Section 3

Rigid Disk Data Controller

Section 3 Contents

National Semiconductor

DP8466A Disk Data Controller

General Description

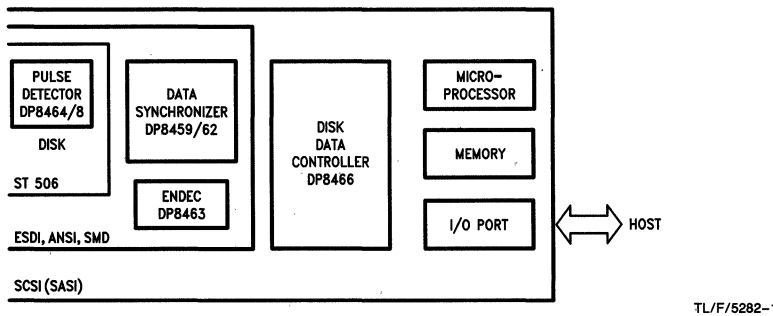
The DP8466A Disk Data Controller (DDC) is an intelligent peripheral which interfaces Winchester or Floppy disk drives to microprocessor based systems. It transfers data between a buffer memory or host system and the serial bit data stream with disk rates up to 25M-bits per second. High speed system data transfer is possible with full on-chip DMA control of buffer or main memory. The 16-bit system I/O interface allows use with any popular 8-bit, 16-bit or 32-bit microprocessor. Programmable track format enables reconfiguration of the DDC for different drive types in a multiple drive environment. Using other National DP8460 series disk data path chips, the DP8466A conforms to ST506, SMD and ESDI standard drive interfaces, as well as to intelligent standard interfaces such as SCSI (SASI) and IPI.

The DP8466A is available in three performance versions DP8466AN-12, DP8466AN-20 and DP8466AN-25.

Part Number	Max Disk Data Rate	Max DMA Transfer Rate
DP8466AN-25	25 Mbit/sec	10 Mbyte/sec
DP8466AN-20	20 Mbit/sec	8 Mbyte/sec
DP8466AN-12	12 Mbit/sec	6 Mbyte/sec

Features

- Easily conforms to any standard drive interface
- Compatible with floppy, hard and optical disk drives
- Compatible with 8, 16 or 32-bit microprocessor systems
- Programmable disk format
- Sector lengths up to 64k bytes, with up to 255 sectors per track
- Programmable 32 or 48-bit ECC polynomial
- Internal ECC correction in less than a sector time
- Disk data rate to 25M bits per second
- Multiple sector transfer capability
- 32 byte internal FIFO data buffer with interleavable burst capability
- 8 or 16-bit wide data transfers
- Single 32-bit or dual 16-bit DMA channel addresses
- Up to 10M bytes per second DMA transfer rate
- +5V supply, 48 pin DIP, microCMOS process



TL/F/5282–1

FIGURE 1. Typical System Configuration

Table of Contents

1.0 Introduction

National's DP8466A Disk Data Controller (DDC) chip is designed to concentrate only on the data aspects of a disk system, leaving the control signals to either a low cost single chip controller or an I/O port from a microprocessor. For this reason, the DDC will work with any standard drive interface.

The DP8466A is an advanced VLSI chip, fabricated in National's latest 2 μ CMOS technology, that allows for operation with disk data rates from the slowest floppy to the fast Winchester and Optical data rates of 25 megabits per second.

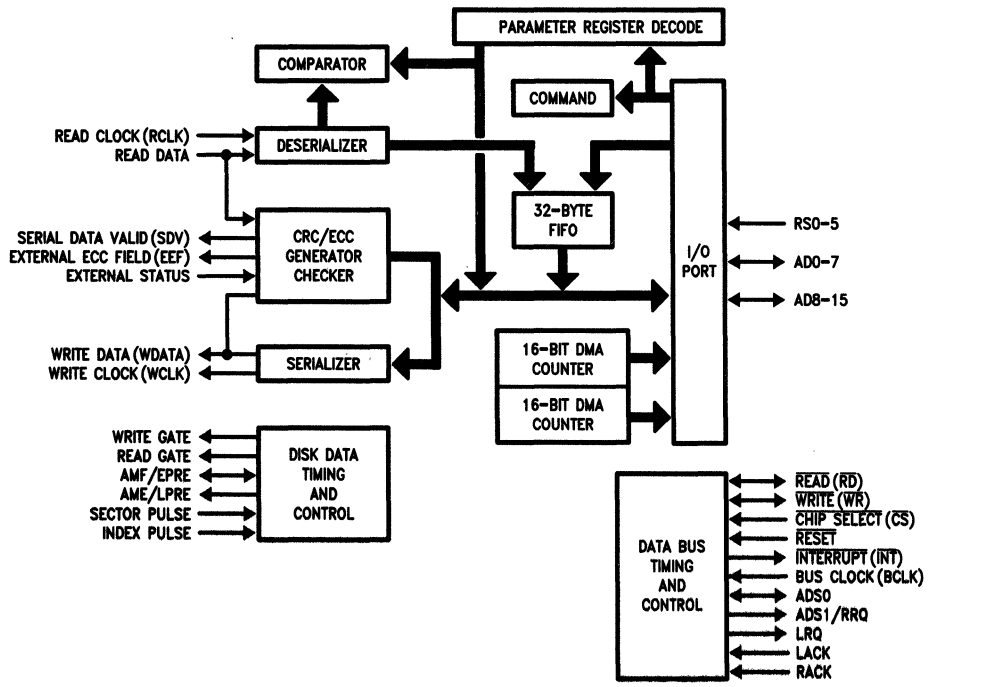
The CMOS design significantly helps the system designer because of reduced power consumption. The chip typically consumes 100 mW.

The DDC is designed for maximum programmability that not only allows the user to select any drive type he wishes, but also allows for different types of drives to be used on the same system. The chip contains 64 registers that can be loaded at any time by a microprocessor connected to the chip's bus. These registers determine the number of bytes in each field of the format, and the byte pattern that each of these fields will repeat. The number of data bytes per sector is selectable from 1 byte to 64k bytes. Finally, both the header field and the data field can each be appended with either a Cyclic Redundancy Check (CRC) field (the 16-bit code used on floppies) or a programmable Error Check and Correct (ECC) field.

The DDC allows the user to load in any 32 or 48-bit ECC polynomial from the microprocessor along with the format parameters. Once an error has been detected, the microprocessor decides whether to re-read the sector during the next revolution of the disk, or to attempt a correction. The DDC can correct errors in a time shorter than that required to read the next sector.

Key blocks in the DDC include a 32-byte FIFO and two 16-bit DMA channels that give the chip a 10 megabyte per second memory transfer capability. This high system data throughout is needed for the high speed drives now becomming available. The small FIFO allows for bursts of data to take place on the bus, thereby leaving the bus free for useful periods of time. The threshold for FIFO data storage is selectable to allow for some degree of system latency. The DDC allows for bursts of 2, 8, 16 or 24 bytes of data to be transferred between the FIFO and memory. The width of the data bus is selectable for either 8 or 16-bit transfers. The system designer selects the threshold so that when the FIFO contains the selected amount of data, the DDC will issue a request. The CPU can continue its operation and then stop to acknowledge the DDC, which then bursts the data between FIFO and memory, before the FIFO has time to overflow or underflow. With a 10 megabit per second disk data rate and a 10 megabyte per second memory transfer cycle, the bus will only be occupied for one-eighth of the time transferring data between FIFO and memory. This leaves the bus free for microprocessor usage for over 80% of the time.

Block Diagram

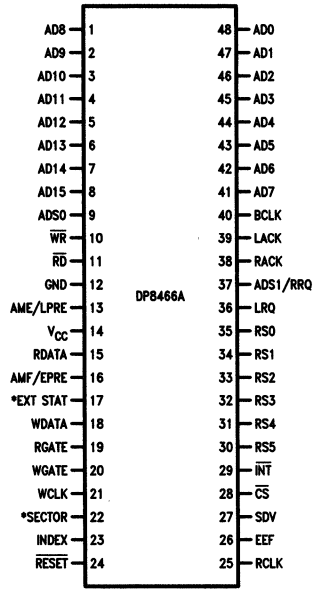


TL/F/5282-2

FIGURE 2. DDC

Connection Diagrams

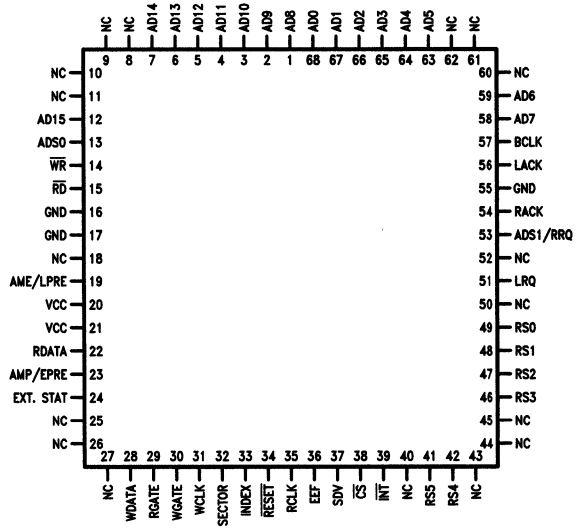
Dual-In-Line Package



TL/F/5282-3

Top View

*This pin must be grounded if not used.



TL/F/5282-76

Top View

Order Number DP8466AN
See NS Package Number N48A

Order Number DP8466AV
See NS Package Number V68A

FIGURE 3

2.0 Pin Descriptions

2.1 BUS INTERFACE PINS

Symbol	DIP Pin No.	PCC Pin No.	Type	Function
$\overline{CS}$	28	38	I	**CHIP SELECT:** Sets DDC as a standard I/O port for reading and writing registers. Configures $\overline{RD}$ and $\overline{WR}$ pins as inputs when DMA is inactive. This pin is ignored if on-chip DMA is enabled and performing a transfer.
$\overline{INT}$	29	39	O	**INTERRUPT:** An interrupt can be generated on any error, or after completion of a command, a correction cycle or any header operation.
$\overline{RESET}$	24	34	I	**RESET:** Clears FIFO, Status and Error registers. Halts DMA immediately. Halts disk read and write immediately. Does not affect parameter and most count and command registers. On power-up, must be held low for at least 32 RCLK cycles and 4 BCLK cycles. Note that both RCLK and BCLK must be active for the reset cycle to complete.
$\overline{RD}$	11	15	I/O	**READ:** • MICROPROCESSOR ACCESS MODE, with $\overline{CS}$ pin low and DMA inactive (RACK AND LACK low): Places data from FIFO or register as selected by pins RS0–5 onto the AD0–7 bus. • SLAVE MODE, with LACK pin high: Places data from FIFO onto the AD0–7/AD0–15 bus. • MASTER MODE: When DMA is active, $\overline{RD}$ pin enables data from the addressed device onto the address/data bus.
$\overline{WR}$	10	14	I/O	**WRITE:** • MICROPROCESSOR ACCESS MODE, with $\overline{CS}$ low and DMA inactive (RACK and LACK low): Latches data from AD0–7 bus to internal registers selected by RS0–5. • SLAVE MODE, with LACK pin high: Latches data from AD0–7/AD0–15 bus to FIFO. • MASTER MODE: When DMA is active, $\overline{WR}$ pin enables data from the address/data bus to the addressed device.

2.0 Pin Descriptions (Continued)

2.1 BUS INTERFACE PINS (Continued)

Symbol	DIP Pin No.	PCC Pin No.	Type	Function
BCLK	40	57	I	**BUS CLOCK:** Used as a reference clock when DDC is bus master. Used only during reset and DMA operations. Maximum ratio of RCLK/BCLK is 4 for Word Mode, and 2 for Byte Mode.
RACK	38	54	I	**REMOTE DMA ACKNOWLEDGE:** System input granting use of the bus for a remote DMA bus cycle. If RACK is de-asserted during a transfer, the current transfer cycle will complete.
LACK	39	56	I	**LOCAL DMA ACKNOWLEDGE:** System input granting use of bus for a local DMA bus cycle. If LACK is deasserted during a transfer, the current transfer cycle will complete. LACK has priority over RACK.
RS0–5	35-30	41, 42 46-49	I	**REGISTER SELECT:** Used as address inputs to select internal registers when $\overline{CS}$ pin is low.
AD0–7	48-41	58, 59 63–68	I/O	**ADDRESS/DATA 0–7:** These pins float if $\overline{CS}$ pin = 1 and DMA is inactive. • STANDARD I/O PORT, With DMA inactive and $\overline{CS}$ pin low: Command, Parameter, Count and Status register data is transferred. • SLAVE MODE, with external DMA controller active and LACK pin high: D0–7 are transferred between FIFO and memory. • MASTER MODE, with internal DMA active, and LACK pin high: A16–23, A0–7 and D0–7 are transferred depending on DMA mode and bus phase.
LRQ	36	51	O	**LOCAL DMA REQUEST:** Requests are automatically generated when the FIFO needs to have data transferred.
AD8–15	1–8	1–7 12	I/O	**ADDRESS/DATA 8–15:** • STANDARD I/O PORT, with DMA inactive and $\overline{CS}$ pin low: These pins are driven high. • SLAVE MODE, with external DMA active and LACK pin high: D8–15 are transferred between FIFO and memory. • MASTER MODE, with internal DMA active and LACK pin high: A24–31, A8–15 and D8–15 are transferred, depending on DMA mode and bus phase.
ADS0	9	13	I/O	**ADDRESS STROBE 0:** • INPUT with DMA inactive: ADS0 latches RS0–5 inputs when low. When high, data present on RS0–5 will flow through to internal register decoder. • OUTPUT: ADS0 latches low order address bits (A0–15) to external memory during DMA transfers.
ADS1/RRQ	37	53	O	**ADDRESS STROBE 1/REMOTE REQUEST:** In 32-bit DMA Mode, ADS1 latches high order address bits (A16–31) to external memory. For remote DMA modes, RRQ pin is active high when SRI or SRO bits in the OC register are set in non-tracking mode, or during a remote transfer in tracking mode. (See RT register description in DMA REGISTERS Section.)

2.2 DISK INTERFACE PINS

Symbol	DIP Pin No.	PCC Pin No.	Type	Function
RCLK	25	35	I	**READ CLOCK:** Disk data rate clock. When RGATE is high, RCLK input will be the recovered/separated clock from the recorded data and is used to strobe data into the DDC. When RGATE is low, this input should become the referenced clock which will be delayed and is used as WCLK to strobe data to the drive. The transition between the recovered/separated clock and reference clock must be made with no short pulses. Short pulses are pulses that are less than the specified minimum RCLK pulse widths which are specified in the AC timing section as rcl and rch. In the event of any short pulses on RCLK or if RCLK is inactive for greater than 10 μs, then the DDC could go into an indeterminant state. If this happens, then the DDC needs to be reset and the format parameters must be updated to ensure normal operation. Maximum ratio of RCLK/BCLK is 4 for word mode, and 2 for byte mode.

2.0 Pin Descriptions (Continued)

2.2 DISK INTERFACE PINS (Continued)

Symbol	DIP Pin No.	PCC Pin No.	Type	Function
RGATE	19	29	O	**READ GATE:** Set active high during any disk read operation. This pin commands data separator to acquire lock. Enables RDATA input pin.
RDATA	15	22	I	**READ DATA:** Accepts NRZ disk data from the data separator/decoder.
WCLK	21	31	O	**WRITE CLOCK:** Used when NRZ data is on WDATA pin. Also active when MFM data is used, but normally not utilized. WCLK frequency follows RCLK pin.
WGATE	20	30	O	**WRITE GATE:** When writing data onto a disk, WGATE is asserted high with the first bit of data and deasserted low after the last bit of data. WGATE is also de-asserted on reset or on detection of an error.
WDATA	18	28	O	**WRITE DATA:** During any write operation, MFM or NRZ encoded data is output to disk, dependent upon MFM bit status in the DF register. This pin is inactive low when WGATE is low.
AMF/EPRE	16	23	I/O	**ADDRESS MARK FOUND/EARLY PRECOMPENSATION:** Address mark input is monitored if the HSS bit in the DF register is low (for soft sectoring). If the MFM bit in the DF register and the EP bit in the OC register are both set, then this pin becomes the EPRE control. If both functions are used, WGATE pin determines the function as follows: • WGATE asserted: EPRE output. • WGATE de-asserted: AMF input.
AME/LPRE	13	19	O	**ADDRESS MARK ENABLE/LATE PRECOMPENSATION:** If the MFM bit in the DF register is low, AME will indicate that an address mark byte(s) is being output on WDATA pin. If the MFM bit in the DF register and the EP bit in the OC register are both set, LPRE control is output (if internal MFM encoding is used).
SECTOR	22	32	I	**SECTOR PULSE:** In hard sectored drives, this signal comes from the start of a sector. In a soft sectored drive this pin must be tied low.
INDEX	23	33	I	**INDEX PULSE:** This signal comes from the disk drive, indicating the start of a track.
SDV	27	37	O	**SERIAL DATA VALID:** Asserted when the DDC is either issuing or receiving header field, internal header CRC/ECC, data field, or internal data CRC/ECC information. Mainly used for external ECC and diagnostics.
EEF	26	36	O	**EXTERNAL ECC FIELD:** Only used if the External ECC Byte Count register(s) are non-zero. Asserted when external ECC check bits are being generated (WGATE high) and checked (RGATE high).
EXT STAT	17	24	I	**EXTERNAL STATUS:** *IMPORTANT NOTE: This pin MUST be tied low if it is not to be used.* This pin has three functions: • 1: If EEW bit in the RT register is set, the read and write strobes are extended for both remote and local transfers as long as this pin is high. This is the External Wait State function. • 2: If the EEW bit in the RT register is low, this pin will accept a pulse granting valid byte alignment on the last bit of the synch byte before header or data bytes. This is an OR function with the internal synch detect. • 3: External ECC Check. Only used if External ECC Byte Count register(s) are non-zero, and EEW bit in the RT register is low. After the last byte of external ECC, this pin will accept a pulse confirming that there has been no error. A CRC/ECC error will be flagged if this pulse is not received.
V_{CC} GND	14, 12	20, 21 16, 17		**POWER, GROUND:** +5V DC is required. It is suggested that a decoupling capacitor be connected between these pins. It is essential to provide a path to ground for the GND pin with the lowest possible impedance. Otherwise any voltage spikes resulting from transient switching currents will be reflected in the logic levels of the output pins.

3.0 Internal Registers of the DDC

The numerous registers within the DDC are presented below, grouped according to their function. A key is given as an aid for the use of each register. The key data is only suggested for common operation, and should not be considered as an absolute requirement. Following this listing is a description of each register, in the order of which they are listed below. The HA column at the left of this listing gives the Hex Address of each register.

KEY

- D May be updated when a different drive type is selected
- C May be updated before each command
- R May be read at any idle time
- F Used during formatting
- I Used during initialization
- NO Operation is not possible

COMMAND

HA	Register	Bits	Write	Read
10	Drive Command Register (DC)	8	C	NO
11	Operation Command Register (OC)	8	C	NO
35	Disk Format Register (DF)	8	D	NO
00	Status Register (S)	8	NO	R
01	Error Register (E)	8	NO	R
12	Sector Counter (SC)	8	C	R
13	Number of Sector Operations Counter (NSO)	8	C	R
0F	Header Byte Count (HBC)/Interlock	3	F	R
36	Header Diagnostic Readback (HDR)	8	NO	R

DMA

HA	Register	Bits	Write	Read
37	DMA Sector Counter (DSC)	8	NO	R
37	Remote Transfer Register (RT)	8	I	NO
36	Local Transfer Register (LT)	8	I	NO
1A	Remote Data Byte Count (L)	8	C	R
1B	Remote Data Byte Count (H)	8	C	R
1C	DMA Address Byte 0	8	C	R
1D	DMA Address Byte 1	8	C	R
1E	DMA Address Byte 2	8	C	R
1F	DMA Address Byte 3	8	C	R

FORMAT (See Note)

HA	Register	Bits	Write	Read
21	ID Preamble Byte Count	5	D	R
31	ID Preamble Pattern	8	D	R
22	ID Synch #1 (AM) Byte Count	5	D	R
32	ID Synch #1 (AM) Pattern	8	D	R
23	ID Synch #2 Byte Count	5	D	R
33	ID Synch #2 Pattern	8	D	R
24	Header Byte 0 Control Register (HC0)	5	D	R
14	Header Byte 0 Pattern	8	D	R
25	Header Byte 1 Control Register (HC1)	5	D	R
15	Header Byte 1 Pattern	8	D	R

FORMAT (Continued)

HA	Register	Bits	Write	Read
26	Header Byte 2 Control Register (HC2)	5	D	R
16	Header Byte 2 Pattern	8	D	R
27	Header Byte 3 Control Register (HC3)	5	D	R
17	Header Byte 3 Pattern	8	D	R
28	Header Byte 4 Control Register (HC4)	5	D	R
18	Header Byte 4 Pattern	8	D	R
29	Header Byte 5 Control Register (HC5)	5	D	R
19	Header Byte 5 Pattern	8	D	R
2B	ID External ECC Byte Count	5	D	R
2C	ID Postamble Byte Count	5	D	R
3C	ID Postamble Pattern	8	D	R
2D	Data Preamble Byte Count	5	D	R
3D	Data Preamble Pattern	8	D	R
2E	Data Synch #1 (AM) Byte Count	5	D	R
3E	Data Synch #1 (AM) Pattern	8	D	R
2F	Data Synch #2 Byte Count	5	D	R
3F	Data Synch #2 Pattern	8	D	R
3B	Data Format Pattern	8	F	R
38	Sector Byte Count L	8	D	R
39	Sector Byte Count H	8	D	R
2A	Data External ECC Byte Count	5	D	R
20	Data Postamble Byte Count	5	D	R
30	Data Postamble Pattern	8	D	R
34	Gap Byte Count	8	F	R
3A	Gap Pattern	8	F	R

CRC/ECC

HA	Register	Bits	Write	Read
02	ECC SR Out 0	8	NO	R
03	ECC SR Out 1	8	NO	R
04	ECC SR Out 2	8	NO	R
05	ECC SR Out 3	8	NO	R
06	ECC SR Out 4	8	NO	R
07	ECC SR Out 5	8	NO	R
02	Polynomial Preset Byte 0 (PPB0)	8	D	NO
03	Polynomial Preset Byte 1 (PPB1)	8	D	NO
04	Polynomial Preset Byte 2 (PPB2)	8	D	NO
05	Polynomial Preset Byte 3 (PPB3)	8	D	NO
06	Polynomial Preset Byte 4 (PPB4)	8	D	NO
07	Polynomial Preset Byte 5 (PPB5)	8	D	NO
08	Polynomial Tap Byte 0 (PTB0)	8	D	NO
09	Polynomial Tap Byte 1 (PTB1)	8	D	NO
0A	Polynomial Tap Byte 2 (PTB2)	8	D	NO
0B	Polynomial Tap Byte 3 (PTB3)	8	D	NO
0C	Polynomial Tap Byte 4 (PTB4)	8	D	NO
0D	Polynomial Tap Byte 5 (PTB5)	8	D	NO
0E	ECC/CRC Control (EC)	8	D	NO
08	Data Byte Count L	8	NO	R
09	Data Byte Count H	8	NO	R

3.0 Internal Registers of the DDC (Continued)

DUAL-PURPOSE REGISTERS

Some of the above listed registers have dual functions depending on whether they are being written to or read from. These registers are repeated below to help clarify their operation.

HA	Register	Bits	Write	Read
02	ECC SR Out 0	8	NO	R
02	Polynomial Preset Byte 0 (PPB0)	8	D	NO
03	ECC SR Out 1	8	NO	R
03	Polynomial Preset Byte 1 (PPB1)	8	D	NO
04	ECC SR Out 2	8	NO	R
04	Polynomial Preset Byte 2 (PPB2)	8	D	NO
05	ECC SR Out 3	8	NO	R
05	Polynomial Preset Byte 3 (PPB3)	8	D	NO
06	ECC SR Out 4	8	NO	R
06	Polynomial Preset Byte 4 (PPB4)	8	D	NO
07	ECC SR Out 5	8	NO	R
07	Polynomial Preset Byte 5 (PPB5)	8	D	NO
08	Polynomial Tap Byte 0 (PTB0)	8	D	NO
08	Data Byte Count (0)	8	NO	R
09	Polynomial Tap Byte 1 (PTB1)	8	D	NO
09	Data Byte Count (1)	8	NO	R
36	Header Diagnostic Readback (HDR)	8	NO	R
36	Local Transfer Register (LT)	8	I	NO
37	DMA Sector Counter (DSC)	8	NO	R
37	Remote Transfer Register (RT)	8	I	NO

Format Note: It is recommended that the Format Registers be reloaded after the following events:

1. A hardware or software reset of the chip
2. A Sector Not Found error
3. A Sector Overrun error
4. A Data Sync Error

3.1 COMMAND REGISTERS

DRIVE COMMAND (DC) Hex Address (10) Write Only

The locations within this register, when written to, initiate disk commands and chip functions. For a disk operation, after the DDC has been configured, this register is loaded to initiate command execution.

Loading the DC register constitutes the initiation of a disk operation and will hence generate an operation complete interrupt.

DO2	DO1	H02	H01	FMT	MSO	SAIS	RED
7	6	5	4	3	2	1	0

RED: Re-enable DDC

A 1 should be written into this location during the power up initialization process (see POWER UP AND INITIALIZATION Section), or after an error has been encountered in order to re-enable the DDC to accept commands. (NOTE: If the RES bit in the OC register has been set, a 0 should be written to that location before this operation is performed.) If no error has been encountered, and a command is being issued, a zero should be written to this bit. The Re-enable is an operation by itself and hence an interrupt will be generated on completion of the operation.

SAIS: Start at Index or Sector

0 Operation begins only upon receipt of an index pulse.

1 Operation begins on either an index pulse or sector pulse for hard sector drives or immediately for soft sector drives.

MSO: Multi-sector Operation

0 Single-sector operation.

1 Multi-sector operation using NSO register.

FMT: Format Mode

0 No Format Operation.

1 When set, along with other DC register bits, will initiate disk formatting upon receipt of an index pulse.

HO1, 2: Header Operation Bits:

H02 H01

0 0 *IGNORE HEADER:* associated data transfer operation will take place with any valid sector encountered.

0 1 *COMPARE HEADER:* Normal mode used to find a specific sector. The Header Pattern registers contain the comparison pattern.

1 0 *WRITE HEADER (Write ID):* Normally used only during Format mode to write ID patterns to disk.

1 1 *READ HEADER (Read ID):* Reads header information from disk for diagnostic purposes.

DO1, 2: Data Operation Bits:

D02 D01

0 0 *NO OPERATION:* Can be used only with an Ignore Header command. No disk operation is performed with this combination, and it can be used along with the RED command to re-enable the DDC (see OPERATING MODES).

0 1 *CHECK DATA:* No DMA action and no data movement between disk and FIFO. CRC/ECC checks are calculated and interrupts, if enabled, are asserted on proper conditions. DFE bit in Error register will be set if a data CRC/ECC error occurs unless in Interlock Mode.

1 0 *WRITE DATA:* Initiates local DMA action to fill the FIFO. Writes data to disk with the proper pre and post appendages in the data field. FIFO is replenished by local DMA.

1 1 *READ DATA:* Data enters FIFO from disk, and local DMA transfer is initiated when the FIFO contains the number of bytes specified by the Burst Length in the LT register.

The following table shows a list of valid commands combining the H01, H02, D01, D02, FMT bits from the DC register and the FTF bit in the DF register. No other DC register combinations are allowed.

3.0 Internal Registers of the DDC (Continued)

Valid DDC Commands

DC Register D02	D01	H02	H01	FMT	DF Reg FTF	Operation
0	0	0	0	0	X	No Operation
0	1	0	1	0	X	Check Data, Compare Header
0	1	1	0	0	X	Check Data, Write Header
0	1	1	1	0	X	Check Data, Read Header
1	0	0	0	0	X	Write Data, Ignore Header
1	0	0	1	0	X	Write Data, Compare Header, (normal write)
1	0	1	0	0	X	Write Data, Write Header
1	0	1	0	1	0	Write Data,Write Header, Format with No FIFO Table
1	0	1	0	1	1	Write Data, Write Header, FIFO Table Format
1	1	0	0	0	X	Read Data, Ignore Header, (recover data)
1	1	0	1	0	X	Read Data, Compare Header, (normal read)
1	1	1	1	0	X	Read Data, Read Header

OPERATION COMMAND (OC)
Hex Address (11) **Write Only**

The fields within this register enable on-chip operations. In non-tracking mode, a remote DMA operation will be initiated by loading the SRO or SRI bits in this register.

IR	SCC	EP	SRO	SRI	EHI	EI	RES
7	6	5	4	3	2	1	0

RES: Reset DDC

0 Clears a previously set RES function. Allows normal operation.

1 DDC immediately enters a stand-by mode. The FIFO is reset, Status and Error registers are cleared and all operations in progress are stopped. DDC is placed in the Reset mode (see OPERATING MODES). RGATE and WGATE pins are de-asserted if active. All DMA counters are cleared. Format Parameter, DMA Address and ECC registers are unaffected.

EI: Enable Interrupts

0 Disabled, INT pin remains inactive high.

1 Enables interrupts generated by the following:

- Correction cycle complete.
- Error which sets ED bit in Status register.
- Command successfully completed (including independent remote DMA transfer).

EHI: Enable Header Interrupt

EI bit must be set if this bit is set.

0 Disabled.

1 Interrupt issued at start of ID postamble field when:

- Header matches in Compare Header operation.
- Header finished in Read, Write or Ignore Header operation.

SRI, SRO: Start Remote Input, Start Remote Output

These bits are only operational in non-tracking mode. The Remote Start Address and Remote Data Byte Count registers must be loaded first.

SRI SRO

0 0 Remote DMA operation unchanged.

0 1 *START REMOTE OUTPUT*: Asserts RRQ pin and RCB flag in Status register, to begin a remote DMA operation from memory to I/O Port.

1 0 *START REMOTE INPUT*: Asserts RRQ pin and RCB flag in Status register, to begin a remote DMA operation from I/O Port to local memory.

1 1 *STOP CURRENT REMOTE OPERATION*: RRQ pin is de-asserted and RCB flag is reset in Status register.

EP: Enable Precompensation

0 Early and late precompensation signals are forced low during a disk write operation.

1 Permits precompensation signals to be output to external precompensation circuitry (see MFM ENCODED DATA). This bit is only valid if the MFM bit is set in the DF register.

SCC: Start Correction Cycle

0 No correction is attempted.

1 Setting this command will begin the internal correction cycle. The CCA flag in the Status register is set and drive commands should not be issued during this time. At the completion of the cycle, an interrupt is issued.

IR: Interlock Required (Interlock Mode)

0 No interlock function.

1 The interlock (HBC) register must be written to after the header operation has completed and before the DDC encounters the data postamble field. This allows updating of header bytes during a Format operation or changing of drive commands during a multi-sector operation. Normally used with the header interrupt enabled.

3.0 Internal Registers of the DDC (Continued)

DISK FORMAT (DF) Hex Address (35) Write Only

ID2	1D1	IH2	1H1	FTF	HSS	SAM	MFM
7	6	5	4	3	2	1	0

MFM: MFM Encode

(See MFM Encoded Data section.)

0 NRZ data is output on the WDATA pin when WGATE is active.

1 MFM data is output on the WDATA pin when WGATE is active. Also configures AMF/EPRE and AME/LPRE pins as EPRE and LPRE outputs when Write Gate is active. Precompensated outputs are enabled by the EP bit in the OC register.

SAM: Start with Address Mark

(See Formatting section)

0 Address Marks will be generated in the synch #1 fields if MFM bit = 1, or AME will be generated if MFM bit = 0.

1 Address Mark Enable will be generated in ID preamble if MFM bit = 0.

HSS: Hard or Soft Sectored

(See Hard Sector vs. Soft Sector Operation).

0 Sets DDC for soft sectored operation.

1 Sets DDC for hard sectored operation.

FTF: FIFO Table Format

0 Formatting is done without the use of DMA.

1 The local DMA channel loads the correct number of header bytes (HBC register) per sector into the FIFO from local memory. This data is then substituted for the header bytes during a format operation.

IH1, 2: Internal Header Appendage

IH2 IH1

0 0 No CRC/ECC is internally appended, but external ECC must be attached.

0 1 16-bit CRC CCITT polynomial is appended.

1 0 32-bit programmable ECC code is appended.

1 1 48-bit programmable ECC code is appended.

External ECC may be used with any internal CRC/ECC selection. 1 to 31 bytes of external ECC may be added.

ID1, 2: Internal Data Appendage

ID2, ID1

0 0 No CRC/ECC internally appended.

0 1 16-bit CRC CCITT polynomial is appended.

1 0 32-bit programmable ECC code is appended.

1 1 48-bit programmable ECC code is appended.

External ECC can be appended to any of the four cases dependent upon the Data External ECC Byte Count register.

STATUS (S) Hex Address (00) Read Only

The RESET pin and the RES bit in the OC register reset all of the bits in this register.

ED	CCA	LCB	RCB	LRQ	HMC	NDC	HF
7	6	5	4	3	2	1	0

HF: Header Fault

This bit is valid after a Compare Header or Read Header operation.

SET CRC/ECC error detected in a header field.

RESET This bit is reset when the DDC begins the next disk operation after a new disk command has been issued.

All ID fields entering the DDC during the operation are checked. The HF bit will be set if an error is detected in any header field encountered. However, if the header being sought is found and has no CRC/ECC error, the HF bit is reset. This bit does not produce an error that will stop operation, assert an interrupt, or set the ED bit in the Status register in a compare header operation, but will in a read header operation.

This bit could provide useful diagnostic information if a Sector Not Found error occurs (see Error Register in this section).

NDC: Next Disk Command

SET DDC will accept a new command into the DC register. The header operation is completing the last sector being operated on.

RESET On receipt of a new disk command.

HMC: Header Match Completed

For each of the following, this bit is set and the interrupt is generated at the start of the header postamble field.

Compare Header Operation:

SET Header field correctly matched with no CRC/ECC error.

RESET At beginning of subsequent header operation.

Read Header Operation:

SET Header field has been read with no CRC/ECC error.

RESET At beginning of subsequent header operation.

Ignore Header or Write Header Operation:

SET Always set at end of header field.

RESET At beginning of subsequent header operation.

LRQ: Local Request

This bit follows the LRQ pin, and allows application of the DDC in a polled mode.

SET LRQ pin is asserted.

RESET LRQ pin is not asserted.

RCB: Remote Command Busy

Non-Tracking Mode:

SET When OC register is loaded with a DMA instruction.

RESET Upon completion of the instruction or upon internal or external reset.

3.0 Internal Registers of the DDC (Continued)

Tracking Mode:

SET When RRQ pin is first asserted in a disk write mode, or when the Drive Command register is loaded in a disk read mode.

RESET Upon completion of the instruction or upon internal or external reset.

LCB: Local Command Busy

SET When command requiring local DMA is loaded.

RESET Upon completion of the last local or remote DMA transfer (in tracking mode) or upon internal or external reset.

CCA: Correction Cycle Active

SET On asserting SCC bit in the OC register.

RESET At the end of the correction cycle, simultaneously with the INT pin, if enabled.

ED: Error Detected

SET On assertion of one or more bits in the Error register.

RESET Upon internal or external reset.

ERROR(E) Hex Address (01) Read Only

Any bit set in this register generates an interrupt (if EI bit in the OC register is set) and stops the current operation. The RESET pin and the RES bit in the OC register reset all of the bits in this register.

LI	CF	FDL	NDS	SO	SNF	DFE	HFASM
7	6	5	4	3	2	1	0

HFASM: Header Failed Although Sector Number Matched

(See HFASM description in ADDITIONAL FEATURES)

SET The header bytes(s) marked with the EHF bit in the corresponding HC register(s) matched correctly, but other header bytes were in error.

RESET Upon internal or external reset.

DFE: Data Field Error

SET On detection of a data field CRC/ECC error in a Read Data or Check Data operation. This bit may be set when another error occurs; especially an error occurring during a Write operation. These errors would be Sector Overrun or FIFO Data Lost.

RESET Upon internal or external reset.

The RED command must be loaded into the DC register if error correction is to be attempted.

SNF: Sector Not Found

SET When header cannot be matched for two consecutive index pulses in any Compare Header operation.

RESET Upon internal or external reset.

SO: Sector Overrun

SET If RGATE is active and FIFO is being written to when a sector or index pulse is received. If WGATE is active, this bit is set when a sector or index pulse is received.

RESET Upon internal or external reset.

An SO error will not occur during a Format operation.

NDS: No Data Synch

SET If a sector or index pulse occurs while the DDC is waiting to byte align on the first data synch field (synch #1 or synch #2), or if the DDC byte aligns to the first synch word of the data field but does not match to subsequent bytes (synch #1 or synch #2).

RESET Upon internal or external reset.

FDL: FIFO Data Lost

SET During a disk read operation if the FIFO overflows, or during a disk write operation if the FIFO is read when it is empty.

RESET Upon internal or external reset.

CF: Correction Failed

SET If correction is attempted (SCC bit set in OC register) and correction failed.

RESET Upon internal or external reset.

LI: Late Interlock

Will only occur if IR bit in OC register is set.

SET Controlling logic has failed to write to the Interlock (HBC) register before the end of the data field of the present sector.

RESET Upon internal or external reset.

SECTOR COUNTER (SC)

Allowable Value 0–255 **Hex Address (12) Read/Write**

In a multi-sector operation, the SC register is first loaded with the starting sector number. It is incremented after each header operation is completed. The contents of the SC register will replce any header Byte if the SSC bit is set in the corresponding HC register.

NUMBER OF SECTOR OPERATIONS COUNTER (NSO)

Allowable Value 0–255 **Hex Address (13) Read/Write**

In a multisector operation, the NSO register is loaded with the number of sectors to be operated on. It is decremented after every header operation. When zero, the command is finished. This counter must be reloaded after a reset of the DDC.

HEADER BYTE COUNT (HBC)/INTERLOCK

Allowable Value 2–6 **Hex Address (0F) Read/Write**

This register loads the DMA with the number of header bytes to expect in a Read Header, or a Format operation where FIFO table formatting is used. This register is also used in interlock mode to signal completion of update. The upper five bits of this register are pulled low when read.

HEADER DIAGNOSTIC READBACK (HDR)

Hex Address (36) Read Only

If a Compare Header/Check Data operation is performed and an HFASM error occurs, the header bytes for that sector will have been loaded into the FIFO. By consecutively reading this address, the header bytes are read from the FIFO to the microprocessor. Data will be valid for only the number of header bytes specified in the parameter RAM. (NOTE: This is a dual function register, sharing operation with the Local Transfer register, see DMA REGISTER.)

SECTOR BYTE COUNT REGISTER (L, H)

Allowable Value 1–64k **Hex Address (38, 39) Read/Write**

The two bytes (most and least significant) that comprise this register are loaded during initialization, and define the data

3.0 Internal Registers of the DDC (Continued)

field size for each sector. The number of bytes transferred with local DMA is always equal to what has been loaded into this register. Loading *both* with zero is not allowed.

3.2 DMA REGISTERS

LOCAL TRANSFER (LT) Hex Address (36) Write Only

This is a dual function register, sharing operation with the Header Diagnostic Readback (HDR) register (see COMMAND REGISTERS). IMPORTANT NOTE: If any internal DMA is being used, or if the Remote Data Byte Count registers will be read by the processor, the LT (and RT) register must be loaded before the Sector Byte Count and Remote Data Byte Count register pairs.

LBL2	LBL1	LTEB	LA	LSRW	RBO	LWDT	SLD
7	6	5	4	3	2	1	0

SLD: Select Local DMA Mode

0 *SLAVE MODE*: External DMA must be used in place of on-chip DMA.

1 *NON-TRACKING MODE*: Local DMA is enabled. Whenever local transfers are needed, the DDC becomes the bus master.

TRACKING MODE: Local and remote DMA are enabled. DMA transfers are interleaved (see DMA in DATA TRANSFER section).

LWDT: Local Word Data Transfer

0 Address increments by 1, 8 bit wide transfers.

1 Address increments by 2, 16 bit wide transfers. Address, A0, remains unchanged as it was set by the DMA address.

RBO: Reverse Byte Order

Valid if LWDT bit is set.

0 First byte to/from FIFO is mapped onto the AD0–7 bus.

1 First byte to/from FIFO is mapped onto AD8–15 bus (e.g. 68000).

LSRW: Local Slow Read And Write

0 DMA cycles are four clock periods.

1 DMA cycles are five clock periods. RD and WR strobes are widened by one clock period.

LA: Long Address

Valid only if SLD = 1, and SRD = 0 in Remote Transfer register.

0 16 address bits are issued and strobed by the ADS0 pin. ADS1/RRQ is available for use by the remote DMA.

