HD64950S

Preliminary

File Data Processor (FDP)

Description

The HD64950S file data processor (FDP) is a hard disk controller LSI produced by 1.5-µm CMOS processing and designed to interface with a 16-bit microprocessing unit (MPU).

The FDP is placed between the host system and a disk drive unit to provide serial-to-parallel and parallel-to-serial data conversion, error checking, and automatic correction of data transferred to and from the disk drive unit.

This FDP provides separate controller section that functions as a file manager (FM: general-purpose microcomputer such as the H8/532, microprocessor-controlled high performance level, various disk drive interfaces, user-specific programmable data processing, compatibility with various standard buses, and expansibility.

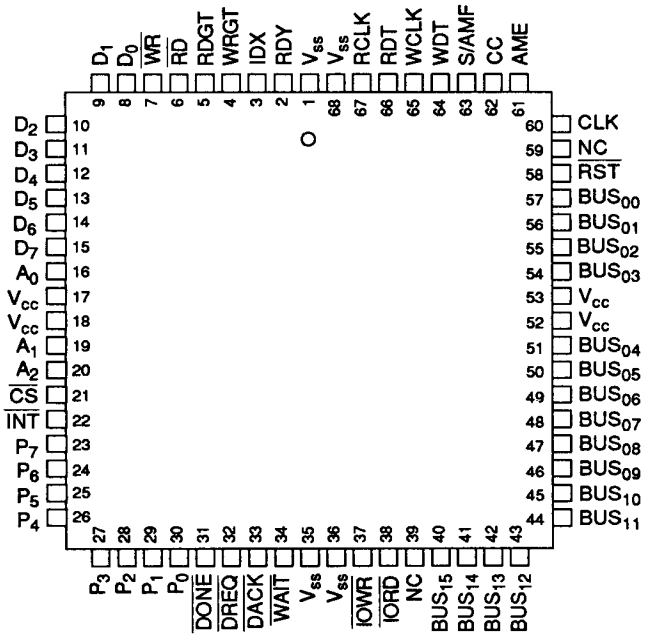
Features

- Versatility
  - Selectable data bus width:
    16-bit or 8-bit data bus (Only the 8-bit data bus is available in master mode.)
  - Selectable operation mode:
    Master mode or slave mode
  - Selectable DMA transfer mode:
    Burst transfer mode or cycle steal transfer mode
- High-speed transfer
  - Simultaneous data transfer between host systems and between hard disk drives
  - HDD transfer rate:
    Maximum 15 Mbits/s
  - Host transfer rate:
    Maximum 6 Mbytes/s (on 16-bit data bus)
    Maximum 4 Mbytes/s (on 8-bit data bus)
  - Multi-sector processing by controller available
  - Retry processing available
- Built-in data buffer
  - The FDP's data buffer consists of two 608-byte static memories (used as two 544-byte memories during DMA transfer).
- Error correction function
  - Selectable ECC polynominal (any)
  - Selectable ECC data length (4, 6, or 7 bytes)
- Highly flexible operation
  - Operations by combinations of 32 format commands
  - Wide sector length range (4 to 1088 bytes)
  - Free format
- Low power dissipation
  - CMOS circuit configuration reduces power dissipation (typically 200 mW)

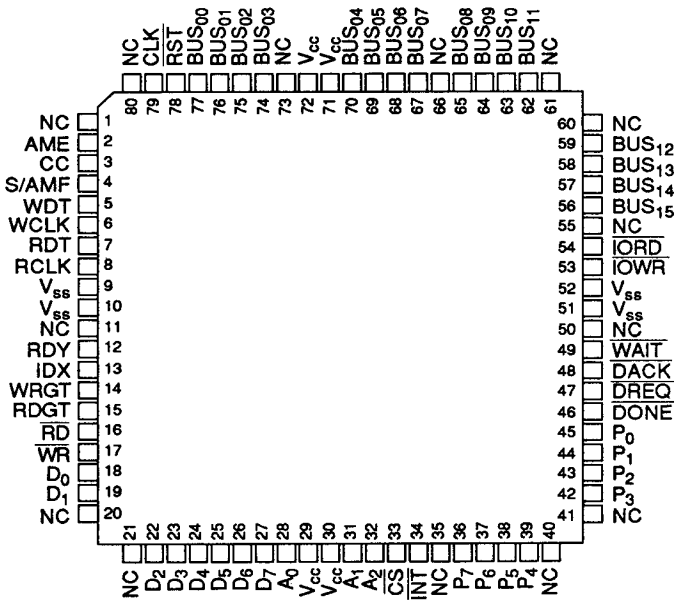
Ordering Information

Product Name	Package
HD64950SCP	68-pin PLCC (CP-68)
HD64950SFP	80-pin flat plastic package (FP-80A)

Pin Arrangement



(CP-68)



(FP-80A)
(Top View)

Pin Description

Class.	Pin No. (PLCC)	Symbol	Pin Name	Type	Function
Supply voltage, clock, reset	17, 18, 52, 53	V_{cc}		—	+5 V supply
	1, 35, 36, 68	V_{ss}		—	Ground (connected to the GND pin)
	60	CLK		Input	Inputs system clock
	58	$\overline{RST}$	Reset	Input	Resets the internal status of the FDP
File manager interface	6	$\overline{RD}$	Read	Input	Controls read operations: the FDP outputs data while this signal is asserted
	7	$\overline{WR}$	Write	Input	Controls write operations: the FDP fetches data at the rising edge of this signal
	21	$\overline{CS}$	Chip select	Input	Enables the file manager to access the FDP: the file manager reads/ writes to the selected register in the FDP while $\overline{CS}$ is low
	16, 19, 20	A_0–A_2	$ADDRESS_0$–$ADDRESS_2$	Input	Direct register address lines for file manager access
	8–15	D_0–D_7	$DATABUS_0$–$DATABUS_7$	Input/ output	Data lines between file manager and FDP
	22	$\overline{INT}$	Interrupt	Output (open drain)	Informs the file manager of the end of a command (open-drain output)

A_2	A_1	A_0	Register
0	0	0	AR
0	0	1	DTR
0	1	0	CEND
0	1	1	CMR
1	0	0	BUF0
1	0	1	BUF1

Pin Description (cont)

Class.	Pin No. (PLCC)	Symbol	Pin Name	Type	Function
Port	23–30	P_7–P_0	$PORT_7$–$PORT_0$	Input/ output	8-bit I/O port whose direction (input or output) is determined by the data direction register (DDR)
Drive interface	5	RDGT	Read gate	Output	Shows that data can be read from the disk drive
	4	WRGT	Write gate	Output	Shows that data can be written to the disk drive
	3	IDX	Index	Input	Inputs a track start signal
	2	RDY	Ready	Input	Shows that the disk drive is ready
	63	S/AMF	Sector, address mark found	Input	Indicates the start of a sector (inputs a sector pulse in hard sector mode or an address mark found signal in soft sector mode)
	67	RCLK	Read clock, reference clock	Input	Inputs disk read/write clock from the disk drive (the same frequency as the serial data transfer rate)
	66	RDT	Read data	Input	Inputs serial data from the disk drive (in NRZ code)
	65	WCLK	Write clock	Output	Outputs disk write clock to the disk drive (the same frequency as RCLK)
	64	WDT	Write data	Output	Contains serial data to be written to the disk (NRZ code is output to the disk drive in synchronism with WCLK)
	62	CC	Command complete	Input	Indicates that the disk drive is ready to receive a command
	61	AME	Address mark enable	Output	Instructs the ESDI disk drive to read or write an address mark

Pin Description

FDP in Slave Mode (MST bit of MDR 1 is 0)

Class.	Pin No.	Symbol	Pin Name	Type	Function
Host interface (slave mode)	32	$\overline{DREQ}$	DMA request	Output	Asserted when the FDP requests the DMAC to transfer data
	33	$\overline{DACK}$	DMA acknowledge	Input	Inputs data transfer timing during a DMA transfer: the FDP is enabled to read or write when this signal is low
	31	$\overline{DONE}$	Done	Input	Indicates the completion of DMA transfer
	37	$\overline{IOWR}$	I/O write	Input	Controls writing to the data buffer: the FDP fetches data at the rising edge of this signal
	38	$\overline{IORD}$	I/O read	Input	Controls reading from the data buffer: the FDP outputs data while this signal is asserted
	48–51, 54–57	BUS_{00}–BUS_{07}	BUS_{00}–BUS_{07}	Input/ output	Bi-directional data bus between host and data buffer (in both 16-and 8-bit modes)

B8 (MDR1)	Bus mode	BUS_{00}–BUS_{07}
1	8-bit mode	Data buffer
0	16-bit mode	Data buffer (Low byte)

	40–47	BUS_{08}–BUS_{15}	BUS_{08}–BUS_{15}	Input/ output	Bi-directional data bus (16-bit mode (B8 = 0) only)

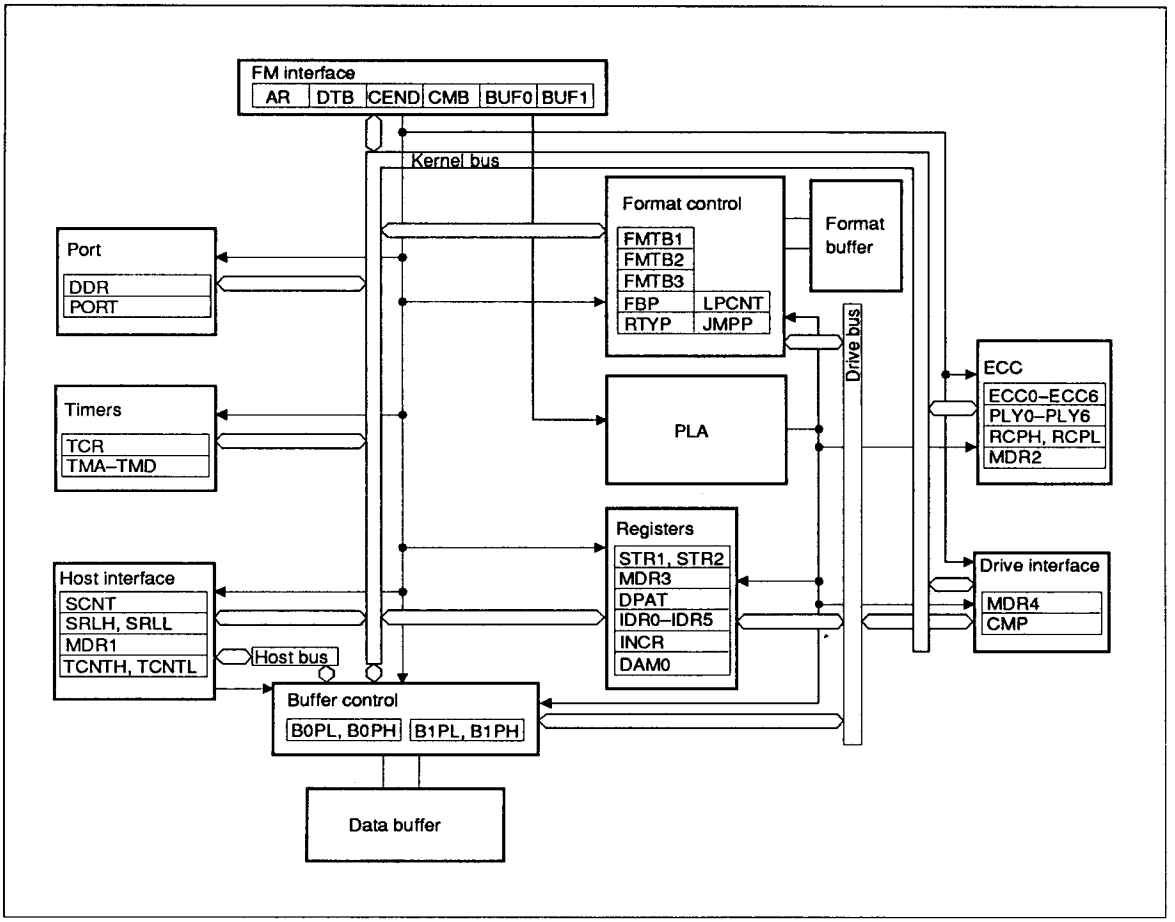
B8 (MDR2)	Bus mode	BUS_{08}–BUS_{15}
1	8-bit mode	Any (must be pulled up)
0	16-bit mode	Data buffer (High byte)

Pin Description

FDP in Master Mode (MST bit of MDR 1 is 1)

Class.	Pin No.	Symbol	Pin Name	Type	Function
Host interface (master mode)	32	$\overline{\text{DREQ}}$	DMA request	Input	Requests the FDP for a DMA transfer
	33	$\overline{\text{DACK}}$	DMA acknowledge	Output	Outputs data-transfer timing during a DMA transfer: the FDP sets this signal low before starting a read or write cycle
	31	$\overline{\text{DONE}}$	Done	Input	Indicates the end of a DMA transfer
	37	$\overline{\text{IOWR}}$	I/O write	Output	Indicates that the FDP is executing a write cycle: the FDP outputs data while this signal is asserted
	38	$\overline{\text{IORD}}$	I/O read	Output	Indicates that the FDP is executing a read cycle: the FDP fetches data at the rising edge of this signal
	34	$\overline{\text{WAIT}}$	Wait	Input	Requests the FDP to prolong the access cycle: the $\overline{\text{IORD}}$ and $\overline{\text{IOWR}}$ signals are not negated while this signal is low
	48–51, 54–57	BUS_{00}–BUS_{07}	BUS_{00}–BUS_{07}	Input/ output	Bi-directional data bus (used to carry data buffer data during a DMA transfer)
	40–47	BUS_{08}–BUS_{15}	BUS_{08}–BUS_{15}	Input/ output	Not used in master mode (must be pulled up)

Block Diagram



The components shown in the block diagram are described below.

FM Interface: Sets data in the indirect registers, data buffer, and format buffer, based on data set in the direct registers by the FM. Decodes a direct command and displays that command's execution status.

Host Interface: Generates signals for data transfer and controls data transfer between host and data buffer.

Format Buffer: Stores format commands.

Format Control: Automatically writes into the format buffer any data that is set in indirect registers FMTB1, FMTB2, and FMTB3. Reads the data in the format buffer at the reception of a START command, and outputs its code to the PLA.

PLA: Controls the disk drive interface, ECC, registers, and formats according to format commands, at the reception of a START command.

Drive Interface: Reads, writes, or compares disk data under PLA control, then performs serial-to-parallel or parallel-to-serial data conversion.

ECC: Generates and performs ECC and CRC checks, and calculates ECC correction patterns.

Register: Stores data and mode information required to access the disk drive.

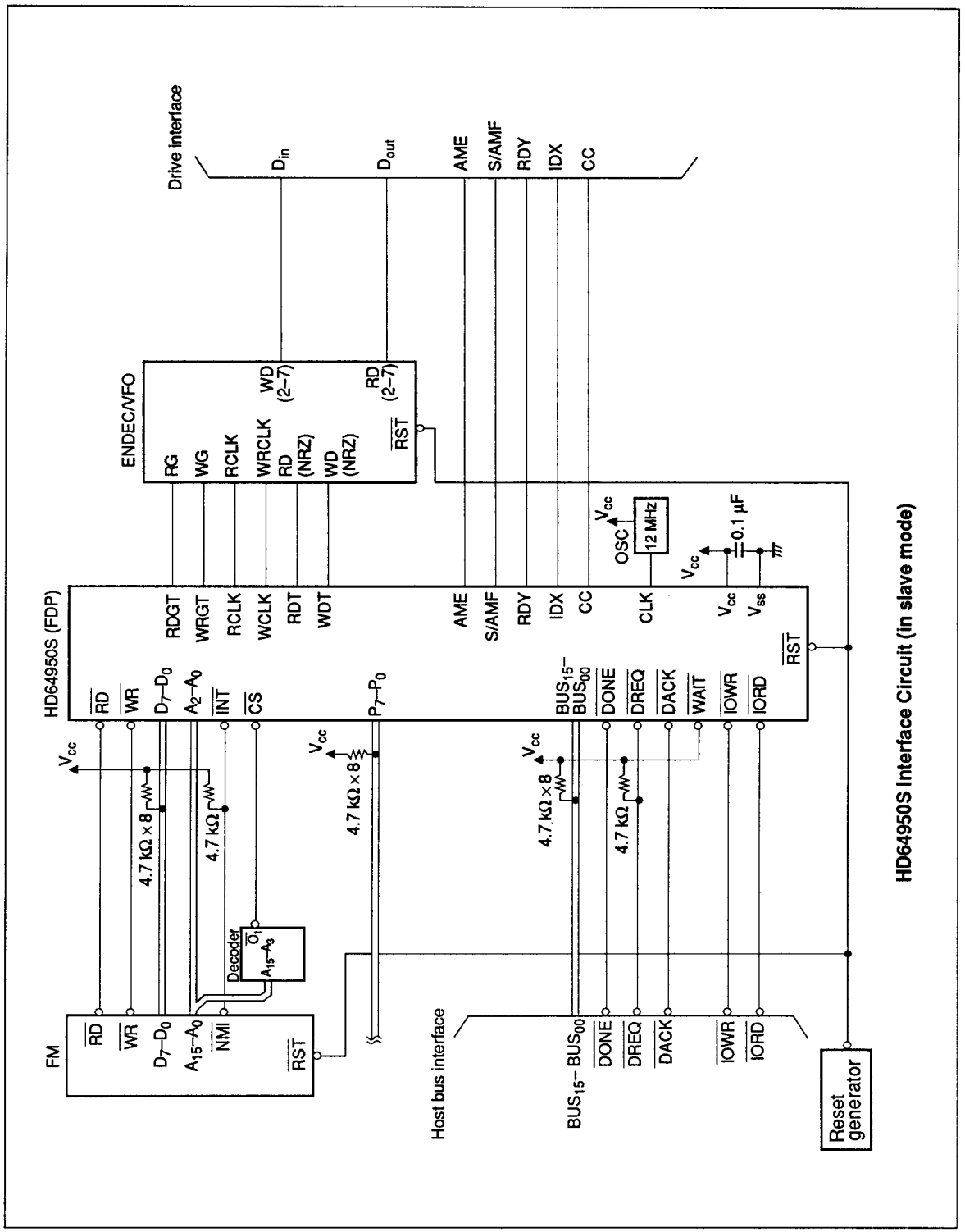
Data Buffer: Stores data transferred between disk drive and host. Also used as external RAM for the file manager.

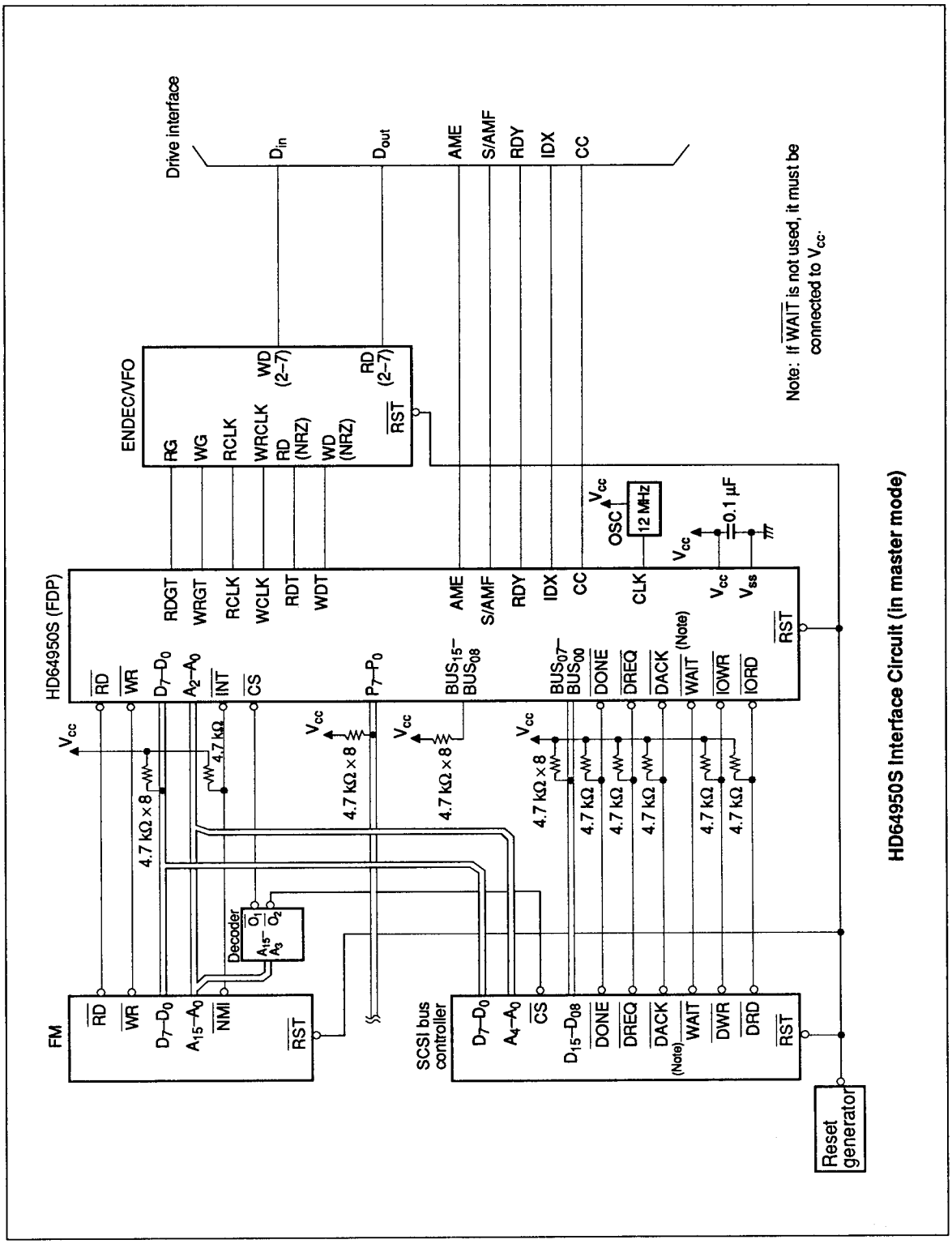
Buffer Control: Switches buffers during command execution to prevent conflicts with accesses from disk drive and host.

Timers: One timer monitors the number of retries and another monitors the index pulse input. Two additional general-purpose timers are available.

Port: I/O data port

Circuit Diagrams





HD64950S Interface Circuit (in master mode)

System Configuration

Figure 1 shows the configuration of a servo drive system containing the HD64950S FDP and controlled by software, and figure 2 shows the bus configurations when the FDP is in slave mode and master mode. In this example, the FM (file manager) is a general-purpose MPU (microprocessing unit) containing software to control the FDP, host bus controller, and drive.

After receiving a command over the host bus, the FM first decodes it to determine if any FDP processing (e.g., read/write of data) is required. If it is required, the FM starts the FDP and issues a command to it to begin the processing. Data is then transferred through the FDP and the host bus controller without FM intervention.

If an ECC error occurs during data read from the drive, the FDP informs the FM. If correction is required, the FM sets the FDP accordingly and issues STECC commands to correct the contents of the data buffer. The FDP supports multi-sector processing, during which time the FM can perform drive control such as servo processing set by software.

