FWD8006
MULTIBUS Compatible
Winchester/Floppy Controller
OEM Manual

FWD8006

MULT[illegible]le

Winchester/Floppy Controller

OEM MANUAL

Document Number: 3000822
Revision: D
Date: May 15, 1984

APPLICABLE PRODUCT NUMBERS
PWB Assy: 0004352-0001
Prog Assy: 1002035-0001

Warning: This equipment generates, uses, and can radiate radio frequency energy and if not installed and used in accordance with the instructions manual, may cause interference to radio communications. As temporarily permitted by regulation it has not been tested for compliance with the limits for Class A computing devices pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference. Operation of this equipment in a residential area is likely to cause interference in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

CONTENTS

FIGURES

TABLES

I. CONTROLLER OVERVIEW

A. INTRODUCTION AND FEATURES

The FWD8006 is a complete single board controller which allows the user to interface up to two 5-1/4" Winchester and two 8" floppy disk drives, two 5-1/4" Winchester and two 5-1/4" floppy disk drives or two 5-1/4" Winchester and two Sony 3-1/2" floppy disk drives* to any Multibus compatible computer system. A unique controller architecture utilizing a multi-processor combination of high integration MOS LSI components and high speed bipolar circuitry is used to eliminate additional PC boards, special cabling, etc., while providing high reliability and performance through careful conservative design, reduced component count, reduced interconnections, and reduced power consumption.

- Fully replaces the three board controller system consisting of the Intel iSBC 215, the Intel iSBX 218, and the Shugart SA1200.
- Software compatible with the iSBC 215 / iSBX 218 combination to allow plug in replacement without costly software handler or file system modifications.
- Comprehensive self test and drive diagnostics on board.
- Patented data recovery phase locked loop design.
- Single PC board requires only a single slot in Multibus backplane.
- Program selectable formats for floppy and Winchester disk.
- Supports 8" diskette compatible with the following formats for industry standard media exchangeability: IBM Diskette 1, IBM Diskette 2, IBM Diskette 2D, Intel SBC 202, and Intel single density floppy development systems and controllers.
- Supports 5-1/4" diskette compatible with the following formats for interchangeability between various systems: Intel iRMX-86 single density, Intel iRMX-86 double density, DEC personal computer, and SMS FDC807-M and FWD0106 controllers.
- Supports Sony 3-1/2" diskette compatible with FM and MFM formats.*
- Seek overlap.
- Supports 8 or 16 bit data transfers on a buffer by buffer basis.

* The connection to Sony drives is via the 50 pin edge connector J1 with a scramble cable to re-route all the drive signals.

- o Supports 16 bit, 2[illegible], or 24 bit (IEEE 796) memory addressing to allow access to up to [illegible] Megabytes of memory.
- o Supports either 8 bit or 16 bit I/O addressing.
- o Low Winchester disk error rate and high data integrity guaranteed through the implementation of error correction (ECC).
- o Programmable full sector buffering or direct transfer for both floppy and Winchester drives.

B. FWD8006 COMMAND and STATUS

Except for hardware Start, Stop, and Reset controls, all command and status information associated with the FWD8006 is communicated through Multibus memory based tables. This mechanism allows for detailed control of the disk system by the host CPU and for extensive status returns to the host with a minimum of hardware handshaking and hardware protocol. The CPU overhead required for disk operation is reduced while the level of data transfer flexibility and control is enhanced. The FWD8006 is an intelligent controller. It is capable of large block transfers across sector, track, and cylinder boundaries without host intervention. It incorporates extensive error processing facilities to improve data recovery and to relieve the host of error handling responsibilities except in the rare case of an unrecoverable error. And if the controller or disk drive system should fail, the FWD8006 contains extensive built in diagnostics to quickly isolate the fault to the controller or a drive for minimum repair time.

C. FWD8006 DATA TRANSFER

Both direct and buffered data transfers are supported by the FWD8006. For Winchester disk transfers where the data rate approaches the bandwidth of the host Multibus system, data transfers can be direct, by locking the Multibus during transfer time, or buffered through an on board full sector buffer. Floppy disk data transfers, on the other hand, occur at a low enough rate that bus access constraints are far less severe. This allows the FWD8006 to do direct disk to Multibus transfers on all floppy accesses without locking the Multibus. In addition, for systems with mixed memory capabilities such as occurs with the combination of the 16K bytes of byte-access-RAM on an iSBC 80/30 CPU board together with a byte/word access iSBC 032 memory board, for example, the FWD8006 allows the access width (byte or word) to be specified for each data buffer independently to take maximum advantage of the increased speed and reduced Multibus bandwidth requirements of word DMA transfers.

D. FWD8006 ECC/CRC

The IBM standard 16 bit CRC is used for error detection on all sector ID fields and floppy disk data fields. The error rate for these fields is lower than for Winchester data fields and the CRC facility provides an excellent error detection capability. Winchester disk data integrity is provided by a 32 bit computer generated error correcting code designed especially for the types of errors found on Winchester technology disks. This code will detect all bursts of 15 bits or less and will correct all bursts of 6 bits or less. This span was chosen to cover the majority of Winchester errors while minimizing the probability of miscorrection. All error correction codes, including the SMS code, have some probability of miscorrection. This occurs when a group of independent errors produce the same error residue as would have occurred with a certain correctable error. To prevent this situation, the FWD8006 saves the error residue from the first read, reads the sector again, and compares the residues before making a correction. This process virtually guarantees user data integrity. The price for this reliability is that two disk revolutions, rather than one, elapse whenever an error is encountered. This results in negligible performance loss.

E. FUNCTIONAL SPECIFICATI...

Maximum number of drives:	Two 8" floppies + two 5-1/4" Winchesters or Two 5-1/4" floppies + two 5-1/4" Winchesters or Two Sony 3-1/2" floppies + two 5-1/4" Winchesters
8" floppy drives supported:	Shugart SA810, SA860, SA850, Tandon TM848-1, TM848-2
8" diskette formats:	128, 256, 512, and 1024 byte sectors in IBM Diskette 1, Diskette 2, and Diskette 2D formats. 128 byte sectors in Intel SBC 202 format.
5-1/4" floppy drives supported:	Shugart SA400, SA450, SA410, SA460, SA455, SA465 Tandon TM100-3, TM100-4 Micro Peripherals MPI91, MPI92 TEAC FD-55B,55F
5-1/4" diskette formats:	128, 256, 512, and 1024 byte sectors in Intel iRMX-86 FM format. 256, 512, and 1024 byte sectors in Intel iRMX-86 MFM format. 512 byte sectors in IBM personal computer format. 512 byte sectors in DEC personal computer format. 128, 256, 512, and 1024 byte sectors in SMS FD0507-M MFM format. 128, 256, 512, and 1024 byte sectors in SMS FWD0106 MFM format.
3-1/2" floppy drives supported:	Sony OA-D30V
3-1/2" diskette formats:	128, 256,512 byte sectors in FM format 256,512,1024 byte sectors in MFM format
5-1/4" Winchester drives supported:	Seagate ST506, ST406, ST412, ST419 and compatible.
Winchester drive formats:	MFM encoding with program selectable sector lengths of 128, 256, 512 or 1024 bytes.
Data transfer:	Winchester disk transfers can be buffered or direct by locking the Multibus. Floppy disk transfers can also be buffered or direct without locking the Multibus. Multibus transfer cycles can be either byte or word wide on a buffer by buffer basis.

I/O address selection:	Defined by hardware strapping. Slip-jumpers select either 8 bit I/O addressing or 16 bit addressing for full Multibus addressing range compatibility.
Interrupt level selection:	Defined by hardware strapping. Wire-wrap jumper selects one of 8 standard Multibus parallel interrupt request lines.

PHYSICAL SPECIFICATIONS

Environmental:

Relative humidity: 10% to 90%

Operating temperature*: 0 to 50 degrees C.

Storage temperature: -65 to 70 degrees C

*Extreme temperatures may require forced air cooling, i.e. 50 degrees C ambient requires 150 linear feet/minute over component side of PC board in a typical card cage application.

Electrical:

Operating voltage: 5V (5% tolerance)

Ripple and noise must not exceed 100 mV P-P.

Operating curren[illegible])
5.2A (maximum)

Power is supplied by the host computer via the standard Multibu[illegible] BUS pins of connector P1.

Mechanical:

The controller PC board conforms to Multibus board form factors as detailed in the IEEE 796 Bus Specification.

C. DISK FORMATS AND C[illegible]TIES

Table I-1. 8" FLOPPY DISKETTE FORMATS

Format	Density	Code	Byte/Sect	Sect/Trk	Sides	Cyl	Bytes/Diskette
IBM 1 *	Single	FM	128	26	1	77	256,256
IBM 1 *			256	15			295,680
IBM 1 *			512	8			315,392
			1024	4			315,392
			128	26	2	77	512,512
IBM 2 *			256	15			591,360
			512	8			630,784
			1024	4			630,784
	Double	MFM	128	44	1	77	433,664
			256	26			512,512
			512	15			591,360
			512	16			630,784
			1024	8			630,784
			128	44	2	77	867,328
IBM 2D *			256	26			1,025,024
IBM 2D *			512	15			1,182,720
			512	16			1,261,568
IBM 2D *			1024	8			1,261,568
SBC 202	Double	MFM	128	52	1	77	512,512

Formats marked * are fully compatible with the indicated IBM format. Formats listed without * are logical extensions of IBM formats.

The 202 format is fully compatible with the Intel SBC 202 double density floppy disk controller.

Table I-2. 5-1/4" FLOPPY DISKETTE FORMATS

Format	Density	Code	Byte/Sect	Sect/Trk	Sides	Cyl	Bytes/Diskette
FD0507-M	Double	MFM	128	30	1	35	134,400
			256	18	1	35	161,280
			512	10	1	35	179,200
			1024	5	1	35	179,200
			128	30	1	40	153,600
			256	18	1	40	184,320
			512	10	1	40	204,800
			1024	5	1	40	204,800
			128	30	1	80	307,200
			256	18	1	80	368,640
			512	10	1	80	409,600
			1024	5	1	80	409,600
			128	30	2	35	268,800
			256	18	2	35	322,560
			512	10	2	35	358,400
			1024	5	2	35	358,400
			128	30	2	40	307,200
			256	18	2	40	368,640
			512	10	2	40	409,600
			1024	5	2	40	409,600
			128	30	2	80	614,400
			256	18	2	80	737,280
			512	10	2	80	819,200
			1024	5	2	80	819,200
FWD0106	Double	MFM	128	26	1	80	266,240
			256	16	1	80	327,680
			512	10	1	80	409,600
			1024	5	1	80	409,600
			128	26	2	80	532,480
			256	16	2	80	655,360
			512	10	2	80	819,200
			1024	5	2	80	819,200

Table I-1 5/4" FLOPPY DISKETTE FORMATS (continued)

Format	Density	Code	Byte/Sect	Sect/Trk	Sides	Cyl	Bytes/Diskette
iRMX-86	Single	FM	128	16	1	35	71,680
			256	9	1	35	80,640
			512	4	1	35	71,680
			1024	2	1	35	71,680
			128	16	2	35	143,360
			256	9	2	35	161,280
			512	4	2	35	143,360
			1024	2	2	35	143,360
			128	16	1	80	163,840
			256	9	1	80	184,320
			512	4	1	80	163,840
			1024	2	1	80	163,840
			128	16	2	80	327,680
			256	9	2	80	368,640
			512	4	2	80	327,680
			1024	2	2	80	327,680
	Double	MFM	256	16	1	35	143,360
			512	8	1	35	143,360
			1024	4	1	35	143,360
			256	16	2	35	286,720
			512	8	2	35	286,720
			1024	4	2	35	286,720
			256	16	1	80	327,680
			512	8	1	80	327,680
			1024	4	1	80	327,680
			256	16	2	80	655,360
			512	8	2	80	655,360
			1024	4	2	80	655,360
IBM PC	Double	MFM	512	8	1	40	163,840
			512	8	2	40	327,680
IBM PC	Double	MFM	512	9	1	40	184,320
			512	9	2	40	368,640
DEC PC	Double	MFM	512	10	1	80	409,600

Table I-3. 3-1/2" FLOPPY DISKETTE FORMATS

Format	Density	Code	Byte/Sect	Sect/Trk	Sides	Cyl	Bytes/Diskette
Sony	Single	FM	128	15	1	70	134,400
			256	9	1	70	161,280
			512	4	1	70	143,360
	Double	MFM	256	15	1	70	268,800
			512	9	1	70	322,560
			1024	4	1	70	286,720

Table I-4. 5-1/4" WINCHESTER DISK FORMATS

Vendor	Model	Byte/Sect	Sect/Trk	Heads	Cyl	Formatted Capacity (MB)
Seagate	ST506	128	54	4	153	4.230
		256	31			4.857
		512	17			5.327
		1024	9			5.640
	ST406	128	54	2	306	4.230
		256	31			4.857
		512	17			5.327
		1024	9			5.640
	ST412	128	54	4	306	8.460
		256	31			9.714
		512	17			10.654
		1024	9			11.280
	ST419	128	54	6	306	12.690
		256	31			14.570
		512	17			15.981
		1024	9			16.921
Atasi	3033	128		5	645	22.291
		256	31			25.594
		512	17			28.070
		1024	9			29.722
	3046	128	54	7	645	31.208
		256	31			35.831
		512	17			39.299
		1024	9			41.610
Maxtor	XT-1065	128	54	7	918	44.417
		256	31			50.997
		512	17			55.932
		1024	9			59.222
	XT-1105	128	54	11	918	69.797
		256	31			80.138
		512	17			87.893
		1024	9			93.063
	XT-1140	128	54	15	918	95.178
		256	31			109.279
		512	17			119.854
		1024	9			126.904

Table I-4. 5 1/4" WINCHESTER DISK FORMAT (continued)

Vendor	Model	Byte/Sect	Sect/Trk	Heads	Cyl	Formatted Capacity (MB)
Rodime	0204	128	54	8	320	17.695
Ampex	PYSIS27	256	31			20.316
		512	17			22.282
		1024	9			23.593
Rodime	0208	128	54	8	640	35.389
		256	31			40.632
		512	17			44.564
		1024	9			47.186
Vertex	V130	128	54	3	987	20.466
		256	31			23.498
		512	17			25.772
		1024	9			27.288
	V150	128	54	5	987	34.110
		256	31			39.164
		512	17			42.954
		1024	9			45.481
	V170	128	54	7	987	47.755
		256	31			54.830
		512	17			60.136
		1024	9			63.673
Tandon	TM703	128	54	5	600	20.736
		256	31			23.808
		512	17			26.112
		1024	9			27.648
CMI	CM6426	128	54	4	640	17.695
		256	31			20.316
		512	17			22.282
		1024	9			23.593
	CM6640	128	54	6	640	26.542
		256	31			30.474
		512	17			33.423
		1024	9			35.389

Table I-4. 5 1/4" WINCHESTER DISK FORMAT (continued)

Vendor	Model	Byte/Sect	Sect/Trk	Heads	Cyl	Formatted Capacity (MB)
Evotek	ET5530	128	54	6	375	15.552
		256	31			17.856
		512	17			19.584
		1024	9			20.736
	ET5540	128	54	8	375	20.736
		256	31			23.808
		512	17			26.112
		1024	9			27.648
Quantum	Q520	128	54	4	512	14.155
		256	[illegible]			16.253
		512	[illegible]			17.826
		1024	9			18.874
	Q540	128	54	8	512	28.311
		256	31			32.506
		512	17			35.651
		1024	9			37.748

Note: Refer to section IV for detailed description of disk formats. Refer to Table I-9 for compatibility.

Table I-5. 8" FLOPPY DRIVE PARAMETERS

Drive Model	Unformatted Capacity (MB)	Cyl	Sides	Rotational Latency [2]	Avg Access Time (ms)	Step Time (ms)	Settling Time (ms)
SA810	0.40 S[1] 0.80 D	77	1	83.33	89	3	16
SA860	0.80 S 1.60 D	77	2	83.33	89	3	16
TM848-1	0.40 S 0.80 D	77	1	83.33	91	4	15
TM848-2	0.80 S 1.60 D	77	2	83.33	91	4	15
TM848-2E	0.80 S 1.60 D	77	2	83.33	89	3	16
M2896-63	0.80 S 1.60 D	77	2	83.33	89	3	16
SA850	0.80 S 1.60 D	77	2	83.33	91	3	15
SA800	0.80 S 1.60 D	77	1	83.33	91	8	8

[1] S = single density; D = double density.

[2] Max data transfer rate is 31.25 KBytes/second for single density and 62.50 Kbytes/second for double density.

[3] Average data transfer rate based on 256 bytes/sector without interleave is 23 KBytes/second for single density and 40 KBytes/second for double density. A 2:1 interleave would cut average transfer rate by two.

-6. 5-1/4" FLOPPY DRIVE PARAMETERS

Drive Model	Unformatted Capacity (Kb)	Cyl	Sides	Rotational Latency (ms)	Avg Access Time (ms)	Step Time (ms)	Settling Time (ms)
SA400	109.4 S[1]	35	1	100	463	40	10
SA450	218.8 S 437.5 D	35	2	100	298	25	15
SA410	250.0 S 500.0 D	80	1	100	158	6	15
SA460	500.0 S 1000.0 D	80	2	100	158	6	15
MPI91	250.0 S 500.0 D	80	1	100	133	5	15
MPI92	500.0 S 1000.0 D	80	2	100	133	5	15
TM100-3	250.0 S 500.0 D	80	1	100	90	5	15
TM100-4	500.0 S 1000.0 D	80	2	100	90	5	15
FD-55B SA455	250.0 S 500.0 D	40	2	100	93	6	15
FD-55F	500.0 S 1000.0 D	80	2	100	94	6	15
SA465	500.0 S 1000.0 D	80	2	100	94	3	15

[1] S = single density; D = double density.

[2] Max data transfer rate is 15.63 Kbytes/second for single density and 31.25 Kbytes/second for double density.

[3] Average data transfer rate based on 256 bytes/sector without interleave is 11.5 KBytes/second for single density and 20.0 Kbytes/second for double density. A 2:1 interleave would cut average transfer rate by two.

Table I-7. 3-1/2" FLOPPY DRIVE PARAMETERS

Drive Model	Unformatted Capacity (MB)	Cyl	Sides	Rotational Latency (ms)	Avg Access Time (ms)	Step Time (ms)	Settling Time (ms)
QA-D33V	250.0 S[1] 500.0 D	80	2	100.0 ms	350	12	30

[1] S = single density; D = double density.

[2] Max data transfer rate is 31.25 Kbytes/second for single density and 62.50 Kbytes/second for double density.

[3] Average data transfer rate based on 256 bytes/sector without interleave is 23 KBytes/second for single density and 38 KBytes/second for double density. A 2:1 interleave would cut average transfer rate by two.

Table -3. 5-1/4" WINCHESTER DRIVE PARAMETERS

Drive Model	Unformatted Capacity (MB)	Cyl	Heads	Rotational Latency (ms)	Avg Access Time (ms)	Data XFR Rate Max.	Avg.
ST506	6.38	153	4	8.33	85	625	476
ST406	6.38	306	2	8.33		625	476
ST412	12.76	306	4	8.33	85	625	476
ST419	19.14	306	6	8.33		625	476

The FWD8006 controller supports all the Winchester drives listed in Table I-8 and c patibles. Table I-9 provides an easy reference for the 5-1/4" Winchester drive co patibility.

Table I-9. Winchester Drive Compatibility

Compatibility	Vender	Model	Cylinders	Heads	Unformatted Capacity (MB)
Fully ST506 Compatible (1)	Seagate	ST506	153	4	6.38
	TI	ST506	153	4	6.38
Fully ST406 Compatible (1)	Ampex (2)	PYXIS7	306	2	6.38
	CMI	CM5206	306	2	6.38
	IMI	5006	306	2	6.38
	Rodime(2)	R0201	306	2	6.38
	Seagate	ST406	306	2	6.38
	Tandon	TM501	306	2	6.38
Fully ST412 Compatible (1)	Ampex (2)	PYXIS13	306	4	12.76
	CMI	CM5412	306	4	12.76
	IMI	5012	306	4	12.76
	Rodime (2)	R0202	306	4	12.76
	Seagate	ST412	306	4	12.76
	Tandon	TM502	306	4	12.76
Fully ST419 Compatible (1)	Ampex (2)	PYXIS20	306	6	19.14
	CMI	CM5619	306	6	19.14
	IMI	5018	306	6	19.14
	Rodime (2)	R0203	306	6	19.14
	Seagate	ST419	306	6	19.14
	Tandon	TM503	306	6	19.14

Table I-9. Winchester Drive Compatibility (continued)

Compatibility	Vender	Model	Cylinders	Heads	Unformatted Capacity (MB)
ST406/412/419 Compatible Except Capacity (3)	Ampex	PYXIS27(4)	320	8	26.67
	Atasi	3033	645	5	33.59
		3046 (4)	645	7	47.03
	CMI	CM6426	640	4	26.00
		CM6640	640	6	40.00
	Evotek	ET5530(4)	375	6	23.00
		ET5540(4)	375	8	31.00
	Quantum	Q520	512	4	21.00
		Q540	512	8	42.00
	Maxtor	XT-1065(4)	918	7	66.94
		XT-1105(4)	918	11	105.19
		XT-1140(4)	918	15	143.44
	Rodime	R0204	320	8	26.67
		R0208 (4)	640	8	53.33
	Tandon	TM703 (4)	600	5	31.25
	Vertex	V130 (4)	987	3	31.00
		V150 (4)	987	5	51.67
		V170 (4)	987	7	72.33

1) All the fully Seagate compatible drives should have the drive straps W1-W4 defined as if they were Seagate drives. (See Section II-C)

2) These drives actually have 320 cylinders. However, the FWD8006 only uses the lower 306 cylinders.

3) All the drives which are Seagate compatible except the capacity have their own drive strap definition for straps W1-W4 as detailed in section II-C.

4) The released firmware supports these drives based on the specifications in the drive OEM manuals. However, due to drive unavailability, the firmware has not been fully tested with these drives.

II. PREPARATION FOR USE

The single board FWD8006 is a complete floppy/Winchester controller. In order to use the controller in a system, however, certain controller characteristics must be selected by the user to suit his particular system requirements and the controller must be cabled to the disk drives.

A. UNPACKING

Examine all shipping containers for evidence of damage prior to unpacking the equipment. If any damage is found, notify an authorized representative of the shipping concern before unpacking the equipment.

No special unpacking instructions are required since the shipping containers are of a standard type and easily disassembled. All sub-assemblies are adequately protected from normal shocks incurred during shipping.

Visually inspect each assembly in the shipment for damage. Check each item against the packing list to ascertain that all items have been received.