1 32 address bits are issued, the lower 16 are strobed by ADS0 pin. The most significant 16 address lines are only issued when a rollover from the least significant 16 address lines occurs, or after loading the upper half of the 32-bit address. When the upper 16 address lines are issued, that DMA cycle is five clock cycles long if no internal or external wait states are used.

LTEB: Local Transfer Exact Burst

0 When DMA tranfer is needed, the FIFO will be filled when writing to disk or emptied when reading from disk.

1 When DMA tranfer is needed, the FIFO will receive (when writing) or deliver (when reading) an exact burst of data.

LBL1, 2: Local Burst Length

LBL2	LBL1	
0	0	1 word (2 byte)
0	1	4 word (8 byte)
1	0	8 word (16 byte)
1	1	12 word (24 byte)

When reading from disk, these bits select the number of bytes needed in the FIFO in order to generate an LRQ signal. When writing, these bits select the number of bytes that need to be removed from a full FIFO in order to generate an LRQ. In either case, if the LTEB bit is set, this bit pair indicate how many data transfers will be allowed before LRQ is removed.

Note: Please refer to Section 17, Helpful Hints #29.

REMOTE TRANSFER (RT) Hex Address (37) Write Only

This is a dual function register, sharing operation with the DMA Sector Counter (DSC) (see DSC at the end of this section). If any internal DMA is being used, or if Remote Data Byte Count registers will be read by the processor, the RT (and LT) register must be loaded before the Sector Byte Count and Remote Data Byte Count register pairs.

RBL2	RBL1	RTEB	TM	RSRW	EEW	RWDT	SRD
7	6	5	4	3	2	1	0

SRD: Select Remote DMA

0 Remote DMA inhibited, ADS1/RRQ pin is configured as ADS1.

1 Remote DMA enabled. This is necessary but not sufficient to start remote transfer.

RWDT: Remote Word Data Transfer

0 Remote address increments by 1.

1 Remote address increments by 2. Address A0 remains unchanged as it was set by the starting DMA address.

EEW: Enable External Wait

0 No external wait states acknowledged. Functions 2 and 3 of EXT STAT pin are enabled (see PIN DESCRIPTIONS).

1 The EXT STAT pin will lengthen RD and WR strobes during DMA transfers as long as it is maintained at a high level.

RSRW: Remote Slow READ/WRITE

0 Remote DMA cycles are four clock periods long.

1 Remote DMA cycles are five clock periods long, if external wait states are not asserted.

TM: Tracking Mode

See Tracking Mode description in DATA TRANSFER Section.

0 DMA channels are independent and addresses are allowed to overlap.

1 DMA channel addresses are not allowed to overlap.

RTEB: Remote Transfer Exact Burst

0 If a remote transfer has been initiated, the RRQ pin will remain asserted until the number of bytes specified by the Remote Data Byte Count registers has been transferred, or until the oper-

3.0 Internal Registers of the DDC (Continued)

ation is reset or SRI and SRO bits in the OC register are both set when in non-tracking mode, or when DMA sector counter reaches zero when in tracking mode.

1 If a remote transfer has been initiated, the RRQ pin will remain asserted until the exact number of bytes specified by RBL1 and RBL2 has been transferred, or if any of the conditions described in the previous paragraph occur.

RBL1, 2: Remote Burst Length

LBL2	LBL1	
0	0	1 word (2 byte)
0	1	4 word (8 byte)
1	0	8 word (16 byte)
1	1	12 word (24 byte)

REMOTE DATA BYTE COUNT (L, H)

Allowable Value 0–64k Hex Address (1A, 1B) READ/WRITE

This pair of registers specifies the number of bytes in one remote transfer using the 16-bit address of the remote DMA channel. In the non-tracking mode, the remote DMA can transfer 1–64k bytes independent of the local DMA. Loading both registers with zero will be interpreted as a 64k byte count. These registers are ignored in tracking mode.

DMA ADDRESS BYTE 0–3

Allowable Value 0–255 Hex Address (1C–1F) READ/WRITE

These address bytes are configured dependent on the current DMA mode. In *32-bit mode*, all four bytes form the physical address with 1F containing the most significant byte. In *16-bit mode*, bytes 0 and 1 form the low and high bytes of the local DMA channel, and bytes 2 and 3 form the low and high of the remote DMA channel, if enabled.

DMA SECTOR COUNTER (DSC)

Hex Address (37) Read Only

This counter is only valid during tracking mode and holds the difference between the number of sectors transferred by the local and remote DMA channels. In tracking mode, when DSC = 0, remote transfer is disabled in a disk read operation so invalid data is not exchanged between local and host memory. This is a dual function register, sharing operation with the Remote Transfer (RT) register described earlier in this section.

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. In each pair, one register is loaded with an appropriate 8-bit pattern that will be written to the disk during a Format or Write command, or will be used during a Read or Compare command for byte alignment or a comparison in locating a sector. Refer to *Figure 4*, below, for a listing of the format registers, and the manner in which they are paired. The FORMAT, READ AND WRITE Section contains a listing and description of each of the format fields.

The other register in the pair is used to control the use of the corresponding pattern register. These Byte Count registers are loaded with a 5-bit binary number indicating the number of times the associated pattern will be repeated, therefore defining the size of that particular field (0–31 bytes). The Gap Byte Count register is the only one with 8 bits, allowing a field of up to 255 bytes in length.

The External ECC Count registers do not perform any pattern repetition. The external ECC appendage is provided from outside the DDC, and must be fit into the field whose length is defined by these registers (0–31 bytes). If any field is to be excluded from the disk format, the Byte Count register associated with that field must be loaded with zero. This is particularly important with the External ECC Byte Count registers. If these are non-zero, the EXT STAT pin will expect a pulse for each external ECC field during a Read operation. If these pulses are not supplied, the operation will be aborted in an error condition. Also, no more than two consecutive format fields may be deleted at one time.

The Header Byte Control registers also do not perform any pattern repetition, nor do they define field size. They are provided for controlling the function of each corresponding header byte.

HEADER CONTROL (HC0–5)

Hex Address (24–29) Read/Write

There is one HC register for each of six Header Byte pattern registers.

NU	NCP	EHF	SSC	HBA
4	3	2	1	0

HBA: Header Byte Active

0 The corresponding Header Byte is not included in the header byte field and will not be used in the ID operation. All other bits in each HC register in which this bit is set to zero must also be set to zero. A minimum of two Header Bytes must be enabled out of six, with no more than two disabled consecutively.

1 The corresponding Header Byte contains valid data and will be used in the ID operation.

SSC: Substitute Sector Counter

0 The corresponding Header Byte as stored in the pattern register is directly written to the disk for a Write Header command, and will be compared for Compare Header command.

1 The contents of the Sector Counter (SC) are substituted for this Header Byte during a Write Header command and compared during a Compare Header command. This is normally used in multisector operations.

EHF: Enable HFASM Function

See HFASM function description in ADDITIONAL FEATURES.

0 HFASM function is disabled.

1 HFASM function is enabled. The corresponding Header Byte is designated as that byte that must match in order to generate an HFASM error, typically the sector number.

NCP: Not Compare

0 The corresponding Header Byte will be compared normally.

1 A valid comparison will always be assumed, regardless of the true outcome.

NU: Not Used

This bit must be set to zero. If set to 1 unspecified operations may occur.

3.0 Internal Registers of the DDC (Continued)

Pattern Register	Hex Addr	Pattern Source	Control Function	Hex Addr	Control Register
ID Preamble	31	*Internal*	*Repeat 0–31x*	21	ID Preamble Byte Count
ID Synch #1 (AM)	32			22	ID Synch #1 (AM) Byte Count
ID Synch #2	33			23	ID Synch #2 Byte Count
Header Byte 0	14		*Define/Control*	24	Header Byte 0 Control
Header Byte 1	15			25	Header Byte 1 Control
Header Byte 2	16			26	Header Byte 2 Control
Header Byte 3	17			27	Header Byte 3 Control
Header Byte 4	18			28	Header Byte 4 Control
Header Byte 5	19			29	Header Byte 5 Control
ID External ECC	*	*External*	*0–31 Bytes*	2B	ID External ECC Byte Count
ID Postamble	3C	*Internal*	*Repeat 0–31x*	2C	ID Postamble Byte Count
Data Preamble	3D			2D	Data Preamble Byte Count
Data Synch #1 (AM)	3E			2E	Data Synch #1 (AM) Byte Count
Data Synch #2	3F			2F	Data Synch #2 Byte Count
Data Format	3B		*Field Size 1–64k Bytes*	38	Sector Byte Count L
				39	Sector Byte Count H
Data External ECC	*	*External*	*0–31 Bytes*	2A	Data External ECC Byte Count
Data Postamble	30	*Internal*	*Repeat 0–31x*	20	Data Postamble Byte Count
Gap	3A		*Repeat 0–255x*	34	Gap Byte Count

*These are not pattern registers.

FIGURE 4. Format Registers

3.4 CRC/ECC REGISTERS

The following registers are for programming and controlling the CRC/ECC functions of the DDC. Many of these registers have dual functions, depending on whether they are being written to or read from. Take care in noting which these are, to avoid confusion later. Only a basic functional description of these are provided here. Detailed instructions on their use can be found in the CRC/ECC section.

ECC SR OUT 0–5 Hex Address (02–07) Read Only

The syndrome bytes for performing a correction are available from these registers, and are externally XOR'ed with the errored data bytes. These are dual function registers, sharing operation with the Polynomial Preset Bytes.

POLYNOMIAL PRESET BYTES 0–5 (PPB0–5)

Hex Address (02–07) Write Only

The ECC shift registers can be preset by loading a bit pattern into these registers. These are dual function registers, sharing operation with the ECC SR Out registers.

POLYNOMIAL TAP BYTES (PTB0–5)

Hex Address (08–0D) Write Only

These registers are used for programming the taps for the internal 32 or 48-bit ECC polynomial. PTB0 and PTB1 are dual function registers, sharing operation with the Data Byte Counters.

DATA BYTE COUNTER 0, 1 (LS, MS)

Hex Address (08, 09) Read Only

The Data Byte Counters indicate the location of the byte in error after an ECC cycle. These are dual function registers, sharing operation with the Polynomial Tap Bytes 0 & 1. The Sector Byte Count Register must be reloaded with the sector length plus the number of ECC bytes before the start of a correction cycle. If the CF bit in the Error register is reset after a correction, the Data Byte Counter will contain an offset pointing to the first byte in error.

ECC/CRC Control (EC)

Hex Address (OE) Write Only

DNE	IDI	IEO	HNE	CS3	CS2	CS1	CS0
7	6	5	4	3	2	1	0

CS0–CS3: Correction Span Selection Bits

These four bits program the number of bits that the ECC circuit will attempt to correct. Errors longer than the correction span will be treated as non-correctable. The allowable correction span is 3–15 bits. If a span outside this range is loaded, the DDC will automatically default to a span of three bits.

For example, a five bit correction span would load as:

CS3	CS2	CS1	CS0
0	1	0	1

HNE: Header Non-Encapsulation

0 Header address mark and/or synch fields are encapsulated in the CRC/ECC calculation.

1 Header address mark and/or synch fields are not encapsulated in the CRC/ECC calculation.

NOTE: The SAM bit in the DF register must be reset when performing a Compare or Read Header operation, and the HNE bit is active low. If this is not done, the CRC/ECC calculation will begin at the synch word of the header, resulting in a Header Fault that will abort a Read operation or a Sector Not Found error for a Compare Header operation.

IEO: Invert ECC Out

See note under IDI bit, below.

0 Checkbits exiting ECC/CRC shift register are unaltered.

1 Checkbits exiting ECC/CRC shift register are inverted.

3.0 Internal Registers of the DDC (Continued)

IDI: Invert Data In

0 Data and checkbits entering the ECC/CRC shift register are unaltered.

1 Data and checkbits entering the ECC/CRC shift register are inverted.

NOTE: This inversion option has been included for compatability with a few systems that require ECC input and/or output inversion.

DNE: Data Non-Encapsulation

0 Data address mark and/or synch fields are encapsulated in the CRC/ECC calculation.

1 Data address mark and/or synch fields are not encapsulated in the CRC/ECC calculation.

4.0 DDC Operation

4.1 MICROPROCESSOR ACCESS

The DDC requires microprocessor control to initiate operations and commands, and to check chip status. All registers in the DDC appear as unique memory or I/O locations. Each can be randomly accessed and operated on. When the DMA is not performing a memory transfer, the chip can be accessed as a memory location or standard I/O port. Only eight bits of data may be transferred at this time, using pins AD0–7 (the upper 8 bits of a 16 bit microprocessor are not used). Six dedicated address pins (RS0–5) individually select all of the DDC's internal registers. By using these dedicated lines with an address strobe input (ADS0), the chip can be used in both multiplexed and demultiplexed address bus environments. The ADS0 and RS0–5 pins operate as a fall through type latch. By asserting CS active low, the DDC recognizes it has to be a slave and allows RD and WR to effect the internal registers. With multiplexed address and data lines, a positive strobe pulse on ADS0 will latch the address. The ADS0 line may be derived from a microprocessor address strobe such as ALE. In systems with a dedicated address bus (demultiplexed), ADS0 may be pulled high to allow address information to flow through the latch. Finally, by applying CS and a RD or WR strobe, any of the 64 internal locations can be accessed. It is important to note that most registers are read or write only. Some registers, however, change function dependent on whether they are being read from or written to (see Dual Function register list in INTERNAL REGISTERS).

4.2 OPERATING MODES

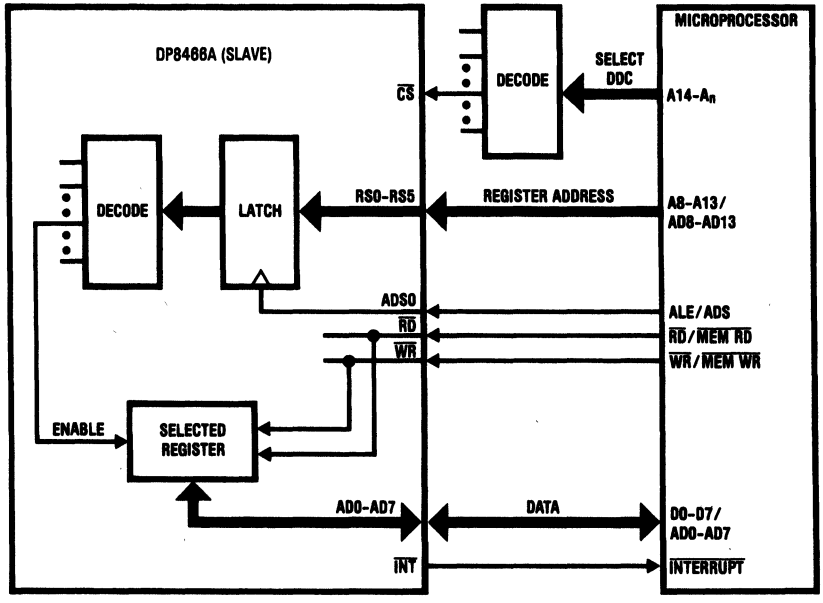
The DDC can be thought of as operating in four modes: *RESET, COMMAND ACCEPT, COMMAND PERFORM* and *ERROR*. These modes are given here in order to provide a functional operating description of the DDC, particularly when an error has been encountered.

Mode 1 *RESET*: All functions are stopped, and no command can be issued. During power up and before initalization, the DDC is held in this mode. To leave this mode, pin 24 (RESET) must be high, a 0 must be written to the RES location in the OC register, and a RED command loaded into the DC register. This places the DDC into MODE 2.

Mode 2 *COMMAND ACCEPT*: The DDC is free and ready to receive the next command (NDC bit set in Status register). Upon receipt of a command, the DDC will enter MODE 3.

Mode 3 *COMMAND PERFORM*: The directed operation is performed. If no error is encountered, the DDC will return to MODE 2. An error will put the DDC into MODE 4.

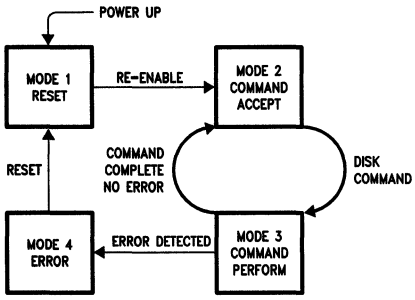
Mode 4 *ERROR*: The error needs to be serviced, and then the DDC can be reset by MODE 1.



TL/F/5282–4

FIGURE 5. Microprocessor Access to DP8466A

4.0 DDC Operation (Continued)

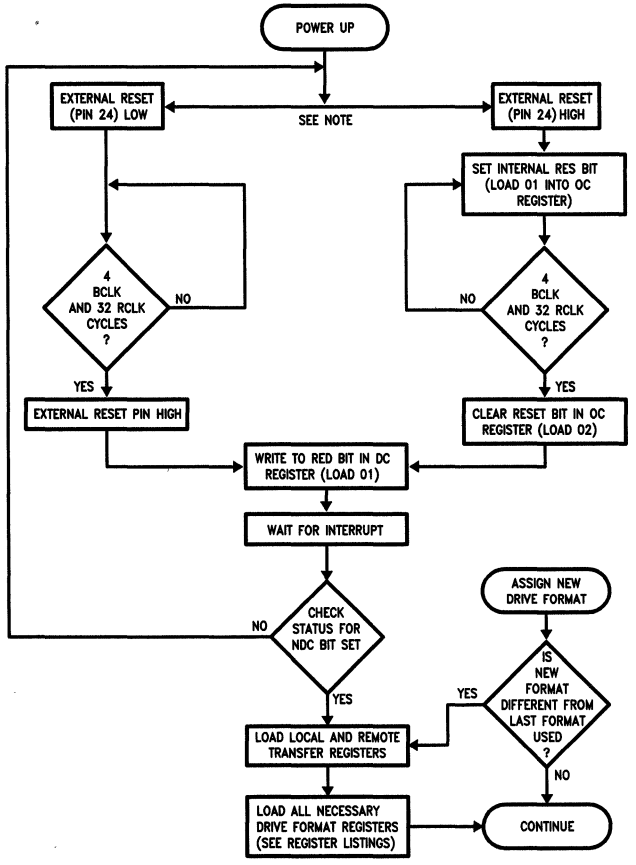


TL/F/5282–5

FIGURE 6. DDC Operating Modes

4.3 POWER UP AND INITIALIZATION

In powering up the DDC, the counters and registers must be initialized before a drive can be assigned and the appropriate information loaded. This can be done by either holding pin 24 (RESET) low, or by setting the internal RES bit in the OC register. Both require that the DDC be held in the reset condition for a minimum of 32 RCLK periods and 4 BCLK periods before the reset condition can be cleared. *Figure 7* shows a general algorithm for both methods. After power up, and whenever a new drive is assigned, the appropriate drive format registers need to be loaded before any drive operation is performed.



TL/F/5282–6

Note 1: If the RE-ENABLE operation is accomplished by polling the status register and not enabling interrupt, then it should be polled for NDC bit set. When set, it should remain set for at least 30 RCLKs before RE-ENABLE can be considered complete. The REN operation under worst case condition could take as long as 270 RCLKs.

Note 2: As shown various methods are possible for power up, and it is up to the user as to which is more suitable. The DDC should be reset and RCLK and BCLK should be applied after power up, otherwise it may draw an excessive amount of current, and may cause bus contention.

FIGURE 7. Power Up and Initialization Algorithm

5.0 Format, Read and Write

5.1 DISK FORMATTING

The formatting process is carried out through the format parameter and pattern registers (see FORMAT REGISTERS). These registers should be loaded during the initialization process for the particular drive in use. The pattern registers are loaded with the specific 8-bit pattern to be written to the disk. The count registers specify the number of times each 8-bit pattern is to be written. In loading these registers, several things need to be kept in mind:

- If any byte count register is loaded with zero, that field will be excluded, and no pattern for the corresponding pattern register need be loaded.
- At least two header bytes must be used, with no more than two consecutive unused header bytes. This also applies to all the fields in the format, where no more than two consecutive fields may be deleted. The one exception is the internal header ECC and external header ECC field. At least one of these fields must be present.
- If the disk is hard sectored, no gap byte count needs to be loaded. See Hard Vs. Soft Sector Operation in the FORMAT, READ AND WRITE Section.

The sector format options that are provided with the DDC are shown in *Figure 8*. The fields common to the ID and data fields, such as the preamble, Synch, CRC/ECC and postamble fields, perform similar functions, and are briefly discussed below.

PREAMBLE: Allows the PLL in the data separator to achieve phase lock.

SYNCH #1 and 2: Synch #1 contains the missing clock address mark for use with soft sectored disks. Generally, this field is not used in hard sectored disks. The synch #1 field can be used to extend the preamble or the synch fields in hard sectored mode. Synch #1 and #2 fields allow for byte alignment of the DDC.

HEADER BYTES: Used to uniquely identify each sector. Examples are sector number, cylinder number, track number, etc.

DATA: Information to be stored.

CRC/ECC: This field is generated and checked internally.

EXT.ECC: Used with external ECC circuitry. Provides space for externally generated ECC bytes.

POSTAMBLE: Allows read gate turn off time for the PLL to unlock. Provides a pad so that the write splice does not occur at the end of the CRC.

GAP 3: Provides protection against speed variation. In soft sectored mode, its length is determined by the Gap Byte Count register. In hard sectored mode, this gap will continue until the next sector pulse.

Format operations always start with an index pulse, and end with the next index pulse, thus making one track. The DDC has three approaches for formating disks:

Internal Sequential
FIFO Table
Interlock Type

INTERNAL SEQUENTIAL

This mode is used where the sector number is incremented for each physically adjacent sector, that is, for an interleave of one. This mode may be used on a multi-sector operation to format a whole track of sequential sectors. The header bytes other than the sector number, such as cylinder number and head number, are loaded. The Sector Counter (SC) is loaded with the first sector number desired on the track and the HC register with SSC = 1. The Number of Sector Operations (NSO) counter is loaded with the number of sectors per track. Finally, the FMT bit is set in the DC register in addition to bits for a Write Header/Write Data, multi-sector operation. Formatting begins upon loading the DC register. The last sector number written will therefore be [SC] + [NSO] − 1.

FIFO TABLE

This approach is ideal for sector interleaving and offers the minimum of microprocessor intervention during the format operation. The microprocessor sets up the header bytes of each sector, contiguously in memory. The local DMA channel or external DMA is used to transfer the header byte sets into the FIFO. Each set transferred is used once for each header field. The local DMA transfers a new set for each sector. The number of sectors transferred is determined by the NSO register.

The format operation follows the sequence below:

(1) Before the format operation, a full track of header byte sets is loaded into a memory area accessible to the local DMA channel. Each header byte set must contain an even number of bytes. If it contains an odd number of bytes, an extra "dummy" byte must be inserted so that each header byte set will be contained in an even byte boundary.

(2) The DMA address is loaded with the location of the first byte of the first header byte set.

ID FIELD

ID PREAMBLE	ID SYNCH #1 (AM)	ID SYNCH #2	HEADER BYTES	ID CRC/ECC*	ID EXT ECC*	ID POSTAMBLE
0–31 Bytes	*0–31 Bytes*	*0–31 Bytes*	*2–6 Bytes*	*0, 2, 4 or 6 Bytes*	*0–31 Bytes*	*0–31 Bytes*

DATA FIELD

DATA PREAMBLE	DATA SYNCH #1 (AM)	DATA SYNCH #2	DATA FORMAT PATTERN	DATA CRC/ECC	DATA EXT ECC	DATA POSTAMBLE	GAP 3
0–31 Bytes	*0–31 Bytes*	*0–31 Bytes*	*1–64k Bytes*	*0, 2, 4 or 6 Bytes*	*0–31 Bytes*	*0–31 Bytes*	*0–255 Bytes*

Note 1: The ID CRC/ECC field and the ID EXT ECC field must not be set to zero simultaneously.

Note 2: The ID and DATA preamble fields need to be at least 3 bytes for proper operation.

FIGURE 8. Sector Format Fields

5.0 Format, Read & Write (Continued)

(3) The Header Byte Count (HBC) is loaded with the number of header bytes in each sector (2–6 bytes).

(4) The Disk Format (DF) register is loaded with the FTF bit set.

(5) The Drive Command (DC) register is loaded for a Write Header/Write Data, multi-sector, format operation.

INTERLOCK TYPE

This approach offers the most versatility, but requires fast microprocessor intervention. It may be used to format a whole track of interleaved sectors. It can also be used for creating files of varying sector length, but this can be very tricky. The DDC can format sectors with data lengths from 1 to 64k bytes with single byte resolution.

Interlock type formatting uses the interlock mode and the header complete interrupt to enable the microprocessor to directly update any format parameter bytes. The Operation Command (OC) register is loaded wth IR (Interlock Mode), EHI and EI bits set. The Disk Format (DF) register should be loaded with the FTF bit reset. The header byte pattern for each selected header byte must be loaded into the relevant register. The NSO register is loaded with the number of sectors to be formatted. The DC register is then loaded for a Write Header/Write Data, multi-sector, format operation.

After the header field is written in the first sector, the DDC issues the header complete interrupt. With interlock mode set, the controlling microprocessor has the block of time until the preamble field of the next sector to read status, load the next sector's header bytes into the DDC registers and confirm this had been accomplished by writing to the Interlock (HBC) register. This must be done after the HMC interrupt for every sector, including the last sector of the operation. If this is not done, a Late Interlock error will occur when a subsequent command is loaded in the DC register.

In a non-format operation, the user has only until the end of the data field to write to the HBC register (see Data Recovery Using The Interlock Feature in ADDITIONAL FEATURES). This operation is repeated until the NSO register decrements to zero. An interrupt will then be issued indicating that the operation has completed.

5.2 READ AND WRITE

For initiating Read/Write operations, the necessary format registers need to be loaded with the appropriate information to enable the DDC to identify the desired sector. Multi-sector operations will also require the Number of Sector Operations (NSO) counter and the Sector Counter (SC). Algorithms outlining the read/write operations are shown in *Figures 10* and *11*. For each of these, it is assumed that the parameters for the desired sector(s) have been loaded, and that the head is positioned over the proper track.

READ

During a read operation, header data passing under the disk head is compared to the header bytes in the DDC parameter RAM. If a match is found after a read command is issued, the data field of the identified sector will start filling the FIFO. Once the selected threshold data level (burst length) is reached, the Local DMA Request (LRQ) pin will be asserted, signaling that a transfer is required. When the LACK pin grants the bus, either the exact burst length or the entire FIFO contents are transferred to memory. The FIFO continues filling, and this process repeats until the entire data field has been transferred to memory.

WRITE

A similar process occurs in reverse for a write operation. The DMA fills the FIFO, and when the correct sector is found, this data begins to be written to disk. When the data in the FIFO falls by an amount equal to the burst length, a transfer request is issued on LRQ. When LACK is granted, the DMA either fills the FIFO or transfers the exact number of bytes specified in the burst length. This process continues until a number of bytes specified by the Sector Byte Count register has been written to the disk.

Multi-sector operations follow the same procedure, but the operation is repeated on the number of sectors specified in the Number of Sector Operations (NSO) counter, with an interrupt being generated on completion of the last sector.

5.3 HARD SECTOR vs. SOFT SECTOR OPERATION

The choice between hard and soft sectored operation is made through the use of the HSS bit in the Drive Format register. This bit, in conjunction with other control bits can set the DDC to perform a number of functions depending on whether a read, write or format operation is to be enacted. HSS = 0 sets the DDC for soft sectored operation, and HSS = 1 sets the DDC for hard sectored operation.

FORMAT

In hard sectored operation, the DDC assumes that sector pulses are present, and will ignore the gap count. Gap bytes will be written until a pulse is detected on the SECTOR pin.

In soft sectored operation, the gap count will be used for every sector except the last. The Gap Byte Count register determines the Gap 3 length. For the last sector, gap bytes will be written until an index pulse is received.

READ

When reading, the need for the AMF input pulse is determined by the HSS bit. For soft sectoring, the AMF input is required for at least one bit time within the Synch #1 fields in both the ID and Data sections of the sector. For hard sectoring, the AMF input is not required.

The HSS bit in the DF register, and the SAIS command in the DC register define when RGATE is asserted for various sector formats. This is outlined below.

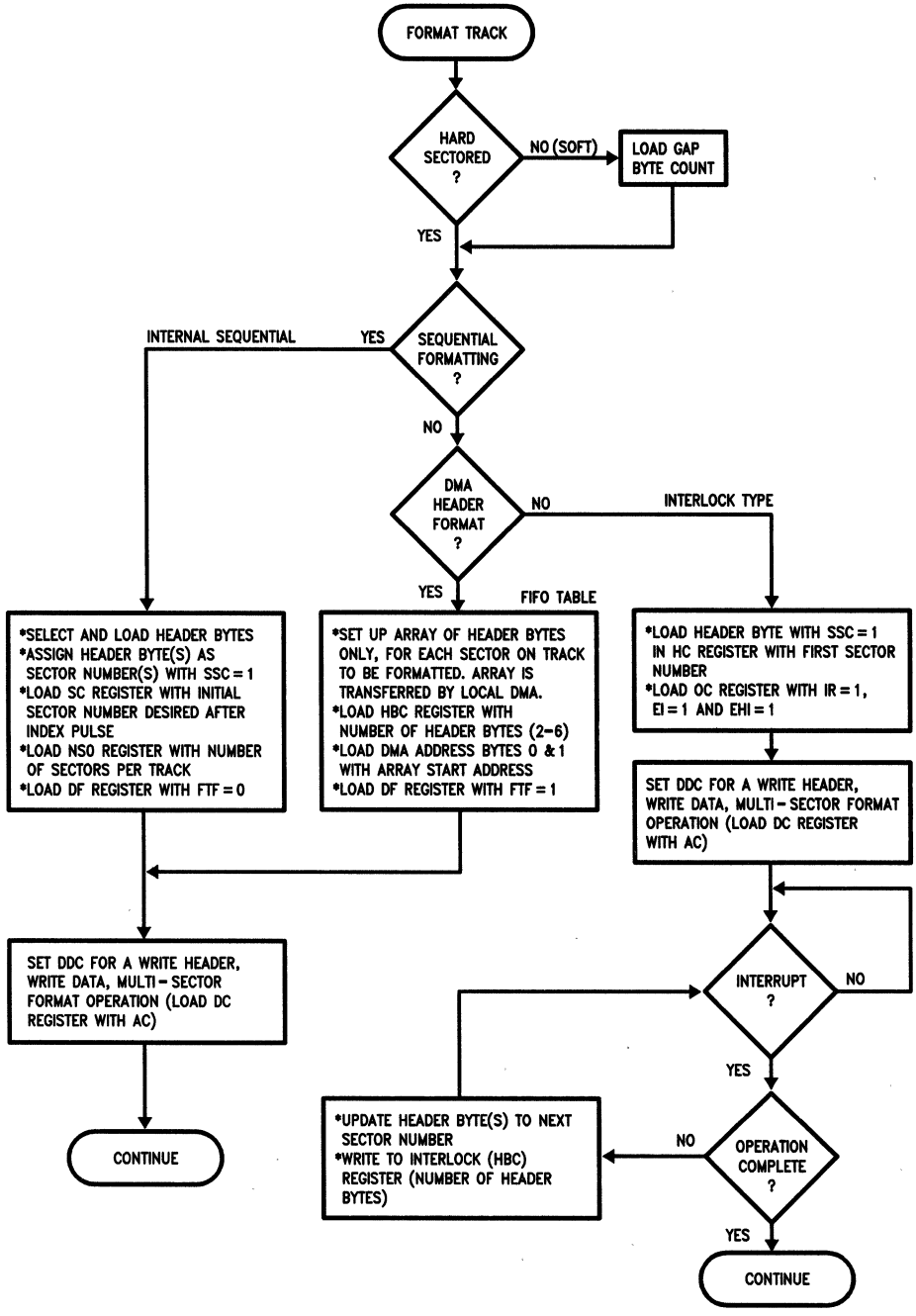
HSS	SAIS	RGATE ASSERTED:
0	0	On index pulse
0	1	On receipt of instruction
1	0	On index pulse
1	1	On index or sector pulse

WRITE

The HSS, MFM and SAM bits in the DF register determine the use of the address mark and the AME pin as follows:

HSS	MFM	SAM	FUNCTION
0	0	0	AME pin activated during ID and data synch #1 fields.
X	0	1	AME pin activated during ID preamble.
0	1	X	Missing clocks inserted in ID and data synch #1 fields. AME pin indicates LPRE (if enabled).
1	0	0	AME pin disabled.
1	1	X	Synch #1 fields written without missing clock pulse.

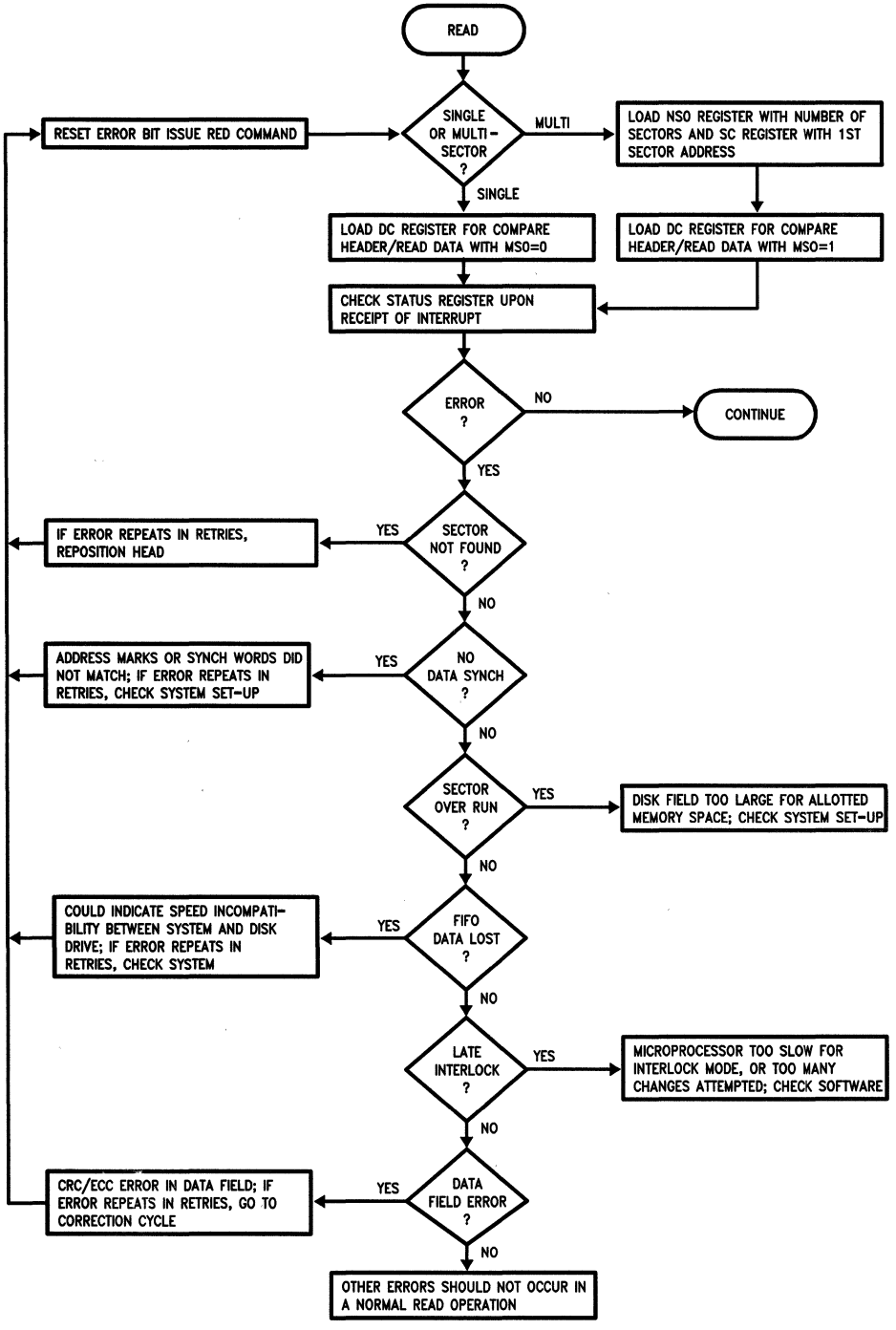
5.0 Format, Read & Write (Continued)



TL/F/5282-7

FIGURE 9. Format Track Algorithm

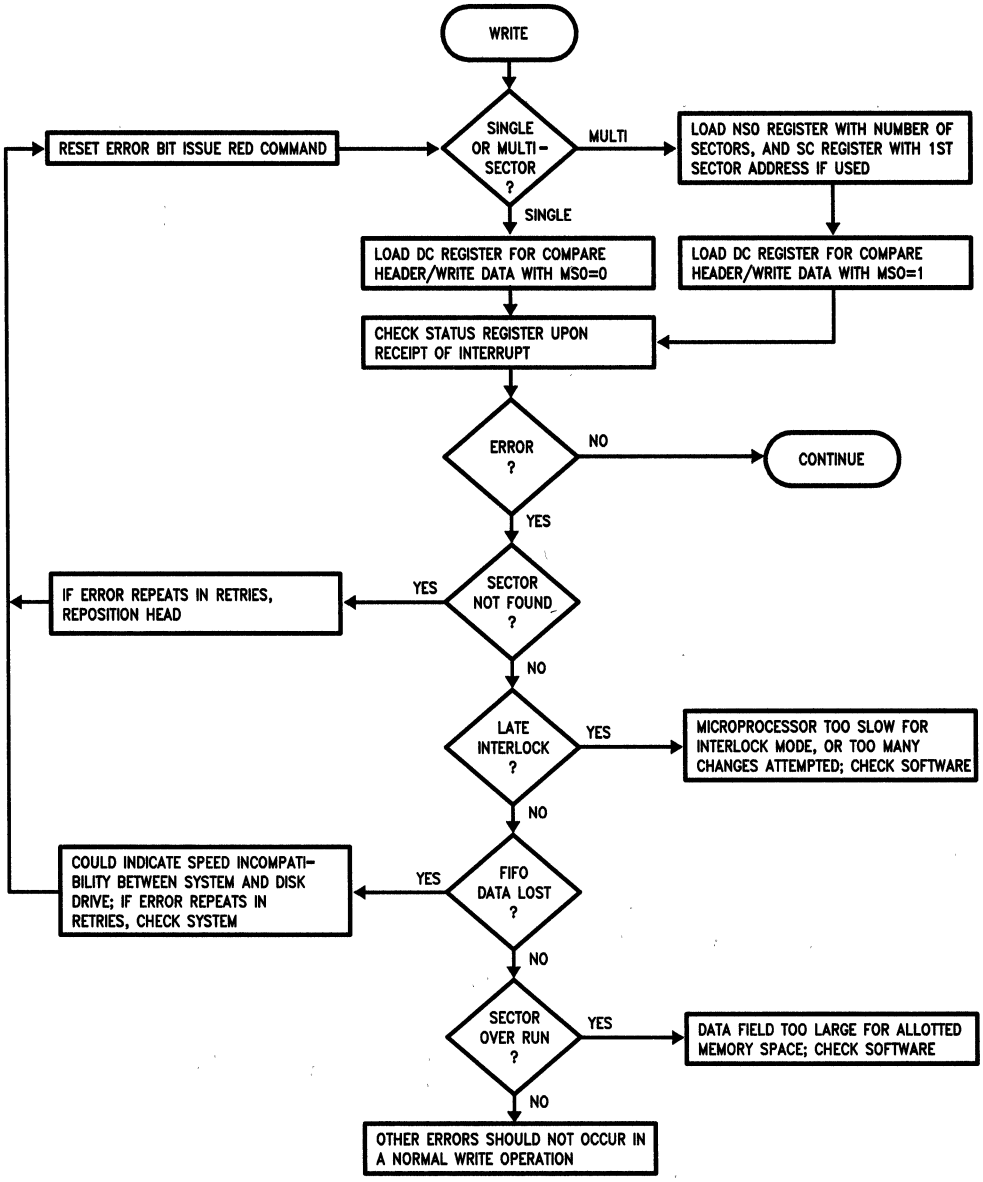
5.0 Format, Read & Write (Continued)



TL/F/5282–8

FIGURE 10. Simple Read Operation

5.0 Format, Read & Write (Continued)



TL/F/5282–9

FIGURE 11. Simple Write Operation

5.0 Format, Read & Write (Continued)

5.4 MFM ENCODED DATA

MFM encoding of write data is controlled by the MFM bit in the DF register MFM = 1 sets the DDC to write MFM data to the disk. MFM = 0 sets the DDC to write NRZ data to the disk.

PRECOMPENSATION OF MFM ENCODED DATA

When the MFM bit in the DF register and the EP bit in the OC register are set, precompensation will be indicated on the EPRE and LPRE pins. Precompensation is issued for the middle bit of a 5-bit field. In the DP8466A, early and late precompensation will be enacted for all of the combinations as shown below. All other patterns will not require precompensation. Precompensation can be disabled by setting the EP bit in the OC register inactive low.