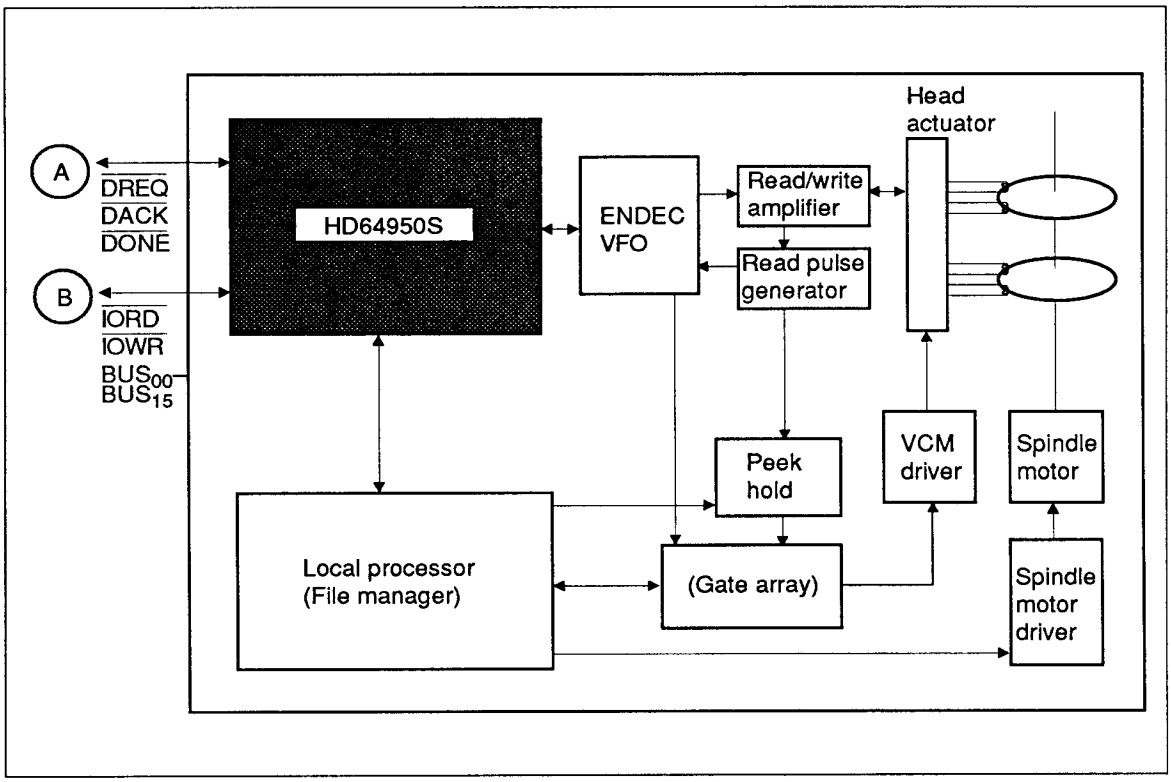


Figure 1 Servo Drive System

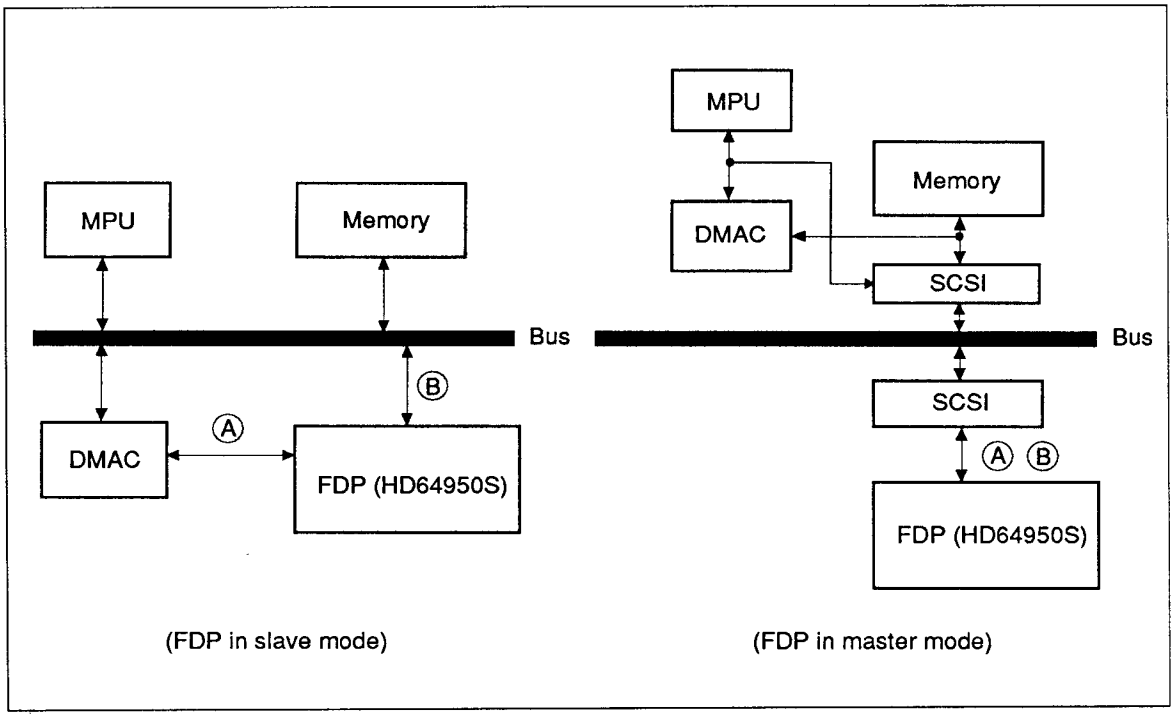


Figure 2 Bus Configurations

FDP Operating Procedure

To operate the FDP, the following preparation is necessary:

1. Set the indirect registers according to the required operation mode, using the direct registers of the file manager interface. Refer to the Registers section for details of direct and indirect registers and their settings.

   During write-formatting or write to the drive without host transfer, data must be stored in the data buffer.

2. Store format commands for the required operation mode in the format buffer. Refer to the Commands section for details of format commands and their storage in the format buffer.

   Note: When format buffer storage is completed, set the desired start address in the indirect register FBP (format buffer pointer).

   When a command other than the START command is issued, it is not necessary to store a format command in the format buffer.

3. Write an ABORT command, then write an FDP start command (START, LOAD, STORE, or STECC) in the direct register CMR (command register).

When these preparations are complete, the FDP will run according to the initialized values and commands. The operating status of the FDP is displayed in the direct register CEND. Determine the completion of an FDP operation by the contents of direct register CEND or by the $\overline{\text{INT}}$ (interrupt) signal.

Registers

The FDP has two kinds of register: direct registers which the file manager (FM) accesses directly with signals A_0 to A_2 and indirect registers which the FM accesses by means of direct registers AR (specifying the address of an indirect register) and DTR (containing data to be written to or read from the indirect register). Direct registers are listed in table 1 and indirect registers are listed in table 2. Refer to the Interface Functions section for details of register settings.

Table 1 Direct Registers

Symbol	Name	Description
AR	Address register	Specifies the address of an indirect register to be read or written
DTR	Data register	Data port used to access a specified indirect register: stores data to be written in a specified indirect register (write operation) or the contents of indirect register AR (read operation)
CMR	Command register	Stores a direct command (START, STECC, LOAD, STORE, or ABORT) for FDP start
CEND	Command end	Stores the operating status of the FDP and operation results (read-only register)
BUF0	Buffer 0	Data port for data buffer 0
BUF1	Buffer 1	Data port for data buffer 1

Table 2 Indirect Registers

Symbol	Name	Description
TCNTH, TCNTL	Transfer counter	Counts the number of bytes sent during DMA transfer (read-only register that loads the value of SRLL and SRLH and decrements them at each transfer)
MDR1–MDR4	Mode register 1 to mode register 4	Specifies the FDP operation mode (host, drive, timer, ECC, etc.)
IDR0–IDR5	ID register 0 to ID register 5	Specifies byte sync patterns and ID field information
DAM0	Data byte sync pattern	Specifies a data byte sync pattern
INCR	Increment register	Specifies automatic incrementation from IDR0 to IDR5
PORT	Data port	General-purpose I/O register
DDR	Data direction register	Specifies input or output for each port bit

Table 2 Indirect Registers (cont)

Symbol	Name	Description
FMTB1, FMTB2, FMTB3	Format buffer 1 to format buffer 3	Stores a format command which specifies the disk format
FBP	Format buffer pointer	Specifies the address of the format buffer to be accessed
RTYP	Retry pointer	Specifies an FBP value for retry (read-only register)
JMPP	Jump pointer	Specifies an FBP value for jumping (read-only register)
LPCNT	Loop counter	Specifies the number of jumps for the format buffer
TCR	Timer control register	Specifies operating periods for the three internal timers (TMA, TMC, and TMD)
RCPH, RCPL	Read correction pointer	Specifies bit location for ECC correction (read-only register)
SRLH, SRLL	Sector length	Specifies the sector data field length
B0PH, B0PL	Buffer 0 pointer	Specifies the address of a unit which read/writes to buffer 0
B1PH, B1PL	Buffer 1 pointer	Specifies the address of a unit which read/writes to buffer 1
ECC0–ECC6	ECC register 0 to ECC register 6	ECC generation/check registers which specify an ECC correction pattern (read-only registers)
PLY0–PLY6	Polynominal 0 to polynominal 6	Specify ECC polynominal
TMA	Timer A	General-purpose timer
TMB	Timer B	Timer for controlling the number of retries
TMC	Timer C	Watch dog timer for IDX pulse
TMD	Timer D	General-purpose timer
STR1, STR2	Status 1, status 2	Indicate the cause of FDP abnormal termination and the status of the disk drive
CMP	Comparator	Compares byte sync patterns and indicates the result (read-only register)
DPAT	Data pattern	Specifies data to be written in data fields (for formatting)
SCNT	Sector counter	Specifies the total number of sectors to be transferred to or from the host

Table 3 Direct Register Addresses and Structures

Address (A_0–A_2)	Register Name	R/W	Data Bits							
			7	6	5	4	3	2	1	0
$00	AR	R/W								
$01	DTR	R/W								
$02	CEND	R	BSY	ERR	HBSY	INT	BFACS		BUSINT	
$03	CMR	R/W						C2	C1	C0
$04	BUF0	R/W								
$05	BUF1	R/W								

Note: ▨ : Expansion bit (set to 0 in writing. Only 0 is read in reading.)

Bit Description

BSY (Busy): Shows that disk drive is being accessed, or error correction processing is being executed. BSY is set when a START or STECC command is stored in the CMR, and is reset when command execution is completed.

ERR (Error): Set when the operation is terminated by an error; reset when data is written in the CMR.

HBSY (Host Busy): Shows that data is being accessed from the host. HBSY is set during a data transfer between the FDP and the host, and is reset when all data has been transferred.

INT (Interrupt): Set by an interrupt caused by the normal completion or abnormal termination of operation; reset when data is written into the CMR.

BFACS (Buffer Access): Shows that data is being accessed between the disk drive and data buffer. BFACS is set during while access, and is reset when all data is transferred.

BUSINT (Bus Interrupt): Shows that the bus was interrupted during data transfer from the host. BUSINT is set when $\overline{\text{DONE}}$ is input during a DMA transfer (in master mode, this bit is set even when no DMA transfer is in progress), and is reset when data is written in the CMR.

Table 4 Indirect Register Addresses and Structures

Address (AR)		Register Name	R/W	Data Bits							
7654 3210	(HEX)			7	6	5	4	3	2	1	0
0000 0000	00	IDR0	R/W								
0000 0001	01	IDR1	R/W								
0000 0010	02	IDR2	R/W								
0000 0011	03	IDR3	R/W								
0000 0100	04	IDR4	R/W								
0000 0101	05	IDR5	R/W								
0000 1000	08	INCR	R/W			IFG5	IFG4	IFG3	IFG2	IFG1	IFG0
0000 1001	09	CMP	R								
0000 1010	0A	TCNTL	R								
0000 1011	0B	TCNTH	R								
0000 1100	0C	SRLL	R/W								
0000 1101	0D	SRLH	R/W								
0001 0000	10	LPCNT	R/W								
0001 0001	11	JMPP	R								
0001 0010	12	RTYP	R								
0001 0011	13	FBP	R/W								
0001 0100	14	STR1	R/W	SNF	IBSNF	DBSNF	VER	CCE	CRFLT	FLG	
0001 0101	15	STR2	R/W	TOA	TOB	TOC	TOD	DR		CC	
0001 1000	18	TCR	R/W	TDC1	TDC0	TCC1	TCC0			TAC1	TAC0
0001 1001	19	TMA	R/W								
0001 1010	1A	TMB	R/W								
0001 1011	1B	TMC	R/W								
0001 1100	1C	TMD	R/W								
0001 1101	1D	MDR3	R/W							FLEN	
0001 1111	1F	RCPL	R								

Table 4 Indirect Register Addresses and Structures (cont)

Address (AR)		Register Name	R/W	Data Bits							
7654 3210	(HEX)			7	6	5	4	3	2	1	0
0010 0000	20	RCPH	R								
0010 0001	21	ECC0	R								
0010 0010	22	ECC1	R								
0010 0011	23	ECC2	R								
0010 0100	24	ECC3	R								
0010 0101	25	ECC4	R								
0010 0110	26	ECC5	R								
0010 0111	27	ECC6	R								
0010 1010	2A	B0PL	R/W								
0010 1011	2B	B0PH	R/W								
0010 1100	2C	B1PL	R/W								
0010 1101	2D	B1PH	R/W								
0010 1110	2E	MDR1	R/W	MST	B8	NIMK	AIMK		DMA	DMAM	DRBF
0010 1111	2F	MDR2	R/W	ECM1	ECM0	ECCI	CRCI	BSEX	ZCK2	ZCK1	ZCK0
0011 0000	30	DDR	R/W								
0011 0001	31	PORT	R/W								
0011 0010	32	PLY0	R/W								
0011 0011	33	PLY1	R/W								
0011 0100	34	PLY2	R/W								
0011 0101	35	PLY3	R/W								
0011 0110	36	PLY4	R/W								
0011 0111	37	PLY5	R/W								
0011 1000	38	PLY6	R/W								
0011 1011	3B	DPAT	R/W								
0011 1100	3C	FMTB1	R/W								

Table 4 Indirect Register Addresses and Structures (cont)

Address (AR)		Register Name	R/W	Data Bits							
7654 3210	(HEX)			7	6	5	4	3	2	1	0
0011 1101	3D	FMTB2	R/W								
0011 1110	3E	FMTB3	R/W								
0011 1111	3F	DAM0	R/W								
0100 0100	44	MDR4	R/W				HS		BURST	BFM	
0100 0101	45	SCNT	R/W								

Notes: 1. ▨: Expansion bit (set to 0 in writing. Only 0 is read in reading.)
2. Only 0 can be written in indirect registers STR1 and STR2.

Bit Description

IFG0–IFG5: Specifies the register (IDR0 to IDR5) to be incremented. For example, IDR0 is incremented after the execution of format command IDINC if IFG0 is set to 1.

SNF (Sector Not Found): Set if no sector is found after the specified number of rotations.

IBSNF (ID Byte Sync Pattern Not Found): Set if the byte sync pattern of the ID field does not match.

DBSNF (Data Byte Sync Pattern Not Found): Set if the byte sync pattern of the data field does not match.

VER (Verify Error): Set if an unmatch error is detected by verify/comparison processing.

CCE (Check Code Error): Set if a CRC/ECC error occurs.

CRFLT (Correction Fault): Set if no ECC error correction pattern or pattern location is detected.

FLG (Flag Error): Set if flags do not match (but CRC/ECC is operating correctly).

TOA–TOD (Timer Overflow A to D): Set if the corresponding timer (TMA to TMD) overflows.

DR (Drive Ready): Set while the RDY signal is input.

CC (Command Complete): Set while the CC signal is input

TAC0/1, TCC0/1, TDC0/1: Specify the decrementation clock cycles of timers TMA, TMC, and TMD. Cycles are determined to be fractions of the CLK frequency, as follows:

(Timer n Control 0/1)

TnC1	TnC0	TMn Input Clock
0	0	$1/2^7$ of source clock (CLK)
0	1	$1/2^{11}$ of source clock (CLK)
1	0	$1/2^{15}$ of source clock (CLK)
1	1	$1/2^{19}$ of source clock (CLK)

FLEN (Flag Enable): Set to 1 when flag check is enabled.

MST (Master Mode): Set to 1 when FDP is in master mode.

B8 (8-Bit Bus Mode): Set to 1 when FDP is in 8-bit host bus mode.

NIMK (Normal Interrupt Mask): Set to 1 if interrupt signals are to be masked at the completion of operation. (START (normal completion or abnormal termination) or STECC (normal completion) command)

AIMK (Abnormal Interrupt Mask): Set to 1 if interrupt signals are to be masked at the termination of operation. (START, STECC, LOAD, or STORE command)

DMA (DMA Enable Mode): Set to 1 if DMA transfer is to be enabled during command execution.

DMAM (DMA Mode): Specifies the switching mode of the data buffer. DMAM is set to 1 during format command execution.

DRBF (Drive Buffer Select): Specifies the data buffer to be used first in DMA transfer. 0 specifies that buffer 0 is used first.

ECM0/1 (ECC Mode 0/1): Specifies the ECC data length, as follows:

ECM1	ECM0	ECC Data Length
0	0	4 bytes
0	1	6 bytes
1	0	7 bytes

ECCI (ECC Initial): Specifies the initial ECC value. 0 specifies that the initial value is all zeros; 1 specifies that it is all ones.

CRCI (CRC Initial): Specifies the initial CRC value. 0 specifies that the initial value is all zeros; 1 specifies that it is all ones.

BSEX (Byte Sync Pattern Excluded): Set to 0 if no byte sync pattern is to be included in the ECC/CRC operation.

ZCK0/1/2 (Zero Check 0/1/2): Specifies the maximum correctable burst error length, as follows:

ZCK2	ZCK1	ZCK0	Max Correctable Burst Error Length
0	0	0	16 bits
0	0	1	11 bits
0	1	0	10 bits
0	1	1	9 bits
1	0	0	8 bits
1	0	1	7 bits
1	1	0	6 bits
1	1	1	5 bits

HS (Hard Sector): Specifies the drive interface mode. 0 specifies soft sector mode; 1 specifies hard sector mode.

BURST (Burst Mode): Specifies the transfer mode. 0 specifies cycle steal mode; 1 specifies burst mode.

BFM (Buffer Mode): Specifies the data buffer mode. 0 specifies that two data buffers are used, 1 specifies that one is used.

Direct Registers

The direct registers are described below.

Address Register (AR)

Bit Configuration

AR Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	00	$00
	Indirect register address									

Function

The AR latches the address of the indirect register to be accessed by the DTR.

Note: Do not specify an address which is not assigned to an indirect register.

Data Register (DTR)

Bit Configuration

DTR Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	Any	$01
Contents of specified indirect register										

Function

The DTR acts as a data port for accessing an indirect register. It stores data which is to be written to or has been read from the indirect register specified by the AR.

Command End (CEND)

Bit Configuration

CEND Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R	00	$02
BSY	ERR	HBSY	INT	BFACS		BUSINT				

Bit Description

BSY (Busy): Is high (1) when a START or STECC command is being executed, to indicate that accessible registers are limited. Refer to the File Manager Interface section for further details.

ERR (Error): Goes high (1) when a command is terminated abnormally or a timer overflows. ERR goes high under any of the following conditions:

- When an error occurs and the SNF, IBSNF, DBSNF, VER, CCE, CRFLT, or FLG bit of status register 1 (STR1) goes high.
- When a timer overflows and the TOA, TOB, TOC, or TOD bit of status register 2 (STR2) goes high.
- When $\overline{\text{DONE}}$ is input during a DMA transfer and the DMA transfer is aborted.
- When the FDP is in master mode, ERR is set whenever $\overline{\text{DONE}}$ is input

To reset ERR to low, clear the high bits in STR1 or STR2 and issue a direct command.

HBSY (Host Busy): Goes high (1) when DMA transfer is performed, to indicate that accessible registers are limited. Refer to the File Manager Interface section for further details. HBSY goes high (1) under any of the following conditions:

- When a LOAD or STORE command is issued while the DMA bit of MDR1 is 1.
- When a DMAW command is executed while the DMA bit of MDR1 is 1.
- When a WIDB, WDTB, RIDB, RDTB, or VDTB command is executed while the DMA bit of MDR1 is 1, and data transfer between the data buffer and disk drive has started or verification of data has started.

HBSY is reset under any of the following conditions:

- When the specified amount of data (SRLH/SRLL × SCNT) has been transferred.

  Note: HBSY is reset within 1.5 CLK after $\overline{\text{IOWR}}$ or $\overline{\text{IORD}}$ of the last data transfer is asserted. Therefore, the HBSY bit may be reset while $\overline{\text{IOWR}}$ or $\overline{\text{IORD}}$ is still asserted.

- When $\overline{\text{DONE}}$ is asserted.
- When a disk access error (which sets the SNF, IBSNF, DBSNF, VER, CCE, or FLG bit of STR1, or the TOB or TOC bit of STR2) occurs.
- When the ABORT command is issued.

INT (Interrupt): Indicates an interrupt caused by the completion of command execution. Interrupts can be masked in different ways at normal completion and abnormal termination by the NIMK and AIMK bits of MDR1.

- INT set by completion of command execution INT is set when START, STECC command execution is completed.
- INT set by abnormal termination of command execution INT is set when START, STECC, LOAD, or STORE command execution is terminated abnormally by an error which sets the ERR bit of CEND.

INT is cleared when another command is issued.

BFACS (Buffer Access): Set to 1 during data transfer between data buffer and drive, or during data verification. BFACS is set only during execution of the WIDB, WDTB, RIDB, or VDTB command.

BUSINT (Bus Interrupt): Set to 1 when DMA transfer between host and data buffer is terminated by $\overline{\text{DONE}}$ input, except when $\overline{\text{DONE}}$ is asserted during the transfer of the last of the specified amount of data (set by SRLL/SRLH × SCNT) when the FDP is in slave mode (when the MST bit of MDR1 is 0).
When the FDP is in master mode, BUSINT is set whenever $\overline{\text{DONE}}$ is input.

BUSINT is cleared when another command is issued.

Command Register (CMR)

Bit Configuration

CMR Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	00	$03
					C2	C1	C0			

Function

The CMR enables direct command activation by specifying the command code of the START, STECC, LOAD, or STORE command for execution. The specified direct command can be issued only when both the BSY and HBSY bits of CEND are low (0).

To abort the execution of the specified direct command, issue the ABORT command. (The ABORT command can be issued regardless of the BSY and HBSY values.) The direct commands codes are listed below. For details on direct command functions, refer to the Direct Commands section.

Command	Command Code
START	$01
STECC	$02
LOAD	$04
STORE	$06
ABORT	$00

Notes: 1. Do not use any other command codes.
2. Make sure that the contents of status registers STR1 and STR2 are cleared before writing a command (except the ABORT command) code to the CMR.

Buffer Registers 0 and 1 (BUF0, BUF1)

Bit Configuration

BUF0 Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	00	$04
Buffer 0 data										

BUF1 Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	00	$05
Buffer 1 data										

Function

BUF0 and BUF1 act as data ports. They access data from the data buffer address specified by the buffer pointer.

Note: Always set the buffer pointer before accessing the buffer.

Indirect Registers

The main indirect registers are described below.

Status Register 1 (STR1)

Bit Configuration

STR Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	00	$14
SNF	IBSNF	DBSNF	VER	CCE	CRFLT	FLG				

Function

STR1 indicates the result of a command's execution. If a command is terminated because of an error, the appropriate data bit of STR1 is set to 1. However, if retry succeeds, no data bits are set.

The set bits can be reset to 0 by the FM writing 0 to them. Conditions for setting the data bits in STR1 vary with commands. Refer to the Commands section for details.

Note: Under some conditions, set bits may be reset to 0 by writing 1 to them.

Bit Description

SNF (Sector not found)

Set When	Commands
No sector is detected • IDX and SCT are not input • ID byte sync pattern is not detected	WPUIS, RIDB, WABSC, RIDS, WTISA, VID
The buffer cannot be accessed by the disk drive. (Either the IBSNF or DBSNF bit is also high)	WIDB, WDTB, RIDB, RDTB, VDTB

IBSNF (ID Byte Sync Pattern Not Found)

Set When	Commands
ID byte sync pattern is not detected	RIDB, RIDS, VID
The buffer cannot be accessed by the disk drive (both the SNF and DBSNF bits are also high)	WIDB, WDTB, RIDB, RDTB, VDTB

DBSNF (Data Byte Sync Pattern Not Found)

Set When	Commands
Data byte sync pattern is not detected	RDTB, RDTS, VDTB
The buffer cannot be accessed by the disk drive (both the SNF and IBSNF bits are also high)	WIDB, WDTB, RIDB, RDTB, VDTB

VER (Verify Error)

Set When	Commands
Drive data does not match the contents of the buffer or the IDR	VID, VDTB

CCE (Check Code Error)

Set When	Commands
A CRC or ECC error occurs	CCRC, CECC, VCRC, VECC, VCRCR, VECCR, VCRCW, VECCW

CRFLT (Correction Fault)

Set When	Commands
No ECC error correction pattern was obtained	STECC

FLG (Flag Error)

Set When	Commands
The flag bytes do not match (but no CRC or ECC error has occurred)	VCRC, VECC, VCRCR, VECCR, VCRCW, VECCW

Status Register 2 (STR2)

Bit Configuration

STR2 Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	00Note	$15
TOA	TOB	TOC	TOD	DR		CC				

Note: The DR and CC bits are set by inputs from the FDP's pins.

Function

STR2 indicates the status of timers and input pins.

Bit Description

TOA, TOB, TOC, and TOD (Timer Overflows A, B, C, and D): Set when the corresponding timer (TMA, TMB, TMC, and TMD) overflows. The set bits can be reset by the FM writing 0 to them, but in some cases they may be reset by writing 1 to them.