B. I/O ADDRESS and MEMORY WAKE-UP ADDRESS SELECTION

As described in section III-B, the programmed I/O interface between the FWD8006 and the host is via a single I/O Write address. In addition, a single Multibus memory address is used as the starting address of the sequence of memory based communication and control tables, as detailed in Section III-D. So that the controller will respond to the correct addresses as chosen by the user a series of jumpers on the FWD8006 PC board must be set. The user must first, however, determine what addresses are appropriate based on three considerations:

1. The addresses chosen must be compatible with the rest of the user's Multibus system. The programmed I/O address must be unique to the FWD8006 and the wake-up address must be the system address of memory suitable for storing the Wake-up Block.
2. Some jumpering options are determined by whether the Multibus system in which the controller is to be installed uses 8 bit or 16 bit I/O addresses. Typically, small to medium sized systems using 8 bit processor based CPU boards (such as the Intel iSBC 80/20-4, iSBC 80/24, or iSBC 80/30) use 8 bit I/O addresses while 16 bit I/O addressing is reserved for larger systems using 16 bit processors or larger.
3. The wake-up address, which is actually the Multibus memory address of the Wake-up Block (Section III-D-1), is always a multiple of either 16 or 256*, e.g. it is possible to select a wake-up address of 0042CH** but not possible to select an address of 00424H. Also the wake-up address and the I/O address are closely related so consideration must be given to choosing a combination which satisfies all of the various system requirements. When 16 bit I/O addressing is to be used, the I/O address is always the wake-up address divided by 16 or 256.

Thus a wake-up address of 00440H or 004400H corresponds to an I/O address of 0044H, a wake-up address of 92CD0H or 92CD00H corresponds to an I/O address of 92CDH, etc. When 8 bit I/O addressing is to be used, the I/O address is the least significant 8 bits of the wake-up address divided by 16 or 256. Thus for 8 bit I/O addressing, a wake-up address of 00440H or 004400H corresponds to an I/O address of 44H, and a wake-up address of 1B680H or 1B6800H corresponds to an I/O address of 68H.

* Note: For detailed information on memory address generation by the FWD8006 refer to Section III-A.

** Note: The H suffix is used throughout this manual to denote a number in hexadecimal (base 16) notation. Similarly, a B suffix is used for binary numbers. Numbers without suffixes are decimal.

Once appropriate addresses have been selected, the following two steps are used to set 8 bit or 16 bit addressing and the actual I/O address on the FWD8006. Figure II-1 may be used to locate the jumpers specified.

1. If the FWD8006 is to recognize an 8 bit I/O address, remove jumper W37 and switch the jumpers at W38 and W39 to connect A-B. If it is to recognise a 16 bit I/O address, install a jumper at W37 and switch the jumpers at W38 and W39 to connect B-C.

2. The actual I/O address bits are set by jumpers W21 thru W36 by following this procedure:

   Divide the wake-up address by 16 or 256 (Section III-A) and then converted to a 16 bit binary number, right justified.

   E.g.

   00440H / 16 = 44H = 0000000001000100B
   92CD00H / 256 = 92CDH = 1001001011001101B

   The 16 bits are copied to the table below, one bit per box.

   Jumpers are INSTALLED at those W locations over 1s in the t
   Jumpers are REMOVED from those W locations over 0s in the table.

W21	W22	W23	W24	W25	W26	W27	W28	W29	W30	W31	W32	W33	W34	W35	W36

The FWD8006 is currently shipped from the factory with:

Wake-up address = 00040H or 000400H
I/O address = 04H
8 bit I/O recognized

This strapping is subject to change without notice.

C. OPTION STRAPPING

In addition to the address selection described in the preceeding section, the FWD8006 PC board contains additional jumper locations which are used to configure the controller to the user's particular Multibus system. They are described below and may be located on the PC board by refering to Figure II-1. The configuration as supplied by SMS is shown in parentheses with each jumper function.

W1-W3 Define First Winchester Drive (W3-AB Installed)

Jumpers W1 through W3 are used to define the first Winchester drive according to the following table:

		W1-AB	W1-BC	W2-AB	W2-BC	W3-AB	W3-BC
	No Winchester	R	R	R	R	R	R
Seagate	ST506	R	R	R	R	R	I
	ST406	R	R	R	R	I	R
	ST412	R	R	R	I	R	R
	ST419	R	R	R	I	R	I
Rodime	R0208	R	R	R	I	I	R
Vertex	V130	R	R	I	R	R	R
	V150	R	R	I	R	R	I
	V170	R	R	I	R	I	R
Tandon	TM703	R	I	R	R	R	R
Atasi	3033	R	I	R	I	R	I
	3046	R	I	R	I	I	R
Maxtor	XT-1065	R	I	I	R	R	R
	XT-1105	R	I	I	R	R	I
	XT-1140	R	I	I	R	I	R
Rodime	R0204						
Ampex	PYXIS27	I	R	R	R	R	R
Evotek	ET5530	I	R	R	R	R	I
	ET5540	I	R	R	R	I	R

CMI	CM6426	I	R	R	I	R	R
	CM6640	I	R	R	I	R	I
Quantum	Q520	I	R	R	I	I	R
	Q540	I	R	I	R	R	R
CDC	9415-36	I	R	I	R	R	I

R = Removed
I = Installed

All other possible combinations are reserved.

For drive compatibility, refer to Table I-9.

W4 Define Second Winchester Drive (both AB, BC Removed)

Jumper W4 is used to define the second Winchester drive as follows:

W4-AB	W4-BC	
R	R	Same as the first one defined by W1-AB thru W3-BC
R	I	Reserved
I	R	Reserved

W5-W7 Define Floppy Drives (W7-AB Installed)

Jumpers W5 through W7 are used to define the two floppy drives according to the following table:

	W5-AB	W5-BC	W6-AB	W6-BC	W7-AB	W7-BC
No floppy	R	R	R	R	R	R
SA810	R	R	R	R	R	I
SA860 M2896-63 TM848-2E	R	R	R	R	I	R
TM848-1	R	R	R	I	R	R
TM848-2	R	R	R	I	R	I
SA850	R	R	R	I	I	R
SA400	R	R	I	R	R	R
SA450	R	R	I	R	R	I
SA410	R	R	I	R	I	R
SA460 FD-55F	R	I	R	R	R	R

MPI91	R	I	R	R	R	I
MPI92	R	I	R	R	I	R
TM100-3	R	I	R	I	R	R
TM100-4	R	I	R	I	R	I
SA455 FD-55B	R	I	R	I	I	R
SA800	R	I	I	R	R	R
SA465	R	I	I	R	R	I
OA-D33V	R	I	I	R	I	R

Note: The two floppy drives should be of the same type.

All other possible combinations are reserved.

W8 Enable Power On Reset Delay (Installed)

Installed: A 15 second power on reset delay is enabled before the controller can respond to any host command. During the 15 second delay, the on board LED will blink for approximately 7 times.

Removed: There will be no delay, i.e. as soon as power is turned on, the controller can respond to host command immediately.

W9 Enable Reset Self Test (Installed)

Installed: Upon detection of the second programmed I/O Start command after reset (Section III-B), a comprehensive set of internal controller diagnostics is executed. If an error is detected status is posted and an interrupt is generated.

Removed: No diagnostics are executed except via the Diagnostic Command.

W10 Enable 16 Bit System Bus (Installed)

Installed: The FWD8006 will assume that all memory to which it is directed supports word transfers which it will use when appropriate.

Removed: The FWD8006 will only do byte data transfers to the Multibus except for data buffers where word transfers are flagged by the appropriate Modifier bit (Section III-F-2) and then only if W11 is installed to enable the command extension.

W11-W12 Enable Command Extension and Direct/Buffered Transfers (W11 Installed)

W11, when installed, enables all the extended mode operations except direct/buffered transfers. In exterded mode, FWD8006 command forms which are not available with the Intel iSBC215/iSBX218 combinations are allowed. (For detail of extended mode operation see modifier byte of IOPB block in section III.D.4.)

If W11 is removed, the extended mode is disabled and only valid iSBC215/iSBX218 command forms will be accepted by the FWD8006. Note that in this iSBC215/iSBX218 compatible mode, the controller will post only operation complete status, not seek complete or media change status, to the host until the "Busy" byte in CCB is cleared, i.e., until the controller finishes executing a command issued by the host. This means that in the compatible mode, the report of seek complete or media change will be delayed until after the controller finishes a command. In the extended mode the controller will post status internally queued in sequence whenever the status semaphore is cleared by the host, regardless of whether it is seek complete, media change or operation complete status. Note that to be compatible with iRMX-86, W11 has to be removed and W43-AB must be installed.

Besides enabling and disabling the command extension, W11 is also used together with W12, to define four direct/buffered transfer modes. The definitions are listed below:

W11	W12	
R	R	Compatible mode, both floppy and Winchester transfers buffered
R	I	Compatible mode, floppy direct, Winchester buffered
I	R	Extended mode, floppy direct, Winchester programmable by modifier byte.
I	I	Extended mode, both floppy and Winchester transfers direct

This implies that bit 6 of the IOPB modifier byte (section III.D.4) is only used to define Winchester direct/buffered transfers when W11 is installed and W12 is removed.

W13 Enable No-Host Diagnostics (BC connected by a trace on board)

AB, BC both removed: Enables normal controller operation. Straps W8 through W12 have the control functions as described above.

AB installed: Enables stand-alone diagnostics. Straps W8 through W12 have control functions related to diagnostics and host control is overriden except for reset control. Refer to section V for details of the comprehensive diagnostic capabilities available.

BC installed: The No-Host diagnostic is enabled (low) or disabled (high) by pin 38 of Multibus P2 connector.

Note: The controller is shipped with a trace on the board connecting W13-BC. The trace has to be cut if pin 38 of P2 is to be used for other purposes in a specific system environment or if AB is to be installed.

!!! Warning !!!

If No-Host diagnostics is enabled (W13-BC installed) and format is enabled (W11 installed) and Winchester write protect is disabled (W14 removed), a format operation will automatically be started after power on and data on Winchester will be overwritten.

W14 Enable Winchester Write Protect (BC connected by a trace on board)

AB, BC both removed: Both Winchester drives are not write protected.

AB installed: Both Winchester drives are write protected.

BC installed: Winchester write protection is controlled by pin 20 of Multibus P2 connector. Write protected if the line is driven low, not protected otherwise.

Note: The controller is shipped with a trace on the board connecting W14-BC. The trace has to be cut if pin 20 of P2 is to be used for other purposes in a specific system environment or if AB is to be installed.

W15 Enable Write Precompensation (BC connected by a trace on board)

AB, BC both removed: Not recommended if the trace between posts B and C is cut; may enable write precomp.

AB installed: Disable write precomp.

BC installed: Enable write precomp.

Note 1): The controller is shipped with a trace on the board connecting W15-BC. The trace has to be cut if AB is to be installed.
2): Even if write precomp is enabled by strap W15, the controller will override it if the Winchester drives connected to the controller are manufactured by Vertex or Maxtor because both manufacturers recommend that precomp not be enabled for their drives.

W18-W19 Select PLL timing for 8" or 5-1/4" floppies (W18-AB, W19-AB installed)

W18-AB, W19-AB installed: Select PLL integrator time constant for 8" floppy.

W18-BC, W19-BC installed: Select PLL integrator time constant for 5-1/4" floppy.

Note: All other possible combinations are illegal.

W21-W39 are described in section II-B.

W41 Disable 8289 ANYRQST (Installed)

Installed: ANYRQST of the 8289 bus arbiter is disabled, that is lower priority master has to acquire Multibus via CBRQ/line.

Removed: ANYRQST is enabled, that is lower priority master will be treated as higher priority master.

W42 Enable BPRO/ (Installed)

Installed: Multibus serial priority output signal is connected to P1-16. Required for serial priority systems.

Removed: BPRO/ is disconnected from P1-16 to allow BPRN/ of the next lower priority board to be driven by external circuitry. Required for non-SMS backplanes using parallel priority.

W43 Select CBRQ/ (BC Installed)

W43-AB installed: FWD8006 Multibus arbitration circuitry is controlled by the Multibus CBRQ/ signal, P1-29. Under these conditions, the FWD8006 will release the bus to a lower priority bus master unless CBRQ/ is asserted. compatible with iRMX-86, W43-AB must be installed.

W43-BC installed: FWD8006 Multibus arbitration circuitry behaves as if CBRQ/ is always asserted. Bus arbitration normally occurs for every Multibus access.

W44 Select Interrupt Level (A to D wirewrapped, INT5/)

This wirewrap jumper is used to select the particular Multibus interrupt line to be asserted whenever the FWD8006 generates an interrupt. A connection is wirewrapped between W44-A and one of the other pins as follows:

Multibus	W44
INT7/	B
INT6/	C
INT5/	D
INT4/	E
INT3/	F
INT2/	H
INT1/	J
INT0/	K

W45 Enable LED Driver (Connected by a trace on board)

Installed: The LED driver which drives the on-board LED is also connected to drive P2-28 line.

Removed: Disconnect the on-board LED driver from P2-28.

Note: The controller is shipped with a trace on the board connecting W45-AB. The trace has to be cut if P2-28 is to be used for other purpose in a specific environment.

W46-W49 IEEE 796 High Order Address Lines (Connected by traces on board)

These four straps connect the FWD8006 to the four most significant IEEE 796 address lines.

W47 ADR17/ P2-56
W46 ADR16/ P2-55
W49 ADR15/ P2-58
W48 ADR14/ P2-57

Note: The controller has these four traces on board. These traces have to be cut if these four pins are to be used for other purpose.

W50 Enable LOCK/ (Removed)

Installed: The LOCK/ signal generated by the controller is connected to P1-25.

Removed: Controller generated LOCK/ signal is disconnected from P1-25.

Note 1): This strap exists only on boards of revision level C and up.
2): The Multibus specifications specify that "LOCK/" cannot be asserted longer than 12 us on the Multibus. However, when the controller asserts "LOCK/" for Winchester direct transfer, it stays asserted during the whole block transfer. Hence it will violate the Multibus timing specification if this strap is installed and Winchester direct transfer is requested. It is the user's responsibility to make sure that the installation of W50 will not cause any problem in his system.

The remainder of the straps on the pc board are used for SMS manufacturing test and are not user options. They must be strapped as follows:

Straps Installed: W20, W40 (W20 is connected by a trace on board. No strap has to be installed unless the trace is cut)

Straps Removed: W16, W17

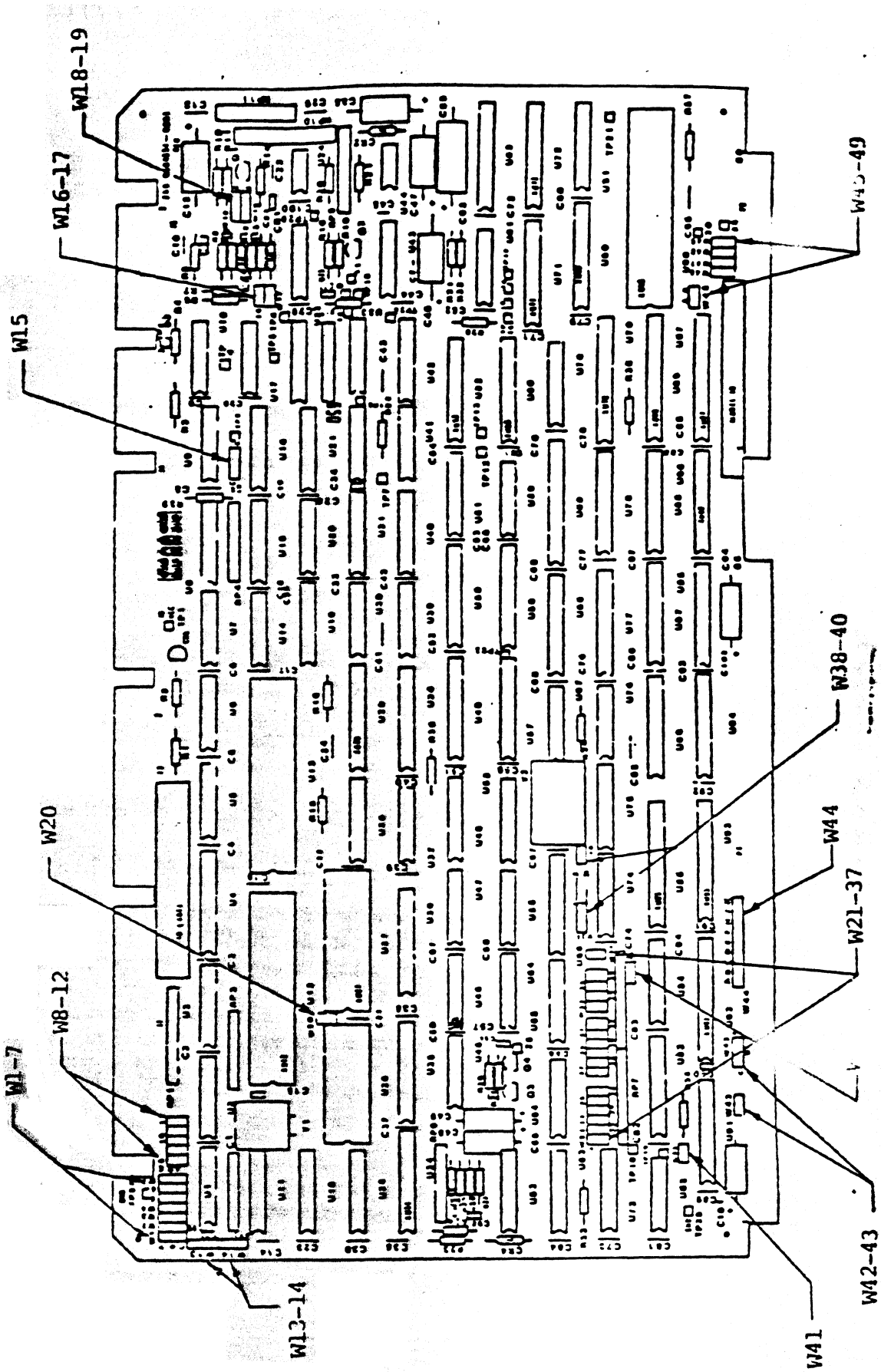


Figure II-1. FWD8006 PWB Jumper Locations

D. DRIVE STRAPPING

All the **last (second)** drives on the J1,J2 or J3 cable should be terminated properly **as recommended** by the drive manufacturer. Tables II-5 and II-6 provide detailed information on option straps for each drive supported by the FWD8006.

Table II-5. Floppy Drive Options

SHUGART SA400/SA450/SA410/SA460

DRIVE	JUMPERS	
DRIVE 0	Remove MX, MS, SD	Install DS1, SS, DD, RI, DA
DRIVE 1	Remove MX, MS, SD	Install DS2, SS, DD, RI, DA

Heads will load when door is closed. Door locks when drive is selected.

MICRO PERIPHERALS INC. MPI 91/92

DRIVE	JUMPERS	
DRIVE 0	Remove T1, T3, T4, T5, T6	Install T7
DRIVE 1	Remove T1, T2, T4, T5, T6	Install T7

Head loads when motor comes on.

TANDON TM100-3/TM100-4

DRIVE	JUMPERS
DRIVE 0	Remove NDS1, NDS2, NDS3, MX
DRIVE 1	Remove NDS0, NDS2, NDS3, MX

TANDON TM848-1/TM848-2

The default strapping as shipped by Tandon must be modified as follows:

DRIVE	JUMPERS	
DRIVE 0	Remove B, M1, M2, M3	Install A,X,C,I,R,HL,Z,M4,DC,DS1
DRIVE 1	Remove B, M1, M2, M3	Install A,X,C,I,R,HL,Z,M4,DC,DS2

The motor on and stepper power circuits are controlled by Head Load (HL).

SHUGART SA800

DRIVE	JUMPERS
DRIVE 0	Install A,X,Y,C,HL,T1,T2,800,L,DS1
DRIVE 1	Install A,X,Y,C,HL,T1,T2,800,L,DS2

SHUGART SA455/SA465

DRIVE	JUMPERS
DRIVE 0	Install MM,HS,HLD,M3,SO,DS,DS1
DRIVE 1	Install MM,HS,HLD,M3,SO,DS,DS2

SONY CA-D30V

DRIVE	JUMPERS
DRIVE 0	Drive switch set to position 1.
	Solder jumper on back of Sony drive PC board, 34 pin connector pin 4 to pin 10. This is required in order to allow a write to the drive. Non-standard minifloppy requirement.
DRIVE 1	Drive switch set to position 2.
	Solder jumper on back of Sony drive PC board, 34 pin connector pin 4 to pin 12. This is required in order to allow a write to the drive.

SHUGART SA810/SA860

DRIVE	JUMPERS	
DRIVE 0	Remove all jumpers	Install MO,SI,S2,TR,2S,DS1,Y,SR,DC
DRIVE 1	Remove all jumpers	Install MO,SI,S2,TR,2S,DS2,Y,SR,DC

Y,SR available on drive PCB P/N 35249 and up.

SHUGART SA850

DRIVE	JUMPERS
DRIVE 0	Install A,M,X,Y,R,I,C-HI,S2,2S,DS1,850,HL,IT,IWI
DRIVE 1	Install A,M,X,Y,R,I,C-HI,S2,2S,DS2,850,HL,IT,IWI

DD,RR, and RI etched on PC board and are assumed to be connected. B,S, and Z are connected via straps at location 4F on the drive and should be cut. Either RS or RM should be jumpered. On SA851 drives, either FS or TS should be jumpered. SMS jumpers RM and FS. No other jumpers are to be implemented.

TANDON TM848-2E

The following jumpers must be installed:

DS1 (or DS2 for a second drive), HL, M4, TR

The following are installed with a trace on the PC board. The jumpers do not have to be installed unless the trace has been cut.

S2, PS, R, DC, 2S, XC

MITSUBISHI M2896-63

DRIVE	JUMPERS
DRIVE 0	DS1,JSG,SE,RFa,HUN,HY,PS,RM,2S,I,R,S2,RS,Z,WP,A,X,MO,C,DC,M2
DRIVE 1	DS2,JSG,SE,RFa,HUN,HY,PS,RM,2S,I,R,S2,RS,Z,WP,A,X,MO,C,DC,M2

Table II-6. Winchester Drive Options

SEAGATE ST506

DRIVE	JUMPERS	
DRIVE C	Remove shunts 16-1, 13-4, 12-5, 11-6, 10-7.	Install 15-2, 9-8.
DRIVE 1	Remove shunts 16-1, 13-4, 12-5, 11-6, 9-8.	Install 15-2, 10-7.

The Half Step option is selected.

SEAGATE ST400 SERIES

DRIVE	JUMPERS	
DRIVE 0	Remove shunts 16-1, 12-5, 11-6, 10-7.	Install 15-2, 13-4, 9-8.
DRIVE 1	Remove shunts 16-1, 12-5, 11-6, 9-8.	Install 15-2, 13-4, 10-7.

CMI CM5000 SERIES

DRIVE	JUMPERS
DRIVE 0	Install shunt 1-2.
DRIVE 1	Install shunt 3-4.

Note: Except for strapping differences, the CM5000 series drives are fully compatible with the ST400 series.

E. DRIVE CABLE REQUIREMENTS

The FWD8006 is connected to the disk drives via five flat ribbon cables: the 50-pin (J1) is for operation of up to two daisy chained 8" floppy disk drives; one 34-pin (J2) is for up to two 5-1/4" floppies; the other 34-pin (J3) is for two 5-1/4" Winchesters. The last drive at the end of these three cables has to be properly terminated as recommended by the drive manufacturers. Note that the controller cannot support 8" and 5-1/4" floppies simultaneously. For each Winchester drive there is an additional 20-pin cable to carry the drive data and control signals. The cable for the first Winchester (W0) is connected to J4 and for the second (W1) to J5. The maximum length of each cable is 15 feet including cabling within the Multibus enclosure or drive enclosures. The proper board edge connectors for the 50-pin cables are Scotchflex ribbon connectors p/n 3415-0001 or equivalent, p/n 3463-0001 or equivalent for the 34-pin cables, and p/n 3461-0001 or equivalent for the 20-pin cables.

