8510/a
Graphics Computer System

Variable Density Disk Drive Controller

Reference Guide

Part No. 50-0010-002 Rev. E

This document describes the use of two types of Floppy Disk Drive Controllers in an 8510/a Graphics Computer System: a Variable Density Disk Drive Controller and a Single Density-only Disk Drive Controller. These modules control direct memory access (DMA) data transfer between system main memory and eight inch floppy diskette drives. Up to four drives, installed in 8510, 8512 and 8515 units, can be controlled. This hardware, in conjunction with appropriate software, supports the following formats: IBM 3740/1 single-sided, single density; 3740/1D single-sided, double density; IBM 3740/2D double-sided, double density. This hardware/software subsystem is completely compatible with original 8510/a Single Density, non-DMA configurations.

TERAK CORPORATION

Scottsdale, Arizona

TERAK CORPORATION believes that the information contained herein is accurate. In no event will TERAK be liable for any losses or damages, whether direct or indirect, resulting from the use of such information, including, without limitation, losses arising from claims of patent, copyright, and trademark infringement. No license is granted hereby for the use of any patent or patent rights of TERAK. TERAK reserves the right to update the information contained herein at any time without further notice.

The information contained herein is proprietary to TERAK CORPORATION and must be treated as confidential. It may not be disclosed to others or used for any purpose without the written consent of TERAK CORPORATION.

8510/a GRAPHICS COMPUTER SYSTEM

VARIABLE DENSITY DISK DRIVE CONTROLLER

REFERENCE GUIDE

FIRST EDITION

COPYRIGHT 1980 BY TERAK CORPORATION

"ALL RIGHTS RESERVED"

TERAK is a trademark of TERAK Corporation. DEC, PDP-11, RT-11 and LSI-11 are trademarks of DIGITAL EQUIPMENT CORPORATION.

CONTENTS

SECTION PAGE

E1. General Description..... E1
E1.1 Diskette Formats & Drive Attributes..... E2
E1.2 Media Considerations..... E3

E2. Controller Description..... E4
E2.1 Processor Interface..... E5
E2.1.1 Floppy Disk Status Register (QBCS)..... E6
E2.1.2 Multi-Purpose Stack (QBMS)..... E6
E2.1.3 DMA Control/Status Register (QBDC)..... E7
E2.1.4 Terminal Count Register (QBTC)..... E7
E2.1.5 Bus Address Register (QBBA)..... E7

E3. Diskette & Track Geometry..... E8

E4. Controller Instruction Set..... E9
E4.1 Byte Descriptions..... E9
E4.2 Bit Descriptions..... E10
E4.3 Instruction Descriptions..... E12
E4.3.1 Read Data Instruction..... E12
E4.3.2 Read Deleted Data Instruction..... E14
E4.3.3 Write Data Instruction..... E15
E4.3.4 Write Deleted Data Instruction..... E16
E4.3.5 Read A Track Instruction..... E17
E4.3.6 Read ID Instruction..... E18
E4.3.7 Format A Track Instruction..... E19
E4.3.8 Scan Instructions..... E20
E4.3.9 Seek Instruction..... E23
E4.3.10 Recalibrate Instruction..... E24
E4.3.11 Sense Drive Status Instruction..... E24
E4.3.12 Sense Interrupt Status Instruction..... E25
E4.3.13 Specify Instruction..... E26
E4.4 Invalid Operation..... E26

E5. Boot ROM..... E33

List Of Tables

Table — Page

E1. ID Information When the CPU Terminates an Instruction..... E27
E2. Transfer Capacity..... E28
E3. Formatting Variables..... E28
E4. SH & SN Status Bits..... E29
E5. Status Register 0 Description..... E29
E6. Status Register 1 Description..... E30
E7. Status Register 2 Description..... E31
E8. Status Register 3 Description..... E32

Program Listings

Name — Page

Variable Density Floppy Disk Controller (FDC) Boot ROM Code.. E34

List Of Figures

Figure Number

E1. Variable Density Disk Controller Q-Bus Interface Registers
E2. QBCS, QBMS Registers
E3. QBDC, QBTC, QBBA Registers
E4. Diskette & Track Geometry

APPENDIX E. Variable Density Disk Drive Controller Reference Guide

E1. General Description

The 8510/a system utilizes eight inch magnetic floppy diskettes for mass storage. This document describes the use of either of two Disk Drive Controllers: a standard, Single Density-only Disk Drive Controller or an optional, Variable Density Disk Drive Controller. The Variable Density Disk Drive Controller, and its appropriate software handler, interfaces to single and dual head drives, and supports the IBM 3740/2D format for double density, modified frequency modulation (MFM) recording, as well as the IBM 3740 single density, frequency modulation (FM) format. As a result, the Variable Density Controller supports the transfer of data to and from diskettes in the following manner: single-sided, single-density (128 bytes per sector); single-sided, double-density (512 bytes per sector); and double-sided, double-density (512 bytes per sector). The Single Density-only Controller has all the attributes of the Variable Density Controller, however, it will only support the IBM 3740/1 single density (FM) format.

The hardware/software subsystem can control up to four drives whose diskettes have combinations of the previously described side and density options; the software handler recognizes the options when the diskette is inserted in a drive.

The Variable Density (and the Single Density-only) Controller provides, at the sector level, DMA data transfer between the main memory in the 8510 unit and up to four diskette drives. The Controller supports parallel-seek operation on all four drives; simultaneous multiple heads-down operation; multi-sector and multi-track data transfer; and diskette formatting for the soft-sectored recording mode. Data transfer rates are 250 K-bits per second for FM encoding and 500 K-bits per second for MFM encoding.

The Controllers also provide the electonics necessary to interface the diskette drives to the 8510/a system bus (identical to the LSI-11 Q-bus), plus record and recover information in single density formats (and double density formats - Variable Density Controller only).

The Controllers also contain a 255 word (16-bit) initial boot ROM. This ROM, and all applicable registers of the Controllers, are accessed via Bank 7 (I/O Page) of the LSI-11 address space.

Controller hardware consists of: a dual-width (8.9 in. W x 5.2 in. H) module, installed in a slot of the 8510 unit backplane, and an External Interface Board (EIB), installed on the rear surface of the 8510 unit, plus, interconnect cables. EIBs are also provided for the 8512 and 8515 units. DIP switches on the EIBs are used to assign a drive as unit 0,1,2, or 3.

Access Times	Single Head Drive	Dual Head Drive
Track to Track:	6 milliseconds	3 milliseconds
Head Settling Times:	8 milliseconds	15 milliseconds
Head Load Times:	35 milliseconds	35 milliseconds

Maximum Error Rates: 1 recoverable error per 10 **9 bits
1 non-recoverable error per 10 **12 bits

These error rates exclude external causes, such as diskette defects or contamination. A non-recoverable error is defined as an error that persists after the error recovery procedure is performed. The error recovery procedure consists of a fixed number (10) of retries to read the record in error.

E1.2 Media Considerations

By means of the information stored on Track 00 of all TERAK-supplied IBM-compatible diskettes, the user can access various formats, on different diskette media, simultaneously; this also allows various drive and media types to be used simultaneously.

Media originally formatted in single density, however, may not necessarily be reliable if re-formatted in double density. Double density formatting requires diskettes that are certified as reliable for double density recording.

E1.1 Diskette Formats & Drive Attributes

The following media formats are used by TERAK:

	Single-sided		Double-sided
	Single-density	Double-density	Double-density
Cylinders (Total)	77	77	77 **
Tracks (Total)	77	77 *	154 **
Index Track(s)	1	1	2
Sectors/Track	26	15	15
Data bytes/Sector	128	512*	512 **
Sectors/Diskette	2002	1166	2332
Data bytes/Diskette	256,256	587,008	1,173,344

* Track 00 is single-density, 128 bytes/sector, 26 sectors.
** Side 0, Cylinder 0, is single-density, 128 bytes/sector, 26 sectors.
** Side 1, Cylinder 0, is double-density, 256 bytes/sector, 26 sectors.

The drive assemblies used (single-head & dual-head) produce the following attributes:

Rotational Speed:	360 RPM +- 2.5 %
Rotational Direction (as would be seen from the label side of a diskette):	CW
Tracks/Inch:	48
Read/Write Track Width:	0.012 in.
Inside Track Radius:	2.029 in.

Recording Density (inside track): 3200 bpi (single density)
6400 bpi (double density, MFM)

Flux Density (inside track): 6400 fci (single density)
6400 fci (double density, MFM)

Encoding Methods: FM (Single Density) MFM (Double Density)

Data Transfer Rates: 250 K-bits/S (Single Density, FM encoding)
500 K-bits/S (Double Density, MFM encoding)

Average Latency: 83 milliseconds

Average Access Time: 260 milliseconds

E2. Controller Description

The Floppy Disk Controller module (FDC) is capable of transferring a minimum of one byte to a maximum of 16,384 bytes, in the Direct Memory Access (DMA) mode. Data anywhere in main memory can be transferred to the sectors of a diskette; conversely, data from any of the diskette sectors can be transferred to any of the locations in main memory. (Bank 7 is normally reserved for peripheral I/O device addresses.) The DMA data transfer is accomplished, with a minimum of processor overhead.

The FDC converts the data bytes from the main memory into a serial bit stream; this data stream is then interleaved with a serial clock bit stream. The resultant clock/data envelope is then recorded on a diskette as a series of flux changes. When a diskette is read, the flux changes are re-converted to a clock/data bit stream. A data recovery circuit separates the clock and data bit streams to recover the data bits. The FDC then converts the serial data bit stream into data bytes for storage in main memory.

The FDC will execute the following 15 instructions:

Read Data	Write Data	Sense Interrupt Status
Read ID	Write Deleted Data	Sense Drive Status
Read Deleted Data	Scan Equal	Seek
Read a Track	Scan High or Equal	Specify
Format a Track	Scan Low or Equal	Recalibrate (Restore to Track 00)

The FDC contains five Q-bus accessible registers, as shown in Figure E-1. The bits of the read-only FDC Status Register (QBCS) indicate data transfer direction, operating mode (DMA, non-DMA), data validity, FDC status and drive status. The Multi-Purpose Stack (QBMS) is used to store bytes in a certain sequence. These bytes specify the parameters of a diskette data transfer (i.e., physical location, data stream length, encoding method, etc.) and the type of operation to be performed.

As shown in Figure E-1, DMA is controlled by 3 Q-bus accessible registers. A 16-bit Bus Address Register (QBBA) is loaded with the starting address that specifies the beginning location, in main memory, for a data transfer. The lower 14 bits of a 16-bit Terminal Count Register (QBTC) are loaded with an ending, or terminal count value, that specifies the length of the data block to be transferred (the upper 2 bits specify read or write operation). When the DMA transfer commences, the starting address is put on the Q-bus and a DATI or DATO(B) operation occurs. The address is then incremented and the read or write operation is repeated. This sequence continues until the number of data transfers equals the value (plus one) loaded in the QBTC register. At that time, the transfer is completed.

The module also contains an 8-bit DMA Control/Status Register (QBDC). This register is used to enable the DMA and Floppy Disks; to enable the DMA interrupt; and to set the data direction (i.e., memory read or write).

E2.1 Processor Interface

The FDC can transfer a single byte (single DMA cycle), or multiple bytes (multiple-DMA cycles; i.e., DMA transfer). The addresses of the Q-bus accessible registers, are as follows:

Q-Bus Address (octal)	Mnemonic	Function
171000	QBCS	Floppy Disk Status Register
171002	QBMS	Multi-Purpose Stack
171012	QBDC	DMA Control/Status Register
171014	QBTC	Terminal Count Register
171016	QBBA	Bus Address Register
173000-173774		Initial bootstrap ROM

The FDC is capable of executing 15 instructions, as listed in Part E2 of this document. Descriptions of these instructions are contained in Part E4. Each instruction requires multiple-byte transfers to set up, execute, and complete an instruction. As a convenience, an instruction may be considered as consisting of three phases:

Command Phase: The FDC receives all information, required to perform a particular operation.

