96761300

CONTROL DATA
CORPORATION

CDC® STORAGE MODULE DRIVE INTERFACE

GB138-A

GENERAL DESCRIPTION
OPERATION AND PROGRAMMING
INSTALLATION
THEORY OF OPERATION
DIAGRAM DESCRIPTION
MAINTENANCE

HARDWARE REFERENCE/MAINTENANCE MANUAL

REVISION RECORD

REVISION	DESCRIPTION
A	Manual released.
(12/76)	
B	Manual revised; includes ECO 14795/A/B, correcting storage module drive timing, memory and stability
(5/77)	problems. Pages ii, iii, v, vii, 2-23, 4-5, 4-14, 4-22, 5-1, 5-2, 5-3, 5-7, 5-9, B-2, B-3, B-4, B-7, B-10, B-12,
	B-13, B-15, B-17 thru B-22, B-25 thru B-29, B-31, B-32, B-34 thru B-36, B-39, and B-41 are revised.
	Appendix C is replaced in its entirety.
Publication No. 96761300	

REVISION LETTERS I, O, Q AND X ARE NOT USED

Address comments concerning this manual to:

Control Data Corporation
Publications and Graphics Division
4455 Eastgate Mall
La Jolla, California 92037

or use Comment Sheet in the back of this manual.

MANUAL TO EQUIPMENT LEVEL CORRELATION SHEET

This manual reflects the equipment configurations listed below.

EXPLANATION: Locate the equipment type and series number, as shown on the equipment FCO log, in the list below. Immediately to the right of the series number is an FCO number. If that number and all of the numbers underneath it match all of the numbers on the equipment FCO log, then this manual accurately reflects the equipment.

EQUIPMENT TYPE	SERIES	WITH FCOs	COMMENTS
GB138-A	02	ECO 14606	
	03	ECO 14738	
	04	ECO 14795	

LIST OF EFFECTIVE PAGES

New features, as well as changes, deletions, and additions to information in this manual, are indicated by bars in the margins or by a dot near the page number if the entire page is affected. A bar by the page number indicates pagination rather than content has changed.

PAGE	REV
Cover	--
Title Page	--
ii	B
iii/iv	B
v/vi	B
vii/viii	B
1-1 thru 1-3	A
2-1 thru 2-22	A
2-23	B
2-24 thru 2-28	A
3-1 thru 3-3	A
4-1 thru 4-4	A
4-5	B
4-6 thru 4-13	A
4-14	B
4-15 thru 4-21	A
4-22	B
4-23	A
4-24	A
5-1 thru 5-3	B
5-4 thru 5-6	A
5-7	B
5-8	A
5-9	B
5-10 thru 5-12	A
6-1	A
A-1	A
A-2	A
B-1	A
B-2 thru B-4	B
B-5	A
B-6	A
B-7	B
B-8	A
B-9	A
B-10	B
B-11	A
B-12	B
B-13	B
B-14	A
B-15	B
B-16	A
B-17 thru B-22	B
B-23	A
B-24	A
B-25 thru B-29	B
B-30	A
B-31	B
B-32	B
B-33	A
B-34 thru B-36	B
B-37	A
B-38	A
B-39	B
B-40	A
B-41	B
B-42 thru B-49	A
C-1 thru C-42	B
Comment Sheet	B
Cover	--

PREFACE

This manual describes the operation, programming, and logic analysis at a maintenance level of the CDC® Storage Module Drive Interface. This device utilizes a combination of hardware and controlware to perform its internal functions. At a systems level, the device functions as an intelligent interface between a central processing unit and a multiple disk drive system controller.

This manual is intended for people who have a working knowledge of programming, operation, and formats used in disk systems. Hardware-oriented individuals should have a basic understanding of micro programming as applied to peripheral micro controllers.

The CDC publications listed below provide additional information related to the operation and programming of the drive interface. Copies can be ordered from CDC's LDS catalog.

Description	Publication No.
CYBER 18 Processor with Core Memory (Macro Level) Reference Manual	88973500
Operational Diagnostic System (ODS) Version 1 Reference Manual	39452100
FA7A8/FA727 SMD Controller/Formatter Hardware Maintenance Manual	83312400
GB138-A Field Print Package	96751700
CYBER 18-20/30 Timeshare Computer Systems Hardware Maintenance Manual, Volumes 1 and 2	96769301 96769302
BJ701/BJ7B1 Storage Module Drive Hardware Maintenance Manual	83311300
BJ701/BJ7B1 Storage Module Drive Hardware Reference Manual	83317300

CONTENTS

APPENDIXES

FIGURES

TABLES

GENERAL DESCRIPTION

The storage module drive interface is an intelligent controller that provides a generalized peripheral interface. As used in the CYBER 18 computer family, the drive interface is an integral part of the storage module system (SMS) and is dedicated to interfacing a multiple disk drive subsystem to the central processing unit (CPU).

PHYSICAL DESCRIPTION

The drive interface is a four-layer, 11- by 14-inch printed circuit assembly. It occupies one slot within the CPU board cage or the expansion chassis.

The storage module system (figure 1-1) is comprised of no more than two drive interfaces (one per CPU for dual configurations) and the disk subsystem, consisting of one controller/formatter and any mixture of up to eight storage module disk drives of either 25- or 50-megabyte capacity.

FUNCTIONAL DESCRIPTION

The drive interface provides a front-end A/Q-direct memory access peripheral controller-type interface to the CPU's backplane. Its back end interfaces through input/output port cabling to a tag bus-type interface presented by the controller/formatter's front end. All communications and data exchanges between the CPU and the storage module system must pass through the drive interface.

Software input and output commands to the storage module system and the status words of the drive interface, controller/formatter, or disk drives are handled through the CPU's A/Q input/output channel. Data transfers are handled through the CPU's direct memory access channel. The drive interface contains a 64-word data buffer first-in/first-out (FIFO) and 16 file registers for setting up the sector, head, and cylinder addressing on the selected file.

Interface handshaking and internal control and data paths are directed by a read-only memory-resident program (controlware) within the drive interface. The data word between the drive interface and controller/formatter consists of an 8-bit byte; between the drive interface and CPU, 16-bit words are used.

LOGIC CIRCUITS

The drive interface utilizes standard TTL-type (SSI and MSI) logic packages for its control and data handling, MOS logic for data storage, and semiconductor memory (read-only memory) for implementing its control program logic (controlware).

POWER

The drive interface logic obtains its power from the CPU system backplane. The drive interface requires approximately 4.0 amperes at 5 vdc for the logic circuits and -12 vdc at 50 milliamperes for the FIFO buffer integrated circuits.

ENVIRONMENTAL CONDITIONS

Drive Interface cooling is accomplished by the external fans in the CPU board cage. The drive interface performs satisfactorily over a temperature range of 40° to 120° F (4.4° to 48.9° C) and a relative humidity factor of 10 to 90 percent, noncondensing.

PROCESSING RATE

The drive interface is designed to handle the maximum peak data transfer rate of the disk. The average data transfer rate of the drive interface to/from the CPU is:

605 kHz (1.65 microseconds per word) – 3600 rpm drives

STORAGE MODULE SYSTEM FEATURES

Because the storage module system consists of several components, standard features as well as options are supported by, but not implemented in, the storage module system hardware. They are mentioned here as an aid to understanding the programming and operation of the drive interface.

ERROR CORRECTION

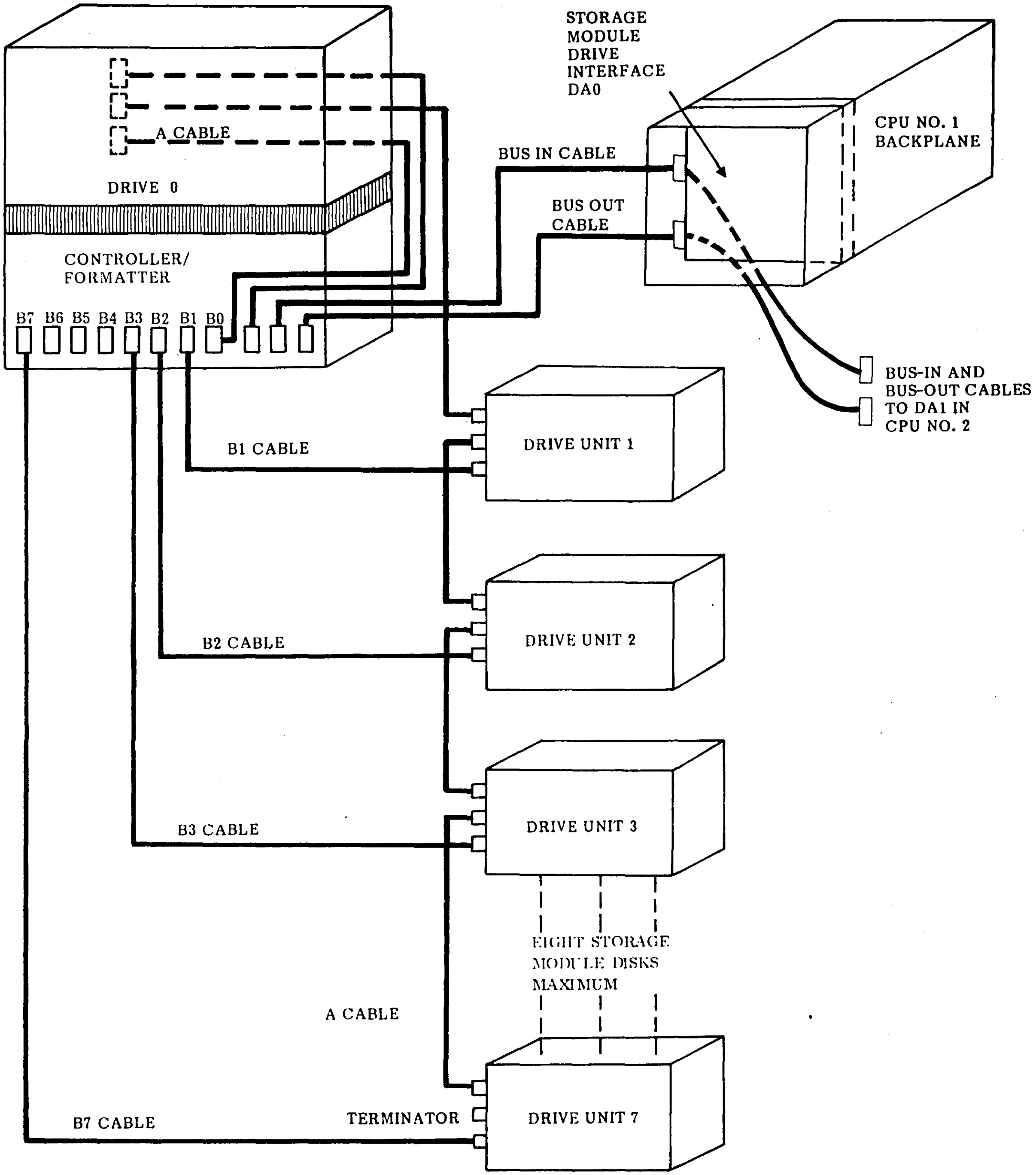
A standard storage module system error correction code (ECC), with the controller/formatter option included, is capable of detecting all single 22-bit (or less) error bursts and correcting all single 11-bit (or less) error bursts in any physical record. The correction is accomplished at the direction of the system software.

FOUR-DRIVE EXPANSION

The storage module system standard 4-drive expansion with the controller/formatter option allows the multiplexing capability of the controller/formatter to expand from the standard four drives to a maximum of eight.

DUAL ACCESS

The dual access feature with the controller/formatter option and disk adapter jumpers provides the capability to access the controller/formatter from two drive interfaces. Control contention must be resolved by the drive interfaces through software.



NOTE: FOR MORE DETAIL, REFER TO THE CYBER 18-20/30 TIMESHARE HARDWARE MAINTENANCE MANUAL,VOLUME 1.

0365

Figure 1-1. Storage Module System

ERROR RECOVERY AIDS

The drive-type dependent error recovery aids, 16 values of drive head offset and two variations of data recovery timing, can be selected by software to assist in error recovery procedures.

ROTATIONAL POSITION SENSING

A rotational position sensing (RPS) capability, included via a status command, allows the program to read the physical number of the sector currently under the read/write heads of the selected drive.

SECTOR INTERLACE

The capability for a software-prescribed sector interlace scheme is included. It allows the software to create logical sectors irrespective of the physical (sequential) sector numbering.

OVERLAPPED SEEKS

A seek complete occurring on any drive is logged by the controller/formatter and flagged to the appropriate drive interface. A software poll command is used to determine which drive is in the seek end condition.

LOGICAL-TO-PHYSICAL DRIVE NUMBER CORRELATION

The logical-to-physical drive number correlation feature provides a software command to correlate the selected drive's physical unit number with its logical unit number.

SELECTABLE SECTORING

The number of sectors per revolution can be selected for a range of 1 to 128 when a sector logic plug is installed in the storage module disk. Not all sector counts result in equal length sectors. For example, a 50-sector plug allows 49 equal length sectors with an odd length (longer) sector at the end. Sector plugs that result in equal length sectors are as follows:

> 4, 5, 6, 7, 8, 10, 12, 14, 15, 16, 20, 21, 24, 28, 30, 32, 35, 40, 42, 48, 56, 60, 64, 70, 80, 84, 96, 105, 112, 120, 128.

The standard number of sectors per revolution is 64.

AUTOLOAD

The autoload feature initiates a bootstrap read of 1,536 words from cylinder, head, and sector 0 on the selected drive of the drive interface jumper.

DRIVE INTERFACE SELF-TEST

The drive interface internal data paths, data buffering capabilities, and direct memory access control logic may be tested independently of the controller/formatter and disk drives via self-test read/write commands.

DATA ECHO

Two levels of data echo capability are included as maintenance and diagnostic aids. The first level allows data to be output through the drive interface to the controller/formatter receivers and transmitters and back to the CPU. The second level verifies the data path to the selected drive's receivers and returns to the CPU through the drive's transmitters.

PARAMETER JUMPERS

The drive interface contains several solder-type printed circuit jumpers that are visible when the drive interface board. is removed from the CPU chassis. These jumpers implement system parameters that define equipment configuration and operating modes. The functions of the jumpers are described here; the jumper pin numbers are identified in table 3-1.

EQUIPMENT NUMBER

On all A/Q input/output commands, bits 7 through 10 of the Q register (E field) contain the input/output device address/equipment number. Four jumpers on the drive interface, representing hexadecimal codes 0 through F, must be set to correspond to the correct drive interface equipment number.

PROTECT OVERRIDE

The drive interface may operate as a protected or unprotected input/output device. One jumper provides for selecting this option, which enables the drive interface to write into protected memory.

AUTOLOAD DRIVE NUMBER

Four jumpers are provided for selecting the logical drive unit number when an autoload (bootstrap) operation is initiated from the console. Upon depressing the AUTOLOAD switch, the selected disk drive seeks to address 0 (cylinder, head, and sector 0) and reads 1,536 sequential words from this area. This data is stored into sequential CPU memory locations, beginning with location 0.

DUAL CPU CONFIGURATION

In a dual CPU system, two drive interfaces can be connected to a single controller/formatter. Communication occurs, not only between the drive interface and the controller/formatter, but also between drive interfaces. For proper operation, each drive interface must be configured with wire jumpers so that it transmits and receives only those signals that correspond to its unit number. Unit 0 is the first drive interface on the daisy chain cable from the controller/formatter. Unit 1 is the remaining drive interface on the cable. The jumpers and their applications follow.

One jumper is required to allow the drive interface to generate an interrupt to the CPU upon receiving a seek end interrupt from its selected disk drive.

One jumper is required to interrupt the alternate drive interface.

One jumper is required to receive the interrupt from the alternate drive interface.

One jumper is required to sample the controller/formatter request line (select hold).

One jumper is required to transmit a request (select hold) to the controller/formatter.

DRIVE INTERFACE REGISTERS

The drive interface contains registers and counters that support software and/or hardware functions. These registers are divided into three categories.

PROGRAM REGISTERS

Program registers are accessible through software A/Q input/output commands. These registers are necessary for setting up and executing disk drive data transfer operations. Table 2-1 lists these program registers and their application. The software function codes that reference these registers are detailed below under A/Q Command Description.

HARDWARE REGISTERS

The hardware registers are not accessible by software; they are used internally by the drive interface for control of storage module system operations. These registers are described in section 3.

FILE REGISTERS

File registers are the internal drive interface working registers. There are 16 eight-bit registers addressed as 0 through F (hexadecimal). All file registers may be read by software; however, only six may be loaded directly from the CPU. They are file registers 1, 2, 3, 4, A, and B.

The first nine files (0 through 8) of the file register contain the information read from the address field of the last sector operated on following a data read or write operation. The remaining seven files contain constants or working data for the particular function being performed. The following text identifies the file registers and their application.

File 0 - Field Length

The field length file is the number of bytes, minus one, in the data field of this sector.

File 1 - Sector Address

File 1 is the actual logical sector address read from each sector's address field on the selected drive. File 1 is also used prior to initiating a read or write operation for temporarily storing the target sector address loaded by a set sector and head function.

TABLE 2-1. DRIVE INTERFACE PROGRAM REGISTERS

Register Name	Length (Bits)	Data Loaded By	Data Read By	Register Cleared By	Application
Buffer length	0-15 (16)	Function code 0	Hardware	None, self clearing	Set to number of words to be transferred between file and main memory
Current word address, lower	0-15 (16)	Function code D	Hardware	None	Set to 16-bit memory address of first data word (FWA) to be transferred between file and main memory under direct memory access control
Current word address, upper	16, 17 (2)	Function code E	Hardware	Master clear or clear drive interface function	Set to most significant two bits of first memory address for memory capacities of up to 256K
Cylinder address (file 3, 4)	0-9 (10)	Function code 5	Function code 5	None	Set to the desired cylinder address
Sector and head address (file A, B)	0-10 (11)	Function code 7	Function code 7	None	Set to desired sector and head address
Interrupt mask	1-6 (6)	Function code 8	Hardware	Master clear or clear drive interface function	Set to enable/disable drive interface interrupt conditions
Drive interface status	0-15 (16)	Hardware	Function code 8	Clear drive interface function	Contains current status and alarm conditions within the drive interface

File 2 - Head Address

File 2 is the same as file 1, except it contains the head address.

File 3 - Cylinder Address, Lower

File 3 contains the least significant eight bits of the cylinder address that was read from the address field on the selected drive. File 3 is also used to hold the software-supplied initial cylinder address following a load cylinder address function and preceding a read or write operation.

File 4 - Cylinder Address, Upper

File 4 is the same as file 3, except it contains the most significant two bits of the cylinder address.

File 5 - Flag Byte 4

This flag byte is not used by the storage module system hardware. The operating system software may use this field for a bad track/sector flag, a write protect flag, or an alternate track assignment.

File 6 - Flag Byte 3

File 6 is the same as file 5.

File 7 - Flag Byte 2

File 7 is the same as file 5.

File 8 - Flag Byte 1

File 8 is the same as file 5.

File 9

This register contains a hexadecimal 30. It is used within the drive interface as a drive ready and on cylinder status compare word for testing drive status during an autoload sequence.

File A - Target Sector Address

Preceding a data read or write, this register receives the initial target logical sector address as specified via a software set sector and head command. Following a data read or write, this register contains the address of the last sector operated on. During data read or write operations, the drive interface increments this register as the transfer extends across sector boundaries.

File B - Target Head Address

File B is the same as file A, except it contains the head address.

File C - Constant of Zero

The file C register always contains zero. It is used internally within the drive interface for testing register conditions of zero.

File D - Timeout Counter

The file D register is used internally within the drive interface to time certain commands and responses for the controller/formatter.

File E - Read/Write Retry Counter

The file E register is used during data read or write operations by the drive interface while searching for the correct sector and head to operate on. The drive interface reads the address field of 128 sectors (two revolutions) in search of the target sector and head address. If this register ever reaches a zero count during a read or write command, it indicates that the drive interface was unable to find the specified sector and head address, and the operation is aborted.

File F - Temporary Storage

File F is used during data read or write operations and for temporary storage of the read file address. If this file is read, it holds its own file address.

PROGRAMMING

The drive interface communicates with the CPU via the A/Q and direct memory access input/output channels. The A/Q channel provides the drive interface with a direct connection to the CPU's A and Q registers. All operations on the A/Q channel are initiated by execution of the software instructions input-to-A or output-from-A. The Q-register contents are always output to the drive interface along with the A-register contents for an output-from-A instruction. For an input-to-A instruction, the Q register is output, but the drive interface inputs a single word into the CPU's A register.

A/Q COMMAND CATEGORIES

The CPU A register transmits to, or receives information from, the drive interface under A/Q input/output command operations. The application of the information contained in the A register falls under the following general categories.

- Single word transfer, output from A – The CPU's A register data is used to load drive interface registers, clear control logic, initiate requests, transmit test data, and select drives as directed by specific bit assignments of the A register.
- Single word transfer, input to A – This A/Q operation is used to read status words, registers, test data, and control logic conditions into the CPU A register.
- Control, output from A – This operation initiates motion control (seek operations) to the selected disk drive's head positioning logic, based on the contents of the CPU A register.
- Data handling, output from A – This operation sets up the drive interface for initiating buffered transfers of data between the drive interface and main memory under direct memory access control.

DIRECT MEMORY ACCESS OPERATION

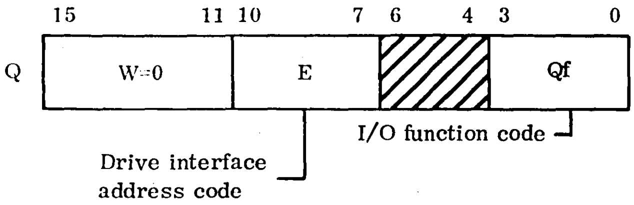
The direct memory access channel provides the drive interface with the control and data paths to the CPU's memory. This allows the drive interface to communicate with the memory for the purpose of data transfers. The direct memory access input/output channel connection with the drive interface is defined initially by a sequence of software operations. The execution of several output-from-A instructions, with the accompanying A/Q registers containing drive addressing and initiate direct memory access command information, sets up the drive interface for direct memory access operation. The drive interface internally verifies the entry into the beginning of the desired target sector requested by software.

As data becomes available, the drive interface logic through the control paths of the direct memory access channel makes a request to the CPU for a memory cycle. The drive interface link to the direct memory access channel is established for one data transfer when the CPU's memory acknowledges the drive interface's memory request. Upon receiving the memory's acceptance, the drive interface transfers a single data word to (direct memory access write) or from (direct memory access read) the CPU memory via the direct memory access data path. The drive interface's memory address register is incremented and its buffer length (word count) register is decremented by one following each data transfer. The direct memory access channel communication continues to be periodically activated by drive interface hardware until the buffer length register has been decremented to zero. At that time, all data transfers have been completed, and the drive interface's direct memory access link with CPU memory is terminated.

A/Q INPUT/OUTPUT COMMAND FORMAT

The Q register is loaded by the CPU program with the equipment number (drive interface device address) plus the drive interface's prime input/output function. The A register is loaded by the CPU program for all output operations (control, single word out, data handling) and is set to either an auxiliary input/output function or input/output function parameters, depending on the particular output command.

For A/Q input operations (single word in and status), the CPU's A register is loaded by the drive interface with the specific data that was requested by the function code contained at this time in the Q register. The Q-register format for all A/Q operations with the drive interface is as follows:



Q-REGISTER FIELDS

The E portion of the Q register defines the disk adapter's equipment address code. This code is configuration-dependent and hardware-implemented by four equipment select jumpers located on the drive interface.

The field contained in Q-register bits 0 through 3 (Qf) defines the prime input/output function to be performed by the storage module system.

All Q-register fields always are output to the drive interface upon execution by the CPU of either an output or input instruction. The A-register contents are output only to the drive interface upon execution of an output-from-A instruction.

A-REGISTER CONTENT

The A register, when used in conjunction with the Q register for output operations, contains single bits or fields (multiple bits). These bits in the A register are used to further define the function or as parameters for the operation. The definition of each bit or field in the A register varies with the particular function code in the Q register (Qf) as explained in detail under A/Q Command Description below.

For A/Q input operations (single word in and status) the definition of the bits coming into the A register is again related to the Qf.

A/Q COMMAND DESCRIPTION

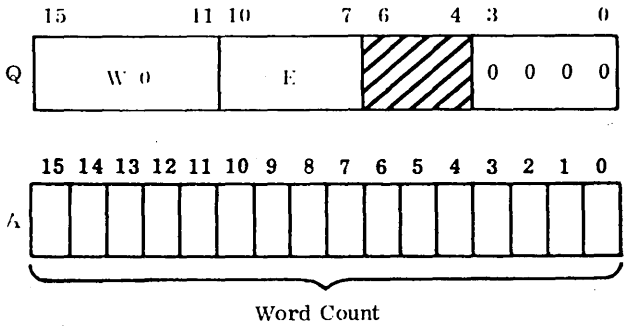
The following section lists the A/Q functions in numerical order with the output command followed by the input command of the same code number. A functional description of each command and an application for the command is provided when necessary.

Table 2-2, which lists the function codes in numerical order, may be used as an index to the functional descriptions. Note the suffix -1 following certain function codes denotes that this same code is used to perform different input/output operations. The differences are qualified by the A-register data accompanying the function; they are explained further within the functional description. Except where noted under programming considerations, all A/Q commands require a not-busy drive interface to be capable of accepting the command (refer to Programming Aids and Considerations at the end of this section).

Tables 2-3, 2-4, and 2-5 list the drive interface functions in a quick reference format by their major functional categories.

BUFFER LENGTH

Format: Output
Function Code: 0



The buffer length loads the drive interface's buffer length register with the transfer word count.

The CPU's A register must be set by the program to the total number of 16-bit words that are transferred under direct memory access control. This word count is then loaded into the drive interface's buffer length register by function code 0. During a direct memory access read or write operation, the drive interface decrements this word count -1 for every 16-bit data word transferred. When the buffer length register equals 0, the direct memory access transfer is complete.

The buffer length function is used to initialize the drive interface in preparation for initiating a direct memory access read/write, address, or data command (function codes 9 or A).

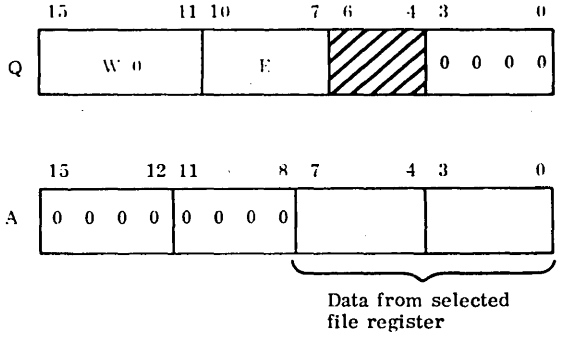
When the buffer length register is initialized to zero, any following initiate direct memory access read or write command is interpreted as a test mode read or write operation.

TABLE 2-2. DRIVE INTERFACE FUNCTION CODES (Qf)

When Q Register Bits 0-3 Equals:	Input-to-A Instruction Causes A Register to Receive:	Output-from-A Instruction Causes the Following Operation:
0	Read drive interface file register data	Set data buffer length register
1	Physical unit number	Set drive request
2	Poll status	Initiate poll
3	Select acknowledge status	Drive unit select
4	Drive echo input data	Drive echo output data
5	Cylinder address status	Load cylinder address and seek
6	Current physical sector address	Format write
7	Sector and head address status	Load sector and head address
8	Drive interface status	Director function
9	Drive status word 2	Initiate read
9-1	- -	Test mode read
A	Error correction code pattern	Initiate write
A-1	- -	Test mode write
B	Error correction code condition status	Error correction code control
C	Drive fault condition status	Status clear control
C-1	- -	Read recovery control
D	Controller/formatter status	Set first word address, lower
E	Drive status word 1	Set first word address, upper
E-1	- -	Select file address
F	Controller/formatter echo input, data	Controller/formatter echo output data

READ FILE

Format: Input
Function Code: 0



The read file reads data from the selected one of 16 eight-bit drive interface file registers.

This input function causes the contents of the drive interface file register, previously selected by a select file function, to be input to A-register bits 0 through 7.

The read file function is primarily a diagnostic or fault isolation aid in that it provides direct access to the primary working storage within the drive interface.

Refer to Select File Address below.

DRIVE REQUEST

Format: Output
Function Code: 1

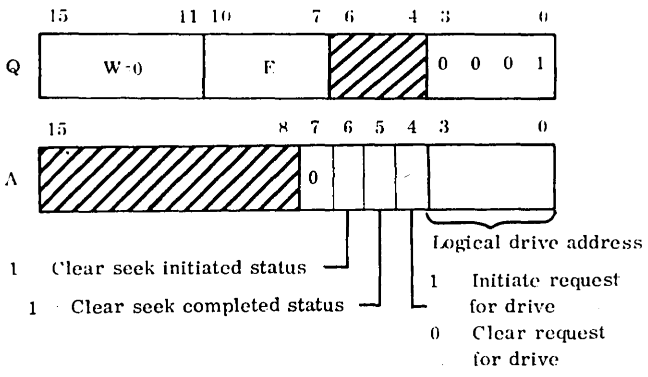

15
11 10
7 6
4 3
0
Q
W-0
F
0 0 0 1
15
8 7 6 5 4 3
0
A
0
Logical drive address
1 Clear seek initiated status
1 Clear seek completed status
1 Initiate request for drive
0 Clear request for drive

TABLE 2-3. SINGLE WORD TRANSFER, INPUT FUNCTIONS

Category	A/Q Function Name	Qf Function Code	A Register Input	Operation
Control and register read back	Poll status	2	Poll response	Inputs drive request and seek condition status
	Physical unit (confirm drive selected)	1	Physical drive number	Inputs physical drive unit number of currently selected logical drive
	Select acknowledge (drive selection reply)	3	Logical drive number	Inputs logical drive number and drive type status following a unit select function
	Cylinder address status	5	Cylinder address	Inputs contents of drive interface cylinder address register
	Current physical sector status	6	Physical sector number	Inputs physical sector number currently under heads of selected drive
	Sector and head status (last track address)	7	Sector and head address	Inputs current contents of drive interface sector/head register
Status	Disk adapter	8	Drive interface status	Inputs current drive interface status
	Control unit	D	Controller/formatter status	Inputs current controller/formatter status
	Drive status 1 (test drive errors)	E	Drive status word 1	Inputs selected drive's basic error conditions
	Drive status 2 (test drive ready)	9	Drive status word 2	Inputs basic status condition of selected drive
	Drive fault	C	Drive fault condition	Inputs fault conditions of selected drive

The drive request requests the drive from the controller/formatter when the alternate CPU is finished with it.

A-register bits 0 through 3 specify the logical drive number. Bit 4 causes the request flag in the controller/formatter associated with that logical drive number to be set or cleared. This flag can then be read by the alternate CPU via a poll command.

Bit 5 = 1 clears the seek complete status bit for the logical drive number specified by A-register bits 0 through 3. Note that this status is also cleared by initiating any seek operation on the selected drive.

Bit 6 = 1 clears the seek initiated status flag in the logical drive designated by A00 through A03. This may be used to establish initial controller/formatter conditions following a power on. Normally, seek-initiated status sets following a seek (load address) command and clears at seek end time.

This command allows two CPU's to share drives, using the following software operations example:

Through an initiate poll function, the requesting CPU determines that the drive it wants is busy performing a seek for the controlling CPU (the seek initiated status bit is set). The requesting CPU should then execute a drive request function with A04 equal to 1 (initiate request) and wait for the controlling CPU to release the drive.

Before the controlling CPU initiates a direct memory access write or read operation, it polls the drive and finds its request flag from the alternate CPU set. Prior to starting the direct memory access operation, the controlling CPU, through use of a director function (code 8), may interrupt the requesting CPU and then initiate the specific direct memory access operation. At its completion, the controlling CPU does not initiate any new operations. Prior to exiting, it sets the drive request bit if it wants the drive back or clears the request bit if it does not want the drive back. Each drive request must be done on a per-drive basis.

The drive interface accepts this command only if it is not busy and the controller/formatter is currently selected. Note that the drive request function does not select the drive; it flags a request to the alternate drive interface, via the controller/formatter, for a drive that is currently in use. The drive request and poll functions must be used by both CPUs for determining channel and disk drive activity.

TABLE 2-4. SINGLE WORD TRANSFER, OUTPUT FUNCTIONS

Category	A/Q Function Name	A/Q Function Code		Description of Operation
		Q00-Q03	A00-A15	
Drive initialization	Reset †	8	0001 (hex)	Clears all control, interrupt, and fault alarm conditions within the drive interface
	Director function	8	Interrupt enable pointers	Enables/disables interrupts within the drive interface and transmits connect/disconnect requests to the alternate drive interface and controller/formatter
	Drive request	1	Logical drive address and request flag	Flags a request for a drive to the alternate drive interface via the controller/formatter
	Initiate poll (sample channel activity)	2	Poll conditions and logical drive group	Queries what current channel activity, seek conditions, or active requests exist for a group of logical drives
	Unit select (select disk drive)	3	Logical drive address	Selects one of 16 logical drives
	Buffer length (load word count)	0	Length of buffer	Loads the drive interface buffer length register with the number of 16-bit words to be transferred.
	Set first word address, lower (set beginning buffer address)	D	Memory address of first data word	Loads the lower 16 bits of memory address into the drive interface's current word address register
	Set first word address, upper †	E	Memory address most significant bits	Loads the upper two bits of 18-bit memory address into the drive interface current word address register
	Status control † (selective status clear)	C	Status conditions to be cleared	Selectively clears general and error status conditions as directed by bits 0 through 6 in the A register
Disk addressing	Sector, head address (select track)	7	Head/sector address	Loads the desired sector and head address into the drive interface head/sector register
Control, out	Load cylinder address (seek)	5	Cylinder address	Loads the desired cylinder address in the drive interface and initiates a seek operation on the selected drive to that address
	Return to zero† (home heads)	C	0080_{16}	Returns the heads in the selected drive to cylinder address zero and clears seek error status
Data handling	Format write (zero track)	6	0000	Initializes selected track (erase)
	Write (data)	A	0000	Initiates direct memory access write data operation
	Write (address)	A	0080_{16}	Initiates direct memory access write track address operation
	Read (data)	9	0000	Initiates direct memory access read data operation
	Read (address)	9	0080_{16}	Initiates direct memory access read track address operation

† Pseudo operation; subset of basic function

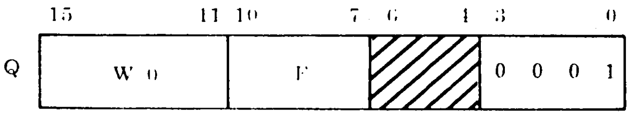
TABLE 2-5. INPUT/OUTPUT FUNCTIONS

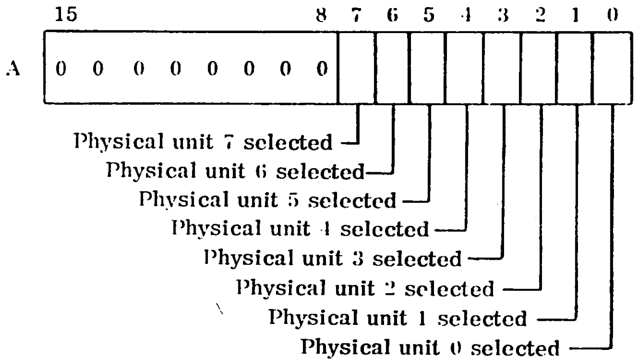
Category	A/Q Function Name	Mode	A/Q Function Code		Operation
			Q00-Q03	A00-A15	
Error recovery	Error correction code control (shift error correction code)	Out	B	Shift code	Manipulates the error correction code register bytes per A-register bit 4 or 5
	Error correction code condition status	In	B	Error correction code byte conditions	Inputs the error correction code byte condition status to the A register
	Error correction code pattern	In	A	Error correction code pattern	Inputs bits 11 through 18 of the error correction code pattern to the A register
	Read recovery control †	Out	C	Data strobe/ head offset	Outputs bits 8 through 15 of the strobe/offset word from the A register
Diagnostic test	Test write †	Out	A	0000 and buffer length = 0	Loads the drive interface first-in/first-out (FIFO) buffer; outputs 64 sequential words starting from first word address in the CPU memory to the drive interface's FIFO
	Test read †	Out	9	0000 and buffer length = 0	Inputs the drive interface FIFO buffer into 64 sequential CPU memory locations starting with first word address
	Select file address †	Out	E	Drive interface file register address	Selects one of 16 internal file registers per bits 0 through 3 of the A register
	Read file	In	0	Contents of selected file register	Inputs the eight-bit selected file register to the A register
	Controller/formatter echo output	Out	F	Test pattern	Outputs the test pattern from bits 0 through 7 of the A register to the controller/formatter
	Controller/formatter echo input	In	F	Test pattern	Inputs the test pattern from the controller/ formatter to A-register bits 0 through 7
	Drive echo output	Out	4	Test pattern	Outputs the test pattern from bits 0 through 6 (bit 7 equals 1) of the A register to the selected drive
	Drive echo input	In	4	Test pattern	Inputs the test pattern from the selected drive to bits 0 through 6 of the A register

† Pseudo operation; subset of basic function

PHYSICAL UNIT NUMBER

Format: Input
Function Code: 1





The physical unit number senses the physical drive number of the selected logical unit.

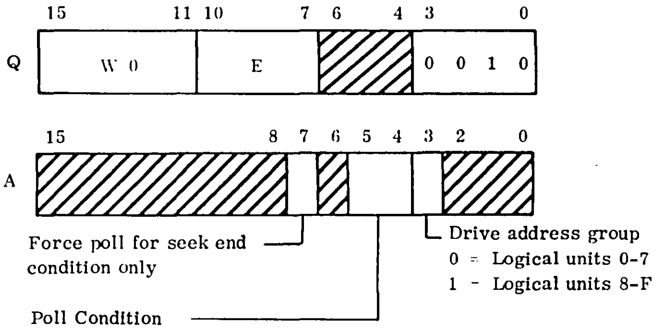
One bit is set in A-register bits 0 through 7 corresponding to the physical drive number of the logical drive unit currently selected.

Logical addresses are assigned initially to each drive by the insertion of a logical address plug. Software can cross-reference logical drive numbers against physical drive numbers by first executing a unit select (select logical unit) function (code 3) to one drive. Executing a physical unit number function then causes the physical unit number of that drive to be input to its corresponding A-register bit. This can be used within the operating system for error reporting when it is more desirable to log errors by physical drive number.

Note that although the drive interface/controller/formatter is designed to interface a maximum of eight drives, the drives can have logical addresses assigned by a manually inserted logic plug anywhere within the range of 0 to F hexadecimal.

INITIAL POLL

Format: Output
Function Code: 2



00 - Alternate channel request
01 - ID of last controlling CPU
10 - Seek end condition
11 - Seek initiated, seek end pending

The initial poll interrogates the controller/formatter's poll table for the requested condition.

This function causes the drive interface to initiate a poll command to the controller/formatter. A-register bit 3 specifies the group of drives being polled, and bits 4 and 5 specify one of four possible poll conditions (see Poll Status below).

The bit 7 force poll need be used only when the CPU does not de-select the controller/formatter following a load address (seek) command. This bit causes the controller/formatter to poll the specified group (per bit 3) of eight drives for seek end conditions only and update its internal poll table.