The following tables show the pin assignments in each cable.

Table II-1. 8" FLOPPY DISK DRIVE CABLE (50-PIN, J1)

Signal Name	Driver	Pin	Signal Function
Low Current	FWD8006	2	Reduced write current
-- not used --		4	
-- not used --		6	
True Ready	Floppy	8	Drive is ready to handle data
Two Sided	Floppy	10	The diskette loaded is two sided
Disk Change	Floppy	12	Disk change has been detected
Side Select	FWD8006	14	Select one of the two sides of the diskette
-- not used --		16	
Motor On	FWD8006	18	Turn on the DC motor
Index	Floppy	20	Beginning of physical disk track
Ready	Floppy	22	Drive is ready for access
-- not used --	Floppy	24	(sector)
Drive Select 0	FWD8006	26	Enable first floppy for control
Drive Select 1	FWD8006	28	Enable second floppy for control
-- not used --		30	
-- not used --		32	
Direction	FWD8006	34	Move heads toward center if true (low)
Step	FWD8006	36	Pulsed for a single cylinder step
Write Data	FWD8006	38	Serial write data to floppy
Write Gate	FWD8006	40	Enables write current
Track 00	Floppy	42	R/W heads at outermost cylinder
Write Protect	Floppy	44	Cannot write on current drive
Read Data	Floppy	46	Serial read data from floppy
-- not used --	Floppy	48	(Separated data)
-- not used --	Floppy	50	(separated clock)

Note: All odd pins are grounded.

Table II-2. 5-1/4" FLOPPY DISK DRIVE CABLE (34-PIN, J2)

Signal Name	Driver	Pin	Signal Function
— not used —		2	
-- not used --		4	
-- not used —		6	
Index	Floppy	8	Beginning of physical disk track
Drive Select 0	FWD8006	10	Enable first floppy for control
Drive Select 1	FWD8006	12	Enable second floppy for control
— not used —		14	
Motor On	FWD8006	16	Turn on the DC motor
Direction	FWD8006	18	Move heads toward center if true (low)
Step	FWD8006	20	Pulsed for single cylinder step
Write Data	FWD8006	22	Serial write data to floppy
Write Gate	FWD8006	24	Enable write current
Track 00	Floppy	26	R/W heads at outermost cylinder
Write Protect	Floppy	28	Cannot write on current drive
Read Data	Floppy	30	Serial read data from floppy
Side Select	FWD8006	32	Select one of the two sides of the diskette
— not used —		34	(drive status)

Note: All odd pins are grounded

For Sony 3-1/2" floppy disk drive, a special scramble cable has to be made to route all signals from J1 to the appropriate pins on the drive.

Table II-3. 5-1/4" WINCHESTER DISK DRIVE CABLE (34-PIN, J3)

Signal Name	Driver	Pin	Signal Function
Low Current	FWD8006	2	Reduced write current/Head Sel 3
Head Select 2	FWD8006	4	Most significant head select control
Write Gate	FWD8006	6	Enable write current
Seek Complete	Winchester	8	Signals the end of carriage movement
Track 00	Winchester	10	R/W heads at the outermost cylinder
Write Fault	Winchester	12	Drive may cause improper writing on disk
Head Select 0	FWD8006	14	Least significant head select control
-- not used —		16	
Head Select 1	FWD8006	18	Middle head select control
Index	Winchester	20	Beginning of physical disk track
Ready	Winchester	22	Drive ready for access
Step	FWD8006	24	Pulsed for a single cylinder step
Drive Select 0	FWD8006	26	Enable first Winchester for control
Drive Select 1	FWD8006	28	Enable second Winchester for control
-- not used --		30	
-- not used —		32	
Direction	FWD8006	34	Move heads toward center if true (low)

Note: All odd pins are grounded.

TABLE II-4. WINCHESTER DIFFERENTIAL DATA CABLE (20-PIN, J4/J5)

Pin No.		Pin No.	
1		11	GND
2	GND	12	GND
3	RECAL/	13	WDATA
4	GND	14	WDATA/
5	CARWP/	15	GND
6	GND	16	GND
7		17	RDATA
8	GND	18	RDATA/
9		19	GND
10		20	GND

DATA TRANSFER CONTROL

In addition to the level of control provided by the variety of commands which may be issued to the FMD8006, many of these functions may be further tailored to the needs of a particular system by controlling certain data transfer parameters. These controls are implemented via bits in the Modifier byte in the IOPB. The definitions of each of the bits in this byte are repeated below.

Bit	Function enabled when bit is set (=1)
7	Enables extensions to iSBC 215 commands. Must be set, and M11 installed, to enable bits 6 or 5.
6	Enables direct Winchester disk/Multibus transfers rather than fully buffered transfers.
5	Enables word data transfer within data buffer even if M10 is removed specifying byte transfers.
4	Undefined - Must be set to zero
3	Undefined - Must be set to zero
2	Enables floppy Deleted Data AM writing/reading
1	Inhibits all error retries
0	Suppresses command completion interrupt

The function of bits 7, 1, 0 are trivial as described above. But bits 6, 5, and 2 need further explanation.

Bit 6 - Direct Winchester Disk/ Multibus Transfer

In systems where higher performance is required, and where sufficient Multibus bandwidth is available, direct disk/memory transfers may be used with the Read Data and Write Data commands. When this option is selected by setting bits 7 and 6 in the modifier, data is transferred in exactly the same way as for buffered transfers except that the on board buffer is not used for any full sectors moved and the Multibus is locked during the transfer. Data is passed directly between the disk and the Multibus memory a byte or word at a time. Because the data is transferred as the disk is written or read, no time need be allocated at the end of each sector for data movement between Multibus memory and the controller. This is especially useful when reading or writing disk which were formatted without interleave; data is moved at full disk speed instead of one disk revolution per sector. Note, however, that transferring data directly between the disk and memory requires that a Multibus memory access be done whenever a byte or word of data has been passed from/to the disk. It is possible that occasionally the controller will either not get access to the Multibus quickly enough to store the data collected before it is overrun by the next byte from the disk when reading, or to fetch the next byte or word before it is required by the disk when writing. Should this occur, the FMD8006 will recover automatically. Multibus memory pointers and counters will be internally reset and the disk sector involved will be accessed again. On this second transfer the data in this single sector will be buffered through the internal FMD8006 buffer. This allows full recovery from data overrun/data late conditions without CPU intervention. Note that because this recovery requires two full disk revolutions, systems which require that the FMD8006 be frequently

III. PROGRAMMING THE FWD8006

The FWD8006 controller presents an interface to the host computer driver program which is simple and straightforward. This interface is composed of a single I/O address where the host goes to control the overall execution of commands, a facility by which the controller may generate host interrupts, and a mechanism by which the host specifies, via blocks of data stored in system memory, which disk operations to perform. Each of the three areas are discussed in sections which follow. Preceeding these discussions, is a section describing in detail the mechanism used by the FWD8006 to generate Multibus memory addresses.

A. MULTIBUS ADDRESS GENERATION

The FWD8006 is capable of operating in Multibus systems based on 16 bit, 20 bit, 24 bit memory addressing as a component of systems ranging from low cost [illegible] based on 8 bit microprocessors to high performance products with 16 or 32 bit processors and containing up to 16 Megabytes of Multibus memory.

Regardless of actual memory size or system addressing capability, the FWD8006 always calculates memory addresses which are 24 bits in length internally. This 24 bit address is driven onto the Multibus address lines during memory read or write operations. In those systems where 16 or 20 bit addressing is used, the most significant 4 address bits (ADR14*, ADR15*, ADR16*, and ADR17*) may be disconnected from the Multibus P2 connector by cutting 4 traces on W46-49 (Section II-C). In those systems using 16 bit addressing, system software will cause the FWD8006 to only generate addresses where the most significant address bits are zero and these may be ignored by other boards on the Multibus.

The 24 bit addresses generated by the FWD8006 are not supplied by the host directly, but are constructed by the controller from pairs of addressing words. The mechanism used is a form of memory block indexing called segmented addressing. Segmented addressing is used to allow the specification of any Multibus address as a truncated representation of the address of a block of memory, called a Segment, and a relative address within that block, called the Offset. Using this scheme has many advantages. First of all, it is identical to the memory addressing scheme used by the Intel iAPX86 (8086) and iAPX88 (8088). Secondly, it allows the addresses of individual buffers, variables, etc. in any one block of memory to be treated as if they had 16 bit addresses even though the Multibus system supports 24 bit addressing. This also means that data areas in several different blocks which are all at the same relative locations to the beginning of their blocks may be addressed by only changing the Segment.

All memory addresses generated by the FWD8006 are produced in the same way regardless of whether the addressed memory contains control information, command parameters, user data, etc. In addition, the initial address used by the controller to access the host communication blocks, the wake-up address, is computed by treating the 16 bit value strapped on the PWB (W21-W36) as a segment for an address with an Offset of zero.

Segmented addresses consist of the two 16 bit numbers mentioned above, the Segment and the Offset. In all cases where segmented addresses are used by the FWD8006 these two numbers are stored in memory in the same format (Figure III-1).

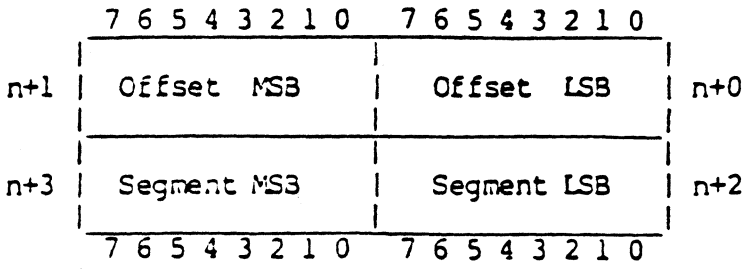


Figure III-1. Segmented Address Format

To arrive at the 24 bit Multibus address which corresponds to a particular segmented address the FWD8006 takes the 16 bit Segment and multiplies it by either 16 or 256 (i.e. shifts it left 4 or 8 bits) and then adds the Offset to the result. Thus the Segment is the truncated base address of a block (segment) of memory which starts on a 16 or 256 byte boundary and the Offset is simply an index from that base address. For example:

```
Segment = 1234H
Offset  = 5678H

   Segment * 16  =  12340H
   plus Offset     + 5678H

Multibus address =  17B38H
```

If the Segment is multiplied by 16 (shifted 4 bits) a 20 bit address will generally result. This mode is used whenever the segmented address is to be converted in the same way as implemented in the iSBC 215 controller, the iAPX86 microprocessor etc. The FWD8006 will use this mode for all address conversions following a programmed I/O Start command where the data written to the controller was 01H.

To use the full 24 bit, IEEE 796, addressing capability of the FWD8006, a programmed I/O Start command is written to the controller with a data value of 03H. For addresses converted after a Start of this type, the Segment value is multiplied by 256 (shifted 8 bits) before the Offset is added. This allows the specification and generation of addresses anywhere in the 16 Megabyte range.

The two different conversion systems also apply to the first programmed I/O Start command following controller reset. Thus if the value written is 01H, the hardware strapped value, which is treated as a Segment, is shifted left four bits to arrive at the wake-up address. On the other hand, if the I/O data written is 03H, the strapped value is shifted 8 bits to produce the wake-up address. E.g. a programmed I/O address of 0070H when set into the address straps on the controller board is interpreted to mean that the Wake-up Block is located at memory address 00700H if a 01H is written to the controller's I/O address after a Reset/Clear sequence. But the same physical strapping is converted to a value of 007000H for the address of the Wake-up Block if the start value written is 03H.

Note:

1. Because the Segment is stored as a 16 bit number, the Multibus address of the start of the block must be a multiple of 16 or 256. That is; 23450H is a possible 20 bit address for the start of a block, 23456H is not because the 6 will be lost when the address is converted to the Segment, 2345H.

2. Any particular Multibus address may be represented in a great many different ways. E.g. 44444H may be a Segment of 4440H and an Offset of 0044H or a Segment of 4000H and an Offset of 4444H.

3. The two conversion methods listed above may be intermixed. Thus the first start issued may use 20 bit addressing to specify the Wake-up address at any 16 byte boundary in the low 1 Megabyte of address space and then subsequent Starts for disk I/O may use 24 bit addressing to allow access anywhere in memory.

B. PROGRAMMED I/O

Because control of the FMD8006 is heavily focused on the interface presented through Multibus memory based tables, the programmed I/O interface to the controller has been kept to the minimum required for overall control functions. The FMD8006 responds to a single I/O address which is selected by hardware straps on the controller board. This address may be either an 8 bit I/O address as used in older Multibus compatible systems or may be a full 16 bit address to be compatible with the latest Multibus specification. In either case, the FMD8006 will respond only to I/O Write operations to this address. Whenever such an I/O Write is detected, the least significant two bits of the data sent from the host are used to select one of four possible hardware functions:

Data	Function
00H	Clear Interrupt - Remove Reset
01H	Start Operation - Use 20 bit addressing
02H	Reset Controller
03H	Start Operation - Use 24 bit addressing

These functions are described in more detail on the next page.

CLEAR (00H): An I/O Write operation with this data value causes any pending hardware interrupt to be removed. This is the normal procedure used by a host interrupt handler in response to a controller interrupt. This I/O Write operation is also used to clear the controller's hardware reset circuitry. This must be done after any programmed I/O Reset, Multibus INIT/ assertion, or power-on reset before the FWD8006 will begin normal operation.

START (01H): An I/O Write to the FWD8006 with this data is treated two different ways. The first Start command after the controller has been reset is used to signal the fact that the controller should go to Multibus memory and collect the addresses of important memory based control tables. This operation returns no status information except that the BUSY flag is cleared. After this initial case, any further Start commands cause the controller to fetch the I/O Parameter Block and to begin executing the command specified. Refer to Section III-E for detail on commands and the IOPB contents.

RESET (02H): Addressing the FWD8006 with an I/O Write operation with this data value will cause the controller to be reset immediately. Current disk operations are terminated, buffer transfers in progress will stop, and no status will be returned.

NOTE: After the hardware reset is removed by a Clear programmed I/O function, the controller will execute start-up diagnostics as specified by option strapping and drive initialization commands will be required before any further disk accesses will be performed.

START (03H): An I/O Write to the FWD8006 with this data is treated in the same way as if the data had been 01H, with a single exception. If the data is 01H, Segment values (included the I/O address straps) are multiplied by 16 before the Offset is added. If the data is 03H, Segment values are multiplied by 256 before the Offset is added.

C. INTERRUPTS

As with any other high performance disk controller, the FMD8006 is capable of alerting the host to significant changes in disk system status by generating hardware interrupts. This is done by asserting one of the 8 possible Multibus interrupt lines, INT7/ - INT0/. The particular interrupt line to be used is user selectable by changing a wire wrapped jumper on the controller board. Once the interrupt is asserted, it is removed either by a Clear or Reset programmed I/O Write from the host to the controller. The interrupt is also cleared by power-on reset or by the assertion of the Multibus INIT/ signal.

There are three events which will cause the FMD3006 to assert an interrupt:

1. Completion of a command
2. Completion of a seek
3. Floppy media change

The first of these, the command completion interrupt, may be disabled by controlling the appropriate bit in the Modifier word in the command IOPB (Section III-D-4). The remaining two interrupt sources may not be disabled.

For a detailed discussion of interrupt processing with the FMD8006 refer to section III-G.

3. MEMORY BASED CONTROL PATHS

The main command and status path between the FWD8006 controller and the host is a set of tables stored in Multibus memory. These tables are used to pass all the command information required for each disk access such as disk function, disk address, buffer address, etc. as well as all status information returned by the controller. The use of this mechanism allows far more detail and flexibility in the interface between host and controller than approaches which rely on passing command and status information through hardware registers with hardware protocols.

The FWD8006 uses four memory based tables for all communication between the host and the controller. These four tables are:

WUB - Wake-up Block

CCB - Channel Control Block

CIB - Controller Invocation Block

IOPB - Input/Output Parameter Block

These four blocks are chained together via address pointers contained in the blocks themselves. A fifth block associated with most controller operations is the Multibus memory buffer where data is either read to be put on the disk, written from the disk, or used to control such functions as disk formatting. This fifth block is also linked into the chain of control blocks.

1. WUB - Wake-up Block

The Wake-Up Block is the first block in the chain and is used only to link the controller to the rest of the chain. It consists of 6 bytes (Figure III-2). The first two bytes are reserved. The remaining four bytes contain the segmented address (Section III-A) of the next block in the chain, the CCB. The address of the WUB is defined by the same hardware straps on the controller board which define the programmed I/O address used for Clear, Start, and Reset programmed I/O Write commands from the host. The value represented in these straps (Section II-G) is multiplied by either 16 or 256 (Section III-A), i.e. shifted 4 or 8 bit positions to the left, to get the Multibus address of the WUB. On recognition of the first programmed I/O Start command from the host after a hardware reset of any type the FWB2006 will go to this address, fetch the WUB, and internally save the CCB address. Since this action is only taken once after a reset, the WUB need not be preserved.

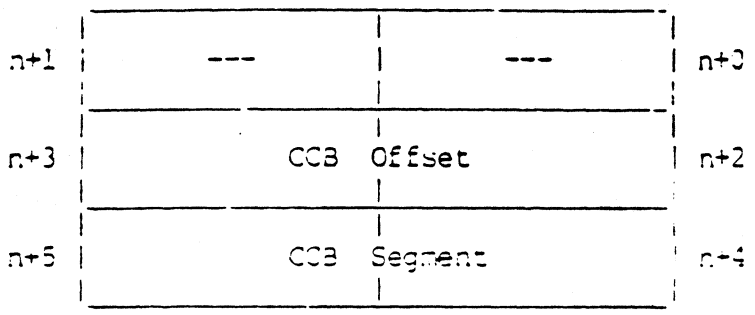


Figure III-2. Wake-up Block

2. CCB - Channel Control Block

The next block in the control chain is the Channel Control Block, or CCB. It is 6 bytes in length as shown in Figure III-3. The first byte is checked by the FWD8006 for compatibility with the INTEL iSBC 215. If it is not 01H processing will cease with no status reported. The second byte is the BUSY flag. It is used to inform the host whether the controller is busy (0FFH) or idle (00H). This information is used in handshaking commands and status between the host and the controller (Section III-G). The remaining four bytes of the CCB contain the segmented address of the fifth byte in the next table in the control chain, the CIB.

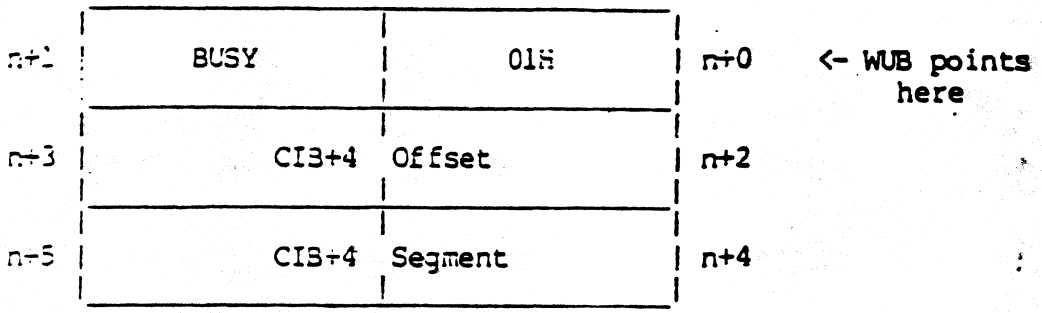


Figure III-3. Channel Control Block

Note:

The actual 24 bit address of the CCB is stored within the controller during the processing of the first programmed I/O Start command after a reset. Its location must not be changed between commands without a controller reset and initialization sequence.

3. CIB - Controller Invocation Block

The Controller Invocation Block, or CIB, is the next block in the control chain. It is 12 bytes long as shown in Figure III-4.

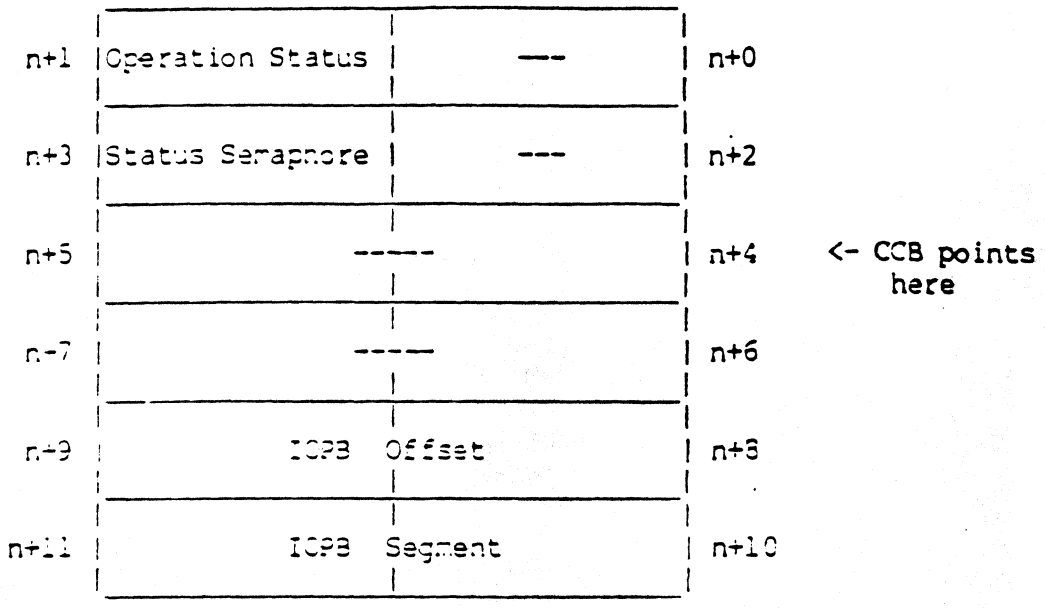


Figure III-4. Channel Invocation Block

The fields in the CIB have the following functions:

n+0 reserved

n+1 Operation Status:

This byte contains the latest controller status and is interlocked by byte n+3 below. The status contained in this byte is encoded on a bit by bit basis. The bits having the following meaning:

7	6	5	4	3	2	1	0
E	H	-	U	-	D	N	S	O

E = Error Detected
H = Hard Error
U = Unit (2 bits)
D = Drive Type
N = New Ready Drive (Media change)
S = Seek Complete
O = Operation Complete

The E bit is set whenever an error is detected. It remains set even if a retry is successful. The H bit indicates hard error. It is set if error is detected and no retry is allowed or if the retries are not successful. In other words, E=H=0 means successful operation without any error. E=1, H=0 means error detected but recovered. E=1, H=1 means error detected with unsuccessful retries or without retry. E=0, H=1 should never occur.

Note: If an Operation Status byte is read in which E=H=1 and N=S=O=0 the controller is indicating that its internal status buffer has overflowed because of lack of host service. Some status bytes have been lost.

n+2 reserved

n+3 Status Semaphore - See Section III-G for details

n+4 - n+7 reserved

n+8 - n+11 Segmented address of IOPB

4. IOPB - Input / Output Parameter Block

The main channel for communication between the FWD8006 and the host is the next block in the control chain, the Input / Output Parameter Block, or IOPB. Stored in this block is all the information required by the controller for each specific command such as disk address, buffer address, and function to be performed. The IOPB is 26 bytes in length as is diagrammed in Figure III-5. A description of each of the fields follows the figure.