EPRE NRZ PATTERNS	LPRE NRZ PATTERNS
00 0 10	00 1 10
00 0 11	00 1 11
01 1 00	10 0 00
01 1 01	10 0 01
11 1 00	10 1 10
11 1 01	10 1 11

Precompensation outputs are aligned to provide symmetrical set-up and hold times relative to the rising edge of the WDATA outputs. This gives a half period of RCLK set-up time on precompensation outputs. This is shown in *Figure 12*. Two bits of zero precede the preamble fields at the leading edge of the write gate when writing MFM data due to MFM encoded delays.

5.5 ADDRESS MARK PATTERNS, MISSING CLOCK

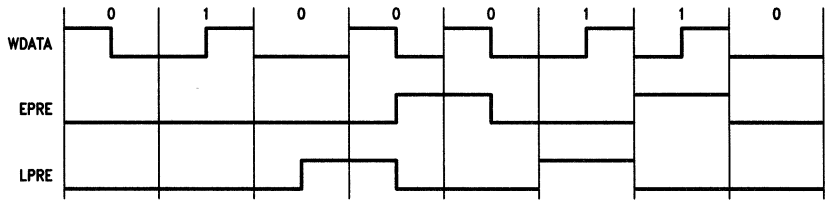
During writing and formatting a sector with MFM encoding enabled, a clock violation, or missing clock pulse, will be inserted in the synch #1 field. This indicates the address mark. For an example of this, refer to *Figure 13*.

When writing MFM encoded data with precompensation enabled, only the following hex values are allowed to be loaded into the synch #1 pattern registers:

A1, C2, C3, E1, 84, 85, 86, 87

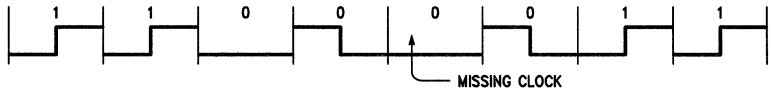
With no precompensation, any pattern containing 100001 is valid.

During a soft sectored read operation, an AMF pulse will be expected on the AMF/EPRE pin during each byte of the synch #1 field.



TL/F/5282–10

FIGURE 12. Example of EPRE and LRPE Outputs



TL/F/5282–11

FIGURE 13. Missing Clock Example

6.0 CRC/ECC

6.1 PROGRAMMING CRC

The DDC is set for internal CRC by programming the disk Format (DF) and ECC/CRC Control (EC) registers. The CRC-CCITT polynomial used by the DDC for the CRC code is given below:

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

The DDC uses the pattern preset to all 1's for the CRC calculation. *Note:* If no CRC/ECC is used for the ID fields, an external ECC must be used.

6.2 PROGRAMMING ECC

There are two sets of six registers used to program the ECC. One set of six is used to program the polynomial taps, while the other set is used to establish a preset pattern (typically all 1's). Bits contained in the ECC Control (EC) register are used to control the correction span. The DF register contains bits for choosing the desired type of appendage: Either 32 or 48-bit programmable ECC polynomials, or the 16-bit CCITT CRC polynomial is possible. A 48-bit computer generated polynomial is also available from National Semiconductor free of charge.

PROGRAMMING POLYNOMIAL TAPS

To program a polynomial into the shift register, each tap position used in the code must be set to 0, and all unused taps should be set to 1. The bit assignment for these registers in 48 and 32-bit modes is shown in the tables that follow. It is important that for 32-bit codes, PTB2 and PTB3 all be set to 1's. Failure to do so will result in improper operation. Also, x^{48} and x^{32} are implied, i.e., a 32-bit ECC will always contain the x^{32} term and a 48-bit ECC will always contain the x^{48} term. For both ECC's, the term x^0 (or 1) is also implied, even though this bit is accessible.

Tap Assignment 48-Bit Mode

REG#	ADDR	*BIT NUMBER*							
		DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
PTB0	*08*	x^7	x^6	x^5	x^4	x^3	x^2	x^1	x^0
PTB1	*09*	x^{15}	x^{14}	x^{13}	x^{12}	x^{11}	x^{10}	x^9	x^8
PTB2	*0A*	x^{23}	x^{22}	x^{21}	x^{20}	x^{19}	x^{18}	x^{17}	x^{16}
PTB3	*0B*	x^{31}	x^{30}	x^{29}	x^{28}	x^{27}	x^{26}	x^{25}	x^{24}
PTB4	*0C*	x^{39}	x^{38}	x^{37}	x^{36}	x^{35}	x^{34}	x^{33}	x^{32}
PTB5	*0D*	x^{47}	x^{46}	x^{45}	x^{44}	x^{43}	x^{42}	x^{41}	x^{40}

Tap Assignment 32-Bit Mode

REG#	ADDR	*BIT NUMBER*							
		DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
PTB0	*08*	x^7	x^6	x^5	x^4	x^3	x^2	x^1	x^0
PTB1	*09*	x^{15}	x^{14}	x^{13}	x^{12}	x^{11}	x^{10}	x^9	x^8
PTB2	*0A*	1	1	1	1	1	1	1	1
PTB3	*0B*	1	1	1	1	1	1	1	1
PTB4	*0C*	x^{23}	x^{22}	x^{21}	x^{20}	x^{19}	x^{18}	x^{17}	x^{16}
PTB5	*0D*	x^{31}	x^{30}	x^{29}	x^{28}	x^{27}	x^{26}	x^{25}	x^{24}

PROGRAMMING PRESET PATTERN

To program the preset pattern that the shift registers will be preset to, PPB0–PPB5 must be initialized. As in the polynomial taps, x^{48}, x^{32}, and x^0 are implied. The assignment of the bits for 48 and 32 bit modes is shown in the tables on the following pages.

The value programmed into each register will be the preset pattern for the eight bits of the corresponding shift register. For typical operation, these will be programmed to all 1's. All unused presets must be set to 0. In 32-bit mode, PPB2 and PPB3 must be set to all 0's. Failure to do so will result in improper operation.

Preset Bit Assignment 48-Bit Mode

REG#	ADDR	*BIT NUMBER*							
		DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
PPB0	*02*	x^7	x^6	x^5	x^4	x^3	x^2	x^1	x^0
PPB1	*03*	x^{15}	x^{14}	x^{13}	x^{12}	x^{11}	x^{10}	x^9	x^8
PPB2	*04*	x^{23}	x^{22}	x^{21}	x^{20}	x^{19}	x^{18}	x^{17}	x^{16}
PPB3	*05*	x^{31}	x^{30}	x^{29}	x^{28}	x^{27}	x^{26}	x^{25}	x^{24}
PPB4	*06*	x^{39}	x^{38}	x^{37}	x^{36}	x^{35}	x^{34}	x^{33}	x^{32}
PPB5	*07*	x^{47}	x^{46}	x^{45}	x^{44}	x^{43}	x^{42}	x^{41}	x^{40}

Preset Bit Assignment 32-Bit Mode

REG#	ADDR	*BIT NUMBER*							
		DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
PPB0	*02*	x^7	x^6	x^5	x^4	x^3	x^2	x^1	x^0
PPB1	*03*	x^{15}	x^{14}	x^{13}	x^{12}	x^{11}	x^{10}	x^9	x^8
PPB2	*04*	0	0	0	0	0	0	0	0
PPB3	*05*	0	0	0	0	0	0	0	0
PPB4	*06*	x^{23}	x^{22}	x^{21}	x^{20}	x^{19}	x^{18}	x^{17}	x^{16}
PPB5	*07*	x^{31}	x^{30}	x^{29}	x^{28}	x^{27}	x^{26}	x^{25}	x^{24}

RECOMMENDED POLYNOMIAL AS AN EXAMPLE

To program the 32-bit polynomial of the form:

$$x^{32} + x^{28} + x^{26} + x^{19} + x^{17} + x^{10} + x^6 + x^2 + 1$$

with a preset of all 1's, a correction span of 5-bits with no header/data encapsulation, the following registers would be programmed as shown. Note that PTB2 and PTB3 must be all 1's and PPB2 and PPB3 must be all 0's in 32-bit mode.

6.0 CRC/ECC (Continued)

Polynomial Taps

REG#	BIT NUMBER 7	6	5	4	3	2	1	0
PTB0	1	0	1	1	1	0	1	0
PTB1	1	1	1	1	1	0	1	1
PTB2	1	1	1	1	1	1	1	1
PTB3	1	1	1	1	1	1	1	1
PTB4	1	1	1	1	0	1	0	1
PTB5	1	1	1	0	1	0	1	1

Preset Pattern

REG#	BIT NUMBER 7	6	5	4	3	2	1	0
PTB0	1	1	1	1	1	1	1	1
PTB1	1	1	1	1	1	1	1	1
PTB2	0	0	0	0	0	0	0	0
PTB3	0	0	0	0	0	0	0	0
PTB4	1	1	1	1	1	1	1	1
PTB5	1	1	1	1	1	1	1	1

ECC Control Register

BIT#	7	6	5	4	3	2	1	0
SET	1	0	0	1	0	1	0	1

6.3 OPERATION DURING CORRECTION

The DDC can be set to correct an error any time one has been detected and before another operation has begun. The user decides when to initiate the correction. The sector in question can be re-read several times to insure that the error is repeatable. If so, the error can be considered a hard error on the disk and a correction can be attempted. Since the DDC does not contain drive control circuitry, it is the user's responsibility to provide the programming for the execution of any re-read operations and the associated decision making.

The syndrome bytes in the ECC shift register will contain the bit error information. The bytes in error will already have been transferred to memory. Once initiated, the correction is performed internal to the DDC, leaving the bus free for other operations. An interrupt will be issued within the time it takes to read a sector, indicating whether the error was corrected or not. During this time, the erroneous sector in memory will remain unchanged.

Error correction time is determined by the error's location in the sector. The nearer to the start of the sector, the longer the DDC takes to locate the error. This time can be determined using the formula shown at right. It should be noted that this is internal correction time only; more time is required for the microprocessor to perform additional operations.

Before initiating a correction operation, the DDC needs to be reset, and re-enabled (see Operating Modes in DDC OPERATION). The Sector Byte Count registers must be initialized to *[sector length]* + 4 for 32-bit mode or *[sector length]* + 6 for 48-bit mode. The correction command should be issued when the counter has been updated.

The DDC will issue an interrupt after the correction cycle is complete. Other activities (such as completion of remote DMA) may issue interrupts before this happens. These interrupts should be serviced to allow the Correction Cycle Complete interrupt to be issued. The CCA bit in the Status register will be high during the entire correction cycle. It will be reset when the cycle has completed. The ED bit in the Status register will remain active throughout the correction cycle.

If after an interrupt, the Status register is read and the CCA bit is low, the Error register is read to see if the correction was successful. If the CF bit is set, this signifies that the error was non-correctable. This usually means that two errors have occurred with extremities exceeding the selected correction span. Failure to correct an error is serious and the system should be notified that the data from that sector is erroneous.

If the CF bit was not set, the error was corrected. The microprocessor then computes the address of the first byte in the data field that contains the error. That address is: *[current value of DMA Address Bytes 0 & 1] - [Sector Length] + [Data Byte Count L & H]* − 1.

Errors are corrected by XOR'ing syndrome bytes (ECC SR Out 0–5) with the bytes in the data record in memory that contain the error. The Data Byte Count can be used to determine whether the error is in the ECC or data field. If the Data Byte Count is greater than the maximum sector length, the error is in the ECC field and no correction should be attempted. If the Data Byte Count is less than the sector length, the error is in the data field (or it may straddle the data and ECC fields) and may be corrected.

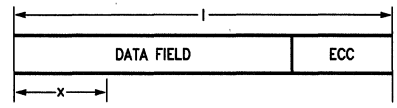
For performing a correction with 32-bit ECC, the following shift registers should be read sequentially to obtain the syndrome byte pattern:

ECC SR Out 1, ECC SR Out 4, ECC SR Out 5

ECC SR Out 2 and 3 are not used in 32-bit mode and will contain 0's if read. ECC SR Out 0 will contain all 0's if the error is correctable, and may contain some set bits if it is not.

ECC SR Out 1 will always contain the first bits in error. The succeeding bits will be contained in ECC SR Out 4 and 5. If the maximum span of 15 bits is used, all three registers may be needed, depending on where the first bit occurs.

To correct the error, the syndrome bits in these registers are XOR'ed with the data bits contained in buffer memory. The corrected data is then written back to the buffer memory, replacing the data in error. The address of the first byte in error is computed by the microprocessor as described above.



TL/F/5282–12

Approximate Correction Time = $(I - x)/f$

I = Entire length of data field and ECC appendage (in bits)

x = Distance from least significant bit to first error location (in bits)

f = read clock frequency (in hertz)

FIGURE 14. Calculating Correction Time

6.0 CRC/ECC (Continued)

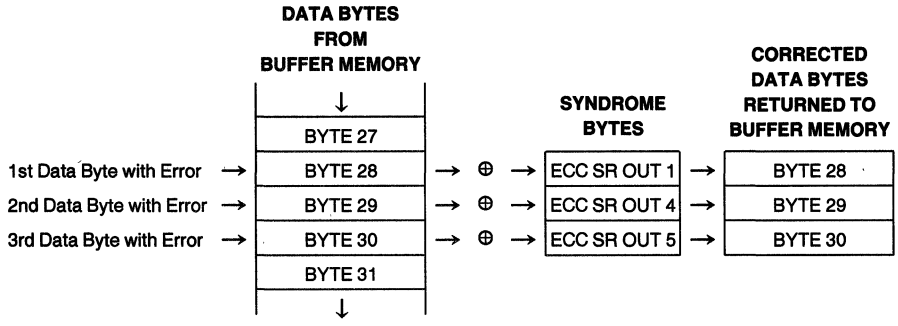


FIGURE 15. 32-Bit ECC Correction Process

To perform a 48-bit ECC correction, the following registers should be read sequentially:

ECC SR Out 1, ECC SR Out 2, ECC SR Out 3

ECC SR Out 0, 4 and 5 are not used for outputting syndrome bits for correction in 48-bit mode and will contain 0's for a correctable error. If the error is non-correctable, these registers may contain some set bits. Syndrome bit location and error correction is performed as in 32-bit mode.

EXAMPLE OF A 32-BIT CORRECTION

Shown in *Figure 17*, is a record with several bits read in error from disk. Bits D4, D11, D13 and D14, now located in memory, were incorrectly and need to be corrected. As can be seen, the correction pattern provided in ECC SR Out 1 and 2 can be used to correct bits D4, D11, D13 and D14. The CPU reads the Data Byte Count and computes that it points to the first byte read from disk. This byte is XOR'ed with ECC SR Out 1 and is written back to memory. The second byte read from the disk is XOR'ed with ECC SR Out 4 and then written back. ECC SR Out 5 need not be used since it contains all 0's.

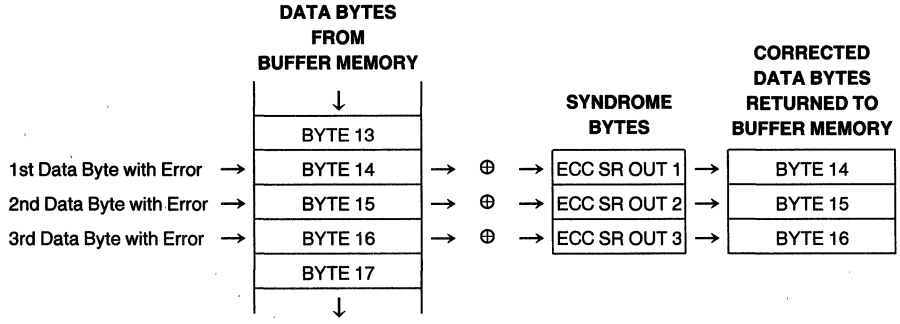


FIGURE 16. 48-Bit ECC Correction Process

Syndrome Pattern

REGISTER	BIT NUMBER 7	6	5	4	3	2	1	0
ECC SR OUT 1	0	0	0	1	0	0	0	0
ECC SR OUT 4	0	1	1	0	1	0	0	0
ECC SR OUT 5	0	0	0	0	0	0	0	0

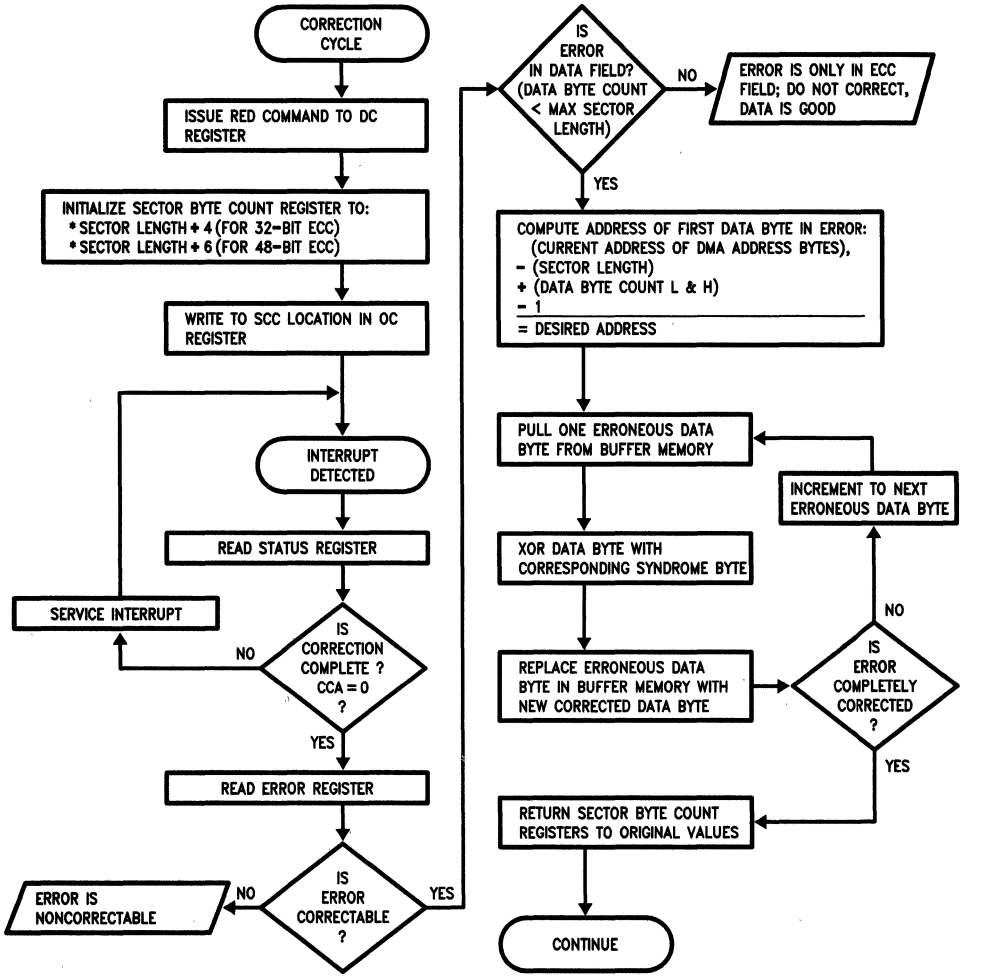
Buffer Memory

CORRESPONDING BUFFER DATA BIT PATTERN							
D7	D6	D5	*	D3	D2	D1	D0
D15	*	*	D12	*	D10	D9	D8
D23	D22	D21	D20	D19	D18	D17	D16

* = location of bits in error

FIGURE 17. Example of a 32-Bit Correction

6.0 CRC/ECC (Continued)



TL/F/5282–13

FIGURE 18. Correction Cycle Algorithm

THIS CYCLE CAN ONLY BE INITIATED AFTER A READ DATA OPERATION HAS BEEN COMPLETED

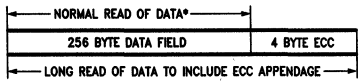
3

6.0 CRC/ECC (Continued)

A note of caution: If the DDC is in the tracking DMA mode when a data error occurs, the remote DMA channel will transfer the sector in error to its destination in the system. The DDC will still interrupt to indicate that it has detected an error. It is then up to the system to get the DDC to correct the error in buffer memory and retransfer the corrected data to the system.

6.4 ECC CHECK USING LONG READ AND LONG WRITE

During a normal read or write operation, the size of the data field is specified by the Sector Byte Count register pair. If the data field is extended during a readback, the ECC appendage can be read in as data and analyzed outside the DDC. This is what is known as a *long read*.



TL/F/5282–14

*Read length defined by Sector Byte Count register pair.

FIGURE 19. Example of a Long Read

Likewise, an externally generated ECC appendage can be added to the data and written to the disk as data with or without the onboard CRC/ECC generator enabled. This is known as a *long write*.

By using long reads and long writes in conjunction with external software used to produce data fields and external CRC/ECC appendages, various diagnostic programs can be devised to test the DDC's internal correction functions and ECC generation circuitry. These tests could be incorporated in the initialization algorithm to test the chip each time it is powered up.

7.0 Data Transfer

7.1 DIRECT MEMORY ACCESS (DMA)

The DDC is designed to work efficiently in two major system configurations:

(1) A single system bus with shared data buffer/system memory (see *Figure 20*).

(2) A dual bus environment with a local microprocessor, buffer memory and DP8466A on a local bus interfacing the host system bus through an I/O port (see *Figure 21*).

All DMA activity is supported by the following three features:

PROGRAMMABLE BURST LENGTH (THRESHOLD)

Here, the transfer of data between the 32-byte FIFO on the DDC and the external memory (local or main) involves the use of internal or external local DMA channel. While writing to the disk, the DDC will initiate a transfer when the FIFO has been depleted by the burst length. It will also initiate a transfer while reading from the disk when the FIFO fills to the burst length. This length is selectable from 2, 8, 16 or 24 bytes, allowing for the variations in bus latency time encountered in most systems.

At the start of a write operation, the FIFO will be filled up in a series of bursts of the programmed length.

If the exact burst option is not selected, the FIFO will be completely filled (if writing to disk) or emptied (if reading from disk) in one DMA operation. The burst length is always the threshold at which the transfer will be requested and is independent of the DMA mode, including slave.

At the end of a sector or an operation, the local burst counter does not reset. This means that the first burst of a subsequent sector will not be what was programmed in the LTR if the burst length was not an exact multiple of the data length. The data length is equal to the sector length times the number of sectors. The DDC would have to be reset between operations if resetting the local burst counter is desired. It is not recommended to count bursts in order to monitor the amount of data transferred.

8-BIT/16-BIT WIDE TRANSFERS

Byte or word wide data transfer can be selected for both local and remote DMA channels. Word wide tranfers with local DMA use the AD0–15 pins, and byte wide use the AD0–7 pins. Both the local and the remote DMA addresses are incremented by 2 for word wide transfers, and 1 for byte wide transfers. Commands and DDC parameter registers are loaded and read only 8-bits at a time, using AD0–7.

REVERSE BYTE ORDER

This option is only valid for 16-bit wide transfers using the local DMA channel. This should not be used for 8 bit wide transfers. It enables the two bytes being transferred to be mapped with the high order byte to AD0–7 and the low order byte to AD8–15, or vice-versa.

The DDC has provisions to accommodate five DMA modes. These are as follows:

EXTERNAL DMA:	1. Slave Mode
INTERNAL DMA, Single Bus:	2. 16-Bit Local Mode
	3. 32-Bit Local Mode
Multiple Bus:	4. Non-Tracking Mode
	5. Tracking Mode

All five modes accommodate the three configurations just described. All DMA modes, except external slave, use an incrementing address. Local channel transfers always have priority over remote channel transfers unless externally reprioritized. If the local channel is used, its transfer length is always automatically loaded from the Sector Byte Count register pair.

7.2 EXTERNAL DMA

SLAVE MODE

In this mode, no on-chip DMA control is used. LRQ and LACK pins are connected to an external DMA controller. After LACK has been granted, I/O RD and I/O WR from the DMA controller are used to strobe data between the internal FIFO and the DDC I/O port. 8-bit and 16-bit wide data transfers are possible. Throughout this data sheet, reference has been made to the use of on-chip DMA for the transfer of data. It is important to note here that external DMA can be used in place of this if so desired.

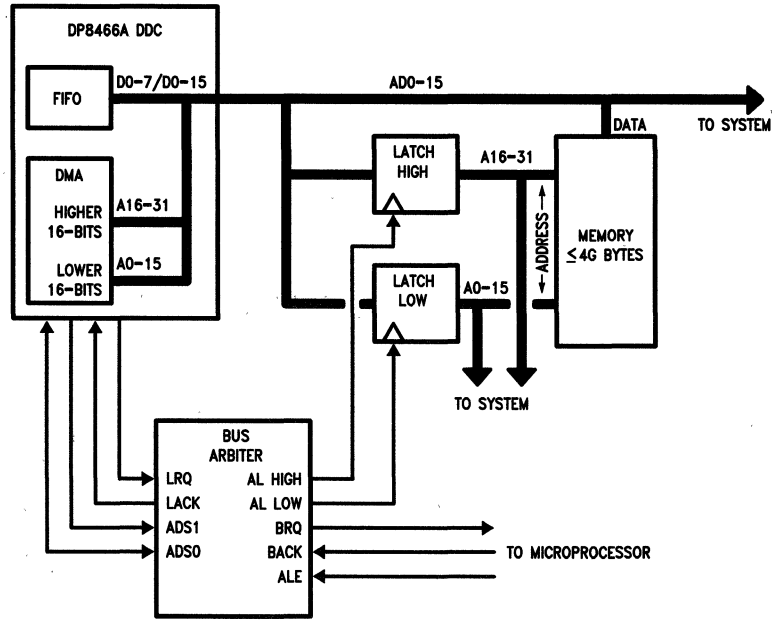
7.3 INTERNAL DMA

The following four modes all use on-chip DMA control with at least the local channel serving as bus master for data transfers between the internal FIFO and memory.

SINGLE BUS SYSTEMS

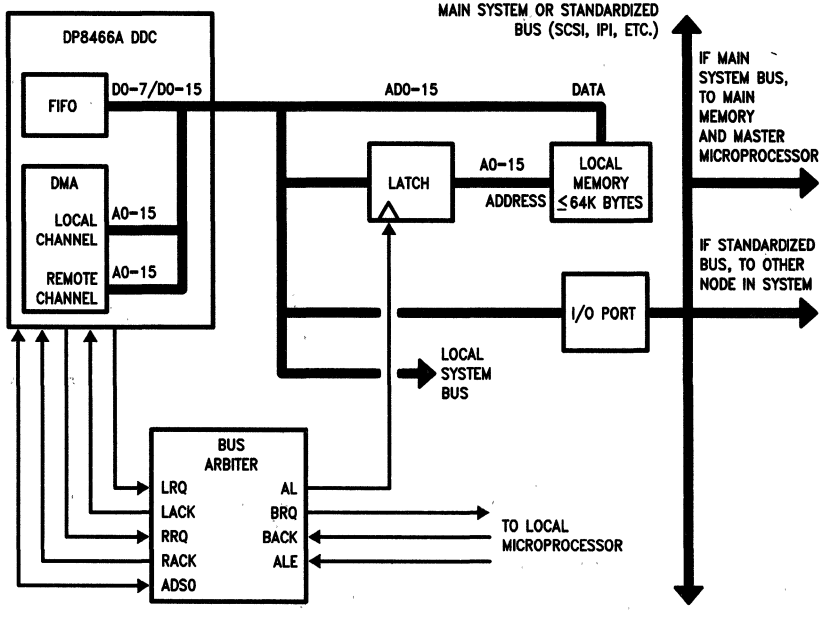
The following two modes support a single bus and a single shared buffer/system memory. Bus access should be guaranteed before the FIFO overflows or empties during a disk transfer operation. A FIFO Data Lost error (FDL bit in Error register) will be flagged and the operation aborted if this fails

7.0 Data Transfer (Continued)



TL/F/5282-16

FIGURE 20. Single System Bus, 32-Bit Address DMA



TL/F/5282-15

FIGURE 21. Dual System Bus, 16-Bit Address DMA

7.0 Data Transfer (Continued)

to happen. Different system latency times can be accommodated by the selectable burst length.

16-BIT LOCAL MODE

SLD bit is set and LA bit is reset in the LT register. Only the 16-bit local DMA channel is enabled. 64k bytes are directly addressable by the DDC. Address data is presented on AD0–15 and latched with ADS0. Transfers always take 4 BCLK cycles if no wait states are issued.

32-BIT LOCAL MODE

SLD bit and LA bit are both set in the LT register. SRD bit in the RT register must be reset. The local DMA channel is now set to issue 32-bit addresses using the remote DMA channel as the upper 16-bit address register. 4 G bytes are addressable by the DDC. During the first DMA cycle of a newly programmed address, or after a roll-over of the lower 16-bit address counter occurs, ADS1 strobes a new high order word (A16-31) into the external address latches. Each time this happens, the DMA cycle is 5 BCLK periods long. When a new high order address is not needed, the DMA cycle is 4 BCLK periods long. ADS0 is used as an output to latch the low order word (A0–15) from the AD0–15 pins into the address latch.

MULTIPLE BUS SYSTEMS

The following two modes support a dual bus environment, where a local microprocessor, buffer memory and the DP8466A interface to the host through an I/O port. The difference between tracking and non-tracking mode is whether the DDC or the controlling microprocessor ensures that an attempt to read data from buffer memory does not occur before data has been written there. Basic algorithms for both are shown in *Figures 22* and *23*.

TRACKING MODE

SLD bit set and LA bit reset in the LT register. SRD bit and TM bit set in the RT register. The DDC ensures that data is not overwritten by data transferred from the FIFO.

This mode effectively turns the buffer memory into a large FIFO. This is accomplished through the use of the DMA Sector Counter (DSC), which keeps track of the difference between sectors read/written to the disk and the sectors transferred to/from the host system. Each time the source transfers a sector of data into buffer memory (length determined by the Sector Byte Count register pair), the DSC register is incremented. It is decremented each time the destination has transferred a sector of data. Whenever the DSC register contents become zero, destination transfers are inhibited. This mode facilitates multi-sector operations.

Example: Tracking Mode, Disk Read

- Source is local DMA
- Destination is remote DMA
- DSC register is reset automatically upon start of operation
- Local and remote start address, SC, NSO, OC and finally DC registers are loaded. Other registers may need to be updated, but this is a minimum set.

A sector is read from the disk and is transferred in bursts from the FIFO to the buffer memory by local DMA. The DSC register then increments and the remote channel can begin transferring the first sector from the buffer memory to the host system. Burst transfers can be interleaved with local DMA, remote DMA and microprocessor all sharing the bus. The local channel bursts have priority over remote bursts. If the remote channel manages to transfer a sector before the local channel has completed the next sector, the DSC register will decrement to zero. Further remote transfers are inhibited until the local channel completes another sector and increments the DSC. In other words, each time a local sector has been transferred, the DSC is incremented and each time a remote sector completes, the DSC is decremented. Therefore, the DDC prevents further buffer memory contents that have not been previously loaded with valid data by the local DMA from being transferred to the host system. The remote channel continues operation until the last byte from the buffer memory has been transferred. An interrupt is issued upon completion of the operation.

NON-TRACKING MODE

SLD bit set and LA bit reset in the LT register. SRD bit set and TM bit reset in the RT register. The remote and local channel addresses are completely independent. The controlling microprocessor must insure that the data to be transferred by the remote channel is not over-written by the local channel and vice-versa. DMA address and count registers are set up independently. Remote start address (DMA Address Bytes 2 and 3) and Remote Data Byte Count registers must be loaded before SRI or SRO bits are set in the OC register. Local or remote transfers may already be in progress when the other channel is started. The local channel has priority over the remote channel. Local bus utilization is then interleaved between the local channel, the remote channel and the controlling microprocessor.

By setting both SRI and SRO simultaneously, any non-tracking remote DMA operation will stop. The present remote address and remote data byte count will be retained and the local DMA will be unaffected. Loading the original OC instruction (input or output) will restart the original instruction from the last remote DMA address.

DMA Mode Select Table

DMA Mode	LT Register SLD	LT Register LA	RT Register SRD	RT Register TM
SLAVE	0	0	0	0
16-BIT LOCAL	1	0	0	0
32-BIT LOCAL	1	1	0	0
TRACKING	1	0	1	1
NON-TRACKING	1	0	1	0

NOTE: In either tracking or non-tracking mode, if either channel is loaded with an odd byte transfer count, the DDC will transfer the next higher even number of bytes. For example, if 511 was loaded into the Remote Data Byte Count registers, 512 bytes would be transferred, with valid data only in the first 511 bytes.

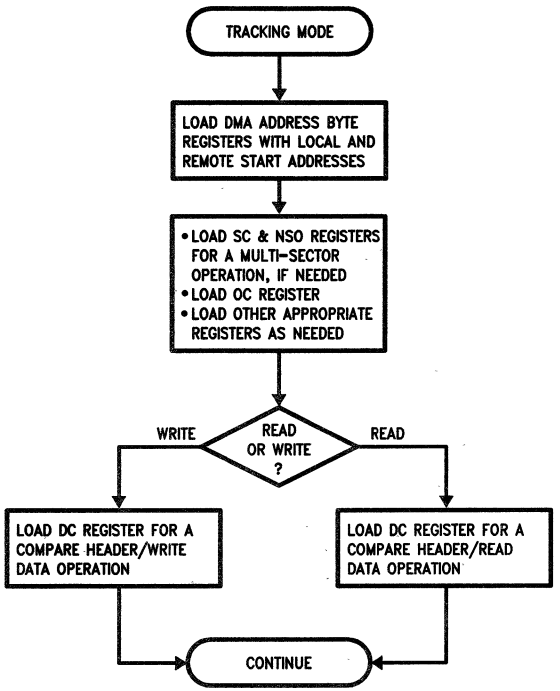
DMA WAIT STATES

INTERNAL

Both DMA channels can independently be set to lengthen the RD and WR strobes by one clock cycle (LSRW bit in the LT register and RSRW bit in the RT register). This lengthens each transfer from 4 cycles to 5 cycles of the BCLK.

EXTERNAL

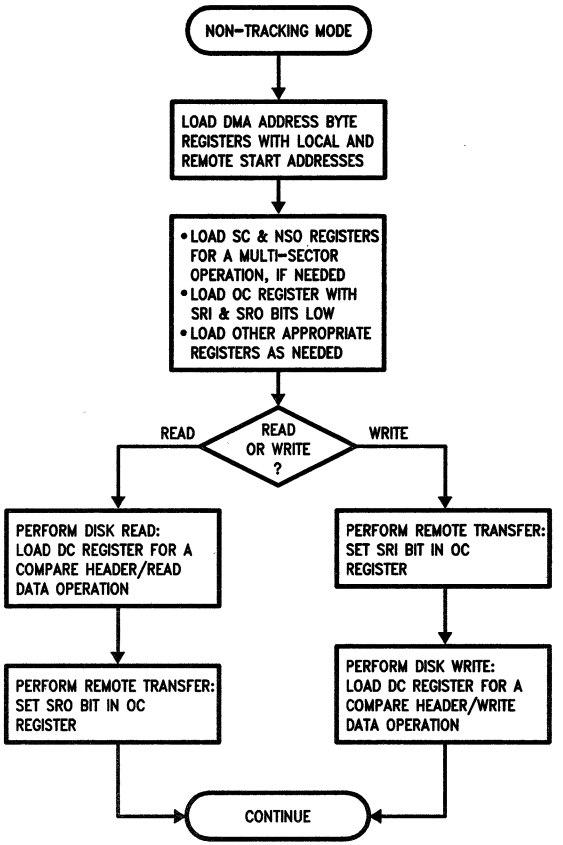
By enabling the external wait states (in the RT register), the EXT STAT pin is configured to insert wait states in each RD and WR pulse as long as this input is high. This is valid for both the local and remote DMA channels.



TL/F/5282–17

NOTE: DMA operation is completely automatic for the duration of the command. For example, when reading disk, local DMA empties/fills the FIFO and remote DMA transfers data at least one sector behind the local channel to an I/O port. For disk write, local channel will be at least one sector behind the remote channel.

FIGURE 22. Tracking Mode for Normal Disk Read/Write



TL/F/5282–18

NOTE: This is the most basic of non-tracking mode operations, and unlimited, more versatile algorithms can be built up from this.

FIGURE 23. Non-Tracking Mode for Normal Disk Read/Write

8.0 Interrupts

Interrupts can only occur if the EI bit in the OC register is set. If it is not set, the INT pin is always de-asserted high. 16 RCLK periods (3.2 μs at 5 Mbit/sec data rate) must pass before servicing an interrupt (i.e. reading Status). Failure to do this will result in servicing the same interrupt twice. There are four general conditions that may cause an interrupt to occur:

Operation Complete

Header Complete

Error

Correction Cycle Complete

OPERATION COMPLETE

This interrupt indicates that the current DDC operation has completed and the DDC is ready to execute a new command. Commands can be loaded sooner by setting EHI bit in the OC register. The Next Disk Command (NDC) bit in the Status register is set coincident with the Header Complete interrupt. New disk commands can be loaded before DMA operation is finished if NDC is set. If the command is a multi-sector operation, the end of operation interrupt will occur only after the operation is completed in the last sector of operation. The INT pin is asserted low when:

- Disk operation is completed for any command that is not a disk read operation.
- A read operation in the tracking DMA mode after the remote transfer is complete.
- A read operation in the non-tracking DMA mode after the local transfer is complete.
- A non-tracking mode remote DMA transfer is completed. This is independent of the disk operation or the local DMA.

HEADER COMPLETE:

If the EHI and EI bits are set in the OC register, an interrupt will occur when any header operation is complete. Multi-sector operations will generate an interrupt after each header in each sector has been operated on. It is asserted two bit times into the ID postamble. This function allows the changing of header bytes (and parameter RAM in general) *on the fly*. The Header Complete interrupt can be used in conjunction with the Interlock Required (IR) bit in the OC register set to insure that changes have been completed before the next sector is encountered (see Interlock Type formatting). Another normal mode of use would be to notify the controlling microprocessor when the next disk command can be loaded. This interrupt is coincident with the Next Disk Command (NDC) bit being set in the Status register.

ERROR

Any bit set in the Error register sets the ED bit in the Status register and causes an interrupt.

CORRECTION CYCLE COMPLETE

An interrupt will occur at the end of an internal correction cycle, regardless of whether the error was corrected or not. If the error was non-correctable, the CF bit will be set in the Error register. This will not generate two interrupts.

CLEARING INTERRUPTS

The INT pin will be forced inactive high any time the Status register is being read. If an interrupt condition arises during a status read, this condition will assert INT as soon as the status read is finished.

Interrupts can also be cleared by setting the internal RES bit, or by asserting the external RESET pin.

9.0 Additional Features

9.1 DATA RECOVERY USING THE INTERLOCK FEATURE

The potential use of the interlock feature is in recovering data from a sector with an unreadable header field. It is assumed that the number of the sector physically preceding the bad sector on the disk is known. A single-sector operation will be performed on these sectors, and the Drive Command register will be changed in between them. The following steps will recover the data:

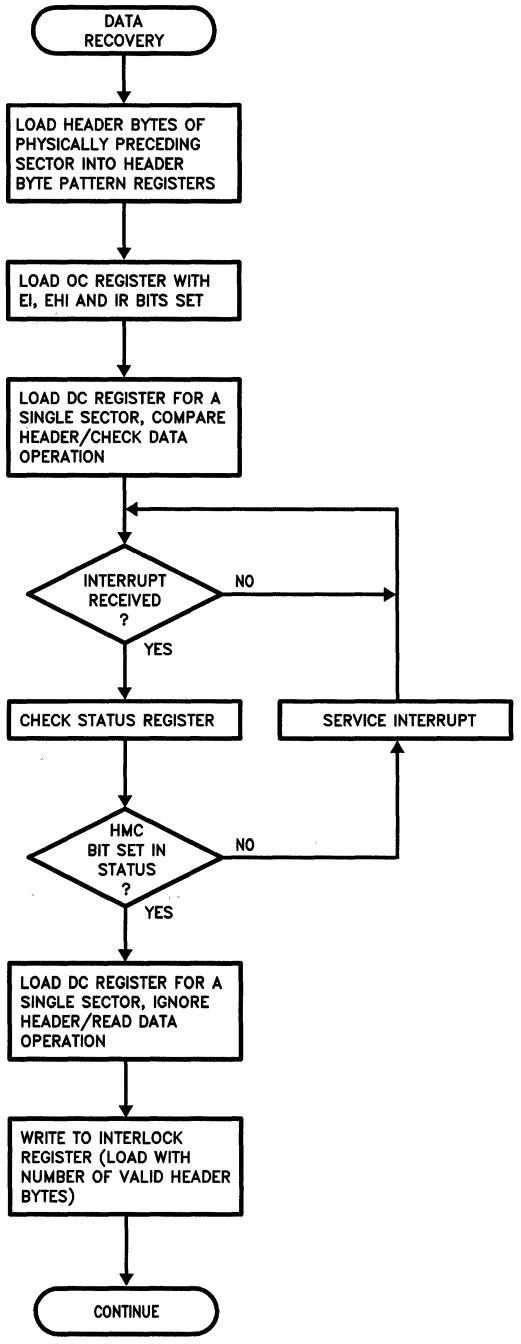
- The header bytes of the physical sector preceding the desired sector are loaded into the relevant byte pattern registers.
- The OC register must be loaded with the EI, EHI and IR bits set. This enables the Header Complete interrupt as well as the interlock feature.
- The DC register is loaded for a single-sector, Compare Header/Check Data operation.
- After the Header Complete interrupt, the DC register must be loaded with an Ignore Header/Read Data operation, and the Interlock (HBC) register written to. If the controlling microprocessor fails to write to the HBC register before the end of the data field of the first sector, a Late Interlock error (LI bit in Error register) will be flagged, and the operation will be terminated with an interrupt.
- When the HMC interrupt occurs on the second sector, the Interlock (HBC) register must be written to again in order to avoid LI error.
- The operation will terminate normally when the data from the badly labeled sector has been read.