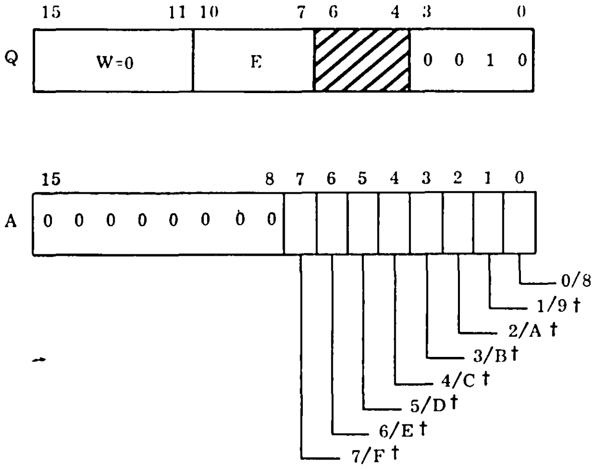
This function may be used to determine channel availability and drive activity in dual CPU configurations where time sharing of drives between CPU's must be contended with through software. Also, in single CPU with multiple drive configurations, this function allows the software to discriminate seek end conditions occurring from more than one drive.

The drive interface accepts this function if the controller/formatter is selected and the drive interface is not busy. Acceptance of this function causes the drive interface to become busy and remain busy until the poll status is available (40 microseconds nominal).

After a poll, the initial drive unit selection is lost and must be re-established via the unit select function.

POLL STATUS

Format: Input
Function Code: 2



† Drive unit numbers and A-register poll bit assignments

The poll status inputs the selected drive's operational condition in response to the poll function.

The group of drives selected by the poll function sets the assigned A-register bits to 1, if they satisfy the polled condition. The initiate poll function may specify drive group 0 through 7 or drive group 8 through F. For either case, their respective poll status is returned in A-register bits 0 to 7.

The following are poll response conditions:

- Controlling CPU ID (source usage) – If the drive's respective A-register bit in the returned poll status is a 1, this drive was last selected by CPU 1. If the assigned A-register bit is a 0, the drive was selected by CPU 0.

- Alternate channel request – If the drive's assigned A-register bit position of the poll status is set, it indicates a request for that drive from the alternate drive interface.

- Seek end condition – If the drive's respective A-register bit in the poll status is set, it indicates that this drive has completed a previously requested seek operation.

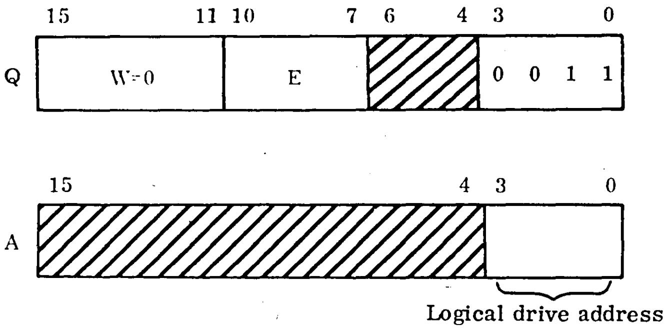
- Seek initiated – When the assigned A-register bit is set, it indicates this drive is currently executing a requested seek operation.

The poll status determines the channel and drive activity in a dual CPU configuration (see Initiate Poll above).

An initiate poll output function must precede a poll status input command. If other intervening operations are performed, requested information may be lost.

UNIT SELECT

Format: Output
Function Code: 3



The unit select function selects the logical unit number.

A-register bits 0 through 3 are used to select logical drives 0 through F hexadecimal. Executing this output function causes one of the following conditions to occur:

- If the controller/formatter is already selected by this drive interface, the drive interface selects the drive specified by this function.
- If the controller/formatter is not already selected by this drive interface, the drive interface initiates a controller/formatter select sequence only. A second unit select function must be initiated to obtain control of the desired drive.
- If the controller/formatter is currently selected by the alternate channel, the drive interface performs the select sequence of the controller/formatter only, as soon as the controller/formatter is released by the alternate channel.

Following completion of the select, the selected status is set and the selected interrupt is activated if it has been enabled previously.

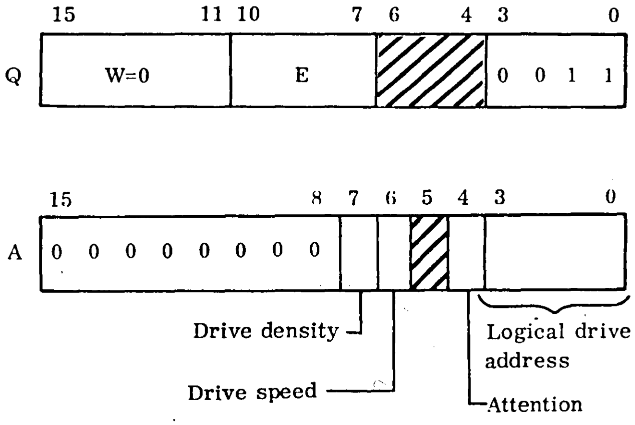
This command provides software with the capability of direct selection of logical drive units in a single CPU configuration. For dual CPU/drive interface configurations, see drive request, output function code 1, above.

A unit select output function should be followed by a select acknowledge status input command.

Although the storage module system supports a maximum of eight physical drives, they can be assigned any logical addresses of 0 through F (hexadecimal) as determined by their logical address plug (LAP).

SELECT ACKNOWLEDGE STATUS

Format: Input
Function Code: 3



Verify drive unit selection.

Logical Drive Address

A-register bits 0 through 3 indicate the storage module disk logical address that responded to the unit select function. The program should check that these bits agree with the address that was output as part of the select command.

Attention

A-register bit 4 equal to 1 indicates that the heads have been unloaded and reloaded on the selected drive, resulting from a pack change or a fault condition. When the drive comes ready, attention stays set until reset by the clear attention function (code C).

Drive Density

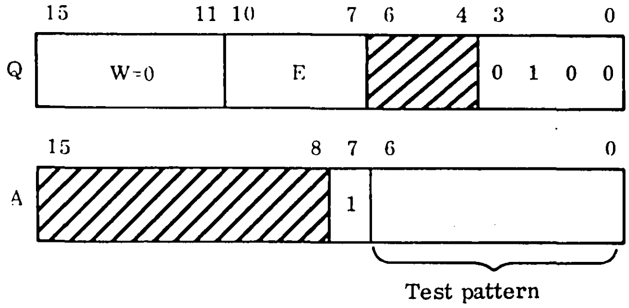
When A-register bit 7 is a 1, a 400-tracks per inch (50-megabyte) storage module disk has been selected; when it is 0, a 200-tracks per inch (25-megabyte) storage module disk has been selected.

The input command allows the CPU to verify drive selection and determine drive unit characteristics.

This status should be read following a unit select command and preceding any other input/output command; otherwise, the status may be lost in the drive interface. The drive interface accepts this input command if it is not busy; however, this status is only meaningful when preceded by a unit select command.

DRIVE ECHO OUTPUT

Format: Output
Function Code: 4



The drive echo outputs the test pattern to the selected drive unit.

The drive echo output function is a diagnostic aid that allows the CPU to verify the bus-out and bus-in data paths between the controller/formatter and the drive. A-register bit 7 must be a 1 and bits 0 through 6 may be any binary data pattern. The data that is output via this function may be input and verified by the drive echo input function.

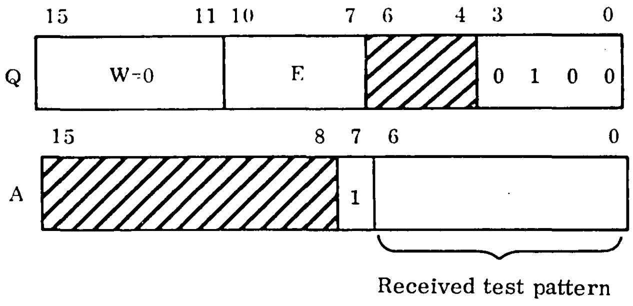
The drive echo output function is a diagnostic test of the controller/formatter/drive data paths.

This command, in conjunction with the corresponding input command, provides a functional verification of the command, status, and data paths between the drive interface, the controller/formatter, and the selected drive. Output data is transmitted to the drive, received, and echoed back to the drive interface. This test data may then be input to the CPU's A-register for verification by execution of a drive echo input command.

First, each drive to be echo-tested must have been selected by a unit select function. Additionally, the data paths between the drive interface and controller/formatter should have been verified through execution of the controller/formatter echo input/output function. Upon confirmation of the integrity of this path, the drive echo commands may be used with a reasonable degree of confidence. Note that a drive echo input command always must be preceded by a drive echo output command.

DRIVE ECHO INPUT

Format: Input
Function Code: 4



The drive echo input reads back the drive test pattern to A-register bits 0 through 7.

This command allows the CPU to input the binary data pattern that was output to the drive via the drive echo output function.

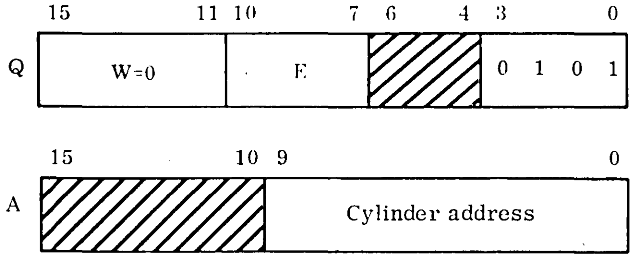
The drive echo input is a diagnostic test of the controller/formatter/drive data paths.

When used in conjunction with the drive echo output function, this command allows full echo checking of the command, status, and data paths between the drive interface, the controller/formatter, and the selected drive.

The drive interface accepts this command if it is not busy; however, the input data is meaningful only when this command is immediately preceded by the drive echo output function.

LOAD CYLINDER ADDRESS

Format: Output
Function Code: 5



The load cylinder address loads the cylinder address register and initiates a seek.

The cylinder address register in the drive interface is first loaded from A-register bits 0 through 9; a seek command to that address is then issued to the selected drive.

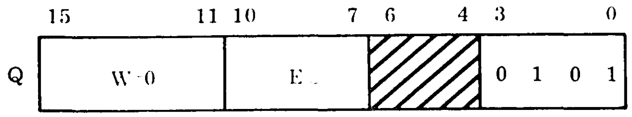
The load cylinder address positions the selected drive's heads to the desired cylinder address in preparation for track and sector selection.

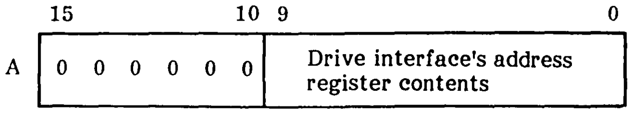
Absolute addressing is used.

The drive interface accepts this command only if it is not busy and the controller/formatter is currently selected.

CYLINDER ADDRESS STATUS

Format: Input
Function Code: 5





The cylinder address status inputs the cylinder address register contents to A-register bits 0 through 9.

Following a load cylinder address function, the cylinder address status is the same as the address that was issued by that function, irrespective of the true cylinder address of the currently selected drive. Following an RTZ (return-to-zero) seek function, the cylinder address status is set to zero.

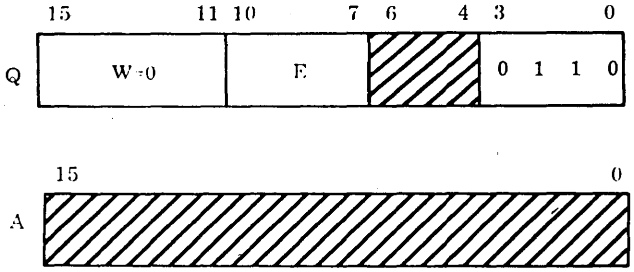
Determine the cylinder address of the drive, specifically the final address following completion of a direct memory access read or write function.

The drive interface will not accept this input command while it is busy; for example, during a read or write operation. Following completion of a read or write data operation, this status represents the last cylinder address operated on.

Care must be taken when attempting to determine the true cylinder address status of a drive. This is particularly true when doing overlapped seek operations. Following a series of load address commands to different drives, the cylinder address register contains the same information that was issued as part of the last load address function. For this case, one way to determine a drive's current cylinder address is to initiate a read address function, and then compare the information transmitted to memory with the expected address. Another method is to initiate the desired read data command, and then check the cylinder address status at its completion.

FORMAT WRITE

Format: Output
Function Code: 6



The format write erases a selected track.

This function causes the controller/formatter to write zeroes beginning at the index mark of the previously specified track. The format write operation is terminated upon detection of the next index mark.

No data is transferred between the CPU and the drive interface. The format write command is a one-track operation, and upon completion, the drive interface clears the busy status bit and sets the transfer complete status.

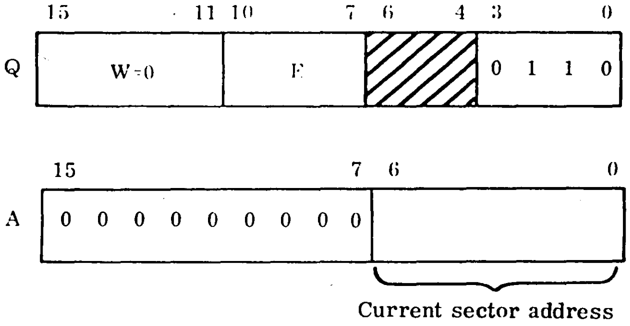
This command is required for pack initialization and setting up the proper head gaps and synchronous patterns.

Note that a format write destroys any data or address information previously recorded on the track.

The drive interface accepts this command if the controller/formatter is selected and the drive interface is not busy.

CURRENT SECTOR STATUS

Format: Input
Function Code: 6



The current sector status inputs the current physical sector number.

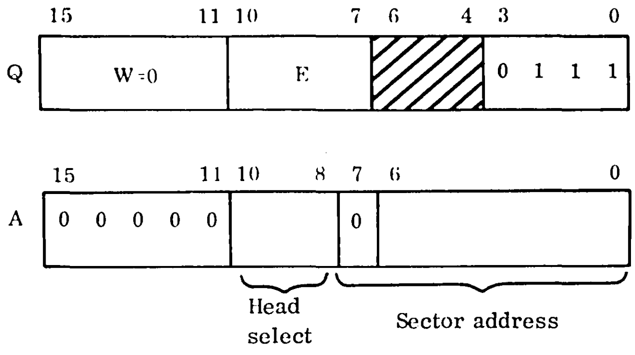
This input command causes the number of the current physical sector under the read/write heads of the selected drive to be input to the CPU A-register.

The current sector status allows software to perform rotational position sensing.

The drive interface accepts this command only if it is not busy and the controller/formatter is currently selected. Note that sectors are numbered consecutively from 0 to N beginning at the index mark. N is one less than the total number of sectors (dependent upon the sector plug installed in the drive) per track, for example, 64 sectors numbered as sectors 0 through 63.

SECTOR AND HEAD ADDRESS

Format: Output
Function Code: 7



The sector and head address function loads the drive interface's sector/head address register.

The desired sector number (maximum of 128) is set in A-register bits 0 through 6 along with the head select code in A-register bits 8 through 10. Upon execution, the A register is output to the drive interface's sector/head address register, and a head select command is issued to the selected drive.

This function, preceded by a load cylinder address function (code 5), provides electronic positioning of the selected drive's head for a specific track and sector access. A direct memory access read or write data operation may then be initiated.

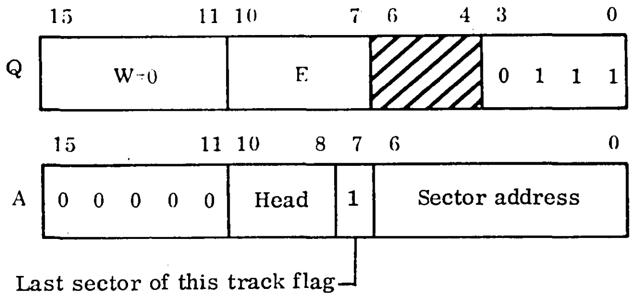
The drive interface accepts this command only if it is not busy and the controller/formatter is currently selected.

Selection of the sector and head address should be considered part of the direct memory access read or write operation and not part of the load address/seek operation.

For direct memory access read or write address functions, the sector address field in the A register is ignored, since the data transfer always starts from the index mark that precedes physical sector number 0.

SECTOR AND HEAD STATUS

Format: Input
Function Code: 7



The sector and head status input command causes the sector and head address status to be input to the CPU's A register as shown above. Following a sector and head output function, the sector and head status is the same as that issued as part of the output function. Following an RTZ seek function, the sector and head status is set to 0.

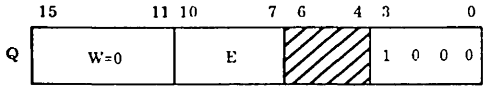
If bit 7 of this status word equals a 1, this sector is the last logical sector of the track. This was established by software at the time the track was originally formatted with a write address function.

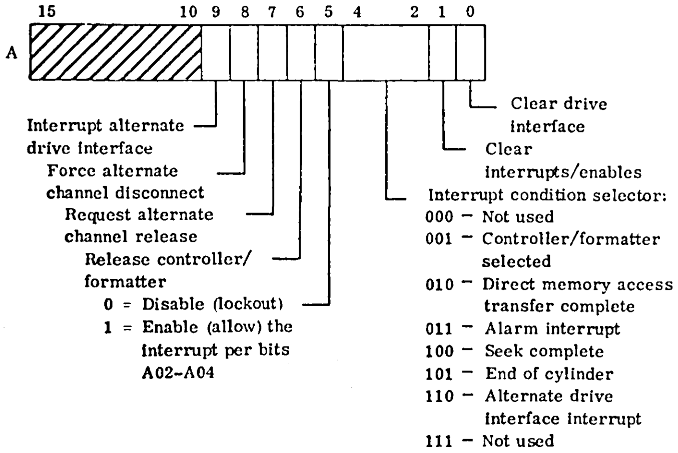
Determine the sector/head that was last addressed by software or the last sector/head address referenced for data.

The drive interface will not accept this input command while it is busy (i.e., during a read or write operation). Following completion of the read or write data operation or the occurrence of any error causing termination, this status represents the last head and sector address that was operated in. For current sector status, refer to input function code 6.

DIRECTOR FUNCTION

Format: Output
Function Code: 8





15
10
9
8
7
6
5
4
2
1
0
A
Interrupt alternate drive interface
Force alternate channel disconnect
Request alternate channel release
Release controller/ formatter
0 = Disable (lockout)
1 = Enable (allow) the interrupt per bits A02-A04
Clear drive interface
Clear interrupts/enables
Interrupt condition selector:
000 – Not used
001 – Controller/formatter selected
010 – Direct memory access transfer complete
011 – Alarm interrupt
100 – Seek complete
101 – End of cylinder
110 – Alternate drive interface interrupt
111 – Not used

The director function sets up the interrupt mask in the drive interface and provides channel control to the controller/formatter per A-register bits 6 through 9.

Clear Drive Interface

The A00 = 1 function bit clears all controls, interrupt requests, interrupt responses, and alarm conditions within the disk adapter. The controller/formatter is de-selected.

Clear Interrupts/Enables

The A01 = 1 function bit clears all interrupts, resets the interrupt mask register, and disables all further interrupts.

A02 through A04 Interrupt Condition Selector Code

All storage module system conditions that result in the generation of an interrupt to the CPU may be selectively enabled or disabled. Each interrupt condition is individually selected, as required, by the following codes placed in A-register bits 2 through 4 and accompanied by an A/Q director function. Note that a clear drive interface function (A00 = 1) or a clear interrupt function (A01 = 1) disables all the conditions from causing interrupts but still allows the condition to set its respective status bit.

A02 through A04 = 001. When the controller/formatter is selected, this interrupt enable allows the drive interface to generate an interrupt to the CPU whenever the controller/formatter is first selected by this drive interface.

A02 through A04 = 010. When the direct memory access transfer is complete, this interrupt enable allows the drive interface to generate an interrupt to the CPU when it has completed a data transfer with the selected drive for a read data, write data, read address, write address, or format write operation.

A02 through A04 = 011. The alarm interrupt enable allows the drive interface to generate an interrupt to the CPU when any alarm condition exists. Refer to Alarm Conditions at the end of this section.

A02 through A04 = 100. The seek complete interrupt enable allows the drive interface to generate an interrupt to the CPU when any one of the drives selected by this drive interface has completed a seek operation. A seek end condition occurs for every load address function, RTZ seek function, or head position offset function.

A02 through A04 = 101. The end-of-cylinder interrupt enable allows the drive interface to generate an interrupt to the CPU when a data or address read or write operation attempts an access beyond the last sector of the last track on the selected drive.

A02 through A04 = 110. The alternate drive interface interrupt enable allows the drive interface to generate an interrupt when it is interrupted by the alternate drive interface in a dual CPU configuration.

A05 = 1, Enable/Disable Interrupts

If A-register bit 5 equals 1, the drive interface is allowed to interrupt the CPU for the condition selected by bits A02 through A04. If A-register bit 5 equals 0, the interrupt, as selected by A02 through A04, is locked out (disabled).

Release Controller/Formatter

A-register bit 6 equal to 1 causes the drive interface to disconnect from the controller/formatter, thereby allowing the alternate channel to gain access or to allow the controller/formatter to poll for seek end conditions.

Request Alternate Channel Release

A-register bit 7 equal to 1 causes the controller/formatter request status bit to be set in the alternate channel's controller/formatter status word. It is used to request the alternate channel to release the controller/formatter.

Force Alternate Channel Disconnect

A-register bit 8 equal to 1 forces the alternate drive interface to unconditionally release control of the disk subsystem. If a data transfer is in progress, it is terminated as soon as the controller/formatter recognizes the force release operation. The controller/formatter is not selected by this drive interface, however, until a unit select command is first issued.

This function bit must be used cautiously, since the termination of the alternate channel is not well controlled. The primary application of this function bit is to force the alternate channel to release when it is determined that the alternate drive interface is otherwise unwilling or unable to release due to some error condition or hardware failure.

Interrupt Alternate Drive Interface

A-register bit 9 equal to 1 causes the drive interface to generate an interrupt signal to the alternate drive interface in a dual CPU configuration.

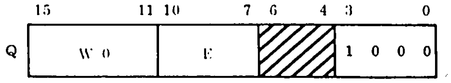
A10 through A15

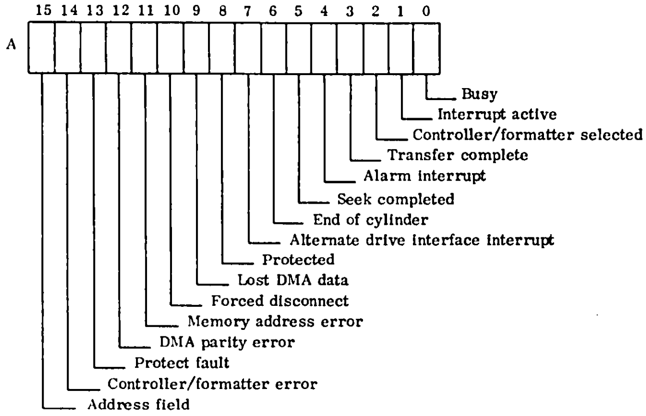
Not used

The drive interface accepts a director function command at any time.

DRIVE INTERFACE STATUS

Format: Input
Function Code: 8





Busy

A-register bit 0 equal to 1 indicates that the drive interface and/or controller/formatter is presently involved in the performance of a previously requested operation. The busy status is set when the drive interface accepts any read or write operation, a format write command, or a seek or an initiate poll command. Busy is cleared when the operation is completed, no longer involves the drive interface, or an abnormal condition is detected that aborts the operation. Busy also is cleared by a clear drive interface command (director function and A-register bit 0 equals 1). Refer to Programming Aids and Considerations at the end of this section for additional information on drive interface busy status.

Interrupt Active

A-register status bit 1 equal to 1 indicates that a selected interrupt condition has occurred. Status bits A02, A03, A04, A05, A06, and A07 define the interrupt that occurred. The interrupt status bit is cleared by a clear drive interface or an interrupt condition selection director function with A-register bit 5 equal to 0.

Controller/Formatter Selected

A-register status bit 2 equal to 1 indicates that the controller/formatter is currently selected by this drive interface. It is set following completion of the controller/formatter select sequence. This bit is cleared by a clear drive interface command or a release command (director function and A-register bit 0 or 6 equals 1).

Transfer Complete

A-register status bit 3 equal to 1 indicates that the drive interface has completed a format write, read data, write data, read address, or write address operation. Setting this bit coincides with clearing of busy status. The transfer complete status bit is cleared by a clear drive interface, clear interrupt director function, or another read or write command.

Alarm Interrupt

A-register status bit 4 equal to 1 indicates that one or more of the following abnormal conditions has been detected by the drive interface:

Lost direct memory access data

Forced disconnect

Memory address error

Direct memory access parity error

Protect fault

Controller/formatter error

The error condition detected is indicated by A-register status bits 9 through 14. The alarm status bit is cleared by a clear drive interface or clear interrupt director function.

Seek Completed

A-register status bit 5 equal to 1 indicates that one or more drives has completed a seek operation that was requested from this drive interface. Following a seek end condition, the CPU should initiate a polling operation to determine which drive has completed the seek. The seek end status bit is cleared by a clear drive interface or clear interrupt director function.

End of Cylinder

A-register status bit 6 equal to 1 indicates that the data or address read or write operation has been suspended, because it extends beyond the last sector of the last track of the current cylinder on the selected drive. When the drive interface detects this condition, the data transfer is terminated, and the busy status is cleared. Since the buffer length register (word count) does not equal zero, the transfer complete status bit is not set. In order to continue the information transfer, cylinder boundaries must be crossed. This can be accomplished by issuing a seek command. After repositioning the heads to the next cylinder address and setting the desired sector and head address, another read or write command may be issued. The end-of-cylinder status bit is cleared by a clear drive interface or clear interrupt director function.

Alternate Drive Interface Interrupt

A-register status bit 7 equal to 1 indicates that the drive interface was interrupted by the other drive interface in a dual CPU configuration. This status is only true if this interrupt is enabled prior to receiving the interrupt signal from the other drive interface. The alternate drive interface interrupt status bit is cleared by a clear drive interface or clear interrupt director function.

Protected

A-register status bit 8 equal to 1 indicates that the drive interface program protect jumper is in and is therefore operating in a protected mode.

Lost DMA Data

A-register status bit 9 equal to 1 indicates that the CPU's direct memory access channel was unable to keep up with the information transfer rate of the controller/formatter and disk drive. Detection of this error causes the read or write operation to terminate at the end of the current sector. The lost data status bit is cleared by a clear drive interface director function.

Forced Disconnect

A-register status bit 10 equal to 1 indicates that the alternate channel has executed a force release function. The force release function causes this drive interface to disconnect from the controller/formatter, allowing the alternate channel to gain access. Any operation in progress is terminated immediately. The forced disconnect status bit is cleared by a clear drive interface director function or a controller/formatter select command.

Memory Address Error

A-register status bit 11 equal to 1 indicates that the drive interface has attempted to address a nonexistent CPU memory location (i.e., beyond the system memory available) during a read or write data or address operation. Detection of this error causes the read or write operation to terminate at the end of the current sector. The memory address error status bit is cleared by a clear drive interface or clear interrupt director function.

DMA Parity Error

A-register status bit 12 equal to 1 indicates that the drive interface has received a parity error signal from the CPU direct memory access channel. Detection of this error causes the operation to terminate at the end of the sector being operated on. The direct memory access parity error status bit is cleared by a clear drive interface or clear interrupt director function.

Protect Fault

A-register status bit 13 equal to 1 indicates that an unprotected drive interface operation attempted to write into a protected CPU memory location. The memory inhibits the attempted write. This error causes the operation to terminate at the end of the sector being operated on. The protect fault status bit is cleared by a clear drive interface or clear interrupt director function.

Controller/Formatter Error

A-register status bit 14 equal to 1 indicates that the disk adapter has detected an abnormal condition during a command sequence or data transfer operation with the controller/formatter. To determine the error condition, the CPU must read drive interface and/or controller/formatter status. The controller/formatter error status bit is cleared by either a clear drive interface director function or clear interrupt director function.

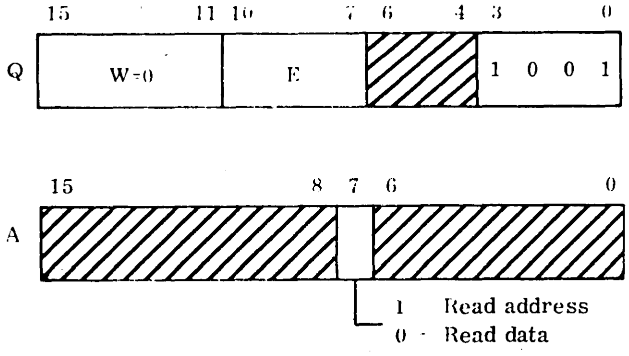
Address Field

When the drive interface is performing a read or write operation, A-register bit 15 equal to 1 indicates that the drive interface is currently operating in the address field of a sector. For other operations, or when the drive interface is not busy, this bit indicates that the last drive interface read or write operation was performed on the sector address field.

The drive interface accepts a drive interface status function at any time.

DISK READ

Format: Output
Function Code: 9



This function causes the drive interface to initiate an information transfer from the selected drive to CPU memory only if the drive interface's buffer length register contents are initially nonzero. The actual information to be read is determined by A-register bit 7.

In the case of a read address function (A07 = 1), the drive interface reads the first five words (address field) per physical sector beginning at the index mark of the previously specified track. The total number of address fields that is read is a function of the buffer length register contents.

For a read data function (A07 = 0), the number of data words within a sector is determined by the drive interface from the length bytes word in the address field of that sector. The transfer begins at the previously specified cylinder, head, and logical sector address.

Previously, the buffer length register within the drive interface was loaded with a nonzero word length by the load buffer length function, and the first word address was loaded by the set first word address, lower and upper functions. When the data transfer begins, the first word address is incremented in the current word address register, and the buffer length register is decremented as each word is transferred to the CPU's memory.

When the buffer length register finally equals 0, the drive interface ends the information transfer to CPU memory, clears the busy status, and sets the transfer complete interrupt and status bit. If the data or address information transfer ends within a sector, the drive interface continues reading to the end of the sector. The controller/formatter then determines if the error correction code word is correct. If the transfer extends beyond the last sector in a cylinder, the drive interface suspends the data transfer, clears the busy status, sets the end of cylinder status bit, and if allowed, interrupts the CPU. The program may continue the data transfer by repositioning the heads, setting the desired sector and head address, and issuing another direct memory access read command. If information transfer is attempted beyond the last cylinder in a file, a drive seek error occurs.

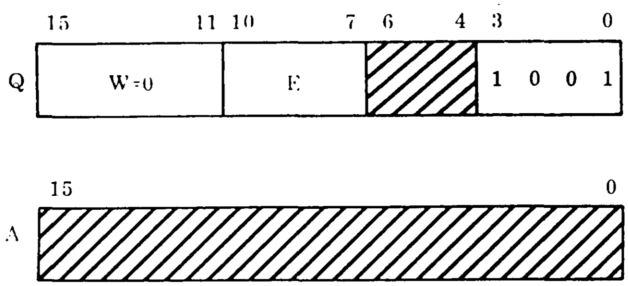
The drive interface accepts this function if it is not busy and the controller/formatter is selected.

The drive interface becomes busy upon acceptance of this function and remains busy until the end of the last sector operated on, until the end of cylinder condition is detected, or until an alarm condition is detected.

Note that a read operation may be as small as one CPU word or as large as one cylinder, depending on the specified buffer length and the starting head and sector address. Refer to Alarm Conditions at the end of this section for additional information related to this function.

TEST MODE READ

Format: Output
Function Code: 9-1



The test mode read function reads the drive interface's 64-word buffer into the CPU's memory.

Any direct memory access read function initiated with the buffer length register containing 0 causes the drive interface to transmit the 64 words stored in its first-in/first-out buffer memory to the CPU memory starting at the address specified by the first word address register. Effectively, this operation causes the drive interface to echo back the 64 data words written into the first-in/first-out buffer memory by a test mode write command. When the transfer is complete, the drive interface busy status is cleared and the transfer complete status is set.

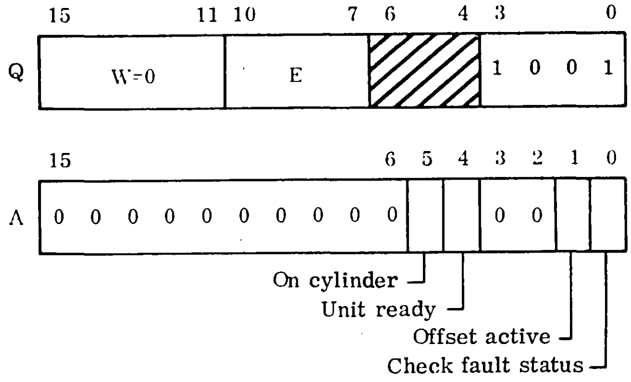
See Test Mode Write below (function code A-1).

A test mode write operation must precede a test mode read operation.

If other operations are performed between a test mode write/read sequence, the data contained in the drive interface's buffer can be altered.

DRIVE STATUS 2

Format: Input
Function Code: 9



This command inputs the currently selected drive's operational status to the CPU A register.

Check Fault Status

A-register bit 0 equal to 1 indicates that there is one or more bits set in the drive fault status word (see Drive Fault Status, function C, below).

Offset Active

A-register bit 1 equal to 1 indicates that the head position is offset plus or minus from the nominal head position on the currently selected drive. Write operations are not permitted on drive units with heads in an offset condition.

Unit Ready

A-register bit 4 equal to 1 indicates that the drive is up to speed, the heads are loaded, and no fault condition exists within the storage module disk.

On Cylinder

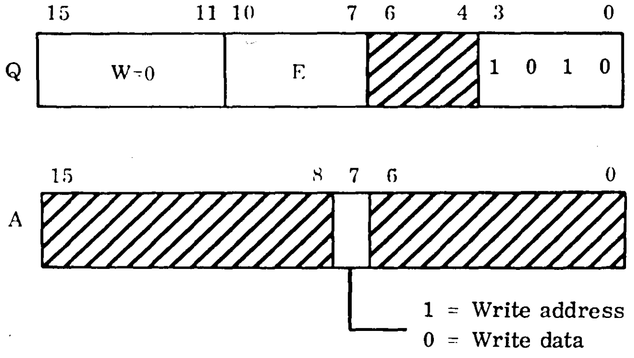
A-register bit 5 equal to 1 indicates that the heads are stationary (i.e., not seeking) at the previously specified cylinder address.

The drive status 2 function allows a quick program check of the selected drive's basic status conditions.

This status is valid any time following the unit select command.

DISK WRITE

Format: Output
Function Code: A



This function causes the drive interface to initiate an information transfer to the selected drive only if the drive interface's buffer length register contents are initially nonzero. The type of information to be written on the disk is determined by this command's A-register bit 7.

In the case of a write address function (A07 = 1), the drive interface writes five words (from CPU memory) per physical sector beginning at the index mark of the previously specified track.

For a write data function (A07 = 0), the number of words written per sector (from CPU memory) is determined by the drive interface from the length bytes word in the address field of that sector.

The total words transferred to the file from memory for write address or data operations is a function of the buffer length register's contents. The transfer begins at the previously specified cylinder, head, and sector address.

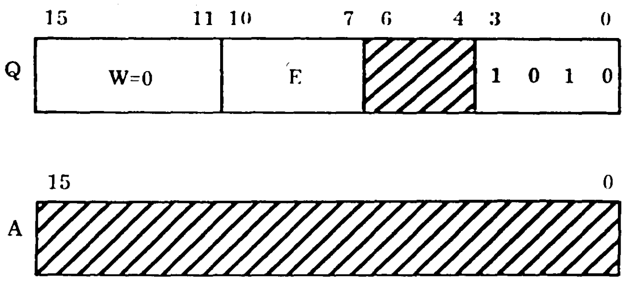
The write address and write data commands operate in basically the same manner as the read address and read data commands. The principle difference is that if the information transfer ends within the address or data field of a sector, the drive interface fills the remainder of the address or data field with 0's and then causes the seven error correction code bytes to be written as the final data.

The drive interface accepts this command only if it is not busy and the controller/formatter is currently selected. A write operation may be as small as one CPU word or as large as one cylinder depending on the specified buffer length and the starting head and sector address.

Additional information relating to the direct memory access write function is contained under Alarm Conditions and Programming Aids and Considerations at the end of this section.

TEST MODE WRITE

Format: Output
Function Code: A-1



The test mode write function loads the drive interface's 64-word first-in/first-out buffer from the CPU's memory.

Any direct memory access write function with the initial buffer length specified as 0 causes the drive interface to transfer 64 words from CPU memory starting at the address (first word address) specified by the current word address register. This data is stored in the drive interface's 64-word first-in/first-out buffer memory. When this transfer is complete, the drive interface busy status is cleared and the transfer complete status is set. Data is not transferred to the controller/formatter during this operation.

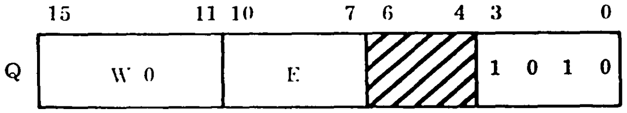
The test mode write function has a self-test and diagnostic capability.

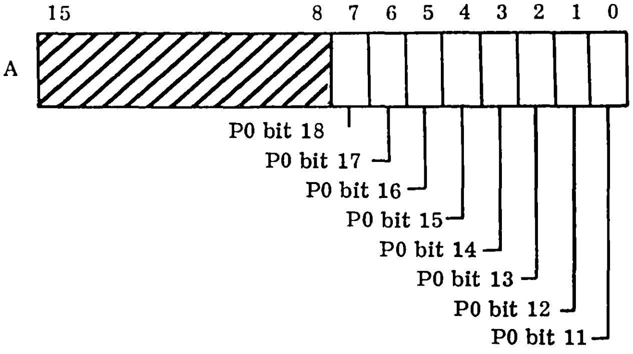
The program may check the input data against that output through execution of a test mode read function, plus check several error conditions through sampling interrupts and status.

Note that between the test mode write and read commands, a clear drive interface director function must not be issued since it causes the data in the first-in/first-out buffer memory to be destroyed.

ERROR CORRECTION CODE PATTERN

Format: Input
Function Code: A





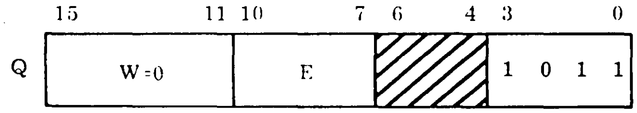
This function inputs the error correction code pattern (error burst) and the controller/formatter's P0-register bits 11 through 18 to A-register bits 0 through 7.

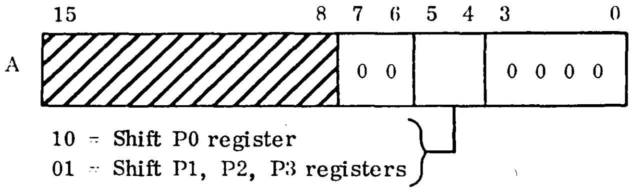
Following detection of an error correction code error through the program status of the controller/formatter-drive interface, this function is used as part of a software error correction routine for possible correction of disk read data. Refer to the FA7A8/FA727 SMP Controller/Formatter Hardware Maintenance Manual for additional information.

The drive interface accepts this command only if it is not busy and the controller/formatter is currently selected.

ERROR CORRECTION CODE CONTROL

Format: Output
Function Code: B





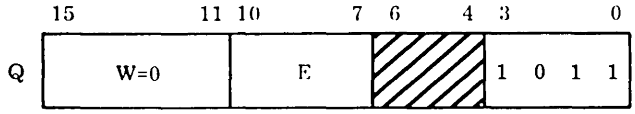
The error correction code control function causes the controller/formatter to shift its error correction code polynomial registers as defined by A-register bits 4 and 5.