DR (Drive Ready) and CC (Command Complete): Indicate the status of the RDY and CC pins, as follows:

RDY Pin	DR (STR2)
Low	0
High	1

CC Pin	CC (STR2)
Low	0
High	1

Mode Register 1 (MDR1)

Bit Configuration

MDR1 Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	00	$2E
MST	B8	NIMK	AIMK		DMA	DMAM	DRBF			

Function

MDR1 controls data transfers and interrupts.

Bit Description

MST (Master Mode): Specifies the mode of the FDP.

- MST = 0: The FDP functions as a slave and the DMAC controls the DMA transfer.
- MST = 1: The FDP functions as the master, controlling the DMA transfer. Only bus lines BUS_{00} to BUS_{07} are used (8-bit data bus); bus lines BUS_{08} to BUS_{15} should be pulled up.

Refer to the Host Interface section for further details of master details of master and slave modes.

B8 (8-Bit Bus Mode): Specifies the width of the host interface bus. When the FDP is in master mode, B8 must always be 1 (8-bit data bus). When the FDP is in slave mode, B8 specifies the width of the host interface bus.

- B8 = 0: 16-bit data bus
- B8 = 1: 8-bit data bus (Only bus lines BUS_{00} to BUS_{07} are used; bus lines BUS_{08} to BUS_{15} should be pulled up.)

NIMK (Normal Interrupt Mask): Specifies whether interrupts are to be masked at the completion of a command. Setting this bit affects the INT bit of the CEND register and the $\overline{INT}$ signal.

- NIMK = 0

  Interrupts are not masked at the completion of a command. If the execution of START command is completed, the FDP sets the INT bit of the CEND register and asserts the $\overline{INT}$ signal. If the execution of a STECC command is completed, the FDP sets the INT bit of the CEND register and asserts the $\overline{INT}$ signal.

- NIMK = 1

  Interrupts are masked at the completion of a command. The FDP dose not set the INT bit of CEND and does not assert the $\overline{INT}$ signal, even when the execution of the START command completed. The FDP does not set the INT bit of CEND and does not assert the $\overline{INT}$ signal, even when the execution of the STECC command is completed.

AIMK (Abnormal Interrupt Mask): Specifies whether interrupts are to be masked at the abnormal termination of a command. Setting this bit affects the INT bit of CEND and the $\overline{INT}$ signal.

- AIMK = 0

  Interrupts are not masked at the abnormal termination of a command. If an error which would set the ERR bit of CEND is detected during the execution of a START, STECC, LOAD, or STORE command, the FDP sets the INT bit of CEND and asserts the $\overline{INT}$ signal.

- AIMK = 1

  Interrupts are masked at the abnormal termination of a command. The FDP does not set the INT bit of CEND and does not assert the $\overline{INT}$ signal, even when an error which would set the ERR bit of the CEND register is detected during the execution of the START, STECC, LOAD, or STORE command.

DMA (DMA Enable Mode): Specifies whether the FDP can start DMA transfer during the execution of a command. Set DMA to 1 to enable DMA transfers to and from the host.

- DMA = 0: Disables DMA transfer, even when a command which starts DMA transfer is executed.
- DMA = 1: Enables DMA transfer for commands which request DMA transfer.

Commands which request DMA transfer are:
- Format commands such as DMAW, WIDB, WDTB, RTDB, and VDTB
- LOAD and STORE commands

DMAM (DMA Mode): Specifies the data buffer switching mode.

- DMAM = 0

  Data buffers are switched each time one sector is transferred. In this mode, the BFM bit of MDR4 must be specified as follows, according to sector size:

  BFM = 0: Sector size of 4 to 544 bytes
  BFM = 1: Sector size of 545 to 1088 bytes

- DMAM = 1

  Data buffers are switched when address 543 is accessed. This switch has no connection with the value of the BFM bit of MDR4.

If two or more format commands that access buffers are executed, each access begins at the address after the address accessed by the preceding command.

Note: DMAM = 1 is valid only for the execution of a format command. In this mode, the DMA bit of MDR1 must be 0.

Refer to the Data Transfer Function section for more details.

DRBF (Drive Buffer Select): Selects the data buffer to be used first for DMA transfer.

- DRBF = 0: BUF0 is used first.
- DRBF = 1: BUF1 is used first.

DRBF is valid as soon as it is set, but it only indicates the data buffer to be used first; not the buffer currently being used.

Mode Register 2 (MDR2)

Bit Configuration

MDR2 Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	00	$2F
ECM1	ECM0	ECCI	CRCI	BSEX	ZCK2	ZCK1	ZCK0			

Function

MDR2 specifies details relating to ECC/CRC.

Bit Description

ECM1 and ECM0 (ECC Modes 1 and 0): Specify the ECC byte length to be used in the execution of an ECC format command or STECC command, as follows:

ECM1	ECM0	Number of ECC Bytes
0	0	4
0	1	6
1	0	7
1	1	Reserved

ECCI (ECC Initial): Specifies the initial ECC value to be used by an ECC-initializing format command.

- ECCI = 0: The initial ECC value is all zeros.
- ECCI = 1: The initial ECC value is all ones.

CRCI (CRC Initial): Specifies the initial CRC value to be used by a CRC-initializing format command.

- CRCI = 0: The initial CRC value is all zeros.
- CRCI = 1: The initial CRC value is all ones.

BSEX (Byte Sync Pattern Excluded): Specifies whether a format command that performs an ECC or CRC operation includes a byte sync pattern.

- BSEX = 0: A byte sync pattern is included in the ECC or CRC operation.
- BSEX = 1: No byte sync pattern is included in the ECC or CRC operation.

ZCK2, ZCK1, and ZCK0 (Zero Checks 2, 1, and 0): Specify the maximum length of burst error that can be corrected during the calculation of an ECC error correction pattern by the STECC command.

ZCK2	ZCK1	ZCK0	Maximum Correctable Burst Error Length
0	0	0	16 bits
0	0	1	11 bits
0	1	0	10 bits
0	1	1	9 bits
1	0	0	8 bits
1	0	1	7 bits
1	1	0	6 bits
1	1	1	5 bits

Refer to the Error Checking and Correction Function section for details of ECC and CRC operations.

Mode Register 3 (MDR3)

Bit Configuration

MDR3 Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	00	$1D
						FLEN				

Function

MDR3 specifies details relating to disk formatting.

Bit Description

FLEN (Flag Enable): Indicates if flag comparison is required with the VID command.

- FLEN = 1

  The contents of IDR5 are checked as a flag byte during the execution of a VID command. If only the flag bytes do not match (but all other information matches and no ECC/CRC error occurs), the FDP sets the FLG bit of STR1 and the ERR bit of CEND and completes the START command execution.

  A verification ECC/CRC command that checks for flag byte errors must be executed after a CRC/ECC check, so the VID command must be followed by a VCRC, VCRCR, VCRCW, VECC, VECCR, or VECCW command.

- FLEN = 0: No flag comparison is performed

Mode Register 4 (MDR4)

Bit Configuration

MDR4 Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	00	$44
			HS		BURST	BFM				

Function

MDR4 specifies the type of disk drive and other details relating to data transfer.

Bit Description

HS (Hard Sector): Specifies the type of disk drive and affects the sequence of WAFSA and WTISA commands.

- HS = 0: Soft sector mode
- HS = 1: Hard sector mode

BURST (Burst Mode): Specifies the DMA transfer mode when the FDP is in slave mode. The DMA transfer mode in FDP master mode is always burst mode.

- BURST = 0 (cycle steal mode)

  In this mode, the FDP negates $\overline{\text{DREQ}}$ when $\overline{\text{DACK}}$ is asserted in response to the assertion of $\overline{\text{DREQ}}$. The FDP continues to negate $\overline{\text{DREQ}}$ while $\overline{\text{DACK}}$ is asserted, and it continues to assert $\overline{\text{DREQ}}$ until the DMA transfer is completed (all sectors are transferred) or aborted ($\overline{\text{DONE}}$ is input, ABORT command is issued, or drive error occurs).

- BURST = 1 (burst mode)

  In this mode, the FDP continues to assert $\overline{\text{DREQ}}$ until a DMA transfer is completed (all sectors are transferred) or aborted ($\overline{\text{DONE}}$ is input, ABORT command is issued, or drive error occurs), or until buffers are switched.

BFM (Buffer Mode): Specifies the mode of the data buffer during DMA transfer. BFM must be specified when DMAM = 0 and DMA = 1.

- BFM = 0: Sector length is 4 to 544 bytes (two-data-buffer mode)
- BFM = 1: Sector length is 545 to 1088 bytes (one-data-buffer mode)

Format Buffer Pointer (FBP)

Bit Configuration

FBP Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	00	$13
			Format buffer pointer							

Function

The FBP holds the format buffer address accessed by the FM or the START command's start address. It is updated by any of the following causes:

- When the START command is not being executed (BSY = 0) and:
  - The FM writes to the FBP,
  - The FM reads the 3 bytes of FMTB1 to FBTB3 (in any order), Note or
  - The FMN writes the 3 bytes of FMTB1 to FBTB3 (in any order). Note

  Note: The FBP is incremented automatically.
- When the START command is being executed (BSY = 1) and:
  - The contents of JMPP are loaded at the end of the execution of a format command specifying JMP,
  - The contents of RTYP are loaded into the FBP when a retry occurs during the execution of a format command specifying JRSR or RTY, or
  - The FBP is incremented when any other format command is completed.

Format Buffer Registers 1, 2, and 3 (FMTB1, FMTB2, and FMTB3)

Bit Configuration

FMTB1 Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	Any	$3C
Format buffer data										

FMTB2 Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	Any	$3D
Format buffer data										

FMTB3 Data Bits								R/W	Initial Value	Address
7	6	5	4	3	2	1	0	R/W	Any	$3E
Format buffer data										

Function

FMTB1, FMTB2, and FMTB3 act as a port through which the FM accesses the format buffer. The format buffer pointer (FBP) must be set before these registers are accessed.

Writing to the Format Buffer: The FM sets data indirect registers FMTB1, FMTB2, and FMTB3 before writing the data to the format buffer, then increments the FBP.

Reading from the Format Buffer: When the FBP is set, data is loaded from the format buffer to the format buffer registers (FMTB1, FMTB2, and FMTB3). The FM reads the 3 bytes of the contents of the format buffer registers, then increments the FBP.

In both read and write, the monitoring of access to the format buffers (FMTB1 to FMTB3) is cleared when a value is specified in the FBP. If the FBP value is specified again after data has been read from or written to FMTB1 and FMTB2, that data must be read from or written to FMTB1, FMTB2, and FMTB3 again. Otherwise, the format buffer cannot be accessed.

The FM can access FMTB1, FMTB2, and FMTB3 in any order.

Commands

The FDP supports direct commands, format commands, and supplementary commands.

A direct command starts or ends FDP operations and must be written in the command register (CMR) for its execution.

A format command controls disk formatting. Format commands are set in the format buffer by indirect registers FMTB1, FMTB2, and FMTB3, and are started by a START command.

A supplementary command controls jumps and retrys.

Direct Commands

When a command code is written in the command register (CMR), the FDP executes the corresponding direct command. Before a direct command is issued, the command's operating conditions must be specified in the relevant registers.

Except for the ABORT command, direct commands can be issued only when both the BSY and HBSY bits of CEND are 0. (The ABORT command can be issued at any time.)

Only START, STECC, LOAD, STORE, and ABORT commands can be written in the CMR. These direct commands are listed in table 5 and further details are given below.

Table 5 Direct Commands

Direct Command Name	Command Code								Function
	7	6	5	4	3	2	1	0	
START						0	0	1	Starts the execution of the format commands stored in the format buffer. The START command ends when an END command is read from the format buffer or if the START command is terminated abnormally.
STECC						0	1	0	Obtains the error correction pattern and the error bit location from the error syndrome left in the ECC registers after an ECC error. The STECC command does not correct errors.
LOAD						1	0	0	Transfers data from data buffer to the host (DMA transfer).
STORE						1	1	0	Transfers data from the host to the data buffer (DMA transfer).
ABORT						0	0	0	Stops the execution of the four direct commands listed above.

START (Start)

Function

The START command executes a series of format commands stored in the format buffer registers FMTB1, FMTB2, and FMTB3, starting from the command at the address specified by the FBP to the END command. During data transfer, the FDP sets bit 7 (BSY) of CEND while it is processing drive data and bit 5 (HBSY) while it is performing DMA transfer.

Registers Available

FMTB1, FBTB2, FMTB3, LPCNT, FBP, and registers required for the execution of the specified format commands.

STECC (Start ECC)

Function

The STECC command obtains the ECC correction pattern and error bit locations. (Refer to the Error Correction Function section for details of ECC correction patterns and error bit locations.) If the FDP fails to obtain the correction pattern and bit locations, it sets the CRFLT bit of STR1.

The FDP sets bit 7 (BSY) of CEND while this command is executing.

Registers Available

PLY0, PLY1, PLY2, PLY3, PLY4, PLY5, PLY6

LOAD (Load from Buffer)

Function

The LOAD command transfers data from the internal data buffer to the host (DMA transfer), provided that all the following conditions are satisfied:

- The transfer start data buffer is specified by DRBF (MDR1).
- The transfer start address is specified by B0PL/B0PH or B1PL/B1PH.
- The byte length to be transferred is specified by SRLL/SRLH.
- The number of sectors to be transferred is specified by SCNT.

The FDP sets bit 5 (HBSY) of CEND while this command is executing.

Registers Available

B0PH, B0PL, B1PH, B1PL, SRLH, SRLL, SCNT

STORE (Store to Buffer)

Function

The STORE command transfers data from the host to the internal data buffer (DMA transfer), provided that all the following conditions are satisfied:

- The transfer start data buffer is specified by DRBF (MDR1).
- The transfer start address is specified by B0PL/B0PH or B1PL/B1PH.
- The number of sectors to be transferred is specified by SRLL/SRLH.
- The number of transfers is specified by SCNT.

The FDP sets bit 5 (HBSY) of CEND while this command is executing.

Registers Available

B0PH, B0PL, B1PH, B1PL, SRLH, SRLL, SCNT

ABORT (Abort)

Function

The ABORT command terminates the currently executing command.

Format Commands

Format commands must be stored in the format buffer before they can be executed. Once stored, format commands can be executed repeatedly until the commands in the format buffer are updated.

The FDP interprets 3-byte format commands. A format command consists of a command, supplementary command (sup CMD), byte length, and pattern. The functions of these components of the format command are given in table 6.

Table 6 Functions of Format Command Components

Item	Function
Command	Specifies how the disk drive interface is controlled
Supplementary Command	Controls the execution of the command (e.g., it sets jump and/or retry)
Byte length	Specifies the length of the command in bytes (some commands specify the length of execution after a specific condition is satisfied, such as after IDX/SCT is input)
Pattern	Specifies the data pattern to be output by the current or the next command

When the START command is issued, the FDP interprets and executes the command at the address specified by the format buffer pointer (FBP). When command execution is completed, the FBP is incremented unless a jump or retry is specified. The FDP then interprets and executes the command at the next address specified by the FBP. In this way, the FDP interprets and executes commands in sequence until the END command is executed or an error occurs.

To control jumps and retrys the format buffer pointer (FBP), jump pointer (JMPP), and retry pointer (RTYP) can be set by the supplementary command. Refer to the Supplementary Commands section for details.

Format commands can be classify into three types: type A, type B, and type C.

Type A Command

FMTB	Bits							
	7	6	5	4	3	2	1	0
1	Command					Sup CMD		
2	Pattern							
3	Byte length							

- Type A commands write predetermined data and ECC or CRC data.
- Byte length specifies the number of bytes (4 to 255 bytes) to be written. Do not specify a value of less than 4 in this field.

Type B Command

FMTB	Bits							
	7	6	5	4	3	2	1	0
1	Command					Sup CMD		
2						Byte length (high)		
3	Byte length (low)							

- Type B commands write, read, and verify data in ID fields and data fields.
- Byte length specifies the number of bytes (4 to 2047 bytes) to be processed. Do not specify a value of less then 4 in this field.

Type C Command

FMTB	Bits							
	7	6	5	4	3	2	1	0
1	Command					Sup CMD		
2	0	0	0	0	0	0	0	0
3	Byte length							

- Type C commands wait for the IDX/SCT signal, read ECC and CRC data, and execute DMA transfer.
- Byte length specifies the number of bytes (4 to 255 bytes) to be processed. Do not specify a value of less than 4 in this field.

Table 8 Format Command Functions

Mnemonic	Command Code	Function
WIDB	10111	Writes the byte sync pattern set in IDR0 and the predetermined ID information set in the data buffer
WDTB	11000	Writes the byte sync pattern set in DAM0 and the predetermined data set in the data buffer
WDTP	11001	Writes the byte sync pattern set in DAM0 and the data pattern set in DPAT
WCRC	00100	Writes 2-byte-long CRC data
WECC	00101	Writes ECC data of the length (in bytes) set in MDR2
WPTN	00001	Writes the value set in FMTB2
WSP	00110	Negates one byte of the WRGT signal, then writes the value set in FMTB2 (used to write a write splice and the following sync field)
WPUIS	00010	Writes the value set in FMTB2 until the FDP receives the IDX or SCT signal
WPAI	00011	Writes the value set in FMTB2, after the FDP receives the IDX signal
WABSC	00111	Writes the value set in FMTB2 until the FDP receives the SCT signal (valid in hard sector mode only)
WAFSA	01000	• Writes the value set in FMTB2, after the FDP receives the SCT signal (valid in hard sector mode only) • Outputs the AME signal, and writes the value set in FMTB2 (valid in soft sector mode only)
RIDB	11010	Compares a byte sync pattern with the one stored in IDR0, and sets ID information in the buffer if the patterns match
RIDS	11011	Compares a byte sync pattern with the one stored in IDR0, and skips ID information if the patterns match
RDTB	11100	Compares a byte sync pattern with the one stored in DAM0, and sets data in the buffer if the patterns match
RDTS	11101	Compares a byte sync pattern with the one stored in DAM0, and skips data if the patterns match
CCRC	01100	Performs a CRC check
CECC	10000	Performs an ECC check
VID	01011	Compares a byte sync pattern and ID information with values set in ID registers IDR0 to IDRn

Table 8 Format Command Functions (cont)

Mnemonic	Command Code	Function
VDTB	10100 ▨	Compares a byte sync pattern with the one stored in DAM0, and compares data with the contents of the buffer if the patterns match
VCRC	01101 ▨	Compares the last two bytes in the ID or data field, and performs a CRC check and a flag check
VCRCR	01110 ▨	Compares the last two bytes in the ID or data field, performs a CRC check and a flag check, and asserts the RDGT signal from CRC data
VCRCW	01111 ▨	Compares the last two bytes in the ID or data field, performs a CRC check and a flag check, negates the RDGT signal and asserts the WRGT signal at the end of CRC data
VECC	10001 ▨	Compares the last two bytes in the ID or data field, and performs an ECC check and a flag check
VECCR	10010 ▨	Compares the last two bytes in the ID or data field, performs an ECC check and a flag check, and asserts the RDGT signal from ECC data
VECCW	10011 ▨	Compares the last two bytes in the ID or data field, performs an ECC check and a flag check, and negates the RDGT signal and asserts the WRGT signal at the end of ECC data
WAITI	10101 ▨	Waits until the FDP receives the IDX signal
WTISA	10110 ▨	• Waits until the FDP receives the IDX or SCT signal (valid only in hard sector mode) • Asserts the AME signal and waits until the FDP receives the AMF signal (valid only in soft sector mode)
SKIP	11110 ▨	Waits
SKIRG	01001 ▨	Asserts the RDGT signal and waits
DMAW	01010 ▨	Starts DMA transfer of data from the host to the buffer
IDINC	11111 ▨	Increments the value (ID field) of the ID register (IDR0 to IDR5) specified by INCR
END	00000 ▨	Terminates the processing of format commands, and waits for the input of a direct command

Note: ▨: Position of supplementary command

Supplementary Commands

Supplementary commands (Sup CMDs) set the jump and/or retry pointer(s) and execute jump and/or retry operation(s). These commands are listed in table 9.

Table 9 Supplementary Commands (Sup CMDs)

Mnemonic	Bits 2 1 0	Function
NOP	0 0 0	Does nothing
JPS	0 0 1	Sets the jump pointer (JMPP)
RPS	0 1 0	Sets the retry pointer (RTYP)
JRS	0 1 1	Sets the jump pointer and retry pointer
JRSR	1 0 0	Sets the jump pointer and retry pointer, and retries
JMP	1 0 1	Jumps
RTY	1 1 0	Retries
NOP	1 1 1	Does Nothing

Jump Pointer (JMPP) Setting: The FDP sets the value of the FBP into the JMPP during the current command execution.

Retry Pointer (RTYP) Setting: The FDP sets the value of the FBP into the RTYP during the current command execution.

Jump: If the value in the loop counter (LPCNT) is not 0, the FDP sets the JMPP value into the FBP to cause a jump corresponding to that value. The LPCNT is decremented after its value is checked.

Retry: If data does not match during the execution of a verify command, the FDP sets the RTYP value into the FBP to cause a jump corresponding to that value.

The JMPP specifies the start address of a series of format commands set in an execution loop to process two or more sectors. To set a jump address in the JMPP, JPS, JRS, or JRSR must be specified in the supplementary command field of each format command. JMP must be set in the supplementary command field of the last format command of the loop.

The LPCNT specifies the number of repetitions of the loop set by JMPP. (This value must be set by the FM in advance.)

The RTYP specifies the start address of the execution loop to be restarted at the detection of a sector pulse after ID data does not match. To set an address in the RTYP, RPS, JRS, and JRSR must be specified in the supplementary command field of the format command.

To set the FBP at the start loop address pointed to by the RTYP, RTY or JRSR must be specified in the supplementary command field of the format command to be retried.

The processing specified by a supplementary command is jump processing and/or retry processing.

Supplementary commands for jump processing are Jump Pointer Set (JPS) which sets a jump destination address and Jump (JMP) which jumps at normal termination to the address specified by the jump pointer (JMPP).

Supplementary commands for retry processing are Retry Pointer Set (RPS) which sets a retry destination address and Retry (RTY) which jumps to the address specified by the retry pointer (RTYP) if an error occurs.

Other supplementary commands combine the above command functions.

For example, the first processing in the reading of or writing to a sector is the verification (by the VID command) of the ID field. In multi-sector processing, this verification must be performed for each sector, and, if data in the ID field does not match, the operation must be retried. The supplementary command JRSR (which sets the JMPP and RTYP and enables retry) simplifies the retry.

Similarly, the first processing in hard sector mode is to wait for a sector pulse (by the WTISA command). This process is performed for each sector. If ID data does not match in the succeeding ID verification, the operation must be restarted from the wait process. In such a case, the supplementary command JRS (which sets JMPP and RTYP) simplifies the process.

Example 1: Loop processing

LPCNT = N

	Address	
	n	CMD
JMPP ←	n + 1	CMD (JPS, JRS, JRSR)
		.
		.
	m	CMD (JMP)

If any one of JPS, JRS, and JRSR is specified in the supplementary command field of the format command at address n + 1, the FDP sets address n + 1 in the JMPP. If the format command at address m has JMP specified in its supplementary command field, the FDP jumps to the specified address (n + 1) every time it encounters that command, until the LPCNT reaches 0. The LPCNT is decremented by each jump. In this example, N + 1 loops from address n + 1 to address m are performed.

Example 2: Retry processing

	Address	
	n	CMD
RTYP ←	n + 1	CMD (RPS, JRS, JRSR)
		CMD (RTY)
		.
		.
	m	CMD

If any one of RPS, JRS, and JRSR is specified in the supplementary command field of the format command at address n + 1, the FDP sets address n + 1 in the RTYP.

If a retry is instructed (after an error occurs during a format command that has RTY or JSRS specified in its supplementary command field), the contents of the RTYP are loaded into the FBP, and processing resumes from the address specified by the RTYP.

Interface Functions

Drive Interface

The status of the HS bit of the indirect register MDR4 is determined by the interface specification of the disk drive (hard sector mode or soft sector mode), as follows:

HS = 1 for hard sector mode
HS = 0 for soft sector mode

File Manager Interface

The FDP has two types of register: direct registers which the file manager can directly access and indirect registers which the file manager accesses via the direct registers. (Refer to the Block Diagram.)

Accessing Direct Registers: There are six direct registers: address register (AR), data register (DTR), command end register (CEND), command register (CMR), and buffer register 0 and 1 (BUF0 and BUF1). CEND is a read-only register but the other five registers can be read from and written to.