Execution Phase: The FDC performs the operation.

Results Phase: After completion of the operation, status and other housekeeping information is made available.

During the Execution phase, data bytes are transferred between the FDC and a drive.

The formats of QBCS and QBMS are shown in Figure E-2. One register monitors the FDC status, the other register location is actually a stack, used to store and retrieve data during the Command and Results phases of an instruction.

The formats of QBDC, QBTC and QBBA are shown in Figure E-3. These registers are used for DMA operations.

E2.1.1 Floppy Disk Status Register (QBCS)

During the Command and Result phases of an instruction, QBCS must be read before each byte is written into, or read from, the Multi-Purpose Stack (QBMS). This alternating QBCS-read/QBMS-write (during the Command phase) and QBCS-read/QBMS-read (during the Result phase) must occur before the FDC will execute the current instruction and accept a new instruction. During the Execution phase, QBCS need not be read.

QBCS is read from the lower byte of a Q-bus word at location 171000, octal.

QBCS bits 0 through 3 reflect drive status; that is, whether or not a drive is busy (in Seek mode, to move the read/write head from track to track).

QBCS bit 4 indicates whether the FDC is busy with a read or a write operation.

QBCS bit 5 indicates the operating mode. This bit is set only during the Execution phase in the non-DMA mode; when bit 5 resets, the Execution phase has ended.

QBCS bit 6 indicates data transfer direction. A "0" means that data is being transferred from the Q-bus to the FDC; a is being transfe

QBCS bit 7, when set, indicates that QBMS is ready to send or receive data.

E2.1.2 Multi-Purpose Stack (QBMS)

QBMS is a stack (of 8-bit registers) that is pushed and popped, respectively, during the Command and Result phases of an instruction. Each instruction requires that a specific sequence of bytes be loaded into, and read from, the stack. As shown in Figure E-2, a 1 or 2 byte command code (that specifies a particular instruction) must be loaded first, and then followed by 0 to 7 configuration bytes. After instruction execution, status bytes (whose bits specify any errors that had occurred during execution) and sector ID bytes (that either duplicate the sector ID bytes that were loaded during the Command phase or have been changed because of instruction termination) are returned to the stack. QBMS must be read (and interpreted) to proceed to a new instruction. The byte codes for QBMS are described in Part E3.

QBMS is written to, and read from, the lower byte of a Q-bus word at location 171002, octal.

E2.1.3 DMA Control/Status Register (QBDC)

QBDC must be programmed after QBTC and QBBA registers have been programmed. QBDC is cleared (reset) whenever a Reset instruction is executed, when the system is reset (boot switch depressed), or when power is cycled. Bit 0 of QBDC must be set to enable the DMA. When read, during a DMA cycle, QBDC bit 0 will be clear until the cycle is completed, at which time it will be set. When read after completion of a DMA cycle, QBDC bit 0 will be reset.

When set, during register programming, QBDC bit 3 will enable the Controller interrupt input upon completion of the DMA transfer.

QBDC bit 4 determines data direction during a DMA cycle; if programmed to "0", data will be transferred from memory to diskette.

QBDC bits 1, 2, 5, 6 and 7 must be programmed to the states shown in the QBDC bit map of Figure E-3 to prevent spurious operations.

QBDC is read from, and written to, the lower byte of a Q-bus word at location 171012, octal, and consumes a single DATI or DATOB bus cycle.

E2.1.4 Terminal Count Register (QBTC)

QBTC must be programmed with a number that is one less than the desired number of byte transfers. For instance, if "0" is loaded, a single DMA cycle would occur. A maximum of 16,384 bytes are allowed per DMA transfer.

The lower 14 bits of QBTC are used for the terminal count number; the upper 2 bits specify the data direction during a DMA operation. Only two combinatorial states are allowed for bits 6 & 7, as shown in the QBTC bit map of Figure E-3.

E2.1.5 Bus Address Register (QBBA)

QBBA must be programmed with the address of the first memory byte location to be accessed during the DMA operation. During the operation, the Controller will increment, (by one), the address put on the bus, until the desired number of byte locations (specified by the value loaded into the lower 14 bits of QBTC) have been accessed.

QBBA can be loaded with any starting address, however, Bank 7 (160000 to 177777, octal), of the LSI-11 address space, is generally reserved for I/O device addresses.

Note: QBTC and QBBA are both 16-bit registers that are written to the lower byte of a Q-bus word at respective locations 171014 and 171016, octal. Two DATOB bus cycles are required to load either register.

E3. Diskette & Track Geometry

The diskettes used with the Variable Density Controller can either be single or double density encoded, and, with the proper drive, can be single or double sided. Figure E-4 identifies the general characteristics of all diskettes used with the 8510/a system.

There are 77 concentric tracks on one surface (side) of a diskette. For double sided diskettes, pairs of tracks , on opposite sides, are termed cylinders. In comparison to single sided diskettes, data transfers to/from a double sided are faster, because the opposing tracks of a cylinder can be accessed, sequentially, without moving the read/write heads.

During diskette formatting, the tracks will be divided into 26 segments, or sectors, for single density recording, or 15 sectors, for double density recording. For all diskettes used with the 8510/a system, track 0, side 0, will always be FM encoded, with 26 sectors and 128 bytes/sector. This track specifies the recording density to be used for the other tracks; the Controller uses track 0, side 0 information (plus a drive status line signal) to determine the format for the diskette. The index hole on a single sided diskette is at a different location (with respect to the slot in the diskette's protective cover) when compared with a double sided diskette. The Controller uses the index hole position to differentiate between a single sided and a double sided diskette.

The gaps and fields of a sector are shown in Figure E-4. During a Format operation, these patterns are created by the system after it has detected the index hole. The system will format the track for single or double density recording. A track will be formatted in a single rotation of the diskette, beginning and ending with the index hole detection. As the track passes the read/write head(s), the R values (Sector No.s), written into the ID Field, need not be in sequential order. The C (Cylinder No.), H (Head No.) and N (Data Bytes/Sector No.) values are a function of the head position, diskette side and required density and cannot be randomly ordered. To read or write to any data field on a diskette, the user must load matching C,H,R,N values (Sector ID) into QBMS, prior to the execution of an instruction.

E4. Controller Instruction Set

The FDC has a repertoire of 15 instructions, as follows:

- Read Data
- Read ID
- Write Data
- Format a Track
- Recalibrate (Restore to Track 0)
- Seek
- Specify
- Sense Interrupt Status
- Sense Drive Status
- Read a Track
- Read Deleted Data
- Write Deleted Data
- Scan Equal
- Scan High or Equal
- Scan Low or Equal

The FDC requires multiple-byte transfers to execute the instructions. These bytes (and certain bits of some bytes) are described as follows.

E4.1 Byte Descriptions

C; Cylinder No. : The current/selected cylinder number (0 to 76)

H; Head No. : The selected read/write head (0 = Side 0; 1 = Side 1)

R; Record No. : The sector (record) number (1 to 26, single density; 1 to 15, double density)

N; Number : The number of data bytes/sector (0 for single density, 2 for double density) written into a sector.

EOT; End Of Track: The final sector number (15 or 26) of a track.

GPL; Gap Length: The Post-ID and Post-Data gap lengths; that is, the inter-record gaps. GPL is is equated to a binary magnitude; see Table E3 for recommended values.

DTL; Data Length: The number of data bytes written into or read from a sector when N = 0. If DTL defines a data length smaller than that which was defined by N when N was not zero, the read or write transfer will cease when the DTL value is reached. The FDC will continue to read or write (with zeroes) the remainder of the data field to perform the CRC function. When N is not zero, DTL must be set to all "1's" (i.e., FF, hex or 377, octal).

STP; Scan Track, Processor: During a Scan (High, Low, or Equal) instruction,the data bytes in a sector are compared with data bytes from the Q-bus. If STP = 1, the data bytes from contiguous sectors are compared. If STP =2, data bytes from alternate sectors are compared.

STO; Status 0:
ST1; Status 1:
ST2; Status 2:
ST3; Status 3: These are four registers that are read at the beginning of the Result phase of an instruction. The registers that are read depend on the particular instruction. The bits of the registers indicate what errors (if any) occurred during the instruction execution, and whether the instruction terminated normally or abnormally. See Tables E5 through E8 for a description of these registers.

SC; Sectors/Cylinder: During a Formatting instruction, SC determines the number of sectors to be written on a (track); (15 for double density; 26 for single density).

D; Data: During a Formatting instruction, D is the byte pattern that is written into the data fields to define the data field locations on a track. D can be any pattern.

PCN; Present Cylinder No. : PCN is read during the Result phase of a Sense Interrupt Status instruction and represents the present position of the read/write head(s) (i.e., at Cylinder 0 through 76).

NCN; New Cylinder No. : NCN is the cylinder number to which the read/write head(s) will move during a Seek instruction. NCN is loaded during the Command phase of Seek with value 0 through 76.

E4.2 Bit Descriptions

MT; Multi-Track: A command code bit in the Read, Write and Scan instuctions. When MT = 1, data will be transferred to/from both tracks of a cylinder, starting at Sector 1, Side 0 and completing at Sector L (last sector) of the opposing track on Side 1, of a double sided diskette.

MF; FM or MFM Mode: A command code bit in the Read, Write, Scan and Format instructions. When MF = 0, the FM mode is selected; when MF = 1, the MFM mode is selected.

SK; Skip: A command code bit in the Read (except Read ID) and Scan instructions. When executing a Read Data or a Scan instruction, and SK = 1, the FDC will skip over, (not read), a sector that has a Deleted Data Address Mark (DDAM). The CRC bits of a skipped sector are not checked. At this time, if a Scan instruction is executing, the Control Mark (CM) bit of the ST 2 register is set, to indicate that a DDAM had been encountered.

When executing a Read Deleted Data instruction, and SK = 1, the FDC will skip over a sector that has a Data Address Mark.

When executing a Read a Track instruction, the SK bit is ignored.

When SK = 0, the FDC will read the field that would have been skipped over, and then terminate the instruction. At this time, the CM bit of ST 2 will be set.

HD; Head : A command code bit that indicates the selected head (0 = Head 0; 1 = Head 1). This bit must have the same value as H, the Head No. byte.

HLT; Head Load Time : Seven bits of a command code of the Specify instruction. HLT controls the head load time of a drive (defined as the interval between the assertion of a head load signal and the commencement of a read/write operation). HLT is programmable in 2 millisecond increments over the range 2 mS to 254 mS; it is equated to a binary magnitude: 2 mS = 01; 254 mS = FE,hex = 376,octal.

HUT; Head Unload Time: Four bits of a command code of the Specify instruction, programmable in 16 millisecond increments over the range 16 mS to 240 mS. (16 mS = 01; 240 mS = 0F,hex = 17,octal). HUT starts at the end of a R/W execution phase. Note: Hardware causes a head to remain loaded for 600 mS after the end of a R/W instruction execution phase.

SRT; Step Rate Time: Four bits of a command code of the Specify instruction. SRT controls the head stepping rate (defined as the time it takes to move from track to adjacent track). SRT is programmable in 1 millisecond increments; it is equated to a binary magnitude: 1 mS = F,hex = 17,octal; 16 mS = 0.

ND; Non-DMA Mode: A single bit of a command code of the Specify instruction. ND controls the operating mode of the Controller. When ND = 1, the non-DMA mode is selected; when ND = 0, the DMA mode is selected.

US1,US0; Unit Select: Two bits of a command code of every instruction (except Specify) that are used to select a drive.

US1	US0	Drive Selected
0	0	0
0	1	1
1	0	2
1	1	3

E4.3 Instruction Descriptions

The following listing describes the instructions and the order in which the byte and bit parameters of Parts E4.1 and E4.2 must be written to and read from the Multi-Purpose Stack (QBMS) to execute the instructions. (Note: An 'x' bit is a 'don't care' state.)