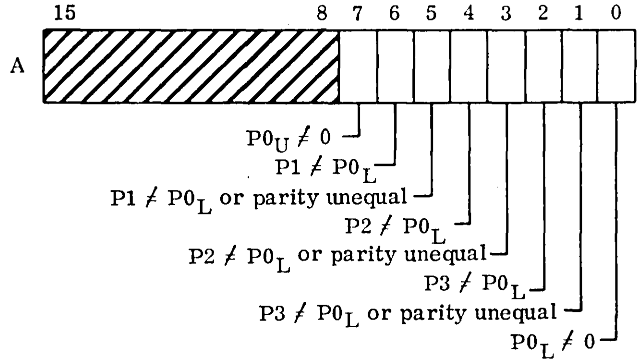
Upon detection of an error correction code error, this command is used as part of the software error correction algorithm to determine the error position within the data stream. Refer to the FA7A8/FA727 SMD Formatter/Controller Hardware Maintenance Manual for additional information.

The drive interface accepts this command only if it is not busy and the controller/formatter is currently selected.

ERROR CORRECTION CODE CONDITION STATUS

Format: Input
Function Code: B




15
8 7 6 5 4 3 2 1 0
A
P0U ≠ 0
P1 ≠ P0L
P1 ≠ P0L or parity unequal
P2 ≠ P0L
P2 ≠ P0L or parity unequal
P3 ≠ P0L
P3 ≠ P0L or parity unequal
P0L ≠ 0

This function inputs the error correction code polynomial register relationships to the A register.

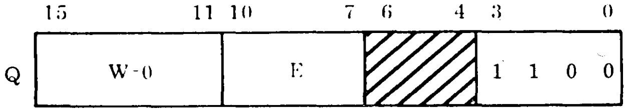
This input command causes the error correction code condition status to be input to the CPU A register. The controller/formatter's P0 lower register (bits 0 through 10) is compared against the P1, P2, and P3 registers for an all zero or all nonzero condition. The results of that comparison are reflected in A-register bits 0 through 7.

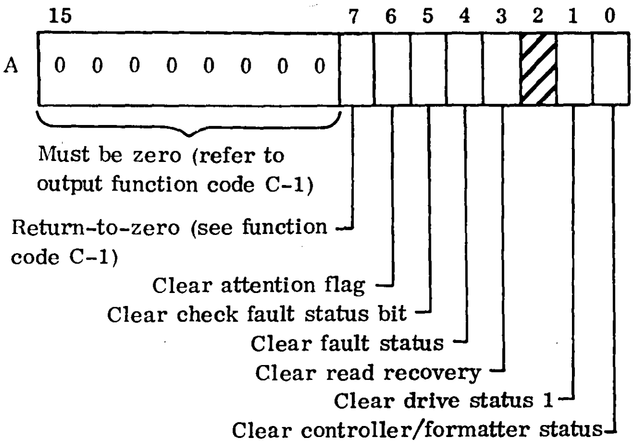
Upon detection of an error correction code error, the software error correction algorithm uses this input function following polynomial register manipulation by the error correction code control function

The drive interface accepts this command only if it is not busy and the controller/formatter is currently selected.

STATUS CONTROL

Format: Output
Function Code: C




15
7 6 5 4 3 2 1 0
A
0 0 0 0 0 0 0 0
Must be zero (refer to output function code C-1)
Return-to-zero (see function code C-1)
Clear attention flag
Clear check fault status bit
Clear fault status
Clear read recovery
Clear drive status 1
Clear controller/formatter status

Clear status words, conditions, and flags.

Clear Controller/Formatter Status

A-register bit 0 equal to 1 causes the controller/formatter status to be reset to 0.

Clear Drive Status 1

A-register bit 1 equal to 1 causes the drive status 1 word to be reset to 0.

A02

Not used

Clear Read Recovery

A-register bit 3 equal to 1 causes the head position offset and timing strobe early/late conditions to be reset to their normal values (see Read Recovery, code C-1, below).

Clear Fault Status

A-register bit 4 equal to 1 causes the drive fault status word to be reset to 0.

Clear Check Fault Status Bit

A-register bit 5 equal to 1 causes the check fault status bit within the drive status 1 and 2 words to be reset.

Clear Attention Flag

A-register bit 6 equal to 1 causes the attention bit within the select acknowledge status word to be reset.

Return-to-Zero

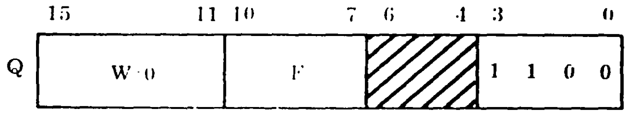
A-register bit 7 equal to 1 is used to command a return-to-zero seek on the selected drive. Additionally, the cylinder, head, and sector address registers within the disk adapter and any drive offset conditions are cleared.

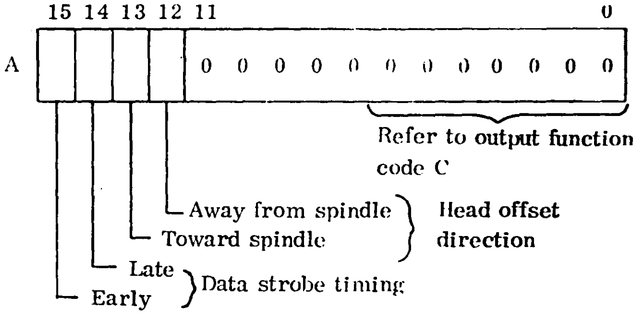
The status control function provides selective clearing of fault, status, and control flags.

The drive interface accepts this command only if it is not busy and the controller/formatter is currently selected.

READ RECOVERY

Format: Output
Function Code: C-1





The read recovery function allows software control of the drive head position offset and read timing strobe as defined by A-register bits 12 to 15. If any of these control bits are set, the lower 12 bits of the A register are ignored.

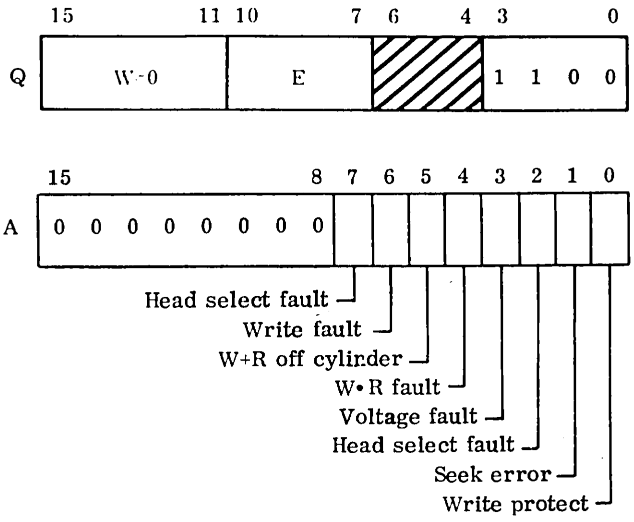
The offset and timing conditions are maintained within the drive until altered via this command or cleared via bit A03 of the status control function or until any seek operation is initiated.

The read recovery function is intended to be an aid in recovering marginal data.

The drive interface accepts this command only if it is not busy and the controller/formatter is currently selected. The head offset or the RTZ seek function operate like any other seek command and result in the normal seek end conditions at completion. A head offset results in a delay of on-cylinder and seek end status for a period of 3.2 millisecond maximum (2.7 millisecond nominal). The maximum time for the heads to move from forward to reverse offset or vice-versa does not exceed 7 milliseconds. Data may not be written while in the offset mode.

DRIVE FAULT STATUS

Format: Input
Function Code: C



The drive fault status function inputs the selected drive's fault condition status to the A register.

Write Protect

A-register bit 0 equal to 1 indicates that the selected drive is write-protected. It is set when head alignment maintenance is being performed during fault conditions that inhibit the writer or when write protection is purposely enabled on the drive.

Seek Error

A-register bit 1 equal to 1 indicates that a seek error condition has occurred due to any of the following: the drive was unable to complete a head move within 500 milliseconds, the carriage has moved to a position outside the recording field, or a cylinder address greater than the physical maximum for the selected drive. A seek error may only be cleared by a return-to-zero command (function code C) that returns the heads to cylinder zero and sets the on-cylinder status signal to the drive interface.

Head Select Fault

A-register bit 2 equal to 1 indicates that more than one head has been selected. Detection of this fault inhibits disk write operations on this drive.

Voltage Fault

A-register bit 3 equal to 1 indicates a below normal voltage is or was present for either positive or negative voltages. Detection of a voltage fault inhibits writing and generates an automatic retraction of the heads.

W·R Fault

A-register bit 4 equal to 1 indicates that the drive's write gate logic is on at the same time as its read gate. Detection of this fault inhibits write operations on this drive.

W+R Off Cylinder

A-register bit 5 equal to 1 indicates that a read or write operation has been attempted, and the heads were not on cylinder. Detection of this fault inhibits write operations on this drive.

Write Fault

A-register bit 6 equal to 1 indicates the loss of write current at the heads. The circuitry employs a fault technique to detect if the head coils are shorted or open. Detection of this fault inhibits writing.

Head Select Fault

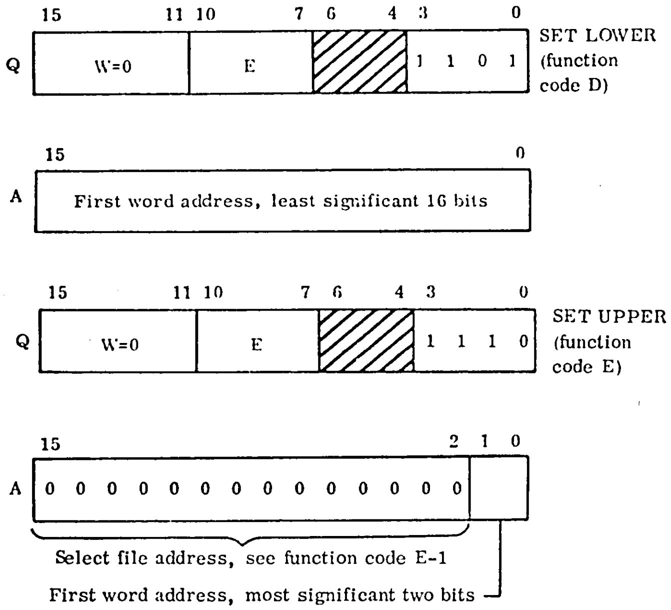
A-register bit 7 equal to 1 indicates that a head select address greater than four has been requested.

The program normally, upon detecting the check fault status flag in the drive status word 1 (bit 5) or drive status word 2 (bit 0), issues this command for definitive error information.

The drive interface accepts a drive fault status input command if the controller/formatter is selected and the drive interface is not busy.

SET FIRST WORD ADDRESS, LOWER OR UPPER

Format: Output
Function Codes: D and E



15
11 10
7 6
4 3
0
Q
W=0
E
1 1 0 1
SET LOWER (function code D)
15
0
A
First word address, least significant 16 bits
15
11 10
7 6
4 3
0
Q
W=0
E
1 1 1 0
SET UPPER (function code E)
15
2 1 0
A
0 0 0 0 0 0 0 0 0 0 0 0 0 0
Select file address, see function code E-1
First word address, most significant two bits

The set first word address, lower or upper, function loads the drive interface's current word address register with the first data word's main memory address (first word address).

The A-register bits, 0 through 15 for a set lower function and bits 0 and 1 for a set upper function, collectively contain the starting address in the data buffer area main memory. The set upper and lower functions cause their respective A-register bits to be output to the 18-bit current word address register in the drive interface.

The set first word address, lower or upper, function is used to initialize the drive interface to prepare for initiating a direct memory access read/write address or data function (function codes 9 or A).

Upon receiving a disk read or write command, the contents of the drive interface's current word address register points to the first main memory address (first word address) where the data transfer between the disk and memory must begin.

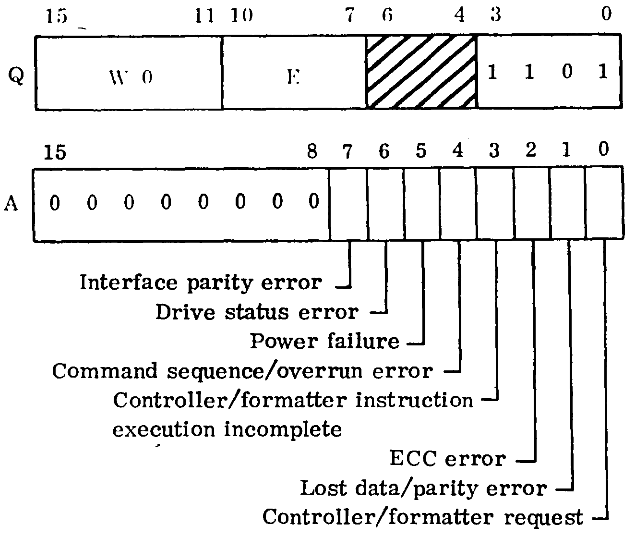
The set upper function must be used to load the drive interface's current word address register when accessing memory sizes above 65K (FFFF$_{16}$). This allows the drive interface to directly address up to 256K (3FFFF$_{16}$) of memory.

The set upper function (function code E) is a dual command. The field of A-register bits 0 through 3 and 7 through 15 are used by the select file function (code E-1) within the drive interface to enable reading or writing into 1 of 16 file registers.

Note that only the upper two bits of the current word address register are set to 0 with a master clear or clear drive interface function command.

CONTROL UNIT STATUS

Format: Input
Function Code: D



The control unit status function inputs the controller/formatter status to the A register.

Controller/Formatter Request

A-register bit 0 equal to 1 indicates that the alternate channel (another CPU via its associated drive interface) has requested this CPU/drive interface to release control of the controller/formatter.

Lost Data/Parity Error

A-register bit 1 equal to 1 indicates that a data byte has been lost due to improper synchronization of the sync-in and sync-out control signals between the drive interface and controller/formatter during a read or write operation or that the controller/formatter has detected a parity error within the error correction code bytes during a read or write.

ECC Error

A-register bit 2 equal to 1 indicates that the controller/formatter has detected an error correction code error on a read data, read address, or write data (during the drive interface's internal read of the sector address field) operation. If the error occurred while reading the sector address field, drive interface status bit A15 also is set.

Controller/Formatter Instruction Execution Incomplete

A-register bit 3 equal to 1 indicates that the controller/formatter has gone into a default condition caused by the controller/formatter failing to generate a normal end or check end interface signal to the drive interface within 50 millisecond after receiving an input/output command.

Command Sequence/Overrun Error

A-register bit 4 equal to 1 indicates that a read or write gap B (data field) command has been received at the controller/formatter without a prior read gap A (address field) command for that sector. This error indicates an internal drive interface or controller/formatter command decode problem.

This bit may also indicate that a read or write address or data command was received too late with respect to sector gaps A and B timing. This error probably indicates a drive interface malfunction.

Power Failure

A-register bit 5 equal to 1 indicates that the controller/formatter has detected a low voltage condition. If the drive interface/controller/formatter is transferring data or processing any input/output command, the operation terminates immediately upon detection of the power failure and a drive interface alarm condition is generated. Following this, the program has approximately 2 milliseconds to read status and/or shut down the disk subsystem before the controller/formatter becomes inoperable. This status bit is cleared automatically when power is restored to normal.

Drive Status Error

A-register bit 6 equal to 1 indicates that a fault exists within one of the drive status words. The program should read the drive status to determine the conditions causing the fault.

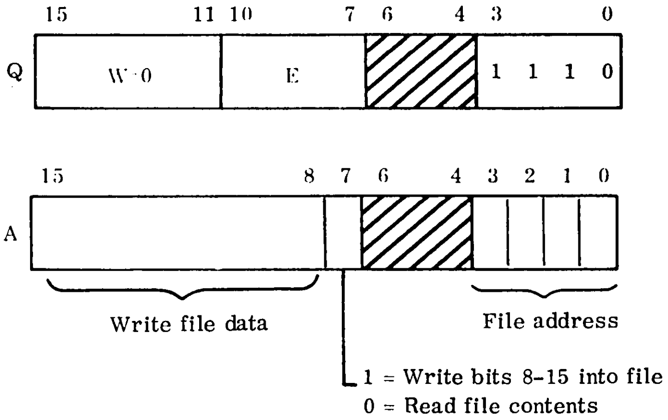
Interface Parity Error

A-register bit 7 equal to 1 indicates that the controller/formatter has detected a parity error on the bus out or tag bus lines between the controller/formatter and drive interface.

The drive interface accepts a controller/formatter status input command if the controller/formatter is selected and the drive interface is not busy.

SELECT FILE ADDRESS

Format: Output
Function Code: E-1



This function command causes two distinctly separate operations.

When A-register bit 7 is a 0:

- A-register bits 0 and 1 are loaded into the two most significant bit positions of the first word address register (i.e., 2 and 2), in a manner identical to the set first word address lower function command. This allows the drive interface to directly address up to 256K words of CPU memory for disk input/output operations. Note that these two bits of the first word address register are set to 0 with a master clear or clear drive interface function command.

- A-register bits 0 through 4 are stored internally and are used in conjunction with the read file command to specify one of 16 internal registers to be read. This is primarily a diagnostic and fault isolation aid in that it provides direct access to the primary working register storage within the drive interface.

Note that both operations occur regardless of which operation is intended.

When A-register bit 7 is a 1:

A-register bits 0 through 4 are stored internally and bits 8 through 15 are written into the file address specified by bits 0 through 4. The intent is to use this in conjunction with the read file command as a means to echo data through all 16 drive interface file registers. Note that this also loads the first word address upper two bits.

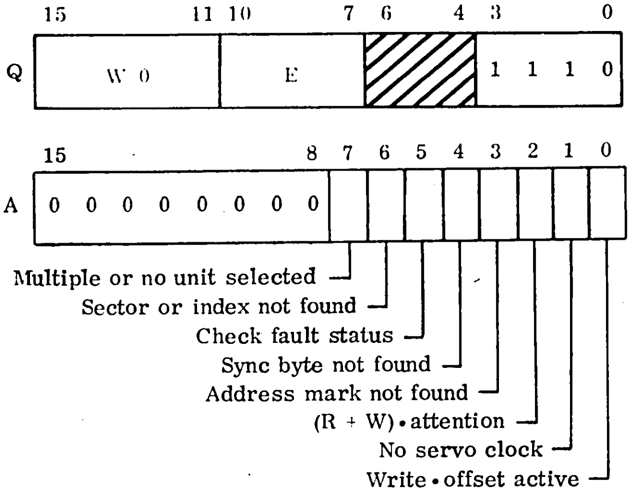
The drive interface accepts this function command if it is not busy.

The select file address function is primarily a diagnostic or fault isolation aid in that it provides direct access to the primary working storage registers within the disk adapter.

A-register bits 0 and 1 of this command are loaded into the current word address register within the drive interface (see set first word address, upper, function above). If select file address functions are to be executed after initializing the drive interface's current word address register, the original upper bit configuration of the current word address register must be maintained or re-established.

DRIVE STATUS 1

Format: Input
Function Code: E



The drive status 1 function inputs the drive status word 1 to the CPU's A register.

Write . Offset Active

A-register bit 0 equal to 1 indicates that the CPU attempted to perform a write operation with the heads offset in a plus or minus condition from the normal position (see Read Recovery/Status Control functions above).

No Servo Clock

A-register bit 1 equal to 1 indicates that the drive has failed to detect the required clock signal from the servo track.

(R+W)· Attention

A-register bit 2 equal to 1 indicates that a read or write operation has been attempted with the drive's heads unloaded (see Select Acknowledge Status above). This is not necessarily a fault condition.

Address Mark Not Found

A-register bit 3 equal to 1 indicates that the drive and/or controller/formatter has failed to detect an address mark within the address field of a sector. The address mark is a required synchronization point for writing or reading disk data.

Sync Byte Not Found

A-register bit 4 equal to 1 indicates that the controller/formatter has failed to detect a sync byte within the address or data field of a sector. The sync byte is required for proper synchronization within the address or data field.

Check Fault Status

A-register bit 5 equal to 1 indicates that there is one or more bits set in the drive fault status word.

Sector or Index Not Found

A-register bit 6 equal to 1 indicates that the drive and/or controller/formatter have failed to detect index or sector pulses from the disk servo track.

Multiple or No Unit Selected

A-register bit 7 equal to 1 indicates that more than one drive responded to a unit select command (i.e., they have the same logical address) or that no unit responded to the unit select command.

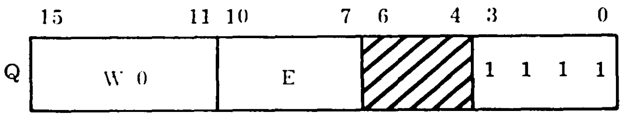
This input command allows the program to detect and/or monitor basic drive errors following any read or write operation.

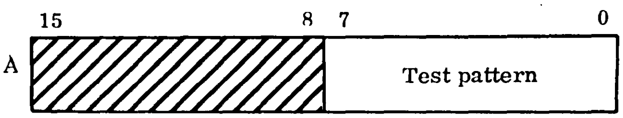
Note that this status is only valid following a direct memory access read or write command to the drive. The drive interface accepts this command only if it is not busy and the controller/formatter is currently selected.

CONTROLLER/FORMATTER ECHO OUTPUT

Format: Output
Function Code: F

The controller/formatter echo output function transmits the test data pattern to the controller/formatter from the A register.





A-register bits 0 through 7 may be any binary pattern. The data output via this function may be input and verified by the controller/formatter echo input function below.

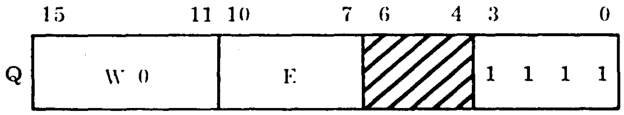
This command, in conjunction with the corresponding echo input command, provides a functional verification of the command, status, and data paths between the drive interface and controller/formatter. Output test data can be transmitted to the controller/formatter and echoed back to the drive interface.

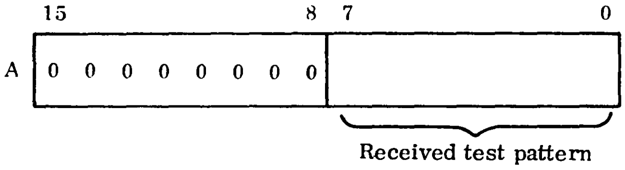
This function is intended to be a diagnostic aid that allows the CPU to verify the bus-out and bus-in data paths between the drive interface and the controller/formatter.

The drive interface accepts this function if the controller/formatter is selected and the drive interface is not busy.

CONTROLLER/FORMATTER ECHO INPUT

Format: Input
Function Code: F





The controller/formatter echo input function reads back controller/formatter test data to the A register.

This command allows the CPU to input the binary data pattern that was output to the controller/formatter via the controller/formatter echo output function above.

When used in conjunction with the controller/formatter echo output function, this command allows full echo checking of the command, status, and data paths between the drive interface and controller/formatter.

The drive interface accepts this command if it is not busy; however, the input data is only meaningful when this command is immediately preceded by the controller/formatter echo output function.

WORD FORMATS

DATA BUFFER ORGANIZATION

For direct memory access data operations, the organization of the data buffer written from or read into main memory is shown in figure 2-1.

ADDRESS FIELD BUFFER ORGANIZATION

The identifying data block written at the start of each physical sector is known as the gap A address field. It is created by initiating a direct memory access write address function with the buffer formatted in memory as in figure 2-2. Each address field consists of five memory words for each sector.

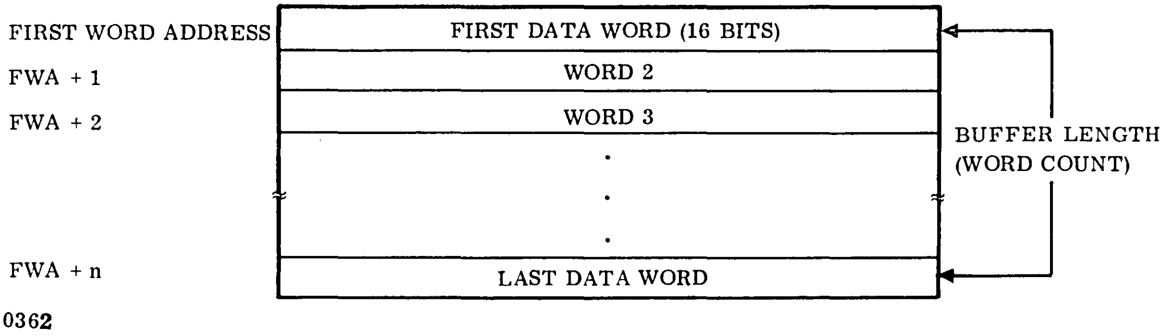


Figure 2-1. Data Buffer

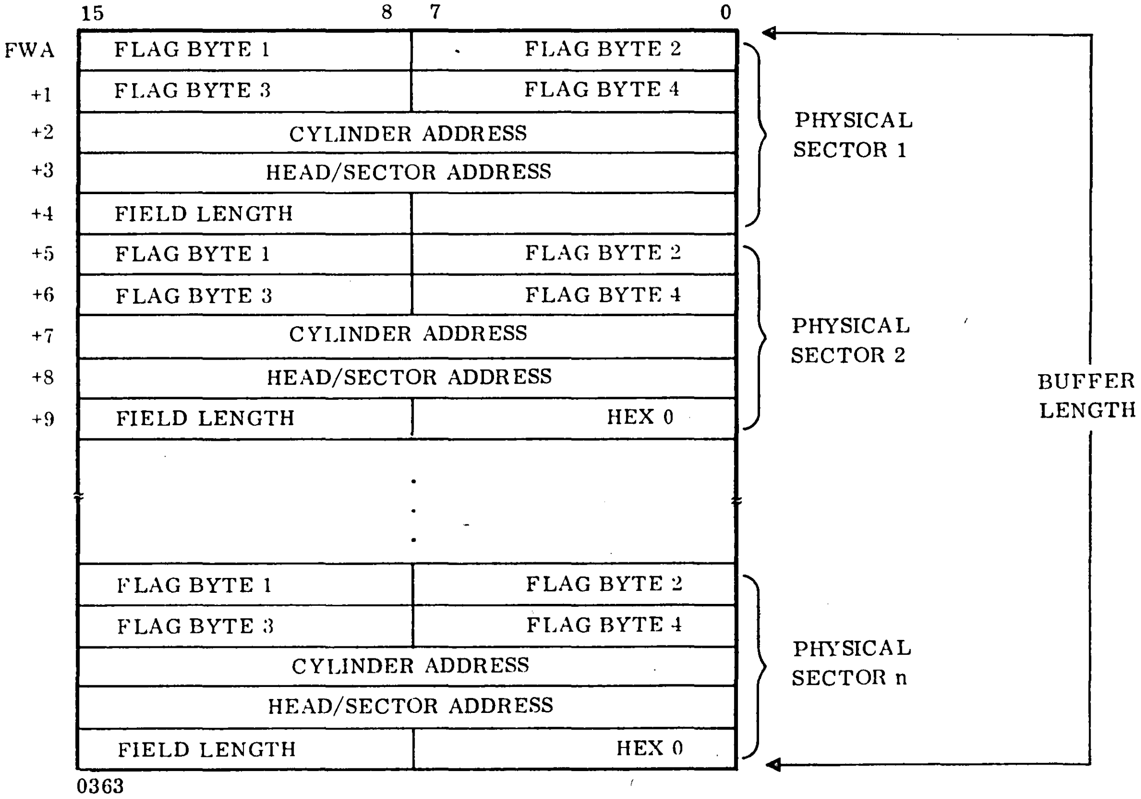


Figure 2-2. Address Field Buffer

The address field is always read by drive interface hardware prior to initiating any data transfers to the file. This allows the drive interface to determine its current address position relative to the software's requested (target) address.

A direct memory access read address function reads the address field from the selected file into main memory as in figure 2-2. Multiple address fields may be read by increasing the buffer length count by five for each sector desired.

ADDRESS FIELD WORDS

Flag Bytes

The four flag bytes are not used by the storage module system hardware. The operating system software may use these fields for bad track/sector flags, write protect flags, or alternate track assignments.

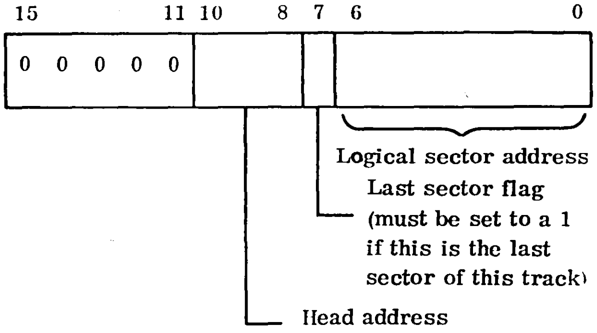
Cylinder Address

The format of this word must be as follows:

15 10	9 0
0 0 0 0 0 0	Absolute cylinder address

Head/Sector

The format of this word must be as follows:



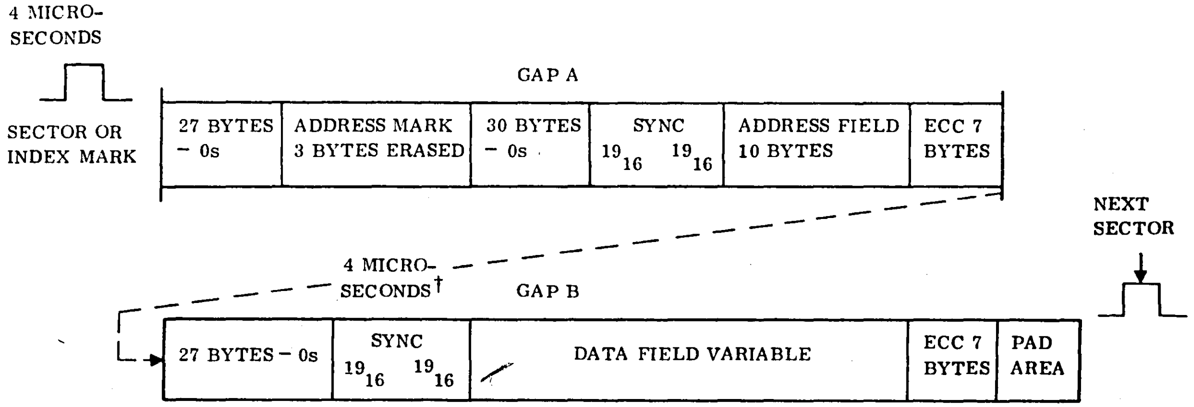
15
11 10
8 7 6
0
0 0 0 0 0
Logical sector address
Last sector flag (must be set to a 1 if this is the last sector of this track)
Head address

Field Length

This word must contain the left-justified eight-bit number equal to the number of bytes contained within the data field of the sector minus 1 (e.g., twice the number of 16-bit data words minus 1).

SECTOR FORMAT

Figure 2-3 illustrates the sector format. This is provided as general information only. The synchronization bytes, address marks, and error correction code words are created



SYNC CHARACTER – 0001 1001

† THIS PERIOD IS THE TIME AVAILABLE FOR THE DRIVE INTERFACE TO DETERMINE THAT THE ADDRESS FIELD IT HAS JUST READ IS THE TARGET (DESIRED) ADDRESS AND THAT THE GAP B FIELD FOLLOWING IS THE DATA THAT MUST BE TRANSFERRED UNDER DIRECT MEMORY ACCESS. THIS TIME IS NOT A SOFTWARE DECISION PERIOD FOR INITIATING A DIRECT MEMORY ACCESS DATA OPERATION FOLLOWING A DIRECT MEMORY ACCESS READ ADDRESS FUNCTION.

0364

Figure 2-3. Sector Format

by hardware in the controller/formatter and cannot be altered by software. Only the gap A address or gap B data fields are even transferred through the drive interface during direct memory access operations.

INTERRUPTS

There are six interrupt-causing conditions within the drive interface. Each condition may be individually enabled (allowed) or disabled (locked out) by software.

Through execution of a director function (function code 8), an interrupt mask register with bits corresponding to the interrupt conditions may be set or cleared (enabled/disabled) in the drive interface. The interrupt conditions available from the drive interface are:

Alternate drive interface †

Transfer complete

Alarm condition

Seek end†

End of cylinder

Controller/Formatter selected

The logical OR of all drive interface interrupts is presented to the CPU as a common single interrupt. Following any drive interface interrupt, the CPU must read drive interface status to determine which of the enabled conditions caused the interrupt.

ALARM CONDITIONS

Various fault or abnormal conditions can occur within the storage module system. The detection of these conditions causes:

- An alarm interrupt (if enabled) to the CPU
- The setting of the alarm status bit
- The immediate termination of any direct memory access read or write operation in progress
- The setting of the associated fault status bit

The abnormal conditions that cause an alarm are:

- Protect fault
- Memory addressing error
- Forced release of controller/formatter
- Lost data
- Controller/formatter error
- Storage parity error (direct memory access read operation only)

The one exception to alarm conditions are their associated termination of the direct memory access operation is error correction code errors. This type of error can only occur during a direct memory access read operation, and it also sets the alarm interrupt and status bit. However, due to the error correction capabilities of the controller/formatter, the detection of an error correction code error does not terminate the operation but allows it to continue to the end of the sector in which the error occurred. Detection of any of the other alarm conditions causes the read operation to end immediately or at the end of the current sector.

PROGRAMMING AIDS AND CONSIDERATIONS

DRIVE INTERFACE BUSY

The only two functions the drive interface accepts while it is busy are the director and drive interface status request functions (function code 8). Initiation of any other A/Q function to the drive interface while it is busy causes an input/output reject condition.

Upon acceptance of an input/output command, the drive interface becomes busy. The following commands result in the drive interface staying busy for more than 6 microseconds.

- Initiate poll – The drive interface goes busy until the controller/formatter responds with the poll status (approximately 40 microseconds).
- Format write – 16.6 to 33.2 milliseconds
- Any seek operation – 20 microseconds
- Any direct memory access read or write command causes the BUSY signal to be set until the end of the last sector being operated on is reached, an end-of-cylinder condition is detected, or an abnormal condition causes the operation to be aborted (see Alarm Conditions above).

STAND-ALONE DRIVE INTERFACE FUNCTIONS

The drive interface accepts any function from the CPU if it is not busy. However, since some functions are directed at the controller/formatter or drives, they require that the drive interface first be connected to the controller/formatter (selected) before they can be executed. Attempting to execute one of these instructions without first selecting the controller/formatter results in an input/output reject condition.

† Requires a hardware jumper on the drive interface board.

The following lists the stand-alone functions that the drive interface can execute without controller/formatter selection:

- Director function
- Buffer length
- Unit select (selects the controller/formatter, if the controller/formatter is not already selected)
- Set first word address, upper/lower
- Read file register data
- Cylinder address status
- Sector and head address status
- Drive interface status

The following A/Q functions are executed by the drive interface without controller/formatter selection. Generally, however, these functions are preceded by functions (shown following the function below) that required controller/formatter selection.

- Poll status (initiate poll)
- Select acknowledge status (unit select)
- Controller/formatter echo in (controller/formatter echo out)
- Drive echo in (drive echo out)

The results obtained from executing the above as stand-alone functions are indeterminate.

TRACK OPERATIONS

The following operations are basically track operations in that only the cylinder and head address are used; any sector address information supplied is ignored.

- Write Address – This operation begins at the index mark and continues until the buffer length register within the drive interface equals zero.
- Format Write – This operation starts at the index mark and terminates the next time the index mark is sensed (one track only).
- Read Address – This operation begins at the index mark and continues until the buffer length register within the drive interface equals zero.

Note that read and write address operations may be greater than one track in length (as determined by the buffer length), but these operations always begin at the index mark.

SEEK OPERATIONS

Following the initiation of any seek operation, a seek end interrupt (and status) is generated if the software program de-selects the controller/formatter, thereby allowing it to poll the drives. Unless the controller/formatter is free to poll, no seek end interrupt or status is generated. The alternative is for the program to repetitively issue an initiate poll output function with the force poll update bit set.

PROTECTED OPERATION

When the drive interface is operating in the protect mode (selectable by optional jumper on the drive interface board), input/output instructions not having their protect bit set cause a protect violation and are rejected by the drive interface (external reject). The drive interface can write into protected CPU memory only if the drive interface is protected and the disk read operation was initiated by a protected A/Q output instruction. The exception to the protect rule is a drive interface status request that is never rejected, regardless of the protect condition.

Table 2-6 defines how the drive interface handles data transfers to memory with respect to CPU protect, drive interface protect, and protected instructions. Note that with the CPU's protect switch off, the protected states of the drive interface and input/output instruction are ignored.

TABLE 2-6. DRIVE INTERFACE PROTECT CONDITIONS

CPU Protect Switch	Drive Interface Protected (Jumper in)	Protected I/O Instruction	Result
Off	No	No	A
Off	No	Yes	A
Off	Yes	No	A
Off	Yes	Yes	A
On	No	No	B
On	No	Yes	B
On	Yes	No	Reject
On	Yes	Yes	A

A = Drive interface can transfer to protected or unprotected memory.

B = Drive interface can transfer to unprotected memory only.

BOARD PREPARATION

Prior to system operation, parameter jumpers must be installed on the drive interface board. Section 2, Operation and Programming, describes the function of each jumper. Table 3-1 identifies the jumpers by name and lists the drive interface's jumper pin designations on the printed circuit assembly.

DRIVE INTERFACE BOARD PLACEMENT

CAUTION

> The storage module drive printed wiring assembly contains electrostatic sensitive devices as indicated by the red solder mask. Exercise extreme care in handling to avoid damage to the board. Common practices, such as touching a grounded surface before handling, storing in anti-static or conductive bags for storage or transfer, repairing only at properly equipped and grounded work stations, etc., should be strictly followed.

The drive interface should be inserted with the component side of the board facing towards the left side of the CPU cabinet. Assignment of the board within the CPU board cage may be to any one of the four A/Q-DMA slot positions available. For CYBER series systems, position H is the standard slot for the storage module disk.

CABLING

Figure 1-1 illustrates the disk subsystem cabling for a dual CPU configuration. The input/output cables (bus in/bus out) run directly from the CPU backplane slot assigned for the drive interface logic board to the controller/formatter connector panel. The connectors for the CPU end of the cable are 50 pin connectors that slip over the backplane pins (identified in table 3-2).

CABLES

The interconnecting cables are 24 twisted pair with outside shield. The standard length supplied is 25 feet (7.62 meters). The maximum cable length supported is 50 feet (15.24 meters).