- Accessing AR, DTR, CEND, and CMR

  Set the address of the target direct register in signals A_2 to A_0, and assert $\overline{CS}$ and $\overline{WR}$ to write to the register or asserts $\overline{CS}$ and $\overline{RD}$ to read from the register. The register stores data at the rising edge of $\overline{WR}$ (write), or outputs data when $\overline{RD}$ is asserted (read).

- Accessing BUF0 and BUF1

  Set the address of the target direct register (BUF0 or BUF1) in signals A_2 to A_0, and assert $\overline{CS}$ and $\overline{WR}$ to write to the register or assert $\overline{CS}$ and $\overline{RD}$ to read from the register. In addition, the register address for buffer pointers (B0PL/B0PH and B1PL/B1PH) must be specified before the register is accessed. Refer to the next section for details on setting indirect registers B0PL/B0PH and B1PL/B1PH.

The addresses of the direct registers are listed in table 10.

Table 10 Direct Register Address

A2	A1	A0	Register	R/W	Remarks
0	0	0	AR	R/W	
0	0	1	DTR	R/W	
0	1	0	CEND	R	
0	1	1	CMR	R/W	
1	0	0	BUF0	R/W	Set the address of this register in the buffer pointer before accessing it, and increment the value of the buffer pointer afteraccessing it
1	0	1	BUF1	R/W	Set the address of this register in the buffer pointer before accessing it, and increment the value of the buffer pointer after accessing it

Accessing Indirect Registers: Write the target indirect register address in the address register (AR). Set the data to be written to or read from the indirect register in the data register (DTR).

The AR retains the specified address so that the file manager can repeatedly access the specified indirect register until the AR is updated.

Refer to the Indirect Registers section for further details.

Register Access Conditions: During some FDP operating states, the file manager may not be able to access some of the registers. Operating states and corresponding inaccessable registers are listed in table 11.

Table 11 Register Access States

FDP Operation	Registers Not Accessable by the File Manager
No operation	None
LOAD or STORE command	B0PL, B0PH, B1PL, B1PH, MDR1, MDR4, BUF0, BUF1, SCNT, SRLH, SRLL, TCNTH, TCNTL
START command	Direct registers except CMR and CEND, and all indirect registers
STECC command	MDR2, ECC0–ECC6, PLY0–PLY6, RCPH, RCPL

Host Interface

The FDP can operate in two modes: master mode (MST bit of MDR1 is 1) in which the FDP has the bus right and slave mode (MST bit of MDR1 is 0) in which the MPU or DMAC has the bus right.

Asynchronous data transfer, with a minimum of 4 access cycles in 16-bit mode or 3 access cycles in 8-bit mode, is enabled in both modes.

Slave Mode: In slave mode (MST = 0), FDP pins relating to the host interface (pins 31, 32, 33, 34, 37, and 38) have the functions listed in table 12.

Table 12 Pin Functions in Slave Mode

Pin No.	Type	Pin Name	Remarks
32	Output	$\overline{\text{DREQ}}$	
33	Input	$\overline{\text{DACK}}$	
38	Input	$\overline{\text{IORD}}$	
37	Input	$\overline{\text{IOWR}}$	
31	Input	$\overline{\text{DONE}}$	
34	Input	$\overline{\text{WAIT}}$	Not used (must be pulled up)

- DMA transfer
  - DMA write (host to FDP)

    When a DMA write transfer is requested, the FDP asserts the $\overline{\text{DREQ}}$ signal. When $\overline{\text{DACK}}$ and $\overline{\text{IOWR}}$ are then asserted in this state, the FDP receives data from the host at the rising edge of the $\overline{\text{IOWR}}$ signal.

  - DMA read (FDP to host)

    When a DMA read transfer is requested, the FDP asserts the $\overline{\text{DREQ}}$ signal. When $\overline{\text{DACK}}$ and $\overline{\text{IORD}}$ are then asserted in this state, the FDP outputs data to the host.

  - DMA transfer modes

    Two DMA transfer modes are available: burst mode and cycle steal mode. The DMA transfer mode is indicated by the BURST bit of MDR4, as follows

    BURST bit = 0: Cycle steal mode
    BURST bit = 1: Burst mode

Burst mode: The FDP asserts the $\overline{\text{DREQ}}$ signal until the DMA transfer is completed (all data sectors are transferred), aborted ($\overline{\text{DONE}}$ is input, the ABORT command is issued, or a drive error occurs), or the buffer is switched. $\overline{\text{DREQ}}$ is temporarily negated when data of one sector has been transferred or after 544 bytes have been transferred if the sector is more than 544 bytes long (figure 3).

Cycle steal mode: When it receives $\overline{\text{DACK}}$ after asserting $\overline{\text{DREQ}}$, the FDP negates $\overline{\text{DREQ}}$ while $\overline{\text{DACK}}$ is asserted. While $\overline{\text{DACK}}$ is not asserted, the FDP asserts $\overline{\text{DREQ}}$ until the DMA transfer is completed (all sectors are transferred) or aborted ($\overline{\text{DONE}}$ is input, ABORT command is issued, or a drive error occurs).

– Data bus

Two data buses can be selected: 8-bit data bus and 16-bit data bus. The data bus width is indicated by the B8 bit of MDR1, as follows:

B8 bit = 0: 16-bit data bus (16-bit mode)
B8 bit = 1: 8-bit data bus (8-bit mode)

In 16-bit mode, BUS_{00} to BUS_{07} are the most significant byte and BUS_{08} to BUS_{15} are the least significant byte. In 8-bit mode, only BUS_{00} to BUS_{07} are used so BUS_{08} to BUS_{15} must be pulled up.

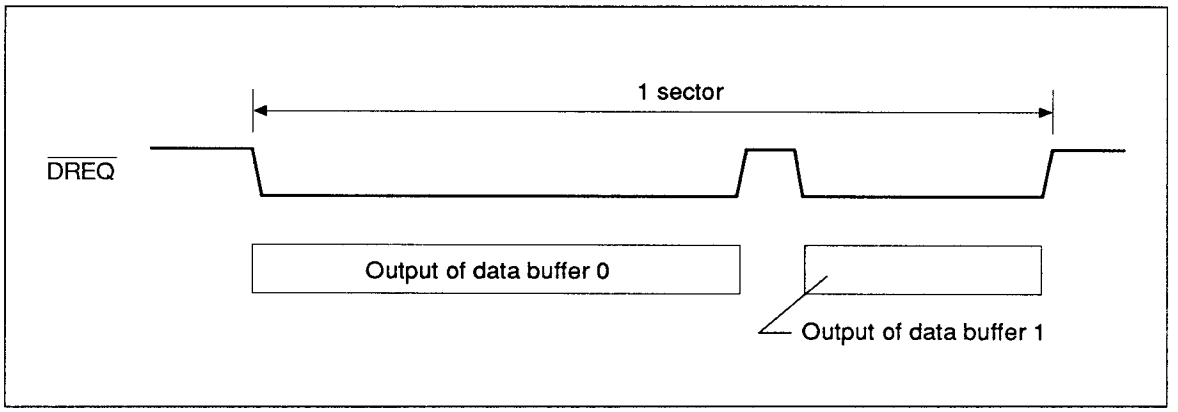


Figure 3 Data Buffer Switching and Negation of $\overline{DREQ}$

Master Mode: In master mode: (MST = 1), the FDP pins relating to the host interface (pins 31, 32, 33, 34, 37, and 38) have the functions listed in table 13.

Table 13 Pin Functions in Master Mode

Pin No.	Type	Pin Name	Remarks
32	Input	$\overline{DREQ}$	
33	Output	$\overline{DACK}$	
38	Output	$\overline{IORD}$	
37	Output	$\overline{IOWR}$	
31	Input	$\overline{DONE}$	
34	Input	$\overline{WAIT}$	

- DMA transfer

  – DMA read (FDP to host)

  When a DMA write transfer is requested, the FDP waits until the $\overline{\text{DREQ}}$ signal is asserted, then it asserts $\overline{\text{DACK}}$ and $\overline{\text{IOWR}}$ and begins to transfer data. (The FDP outputs data at the falling edge of $\overline{\text{IOWR}}$.) If $\overline{\text{DREQ}}$ is asserted and $\overline{\text{WAIT}}$ is negated, the FDP transfers data for two asserted cycles and one negated cycle of $\overline{\text{IOWR}}$. If the transfer rate of the host is too slow to complete the transfer during these cycles, $\overline{\text{WAIT}}$ must be asserted or $\overline{\text{DREQ}}$ negated. The FDP does not negate $\overline{\text{IOWR}}$ while $\overline{\text{WAIT}}$ is asserted, and it does not assert $\overline{\text{IOWR}}$ while $\overline{\text{DREQ}}$ is negated.

  – DMA write (host to FDP)

  When a DMA read transfer is requested, the FDP waits until the $\overline{\text{DREQ}}$ signal is asserted, then it asserts $\overline{\text{DACK}}$ and $\overline{\text{IORD}}$ and begins to transfer data. (The FDP fetches data at the rising edge of $\overline{\text{IORD}}$.) If $\overline{\text{DREQ}}$ is asserted and $\overline{\text{WAIT}}$ is negated, the FDP transfers data for two asserted cycles and one negated cycle of $\overline{\text{IORD}}$. If the transfer rate of the host is too slow to complete the transfer during these cycles, $\overline{\text{WAIT}}$ must be asserted or $\overline{\text{DREQ}}$ negated.

  – DMA transfer mode

  In master mode, only burst mode is available. The FDP asserts the $\overline{\text{DACK}}$ signal until DMA transfer is aborted or completed. The FDP negates the $\overline{\text{DACK}}$ signal under any of the following conditions:
  - All data sectors are transferred.
  - $\overline{\text{DONE}}$ is input.
  - ABORT command is issued.

  Note: If DMA transfer is aborted because of a drive error, $\overline{\text{DACK}}$ will not be negated. Therefore, an ABORT command must be issued from the FM. (The setting and resetting of CEND bits are the same as those in slave mode.)

  – Data bus

  In master mode, only the 8-bit data bus (BUS_{00} to BUS_{07}) is used. The remaining data bus lines (BUS_{08} to BUS_{15}) must be pulled up.

Notes:
1. Assert $\overline{\text{DONE}}$ only during a DMA transfer. In master mode, the FDP accepts $\overline{\text{DONE}}$ and sets the BUSINT bit of CEND to 1.
2. If $\overline{\text{DONE}}$ is input when the last data of each sector is transferred, that sector's data may be written to the drive, depending on the input timing of $\overline{\text{DONE}}$.
3. $\overline{\text{WAIT}}$ is ignored while $\overline{\text{DONE}}$ is input.
4. If $\overline{\text{DONE}}$ is input during a DMA transfer (except as described in note 2), the transfer ($\overline{\text{IORD}}$, $\overline{\text{IOWR}}$, and data AC characteristics) cannot be guaranteed.
5. If a sector is 545 bytes or longer, the negation width of $\overline{\text{IORD}}$ or $\overline{\text{IOWR}}$ increases while data buffers are switched, as shown in figure 4.

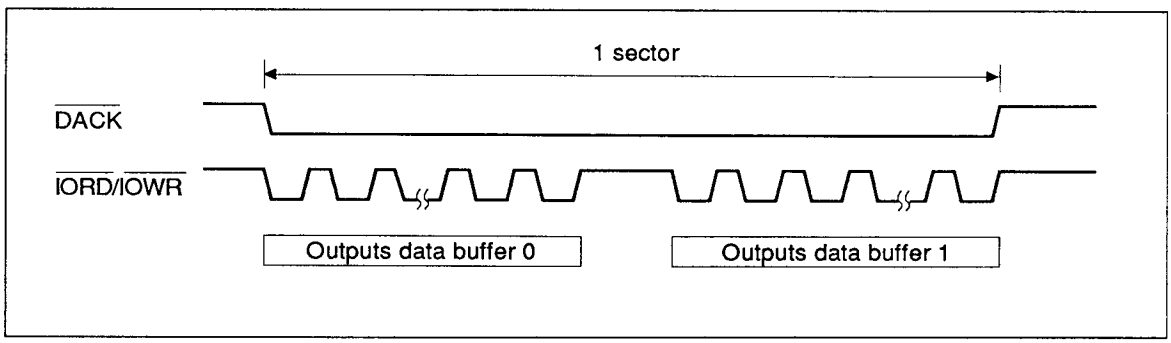


Figure 4 Negation of $\overline{\text{IORD}}/\overline{\text{IOWR}}$

Data Transfer Function

Various data transfer modes are available, depending on the settings of registers. All data transfers are DMA transfers.

Data Transfer Modes

The bits shown in tables 14 and 15 must be set to suit the mode selected for data transfer.

Table 14 Host Interface Mode Settings

Register	Bit	Value	Mode
MDR1	MST	0	FDP slave mode
		1	FDP master mode
	B8	0	16-bit data bus
		1	8-bit data bus
	DMA	0	A DMA transfer is not performed when a command is issued
		1	A DMA transfer is performed when a command is issued
MDR4	BURST	0	Cycle steal mode
		1	Burst mode

Note: The B8 and BURST bits are valid only when MST = 0.

Table 15 Data Buffer Mode Settings

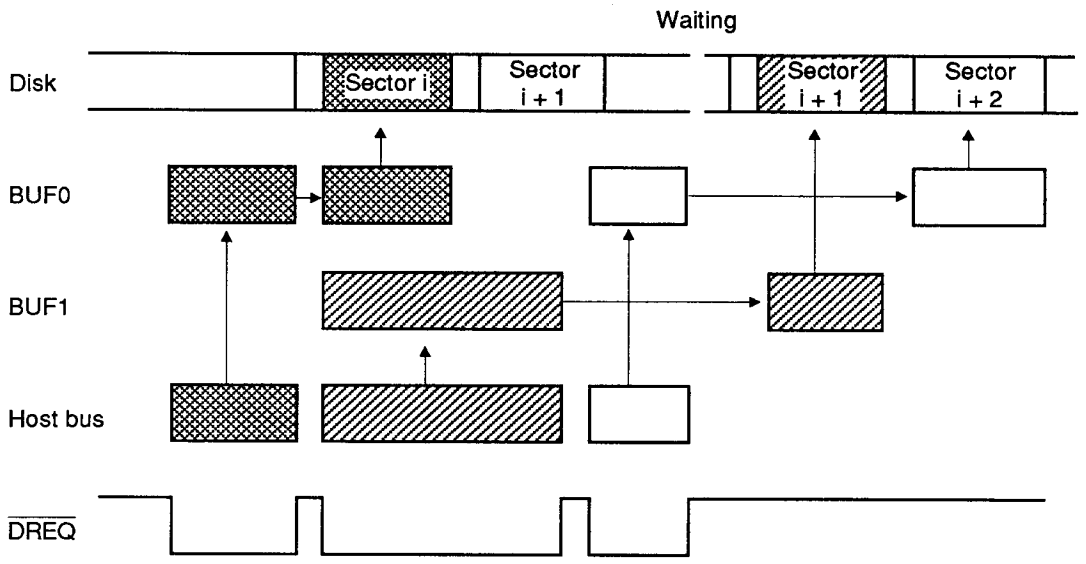
Register	Bit	Value	Mode
MDR1	DRBF	0	Starts DMA transfer from data buffer 0 (BUF0)
		1	Starts DMA transfer from data buffer 1 (BUF1)
	DMAM	0	Changes data buffers for each sector
		1	Does not change data buffers for each sector
MDR4	BFM	0	Two-buffer mode (one sector is 4 to 544 bytes long)
		1	One-buffer mode (one sector is 545 to 1088 bytes long)

Notes: 1. The DMAM = 1 setting is valid only in write format mode. (The DMA bit must be 0 in write format mode.)
2. BFM is valid only when DMAM = 0.

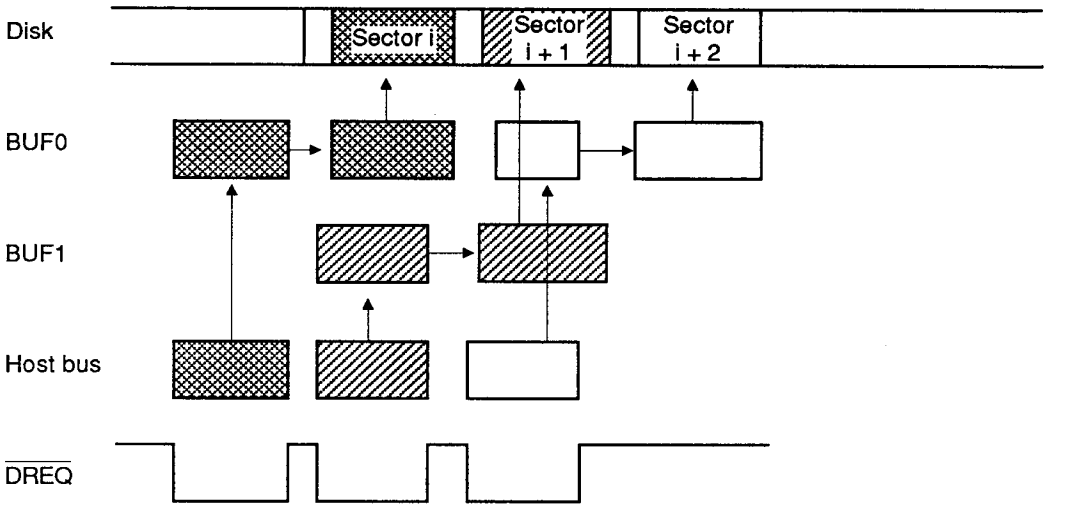
- Data transfer when DMAM=0

  For a DMA transfer in this mode, the data buffer receives one sector of data from disk, for example, and transfers it to the host. Since the data buffers (BUF0 and BUF1) hold a maximum of 1088 bytes (544 bytes each), each sector must be no more than 1088 bytes long.

  If the sector length is no more than 543 bytes, multi-sector processing without interleaving is possible. The writing of data to disk by DMA transfer in this case is shown in figure 5 and the reading of data from disk is shown in figure 6.

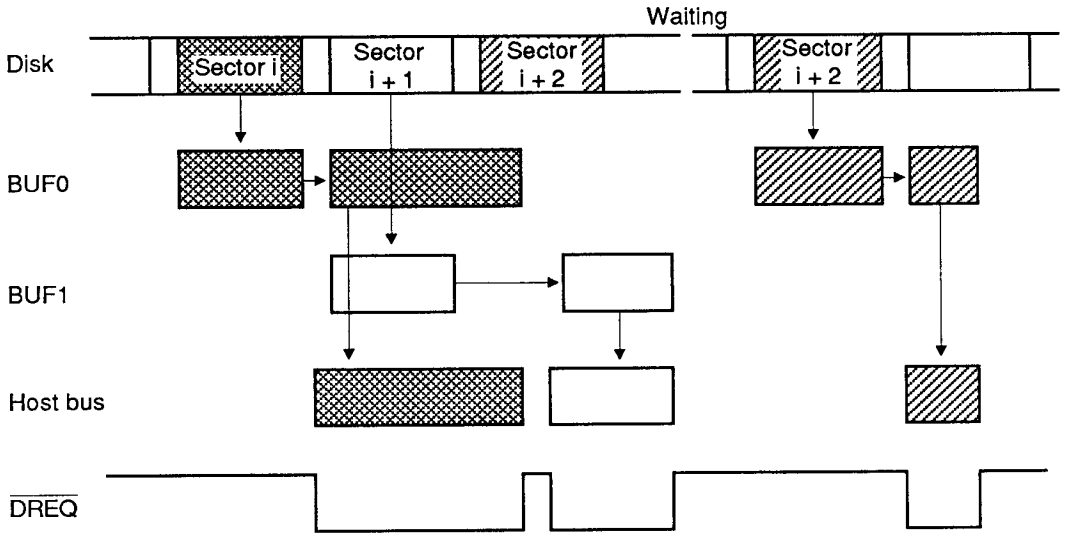


(a) Slow Host Transfer Rate

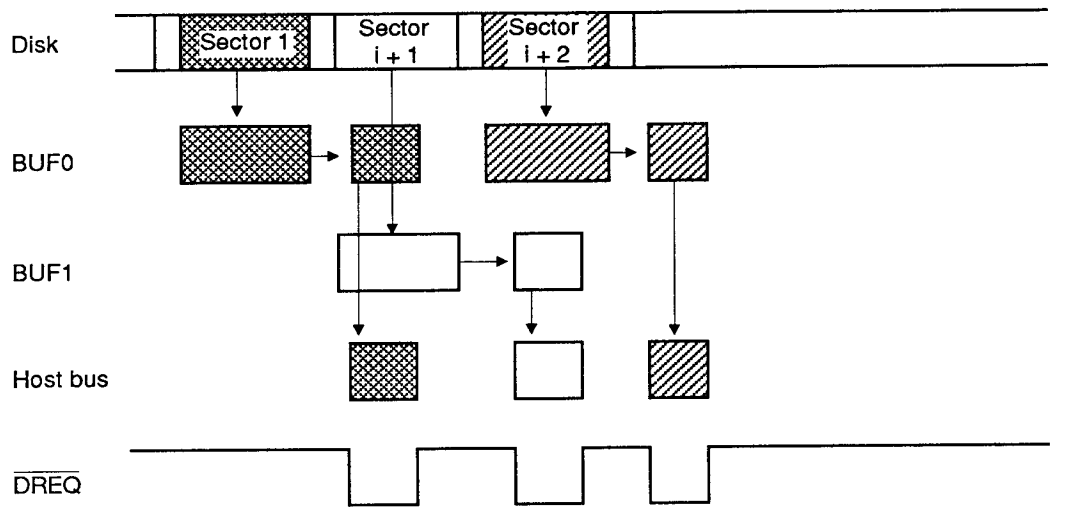


(b) Fast Host Transfer Rate

Figure 5 DMA Write (up to 543 bytes/sector) Started by DMAW Command



(a) Slow Host Transfer Rate



(b) Fast Host Transfer Rate

Figure 6 DMA Read (up to 543 bytes/sector)

- Data transfer when DMAM = 1

  In this mode, data buffers are switched when address 543 is accessed. The next address is accessed by the next format command (Figure 7).

  Use this transfer method for write formatting when DMA = 0.

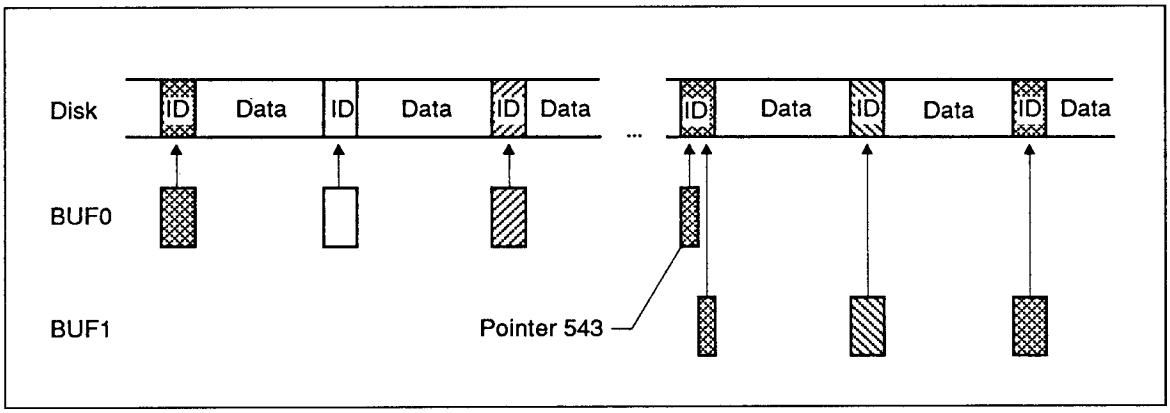


Figure 7 Write Formatting

Data Transfer

Start of DMA Transfer: The FDP starts DMA transfer between the data buffer and the host when a LOAD or STORE command is issued or when a WIDB, WDTB, RIDB, RDTB, VDTB, or DMAW command is executed while DMA = 1.

Completion of DMA Transfer: The FDP completes DMA transfer when any of the following conditions is satisfied:

- The specified amount of data (SRLL/SRLH value × SCNT value) has been transferred.
- A drive access error aborts a command.
- $\overline{\text{DONE}}$ is asserted.
- An ABORT command is issued.

Stopping DMA Transfer (to monitor the status of data buffers): When a LOAD or STORE command is executing, the FDP (in slave mode) asserts $\overline{\text{DREQ}}$ until the specified amount of data has been transferred. If data to be stored overflows the data buffers, the data stored last is valid.

Example: If the STORE command specifying DMAM = 0, SCNT = 3, and DRBF = 0 is executed, the third sector's data is stored in data buffer BUF0 and the second sector's data is stored in BUF1. The first sector's data is not stored.

If LOAD commands with the same specifications are executed, the same data is repeatedly output.

