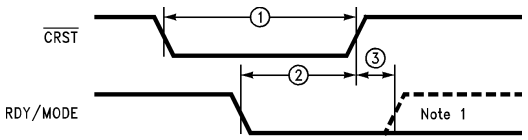
7.4 READY PIN



TL/F/11212–38

ID #	Symbol	Parameter	Min	Max	Units
4.1	srdyt	Strobe to Assertion of RDY		55	ns
4.2	rdys	RDY Released to End of Strobe	55		ns

7.5 MODE PIN



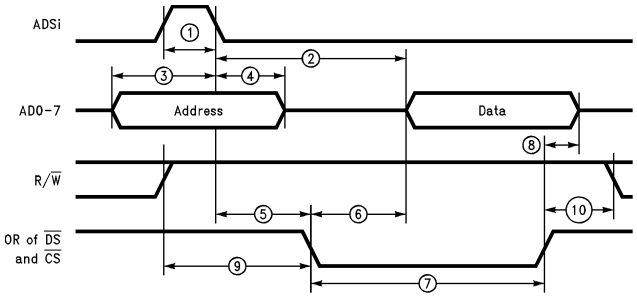
TL/F/11212–39

ID #	Symbol	Parameter	Min	Max	Units
5.1	crstwi	Chip Reset Strobe Width	5 bcp		ns
5.1	mds	Mode Setup to CRST	100		
5.1	mdh	Mode Hold from CRST	50		ns

Note 1: The RDY/MODE pin will be driven low internally. It no longer functions as an output after reset goes inactive.

7.0 A.C. Specifications (Continued)

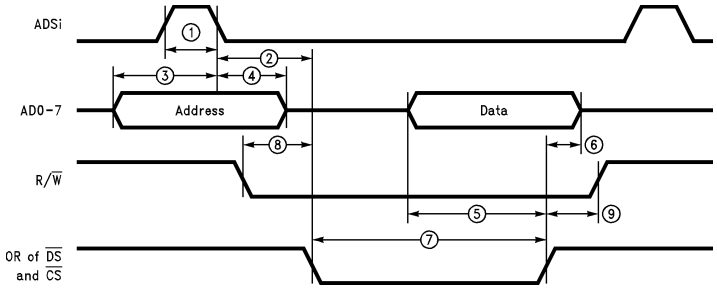
7.6 REGISTER READ (Zilog/Motorola Mode)



TL/F/11212–40

ID #	Symbol	Parameter	Min	Max	Units
6.1	aswi	Address Strobe Width	20		ns
6.2	asdv	Address Strobe to Data Valid		120	ns
6.3	asas	Address Setup to Address Strobe	9		ns
6.4	ahas	Address Hold from Address Strobe	10		ns
6.5	asds	Address Strobe to Data Strobe	20		ns
6.6	dvds	Data Valid from Data Strobe		50	ns
6.7	dswi	Data Strobe Width	50		ns
6.8	dhds	Data Hold from Data Strobe	20	30	ns
6.9	rwas	R/$\overline{W}$ Strobe to Address Strobe	30		ns
6.10	dsrw	Data Strobe to R/$\overline{W}$	30		ns

7.7 REGISTER WRITE (Zilog/Motorola Mode)

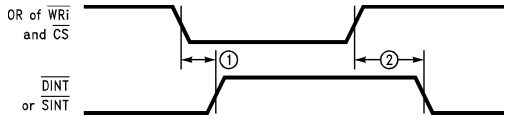


TL/F/11212–41

ID #	Symbol	Parameter	Min	Max	Units
7.1	aswi	Address Strobe Width	20		ns
7.2	asds	Address Strobe to Write Strobe	20		ns
7.3	asas	Address Setup to Address Strobe	9		ns
7.4	ahas	Address Hold from Address Strobe	10		ns
7.5	dvds	Data Valid to WR Strobe	100		ns
7.6	dhds	Data Hold from WR Strobe	10		ns
7.7	dswi	WR Strobe Width	100		ns
7.8	rwds	R/$\overline{W}$ Strobe to Data Strobe	10		ns
7.9	dsrw	Data Strobe to R/$\overline{W}$ Strobe	10		ns

7.0 A.C. Specifications (Continued)

7.8 INTERRUPTS (Note 1)

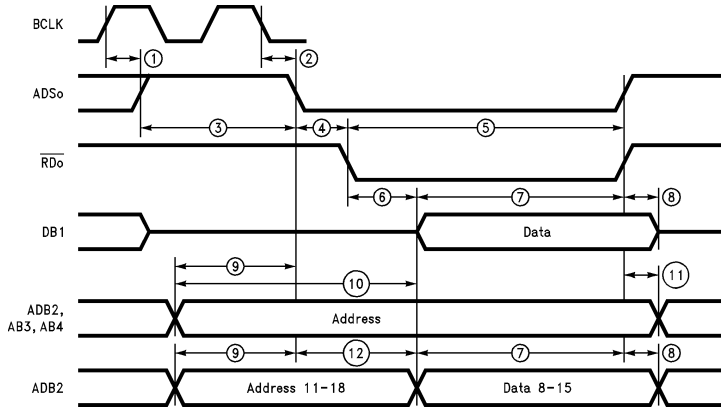


TL/F/11212–42

ID#	Symbol	Parameter	Formula	BCLK = 20 MHz		Units
				Min	Max	
8.1	wraint	WR Strobe Assertion to INT Deassertion			30	ns
8.1	wrdint	WR Strobe Deassertion to INT Assertion	Min = 2bcp, Max = 3bcp + 30	100	180	ns

Note 1: This waveform represents when the Disk Interrupt or SCSI Interrupt register is written to, but interrupts are still pending.

7.9 BUFFER MEMORY STATIC RAM READ



TL/F/11212–43

ID#	Symbol	Parameter	Formula	BCLK = 20 MHz		Units
				Min	Max	
9.1	cash	BCLK to Address Strobe High			30	ns
9.2	casl	BCLK to Address Strobe Low			30	ns
9.3	asw	Address Strobe Width	bcp − 5	45		ns
9.4	asrd	Address Strobe to RD Strobe	bch − 5	18		ns
9.5	rdw	RD Strobe Width	W + bcp + bcl − 5	68		ns
9.6	rdd	RD Strobe to Data	W + bcp + bcl − 13		60	ns
9.7	ds	Data Setup to RD Strobe		13		ns
9.8	dh	Data Hold from RD Strobe		10		ns
9.9	adsas	Address Setup to Address Strobe	bcl − 5	18		ns
9.10	adsd	Address Setup to Data	W + 2bcp + bcl − 13	110		ns
9.11	adhrd	Address Hold from RD Strobe	bch − 15	8		ns
9.12	adhas	Address Hold from Address Strobe (16-Bit Mode)	bch − 15	8		ns

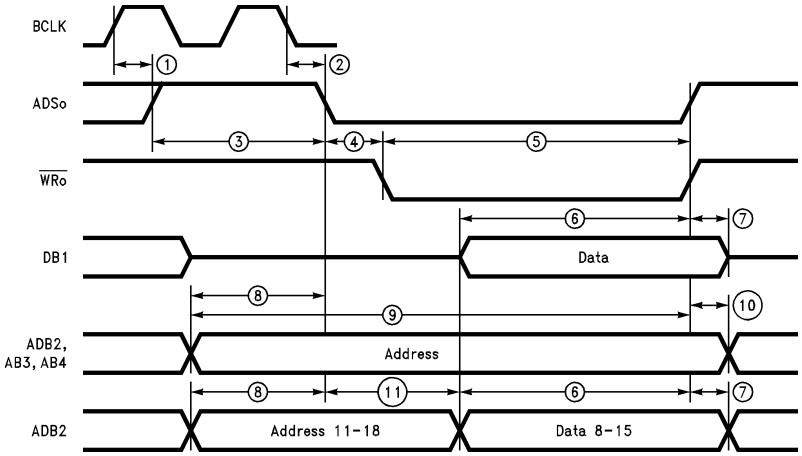
Note 1: While in the 8-bit mode, the last waveform is not used. While in the 16-bit mode, the last waveform is used and the previous waveform represents AB3 and AB4 only.