9.2 HFASM FUNCTION

The Header Failed Although Sector number Matched (HFASM) function on the DDC can be used to perform maintainance and diagnostic functions, both of which will be briefly outlined here.

The HFASM function is enabled by setting the EHF bit in at least one of the Header Control registers, with a Compare Header command loaded into the DC register. More than one header byte may have its EHF bit set. If any one of the header byte(s) with it's EHF bit set matched, but any other header byte(s) (regardless of the state of their EHF bit) don't match, an HFASM error will occur.

9.0 Additional Features (Continued)



TL/F/5282–19

FIGURE 24. Data Recovery Algorithm

9.0 Additional Features (Continued)

In this way, the HFASM function performs a maintenance type function, and can often indicate that the head is positioned over the wrong track. It is independent of whether or not a CRC failure has occurred. An HFASM failure will not stop operation until the header CRC bytes have been compared and the CRC check is completed.

To perform a diagnostic function, the header can be read and analyzed. This can be done only during a Compare Header/Check Data operation with HFASM enabled. This causes the header patterns coming from the disk to be written into the FIFO. We must assume that the FIFO is empty (or has been reset before the operation) in order for this operation not to interfere with data transfers. If an HFASM error occurs during a Header Compare, the FIFO will be left intact and the header with the error can be read out of the FIFO from the Header Diagnostic Readback (HDR) register. (Note: LWDT of the local transfer register must be set to match the bus width of the accessing MP for this function.)

If an HFASM error did not occur, the FIFO will be cleared and the header patterns that were stored there will be lost.

This process can only be enabled for one disk command. The Compare Header/Check Data command will enable this function. Any other command will disable it.

10.0 Typical System Configurations

10.1 LOW COST SYSTEM

In a single bus system, the DDC can directly address 4G bytes of main memory. The 16-bit I/O port (AD0–15) is externally demultiplexed and buffered with the octal latches and drivers. The main microprocessor, through a separate disk drive control I/O block, is responsible for commands like Head Select, Seek, TRK 000, Drive select, etc. Bus access must be guaranteed before the FIFO overflows or empties. A short burst length (LT and RT registers) accommodates longer bus latency times and helps to insure this. The burst capability allows for other bus operations to be interleaved while the FIFO is filling (during a read) or emptying (during a write). If long, important CPU operations are required, the next configuration must be used.

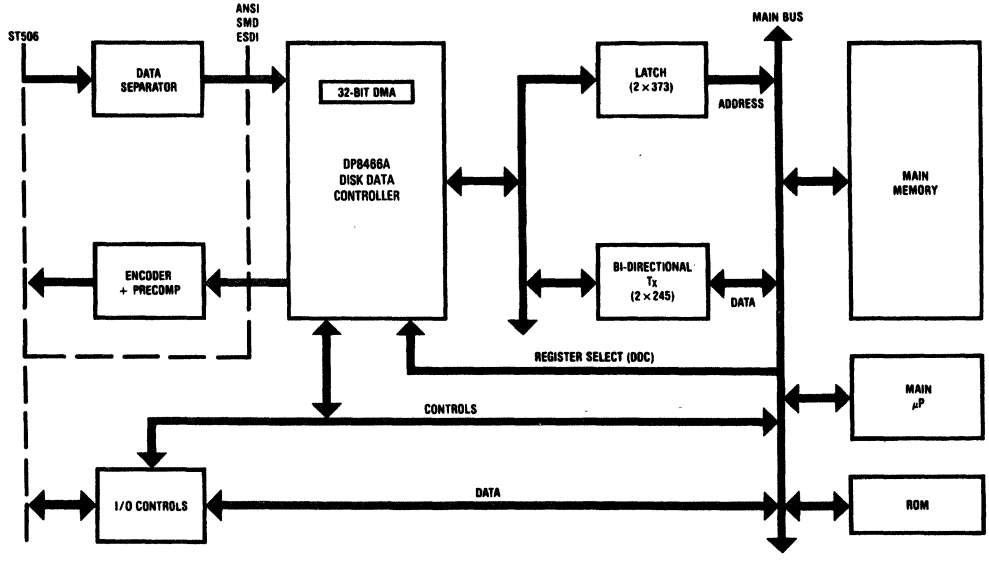


FIGURE 25. Low Cost System Configuration

TL/F/5282–20

10.0 Typical System Configuration (Continued)

10.2 HIGH PERFORMANCE SYSTEM

This configuration provides a local bus for the DDC to share with the local microprocessor and a buffer memory. Here, whole blocks of data can be transferred between the DDC and buffer memory without interfering with the system bus. This leaves the main CPU to perform important operations and to allow data transfers when it is ready. This configuration is also used in intelligent drives or systems that comply to SCSI or IPI specifications. A local bus, dedicated microprocessor and buffer memory are main characteristics of an intelligent disk interface. The buffer memory can be used as a cache for track or file buffering and command lists can be down-loaded for execution by the microprocessor. The two DMA channels can both directly address 64k bytes of buffer memory. The local DMA channel transfers data between the buffer memory and the internal FIFO. The remote DMA channel transfers data between the buffer memory and the host I/O port. With the addition of a bi-directional buffer isolating the DDC from the microprocessor, simultaneous drive operations can be accomplished. While the DDC is transferring data via DMA with the buffer memory, the local microprocessor can issue drive control commands.

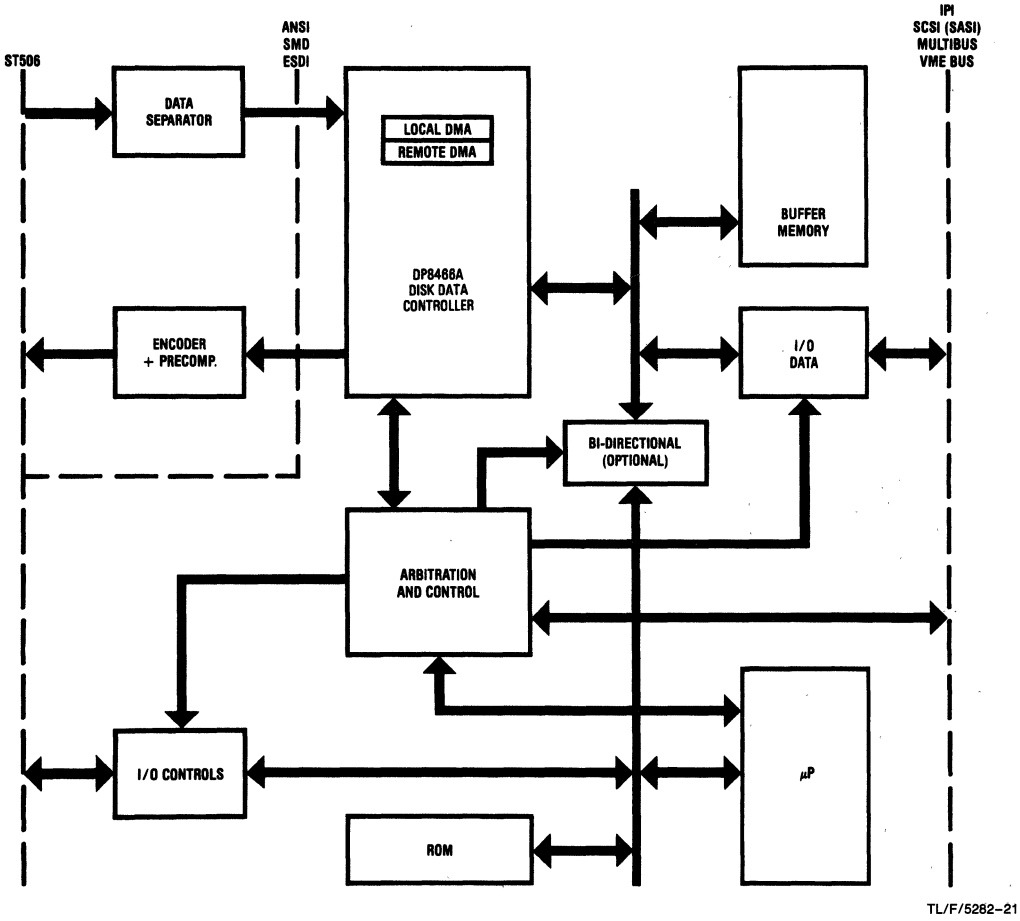


FIGURE 26. High Performance System

INDEX

11.0 Absolute Maximum Ratings*

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

Supply Voltage (V_{CC})	−0.5 to +7.0V
DC Input Voltage (V_{IN})	−0.5 to V_{CC}+0.5V
DC Output Voltage (V_{OUT})	−0.5 to V_{CC}+0.5V
Storage Temperature Range (TSTG)	−65°C to +150°C
Power Dissipation (PD)	500 mW
Lead Temperature (TL) (Soldering 10 sec.)	260°C
ESD Tolerance: C_{ZAP} = 100 pF, R_{ZAP} = 1500Ω	1600V

*Absolute Maximum Ratings are those values beyond which damage to the device may occur.

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

Symbol	Parameter	Conditions	Typ	Limit	Units
V_{IH}	Minimum High Level Input Voltage	(Note 1)		2.0	V
V_{IL}	Maximum Low Level Input Voltage	(Note 1)		0.8	V
V_{OH1}	Minimum High Level Output Voltage (Note 2)	$\|I_{OUT}\|$ = 20 μA		V_{CC} − 0.1	V
V_{OH2}		ADS0, ADS1 $\|I_{OUT}\|$ = 4.0 mA For All Other Outputs $\|I_{OUT}\|$ = 2.0 mA		3.5	V
V_{OL1}	Maximum Low Level Output Voltage (Note 2)	$\|I_{OUT}\|$ = 20 μA		0.1	V
V_{OL2}		ADS0, ADS1 $\|I_{OUT}\|$ = 4.0 mA For All Other Outputs $\|I_{OUT}\|$ = 2.0 mA		0.4	V
I_{IN}	Maximum Input Current	V_{IN} = V_{CC} or GND		±1	μA

12.0 DC Electrical Characteristics

(V_{CC} = 5V ±10%, unless otherwise specified) T_A = 0°C to +70°C (Continued)

Symbol	Parameter	Conditions	Typ	Limit	Units
I_{OZ}	Maximum TRI-STATE Output Leakage Current	$V_{OUT} = V_{CC}$ or GND		±10	μA
I_{CC}	Average Supply Current DP8466AN-12 (Note 3)	$V_{IN} = V_{CC}$ or GND BCLK = RCLK = 12 MHz I_{OUT} = 0 μA	12	30	mA
	Average Supply Current DP8466AN-20 (Note 3)	$V_{IN} = V_{CC}$ or GND RCLK = 20 MHz BCLK = 16 MHz, I_{OUT} = 0 μA	20	40	mA
	Average Supply Current DP8466AN-25 (Note 3)	$V_{IN} = V_{CC}$ or GND BCLK = 20 MHz RCLK = 25 MHz I_{OUT} = 0 μA	25	45	mA

Note 1: Limited functional test patterns are performed at these levels. The majority of functional test patterns are performed with input levels of 0V and 3V for AC Timing Verification.

Note 2: Outputs are "conditioned" for Tested States by normal functional test patterns. Device clocks are disabled and a purely static measurement is performed.

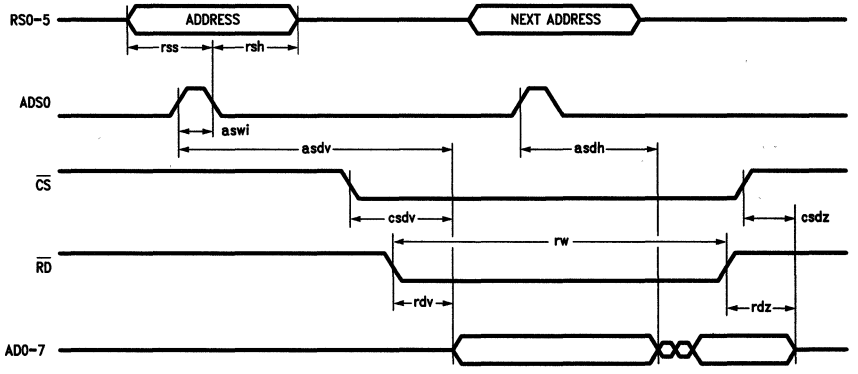
Note 3: Device is in normal operating mode and is measured with bypass capacitor of 0.1 μF between V_{CC} and Ground.

13.0 AC Electrical Characteristics & Timing Diagrams

NATIONAL SEMICONDUCTOR PRELIMINARY TIMING FOR THE DP8466A

Note: Refer to 11.4 for AC Timing Test Conditions.
Refer to 11.5.6 for derating factor.

13.1 REGISTER READ (Latched Register Select: ADS0 Active)



TL/F/5282–22

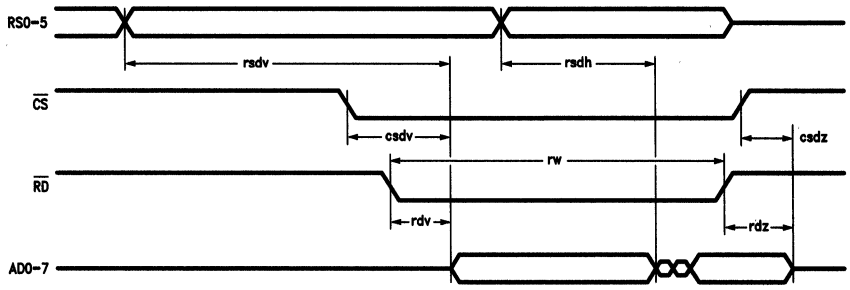
Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
rss	Register Select Setup to ADS0 Low	10		15		ns
rsh	Register Select Hold to ADS0 Low	10		15		ns
aswi	Address Strobe Width In	20		30		ns
asdv	Address Strobe to Data Valid (Note 1)		150		200	ns
csdv	Chip Select to Data Valid		125		150	ns
rdv	Read Strobe to Data Valid		125		150	ns
rw	Read Strobe Width		10		10	μs
csdz	Chip Select to Data TRI-STATE (Note 2)	20	80	20	90	ns
rdz	Read Strobe for Data to TRI-STATE (Note 2)	20	80	20	90	ns
asdh	Data Hold from ADS0 (Note 1)	20		20		ns

Note 1: asdv and asdh timing is referenced to the leading edge of ADS0 or the leading edge of valid address, whichever comes last.

Note 2: TRI-STATE note: These limits include the RC delay inherent in our test method. These signals typically turn off within 15 ns, enabling other devices to drive this line with no contention.

13.0 AC Electrical Characteristics & Timing Diagrams (Continued)

13.2 REGISTER READ (Non-Latched Register Select: ADS0 = 1)



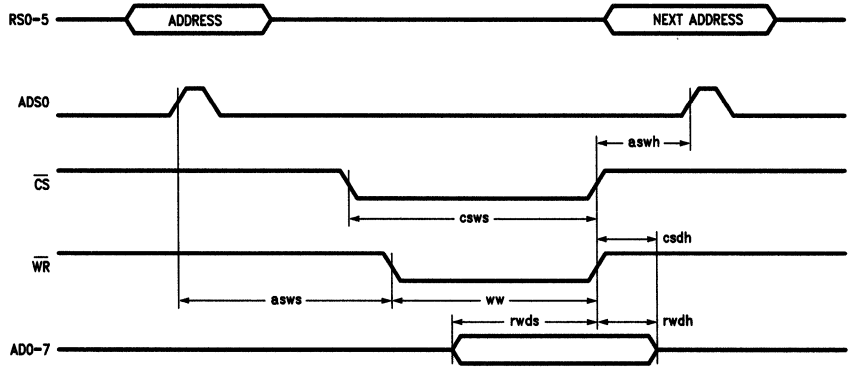
TL/F/5282–23

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
rsdv	Register Select to Data Valid (ADS0 = 1) (Note 1)		150		200	ns
csdv	Chip Select to Data Valid		125		150	ns
rdv	Read Strobe to Data Valid		125		150	ns
rw	Read Strobe Width		10		10	μs
csdz	Chip Select to Data TRI-STATE (Note 2)	20	80	20	90	ns
rdz	Read Strobe for Data to TRI-STATE (Note 2)	20	80	20	90	ns
rsdh	Data Hold from Register Select Change (Note 1)	20		20		ns

Note 1: rsdv and rsdh timing assumes that ADS0 is true when RS0–5 changes.

Note 2: TRI-STATE note: These limits include the RC delay inherent in our test method. These signals typically turn off within 15 ns, enabling other devices to drive this line with no contention.

13.3 REGISTER WRITE (Latched Register Select: ADS0 Active)



TL/F/5282–24

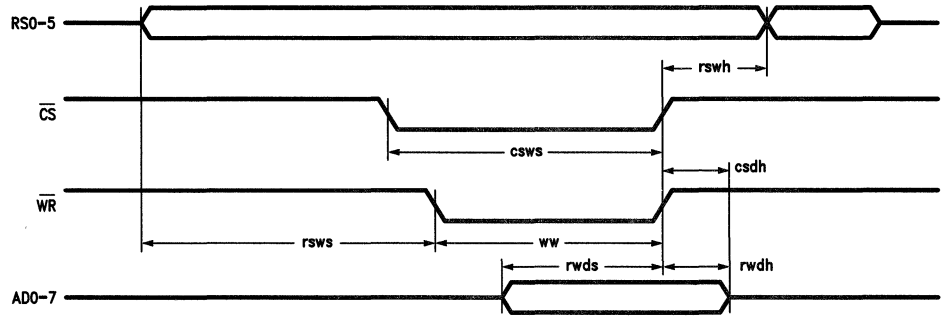
Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
asws	Address Strobe to Write Setup (Note 1)	15		20		ns
csws	Chip Select to Write Setup	50		70		ns
csdh	Chip Select Data Hold (Note 2)	7		10		ns
rwds	Register Write Data Setup	40		50		ns
rwdh	Register Write Data Hold (Note 2)	3		5		ns
ww	Write Strobe Width	50		70		ns
aswh	ADS0 Hold from Write (Note 1)	10		15		ns

Note 1: asws and aswh timing is referenced to the leading edge of ADS0 or the leading edge of valid address, whichever comes last.

Note 2: Minimum data hold time for a register write is referenced to $\overline{CS}$ or $\overline{WR}$, whichever goes inactive high first.

13.0 AC Electrical Characteristics & Timing Diagrams (Continued)

13.4 REGISTER WRITE (Non-Latched Register Select: ADS0 = 1)



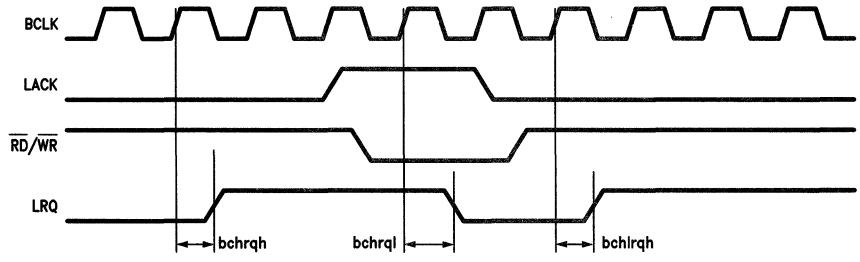
TL/F/5282–25

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
rsws	Register Select to Write Setup (Note 1)	10		15		ns
csws	Chip Select to Write Setup	50		70		ns
csdh	Chip Select to Data Hold (Note 2)	7		10		ns
rwds	Register Write Data Setup	40		50		ns
rwdh	Register Write Data Hold (Note 2)	3		5		ns
ww	Write Strobe Width	50		70		ns
rswh	Register Select Hold from Write (Note 1)	15		20		ns

Note 1: rsws and rswh assume that ADS0 is true when RS0–5 changes.

Note 2: Minimum data hold time for a register write is referenced to $\overline{CS}$ or $\overline{WR}$, whichever goes inactive high first.

13.5 LRQ TIMING WITH EXTERNAL DMA



TL/F/5282–26

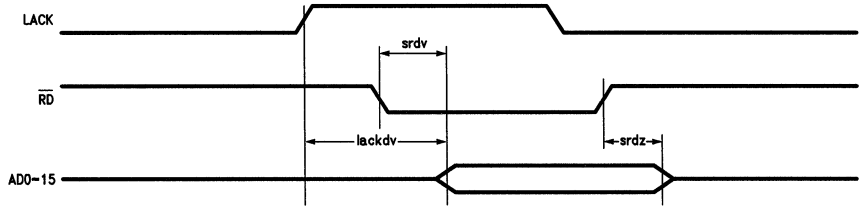
Symbol	Parameter	DP8466A-25/20		DP8466AN-12		Units
		Min	Max	Min	Max	
bchrqh	BCLK High to LRQ High		75		100	ns
bchrql	BCLK High to LRQ Low		75		100	ns

Note 1: The "ON" condition for the slave mode DMA, once the LRQ is active, is when both LACK and the $\overline{RD}$ or $\overline{WR}$ strobes are active. The LRQ is then removed after the next BCLK as shown. The "OFF" condition for the slave mode DMA is determined by the $\overline{RD}$ or $\overline{WR}$ strobe becoming inactive and the LRQ could be deasserted from the next BCLK rising edge. Lack does not play a role in determining the "OFF" condition.

Note 2: National recommends to use the same clock that generates the external $\overline{RD}$ & $\overline{WR}$ strobes for BCLK.

13.0 AC Electrical Characteristics & Timing Diagrams (Continued)

13.6A READING FIFO DATA IN DMA SLAVE MODE



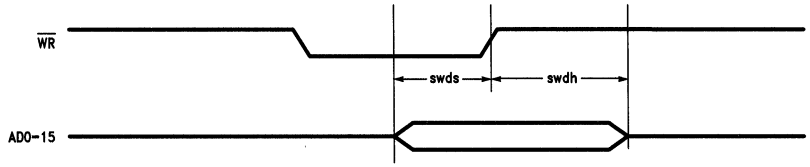
TL/F/5282–27

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
lackdv	LACK to Data Valid		80		90	ns
srdv	Slave Read Strobe to Data Valid		60		70	ns
srdz	Slave Read Strobe to Data TRI-STATE (Note 3)	20	80	20	90	ns

Conditions: Disk read operation, DMA disabled, LRQ output true.

Note 3: TRI-STATE Note: These limits include the RC delay inherent in our test method. These signals typically turn off within 15 ns, enabling other devices to drive this with no contention.

13.6B WRITING FIFO DATA IN DMA SLAVE MODE

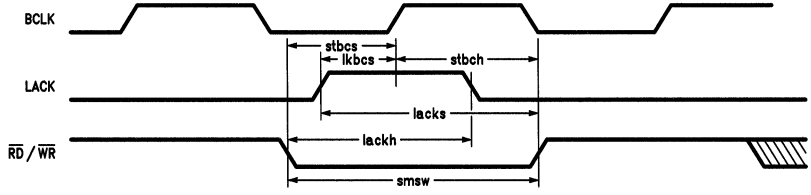


TL/F/5282–28

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
swds	Slave Write Data Setup	5		10		ns
swdh	Slave Write Data Hold	20		28		ns

Conditions: Disk write operation, DMA disabled, LRQ output true.

13.7 ADDITIONAL SLAVE MODE DMA TIMING



TL/F/5282–78

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
smsw	Slave Mode Strobe Width	40	(Note 2)	50	(Note 2)	ns
lacks	Lack to Strobe Setup	50		60		ns
lackh	Strobe to Lack Hold	10		15		ns
lkbcs	Lack to Bus Clock Setup	35		40		ns
stbch	Strobe from Bus Clock Hold	10		15		ns
stbcs	Strobe to Bus Clock Setup	20		25		ns

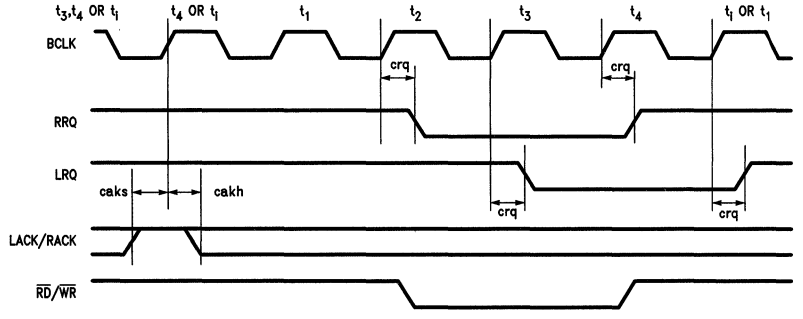
Conditions: Disk read or disk write operation, internal DMA disabled, and LRQ output active.

Note 1: The Read or Write Cycle begins when Lack and ($\overline{WR}$ or $\overline{RD}$) are true. From this point Lack must be held true for lackh and $\overline{WR}$ or $\overline{RD}$ must remain true for smsw.

Note 2: Disk Read or Write Byte Transfer Rate cannot exceed DMA Byte Transfer Rate. The inactive $\overline{RD}/\overline{WR}$ pulse width must be at least 2 BCLK cycles.

13.0 AC Electrical Characteristics & Timing Diagrams (Continued)

13.8 LOCAL AND REMOTE DMA ACKNOWLEDGE



TL/F/5282–29

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
crq	Bus Clock to Request (Notes 5, 6)		85		100	ns
caks	Acknowledge Setup to Clock	20		25		ns
cakh	Bus Clock to Remote Status	10		15		ns

Note 1: The Local and Remote Acknowledges are sampled at the beginning of bus cycles t4 and t1.

Note 2: Local Acknowledge has internal priority over Remote Acknowledge.

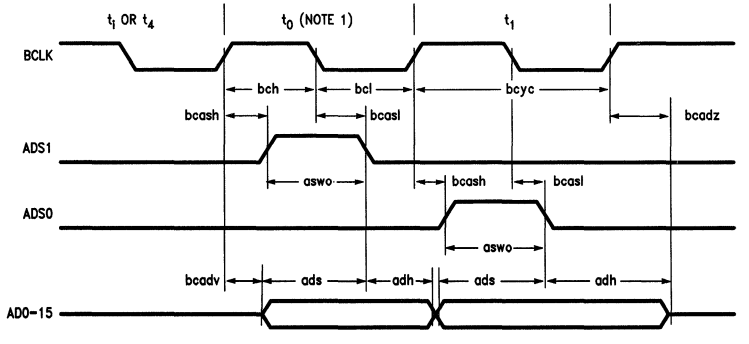
Note 3: Local and Remote Acknowledge are ignored if their respective Request output line is false.

Note 4: Above timing is for 16 bit address updates. For 32 bit Local address mode, cycle t0 occurs on the first transfer of an operation or when the lower 16 bits of the address rollover.

Note 5: crq is implied to be the same for both assertion and deassertion of LRQ or RRQ.

Note 6: LRQ will deassert on t2 for the final deassertion.

13.9 DMA ADDRESS GENERATION



TL/F/5282–30

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
bcyc	Bus Clock Cycle Time (Notes 2, 3)	50	10,000	80	10,000	ns
bch	Bus Clock High Time (Note 3)	22.5	10,000	32	10,000	ns
bcl	Bus Clock Low Time (Note 3)	22.5	10,000	32	10,000	ns
bcash	Bus Clock to Address Strobe High		45		55	ns
bcasl	Bus Clock to Address Strobe Low		50		60	ns
aswo	Address Strobe Width Out	bch		bch		ns
bcadv	Bus Clock to Address Valid		60		70	ns
bcadz	Bus Clock to Address TRI-STATE (Note 4)	20	80	20	90	ns
ads	Address Setup to ADS0/1 Low	bch − 17		bch − 22		ns
adh	Address Hold from ADS0/1 Low	bcl − 10		bcl − 10		ns

Note 1: Cycle t0 occurs only on the first transfer of an operation or when the lower 16 bits of the address rolls over.

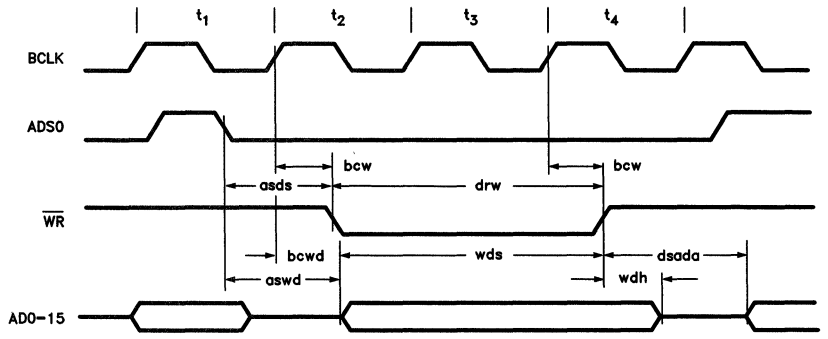
Note 2: The rate of bus clock must be high enough that data will be transferred to and from the FIFO faster than the data being transferred to and from the disk.

Note 3: For DP8466A-20, minimum bcyc = 60 ns minimum bch = bcl = 28 ns.

Note 4: TRI-STATE note: These limits include the RC delay inherent in our test method. These signals typically turn off within 15 ns, enabling other devices to drive this line with no contention.

13.0 AC Electrical Characteristics & Timing Diagrams (Continued)

13.10 DMA MEMORY WRITE



TL/F/5282–31

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
bcw	Bus Clock to Write Strobe		50		60	ns
wds	Data Setup to $\overline{WR}$ High (Note 1)	2bcyc − 35		2bcyc − 45		ns
wdh	Data Hold from $\overline{WR}$ high (Note 1, 3)	8	50	8	60	ns
bcwd	Data Valid from t2 Clock (Note 1)		75		90	ns
asds	Address Strobe to Data Strobe (Note 2)		bcl + 10		bcl + 20	ns
aswd	Address Strobe to Write Data Valid		bcl + 40		bcl + 60	ns

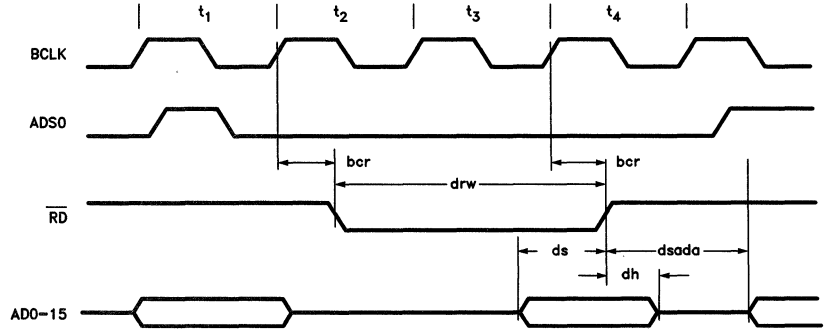
Conditions: DMA write, Local or Remote transfer, internal DMA.

Note 1: Data is enabled on AD0–15 only in local DMA transfers.

Note 2: Data strobe is either $\overline{RD}$ or $\overline{WR}$ out.

Note 3: TRI-STATE Note: These limits include the RC delay inherent in our test method.

13.11 DMA MEMORY READ



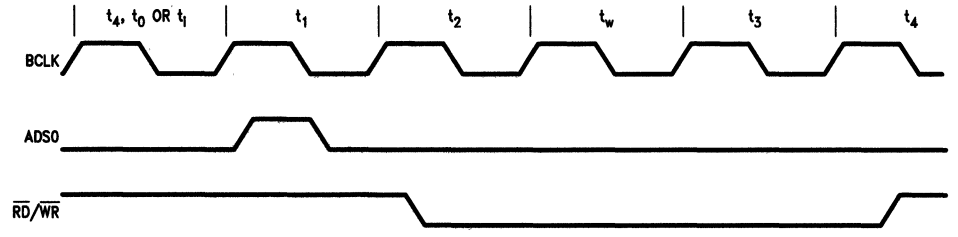
TL/F/5282–32

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
bcr	Bus Clock to Read Strobe		50		60	ns
ds	Data Setup to Read Strobe High	30		35		ns
dh	Data Hold from Read Strobe High	0		0		ns
drw	DMA Data Strobe Width Out	2bcyc − 10		2bcyc − 15		ns
dsada	DMA Data Strobe to Address Bus Active	bcyc − 10		bcyc − 10		ns

Note 1: ds and dh timing are for Local transfers only.

13.0 AC Electrical Characteristics & Timing Diagrams (Continued)

13.12 DMA WITH INTERNAL WAIT STATES



TL/F/5282–33

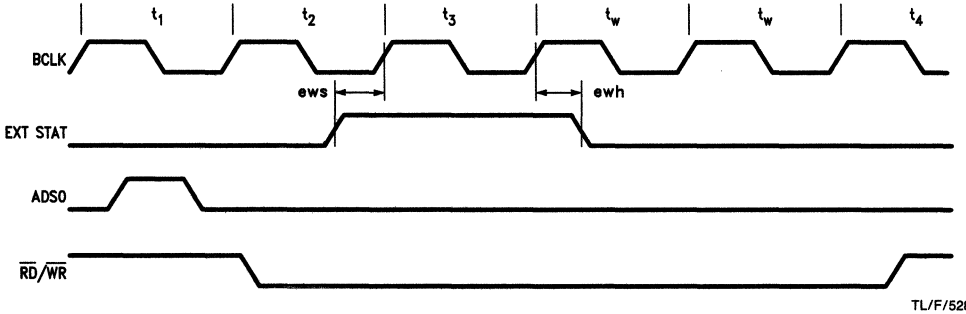
Conditions: Local or Remote DMA transfer, read or write, internal DMA.

Note 1: Addition of an internal wait state will lengthen $\overline{RD}/\overline{WR}$ strobes by an additional bus clock cycle.

Note 2: Internal wait states are enabled by setting the Slow Read/Write bits in the Local and Remote Transfer registers.

Note 3: If used, external wait states will be added between cycles t3 and t4.

13.13 DMA WITH EXTERNAL WAIT STATES



TL/F/5282–34

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
ews	External Wait Setup to t3 Clock	15		20		ns
ewh	External Wait Hold after tw Clock	10		15		ns

Conditions: Read or write, internal DMA mode. Local or Remote transfer.

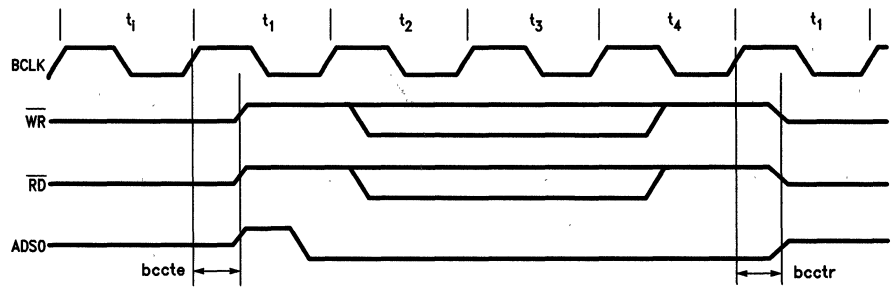
Note 1: Addition of external wait states will extend $\overline{RD}/\overline{WR}$ strobes by an integral number of bus clock cycles.

Note 2: If enabled, an internal wait state is added between cycles t2 and t3.

Note 3: EXT STAT is sampled upon entering states t3 and tw, and adds wait states one bus clock cycle later.

13.0 AC Electrical Characteristics & Timing Diagrams (Continued)

13.14 DMA CONTROL SIGNALS

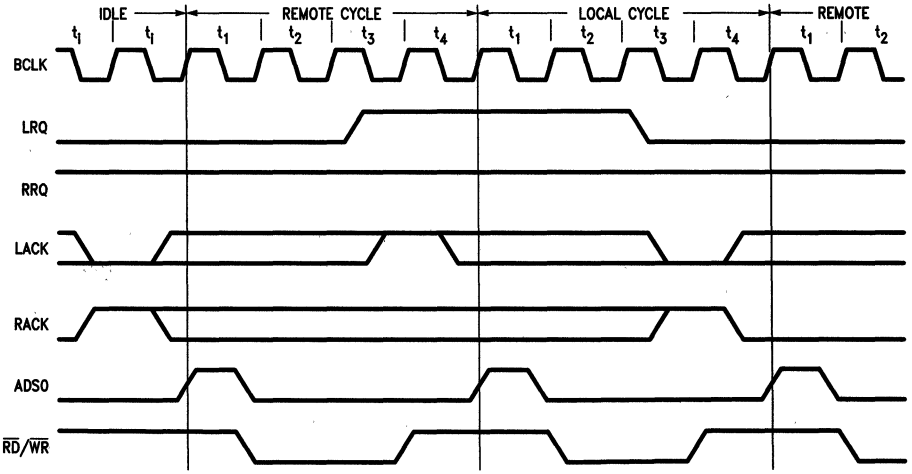


TL/F/5282–35

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
bccte	Bus Clock to Control Enable (WR, RD, ADS0)		55		70	ns
bcctr	Bus Clock to Control Release (WR, RD, ADS0) (Note 1)		60		70	ns

Note 1: TRI-STATE note: These limits include the RC delay inherent in our test method. These signals typically turn off within 15 ns, enabling other devices to drive this line with no contention.

13.15 LOCAL AND REMOTE DMA INTERLEAVING



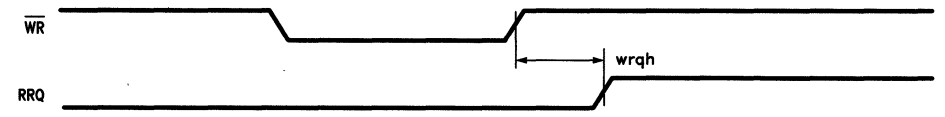
TL/F/5282–36

Note 1: Timing of the acknowledge pulses are used for illustration. Acknowledges need only to be set up with respect to t4 and t1 clock cycle.

Note 2: If both LACK and RACK are asserted with both LRQ and RRQ pending, a local DMA transfer will be performed.

13.0 AC Electrical Characteristics & Timing Diagrams (Continued)

13.16 RRQ ASSERTION AFTER WRITING TO OC REGISTER FOR REMOTE TRANSFER

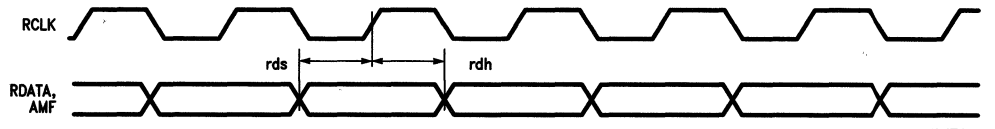


TL/F/5282–37

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
wrqh	Write Strobe to Remote Request High		100		150	ns

Conditions: Non-tracking mode, writing "Start Remote Input/Output" to the Operation Command register.