If DMA transfer is initiated by the execution of command other than STORE or LOAD, the FDP monitors the status of the data buffers to ensure that the data buffers' contents and disk data are not lost. If a data transfer between the host and data buffer overflows, the FDP negates $\overline{\text{DREQ}}$ if it is in slave mode, or asserts neither $\overline{\text{IORD}}$ nor $\overline{\text{IOWR}}$ if it is in master mode.

DMA transfer is requested by a WIDB, WDTB, RIDB, RDTB, or VDTB command, the FDP inhibits transfers between the host and the data buffers, but allows transfers between the data buffers and the disk drive. If DMA transfer is requested by a DMAW command, the FDP allows transfers between the host and the data buffers but inhibits transfers between the data buffers and the disk drive.

Error Checking and Correction Function

The FDP has two error checking and correction circuits: a detection and generation circuit for 16-bit CRC (cyclic redundancy code) and a detection, generation, and correction circuit for ECC (error checking and correction) code (max. 7-byte fire code). These codes are attached to each disk field if specified with the format commands. The characteristics of the CRC and ECC circuits are listed in tables 16 and 17.

Table 16 CRC Detection and Generation Circuit

Item	Description
Polynominal	$X^{16} + X^{12} + X^5 + 1$ (fixed)
Initial value	Either all zeros or all ones (set by the CRCI bit of MDR2)
CRC operation	Either addition or non-addition of the byte sync pattern (set by the BSEX bit of MDR2)
Format command	(Generation) WCRC (Detection) CCRC, VCRC, VCRCW, VCRCR

Table 17 ECC Detection, Generation, and Correction Circuit

Item	Description
ECC data length	4, 6, or 7 bytes (set by the ECM0 and ECM1 bits of MDR2)
Polynominal	any (set by PLY0 to PLY6)
Initial value	Either all zeros or all ones (set by the ECCI bit of MDR2)
ECC operation	Either addition or non-addition of the byte sync pattern (set by the BSEX bit of MDR2)
Format command	(Generation) WECC (Detection) CECC, VECC, VECCW, VECCR
Maximum length of correctable burst error	5, 6, 7, 8, 9, 10, 11, or 16 bits Note (Set by ZCK2, ZCK1, and ZCK0 bits of MDR2)
Correction pattern generation command	STECC (direct command)

Note: Limited by polynominal and error pattern

ECC/CRC Generation and Detection Functions

Setting of ECC Data Length and Polynominal: To use ECC codes, it is necessary to set their data length and their polynominal first.

- Setting of ECC data length

The ECC data length is specified by the ECM1 and ECM0 bits of MDR2, as follows:

ECM1	ECM0	ECC Data Length
0	0	4 bytes
0	1	6 bytes
1	0	7 bytes
1	1	(Reserved)

The relations between ECC data lengths and polynominal registers are given below.

– 4-byte mode

Bits	Plynominal Register	ECC
0–7	PLY0	ECC0
8–15	PLY1	ECC1
16–23	PLY5	ECC5
24–31	PLY6	ECC6

– 6-byte mode

Bits	Plynominal Register	ECC
0–7	PLY0	ECC0
8–15	PLY1	ECC1
16–23	PLY2	ECC2
24–31	PLY4	ECC4
32–39	PLY5	ECC5
40–47	PLY6	ECC6

– 7-byte mode

Bits	Plynominal Register	ECC
0–7	PLY0	ECC0
8–15	PLY1	ECC1
16–23	PLY2	ECC2
24–31	PLY3	ECC3
32–39	PLY4	ECC4
40–47	PLY5	ECC5
48–55	PLY6	ECC6

Note: Polynominal registers PLY0 to PLY6 not listed in the above tables need not be set.

- Setting of polynominals

The polynominals that generate ECC codes are specified by PLY0 to PLY6. The bits of each polynominal register (PLY0 to PLY6) are in a one-to-one relationship with the bits of the ECC registers (ECC0 to ECC6). The 1 bits specified in PLY0 to PLY6 are fed back, and the 0 bits are directly shifted. The least significant bit (LSB) represents the data which was shifted from the most significant bit (MSB), independently of the PLY0 value.

The examples below show typical settings of PLY0 to PLY6 for each byte mode.

– Typical polynominal for 4-byte mode

$(x^{21} + 1)(x^{11} + x^2 + 1)$
$= x^{32} + x^{23} + x^{21} + x^{11} + x^2 + 1$

Exponent =32 31 30 29 28 27 26 25 24
PLY6 =1 0 0 0 0 0 0 0 0 = $00

Exponent = 23 22 21 20 19 18 17 16
PLY5 = 1 0 1 0 0 0 0 0 = $A0

Exponent = 15 14 13 12 11 10 9 8
PLY1 = 0 0 0 0 1 0 0 0 = $08

Exponent = 7 6 5 4 3 2 1 0
PLY0 = 0 0 0 0 0 1 0 1 = $05

– Typical polynominal for 6-byte mode

$(x^{13} + 1)(x^{35} + x^{23} + x^8 + x^2 + 1)$
$= x^{48} + x^{36} + x^{35} + x^{23} + x^{21} + x^{15} + x^{13} + x^8 + x^2 + 1$

PLY6 = $00
PLY5 = $18
PLY4 = $00
PLY2 = $A0
PLY1 = $A1
PLY0 = $05

– Typical polynominal for 7-byte mode

$(x^{22} + 1)(x^{12} + x^{11} + x^{10} + x^9 + x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1)(x^{11} + x^7 + x^6 + x + 1)(x^{11} + x^9 + x^7 + x^6 + x^5 + x + 1)$
$= x^{56} + x^{55} + x^{49} + x^{45} + x^{41} + x^{39} + x^{38} + x^{37} + x^{36} + x^{31} + x^{22} + x^{19} + x^{17} + x^{16} + x^{15} + x^{14} + x^{12} + x^{11} + x^9 + x^5 + x + 1$

PLY6 = $82
PLY5 = $22
PLY4 = $F0
PLY3 = $80
PLY2 = $4B
PLY1 = $DA
PLY0 = $23

Setting of Initial Values and Treatment of the Byte Sync Pattern: Before an CRC/ECC operation is started, the initial CRC and ECC values must be specified by the CRCI and ECCI bits of MDR2. The initial values set by the CRCI and ECCI bits are used each time a CRC/ECC initialization command is executed.

CRCI = 0: The initial CRC value is all zeros.
CRCI = 1: The initial CRC value is all ones.

ECCI = 0: The initial ECC value is all zeros.
ECCI = 1: The initial ECC value is all ones.

Specify the treatment of the byte sync pattern by the BSEX bit of MDR2, as follows:

BSEX = 0: A byte sync pattern is included in the ECC/CRC operation.
BSEX = 1: No byte sync pattern is included in the ECC/CRC operation.

Generation and Detection of ECC and CRC Codes: The FDP can generate ECC and CRC codes and detect errors if specified format commands are run after the ECC data length, polynominals, and initial values have been set. To ensure normal ECC/CRC generation and detection, the format commands must be combined as follows:

Command to initialize ECC/CRC	→	Command to execute ECC/CRC operation	→	Command to determine or output an ECC/CRC error

Repeat this combination of format commands to process two or more sectors (for multi-sector processing). Valid format commands are listed in table 18.

Table 18 ECC/CRC Generation and Detection Commands

Command	ECC/CRC Initialization	ECC/CRC Operation	ECC/CRC Error Determination/Output
WPTN	○	×	×
WPUIS	(○)	×	×
WPAI	(○)	×	×
WCRC	○	×	○
WECC	(○)	×	○
WSP	○	×	×
WASFI	(○)	×	×
WAFSA (hard sector)	(○)	×	×
WAFSA (soft sector)	○	×	×
DMAW	○	×	×
VID	○	(○)	×
CCRC	(○)	○	○
VCRC	(○)	(○)	○
VCRCR	(○)	(○)	○
VCRCW	(○)	(○)	○
CECC	(○)	○	○
VECC	(○)	(○)	○
VECCR	(○)	(○)	○
VECCW	(○)	(○)	○
VDTB	×	(○)	×
WAITI	(○)	×	×
WTISA (hard sector)	(○)	×	×
WTISA (soft sector)	(○)	×	×
WIDB	×	(○)	×
WDTB	×	(○)	×

Table 18 ECC/CRC Generation and Detection Commands (cont)

Command	ECC/CRC Initialization	ECC/CRC Operation	ECC/CRC Error Determination/Output
WDTP	×	○	×
RIDB	(○)	○	×
RIDS	○	(○)	×
RDTB	(○)	(○)	×
RDTS	○	(○)	×
IDINC	○	×	×
SKIP	○	×	×
END	×	×	×

○: Always enabled (○): Conditionally enabled. ×: Disabled

Error Correction Function

Any ECC error found in read data must be corrected. The FDP uses the error diagnosis (the contents of ECC0 to ECC6 after the ECC check) to calculate error positions and a correction pattern. Therefore, the FM must correct the data stored in the data buffers, based on the result of the FDP's calculation.

Setting of Reciprocal Polynominals: To generate a correction pattern, the FDP reverses the generation and checking operations using the error diagnosis. Therefore, it is necessary to specify a reciprocal polynominal before issuing the STECC command to obtain the correction pattern. Assuming that the original polynominal is f(x), the reciprocal polynominal can be expressed by:

$$x^n \cdot f\left(\frac{1}{x}\right)$$

The following polynominals are the reciprocals of the original typical polynominals for 4-, 6- and 7-byte modes given above:

4-byte ECC mode

$$x^{32} + x^{30} + x^{21} + x^{11} + x^9 + 1$$

6-byte ECC mode

$$x^{48} + x^{46} + x^{40} + x^{35} + x^{33} + x^{27} + x^{25} + x^{13} + x^{12} + 1$$

7-byte ECC mode

$$x^{56} + x^{55} + x^{51} + x^{47} + x^{45} + x^{44} + x^{42} + x^{41} + x^{40} + x^{39} + x^{37} + x^{34} + x^{25} + x^{20} + x^{19} + x^{18} + x^{17} + x^{15} + x^{11} + x^7 + x + 1$$

These reciprocal polynominals are set in polynominal registers PLY0 to PLY6 in the same manner as described in the previous section, and the specification of ECC data length and the relations between polynominal registers and ECC registers are also exactly the same. The values in the polynominal registers in each ECC byte mode are as follows:

4-byte ECC mode	6-byte ECC mode	7-byte ECC mode
PLY6 = $40	PLY6 = $41	PLY6 = $88
PLY5 = $20	PLY5 = $0A	PLY5 = $B7
PLY1 = $0A	PLY4 = $0A	PLY4 = $A4
PLY0 = $01	PLY2 = $00	PLY3 = $02
	PLY1 = $30	PLY2 = $1E
	PLY0 = $01	PLY1 = $88
		PLY0 = $83

Setting of Correction bit Length: Generally speaking, ECC accuracy decreases as the correction bit length increases. To avoid this deterioration, the necessary correction bit length should be specified for the FDP, using the ZCK2, ZCK1, and ZCK0 bits of MDR2.

ZCK2	ZCK1	ZCK0	Correction Bit Length
0	0	0	16 bits
0	0	1	11 bits
0	1	0	10 bits
0	1	1	9 bits
1	0	0	8 bits
1	0	1	7 bits
1	1	0	6 bits
1	1	1	5 bits

Activation by Command: Issue a STECC command after setting the reciprocal polynominal and the correction bit length. The FDP begins the operation from the error diagnosis left in ECC0 to ECC6, and stops after the specified number of operations (corresponding to the sector length), regardless of whether it has obtained a correction pattern. This operation limit is specified by the indirect register SRLH as follows:

SRLH = ****000X: $(2^{13} - 1)$ times (8191 times)
****001X: $(2^{14} - 1)$ times (16383 times)
****01XX: $(2^{15} - 1)$ times (32767 times)

If it does not obtain a correction pattern after the specified number of operations the FDP stops command execution, setting the CRFLT bit of STR1 and the ERR bit of CEND. In this case, the contents of RCPH, RCPL, and ECC0 to ECC6 are meaningless.

Correction: If it obtains a correction pattern, the FDP stops command execution without setting the CRFLT bit of STR1 and the ERR bit of CEND. In this case, the contents of RCPH and RCPL indicate the error bit position, counting from the end of all the data (including ECC codes), and ECC0 and ECC1 contain the correction pattern. The FM can now correct the buffer data from the contents of ECC0 and ECC1.

Example: Sector length = 256 bytes, ECC data length = 32 bits, number of correction bits = 11, RCPL, RCPH = 410 (decimal), ECC0 = $0A, ECC1 = $05 (figure 8)

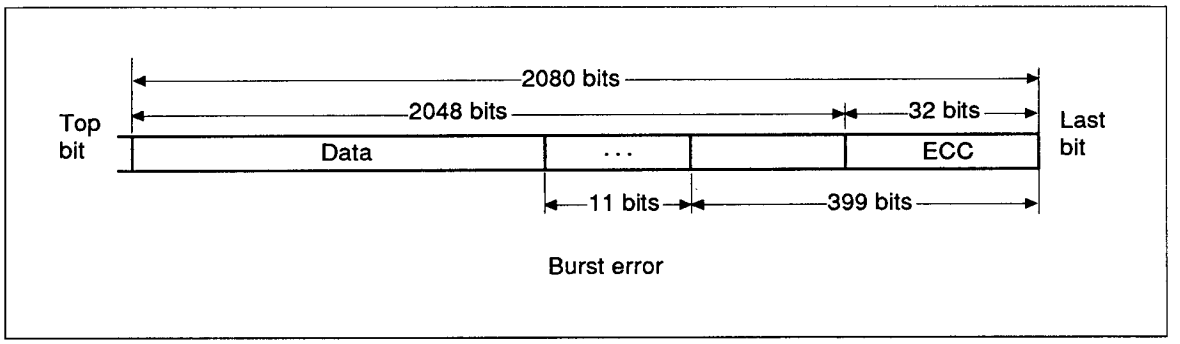


Figure 8 Burst Error Data

In this example, a burst error is detected between bit 410 and bit 400 (11 bits), counting from the last data bit. The correction pattern is represented by bits 0 to 7 of ECC0 and bits 0 to 2 of ECC1 (for 11 correction bits), as shown in figure 9.

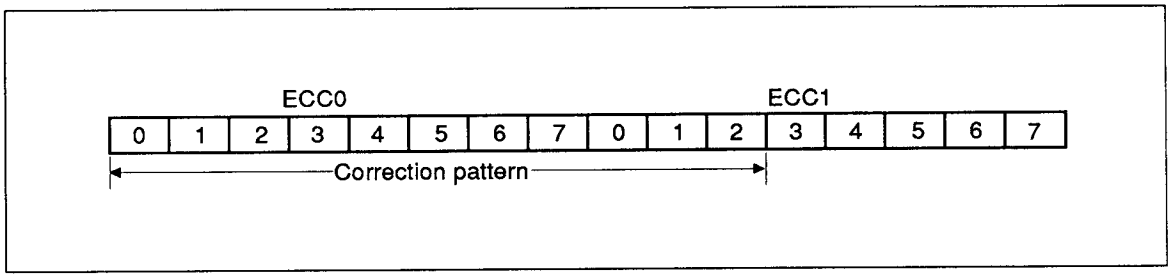


Figure 9 Correction Pattern

1 bits in the correction pattern invert the corresponding bits in drive data. 0 bits keep drive data unchanged. The following shows the one-to-one relationship of the bits of registers ECC0 and ECC1 to data bits.

ECC0 Bit 0: Bit 410
Bit 1: Bit 409
Bit 2: Bit 408
Bit 3: Bit 407
Bit 4: Bit 406
Bit 5: Bit 405
Bit 6: Bit 404
Bit 7: Bit 403
ECC1 Bit 0: Bit 402
Bit 1: Bit 401
Bit 2: Bit 400

The following is an example of error correction:

Burst error data	10100011100
Correction pattern	01010000101
Corrected data	11110011001

Operation Examples

Formatting

Conditions: Hard sector mode, 3 sectors to be formatted, ECC data 4 bytes long (figure 10)
The result of the formatting is shown in figure 11.

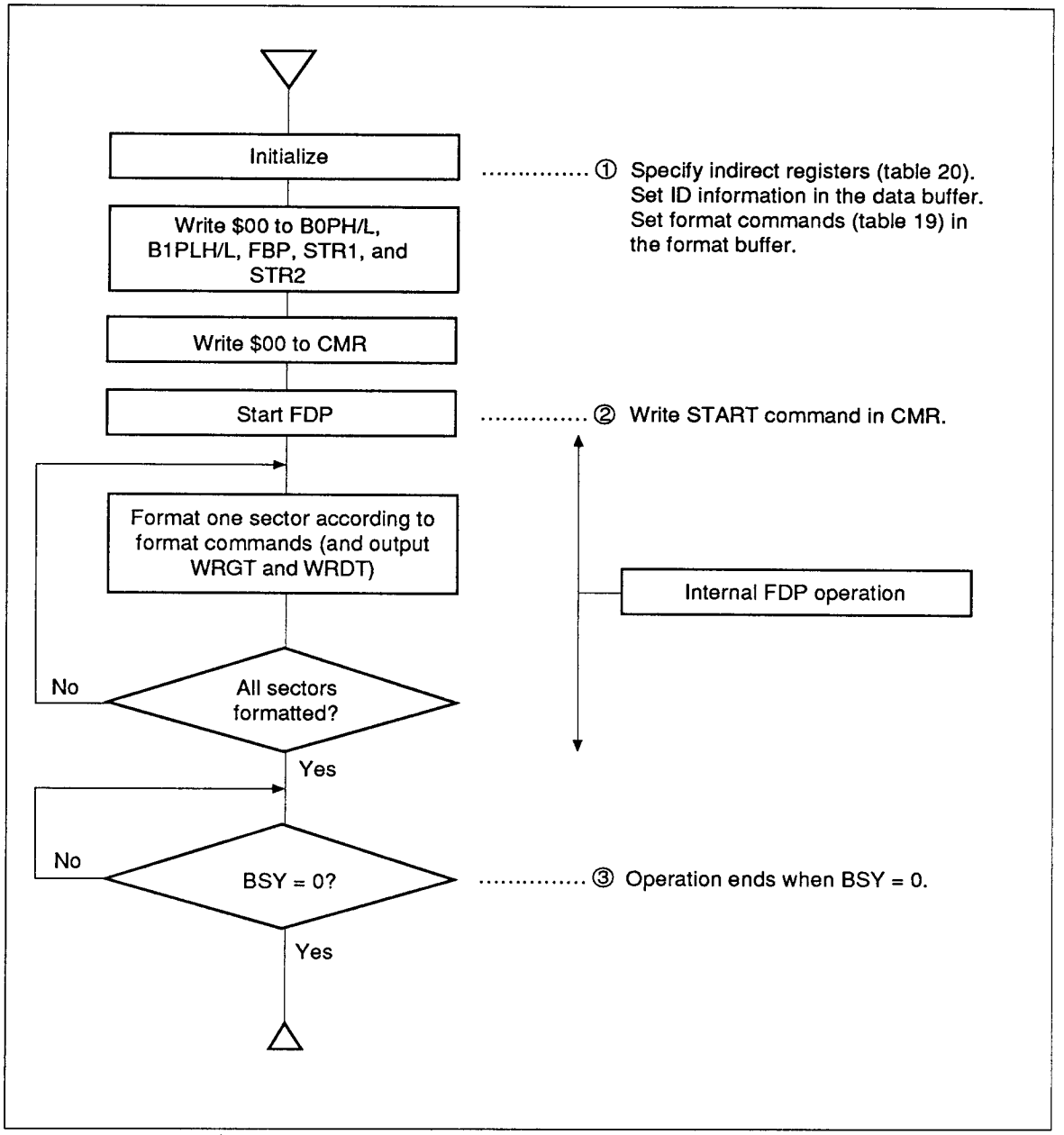


Figure 10 Formatting Flowchart

Table 19 Formatting Commands

FBP	FMTB1		FMTB2	FMTB3
00	WPAI + NOP	$18	$00	$07
01	WPTN + JRS	$0B	$00	$0B
02	WIDB + NOP	$B8	$00	$06
03	WCRC + NOP	$20	$00	$06
04	WPTN + NOP	$08	$00	$0B
05	WDTP + NOP	$C8	$02	$01
06	WECC + NOP	$28	$00	$06
07	WABSC + JMP	$3D	$00	$07
08	WPTN + NOP	$08	$00	$0B
09	WIDB + NOP	$B8	$00	$06
0A	WPTN + NOP	$08	$00	$0B
0B	WDTP + NOP	$C8	$02	$01
0C	WECC + NOP	$28	$00	$06
0D	WPUIS + NOP	$10	$00	$04
0E	END	$00	$00	$00

Table 20 Register Setting Example

Register	Setting	Register	Setting
IDR0	$A0	DAM0	$A1
TCR	$A2	DPAT	$00
LPCNT	$02	PLY0	$05
TMB	$03	PLY1	$A0
TMC	$FF	PLY2	($00)
MDR1	$32	PLY3	($00)
MDR2	$3D	PLY4	($00)
MDR3	($02)	PLY5	$A0
MDR4	$14	PLY6	$00

( ): Omittable

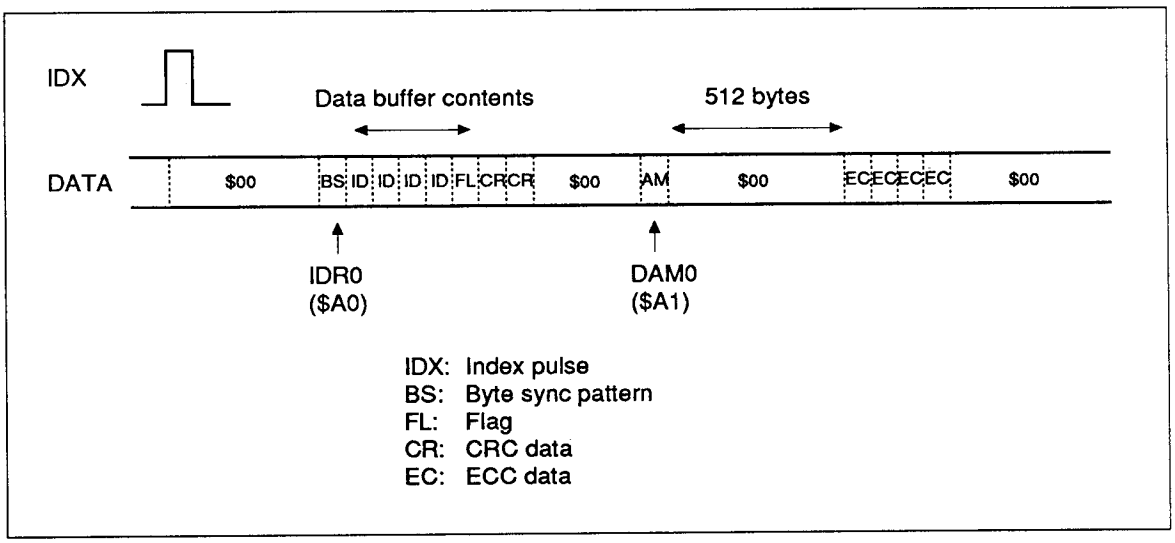


Figure 11 Formatting Example

Writing

Conditions: Hard sector mode, 3 sectors to be written, ECC data 4 bytes long (figure 12)