CONNECTOR PART NUMBERS

Backplane connector – CDC part no. 94360805

Controller/formatter connector – CDC part no. 24502303

TABLE 3-1. DRIVE INTERFACE JUMPER PIN DESIGNATIONS

Jumper Name	Pin Designation and Connections
Select hold 0 (out)	Connect E16 to E17.
Select hold 1 (in)	Connect E14 to E15.
AUTOLOAD switch	Disconnect E19 to E20. Connect E18 to E20.
Drive interface equipment number (E_{16} is standard equipment code)	2^0 = E54, 2^1 = E53, 2^2 = E52, 2^3 = E51 High+ (1) = E43-E46, low- (0) = E47-E50
Drive interface protected/unprotected	Connect E23 to E21 or E22.
ALERT 0/1, controller/formatter to drive interface 0/1	Connect E37 to E36 or E38.
Interrupt to drive interface 0/1	Connect E42 to E39 or E41.
Interrupt from drive interface 0/1	Connect E40 to E39 or E41.
Autoload drive number	E33 = 2^0, E32 = 2^1, E35 = 2^3, E34 = 2^4 High (+) = E24-E27,low (-) = E28-E31

TABLE 3-2. DRIVE INTERFACE/CONTROLLER/FORMATTER SIGNAL PIN ASSIGNMENTS

Signal	Drive Interface Backplane Pin	Controller/Formatter Connector Pin	Direction	
Recycle	88	u/w	To controller/formatter	bus-out cable
Select hold 0	98	A/C	To controller/formatter	bus-out cable
Select hold 1	99	B/D	To controller/formatter	bus-out cable
Bus out 0	297	E/H	To controller/formatter	bus-out cable
Bus out 1	298	F/J	To controller/formatter	bus-out cable
Bus out 2	296	K/M	To controller/formatter	bus-out cable
Bus out 3	299	L/N	To controller/formatter	bus-out cable
Bus out 4	91	P/S	To controller/formatter	bus-out cable
Bus out 5	94	R/T	To controller/formatter	bus-out cable
Bus out 6	92	U/U	To controller/formatter	bus-out cable
Bus out 7	93	V/X	To controller/formatter	bus-out cable
Bus out parity	87	y/a	To controller/formatter	bus-out cable
Tag bus 0	86	z/b	To controller/formatter	bus-out cable
Tag bus 1	85	c/e	To controller/formatter	bus-out cable
Tag bus 2	84	d/f	To controller/formatter	bus-out cable
Tag bus 3	83	h/k	To controller/formatter	bus-out cable
Tag bus parity	89	j/m	To controller/formatter	bus-out cable
Tag out	95	n/r	To controller/formatter	bus-out cable
Sync out	90	p/s	To controller/formatter	bus-out cable
Response	96	t/v	To controller/formatter	bus-out cable
Select active in	19	a/c	From controller/formatter	bus-in cable
Bus in 0	13	B/D	From controller/formatter	bus-in cable
Bus in 1	14	E/H	From controller/formatter	bus-in cable
Bus in 2	12	F/J	From controller/formatter	bus-in cable
Bus in 3	15	K/M	From controller/formatter	bus-in cable
Bus in 4	23	L/N	From controller/formatter	bus-in cable
Bus in 5	22	P/S	From controller/formatter	bus-in cable
Bus in 6	17	R/T	From controller/formatter	bus-in cable
Bus in 7	18	U/W	From controller/formatter	bus-in cable
Bus in parity	21	V/X	From controller/formatter	bus-in cable
Tag valid	4	y/a	From controller/formatter	bus-in cable
Sync in	24	z/b	From controller/formatter	bus-in cable
Normal end	7	c/e	From controller/formatter	bus-in cable
Check end	5	d/f	From controller/formatter	bus-in cable
Index	6	t/v	From controller/formatter	bus-in cable
Sector	8	u/w	From controller/formatter	bus-in cable
Alert 0	9	h/k	From controller/formatter	bus-in cable
Alert 1	10	j/m	From controller/formatter	bus-in cable

TABLE 3-2. DRIVER INTERFACE/CONTROLLER/FORMATTER SIGNAL PIN ASSIGNMENTS (Contd)

Signal	Drive Interface Backplane Pin	Controller/Formatter Connector Pin	Direction
Select active out	20	None	DA0 to DA1
Int. DA0	11	None	DA1 to DA0
Int. DA1	16	None	DA0 to DA1

BACKPLANE PIN ASSIGNMENTS

Table 3-2 identifies the bus in and bus out cable lines and correlates the backplane pin numbers to the controller/formatter connector pins.

CHECKOUT

Powering up the CPU system provides the drive interface with its required input voltages of +5 volts and -12 volts through the CPU's backplane voltage bus.

With the exception of the autoload switch, the drive interface does not contain any operating switches or indicators. Connection of the external autoload switch requires the normally open section of the switch to be connected to backplane pin 42 of the drive interface slot. The normally closed section and normally connected section should be connected to pin 45. The wiper (common) section of the switch should be connected to pin 43 (ground).

Subsystem checkout may proceed using the appropriate diagnostics. After confirming that remaining components of the subsystem are powered up and made ready the controller/formatter/drive circuit breakers are turned on and the scratch pack is loaded on selected drives.

THEORY OF OPERATION

The drive interface consists of general logic blocks (registers, counters, and flip-flops) with connection paths to and from a single internal data bus. The gating of data onto this bus is controlled by hardware multiplexers, which, in turn, are enabled by the drive interface's control logic. This control logic is implemented by an internally stored micro program and associated hardware decode and control logic. Figure 4-1 is a simplified block diagram of the drive interface. The box in the upper right-hand corner of the diagram blocks refer to the logic drawing page numbers.

The drive interface's micro program is called controlware, because processing logic does not have the capability for arithmetic and Boolean operations, only control.

The micro program is made up of many individual subprograms called micro routines. A micro routine may contain anywhere from two to 20 individual micro instructions. Each micro instruction directs its control logic to select one of several available logic blocks as an input through a multiplexer. The selected logic block is known as the source; if it is a register, for this example it supplies the source data used in this micro operation. The same micro instruction also selects a specific logic block as a final destination for the source data.

By this method, controlware directs and controls the data flow and command paths in a micro controller.

Each micro instruction resides within a read-only memory (ROM) circuit at a specific location or address in the read-only memory. Access to a particular read-only memory address and the subsequent execution of the instruction resident there is accomplished by any one of three methods:

- The instruction there is sequentially stepped to by the previous micro instruction.

- The micro instruction currently being executed is a jump instruction that references that address.

- An input/output command is sent to the drive interface from the CPU's A/Q registers. The storage module system function code contained in Q-register bits 0 through 3 of that command is combined with other bits in the drive interface to form a unique read-only memory address. This transposing of an input/output function code into a read-only memory address and the referencing of the micro instruction at that address is known as the Qf transform.

Data to be written or read from the selected disk drive initially passes through a first-in/first-out (FIFO) buffer in the drive interface. This eliminates synchronizing the timing of the CPU's direct memory access (DMA) channel to the controller/formatter's interface timing.

Write data originating from the CPU's memory is transmitted to and held in the drive interface's FIFO buffer. The first data word stored in the FIFO buffer is the first data word written on the file via the drive interface's data bus (bus out) interface to the controller/formatter.

Data read from the file is sent to the drive interface by the controller/formatter via the data bus (bus in). It is held in the FIFO buffer until the drive interface's DMA control logic connects into the CPU's DMA channel. At that time, the first data word written into the FIFO buffer is the first word to be transferred to the CPU's memory.

A/Q INTERFACE

All input/output operations for the storage module system are first initiated through the drive interface's A/Q interface. During a CPU output from the A instruction, the contents of the A and Q registers are transmitted to the drive interface over the CPU backplane lines, along with the appropriate input/output handshaking signals.

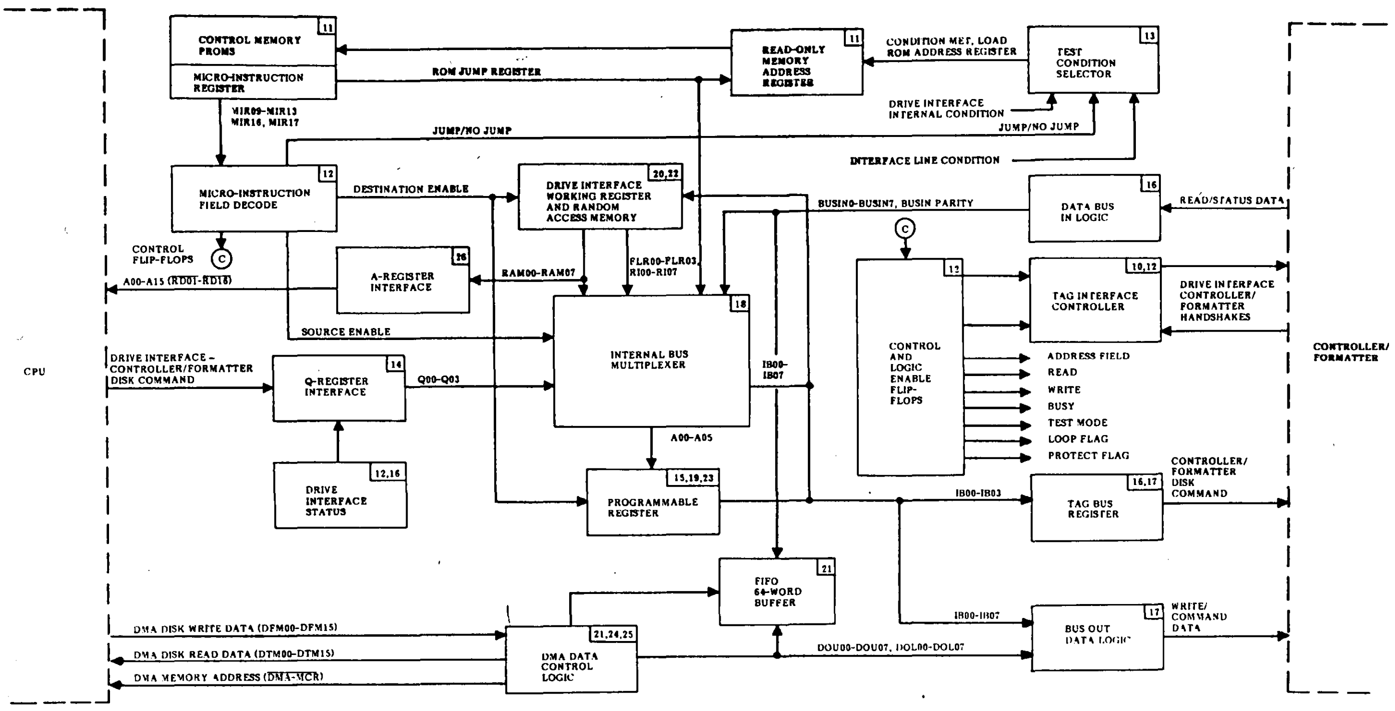
Based on the input/output function code contained in Q-register bits 0 through 3, the contents of the A and Q registers are handled in one of the two following ways: The A/Q register data is acted on entirely within the drive interface or some part of the A or Q register's data is passed on to the controller/formatter or drive for further interpretation.

Figure 4-2 illustrates the data paths for the A and Q registers in the drive interface.

For a CPU input to A instruction, contents of the Q register are sent to the drive interface's A/Q interface. However, it is the drive interface that now supplies information to the A-register via the CPU's RDxx backplane lines. This information may originate from the drive interface, controller/formatter, or drive unit.

DRIVE INTERFACE ADDRESSING

The Q-register contents are always sent to the drive interface for any command setup. The W and E fields of the Q register are the input/output device code. As used to address the drive interface, the W field (converter code) of the Q register is decoded within the CPU to provide the WE0 signal. The E (equipment) field enters the drive interface as ADR08/ through ADR11/ and is compared to the drive interface's jumper-selected equipment address code. If the input/output device address requested by the input/output command equals the drive interface's assigned address, the Q function decode logic is now enabled to act on Q-register bits 0 through 3 (Qf).



NOTE NUMBERS IN BLOCKS REFER TO FIELD PRINT PACKAGE LOGIC DRAWING SHEET

0348

Figure 4-1. Drive Interface Simplified Block Diagram

Q-FUNCTION DECODE

The F field of the Q register enters the drive interface as logic signals ADR01/ through ADR04/, where it fans out to three logic areas: the function code 8 detect, Qf transform, and drive function logic.

FUNCTION CODE 8 DETECT

This logic samples Q00 through Q03 for the only hardware-implemented functions in the drive interface. These are the drive interface director and drive interface status request functions, and both are accepted and executed without controlware intervention. During the drive interface's hardware execution of these two functions, its micro-control program is performing a do-nothing idle loop or processing some previously requested function.

QF TRANSFORM

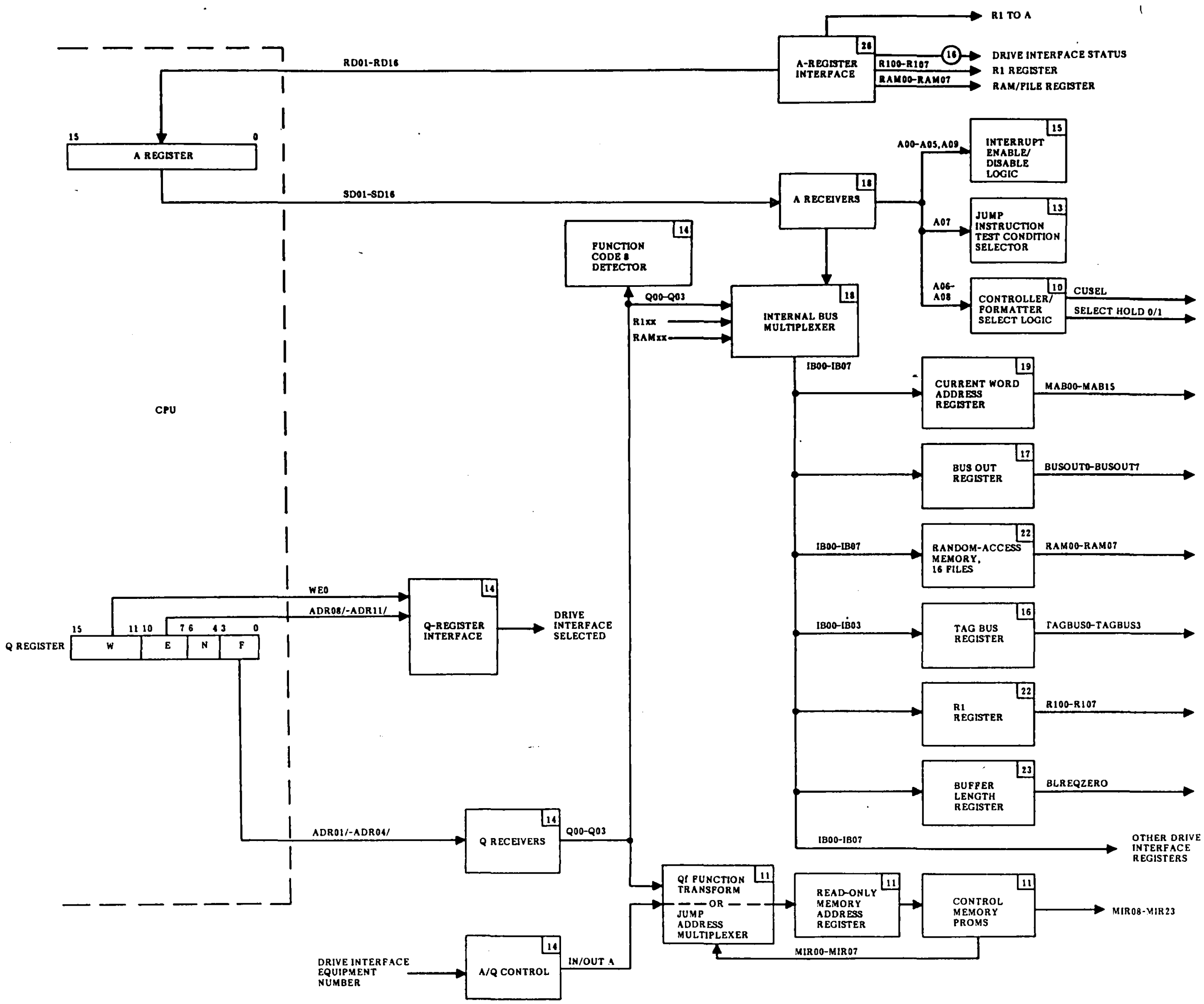
The read-only memory program contains a dedicated micro routine for each of the storage module system's 16 input and 16 output A/Q commands. Excluding Q function code 8, the actual execution of the input/output command is controlled by the dedicated micro routine (combined in some cases with micro routines common to several drive interface functions).

The Qf transform allows a single micro instruction to decode the storage module system function code contained in the Q register. Access to the associated micro control routine is accomplished by transforming the four-bit Q-register function code field (Qf) into a nine-bit read-only memory address. The Qf is ORed with a hardware supplied, hexadecimal constant of 1E0 for input and 1F0 for output functions. This ORing forms a unique read-only memory program address in the range of 1E0 through 1FF, which is then loaded into the read-only memory address register and accessed. The micro instruction at this transform-generated address is a jump instruction to the starting read-only memory address of the micro routine for the specific input/output operation.

DRIVE FUNCTIONS

If the current input/output command (Qf) relates to a drive such as reading drive fault status, the Qf field is passed through the drive interface to the selected drive unit via the controller/formatter.

A transform operation on Qf occurs first, with the controlling micro routine selecting Q-register bits 0 through 3 as the input source to the internal bus multiplexer. The drive interface's tag out register connecting the drive interface to the controller/formatter, is selected by the



NOTE: NUMBERS IN BLOCKS REFER TO FIELD PRINT PACKAGE LOGIC DRAWING SHEET

0347

Figure 4-2. A/Q Data Flow

micro program as the destination for the Qf. As seen in figure 4-2, the Qf is output to the controller/formatter. The appropriate tag control lines are asserted by the drive interface, instructing the controller/formatter to transmit the Qf code it has received to the selected disk drive.

A-REGISTER USAGE

During an A/Q command to the drive interface, the A-register content is directly related to the Qf field. The program sets the A register to any one of the following:

- File addressing information
- Qf modifiers or operation parameters
- Data patterns used for hardware testing

For A/Q input operations, the A register receives input from the drive interface. This data may originate either from the drive interface, controller/formatter, or selected drive, depending on the operation specified by Qf.

The information input to the A register can be any one of the following:

- Hardware status
- Received data test patterns
- Register data

A-REGISTER OUTPUT FLOW

Upon receiving an A/Q output command, the Qf transform hardware directs the controlware program to the micro routine for that particular function. The micro-routine instructions select the A register (A00 through A15) as the source data input to the internal bus multiplexer. The destination of that A-register data may then be to any one of the logic blocks connected to the internal bus as shown in figure 4-2.

As an example, if the Qf code is a (set) buffer length (register) command, the A register has been set by software to a value representing the data transfer word count. To accomplish the loading of the buffer length register with this word count, one micro instruction of the unique micro routine for this input/output command selects A register lower (AL) as the source. In that same instruction, the buffer length lower (BLL) register is selected as the destination for the A-register data. Upon execution, this micro instruction causes A-register bits 0 through 7 to be loaded into BL0 through BL7.

The next sequential micro instruction in the micro routine is: A register upper (AU) as the source of data to drive interface buffer length register upper (BLU). This completes the transfer of the 16-bit CPU A register to the 16-bit buffer length register over an 8-bit wide data bus. As seen by this example, the execution of the software input/output command, set buffer length register, was accomplished within the drive interface by the execution of two micro instructions.

A-REGISTER INPUT FLOW

For all drive interface A/Q input-to-A operations, the A register receives the contents of backplane lines RD01 through RD16. These lines are driven by the drive interface's output multiplexer, whose input-to-output selection is controlled by both hardware and controlware.

A software command requesting drive interface status is performed purely by hardware; and for this function, the hardware status conditions of the controller are enabled through the output multiplexer to the A register by the function code 8 detect logic. Any other types of A/Q input operation that request controller/formatter or drive status or drive interface/controller/formatter register data require the intervention of the micro control program.

External information that can be requested by software such as controller/formatter or drive status, drive unit selection, and error correction code (data correction codes) or test pattern data, is first brought into the drive interface from the controller/formatter's interface and temporarily stored in the drive interface's general-purpose R1 register. Drive addressing information, such as cylinder, track, and sector numbers, is maintained in, and read directly from, one of the drive interface's 16 random-access memory file registers to the A register.

The CPU's A-register input (figure 4-2) is fed by the drive interface output multiplexer. The multiplexer receives inputs from the drive interface's R1 register as well as random-access memory. The contents of the CPU's Q register bits 0 through 3 (Qf) during the input-to-A operation specifies the data that is to be transmitted to the A register. This data is either equipment status from the R1 register, as previously mentioned, or drive addressing information from the random-access memory files. The drive interface's controlware to the micro routine associated with executing the input-to-A operation selects the source of data for the A register by selecting either the R1 register or a previously enabled file register in the random-access memory. This is accomplished with a destination control flip-flop (designated as the R1-to-A control flip-flop) within the drive interface that can be set by a micro instruction. When this flip-flop is set, it enables the R1 register to pass through the drive interface's output multiplexer. If the R1-to-A path is not enabled by the micro routine, the contents of the selected random-access memory file register are gated through the multiplexer.

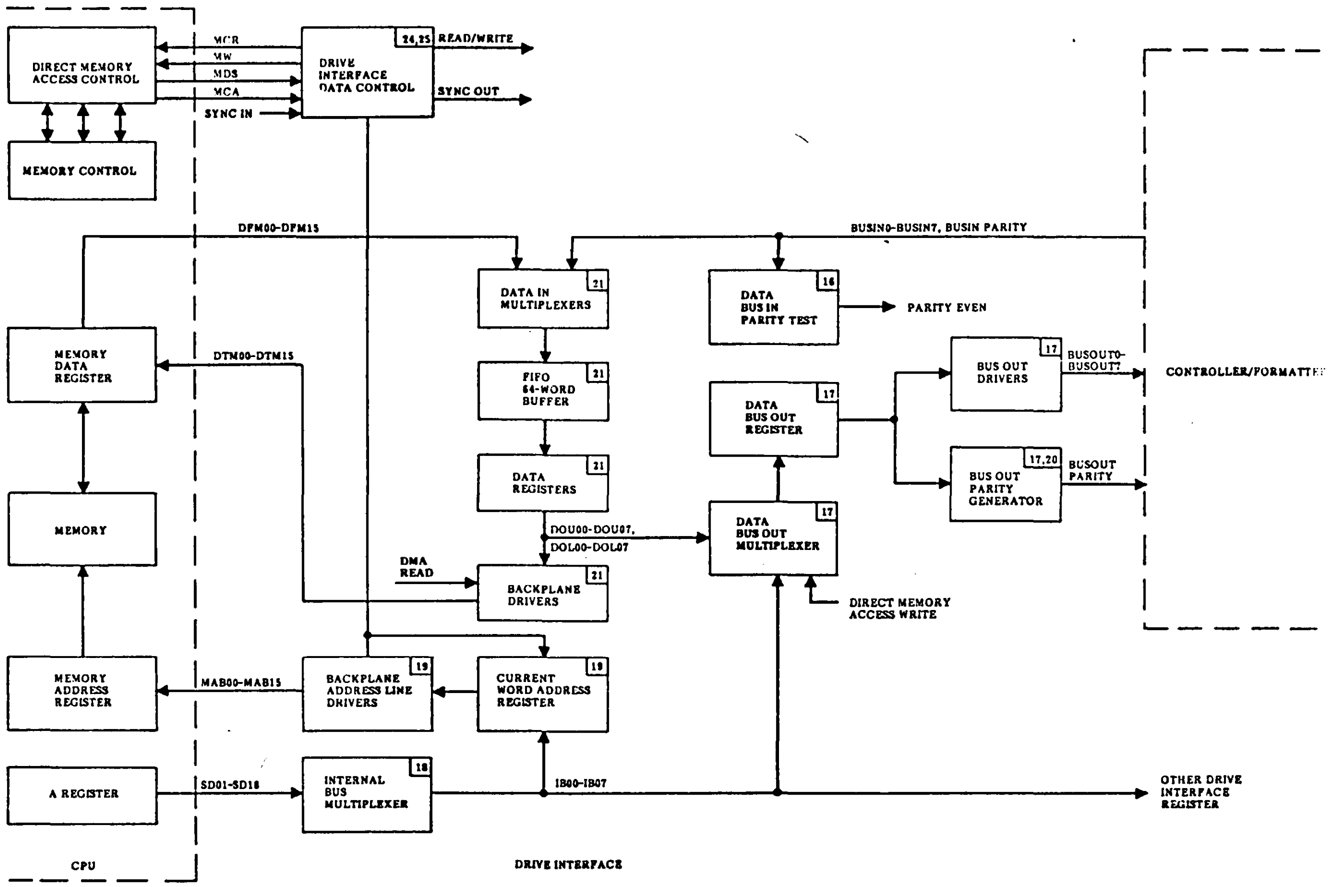
DIRECT MEMORY ACCESS INTERFACE

All data read from or written to the disk files is transferred between the CPU memory and the storage module system through direct memory access. The direct memory access channel consists of backplane lines and handshaking logic between the CPU's memory and the drive interface's A/Q-DMA card slot position in the CPU chassis. Sixteen lines each are allocated for data from (DFM01 through DFM16) and to (DTM01 through DTM16) the CPU's memory. Figure 4-3 illustrates the direct-memory access data paths and drive interface logic. The drive interface also has a connection path into 18 backplane lines that drive the CPU's memory address register. These lines select the specific memory address (MAB01 through MAB18) accessed under direct memory access control for data transferred over the DTM or DFM backplane lines.

DIRECT MEMORY ACCESS ADDRESSING

The drive interface's 18-bit current word address register drives the MAB backplane lines and is initially loaded with the first word address from the A register by a set first word address upper or lower command. This first word address points to the first CPU memory location accessed by the drive interface under direct memory access control.

The 18-bit first word address word is loaded via the SD backplane lines through the internal bus multiplexer into the 18-bit current word address register eight bits at a time. For addressing CPU memory addresses above 65K, the two most significant address bits of the current word address are loaded separately by the set first word address upper command.



NOTE: NUMBERS IN BLOCKS REFER TO FIELD PRINT PACKAGE LOGIC DRAWING SHEET

8349

Figure 4-3. Direct Memory Access Interface

During the direct memory access read or write operation a request (signal MCR) for a memory cycle is made to the CPU by the drive interface's direct memory access logic. Upon accepting the direct memory access request, the CPU memory's response signal to the drive interface (MCA) enables the contents of the current word address register out to the MAB lines.

The CPU's memory address register latches the first word address, and a memory cycle is initiated to access that address.

Following completion of the direct memory access data transfer, the memory cycle accept (MCA) line is released, causing the current word address register's contents to be incremented by +1. This register now points to the next sequential memory address (first word address + 1) in the data buffer area that is accessed by the drive interface on the following direct memory access request.

DISK WRITE OPERATION (READ FROM MEMORY)

The transforming of a disk write command (function code A) causes the controlling micro routine to activate the drive interface's direct memory access logic. The drive interface requests a memory cycle (MCR) and, upon acceptance by memory (MCA), it asserts the memory write line (MW) at a low level, indicating a memory read operation.

The first data word fetched from memory is placed on the DFM backplane lines. The drive interface is notified of this by the memory data strobe (MDS) signal. The drive interface's data-in multiplexer transfers the 16-bit data word to its FIFO buffer, where it is strobed as two eight-bit bytes by the MDS signal. The drive interface fills the FIFO buffer to a maximum of 64 words through continued direct memory access read requests.

The FIFO buffer is unloaded one word at a time when the controller/formatter is ready to accept data (indicated by the controller/formatter asserting SYNCIN to the drive interface's controller/formatter/direct memory access control logic). The drive interface's FIFO buffer data register, which now contains the first 16-bit data word that was stored in the FIFO buffer, is connected to the bus-out multiplexer. One byte at a time is passed through the multiplexer and the bus-out register. At this point, a parity assignment is made on the byte, so that the total count of set bits, including parity, is odd. This byte is transmitted to the controller/formatter interface along with the drive interface's data notification signal, SYNCOUT. The remaining byte of the word is transferred to the controller/formatter in an identical manner.

The FIFO buffer continues to be loaded from the CPU's memory with 16-bit words on every MDS signal and unloaded eight bits at a time on every SYNCIN signal received from the controller/formatter.

The drive interface's buffer length register (BL), previously loaded by software with the transfer word count, is decremented -1 on each direct memory access cycle. The loading of the FIFO buffer ceases when the buffer length register goes to a zero count.

DISK READ OPERATION (WRITE TO MEMORY)

A disk read command, function code 9, initiates a direct memory access operation similar to a direct memory access write, except the direction of data is to, rather than from, CPU memory.

A 9-bit byte (data plus parity) is placed on the bus in by the controller/formatter. The controller/formatter asserts SYNCIN, notifying the drive interface that a byte is available. The byte is input through the data-in multiplexer, and the drive interface responds by issuing the SYNCOUT signal to the controller/formatter. Each byte is stored in its associated FIFO buffer and then is output to the data register in synchronization with the direct memory access handshake signals. The FIFO buffer data register's output is enabled through the drive interface backplane drivers and onto the DTM lines. The current word address register points to the CPU memory address where the 16-bit word is stored.

The handshaking of direct memory access signals MCR and MCA continues asynchronous to the controller/formatter/drive interface SYNCIN/SYNCOUT handshaking. Direct memory access control waits until the controller/formatter catches up if the FIFO buffer becomes temporarily empty.

The controller/formatter data transfer to the FIFO buffer is terminated when the drive interface's buffer length register has been decremented to 0. The direct memory access data transfer to memory ceases as soon as the FIFO buffer register is empty.

DRIVE INTERFACE/CONTROLLER/ FORMATTER INTERFACE

The drive interface and controller/formatter are physically connected by two cables, identified as the bus-in and bus-out cables. The bus-out cable contains the output data bus that is used for transmitting data and byte control commands to the controller/formatter. It also contains the tag bus that is used to transmit the actual binary-coded function (read, write, and select) to the controller/formatter.

The bus-in cable contains the input data bus and tag control lines. The input data bus carries data from the controller/formatter (controller/formatter/drive status, register, or file read data). The tag control lines establish the communication protocol between the controller/formatter and drive interface that is shown in figure 4-8.

The signal interface on these cables is implemented using integrated circuit transmitters and receivers as shown in figure 4-4. These devices allow for transmission of data over cables up to 50 feet (15.24 meters) in length.

INTERFACE CONTROL LINES

SELECT HOLD

When the drive interface controlware program determines, through decoding of software output function code 3, that controller/formatter selection is required, it sets the need controller/formatter flip-flop, causing the select hold interface line to be asserted high true.

This line is not asserted when interface signal select active is true (the controller/formatter is already selected and active) unless the following input/output functions are requested:

- Request Release – If SELECT HOLD is held true for not more than five microseconds, the controller/formatter interprets this as requesting the alternate drive interface to release the controller/formatter. An A/Q director request release command triggers a five-microsecond one-shot, driving the select hold line to the controller/formatter. The controller/formatter acknowledges this request by presenting request release status to the controlling drive interface.

- Forced Release – A 100-microsecond one-shot that also drive the select hold line is triggered by a director function's force release bit. The controller/formatter translates this timed condition as a demand to terminate its current operation and disconnect its selection link with the alternate drive interface.

  The select hold and select active lines are interlocked so that if the controller/formatter drops its select active line to the drive interface (forced release) or the drive interface drops its select hold line, the corresponding select signal drops.

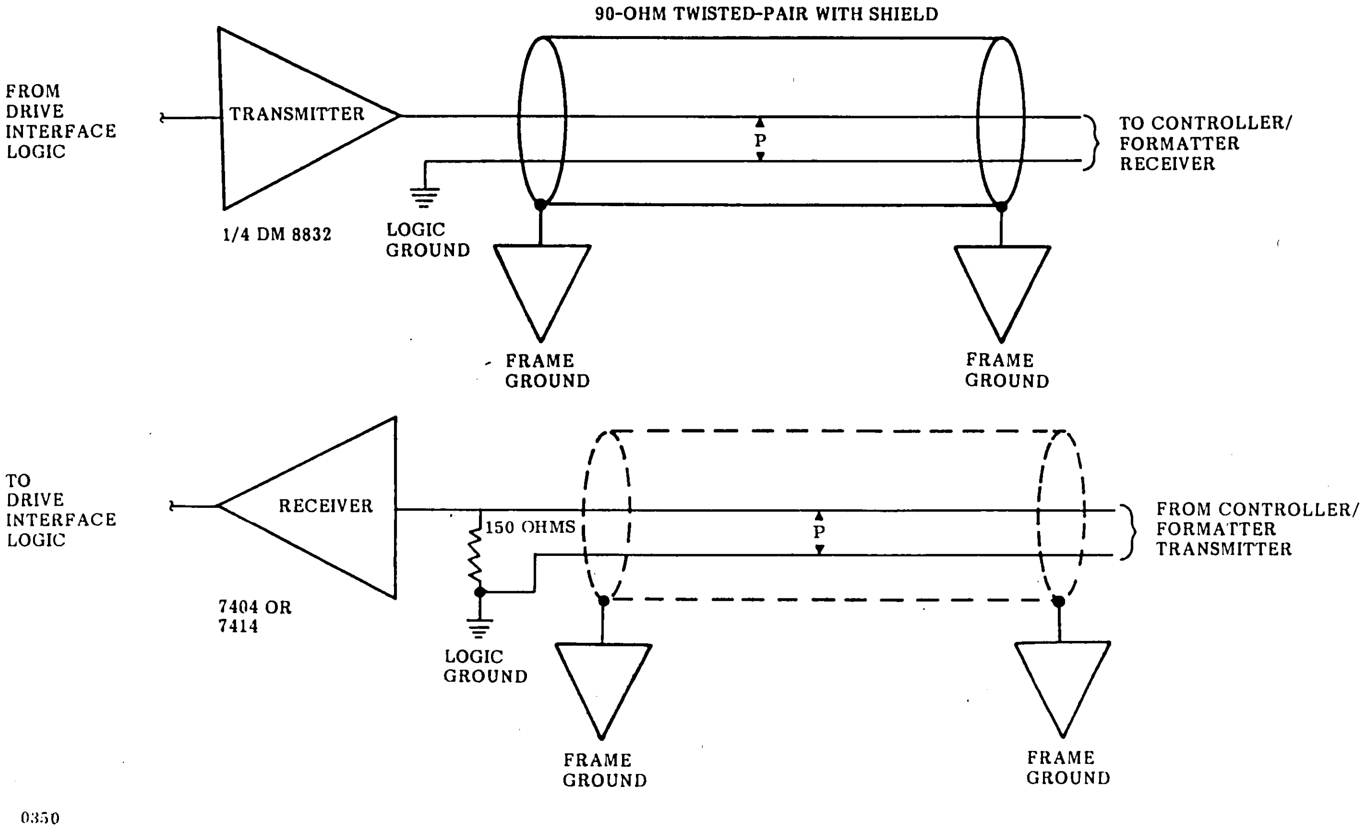


Figure 4-4. Drive Interface Transmitter/Receiver

SELECT ACTIVE

This is the controller/formatter's response to the drive interface's SELECT HOLD request. SELECT ACTIVE goes true when the controller/formatter acknowledges the disk adapter SELECT HOLD, and it remains high during all drive interface/controller/formatter operations. Select active drops when the drive interface clears SELECT HOLD or when the alternate drive interface initiates a forced release function.

In the case of a dual CPU configuration (figure 4-5), the SELECT ACTIVE line from the controller/formatter is serially connected through the drive interfaces. If the first drive interface on this line does not have its SELECT HOLD signal up, it passes the controller/formatter's SELECT ACTIVE on to the alternate drive interface. Conversely, if DA0 is requesting the controller/formatter, it blocks the passing of the controller/formatter's SELECT ACTIVE line to the alternate (DA1).

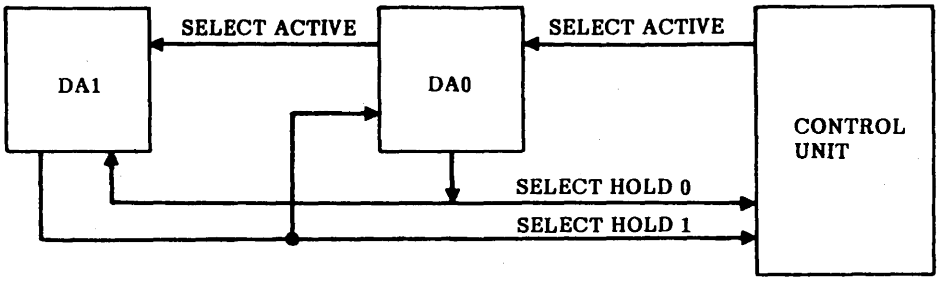
The drive interface's request for the controller/formatter (SELECT HOLD) is also daisy-chained serially through the drive interfaces. The requesting drive interface's SELECT HOLD line passes through the alternate drive interface's logic and is sampled prior to that drive interface issuing any controller/formatter requests. In this way, controller/formatter select contentions are resolved at the hardware level through the priority scheme implemented by the controller/formatter select wire jumpers on both drive interfaces.

TAGOUT

TAGOUT is an interface line from the drive interface indicating that TAGBUS and BUSOUT are to be sampled. TAGOUT remains up until TAGVALID is received or until two microseconds have elapsed, resulting in an abnormal sequence.

TAGVALID

TAGVALID is an interface line from the controller/formatter that indicates the controller/formatter has received a valid command via the TAGBUS and BUSOUT, parity on both busses is correct, and BUSIN now contains information in response to the command. TAGVALID remains up until the drive interface drops TAGOUT.



0351

Figure 4-5. Dual Drive Interface/Controller/Formatter Select Configuration

SYNCOUT

SYNCOUT is an interface line from the drive interface that identifies and synchronizes data transfers. During write data (to the disk) operations, the 100-nanosecond high true pulse on this line indicates that BUSOUT contains valid data. When the controller/formatter is ready for the next byte, it replies to the drive interface with SYNCIN. During read data (from the disk) operations, the SYNCOUT signal to the controller/formatter indicates that the drive interface is ready to receive another byte of data on the BUSIN. When the next byte has stabilized on BUSIN, the controller/formatter signals the drive interface by sending SYNCIN.

SYNCIN

SYNCIN is an interface line to the drive interface that identifies and synchronizes data transfers. The use of the 100-nanosecond high true control pulse on this line is described in SYNCOUT above.

NORMALEND

NORMALEND is an interface line from the controller/formatter that indicates an operation has been terminated successfully. NORMALEND occurs for every successful tag command executed and remains up until the RESPONSE signal is received by the controller/formatter.

CHECKEND

CHECKEND is an interface line from the controller/formatter that indicates that an operation was unsuccessful. Every operation results in either a NORMALEND or CHECKEND signal. The CHECKEND signal remains up until the RESPONSE signal is received by the controller/formatter.

RESPONSE

RESPONSE is an interface signal from the drive interface that replies to the end conditions (NORMALEND or CHECKEND) and remains up until the controller/formatter drops the END signal.

INDEX

INDEX is an interface line from the controller/formatter. The INDEX pulse marks the beginning of physical sector zero from the selected disk drive.

SECTOR

SECTOR is an interface line from the controller/formatter. This signal represents the physical sector pulse from the selected disk drive and marks the beginning of a data record.

RECYCLE

RECYCLE is an interface line from the drive interface that is used to control the modulus 16 counter in the controller/formatter that identifies the end of the address or data field and the beginning of the error correction code bytes. RECYCLE is held in a high state (logical 1) until the drive interface's field length register is decremented to a count of 7. This indicates to the controller/formatter that less than seven information bytes are left to transfer. For address field read or write operation, RECYCLE is true low during the entire operation.

ALERT

There are two ALERT interface lines from the controller/formatter; one line is assigned to each drive interface. A 100-nanosecond pulse on this line indicates that a seek end condition exists on one of the disk drives. In a system with two drive interfaces connected to one controller/formatter, the ALERT signal is sent only to the drive interface that initiated the seek.

INTERFACE BUS DESCRIPTION

Bus In

Bus in consists of nine lines (eight plus parity) from the controller/formatter to the drive interface that are used to:

- Convey status information from the controller/formatter to the drive interface. It is accompanied by TAGVALID, NORMALEND, or CHECKEND control signals from the controller/formatter.

- Transmit data bytes to the drive interface. Data is signified by the controller/formatter's assertion of SYNCIN.

Bus Out

Bus out consists of nine lines (eight plus parity) from the drive interface to the controller/formatter that are used to:

- Convey parameter information related to the tag bus command

- Transmit data bytes to the controller/formatter

Tag Bus Out

The tag bus consists of five lines (four plus parity) from the drive interface to the controller/formatter to define the controller/formatter command. It is stabilized 100 nanoseconds before raising TAGOUT.