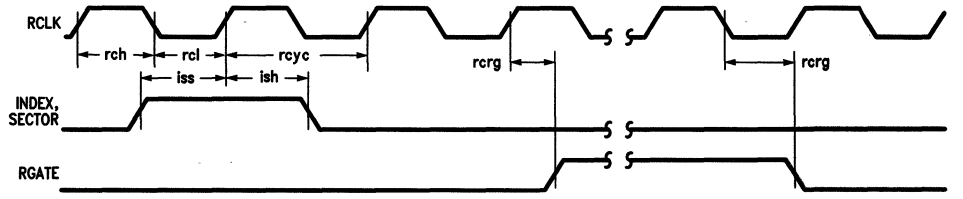
13.17 READ DATA TIMING



TL/F/5282–38

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
rds	Read Data/AMF Setup to Read Clock	10		15		ns
rdh	Read Data/AMF Hold to Read Clock	10		15		ns

13.18 RGATE ASSERTION FROM INDEX OR SECTOR PULSE INPUT



TL/F/5282–39

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
rcyc	Read Clock Cycle Time (Notes 2, 3)	39	10,000	80	10,000	ns
rch	Read Clock High Time (Note 2)	16	10,000	32	10,000	ns
rcl	Read Clock Low Time (Note 2)	16	10,000	32	10,000	ns
iss	Index/Sector Setup to Read Clock	10		15		ns
ish	Index/Sector Pulse Hold	10		15		ns
rcrg	Read Clock to Read Gate		55		70	ns

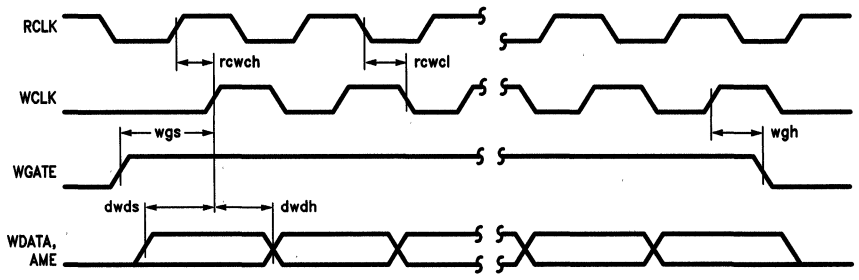
Note 1: INDEX/SECTOR low must meet iss/ish timing for proper INDEX/SECTOR pulse detection.

Note 2: For DP8466A-20, minimum rcyc = 50 ns, minimum rch and rcl = 20 ns.

Note 3: For DP8466A-25, this parameter is not tested directly, but is guaranteed through correlation.

13.0 AC Electrical Characteristics & Timing Diagrams (Continued)

13.19 WRITE DATA TIMING FOR NRZ TYPE DATA

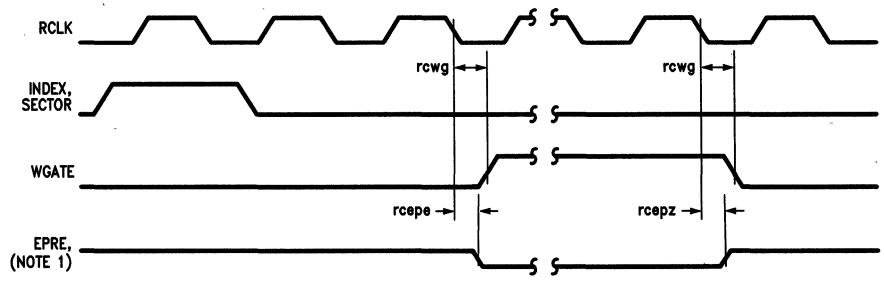


TL/F/5282–40

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
rcwch	Read Clock to Write Clock High Delay		30		40	ns
rcwcl	Read Clock to Write Clock Low Delay		30		40	ns
rcwcs	Absolute Value of (rcwcl — rcwch)		6		7	ns
dwds	Drive Write Data Setup to Write Clock	rcl − 10		rcl − 15		ns
dwdh	Drive Write Data Hold to Write Clock	rch − 5		rch − 8		ns
wgs	Write Gate Setup to Write Clock	rcl − 10		rcl − 15		ns
wgh	Write Gate Hold to Write Clock	rch		rch		ns

Note 1: rcl and rch are described in Timing Diagram 13.18.

13.20 WGATE ASSERTION FROM INDEX OR SECTOR PULSE INPUT



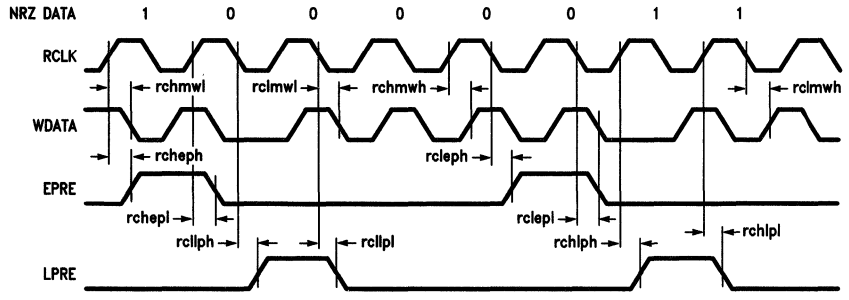
TL/F/5282–41

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
rcwg	Read Clock to Write Gate		40		50	ns
rcepe	Read Clock to Early Precomp Enabled		50		60	ns
rcepz	Read Clock to Early Precomp TRI-STATE		50		60	ns

Note 1: Early Precompensation (EPRE) is used as an output only when writing MFM data.

13.0 AC Electrical Characteristics & Timing Diagrams (Continued)

13.21 WRITE DATA TIMING FOR MFM TYPE DATA



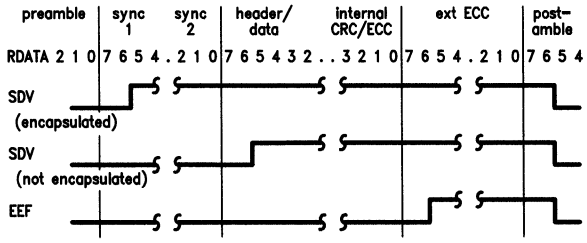
TL/F/5282–42

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
rchmwh	RCLK High to MFM WDATA High		40		50	ns
rchmwl	RCLK High to MFM WDATA Low		40		50	ns
rclmwh	RCLK Low to MFM WDATA High		40		50	ns
rclmwl	RCLK Low to MFM WDATA Low		40		50	ns
rcheph	RCLK High to EPRE High		40		50	ns
rchepl	RCLK High to EPRE Low		40		50	ns
rcleph	RCLK Low to EPRE High		40		50	ns
rclepl	RCLK Low to EPRE Low		40		50	ns
rchlph	RCLK High to LPRE High		40		50	ns
rchlpl	RCLK High to LPRE Low		40		50	ns
rcllph	RCLK Low to LPRE High		40		50	ns
rcllpl	RCLK Low to LPRE Low		40		50	ns

13.0 AC Electrical Characteristics & Timing Diagrams (Continued)

13.22 POSITIONAL TIMING FOR SDV AND EEF

Read operation (Compare Header, Read Header, Compare Data or Read Data)

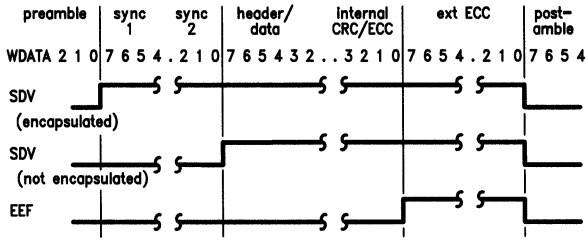


TL/F/5282–43

Note 1: Data should be delayed 2 bit times before entering external ECC circuitry in order for it to properly align correctly with SDC and EEF.

Note 2: Encapsulation is controlled by the HEN and DEN bits in the EC register, and causes the sync patterns to be included in the CRC/ECC calculation.

Write operation (Write Header, Write Data or Format Track)

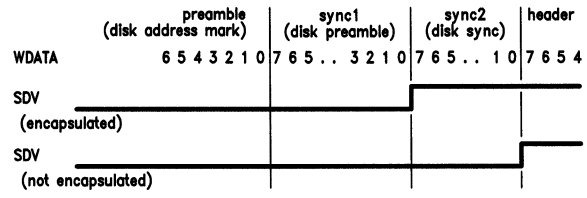


TL/F/5282–44

Note 1: Write operation shown is for NRZ data. For MFM encoding, Write data is delayed two bit times relative to NRZ data.

Note 2: Encapsulation is controlled by the HEN and DEN bits in the EC register, and causes the sync patterns to be included in CRC/ECC calculation.

Write header operation (Start with Address Mark)



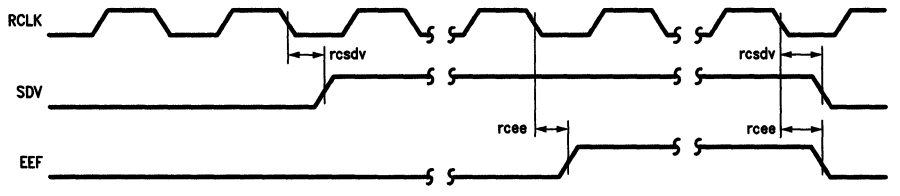
TL/F/5282–45

Note 1: Field names within parenthesis are the names of the fields on disk.

Note 2: Encapsulation is controlled by the HEN bit in the EC register, and causes the sync patterns to be included in CRC/ECC calculation.

13.0 AC Electrical Characteristics & Timing Diagrams (Continued)

13.23 FIELD ENVELOPE TIMING



TL/F/5282–46

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
rcsdv	Read Clock to Serial Data Valid		35		50	ns
rcee	Read Clock to External ECC		35		50	ns

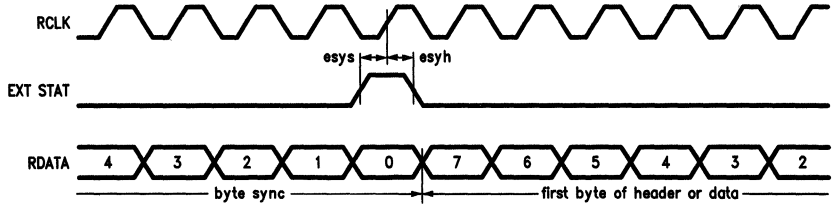
Note 1: SDV is asserted after sync fields, and is deasserted at the start of the postamble field. If sync field encapsulation is enabled, SDV is asserted at the start of the sync fields.

Note 2: EEF is asserted at the start of the external ECC field, and is deasserted at the start of the postamble field.

Note 3: When the DDC is receiving data from the disk, the SDV and EEF are delayed by two bit times from incoming read data due to internal delays.

Note 4: If the external ECC count is set to zero, no EEF output will be generated.

13.24 EXTERNAL STATUS TIMING WHEN USING EXTERNAL BYTE SYNC



TL/F/5282–47

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
esys	External Byte Sync Setup to Rising Edge of Bit Clock 0 of Byte Sync	15		20		ns
esyh	External Byte Sync Hold to Rising Edge of Bit Clock 0 of Byte Sync	10		15		ns

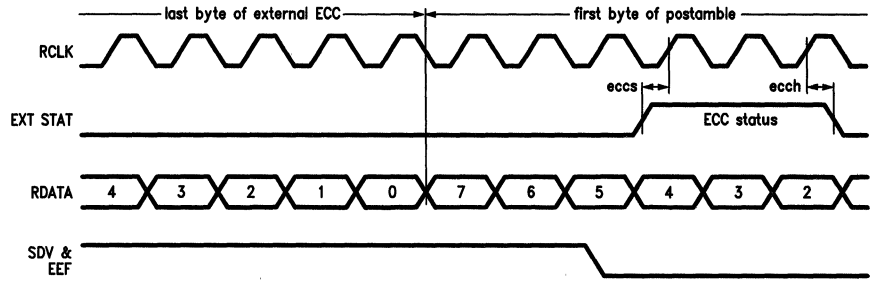
Note 1: The external sync feature can only be used if the Enable External Wait states (EEW) bit of the Remote Transfer (RT) register is not set.

Note 2: External circuitry is needed to feed the DDC with NRZ zeros until the external sync signal has been generated to prevent the DDC from trying to detect sync.

Note 3: If External Sync and External Wait states are not being used, the EXT. STAT. pin must be false during preamble and sync fields.

13.0 AC Electrical Characteristics & Timing Diagrams (Continued)

13.25 EXTERNAL STATUS TIMING WHEN USED FOR EXTERNAL ECC



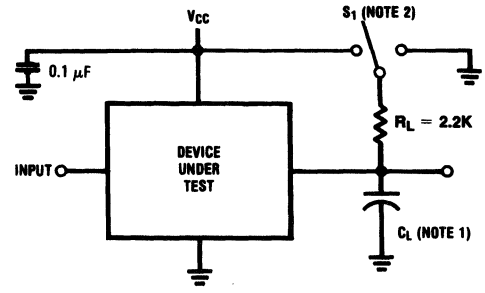
TL/F/5282–48

Symbol	Parameter	DP8466A-25/20		DP8466A-12		Units
		Min	Max	Min	Max	
eccs	External ECC Status Setup to Rising Edge of Bit Clock 4 of Postamble	15		20		ns
ecch	External ECC Status Hold to Rising Edge of Bit Clock 2 of Postamble	10		15		ns

Note 1: The external ECC error detection feature can only be used if the Enable External Wait states (EEW) bit of the Remote Transfer register (RT) is zero.

14.0 AC Timing Test Conditions

Input Pulse Levels	GND to 3.0V
Input Rise and Fall Times	5 ns
Input and Output Reference Levels	1.3V
TRI-STATE Reference Levels	Float (ΔV) ± 0.5V
Output Load (See Figure 27)	



TL/F/5282–77

FIGURE 27

Note 1: C_L = 50 pF, includes scope and jig 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.

Capacitance (T_A = 25°C, f = 1MHz)

Parameter	Description	Typ	Max	Unit
C_{IN}	Input Capacitance	7	12	pF
C_{OUT}	Output Capacitance	7	12	pF

Note: This parameter is sampled and not 100% tested.

15.0 Miscellaneous Timing Information

15.1 STATUS REGISTER TIMING

HEADER FAULT: This bit is set at the start of the Header Postamble field of a header with a CRC/ECC error. It is reset at the start of the Header Postamble of the header requested, or upon receipt of a new disk command. No interrupt is generated.

NEXT DISK COMMAND: This bit is set at the start of the Header Postamble of the last sector of an operation, and is reset upon loading the Drive Command register. No interrupt is generated.

HEADER MATCH COMPLETED: This bit is set at the start of the Header Postamble field of the header of interest. This bit is reset when the DDC begins the next header operation. An interrupt is generated if enabled.

LOCAL REQUEST: This bit has the same timing as the Local Request pin. When the FIFO requires servicing, this bit is set. When service is no longer required, this bit is cleared. No interrupt is generated.

REMOTE COMMAND BUSY: In the tracking mode, this bit is set 3–5 RCLK's after receipt of a drive command. In the non-tracking mode, this bit is set when either a Start Remote Input or Start Remote Output command is received in the Operation Command register. This bit is reset and interrupt is generated upon completion of the initiating operation.

LOCAL COMMAND BUSY: This bit is set 3–5 RCLK's after receipt of a drive command which requires the use of the local channel. It is reset after the last transfer of the local channel if in the non-tracking mode or writing the disk, or after the last transfer of the remote channel if in the tracking mode and reading disk. Interrupt is generated upon completion of the initiating operation.

CORRECTION CYCLE ACTIVE: This bit is set upon receipt of the Start Correction Cycle in the Operation Command register, and is reset at the end of the correction operation. An interrupt is generated at the end of the correction cycle.

ERROR DETECTED: This bit is a logical OR function of all the bits in the Error register. An interrupt is generated when an error is detected.

15.2 ERROR REGISTER TIMING

HFASM ERROR: If while in the HFASM mode the sector address matches and another header byte does not, this bit will be set at the start of the Header Postamble field.

DATA FIELD ERROR: If the Data field contains a CRC/ECC error, this bit will be set at the start of the Data Postamble field.

SECTOR NOT FOUND: If the header of the desired sector is not located before two index pulses are received, this bit will be set upon receipt of the second index pulse.

SECTOR OVERRUN: If an index or sector pulse is detected while reading the Header or Data field, or while writing and not in the Gap field, this bit will be set upon receipt of the sector/index pulse.

NO DATA SYNC: If an index or sector pulse is received before data sync is detected, this bit is set upon receipt of the sector/index pulse. If there is a data sync error after the first sync byte has been detected, this bit will be set during the byte following the byte in error.

FIFO DATA LOST: If a transfer between the disk and FIFO causes the FIFO to underrun or overrun, this bit will be set within the next byte time creating a write splice if write gate was on. This is reflected as an ECC error and can be removed if sector is rewritten.

CORRECTION FAILED: This bit is set at the end of the correction cycle if the error is non-correctable.

LATE INTERLOCK: This bit is set at the start of Data Postamble field for Read operations and at the end of the postamble field for non-format Write operations. While formatting, this bit is set at the end of the Gap field.

15.3 GENERAL TIMING FOR READ GATE

Whenever the DDC is reading, comparing, or in some cases, ignoring information, RGATE is asserted. The use of RGATE can be separated into three groups: Header search (soft sectored mode), header examination, and data examination.

SEARCHING FOR HEADERS

When the DDC is searching for a header in the soft-sectored mode, RGATE is asserted in a somewhat random location in the format. After being asserted, if the DDC does not recognize the address mark pattern within eight bit times of detecting a one, RGATE will be de-asserted in 18½ RCLK's. RGATE will then remain low for 17½ RCLK's before another search attempt is made.

In modes where the DDC starts a Read, Compare or Ignore Header operation at an index or sector pulse, RGATE will be asserted 3–4 RCLK cycles from detection of the index or sector pulse.

DATA OPERATIONS

After the header operation has completed, RGATE will be removed two bits after the start of the Header Postamble. If a Read or Check Data operation is to follow, RGATE will be reasserted 11½ bits after the Header Postamble.

At the end of the Data field, RGATE will be removed two bits into the start of the Data Postamble.

15.4 WRITE GATE TIMING

Whenever the DDC is writing information, WGATE is asserted. WGATE can be separated into three uses: Writing header, writing data or track formatting.

WRITING HEADERS

When the DDC writes the header, the write operation does not begin until the receipt of an index or sector pulse. After the pulse is detected, WGATE will be asserted 2½–3½ RCLKs from the detection of the pulse. WGATE will stay true until the end of the Header Postamble, unless the Data field is to be written. If the Data field is to be written, WGATE will not be de-asserted between the Header and Data fields.

WRITING DATA

After a header operation has properly completed, WGATE will be asserted 3 bit times into the Data Preamble. The WGATE will remain active until the end of the Data Postamble. Because of internal delays within the DDC, the Write Data operation is delayed three bit times from the header patterns.

15.0 Miscellaneous Timing Information (Continued)

FORMAT TRACK

In a format track operation, WGATE is asserted 2½–3½ RCLK's from the detection of the index pulse. WGATE will remain active until the next index pulse is detected, and will then be removed.

Note: Detection of an index or sector pulse is defined as the rising edge of the RCLK where index/sector input has met the setup time.

15.5 NORMAL INTERRUPTS

Interrupts are generated by the DDC for a variety of reasons, but they all fall into one of three categories: Either they signal normal completion, a synchronization point, or an error condition. If an interrupt is generated because of an error, the interrupt will have timing as described in the Error register timing section.

The Header Operation Complete interrupt is used for synchronization, and is enabled with the Enable Header Interrupt bit of the Operation Command register. This interrupt will occur when the DDC finishes the header operation, and starts the data operation. For Read, Compare, Write, or Ignore Header operations, the interrupt will be generated at the start of the Header Postamble field.

The normal Operation Complete interrupt is dependent on the operation being performed. If the operation is to Check Data, the interrupt is generated at the start of the Data Postamble field. For Write Data operations, an interrupt will be generated at the end of the Data Postamble. When the DDC is formatting, the interrupt will be delayed by the length of the Header Preamble after the format has finished. The fourth event is further defined by the DMA mode used. For all local channel operations except for tracking mode disk read, the interrupt will be generated during the last transfer of data from the FIFO. In the configuration, tracking mode disk read, the interrupt will be delayed until the last transfer is made by the remote DMA. For all non-tracking remote DMA operations, the interrupt will be generated during the last transfer of the remote DMA.

When a correction operation is being performed, an interrupt is generated at the end of the correction cycle, regardless of the outcome.

15.6 DERATING FACTOR

Output timings are measured with a purely capacitive load for 50 pF. The following correction factor can be used for other loads:

DP8466A-25/20	$C_L \geq 50$ pF:	+.13 ns/pF (ADS0, ADS1) +.20 ns/pF (all other outputs)
DP8466A-12	$C_L \geq 50$ pF:	+.18 ns/pF (ADS0, ADS1) +.25 ns/pF (all other outputs)

16.0 DP8466A Functional Status

Introduction

This section is intended to provide some relevant information on the functional status of the Disk Data Controller, DP8466A. Several problems have been identified in the DP8466 from the numerous beta sites. All of these were investigated and many were rectified resulting in the DP8466A as the production version. However there are still a few shortcomings which are outlined below for reference:

1.0 Correction Cycle Failure

If a correction cycle is attempted when an ECC/CRC error occurs in a multisectored disk operation with the sync word being encapsulated, then it will always fail because the ECC shift register gets preset. In order to ensure proper correction, a single sector retry must be attempted on the erroneous sector before correction cycle is initiated.

2.0 Error-Correction Handling Feature in Tracking Mode (Remote transfer of data conditional on Data ECC Error)

During tracking-mode read data operation, data will be transferred to local memory and then to a remote port. The DMA should prevent a remote transfer of the data until the DDC has checked for a data ECC error. Hence if correction is to be attempted, then it can be done in the local memory and then remote transfer can continue. However, the bad data will be sent to remote system memory without regard to its integrity and hence it's the responsibility of the user to correct the data in his system memory or send the correct data block again.

3.0 Odd Byte Remote DMA Transfer

Odd byte remote transfers are not allowed by the DMA mode. Therefore if only one transfer is desired to the remote port, it cannot be done. The only way to overcome this problem is to do a transfer of two bytes and ignore the second byte by reloading the remote data byte counter, etc.

4.0 Parameter RAM Registers Losing Contents

If at anytime the Read Clock input sees a glitch, then there is a good probability for some of the registers in the parameter RAM to lose their contents, e.g., ID sync #1, ID sync #2 etc. Whenever the Read Clock goes below the minimum specifications of 'rch' (read clock high time) and 'rcl' (read clock low time), it is considered as glitching the Read Clock. Hence it is the users responsibility to ensure that there are no glitches in the Read Clock input. In the future version, redesign will be attempted to decrease or totally remove the susceptibility of the chip to glitches on the Read Clk.

5.0 Remote DMA Interrupt Handling

In the non-tracking-mode remote DMA operation the operation complete interrupt could be held off or remain asserted despite servicing attempts whenever it happens while the disk header search is being attempted simultaneously. This will have to be taken care of in software. More details of this situation are provided in Chapter 2.

6.0 AME/AMF Handshake for ESDI (Softsectored Drives)

The DDC does not incorporate the handshake for ESDIsoft sectored disk operation. The DDC generates the AME signal only during the format operation and not during the read/write operation, when in the hardsectored, NRZ data mode. In the ESDI spec. Address Mark Found, AMF, responds only after AME is asserted. If AME is not asserted then AMF from the drive will not occur and the beginning of the sector will not be determined. The external logic and software methods needed to implement this handshake protocol is discussed in the design guide application note (AN413), in the MASS STORAGE data book.

7.0 Post Index/Sector Gap Field

The DDC has no defined field to implement the post index or post sector pulse gap. This can however be still imple-

16.0 DP8466A Functional Status
(Continued)

mented using a combination of software manipulation and external circuitry, as outlined in the design guide application note (AN413) in the MASS STORAGE data book.

8.0 Write Clock with Respect to Write Gate

In the DDC, Write Clock is generated 0.5 bit times after Write Gate is asserted. However in case of the SMD and ESDI drives they expect write clock to be active 250 ns (worst case) before write gate is asserted. This would have to be accomplished using external circuitry if desired.

17.0 Some Helpful Hints, When Designing a Disk Controller Subsystem with National Semiconductor's DP8466A, Disk Data Controller (DDC)

The following section provides some useful hints/application information for designing with the DP8466A. The suggestions given in this document are the results of situations encountered while debugging the designs at National and also from the feedback provided by the numerous beta site designs during their debug stages. This is an unending list and users are welcome to add their experiences, for they may save someone else a lot of trouble in the process. It should be understood that some of the situations outlined may be dependent on that particular system design approach and may not necessarily present itself in a different system environment. Hence National assumes no guarantees regarding these situations. A lot of the suggestions are explanations of inherent operational rules that may not be very evident in the chips documentation. For a detailed technical reference for design purposes, users are recommended to consult the DP8466 design guide in the MASS STORAGE handbook, while the DATA SHEET gives the features and timing specifications of the chip.

Sync #1 and Sync #2 Pattern Restrictions

When the DDC is in the read mode, i.e., read/compare/ignore header/or read/ check data, then it starts out looking for the sync byte. The data separator usually sends out zeroes when it is attempting to lock and when it has, it sends out the data coming off the disk. Hence the DDC is looking for the first non-zero bit to initiate sync byte comparison. If the DDC is programmed in the soft sectored mode then it basically attempts to do a compare for eight clocks before it asserts the abort address mark function internally and recycles Read Gate. In the hard sectored mode it will essentially be waiting for the sync match forever, till two revolutions of the disk, after which it gives a SNF error. Therefore it is not advisable to use a pattern of zeroes for the sync #1 or sync #2 bytes as that would result in an immediate sync byte alignment when read gate is asserted, as the serializer has been cleared to all zeroes. However, when writing information on the disk the sync #1 and sync #2 fields could be used to write a pattern of zeroes. This would probably be the case when some software manipulation is being attempted with the various fields of the DDC to implement some additional function like the post index gap, etc. *Hence, a pattern of zeroes is not recommended for sync #1 and sync #2 fields during a READ operation.*

Most Significant Bit of Sync Byte

When the sync byte is included in the CRC/ECC calculation, i.e., the encapsulated mode, controlled by the ECC Control Register, then it is **mandatory** that the most significant bit of the first sync byte be a 1. Hence, the most significant bit of the sync byte must be a 1.

Proper Sequence for Reset and Renable

The proper reset sequence for the chip consists of holding the RESET line active or the reset bit set in the OC register for 32 RCLKS and 4 BCLKS. Then this is deactivated and the RENABLE operation is initiated with a 01 in the DC register. It is possible, although not necessary for the renable operation to take as long as 260 RCLKS after which the operation complete interrupt would be generated. In case the status register is polled to detect operation completion, then the status register should be polled for the NDC bit set. Once set, it should be read after 30 RCLKS. If the NDC bit is still set then it signals the proper completion of the RENABLE operation.

Read/Write Registers

In the DDC some of the registers are defined as read only, while some are defined as write only. Care should be taken that read only registers should not be written to and write only registers should not be read from.

Write Header—Write Data Operation Variations

For the WRITE HEADER-WRITE DATA operation the DDC will fetch the data for the ID and DATA fields from the on-chip parameter RAM if the FMT bit is set in the DC register, (this also constitutes a format operation). The same scenerio with the FTF bit set in the DISK FORMAT register will fetch the ID and DATA information from the local buffer memory by the local DMA channel, and constitutes a full format operation. If however, the FMT bit is reset then the information is fetched from the local buffer memory by the local DMA channel and the operation is a regular one.

Status Reads on Interrupts

The STATUS register is read when an interrupt occurs to determine its cause and also serves to reset the interrupt line. However, if the status is read before 16 RCLKS after the interrupt, then the interrupt line will not be reset by the status register read. If the STATUS register is being constantly polled in the software, then it must not go faster then once in 16 RCLKS.

LCB Bit Behaviour

Whenever a disk operation is initiated, the LCB bit in the status register is set until the operation is complete. However if the operation is truncated due to any error condition in the header or the data fields, the LCB bit will remain set in the status register.

Resetting the DDC

After a normal reset of the DDC, none of the registers in the parameter RAM are affected. The STATUS and ERROR registers are cleared. The internal counters are reset and the FIFO pointers point to the beginning of the FIFO. There are, however, two other conditions that need to be taken into account.

If the DDC is reset while it is reading, writing, or formatting, the entire format RAM is potentially corrupted. For this reason, hex addresses 14–33, 38, 39, 3B–3F need to be reloaded after such a reset.

17.0 Some Helpful Hints, When Designing a Disk Controller Subsystem with National Semiconductor's DP8466A, Disk Data Controller (DDC) (Continued)

There is a certain relationship between BCLK and $\overline{WR}$ that can cause the DDC to be fooled into thinking it is in a DMA cycle for 2 BCLKs following a software reset (setting the reset bit in the Operation Command Register). This is also true if BCLK and $\overline{RST}$ have a certain relationship. The implications of this are that ADS0 will be asserted on the first BCLK and $\overline{WR}$ will be asserted on the second. Both will be cleared by the third. Internally, the following registers may change:

HA	Register
0F	Header Byte Count
13	NSO Counter
1A	Remote Data Byte Count (L)
1B	Remote Data Byte Count (H)
1C	DMA Address Byte 0
1D	DMA Address Byte 1
1E	DMA Address Byte 2
1F	DMA Address Byte 3
38	Sector Byte Count (L)
39	Sector Byte Count (H)

For this reason all of the above registers should be reloaded after a reset if it is not known that the timing relationship is such that the problem will not occur. Additionally, if $\overline{CS}$ is active at the time of the false $\overline{WR}$, then the register selected by the RS0–5 signals will be altered. In order to avoid a problem here, putting a zero on these lines will cause the write to go to the Status Register which cannot be written to, so no destructive write will occur.

Queing Disk Commands

After header match is successfully accomplished in a disk operation, also indicated by the header match complete interrupt if enabled, the NDC bit is set in the status register, indicating that the DDC is ready to accept the next disk command. Hence the next disk operation could be queued by a load of the DC register, however, it should be noted that the operation will not commence until the previous one has completed. Hence, care should be taken that registers being used while the data segment of the previous operation is in progress should not be changed, e.g., the registers associated with the DMA etc.

Assertion/Deassertion of LRQ/RRQ

In the burst DMA mode the request (LRQ/RRQ) is asserted when the set threshold is reached in the FIFO/LOCAL BUFFER MEMORY, and deasserted after the set burst is transferred even if the threshold has been reached again for the next transfer. The request is then reasserted for the next transfer.

Causes of Interrupts

There is only a single interrupt line on the DDC. There may be more than one source for the interrupt at times. It is hence recommended that every time an interrupt is serviced all the possibilities be checked to safeguard against more than one completion condition occurring at the same time.

HMC Bit in the Status Register

The HMC bit in the STATUS register is functional even if the header match complete interrupt is not enabled in the OC register. In a similar context it should be noted that even if the interrupts are not enabled in the OC register, the interrupt condition is generated internally when it happens. This EN bit in the OC register essentially controls the physical availability of the interrupt on the pin to the outside world.

Correction Cycle Initiation Sequence

When a CRC/ECC error occurs in a disk operation, the DDC has to be reset before a correction cycle can be attempted. On completion of the correction cycle the chip needs to be reset only if the correction cycle failed and hence an error condition resulted. In general, the DDC should be reset following any operation terminating in an error condition.

DFE (Data Field Error) Exceptions

Usually the Data Field Error condition in the DDC is terminal and the operation is aborted with an interrupt. However there is one exception to the rule. This is if the operation is a multisector check data operation in the interlock mode, then the DFE error will set the bit in the ERROR register but will not generate an interrupt and hence will not terminate the operation.

Read Header—Check Data Operation Exception

Normally the operation complete interrupt comes at the end of the data field for the disk operation. It is usually signified by the LCB bit reset in the STATUS register in case of non-tracking mode or by the LCB and RCB bit reset in the tracking mode. However there is one exception to the rule. In the case of a read header-check data operation, because local DMA transfers only the header and there is no DMA activity for the data field, the LCB bit is reset just after the header and the operation complete interrupt is generated. There is no interrupt at the end of the check data unless there is an ECC error in which case the operation is terminated with the error signalled through the STATUS and ERROR registers.

Header Fault Exceptional Behaviour

The HF (header fault) bit in the STATUS register is a passive error condition bit which is set if there is a CRC/ECC error in the header. This does not generate an interrupt nor terminate the operation normally. In a normal operation this bit is set if there is a header fault in the header, while searching for a sector and its gets reset, only if there is no CRC/ECC error in the header of the sector being sought. It should be noted that this behaviour is exhibited even when the DDC is searching for headers. However there is one exception to the rule. In case of a Read Header operation, if there is a CRC/ECC error in the header, then an interrupt is generated, the operation is terminated and the STATUS register will have the ED bit and the HF bit set while the ERROR register will read zeroes.

SC and NSO Counter Updates

The Number of Sector operations counter (NSO) in the DDC should be handled with care. Although addressed as one register, internally it is downloaded into two separate counters; one for the disk side and the other for the DMA logic. Whenever a read is done of the NSO counter, the value read back is the contents of the disk side NSO counter. The disk side NSO counter is decremented just after the header match complete interrupt, while the DMA side NSO counter is decremented while the local DMA channel is transferring

17.0 Some Helpful Hints, When Designing a Disk Controller Subsystem with National Semiconductor's DP8466A, Disk Data Controller (DDC) (Continued)

the last byte of the data field. If the SC and NSO counters have to be read/written by the microprocessor for some reason, care should be taken that they are not read/written when the DDC is accessing them internally, otherwise they might be zeroed. So it is recommended that they be read or updated about 1 μs after the HMC bit is set in the STATUS register. By the same token, this applies to other registers like the Remote Data Byte Count registers, Sector Byte Count registers, and DMA address registers. Also if the NSO counter has to be updated before the operation is completed to fool the DDC to go on some more without reloading the command then certain precautions need to be observed. Firstly the NSO register can be written to only after the DMA side NSO has been decremented. Secondly, the update cannot be done after the NSO on the DMA side has decremented to a 1, in other words the update cannot be done after the second to last sector, and hence has to be done at the latest before two sectors remain for the completion of the current multisector operation.

LT and RT Register Loading Restrictions

It is mandatory that the LT (RT) register must be loaded before the Sector Byte Count (Remote Data Byte) register pairs for any of the following situations.

a) If any internal DMA is being used or **b)** if the Remote Data Byte Count registers are going to be read by the processor, or **c)** if one needs to rewrite the LT or RT registers at anytime, (like when one wants to shift from tracking to non-tracking mode etc).

DMA Burst Mode Behaviour

One of the features of the DDC is that it can be programmed to do DMA transfers in the burst mode. The size of the burst is selectable through the LT & RT registers. Internally the burst value is downloaded to the burst counter which will reload itself only when the terminal count is reached or if the DDC is reset. The size of the DMA transfer is the length of the sector in case of a single sector transfer while in case of a multisector operation the DDC looks at it like one big transfer of length equal to the sector length times the number of sectors requested for the operation. This value is divided by the burst length which determines the number of bursts in the total transfer. If the total transfer length were not an even multiple of the burst length, then the very last burst would be less than the burst length selected. Control logic in the FIFO ensures that the remainder bytes are transferred even though it is less than the burst threshold. However, the internal burst counter remains at that lesser number and does not get reinitialized to the original burst value at the end of the operation. Hence the length of the first burst transfer of the next DMA operation may not be the same as that specified in the LT & RT registers.

Glitches on the Read Clock Input

The DDC has a minimum specification for the RCLK high time (rch) and the RCLK low time (rcl). Any RCLK not within these specifications is taken as a clock with the glitch. If such a situation is presented to the DDC then a number of things happen. This glitch results in throwing the Disk Sequencer in an unknown state, away from the standby state. Hence, in order for the DDC to be able to accept commands the chip has to be reset and reenabled in order to bring the sequencer to standby. A glitch on the RCLK can also potentially cause a situation leading to the altering of some register contents in the parameter RAM. Hence it is the designers responsibility to ensure that there are no possibilities of a glitch as defined by the specs on the RCLK line to the DDC and if it does reach the DDC, he should be aware of what to expect.

The DDC doesn't tolerate glitches on the RCLK input.

Remote DMA Completion Interrupt

The DMA on board the DDC is controlled by a separate sequencer. This DMA sequencer is responsible for generating the DMA completion interrupts and also controlling the LCB & RCB bits in the STATUS register. It is oblivious to the disk sequencer, in terms of the errors on the disk etc. However the interrupt generating mechanism for the remote DMA uses a clock from the disk PLA for synchronization purposes. This clock becomes inactive at certain times referred to as the freeze condition for the disk sequencer. This happens whenever a command is loaded in the DC register and the sequencer is waiting for a sync match in a disk read operation. Hence in the non-tracking mode if the remote DMA finishes at an instant when the disk sequencer is frozen then the remote DMA completion interrupt is held off till the next header comes along where the sequencer comes out of the frozen state and the clock is available. So this is more apt to happen when the remote DMA is under way while the disk sequencer is off looking for a header match. The other instances where the disk sequencer freezes is in a multisector operation; 1) the time after the header CRC and before the sync match for the data field occurs; 2) the time after the data field and just before the sync match of the header of the next field. Hence, if the remote DMA finishes around those instances then the completion interrupt could be delayed. The more serious implication of this situation is if the remote happened to finish just before the disk sequencer was entering the freeze mode, than the remote DMA completion interrupt would be held active till the sequencer comes out of the freeze state. Until then all efforts to service the interrupt by doing a status read will not deactivate the interrupt.

Hence the recommendation would be to initiate a remote operation only after a header match has occurred and to wait for the remote DMA completion before issuing another disk command. The other alternative would be to accommodate in software to look for such a situation and work around it. Software polling could be used to determine remote DMA completion and the interrupts from it ignored.

LRQ/RRQ Synchronization and Hold Off

In the DDC, the acknowledge signal in response to a request is sampled at the T4 state of the DMA transfer cycle. The chip does not require cycling of the acknowledge signal with the request from the chip. Also the initial assertion of the LRQ/RRQ signals is not synchronous to the BCLK.

Read Gate Algorithm for Harmonic Lock

If the read head was turned on over a write splice, the data separator may go into harmonic lock, which will prevent it from detecting the preamble pattern it is looking for. This forces zeroes data out of the data separator to the DDC and hence the DDC allows read gate to remain asserted, indefinitely. This is a lock up situation which must be avoided using external hardware.

17.0 Some Helpful Hints, When Designing a Disk Controller Subsystem with National Semiconductor's DP8466A, Disk Data Controller (DDC) (Continued)

Write Splice During a Disk Write

If a genuine FDL error occurs during a disk write function, the write gate will be deasserted as soon as the FIFO gets over-read. If this happens in the middle of a sector, it will result in a write splice to occur.

Read Gate Timing

Usually in most drives, when write gate is asserted, data actually gets written after 8 RCLKS, because of write driver delays etc. Hence the exists a write splice. In the DDC for a write data operation, the write gate is asserted 3 bit times into the data preamble. The read gate is asserted 8.5 bit times after the write gate, which is just sufficient to ensure assertion of read gate beyond the write splice associated with the write gate assertion. However at the beginning of the sector, both read and write gate are asserted 2–4 bit times from the index or sector pulse, hence resulting in the read gate being asserted in the write splice. External circuitry must be implemented to prevent this from happening.

FIFO Table Format

In the FIFO TABLE format mode the local DMA loads the correct number of header bytes (given by the HBC register) per sector into the FIFO from the local buffer memory. This data is then substituted for the header bytes during a format operation. It should be noted that each header byte set must contain an even number of bytes. If it contains an odd number of bytes, an extra dummy byte must be inserted so that each header byte set starts at an even byte boundary.

Two Interrupts in a Read Disk Operation

In a read disk operation, there is a potential for the controller μP to see two interrupts from the DDC, if a DFE error occurred during the operation. One of them is due to the error condition reflecting the DFE error, while the other reflects the operation completion by the local DMA, i.e., when the local DMA has finished transferring the data. Depending on the local DMA speed and bus latency, this could occur before or after the DFE interrupt, and they could be within 16 RCLKS or further apart. If the two interrupts are within 16 RCLKS of each other, the μP sees only one interrupts, while if they occur more than 16 RCLKS apart, then there is a potential of two interrupts being presented to the controller μP. This situation should be kept in mind and handled in firmware accordingly.

ADS0 Glitch During First DMA Transfer

The ADS0 line is a bidirectional line. Hence when the DMA transfer is initiated, the ADS0 line changes from an input to an output. It is released from the input mode into a tristate condition. When released for the DMA operation, it tends to touch the high level and goes low when the address needs to be latched in the t1 cycle of the first DMA transfer. It has been observed that just prior to that instant due to an internal race condition it is possible that the ADS0 may momentarily go low in a glitch fashion. This does not really hurt the system because it will go low at the appropriate time to latch the correct address in the t1 cycle, however if the trailing edge of the strobe is monitored to initiate some operation in a system design, then this could pose a problem. This needs to be kept in mind while designing.

Restrictions for the 2 Byte Exact Burst DMA Transfer Mode

The two byte exact burst mode was intended to be used for systems with very fast BCLKS relative to the RCLKS, such as when using the DDC to write a floppy as back up. The 2 byte exact mode is not needed for quick bus access since this can always be accomplished by the arbitration logic (in any burst mode) by deasserting LACK and waiting a minimum of 4 BCLKS. This is a better response than the 2 byte exact burst mode when waiting for the LRQ to be deasserted.

The performance degradation of the DDC when in the exact burst mode is due to the following sequence of events.

1. A burst of data is transferred causing the LRQ to go inactive.
2. Because the FIFO is still in a condition which requires more data to be transferred, the LRQ must be reasserted.
3. This reassertion of LRQ is held off until both the FIFO address counters match in parity; that is until both counters are either odd or even. This results in the LRQ being held off until the disk strobe occurs which in some cases (see table below) will allow the DMA to transfer only at the same data rate as the disk.