E4.3.1 Read Data Instruction

Nine bytes are written to QBMS to perform the Read Data instruction. After the instruction has been issued, the FDC loads the head (if it is in the unloaded state), waits the specified head settling time (defined in the Specify instruction) and begins reading ID Address Marks & ID Fields. When the current sector number (R), stored in an ID Register compares with the sector number read off the diskette track, the FDC will output data (from the data field), byte-by-byte, to the Q-bus.

After the current-sector data field is read, the Sector Number is incremented, by 1, and the data from the next sector is read and outputted to the Q-bus. This continuous-read is a 'Multi-Sector Read Operation'. The Read Data instruction may be terminated by the FDC reaching DMA Terminal Count (TC), (from QBTC). When the FDC reaches TC, the FDC will stop outputting data to the Q-bus, but will continue to read data from the current sector, check the CRC bytes, and then terminate the Read Data instruction at the end of the current sector.

The 'Multi-Track' function (MT bit) allows the FDC to read data from both tracks of a cylinder. For any cylinder, data will be transferred starting at Sector 1, Side 0 and completing at Sector L (last sector) of Side 1.

When N = 0, then DTL defines the data length which the FDC must treat as a sector. If DTL is smaller than the actual data length in a sector, the data beyond DTL is not sent to the Q-bus. The FDC will read the remaining sector data to perform the CRC check, and , depending on the manner of instruction termination, may perform a "Multi-Sector' Read operation. When N is non-zero, DTL has no significance, and must be set to FF,hex or 377,octal.

At the completion of the Read Data (or Write Data) instruction, a head is not unloaded until a 600 mS interval has elapsed. If the CPU issues another instruction before the head unloads, then the delay caused by the Head Load Time (HLT) interval may be eliminated for subsequent read/write operations.

If the Controller detects the diskette Index Hole twice, without finding the correct sector (as defined by R), then the FDC will set the ND flag in ST1, and terminate the instruction. ST0 bits 7 & 6, (Interrupt Code bits), will be toggled to 0 and 1, respectively, to flag an abnormal termination.

After reading the ID and Data Fields in each sector, the FDC checks the CRC bytes. If an incorrect CRC in the ID Field is detected, the DE (Data Error) flag in ST1 will be set. If an incorrect CRC is detected in the Data Field, the DD (Data Error in Data Field) flag in ST2 and the DE flag of ST1 will both be set. In both cases, the Read Data instruction will be terminated.

If the FDC reads a Deleted Data Address Mark (DDAM) off the diskette, and SK = 0, the Control Mark (CM) flag in ST2 will be set, and the Read Data instruction will be terminated after the sector data is read. If SK = 1, the FDC will skip over the sector with the Deleted Data Address Mark and read the next sector. The CRC bytes in the deleted data field are not checked when SK = 1.

During data transfers between the FDC and the Q-bus, the FDC must be serviced every 27 microseconds in the FM mode, and every 13 microseconds in the MFM mode, or the FDC will set the Over Run (OR) flag in ST1, and terminate the Read Data instruction.

If a Read (or a Write) instruction is terminated, the Sector ID information, during the Result phase, will be dependent upon the state of the MT bit and EOT byte loaded during the Command phase. The Sector ID information, (C, H, R, N), at termination, is shown in Table E1. The amount of data that can be transferred depends upon MT (Multi-Track), MF (Mode), and N (Number of Bytes/Sector). Table E2 lists the transfer capacity.

READ DATA

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	MT	MF	SK	0	0	1	1	0	Command codes
	W	x	x	x	x	x	HD	US1	US0	
	W	----------C----------								Sector ID
	W	----------H----------								codes prior
	W	----------R----------								to instruction
	W	----------N----------								execution.
	W	----------EOT----------								
	W	----------GPL----------								
	W	----------DTL----------								
Execution		Data transfer from a drive to the Q-bus.								
Result	R	----------ST0----------								Status infor-
	R	----------ST1----------								mation after
	R	----------ST2----------								execution.
	R	----------C----------								Sector ID
	R	----------H----------								codes after
	R	----------R----------								instruction
	R	----------N----------								execution.

E4.3.2 Read Deleted Data Instruction

This instruction is the same as the Read Data instruction, except that when the FDC detects a Data Address Mark, at the beginning of a Data Field (and SK = 0), it will read all the data in the sector, set the CM (Control Mark) flag of ST2, and terminate the instruction. If SK = 1, then the FDC skips over the sector with the Data Address Mark and reads the next sector.

READ DELETED DATA

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	MT	MF	SK	0	1	1	0	0	Command codes
	W	x	x	x	x	x	HD	US1	US0	
	W	----------C----------								Sector ID
	W	----------H----------								codes prior
	W	----------R----------								to instruction
	W	----------N----------								execution.
	W	----------EOT----------								
	W	----------GPL----------								
	W	----------DTL----------								
Execution		Data transfer from a drive to the Q-bus.								
Result	R	----------ST0----------								Status infor-
	R	----------ST1----------								mation after
	R	----------ST2----------								execution.
	R	----------C----------								Sector ID
	R	----------H----------								codes after
	R	----------R----------								instruction
	R	----------N----------								execution.

E4.3.3 Write Data Instruction

Nine bytes are written to the FDC Data Register to perform the Write Data instruction. After the instruction has been issued, the FDC loads the head (if it is in the unloaded state), waits the specified head settling time (defined in the Specify instruction), and begins reading ID fields. When the current sector number (R), stored in an ID Register, compares with the sector number read off the diskette track, the FDC will transfer data from the Q-bus to the diskette drive.

After writing data into the current sector, the Sector number is incremented, by 1, and the next data field is written onto the track. The FDC continues this 'Multi-Sector Write Operation' until a DMA Terminal Count (TC) is reached (from QBTC). When the DMA TC occurs, the FDC continues to write into the current sector; if the DMA TC is reached while a data field is being written, the FDC will write zeroes into the remainder of the field.

The FDC reads the ID field of each sector and checks the CRC bytes. If an incorrect CRC is detected, the FDC will set the Data Error (DE) flag of ST1, and terminate the Write Data instruction. If the FDC tries to access a sector beyond the final sector (as configured by SC during a Format a Track instruction), of a cylinder, the EN (End of Cylinder) flag bit, of ST1, will be set.

In the Write Data instruction, data transfers between the FDC and the Q-bus must occur every 31 microseconds in the FM mode, and every 15 microseconds in the MFM mode, or the FDC will set the Over Run (OR) flag in ST1, and terminate the instruction. ST0 bits 7 & 6 (Interrupt Code bits) will also be toggled to 0 and 1, respectively, to flag an abnormal termination.

The Write Data instruction operates in much the same manner as the Read Data instruction. The following items are the same:

1. Sector ID Information when the CPU terminates an instruction. (Refer to Table E1.)

2. Transfer Capacity (refer to Table E2).

3. Definition of DTL when N is and is not zero.

4. Head Load and Unload Time intervals.

5. ND (No Data) flag bit of ST1.

WRITE DATA

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	MT	MF	0	0	0	1	0	1	Command codes
	W	x	x	x	x	x	HD	US1	US0	
	W	----------C----------								Sector ID
	W	----------H----------								codes prior
	W	----------R----------								to instruction
	W	----------N----------								execution.
	W	----------EOT----------								
	W	----------GPL----------								
	W	----------DTL----------								
Execution		Data transfer from the Q-bus to a drive.								
Result	R	----------ST0----------								Status infor-
	R	----------ST1----------								mation after
	R	----------ST2----------								execution.
	R	----------C----------								Sector ID
	R	----------H----------								codes after
	R	----------R----------								instruction
	R	----------N----------								execution.

E4.3.4 Write Deleted Data Instruction

This instruction is the same as the Write Data instruction, except that a Deleted Data Address Mark is written at the beginning of the Data Field, instead of the normal Data Address Mark.

WRITE DELETED DATA

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	MT	MF	0	0	1	0	0	1	Command codes
	W	x	x	x	x	x	HD	US1	US0	
	W	----------C----------								Sector ID
	W	----------H----------								codes prior
	W	----------R----------								to instruction
	W	----------N----------								execution.
	W	----------EOT----------								
	W	----------GPL----------								
	W	----------DTL----------								
Execution		Data transfer from the Q-bus to a drive.								
Result	R	----------ST0----------								Status infor-
	R	----------ST1----------								mation after
	R	----------ST2----------								execution.
	R	----------C----------								Sector ID
	R	----------H----------								codes after
	R	----------R----------								instruction
	R	----------N----------								execution.

E4.3.5 Read A Track Instruction

This instruction is similar to the Read Data instruction, except that this is a continuous read operation, where the entire data field, from each of the sectors, is read. After the Index Hole is detected, the FDC starts reading all the data fields on the track, as continuous blocks of data. If the FDC finds an error in the ID or Data CRC bytes, it continues to read data from the track. The FDC compares the ID information read from each sector with a value stored in an ID Register; if the values do not match, the ND (No Data) flag bit of ST1 will be set. Multi-track and skip operations are not allowed with the Read A Track instruction.

This instruction terminates when the EOT number of sectors have been reached. If the FDC does not find an ID Address Mark on the diskette after the Index Hole has been detected, for the second time, then the MA (Missing Address Mark) flag bit in ST1 will be set, and the instruction will be terminated. ST0 bits 7 &6 (Interrupt Code bits) will also be toggled to 0 & 1, respectively.

READ A TRACK

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	0	MF	SK	0	0	0	1	0	Command codes
	W	x	x	x	x	x	HD	US1	US0	
	W	----------C----------								Sector ID
	W	----------H----------								codes prior
	W	----------R----------								to instruction
	W	----------N----------								execution.
	W	----------EOT----------								
	W	----------GPL----------								
	W	----------DTL----------								
Execution		Data transfer from a drive to the Q-bus. The FDC reads all data fields from the Index Hole to EOT.								
Result	R	----------ST0----------								Status infor-
	R	----------ST1----------								mation after
	R	----------ST2----------								execution.
	R	----------C----------								Sector ID
	R	----------H----------								codes after
	R	----------R----------								instruction
	R	----------N----------								execution.

E4.3.6 Read ID Instruction

The Read ID instruction is used to give the present position of the read/write head. The FDC stores the values from the first ID Field it is able to read. If no proper ID Address Mark is found on the diskette, before the Index Hole is encountered for the second time, then the MA (Missing Address Mark) flag bit in ST1 is set; if no data is found, the ND (No Data) flag bit in ST1 is also set. The instruction is then terminated, and ST0 bits 7 & 6 (Interrupt Code bits), are also toggled to 0 &1, respectively, to flag an abnormal termination.

READ ID

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	0	MF	SK	0	0	0	1	0	Command codes
	W	x	x	x	x	x	HD	US1	US0	
Execution		The first correct ID information read from the cylinder is stored in the FDC Data Register.								
Result	R	ST0								Status information after execution.
	R	ST1								
	R	ST2								
	R	C								Sector ID codes after instruction execution.
	R	H								
	R	R								
	R	N								

E4.3.7 Format A Track Instruction

The Format instruction allows an entire track to be formatted. After the Index Hole is detected, format data is written on the track: Gaps, Address Marks, ID Fields and Data Patterns, for single or double density encoding, are stored. The particular format to be written is controlled by the values programmed into: N (no. of bytes/sector), SC (sectors/cylinder), GPL (gap length of Post-ID & Post-Data gaps) and D (Data Pattern) byte. The Data Field is filled with the D byte, for definition. The ID Field for each sector is supplied from the Q-bus; that is, four requests, per sector, are made by the FDC for: C (Cylinder No.), H (Head No.), R (Sector No.) and N (No. of bytes/sector). This allows the diskette to be formatted with non-sequential sector numbers, if so desired.

After formatting each sector, the Q-bus is referenced for new values for C, H, R and N for each sector. The contents of the FDC R Register are incremented by one, after each sector is formatted, so that the R Register contains a value of R when it is read, during the Result phase. This incrementing & formatting continues until the FDC detects the Index Hole for the second time, and terminates the instruction. The effects that N, SC and GPL have on formatting are shown in Table E3.