INTERFACE COMMUNICATION

Controller/Formatter Selection

The selection of the controller/formatter is achieved by an interlocked handshaking technique. This interlocked feature means that there are no stringent timing requirements for the selection signals. Figure 4-6 illustrates the controller/formatter selection signal relationships. Figure 4-7 illustrates flow charts of the selection process with the top half of the symbols defining the conditions the logic is sampling for and the lower portion of the symbol identifying the logic involved.

Tag Bus Sequence

A controller/formatter command requires a handshaking tag control signal sequence between the drive interface and controller/formatter (figure 4-8).

Data Transfers

Data transfers for read or write operations are synchronized between the controller/formatter and drive interface by handshaking SYNCIN/SYNCOUT signals. The read/write timing sequences are shown in figures 4-9 and 4-10.

REGISTERS

A general description of the drive interface's registers is in section 2. Table 2-1 defines the usage of the program registers and their related software commands. As previously mentioned, there are three groupings of registers: programmable, hardware, and file.

The hardware registers perform internal storage and interface functions and are used by the drive interface's control logic in executing the micro control program.

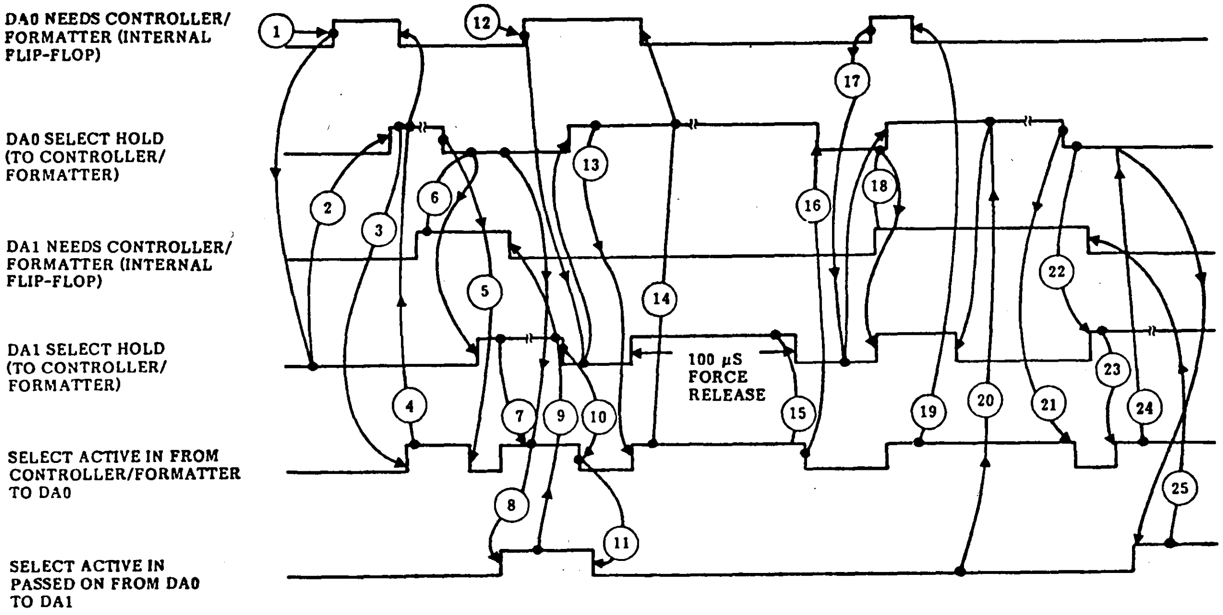
The file registers are 16 addressable 8-bit locations contained in a random-access memory. The usage of the file registers is mixed in that four of the files are software-programmable through controlware intervention. The remaining files are used as internal working and storage registers. The contents of all file registers may be read to the CPU's A register through execution of select file and read file software commands.

FILE REGISTERS/RANDOM-ACCESS MEMORY

Table 4-1 lists the file register names as they are organized in the random-access memory. Files 0 through 8 are loaded directly with the gap A address field that is read from the selected drive during a direct memory access read data or write data operation. When data has been read in, the contents of files A and B that were preset by software are compared by controlware to the data in files 1 and 2. When an equal compare occurs between the actual address information read from the record and the program's requested (target) address, it defines the start of the direct memory access read or write operation. A no-compare condition results in continued reading, under micro control, of the next sequential address field. After 128 attempts (denoted by an initial count preset by controlware in file E being incremented to 0), an error condition is flagged in the drive interface status word (set controller/formatter error flip-flop).

As sector and head boundaries are crossed during the data transfer, the applicable files A and B are incremented by controlware to reflect the current target track address.

Since none of the source registers in the drive interface can reliably be considered as containing zeros at any given time, file C contents are dedicated to providing a constant of zero, which is loaded by micro instructions during a clear disk adapter function. This zero word is used for comparing against various sources of data within the drive interface, resulting in a jump-if-register-contains-zero capability for the micro-instruction set.


DA0 NEEDS CONTROLLER/ FORMATTER (INTERNAL FLIP-FLOP)
DA0 SELECT HOLD (TO CONTROLLER/ FORMATTER)
DA1 NEEDS CONTROLLER/ FORMATTER (INTERNAL FLIP-FLOP)
DA1 SELECT HOLD (TO CONTROLLER/ FORMATTER)
SELECT ACTIVE IN FROM CONTROLLER/FORMATTER TO DA0
SELECT ACTIVE IN PASSED ON FROM DA0 TO DA1
1
12
17
13
2
6
3
16
18
5
14
22
100 µS FORCE RELEASE
23
4
7
9
10
15
19
20
21
24
8
11
25

STEPS 1-4:

DA0 SELECTS CONTROLLER/ FORMATTER.

STEP 5:

DA0 RELEASES CONTROLLER/ FORMATTER.

STEPS 6-9:

DA1 SELECTS CONTROLLER/ FORMATTER.

STEPS 10, 11:

DA1 RELEASES CONTROLLER/ FORMATTER.

STEPS 12-14:

DA0 SELECTS CONTROLLER/ FORMATTER.

STEPS 15-16:

DA1 FORCES RELEASE OF CONTROLLER/ FORMATTER FROM DA0.

STEPS 17, 18:

DA0 AND DA1 SIMULTANEOUSLY SELECT.

STEPS 19, 20:

SELECT CONTENTION RESOLVED TO DA0.

STEP 22:

DA0 RELEASES CONTROLLER/ FORMATTER WITH DA1 STILL REQUESTING.

STEPS 22-25:

DA1 SELECTS CONTROLLER/ FORMATTER.

ALL SIGNALS SHOWN AS HIGH TRUE.

0352

Figure 4-6. Controller/Formatter Selection Protocol

The hexadecimal constant 30 preset in file 9 by controlware equates at an individual bit level to a ready and on cylinder status condition from the selected drive. During an autoload sequence, the micro program uses this constant for comparison against the drive status word number 2 following an RTZ (return to zero) seek command.

RANDOM-ACCESS MEMORY ADDRESSING/FIELD-LENGTH REGISTER

The random-access memory files are accessed by the controlware program during the setup of software commands and by drive interface hardware while reading the gap A address field.

When the random-access memory is selected as a source or destination by the operation field of a micro instruction, the actual file register in the random-access memory that is accessed will be the one addressed by the value contained in FLR00 through FLR03. It should be noted that these lower four bits of the field-length register were previously loaded by a separate micro instruction referencing the field-length register as its destination.

Prior to the start of a read address (gap A) operation, the micro program presets the lower half of the field length register (FLRL) to a value of 8 (hexadecimal). As the first byte read from the address field of the selected drive enters the drive interface, it is strobed by hardware into random-access memory address 8. The FLRL is decremented minus 1 by hardware, and the second byte read is now strobed into random-access memory address/file register 7 (table 4-1). The upper four bits of the field-length register are ignored during these random-access memory references. This operation continues until all ten bytes of the address field have been read from the drive into their assigned file registers (files 8 through 0 and F).

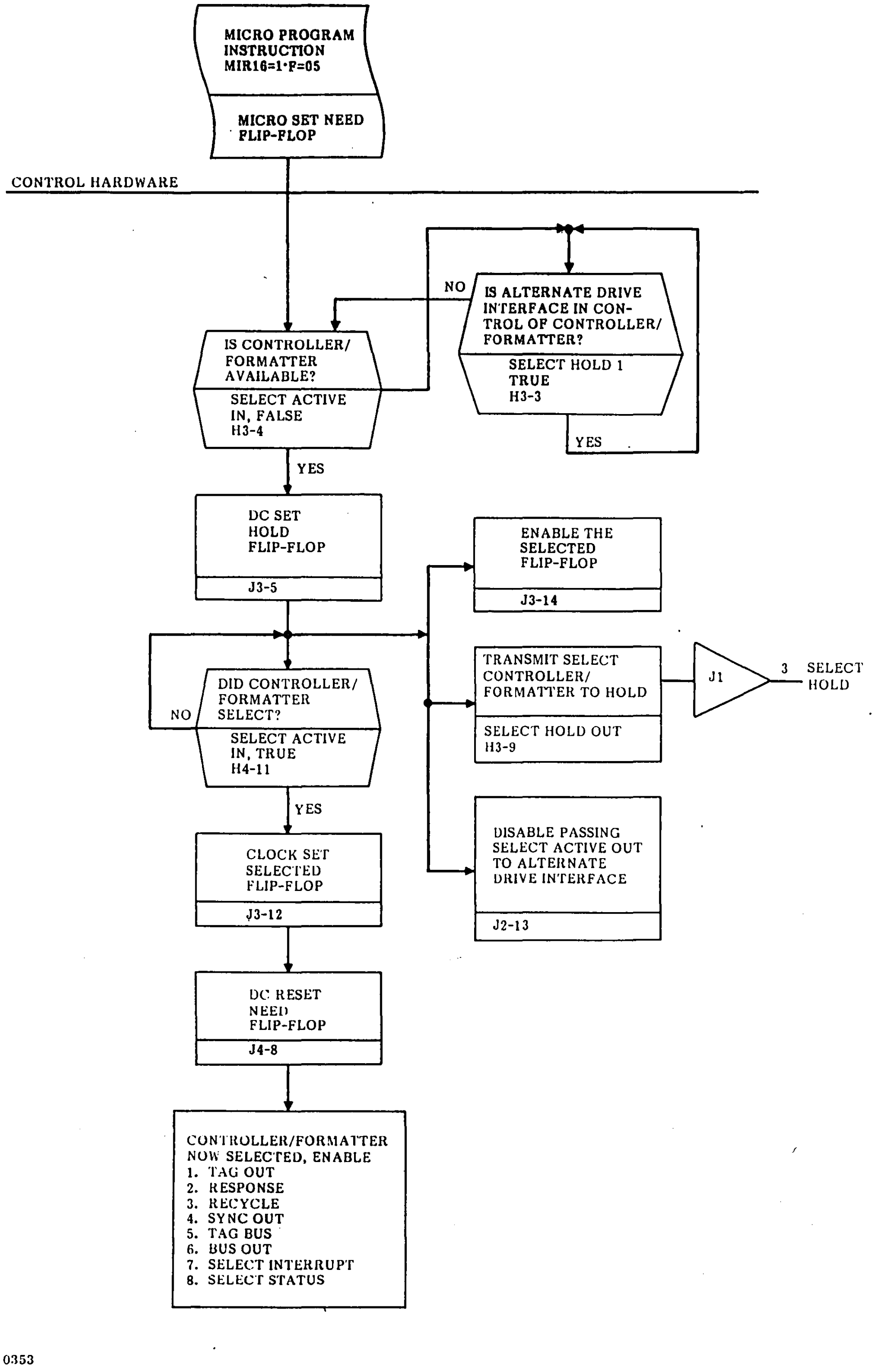
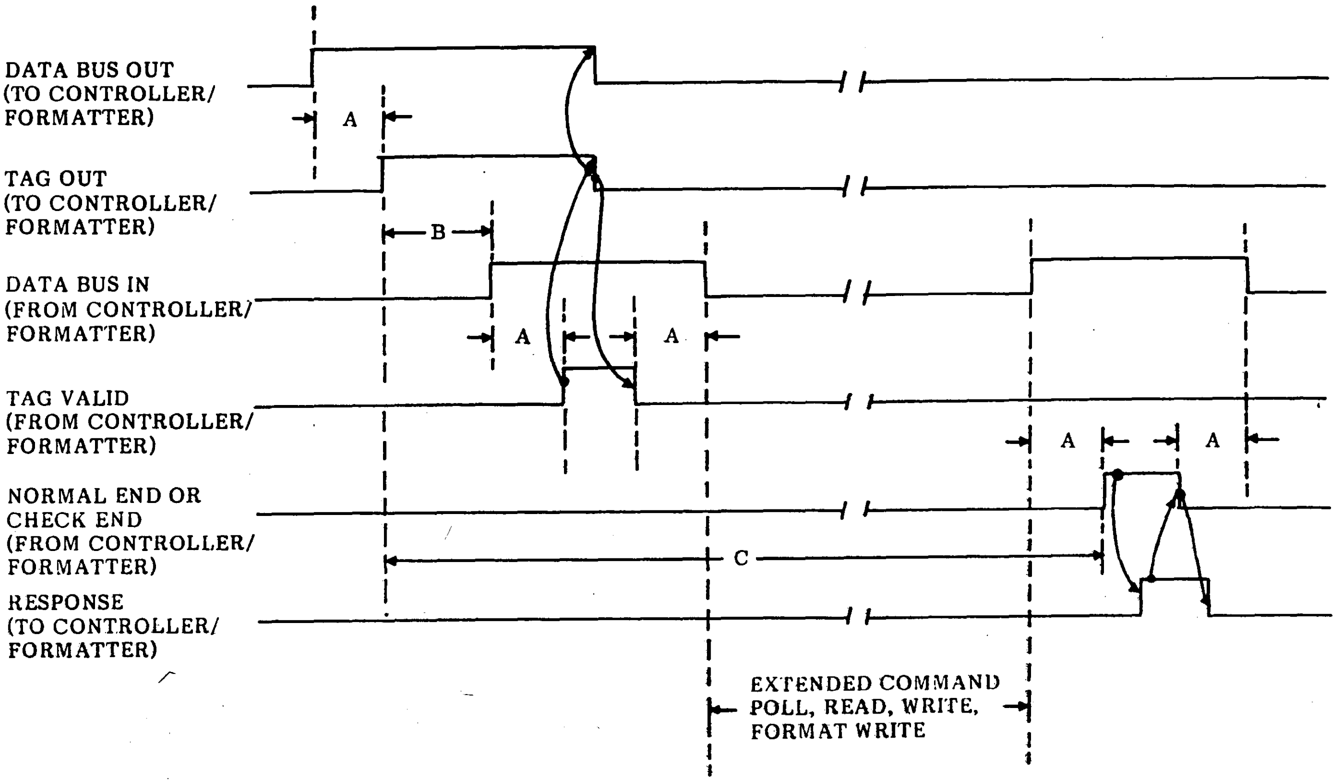

MICRO PROGRAM INSTRUCTION MIR16=1·F=05
MICRO SET NEED FLIP-FLOP
CONTROL HARDWARE
NO
IS ALTERNATE DRIVE INTERFACE IN CONTROL OF CONTROLLER/ FORMATTER?
SELECT HOLD 1 TRUE H3-3
YES
IS CONTROLLER/ FORMATTER AVAILABLE?
SELECT ACTIVE IN, FALSE H3-4
YES
DC SET HOLD FLIP-FLOP
J3-5
ENABLE THE SELECTED FLIP-FLOP
J3-14
TRANSMIT SELECT CONTROLLER/ FORMATTER TO HOLD
SELECT HOLD OUT H3-9
J1
3 SELECT HOLD
NO
DID CONTROLLER/ FORMATTER SELECT?
SELECT ACTIVE IN, TRUE H4-11
YES
CLOCK SET SELECTED FLIP-FLOP
J3-12
DISABLE PASSING SELECT ACTIVE OUT TO ALTERNATE DRIVE INTERFACE
J2-13
DC RESET NEED FLIP-FLOP
J4-8
CONTROLLER/FORMATTER NOW SELECTED, ENABLE
1. TAG OUT
2. RESPONSE
3. RECYCLE
4. SYNC OUT
5. TAG BUS
6. BUS OUT
7. SELECT INTERRUPT
8. SELECT STATUS
0353

Figure 4-7. Controller/Formatter Select Operation


DATA BUS OUT (TO CONTROLLER/ FORMATTER)
TAG OUT (TO CONTROLLER/ FORMATTER)
DATA BUS IN (FROM CONTROLLER/ FORMATTER)
TAG VALID (FROM CONTROLLER/ FORMATTER)
NORMAL END OR CHECK END (FROM CONTROLLER/ FORMATTER)
RESPONSE (TO CONTROLLER/ FORMATTER)
A
B
A
A
C
A
A
EXTENDED COMMAND POLL, READ, WRITE, FORMAT WRITE

A = 100 NANOSECONDS, MINIMUM
B = 1.8 MICROSECONDS, MAXIMUM
C = IMMEDIATE: 3.0 MICROSECONDS MAXIMUM
EXTENDED: 51.8 MILLISECONDS MAXIMUM

EXTENDED OPERATIONS: 51.8 MILLISECONDS MAXIMUM

0354

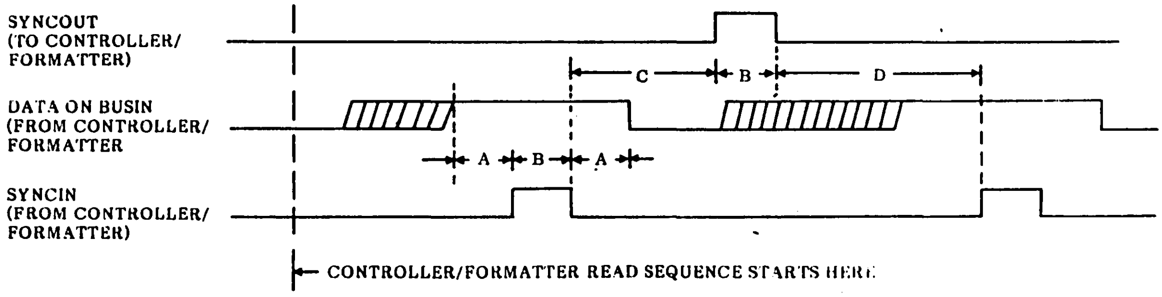
Figure 4-8. Tag Bus Command Sequence Timing

The other application of the field-length register is as a data record byte counter. Once the desired sector is found, the field length word read from the address field and residing in random-access memory file register 0 is transferred to the field-length register (FLR00 through FLR07) by the micro program. The field-length word is a value representing the number of bytes minus 1 in the data field of this sector. The lower four bits of the field-length register are transmitted to the controller/formatter as part of the read or write gap B tag command.

No further random-access memory references are made from this point until the data transfer within this sector is completed. The field-length register is used by drive interface hardware to track the number of data bytes transferred to or from the controller/formatter. With each byte transferred, the field-length register is decremented by 1. In addition, within the controller/formatter, the 4-bit modulo 16 counter (originally set to the same value as in the drive interface field-length register lower four bits) is decremented with each byte transferred. When the counter in the controller/formatter reaches 0, it is allowed to recycle to F and count down again. When the field-length register in the drive interface decrements to a count of 7 or less, this count is decoded and causes the recycle interface line to the controller/formatter to go low. This inhibits the controller/formatter's counter from recycling. Now the controller/formatter's counter and the drive interface's field-length register are in sync. Both reach zero simultaneously, signifying the end of the data transfer. The controller/formatter then reads or writes the seven error correction code bytes at the end of the gap B data.

HARDWARE REGISTERS

These registers may be grouped by their end application into three types: dedicated interface, dedicated function, and general working storage. Those registers falling under the dedicated category are hardwired to interface lines or to some associated logic and serve a single purpose. The general-type register can communicate with most other

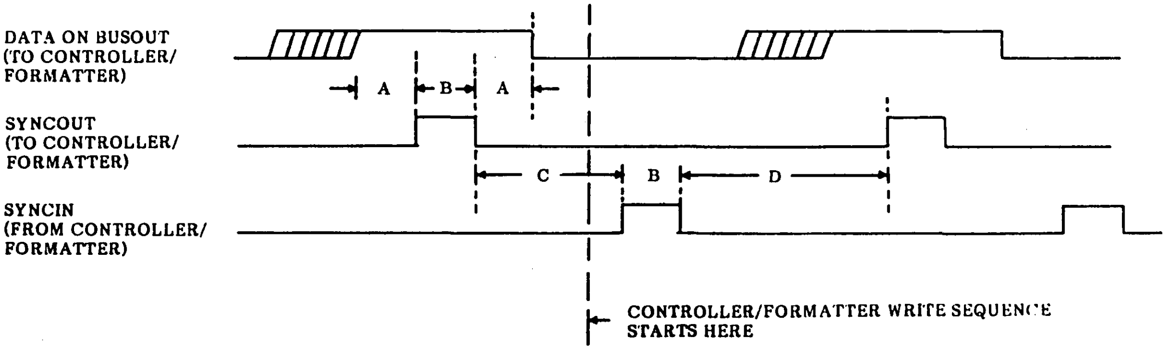


A = 100 NANOSECONDS, MINIMUM
B = 100 NANOSECONDS PULSE
C = 100 NANOSECONDS, MINIMUM; 300 NANOSECONDS, MAXIMUM
D = 100 NANOSECONDS, MINIMUM; 500 NANOSECONDS, MAXIMUM

NOTE: THE CONTROLLER/FORMATTER INITIATES THE TRANSFER BY ISSUING A SYNCIN SIGNAL ALONG WITH THE BUSIN DATA.

0355

Figure 4-9. Drive Interface Data Read Timing



A = 100 NANOSECONDS, MINIMUM
B = 100 NANOSECONDS PULSE
C = 100 NANOSECONDS, MINIMUM; 300 NANOSECONDS, MAXIMUM
D = 100 NANOSECONDS, MINIMUM; 500 NANOSECONDS, MAXIMUM

NOTE: THE CONTROLLER/FORMATTER INITIATES THE DATA TRANSFER BY ISSUING THE SYNCIN SIGNAL TO THE DRIVE INTERFACE.

0355A

Figure 4-10. Drive Interface Data Write Timing

TABLE 4-1. FILE REGISTERS

Random-Access Memory Address	File Register Name		Usage/Contents
0	Read from gap A address field	Data field length	Number of data bytes in the record, minus 1
1		Sector address	Actual sector address read from the record address field
2		Head address	Actual head address read
3		Cylinder address	Lower bits of the target cylinder
4		Cylinder address	Upper bits of the target cylinder
5		Flag byte 4	Usage defined by system software, or not used
6		Flag byte 3	
7		Flag byte 2	
8		Flag byte 1	
9		Autoload status	Constant of 30 (hexadecimal) used as the compare word during the autoload sequence
A		Sector address	Target sector requested by the program or the last sector operated on
B		Head address	Target head selected by the program or the last head selected
C		Zero compare	Contains the constant of zero
D		Timeout counter	Micro-program loop counter
E		Read/write retry counter	Micro-program loop counter
F		Working storage	Holds the temporary file address

logic components in the system and may contain any form of information (data, status, or command) at any given time. Figure 4-11 shows the drive interface registers and their internal bus connections and data paths.

TAG BUS REGISTER

This register is dedicated to the interface transmission of tag codes to the controller/formatter. For most of the A/Q output commands, there is a direct correlation between the lower four bits of the Q register (Qf) and the tag code. The micro-control program, upon transforming Qf, generally passes Q00 through Q03 directly through the bus multiplexer and into the tag bus register where parity is assigned to the tag code prior to transmission. Where differences exist between the tag code and Qf, the controlware loads the tag bus register with the correct tag code from an immediate format instruction. Table 4-2 lists the tag codes as a reference in understanding drive interface/controller/formatter operation and can be compared with the Qf field shown in table 2-4.

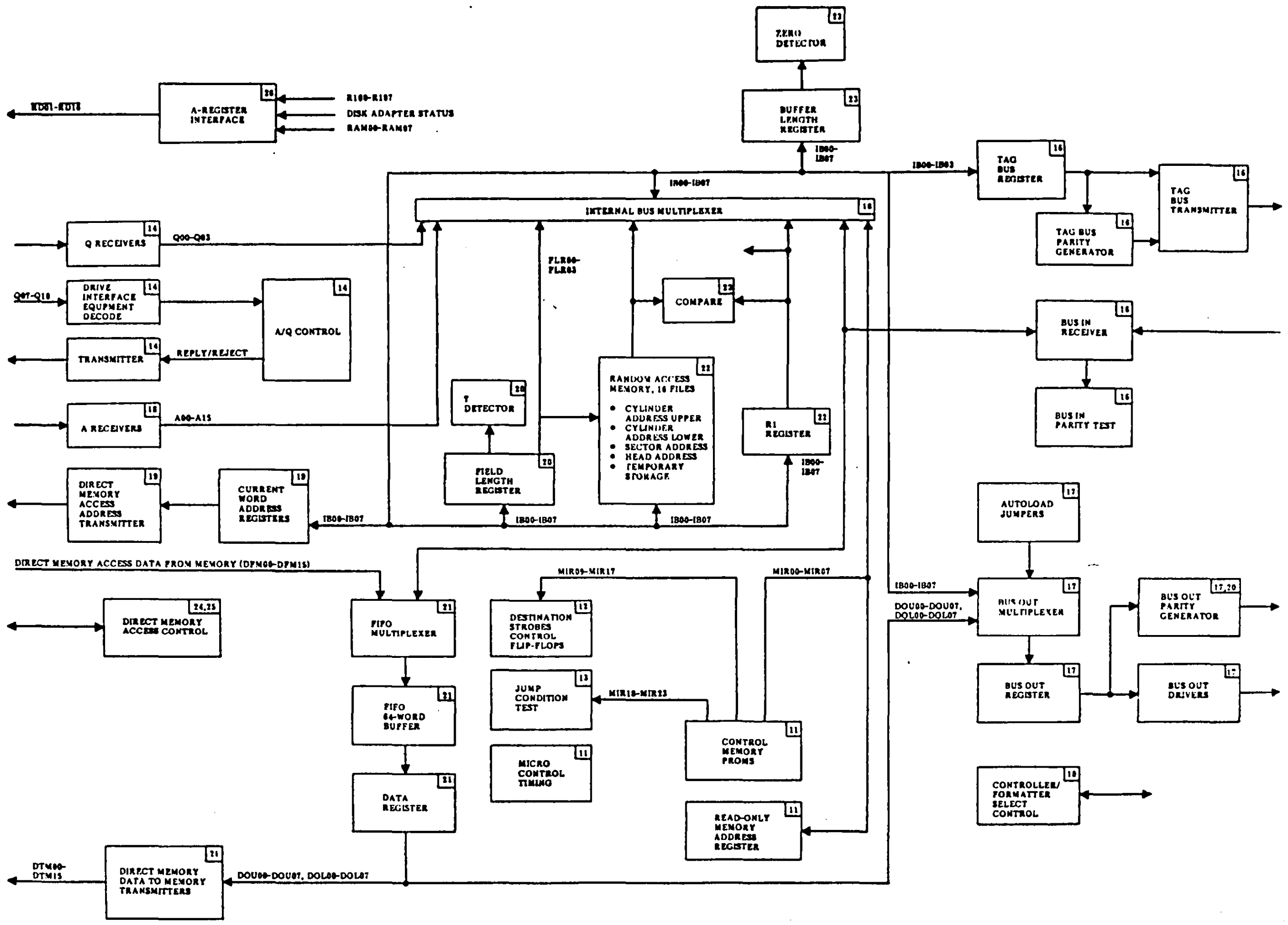
BUS-OUT REGISTER

Receiving its inputs from the data multiplexer, the bus-out register transmits an 8-bit data word to the controller/formatter interface. As evidenced by table 4-2, the bus-out also carries tag code parameter information during a controller/formatter tag command. Once the data transfer starts for a disk write operation, the bus-out register receives CPU memory data via the FIFO data register. For either case, parity is always assigned to the data in the bus-out register and transmitted with it to the controller/formatter.

DEDICATED FUNCTION REGISTERS

Registers in this category have been described both in this section and in section 2. For the purpose of definition, the drive interface's dedicated function registers are:

- Read-only Memory Address Register (MAR) – A nine-bit register/counter loaded with the read-only memory



NOTE: NUMBERS IN BLOCKS REFER TO FIELD PRINT PACKAGE LOGIC DRAWING SHEET

0358

Figure 4-11. Drive Interface Registers

address contained in the operand field (MIR00 through MIR08) of a jump micro instruction or the Qf transform address during a software A/Q input or output command execution; it is used to access read-only memory locations with an address range of 00-1FF.

Note that only the lower eight bits operate as a counter. The most significant bit (page flip-flop) is set or cleared via a jump micro instruction.

- Field-Length Register (FLR) – An 8-bit register/counter loaded from the internal bus. Used to address the 16 random-access memory file registers or maintain data record byte count. During disk read/write operations, the field-length register tracks data bytes transferred between the controller/formatter and drive interface and through hardwired logic and asserts the control signal RECYCLE when there are seven or less bytes remaining in the sector. Also, it sets the FLREQZERO latch when its count is decremented to zero, resulting in the XFERCOMP status flag being set.

- Data Register – Loaded with each 16-bit data word output from the FIFO buffer, this register is loaded at the end of the FIFO output cycle. Its output is gated onto the direct memory access's DTM backplane lines for a disk read operation or passed through the data multiplexer to the bus-out register for disk write operations.

TABLE 4-2. TAG BUS CODES

Tag Bus Bits 0-3	Controller/ Formatter or Drive Command	Bus-Out Bits							
		0	1	2	3	4	5	6	7
0001 (1)	Drive request	-	1 = Clear seek initiated	1 = Clear seek complete	1 = Initiate; 0 = Drop channel request	Logical drive address 2^3	2^2	2^1	2^0
0010 (2)	Poll for seek end	1 = Force poll update	0	0	0	0 = Drive logical addresses 0-7	0	0	0
	Poll, alternate request	0	-	0	0		0	0	0
	Poll, CPU number	0	-	0	1	1 = Drive logical addresses 8-F	0	0	0
	Poll, seek complete	0	-	1	0		0	0	0
	Poll, seek Initiated	0	-	1	1		0	0	0
0011 (3)	Unit select	-	-	-	-	Logical drive address 2^3	2^2	2^1	2^0
0100 (4)	Target sector and drive echo	1 = Load sector and echo 0 = Read sector count	Sector count 2^6	Sector count 2^5	Sector count 2^4	Sector count 2^3	Sector count -2^2	Sector count 2^1	Sector count 2^0
0101 (5)	Load cylinder lower, seek	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0110 (6)	Load cylinder upper	-	-	-	-	-	-	2^9	2^8
0111 (7)	Select head	-	-	-	-	-	Head address 2^2	Head address 2^1	Head address 2^0
1000 (8)	Status - controller/ formatter or drive	-	-	-	1 = Rotational position status	1 = Drive word 2	1 = Physical unit	1 = Controller/ formatter	1 = Drive word 1
1001 (9)	Read	1 = Transfer sector	0 = Address 1 = Data	-	-	Residue byte count remainder after dividing data field byte count + 16.			
1010 (A)	Write	1 = Transfer sector	0 = Address 1 = Data	-	-	2^3	2^2	2^1	2^0
1011 (B)	Error correction code control	-	-	1 = Shift P0 register	1 = Shift P1, P2, P3 registers	-	-	1 = Input error correction code control conditions	1 = Input error correction code control pattern
1100 (C)	Diagnostic	1 = RTZ	1 = Clear attention	1 = Clear check diagnostic	1 = Clear fault status	1 = Clear error recovery	1 = Clear RPS	1 = Clear drive status	1 = Clear controller/ formatter status

TABLE 4-2. TAG BUS CODES (Contd)

Tag Bus Bits 0-3	Controller/ Formatter or Drive Command	Bus-Out Bits							
		0	1	2	3	4	5	6	7
1101 (D)	Error recovery	1 = Early strobe	1 = Late strobe	1 = Positive offset	1 = Negative offset	Offset 2^3	Offset 2^2	Offset 2^1	Offset 2^0
1110 (E)	Format write	-	-	-	-	-	-	-	-
1111 (F)	Drive interface echo data	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7

GENERAL REGISTERS

The 16 random-access memory registers are considered general working registers. Their contents and applications are discussed under Registers above and in section 2.

R1 REGISTER

Loaded from the internal bus, this 8-bit register/counter is completely controlled by controlware. The STROBE R1 signal clocks the source data selected by the current micro instruction into the R1 register. The micro-control program has the capability to add +1 to the R1 register contents through activating the enable, R1INC. It may also compare the contents of the R1 register to the contents of the currently selected file register. This is available through the hardwired comparator on both its and the random-access memory's output. The compare output signal R1EQRAM is one of the conditions that may be tested by the conditional jump program logic. The R1 register can be loaded from or transferred to (figure 4-11) any element in the drive interface with the exception of direct memory access logic. Its output may also be transmitted to the CPU's A register when micro-control signal R1TOA is asserted. The inherent flexibility of the R1 register provides the controlware with a temporary buffer for status, echo pattern data, and error correction code information, as well as a timing or loop counter for program logic decisions. When used in conjunction with the random-access memory and comparator, a software function decode and rudimentary arithmetic capability exists.

DRIVE INTERFACE MICRO PROGRAMMING

Most of the control logic within the drive interface is implemented by micro instructions contained within read-only memory (ROM). This design approach replaces the more conventional hard-wired random logic (interconnected gates and flip-flops). To understand the drive interface control logic, the reader must first understand how the micro-control section operates and then understand how to follow the logical program flow as described by the flow charts.

The following describes the drive interface's micro controller and its operation and implementation. The micro-program flow charts are discussed in section 5 and in the appendixes.

Micro-Control Implementation

The micro controller consists of only three basic sections:

- Read-only memory
- Program sequence control
- Micro-operation decode

Read-only Memory

The read-only memory consists of three factory-programmed integrated circuits, each containing 512 eight-bit words. The three circuits are paralleled within the logic and are accessed simultaneously to provide a 24-bit micro-instruction word.

Program Sequence Control

This section contains the logic necessary to control micro-program sequencing. As each micro instruction is executed, it must determine the address of the next micro instruction to be executed. This is done by conditional jumps. Every micro instruction contains a jump condition field. If the jump condition is satisfied, the sequence control logic causes a program jump to the address specified within the jump address field of the micro instruction. If the jump condition is not satisified, the sequence control causes the current micro-instruction address to be incremented by one.

Micro-Operation Decode

The execution of each micro instruction causes a small part of the total drive interface operation to occur. The operation decode section determines the specific micro operation to be performed. Within the drive interface, there are only two types of operations:

- Moving data from one point to another. This is specified by a source and destination field within the micro instruction. Source data may originate within the CPU's A or Q registers, which are working registers within the drive interface, or within the bus in from the controller/formatter. Destinations consist of CPU interface registers within the drive interface, controller/formatter interface registers within the drive interface, or internal drive interface working registers.
- Setting and clearing control flip-flops. The flip-flops are used to drive CPU and controller/formatter interface lines, to enable and disable internal drive interface hardware functions, and to provide CPU status signals.

Micro-Instruction Word Format

Figure 4-12 shows the micro-instruction word format. The specific fields within the micro instruction are defined below.

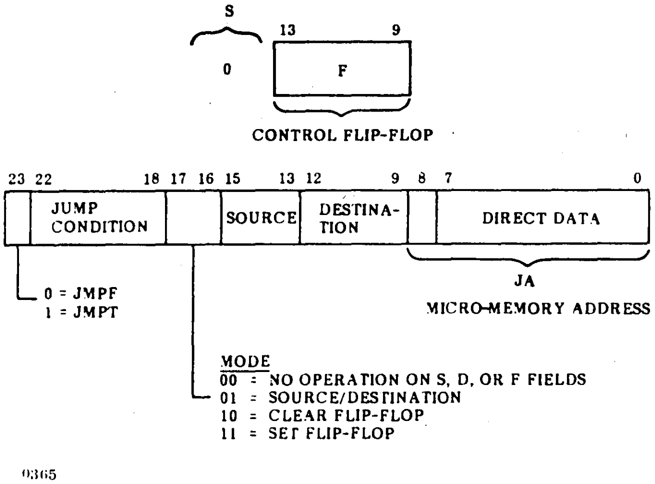


Figure 4-12. Drive Interface Micro-Instruction Word Format

Jump Condition (JC)

Bits 18 through 22 are used to select one of 32 possible jump conditions. Within the drive interface, only 24 of the possible jump conditions are used. The jump conditions allow the micro program to make true/false decisions on both internal and external conditions and branch (or jump) accordingly. Bit 23 of the micro instruction provides added micro-programming flexibility by allowing the selected condition to be tested for either the true or false state.

Mode (M)

Bits 16 and 17 are used to define the specific micro operation to be performed (move data or set/clear control flip-flops). The binary coding of these two bits determines the use of micro instruction bits 9 through 15.

Source (S)

Bits 13 through 15 specify one of eight possible data sources. In this case, data is information gated onto the drive interface's internal data bus. The source field of the micro instruction is meaningful only if the M field is a binary 01.

Destination (D)

Bits 9 through 12 specify one of 16 possible destinations. Decoding of the D field results in a strobe signal that is generally used to load data into a destination register connected to the drive interface's internal data bus. However, for three of the D field codes, the strobes are used for internal drive interface operations that are not related to moving data.

Control Flip-Flops (F)

Bits 9 through 13 define one of 32 possible control flip-flops. Within the drive interface, only 24 of the possible flip-flops are used. This coding for bits 9 through 13 is enabled only when M field bit 17 is a logical 1. In this case, bit 16 of the M field determines whether the addressed flip-flop is set or cleared.

Jump Address (JA)

Bits 0 through 8 specify the address of the next micro instruction to be executed if the selected jump condition is satisfied. Since the JA field consists of nine bits, the micro program jumps directly to any address within the total 512 words of the read-only memory.

Direct Data (DD)

Micro-instruction bits 0 through 7 can be used directly as a data source onto the drive interface's internal data bus. These bits are used for direct data only when it is certain that the jump condition is not satisfied and therefore the jump address is not used. In this case, micro-instruction execution becomes sequential. Note that bits 0 through 7 are never used for both direct data and a jump address within the same micro instruction.

Tables 4-3 through 4-7 specify the binary coding, mnemonics, and definitions for the JC, M, S, D, and F micro-instruction fields, respectively.