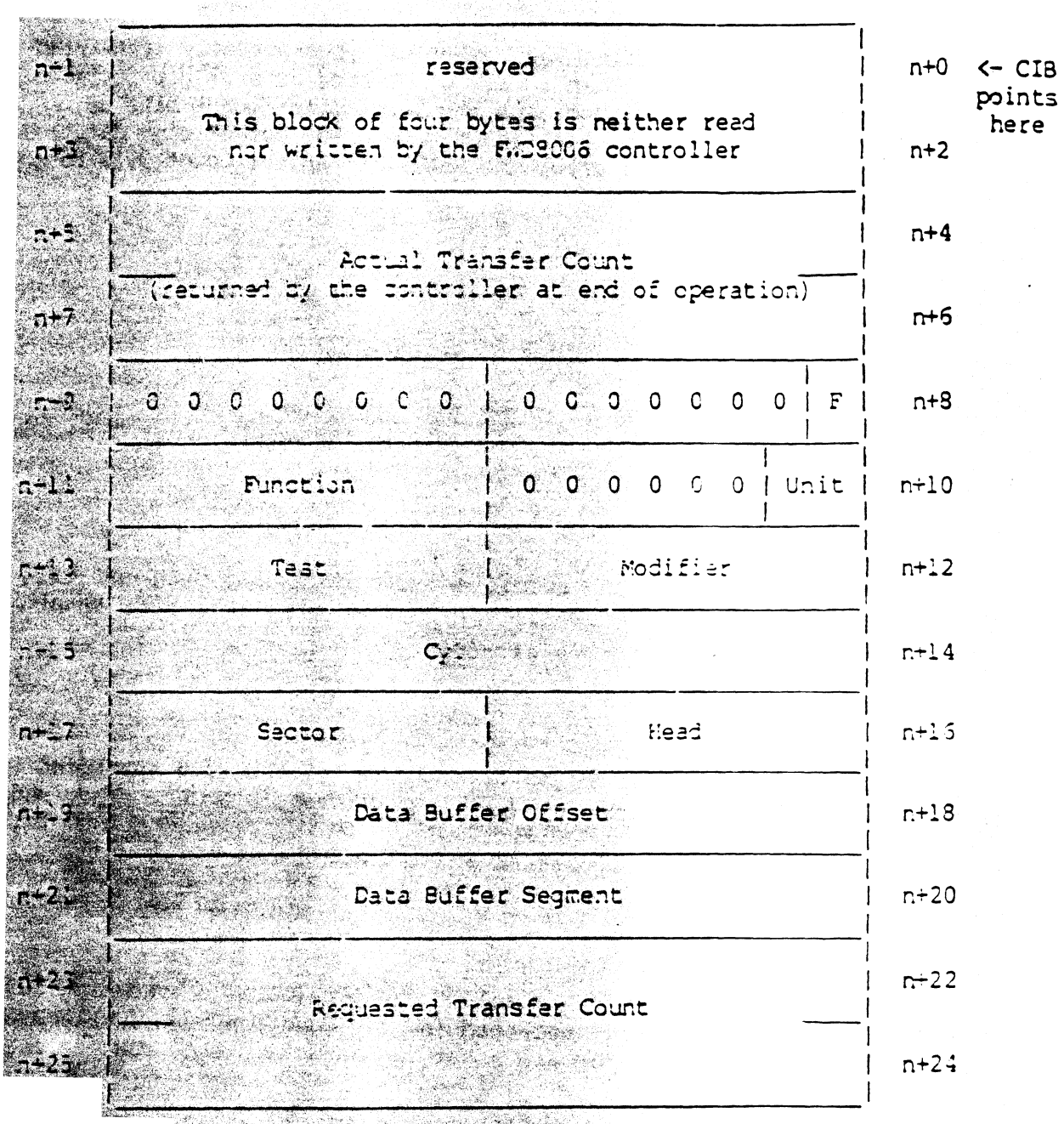


Figure III-5. Typical I/O Parameter Block

The individual fields in the FWD8006 IOPB are defined below. Note that fields filled with zeros in Figure III-5 and in the figures for the various commands are reserved for future expansion and should always be set to zero by the host driver to insure continued compatibilty. Those fields filled with dashes are ignored by the commands shown and may assume any value.

Actual Transfer Count:

This four byte field specifies the number of bytes actually transfered in the process of executing the command. It is treated as a 32 bit positive binary number. Byte 4 of the IOPB contains the least significant bits of and byte 7 contains the most significant bits. For normal disk data which are completed without error the Actual Transfer Count will be equal to the Requested Transfer Count.

F:

This bit specifies the device type to be accessed.

0 - Winchester disk (iSBC 215 equivalent)
1 - Floppy diskette (iSBX 218 equivalent)

Function:

A single byte field which specifies the operation; such as READ, FORMAT, etc. which is to be performed. The value in this byte determines how some other fields in the IOPB are interpreted.

Unit:

This field specifies which of the disk drives of the type specified in the F bit is to be accessed. Since at most two drives of each type are supported, the normal values for this field are 0 or 1. On commands which set this field to 2 or 3 the controller will return a DRIVE NOT READY error.

Test:

This byte is only used to specify the particular diagnostic test to be executed when the operation specified in the Function byte is the Diagnostic Command (Function = 0FH). For all other commands this byte must be zero.

Modifier:

This byte is actually treated by the controller as a field of 8 single bit control flags. The bits are independent of one another and are assigned the following functions:

Bit	Function enabled when bit is set (=1)
7	Enables extensions to iSBC 215 commands
6	Enables direct Multibus <-> Winchester disk transfers
5	Enables word data transfers within data buffer
4	Undefined - Must be set to zero
3	Undefined - Must be set to zero
2	Causes floppy read/write to use deleted data AMs
1	Inhibits all error retries
0	Suppresses command completion interrupt

For more detailed explanation of the modifier byte, refer to Section III-F.

Cylinder:

This word is used to specify the starting cylinder (track) number from which any disk write, read, or format will begin. The range of values acceptable for this word are dependent on the drive type and drive parameters specified at initialization time. The smallest cylinder number is always 0, which is the cylinder farthest from the center of the physical disk. An illegal value in this field will cause the heads on the specified drive to be retracted to cylinder zero (recalibrated) and an error will be returned.

Sector:

This byte is much like the Cylinder word in its function. It specifies the starting sector number for disk write or read operations. The range of legal values is dependent on drive type and format (i.e. the number of sectors per track). Note that the smallest sector number is always 1 for floppy disks and always 0 for Winchester disks.

Head:

This byte is also much like the Cylinder word in its function. It specifies the starting head number for disk write or read operations. The range of legal values is dependent on drive type. Like cylinder numbers, head numbers start at 0. Note that for single sided floppy diskettes the head number is always 0 and that for double sided floppy diskettes the only values allowed are 0 and 1.

E. FMD8006 COMMANDS

The FMD8006 presents the programmer with a comprehensive repertoire of commands designed to take full advantage of the capabilities of the disk drives supported. These include not only normal READ and WRITE commands but also commands designed to tailor the disk system's operation to different applications and a complete set of diagnostic capabilities.
The various commands are invoked by initializing the previously described tables properly and then executing a programmed I/O Start to the controller. The memory based tables leading up to the IOPB have been described in previous sections. The following sections detail the IOPB and data buffer requirements for each of the commands executed by the FMD8006.

The particular command to be executed is determined by the Function field in the IOPB. The 16 possible values and the commands associated with these values are:

Value (Hex)	Function
0	INITIALIZE
1	TRANSFER STATUS
2	FORMAT
3	READ SECTOR ID
4	READ DATA
5	READ TO BUFFER AND VERIFY
6	WRITE DATA
7	WRITE BUFFER DATA
8	INITIATE TRACK SEEK
9	reserved
A	reserved
B	reserved
C	reserved
D	reserved
E	BUFFER
F	DIAGNOSTIC

The following descriptions detail these functions in the order given above. The description of each function includes a diagram of the IOPB showing those fields which must be properly set by the host before the command is executed. Those fields which are not examined by the controller for the function specified, and which thus may have any value, are shown in these diagrams with dashes inserted in place of the field name.

Data Buffer Address:

This four byte field contains the segmented address of the data buffer. For normal read or write operations this is the address of the Multibus memory buffer where the data is to be stored or fetched. For some commands, this is the address of additional control information.

Requested Transfer Count:

This four byte field is set by the host to specify the number of bytes which are to be transfered in the process of executing the command. This field has the same format as the Actual Transfer Count and it is also treated as a 32 bit positive binary number.

1. Initialize - Set Disk Drive Parameters

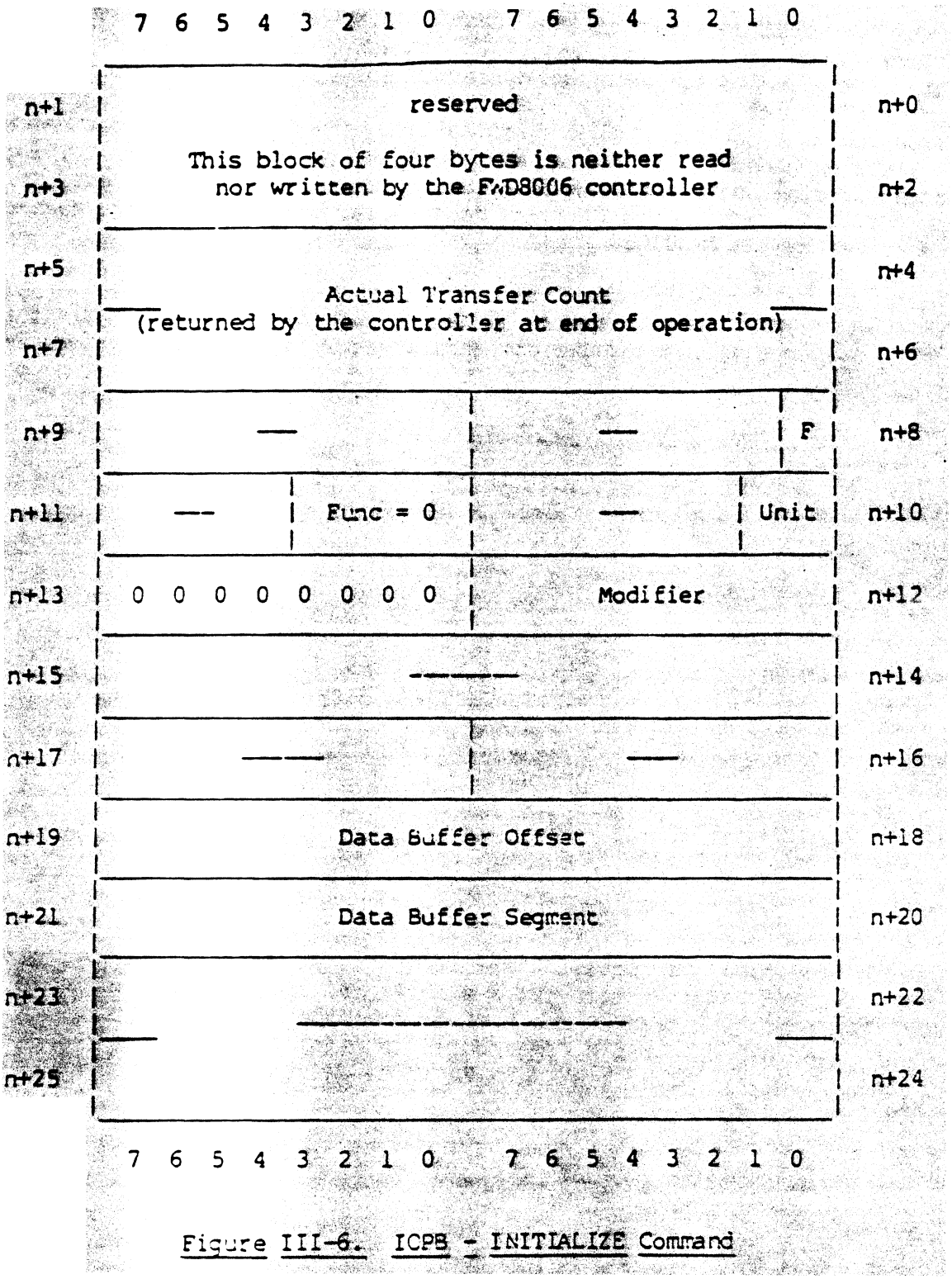


Figure III-6. ICPB - INITIALIZE Command

INITIALIZE

The Initialize command is used to transfer drive related parameters to the controller and to seek the drive heads to cylinder zero to synchronize positioning. The parameters include such things as the number of cylinders, heads, bytes per sector, etc. The information to be passed to the controller is contained in an extension to the IOPB. This extension is eight bytes in length (Figure III-7) and is addressed by the Data Buffer Offset and Segment stored in the IOPB.

n+1	Total Number of Cylinders	n+0
n+3	Sides - floppy / Heads - Winch	n+2
n+5	Bytes/Sector LSB / ----	n+4
n+7	Encoding / Bytes/Sector MSB	n+6

Figure III-7. IOPB Extension - Initialize Command

Because the information contained in the IOPB extension is used by the controller for all disk related commands, the Initialize Command should be issued for each drive in the system following any hardware reset whether caused by power on, Multibus INIT, or a programmed I/O Reset command (Section III-8). Commands issued for drives which have not been initialized will not be executed and an error will be returned.

The fields in the IOPB extension have the following meanings:

Total Number of Cylinders:

This word is used to specify the total number of physical cylinders available on a disk drive. For the proper value of this word for the different Winchester drive types supported, refer to tables I-6 and I-7.

All the 8" floppy drives supported by the controller have 77 cylinders, but 5 1/4" floppy drives may have 35, 40 or 80 cylinders. To provide flexible media exchangeability when the 5 1/4" floppy drives are used, the controller allows a diskette formatted by a 48 TPI drive to be read by a 96 TPI drive. The

process is as follows: The controller recognizes the 96 TPI drive by straps W4-W7. If the total number of cylinders specified in this word is 35 or 40, the controller will issue double stepping pulses for each cylinder. If the total number of cylinders specified is any value other than 35 or 40, no double stepping will be provided. Also note that the double stepping is allowed only for read operation but not write operation because a diskette written by a 96 TPI drive may not create a strong enough signal when read by a 48 TPI drive.

Heads - Winch:

This byte specifies the number of read/write heads on a Winchester drive. The proper value for this byte for each of the drives supported is also contained in Table I-6. When a floppy drive is specified, this byte is ignored.

Sides - floppy:

This byte specifies whether the diskette to be used is single sided if the byte is one, or double sided if the byte is two. This is needed because the 5 1/4" floppy drives do not generate a "double sided" signal and the controller has no way of knowing whether the diskette is single sided or double sided without this byte. The 8" floppy drives, however, do not need this byte because the "double sided" signal tells the controller directly whether the inserted diskette is double sided or single sided. This implies that if there is a media change after a 5 1/4" floppy drive is initialized, the controller does not know whether the diskette is changed from single to double sided or vice-versa unless the drive is re-initialized properly. This problem does not exist for the 8" floppy drives.

Bytes/Sector LSB:
Bytes/Sector MSB:

These two bytes form a word specifying the number of data bytes in a disk sector. This sector length must match the format of the disk and must be either 128, 256, 512, or 1024.

Encoding:

If the drive specified is a floppy, this byte is used to specify the recording density and data encoding scheme used on diskettes in this drive. If drive specified is a Winchester, this byte is ignored. Possible values a

Value	Density	Encoding
[illegible]	Single	FM - 5 1/4" IBM compatible, 5 1/4" iRMX 80 Compatible or 3-1/2" Sony compatible
[illegible]	Double	MFM - 5 1/4" IBM compatible, 5 1/4" iRMX 80, IBM PC Compatible or 3-1/2" Sony compatible
[illegible]	Double	MFM - 5 1/4" Shugart 800 compatible
[illegible]	Double	MFM - 5 1/4" QUME QUIC 10 Compatible
[illegible]	Double	MFM - 5 1/4" DEC RX50 and SMS FWD501C5 compatible
[illegible]	Double	MFM - 5 1/4" SMS FWD5074 Compatible
[illegible]	Double	MFM - 5 1/4" IBM PC Compatible, (9 sectors/trk, 512 bytes/sect)

The Sony 3-1/2" drive (OA-D33V) can use any of the 5-1/4" formats listed.

Note that if this byte is set to 22H, the only acceptable value for bytes/sector is 128.

2. Transfer Status - Read Extended Status Buffer

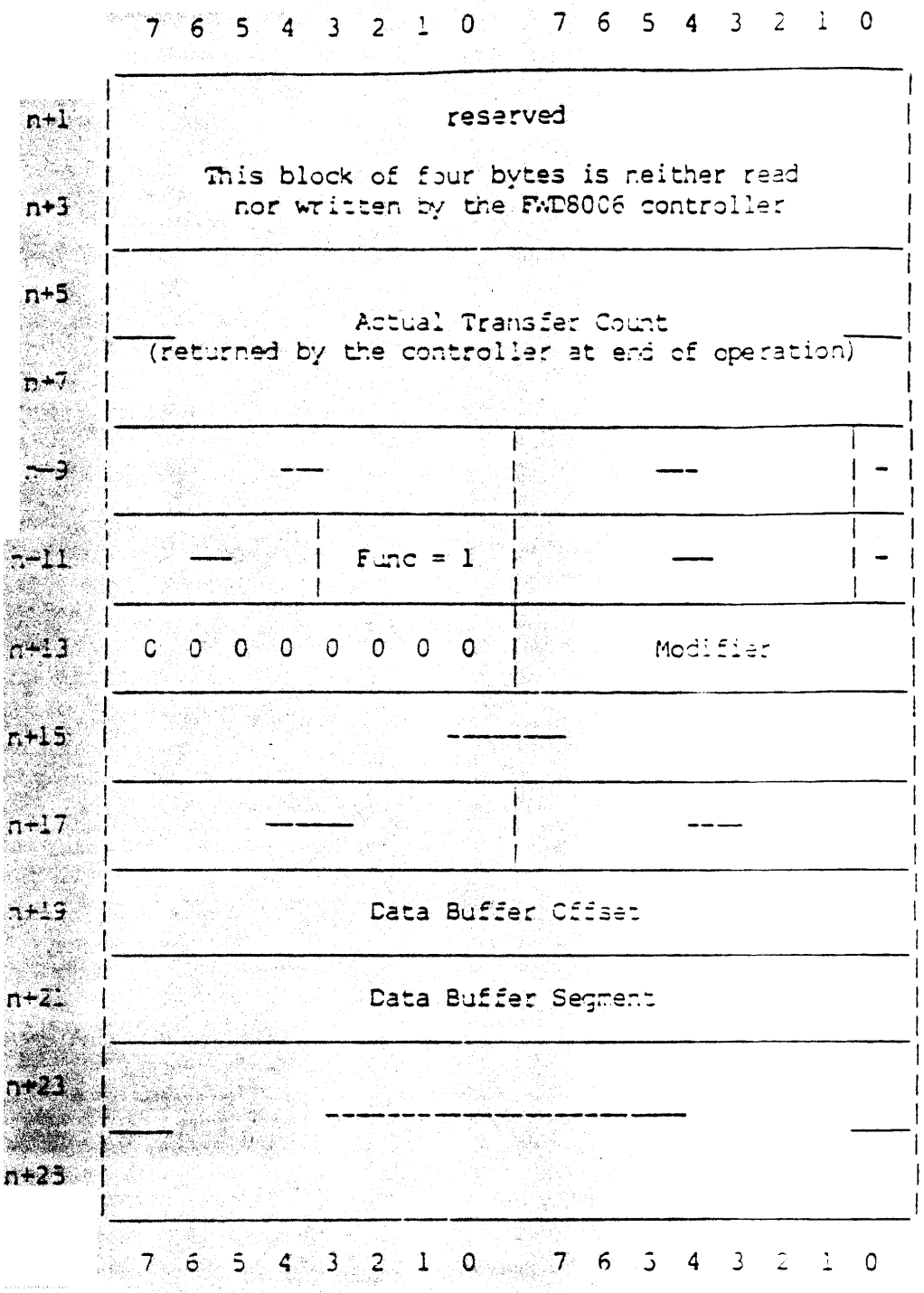


Figure III-8. ICPB - Transfer Status Command

TRANSFER STATUS

The Transfer Status Command is used to read the contents of the Extended Status Buffer from the controller's internal memory. It is normally used to gain detailed information concerning an operation error which was reported via the Error Detected bit in the Operation Status byte in the CIB. Executing this command will cause the latest contents of the buffer to be read to the data buffer specified in the Data Buffer Offset and Data Buffer Segment. The contents of the Extended Status Buffer are updated during the execution of all commands except Transfer Status. Therefore, if extended error information is required, the Transfer Status command should be issued immediately following the erroneous command execution.

The data returned consists of 12 bytes if the extension bit (bit 7) is not set in the Modifier in the IOPB and 13 bytes otherwise. The bytes contain the following information:

Byte	Bit	Error indicated when bit is set (=1)
1	7	End of media was detected before the requested transfer count was exhausted.
	6	Attempt to format a track as both an alternate and a defective track or a defective track pointed to a track which was not formatted as an alternate track.
	5	Attempt to issue a command for a drive which is currently executing a seek.
	4	ROM error (Section V)
	3	RAM / Controller error (Section V)
	2-0	reserved
2	7	Attempt to write or format a write protected floppy diskette.
	6	Drive not ready, not connected, or not initialized
	5	Invalid disk address (cylinder, head, or sector)
	4	Desired sector not found on specified track
	3	Invalid command specification
	2	No index pulse detected on selected drive
	1	Diagnostic fault (Section V)
	0	Sector size read is different than specified

2	7	reserved
	6	Cylinder address mismatch
	5	drive fault
	4	reserved
	3	Data field ECC/CRC error (If error could not be recovered, H bit is set in Operation Status in CIB)
	2-0	reserved
3-4	-	Word containing desired cylinder
5	-	Desired head
6	-	Desired sector
7-8	-	Word containing actual cylinder and flags

Bits 7-4 of byte 8 are flag bits as described below.
Bits 7-6 provide code of the track type:

00B Normal track
01B Alternate track
10B Defective track
11B Reserved

Bits 5-4 define the sector sizes:

00B 128 bytes/sector
01B 256 bytes/sector
10B 512 bytes/sector
11B 1024 bytes/sector

9	-	Actual head
10	-	Actual sector
11	-	Number of retries attempted / required
12	-	Detailed event code (Appendix A)

Note: Byte 12 is only returned if the extension bit (bit 7) is set in the Modifier in the ICPB for the Transfer Status command.