Note 2: bcp, bcl & bch refer to the BCLK frequency in use.

Note 3: If the Wait States field in the Setup 1 register is ''00'', then W = 0. Otherwise, W = bcp.

7.0 A.C. Specifications (Continued)

7.10 BUFFER MEMORY STATIC RAM WRITE



TL/F/11212–44

ID#	Symbol	Parameter	Formula	BCLK = 20 MHz		Units
				Min	Max	
10.1	cash	BCLK to Address Strobe High			30	ns
10.2	casl	BCLK to Address Strobe Low			30	ns
10.3	asw	Address Strobe Width	bcp − 5	45		ns
10.4	aswr	Address Strobe to WR Strobe	bch − 5	18		ns
10.5	wrw	WR Strobe Width	W + bcp + bcl − 5	68		ns
10.6	dswr	Data Setup to WR Strobe	W + bcp + bcl − 15	58		ns
10.7	dhwr	Data Hold from WR Strobe	bch − 15	8		ns
10.8	adsas	Address Setup to Address Strobe	bcl − 5	18		ns
10.9	adswr	Address Setup to WR Strobe	W + 2bcp + bcl − 5	118		ns
10.10	adhwr	Address Hold from WR Strobe	bcl − 15	8		ns
10.11	adhas	Address Hold from Address Strobe (16-Bit Mode)	bch − 15	8		ns

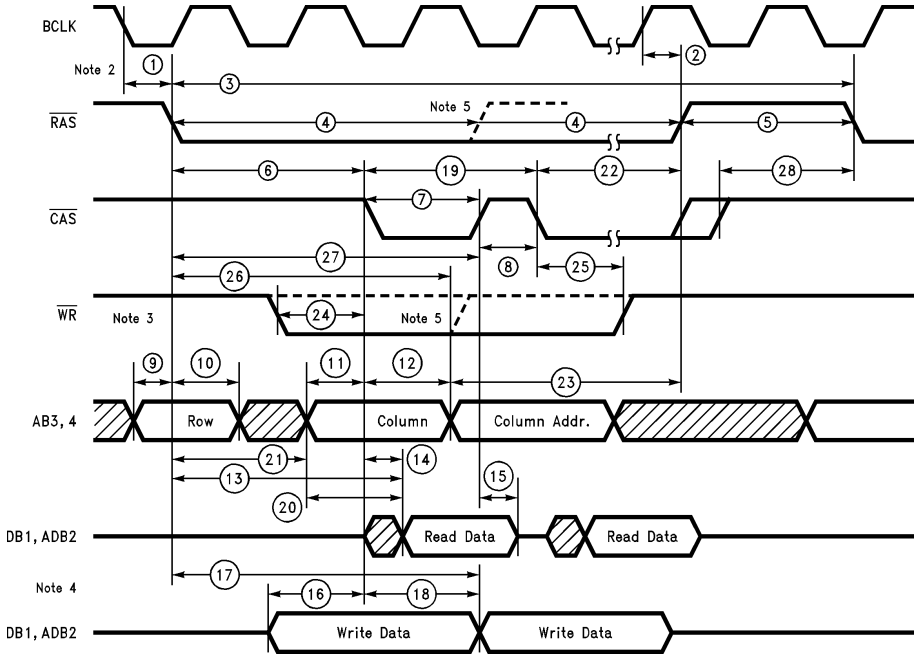
Note 1: While in the 8-bit mode, the last waveform is not used. While in 16-bit mode, the last waveform is used and the previous waveform represents AB3 and AB4 only.

Note 2: bcp, bcl & bch refer to he BCLK frequency in use.

Note 3: If the Wait States field in the Setup 1 register is ''00'', then W = 0. Otherwise, W = bcp.

7.0 A.C. Specifications (Continued)

7.11 BUFFER MEMORY DYNAMIC RAM READ/WRITE



TL/F/11212–45

Timing parameter descriptions and specifications are shown on next page.

Note 1: This diagram shows a fast page mode type operation. Normal one transfer operation would not have multiple CAS cycles and associated column address and data changes as shown here.

Note 2: $\overline{RAS}$ may be asserted on either edge of BCLK.

Note 3: $\overline{WR}$ remains deasserted during a read operation.

Note 4: ADB2 is used for 16-bit mode only.

Note 5: Transition edges shown in dashed line would occur in the normal mode (non fast page mode) transfers.

Note 6: For a Read operation, the last waveform is not used. For a Write operation the second from last waveform is not used.

7.0 A.C. Specifications (Continued)

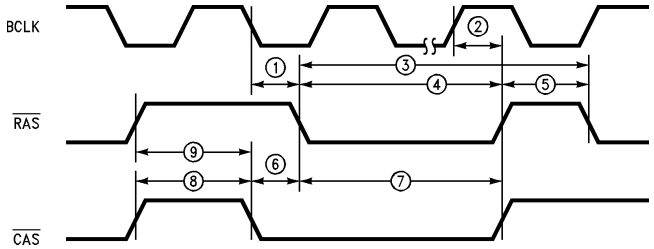
ID #	Symbol	Parameter	Formula		BCLK = 20 MHz		Units
			Min	Max	Min	Max	
11.1	crasa	BCLK to RAS Asserted				30	ns
11.2	crasd	BCLK to RAS and CAS Deasserted				30	ns
11.3	rc	Read/Write Cycle Time Single xfr 4x Burst 6x Burst	5*bcp 9*bcp 12*bcp		250 450 600		ns
11.4	ras	$\overline{RAS}/\overline{WR}$ Pulse Width Single xfr 4x Burst 6x Burst	2*bcp + bcl − 10 7*bcp − 5 10*bcp − 10	3*bcp − 10 7*bcp + 5 10*bcp + 10	112 345 490	140 355 510	ns
11.5	rp	$\overline{RAS}$ Precharge Time	2*bcp − 5		95		ns
11.6	rcd	$\overline{RAS}$ to $\overline{CAS}$ Delay Time	bcp + bcl − 10	bcp + bcl − 10	62	68	ns
11.7	cas	$\overline{CAS}$ Pulse Width	bcp − 5	bcp + 5	45	55	ns
11.8	cp/cpn	$\overline{CAS}$ Precharge Time	bcl − 5		17		ns
11.9	asr	Row Address Setup Time	bch − 10		12		ns
11.10	rah	Row Address Hold Time	bcp − 10		40		ns
11.11	asc	Column Address Setup Time	bch − 10		12		ns
11.12	cah	Column Address Hold Time	bcp − 10		40		ns
11.13	rac	Access Time from $\overline{RAS}$ (Read Only)		2*bcp + bcl − 10		112	ns
11.14	cac	Access Time from $\overline{CAS}$ (Read Only)		bcp − 10		40	ns
11.15	off	Data Hold from $\overline{CAS}$ High (Read)	0	bcp + bcl	0	78	ns
11.16	ds	Data Setup (Write Only)	bc_{min} − 10		32		ns
11.17	dhr	Data Hold from $\overline{RAS}$ (Write Only)	2*bcp + bcl − 10		112		ns
11.18	dh	Data Hold from $\overline{CAS}$ (Write Only)	bcp − 10		40		ns
11.19	pc	Fast Page Mode Cycle Time	bcp + bcl − 10		62		ns
11.20	aa	Access Time from Column Address		bcp + bc_{max} − 10		68	ns
11.21	rad	$\overline{RAS}$ to Column Addr. Delay Time		bcp − 10		40	ns
11.22	rsh	$\overline{RAS}$ Hold Time	bcp − 10		40		ns
11.23	ral	Column Addr. to $\overline{RAS}$ Lead Time	bcp + bch − 10		68		ns
11.24	wcs	Write Command Setup Time	bcp + bcl − 10		68		ns
11.25	wch	Write Command Hold Time	bcp − 10		40		ns
11.26	ar	Column Add. Hold Time from $\overline{RAS}$	2*bcp + bcl −10		112		ns
11.27	csh	$\overline{CAS}$ Hold Time	2*bcp + bcl − 10		112		ns
11.28	crp	$\overline{CAS}$ to $\overline{RAS}$ Precharge Time	2*bcp − 10		90		ns

Note: bcp means Bus Clock period being used. bch and bcl refer to the positive and negative pulse widths of Bus Clock, respectively.

bc_{max} means the greater of the bch or bcl, depending on the Bus Clock's duty cycle; bc_{min} means the lesser of the bch or bcl, depending on the Bus Clock's duty cycle.