TABLE 4-3. JUMP CONDITION FIELD

Micro-Instruction Register Bits					Mnemonic	Hardware Condition – False/True State
22	21	20	19	18		
0	0	0	0	0	TRUE	Unconditional one (true)
0	0	0	0	1	AQEXEC	A/Q execute – flip-flop with clear/set
0	0	0	1	0	A7	A register, bit 7, zero/one
0	0	0	1	1	ADFLD	Address field (Gap A) flip-flop, clear/set
0	0	1	0	0	WRTFF	Drive interface write flip-flop, clear/set
0	0	1	0	1	LFLAG	Loop flag flip-flop, clear/set
0	0	1	1	0	TSTMD	Test mode flip-flop, clear/set
0	0	1	1	1	R107	R1 register, bit 7, zero/one
0	1	0	0	0	PROT	A/Q drive interface operations, unprotected/protected
0	1	0	0	1	BIEVEN	Bus in data parity, odd/even (error)
0	1	0	1	0	CE	Check end interface line, false/true
0	1	0	1	1	--	Not used
0	1	1	0	0	NEORCE	Normal end or check end line, false/true
0	1	1	0	1	XCOMP	Direct memory access transfer complete
0	1	1	1	0	ALARM	Alarm condition false/true
0	1	1	1	1	EMPTY	First-in/first-out data buffer register full/empty
1	0	0	0	0	CUSEL	Controller/formatter presently not/is selected by drive interface
1	0	0	0	1	TAGVAL	Tag valid line, false/true
1	0	0	1	0	SECTOR	Sector pulse, true/false
1	0	0	1	1	INDEX	Index pulse, false/true
1	0	1	0	0	R1FULL	R1 register full flip-flop, clear/set
1	0	1	0	1	BLEQZ	Buffer length register equals zero, false/true
1	0	1	1	0	R1EQRAM	R1 register not/is equal to random-access memory
1	0	1	1	1	AUTOL	Autoload switch not/is set
1	1	0	0	0		Not available
1	1	0	0	1		Not available
1	1	0	1	0		Not available
1	1	0	1	1		Not available

TABLE 4-3. JUMP CONDITION FIELD (Contd)

Micro-Instruction Register Bits					Mnemonic	Hardware Condition – False/True State
22	21	20	19	18		
1	1	1	0	0		Not available
1	1	1	0	1		Not available
1	1	1	1	0		Not available
1	1	1	1	1		Not available

TABLE 4-4. MODE CODES

Micro-Instruction Register Bits		Mode	Definition
17	16		
0	0	No operation	No micro operation is performed; the S, D, and F fields MIR09 through MIR15 are ignored.
0	1	S/D	MIR13 through MIR15 equals the source and MIR09 through MIR12 equals the destination register address (refer to tables 4-5 and 4-6).
0	1	Strobe	MIR13 through MIR15 contain zeros (the source field is not used); the destination field (MIR09 through MIR12) contains a hexadecimal code of B, C, or D (refer to table 4-6).
1	0	Clear	MIR14 and MIR15 equals 0; MIR09 through MIR13 equals the identifier code of the flip-flop/control line to be reset (refer to table 4-7).
1	1	Set	MIR14 and MIR15 equals 0; MIR09 through MIR13 equals the identifier code of the flip-flop/control line to be set (refer to table 4-7).

TABLE 4-5. SOURCE FIELD (MODE CODE = 01 BINARY)

Micro-Instruction Register Bits			Mnemonic	Register Name or Source
15	14	13		
0	0	0	BUSIN	Data bus in
0	0	1	MIR	MIR bits 0 through 7 (direct data)
0	1	0	SR1	R1 Register
0	1	1	SCRAM	Selected file register random access memory
1	0	0	SFLR	Field length register bits 0 through 3 and MIR bits 4 through 7 combined
1	0	1	Q	Q-register bits 0 through 3
1	1	0	AL	A-register lower, bits 0 through 7
1	1	1	AU	A-register upper, bits 8 through 15

TABLE 4-6. DESTINATION FIELD (MODE CODE = 01 BINARY)

Micro-Instruction Register Bits				Mnemonic	Destination Register or Strobe
12	11	10	9		
0	0	0	0	R1	R1 register
0	0	0	1	FLR	Field length register FLR00 through FLR07
0	0	1	0	–	Not used
0	0	1	1	BLL	Buffer length register, lower bits 0 through 7
0	1	0	0	BLU	Buffer length register, upper bits 8 through 15
0	1	0	1	CWAL	Current word address register, lower bits 0 through 7
0	1	1	0	CWAU	Current word address register, upper bits 8 through 15
0	1	1	1	CWAUU	Current word address register, bits 16 and 17
1	0	0	0	TBR	Tag bus register
1	0	0	1	BOR	Bus out register
1	0	1	0	RAM	One of 16 file registers selected by the contents of FLR00 through FLR03
1	0	1	1	R1INC	Increment R1 register (+1 strobe)
1	1	0	0	BLDEC	Decrement buffer length register (-1 strobe)
1	1	0	1	DMACLR	Direct memory access clear strobe
1	1	1	0	-	Not used
1	1	1	1	–	Not used

TABLE 4-7. CONTROL FLIP-FLOP FIELD (MODE CODE = 10 or 11 BINARY)

Micro-Instruction Register Bits					Mnemonic	Control/Status Flip-Flop or Enable Definitions
13	12	11	10	9		
0	0	0	0	0	REPLY	Reply flip-flop (set mode only)
0	0	0	0	1	REJECT	Reject flip-flop (set mode only)
0	0	0	1	0	–	Not used
0	0	0	1	1	–	Not used
0	0	1	0	0	CUERR	Controller/formatter error (set mode only)
0	0	1	0	1	NEED	Need controller/formatter (set mode only)
0	0	1	1	1	XFER	Transfer complete
0	0	1	1	1	EOC	End of cylinder

TABLE 4-7. CONTROL FLIP-FLOP FIELD (MODE CODE = 10 or 11 BINARY) (Contd)

Micro-Instruction Register Bits					Mnemonic	Control/Status Flip-Flop or Enable Definitions
13	12	11	10	9		
0	1	0	0	0	TAGOUT	Tag out
0	1	0	0	1	RESP	Response
0	1	0	1	0	LPFLG	Loop flag †
0	1	0	1	1	TM	Test mode †
0	1	1	0	0	WRT	Write †
0	1	1	0	1	ADRFLD	Address field (gap A) †
0	1	0	1	0	READ	Read
0	1	1	1	1	ALENB	Autoload enable
1	0	0	0	0	BUSY	Busy
1	0	0	0	1	DMAGO	Direct memory access go
1	0	0	1	0	PP	Program protect bit
1	0	0	1	1	–	Not used
1	0	1	0	0	R1TOA	Enable R1 register to A register
1	0	1	0	1	FULWRT	Enable full memory write cycles
1	0	1	1	0	R1MSB	Inhibit output of R1 bit 7
1	0	1	1	1	–	Not used
1	1	0	0	0	–	Not available

† Set/reset condition of flip-flops may be tested by conditional jump instructions.

CONDITIONAL AND UNCONDITIONAL JUMPS AND SEQUENTIAL EXECUTION

Every micro instruction contains a jump condition field. If the jump condition is satisfied, the sequence control logic causes a program jump to the address specified within the jump address field of the micro instruction.

In many micro-programming situations, it is necessary to force an unconditional program jump or to force sequential program execution (sequential execution is required when micro-instruction bits 0 through 7 are used for direct data). These operations are forced by selecting jump condition 0, which is hard-wired to a logical 1 (refer to table 4-3). If MIR bit 23 is a 0 (jump if the condition is false), the jump condition is not satisfied and the drive interface executes the next micro instruction in sequence (sequential execution). However, if MIR bit 23 is a 1 (jump if the condition is true), the jump condition is satisfied. This then becomes an unconditional jump micro instruction.

MICRO-INSTRUCTION CODING

The above capability is used in the coding of micro instructions. The conditional jump micro instructions are coded with the mnemonic JMPT (jump if the condition is true) and JMPF (jump if the condition is false). An unconditional jump is coded as UIP, and sequential micro instructions are coded as SEQ. In addition, the remaining micro-operation mnemonics (S, D, and F) are combined with the above mnemonics to form the possible combinations of micro instructions. These combined micro-instruction mnemonics and abbreviations are used throughout the micro-program flow charts and listing (appendixes A and B) to facilitate understanding.

The 11 different drive interface micro-instruction formats are shown in figure 4-13.

SEQSD

23 16	15 13	12 9	8	7 0
0 0 0 0 0 0 0 1	S	D	▨	DD

SEQFF

23 17	16	15 14	13 9	8 0
0 0 0 0 0 0 1		0 0	F	0 0 0 0 0 0 0 0 0

0 = CLEAR
1 = SET

UJP

23 9	8 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0	JA

UJPSD

23 16	15 13	12 9	8 0
1 0 0 0 0 0 0 1	S	D	JA

UJPFF

23 17	16	15 14	13 9	8 0
1 0 0 0 0 0 1		0 0	F	JA

0 = CLEAR
1 = SET

JMPF

23	18	17 9	8 0
0	JC	0 0 0 0 0 0 0 0 0	JA

JMPFSD

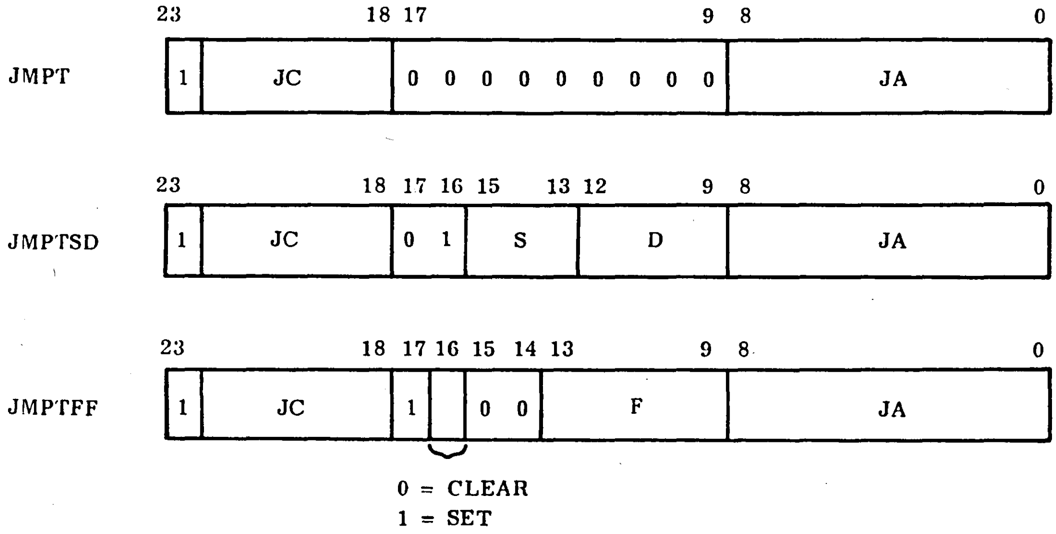
23	18	17 16	15 13	12 9	8 0
0	JC	0 1	S	D	JA

JMPFFF

23	18	17	16	15 14	13 9	8 0
0	JC	1		0 0	F	JA

0 = CLEAR
1 = SET

Figure 4-13. Micro-Instruction Formats (Sheet 1 of 2)



0484

Figure 4-13. Micro-Instruction Formats (Sheet 2 of 2)

This section provides the facing page logic descriptions for the drive interface. To use the information contained in this section, refer to the field print package associated with the drive interface.

Logic drawing sheets 1 through 9 list the voltage distribution and external signal mnemonics in alphabetical order and their corresponding input or output backplane pin. The Page-Zone column allows the user to locate the source and destination of a drive interface interface signal within the drawing package.

Figure 5-1 is the complete block diagram of the disk adapter. It is provided as an aid in relating the theory of operation for any particular logic block to the rest of the drive interface's logic.

CONTROLLER/FORMATTER SELECTION

Controller/formatter selection occurs when a software unit select function is output to the drive interface and the controller/formatter is not already selected. Selection is accomplished through the execution of a dedicated micro routine in the controlware program. In this routine, a micro instruction causes the assertion of micro control signal MSNEED, which sets need flip-flop H2-6. Hold flip-flop J3-6 is next set if the following conditions are met by the other inputs of NAND gate H3-6: SELACTIVEIN is low (H4-11) indicating the controller/formatter is not presently selected, and SELECTHOLD (0 or 1) from the alternate drive interface, via jumper connection E14, is low meaning the alternate drive interface is not selecting the controller/formatter.

Once the hold flip-flop sets, selection of the controller/formatter occurs when it is available, barring a controller/formatter fault condition.

The controller/formatter responds to successful selection by asserting SELACTIVEIN to this drive interface. If ANDed at NAND gate J4-8 with the set condition of the hold flip-flop, the resultant low signal passes through AND gate J2-6 and resets the need flip-flop.

The low-to-high transition occurring on the Q output of the need flip-flop (H2-7) when it resets combined with the hold flip-flop being set clocks set the selected flip-flop J3-10.

This drive interface continues to hold the controller/formatter selected until any one of the following conditions occur:

- A director function with release control bit A06 set is received causing a low-to-high transition at flip-flop J3-4.
- A clear drive interface director function asserts signal CLRDA at NOR gate H5-12.
- A force release condition exists, causing forced release flip-flop K2-6 to be set. Forced release flip-flop K2-6 is enabled set at its pin 2 by the hold flip-flop being set. If for any reason the controller/formatter drops its SELACTIVEIN signal to a low level prior to the normal end of the operation, this low-to-high transition at inverter H4-10 clocks the forced release flip-flop set. This flip-flop's Q output going high passes through NOR gate H5-11. The resultant low resets the hold flip-flop via AND gate J2-10.

If the alternate drive interface requests the controller/formatter via its select hold and the controller/formatter is available for selection as indicated by its SELACTIVEIN signal at a low level, NAND gate J4-11 outputs a low. This prevents the hold flip-flop from setting via its pin 1. This is used to resolve simultaneous controller/formatter select attempts by two drive interfaces in a dual CPU configuration.

During the time that DA0 has the controller/formatter selected, the alternate drive interface is inhibited from receiving the controller/formatter's SELACTIVEIN signal. The hold flip-flop being set places a low from its pin (L4-8) to AND gate J2-13. This disables the transmitting of the active high level SELACTIVEIN signal to DA1. When DA0 does not have the controller/formatter selected as indicated by its hold flip-flop being reset, the SELACTIVEIN signal is propagated to DA1 as signal SELACTIVEOUT (refer to logic sheet 12).

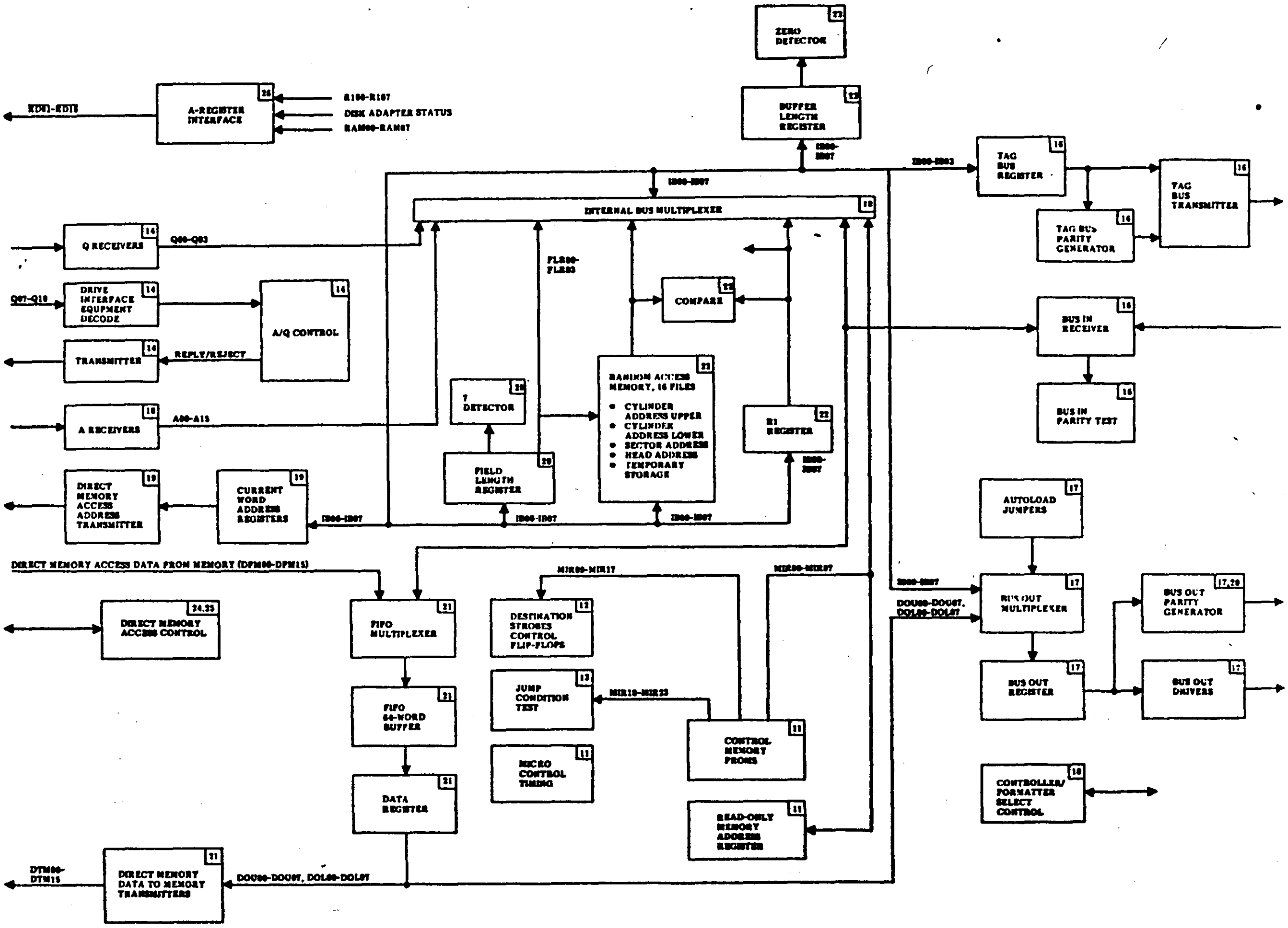
Additional control of the controller/formatter selection process is obtained through drive interface director function commands. If A-register bit 7 is a 1, the director function triggers 5 microsecond request release one-shot K3-7. Its output asserts the SELECTHOLD request line to the controller/formatter for the one-shot time period. The controller/formatter interprets this timed action as a request rather than a forced release.

To force the release of the controller/formatter by the alternate drive interface, a director function with A08 equal to 1 triggers 100-microsecond force release one-shot K3-9. Its output also drives the SELECTHOLD line through OR gate H3-11.

The controller/formatter error flip-flop is shown on sheet 10. It may be set either by the controlware program via the MSERR signal or by the detection of a parity error on bus in during a data input from the controller/formatter. The Q output (H2-9) of the controller/formatter error flip-flop is logically ORed with the other fault conditions at H1-8 to create the ALARM signal.

MASTER TIMING

The clock on sheet 11 provides the only source of master timing within the drive interface. It is free-running and is



NOTE: NUMBERS IN BLOCKS REFER TO FIELD PRINT PACKAGE LOGIC DRAWING SHEET

6366

Figure 5-1. Drive Interface Functional Block Diagram

stopped only by the CLRDA term, which is a result of a master clear, a clear drive interface output function from the CPU, or a power on master clear.

The drive interface master clock consists of two series-connected one-shots with a recirculating output, creating a closed loop oscillator. One-shot E3-10 sets the clock's pulse width at approximately 70 nanoseconds. It is triggered when the pulse rate one-shot E3-6 times out after approximately 130 nanoseconds. The ganged output inverters H4 feed the clock output back to input driver E5-2. The positive trailing edge of the 70 nanoseconds CLK (referenced at H4-2) is passed through NAND gate E5 with the resulting low out retriggering the 130 nanoseconds one-shot, causing a positive pulse to appear on its pin 6. When this one-shot times out, the resultant high-to-low transition on the input of one-shot E3-11 causes it to trigger for approximately 70 nanoseconds. This clock output is used by the read-only memory address register, the A/Q reply and reject flip-flops, and the drive interface register's micro-control strobes.

Note that the drive interface clock timing is particularly critical and should be within ±2 percent of the values shown (130 nanoseconds ± 2.6 nanoseconds ±1.4).

The total clock period is approximately 200 nanoseconds. An individual clock cycle is defined as beginning with the positive leading edge of the 130 nanoseconds pulse.

During this period, several logic events occur. First, the address contained in the read-only memory address register is fetched and the instruction resident there is decoded. Data from the source specified by that instruction is allowed to settle through the internal bus multiplexer, and a destination flip-flop or register is being addressed. On the leading low edge of the 70-nanoseconds clock pulse, the destination register or control flip-flop is strobed.

The positive edge of the 70-nanoseconds clock is used for either incrementing or loading the read-only memory address register integrated circuits, E7 and E8. Selection of the increment or load term is a function of signal JCOND (see Control Memory below).

CONTROL MEMORY

The address of the current micro instruction is contained in read-only memory address register integrated circuits, E7 (higher order) and E8 (lower order), plus read-only memory page flip-flop E6 (sheet 11). Selection of the address loaded into this address register is controlled by two logic signals, JCOND and TRANSFORM, with the address itself supplied by address multiplexer integrated circuits, B7 and B8.

During the execution of a jump format micro instruction, the eight bits of its jump address field (MIR00 through MIR07) are applied to the input of the B7 and B8 multiplexers. If this read-only memory address is not a hexadecimal 1E0, then logic signal TRANSFORM is at a high level. This selects the MIR00 through MIR07 bits on the I1 inputs of the address (B7 and B8) multiplexer, through it to the input of the read-only memory address register. These micro-instruction register bits are only loaded into this register if the test made by the jump micro instruction is successful. In this case, the read-only memory address register's load term JCOND is asserted low, resulting in the loading of a new read-only memory address into this register. Page flip-flop E6 is also enabled by JCOND at NAND gate E5-12 to latch on the state of read-only memory address bit MIR08. If the logic condition does not meet the test made by the micro instruction's jump condition, then signal JCOND remains at a high level. This allows the read-only memory address currently contained in the address register to be incremented by 1 on the next clock transition. The sequential address immediately following the unsuccessful jump instruction is fetched next and executed. The read-only memory address register is clocked with the positive edge of the CLK signal. With every clock cycle, it is either loaded or incremented.

The other form of addressing the read-only memories is through the Q function transform (Qf). A micro routine designated by the program label idle is initiated when the drive interface is powered on. Recognizing any software A/Q command causes the drive interface to set busy status and jump to an area in the read-only memory designated by the program label high. This exit jump from the idle routine is actually a reference to a unique read-only memory address of 1E0 hexadecimal. The only time this read-only memory address is ever referenced is when the drive interface first goes busy; this special address is sensed by NAND gate B6-8 located at the output of read-only memory bits MIR05 through MIR08. Any micro instruction ending with a hexadecimal value of 1E0 through 1FF is decoded as a Qf transform. Decoding of the read-only memory address labeled HIGH in the flow charts asserts signal TRANSFORM at a low that selects the 8 inputs when applied to address multiplexer B8 and B7. These inputs on multiplexer B7 are tied high, providing constant 1 inputs, while the inputs on multiplexer B8 originate from the CPU's Q register, bits 00 through 03. Pin 2 of multiplexer B7 (AQWRT) is high when this is a CPU output from the A operation and low when the operation is an input to the A register. The end result is a read-only memory address in the range of E0 through FF hexadecimal that is generated on the output of the address multiplexer. The specific address between E0 and FF hexadecimal is determined by the state of logic signal AQWRT plus the Qf field bit configuration.

Page flip-flop E6-6, set by MIR08 being a 1 in the IDLE micro routine exit jump instruction, enables the upper 256 addresses of the read-only memories at their A8 address inputs. The actual address range referenced by the transform logic and the page flip-flop equates to 1E0 through 1FF. Each of these addresses contains an unconditional jump micro instruction to the particular execute micro routine that performs the software input/output function defined by the Qf bits. This transform capability allows the drive interface to decode one of 32 possible input/output commands in only one drive interface micro instruction.

READ-ONLY MEMORY ADDRESS REGISTER

The read-only memory address register/counter is preset via its P0 through P3 inputs whenever JCOND (jump condition met) is low at the PE input. The least significant four bits of the address occupy integrated circuit E8 with the count carry out term (pin 15) being connected to the higher order stage's carry-in at E7-10. At each positive edge of the clock applied to pin 2, the lower order stage E8 increments +1. E7 increments upon receiving a carry into its pin 10.

Note that even though the three-state-type read-only memories have a final address of 1FF, the maximum address count of the memory address register is 0FF. The drive interface control program cannot increment across the boundary between the lower and upper 256 read-only memory locations but must jump across the boundary by setting or clearing the page flip-flop.

DESTINATION STROBES AND CONTROL FLIP-FLOPS

Refer to the drive interface micro-instruction word format.

Sheet 12 shows the logic for decoding the destination strobes and for setting and clearing the control flip-flops. The control flip-flop field of the MIR register allows addressing of 32 flip-flops (five bits). The drive interface includes logic for only 24 flip-flops. The flip-flops are divided into three groups of eight flip-flops.

Circuit G4 decodes which group of eight flip-flops are addressed as determined by MIR bits 12 and 13. The decode is enabled by MIR bit 17.

Circuit F2 is enabled when flip-flops 0 to 7 are addressed. The specified flip-flop is decoded from MIR bits 9, 10, and 11. This group of eight consists of discrete flip-flops scattered throughout the drive interface logic.

The second group of eight flip-flops is located within the eight-bit addressable latch at G1, and the third group of flip-flops is located at G2. Each group is enabled out of the group decode at G4. The specified flip-flop is addressed via MIR bits 9, 10, and 11 within the addressable latch and is set or cleared as directed by MIR bit 16.

The drive interface micro-instruction word includes a four-bit destination field. The term destination refers to registers within the drive interface that are loaded from the internal bus as specified within the S field. The 16 destination strobes are decoded from MIR bits 9 through 12 at circuits F5 and G5.

Note that the destination strobes are not used exclusively for loading destination registers; the three exceptions are $\overline{\text{R1INC}}$ (increments the R1 register), $\overline{\text{DMACLR}}$ (clears the DMA section logic), and $\overline{\text{BLDEC}}$ (decrements the buffer length register).

The set/clear control flip-flop field decode on sheet 12 is enabled by MIR bit 17 being a logical 1, and the destination strobes are enabled by MIR bit 16 being a 1 and MIR bit 17 being a 0. This is as specified by the M field of the micro-instruction word format.

JUMP TEST LOGIC

Sheet 13 contains the multiplexing for jump condition (JC) testing. The JC field within the micro-instruction word allows selection of 32 conditions (five bits). The drive interface includes logic for selecting only 24 conditions. The multiplexing of these conditions is divided into three groups of eight conditions each.

The group decode is accomplished by circuit D3 from the decode of MIR bits 21 and 22.

The individual condition to be tested is selected by the multiplexers at locations F1, D1, and D2. The output of the F1, D1, and D2 multiplexers is ORed together at gate H3-12 and then exclusive ORed with MIR bit 23 at gate M9-8. This use of MIR bit 23 provides added micro-programming flexibility in that any condition may be tested for either a true or false state.

If the jump condition is satisfied (i.e., $\overline{\text{JCOND}}$ is low), a new address (the jump address, MIR bits 0 through 8) is loaded into the read-only memory address register on the positive leading edge of the next clock signal.

Note that some of the jump conditions being tested are asynchronous with respect to the drive interface. To avoid the possibility of testing a condition that is changing, these conditions are latched into the registers at locations E1 and E2. The register is updated with every clock cycle.

The cross-coupled gate with output N2-8 is used to debounce the AUTOLOAD switch.

A/Q CONTROL

During an A/Q input/output command, the input/output device address from the Q register (sheet 14) appears on the ADR08/ through ADR11/ backplane lines at comparator N13. The equipment address bits are compared against the address assigned to the drive interface by jumpers E43 through E54. A match causes N13-6 to go high, indicating drive interface selection. The DAROW signal from NOR gate A7-1 is the logical AND of read or write and the drive interface select signal from the comparator.

The four least significant bits from the CPU Q register (signals ADR01/ through ADR04/) are buffered by the A8 inverters and then decoded through circuit A6-8 for a function code of eight (binary 1000). The Q = 8 output of AND gate A6 indicates the drive interface received an A/Q director function or status request command.

The signal PROTOK from circuit A4-6 is a logical 1 if the drive interface is unprotected (as determined by jumper E21, E22, or E23) or if the drive interface is protected and the input/output command is also protected as indicated by the signal PROG-PROT/ being low.

The A/Q read or write flip-flop at C2-6 is enabled by the output of AND gate A2-3. Gate A2-3 is a 1 if the DAROW signal is true and the output of circuits A5-11 and A5-8 are both high (true). Circuits A5-11 and A5-8 are open collector NAND circuits that are wire-ORed together. The inputs to these two NAND gates represent those conditions that must be satisfied for the drive interface to accept and execute any input/output command. Since almost all input/output commands are executed under firmware control, the drive interface is only able to accept input/output commands when it is not busy. The drive interface being not busy requires that the controlware be in the idle state and not executing a previously initiated command. The exception is either a read or write with a function code of eight (drive interface status or drive interface director function). These two commands are accepted at any time, irrespective of a drive interface BUSY signal. The above conditions are logically implemented by circuit A5-8. Output A5-8 is a logical 1 only if the drive interface is not busy or Q = 8 is true.

The other condition that must be satisfied is program protect. This is logically implemented by circuit A5-11. Output A5-11 is true if PROTOK is a logical 1. Gate A4-3 is used to effectively disable gate A5-11 when the drive interface status command is decoded (Q = 8 true and read true), because the drive interface's protected state does not affect the execution of a drive interface status request command.

If the above conditions are satisfied, AND gate A2-3 is true, enabling the A/Q read or write flip-flop to set on the next leading edge of the CLK signal. The flip-flop output signal, AQROW, is sampled by the controlware while in the idle loop. When true, the AQROW signal causes the controlware to initiate a Q transform to decode the function command.

If the conditions described above are not properly satisfied (for instance, a protect violation occurs), then the output of gates A5-8 and/or A5-11 is false. This causes the output of NAND gate A3-13 to be true, which enables the reject flip-flop to set. Note that with every input/output command to the drive interface, either the reject flip-flop or the A/Q read or write flip-flop is set, but never both. The reject flip-flop may also be set under firmware control via the MSREJ signal.

Reply flip-flop C3-6 may be set in either of two ways. For all input/output commands except function code 8, the reply flip-flop may be set under firmware control via the MSREP signal. For function code 8 commands (drive interface status or drive interface director function) the reply flip-flop is enabled by AND gate A2-11. This represents the logical ANDing of AQROW, Q = 8 (via NOR gate A3-1), and the protect conditions of the drive interface being satisfied. In this case, the setting of the reply flip-flop follows the setting of the A/Q read or write flip-flop by one clock cycle, since both flip-flops are triggered by the same signal (CLK).

The A/Q read or write flip-flop and the reply flip-flop or reject flip-flop are cleared when the DAROW signal drops. DAROW goes false when the CPU drops read or write indicating the completion of the input/output command.

INTERRUPT CONTROL

Interrupts to and from the drive interface are handled by the logic shown on sheet 15. A director function is decoded by AND gate A6-8 with its high true output signal FCTN enabling the remaining interrupt enable logic to sample their related A-register function bits.

The interrupt enable register B5 is an eight bit addressable latch. One of its eight flip-flops is addressed by A-register bits 2 through 4 during a director function, while A05 sets or clears the addressed latch. Each 1 set into this register enables its related interrupt driver gate A5 or B4 and flip-flop C2. In this way, software may enable or disable receiving any of the six interrupt conditions shown on the input of the drivers. As an example of interrupts in the drive interface, refer to the ALERT 0/1 interface line at inverter B12. When a drive selected by this drive interface has completed a requested seek operation, the controller/formatter sends a 1 microsecond pulse on the assigned (by jumper) ALERT line associated with this drive interface. This sets the SEEKEND cross-coupled latch B3. In turn, its output sets the A-register bit assigned as the seek complete flag during a status request function. Additionally, SEEKEND is applied to NAND gate A5-4. If software has enabled the seek end interrupt by setting B5-9 to a 1, then NAND gate A5-6 drives RPINTxx1 low. This results in a macro interrupt to the CPU. Note that the NAND drivers A5 and B4 are OR-wired, so that any driver with both inputs high pulls the RPINTxx1 line low true.

Gate A4-11 implements the CLRDA signal, which is the logical OR of the master reset (MR1) signal from the CPU and the clear drive interface direction function from gate A4-8. Gate A3-10 (CLRINT) implements the logical OR of CLRDA and the clear interrupts function from gate A6-11. The CLRINT signal resets all interrupt enables in register B5.

TAG BUS REGISTER

The tag bus register (sheet 16) is loaded by controlware with a four-bit tag code as part of every command to the controller/formatter. The tag bus register receives its source data via the internal bus. Odd parity is generated by the exclusive M9 OR gates and transmitted with the four-bit code to the controller/formatter, via transmitter circuit N9. The transmitter output is enabled only when the controller/formatter is selected (CUSEL).

BUS-IN RECEIVERS

An eight-bit byte plus one parity bit, originating from the controller/formatter, is tested for odd parity by integrated circuit A12 (sheet 16). Signal BUSINEVEN at A12-9 when a high flags a parity error condition sets the controller/formatter/err flip-flop (sheet 10), indicating an error was detected during a data transfer. The BUSINEVEN signal is also tested by the controlware program during status inputs from the controller/formatter.

The data on the bus in also fans out to the internal bus selector (sheet 18) and the data FIFO buffer (sheet 21).

Sheet 16 also shows memory error latch F12 which stores memory fault conditions detected during direct memory access data transfers. The output of this latch drives the alarm logic (sheet 10) and the A register multiplexer for drive interface status (sheet 26).

BUS-OUT REGISTER

The bus-out register consisting of flip-flops J7 and K7, is loaded by either of two load terms via NAND gate K5-11. Controlware asserts signal STBBOR when transmitting commands to the controller/formatter, and the bus-out register is loaded with data associated with that command. During a disk write operation, the direct memory access control logic transfers the write data to the controller/formatter by asserting signal WRTDMAGO. This signal remains low during the entire data transfer and causes the bus-out register to directly pass input data through to its output (become transparent). Parity is assigned to the bus-out register 8-bit byte by parity generator L7. If an even count of bus-out register bits is indicated by a high at L7-9, then the QEV's (Q even) output of L7 is sent as the bus-out parity bit. This makes the total amount of 1s transmitted on bus out an odd count. Transmitter circuits L6 and K6 are enabled by selection of the controller/formatter and output the bus-out register onto the bus out cable.

BUS-OUT MULTIPLEXER

The data input to the bus-out register is selected by bus-out multiplexers J8, J9, K8 and K9. The selection code applied to the multiplexers S1 and S0 inputs is conditioned upon an autoload (ALENABLE), direct memory access write (WRTDMAGO), and the OUTBYTED signal. The multiplexer selection codes are illustrated in table 5-1.

TABLE 5-1. MULTIPLEXER SELECTION CODES

S0	S1	Through Multiplexer		Application
L	L	–	Autoload jumper code	Autoload drive unit number
L	H	(DOUxx)	FIFO Data, ·first byte	Direct memory access data to controller/formatter
H	H	(DOLxx)	FIFO Data, second byte	Direct memory access data to controller/formatter
H	L	(IBxx)	Internal bus data	Command parameter data to controller/formatter (non-direct memory access)

INTERNAL BUS SELECTOR

The source field of the micro instruction, MIR13 through MIR15, selects the source of data applied to the drive interface's internal bus (sheet 18). The multiplexer selection is unqualified; e.g., irrespective of whether the source data is used by this micro instruction or not, multiplexer selection always occurs. Note in table 5-2 that any micro instruction format with a source field code containing 0s still results in the selection of data to the internal bus, specifically the BUSIN from the controller/formatter. The CPU's 16-bit A-register data is handled at the D6 and D7 inputs of the multiplexers as two eight-bit bytes. Data brought through a multiplexer and onto an internal bus (IBxx) is strobed to a destination register only if the format of the current micro instruction specifies source and destination register gating. Source data and selection codes for the internal bus selector are shown in table 5-2.

MEMORY ADDRESS REGISTER

The current word address or memory address register (sheet 19) is an 18-bit counter. It is initialized to a main memory address in the CPU via the set first word address lower and upper output commands. Under control of the drive interface micro program, the memory address register is loaded eight bits at a time by the $\overline{\text{CWAL}}$ and $\overline{\text{CWAU}}$ destination strobes. The remaining two bits are loaded into the memory address register by strobe signal $\overline{\text{CWAUU}}$. When main memory in the CPU accepts the request for a direct memory access cycle from the drive interface, it responds with the MCA (memory cycle accept) signal. This gates memory address register bits 0 through 15 through the open collector transmitter gates G13, H13, G11 and H11. Gate G13-11, whose output is shown as DMA-MAB01/, forms the least significant bit of the requested memory address.

TABLE 5-2. SOURCE DATA SELECTION

Selector Code			Mnemonic	Data Through Multiplexer
MIR15	MIR14	MIR13		
0	0	0	BUSIN0-BUSIN7	Bus in data from controller/formatter
0	0	1	MIR00-MIR07	Direct data from micro-instruction register
0	1	0	RI00-RI07	RI register contents
0	1	1	RAM00-RAM07	Random-access memory file register data
1	0	0	FLR00-FLR03 and MIR04-MIR07	Field-length register and micro-instruction register combination
1	0	1	Q00-Q03	CPU Q-Register bits 0 through 3
1	1	0	SD01-SD08	CPU A-Register bits 0 through 7
1	1	1	SD09-SD16	CPU A-Register bits 8 through 15

For memory sizes above 65K, the two most significant bits of the memory address register, DMA-MAB17/ and DMA-MAB18/, must be connected. This allows the drive interface to directly address up to 131K of memory in a single CPU system or 262K in a dual CPU configuration. Note that MAB18 is gated onto the backplane with the earlier MCR (memory cycle request) signal. This allows the memory's bank control logic adequate time to determine the selected memory (local or external) bank.

At the trailing positive edge of the direct memory access request-accept signal (DMA1-RA/), the memory address register is incremented +1 at G12-5. The memory address register continues to increment for each data word transferred until the buffer length register (sheet 23) goes to a zero count. In the event the memory address register reaches a maximum count (all 1s), the next increment causes the memory address register to equal zero.

FIELD-LENGTH REGISTER

The field-length register (sheet 20) is loaded from the internal bus by controlware via the destination strobe, FLR. The field-length register serves a two-fold purpose.

H7 of the field-length register, whose outputs are FLR00 through FLR03, points to one of 16 random-access memory file addresses. Any micro-program access to the random-access memory is with the file register whose address was previously loaded into this portion of the field-length register.

The second application of the field-length register is during data transfers between the drive interface and the controller/formatter. In this application, the field-length register is used to determine the end of the data record and the beginning of the error correction code bytes for both read and write operations on either the address or data field of a sector. From the controller/formatter's standpoint, either the address or data field may be any length from one byte up to the maximum number of bytes per track. This is restricted by the drive interface so that the address field always contains 10 bytes, but the data field may contain anything from one to 256 bytes. Note that this may be further restricted by CPU software or the disk sector conventions of the system. In any case, the drive interface must signal the controller/formatter during the data transfer when the data field ends and the error correction code field begins.

Prior to a read or write command from the drive interface to the controller/formatter, the field-length register is set by controlware to a value dependent upon the operation being performed. In the case of a read or write address field, the field-length register is set to 9 (i.e., the number of bytes in the address field, minus one). In the case of a read or write data field, the field-length register is set to the field length value that is always read from the address field of the sector prior to initiating the read or write data operation. As part of the read or write command transmitted to the controller/formatter from the drive interface via the tag bus, the least significant four bits of the bus-out register are set (via controlware) from the least significant four bits of the field-length register. Within the controller/formatter, this value is stored in a four-bit modulo 16 counter (table 4-2).

During the data transfer between the drive interface and the controller/formatter, the field-length register is decremented by 1 with each byte transferred by signal DECLFR, which originates in the drive interface's direct memory access control logic. At the same time, the modulo 16 counter in the controller/formatter is also decremented. As the transfer continues, the controller/formatter's counter is allowed to underflow (i.e., count sequence goes 2, 1, 0, F, E, D, C, B. . .) as long as the RECYCLE signal originating in the drive interface is a logical 1 (high). The RECYCLE signal (sheet 20) goes false (low) when the drive interface's field-length register contains a count equal to or less than 7. This is determined by comparator J6, whose A inputs are hardwired for detecting a count of 7 remaining in the lower stage of the field-length register (H7), and its B inputs sampling for a 0 content in the higher stage (H6). Note that the RECYCLE signal is held false during address field reads or writes by the ADRFIELD signal through inverter G3-6. When the RECYCLE signal goes false, the counter in the controller/formatter is inhibited from underflowing (recycling). As the transfer completes, both the field-length register within the drive interface and the modulo 16 counter within the controller/formatter both equal zero with the last byte transferred. When this occurs, cross-coupled latch M6 sets by the low-going borrow term from field-length register H6-13. At this point, the controller/formatter either reads or writes the error correction code bytes, depending on the operation being performed.