3. Format - Write Sector Data to a Disk Track -

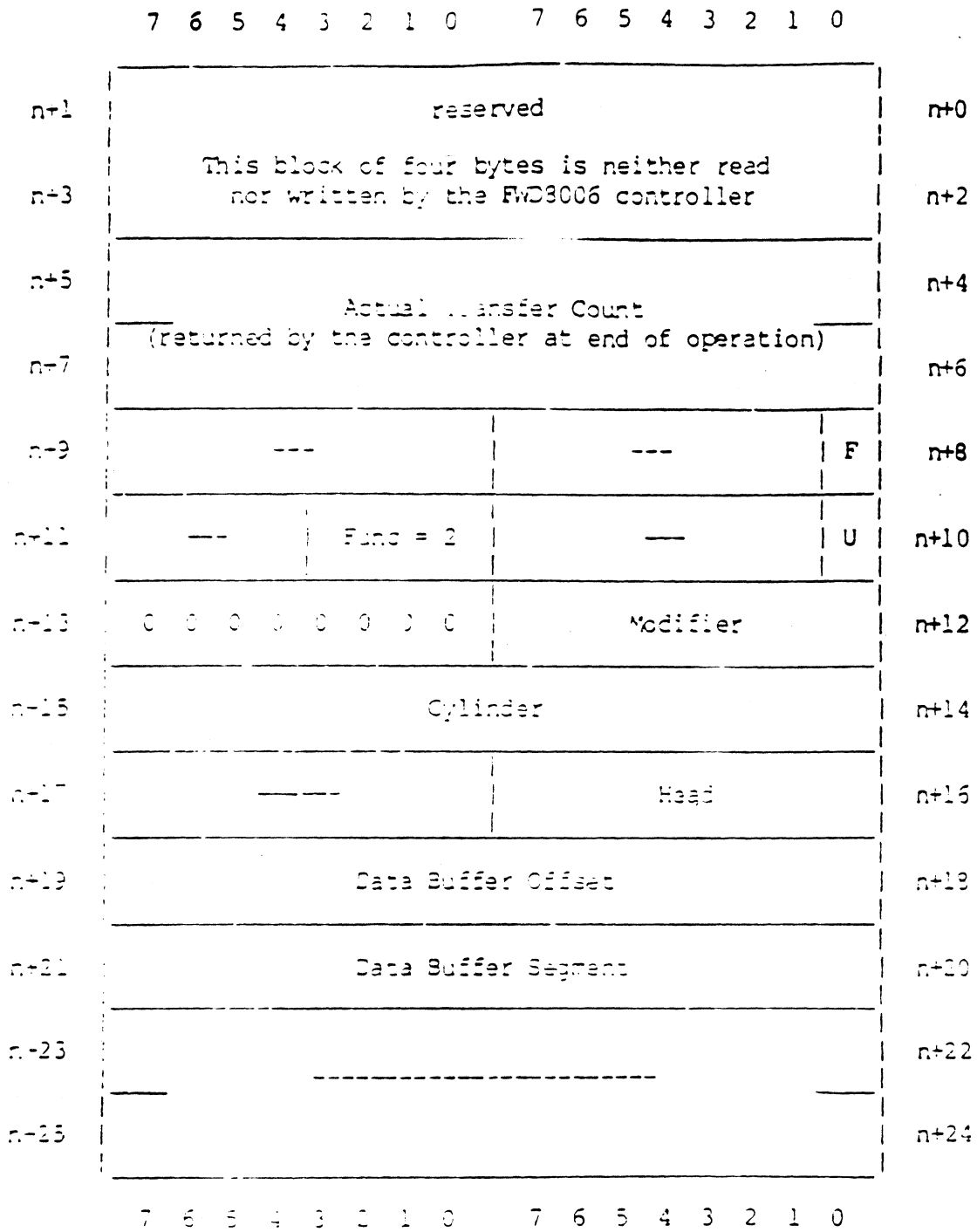


Figure III-9. 1088 - Format Command

FORMAT

The Format command is used to write the sector header information onto a single track of the specified disk drive. These sector ID fields segment the track, allocating space for each of the data sectors. They also contain information used in subsequent write or read operations to locate the correct sector data area and to verify that the correct cylinder and head have been reached and that the sector data area allocated matches the sector length to be written or read. In addition to the sector ID fields, default data fields are also written. The length of these data fields is the length specified in the last Initialize command for this dri This length also controls sector spacing and the number of sectors on the track. For further details on the actual information recorded during the formatting process refer to Section IV. Information required by the controller in order to format the track is passed by the host in an extension to the IOPB. The extension is six bytes long and is addressed by the data buffer offset and segment words in the IOPB. There are three versions of this extension which is shown in Figure III-10 below:

n+1	User Pattern 1	Track Type	n+0
n+3	User Pattern 3	User Pattern 2	n+2
n+5	Interleave Factor	User Pattern 4	n+4

Figure III-10. IOPB Extension - Format Command

The three extension types correspond to the three types of track formatting which may be specified for a Winchester disk. These are a normal data track, an assigned alternate track, or a defective track. The assignment of track type is part of media flaw management, covered in Section III-I, and is controlled by the ... type contained in the first byte of the extension. This byte is assigned the following values:

Value	Track Type
00H	Normal Data
40H	Assigned Alternate
80H	Defective

When formatting a floppy track this byte must be zero.

The user pattern is a four byte sequence which is written into the sector data areas on the track. For floppy diskettes the first byte (User Pattern 1) is repeated throughout the sector, i.e. 128 times, 256 times, etc. For Winchester tracks the whole sequence is repeated; i.e. 32 times for 128 byte sectors, 64 times for 256 byte sectors, etc. The pattern is written to the Winchester in the order it appears in the extension: User Pattern 1 is written first, User Pattern 4 is written last.
When the track is to be formatted as a Defective Winchester track the User Pattern bytes written are used on subsequent accesses to locate the assigned alternate track where the data is written. In this case they are interpreted as shown below.

User Pattern 1	Alternate Cylinder - LSB
User Pattern 2	Alternate Cylinder - MSB
User Pattern 3	Alternate Head
User Pattern 4	Reserved - must be 00H

The interleave factor is used to control the order in which the sectors appear on the physical track. An interleave factor of 1 specifies that sectors are to be written in sequence around the track; i.e. index, sector 1, sector 2, etc. Other values may be used to increase the disk rotational time between sequential sector numbers so that the data from each sector may be processed before the next sector on the track comes under the disk drive read/write head. For example, since Winchester disk data transfers are normally buffered thru the controller's on board buffer, at least one sector time should be allowed (more in some systems) between sequential sector numbers to allow the buffer contents to be passed between the controller and the Multibus memory. The interleave factor is the minimum number of sector intervals between the start of one sector and the start of the next sequential sector. Sector 1 for floppy and sector 0 for Winchester is always written immediately after the physical track index. Some examples of tracks written with interleave are diagrammed below:

Assuming 8 sectors/track (e.g. 1024 byte floppy single density):

Factor	Order from index
1	1 2 3 4 5 6 7 8
2	1 5 2 6 3 7 4 8
3	1 4 7 2 5 8 3 6
4	1 3 5 7 2 4 6 8

4. Read ID - Read Next Sector ID Field

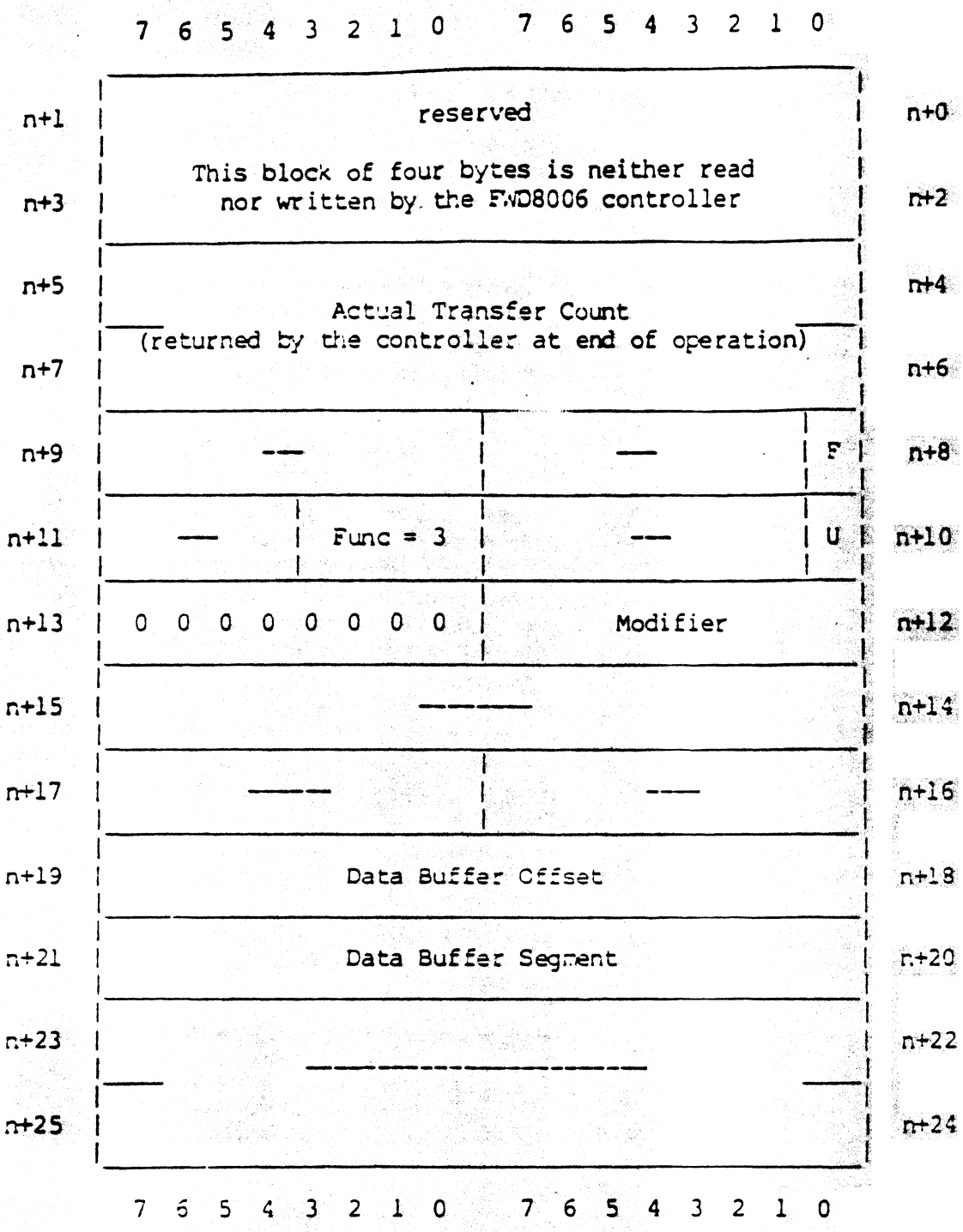
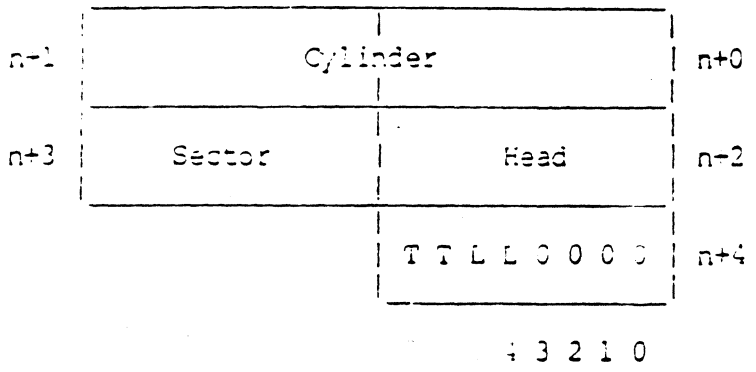


Figure III-11. ICPB - Read Sector ID Command

READ SECTOR ID

The Read Sector ID command is used to transfer the contents of the next available sector ID field (Section IV) into the data buffer. This information may then be used for a number of purposes such as verification of cylinder and head selection, sector length determination, rotational access optimization, etc. Since this command may be used to verify disk position, no implied seek or head selection is performed, and the sector ID is read from the disk track which was last referenced on the drive. The data buffer to be used is addressed by the data buffer offset and segment in the same way as the buffer for a read or write command. The data written to this buffer is five bytes in length and is diagrammed in Figure III-12 below:



Where:

TT	Track Type	LL	Length of Sector
00	Normal	00	128 bytes
01	Assigned Alt	01	256
10	Defective	10	512
11	Not used	11	1024

Figure III-12. Data Buffer - Read Sector ID Command

5. Read Data - Read Disk Data Into Memory

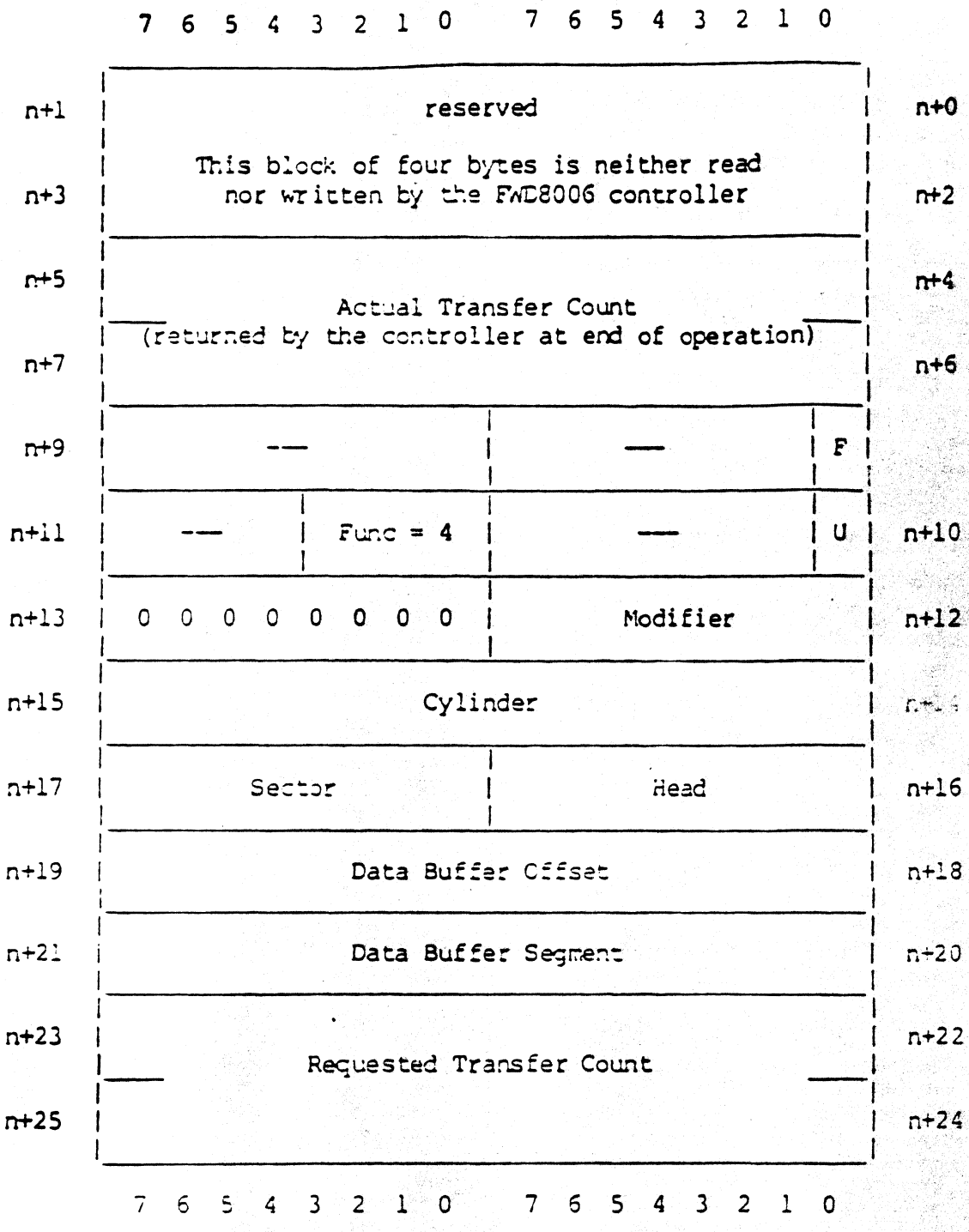


Figure III-13. IOPB - Read Data Command

READ DATA

The Read Data command is, with Write Data, one of the two most often used commands. Its function is to transfer data from a disk drive into a Multibus memory buffer. The IOPB fields F and U specify the drive to be accessed and the Cylinder, Head, and Sector fields determine the starting location on that disk. Data is transfered, starting from the first byte of that sector, into Multibus memory starting at the byte addressed by the Data Buffer Offset and Segment. Subsequent bytes are transfered to sequential bytes in memory until the number of bytes transfered is equal to the Requested Transfer Count field in the IOPB or until an error occurs. At this point the Actual Transfer Count field in the IOPB is updated with the number of bytes actually written to the memory buffer and status is returned. If buffered transfers have been selected (defined by W11, W12 and the modifier byte), each sector is read from the disk into the controller's on board buffer and then transfered to Multibus memory.

If the Requested Transfer Count has not been exhausted by the time the last sector on a track has been transfered, the FMD8006 will automatically continue the reading of data from the first sector on the next track of the cylinder by switching head selection in sequence. In addition, if the count has not been exhausted by the time the last sector on the last track of the cylinder has been transfered, the FMD8006 will automatically seek the drive to the next sequential cylinder and begin reading at head 0 and the first sector. If the Requested Transfer Count does not specify an integral number of sectors, the last sector containing part of the data will be read into the on board buffer in full, but only enough data will be moved to the Multibus buffer to exhaust the count.

Several bits in the Modifier field in the IOPB may be set to tailor the Read Data command. These are listed in the table below and described in detail in Section III-F.

Bit	Function enabled when bit is set (=1)
7	Enables extensions to iSBC 215 commands. Must be set to enable bits 6 or 5.
6	* Enables direct Winchester disk to Multibus transfers rather than fully buffered transfers.
5	Enables word data transfer within data buffer even if W10 is removed specifying byte transfers.
4	Undefined - Must be set to zero
3	Undefined - Must be set to zero
2	Causes floppy read to use deleted data AMs
1	Inhibits all error retries
0	Suppresses command completion interrupt

* Note that although the controller can do direct Winchester to Multibus data transfer by locking the Multibus, data overrun may occur if the Multibus data path is byte wide, or if the Multibus starting address is odd (the Multibus protocol requires that a single byte transfer be followed by a string of word transfers if the data path is word wide but the starting address is odd) or if the Multibus memory is a dual port memory which has to be shared by some other non-Multibus masters like memory on a CPU board because the controller can only lock the Multibus arbitration but has no control over the dual port arbitration which does not reside on Multibus.

6. Read Buffer - Read Disk Data Into Controller

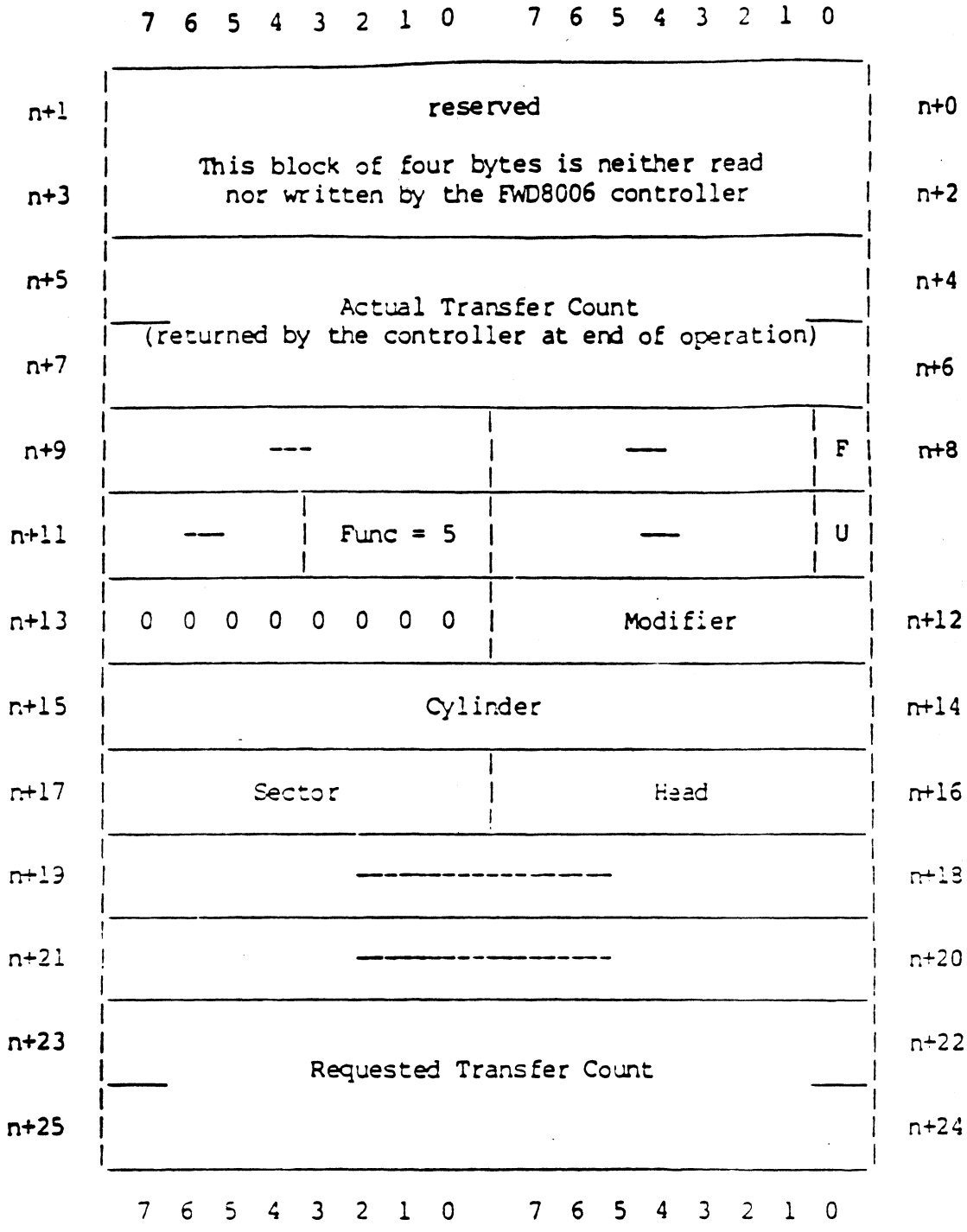


Figure III-14. IOPB - Read to Buffer and Verify Command

READ TO BUFFER AND VERIFY

The Read to Buffer and Verify command is identical to the Read Data command except that no data is transfered to the Multibus buffer. Typically this command is used to verify that certain sectors on a disk may be read with proper ECC/CRC checks or to fill the controller's on board buffer for access by subsequent commands such as Write Buffer Data or Buffer I/O. If the Requested Transfer Count specified in the IOPB is more than the number of bytes in the first sector specified, sequential sectors will be read into the buffer with each sector overlaying the one just previously read. If the Requested Transfer Count is not a multiple of the number of bytes in a sector the count will be rounded up; i.e., full sectors are always read into the controller buffer.

For details concerning IOPB parameters required for use of this command refer to the description of the buffered Read Data command.