7.0 A.C. Specifications (Continued)

7.12 BUFFER MEMORY DYNAMIC RAM REFRESH



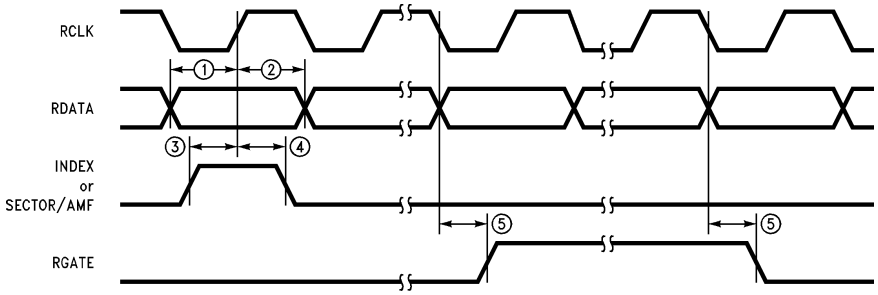
TL/F/11212–46

ID #	Symbol	Parameter	Formula		BCLK = 20 MHz		Units
			Min	Max	Min	Max	
12.1	crasa	BCLK to $\overline{RAS}$ Asserted (See 11.1)				30	ns
12.2	crasd	BCLK to $\overline{RAS}$ and $\overline{CAS}$ Deasserted (11.2)				30	ns
12.3	rcr	Read/Write Cycle Time for Refresh	bcp + bcl		72		ns
12.4	rasr	$\overline{RAS}$ Pulse Width for Refresh	2*bcp + bch − 5		117		ns
12.5	rpr	$\overline{RAS}$ Precharge Time for Refresh	2*bcp + bcl − 5		117		ns
12.6	csr	$\overline{CAS}$ to $\overline{RAS}$ Setup Time	bcp − 5	bcp + 5	45	55	ns
12.7	chr	$\overline{CAS}$ Hold Time	2*bcp + bch − 5		117		ns
12.8	cpn	$\overline{CAS}$ Precharge Time	bcp + bcl − 5		67		ns
12.9	rpc	$\overline{RAS}$ to $\overline{CAS}$ Precharge Time	bcp + bcl − 5		67		ns

Note 1: bcp means Bus Clock Period being used. bch and bcl refer to the positive and negative pulse widths of Bus Clock in use.

Note 2: $\overline{RAS}$ may be asserted on either edge of BCLK.

7.13 DISK READ DATA AND READ GATE TIMING

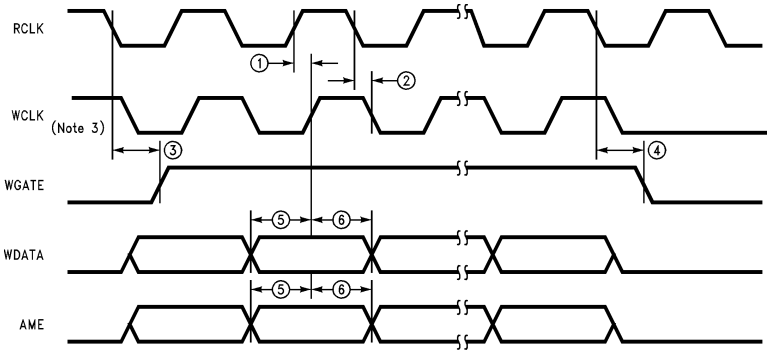


TL/F/11212–47

ID #	Symbol	Parameter	DP8496/7-33	DP8496/7-50	Max	Units
			Min	Min		
13.1	rds	Read Data Setup to RCLK	7	5		ns
13.2	rdh	Read Data Hold from RCLK	10	9		ns
13.3	is	Index Setup to RCLK	10	10		ns
13.4	ih	Index Hold from RCLK	10	10		ns
13.5	crg	RCLK to Read Gate			15	ns

7.0 A.C. Specifications (Continued)

7.14 DISK WRITE DATA AND WRITE GATE TIMING



TL/F/11212–48

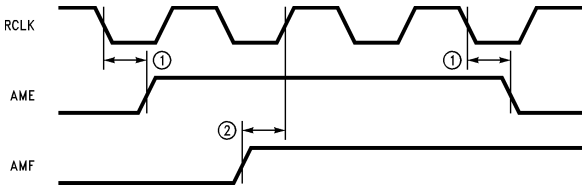
ID #	Symbol	Parameter	Min	Max	Units
14.1	rcwch	RCLK to WCLK High (Note 1)		15	ns
14.2	rcwcl	RCLK to WCLK Low (Note 1)		15	ns
14.3	rcwga	RCLK to Write Gate Asserted	15		ns
14.4	rcwgd	RCLK to Write Gate Deasserted	18		ns
14.5	wds	Write Data Setup to WCLK (Note 2)	rcl − 5		ns
14.6	wdh	Write Data Hold from WCLK (Note 2)	rch − 5		ns

Note 1: The absolute value of rcwch–rcwcl is 5 ns (max).

Note 2: rcl refers to the RCLK low time in use. rch refers to the RCLK high time in use. This formula is tested ony at the frequencies listed in Table 8.1.

Note 3: The WCLK transitions start before the WGATE is asserted and three cycles occur before WGATE assertion.

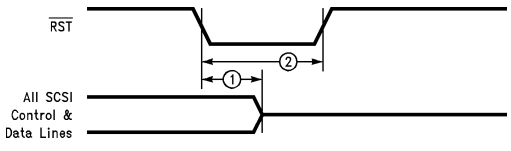
7.15 DISK ADDRESS MARK (PSEUDO-HARD SECTORED FORMAT)



TL/F/11212–49

ID #	Symbol	Parameter	Min	Max	Units
15.1	came	RCLK to Address Mark Enable		15	ns
15.2	amfs	Address Mark Found Setup to RCLK	10		ns

7.16 SCSI RESET



TL/F/11212–50

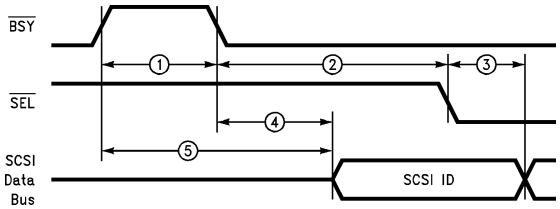
ID #	Symbol	Parameter	Min	Max	Units
16.1	srst	SCSI $\overline{\text{RST}}$ to All SCSI Lines Undriven		600	ns
16.2	rstw	SCSI $\overline{\text{RST}}$ Width (Note 1)	2*bcp		ns

Note 1: bcp refers to the BCLK period in use. This formula is tested only at the frequencies listed in Table 8.1.

All SCSI signals are active-high on the DP8497.