The above sequence is repeated for each sector operated on. The hardware-implemented address field store sequence is not directly shown in the logic but is a part of the field-length register operation. This sequence occurs during the read address portion of a data field read or write operation. Prior to reading or writing the data field of a sector, the drive interface must first read the address field of the sector and verify that the information contained in it indicates that this is the proper (target) sector to be operated on. The address field read operation is accomplished as described above, but in addition, the information bytes input from the controller/formatter are stored in the drive interface's random-access memory file register in sequential addresses as determined by the field-length register. The address field data is read onto the drive interface's internal bus from the BUSIN by the instruction source field in the micro-instruction register being zero. The address field data is stored into the random-access memory file registers by logic on sheet 24. Each data byte sent to the drive interface on the BUSIN is accompanied by interface signal SYNCIN. Through inverter N8 (sheet 24), this signal becomes DMASYNCIN. Applied to NAND gate M6-6, it is ANDed with CUSEL, creating signal DECFLR. The FLR was previously set to a value of 9 by controlware. Signal DECFLR causes the field-length register to decrement to a count of 8. The field-length register is now pointing at the random-access memory file address that holds flag byte 1, which was read from the address field (table 4-1).

Signal DMASYNCIN is also applied to NAND gate M8-11 (sheet 24). NAND gate M8-8 is allowed to assert SETR1FULL low true coincident with the controller/formatter's SYNCIN because the controlware does not set control flip-flop DMAGO (N2-13) until the address field read completes.

On logic sheet 22, signal $\overline{\text{SETR1FULL}}$ through AND gate F4-3 causes the contents of the internal bus (flag byte 1 read from the address field) to be written into random-access memory file address 8. The second address field byte (flag byte 2) is written into random-access memory file address 7 through the automatic decrementing of the field-length register (caused by the controller/formatter's SYNCIN signal).

The remaining address field bytes (sheet 22) continue to be written into the files, in descending order, until the controller/formatter terminates the operation by asserting its END (NORMAL or CHECK) interface signal to the drive interface. R1FULL cross-coupled latch E5 (sheet 22) is not used for this operation.

FIRST-IN/FIRST-OUT DATA BUFFER

INPUT DATA MULTIPLEXER

The input to this multiplexer is 16 bits of data from main memory for a disk write operation (multiplexer pin 1, WRITE high) or two eight-bit bytes of data (for a disk read operation originating from the BUSIN signal (sheet 21). The outputs of multiplexers J13, K13, L13, and M13 are high true for data ones.

FIRST-IN/FIRST-OUT BUFFER MEMORY

Each FIFO circuit (J12, K12, L12, and M12) consists of a four-bit wide by 64 words deep data buffer memory (sheet 21). Bytes are input parallel by the SHIFTIN0/1 signal and fall through to the lowest available output stage. The input ready signal (INPRx) from each FIFO circuit is the only indication that an available cell exists for input data.

The first byte put into the FIFO circuit becomes the first byte available at the output as indicated by output ready signal OUTRx. The second byte in is the second byte out, and so on. Data is taken from the FIFO buffer by the SHIFTOUT signal and stored in the data output register (J11, L11, and M11). As each word is taken out of the FIFO buffer, the remaining data stored falls through to the next lower output location. Note that SHIFTIN (data into the FIFO circuit) and SHIFTOUT (data out of the FIFO circuit) operations occur asynchronously with each other.

The purpose of the FIFO circuit is to provide rate averaging and temporary storage for data transferred between CPU memory and the controller/formatter. During disk write operations, the FIFO buffer enables the drive interface to get ahead of the relatively high speed transfer demands of the controller/formatter and drive. During disk read operations, the FIFO buffer provides temporary storage for the incoming data and allows the drive interface to store data in CPU memory at a rate determined by other system demands on the memory.

The shifting in of data to the FIFO buffer is synchronized to direct memory access data available from main memory ($\overline{\text{DMA-MDS}}$) for a disk write operation. For a disk read operation, each data byte input on BUSIN is flagged by the controller/formatter's SYNCIN. FIFO circuits L12 and M12 receive the first (left or most significant) byte and store it with SHIFTIN1. The second byte (right or least significant) is stored in FIFO circuits J12 and K12 by the SHIFTIN0 term. The storing of successive bytes continues to bounce back and forth between the FIFO stages with each SYNCIN.

After input data falls through to the FIFO output, a full 16-bit data word is transferred to the CPU's main memory. The OUTRx outputs from pin 4 of the FIFO circuits go to a high state, indicating the FIFO buffer has output data ready. This triggers a 100-nanosecond one-shot, generating the SHIFTOUT signal to the FIFO buffers. At the trailing edge of SHIFTOUT, the data output register flip-flops (J11, L11, and M11) are clocked to latch the output of the FIFO buffer. Data from the output register is transmitted to memory by the open collector circuits J10, K10, L10, and M10 for a disk read operation. For disk write operations, data from the output register is routed to the bus out multiplexer for transmission to the controller/formatter eight bits at a time.

RANDOM-ACCESS MEMORY AND R1 REGISTER

The random-access memory integrated circuits, H8 and H9, are addressed by the contents of FLR00 through FLR03, and input data from the internal bus is stored by signals $\overline{\text{STBRAM}}$ or $\overline{\text{SETR1FULL}}$ (sheet 22). Each random-access memory integrated circuit holds 16 four-bit words. The output data from the selected file is the compliment of the input data to that file.

The R1 register is a general-purpose internal data handling register, consisting of F9 and F8. It can be incremented (+1) under firmware control via the $\overline{\text{R1INC}}$ signal.

The contents of the R1 register and the random-access memory can be tested for equality by comparators G9 and G8. Under micro control, signal $\overline{\text{R1MSB}}$, when asserted low, causes the comparator to ignore bit 7 of the R1 register by forcing that input to a 0. The high true output of comparator G8-6, R1EQRAM, can be tested by a conditional jump micro instruction.

BUFFER-LENGTH REGISTER

The buffer-length register (sheet 23) is used during disk read and write operations to keep track of the number of words transferred to or from CPU memory.

The buffer-length register is initially loaded under control of the drive interface's micro program via the $\overline{\text{BLL}}$ and $\overline{\text{BLU}}$ destination strobes as a result of a software set buffer length output command. As direct memory access data transfers occur, the buffer-length register is decremented (-1) by the $\overline{\text{DECBLR}}$ signal with each 16-bit word transferred. This decrement occurs as a result of the SHIFTIN0 signal, which goes low when the second byte of the 16-bit word has been stored in the FIFO buffer. During test mode read operations, the buffer-length register decrement is initiated through NOR gate N1-4 by the $\overline{\text{MCRCLR}}$ signal. The buffer-length register is also decremented under micro program control via the $\overline{\text{BLDEC}}$ signal.

Cross-coupled latch N6 is used to capture the buffer length equal zero condition (underflow) via the BOR (borrow) term from G6-13. The circuitry required to clear the FIFO buffer memory is also shown on sheet 23. The DMACLR signal occurs under firmware control. The discrete circuitry (R2 and C24), is required to produce a pulse of sufficient duration to clear the FIFO buffer.

DIRECT MEMORY ACCESS CONTROL

Sheet 24 shows part of the drive interface direct memory access control logic. Because of the high data transfer rate of the controller/formatter and disk drive, the direct memory access logic is not directly implemented by firmware but by hardwired logic. This logic is enabled by the controlware via three micro-control signals: READ, WRITE, and DMAGO. READ and WRITE govern the direction of the data transfer, and DMAGO is used to start and stop the actual data transfer operation. Note that the terms READ and WRITE are referenced to the controller/formatter; that is, READ implies read from disk.

During read from disk operations, the in data byte flip-flop, INBYTE0, at M3 determines whether the incoming byte is stored in the upper or lower half of the FIFO buffer memory word. The outputs of the in data byte flip-flop are fed back to NAND gates M8-6 and M8-12. As long as the buffer length register does not equal zero (BLREQZERO), the SYNCIN pulses from the controller/formatter that accompany each data byte sent to the drive interface alternately generate the SHIFTIN0 or SHIFTIN1 signals to the FIFO buffer. Refer to the operational description of the FIFO data buffer above (logic sheet 21) for the explanation of the SHIFTIN0/1 signals. The logical OR of these signals at NOR gate L5-10 causes the in data byte flip-flop to toggle (alternately set and clear). The storage of data in the FIFO is disabled when the buffer length register reaches zero.

During a disk write operation, both SHIFTIN terms are generated simultaneously by the low output of NAND gate M6-8 each time main memory data is available to the drive interface (MCRCLR). The out data byte flip-flop at M3 (OUTBYTE0) is toggled only for disk write operations. Each SYNCIN signal to the drive interface, indicating the controller/formatter is ready for the next data byte, clocks this flip-flop from NAND gate M5-8. The output of the out data byte flip-flop at M3 alternately selects the upper and lower byte from the FIFO output via selection of the bus out register multiplexer.

When set, the MEMORY REQUEST flip-flop, K2-10, initiates a direct memory access data transfer cycle between main memory and the drive interface's FIFO buffer. NAND gate L4-6 causes direct memory access requests during disk write operations. The buffer-length register being nonzero (pin 5 high) and the FIFO input ready (IRC, pin 3 high) requests direct memory access data until either the FIFO buffer becomes full or the buffer-length register equals zero.

For disk read operations, the FIFO buffer is unloaded one word at a time and sent to the CPU's main memory. In this case, memory requests are initiated via NAND gate N2-3 as long as the output buffer register becomes full (N2-2 high).

For either read or write operations, the MEMORY REQUEST flip-flop, M2, is reset by the memory data strobe signal, DMA-MDS/. The PROTECT signal inputs to NAND gate L2-8 are set by the controlware during disk read operations if the drive interface's program protect conditions are such that writing into protected CPU memory is permitted.

The DECFLR signal from M6-6 causes the field length register to decrement with each byte transferred during either read or write operations.

The SETR1FULL signal from NAND gate M8-8 is asserted only during the read address field of a read or write operation. Once the address field bytes are read into the random-access memory, the data field transfer follows. Upon reaching this point, controlware sets micro-control DMAGO, disabling SETR1FULL.

The DATAENABLE signal from gate J5-11 enables the open collector transmitter gates (sheet 21) for all data-to-memory (DMA-DTMxx) lines. Data is transmitted to memory when the request accept signal (DMA1-RA) is logically true (low).

FIRST-IN/FIRST-OUT CONTROL

Sheet 25 shows that portion of the direct memory access control logic not shown on sheet 24. Additionally, the logic controlling the FIFO operation is also described in this section.

The output data register full flip-flop at N5-10 operates in conjunction with the one-shot circuit at N3 to move data from the FIFO buffer to the output data buffer register. The SHIFTOUT signal from one-shot N3-6 causes data to be output from the FIFO buffer and loaded into the buffer register. The one-shot is conditioned by the buffer register being empty (REGFULL signal false) and the output of all FIFO integrated circuits being ready (OUTRx signals). When the 100-nanoseconds SHIFTOUT one-shot is triggered, the output data register full flip-flop, N5, is set. This flip-flop is reset when the data word has been stored in memory during disk read operations via AND gate M4-11 or when the second byte of the word has been output to the controller/formatter during disk write operations via AND gate M4-8.

A transfer-complete condition indicated at NAND gate N6-8 (XFERCOMP) is based on the following states: NAND gate N11-12 is high (BLREQZERO) when all data words have been transferred between the drive interface's FIFO buffer and main memory, pin 9 is high 5 microseconds after the FIFO buffer has been emptied of all data, pin 13 is high when the byte count contained in the field-length register is zero (FLREQZERO), and pin 10 is high for the FIFO data register empty (BREMPTY). In test mode, the only condition sensed is that the buffer-length register previously set to 64 by the micro program is now zero. The XFERCOMP signal from N6-8 is sampled as a conditional jump condition by the controlware.

The lost data flip-flop at N5 indicates an error condition due to the drive interface and/or CPU's inability to keep up with the data transfer rate demands of the controller/formatter. For a read-from-disk operation, this error occurs if the controller/formatter has an input byte available but the drive interface's FIFO buffer is full (i.e., no place to store the data byte). The FIFO full condition is indicated by a logical 1 at NOR gate N10-1 as a result of the FIFO input not being ready (INPRx signals).

For a write-to-disk operation, a lost data error occurs if the controller/formatter requires an output byte but the drive interface's FIFO data register is empty. This condition is generated by NOR gate N7-10.

The logical NOR of these read or write lost data conditions is generated by NOR gate N7-4. For both read and write operations, the lost data flip-flop is clocked by the DMASYNCIN signal. A clear drive interface director function ($\overline{\text{CLRDA}}$) command clears the lost data flip-flop.

The $\overline{\text{CLRBR}}$ signal from M5-6 causes the FIFO data register to clear between output words during disk write operations. This is required for a partial sector write operation where the remainder of the sector must be filled with zeroes.

A-REGISTER MULTIPLEXER

The multiplexer shown on sheet 26 feeds the CPU's A-register input lines (RD01 through RD16). With the exception of the hardware-asserted $\overline{\text{DASTAT}}$ signal at NAND gate F11-4 for drive interface status requests, the controlware program controls the multiplexer selection. Depending on the type of CPU input operation being performed, the multiplexer can input any of the following to the A register: drive interface status, random-access memory (file register data), or the R1 register containing controller/formatter or drive status. Micro control signal R1TOA controls the placement of the R1 register to either the upper (bits 8 through 15) or lower (bits 0 through 7) half of the A register.

Two types of multiplexer integrated circuits are used with different enable/selection designations; i.e., S1 and S0 versus E and S terms. The following illustrates multiplexer input selection with the logic levels obtained from NAND gates F11-6 and F11-8.

Multiplexer Inputs		A-Register Inputs	
S/S0	E/S1	AU	AL
L	L	R1 register	Random Access Memory
H	L	Drive Interface status	Drive Interface status
L	H	Zeroes	R1
H	H	Zeroes	Zeroes

CONTROLWARE

Previous sections of this manual described the drive interface's operation, a repertoire of micro instructions, and a logic analysis of the hardware. The assimilation of this information suffices in comprehending those drive interface operations where the actions of its internal logic are governed by hardware alone or combined with controlware. However, there are unique operations, such as the drive interface's internal reading of the address field from the drive. For this, the hardware and controlware, though inter-related, operate asynchronous to each other. Here, a series of micro instructions is operating parallel with a sequence of hardware-driven operations, both related by drive interface operation but individually controlled by different logic. The key to analyzing this type of operation lies in understanding the execution sequence and hardware interaction of grouped micro instructions. The controlware comprising the drive interface micro program is documented by instruction flow charts and a program listing. The program listing contains both the assembler program's macros and the micro-instruction code.

MACROS

The assembler macro is an abbreviated wording of the operation performed by the various fields of the micro instruction. An example of an assembler macro for the drive interface is JMPT R1EQRM, JA251. This macro states that the program flag is true, indicating that the (drive interface's) R1 register equals (the selected contents of) the random-access memory then jumps to jump address 251; otherwise, execute the next sequential micro instruction. The macro assembler program assembles this macro statement into a 24-bit micro instruction code shown here in hexadecimal notation as D80152. This example shows there is a one-to-one relationship between a macro statement and a micro instruction. The following text lists the macro fields and abbreviations used to define the fields in the assembly listing of the drive interface control program.

JMPT Jump to (JA) if the tested condition is true (set/one).

JMPF Jump to (JA) if the tested condition is false (clear/zero).

UJP Unconditional jump to (JA)

JA Jump address; read-only memory address referenced by jump macro

JC Jump condition selected by MIR18 through MIR22 for true/false jump test

SEQ Sequential; execute the next micro instruction in sequence (no jump) following this one.

NOP No operation is performed by this micro instruction.

M Mode field (MIR16 and MIR17) defines the instruction mode.

SET Set mode; the control flip-flop defined by the flip-flop field is set.

CLR Clear mode; the control flip-flop defined by the flip-flop field is cleared.

FF Flip-flop set or clear mode of operation

S Source field; specifies the source of data for operation

D Destination field; specifies the destination for data of this operation

SD Source (S) to destination (D) data transfer operation

DD Direct data; the data field that is part of this micro instruction, used as source data for this operation

F The control flip-flop or enable line that is set or cleared in a flip-flop operation

MACRO FIELDS

Each of the macros or abbreviations listed are directly related to a specific field of the micro instruction. Within the assembly listing of the drive interface micro program, each macro is assigned to a field of the macro statement, based on the format of that micro instruction. This field arrangement is illustrated in table 5-3. Note that the same macro can appear in different fields of the macro-stated micro instruction.

The mode (M), source (S), destination (D), jump condition (JC), and control flip-flop fields of the macro operation contain abbreviated mnemonics for the program listing. These mnemonics are listed in tables 4-3 through 4-7, along with their related codes.

PROGRAM LISTING

The drive interface micro program is initially assembled from the macro operators and statements previously described. The program, once assembled, provides a listing of assembler macros and micro-instruction codes. Note that the micro-instruction code (24 bits) is assembled into two 16-bit computer words in the program listing. The most significant eight bits of the first word are the address of the micro instruction with read-only memory. Figure 5-2 illustrates one micro instruction assembled in the format of the program listing.

TABLE 5-3. DRIVE INTERFACE MACRO-FIELD FORMAT

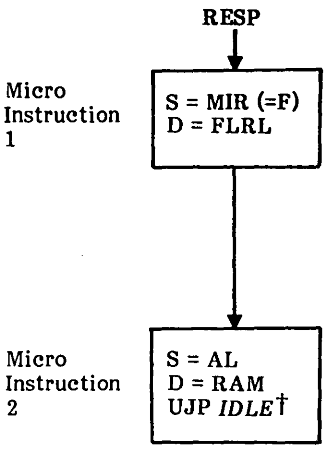
Field 1	Field 2	Field 3	Field 4	Field 5
SEQSD	S	D	DD	
SEQFF	M	F		
UJP	JA			
UJPSD	S	D	JA	
UJPFF	M	F	JA	
JMPF	JC	JA		
JMPFSD	JC	S	D	JA
JMPFFF	JC	M	F	JA
JMPT	JC	JA		
JMPTSD	JC	S	D	JA
JMPTFF	JC	M	F	JA
NOPP				

The assembled drive interface micro-program listing may be found in appendix C.

FLOW CHARTS

The flow charts that illustrate the controlware's program logic are contained in appendix B. Each single or double contacting symbol shown contains an abbreviated form of text defining the operation performed by a single micro instruction. Entrance to a routine is from the top of the symbol, with the routine identified by its assembler label. Decision or question blocks requiring a yes and no path are denoted by diamond symbols. Exits from a flow chart are shown as an arrowhead pointing to the assembler label or by an unconditional jump (UJP) to the name of the routine.

Using the example of transferring the contents of the lower half (bits 0 through 7) of the CPU's A register to random-access memory file register F requires two micro instructions and, therefore, two flow chart steps. The routine is entered by a jump to program label RESP.



This operation accesses file address F by first loading the random-access memory file address register (FLRL) with an F. This constant of F was obtained from the direct data field of an immediate format micro instruction.

This second micro instruction selects A-register lower as the source of data onto the internal bus. The random-access memory, which previously had its file F addressed, is the destination. The final operation performed by this instruction is an exit jump to the micro routine designated by the label IDLE.

TRANSFORM JUMP TABLE

The input/output function code, contained in Q register bits 0 through 3 (Qf) during an A/Q command operation, is ORed with read-only memory address 1E0. This address, designated by the program label HIGH and combined with the Qf, provides the transform logic with a single read-only memory address in the range of 1E0 through 1FF hexadecimal. Accessing this address results in the execution of an unconditional jump micro instruction at that address. The read-only memory address that is jumped to is shown in the jump table as a micro-routine program label. This label and its associated micro routine may be referenced on the indicated flow chart page and in the program listing in appendix C.

† Instructions in italics refer the reader to the corresponding program label in italics.

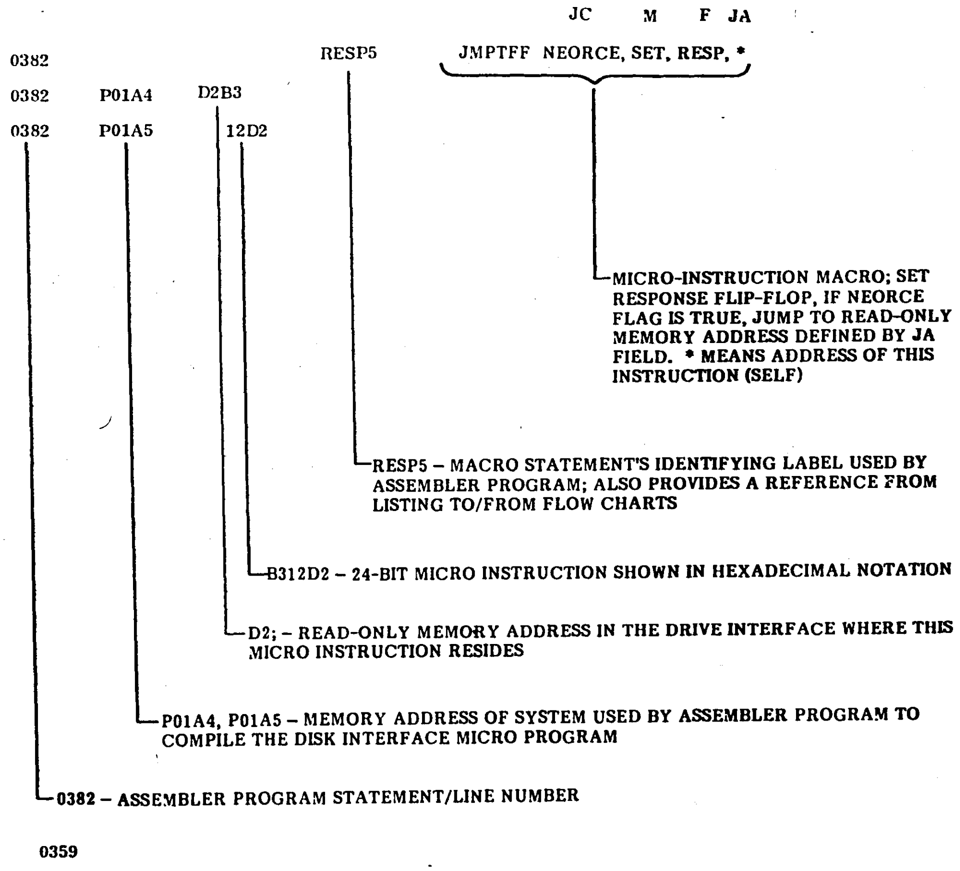

JC M F JA
0382 RESP5 JMPTFF NEORCE, SET, RESP, *
0382 P01A4 D2B3
0382 P01A5 12D2
MICRO-INSTRUCTION MACRO; SET RESPONSE FLIP-FLOP, IF NEORCE FLAG IS TRUE, JUMP TO READ-ONLY MEMORY ADDRESS DEFINED BY JA FIELD. * MEANS ADDRESS OF THIS INSTRUCTION (SELF)
RESP5 – MACRO STATEMENT'S IDENTIFYING LABEL USED BY ASSEMBLER PROGRAM; ALSO PROVIDES A REFERENCE FROM LISTING TO/FROM FLOW CHARTS
B312D2 – 24-BIT MICRO INSTRUCTION SHOWN IN HEXADECIMAL NOTATION
D2; – READ-ONLY MEMORY ADDRESS IN THE DRIVE INTERFACE WHERE THIS MICRO INSTRUCTION RESIDES
P01A4, P01A5 – MEMORY ADDRESS OF SYSTEM USED BY ASSEMBLER PROGRAM TO COMPILE THE DISK INTERFACE MICRO PROGRAM
0382 – ASSEMBLER PROGRAM STATEMENT/LINE NUMBER
0359

Figure 5-2. Micro-Instruction Example

MAINTENANCE 6

PREVENTIVE MAINTENANCE

This equipment consists of a printed circuit board that plugs into the central processing unit chassis. All preventive maintenance for the Storage Module Drive Adapter is covered by the preventive maintenance procedure in the CYBER 18-20/30 Timeshare Computer Systems Hardware Maintenance Manual. This consists of cleaning dirt and dust from the boards while performing preventive maintenance on the central processing unit. No additional preventive maintenance is required.

CALIBRATION AND ALIGNMENT

No field calibration and alignment procedures are required.

TROUBLESHOOTING

Troubleshoot to the board level using the following diagnostics:

MSMP17	SMD094
ODS	SMDDA SMDCU SMD01

GLOSSARY A

GLOSSARY OF TERMS USED IN FLOW CHARTS

ADCOM	Address compare
ADFIND	Address find
ADREP	Address reply
ADRF	Address field
AL	A register, lower
ALDON	All done
ALENB	Autoload enable
A/Q REJ	A/Q reject flip-flop
AU	A register, upper
AUTOL	Autoload flip-flop
AUTSEL	Autoload select
AQEXEC	A/Q execute flip-flop
AQREP	A/Q reply flip-flop
BL	Buffer length
BLEQZERO	Buffer length equals zero
BLR	Buffer length register
BOR	Bus-out register
BREMPTY	Output buffer register empty
BUSY	Busy flip-flop
CE	Check end
CHKCU	Check controller/formatter
CHKHD	Check head
CLR	Clear
CLR2	Clear direct memory access logic 2
COMAD	Common address
COMPL	Complete
CUOP	Controller/formatter operation
CUSEL	Selected flip-flop
CWA	Controller/formatter current word address
DECHO	Drive echo
DMA	Direct memory access
DMA CLR	Clear direct memory access logic
ECC	Error correction code
EOC	End of cylinder
ERR	Error
FLR	Field-length register
FLTCL	Fault clear
FULWRT	Full write cycle flip-flop
IDLE	Idle loop entry point
ILL	Illegal
INC	Increment
IOER	Input/output error
JMPF	Jump false
JMPT	Jump true
L	Lower
LPFLG	Loop flag
MIR	Micro-instruction register
NE	Normal end
NEORCE	Normal end or check end
NOP	No operation
NOTTM	Not test mode

NTM	Not test mode
PP	Program protect
RAM	Random-access memory
REP	Reply
REPT	Repeat
RESP	Response
ROWCOM	Read or write common entry point
R1	R1 register
R1EQRAM	R1 register equals random-access memory
R1MSB	R1 register most significant bit
R1 to A	R1 register to the A-register flip-flop
RRET	Read return
SEL	Select
SFWAU	Set first word address upper
TAGOUT	Tag out flip-flop
TAGVALID	Tag valid flip-flop
TBR	Tag bus register
TCOMP	Transfer complete
TM	Test mode
TMW	Test mode write
TXCPL	Transfer complete
U	Upper
UJP	Unconditional jump
XCOMP	Transfer complete
WRET	Write return

This appendix includes table B-1, the transform jump table, and figure B-1, the controlware flow charts. It is intended to be used in conjunction with the drive interface logics and appendix C, the controlware listing. Appendix C can be used to determine the actual read-only memory cell location of a particular flow block.

PROGRAM LABELS

A/Q FUNCTION/STATUS ENTRY POINTS

Program Labels	Sheet No.	Description
BUFFL	6	Load buffer length register
CSS	5	Current sector status
CUSTA	5	Control unit status
COND	5	Error correction code condition status
CYLST	6	Cylinder address status
DECHO	7	Drive echo
DFS	5	Drive fault status
DRST1	5	Drive status one
DRST2	5	Drive status two
FORMAT	13	Format write
PATT	5	Error correction code pattern status
POLL	11	Initiate poll
PHYS	5	Physical unit status
RCONT	17	Read recovery control
RTZS	17	Return to zero seek
SEC	16	Set section and head address
SECST	6	Sector and head status
SFWAL	6	Set first word address lower
SFWAU	45	Set first word address upper
SEL	7	Select control unit or drive
SEEK	8	Load cylinder address
SPEC	45	Special for diagnostic testing
TMW	25	Test mode write data

Program Label	Sheet No.	Description
TMR	27	Test mode read data
WRITE	19	Write data or address field
REED	26	Read data or address field
WAIT	2	Director function or status
CUOP	3	Drive request
TFORM	1	Q transform
HIGH	0	Jump table instructions

MISCELLANEOUS ENTRY POINTS

Program Label	Sheet No.	Description
CLR2	1	Idle loop entry from read or write
IDLE	1	Idle loop
ILL	2	A/Q reject
REP	2	A/Q reply
CHKCU	3	Check for control unit select
RESP1	4	Response to control unit
OUT	8	Output to the control unit
OUTLP	14	Format write timeout
INLP	14	Format write timeout
FWEND	15	Format write timeout complete
BRCHK	19	Check output buffer register for empty
CHKBR	19	Check output buffer register for empty
NOTTM	20	Not test mode data transfer
NTM	19	Not test mode data transfer
ADFIND	20	Search for requested address field
ADCOM	20	Compare current address field to requested
REPT	20	Read next address field

Program Label	Sheet No.	Description
NOCOM	24	Current address field does not compare with requested
ALDON	25	Test mode write or data transfer is complete
TCOMP	25	Test mode read transfer is complete
ROWCOM	28	Read or write data common entry point
WRET	28	Entry to write data
RRET	28	Entry to read data
DONE	29	Transfer is complete
GO ON	30	More data to transfer
CHKHD	31	Check current head address for equal to 4
ADR	34	Read or write address field
ADREP	34	Read or write more address field
COMAD	35	Common address field output to control unit
MORE	36	Read or write more address field
DELY	37	Read dely timeout
AUTO	38	Autoload entry point
AUTSEL	38	Autoload control unit select
DRSEL	38	Autoload drive select
FLTCL	40	Autoload clear faults
AGAIN	41	Autoload read drive status 2 again
LDRAM	45	Write into random-access memory

ERROR ENTRY POINTS

Program Labels	Sheet No.	Description
RESP2 through RESP8		Response to control unit
CE1 through CE3		Check end error
ERR1 through ERR4 and ERR6 through ERR10		Miscellaneous errors
IOER1 through IOER7		Input/output errors
ERRX	30	Read or write error
COMPL	30	Transfer complete with error
DELX	37	Transfer complete delay with error

CROSS REFERENCE

The following is a list of instructions found in the flow charts with the sheet numbers where they can be found. These instructions are highlighted in italics in the flow charts.

ADCOM	20, 30
ADFIND	20, 26, 33, 44
ADR	20, 26, 33, 44
ADREP	34, 36
AGAIN	41, 42
ALDON	25, 29
AUTO	1, 38
AUTSEL	38
BUFFL	location 0, 6
CE1	3, 4
CE2	9, 10, 12, 14, 39, 40, 41, 43
CE3	32, 33
CHKCU	3, 5, 16, 17
CHKHD	30, 31, 36
CLR2	1, 22, 29
COMAD	34, 35
COMPL	30, 37
COND	location 0, 5
CSS	location 0, 5
CUOP	location 0, 3, 7
CUSTA	location 0, 5
CYLST	location 0, 6
DECHO	location 0, 7
DELX	30, 37
DELY	29, 36, 37
DFS	location 0, 5
DONE	29, 36, 37
DRSEL	38, 41
DRST1	location 0, 5
DRST2	location 0, 5
ERR1	3
ERR2	9, 10
ERR3	12
ERR4	14, 15
ERR6	21, 22, 28, 35
ERR7	22, 30, 33, 37
ERR8	24, 25, 32
ERR9	24, 27
ERR10	15
ERRX	29, 30, 36
FLCTL	40
FORMAT	location 0, 13

FWEND	14, 15
GO ON	29, 30
HIGH	1
IDLE	1, 2, 9, 10, 12, 14, 15, 25, 31
ILL	2, 3, 7, 8, 11, 13, 17, 19, 26
INLP	14, 15
IOER1	4
IOER2	12
IOER3	15
IOER4	21, 22
IOER5	28, 29
IOER6	32, 33
IOER7	35
MORE	36
NOCOM	22, 23, 24
NOTTM	19, 20
NTM	19
OUT	8, 10, 18
OUTLP	14, 15
PATT	location 0, 5
PHYS	location 0, 5
POLL	location 0, 11
RCONT	location 0, 17
REED	location 0, 26
REP	location 0, 2, 4, 6, 7, 45
REPT	20, 24
RESP1	4
RESP2	9, 10, 39, 40, 41, 42, 43
RESP3	12
RESP4	15
RESP5	22
RESP6	29
RESP7	33
RESP8	35
ROWCOM	28
RRET	23, 28
SEC	location 0, 16
SECST	location 0, 6
SEEK	location 0, 8
SEL	location 0, 7
SETB	5
SET8	5
SFWAL	location 0, 6
SFWAU	location 0, 45
SPEC	location 0, 45
TCOMP	25, 27
TMR	26, 27
TMW	19, 25
WAIT	location 0, 2
WRET	23, 28

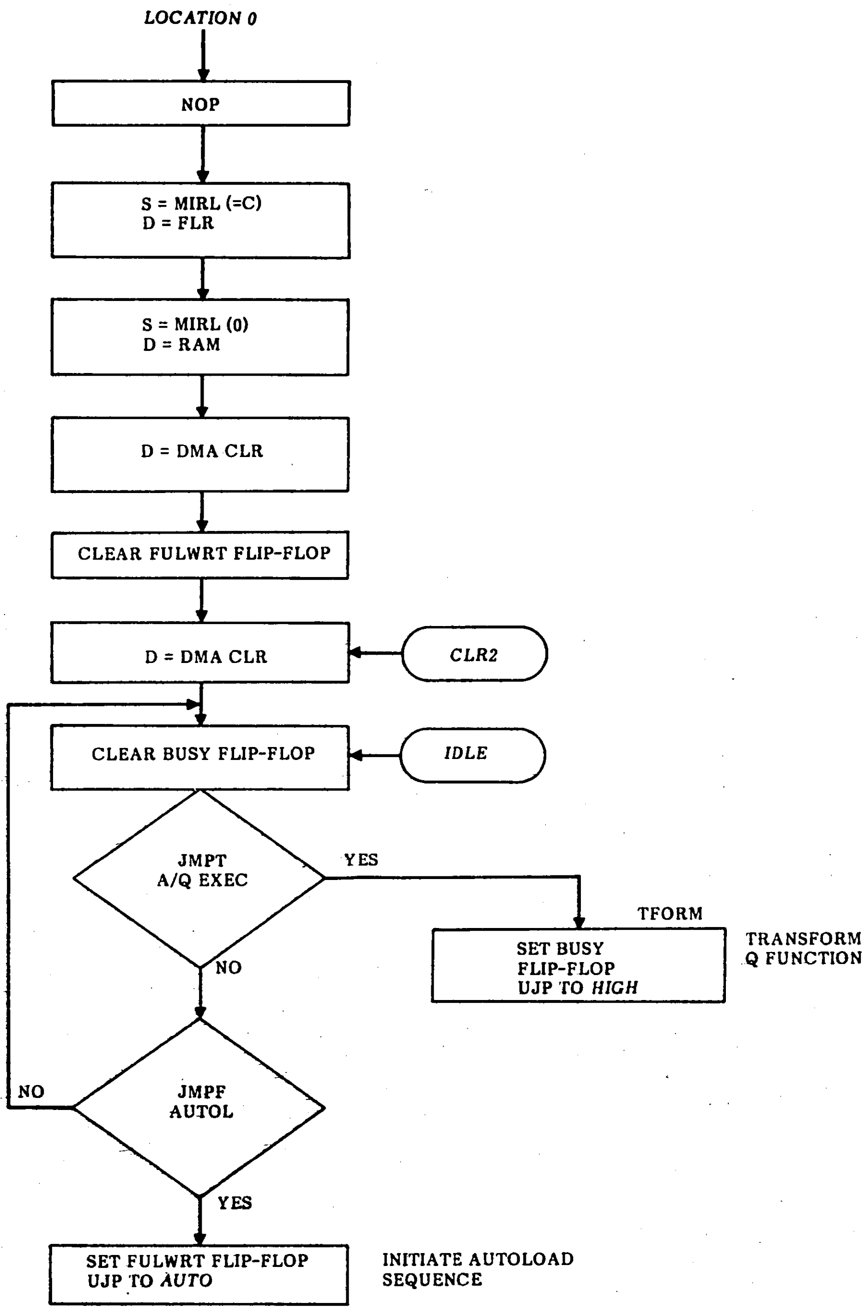
TABLE B-1. TRANSFORM JUMP TABLE

Read-Only Memory Address HIGH		Q00 through Q03 (Qf)	Instruction	Figure B-1 Sheet No.	I/O Function	
Q00 through Q03 ORed with 1E0	1E0	0	UJP SPEC	45	Read-random access memory	[†]
	1E1	1	UJP PHYS	5	Physical unit number	
	1E2	2	UJP REP, set R1 to A flip-flop	2	Poll status	
	1E3	3	UJP REP, set R1 to A flip-flop	2	Select acknowledge status	
	1E4	4	UJP REP, set R1 to A flip-flop	2	Drive echo input	
	1E5	5	UJP CYLST	6	Cylinder address status	
	1E6	6	UJP CSS	5	Currect sector status	
	1E7	7	UJP SECST	6	Sector and head status	
	1E8	8	UJP WAIT, clr BUSY flip-flop	2	Drive interface status	
	1E9	9	UJP DRST2	5	Drive status 2	
	1EA	A	UJP PATT	5	Error correction code pattern	
	1EB	B	UJP COND	5	Error correction code conditions	
	1EC	C	UJP DFS	5	Drive fault status	
	1ED	D	UJP CUSTA	5	Controller/formatter status	
	1EE	E	UJP DRST1	5	Drive status 1	
	1EF	F	UJP REP, set R1 to A flip-flop	2	Controller/formatter echo input	
Q00 through Q03 ORed with 1F0	1F0	0	UJP BUFFL, S=AU, D=BLU	6	Buffer length	[††]
	1F1	1	UJP CUOP, S=Q, D=TBR	3	Drive request	
	1F2	2	UJP POLL, S=Q, D=TBR	11	Initiate poll	
	1F3	3	UJP SEL, S=Q, D=TBR	7	Unit select	
	1F4	4	UJP DECHO, S=Q, D=TBR	7	Drive echo output	
	1F5	5	UJP SEEK	8	Load address	
	1F6	6	UJP FORMAT	13	Format write	
	1F7	7	UJP SEC, S=Q, D=TBR	16	Sector and head address	
	1F8	8	UJP WAIT	2	Director function	
	1F9	9	UJP REED, clr TXCPL flip-flop	26	Read	
	1FA	A	UJP WRITE, clr TXCPL flip-flop	19	Write	
	1FB	B	UJP CUOP, S=Q, D=TBR	3	Error correction code control	
	1FC	C	UJP RCONT	17	Read recovery (status control)	
	1FD	D	UJP SFWAL, S=AU, D=CWAU	6	Set first word address lower	
	1FE	E	UJP SFWAU, S=AL, D=CWAUU	45	Set first word address upper, random-access memory address	
	1FF	F	UJP CUOP, S=Q, D=TBR	3	Controller/formatter echo output	

[†] Input to A-register operation (read)