FORMAT A TRACK

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	0	MF	0	0	1	1	0	1	Command codes
	W	x	x	x	x	x	HD	US1	US0	
	W	----------N----------								Bytes/Sector
	W	----------SC----------								Sectors/Track
	W	----------GPL----------								Gap 3
	W	----------D----------								Filler Byte
Execution		The FDC formats an entire track.								
Result	R	----------ST0----------								Status infor-
	R	----------ST1----------								mation after
	R	----------ST2----------								execution.
	R	----------C----------								In this case,
	R	----------H----------								sector ID
	R	----------R----------								has no
	R	----------N----------								meaning.

E4.3.8 Scan Instructions

The Scan instructions allow data that is being read from a diskette to be compared against data that is on the Q-bus (in the DMA mode). The FDC compares the data on a byte-by-byte basis, and looks for a sector that meets the requirement of the Scan instruction (i.e. Data Equal; Data Equal or Low; Data Equal or High). One's complement arithmetic is used for the comparison (FF, hex = largest number; 00 = smallest number). After a whole sector of data is compared, if the conditions are not met, the sector number (R) is incremented by STP (R + STP returned to R), and the Scan operation is continued. The operation continues until one of the following events occur: the conditions for Scan (Equal, Low or High) are met, the last sector on the track is reached (EOT), or the DMA TC (Terminal Count) is reached.

If the conditions for Scan are met, the FDC will set the SH (Scan Equal Hit) flag bit of ST2, and terminate the instruction. If the conditions for Scan are not met, between the starting sector (as specified by R) and the last sector (as specified by EOT), then the FDC will set the SN (Scan Not Satisfied) flag bit of ST2, and terminate the instruction. If the DMA TC is reached during the Scan operation, the FDC will complete the comparison of the current byte, and then terminate the instruction. Table E4 lists the states of SH and SN under various conditions of Scan.

If the FDC encounters a Deleted Data Address Mark on one of the sectors, and SK = 0, then this sector will be regarded as the last sector on the cylinder; the FDC will set the CM (Control Mark) flag bit of ST2, and terminate the instruction. If SK = 1 when the Deleted Data Address Mark is encountered, the FDC will skip over that sector and read the next sector; the FDC will also set the CM bit of ST2 to indicate that a Deleted Data Address Mark had been encountered.

When either the STP (scan contiguous sectors = 01; scan alternate sectors = 02) or the MT (access both tracks of a cylinder = 1) variables are programmed, it is necessary to remember that the last sector on the track must be read. For example, if STP = 02, and MT = 0, and if the sectors are numbered sequentially 1 through 26, and the Scan instruction is started at Sector 21, then the following will occur: Sectors 21, 23 and 25 will be read, Sector 26, (the last sector), will be skipped and the Index Hole will be encountered, thereby terminating the instruction, abnormally. If EOT had been set to 25, (instead of 26), or if the Scan had started at Sector 20, instead of 21, the instruction would have terminated normally.

During a Scan instruction, data is transmitted from the Q-bus and the diskette and compared in the FDC. The data bytes must arrive at the FDC within 27 microseconds (for FM mode) or 13 microseconds (for MFM mode), or the OR (Over Run) flag bit of ST1 will be set and the instruction will terminate abnormally (the Interrupt Code bits 7 & 6, of ST0 will be toggled to 0 &1, respectively).

SCAN EQUAL

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	MT	MF	SK	1	0	0	0	1	Command codes
	W	x	x	x	x	x	HD	US1	US0	
	W	C								Sector ID
	W	H								codes prior
	W	R								to instruction
	W	N								execution.
	W	EOT								
	W	GPL								
	W	DTL								
Execution		Drive and Q-bus data is compared.								
Result	R	ST0								Status infor-
	R	ST1								mation after
	R	ST2								execution.
	R	C								Sector ID
	R	H								codes after
	R	R								instruction
	R	N								execution.

SCAN LOW OR EQUAL

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	MT	MF	SK	1	1	0	0	1	Command codes
	W	x	x	x	x	x	HD	US1	US0	
	W	C								Sector ID
	W	H								codes prior
	W	R								to instruction
	W	N								execution.
	W	EOT								
	W	GPL								
	W	DTL								
Execution		Drive and Q-bus data is compared.								
Result	R	ST0								Status infor-
	R	ST1								mation after
	R	ST2								execution.
	R	C								Sector ID
	R	H								codes after
	R	R								instruction
	R	N								execution.

SCAN HIGH OR EQUAL

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	MT	MF	SK	1	1	1	0	1	Command codes
	W	x	x	x	x	x	HD	US1	US0	
	W	----------C----------								Sector ID
	W	----------H----------								codes prior
	W	----------R----------								to instruction
	W	----------N----------								execution.
	W	----------EOT----------								
	W	----------GPL----------								
	W	----------DTL----------								
Execution		Drive and Q-bus data is compared.								
Result	R	----------ST0----------								Status infor-
	R	----------ST1----------								mation after
	R	----------ST2----------								execution.
	R	----------C----------								Sector ID
	R	----------H----------								codes after
	R	----------R----------								instruction
	R	----------N----------								execution.

E4.3.9 Seek Instruction

The read/write heads of the drives are moved from cylinder to cylinder under control of the Seek instruction. The FDC compares the PCN (Present Cylinder Number) to the NCN (New Cylinder Number), and if there is a difference, steps to a new cylinder.

The rate at which steps occur is controlled by SRT (Stepping Rate Time), in the Specify instruction. After each step, NCN is compared to PCN; when NCN = PCN, the SE (Seek End) flag bit of ST0 will be set, and the instruction will be terminated.

During the Command phase of Seek, the FDC is in the FDC Busy state; during the Execution phase, it is in the Non-Busy state. Another Seek instruction can be issued while the FDC is in the Non-Busy state; in this manner, parallel Seek operations can be performed on up to 4 drives, simultaneously.

If a drive is in a Not-Ready state at the beginning of the Seek Execution phase, or during the Seek operation, the NR (Not Ready) flag bit of ST0 will be set, and the instruction will be terminated. The Interrupt Code bits, 7 & 6, of ST0, will be toggled to 0 & 1, respectively.

SEEK

Phase	R/W	FDC Data Register Bits								Remarks
		7	6	5	4	3	2	1	0	
Command	W	0	0	0	0	1	1	1	1	Command codes
	W	x	x	x	x	x	HD	US1	US0	
	W	----------				NCN			----------	
Execution		Head is positioned over the proper cylinder on the diskette.								

Note: Because Seek does not have a Result phase, a Sense Interrupt Status instruction must always be performed after a Seek instruction, to effectively terminate Seek.

E4.3.10 Recalibrate Instruction

The function of this instruction is to retract the read/write head of a drive to the Track 00 position. The FDC clears the contents of the PCN (Present Cylinder Number) counter, and checks the status of a Track 00 signal from the drive. As long as this Track 00 signal is low, Steps are issued to move the head. When the Track 00 signal goes high, (indicating that the head is at Track 00), the SE (Seek End) flag bit in ST 0 will be set and the instruction terminated.

If the Track 00 signal is still low after 77 Steps have been issued, the FDC will set the SE and EC (Equipment Check) flag bits of ST0, and terminate the instruction. The Interrupt Code bits of ST0, bits 7 & 6, will be toggled to 0 & 1, respectively.

The ability to do overlapping Recalibrate instructions to multiple drives, and the loss of a drive Ready signal, (as described in the Seek instruction), also applies to the Recalibrate instruction.

RECALIBRATE

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	0	0	0	0	0	1	1	1	Command codes
	W	x	x	x	x	x	0	US1	US0	
Execution		Head retracted to Track 00.								

Note: Because Recalibrate does not have a Result phase, a Sense Interrupt Status instruction must be performed after a Recalibrate instruction to effectively terminate Recalibrate.

E4.3.11 Sense Drive Status Instruction

This instruction can be used to obtain the status of the drives. ST3 contains the drive Status information.

SENSE DRIVE STATUS

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	0	0	0	0	0	1	0	0	Command codes
	W	x	x	x	x	x	HD	US1	US0	
Result	R	----------ST3----------								Drive status

E4.3.12 Sense Interrupt Status Instruction

An interrupt signal is generated, by the FDC, for one of the following reasons:

1. Upon entering the Result Phase of the following instructions:

   A. Read Data
   B. Read a Track
   C. Read ID
   D. Read Deleted Data
   E. Write Data
   F. Format a Track
   G. Write Deleted Data
   H. Scan (High, Low or Equal)

2. A drive changes its Ready state.
3. The end of a Seek or Recalibrate instruction.

Interrupts caused by 1 occur during normal command operations and are easily discerned. Interrupts caused by 2 & 3, however, can be uniquely identified by use of the Sense Interrupt instruction. When this instruction is issued, the cause of the interrupt will be identified, via bits 5, 6 and 7 of ST0, as follows:

Bit 7 (Interrupt Code)	Bit 6	Bit 5 (Seek End)	Cause of Interrupt
1	1	0	Ready line changed state, either polarity.
0	0	1	Normal termination of Seek or Recalibrate.
0	1	1	Abnormal termination of Seek or Recalibrate.

Note: Neither the Seek or the Recalibrate instructions have a Result phase; therefore, it is mandatory that the Sense Interrupt Status instruction be used, immediately afterwards, to effectively terminate Seek or Recalibrate, and to provide head position verification.

SENSE INTERRUPT STATUS

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	0	0	0	0	1	0	0	0	Command code
Result	R	ST0								FDC Status at the end of seek-operation.
	R	PCN								

E4.3.13 Specify Instruction

The Specify instruction sets the initial values of three timers in the FDC: HLT (Head Load Time), HUT (Head Unload Time), and SRT (Step Rate Time). This instruction also determines the Controller operating mode via the ND (Non-DMA) bit. These time and control variables are described in Parts E4.1 & E4.2 of this document.

SPECIFY

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	0	0	0	0	0	0	1	1	Command codes
	W	<------SRT------>				<-----HUT----->				
	W	<-------------HLT------------->							ND	

E4.4 Invalid Operation

If an invalid instruction (one not described in Part E4.3) is sent to the FDC, the FDC will terminate the instruction after Interrupt Code bits 7 & 6, of ST0, are toggled to 0 & 1, respectively. Bit 7 (QBMS Status) and bit 6 (Data Transfer Direction) of QBMS will both be set, indicating that the FDC is in a Result phase and that the contents of ST0 must be read. When ST0 is read, it will be found to contain 80 (hex), indicating that an invalid instruction was issued.

A Sense Interrupt instruction must be sent after a Seek or Recalibrate instruction, or else the FDC will interpret the next instruction to be an invalid one.