7.0 A.C. Specifications (Continued)

7.17 SCSI ARBITRATION



TL/F/11212–51

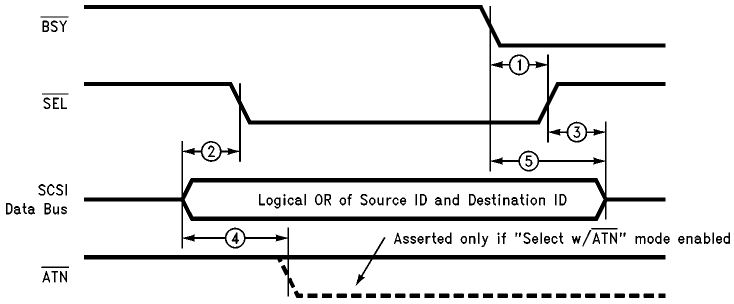
ID #	Symbol	Parameter	Optimal BCLK		Worst Case		Units
			Min	Max	Min	Max	
17.1	bfbl	Bus Free to $\overline{\text{BSY}}$ Out Asserted	1.2	1.5	1.2	2.2	μs
17.2	blsl	$\overline{\text{BSY}}$ Out Asserted to $\overline{\text{SEL}}$ Out Asserted	2.4	4.0	2.4	5.0	μs
17.3	sea	$\overline{\text{SEL}}$ Out Asserted to SCSI Data Change	1.2	1.4	1.2	2.1	μs
17.4	blidv	$\overline{\text{BSY}}$ Out Asserted to Valid SCSI ID	−30	30	−30	30	ns
17.5	bfidi	Bus Free to Valid SCSI ID	1.2	1.5	1.2	2.2	μs

Note: BCLK is defined in the Setup 1 register. Optimal BCLK is defined as follows:

Bclk Freq Code		
7	6	Optimal Frequency
0	0	15 MHz
1	0	17.5 MHz
0	1	20 MHz
1	1	25 MHz

ALL SCSI Signals are active-high on the DP8497.

7.18 SCSI SELECTION AS INITIATOR (W/O Arbitration)



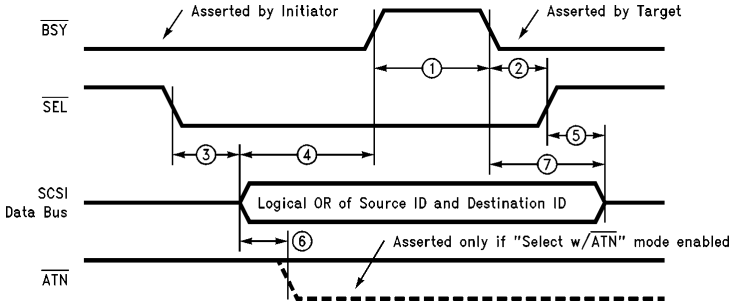
TL/F/11212–52

ID #	Symbol	Parameter	Optimal BCLK		Worst Case		Units
			Min	Max	Min	Max	
18.1	bsel	$\overline{\text{BSY}}$ In Asserted to $\overline{\text{SEL}}$ Out Deasserted	90		90		ns
18.2	idsel	Valid SCSI ID Out to $\overline{\text{SEL}}$ Out Asserted	90	230	90	350	ns
18.3	selid	$\overline{\text{SEL}}$ Out Deasserted to SCSI ID Out Invalid	−30	30	−30	30	ns
18.4	idatn	Valid SCSI ID Out to $\overline{\text{ATN}}$ Out Asserted (if Enabled)	−30	30	−30	30	ns
18.5	bidi	$\overline{\text{BSY}}$ In Asserted to SCSI ID Out Invalid	90		90		ns

Note: ALL SCSI Signals are active-high on the DP8497.

7.0 A.C. Specifications (Continued)

7.19 SCSI SELECTION AS INITIATOR (With Arbitration)



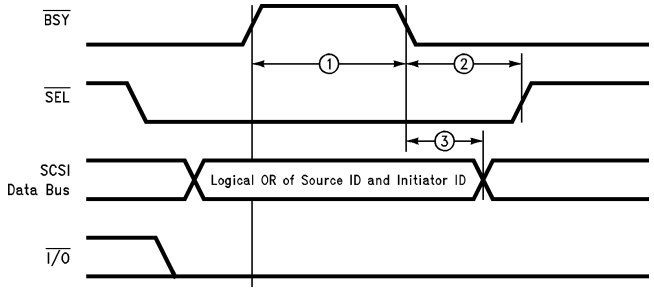
TL/F/11212–53

ID #	Symbol	Parameter	Optimal BCLK		Worst Case		Units
			Min	Max	Min	Max	
19.1	bobi	$\overline{BSY}$ Out Released to $\overline{BSY}$ In Asserted (Note 1)	400		400		ns
19.2	bisel	$\overline{BSY}$ In Asserted to $\overline{SEL}$ Out Deasserted	90		90		ns
19.3	selid	$\overline{SEL}$ Out Asserted to Valid SCSI ID Out	1.2	1.4	1.2	2.1	μs
19.4	selbo	Valid SCSI ID Out to $\overline{BSY}$ Out Deasserted	90	230	90	350	ns
19.5	seld	$\overline{SEL}$ Out Deasserted to SCSI ID Out Invalid	−30	30	−30	30	ns
19.6	datn	Valid SCSI ID Out to $\overline{ATN}$ Out Asserted (If Enabled)	−30	30	−30	30	ns
19.7	bIIDI	$\overline{BSY}$ In Asserted to SCSI ID Out Invalid	90		90		ns

Note 1: BSY may be asserted sooner, but it will not be detected until (bobi) time has passed.

All SCSI signals are active-high on the DP8497.

6.20 SCSI RESELECTION AS INITIATOR



TL/F/11212–54

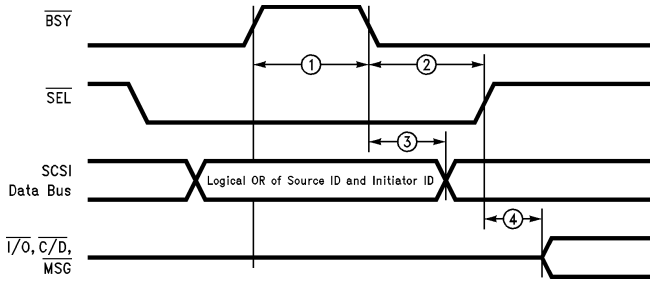
ID #	Symbol	Parameter	Optimal BCLK		Worst Case		Units
			Min	Max	Min	Max	
20.1	bihbol	$\overline{BSY}$ In Deasserted to $\overline{BSY}$ Out Asserted (Note 1)	400	800	400	1300	ns
20.2	blsh	$\overline{BSY}$ Out Asserted to $\overline{SEL}$ In Hold	0		0		ns
20.3	blih	$\overline{BSY}$ In Asserted to SCSI ID Hold	0		0		ns

Note 1: This time starts when $\overline{BSY}$ is deasserted AND $\overline{SEL}$ is asserted AND I/O is asserted AND the logical OR of SCSI ID is valid.

All SCSI signals are active-high on the DP8497.

7.0 A.C. Specifications (Continued)

7.21 SCSI SELECTION AS TARGET



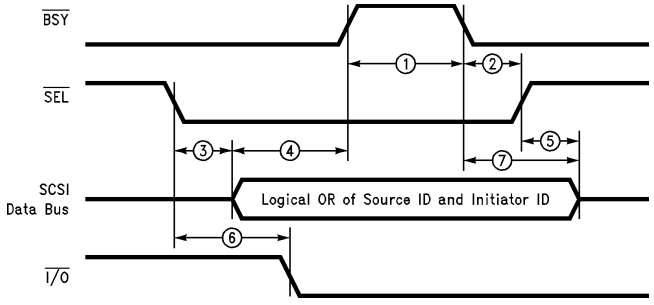
TL/F/11212–55

ID #	Symbol	Parameter	Optimal BCLK		Worst Case		Units
			Min	Max	Min	Max	
21.1	bihbol	$\overline{BSY}$ In Deasserted to $\overline{BSY}$ Out Asserted (Note 1)	400	800	400	1300	ns
21.2	blsh	$\overline{BSY}$ Out Asserted to $\overline{SEL}$ In Hold	0		0		ns
21.3	blih	$\overline{BSY}$ In Asserted to SCSI ID Hold	0		0		ns
21.4	shiov	$\overline{SEL}$ In Deasserted to $\overline{I/O}$, $\overline{C/D}$, $\overline{MSG}$ Asserted	100		100		ns

Note 1: This time starts when $\overline{BSY}$ is deasserted AND $\overline{SEL}$ is asserted AND $\overline{I/O}$ is asserted AND the logical OR of SCSI ID is valid.