The most exaggerated effect of this problem is when in the 2 byte exact burst mode when the data bus is in the byte mode. In this mode the following ratios of BCLK to RCLK must be observed for the corresponding DMA to disk transfer rates.

Byte Mode

BCLK/RCLK Ratio	Max DMA Transfer Rate
< 1/1.6	Will not be able to keep up with disk rate,
> 1/1.6 but < 1/0.6	will get FDL. Can only transfer at the disk rate, therefore any bus sharing will result in depleting the FIFO, with no ability to refill it.
> 1/0.6	Can transfer at least at 2X the disk rate. Can easily refill the FIFO if depleted.

Word Mode

< 1/1	Can transfer only at the disk rate. Depleted FIFO cannot be refilled.
> 1/1	Can transfer at least at 2X the disk rate.

Lost BCLK Cycles in DMA Burst Mode

During DMA burst mode operation, LRQ or RRQ is deasserted for two BCLK cycles between bursts of local or remote DMA, i.e., When a remote burst is followed by another remote burst, an extra BCLK cycle occurs between t4 of the prior burst and the t1 of the subsequent burst. Likewise this is true for a local burst followed by another local burst, with the exception that here there is a possibility of two dummy BCLK cycles being inserted between t4 and t1. However, if a remote burst is followed by a local burst or vice versa, no dummy BCLK cycles are introduced.

18.0 Appendix

18.1 DDC REGISTERS, INDEX BY HEX ADDRESS

The following is a repeat of what can be found in the DDC INTERNAL REGISTERS Section. This listing is arranged numerically by hex address, and is provided as a quick reference. The section numbers provided indicate where the best description for the particular register can be located. For an explanation of the information contained in the WR and RD columns, refer to the key in the INTERNAL REGISTERS Section.

COLUMN KEY:

HA: *Hex Address* **#B**: *Number of bits* **WR**: *Write* **RD**: *Read* **SC**: *Section*

HA	REGISTER	#B	WR	RD	SC
00	Status Register (S)	8	NO	R	3.1
01	Error Register (E)	8	NO	R	3.1
02	ECC SR Out 0	8	NO	R	3.4
02	Polynomial Preset Byte 0 (PPB0)	8	D	NO	3.4
03	ECC SR Out 1	8	NO	R	3.4
03	Polynomial Preset Byte 1 (PPB1)	8	D	NO	3.4
04	ECC SR Out 2	8	NO	R	3.4
04	Polynomial Preset Byte 2 (PPB2)	8	D	NO	3.4
05	ECC SR Out 3	8	NO	R	3.4
05	Polynomial Preset Byte 3 (PPB3)	8	D	NO	3.4
06	ECC SR Out 4	8	NO	R	3.4
06	Polynomial Preset Byte 4 (PPB4)	8	D	NO	3.4
07	ECC SR Out 5	8	NO	R	3.4
07	Polynomial Preset Byte 5 (PPB5)	8	D	NO	3.4
08	Data Byte Count (0)	8	NO	R	3.4
08	Polynomial Tap Byte 0 (PTB0)	8	D	NO	3.4
09	Data Byte Count (1)	8	NO	R	3.4
09	Polynomial Tap Byte 1 (PTB1)	8	D	NO	3.4
0A	Polynomial Tap Byte 2 (PTB2)	8	D	NO	3.4
0B	Polynomial Tap Byte 3 (PTB3)	8	D	NO	3.4
0C	Polynomial Tap Byte 4 (PTB4)	8	D	NO	3.4
0D	Polynomial Tap Byte 5 (PTB5)	8	D	NO	3.4
0E	ECC CONTROL (EC)	8	D	NO	3.4
0F	Header Byte Count (HBC)/Interlock	3	F	R	3.1
10	Drive Command Register (DC)	8	C	NO	3.1
11	Operation Command Register (OC)	8	C	NO	3.1
12	Sector Counter (SC)	8	C	R	3.1
13	Number of Sector Operations Counter (NSO)	8	C	R	3.1
14	Header Byte 0 Pattern	8	C	R	3.3
15	Header Byte 1 Pattern	8	C	R	3.3
16	Header Byte 2 Pattern	8	C	R	3.3
17	Header Byte 3 Pattern	8	C	R	3.3
18	Header Byte 4 Pattern	8	C	R	3.3
19	Header Byte 5 Pattern	8	C	R	3.3
1A	Remote Data Byte Byte Count (L)	8	C	R	3.2
1B	Remote Data Byte Byte Count (H)	8	C	R	3.2
1C	DMA Address Byte 0	8	C	R	3.2

HA	REGISTER	#B	WR	RD	SC
1D	DMA Address Byte 1	8	C	R	3.2
1E	DMA Address Byte 2	8	C	R	3.2
1F	DMA Address Byte 3	8	C	R	3.2
20	Data Postamble Byte Count	5	D	R	3.3
21	ID Preamble Byte Count	5	C	R	3.3
22	ID Sync #1 (AM) Byte Count	5	D	R	3.3
23	ID Sync #Byte 2 Count	5	D	R	3.3
24	Header Byte 0 Control	5	D	R	3.3
25	Header Byte 1 Control	5	D	R	3.3
26	Header Byte 2 Control	5	D	R	3.3
27	Header Byte 3 Control	5	D	R	3.3
28	Header Byte 4 Control	5	D	R	3.3
29	Header Byte 5 Control	5	D	R	3.3
2A	Data External ECC Byte Count	5	D	R	3.3
2B	ID External ECC Byte Count	5	D	R	3.3
2C	ID Postamble Byte Count	5	D	R	3.3
2D	Data Preamble Byte Count	5	D	R	3.3
2E	Data Sync #1 (AM) Byte Count	5	D	R	3.3
2F	Data Sync #2 Byte Count	5	D	R	3.3
30	Data Postamble Pattern	8	D	R	3.3
31	ID Preamble Pattern	8	D	R	3.3
32	ID Sync #1 (AM) Pattern	8	D	R	3.3
33	ID Sync #2 Pattern	8	D	R	3.3
34	Gap Byte Count	8	F	R	3.3
35	Disk Format Register (DF)	8	D	NO	3.1
36	Header Diagnostic Readback (HDR)	8	NO	R	3.1
36	Local Transfer Register	8	I	NO	3.2
37	DMA Sector Counter (DSC)	8	NO	R	3.2
37	Remote Transfer Register	8	I	NO	3.2
38	Sector Byte Count 0	8	D	R	3.2
39	Sector Byte Count 1	8	D	R	3.2
3A	Gap Pattern	8	F	R	3.3
3B	Data Format Pattern	8	F	R	3.3
3C	ID Postamble Pattern	8	D	R	3.3
3D	Data Preamble Pattern	8	D	R	3.3
3E	Data Sync #1 (AM) Pattern	8	D	R	3.3
3F	Data Sync #2 Pattern	8	D	R	3.3

18.0 Appendix (Continued)

18.2 ALPHABETICAL MNEMONIC GLOSSARY AND INDEX

Listed on the following pages are the majority of the abbreviations used within this data sheet as mnemonics to describe portions or functions of the DDC. The section numbers referenced indicate where the terms are first defined. Mnemonics from the specifications section are not included here.

MNEMONIC	DESCRIPTION	SECTION
AD0–7	Address/Data 0–7 (pins 41–48)	2.0
AD8–15	Address/Data 8–15 (pins 1–8)	2.0
ADS0	Address Strobe 0 (pin 9)	2.0
ADS1	Address Strobe 1 (attached to RRQ, pin 37)	2.0
AME	Address Mark Enable (attached to LPRE, pin 13)	2.0
AMF	Address Mark Found (attached to EPRE, pin 16)	2.0
BCLK	Bus Clock (pin 40)	2.0
CCA	Correction Cycle Active (bit in Status register)	3.1
CF	Correction Failed (bit in Error register)	3.1
$\overline{CS}$	Chip Select (pin 28)	2.0
CS0–3	Correction Span Selection (bits in EC register)	3.4
DC	Drive Command register	3.1
DNE	Data Non-Encapsulation (bit in EC register)	3.4
DF	Disk Format register	3.2
DFE	Data Field Error (bit in Error register)	3.1
D01, 2	Data Operation bits (command in DC register)	3.1
DSC	DMA Sector Counter	3.2
E	Error register	3.1
EC	ECC Control register	3.4
ED	Error Detected (bit in Status register)	3.1
EEF	External ECC Field (pin 26)	2.0
EEW	Enable External Wait (bit in RT register)	3.2
EHF	Enable HFASM Function (bit in HC0–5 registers)	3.3
EHI	Enable Header Interrupts (command in OC register)	3.1
FTF	FIFO Table Format (bit in DF register)	3.1
HBA	Header Byte Active (bit in HC0–5 registers)	3.3
HBC	Header Byte Count register	3.1
HC0–5	Header Byte 0–5 Control registers	3.3
HDR	Header Diagnostic Readback register	3.1
HNE	Header Non-Encapsulation (bit in EC register)	3.4
HF	Header Fault (bit in Status register)	3.1
HFASM	Header Failed Although Sector number Matched (bit in Error register)	2.0
HMC	Header Match Completed (bit in Status register)	3.1
H01, 2	Header Operation bits (command in DC register)	3.1
HSS	Hard or Soft Sectored (bit in DF register)	3.1
ID1, 2	Internal Data Appendage (bits in DF register)	3.1
IDI	Invert Data In (bit in EC register)	3.4
IH1, 2	Internal Header Appendage (bits in DF register)	3.1
$\overline{INT}$	Interrupt (pin 29)	2.0
LA	Long Address (bit in LT register)	3.2
LACK	Local DMA Acknowledge (pin 39)	2.0
LBL1, 2	Local Burst Length (bits in LT register)	3.2
LCB	Local Command Busy (bit in Status register)	3.1
LI	Late Interlock (bit in Error register)	3.1
LPRE	Late Precompensation (attached to AME, pin 13)	2.0
LRQ	Local DMA Request (pin 36)	2.0
LRQ	Local Request (bit in Status register)	3.1
LSRW	Local Slow Read/Write (bit in LT register)	3.2
LT	Local Transfer register	3.2
LTEB	Local Transfer Exact Burst (bit in LT register)	3.2
LWDT	Local Word Data Transfer (bit in LT register)	3.2
MFM	MFM Encode (bit in DF register)	3.1
MSO	Multi-Sector Operation (command in DC register)	3.1
NCP	Not Compare (bit in HC0–5 registers)	3.3
NDC	Next Disk Command (bit in Status register)	3.1
NDS	No Data Synch (bit in Error register)	3.1
NSO	Number of Sector Operations counter	3.1
OC	Operation Command register	3.1
PPB0–5	Polynomial Preset Byte 0–5	3.4
PTB0–5	Polynomial Tap Byte 0–5	3.4
RACK	Remote DMA Acknowledge (pin 38)	2.0
RBL1, 2	Remote Burst Length (bits in RT register)	3.2
RBO	Reverse Byte Order (bit in LT register)	3.2
RCB	Remote Command Busy (bit in Status register)	3.1
RCLK	Read Clock (pin 25)	2.0
$\overline{RD}$	Read (pin 11)	2.0
RDATA	Read Data (pin 15)	2.0
RED	Re-Enable DDC (command in DC register)	3.1
RES	Reset DDC (bit OC register)	3.2
RGATE	Read Gate (pin 19)	2.0
RRQ	Remote Request (attached to ADS1, pin 37)	2.0
RS0–5	Register Select 0–5 (pins 30–35)	2.0
RSRW	Remote Slow Read/Write (bit in RT register)	3.2
RT	Remote Transfer register	3.2
RTEB	Remote Transfer Exact Burst (bit in RT register)	3.2
RWDT	Remote Word Data Transfer (bit in RT register)	3.2
S	Status register	3.1
SAIS	Start At Index or Sector (command in DC register)	3.1
SAM	Start at Address Mark (bit in DF register)	3.1
SC	Sector Counter	3.1
SCC	Start Correction Cycle (command in OC register)	3.1
SDV	Serial Data Valid (pin 27)	2.0
SLD	Select Local DMA (bit in LT register)	3.2
SNF	Sector Not Found (bit in Error register)	3.1
SO	Sector Overrun (bit in Error register)	3.1
SRD	Select Remote DMA (bit in RT register)	3.2
SRI	Start Remote Input (command in OC register)	3.1
SRO	Start Remote Output (command in OC register)	3.1
SSC	Substitute Sector Counter (bit in HC0–5 registers)	3.3
TM	Tracking Mode (bit in RT register)	3.2
WCLK	Write Clock (pin 21)	2.0
WDATA	Write Data (pin 18)	2.0
WGATE	Write Gate (pin 20)	2.0
$\overline{WR}$	Write (pin 10)	2.0

Designing an ESDI (Enhanced Small Device Interface) Disk Controller Subsystem with National's DP8466A (Disk Data Controller)

National Semiconductor
Application Note 500

1.0 INTRODUCTION

The ESDI (Enhanced Small Disk Interface) is designed to handle a variety of 5¼" Winchester disk, tape and optical drives. It opens the door to higher performance system designs by incorporating more intelligence onto the drives, and by allowing higher data transfer rates—10 Mbits/s to 24 Mbits/s. This is achieved by incorporating data separation, data encoding and decoding in the drive itself and the smarter interface protocol allows dissemination of more information between the drive and the controller. Thus by removing the restrictions placed by ST506 on 5¼" hard disk transfer rates, the ESDI interface clears the way for higher recording densities and ultimately, higher storage capacities in the 5¼" form factor, up to 700 Mbytes and beyond.

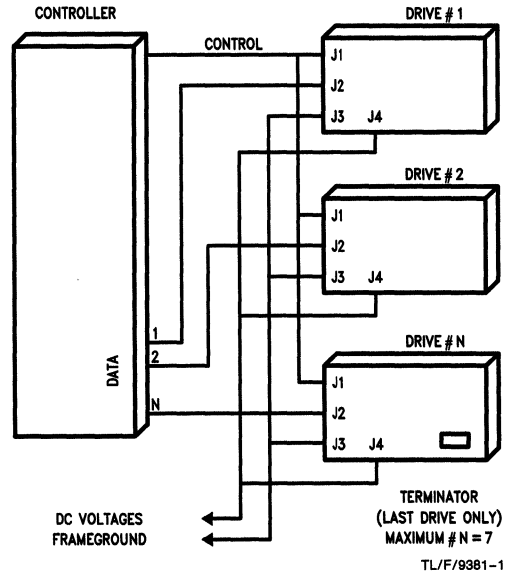
National Semiconductor's DP8466A Disk Data Controller integrates a number of functions originally supported by discrete logic in conventional disk controller designs. This results in a decrease in complexity and parts count in a disk controller design. The DP8466A is a data path controller and hence can support the various disk interfaces viz. ST506, ESDI, SMD, etc. By its versatility and programmability, it greatly simplifies the task of designing a disk controller. There are basically two types of ESDI drives viz. hard sectored ESDI drives and soft sectored ESDI drives. This application note discusses the ESDI interface and the various steps involved in designing an ESDI disk controller with the DP8466A. The emphasis is predominantly on the disk side as that is of utmost relevance with respect to intefacing the DP8466A.

2.0 ENHANCED SMALL DEVICE INTERFACE (ESDI)

The ESDI consists of a control cable and a data cable. The control cable allows for a daisy chain connection of up to seven drives with only the last drive being terminated. The data cable must be attached in a radial fashion. *Figure 1* shows a typical connection in a multiple drive system. All control lines are digital in nature (open collector TTL) and either provide signals to the drive (input) or signals to the controller (output). The data transfer signals are differential in nature and provide data either to, (write) or from, (read), the drive.

2.1 Control Cable

The control cable definition is shown in *Figure 2*. It basically consists of some input lines and some output lines. The control input signals are of two kinds; those to be multiplexed in a multiple drive system and those intended to do the multiplexing. The control input signals to the drive to be multiplexed are WRITE GATE, READ GATE, HEAD SELECT 2^0, 2^1, 2^2, 2^3, TRANSFER REQUEST and COMMAND DATA. The signals to do the multiplexing are the DRIVE SELECT 1,2,3. ADDRESS MARK ENABLE (AME) is a control input on the radial data cable and is not multiplexed. The drive select lines accept a binary input combination decoded internally to allow 1–7 drives. Decode 000 is a no select condition. The four Head select lines allow selection of each individual read/write head in a binary coded sequence. Heads are numbered 0 thru 15. Write Gate and Read Gate are control signals which initiate writing and reading of data respectively at the disk.



TL/F/9381–1

FIGURE 1. Typical Connection in a Multiple Drive System

Control Cable		Data Cable	
Pin No.	Signal	Pin No.	Signal
2	Head Select 2^2	1	Drive Selected
4	Head Select 2^1	2	Sector/AMF
6	Write Gate	3	Command Complete
8	Config/Status	4	Address Mark Enable
10	Transfer ACK	7,8	Write Clock ±
12	Attention	10, 11	Read/Reference CLK ±
14	Head Select 2^0	13,14	NRZ Write Data ±
16	Sector/AMF	17,18	NRZ Read Data ±
18	Head Select 2^3	20	Index
20	Index	Ground = 5,6,9,12,15,16,19	
22	Ready		
24	Transfer Request		
26	Drive Select 1		
28	Drive Select 2		
30	Drive Select 3		
32	Read Gate		
34	Command Data		
Ground = All Odd # Pins			

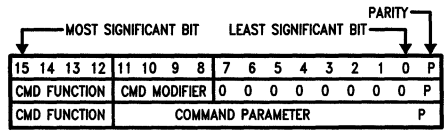
FIGURE 2. ESDI Cable Assignments

The ESDI is an intelligent interface and provides for certain commands, to do certain specific functions, thereby freeing the controller from a number of mundane tasks, and also from being tied to a particular drive. These commands are presented to the drive as 16 information bits of serial data, plus a parity bit. The transfer of this data is controlled by the handshake protocol with the TRANSFER REQUEST and TRANSFER ACKNOWLEDGE signals. Upon receipt of this serial data, the drive will perform the required function as specified by the bit configuration. *Figure 3* lists the various commands supported by ESDI.

The Address Mark Enable (AME) signal behaves differently for soft and hard sectored drives. In soft sectored drives this signal, when active with Write Gate, writes an Address Mark on the disk. When AME is asserted without Write Gate or Read Gate, it causes a search for Address Marks. The address mark written is usually a gap of no flux transitions, exactly 24 bits long. In case of hard sectored drives, the AME does not cause an Address Mark to be written on the media. The trailing edge of AME with Write Gate asserted, initiates the writing of the ID PLO sync field.

The output control signals are driven with an open collector output stage capable of sinking a maximum current of 48 mA. They consist of the DRIVE SELECTED, READY, ATTENTION, INDEX, SECTOR, TRANSFER ACKNOWLEDGE, and CONFIGURATION & STATUS. COMMAND COMPLETE is a control output which allows the host to monitor the drive's command completion status. The ATTENTION line is activated whenever there is an erroneous condition at the drive. In response to the Request Configuration and Request Status commands, the drive provides some status information which is sent to the controller in a

Command Data Word Structure



TL/F/9381–2

Command Data Definition

Command Function Bit 15	14	13	12	Command Function Definition	Command Modifier Applicable Bits 11–8	Command Parameter Applicable Bits 11–0	Status Data Returned to Host
0	0	0	0	Seek	No	Yes	No
0	0	0	1	Recalibrate	No	No	No
0	0	1	0	Request Status	Yes	No	Yes
0	0	1	1	Request Config	Yes	No	Yes
0	1	0	0	Select Head Group*	No	Yes	No
0	1	0	1	Control	Yes	No	No
0	1	1	0	Data Strb Offset*	Yes	No	No
0	1	1	1	Track Offset	Yes	No	No
1	0	0	0	Init Diagnostics*	No	Yes	No
1	0	0	1	Set Bytes/Sector*	No	Yes	No
1	0	1	0	Reserved			
1	0	1	1	Reserved			
1	1	0	0	Reserved			
1	1	0	1	Reserved			
1	1	1	0	Set Config*	No	Yes	No
1	1	1	1	Reserved			
*Optional Commands					All Unused Bits Set to Zero		

FIGURE 3. ESDI Command Structure

serial manner over the CONFIGURATION & STATUS line using a handshake protocol between TRANSFER REQUEST and TRANSFER ACKNOWLEDGE.

Index and sector are interface signals from the drive which indicate the start of a track and sector respectively. In case of a soft sectored drive there are no sector pulses, but if an address mark is found then it signals the end of an address mark, indicated by the AMF signal from the drive.

2.2 Data Cable

All lines associated with the transfer of data between the drive and the host system are differential in nature and may not be multiplexed. Four pairs of balanced signals are used for the transfer of data and clock signals: NRZ Write Data, NRZ Read Data, Write Clock, and Read/Reference Clock. Differential drivers and receivers are needed to interface the controller to the drive, like National's DS8922A/DS8923A. Connection details are shown in *Figure 13*. The NRZ Write Data is clocked by the Write Clock signal while the NRZ Read Data is clocked by the Read Clock signal. The Reference Clock signal from the drive will determine the data transfer rate. The transitions from Reference Clock to Read Clock must be performed without glitches, i.e. the clock should not violate the minimum allowable specifications of the controller chip.

2.3 ESDI Format Rules

The ESDI essentially supports a fixed sector implementation, (Drive hard sectored) and a soft sectored implementation, (Drive soft sectored). The record format on the disk is under control of the controller, however, the ESDI standard recommends a certain format structure which must be implemented. In a hard sectored drive the index pulse signifies the start of a track, while the sector pulse signifies the start of a sector. *Figure 4* shows a fixed sector format and associated timings. In a soft sectored drive, the index signifies the beginning of a track while the beginning of each sector is defined by an Address Mark, followed by the ID field which contains the header information. The AME/AMF handshake is utilized to detect these address marks. *Figure 5* shows the soft sectored format and associated timings. In a hard sectored drive, the beginning of the ID PLO sync field is specified by the trailing edge of the AME, when Write Gate is active. From *Figures 4, 5* and *6* it can be seen that there are some minor differences between the DDC's format and the ESDI recommended format. The ESDI recommended format supports a post index/sector gap field and a write splice field between the ID and DATA fields, which is not directly supported by the DDC. Also Write Gate needs to be optionally deasserted in the write splice area between the ID and DATA segments as shown in *Figures 4* and *5*. These shortcomings can however be overcome through a combination of hardware and software considerations as discussed in the following sections.

3.0 CONTROLLER DESIGN—DISK SIDE

Perhaps of greatest significance to system designers is the fact that the ESDI drives provide the data separation function internally. The performance benefits attained by putting the data separator on the drive more than offset the cost in terms of system efficiency and reliability. Data is transferred over the interface in NRZ format. This results in the use of high density encoding schemes to be implemented in the drive, like 2,7 RLL, etc. These factors greatly simplify the task of designing a controller for ESDI drives. This task is even more simplified with the availability of VLSI disk controller IC's like the DP8466A, which integrates numerous

ESDI Hard Sectored Format

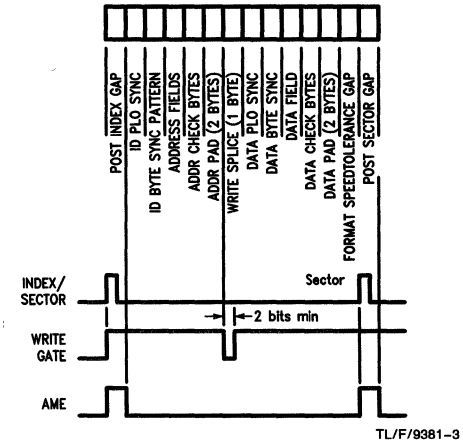


FIGURE 4. ESDI Fixed Sector Format and Relevant Timing

functions in the disk path. The disk side involves the interface of the DDC to a disk interface, like the ESDI. This is made up of two main paths—the data path and the control path. The control path is responsible for the disk related functions like sending commands, etc (as discussed in Section 2.1), while the data path is responsible for the data transfer, (refer to Section 2.2). The DP8466A is a disk data path controller which does not involve itself with the slower tasks of the control path. It features full format programmability, fully programmable ECC, 16-bit dual channel DMA and a 32-byte FIFO. Data is transferred from the FIFO in selectable bursts, which minimizes bus occupancy and can thereby accommodate some degree of latency. For more details refer to AN-413. As mentioned in the previous section, the disk formats suggested by the ESDI standard produce some compatibility problems with the DP8466A, mostly in the area of control line timing with the drive. The techniques to handle them with minimum overhead is discussed below for the two types of ESDI drive systems.

ESDI Soft Sectored Format

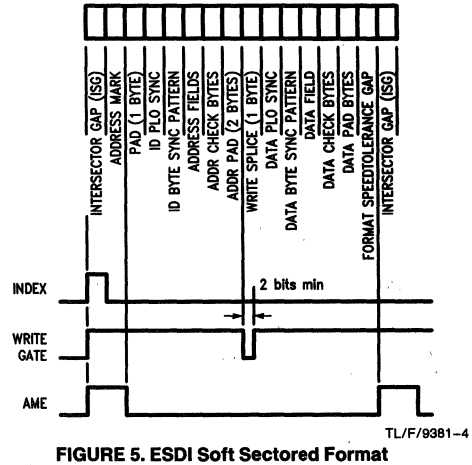


FIGURE 5. ESDI Soft Sectored Format and Relevant Timing

3.1 ESDI Hard Sectored Drive

For an ESDI drive which is hard sectored, the specification calls for an Inter-Sector Gap (ISG) which is to precede and follow the index/sector pulse,also referred to as the post index/sector pulse. This gap is needed to provide the drive with an area for the embedded servo (if used) and also gives the controller time to assert read gate. While formatting the drive, the end of the ISG (post index/sector gap), is indicated by the trailing edge of AME. This is needed by the drive to indicate the beginning of the PLO sync (preamble) field, necessary when the drive encodes the PLO sync field with a non-standard preamble pattern, e.g. as in 2,7 encoding with 3T or 4T preambles. The DDC generates the AME signal with the necessary timing during a format operation. To do so the DDC must be programmed to be in the hard sectored mode and have the start with address mark bit enabled in the Disk Format Register, (HSS and SAM bits in the DF register). *Figure 6* illustrates the manner in which the DDC format parameter registers need to be manipulated for a format operation and a read/write operation. It should be noted that the ISG field is implemented by the DDC's ID preamble field while the PLO sync is implemented by the ID sync #1 field. This feature is needed only during formatting, and hence should be disabled at other times. On the other hand when a disk read/write operation is attempted, the header field is compared (or read) usually. The DDC asserts read gate just after the index or sector pulse, which initiates the data separator to start locking the PLL to the preamble. Since there is the ISG field after the index/sector pulse, the read gate to the drive needs to be delayed until after the ISG. This can be achieved in two ways. One technique is to delay index and sector to the DDC by the length of the ISG, so that the DDC would be asserting Read Gate in the preamble area. *Figure 6* shows the format and control signals for this situation. This is the technique adopted for the combined solution proposed in section 3.3 to implement both hard and soft sectored ESDI.

Alternate Technique

The other technique for implementing the post index sector gap is explained below and could be used when only ESDI hard sectored drives are under consideration. Essentially when formatting the drive, the DDC is set up to use ID preamble field as the ISG, the ID sync #1 as the PLO synch and the ID sync #2 as the sync byte. Hence when reading or writing, the DDC is set up to have ID preamble as the PLO sync, ID sync #1 is skipped and ID sync #2 as sync byte. External hardware is used to delay the read gate at the beginning of every sector, by the length of the ISG so that it gets asserted over the preamble on the drive. It should however be noted that the read gate needs to be delayed only at the beginning of the sector and not in the middle of the sector before the data field. *Figure 7* outlines the format manipulating and control signals behaviour.

3.2 ESDI Soft Sectored Drive

The soft sectored specification of ESDI provides two major stumbling blocks for the DDC. First there is a need for providing the handshaking for the AME and AMF lines. The controller needs to raise AME when it wishes to be notified of the start of a sector. The controller asserts Read Gate on detecting AMF, generated by the drive on finding an Address Mark of 24 bits. After seeing the drive assert AMF, the controller removes AME, finishing the handshake. Since the DCC does not provide the AME/AMF handshake, external hardware is required to do so when the DDC is not formatting. As the Address Mark is at the beginning of every sector, the AMF signal is seen by the DDC as a sector pulse. The DDC is programmed to believe that it is looking at a hard sectored drive, and thereby the soft sectored ESDI can be handled in this pseudo fashion.

The second stumbling block is that the soft sectored ESDI drives generates an AMF signal for each sector on the disk, including the first. Because the DDC starts its operation on either index or sector, (AMF instead of sector in this case), it will believe that there is a sector between index and the first AMF. External circuitry is needed to eliminate the first AMF pulse and replace it with a delayed version of index.

One other problem with soft sectored ESDI, is formatting the drive. In these drives there is usally a ISG following the index pulse. In addition to this is the Address Mark field, which fits between the ISG and the PLO sync fields. The DDC has only two fields (ID preamble and ID sync #1) before the sync byte, with which these three fields have to be created. One way to implement it with external circuitry is to use the ID preamble field of DDC for writing the ISG and AM. The external circuitry delays the AME from the DDC by the length of the ISG. The address mark (AM) field will hence be the length of the ID preamble less the delay and the PLO sync is implemented using the ID sync #1 field. *Figure 8* shows the format manipulation required and the control signals behaviour.

3.3 Combined Solution

A combined solution to the above problems, (for a system supporting both soft and hard sectored ESDI), can be provided by a single PAL® device and something to provide a delay (possibly another PAL device). The interface solutions can be grouped into two main areas, implemented as state machines.

1. Index and sector pulse generation to the DDC
2. AME/AMF handshaking between the DDC and the drive.

3.3.1 The Index/Sector Machine

There are essentially 4 types of drive operations which are encoded using the two PAL inputs (soft/hard/ and format). These are outlined below:

1. Hard sectored drive, read or write data operation
   PAL Inputs: soft/hard/ = 0 and format = 0

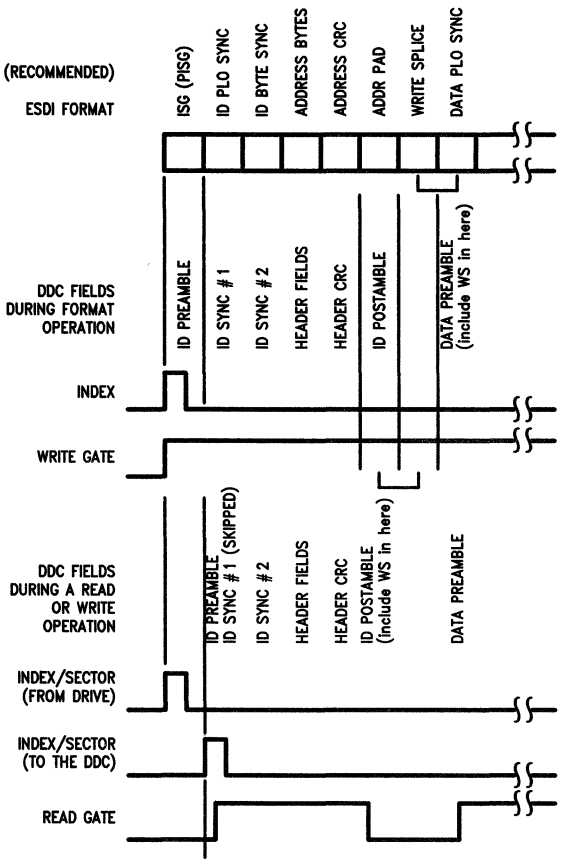
In this case the index/sector machine essentially delays both the index and sector pulse from the drive by the length of the ISG (post index/sector gap) and then present it to the DDC. *Figure 10* shows the state diagram and truth table for this part of the index/sector machine. Three state variables are used: F1 and F2 to indicate the state, and G1 to signal if an index pulse has been received from the drive. *Figure 9* shows the state diagram, truth tables and timing relationships for this part of the machine.

2. Hard sectored drive format operation
   PAL inputs: soft/hard/ = 0 and format = 1

In this case the index/sector machine essentially lets the index and sector pulses from the drive flow through to the DDC.

3. Soft sectored drive, read or write operation
   PAL inputs: soft/hard/ = 1 and format = 0

In this case the index/sector machine essentially follows the AMF from the drive, translating it as the sector input to the DDC; however if an index pulse occurs from the drive then it waits for an AMF from the drive and then generates an index and no AMF/sector pulse to the DDC. This is the scenario which happens in the sector just after the index



TL/F/9381–5

FIGURE 6. Programming the DDC Format Parameters in the Case of an ESDI Hard Sectored Implementation

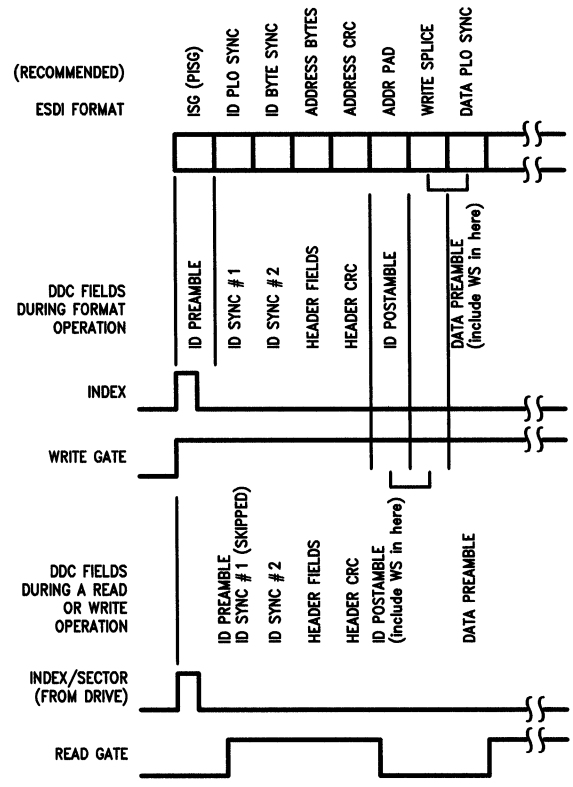
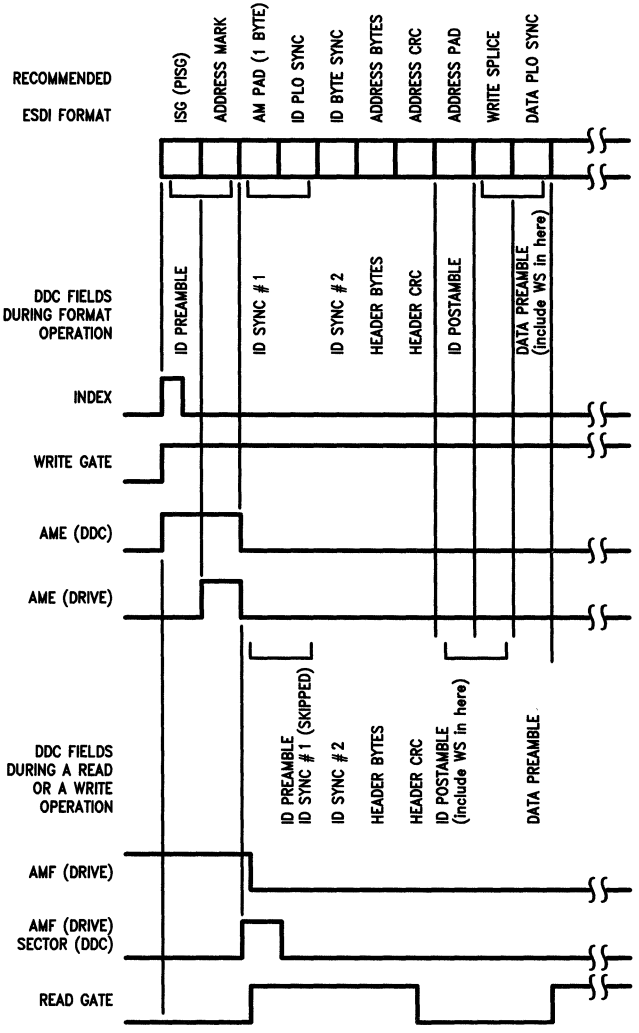


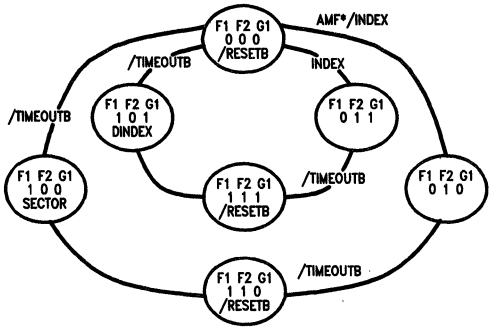
FIGURE 7. Programming the DDC Format Parameters in the Case of an ESDI Hard Sectored Implementation (Alternative Technique)



TL/F/9381–7

FIGURE 8. Programming the DDC Format Parameters in the Case of an ESDI Soft Sectored Implementation

State Machine

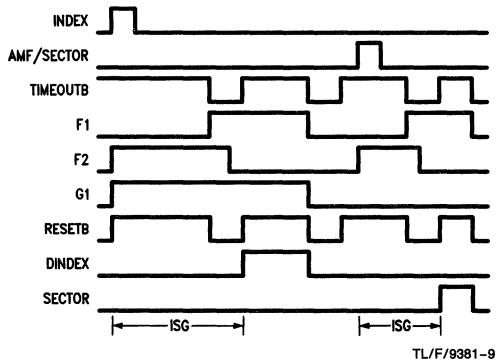


TL/F/9381–8

Truth Table

F1	F2	G1	Index	AMF	TIMEOUTB	F1	F2	G1	RESETB	DINDEX	Sector
Present State			Inputs			Next State			Outputs		
0	0	X	0	0	X	0	0	0	0	0	0
0	0	0	1	X	X	0	1	1	1	0	0
0	0	0	0	1	X	0	1	0	1	0	0
0	1	0	X	X	1	0	1	0	1	0	0
0	1	1	X	X	1	0	1	1	1	0	0
0	1	0	X	X	0	1	1	0	0	0	0
0	1	1	X	X	0	1	1	1	0	0	0
1	1	0	X	X	X	1	0	0	1	0	0
1	1	1	X	X	X	1	0	1	1	1	0
1	0	0	X	X	1	1	0	0	1	0	1
1	0	1	X	X	1	1	0	1	1	1	0
1	0	X	X	X	0	0	0	0	1	0	0

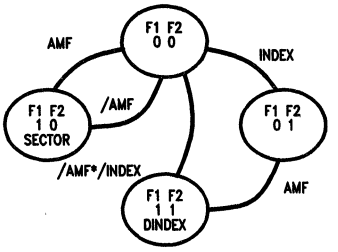
Index/Sector Machine Timing Diagrams



TL/F/9381–9

FIGURE 9. Index/Sector Machine Hard Sectored ESDI (Non-Format) State Diagram, Truth Tables and Timing

State Diagram



TL/F/9381–10

Truth Table

F1	F2	Index	AMF	F1	F2	DINDEX	Sector
Present State		Inputs		Next State		Outputs	
0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	0
0	1	X	0	0	1	0	0
0	1	X	1	1	1	1	0
1	1	1	X	1	1	1	0
1	1	X	1	1	1	1	0
1	1	0	0	0	0	0	0
0	0	X	1	1	0	0	1
1	0	X	1	1	0	0	1
1	0	X	0	0	0	0	0

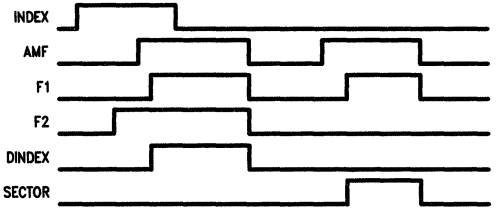
hole. *Figure 10* shows the state diagram, truth table and timing relationships for this section of the Index/Sector machine. It uses two state variables, F1 and F2.

4. Soft sectored drive, format operation
   PAL inputs: soft/hard/ = 1 and format = 1

In this case the index/sector machine essentially follows the index pulse from the drive and passes it on to the DDC while it forces the sector input to the DDC to be inactive (low).

3.3.2 The Address Mark Machine

The Address Mark Machine consists of a pair of multiplexers which feed the AME input to the drive and translate the AMF from the drive as the sector pulse input to the DDC. Once again as before there are 4 types of drive operations, encoded using the two PAL inputs soft/hard/ and format.



TL/F/9381–11

FIGURE 10. Index/Sector Machine Soft Sectored ESDI (Non-Format) State Diagram, Truth Table and Timing

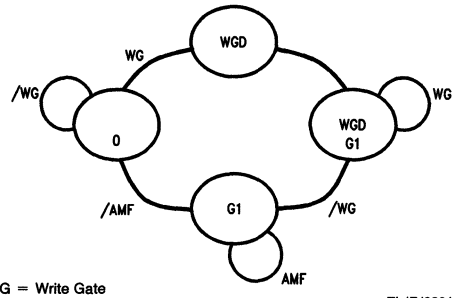
The working of the address mark machine in these cases is outlined below.