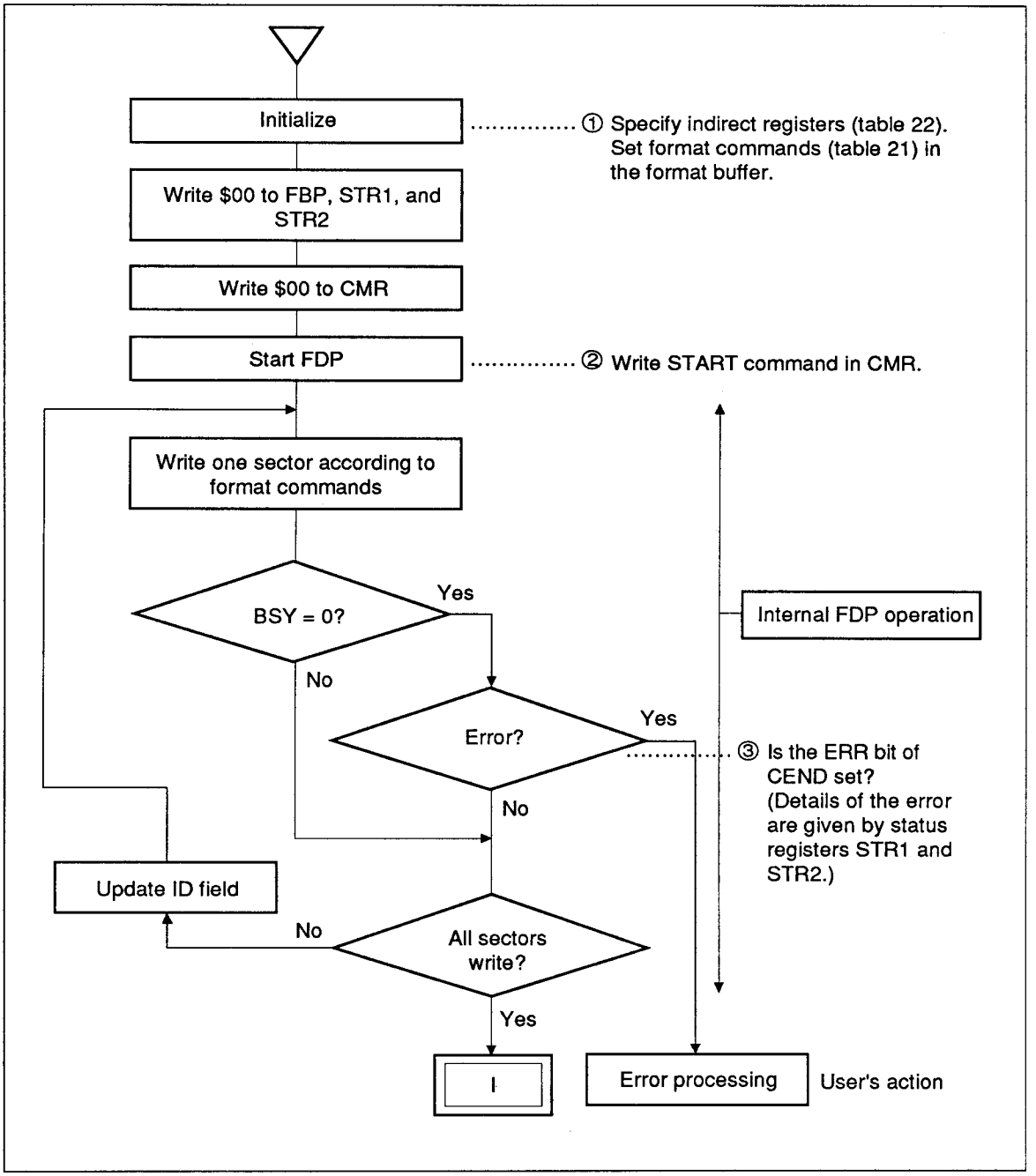


Figure 12 Write Flowchart

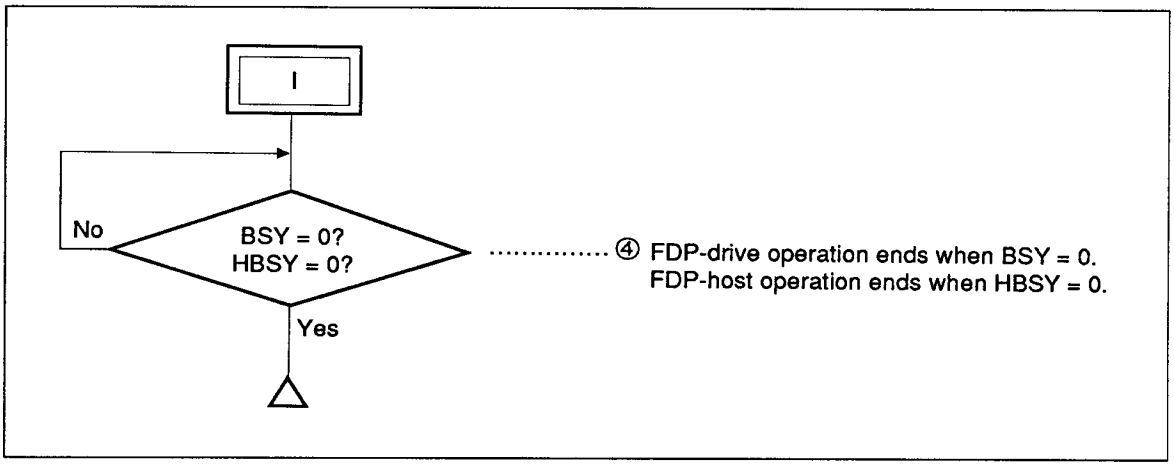


Figure 12 Write Flowchart (cont)

Table 21 Write Commands

FBP	FMTB1		FMTB2	FMTB3
00	DMAW + NOP	$50	$00	$04
01	WTISA + JRS	$B3	$00	$04
02	VID + RTY	$5E	$00	$06
03	VCRCW + RTY	$7E	$00	$0C
04	WDTB + RTY	$C6	$02	$01
05	WECC + NOP	$28	$00	$06
06	IDINC + JMP	$FD	$00	$04
07	END	$00	$00	$00
08				
09				
0A				
0B				
0C				

Table 22 Register Setting Example

Register	Setting	Register	Setting
IDR0	$A0	DAM0	$A1
IDR1	$00	B0PL	$00
IDR2	$01	B0PH	$00
IDR3	$01	B1PL	$00
IDR4	$01	B1PH	$00
IDR5	$00	SCNT	$03
TCR	$A2	PLY0	$05
SRLL	$00	PLY1	$A0
SRLH	$02	PLY2	($00)
LPCNT	$02	PLY3	($00)
TMB	$03	PLY4	($00)
TMC	$FF	PLY5	$A0
MDR1	$34	PLY6	$00
MDR2	$3D		
MDR3	$02		
MDR4	$14		

Reading

Conditions: Hard Sector mode, 3 sectors to be read, ECC data 4 bytes long

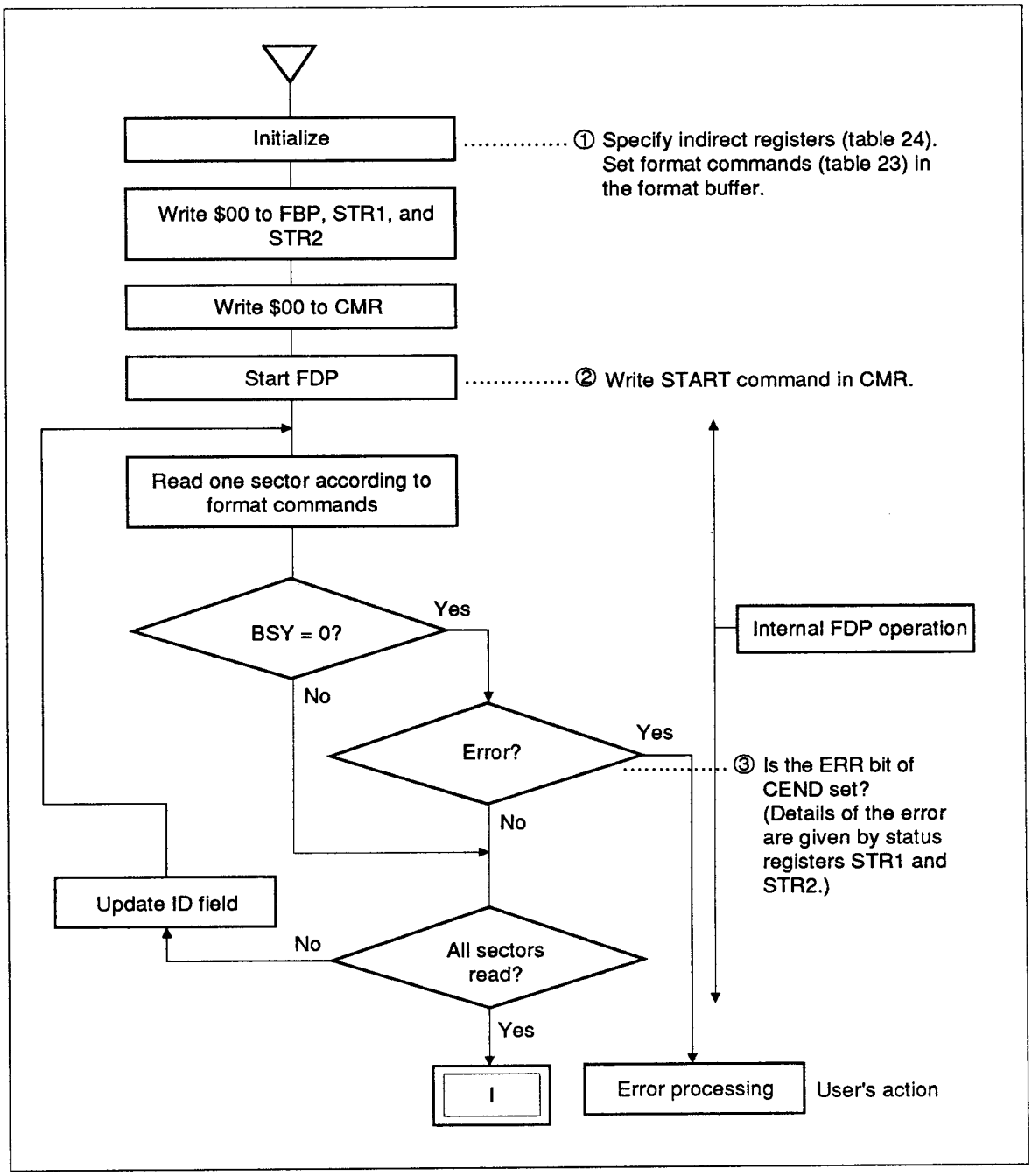


Figure 13 Read Flowchart

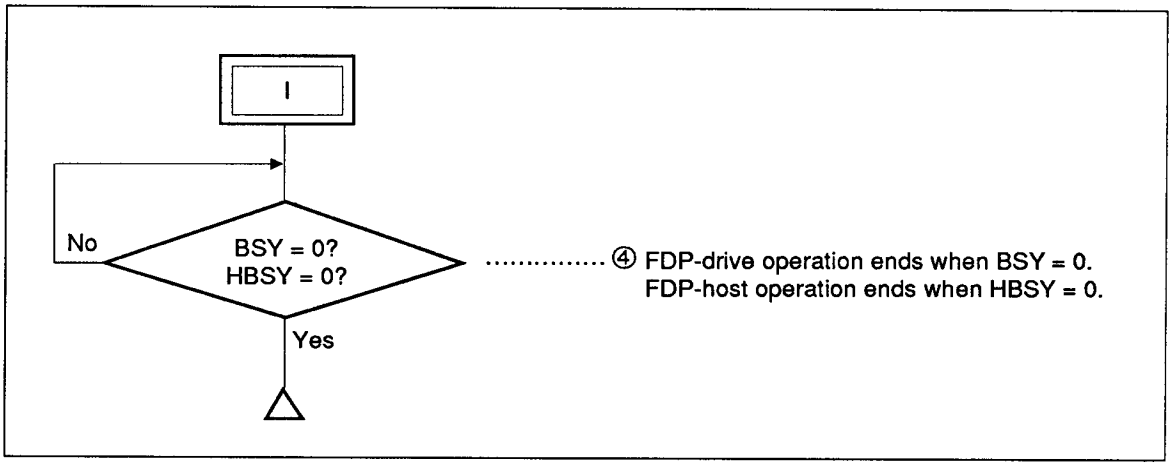


Figure 13 Read Flowchart (cont)

Table 23 Read Commands

FBP	FMTB1		FMTB2	FMTB3
00	WTISA + JRS	$B3	$00	$04
01	VID + RTY	$5E	$00	$06
02	VCRC + RTY	$6E	$00	$04
03	RDTB + RTY	$E6	$02	$02
04	CECC + NOP	$80	$00	$05
05	IDINC + JMP	$FD	$00	$04
06	END	$00	$00	$00
07				
08				
09				
0A				
0B				
0C				

Table 24 Register Setting Example

Register	Setting	Register	Setting
IDR0	$A0	DAM0	$A1
IDR1	$00	B0PL	$00
IDR2	$01	B0PH	$00
IDR3	$01	B1PL	$00
IDR4	$01	B1PH	$00
IDR5	$00	SCNT	$03
TCR	$A2	PLY0	$05
SRLL	$00	PLY1	$A0
SRLH	$02	PLY2	($00)
LPCNT	$02	PLY3	($00)
TMB	$03	PLY4	($00)
TMC	$FF	PLY5	$A0
MDR1	$34	PLY6	$00
MDR2	$3D		
MDR3	$02		
MDR4	$14		

Electrical Characteristics

Absolute Maximum Ratings

Item	Symbol	Rating	Unit	Note
Supply voltage	V_{cc}	−0.3 to +7.0	V	1
Input voltage	V_{in}	−0.3 to V_{cc} +0.3	V	1
Maximum output current	$\|I_o\|$	5	mA	2
Total allowable output current	$\|\Sigma I_o\|$	150	mA	3
Operating temperature	T_{opr}	0 to +70	°C	
Storage temperature	T_{stg}	−55 to +150	°C	

Notes: 1. With reference to V_{ss} = 0 V.
2. The maximum output current is the maximum current that may be drawn from, or flow out to, one output pin or one common input/output pin.
3. The total allowable output current is the total sum of currents that may be drawn from, or flow out to, output pins or common input/output pins.
4. Using an LSI beyond its maximum ratings may result in its permanent destruction. The FDP must be used under the recommended operating conditions. Exceeding any of these conditions may adversely affect its reliability.

Recommended Operating Conditions

Item	Symbol	Min	Typ	Max	Unit	Note
Supply Voltage	V_{cc}	4.75	5.0	5.25	V	1
Input high voltage	V_{IH}	2.2	—	V_{cc}	V	1
Input low voltage	V_{IL}	0	—	0.8	V	1
Operating temperature	T_{opr}	0	25	70	°C	

Note: 1. With reference to V_{ss} = 0 V.

Output Timing Values

The levels at which output signal timings are measured are shown in figure 14.

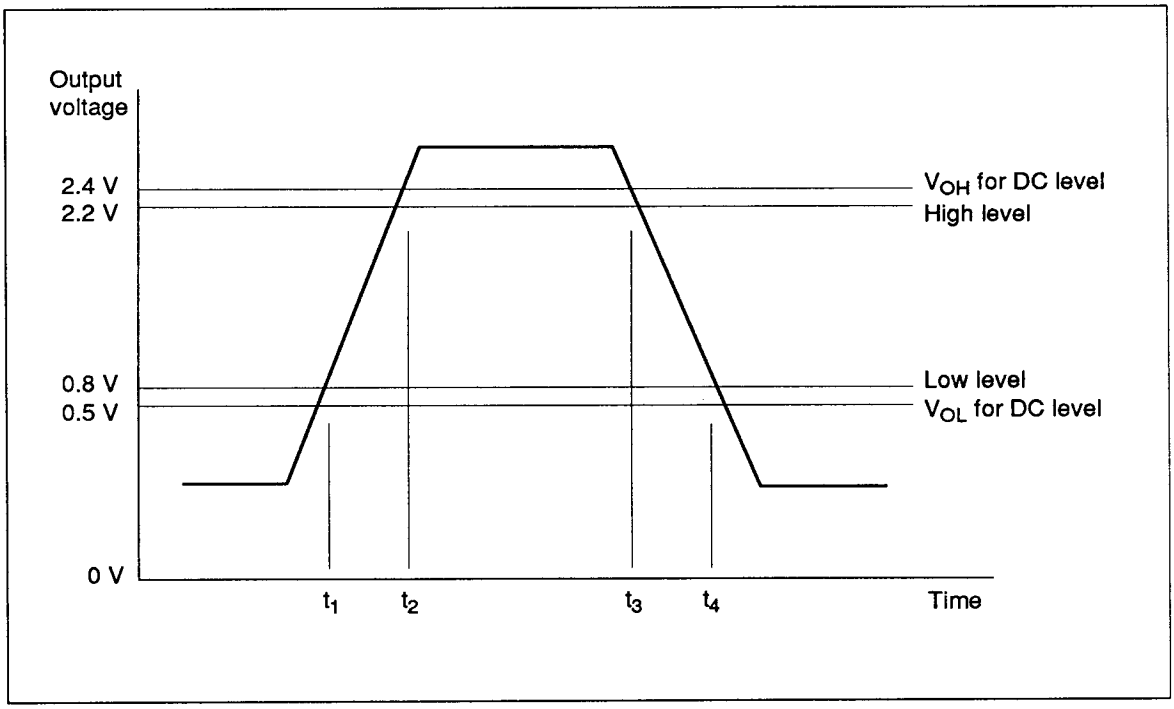


Figure 14 Measurement Points for Output Signal Timing

Electrical Characteristics

DC Characteristics (V_{cc} = 5.0 V ± 5%, V_{ss} = 0 V, Ta = 0 to + 70°C, unless otherwise specified)

Item		Symbol	Test Conditions	Limits Min	Typ	Max	Unit	Note
Supply voltage		V_{cc}		4.75	5.0	5.25	V	1
Input high voltage	All input pins	V_{IH}		2.2	—	V_{cc}	V	1
Input low voltage	All input pins	V_{IL}		–0.3	—	0.8	V	1
Output high voltage		V_{OH}	I_{OH} = –400 µA	2.4	—	—	V	1
Output low voltage		V_{OL}	I_{OL} = 1.6 mA	—	—	0.5	V	1
Input leakage current	RCLK, CC, CLK, S/AMF, RDT, RDY, $\overline{DONE}$, IDX, $\overline{WAIT}$, $\overline{RD}$, $\overline{WR}$, A_0–A_2, $\overline{CS}$ $\overline{RST}$	I_{in}	V_{in} = 0 V – V_{cc}	–2.5	—	2.5	µA	
3-state input current (off state)	D_0–D_7, $\overline{DREQ}$, BUS_{00}–BUS_{15}, P_0–P_7, $\overline{DACK}$ $\overline{IORD}$, $\overline{IOWR}$	I_{TSI}	V_{in} = 0.4 V – V_{cc}	–10	—	10	µA	
Output leakage current (off state)	$\overline{INT}$	I_{LOD}	V_{OH} = V_{cc}	—	—	10	µA	
Input capacitance	All input pins	C_{in}	V_{in} = 0 V f = 1 MHz Ta = 25°C	—	—	20	pF	
Output capacitance	All output pins	C_{out}	V_{in} = 0 V f = 1 MHz Ta = 25°C	—	—	20	pF	
Current dissipation	During operation	I_{cc}	CLK = 12 MHz, RCLK = 15 MHz, V_{cc} = 5.25 V, Rated load	—	40	70	mA	

Note: 1. With reference to V_{ss}= 0 V.

AC Characteristics

Clock Cycle

Numbers in the leftmost columns of the tables on this and the following pages refer to numbers in figures 15 to 28.

No.	Item	Symbol	Min	Max	Unit	Note
1	Clock cycle time	t_{CYC}	81	250	ns	
2	Clock low level pulse width	t_{PWCL}	36	—	ns	
3	Clock high level pulse width	t_{PWCH}	36	—	ns	
4	Clock rise time	t_{CR}	—	5	ns	
5	Clock fall time	t_{CF}	—	5	ns	
6	Read clock cycle time	t_{RCYC}	66	250	ns	
7	Read clock low level pulse width	t_{RCL}	0.45	0.55	t_{RCYC}	
8	Read clock high level pulse width	t_{RCH}	0.45	0.55	t_{RCYC}	
9	Read clock rise time	t_{RCR}	—	5	ns	
10	Read clock fall time	t_{RCF}	—	5	ns	
11	Write clock cycle time	t_{WCYC}			= t_{RCYC}	
12	Write clock low level width	t_{WCL}	t_{RCL} – 10	t_{RCL} + 10	ns	
13	Write clock high level width	t_{WCH}	t_{RCH} – 10	t_{RCH} + 15	ns	
14	Write clock rise time	t_{WCR}	—	10	ns	
15	Write clock fall time	t_{WCF}	—	10	ns	
17	Read clock low level width 1	t_{RCL1}	—	400	ns	
18	Read clock high level width 1	t_{RCH1}	—	400	ns	
20	$\overline{RST}$ input pulse width	t_{RES}	30	—	t_{CYC}	
23	V_{cc} to CLK time	t_{VCT}	—	100	ms	
24	V_{cc} to $\overline{RST}$ time	t_{VRT}	100	—	ms	
25	$\overline{RST}$ rise time	t_{RRT}	—	20	ns	
26	$\overline{INT}$ delay time	t_{IDT}	—	110	ns	
27	$\overline{INT}$ release time	t_{IRT}	—	300	ns	

File Manager Interface

No.	Item		Symbol	Min	Max	Unit	Note
30	Address setup time		t_{AS1}	5	—	ns	1
31			t_{AS2}	10	—	ns	
32	Address hold time		t_{AH1}	10	—	ns	
33	$\overline{RD}$, $\overline{WR}$ pulse width	Except during TMA, TMC, or TMD read	t_{PW1}	2	—	t_{CYC}	
		During TMA, TMC, or TMD read		3	—		2
34	Read data delay time	Except during TMA, TMC, or TMD read	t_{DDR1}	—	100 ns	—	
		During TMA, TMC, or TMD read		—	1 t_{CYC} + 100 ns		2
35	Read data hold time		t_{HR1}	20	—	ns	
36	Write data setup time		t_{DSW1}	30	—	ns	
37	Write data hold time		t_{HW1}	20	—	ns	

Notes: 1. If t_{PW1} or t_{DDR1} is incremented by 1 t_{CYC}, the minimum value of t_{AS1} is −40 ns.
2. When the timers (TMA, TMB, TMC, and TMD) are being read, t_{PW1} min is 3 t_{cyc} and t_{DDR1} max is 1 t_{CYC} + 100 ns. The status of a timer is detected by reading the contents of STR2 and CEND.

Host Interface (slave mode)

No.	Item		Symbol	Min	Max	Unit	Note
40	$\overline{IORD}$, $\overline{IOWR}$ cycle time	8 bits	t_{HCYC}	3	—	t_{CYC}	
		16 bits		4	—		
41	$\overline{DACK}$ release after $\overline{IORD}$, $\overline{IOWR}$		t_{AKD}	0	—	ns	
42	$\overline{IORD}$, $\overline{IOWR}$ delay		t_{ACDS}	5	—	ns	1
43	$\overline{DREQ}$ release after $\overline{DONE}$		t_{DRQD}	0.5 t_{CYC}	1.5 t_{CYC} + 100 ns		
45	$\overline{DREQ}$ release after $\overline{DACK}$		t_{ARQD}	0.5 t_{CYC}	1.5 t_{CYC} + 100 ns		
46	$\overline{DREQ}$ release after $\overline{IORD}$, $\overline{IOWR}$		t_{CRQD1}	0.5 t_{CYC}	1.5 t_{CYC} + 100 ns		
48	$\overline{DREQ}$ release after $\overline{IORD}$, $\overline{IOWR}$ negation	8 bits	t_{CRQD2}	1.5 t_{CYC}	2.5 t_{CYC} + 100 ns		
		16 bits		2.5 t_{CYC}	3.5 t_{CYC} + 100 ns		
49	$\overline{DONE}$ low level pulse width		t_{DONE}	2	—	t_{CYC}	
50	$\overline{IORD}$, $\overline{IOWR}$ low pulse width		t_{PWL}	2 t_{CYC} – 10 ns	—		
51	$\overline{IORD}$, $\overline{IOWR}$ high level pulse width	8 bits	t_{PWH}	1 t_{CYC} – 10 ns	—		
		16 bits		2 t_{CYC} – 10 ns	—		
52	BUS_{00}-BUS_{15} read data delay time		t_{DDR2}	—	1 t_{CYC} + 70 ns		
53	BUS_{00}-BUS_{15} read data hold time		t_{HR2}	20	—	ns	
54	BUS_{00}-BUS_{15} write data setup time		t_{DSW2}	30	—	ns	
55	BUS_{00}-BUS_{15} write data hold time		t_{HW2}	20	—	ns	

Host Interface (slave mode) (cont)

No.	Item		Symbol	Min	Max	Unit	Note
56	BUS_{00}-BUS_{15} read data hold time after $\overline{IORD}$, $\overline{IOWR}$ negation	8 bits	t_{DDR3}	—	2 t_{CYC} + 70 ns		
		16 bits		—	3 t_{CYC} + 70 ns		
57	$\overline{DACK}$ response time		t_{AKR}	0	—	ns	
58	$\overline{DREQ}$ delay after $\overline{DACK}$		t_{RQDA}	0.5 t_{CYC}	1.5 t_{CYC} + 100 ns		

Note: 1. If t_{PWL} or t_{DDR2} is incremented by 1 t_{CYC}, the minimum value of t_{ACDS} is −40 ns.