All SCSI signals are active-high on the DP8497.

7.22 SCSI RESELECTION AS TARGET



TL/F/11212–56

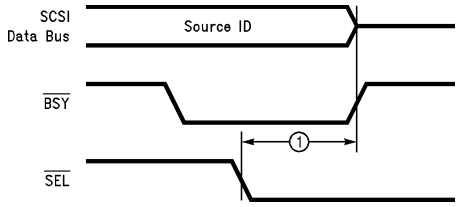
ID #	Symbol	Parameter	Optimal BCLK		Worst Case		Units
			Min	Max	Min	Max	
22.1	bobv	$\overline{BSY}$ Out Deasserted to $\overline{BSY}$ In Valid (Note 1)	400		400		ns
22.2	blsu	$\overline{BSY}$ In Asserted to $\overline{SEL}$ Out Deasserted	90		90		ns
22.3	sliv	$\overline{SEL}$ Out Asserted to SCSI ID Out Valid	1.2	1.4	1.2	2.1	μs
22.4	ivbo	SCSI ID Out Valid to $\overline{BSY}$ Out Deasserted	90	230	90	350	ns
22.5	shdz	$\overline{SEL}$ Out Deasserted to SCSI Data Deasserted	−30	30	−30	30	ns
22.6	sliol	$\overline{SEL}$ Out Asserted to $\overline{I/O}$ Asserted	1.2	1.4	1.2	2.1	μs
22.7	blidi	$\overline{BSY}$ in Asserted to SCSI ID Out In Valid	90		90		ns

Note 1: $\overline{BSY}$ may be asserted sooner, but it will not be detected until (bobv) time has passed.

All SCSI signals are active-high on the DP8497.

7.0 A.C. Specifications (Continued)

7.23 SCSI LOST ARBITRATION

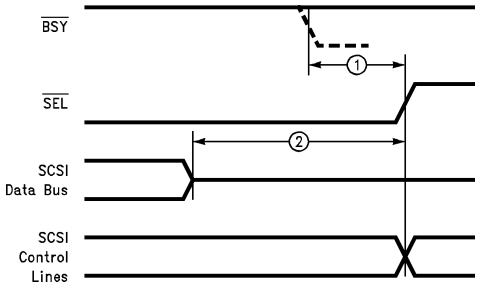


TL/F/11212–57

ID #	Symbol	Parameter	Min	Max	Units
23.1	slbf	$\overline{SEL}$ In Low to Bus Free	bcp	400	ns

Note: All SCSI signals are active-high on the DP8497.

7.24 SCSI ABORT (RE)SELECTION



TL/F/11212–58

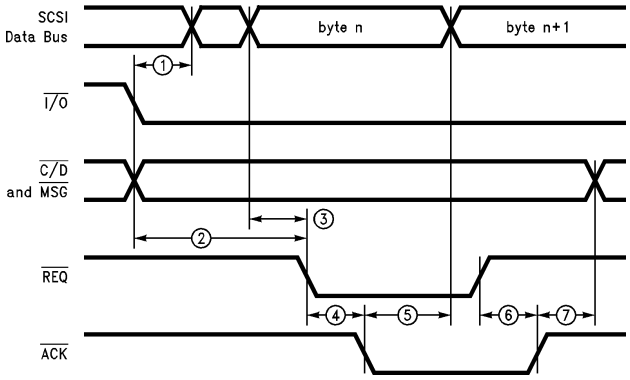
ID #	Symbol	Parameter	Optimal BCLK		Worst Case		Units
			Min	Max	Min	Max	
24.1	bss	$\overline{BSY}$ Setup to $\overline{SEL}$ (Last Chance) (Note 1)	1 bcp	4 bcp	1 bcp	4 bcp	ns
24.2	iuso	SCSI ID Released to $\overline{SEL}$ Out High	201	250	201	400	μs

Note 1: bcp refers to the BCLK period in use. This formula is tested only at the frequencies listed in Table 8.1.

All SCSI signals are active-high on the DP8497.

7.0 A.C. Specifications (Continued)

7.25 SCSI ASYNCHRONOUS INFORMATION IN PHASE AS INITIATOR



TL/F/11212–59

ID #	Symbol	Parameter	Min	Typ	Max	Units
25.1	iodz	I/O Valid to SCSI Data Released			4 bcp	ns
25.2	pcreq	Phase Change to $\overline{REQ}$	50			ns
25.3	sds	Data Setup to $\overline{REQ}$	0			ns
25.4	reqlackl	$\overline{REQ}$ Low to $\overline{ACK}$ Low (Notes 1 and 2)		90	185	ns
25.5	sdh	Data Hold from $\overline{ACK}$	0			ns
25.6	reqhackh	$\overline{REQ}$ High to $\overline{ACK}$ High (Note 3)		40	60	ns
25.7	phack	Phase Hold from $\overline{ACK}$ High	0			ns

Note 1: bcp is the BCLK period in use.

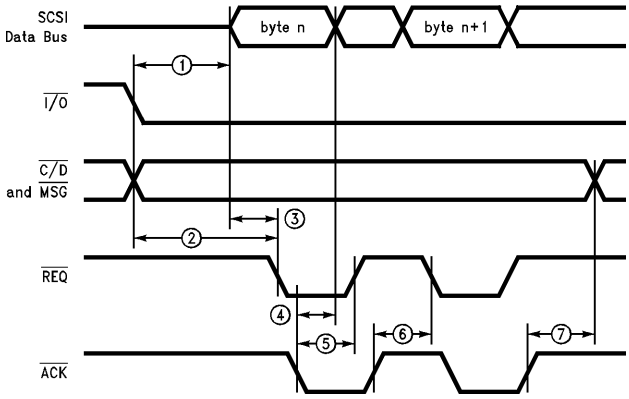
Note 2: Transfer Period (in Synchronous Transfer register) = 0. If Transfer Period = 1 then add an additional bcp.

Note 3: This value is not valid for very last transfer. $\overline{ACK}$ will stay asserted.

All SCSI signals are active-high on the DP8497.

7.0 A.C. Specifications (Continued)

7.26 SCSI ASYNCHRONOUS INFORMATION IN PHASE AS TARGET



TL/F/11212–60

ID#	Symbol	Parameter	Optimal BCLK			Worst Case		Units
			Min	Typ	Max	Min	Max	
26.1	iodz	$\overline{I/O}$ Valid to SCSI Data Driven	800		1100	800	1700	ns
26.2	pcreq	Phase Change to $\overline{REQ}$	400		550	400	800	ns
26.3	sds	Data Setup to $\overline{REQ}$	(Note 3)			(Note 3)		ns
26.4	sdh	Data Hold from $\overline{ACK}$	0			0		ns
26.5	acklreqh	$\overline{ACK}$ Low to $\overline{REQ}$ High		30	45		TBD	ns
26.6	ackhreql	$\overline{ACK}$ High to $\overline{REQ}$ Low (Notes 1 and 2)		115	185		TBD	ns
26.7	phack	Phase Hold from $\overline{ACK}$ High	40			40		ns

Note 1: bcp refers to the BCLK period in use. This formula is tested only at the frequencies listed in Table 8.1.

Note 2: Transfer Period (in Synchronous Transfer register) = 0. If Transfer Period = 1 then add an additional bcp.