1. Hard sectored drive, Read and Write operation
   PAL inputs: soft/hard/ = O and format = O.

The AME to the drive is kept inactive (low).

2. Hard sectored drive, format operation
   PAL inputs: soft/hard/ = O and format = 1

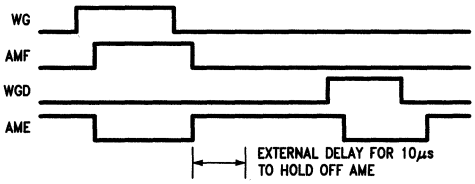
State Diagram



WG = Write Gate
WGD = Write Gate Delayed by One Clock

TL/F/9381–12

Timing Diagrams—Soft Sectored (Non-Format) Operation



TL/F/9381–13

In this case the AME to the drive follows the DDC generated AME.

3. Soft sectored drive, read or write operation
   PAL inputs: soft/hard/ = 1 and format = O

In this case the PAL generates AME and handshakes with the AMF from the drive, translating it to the DDC as the sector input.

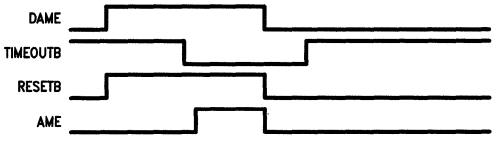
4. Soft sectored drive format operation
   PAL inputs: soft/hard/ = 1 and format = 1

In this case the leading edge of AME generated by the DDC is delayed by the length of the ISG (post index/sector gap) and presented to the drive.

Note: The trailing edge is not delayed.

Figure 11 shows the state diagram and timing relationships for the Address Mark Machine. Given below are the PAL equations for this control PAL implemented in a 16R4. These include some simplifications from the above information. In particular the hard sector non-format equations for F1, F2 and G1 can drop the use of the term /FORMAT, since during the format operation it is quite acceptable to have these output behaving as for non-format since the index to the DDC follows the index from the drive. The equations are written in PLAN format.

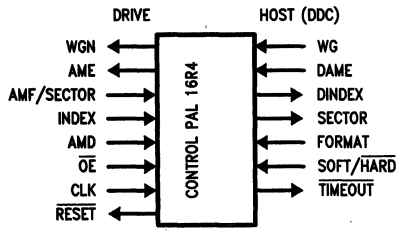
Timing Diagrams—Soft Sectored Format Operation



TL/F/9381–14

FIGURE 11. Address Mark Machine Details

Control PAL Pin Description



TL/F/9381–15

Control PAL Equations (Plan Format)
Version 1.3 Dated: 05/12/86

```
/wgd := /wg

gl   := (soft*wgd*wg)+(soft*wgd*gl)+
        (soft*/wgd*/wg*amf)+
        (/soft*/gl*index)+(/soft*gl*/fl)+
        (/soft*gl*f2)+(soft*gl*/timeout)

fl   := (soft*/format*f2*amf)+
        (soft*/format*fl*ff2*index)+
        (soft*/format*fl*/f2*amf)+
        (soft*/format*/fl*/f2*/index*amf)+
        (/soft*/fl*f2*timeout)+(/soft*fl*f2)+
        (/soft*fl*/f2*/timeout)

f2   := (soft*/format*/fl*/f2*index)+
        (soft*/format*/fl*f2)+
        (soft*/format*fl*f2*amf)+
        (soft*/format*fl*f2*index)+
        (/soft*/fl*/f2*index)+
        (/soft*/fl*/f2*amf)+
        (soft*/fl*f2)

if (vcc) /ame   = (soft*/format*gl)+
                  (soft*/format*wg)+
                  (soft*format*/dame)+
                  (soft*format*/timeout)+
                  (/soft*/format)+
                  (/soft*format*/dame) + (amd)

if (vcc) reset  = (soft*/format)+
                  (soft*format*/dame)+
                  (/soft*/format*/fl*/f2)+
                  (/soft*/format*fl*f2)+
                  (/soft*format)

if (vcc) /dindex =(soft*/format*/fl)+
                  (soft*/format*/f2)+
                  (format*/index)+
                  (/soft*/format*/fl)+
                  (/soft*/format*f2)+
                  (/soft*/format*/gl)+
                  (/soft*/format*timeout)

if (vcc)/sector  =(/format*/fl)+
                  (soft*/format*f2)+
                  (soft*format)+
                  (/soft*/format*f2)+
                  (/soft*/format*gl)+
                  (/soft*/format*timeout)+
                  (/soft*format*/amf)
```

3.3.3 Timer PAL

This is essentially the PAL used to generate the delay. When the RESET input to the PAL is low it does not count. It starts counting and when the desired time delay is reached it produces an active low output ($\overline{\text{TIMEOUT}}$). The value of the delay is variable from 1 to 32 byte times. The counter is clocked by the disk's RCLK which is divided by 8 to provide a byte-rate clock to a five-stage counter. When $\overline{\text{RESET}}$ is low (active), the inverse of the values GA–GE are loaded into the counter, and one byte-time after the counter reaches 11111 the TIMEOUT output goes low and counting stops. The value on GA–GE (GA is least significant bit) should be set to (number of bytes delay—1). Given below are the equations for this PAL 20X10). Note that PLAN requres the use of a dummy term to complete the OR function before the XOR. Hence for example, in the equation for F1 the term "F1*/F1" is put in, which always equals to 0.

Timer PAL Equations (Plan Format)
Version 1.3 Dated: 05/12/86

```
f3 := (/timeout*/reset)+
      (f3*/f3) :+: (f3*/reset)

f2 := (/timeout*/reset*f3)+
      (f2*/f2) :+: (f2*/reset)

fl := (/timeout*/reset*f2*f3)+
      (fl*/fl) :+: (fl*/reset)

qa := (/reset*f3*f2*fl)+
      (reset*ga) :+: (qa*/reset)

qb := (/reset*f3*f2*fl*/qa)+
      (reset*gb) :+: (qb*/reset)

qc := (/reset*f3*f2*fl*/qa*/qb)+
      (reset*gc) :+: (qc*/reset)

qd := (/reset*f3*f2*fl*/qa*/qb*/qc)+
      (reset*gd) :+: (qd*/reset)

timeout :=
   (/reset*f3*f2*fl*/qa*/qb*/qc*/qd*/qe)+
   (timeout*/timeout) :+: (timeout*/reset)
```

3.3.4 Some Other Timing Considerations

The ESDI specification imposes some additional timing restrictions which have to be accommodated for with external logic. These are outlined below:

1. AME to the drive cannot be asserted till at least 10 μs, after deassertion of Write Gate to the drive.
2. In the PAL solution discussed above, when the drive drops AMF at the end of the handshake, the PAL reasserts AME. The MAXTOR doesn't seem to like that and hence the AME to the drive must be held off for at least 8 μs, from the trailing edge of AMF.
3. On a similar token, the MAXTOR drive doesn't like AME to be active when Read Gate is asserted, (it usually activates ATTENTION on the drive). Hence AME needs to be held off at least 8 μs from trailing edge of Read Gate. This is accomplished using a mono shot, which generates a disable signal (Address Mark Disable, AMD), for 10 μs and while this is active the control PAL disables AME to the drive. Also since we are presenting the drive like a hard sectored one to the DDC, we could safely assume that AME need not be active with Read Gate. This is accomplished by gating the AME to the drive with the read gate. Hence if read gate is active, AME gets disabled to the drive. These could be incorporated within the control PAL if desired.

3.4 Handling the Optional ESDI Format Specifications

The ESDI has certain optional specifications which are not directly supported by the DDC. These are discussed below with explanations of the way they can be implemented.

3.4.1 Optional Deassertion of Write Gate between the ID and DATA Fields in a Format Operation

This option is not supported by the DDC, as once its starts the format operation, the Write Gate remains asserted for the entire track. The purpose of the deassertion of Write Gate between the ID and Data fields is usually to indicate to the 2,7 RLL encoder, the start of the data preamble. This enables the encoder to substitute the 3T or 4T preamble pattern. This feature could be implemented in two ways. Using external logic, the trailing edge of the SDV (Serial Data Valid) is used to gate the logic, count until the header postamble has been written and then force the Write Gate low for the required two bit times. This problem could also be circumvented in software by incorporating a two pass format. The first pass involves a regular format operation which will write the headers for all the sectors, but since there will be no de-assertion of Write Gate before the data fields, the proper data preamble will not be written. In the second pass a compare header-write data operation is done, where the Write Gate edge is used to initiate the drive generated preamble. It should be noted (as also pointed out in the ESDI specification), that this is necessary only if a read will be attempted after a format.

3.4.2 Handling the Write Splice Field between the ID and DATA Fields in the ESDI Format

The ESDI format specification recommends a 2-byte header postamble and a 1-byte write splice. The DDC does not have a separate field to implement the write splice. It is accomplished by software manipulation as follows. The format is programmed to have a 2-byte header postamble and a data preamble which is one byte longer than the desired length. This byte is taken as the write splice (a floating Byte). During a write operation, this floating byte is considered as part of the data preamble. As Write Gate is asserted 3 RCLKS into the data preamble, the "write splice" associated with Write Gate assertion, due to write driver turn on time, etc, occurs during the 1 byte of the data preamble which is the floating byte. When the sector is being read, this byte is attached to the header postamble. Since Read Gate is reasserted 11.5 RCLKS into the data preamble, this ensures that it doesn't get asserted in the splice. From the above data it can be concluded that for a normal operation in the DDC, Read Gate and Write Gate assertion in the data field are separated by 8.5 RCLKS in the data preamble, hence automatically taking care of the write splice as the first byte of the data preamble.

3.5 Critical Read and Write Parameters

There are a number of drive dependent parameters which must be met in order to ensure proper operation with an ESDI drive. These are summarized below, for consideration during actual design.

3.5.1 Read Function Parameters

1. A read operation may not be initiated until 15 μs following head switch.
2. Read Gate may not be asserted during a write splice or within ±1 bit time of a write splice.
3. Read gate must be asserted within 16 bit times from the write gate assertion point when the current field was written.
4. Data (read) at the interface could be delayed by up to 9 bit times from the data recorded on the disk media.
5. RCLK and RDATA are valid within the number of PLO sync field bytes specified by the drive configuration after read enable and a PLO sync field is encountered.

3.5.2 Write Function Parameters

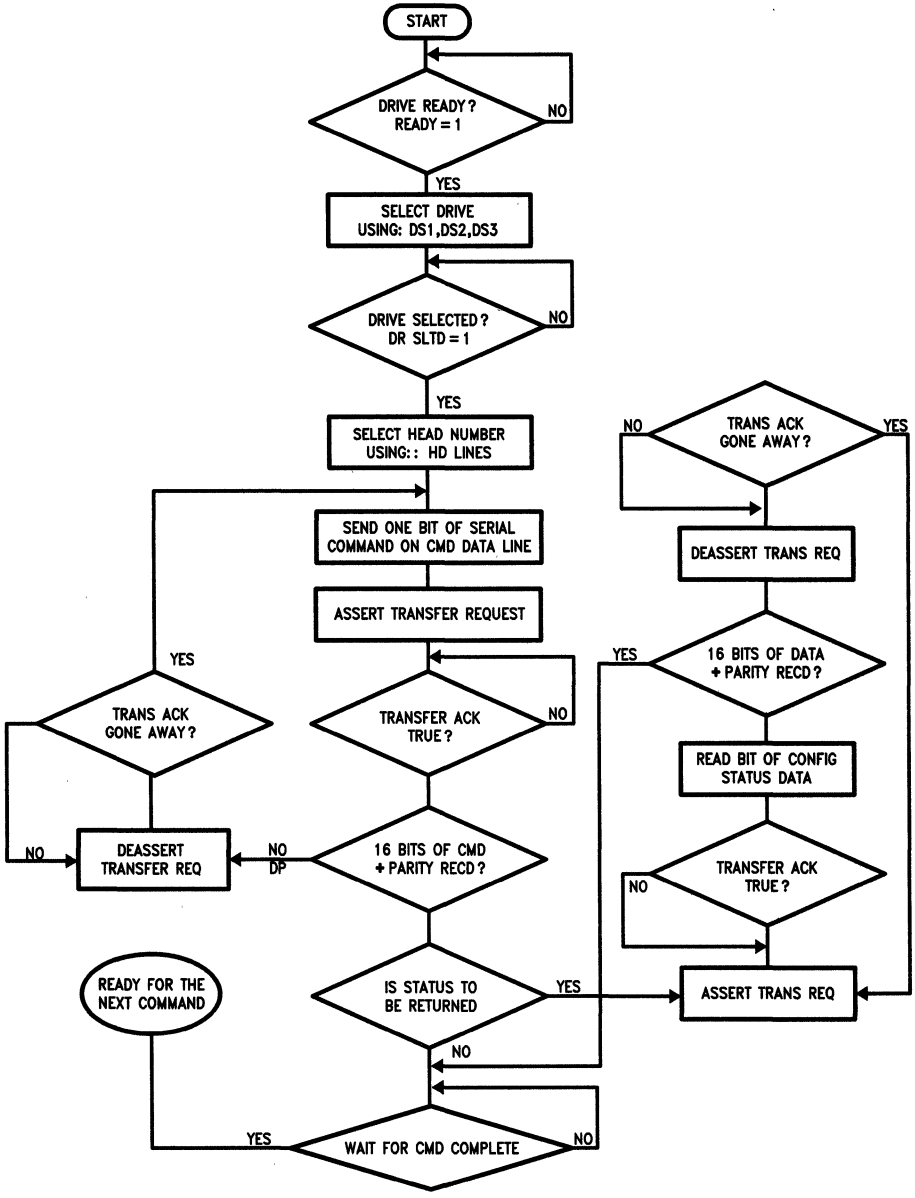
1. Assuming head selection is stabilized, the time lapse from deassertion of Read Gate to assertion of Write Gate shall be five reference clock periods minimum.
2. Write Clocks must precede Write Gate by a minimum of two and a half Reference Clock periods.
3. Write driver plus data-encoder turn on time (write splice width) is between 3 and 7 reference clock periods.
4. To account for data-encoding delays, write gate must be held on for at least 2 byte times after the last bit of information to be recorded. This implies a minimum data postamble length of 2 bytes.
5. The time lapse before Read Gate or AME can be asserted after deassertion of Write Gate is defined by the "ISG bytes after index/sector" in the configuration data response (10 μs).
6. Write Gate must be deasserted at least 1 μs before a head change and shall not be asserted until 15 μs after a head change or command complete.
7. Write data received at the I/O connector will be delayed by the encoder by up to 8 bits maximum prior to being recorded on the media.

Some of these parameters were accommodated in the combined PAL solution, while others need to be accommodated in firmware. The above discussion covered the hardware and relevant firmware considerations for designing the disk side data path section of the controller subsystem. The other aspect of the disk side design is the control path which is discussed in the following sections.

3.6 Disk Side Control Path Design

Since the DDC is a data path controller, the control path functions of the drive have to be controlled by the local intelligence in the disk controller subsystem. This offers more versatility and is less of a handicap, for it allows the DDC to be used with any of the disk interface standards. Also the control path functions like seeks, etc, are very slow operations and by not handling them, the DDC is able to achieve maximum operating speeds of 25 Mbits/sec. Any μP or μC with simple I/O ports would suffice for the control path functions. This usually is no extra overhead, because a local μP or μC is necessary to handle the protocol over the system bus and accordingly set up and activate the DDC anyway.

The control signals are sent over the drive's "A" control cable. Details of the respective signals are discussed in section 2.1. These signals are sent/received, to/from the drive through industry standard open collecter drivers (DP8311) and receivers (74ALS240), with appropriate terminations in accordance with the ESDI specification. These are also shown in the complete design, schematic *(Figure 13)*. In the system design under consideration, the control path functions are controlled by the NSC800, through the NSC810 programmable I/O port. The ideal flow of operations for the control path is shown in *Figure 12* and is self-explanatory.



TL/F/9381–16

FIGURE 12. Control Path Flow of Operations

The above discussion covered the hardware and relevant firmware considerations for designing the disk side of the disk controller subsystem. The other aspect of the disk controller subsystem design is the system side. This essentially involves the interface of the DDC to the local buffer memory and interface of this local buffer over the system bus to the host system. Discussion of the system side design follows in the next section.

4.0 CONTROLLER DESIGN—SYSTEM SIDE

4.1 System Side Hardware Considerations

The system side architecture in controller subsystems usually consists of two buses. A local bus accommodates the DDC, local microprocessor or microcontroller and the buffer memory while the remote bus is usually a standard bus like VME, STD100, MULTIBUS®, SCSI, etc, which connects the controller to the host system. The buffer memory on-board is a disk buffer which could have a maximum size of 64k. The DDC supports two 16-bit DMA channels, local and remote DMA. The local DMA transfers the data between the on-board FIFO and local buffer memory, while the remote DMA transfers data between the local buffer memory and the host over the system bus.

In the controller subsystem under consideration *(Figure 13)*, the local μP is the NSC800™. This can access the registers of the DDC in the peripheral mode and executes program code from an EPROM. All the peripheral chips are memory mapped and the 74HC138 decodes the address lines to generate the various chip selects. A PAL has been designed to provide arbitration between the DDC and the NSC800 for use of the local bus. This design was intended to provide a good exerciser for the DP8466 and hence a terminal is connected to the μP (NSC800) through a UART (NSC858). A monitor has been developed which allows exercise of the DDC in a lot of modes and interacts with the user through the terminal. Hence the remote DMA is not really used. Also in order to be able to do both byte-wide and word-wide transfers, a detection logic was implemented. Only an 8k local buffer RAM is used, accommodating 8k bytes or 8k words. LEDs are provided for visual indications of certain disk parameters.

As mentioned above, from a general design point of view, this could be easily extended to a standard system bus like VME, MULTIBUS II, etc, using the remote DMA channel of the DDC and using the local microprocessor to handle the communications protocol to the DDC and to the host operation system over the system bus.

4.2 System Side Programming Considerations

The firmware in conjunction with the microprocessor is essentially responsible for deciphering the protocol sent over the system bus and then based on the requested operations, set up the control path through the I/O ports and set up the DDC to initiate the disk operation. The driver routines to handle the control path are usually very simple and need to do the defined task as outlined in Section 3.6. The driver would consist of various blocks to implement the different ESDI commands and interpret the status reported; however, the flow would be as shown in *Figure 12*. Once the drive is positioned at the right track and ready to start the operation, control is handed over to the DDC. Before the DDC can be instructed to initiate a disk operation it has to be prepared, i.e., all registered set up to achieve the desired task.

Figure 14 shows the sequence of actions to be done with parameter RAM is initialized with the pattern and count values in accordance with the format desired. The DMA registers should be set up next starting with the LT and RT registers followed by the DMA Address Byte #, indicating start boundaries in memory and finally the sector byte count # and remote data byte count # registers. The ECC registers (preset, taps and control) are set up and the DF register is set up accordingly. Having set up the various registers, the format registers are manipulated if desired, sector counter, NSO counters initialized, and DC register loaded to initiate a disk operation. The interrupts are monitored with reads of the status register to determine the results of the disk operation. If desired, certain fancy operations like error correction cycle, interlock mode of operations, FIFO table formatting could be implemented with additional firmware.

An effort has been made in this application note to introduce the designer to the ESDI interface, and explain the intricacies of designing to its specifications with the DDC. The emphasis was on the disk side, as system side requirements may vary with different design situations. A representative design has been included which was built and tested at National (shown in *Figure 13*). For more details on the ESDI standard, refer to the official specifications provided by the ESDI committee. For more details on the DP8466A refer to AN-413 and the data sheet.

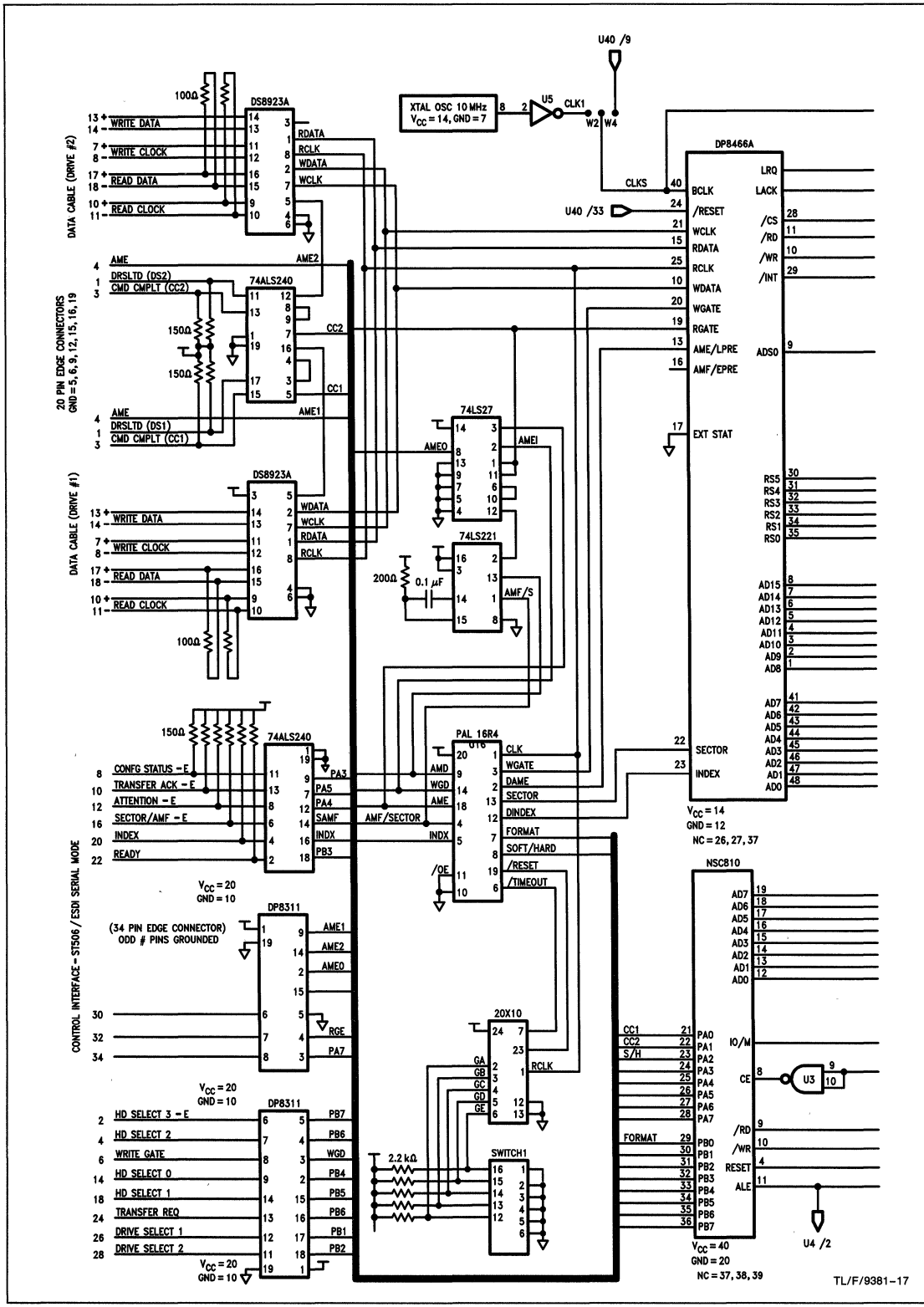
APPENDIX
Schematic of a Representative Design of an ESDI Disk Controller

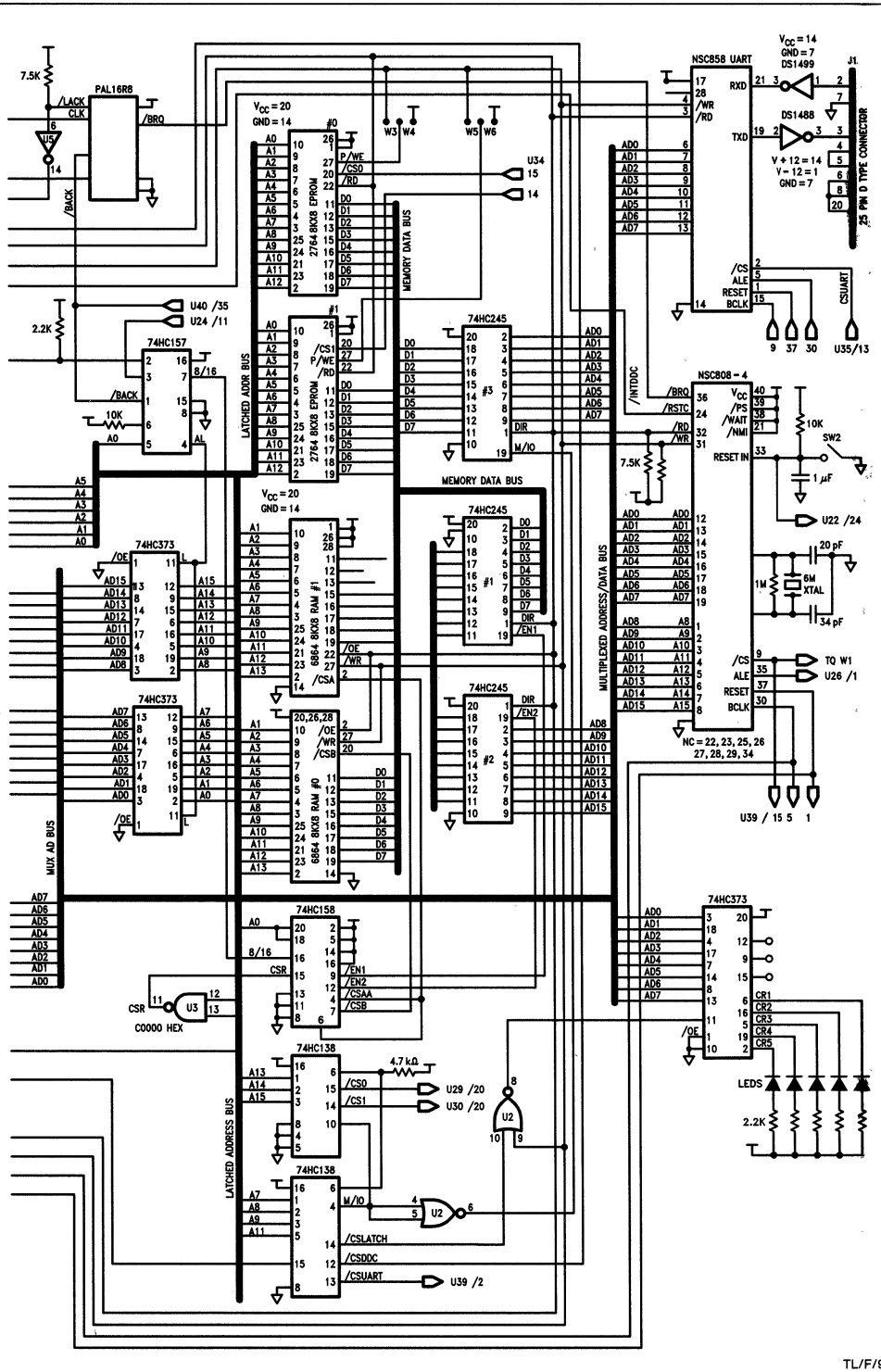
Shown in the attached schematic is the design of an ESDI disk controller system. Emphasis has been placed on the disk side of the design, as that is of foremost concern with respect to the DDC. The design incorporates the PALs as described in the previous sections, to implement the shortcomings of the DDC, in order to completely support the Enhanced Small Device Interface specifications.

The control signals are generated using the NSC810 programmable I/O port. They are connected to the "A" cable through industry standard open collector drivers, (DP8311) and receivers, (74ALS240). The receivers support a termination of 150Ω. The Data path signals are directly controlled by the DDC. They are connected to the "B" cable through industry standard differential line drivers and receivers (DS8923A). The appropriate terminations required are shown in the schematic. The design supports two drives and the appropriate data cable is enable by the drive selected line as shown.

The local microprocessor used is the NSC800, which is responsible for controlling the NSC810 to generate the appropriate control signals, and also programming the DDC to initiate the desired disk operation. This design is essentially a demonstration system and hence supports a UART (NSC858), through which the user is allowed to interact through a special monitor to initiate specific operations for the DDC. The memory design has been done so that the system could be operated in both the byte-wide transfer mode and also the word-wide transfer mode. The arbitration PAL is a simple one to arbitrate the bus between the DDC's DMA activity and the microprocessor accesses of the DDC.

It should be noted that the DDC has the capability of two 16-bit DMA channels, local and remote. The local DMA is used to transfer data between the FIFO and the local buffer memory, while the remote DMA is used to transfer data between the local buffer memory and the host system over a system bus like VME, MULTIBUS, SCSI, etc. Hence the design could easily be extended to these situations, if desired.


U40 /9
XTAL OSC 10 MHz
VCC = 14, GND = 7
U5
CLK1
W2 W4
DP8466A
CLKS
U40 /33
BCLK
/RESET
WCLK
RDATA
RCLK
WDATA
WGATE
RGATE
AME/LPRE
AMF/EPRE
EXT STAT
SECTOR
INDEX
LRQ
LACK
/CS
/RD
/WR
/INT
ADS0
RS5
RS4
RS3
RS2
RS1
RS0
AD15
AD14
AD13
AD12
AD11
AD10
AD9
AD8
AD7
AD6
AD5
AD4
AD3
AD2
AD1
AD0
VCC = 14
GND = 12
NC = 26, 27, 37
DATA CABLE (DRIVE #2)
WRITE DATA
WRITE CLOCK
READ DATA
READ CLOCK
DS8923A
RDATA
RCLK
WDATA
WCLK
100Ω
2D PIN EDGE CONNECTORS
GND = 5, 6, 9, 12, 15, 16, 19
AME
DRSLTD (DS2)
CMD CMPLT (CC2)
74ALS240
150Ω
CC2
CC1
AME2
AME1
AME
DRSLTD (DS1)
CMD CMPLT (CC1)
DATA CABLE (DRIVE #1)
74LS27
AMEO
AMEI
74LS221
200Ω
0.1 μF
AMF/S
CONTROL INTERFACE - ST506 / ESDI SERIAL MODE
CONFG STATUS - E
TRANSFER ACK - E
ATTENTION - E
SECTOR/AMF - E
INDEX
READY
PA3
PA5
PA4
SAMF
INDX
PB3
PAL 16R4
CLK
WGATE
DAME
SECTOR
DINDEX
FORMAT
SOFT/HARD
/RESET
/TIMEOUT
AMD
WGD
AME
AMF/SECTOR
INDX
/OE
VCC = 20
GND = 10
DP8311
(34 PIN EDGE CONNECTOR)
ODD # PINS GROUNDED
AME1
AME2
AMEO
RGE
PA7
20X10
RCLK
GA
GB
GC
GD
GE
2.2 kΩ
SWITCH1
HD SELECT 3 - E
HD SELECT 2
WRITE GATE
HD SELECT 0
HD SELECT 1
TRANSFER REQ
DRIVE SELECT 1
DRIVE SELECT 2
PB7
PB6
WGD
PB4
PB5
PB6
PB1
PB2
NSC810
CC1
CC2
S/H
FORMAT
PA0
PA1
PA2
PA3
PA4
PA5
PA6
PA7
PB0
PB1
PB2
PB3
PB4
PB5
PB6
PB7
IO/M
CE
U3
/RD
/WR
RESET
ALE
U4 /2
VCC = 40
GND = 20
NC = 37, 38, 39
TL/F/9381-17


PAL16R8
/LACK
CLK
/BRQ
/BACK
7.5K
2.2K
10K
74HC157
8/16
74HC373
74HC158
74HC138
2764 8KX8 EPROM
6864 8KX8 RAM
LATCHED ADDR BUS
MEMORY DATA BUS
MUX AD BUS
LATCHED ADDRESS BUS
MULTIPLEXED ADDRESS/DATA BUS
74HC245
NSC858 UART
DS1499
DS1488
25 PIN D TYPE CONNECTOR
NSC808-4
4.7 kΩ
LEDS
2.2K
20 pF
1M
6M XTAL
34 pF
1 µF
SW2
RESET IN
C0000 HEX
TL/F/9381-18

Interfacing National's DP8466A to the SMD Storage Module Device Interface Standard (Hard Sectored Drive)

National Semiconductor
Application Note 501
Ramachandran Gopalan

1.0 INTRODUCTION

With the advent of computer technology, the demand for high performance memory devices has been increasing. Before the introduction of 5¼ and 8 inch Winchester disk drives in the late 1970's, minicomputers and mainframes were the only systems that utilized rigid disks. Storage capacity, data transfer rates and to some extent cost, are the main factors which determine the performance of Winchester drives. The data transfer rate is highly dependent on the interface protocol adopted by the drive and also the electronics. The defacto industry standard drive interface for low end systems, ST506 by Seagate Technology, supports a maximum data transfer rate of 5 Mbits/sec and requires the disk controller to take care of data separation (synchronization and decoding). With the intention of supporting higher data transfer rates, newer standards were defined, like the ESDI (Enhanced Small Device Interface), which incorporates data separation on the drive itself and supports a data transfer rate of 10 Mbits/sec. The intelligent disk interface standards like SCSI and IPI, incorporate an interface to the host system with a well established high level communication protocol, however they also need to incorporate a drive-level interface like ESDI, SMD, etc.

With the intention of standardization of a common interface and to prevent product obsolescence, Control Data Corporation developed an intelligent interface called the Storage Module Device (SMD). This interface allows different drives to use the same hardware signals, even though their capacities and physical sizes were different. Variations of the SMD were also introduced. SMD started out at 9.667 Mbits/s data transfer rate and has since gone through several upgrades to a 24 Mbit/s option (SMD-E), introduced recently by Control Data Corporation. The SMD disk interface is a high quality proven attachment and is well on its way to becoming a defacto industry standard. It incorporates error recovery facilities, includes power sequencing for multiple units and is adaptable to many different rigid-disk storage units. Although it requires two very bulky and expensive cables, in high-end products this is an acceptable drawback and SMD still remains a popular choice for the higher capacity 5¼ and 8 inch Winchesters. It offers several advantages over the ST506 type interface in the high capacity arena (like parallel seek instead of serial step pulses and better status reporting), however, it is not a trivial interface when it comes down to designing a controller.

This application note looks into the definition of the SMD interface and the various design aspects of building a Disk Controller for a hard sectored drive supporting the SMD interface, using National Semiconductor's Disk Data Controller IC, DP8466A (DDC). Emphasis is laid on the disk interface design as that is of relevance to the DDC. The DDC is the most versatile LSI disk controller in the market today, offering fully programmable format features, maximum range in data rate (50 Kbits/s to 25 Mbits/s, dual DMA capability with a transfer rate of 10 Mbytes/sec, programmable error checking and correction and many other features which makes the design of a disk controller simpler and less complex. To be able to support high performance drives in the future, disk controller IC's must be capable of handling data transfer rates >20 Mbits/s. The DDC concentrates on the high speed data path signals, while the slower drive control operations are left to an inexpensive microcontroller, or the local (on board) microprocessor. The DDC in conjunction with this local intelligence, can achieve the Disk Controller function which was accomplished by 100–150 SSI/MSI integrated circuits in earlier Disk Controller systems, minimizing complexity, cost and system overhead.

2.0 SMD INTERFACE DEFINITION

The SMD disk interface standard started out as a dominant de-facto standard for 14" OEM Winchester drives and is virtually the basis of 8"-14" OEM disk drive industry today, with an eye out for the 5¼" OEM market in the future. It was approved by the ANSI committee (ANSI X3.91), in 1982. The interface consists of a 60-pin twisted pair control (A) cable and a 20-pin data (B) cable with a molded foil ground plane on one side. The 'A' cable is responsible for all head movements, status reporting and issuing commands, while the 'B' cable is used for reading and writing NRZ data. The 'A' cable assignments are shown in *Figure 1.* Address and Control functions are transferred on ten BUS OUT lines. The significance of the information on these lines is indicated by one of the six TAG functions as shown in *Figure 2.* Status for real time control, device identity and current sector status are returned on eight BUS IN lines. Drives are selected by separate UNIT SELECT lines on the 'A' cable, which have their own strobe line, UNIT SELECT TAG. All TAG lines except UNIT SELECT TAG are gated by the UNIT SELECTED signal, unit referring to the drive. The 'B' cable (Data) I/O signal assignments are shown in *Figure 1.* This cable essentially handles the transmission of data and clock information, which could be at very high transfer rates. The 'A' interface cable may be connected to the controller in a star-chained mode or daisy-chained mode. Most of the drive manufacturers support a dual channel option, where the drive could be accessed by two controllers. This is controlled by three special signals on the 'A' cable—PRIORITY SELECT, BUSY and RELEASE lines.

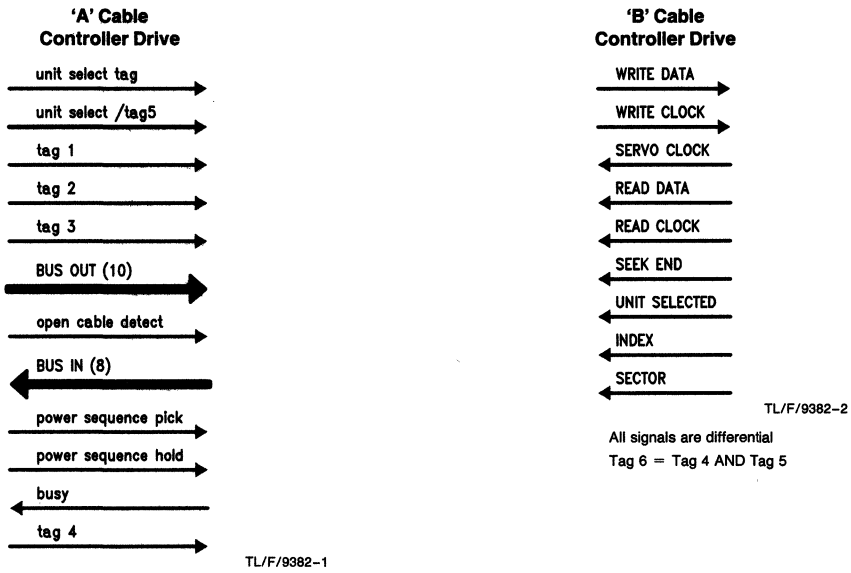


FIGURE 1. SMD Interface Cable Assignments

All input and output signals are differential in nature, and utilize industry standard transmitters and receivers. When used with properly shielded cables, this interface provides a terminated, differential transmission system for long distances, up to 50 feet, in noisy electrical environments. The maximum number of drives connected to the 'A' cable is 16. The recommended TTL differential drivers and receivers are the MC3453 and MC3550. The appropriate terminations for the driver/receiver combinations are shown in *Figure 3.* The detailed schematic in the appendix also shows them in the design. There are certain drives like CDC's 9772-XMD drive (24 MHz), where the high data rates and timing requirements at the drive necessitate the use of ECL drivers and receivers, with the appropriate terminations. The schematic in the appendix outlines the connection requirements for these with the appropriate terminations.