7. Write Data - Write Memory Data onto Disk

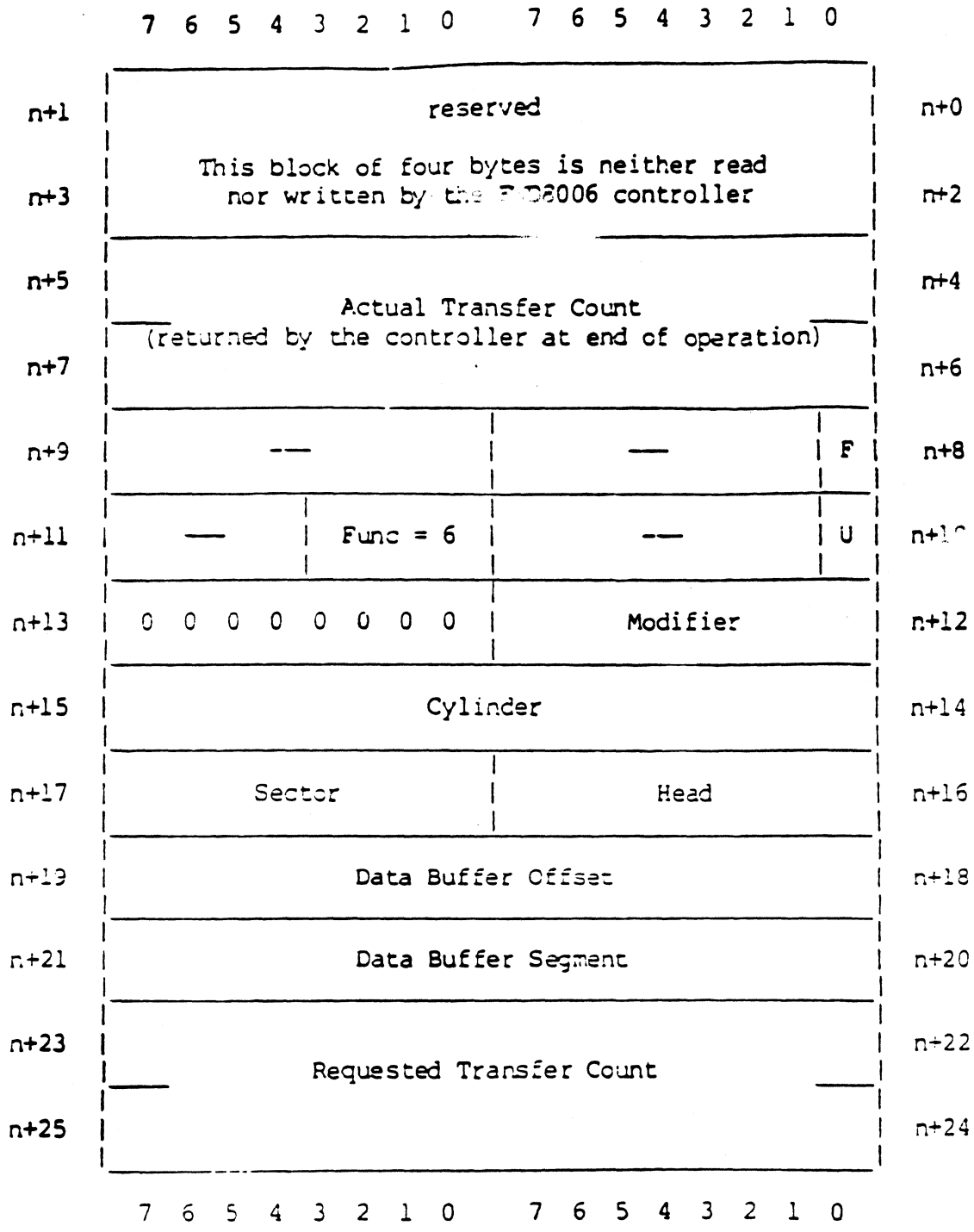


Figure III-15. IOPB - Write Data Command

WRITE DATA

The Write Data command is, with Read Data, one of the two most often used commands. Its function is to transfer data from a Multibus memory buffer to a disk drive. The IOPB fields F and U specify the drive to be accessed and the Cylinder, Head, and Sector fields determine the starting location on that disk. Data is transfered, starting from the first byte of the Multibus buffer addressed by the Data Buffer Segment and Offset to that starting disk sector. Subsequent bytes are transfered from sequential bytes in the buffer until the number of bytes transfered is equal to the Requested Transfer Count field in the IOPB or until an error occurs. At this point the Actual Transfer Count field in the IOPB is updated with the number of bytes actually read from the Multibus, and status is returned. If buffered transfers have been selected (defined by W11, W12 and the modifier byte), each sector full of data is transfered from the Multibus buffer into the controller's on board buffer first, and then this data is written onto the disk.

If the Requested Transfer Count has not been exhausted by the time the last sector on a track has been transfered, the FMD8006 will automatically continue the writing of data to the first sector on the next track of the cylinder by switching head selection in sequence. In addition, if the count has not been exhausted by the time the last sector on the last track of the cylinder has been transfered, the FMD8006 will automatically seek the drive to the next sequential cylinder and begin writing at head 0 and the first sector. If the Requested Transfer Count does not specify an integral number of sectors, the last sector written will contain the last partial sector's data and the balance of the sector will be filled with zeros.

Several bits in the Modifier field in the IOPB may be set to tailor the Write Data command. These are listed in the table below and described in detail in Section III-F.

Bit	Function enabled when bit is set (=1)
7	Enables extensions to iSBC 215 commands. Must be set to enable bits 6 or 5.
6	* Enables direct Multibus to Winchester disk transfers rather than fully buffered transfers.
5	Enables word data transfer within data buffer even if W10 is removed specifying byte transfers.
4	Undefined - Must be set to zero
3	Undefined - Must be set to zero
2	Causes floppy write to use deleted data AMs
1	Inhibits all error retries
0	Suppresses command completion interrupt

* The same restrictions as those mentioned in Read Data Command apply to direct Multibus to Winchester transfer. If any of these three conditions exist, a data late error may occur.

8. Write Buffer - Write Controller Data onto Disk

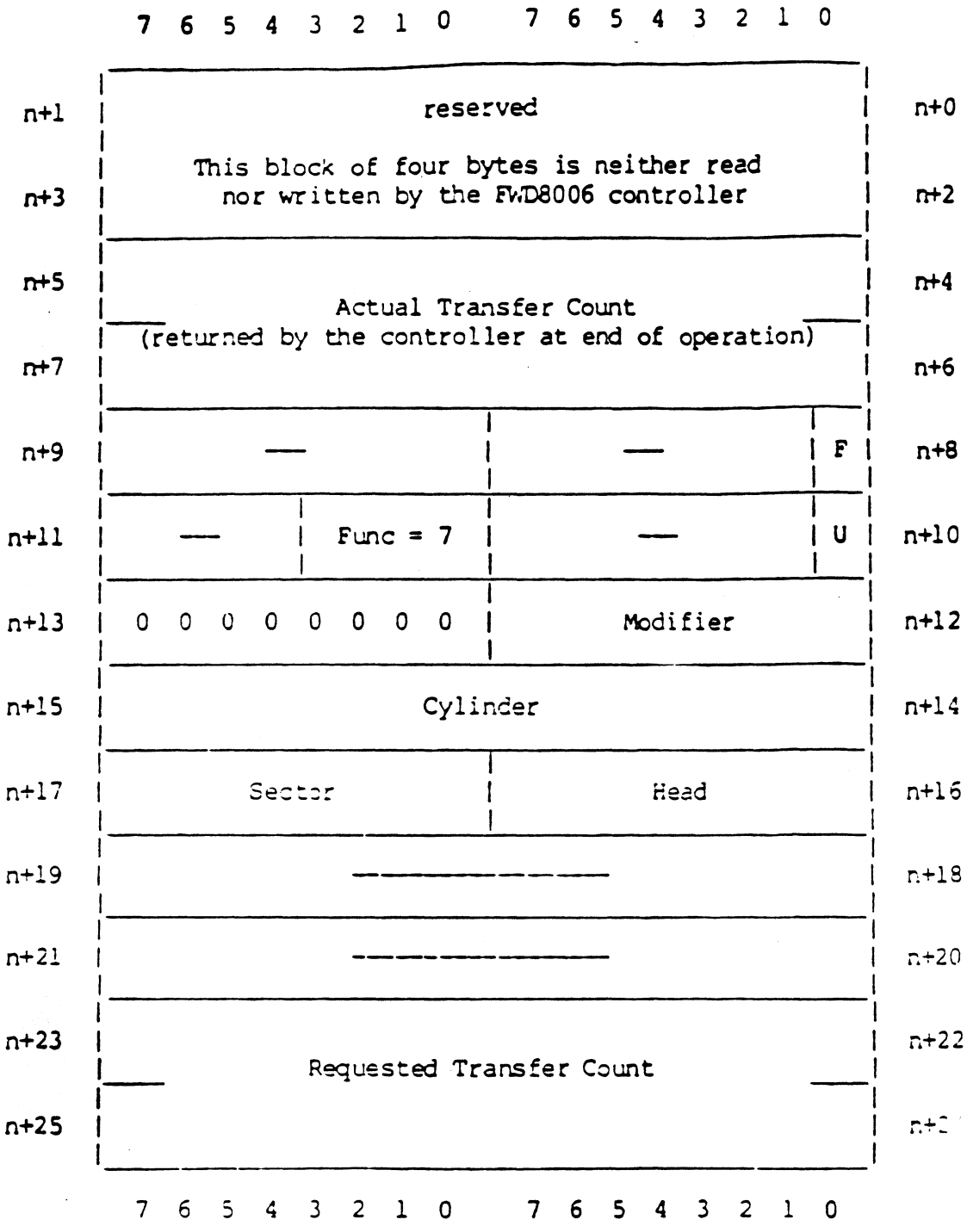


Figure III-16. IOPB - Write Buffer Data Command

WRITE BUFFER DATA

The Write Buffer Data command is identical to the Write Data command except that no data is transfered from the Multibus buffer. Typically this command is used to write the same data pattern to multiple sectors on a disk. If the Requested Transfer Count specified in the IOPB is more than the number of bytes in the first sector specified, sequential sectors will be written with the same data starting each time at the first byte in the controller's on board sector buffer. If the Requested Transfer Count is not a multiple of the number of bytes in a sector the last sector written will be written with partial buffer data. The balance of the sector will be written with zero data bytes.

For details concerning IOPB parameters required for use of this command refer to the description of the buffered Write Data command.

9. Seek - Initiate Drive Carriage Positioning

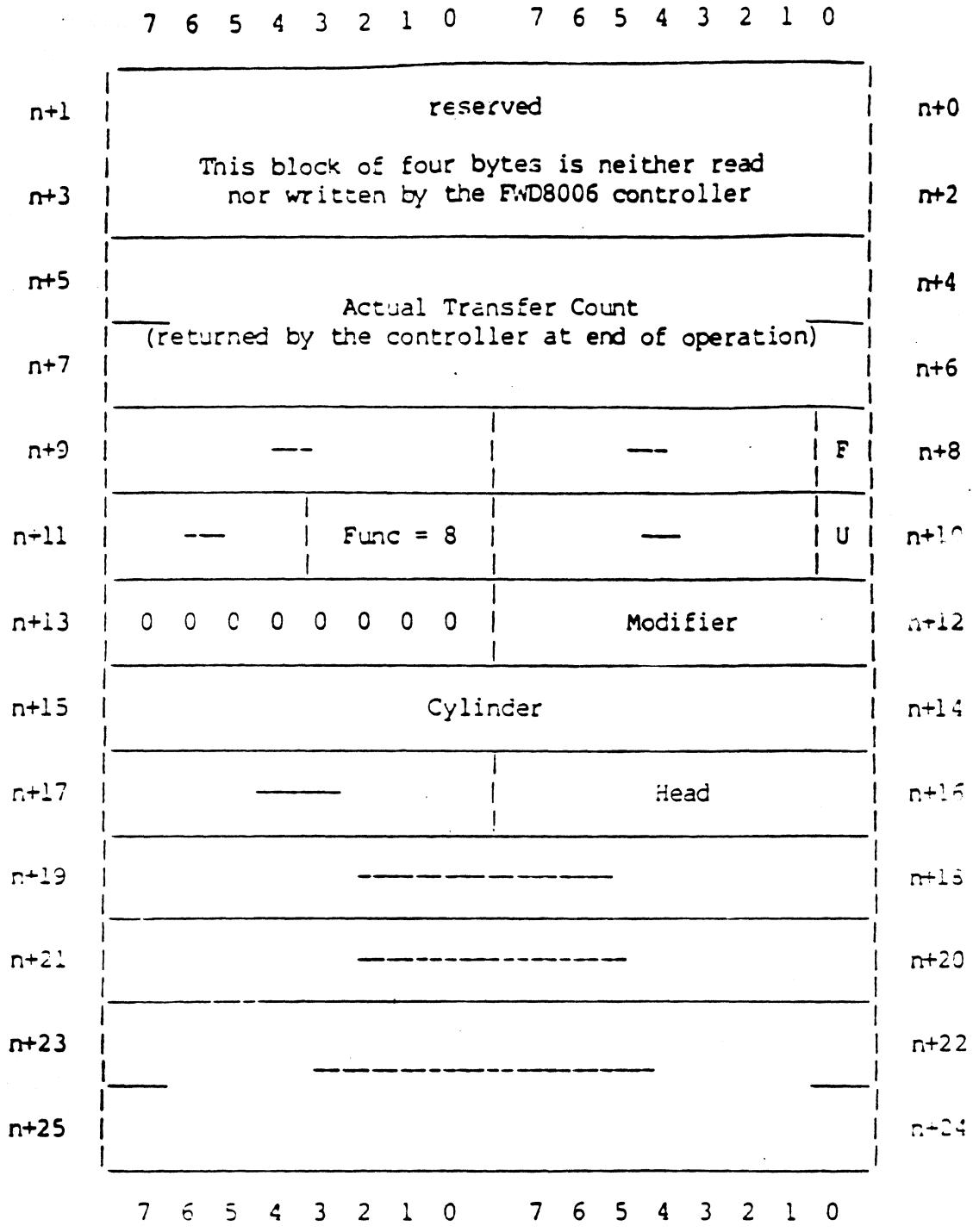


Figure III-17. IOPB - Initiate Track Seek Command

INITIATE TRACK SEEK

The Initiate Track Seek command is used to position the read/write heads on the specified drive without transfering data. Because each of the data transfer commands includes implied seek, the primary use of this command is to allow the controller to perform other activities on other drives in the system while the heads are being positioned. Unlike other commands, the Initiate Track Seek command does not wait until the heads have reached the requested track before continuing. Once the controller has determined that the drive is ready to be positioned the seek is initiated and status is returned. When the heads on the drive have reached the specified track, seek complete status is posted and a seek complete interrupt is generated. Because the command is completed when the seek is initiated, several seeks may be in progress on different drives at the same time (overlapped seek). This facility allows the host to start seeks on multiple drives and then read or write data from the first of them to reach the specified track.

10. Buffer I/O - Transfer Controller/Memory Data

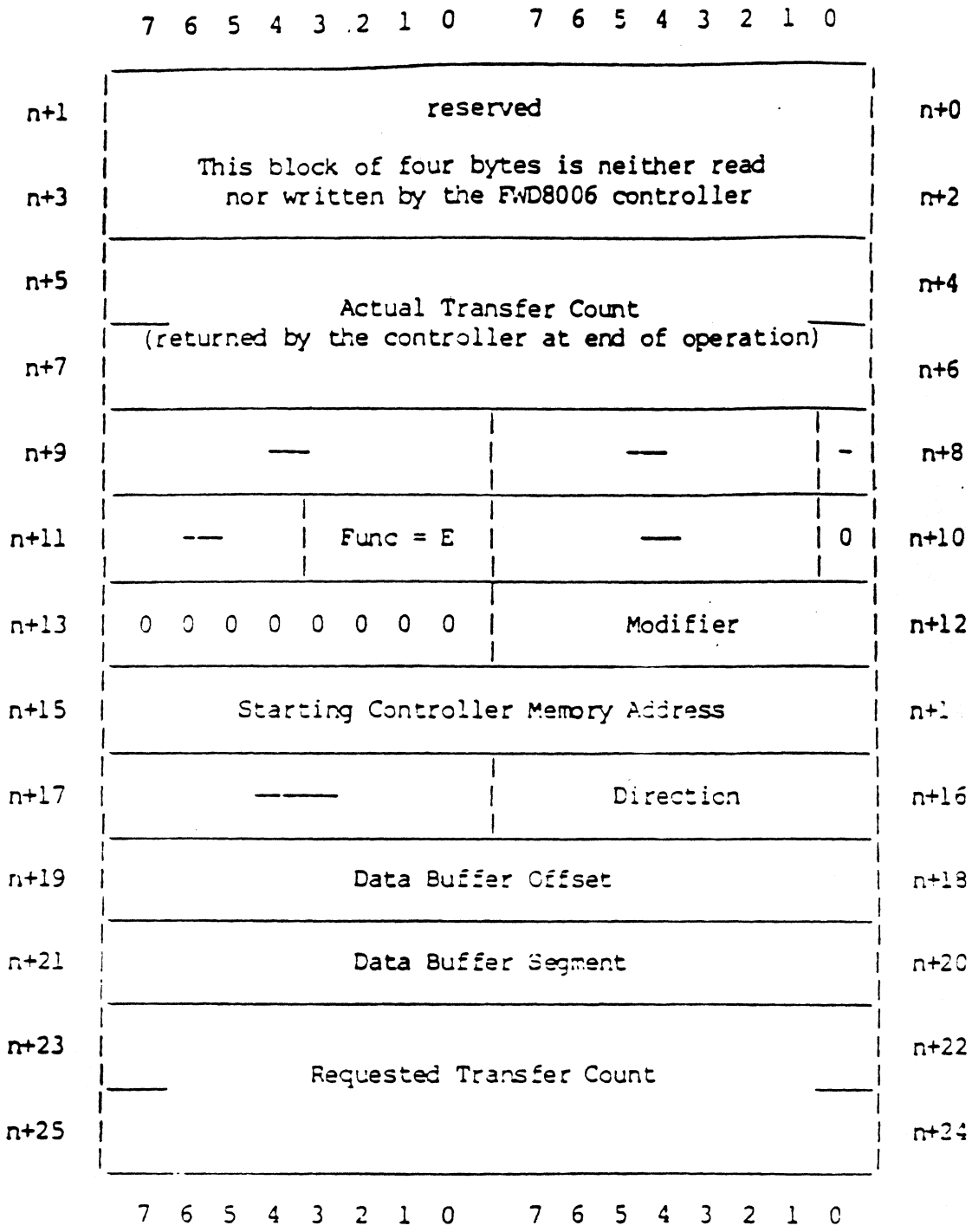


Figure III-18. ICPB - Buffer I/O Command

BUFFER I/O

The Buffer I/O command allows the host to transfer data between the controller's on board buffer and a Multibus memory buffer. It is used primarily for diagnostic purposes and for filling the buffer for subsequent Write Buffer Data commands. No disk accesses are involved. The Multibus buffer is addressed by the Data Buffer Offset and Segment fields in the ICPB as in disk transfer commands. The on board buffer starting address is specified in the Cylinder field of the ICPB. For compatibility with the iSBC 215, all addresses in the on board buffer must be between 4000H and 45FFH. Similarly, the [illegible] of the on board buffer used as the sector data buffer for disk transfer [illegible] begins at address 4010H. The Head field in the ICPB is used to specify the direction of data transfer; 00H for on board to Multibus (read the buffer) and FFH for Multibus to on board (write the buffer) transfers. The Requested Transfer Count is used to specify the number of bytes to transfer in the same way as for disk data transfer commands. Note that, unlike the iSBC 215, attempts to specify a transfer which begins or ends at an internal controller address outside of the on board buffer is illegal and will return an appropriate error status.

11. Diagnostic - Begin Fault Detection Diagnostic

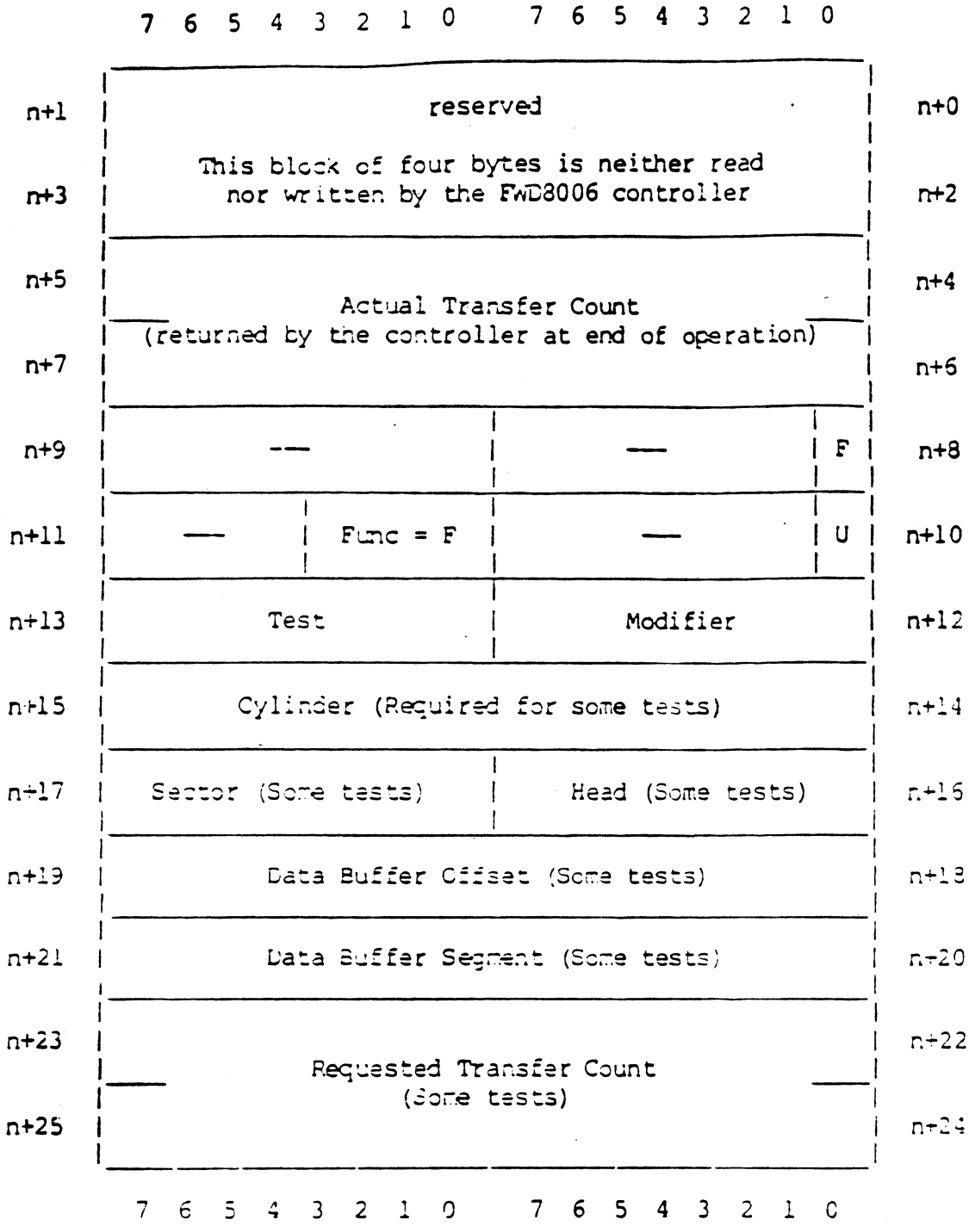


Figure III-19. IOPB - Diagnostic Command

DIAGNOSTIC

The Diagnostic command is used to exercise the controller and disk drive system either to confirm that it is working correctly or to help in determining what component of the system has failed. A number of different diagnostic tests are available and are selected by setting proper values in the Test and Modifier fields in the IOPB.