INVALID

Phase	R/W	FDC Data Register Bits 7	6	5	4	3	2	1	0	Remarks
Command	W	----------Invalid Codes----------								
Result	R	1	0	0	0	0	0	0	0	ST0 = 80,hex = 200,octal

MT	EOT	Final Sector Transferred	ID Info @ Result Phase			
			C	H	R	N
0	32	Sector 1 to 25 @ Side 0	NC	NC	R+1	NC
0	17	Sector 1 to 14 @ Side 0	NC	NC	R+1	NC
0	10	Sector 1 to 7 @ Side 0	NC	NC	R+1	NC
0	32	Sector 26, @ Side 0	C+1	NC	R=01	NC
0	17	Sector 15, @ Side 0	C+1	NC	R=01	NC
0	10	Sector 8, @ Side 0	C+1	NC	R=01	NC
0	32	Sector 1 to 25 @ Side 1	NC	NC	R+1	NC
0	17	Sector 1 to 14 @ Side 1	NC	NC	R+1	NC
0	10	Sector 1 to 7 @ Side 1	NC	NC	R+1	NC
0	32	Sector 26, @ Side 1	C+1	NC	R=01	NC
0	17	Sector 15, @ Side 1	C+1	NC	R=01	NC
0	10	Sector 8, @ Side 1	C+1	NC	R=01	NC
1	32	Sector 1 to 25 @ Side 0	NC	NC	R+1	NC
1	17	Sector 1 to 14 @ Side 0	NC	NC	R+1	NC
1	10	Sector 1 to 7 @ Side 0	NC	NC	R+1	NC
1	32	Sector 26, @ Side 0	NC	LSB	R=01	NC
1	17	Sector 15, @ Side 0	NC	LSB	R=01	NC
1	10	Sector 8, @ Side 0	NC	LSB	R=01	NC
1	32	Sector 1 to 25 @ Side 1	NC	NC	R+1	NC
1	17	Sector 1 to 14 @ Side 1	NC	NC	R+1	NC
1	10	Sector 1 to 7 @ Side 1	NC	NC	R+1	NC
1	32	Sector 26, @ Side 1	C+1	LSB	R=01	NC
1	17	Sector 15, @ Side 1	C+1	LSB	R=01	NC
1	10	Sector 8, @ Side 1	C+1	LSB	R=01	NC

Note 1: NC = No Change; the same value as was loaded during the Command Phase.

Note 2: LSB (Least Significant Bit); the LSB of H is complemented.

Note 3: EOT is listed in octal; Sector is listed in decimal.

TABLE E1. ID Information When The CPU Terminates An Instruction

MT	MF	N	Maximum Transfer Capacity (Bytes/Sector)x(No. of Sectors)	Final Sector Read from Diskette
0	0	00	(128)x(26) = 3,328	26 @ Side 0 or
0	1	01	(256)x(26) = 6,656	26 @ Side 1
1	0	00	(128)x(52) = 6,656	26 @ Side 1
1	1	01	(256)x(52) =13,312	
0	0	01	(256)x(15) = 3,840	15 @ Side 0 or
0	1	02	(512)x(15) = 7,680	15 @ Side 1
1	0	01	(256)x(30) = 7,680	15 @ Side 1
1	1	02	(512)x(30) =15,360	
0	0	02	(512)x(8) = 4,096	8 @ Side 0 or
0	1	03	(1024)x(8) = 8,192	8 @ Side 1
1	0	02	(512)x(16) = 8,192	8 @ Side 1
1	1	03	(1024)x(16) =16,384	

Table E2. Transfer Capacity

N	SC	GPL(1)	GPL(2)	Mode	Sector Size (Bytes/Sector)	Format
00	32	07	13	FM	128	IBM Diskette 1
01	17	16	52	FM	256	IBM Diskette 2
02	10	13	72	FM	512	
03	04	x	x	FM	1,024	
04	02	x	x	FM	2,048	
05	01	x	x	FM	4,096	
01	32	16	66	MFM	256	IBM Diskette 2D
02	17	13	124	MFM	512	
03	10	65	164	MFM	1,024	IBM Diskette 2D
04	04	x	x	MFM	2,048	
05	02	x	x	MFM	4,096	
06	01	x	x	MFM	8,192	

Note: The GPL(1) values are suggested for Read and Write instructions to avoid splice points between the Data Field and ID Field of contiguous sectors.

The GPL(2) values are suggested for the Format instruction.

The N, SC and GPL values are in hexadecimal.

The x values are to be determined by the user.

SC and GPL are listed in octal.

Table E3. Formatting Variables

Command	Status Register 2 SN = Bit 2, SH = Bit 3		Comments
Scan Equal	0	1	Data(Drive)=Data(Q-bus)
	1	0	Data(Drive)not=Data(Q-bus)
Scan Low or Equal	0	1	Data(Drive)=Data(Q-bus)
	0	0	Data(Drive)<Data(Q-bus)
	1	0	Data(Drive)>Data(Q-bus)
Scan High or Equal	0	1	Data(Drive)=Data(Q-bus)
	0	0	Data(Drive)>Data(Q-bus)
	1	0	Data(Drive)<Data(Q-bus)

Table E4. SH & SN Status Bits

BIT 7, BIT 6 : INTERRUPT CODE (IC).

Bit 7 = 0: Normal termination of instruction (NT).
Bit 6 = 0: Instruction was properly executed and completed.

Bit 7 = 0: Abnormal termination of instruction (AT).
Bit 6 = 1: Execution was started, but not sucessfully completed.

Bit 7 = 1: Invalid instruction issued (IC). Instruction
Bit 6 = 0: was issued, but never started.

Bit 7 = 1: Abnormal termination due to state change of a
Bit 6 = 1: drive Ready signal, during instruction execution.

BIT 5 : SEEK END (SE).

When the FDC completes the SEEK instruction, this flag is set (1).

BIT 4 : EQUIPMENT CHECK (EC).

If the drive fails to reach Track 00 after 77 Steps (during the Recalibrate instruction), then this flag will be set (1).

BIT 3 : NOT READY (NR).

When a drive is in a Not-Ready state and a read/write instruction is issued, this flag will be set. If a read/write instruction is issued to Side 1 of a single-sided drive, this flag will be set.

Table E5. Status Register 0 Description

BIT 2 : HEAD ADDRESS (HD).

This flag is used to indicate the state of a head when an interrupt occurs. When set, the head is down.

BIT 1 : UNIT SELECT 1 (US1).
BIT 0 : UNIT SELECT 0 (US0).

These bits indicate the selected drive when an interrupt occurs:

Bit 1	Bit 0	Drive Selected
0	0	0
0	1	1
1	0	2
1	1	3

Table E5. (Cont.) Status Register 0 Description

BIT 7 : END OF CYLINDER (EN).

When the FDC tries to access a sector beyond the final sector of a cylinder, this bit will be set.

BIT 6 :

Not used; always low (0).

BIT 5 : DATA ERROR (DE).

When the FDC detects a CRC error in either an ID field or a Data field, this flag is set.

BIT 4 : OVER RUN (OR).

If the FDC is not serviced within a certain time, during data transfers, this flag is set. Refer to the Read and Write Data and the Scan instruction descriptions for the time intervals.

BIT 3 :

Not used, always low (0).

BIT 2 : NO DATA (ND).

During the execution of a READ DATA, WRITE DELETED DATA or SCAN instruction, if the FDC cannot find the Sector specified in an ID register, this flag will be set.

During the execution of a READ ID instruction, if the FDC cannot read the ID field without an error, this flag will be set.

Table E6. Status Register 1 Description

During the execution of a READ A TRACK instruction, if the FDC cannot find the starting sector, this flag will be set.

BIT 1 : NOT WRITABLE (NW).

During the execution of a WRITE DATA, WRITE DELETED DATA or FORMAT A TRACK instruction, if the FDC detects a Write Protect condition from a drive, this flag will be set.

BIT 0 : MISSING ADDRESS MARK (MA).

If the FDC cannot detect the ID Address Mark after encountering the Index Hole twice, this flag will be set.

If the FDC cannot detect the Data Address Mark or Deleted Data Address Mark, this flag will be set; at the same time, the MD bit (Missing Address Mark in Data Field) of ST2 will also be set.

Table E6. (Cont.) Status Register 1 Description

BIT 7 : Not used; always low.

BIT 6 : CONTROL MARK (CM).

During a READ DATA or a SCAN instruction execution, if the FDC encounters a sector with a Deleted Data Address Mark, this flag will be set.

BIT 5 : DATA ERROR IN DATA FIELD (DD).

If the FDC detects a CRC error in the data field, this flag will be set.

BIT 4 : WRONG CYLINDER (WC).

When the C (Cylinder Number) value written on the track differs from the value stored in an ID register, this flag will be set. When WC is set, ND of ST1 will also be set.

BIT 3 : SCAN EQUAL HIT (SH).

During the execution of a Scan instruction, if the condition "Equal" is met, this flag will be set.

BIT 2 : SCAN NOT SATISFIED (SN).

During the execution of a Scan instruction, if the desired condition, Equal, High, or Low, is not met, this flag will be set.

Table E7. Status Register 2 Description

BIT 1 : BAD CYLINDER (BC).

When the value of C (Cylinder Number) written on the track is FF,hex (377,octal), this flag will be set. Also, when the C (Cylinder Number) value written on the track differs from the value stored in an ID register, this flag will be set. When BC is set, ND of ST1 will also be set.

BIT 0 : MISSING ADDRESS MARK IN DATA FIELD (MD).

If the FDC cannot read a Data Address Mark or Deleted Data Address Mark, this flag will be set.

Table E7. (Cont.) Status Register 2 Description

BIT 7 : Not used. Always 0.

BIT 6 : WRITE PROTECTED (WP).

This bit indicates the Write Protected state of a drive. Write Protected = 1.

BIT 5 : READY (RY).

This bit indicates the Ready state of a drive. Ready = 1.

BIT 4 : TRACK 00 (TO).

This bit indicates whether a drive head is positioned over Track 00. Track 00 = 1.

BIT 3 : TWO SIDE (TS).

This bit indicates if a two-headed drive AND a two-sided diskette are in use. Two-Sided = 1.

BIT 2 : HEAD ADDRESS (HD).

This bit indicates the side selected for a two-headed drive. Side 1 Selected = 1.

BIT 1 : UNIT SELECT 1 (US1).
BIT 0 : UNIT SELECT 0 (US0).

These two bits indicate the drive unit selected.

US1	US0	DRIVE SELECTED
0	0	0
0	1	1
1	0	2
1	1	3

Table E8. Status Register 3 Description

E5. Boot ROM

The FDC contains a ROM bootstrap program. The function of the program is to read logical block 0, from the system disk in Unit 0, and transfer it to memory locations 0 through 777, octal. When the transfer is completed, LSI-11 CPU control is transferred to location 0, where the software bootstrap for the system resides, to complete the boot process for the operating system.

The ROM bootstrap begins at location 173000, (octal), in the I/O page of the LSI-11 address space and is 256 words in length. Whenever a Reset instruction is executed, or the system reset (boot switch depressed), or when power is cycled, CPU control is transferred to location 173000. The bootstrap program will then:

1. Suspend CPU interrupts.

2. Create a temporary stack beginning at location 17000, octal.

3. Relocate itself, beginning at 10000, octal.

Once relocated, the bootstrap programs the FDC to read Sector 7, Cylinder 0, Side 0, of the diskette in Unit 0. At byte locations 72 and 76, diskette attribute identifiers are stored, as per the IBM specification. When the locations have been examined, and the diskette attributes determined, the FDC is appropriately programmed.

If the FDC and the disk drive are capable of supporting these attributes, logical block 0, of the system disk, is read into memory. If errors occur during this transfer, the bootstrap will attempt retries to correct the transfer process. If the prescribed number of retries will not complete the transfer, the CPU will suspend operation by executing a branch-to-self instruction. At this time, CPU register R0 will contain the memory location of the FDC command sequence being processed at the time of the error. CPU register R1 contains the memory location of the FDC status returned during the command sequence.

There are several special attributes of the ROM bootstrap program. One is that the bootstrap program will wait until drive unit 0 becomes ready, meaning that a diskette need not be inserted in the drive at the beginning of the boot process. The program will wait indefinitely, until a diskette is inserted and the drive door closed. Another feature is that the ROM bootstrap contains a Read-Only-Handler, which is activated via the breakpoint (BPT) instruction. Activation of the bootstrap, and its relocation, will set up locations 14 and 16, octal, to utilize the Read-Only-Handler.