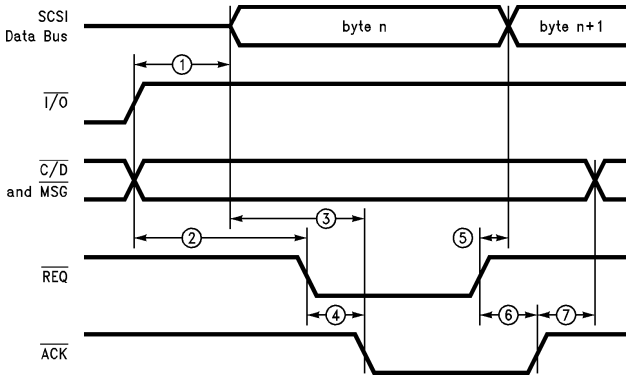
Note 3: This value is programmable. It is a functon of the BCLK Frequency bits in SUP1 register and the SCSI clock bits in SDIF2 register.
The SCSI 1 specification (including cable skew) is 55 ns. The SCSI 2 specification (including cable skew) is 25 ns.

		SCSI Clock Bits (1:0)			
		00	01	11	10
BCLK Freq (1:0)	00	1.0	1.0	2.0	1.5
	01	1.5	1.0	2.0	1.5
	10	1.5	1.0	2.0	1.5
	11	1.5	1.0	2.0	1.5

Where 1.0 is defined as bcp − k; k is a characterization constant
Where 1.5 is defined as bcp + bc − k; bc is either bcl or bch
Where 2.0 is defined as 2*bcp − k; k is a characterization constant
All SCSI signals are active-high on the DP8497.

7.0 A.C. Specifications (Continued)

7.27 SCSI ASYNCHRONOUS INFORMATION OUT PHASE AS INITIATOR



TL/F/11212–61

ID #	Symbol	Parameter	Min	Typ	Max	Units
27.1	iodz	$\overline{I/O}$ Valid to SCSI Data Valid	45			ns
27.2	pcreq	Phase Change to $\overline{REQ}$	50			ns
27.3	sds	Data Setup to $\overline{ACK}$	(Note 4)			ns
27.4	reqlackl	$\overline{REQ}$ Low to $\overline{ACK}$ Low (Note 1)		90	235	ns
27.5	sdh	Data Hold from $\overline{REQ}$	0			ns
27.6	reqhackh	$\overline{REQ}$ High to $\overline{ACK}$ High (Note 3)		40	60	ns
27.7	phack	Phase Hold from $\overline{ACK}$ High	0			ns

Note 1: Transfer Period (in Synchronous Transfer register) = 0. If Transfer Period = 1 then add an additional bcp.

Note 2: bcp refers to the BCLK period in use. This formula is tested only at the frequencies listed in Table 8.1.

Note 3: This value is not valid for the very last transfer. $\overline{ACK}$ will stay asserted.

Note 4: This value is programmable. It is a functon of the BCLK Frequency bits in SUP1 register and the SCSI clock bits in SDIF2 register. The SCSI 1 specification (including cable skew) is 55 ns. The SCSI 2 specification (including cable skew) is 25 ns.

		SCSI Clock Bits (1:0)			
		00	01	11	10
BCLK Freq (1:0)	00	1.0	1.0	2.0	1.5
	01	1.5	1.0	2.0	1.5
	10	1.5	1.0	2.0	1.5
	11	1.5	1.0	2.0	1.5

Where 1.0 is defined as bcp − k; k is a characterization constant

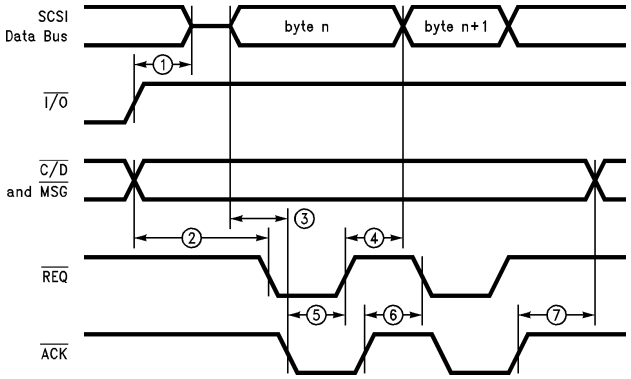
Where 1.5 is defined as bcp + bc − k; bc is either bcl or bch

Where 2.0 is defined as 2*bcp − k; k is a characterization constant

All SCSI signals are active-high on the DP8497.

7.0 A.C. Specifications (Continued)

7.28 SCSI ASYNCHRONOUS INFORMATION OUT PHASE AS TARGET



TL/F/11212–62

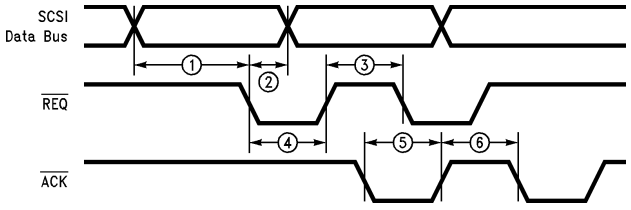
ID #	Symbol	Parameter	Optimal BCLK			Worst Case		Units
			Min	Typ	Max	Min	Max	
28.1	iodz	$\overline{I/O}$ Valid to SCSI Data Released			45		45	ns
28.2	pcreq	Phase Change to $\overline{REQ}$	400		550	400	800	ns
28.3	sds	Data Setup to $\overline{ACK}$	0			0		ns
28.4	sdh	Data Hold from $\overline{REQ}$	0			0		ns
28.5	acklreqh	$\overline{ACK}$ Low to $\overline{REQ}$ High		30	45		TBD	ns
28.6	ackhreql	$\overline{ACK}$ High to $\overline{REQ}$ Low (Notes 1 and 2)		70	135		TBD	ns
28.7	phack	Phase Hold from $\overline{ACK}$ High	40			40		ns

Note 1: bcp refers to the BCLK period in use. This formula is tested only at the frequencies listed in Table 8.1.

Note 2: Transfer Period (in Synchronous Transfer register) = 0. If Transfer Period = 1 then add an additional bcp.

All SCSI signals are active-high on the DP8497.

7.29 SCSI SYNCHRONOUS INFORMATION IN PHASE AS INITIATOR



TL/F/11212–63

ID #	Symbol	Parameter	Min	Max	Units
29.1	syds	Synchronous Data Setup	0		ns
29.2	sydh	Synchronous Data Hold (Note 1)	20		ns
29.3	reqdp	$\overline{REQ}$ Deassertion Period	20		ns
29.4	reqap	$\overline{REQ}$ Assertion Period	20		ns
29.5	ackap	$\overline{ACK}$ Assertion Period (Note 2)	90		ns
29.6	ackdp	$\overline{ACK}$ Deassertion Period (Note 2)	90		ns

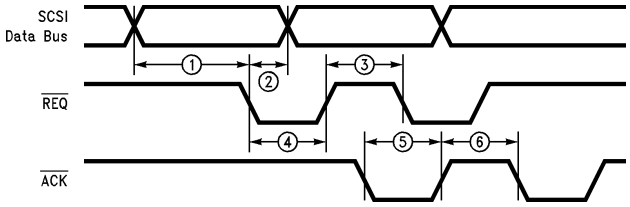
Note 1: This parameter meets the SCSI 1 specification of 45 ns. Characterization data will determine if the SCSI 2 specification of 10 ns can be obtained.

Note 2: The $\overline{ACK}$ assertion and deassertion time is programmable. The following equations determine these times:
For n even: 0.5 * n * bcp-k
For n odd: (0.5 * (n − 1) * bcp + bc)-k; where bc is bcl or bch
k is a characterization constant—target value of which is 10 ns.

All SCSI signals are active-high on the DP8497.