Host Interface (master mode)

No.	Item	Symbol	Min	Max	Unit	Note
60	$\overline{DACK}$ delay time	t_{ACD}	0	80	ns	
61	$\overline{IORD}$, $\overline{IOWR}$ delay time	t_{RWD}	0	80	ns	
62	$\overline{DREQ}$ setup time	t_{RQS}	10	—	ns	
63	$\overline{DREQ}$ hold time	t_{RQH}	20	—	ns	
64	BUS_{00}-BUS_{07} read data setup time	t_{DSR}	60	—	ns	
65	BUS_{00}-BUS_{07} read data hold time	t_{HRM}	10	—	ns	
66	BUS_{00}-BUS_{07} write data delay time	t_{DDW}	—	50	ns	
67	BUS_{00}-BUS_{07} write data hold time	t_{HWM}	10	—	ns	
68	$\overline{WAIT}$ setup time	t_{WTS}	10	—	ns	
69	$\overline{WAIT}$ hold time	t_{WTH}	20	—	ns	
70	$\overline{DACK}$ release after $\overline{DONE}$	t_{DAKD}	2.5 t_{CYC}	3.5 t_{CYC} + 100 ns		
71	$\overline{DACK}$ release after $\overline{DREQ}$	t_{RAKD}	2.5 t_{CYC}	3.5 t_{CYC} + 100 ns		

Drive Interface

No.	Item	Symbol	Min	Max	Unit	Note
80	RDT setup time	t_{RDTS}	10	—	ns	
81	RDT hold time	t_{RDTH}	10	—	ns	
82	RDGT delay time	t_{RGTD}	0	100	ns	
83	WDT delay after WCLK	t_{WDTD}	—	15	ns	1
84	WDT hold after WCLK	t_{WDTH}	20	—	ns	
85	WRGT delay time	t_{WGTD}	0	100	ns	
86	AME delay time	t_{AMED}	0	100	ns	
87	IDX pulse width	t_{PWIDX}	2	—	t_{RCYC}	
88	S/AMF pulse width	t_{PWSA}	2	—	t_{RCYC}	
89	RDY, CC pulse width	t_{RDYCC}	2	—	t_{CYC}	
100	WDT delay after RCLK	t_{WDTD2}	—	30	ns	
101	WDT hold after RCLK	t_{WDTH2}	30	—	ns	

Note: 1. WDT setup time minimum from WCLK is 10 ns.

Port

No.	Item	Symbol	Min	Max	Unit	Note
90	P_0–P_7 $\overline{WR}$ setup time	t_{PWS}	30	—	ns	
91	P_0–P_7 write data delay time	t_{PWD}	—	100	ns	
92	P_0–P_7 read data setup time	t_{PRS}	30	—	ns	
93	P_0–P_7 read data hold time	t_{PRH}	30	—	ns	
94	P_0–P_7 $\overline{RD}$ delay time	t_{PRD}	30	—	ns	

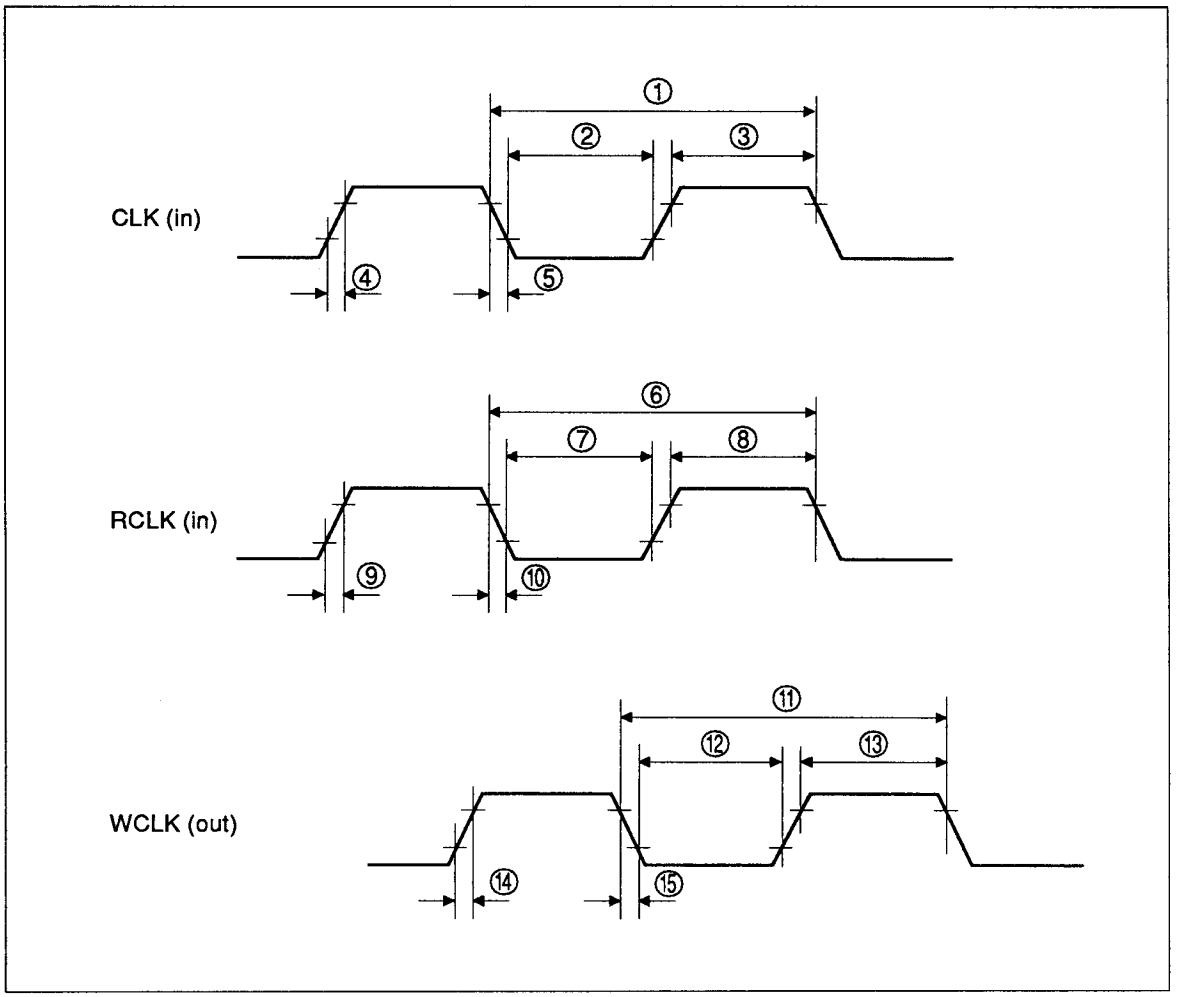


Figure 15 Clocks

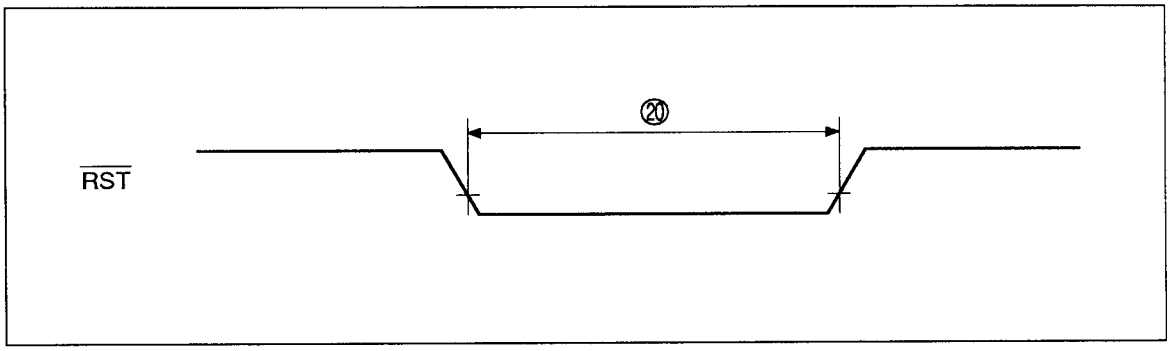


Figure 16 Reset

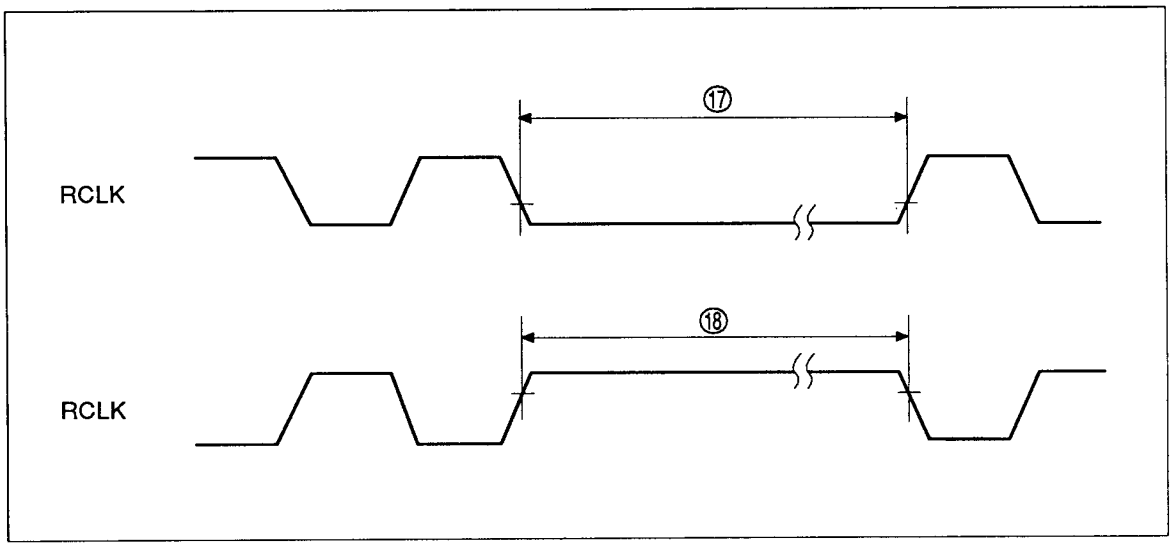


Figure 17 RCLK Waveform at Clock Switching (Non-operating)

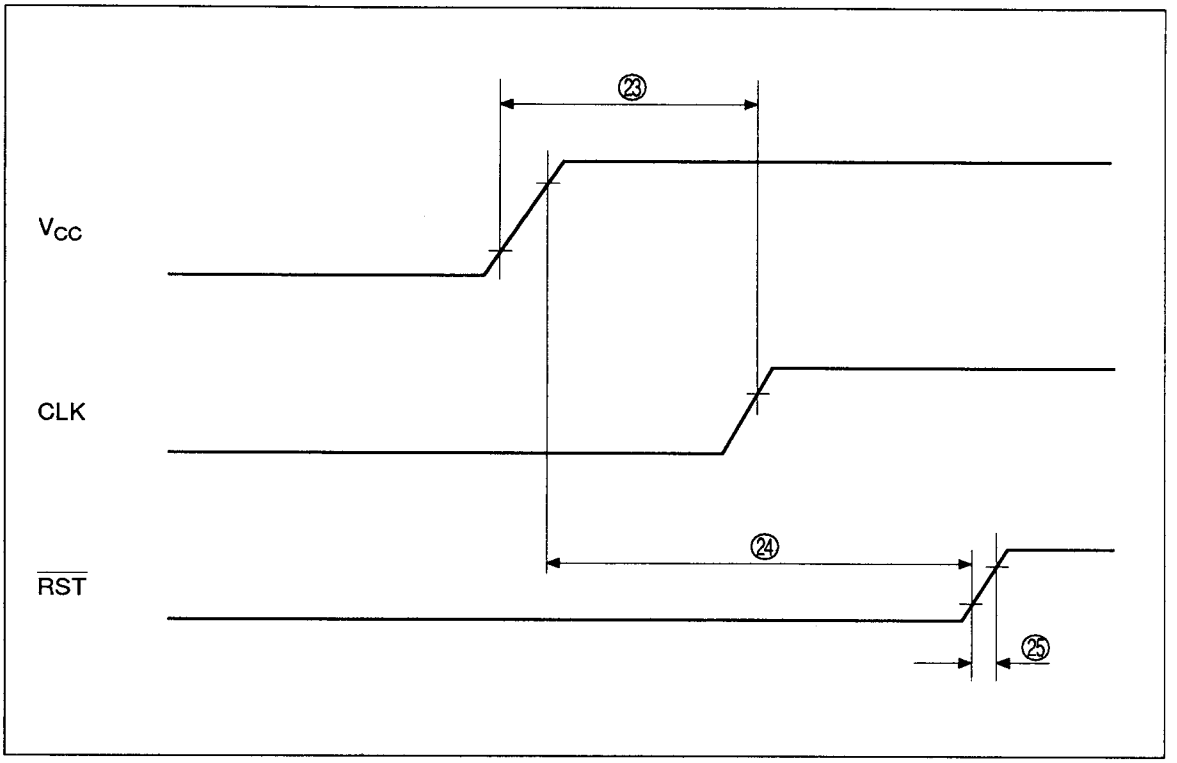


Figure 18 Power-on Timing

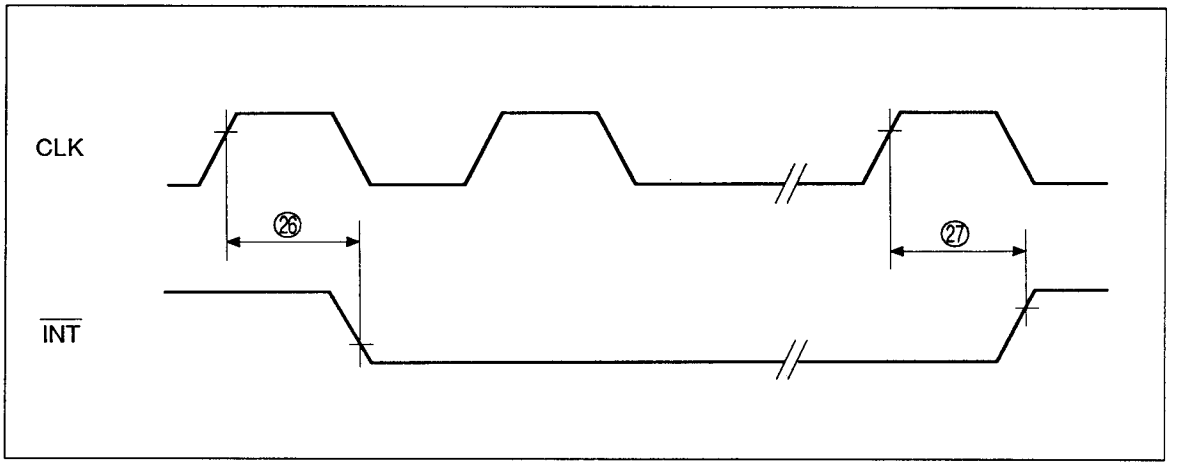


Figure 19 $\overline{INT}$ Output Timing

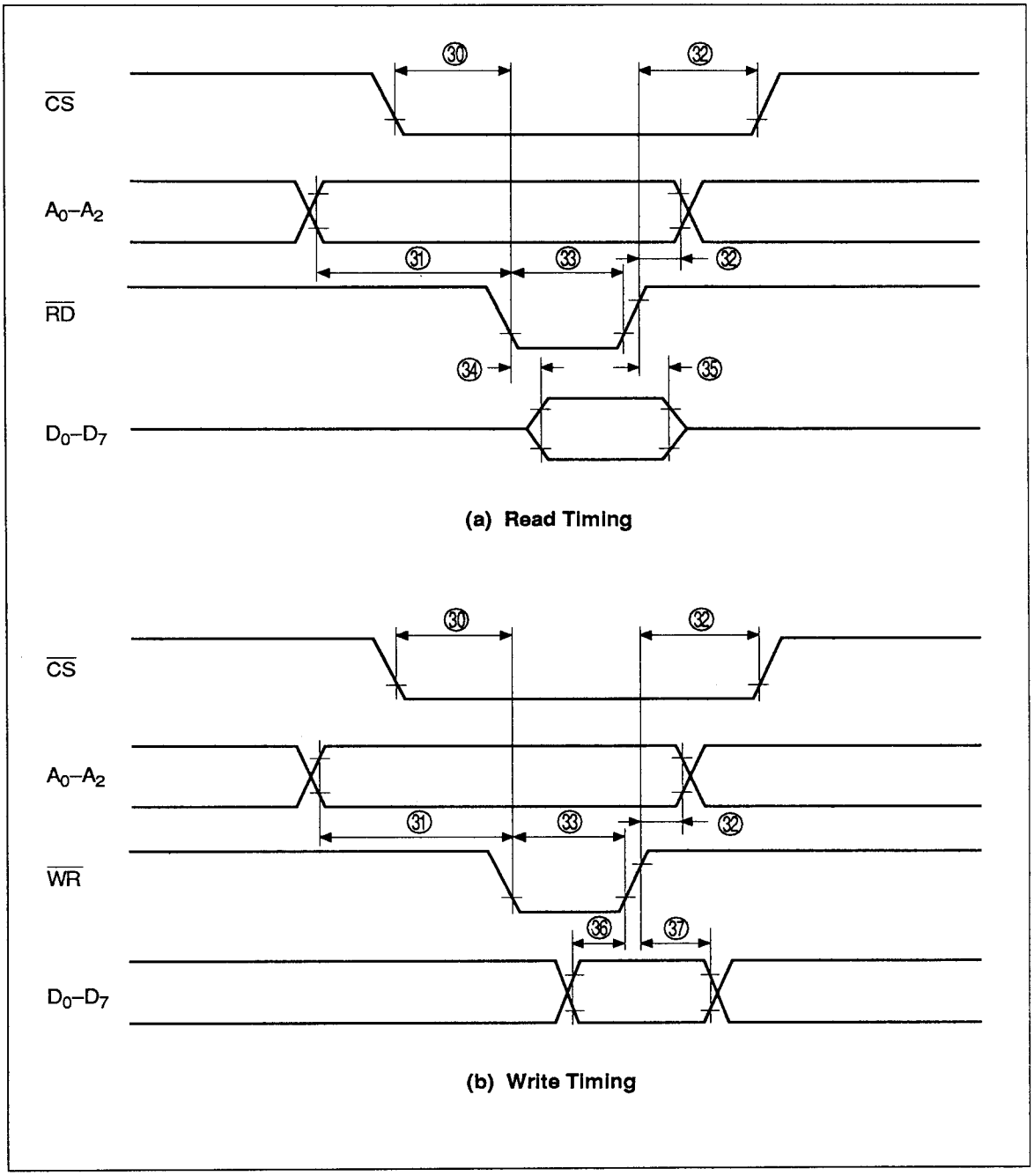


Figure 20 File Manager Interface

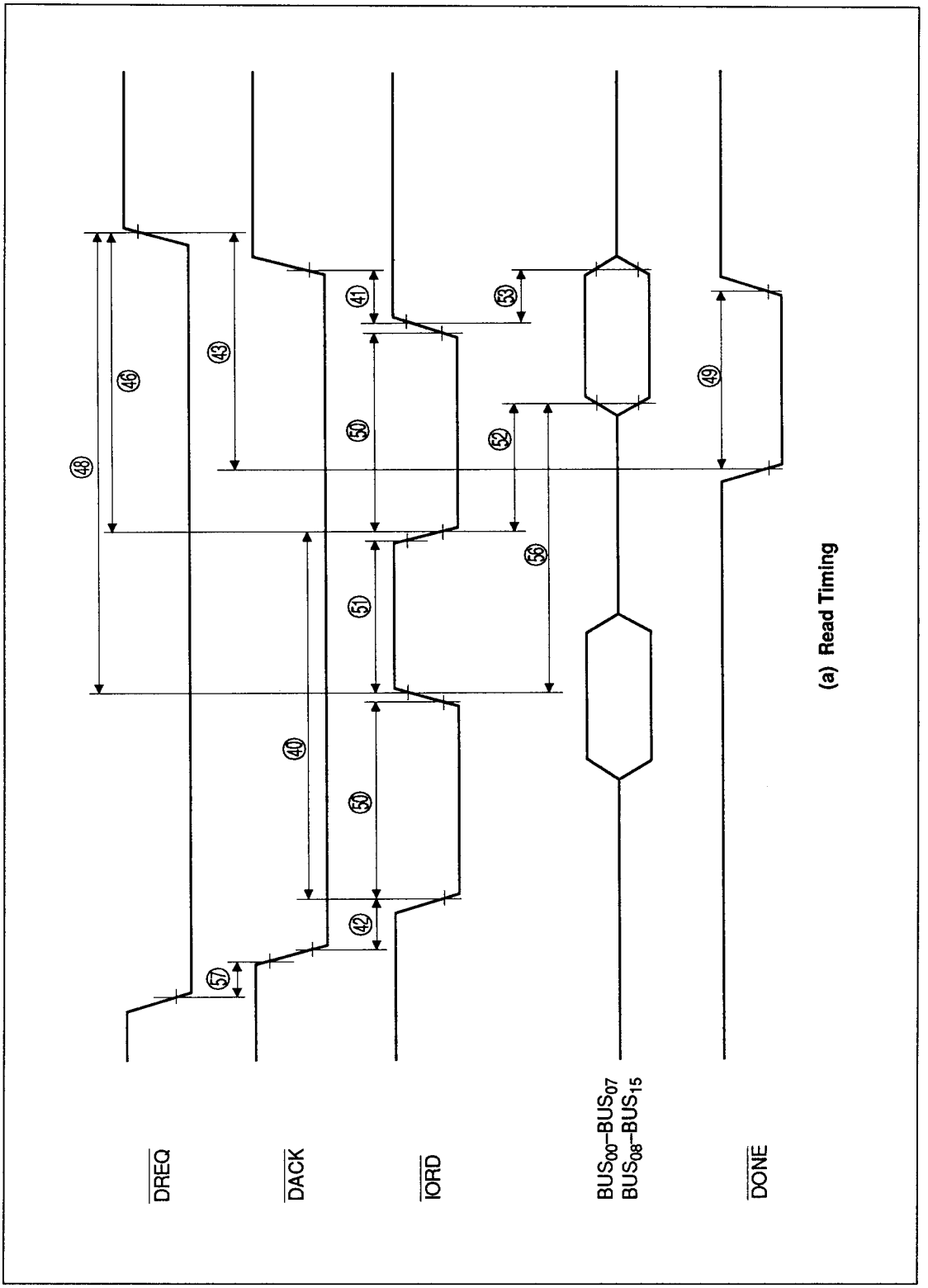


Figure 21 Host Interface (slave, burst mode DMA transfer)

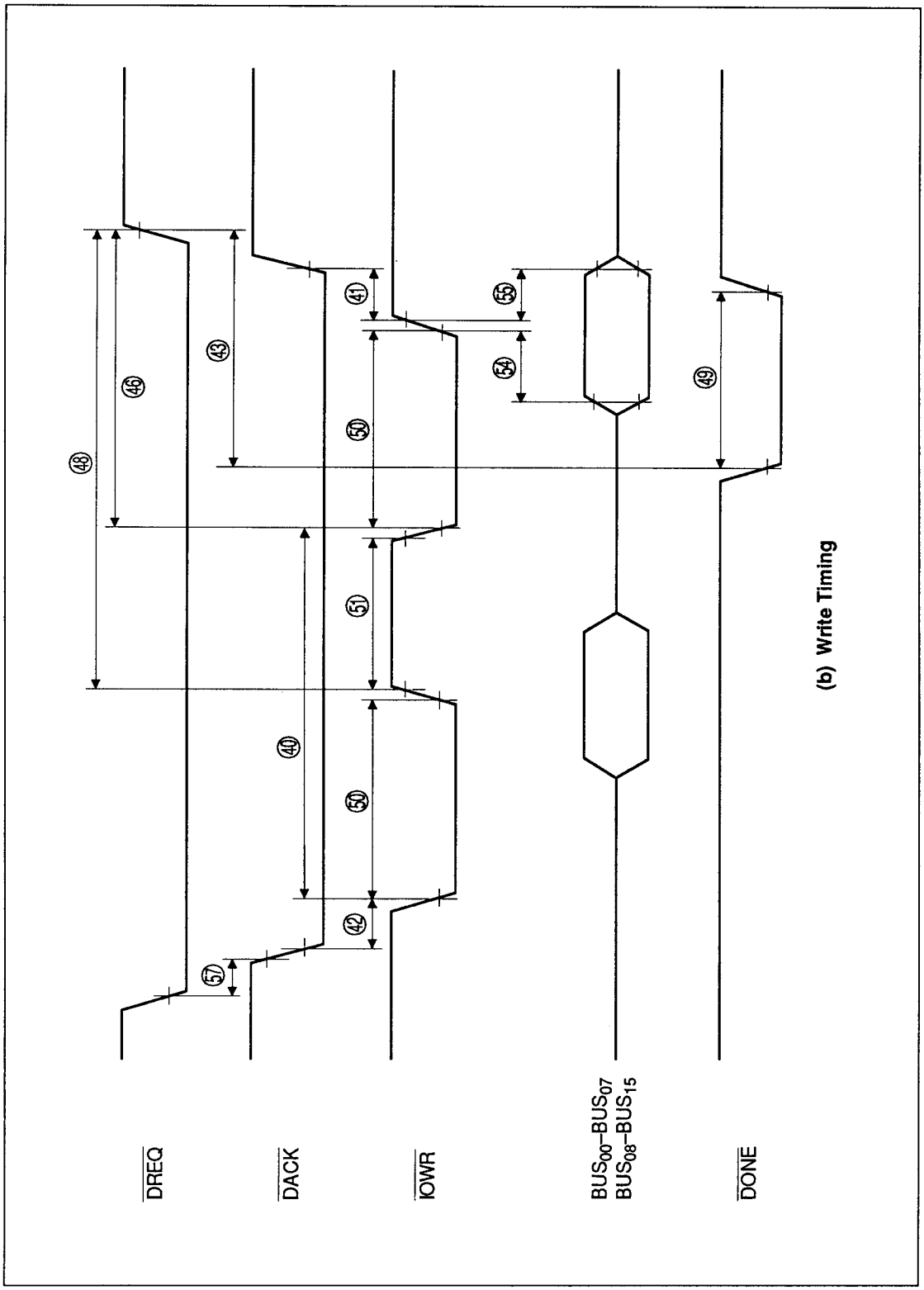


Figure 22 Host Interface (slave, burst mode DMA transfer) (cont)