If the Extension bit in the Modifier field (bit 7) is not set, the diagnostics available are equivalent to those of the IBC 215:

Test	Diagnostic Performed
00H	A seek is executed to the last cylinder on the drive. Head 0 is selected and a read ID is performed to verify position. The first sector is written with a 55AAH pattern. The same sector is read to verify the data and ECC/CRC.
01H	The controller self test is executed (Section V-A).
02H-FFH	The drive heads are re-positioned to cylinder 0 (recalibrated).

If the Extension bit in the Modifier field (bit 7) is set, the value of the Test field is used to select exercises as shown below. Note that many of these sub-functions will execute continuously until an error occurs or the controller is reset.

Test	Diagnostic Performed
00H	No activity - returns immediate status
01H	Executes controller self test once (Section V-A)
02H	Executes controller self test continuously (Section V-A)
03H	Executes system and drive test continuously (Section V-B)
04H	This sub-function is for various drive alignment and maintenance activities. The specified drive is positioned to the cylinder and head specified, the head is loaded if the floppy is an AC drive, and the drive is left selected indefinitely.

05H This sub-function is also for drive alignment and test and is for floppy disks only. The specified drive is selected, a seek is performed to the cylinder and head specified, and the head is loaded and unloaded about 5 times per second indefinitely if it is an AC drive. For DC drives, 04H and 05H are identical.

06H This sub-function is also for drive alignment and test. The specified drive is selected, a seek is performed to the cylinder and head specified, the head is loaded if the floppy is an AC drive, and then the drive is stepped. between the specified cylinder and the specified cylinder + 1 indefinitely.

07H This sub-function is also for drive alignment and test. The specified drive is selected, a seek is performed to the cylinder and head specified, the head is loaded if the floppy is an AC drive, and the entire track is written from index to index with an unbroken repetition of the data byte in the Sector field of the IOPB. For floppy disks the track is written with the density and encoding as specified in the last Initialize Command for the drive. Note that after this sub-function executes the specified track must be re-formatted before it may be used for data.

08H This sub-function is used in conjunction with the next sub-function to exercise the Winchester error correction circuitry to verify its proper operation. Its execution very similar to a Write Buffer Data command except the Requested Transfer Count is assumed to equal the length of one sector and the CRC or ECC bytes written to the disk following the sector data are taken from the controller's on board buffer instead of from the CRC/ECC circuitry. This allows a sector to be written to either a floppy or Winchester disk drive with a previously calculated CRC or ECC error included.

09H This sub-function is the complement of sub-function 08H described above. Its execution is similar to a Read to Buffer and Verify command except that the Requested Transfer Count is assumed to be equal to the length of one sector and the CRC or ECC bytes following the data on the disk are also read into the controller's on board buffer. No CRC or ECC error detection or correction is enabled when this sub-function is invoked.

0AH This sub-function is used to turn on the on-board LED.

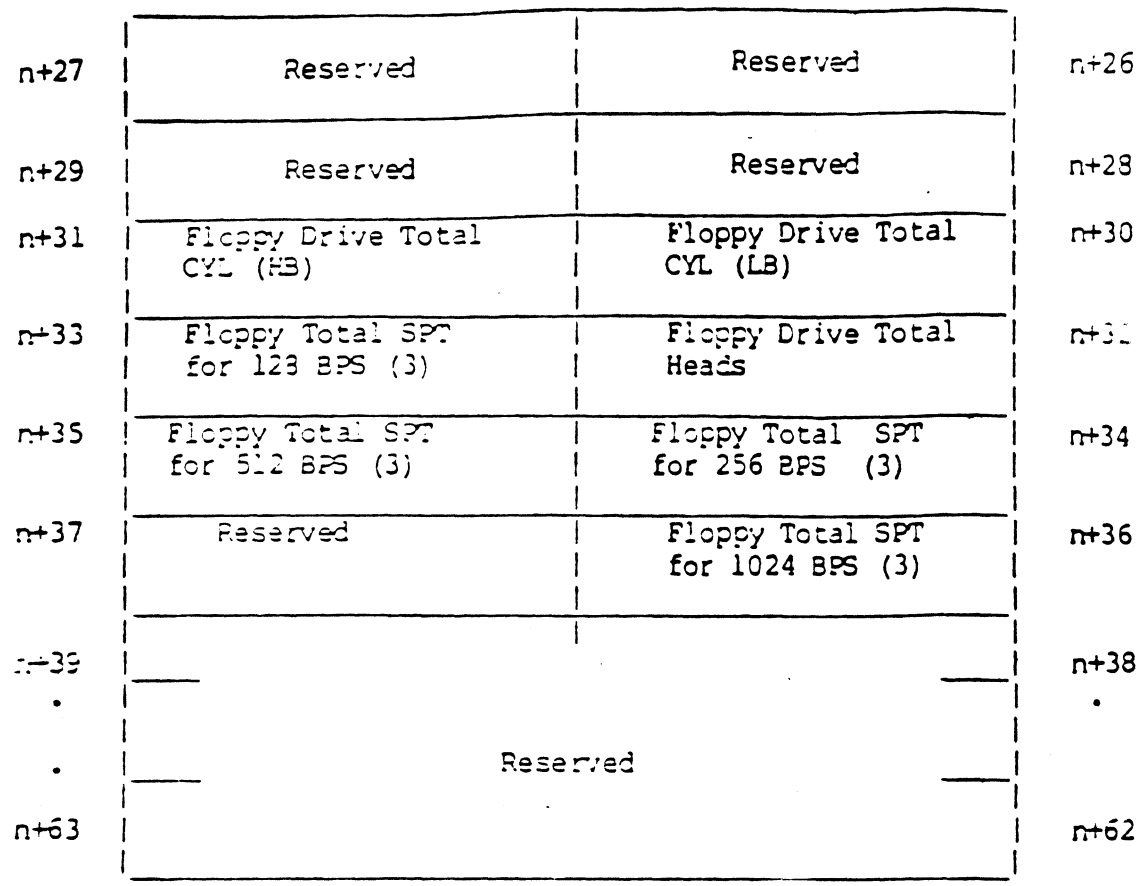
0BH This sub-function is used to turn off the on-board LED.

0CH This sub-function moves Winchester heads to the shipping track of the drive type defined by straps W1-W4 (section II-D).

0DH This sub-function allows the host to read the on-board drive straps, option straps, controller type and firmware version. The host defines the four byte extension (bytes n+0 to n+3) addressed by the Data Buffer Offset and Segment fields in the ICPB. The controller then returns a 60-byte (bytes n+4 to n+63) information block to the address immediately following the four byte extension as shown below: (Test numbers 0EH through 0FFH are reserved for future use)

	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	
n+1	Reserved	Floppy Encoding	n+0
n+3	Reserved	Reserved	n+2
n+5	Controller Type (HB)	Controller Type (LB)	n+4
n+7	Firmware Version (HB)	Firmware Version (LB)	n+6
n+9	Reserved	Winchester Type	n+8
n+11	Reserved	Floppy Type	n+10
n+13	Controller Straps (Byte1)	Controller Straps (Byte0)	n+12
n+15	Reserved	Reserved	n+14
n+17	Winch Total Cyl (HB)	Winch Total Cyl (LB)	n+16
n+19	Winch Total SPT (1) for 128 BPS (2)	Winch Total Heads	n+18
n+21	Winch Total SPT for 512 BPS	Winch Total SPT for 256 BPS	n+20
n+23	Reserved	Winch Total SPT for 1024 BPS	n+22
n+25	Reserved	Reserved	n+24

Figure III-20. ICPB Execution - Diagnostic Command, Sub-function 0DH



(1) SPT: sectors per track
(2) BPS: Bytes per sector
(3) If the byte is zero, it means that the corresponding sector size is not supported.

Figure III-20. IOPB Extension - Diagnostic Command, Sub-function 0DH continued

Byte 0: Floppy Encoding Byte

This byte defines the floppy encoding format so that the controller can, based on the floppy drive straps defined on board, determine number of sectors per return it to the information block. The byte is assigned the following different formats:

00H FM - 8" IBM compatible or 5 1/4" iRMX 86 compatible
01H MFM - 8" IBM compatible or 5 1/4" iRMX 86 or IBM PC compatible.
22H MMFM - 8" Intel SBC 202 compatible
06H MFM - 8" SMS FWD0106 compatible
07H MFM - 5 1/4" DEC PC and SMS FWD0106 compatible
08H MFM - 5 1/4" SMS FWD0507-M compatible

All other values are reserved.

Byte 1 thru Byte 3 are reserved.

Controller Type is 8006 decimal or 1F46H

Firmware Version bytes have the following format:

high byte=000000008
bits 7,6 of low byte = version number
bits 5,2 of low byte = revision number
bits 1,0 of low byte = 00B

Winch Type	00H	No Winch
	01H	Seagate ST406
	02H	Seagate ST412
	03H	Seagate ST419
	04H	Seagate ST506
	05H-07H	Reserved
	08H	Rodime R0208
	09H	Vertex V130
	0AH	Vertex V150
	0BH	Vertex V170
	0CH	Tandon TM703
	0DH-0FH	Reserved
	10H	Atasi 3033
	11H	Atasi 3046
	12H	Maxtor XT-1065
	13H	Maxtor XT-1105
	14H	Maxtor XT-1140
	15H	Rodime R0204
	16H-FFH	Reserved
Floppy Type	00H	No Floppy
	01H	SA400
	02H	SA410
	03H	SA450
	04H	SA460 and FD-55F
	05H	MPI91
	06H	MPI92
	07H	TM100-3
	08H	TM100-4
	09H	SA810
	0AH	SA850
	0BH	SA860
	0CH	TM848-1
	0DH	TM848-2
	0EH	FD-55B
	0EH-0FFH	Reserved

Controller Strap Byte 0 reports the strap status: a 1 means strap installed, 0 means removed.

Bit 7-6	Reserved	
Bit 5	W13	Enable no host diagnostics
Bit 4	W8	Enable 2 minute POR delay
Bit 3	W9	Enable self test
Bit 2	W10	Enable word transfer
Bit 1	W11	Enable extended mode and buffered/direct transfers
Bit 0	W12	" "

Controller Strap Byte 1 reports Led state and floppy size

Bit 7-2	Reserved
Bit 1	Floppy Size=5 1/4" if 1, else 8"
Bit 0	LED state: 1=on, 0=off

Bytes n+0—n+3 are supplied by the host prior to the command execution.
n+4—n+63 are returned by the controller when the command is finished.

DATA TRANSFER CONTROL

In addition to the level of control provided by the variety of commands which may be issued to the FMD8006, many of these functions may be further tailored to the needs of a particular system by controlling certain data transfer parameters. These controls are implemented via bits in the Modifier byte in the IOPB. The definitions of each of the bits in this byte are repeated below.

Bit	Function enabled when bit is set (=1)
7	Enables extensions to iSBC 215 commands. Must be set, and W11 installed, to enable bits 6 or 5.
6	Enables direct Winchester disk/Multibus transfers rather than fully buffered transfers.
5	Enables word data transfer within data buffer even if W10 is removed specifying byte transfers.
4	Undefined - Must be set to zero
3	Undefined - Must be set to zero
2	Enables floppy Deleted Data AM writing/reading
1	Inhibits all error retries
0	Suppresses command completion interrupt

The function of bits 7, 1, 0 are trivial as described above. But bits 6, 5, and 2 need further explanation.

Bit 6 - Direct Winchester Disk/ Multibus Transfer

In systems where higher performance is required, and where sufficient Multibus bandwidth is available, direct disk/memory transfers may be used with the Read Data and Write Data commands. When this option is selected by setting bits 7 and 6 in the modifier, data is transferred in exactly the same way as for buffered transfers except that the on board buffer is not used for any full sectors moved and the Multibus is locked during the transfer. Data is passed directly between the disk and the Multibus memory a byte or word at a time. Because the data is transferred as the disk is written or read, no time need be allocated at the end of each sector for data movement between Multibus memory and the controller. This is especially useful when reading or writing disk which were formatted without interleave; data is moved at full disk speed instead of one disk revolution per sector. Note, however, that transferring data directly between the disk and memory requires that a Multibus memory access be done whenever a byte or word of data has been passed from/to the disk. It is possible that occasionally the controller will either not get access to the Multibus quickly enough to store the data collected before it is overrun by the next byte from the disk when reading, or to fetch the next byte or word before it is required by the disk when writing. Should this occur, the FMD8006 will recover automatically. Multibus memory pointers and counters will be internally reset and the disk sector involved will be accessed again. On this second transfer the data in this single sector will be buffered through the internal FMD8006 buffer. This allows full recovery from data overrun/data late conditions without CPU intervention. Note that because this recovery requires two full disk revolutions, systems which require that the FMD8006 be frequently

excluded from Multibus access for extended periods should exclusively use buffered transfers with an appropriate interleave factor for optimum disk throughput. Note that the above description also applies to floppy drives except that:

1) THe floppy direct/buffered transfer is controlled by W11 and W12 (Section II-C), not by bit 6.

2) The Multibus is not locked when floppy direct transfer is enabled.

Bit 5 - Selective Word Wide Multibus Transfers

In most systems the Multibus memory accessed by the FWD8006 is capable of either byte or word transfers. In these systems the controller will normally be optioned to do word transfers wherever possible to reduce Multibus usage (Section II-C). There are cases, however, where portions of system memory will allow only byte wide transfers while the remainder will support either. In these systems, it is useful to specify the transfer width which the FWD8006 may use on a command by command basis. This flexibility is provided by bit 5 in the IOPB modifier byte. Setting this bit together with bit 7 informs the controller that even though its normal mode of Multibus access is byte wide only, the data buffer addressed by the Data Buffer Segment and Data Buffer Offset for this command may be accessed using word wide transfers.

Bit 2 - Floppy Deleted Data Address Marks

For a variety of reasons it is often valuable to be able to mark the data written to floppy disks in a way which may be detected by the controller when the sector is read. This might be done, for example, to flag bad sectors, to mark free sectors, etc. This may be done with the FWD8006 by using Deleted Data address marks. As shown in Section IV, the data address mark is recorded on the disk immediately preceeding the sector data whenever a sector is written. The data AM is read and interpreted by the controller but is not transferred as part of the sector data. For most operations a normal data AM is used. Under these conditions the data AM is only used by the controller and is invisible to the rest of the system. In those cases where a floppy disk sector is to be specially marked, however, a different data address mark is written along with the data. This special AM, the Deleted Data AM, is recognized by the controller when the sector is read and special action taken.

Both the Write Data command and the Write Buffer Data command may be made to write floppy disk sectors with Deleted Data AMs in place of Normal Data AMs. For each of these commands, if bit 2 in the IOPB modifier byte is set when the command is invoked, all sectors written will have Deleted Data AMs. Note that this will be multiple sectors if the Requested Transfer Count in the IOPB is larger than the length of a single sector.

Floppy sectors with Deleted Data AMs are recognized by both the Read Data command and the Read to Buffer and Verify commands. In each case, the action taken is determined by the setting of two bits in the ICPB modifier byte, bits 7 and 2. The table below lists the four combinations of these two bits and the processing which results:

Bit 7	Bit 2	Action
0	0	Sectors with Deleted Data AMs are skipped
0	1	Sectors with Normal Data AMs are skipped
1	0	Sectors with Deleted Data AMs are skipped
1	1	Transfer stops after the first sector with a Deleted data AM is encountered and an error status will be returned with an event code of 42H: deleted data AM encountered.

Sectors which are skipped are treated normally at the diskette interface, i.e. they are read and counted, but the data contained in the Multibus buffer is written as if these sectors did not exist. In addition, the Actual Transfer Count returned in the ICPB and used to compare against the Requested Transfer Count is not affected by the number of bytes contained in the skipped sectors. Note that by skipping sectors in this way it is possible to run out of sectors on the media before even a small Requested Transfer Count

G. ISSUING COMMANDS and RECEIVING STATUS

Because the main channel of communication between the host and the FWD8006 is the memory based tables, the protocol required for issuing commands and receiving status is very simple and straightforward. It is also essentially constant for all commands. The sequence required for command execution by the host is:

After any reset or if the controller alone is to be reset:

A. Put FWD8006 into Reset
B. Initialize WUB, CCB, and CIB as required
C. Issue Clear and Start
D. Wait for not BUSY

Then for any subsequent commands:

E. Process any pending status while waiting for not BUSY
F. Initialize CCB, CIB, and ICPB for command
G. Issue Start
H. Wait for interrupt or poll for status

Each of these steps may be broken down into details as follows:

A. Put FWD8006 into Reset

Issuing a programmed I/O Reset command (Section III-B) clears any pending interrupts and applies a hardware reset to the FWD8006 controller hardware. All drives are de-selected. Any disk writes in progress are abruptly terminated and floppy heads are unloaded if it's an AC drive.

B. Initialize WUB, CCB, and CIB as required

Wake-up block (WUB) is set up pointing to the CCB.
Channel control block (CCB) is set up pointing to the CIB.
Channel Invocation Block is set up.
Status Semaphore in CIB is cleared (00H).
BUSY in CCB is set (0FFH).

C. Issue Clear and Start

Issuing a programmed I/O Clear removes the hardware reset from the FWD8006 hardware allowing it to recognize Start.
Because the controller has just been in Reset, the first programmed I/O Start command it receives is treated in a special way. Rather than attempting to fetch an ICPB and execute a command, the FWD8006 will examine it's hardware straps to determine the Multibus memory address of the WUB and then will chain from the WUB to the CCB internally saving the address of this block. It then clears BUSY in the CCB without issuing any status.

D. Wait for [illegible] BSY

While the FWD8006 is collecting its table addresses, the host must refrain from issuing any further commands. When the BUSY flag in the CCB has been cleared the FWD8006 is ready to receive commands. These normally will start with commands to initialize the various disk drives in the system.

E. Process any pending status while waiting for not BUSY

Under normal circumstances the controller will be not BUSY already. If overlapped operations are used, however, the host may have to wait for a previously issued command to complete before executing another. During this time status may be posted for seek completion or for drives coming ready. This status should be processed as it is posted.

F. Initialize CCB, CIB, and IOPB for command

An Input/Output Parameter Block (IOPB) is set up for the command which is to be executed next.
The CIB must point at this IOPB.
BUSY is set (0FFH) in the CCB.

G. Issue Start

Once the proper table entries have been set, a programmed I/O Start command is issued. This causes the FWD8006 to fetch the table contents and begin execution of the command.

H. Wait for interrupt or poll for status

When the controller has [illegible]shed execution of a command, when a seek completes, or when a disk drive becomes ready, status will be posted. The host may then examine the status in order to determine if an error occured and to decide what command to issue next.

Whenever the FWD8006 has status information to pass to the host it examines the Status Semaphore in the CIB. If it is zero, the FWD8006 will assume that previous status information has been accepted by the host. It then writes the new status byte to the Operation Status byte in the CIB and sets the semaphore non-zero. The host should follow the reverse protocol. That is, whenever the Status Semaphore is non-zero the host should assume that the Operation Status byte contains new status information. It should fetch that information as required and then clear the Status Semaphore to tell the controller that the status has been received. Bits in the Operation Status byte contents may be decoded to determine status type and whether or not an error has occured.

If the FWD8006 finds the Status Semaphore to be non-zero when it has status to pass to the host it will save the status information in an internal queue and poll the Status Semaphore periodically waiting for it to be cleared. Note that if the status to be passed is for Operation Complete the FWD8006 will not clear the Busy byte in the CIB until it has put the status into the Operation Status byte.

H. ERROR PROCESSING

Under normal operating circumstances, commands issued to the FWD8006 will be processed without error. The Operation Status byte for these commands will reflect this fact by having the Summary Error bit (bit 7) equal to zero when status is reported. Under these circumstances, the host will take appropriate action whenever status is reported based on whether the status is for the end of an operation, a seek completion, or a new drive ready.
If an error does occur, the level of host action must be determined by the application and the type of error. The Transfer Status command will often be used at these times to determine details of the error. Errors will generally fall into one of three categories. The first of these is operational errors. These errors are caused by user software or operator errors and include such things as illegal cylinder, illegal sector, wrong sector length, write protect error, etc. The FWD8006 will report these errors so that the host or operator may take whatever corrective measures are required. The second category is disk media errors. These occur because of flaws in the media itself (hard errors) or because of an occasional error on media read operations by the disk system (soft errors). The FWD8006 is pre-programmed to handle the majority of these errors directly, without host intervention. If automatic retries are not disabled by setting the NRE bit (bit 1) in the Modifier field in the IOPB, the FWD8006 will attempt recovery from these errors using a retry policy which is dependent on the type of error encountered. For problems which may have been caused by a seek error the procedure used is to seek the drive to cylinder 0 and then back to the proper cylinder. This will be done twice before the error is assumed to be hard. The errors handled in this way are:

Illegal alternate track
Cylinder address mismatch
No address marks on track

For errors which may have been caused by media read errors the read operation is repeated up to eight times before an error is considered to be hard. Errors handled in this way are:

Data CRC errors on floppy
Uncorrectable data ECC errors on Winchester
Missing Data Address Mark
Sector ID not found

In each of the cases above, the number of retries attempted is always available at the end of the command by executing the Transfer Status Command.

In addition to the error recovery procedures covered above, the FWD8006 supports error correction on the data fields of the Winchester drives. When an ECC error is detected, the sector is re-read until either it is read correctly or until the same residue is read twice. When the latter occurs, an attempt is made to recover the data and continue.

Whenever the FWD8006 is transferring data directly between a Multibus data buffer and a disk it is possible to have a data late (disk write) or data overrrun (disk read) error. These are often a result of the fact the the disk must transfer data at a fixed rate. Thus for direct transfers with no buffering, the Multibus memory must be accessed at regular intervals. If either of these errors occurs, the FWD8006 will revert to a buffered transfer for that sector to insure that the command is eventually finished. Because this results in an additional two revolution delay for each sector recovered, the system designer who wishes to use direct transfers must insure that these errors will occur infrequently enough to not degrade system performance excessively.

When the Transfer Status command is issued with the extension bit in the Modifier field (bit 7) set, the last byte returned is an internal event code for the precise error detected. As an additional aid to determining the exact cause of an error or malfunction this byte may be decoded using the values in Appendix A. If the most significant bit of the byte is set, the error reported was successfully recovered with ECC or controller retry.

I. MEDIA FLAW MANAGEMENT

Because of a number of physical factors including high bit density, manufacturing processes, etc., Winchester disk surfaces generally contain a small number of flaws. These imperfections in the magnetic coating can cause data written over them to be unreadable. These flaws range from less than a bit to many bits in length. Although the FWD8006 features an error correcting code scheme on these disks there are a number of reasons why additional measures are required to deal with these media flaws. They include:

Flaws may occur in parts of the track outside of the data fields such as sync fields.

Flaws are sometimes longer than the error correcting capabilities of the ECC polynomial.

Permanent hard errors significantly reduce the probability of a separate soft error being correctable.