7.0 A.C. Specifications (Continued)

7.30 SCSI SYNCHRONOUS INFORMATION IN PHASE AS TARGET



TL/F/11212–64

ID #	Symbol	Parameter	Min	Max	Units
30.1	syds	Synchronous Data Setup (Note 1)	(Note 1)		ns
30.2	sydh	Synchronous Data Hold (Note 2)	(Note 2)		ns
30.3	reqdp	$\overline{REQ}$ Deassertion Period (Note 3)	(Note 3)		ns
30.4	reqap	$\overline{REQ}$ Assertion Period (Note 3)	(Note 3)		ns
30.5	ackap	$\overline{ACK}$ Assertion Period	20		ns
30.6	ackdp	$\overline{ACK}$ Deassertion Period	20		ns

Note 1: The setup time is programmable. The following equations determine these typical times:

For n = 2: bcp-k

For n = 4, 6: (0.5 * (n − 2) * bcp + bc)-k; where bc is bcl or bch

For n Odd: (0.5 * (n − 1) * bcp)-k

Note 2: The hold time is programmable. The following equations determine these typical times:

For n = 2: bcp-k

For n = 4, 6: (0.5 * n * bcp + bc)-k; where bc is bcl or bch

For n Odd: (0.5 * (n + 1) * bcp)-k

Note 3: The $\overline{ACK}$ assertion and deassertion time is programmable. The following equations determine these typical times:

For n Even: (0.5 * n * bcp)-k

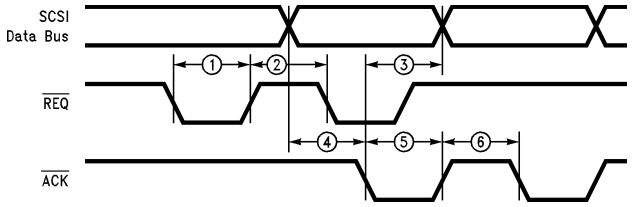
For n Odd: (0.5 * (n − 1) * bcp + bc)-k; where bc is bcl or bch

k is a characterization constant—target value of which is 10 ns.

All SCSI signals are active-high on the DP8497.

7.0 A.C. Specifications (Continued)

7.31 SCSI SYNCHRONOUS INFORMATION OUT PHASE AS INITIATOR



TL/F/11212–65

ID #	Symbol	Parameter	Min	Max	Units
31.1	reqap	$\overline{\text{REQ}}$ Assertion Period	20		ns
31.2	reqdp	$\overline{\text{REQ}}$ Deassertion Period	20		ns
31.3	sydh	Synchronous Data Hold (Note 2)	(Note 2)		ns
31.4	syds	Synchronous Data Setup (Note 1)	(Note 1)		ns
31.5	ackap	$\overline{\text{ACK}}$ Assertion Period (Note 3)	(Note 3)		ns
31.6	ackdp	$\overline{\text{ACK}}$ Deassertion Period (Note 3)	(Note 3)		ns

Note 1: The setup time is programmable. The following equations determine these typical times:

For n = 2: bcp-k

For n = 4, 6: (0.5 * (n − 2) * bcp + bc)-k; where bc is bcl or bch

For n Odd: (0.5 * (n − 1) * bcp)-k

Note 2: The hold time is programmable. The following equations determine these typical times:

For n = 2: bcp-k

For n = 4, 6: (0.5 * n * bcp + bc)-k; where bc is bcl or bch

For n Odd: (0.5 * (n + 1) * bcp)-k

Note 3: The $\overline{\text{ACK}}$ assertion and deassertion time is programmable. The following equations determine these typical times:

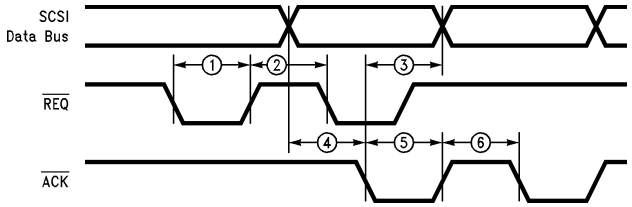
For n Even: (0.5 * n * bcp)-k

For n Odd: (0.5 * (n − 1) * bcp + bc)-k; where bc is bcl or bch

k is a characterization constant—target value for which is 10 ns.

All SCSI signals are active-high on the DP8497.

7.32 SCSI SYNCHRONOUS INFORMATION OUT PHASE AS TARGET



TL/F/11212–66

ID #	Symbol	Parameter	Min	Max	Units
32.1	reqap	$\overline{\text{REQ}}$ Assertion Period	(Note 1)		ns
32.2	reqdp	$\overline{\text{REQ}}$ Deassertion Period	(Note 1)		ns
32.3	sydh	Synchronous Data Hold	20		ns
32.4	syds	Synchronous Data Setup	0		ns
32.5	ackap	$\overline{\text{ACK}}$ Assertion Period	20		ns
32.6	ackdp	$\overline{\text{ACK}}$ Deassertion Period	20		ns

Note 1: The $\overline{\text{REQ}}$ assertion and deassertion time is programmable. The following equations determine these typical times:

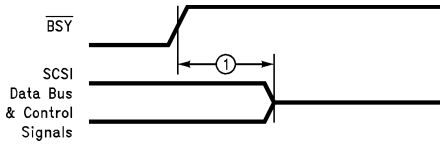
For n even: (0.5 * n * bcp)-k

For n odd: (0.5 * (n − 1) * bcp + bc)-k; where bc is bcl or bch

k is a characterization constant—target value for which is 10 ns.

7.0 A.C. Specifications (Continued)

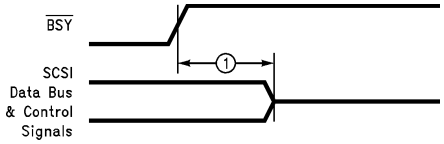
7.33 SCSI BUS FREE FROM INFORMATION PHASE AS INITIATOR



TL/F/11212–67

ID#	Symbol	Parameter	Optimal BCLK		Worst Case		Units
			Min	Max	Min	Max	
33.1	bihbf	BSY Deasserted to Bus Free	400	800	400	1200	ns

7.34 SCSI BUS FREE FROM INFORMATION PHASE AS TARGET

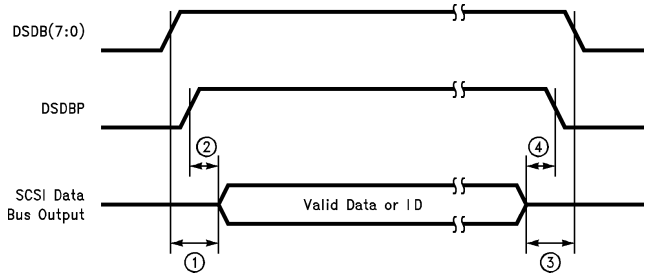


TL/F/11212–68

ID#	Symbol	Parameter	Min	Max	Units
34.1	bohbf	BSY Deasserted to Bus Free		200	ns

Note: All SCSI signals are active-high on the DP8497.

7.35 DIFFERENTIAL TRANSCEIVER DIRECTION CONTROL (DP8497 Only)



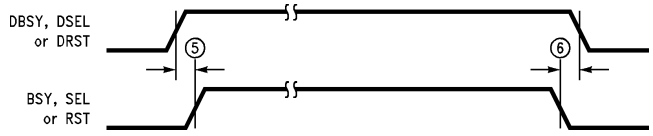
TL/F/11212–69

ID#	Symbol	Parameter	Arbitration		Selection		Info Transfer		Units
			Min	Max	Min	Max	Min	Max	
35.1	ddsad	Data Direction Signal Assertion to Data Valid	−30	30	−30	30	0		ns
35.2	pdsap	Parity Direction Signal Assertion to Parity Valid	N/A	N/A	−30	30	0		ns
35.3	diddsd	Data Invalid to Data Direction Signal Deassertion	−30	30	−30	30	0		ns
35.4	pipdsd	Parity Invalid to Parity Direction Signal Deassertion	N/A	N/A	−30	30	0		ns

Note: All SCSI signals are active-high on the DP8497.