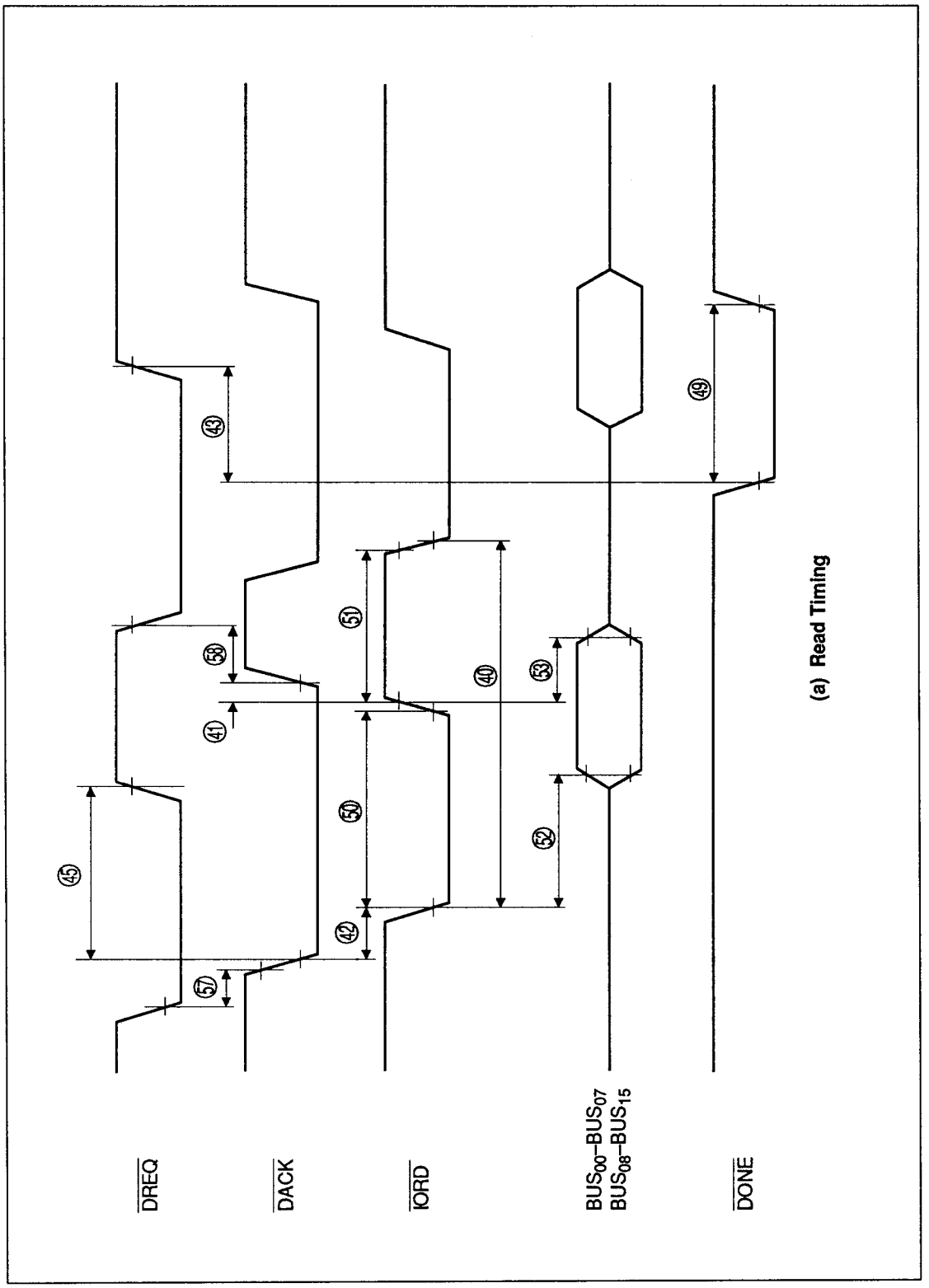


Figure 23 Host Interface (slave, cycle steal mode DMA transfer)

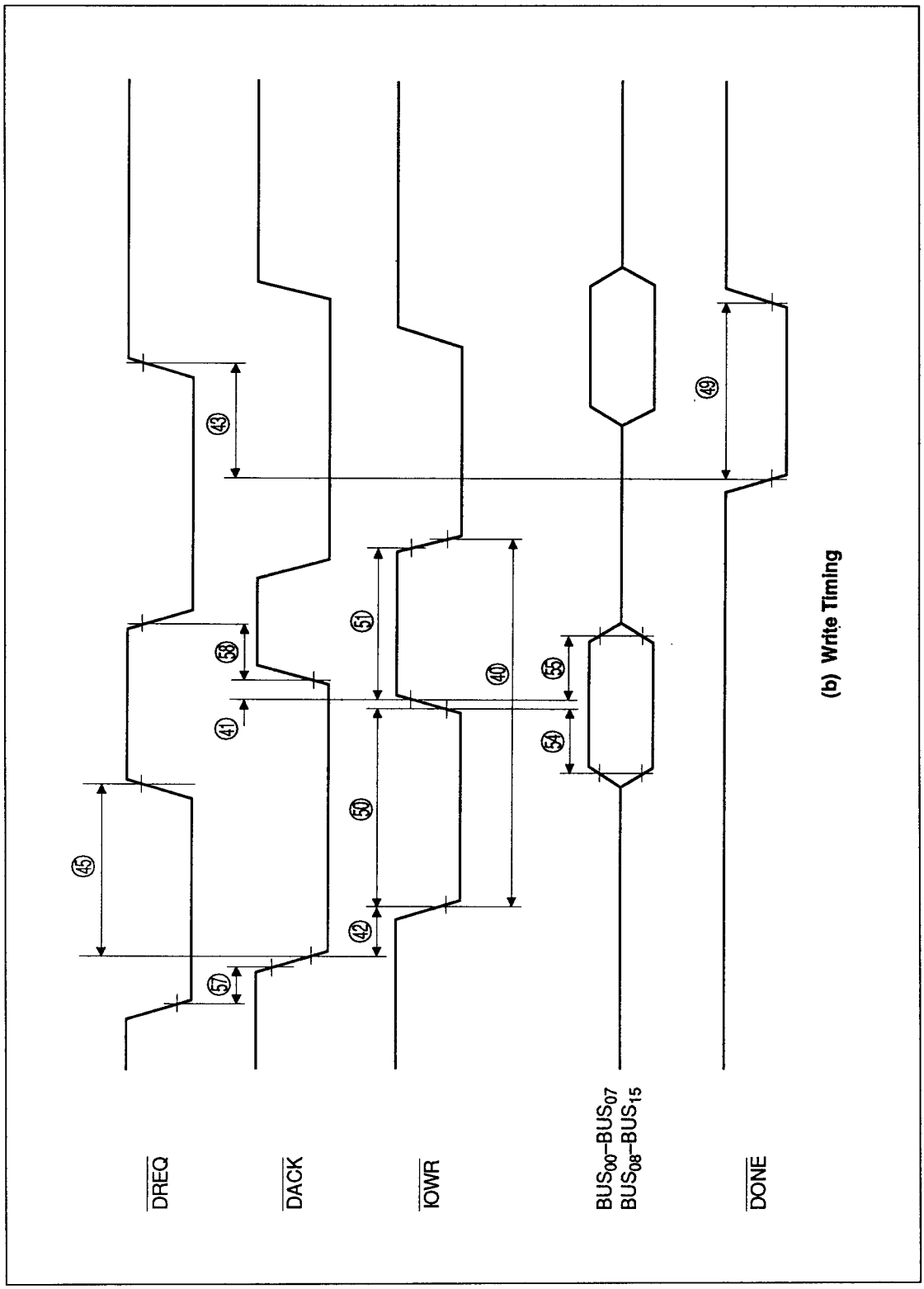
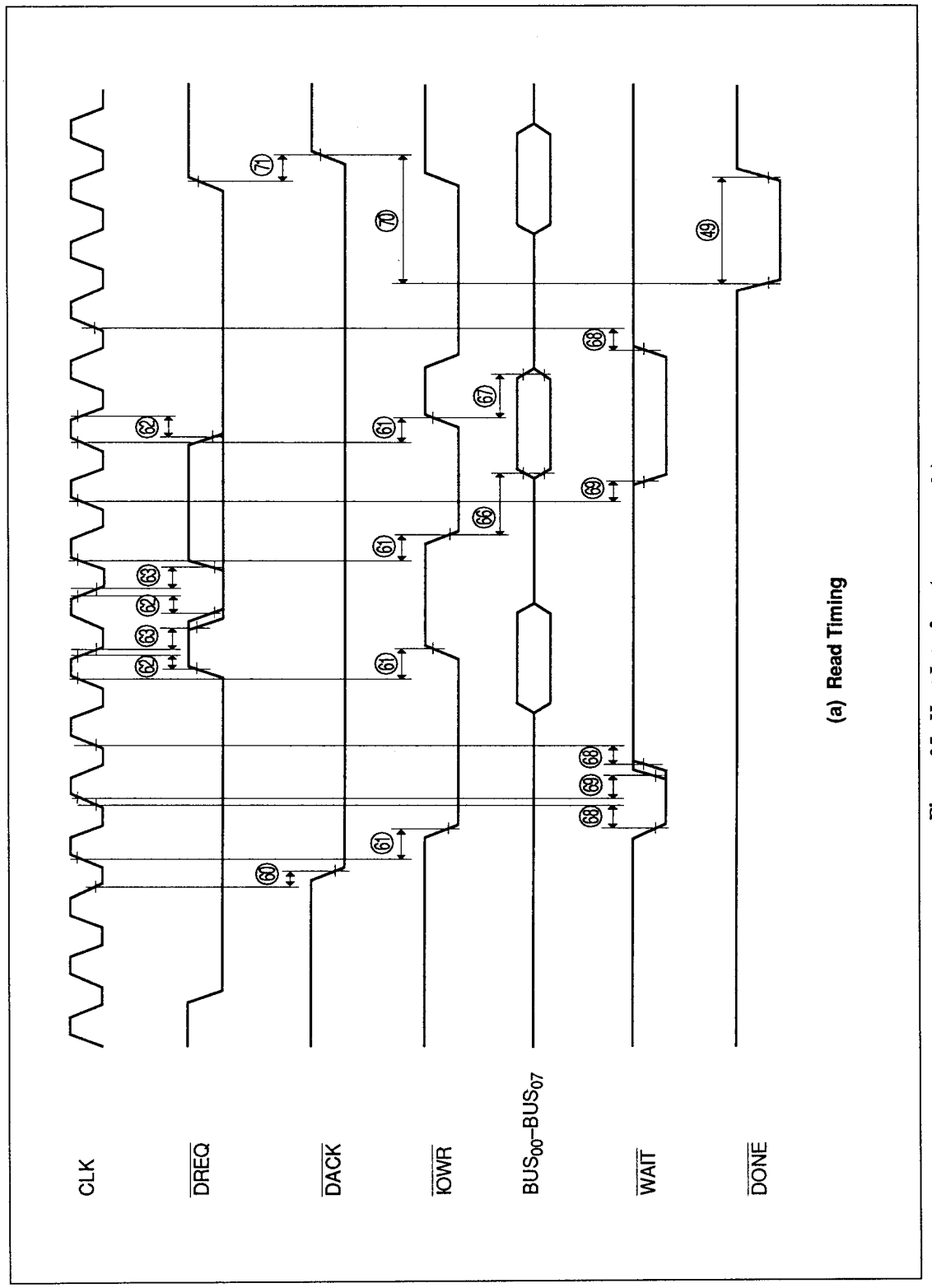


Figure 24 Host Interface (slave, cycle steal mode DMA transfer) (cont)



(a) Read Timing

Figure 25 Host Interface (master mode)

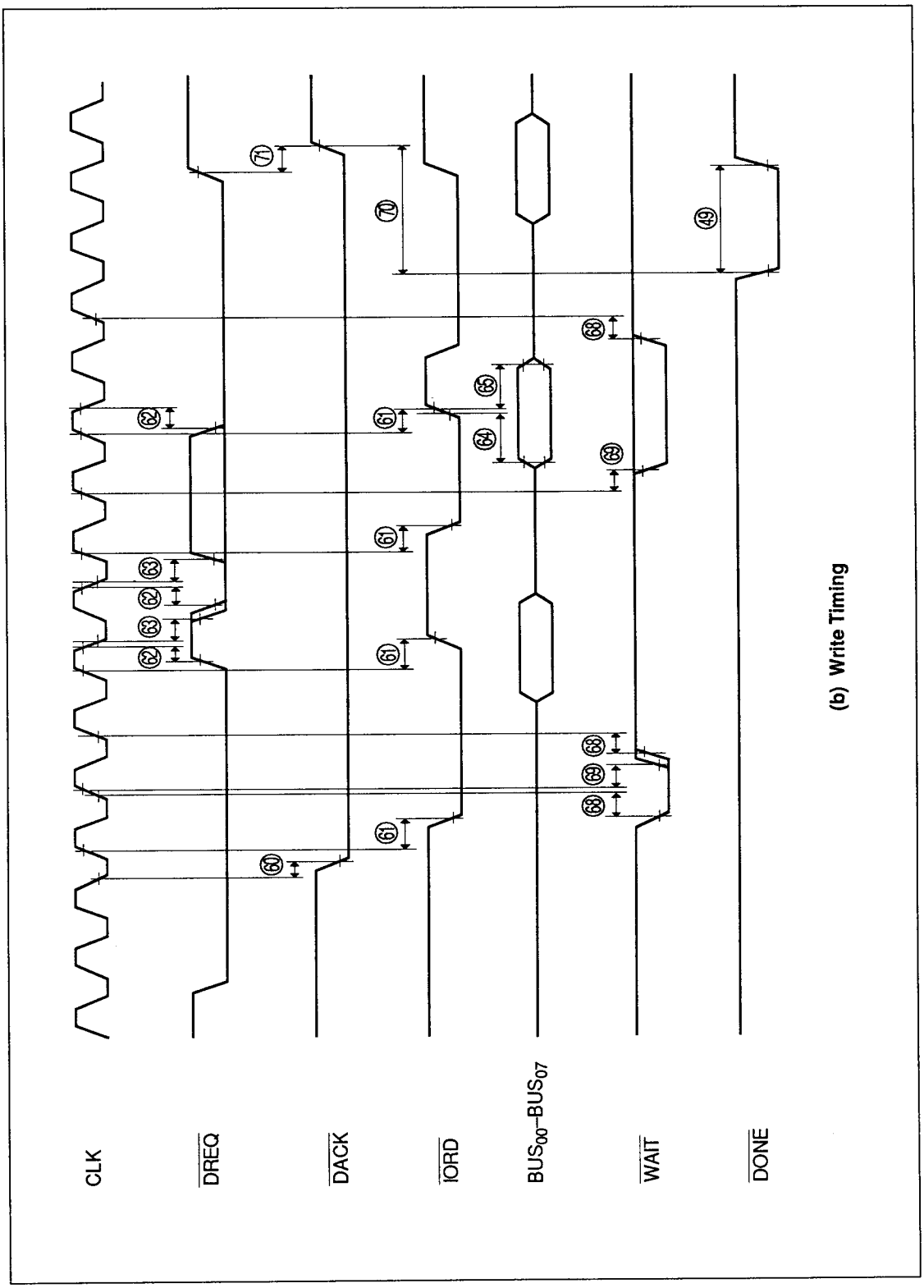


Figure 26 Host Interface (master mode) (cont)

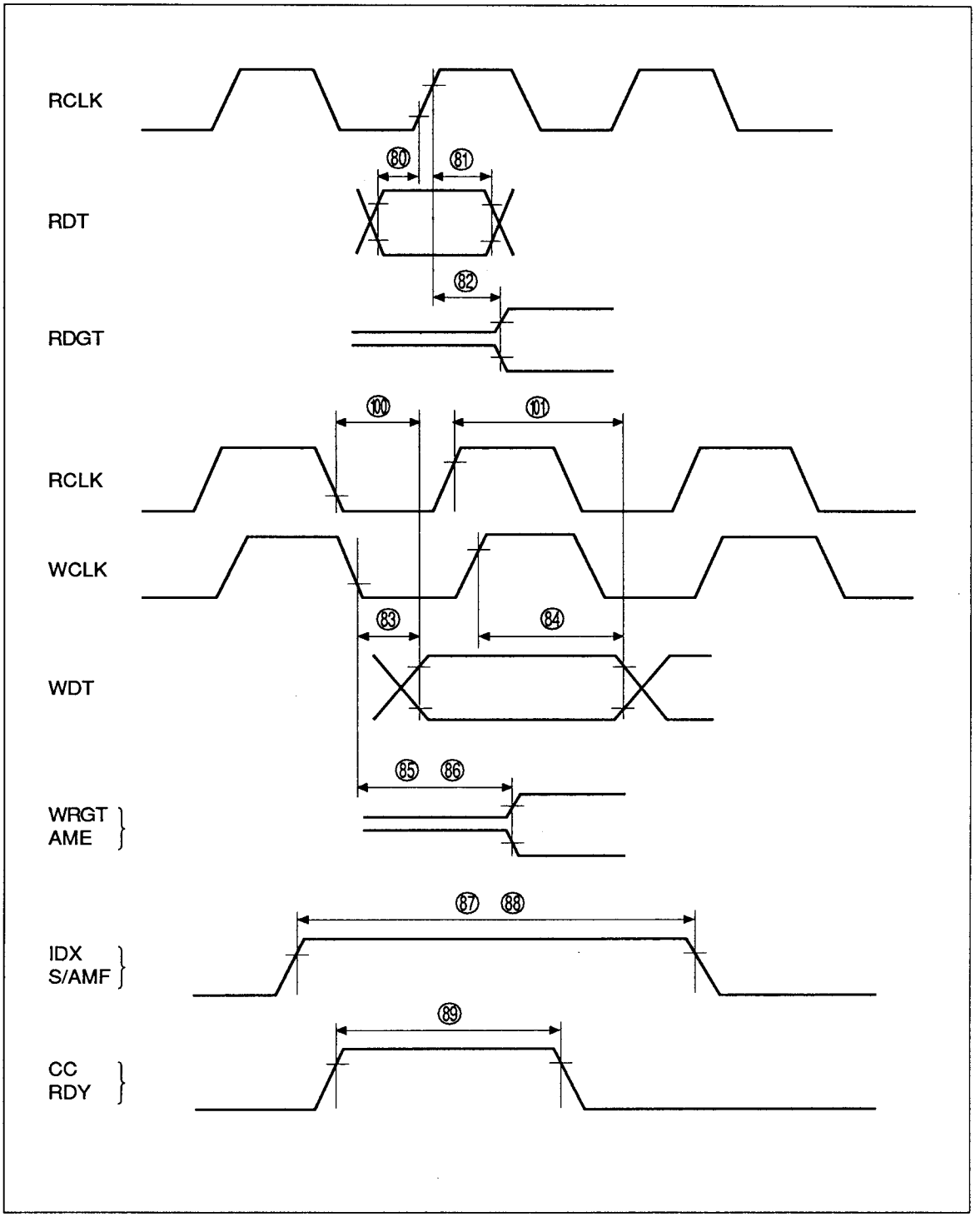


Figure 27 Drive Interface

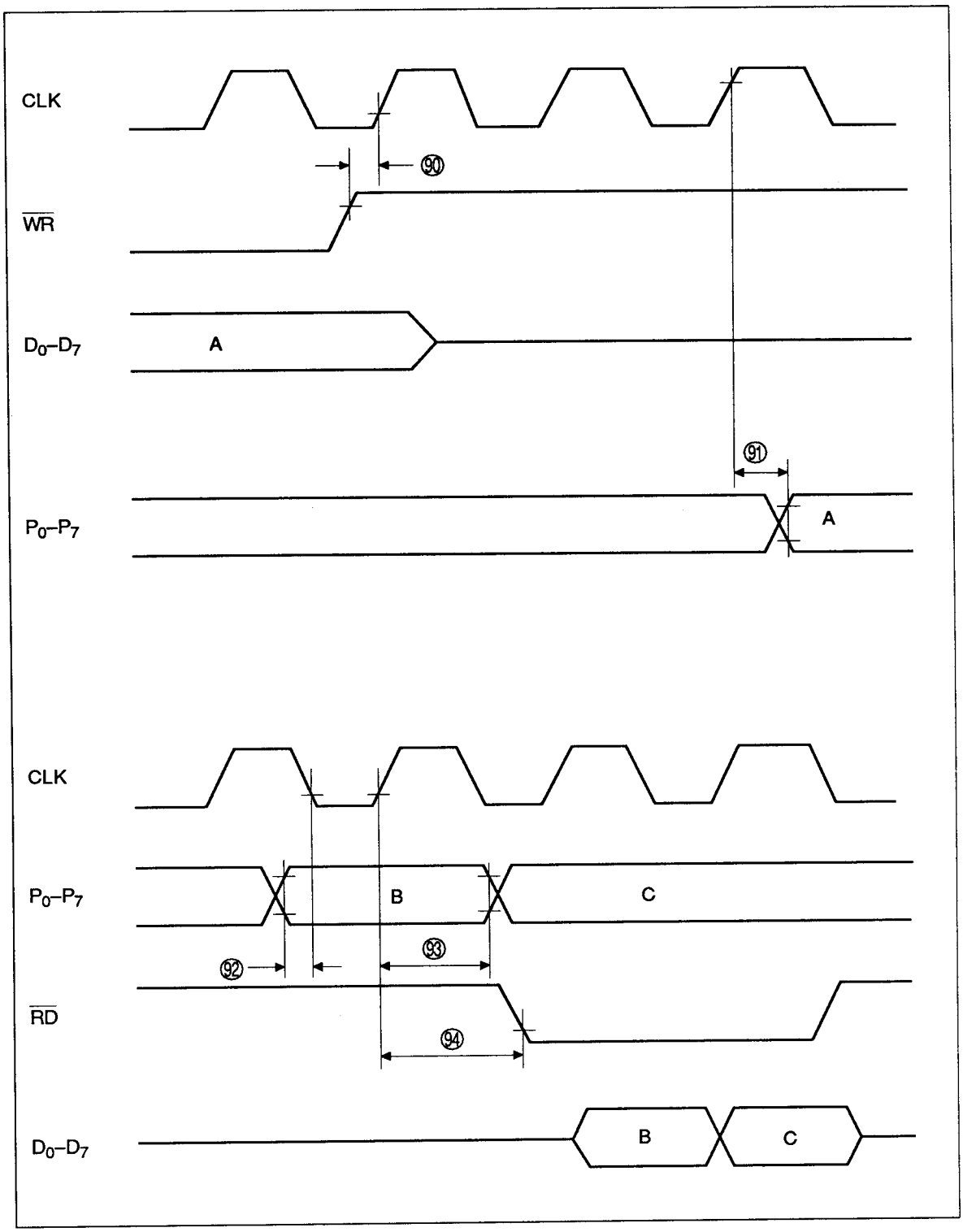


Figure 28 Ports