3.0 DISK SIDE INTERFACING

In the SMD interface it is fairly obvious that a lot of control is necessary to perform even a simple operation. The disk controller must perform simultaneous operations on both cables, as well as monitor status signals to determine if the command was executed properly. The disk controller board essentially provides a connection from the drive-level interface to the system. Its main functions are to handle the disk's control and data path, to transfer data between the disk and the system. The Disk Controller interfaces to the disk drive at one end and the main system at the other end. On the disk side the DDC interface to the drive can be broken down into two main paths—control path and data path. The control path essentially comprises of the 'A' cable signals on the SMD interface. These are differential in nature and require differential drivers/receivers to drive the cable between the chip and the interface connectors. The local μP has to activate and monitor the control lines in a certain sequence to achieve a desired operation, defined by the SMD protocol. The interface is activated by asserting the OPEN CABLE DETECT signal. This signal is sent through two drivers paralleled and the regular termination resistor is omitted. The first task then is to select the drive using the UNIT SELECT lines, latching them with the UNIT SELECT TAG. UNIT SELECTED is asserted by the drive indicating the selection of the drive. The unit is checked to see if it is ready for the next operation by monitoring the READY line. Once ready, the head and cylinder addresses are provided to place the drive's head assembly at the desired position on the media, and then the desired sector of data is sought. On arriving at the desired sector, the read/write operation is performed. In case of an unsuccessful operation, an error condition is flagged which could be determined by reading the status and monitoring other signal lines. *Figure 4* gives a detailed flow chart of the sequence of drive control operations performed by the local μP for the SMD interface standard. The status information presented on the interface is a function of the various TAG lines as seen in *Figure 2.*

Bus Out Bit	Unit Select Tag	Tag1 (cyl se.l)	Tag2 (head and upp cyl sel)	Tag3 (control sel)	Tag4 (curr sector)	Tag5 (extnd status)				Tag6 (Device Type)
0		2^0	2^0	Write Gate	0	0	1	0	1	0
1		2^1	2^1	Read Gate	0	0	0	1	1	0
2		2^2	2^2	Servo Offset +	0	0	0	0	0	0
3		2^3	2^3	Servo Offset −	0	0	0	0	0	0
4		2^4	2^4	Fault Clear	0	0	0	0	0	0
5		2^5		AME	0	0	0	0	0	0
6		2^6		Return to Zero	0	0	0	0	0	0
7		2^7	2^{10}	dztz str (Early)	0	0	0	0	0	0
8		2^8	2^{11}	dztz str (Late)	0	0	0	0	0	0
9		2^9		Release	0	0	0	0	0	0
						①	②	③	④	

Bus in Bits	Drive Status	curr sector cnt	Fault Status①	Operating Status②	Diagnostic Status③	Diagnostic Execute Status④	Device Type Status (User Defined)
0	Unit Ready	2^0	RD * WR				
1	On Cylinder	2^1	(RD + WR) * oncyl				
2	Seek Error	2^2	First Seek				
3	Fault	2^3	Write				
4	WR Protect	2^4	WR * WR Protect				
5	AMF	2^5	Head Select				
6	Index Mark	2^6	Voltage				
7	Sector Mark	2^7	Valid Status Available			tst	

FIGURE 2. Tag Bus Decode Information Available on the 'A' Cable

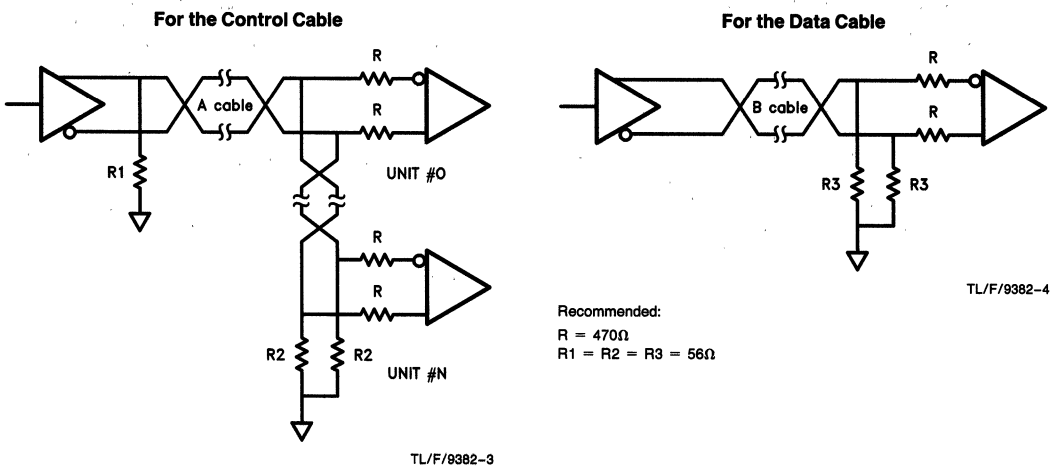
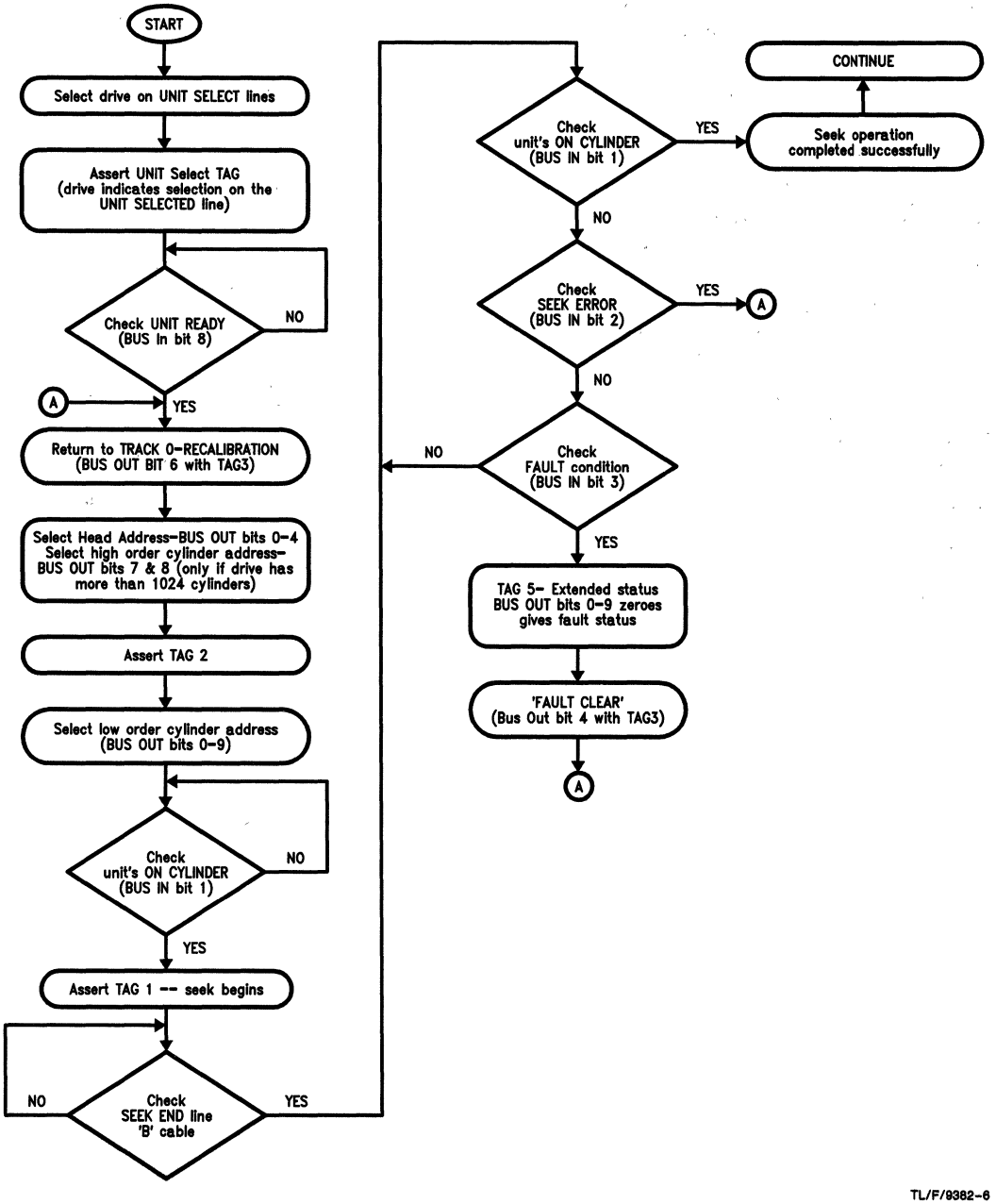


FIGURE 3. Driver/Receiver Combinations with the Appropriate Terminations for 'A' and 'B' Cable



TL/F/9382-6

FIGURE 4. Control Path—Flow of Operations

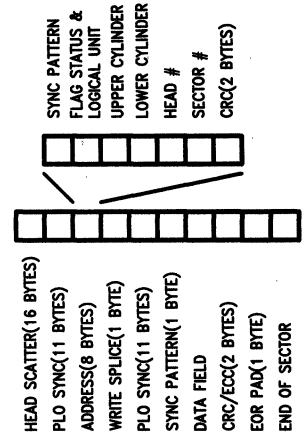
The other component of the Disk Controller system is the data path. The SMD 'B' cable essentially contains the data path signals to the drive. It consists of lines to transmit serial NTZ data to and from the drive. Associated with the read/write lines are the clocks: Read Clock for reading data, Write Clock for writing data and a Servo Clock synchronous with the rotation of the spindle, for reference. Additional signals help in determining the status of each drive on the bus. The main component of the data path in a disk controller system is the Disk Data Controller IC, like the DP8466A. The DDC can be programmed to operate in accordance to the SMD interface specification for the associated operations like formatting, reading and writing. Read Gate and Write Gate are the two main read/write control signals in the data path. Write Gate enables the write operation and is validated only when UNIT READY, ON CYLINDER, SEEK END are false. If Write Gate is asserted under any other conditions, a fault occurs and writing is inhibited. There are also certain drive dependent constraints which affect the read/write timing and must be taken care of. Listed below are some of them. Details can be found in the SMD specification.

1. write circuit turn-on delay
2. head select transients
3. read after write transients
4. read/write—encoding/decoding delays
5. write after read transients
6. PLL synchronization time

The DDC supports most of the specifications of the SMD standard, however there are certain specifications that the DDC does not directly support. These can be taken care of through software or with the help of external logic. A discussion of these considerations is given in the following sections.

4.0 HANDLING THE SMD RECOMMENDED FORMAT AND INTERFACE SPECIFICATIONS

Figure 5 shows the recommended format for SMD drives. There are certain fields in the format which are not directly supported by the DDC, which however can be implemented as discussed below.



TL/F/9382–7

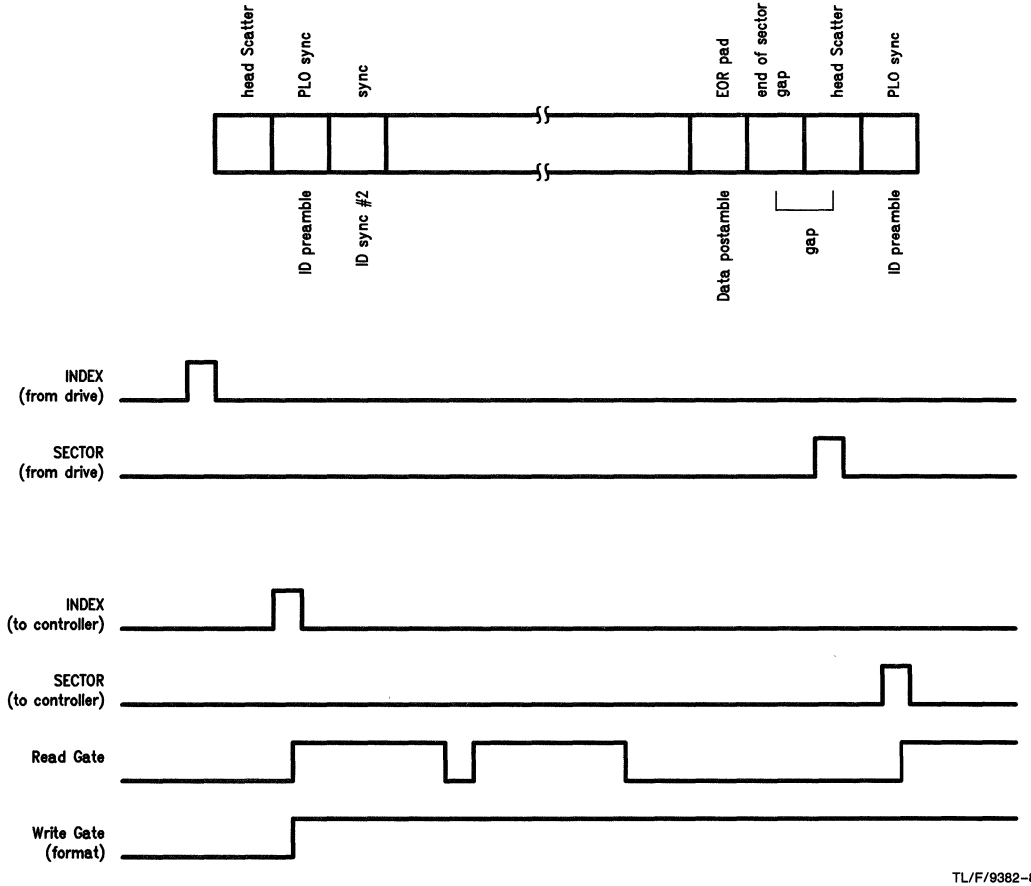
FIGURE 5. SMD Recommended Format (Hardsectored)

4.1 The Write Splice Field

The SMD format specification recommends a zero byte header postamble, a one byte write splice field and an eleven byte data preamble, (PLO sync field). The DDC does not support a separate register for the write splice field between the header and the data segments in its format parameter RAM. However this can be easily taken care of by the programmable format feature of the DDC. The format parameter registers are programmed to have a 12 byte data preamble, one byte longer than the desired length. This byte is considered for practical purposes as the write splice. Hence when formatted this will have the same pattern as the data preamble. During a write operation, this byte is kept as part of the data preamble and with the standard requirement of 8 clocks write propagation delay in the drive (due to write circuitry), the data gets written on the media beyond the write splice field, at the correct place in the preamble. In case of a read operation, this byte is considered to be part of the header postamble. Since the DDC asserts read gate 11.5 bits into the data preamble, it will never be asserted in the write splice. In this manner the write splice field can be implemented to support the recommended SMD format specification. It should be noted however that it is mandatory to use at least a one byte header postamble with the DDC for proper operation.

4.2 The Post Index/Sector Gap Field (Head Scatter)

The recommended format in the SMD standard supports a gap felled after the index/sector pulse, referred to as the post index/sector gap or head scatter. This is necessary mainly to accommodate drive dependent transients as mentioned earlier. The DP8466A does not support a separate field for this gap and also uses the index/sector pulse as a reference for the read/write control signals. To implement this gap field, an external counter is used to delay the index/sector pulse from the drive, by the desired gap count, before presentation to the DDC. The circuitry to achieve this is shown in the schematic in Appendix A. Since the index/sector pulse is presented to the DDC delayed by the length of the Head Scatter field, at the time it starts writing the PLO field etc., it's at the right area on the media. The gap field at the end of the sector is written till a sector pulse is received by the DDC. Hence the gap pattern gets written for the Head Scatter field of the next sector. *Figure 6* shows the details of this technique and the manipulation of the parameter fields.



TL/F/9382–8

The disk controller writes the end of sector gap over the head scatter area, hence achieving the writing of the post index/sector gap. The last sector overlaps to write the head scatter for the first sector.

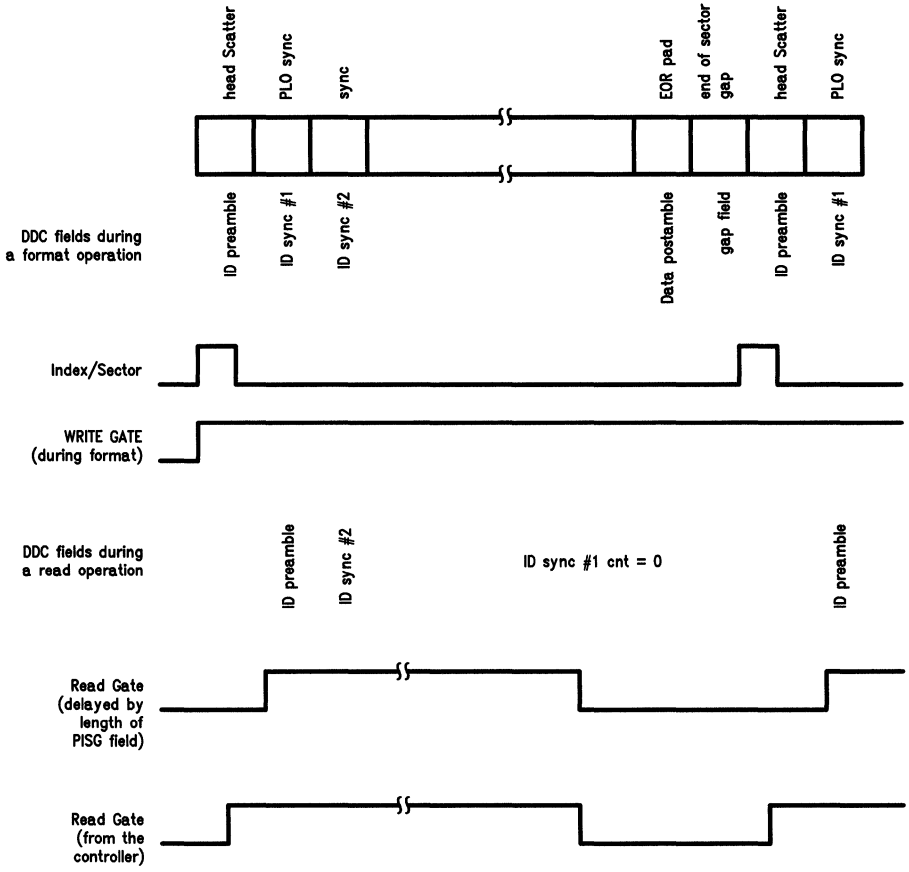
FIGURE 6. PISG Implementation (Technique 1)

4.3 Read/Write Gate Timing Restrictions

On assertion of Write Gate, there is a write splice before data is actually written on the media. Hence when reading the data, it must be ensured that Read Gate is asserted sometime after the write splice. In the DDC the delay between Write and Read Gate in the Data field is 8.5 bit times, which satisfactorily covers the write splice. However at the beginning of the sector it is only 0.5 bit times, hence Read Gate would have to be delayed to prevent assertion of the Read Gate during the write splice. This would be a problem only in the first sector after the index pulse, as during a format operation Write Gate remains asserted till the Index pulse is encountered again. This is done by delaying the Read Gate by 8 bits from the DDC to the drive, using a counter as shown in the schematic in the appendix. This results in the read gate assertion delayed by a byte even in the data field which does not make any difference to the performance.

4.4 Alternative Technique to Implement the Shortcomings in Section 4.2 and 4.3

An alternative technique to implement the Post index/sector gap and the problem of asserting read gate in splice at the beginning of the sector is to have a single delay circuit which delays the read gate qualified by index/sector, assertion to the drive by the length of this gap. The manipulations of the DDC fields to achieve this is shown in *Figure 7.*



TL/F/9382–9

Note: Read Gate assertion needs to be delayed only at the index or sector pulse.

FIGURE 7. PISG Implementation (Technique 2)

4.5 MUXing the Two Clocks on the SMD INTERFACE 'B' CABLE

The 'B' cable has two clock outputs— the SERVO CLOCK and the READ CLOCK. The SERVO CLOCK signal is a phase locked clock, (frequency dependent on the drive), generated from the servo track bits and is available at all times. The READ CLOCK signal defines the beginning of a data cell and is synchronous with the detected data. This signal is derived from the SERVO CLOCK. At the start of a read operation, READ GATE is asserted by the controller. This initiates the PLL on the drive to begin locking on the data from the media. Till this point, the clock sent to the controller is the reference clock (similar to the SERVO CLOCK frequency). When the PLL achieves phase lock, the clock transmitted on the READ CLOCK line to the controller is the one in phase sync with the data. An undefined clock may be transmitted at the point of obtaining phase sync, upon initiating or ceasing a read operation. Read Clock is in phase sync within 2.5 μs after Read Gate is asserted, (worst case). Also the WRITE CLOCK generated by the DP8466A is essentially the SERVO CLOCK retransmitted to the drive synchronized to the NRZ Write Data. The DP8466A has only one clock input line (READ CLOCK). Hence the switching of the SERVO or READ CLOCK to the DP8466A must be done externally. Also since the DP8466A's clock input cannot accommodate short pulses, this switching must be done without short pulses. The external circuitry to multiplex the two clocks and the deglitcher is shown in the schematic. This can safely operate up to clock frequencies of 25 MHz. To take care of the undefined pulse occurring on obtaining phase sync, the Read Gate signal delayed by 3 μs (to accommodate worst case lock time), is used as the 'switch' control input.

4.5.1 GLITCHLESS CLOCK MULTIPLEXER CIRCUIT

With Read Gate active (high), the source of the 8466 Read Clock is the Read Clock input signal. When Read Gate is inactive, the 8466 Read Clock will come from the Servo Clock input. If the switching between the two clocks would result in a less-than-normal-width pulse in a simple multiplexer, this glitchless circuit allows the currently active clock to finish a full one or two clock pulses before the output goes low and waits one or two clock times until the new clock appears at the "8466 Read Clock" output. So there may be one or two missing pulses but there will never be a "glitch" (narrow pulse width).

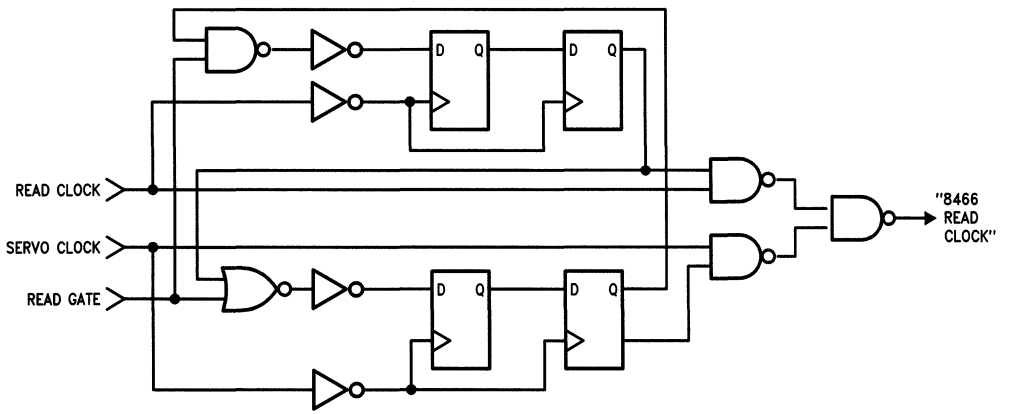
If the disk system also uses the National Semiconductor DP8463B (2, 7) RLL ENDEC, no external circuitry is required as this glitchless multiplexer is already incorporated into the DP8463B.

5.0 SYSTEM CONSIDERATIONS AND CONCLUSIONS

The DP8466A supports a dual channel onboard DMA controller which simplifies the task of interfacing to any system bus. System design considerations are flexible to the designer as outlined in the Application Note: "Designing an ESDI Disk Controller system with National's DP8466A" and AN-413. An effort has been made in this application note to introduce the designer to the SMD interface, and explain the intricacies of designing to its specifications with the DP8466A. The emphasis was on the disk side design and a representative design of the disk side is included in the appendix. This is the disk side for a disk controller designed for the 25 MHz XMD drive from Control Data Corporation. The board was built and tested in the Labs at National and was found to perform satisfactorily. For more details on the SMD standard, refer to the ANSI document on SMD interface specifications. For more details on the DP8466A refer to AN-413 and the datasheet.

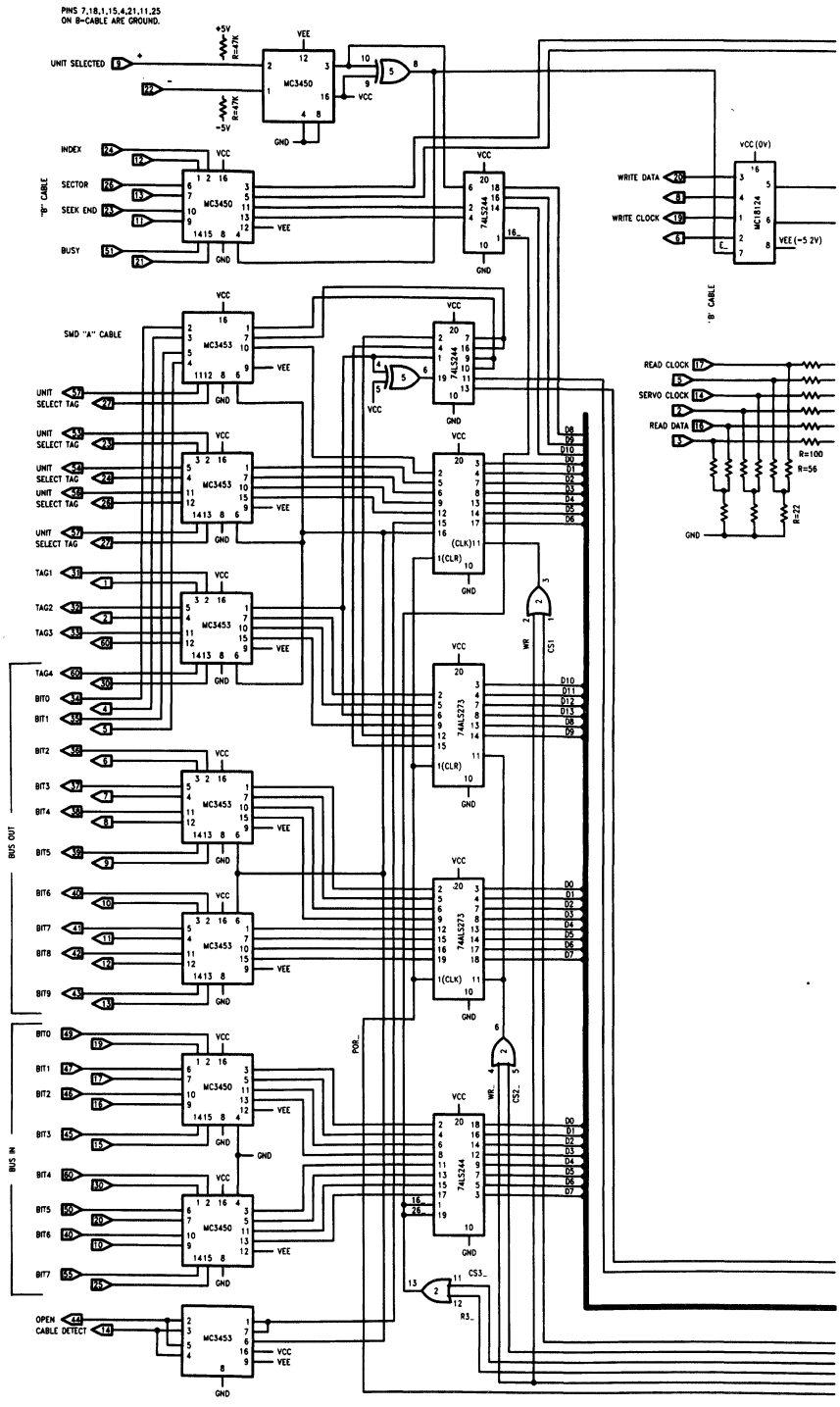
GLITCHLESS MULTIPLEXER

Block Diagram



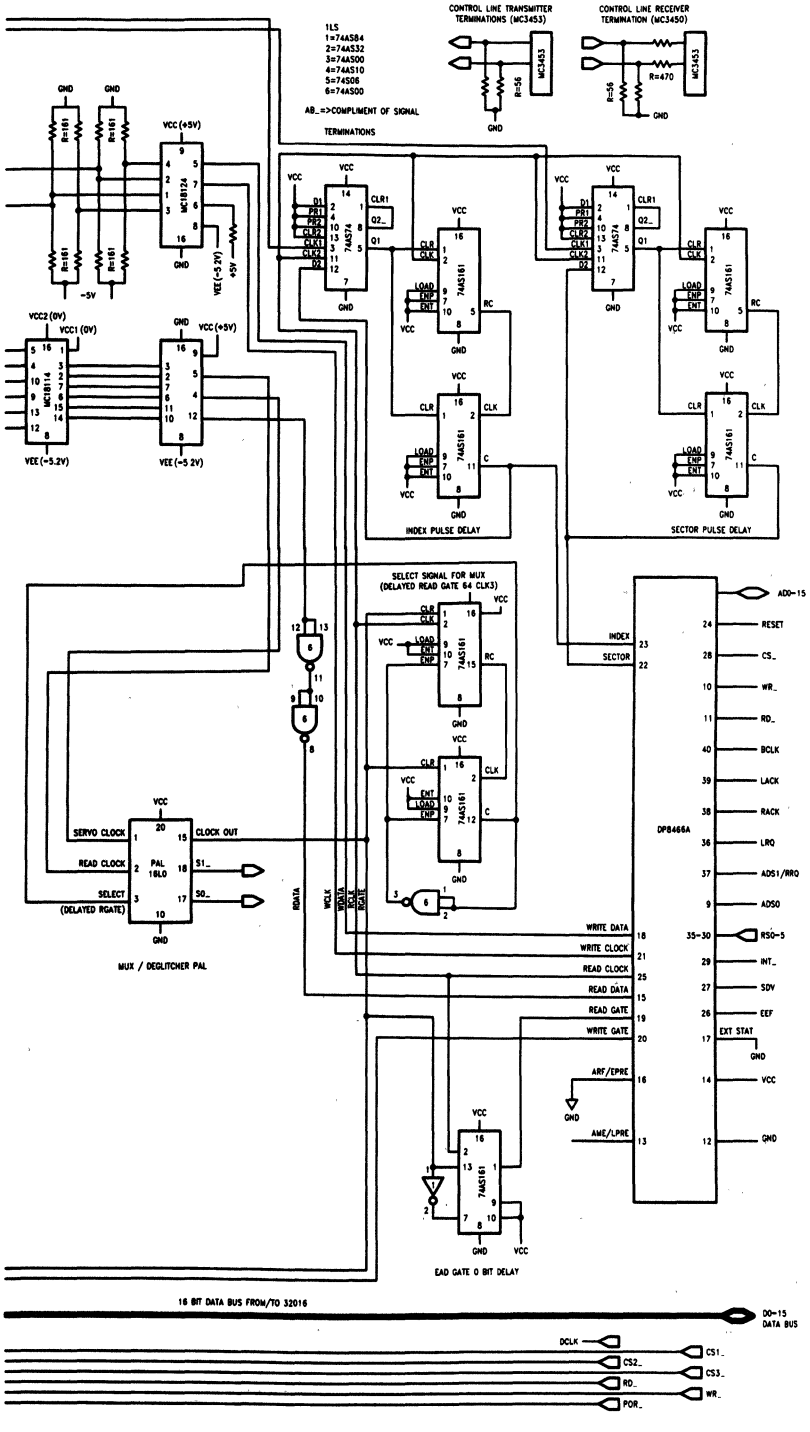
TL/F/9382–10

FIGURE 8. Glitchless Multiplexer



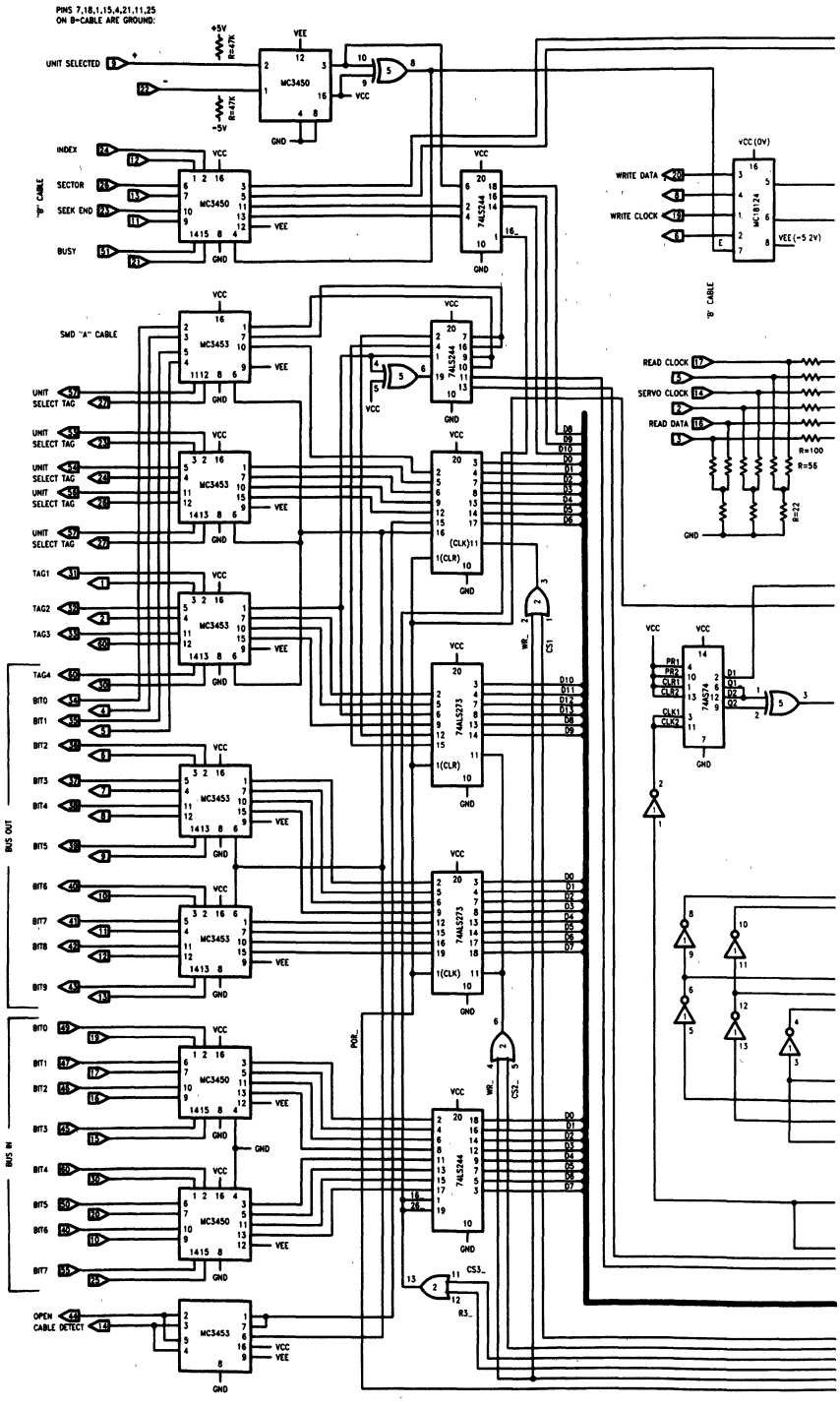
TL/F/9382–13

FIGURE 9



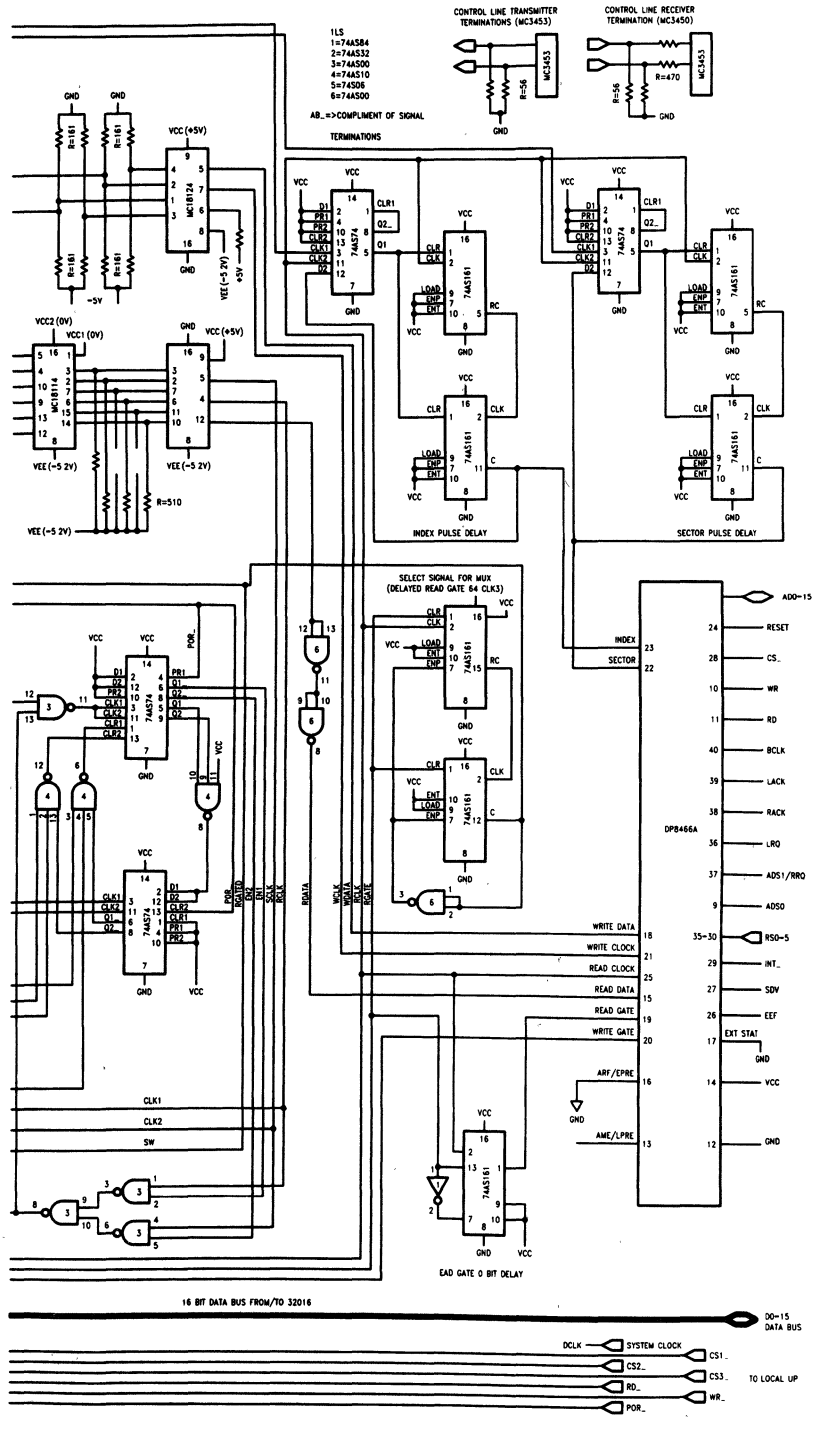
TL/F/9382–14

FIGURE 9 (Continued)



TL/F/9382-15

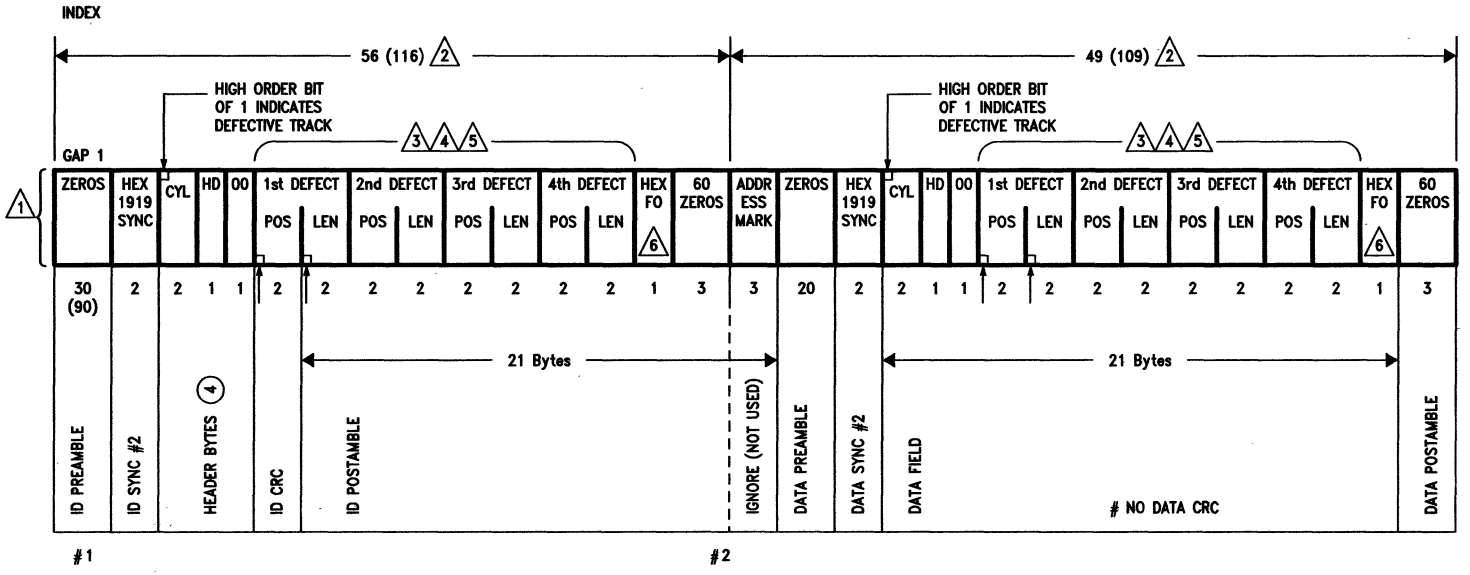
FIGURE 10



TL/F/9382–16

FIGURE 10 (Continued)

Defect List Format (SMD Drives)



TL/F/9382–12

Notes:

No data CRC is selected, OPERATION is IGNORE HEADER—READ DATA

If in case of a defect, so that #1 field becomes 90 and #2 field becomes 60 then it poses no complications for reading the defect information. However if the defect format is to be written again then it might cause complications as the max. byte count for a field is 32 bytes, hence cannot be done unless delayed physically.

Under no conditions can sync #1 be used with a pattern of zeroes. Its count must be kept zero.

Note that this technique to enable the DP8466 to be able to read the defect list format is possible only if written in two parts originally. HITACHI, NEC and CDC support the above but *not FUJITSU.*