7.0 A.C. Specifications (Continued)

7.35 DIFFERENTIAL TRANSCEIVER DIRECTION CONTROL (DP8497 Only) (Continued)



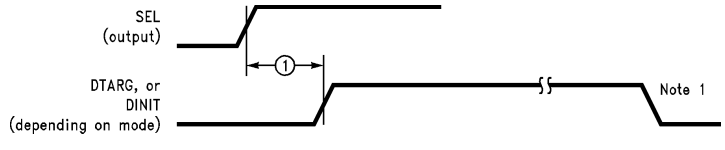
TL/F/11212–70

ID #	Symbol	Parameter	Min	Max	Units
35.5	bdsab sdsas rdsar	BSY Direction Signal Assertion to BSY Assertion SEL Direction Signal Assertion to SEL Assertion RST Direction Signal Assertion to RST Assertion	−30	30	ns
35.6	bdbdsd sdsdsd rdrdsd	BSY Deassertion to BSY Direction Signal Deassertion SEL Deassertion to SEL Direction Signal Deassertion RST Deassertion to RST Direction Signal Deassertion	−30	30	ns

Note: All SCSI signals are active-high on the DP8497.

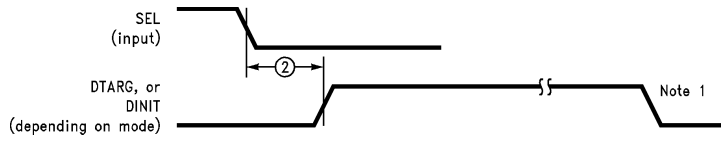
7.36 DIFFERENTIAL TRANSCEIVER DIRECTION CONTROL FOR TARGET AND INITIATOR SIGNALS (DP8497 Only)

a) DP8496/DP8497 Attempting SELECT or RESELECT



TL/F/11212–71

b) DP8496/DP8497 Being SELECTed or RESELECTed



TL/F/11212–72

ID #	Symbol	Parameter	Min	Max	Units
36.1	soatds soaids	SEL Output Assertion to TARG Direction Signal Assertion SEL Output Assertion to INIT Direction Signal Assertion	bcp − 30	bcp + 30	ns
36.2	sidtds sidids	SEL Input Assertion to TARG Direction Signal Assertion SEL Input Assertion to INIT Direction Signal Assertion	bcp	6*bcp	ns

Note 1: DTARG and DINIT are deasserted in the BUS FREE phase.

Note 2: bcp refers to the Bus Clock Period in use.

All SCSI signals are active-high on the DP8497.

8.0 A.C. Test Conditions

The A.C. Characteristics in Section 7.0 are tested under the conditions described below. If a variable is included in the Min or Max column, the value of that variable should be based on the conditions used in the current application—not the Min or Max specification from a table.

The AC specifications are tested at different BCLK and RCLK frequencies. The frequencies tested are listed in Table 8.1. If a variable is included in the Min or Max column, you may calculate an estimate for that characteristic for any frequency. However, only the frequencies listed in Table 8.1 are guaranteed.

TABLE 8.1. Guaranteed Frequencies

BCLK
15 MHz
17.5 MHz
20 MHz

TABLE 8.2. Test Conditions

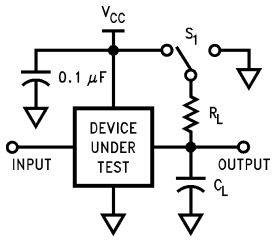
Input Pulse Levels	GND to 3V
Input Rise and Fall Times	5 ns
Input and Output Reference Levels	1.3V
TRI-STATE Reference Levels	Active High − 0.5V
	Active Low + 0.5V

TABLE 8.3. Capacitance

Symbol	Parameter	Typ	Units
C_{IN}	Input Capacitance	5	pF
C_{OUT}	Output Capacitance	8	pF

Note 1: $T_A = 25°C$, $f = 1$ MHz.

Note 2: These parameters are not 100% tested.



TL/F/11212–9

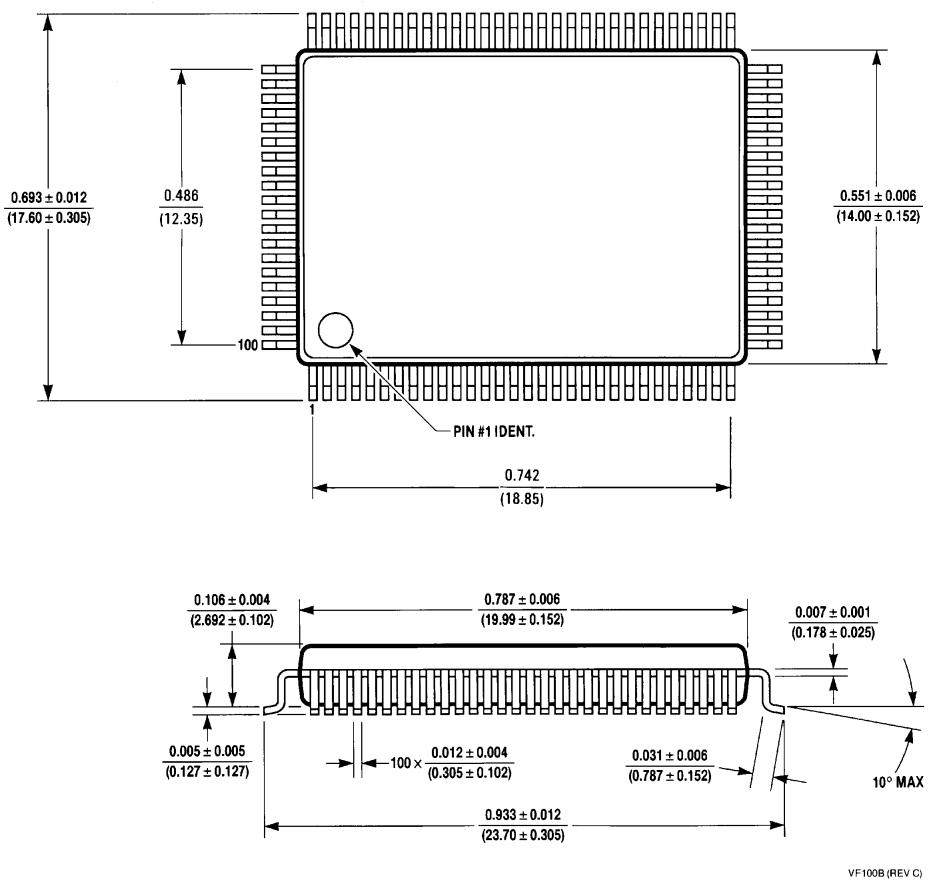
FIGURE 8.1. Test Jig

Note 1: $C_L = 110$ pF for Buffer Memory Interface pins; 50 pF for all other pins. Includes jig and scope capacitance.

Note 2: S1 = open for push-pull outputs. S1 = V_{CC} for high impedance to active low and active low to high impedance measurements. S1 = GND for high impedance to active high and active high to high impedance measurements. $R_L = 1.0$ kΩ (See Note 3).

Note 3: For the SCSI interface pins (SCSI Data bus and SCSI Control lines) S1 = V_{CC} and $R_L = 150\Omega$.

Physical Dimensions inches (millimeters)



Plastic Quad Flatpack, EIAJ, 100 Lead
Order Number DP8496VF or DP8497VF
NS Package Number VF100B

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
1111 West Bardin Road
Arlington, TX 76017
Tel: 1(800) 272-9959
Fax: 1(800) 737-7018

National Semiconductor Europe
Fax: (+49) 0-180-530 85 86
Email: cnjwge@tevm2.nsc.com
Deutsch Tel: (+49) 0-180-530 85 85
English Tel: (+49) 0-180-532 78 32
Français Tel: (+49) 0-180-532 93 58
Italiano Tel: (+49) 0-180-534 16 80

National Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: (852) 2737-1600
Fax: (852) 2736-9960

National Semiconductor Japan Ltd.
Tel: 81-043-299-2309
Fax: 81-043-299-2408