The argument sequence is as follows:

R0 = Physical Track No. (0-76, decimal)
R1 = Running word count
R2 = Running buffer address
R3 (high byte) = Head No. (Always 0 for single-sided drives)
R3 (low byte) = Sector No. (1-26, decimal, for single-density)
(1-15, decimal, for double-density)

This routine may be used for diagnostic read, or tests.

```
; VARIABLE DENSITY FLOPPY DISK CONTROLLER (FDC) BOOT ROM CODE
;
; CHARACTERISTICS
;
;       SIZE AVAILABLE    = 1000 (512.) BYTES, 400 (256.) WORDS
;
;       LOCATION          = 173000
;
;       DEVICE VECTOR     = 173776 (776 RELATIVE)
;
;       CHKSUM            = 173774 (774 RELATIVE)
;
;       ID VALUES         = 173764,173766,173770,173772 (764,766,770,772
;                                                        RELATIVE)
;
; CONTROLLER ADDRESSES
        QBCS    =171000         ;FDC CONTROL + STATUS REGISTER
        QBMS    =QBCS+2         ;FDC DATA REGISTER

        QBDC    =171012         ;DMA CONTROLLER STATUS REGISTER
        QBTC    =QBDC+2         ;DMA WORD COUNT REGISTER
        QBBA    =QBDC+4         ;DMA ADDRESS REGISTER


; FDC COMMAND VALUES
        SPECFI  =003            ;SPECIFY COMMAND
        RECAL   =007            ;RE-CAL COMMAND
        REED    =006            ;READ COMMAND
        SEEK    =017            ;SEEK COMMAND
        SENSAT  =010            ;SENSE INTERRUPT STATUS COMMAND

; DMA COMMAND VALUES
        DMARED  =40000          ;DMA READ (XFER TO MEMORY)
        DMAENB  =  121          ;DMA READ ENABLE

; FDC DATA VALUES
        STEP    =-6.            ;STEP RATE FOR BOOT PROCESS (6. X 1 MSEC)
        UNHLD   =17             ;HEAD UNLOAD TIME
        HLOADS  =18.            ;HEAD LOAD TIME (18. X 2 MSEC)
        DMAMOD  =000            ;DMA MODE
        SD      =000            ;SINGLE DENSITY
        DD      =100            ;DOUBLE DENSITY
        DS.ID   =324            ;VOLUME ID FOR DS
        DD.ID   =362            ;VOLUME ID FOR DD
        SIZE1   =64.            ;WORDS PER SECTOR SSSD
        SIZE2   =256.           ;WORDS PER SECTOR SSDD, DSDD
        N1      =0              ;SSSD = ((128. BYTES PER SECTOR)/128.)-1
        N2      =2              ;SSDD,DSDD = ((512. BYTES PER SECTOR)/128.)-1
        EOT1    =26.            ;SSSD = 26. SECTORS PER TRACK
        EOT2    =15.            ;SSDD,DSDD = 15. SECTORS PER TRACK
        GPL1    =7.             ;SSSD GAP LENGTH
        GPL2    =27.            ;SSDD,DSDD GAP LENGTH
        DTL1    =377            ;SSSD DATA LENGTH
        DTL2    =377            ;SSDD,DSDD DATA LENGTH
        QV1.1   =EOT1*400!N1
        QV2.1   =DTL1*400!GPL1
        QV1.2   =EOT2*400!N2
        QV2.2   =DTL2*400!GPL2
```

```
; ROM BOOTSTRAP VALUES
        BBPT    =014            ;BPT ADDRESS
        STACK   =17000          ;ADDRESS FOR TEMPORARY STACK
        BTLOAD  =10000          ;ADDRESS TO RELOCATE ROM TO IN RAM
        ROMSIZ  =<VECT+2-BOOT>/2        ;ROM SIZE IN WORDS
        LOAD1   =200            ; ADDRESS FOR SSSD,DSSD TRACK1,SECTOR3
        LOAD2   =1000           ; ADDRESS FOR SDDD,DSSD TRACK1,SECTOR3

; START VDC EXECUTABE ROM CODE
;
BOOT:   MTPS    #340            ;DISABLE INTERRUPTS
        MOV     #STACK,SP       ;SET A TEMPORARY STACK

; RELOCATE THE ROM INTO RAM
; DO IT IN SUCH A WAY THAT IT CAN BE LOCATED ANYWHERE IN MEMORY
;
        MOV     PC,R2           ;GET CURRENT PC
        SUB     #<.-BOOT>,R2    ;R2 = LOC (BOOT:)

        MOV     #BTLOAD,R1      ;R1 = LOC (TARGET LOCATION FOR ROM)

; INIT CHECK SUM ACCUMULATORS
;
        CLR     R4
        CLR     R5

; RELOCATION + CHKSUM VERIFY LOOP
;
        MOV     #ROMSIZ,R0      ;SET WORD COUNT TO XFER
RELOOP: MOV     (R2)+,(R1)
        MOV     (R1)+,R3
        ADD     R3,R4
        SWAB    R3
        ADD     R3,R5
        SOB     R0,RELOOP

;TEST FOR RELOCATION ERROR
;
        MOVB    R4,(R1)+        ; LOW BYTE CHECKSUM
        MOVB    R5,(R1)+        ; HIGH BYTE CHECKSUM
        TST     -(R1)
        BEQ     XFERTO          ;BRANCH IF NO XFER FAILURE
        BR      .               ;IDLE AS HAD AN ERROR

; JUMP TO RELOCATED CODE
;
XFERTO: MOV     #BTLOAD+<BSTRT-BOOT>,PC
```

```
; START OF THE FUNCTIONAL PORTION OF THE BOOTSTRAP
;
 STRT:

; SET UP BPT FOR READ ONLY HANDLER
        MOV     #BTLOAD+<QROH-BOOT>,-(SP)       ; SAVE FOR FUTURE REFERENCE
        MOV     (SP),@#BBPT
        MOV     #340,@#BBPT+2

; USE THE "SPECIFY" COMMAND TO PROGRAM NEC CHIP
;
        CALL    XQT
        .WORD   BTLOAD+<SPECK-BOOT>

; SEEK INTERNAL TRACK FOR GUARANTEED RE-CAL AND DRIVE READY
;               AUTOMATIC SENSE INTERRUPT GIVEN
;
FSEEK:  DECB    SENCNT          ; CLEAR RETRY COUNT
        CALL    XQT
        .WORD   BTLOAD+<SEEKT-BOOT>

        BCS     FSEEK

; DO A "RE-CAL" SO WE GET IN SYNC
;               AUTOMATIC SENSE INTERRUPT GIVEN
;
RECAL:  CALL    XQT
        .WORD   BTLOAD+<REKAC-BOOT>

        BCS     RECAAL          ; ERROR TRY AGAIN


;READ TRACK 0 SECTOR 7
;
RDT057: CLR     R0              ;TRACK 0
        MOV     #7,R3           ;SECTOR 7
        MOV     #STACK,R2       ;BUFFER
        MOV     SECSIZ,R1       ;WORD COUNT = 64. WORDS
        BPT                     ;TRAP TO ROH
```

```
;DECODE VOLUME CHARACTERISTICS
;
DECODE: MOV     (PC)+,R5        ; ADDRESS OF CYL=0,SECT=7 BUFFER
        .WORD   STACK+<72.-1>

        BISB    4(R5),(R5)      ; "OR" TWO MEDIA CHAR BYTES TOGETHER
                                ; SSSD = 100 (SS=100 "OR" SD=100)
                                ; SSDD = 362 (SS=100 "OR" DD=362)
                                ; DSSD = 324 (DS=324 "OR" SD=100)
                                ; DSDD = 366 (DS=324 "OR" DD=362)
        MOVB    (R5),R5         ; GET IT IN A REGISTER
        BIC     #177770,R5      ; CLEAR ALL HIGH BITS
        ASR     R5              ; DIVIDE BY 2
                                ; SSSD = 0
                                ; SSDD = 1
                                ; DSSD = 2
                                ; DSDD = 3
        MOV     R5,-(SP)        ; SAVE FOR FUTURE MEDIA ID
        ASR     R5              ; TEST FOR DOUBLE DENSITY
        BCC     RDT1S3

        BISB    #DD,QREED       ; SET NEW READ COMMAND =  DD

        MOV     #QV1.2,QV1      ;NEED TO RESET SOME FDC VALUES
        MOV     #QV2.2,QV2
        MOV     #SIZE2,SECSIZ   ;MUST ALSO SET NEW SECTOR SIZE
        MOV     #LOAD2,LOADPT   ;SET LOAD START ADDRESS

; READ TRACK 1 SECTOR 3
;
RDT1S3: MOV     #1,R0           ;TRACK 1
        MOV     #3,R3           ;SECTOR 3
        MOV     (PC)+,R2        ;BUFFER ADDRESS
LOADPT: .WORD   LOAD1
        MOV     #2000,R1        ;WORD COUNT = MAX SECTOR SIZE X 4 IN WORDS
        BPT                     ;TRAP TO ROH

; READ TRACK 1 SECTOR 5
;
RDT1S5: MOV     #5,R3           ;SECTOR 5
        BPT                     ;TRAP TO ROH

; READ TRACK 1 SECTOR 7
;
RDT1S7: MOV     #7,R3           ;SECTOR 7
        BPT                     ;TRAP TO ROH

; READ TRACK 1 SECTOR 1
;
RDT1S1: MOV     #1,R3           ;SECTOR 1
        CLR     R2              ;READ INTO LOC (0)
        BPT                     ;TRAP TO ROH
```

```
; SET MEDIA CHARACTERSTIC ID

        CLR     R1              ; SET TO STORE IN LOC = 0
        MOV     (PC)+,(R1)      ; FLOPPY CONTROLLER ID
        MOV     R0,R2
        ADD     (SP)+,(R1)      ; ADD IN MEDIA CHARACTERISTICS

; DONE SO XFER CONTROL TO THE SOFT BOOT (JMP LOC 0)
;
JLOC0:  CLR     PC


        .SBTTL  VDC READ ONLY HANDLER

; THIS ROH PROVIDES ALL READ CAPABILITIES REQUIRED FOR THE BOOT PROCESS
;
; IT ADHERES TO THE ROH CALLING SEQUENCE DEFINED IN THE SOFT BOOT FOR
;                       . DEC RT-11 V3B
;                       . UCSD PASCAL VII.0
;
; SAMPLE CALLING SEQUENCE
;       MOV     TN,R0           ;TN = TRACK #
;       MOV     WC,R1           ;WC = WORD COUNT
;       MOV     BF,R2           ;BF = TARGET BUFFER ADDRESS
;       MOV     SN,R3           ;SN = (HI-BYTE=HEAD, LO-BYTE=SECTOR #)
;       BPT
;       MOV     R0,TNS          ;TNS = SAME VALUE AS INPUT
;       MOV     R1,WCL          ;WCL = WORD COUNT - # WORDS XFERED
;       MOV     R2,BFL          ;BFL = BUFFER ADDRESS + # WORDS XFERED
;
;       R4,R5 ARE DESTROYED
;
QROH:   MOV     R0,-(SP)        ;SAVE R0 AS DEFINED BY ROH CONVENTION


; SET TRACK AND SECTOR
;
        MOVB    R0,QTRAK
        MOVB    R3,QSECT

; SET HEAD
;
        SWAB    R3
        BNE     1$
        CLRB    QMOD            ;SET FOR HEAD 0
        CLRB    QHED
        BR      2$
1$:     MOVB    #4,QMOD         ;SET FOR HEAD 1
        MOVB    R3,QHED
```

```
;SEEK THE TRACK
;       AUTOMATIC SENSE INTERRUPT GIVEN
;
2$:     CMPB    R0,STRAK        ;SEE IF NEED A TRACK SEEK
        BEQ     PRODMA          ;ALREADY THERE, NO NEED TO SEEK
        MOVB    R0,STRAK
QSEEK:  CALL    XQT
        .WORD   BTLOAD+<SEEKT-BOOT>

        BCS     QSEEK           ;STILL HOPE, TRY AGAIN

;       SET DMA XFER COUNT + "DMA XFER TO MEMORY"
;
PRODMA: MOV     (PC)+,R4        ;FULL SECTOR SIZE
SECSIZ: .WORD   SIZE1
        CMP     R4,R1           ;COMPARE WITH WHAT WAS ASKED FOR
        BLE     4$
        MOV     R1,R4           ;ONLY TRANSFER WHAT WAS ASKED FOR
4$:     MOV     R4,-(SP)        ;SAVE WORD COUNT

        ASL     R4              ;CONVERT TO BYTES
        DEC     R4              ;DMA XFERS COUNT + 1 (ODDITY)
        BIS     #DMARED,R4      ;SET DMA DISK READ BITS

; PROGRAM DMA FOR READ
QDMA:   MOV     #QBBA,R5        ;DMA ADDRESS REGISTER
        MOV     R2,(R5)         ;BUFFER LOW ORDER ADDRESS BYTE
        SWAB    R2
        MOV     R2,(R5)         ;BUFFER HI ORDER ADDRESS BYTE
        SWAB    R2              ;RESTORE
        MOV     R4,-(R5)        ;DMA COUNT +DISK READ
        SWAB    R4
        MOV     R4,(R5)
        MOV     #DMAENB,-(R5)   ;ENABLE DMA

; DO THE NEC COMMAND SEQUENCE TO READ
;
QREAD:  CALL    XQT
        .WORD   BTLOAD+<READS-BOOT>

        BCS     QREAD           ;DO IT AGAIN

; CLEAN UP OPERATION
3$:     MOV     (SP)+,R0        ;GET WORD COUNT XFERD
        SUB     R0,R1           ;ADJUST WORD COUNT
        ASL     R0              ;CONVERT TO ADDRESS
        ADD     R0,R2           ;NEW BUFFER ADDRESS
        MOV     (SP)+,R0
        RTI
```

```
;EXECUTE A FDC COMMAND SEQUENCE
;
;       CALL    XQT
;       .WORD   COMTBL
;
;       COMTBL IS NEC COMMAND TABLE
;       R5,R4,R3 ARE DESTROYED, OTHER REGISTERS UNTOUCHED
;
;       RETURN  CARRY=0         NO ERROR
;               CARRY=1         ERROR
;
XQT:    MOV     @(SP),R5        ;SET COMMAND TABLE ADDRESS
        ADD     #2,(SP)         ; BUMP AROUND DATA TABLE ADDRESS

        MOVB    (R5)+,R4        ;SET BYTE COUNT TO XFER

TSTRDY: MOV     @#QBCS,R3       ;FDC READY
        BMI     4$              ; MINUS IS A YES
        MOV     (SP)+,-(SP)     ; NOPE SO DELAY
        BR      TSTRDY          ; AND TRY AGAIN

4$:     ROL     R3              ; TEST FDC DIRECTION
        BPL     5$              ; NOT = 0, ERROR
        CLR     R4              ; CLEAR SO WILL GET ENTIRE STATUS
        BR      STAT            ; CONFUSED, WANTS TO SEND STATUS

5$:     MOVB    (R5)+,R3        ;MOVE BYTE TO REGISTER TO EXPAND TO WORD
        MOV     R3,@#QBMS       ;MOVE COMMAND TO FDC
        SOB     R4,TSTRDY       ;LOOP UNTIL DONE

WCMPLT: MOV     #400,R3         ; DELAY BEFORE TESTING

1$:     MOV     (SP)+,-(SP)     ; DELAY APPROXIMATELY 10 MICRO-SEC
        SOB     R3,1$

        MOV     @#QBCS,R3       ; TEST RQM
        BPL     WCMPLT          ; POSITIVE, FIX NOT READY

7$:     MOVB    (R5)+,R4        ;GET NUMBER OF BYTES IN NEXT COMMAND IN SEQUEN
        BNE     TSTRDY          ; ZERO MEANS COMMAND SEQUENCE DONE
        MOVB    (R5)+,R4        ; GET NUMBER OF BYTES IN STATUS
        BEQ     RETRN           ; ZERO MEANS NO STATUS BYTES TO GET


STAT:   MOV     R5,-(SP)        ; SAVE COMMAND TABLE POINTER
        MOV     SP,R5           ; USE STACK AREA FOR STATUS VALUES
        SUB     #<7+1>*2,R5     ; MAXIMUM NUMBER OF STATUS WORDS IS 7.
        MOV     R5,-(SP)        ; SAVE START OF STATUS AREA

1$:     MOV     @#QBCS,R3       ;TEST IF READY
        BMI     2$              ; MINUS IS A YES
        MOV     (SP)+,-(SP)     ; DELAY
        BR      1$              ; AND TRY AGAIN
2$:     ROL     R3              ; CHECK DIRECTION
        BPL     3$              ; ERROR, NO MORE STATUS
```

```
        MOV     @#QBMS,(R5)+    ;STORE STATUS WORD
        SOB     R4,1$           ;SEE IF MORE STATUS WORDS

3$:     MOV     (SP)+,R4        ; RESTORE START OF STATUS TABLE AREA
        MOV     (SP)+,R5        ; RESTORE COMMAND TABLE POINTER

        CMPB    #77,(R4)        ;TEST FOR ERROR CONDITION
        BCS     ERR             ; CARRY SET MEANS ERROR

RETRN:  RETURN

ERR:    INCB    (R5)            ; INCREMENT COMMAND RETRY
        BPL     RETRN           ; WHEN 200 BIT SET, RETRIED ENOUGH
FERR:   MOV     (SP),R0         ; ADDRESS OF COMMAND IN R0
        TST     -(R0)           ; DECREMENT BY 2
        MOV     R4,R1           ; ADDRESS OF STATUS IN R1
        BR      .

;NEC COMMAND TABLE FORMAT
;       .BYTE   N       = # BYTES IN COMMAND SEQUENCE
;       .BYTE   ARG#1
;       .BYTE   ....
;       .BYTE   ARG#N
;       ............    THE COMMAND SEQUENCE CAN BE ANY LENGTH
;       ............    THE SEQUENCE IS TERMINATED BY N=0
;       .BYTE   NS      =# BYTES IN STATUS (0 MEANS NO STATUS)
;       .BYTE   RTRY    = RETRY COUNT (IF NS = 0, THEN THIS BYTE NOT REQUIRED)

;NEC COMMAND SEQUENCE "SPECIFY FDC CHARACTERISTICS"
        ;NEC COMMAND SEQUENCE "SPECIFY FDC CHARACTERISTICS"
SPECK:  .BYTE   3
        .BYTE   SPECFI
        .BYTE   STEP*20!UNHLD
        .BYTE   HLOADS*2!DMAMOD
        .BYTE   0               ; NO MORE COMMANDS
        .BYTE   0               ; NO STATUS BYTES

;NEC COMMAND SEQUENCE "SEEK TRACK" + "SENSE INTERRUPT STATUS"
        ;NEC COMMAND SEQUENCE "SEEK TRACK"
SEEKT:  .BYTE   3
        .BYTE   SEEK
        .BYTE   0               ; UNIT 0
STRAK:  .BYTE   5
        ;NEC COMMAND SEQUENCE "SENSE INTERRUPT STATUS"
        .BYTE   1
        .BYTE   SENSAT          ;SENSE STATUS COMMAND
        .BYTE   0               ; NO MORE COMMANDS
        .BYTE   2               ; 2 BYTES OF STATUS
SENCNT: .BYTE   0               ; RETRY COUNTER
```

```
;NEC COMMAND SEQUENCE RECALIBRATE DISK DRIVE" + " SENSE INTERRUPT STATUS"
        ;NEC COMMAND SEQUENCE "RECALIBRATE DISK DRIVE"
REKAL:  .BYTE   2
        .BYTE   RECAL
        .BYTE   0               ;UNIT 0
        ;NEC COMMAND SEQUENCE "SENSE INTERRUPT STATUS"
        .BYTE   1
        .BYTE   SENSAT
        .BYTE   0               ; NO MORE COMMANDS
        .BYTE   2               ; 2 BYTES OF STATUS
        .BYTE   0               ; RETRY COUNTER

;NEC COMMAND SEQUENCE "READ SECTOR" + COMMAND IMPLIED 'SENSE INTERRUPT STATUS'
        .EVEN
READS:  .BYTE   9.
QREED:  .BYTE   REED            ;READ COMMAND
QMOD:   .BYTE   0               ;READ COMMAND HEAD + UNIT
QTRAK:  .BYTE   0               ;TRACK
QHED:   .BYTE   0               ;HEAD
QSECT:  .BYTE   0               ;SECTOR
QV1:    .WORD   QV1.1
QV2:    .WORD   QV2.1
        .BYTE   0               ; NO MORE COMMANDS
        .BYTE   7               ; 7 BYTES OF STATUS
        .BYTE   0               ; RETRY COUNTER


  DATA AREA
 .BOOT+764
CTRLID: .WORD   0               ; THESE WORDS RESERVED FOR
        .WORD   0               ; CONTROLLER INDENTIFICATION
        .WORD   0
        .BYTE   'A              ; REVISION
        .BYTE   ATRIBS          ; ATTRIBUTES
CHKSUM: .WORD   0               ; CHECKSUM
VECT:   .WORD   234             ; DEVICE VECTOR

        .END
```