In order to allow the user to avoid these flaws, the FWD8006 incorporates a mechanism for mapping flawed tracks. Whenever a Winchester track is formatted by the FWD8006, flag bits are set in each of the sector header fields to indicate whether that track is a normal data track, a defective track containing media flaws, or an alternate track reserved to replace a defective track. This system allows a track to be formatted as a defective track and to include in the data on that track a pointer to an alternate track to be used for the data. Once this has been done the FWD8006 will automatically seek to the alternate track whenever it encounters a defective track. This works as follows: The controller starts the sector search assuming that it is on a normal track, and the controller will time out after one revolution if a defective track is encountered because the sector ID of a normal and defective track can never match. The controller then issues a Read ID command to find out that it is a defective track followed by a read command to read the alternate track number from the defective track. The controller then seeks to the alternate track and execute the command. This operation is invisible to the host system except for the extra time required to seek to the alternate track and to return after the track has been accessed. In order to use this facility, the user will normally reserve several cylinders at the end of the disk for alternate tracks. To reduce access time, however, alternate tracks may be located away from the innermost cylinders. When the controller encounters an alternate track in place of a normal data track, the alternate track is skipped without any data transfer. That is, the next data is transferred at the sequential head address or at the next cylinder if the alternate track skipped is at the highest head address of the cylinder. Note, however, that even though this process is invisible to the host system, proper allocation of this skipped track must be made in the host system's disk access routines, because if the alternate is treated as a normal track, the controller cannot find the pointer information from the data field of an alternate track. For information on formatting defective and alternate tracks refer to Section III-E-3.

IV. MEDIA ORGANIZATION

A. PHYSICAL MEDIA ORGANIZATION

The physical characteristics of the various disk drives supported by the FWD8006 and the requirement that the storage available on the disk be addressable in manageable parcels combine to present a number of parameters to the host which must be manipulated properly to access the disk drive completely and efficiently. In general five parameters are used to address the smallest block of data in the disk system. They are:

Type
Unit
Cylinder
Head
Sector

Because the FWD8006 is a combined controller, the host must first specify the Type of disk to be accessed, floppy or Winchester; and because the FWD8006 will support as many as two of each drive Type, the host must specify which Unit (0 or 1) is to be used.

Having specified a particular disk drive, the physical construction of the drive now plays a role in further specifying the data's location. Each of the drives supported by the FWD8006 records data on flat disk surfaces which somewhat resemble phonograph records (without grooves) which are coated with a thin layer of magnetic material. In the case of the floppy disk drive, these disks are made of flexible plastic and may be removed, in their protective jackets, from the drive. Each drive may contain only one disk at a time. The Winchester drives, on the other hand, use rigid disks made of aluminum which are not removable; being sealed inside the clean environment of the drive. Unlike the floppy drive, the Winchester often has mutiple disks in one drive mounted on a common central axis. These disks are parallel to each other and turn together.

The number of disks in any single drive and the number of sides, or surfaces, on these disks which may be used for recording data varies depending on drive type, manufacturer, and capacity. In each case, each data recording surface is associated with a magnetic head assembly which is used to read or write data onto that surface. The magnetic heads for the various surfaces are mounted in a line, parallel to the axis of the disks. As the disks turn, a band of magnetic material on each surface sweeps under the head where it may be written or read. Each band of material is called a track. The collection of tracks under all the heads is called a cylinder because they resemble a physical cylinder; concentric with the various disks and with a radius equal to the distance from the center of the disks to each of the heads. The heads may not be moved relative to one another, but the entire assembly (sometimes called a carriage or comb) may be moved so that the heads are all closer to, or farther from, the center of their respective disks. This movement causes a different set of magnetic bands (tracks) to sweep under the heads and thus defines a new cylinder.

This physical drive construction leads to two more of the parameters used to locate data in the disk system. The host must specify which of the cylinders contains the data. The controller uses this information to move the heads in or out to the correct location. This movement, called seeking, is done in discrete steps by either the controller or the drive itself and accounts for much of the time it takes to access a particular piece of data. Cylinders are numbered sequentially from zero which is the cylinder at the outside edge of the disks. In addition to specifying the cylinder number required, the host must also specify the surface to be used. This is done by specifying a head number. Heads are also numbered sequentially, starting from zero. Since switching between heads may be done at much higher rates than moving the heads between cylinders, all of the tracks on one cylinder are normally used before the heads are stepped to a new cylinder. The combination of cylinder number and head number uniquely locates one track on the disk drive. Because floppy disks are always used one diskette at a time, floppy cylinders are sometimes called tracks and floppy heads called sides. Thus "track 40 - side 1" really means "cylinder 40 - head 1". Single sided diskettes may only be recorded on one side: head 0. Double-sided diskettes may be used on both sides: head 0 and head 1.

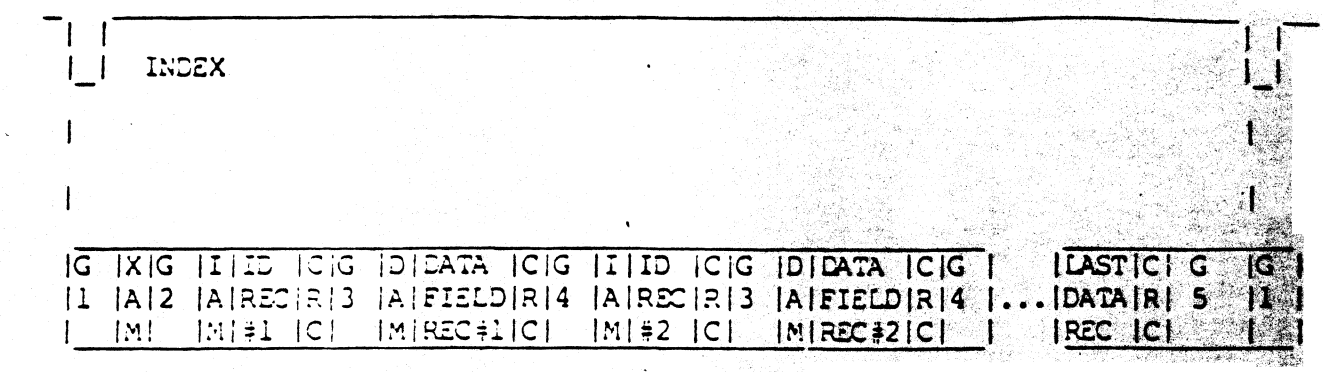
The data capacity of each track on a disk is still too large for most applications, so the track is further divided into a number of equal length records called sectors. Some disk drives provide hardware signals which may be used to determine sector locations within the track. This is not true of the drives supported by FMD3106, however, and for these drives special data is recorded on the track with normal data, to locate each sector. This scheme, called soft-sectoring, relies on writing special bit patterns on the disk which may be located by search hardware later. The process of writing these special patterns, sector identification data, and default data sectors is called formatting. Since the special data used for soft-sectoring is not over-written when user data is written, tracks need only be formatted once unless a new allocation of the track is to be specified, e.g. longer or shorter data sector areas, or in response to certain hard errors.

Once the tracks on a disk have been formatted, the last piece of addressing information is available: the sector number. Because any particular sector will only pass under the head once per disk revolution, rotational latency also contributes significantly to disk access time. Schemes using sector interleave help to reduce this overhead by increasing the probability that the next sector to be read or written will be one of the first to come around to the head.

B. 8" FLOPPY DISK FORMAT

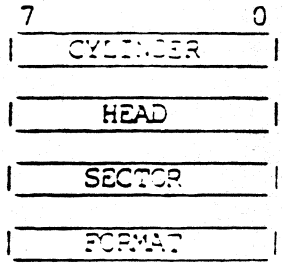
The FMD8006 supports floppy diskettes in both IBM single/double density formats and the SBC 202 double density format. In these formats the position of each sector on a track is marked by the controller by writing a special pattern, called an address mark (AM), on the diskette. The address mark, along with other identification data (track, head, sector and format), is written only when the diskette is formatted. In normal write operations only the data, data address mark and data CRC bytes are written.

The arrangement of sectors, IDs, etc. on each floppy track is shown below:



where G1 to G5 are GAP1 to GAP5
XAM is Index Address Mark
IAM is ID Address Mark
ID REC is ID Record (4 bytes - see below)
DAM is Data Address Mark
CRC is Cyclic Redundancy Check (2 bytes)

Each ID record consists of 4 bytes in the following format:



where format 00H = 128 bytes/sector
01H = 256 bytes/sector
02H = 512 bytes/sector
03H =1024 bytes/sector

The following table summarizes the aspects of the various formats:

Format	Code	Bytes/ IAM	Bytes/ Sect	Sect/ Track	Gap 1	Gap 2	Gap 3	Gap 4	Gap 5
IBM SD	FM	1	128	26	40+6	26+6	11+6	27+6	274
		1	256	15	40+6	26+6	11+6	42+6	212
		1	512	8	40+6	26+6	11+6	66+6	313
		1	1024	4	40+6	26+6	11+6	80+6	667
IBM DD	MFM	4	128	44	80+12	50+12	22+12	38+12	277
		4	256	26	80+12	50+12	22+12	54+12	653
		4	512	15	80+12	50+12	22+12	84+12	485
		4	1024	8	80+12	50+12	22+12	116+12	771
202	MMFM	1	128	52	36+10	18+10	13+10	18+10	282

Note: Gap lengths listed in the form: n+m are actually a gap of n bytes followed by a sync field of length m. Note that the actual length of Gap 4 and of Gap 5 will be somewhat different than the nominal values listed above because of variations in disk rotational velocity.

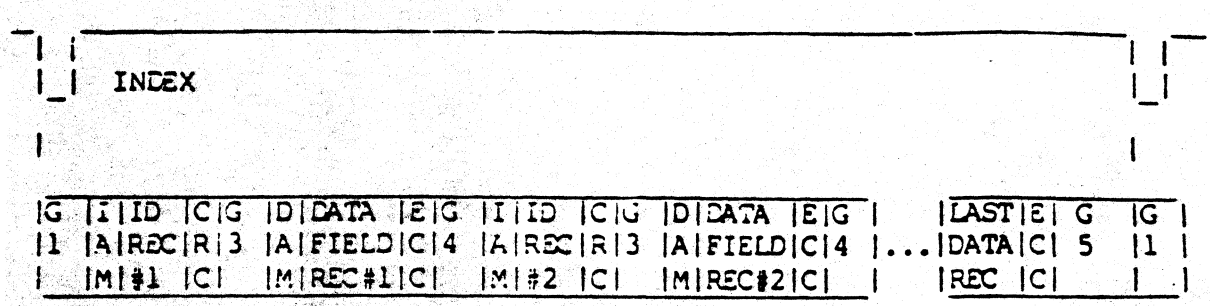
C. 5-1/4" AND 3-1/2" FLOPPY DISK FORMAT

The 5 1/4" and 3 1/2" floppy diskettes have the same arrangement of sectors, ID's 8" floppies. The only difference is the number of sectors per track and gap sizes as listed below:

Format	Code	Bytes/ IAM	Bytes/ Sect	Sect/ Track	Gap 1	Gap 2	Gap 3	Gap 4	Gap 5
FMD0106	MFM	4	128	26	32+12	----	22+12	1+41+12	227
		4	256	16	32+12	----	22+12	1+59+12	229
(DEC PC)		4	512	10	32+12	----	22+12	1+42+12	81
		4	1024	5	32+12	----	22+12	1+120+12	303
FMD050T-M	MFM	4	128	30	34+6	----	14+6	28+6	304
		4	256	18	34+6	----	14+6	36+6	257
		4	512	10	34+6	----	14+6	49+6	234
		4	1024	5	34+6	----	14+6	81+6	529
iRMX-86	FM	1	128	16	40+6	26+6	11+6	23+6	131
		1	256	9	40+6	26+6	11+6	38+6	147
		1	512	4	40+6	26+6	11+6	60+6	512
		1	1024	2	40+6	26+6	11+6	100+6	818
iRMX-86	MFM	4	256	16	80+12	50+12	22+12	50+12	266
(IBM PC)		4	512	8	80+12	50+12	22+12	90+12	832
		4	1024	4	80+12	50+12	22+12	120+12	1400
IBM PC	MFM	4	256	16	80+12	50+12	22+12	54+12	577
(512BPS)		4	512	9	80+12	50+12	22+12	84+12	265
		4	1024	4	80+12	50+12	22+12	116+12	1411

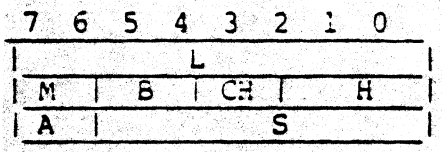
D. WINCHESTER DISK FORMAT

The Winchester drives supported by the FWD8006 are soft sectored and the track format is similar to the floppy disk except that there is no index address mark (XAM) or gap 2 (G2). This means that gap 1 immediately precedes the ID address mark for sector 1. Also, the 2 byte data CRC which is used on the floppy disk is replaced with a 4 byte data ECC capable of detecting and correcting up to a 6 bit burst error. The arrangement of sectors, ID, etc., on each track is shown below:



where G1 to G5 is GAP1 to GAP5
IAM is ID Address Mark (2 bytes)
ID REC is ID Record (3 bytes - see below)
DAM is Data address mark (2 bytes)
CRC is Cyclic Redundancy Check (2 bytes)
ECC is Error Correction Code (4 bytes)

Each ID record consists of 3 bytes in the following format:



Where L = 8 LSB of cylinder number (bit 7-0)
M = bit 9,8 of cylinder number
CH = Logical OR of cylinder number bit 10 and head number bit 3 This allows the cylinder number to exceed 1023 or the head number to exceed 7. If either head number >8 or cylinder >1024, then CH=1. CH will be zero only if cylinder number <1024 and head number <8.

H = 3 LSB of head number (bit 2-0)
*B = bytes/sector code bits
00 B 256 bytes/sector
01 B 512 bytes/sector
10 B 1024 bytes/sector
11 B 128 bytes/sector
A = Alternate/defective track bits
00 B normal track
01 B alternate track
10 B defective track
11 B reserved

*Note that the bytes/sector code defined by B are different from those described in Fig. III-12. The user should use the code defined in Fig. III-12 although the actual code stored on disk is the code defined by B. The controller automatically translates the figure III-12 code into code defined by B before any Winchester access is started.

The following table summarizes the aspects of the various formats:

Bytes/ IAM	Bytes/ Sect	B	Sect/ Track	Gap 1	Gap 3	Gap 4	Gap 5
2	128	11B	54	4+12	3+8	1+26+12	124
2	256	00B	31	4+12	3+8	1+40+12	129
2	512	01B	17	4+12	3+8	1+56+12	183
2	1024	10B	9	4+12	3+8	1+81+12	215

Note: Gap lengths listed in the form: n+m are actually a gap of n bytes f... by a sync field of length m. Gap 4 lengths are composed of 1 byte for write turn-off, a number of actual gap bytes and 12 bytes of sync field. Also, as in floppy formats, the actual length of Gap 4 and of Gap 5 will be somewhat different from the nominal values listed above because of variations in disk rotational velocity.

FAULT DETECTION and DIAGNOSTICS

The FWD8006 is conservatively designed and requires no periodic maintenance or adjustments. Field failures may sometimes occur, however, due to the occasional failure of an IC or other component on the controller. The FWD8006 contains a combination of facilities both to allow the prompt detection of most controller hardware failures and to aid in diagnosing not only controller failures but a variety of drive or cable failures as well. This combination not only adds to system reliability and end user confidence but also shortens the time required to return a failed system to normal operation either in the field or at a local repair facility without the requirement of returning the system to SMS.

A. CONTROLLER SELF TEST

Built into the FWD8006 is the ability to run a series of internal hardware tests following the second programmed I/O Start command after the controller is taken out of hardware reset. Since the first Start is used to collect block addresses, this second Start will be issued for the first controller command. Normally this will be the Initialize command for the first disk drive. These on board tests are enabled by installing option strap W9 on the FWD8006 PC board (Section II-C). Except for a slight lengthening in the response time to the first command issued and the reporting of an error if a fault is detected, the self test execution is entirely transparent to the host. If a fault is detected, status is posted in the Operation Status byte in the CIB. This status byte will have both the "Error" and the "Hard" bits set, but neither "Seek Complete" nor "Operation Complete" will be set. The self test sequence includes:

MCS processor exercise
MCS processor program checksum
Controller RAM and buffer check
Timer and port check
Disk data path check

B. SYSTEM and DRIVE TEST

In order to allow the disk storage system to be tested or exercised either without a host system operating or with a minimum amount of host driver software, the FMD8006 provides a pre-programmed exercise of the controller itself and of all drives attached. This test may be run either by executing the appropriate version of the Diagnostic command (Section III-E-11) or by properly strapping the controller PC board. This strapping passes appropriate disk parameter data to the system test facility within the controller. Strapping options are:

W13-AB	installed (enables no host diagnostics)
W3	not used
W9	not used
W10	not used
W11	installed (format enable, see discussion below)
W12	not used

W1-A3 through W7-BC still have the same drive type definition as the normal operate mode (Section II-C).

Once the controller is powered up, it will access each of the drives which are ready. If the W11 jumper is installed at power up time, and the Winchester write protect is disabled (i.e. W14 disabled), a 15 second delay is automatically enabled, and then each drive will be formatted with 256 byte sectors (IBM DD on floppy) once before the continuous read tests begin. If W11 is installed with Winchester write protect enabled, the controller will format only floppies before the continous read test. If W11 is not installed, the controller goes directly to read from drives and will continue indefinitely.

In addition, each cycle of the test will be preceeded by an execution of the self test (Section V-A above). If a fault is detected during the self test the LED on the controller PC board will be illuminated and drive testing will stop. If one of the drives fails the drive test it will not be read again. This situation will be reflected in the fact that the heads on that drive will not be seeking from cylinder to cylinder. In addition, at the point in the normal test sequence where the failed drive would have been accessed, the LED will illuminate for about one half a second to signal a failure. For action to take if any part of the test should fail, refer to Section V-C below.

!!! Warning !!!

If No-Host diagnostics is enabled (W13-BC installed) and format is enabled (W11 installed) and Winchester write protect is disabled (W14 removed), a format operation will automatically be started after power on and data on Winchester will be overwritten.

C. TROUBLESHOOTING GUIDELINES

Whenever there is reason to believe that the disk system has failed, a number of capabilities of the FWD8006 may be used to aid in determining whether or not there really is a failure and, if so, in which system component the failure has occurred. In most computer systems, the best method for collecting this information is to execute diagnostic programs via the host CPU. Normal FWD8006 commands, particularly the Diagnostic command, may be used to determine quickly whether there is a fault and whether it is located in one of the drives or in the controller itself. If for some reason this method is impractical in a particular situation, the following procedure may be used to isolate the majority of disk system errors. This sequence may also be follo[illegible] a controller is initially received to verify that no damage was done in [illegible]ng.

1. Disconnect all drive cables from the controller PC board.

2. Set the option straps properly for the system in which the controller is to be tested except for straps W1-W13 which should be set as shown below. This unique strapping forces the controller to execute the self test immediately after power on.

W1-W4	removed - no Winchester
W5-W7	removed - no floppy
W13-AB	installed - enable no-host diagnostics
W8-W12	removed

3. Install the controller in a suitable Multibus compatible backplane.

4. Power the controller and insure that the supply voltage is within tolerance (Section 1-F).

5. Whenever reset is applied and then removed, the LED on the controller PC board should illuminate for a brief period and then be extinguished.

If the LED fails to come on, or comes on and stays on, an internal controller failure is indicated. Insure that socketed components on the board are fully seated in their sockets and that no pins are bent or broken. If the board continues to fail this test, contact SMS customer support for additional aid.

If the controller board appears to work properly when the test described above is performed the following procedure may be used to test the complete disk system without a host driver program.

1. Insure that the PC board option straps are set correctly for the system in which the controller is to be tested, again except for W1-W13 which should be set as indicated for the System and Drive Test in Section V-B above.

2. Attach drive cables paying special attention to proper termination.

3. Install the controller in a suitable Multibus compatible backplane.

4. Apply power to the controller system and disk drives, checking that all operating voltages are within their tolerances.

5. When the controller is powered up it will begin running the System and Drive Test (Section V-B above). Each cycle of the test will be preceeded by a repetition of the internal self test (Section V-A).

If a fault is detected during one of the executions of the self test the LED on the controller PC board will come on and stay on. If this occurs the fault should be treated as described for the controller only test above. If a fault is detected during the test of one of the drives it will no longer be accessed and the LED will flash. This fact may be recognized by the operator as a drive or cable failure.

Appendix A — Detailed FWD8006 Event Codes

Detailed event codes are status information returned by the FWD8006 describing, in as much detail as the controller resident processors have available, the most recent controller internal event. Typically, this event is the cause of the last command termination. This information may be used to determine the exact nature of a disk error, the specific IOPB field in error, etc. The event code is returned by the Transfer Status function as the last byte (byte 12) in the buffer (Section III-E-2).

For more information on error processing, see also Section III-H. The most significant bit of the event code in the buffer is not part of the code. It is used to indicate that an error occurred which was later recovered by internal controller retry. The 128 event codes may be divided into four groups:

00H - 1FH	Non - Error Events
20H - 3FH	User Error Events
40H - 5FH	System Error Events
60H - 7FH	Controller Error Events

NON-ERROR EVENTS

00H	No Event / No errors
01H - 1FH	Reserved for internal controller events

USER ERROR EVENTS

20H	reserved
21H	Illegal function
22H	Invalid command
23H	Illegal unit
24H	Invalid cylinder number
25H	Invalid head number
26H	Invalid sector number
27H	Illegal bytes per sector
28H	Too many Winchester cylinders
29H	Too many Winchester heads
2AH	Illegal floppy sides
2BH	Illegal encoding
2CH	Illegal format code byte
2DH	No defective/alternate track format on floppy
2EH	Buffer starting address too small
2FH	Buffer starting address + length too large
30H	Can't execute 8089 code
31H	Command issued for a drive which is in use
32H	Command issued for an uninitialized drive
33H	Illegal test number specification
34H	End of media before requested transfer exhausted
35H	Reserved
36H	Sector size not supported by specified encoding
37H	Reserved

35H	Command extensions need to be enabled
36H	non-existent or reserved drive strapping
37H - 3FH	reserved

SYSTEM ERROR EVENTS

40H	Drive not ready
41H	reserved
42H	Deleted Data AM encountered
43H	Data Address Mark mismatch
44H	Write error
45H	Format pass timeout
46H	No Sector ID address marks found on track
47H	No good sector ID found on track
48H	Sector length doesn't match Initialization value
49H	Cylinder number mismatch - positioning error
4AH	Head (Side) number mismatch
4BH	Specified sector not found on track
4CH	Data field CRC/ECC error
4DH	Internal BP timeout error
4EH	Track write timeout
4FH	reserved
50H	Normal track at alternate track location
51H	Defective track at alternate track location
52H	Couldn't recover alternate track address
53H	Illegal alternate track vector
54H	reserved
55H	Winchester data corrected - ECC
56H - 57H	reserved

CONTROLLER/HARDWARE ERROR EVENTS

60H	Default error - EST
61H	Carriage registration timeout
62H	No Winchester seek complete
63H	Immediate seek complete
64H	Winchester seek timeout
65H	reserved
66H	Illegal track type in Winchester sector ID
67H	reserved
68H	8085 failure
69H	Program checksum error
6AH	2Kx8 RAM failure
6BH	8155 RAM failure
6CH	Millisecond clock too slow
6DH	Millisecond clock too fast
6EH	8155 port failure
6FH	BP hung in EST data loop
70H	Computed write ECC error
71H	Pattern error - Last cylinder write/read
72H	reserved
73H	No stable ECC residue
74H - 7FH	reserved