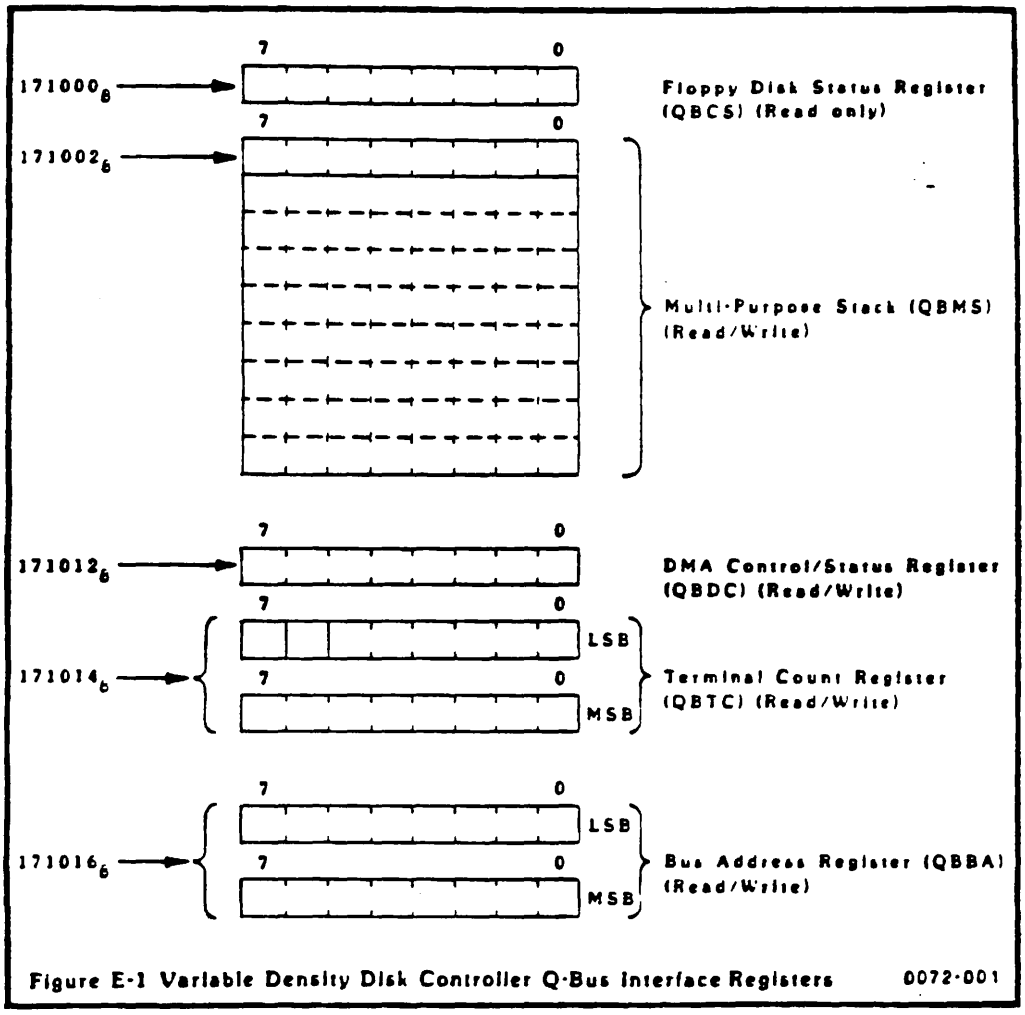
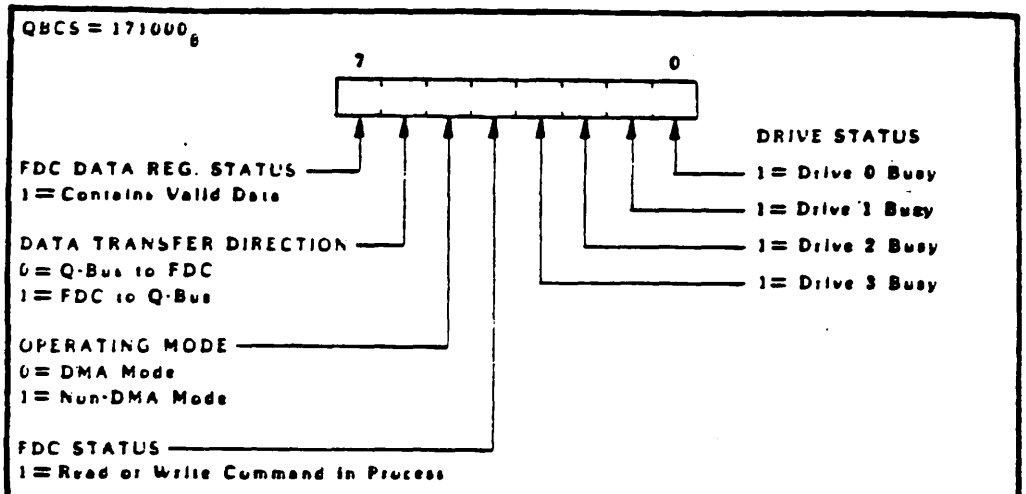
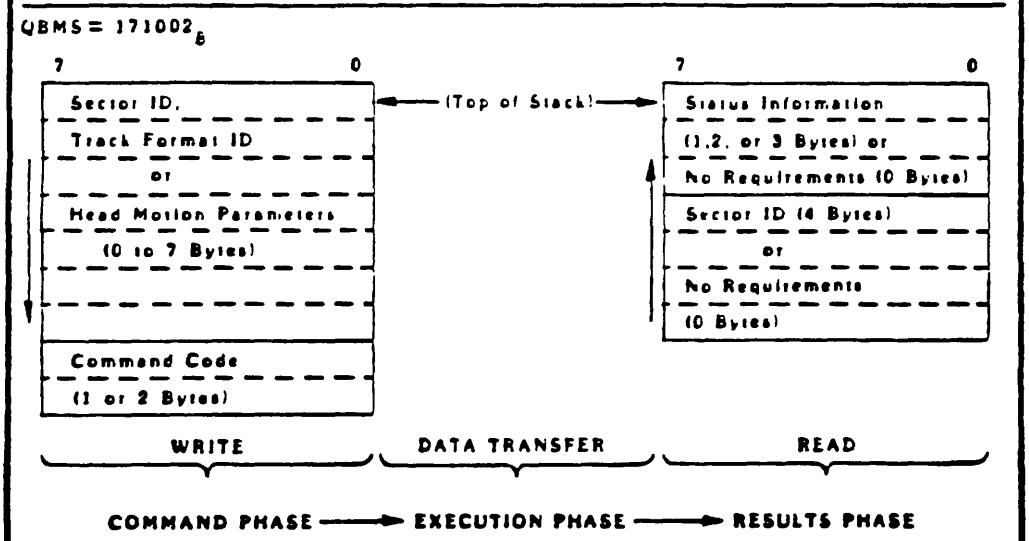


Figure E-1 Variable Density Disk Controller Q-Bus Interface Registers 0072-001



(QBCS) Floppy Disk Status Register (Read Only)



Multi-Purpose Stack (QBMS) (Read/Write Stack)

Figure E-2 QBCS , QBMS Registers

0073-001

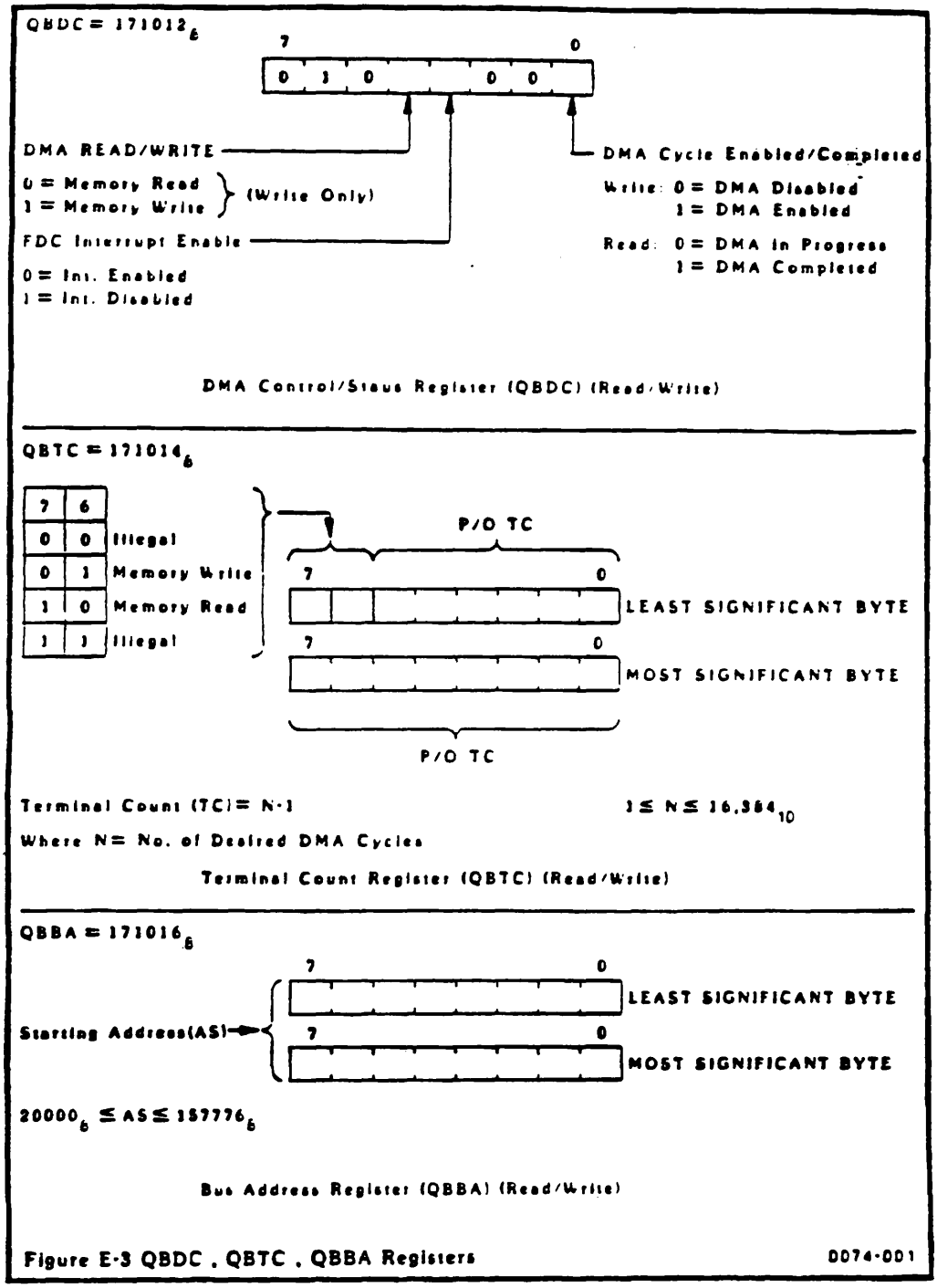


Figure E-3 QBDC , QBTC , QBBA Registers

0074-001

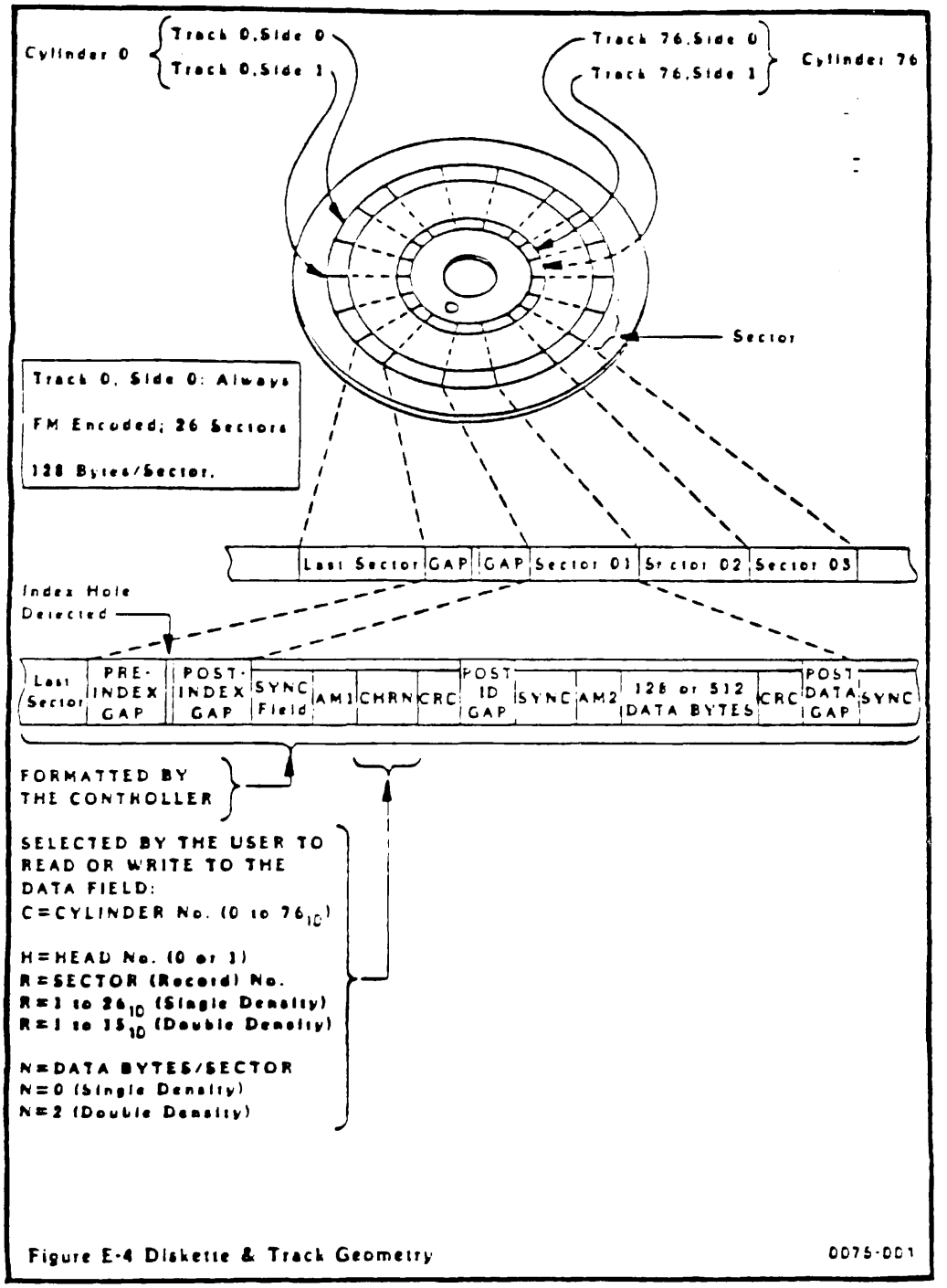


Figure E-4 Diskette & Track Geometry

0075-001