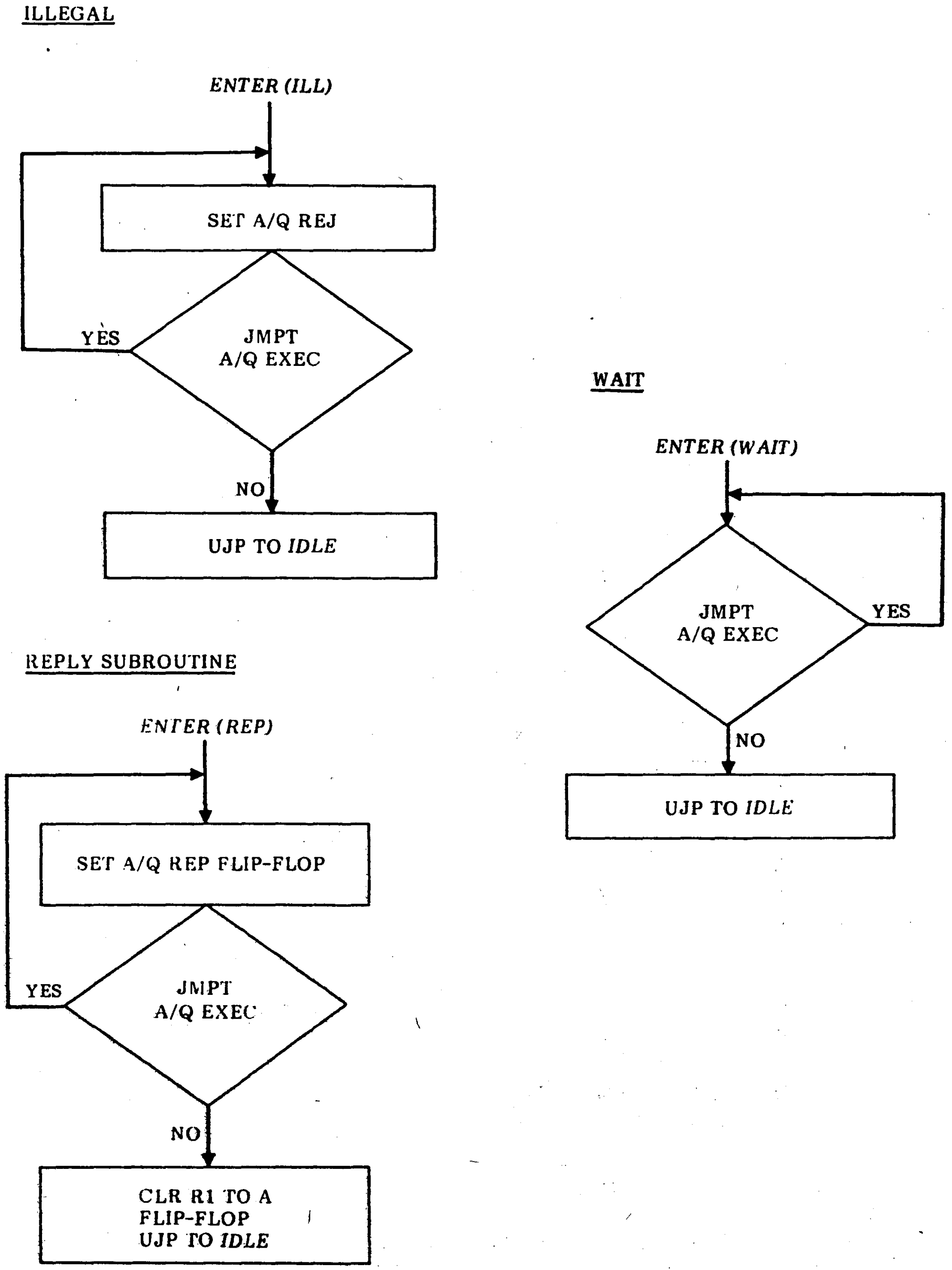
[††] Output from A-register operation (write)

IDLE LOOP



0485

Figure B-1. Storage Module Drive Flow Charts (Sheet 1 of 45)



ILLEGAL

REPLY SUBROUTINE

WAIT

0486

Figure B-1. Storage Module Drive Flow Charts (Sheet 2 of 45)

OUTPUT ONLY TO CONTROLLER/FORMATTER → W•1 W•B

DRIVE REQUEST
ERROR CORRECTION
CODE CONTROL

ENTRY CONDITIONS:

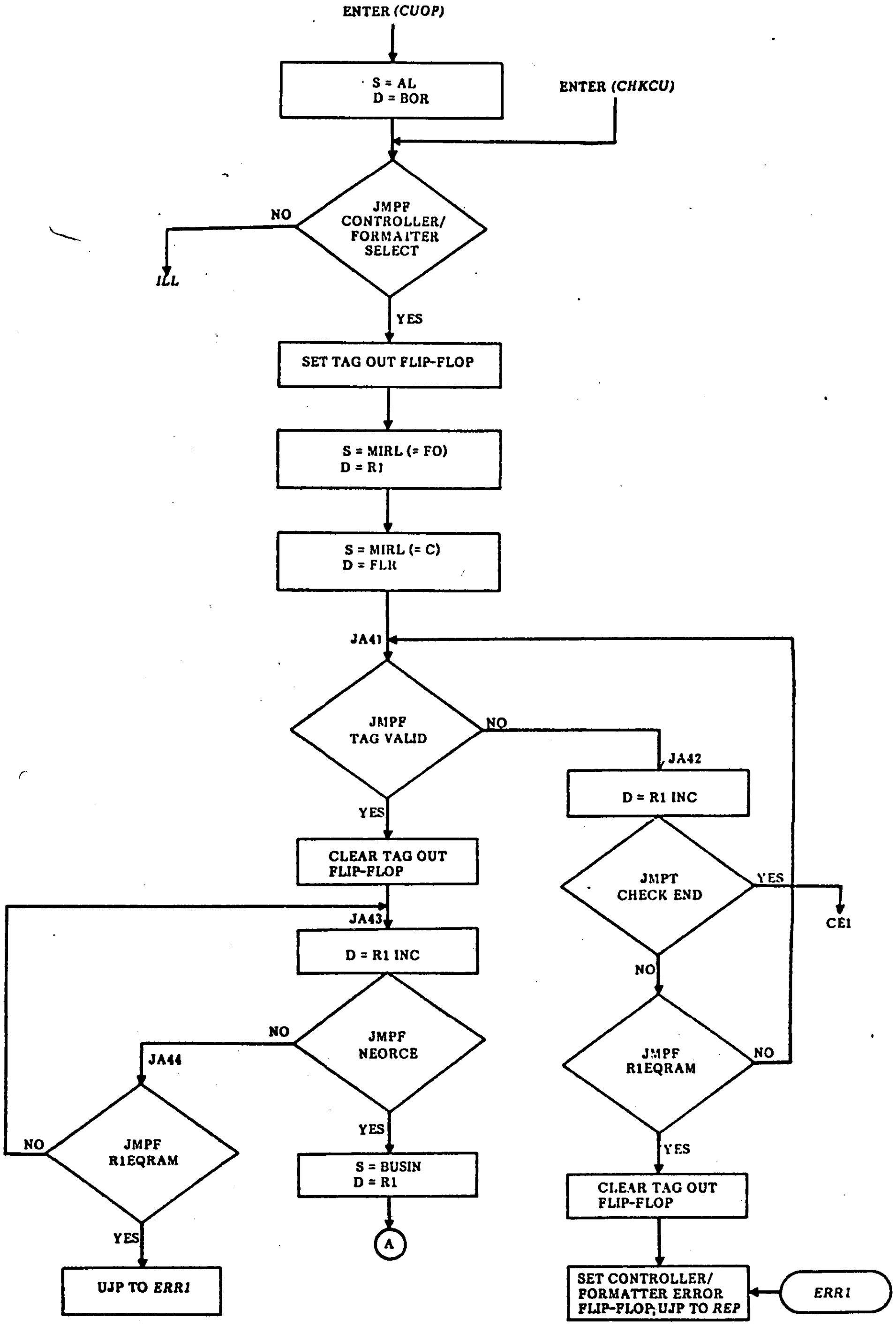
(CUOF): THE TAG BUS REGISTER IS SET TO THE APPROPRIATE VALUE.

(CHKCU): BOTH THE TAG BUS REGISTER AND BUS-OUT REGISTER ARE SET AS DESIRED.

NOTE: THIS ROUTINE IS ALSO USED FOR:

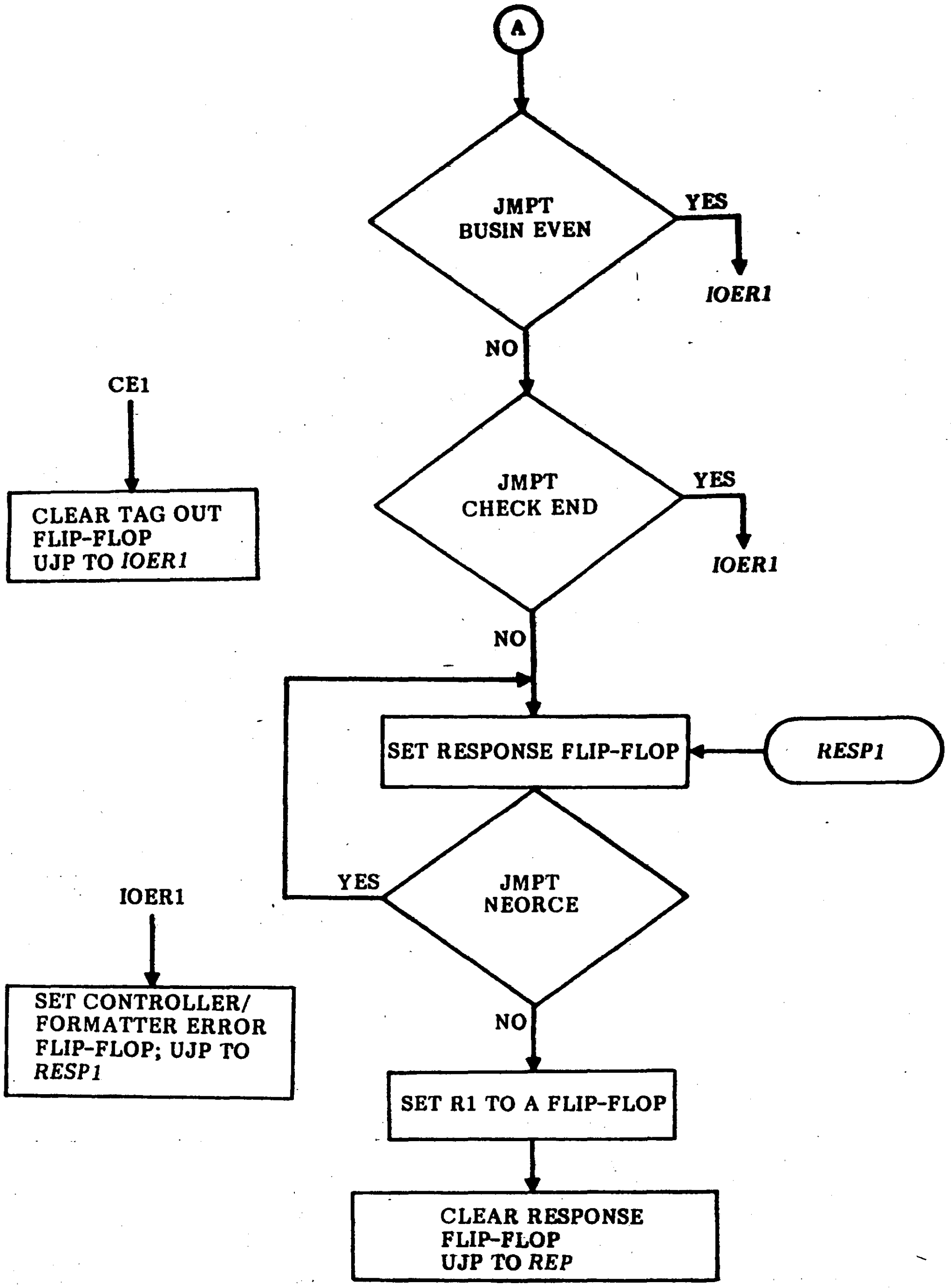
- CONTROLLER/FORMATTER INPUT/ OUTPUT IMMEDIATE
- CONTROLLER/FORMATTER INPUT/ OUTPUT EXTENDED

ENTRY POINT USED IS CHKCU.



0487

Figure B-1. Storage Module Drive Flow Charts (Sheet 3 of 45)



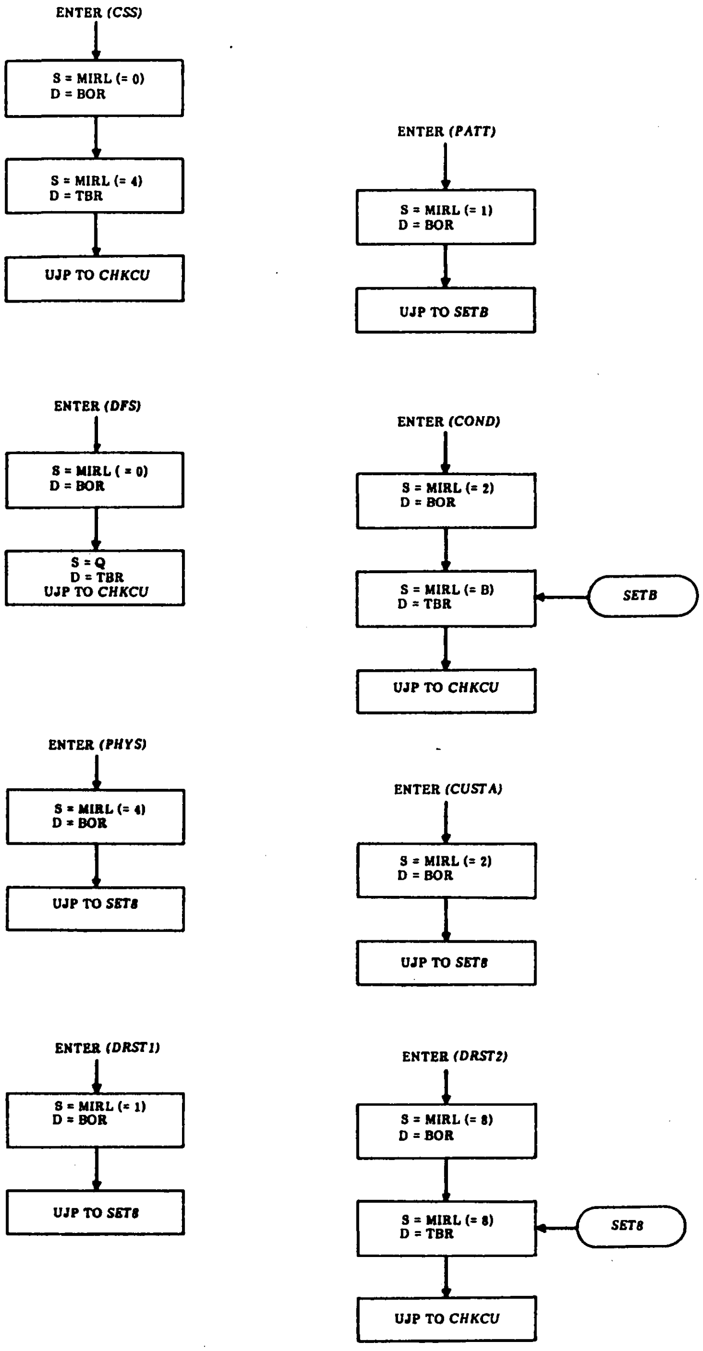
0488

Figure B-1. Storage Module Drive Flow Charts (Sheet 4 of 45)

CONTROLLER/FORMATTER INPUT/OUTPUT IMMEDIATE

ENTRY CONDITIONS: NONE

R • 1	PHYSICAL UNIT NUMBER
R • 6	CURRENT SECTOR STATUS
R • 9	DRIVE STATUS 2
R • A	ERROR CORRECTION CODE PATTERN
R • B	ERROR CORRECTION CODE CONDITIONS
R • C	DRIVE FAULT STATUS
R • D	CONTROLLER/FORMATTER STATUS
R • E	DRIVE STATUS 1



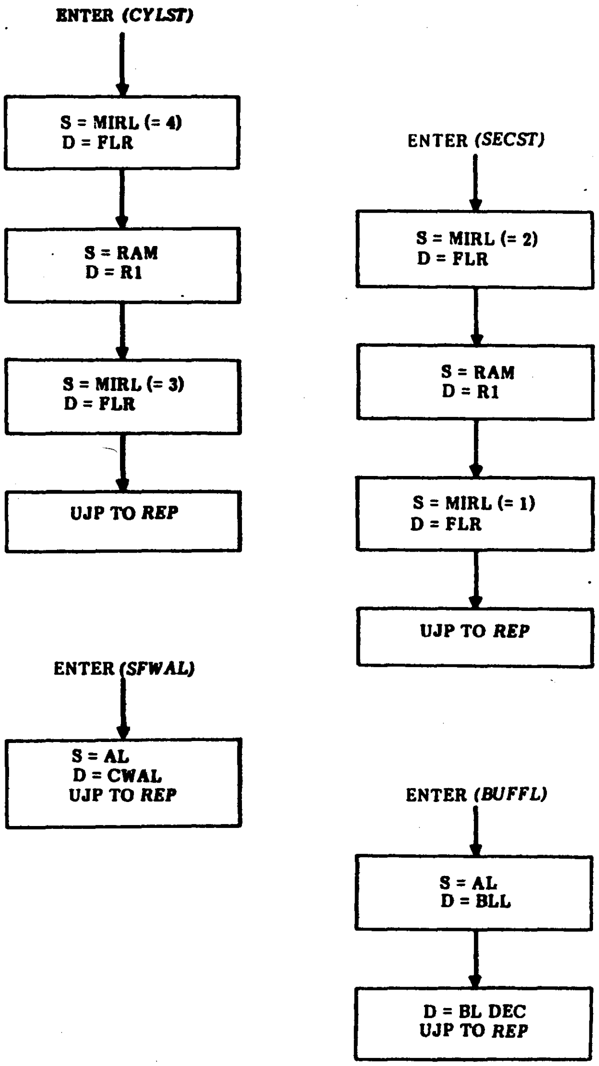
0489

Figure B-1. Storage Module Drive Flow Charts (Sheet 5 of 45)

DRIVE INTERFACE OPERATIONS ONLY

W · 0	BUFFER LENGTH
R · 2	POLL STATUS†
R · 3	SELECT ACKNOWLEDGE STATUS†
R · 4	DRIVE ECHO INPUT†
R · 5	CYLINDER ADDRESS STATUS
R · 7	SECTION/HEAD STATUS
W · D	SET FIRST WORD ADDRESS LOWER
R · F	CONTROLLER/FORMATTER ECHO INPUT†

†NO MICRO-CONTROL ENTRY POINT IS REQUIRED. CODE IS CONTAINED WITHIN THE JUMP TABLE INSTRUCTION.



0490

Figure B-1. Storage Module Drive Flow Charts (Sheet 6 of 45)

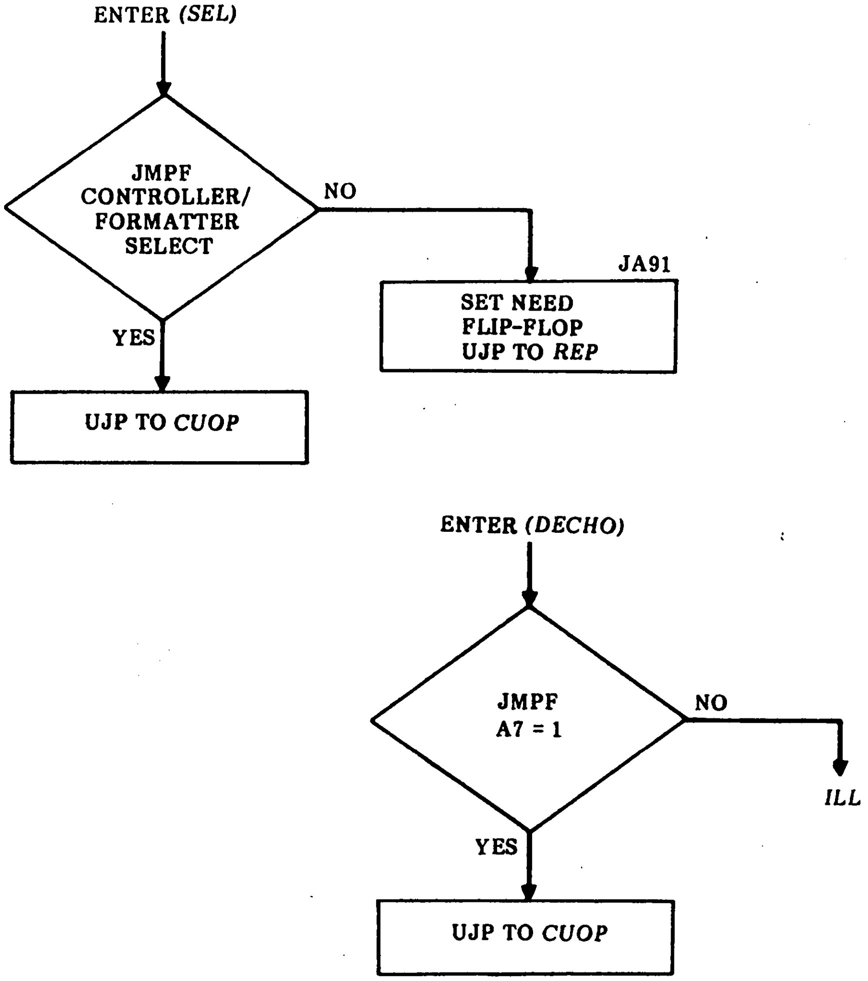
CONTROLLER/FORMATTER
INPUT/OUTPUT EXTENDED

ENTRY CONDITIONS:

THE TAG BUS REGISTER IS SET AS DESIRED.

W · 3	UNIT SELECT
W · 4	DRIVE ECHO OUTPUT
W · F	CONTROLLER/FORMATTER ECHO OUTPUT†

†ENTRY POINT IS CUOP.

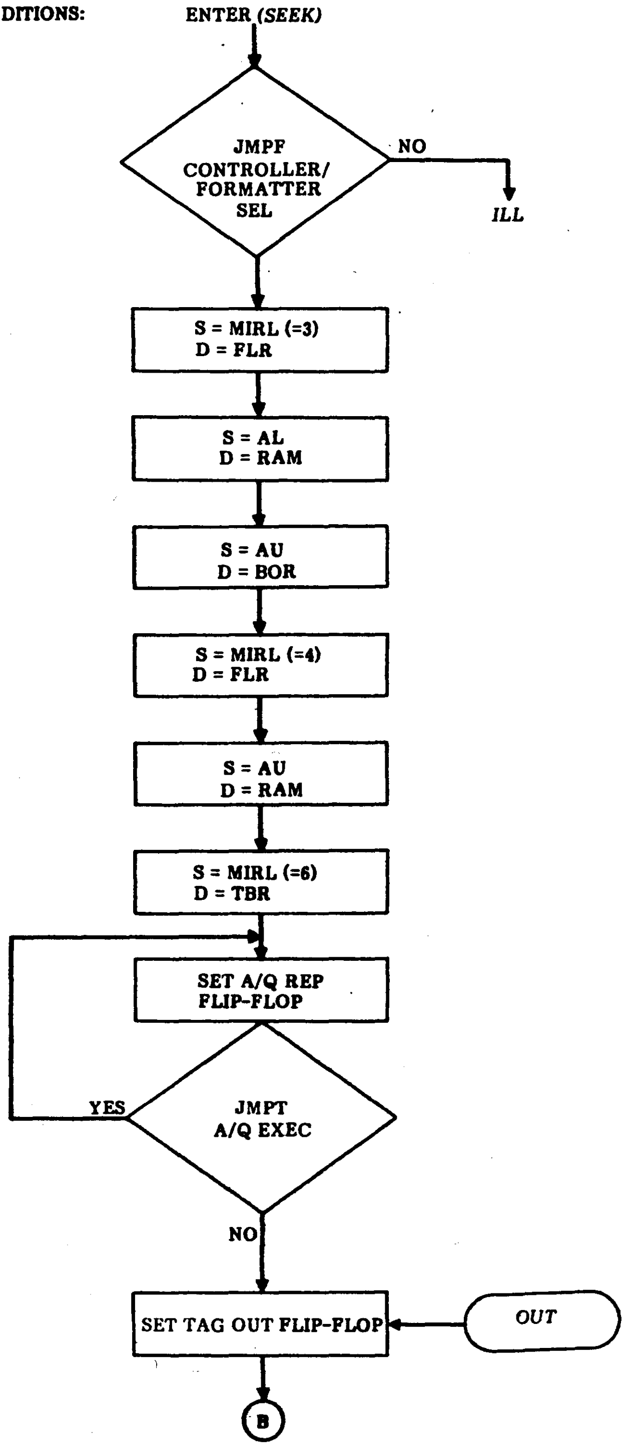


0491

Figure B-1. Storage Module Drive Flow Charts (Sheet 7 of 45)

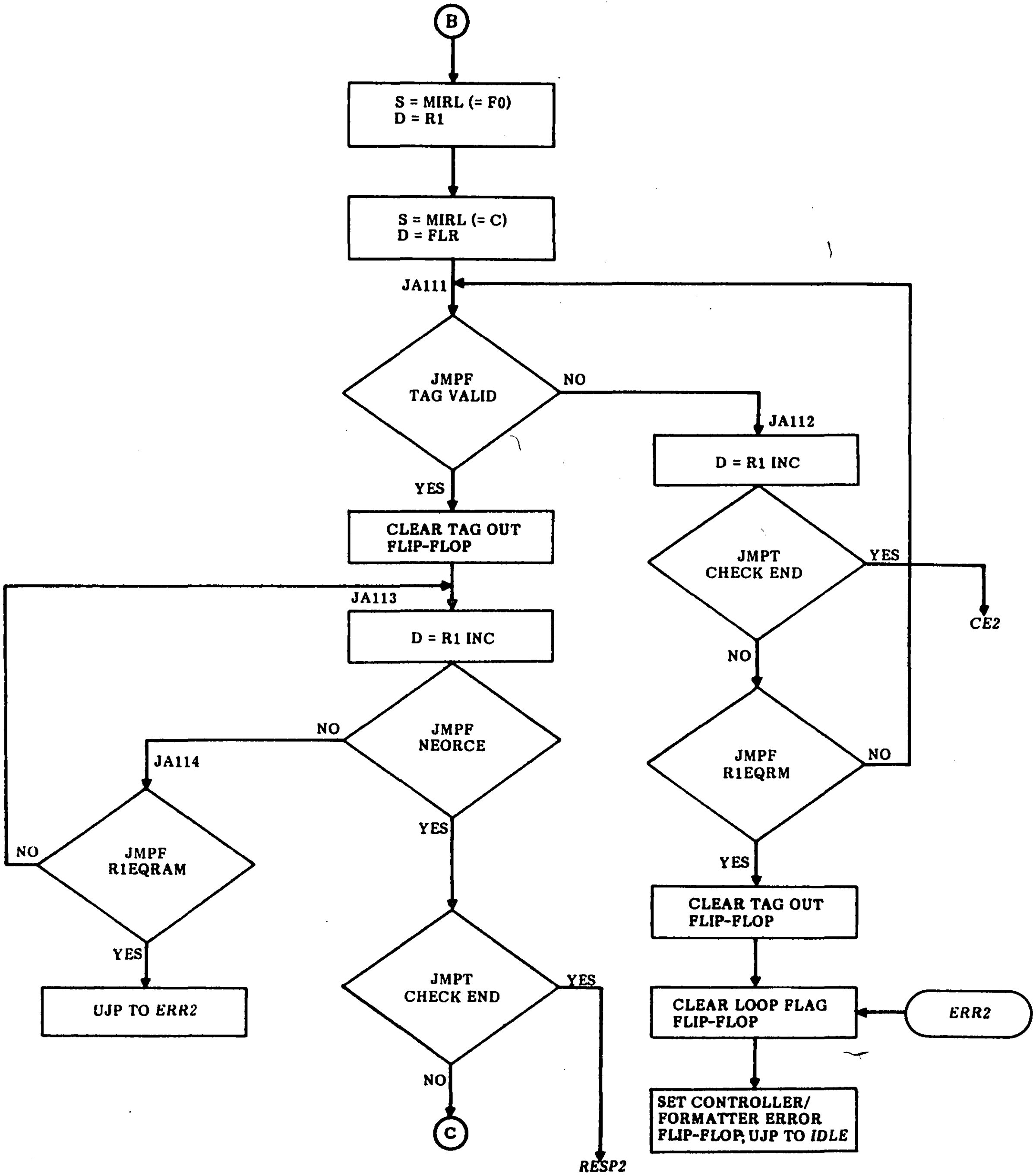
LOAD ADDRESS W • 5

ENTRY CONDITIONS:
NONE



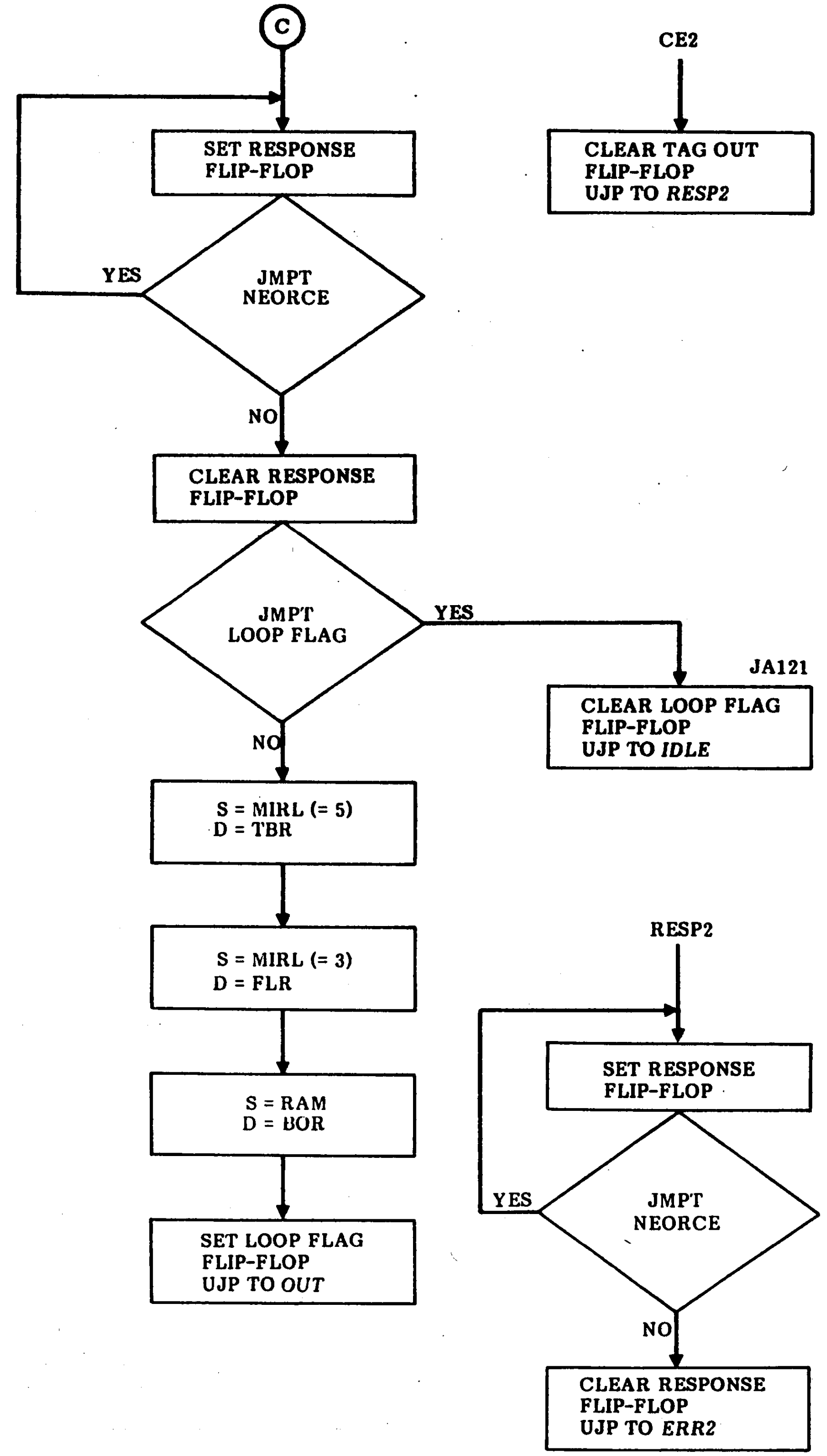
0492

Figure B-1. Storage Module Drive Flow Charts (Sheet 8 of 45)



0493

Figure B-1. Storage Module Drive Flow Charts (Sheet 9 of 45)

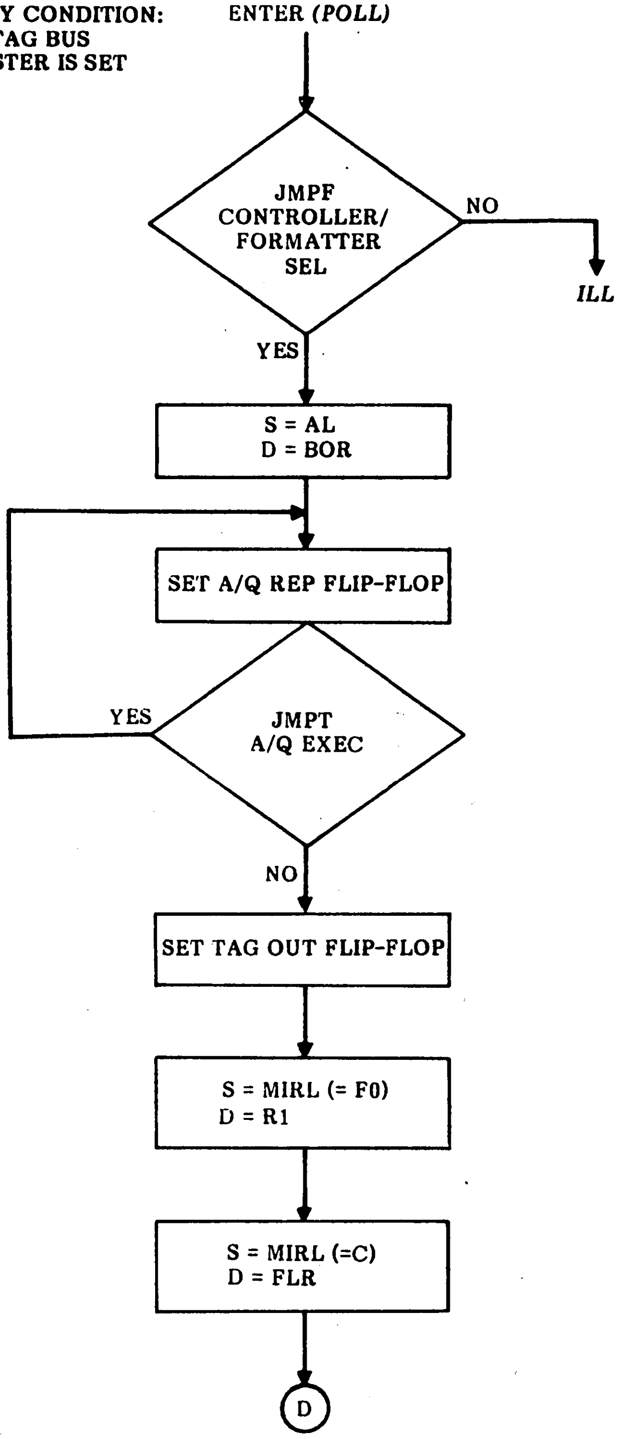


0494

Figure B-1. Storage Module Drive Flow Charts (Sheet 10 of 45)

INITIATE POLL W • 2

ENTRY CONDITION:
THE TAG BUS
REGISTER IS SET
TO 2.



0495

Figure B-1. Storage Module Drive Flow Charts (Sheet 11 of 45)

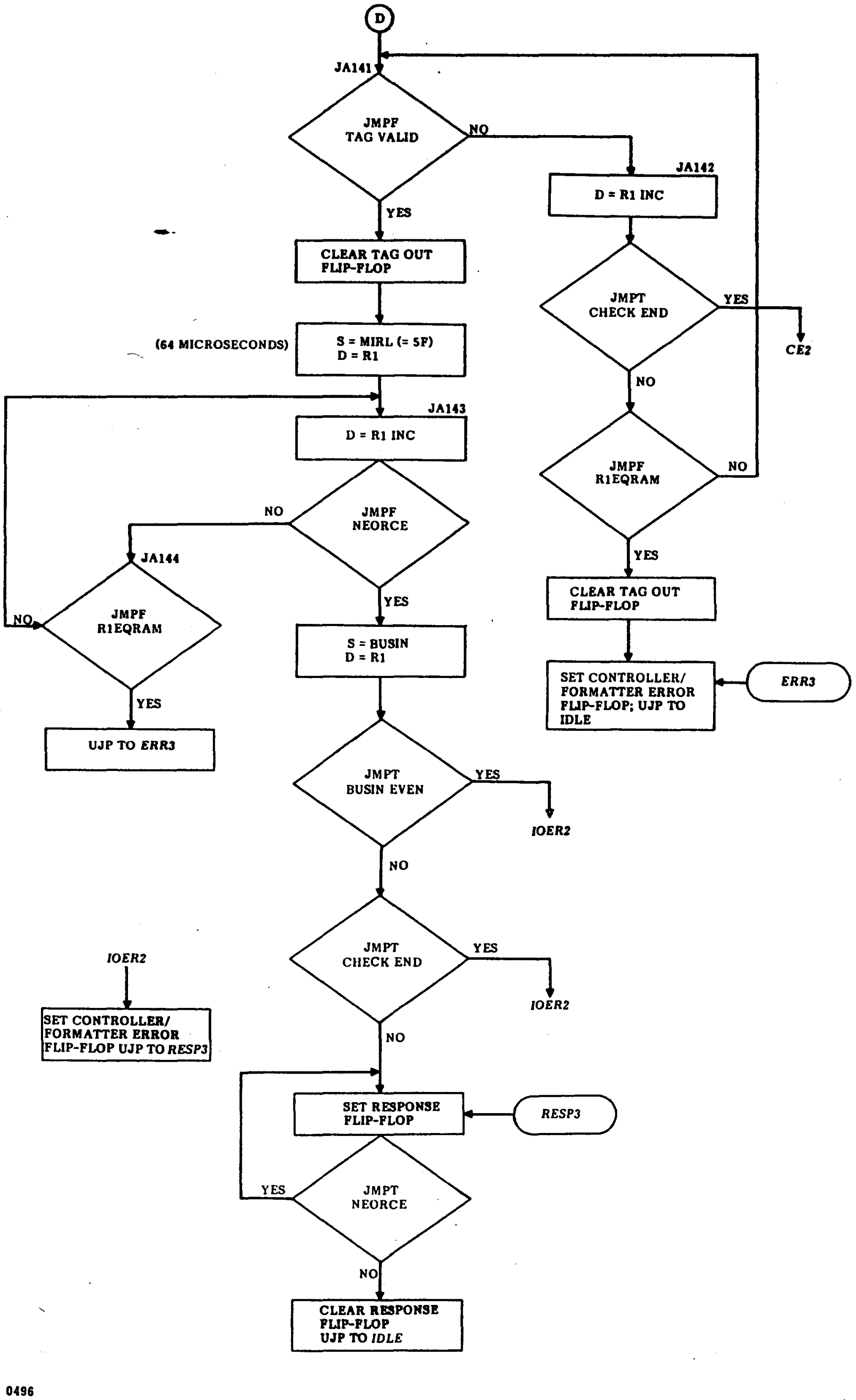
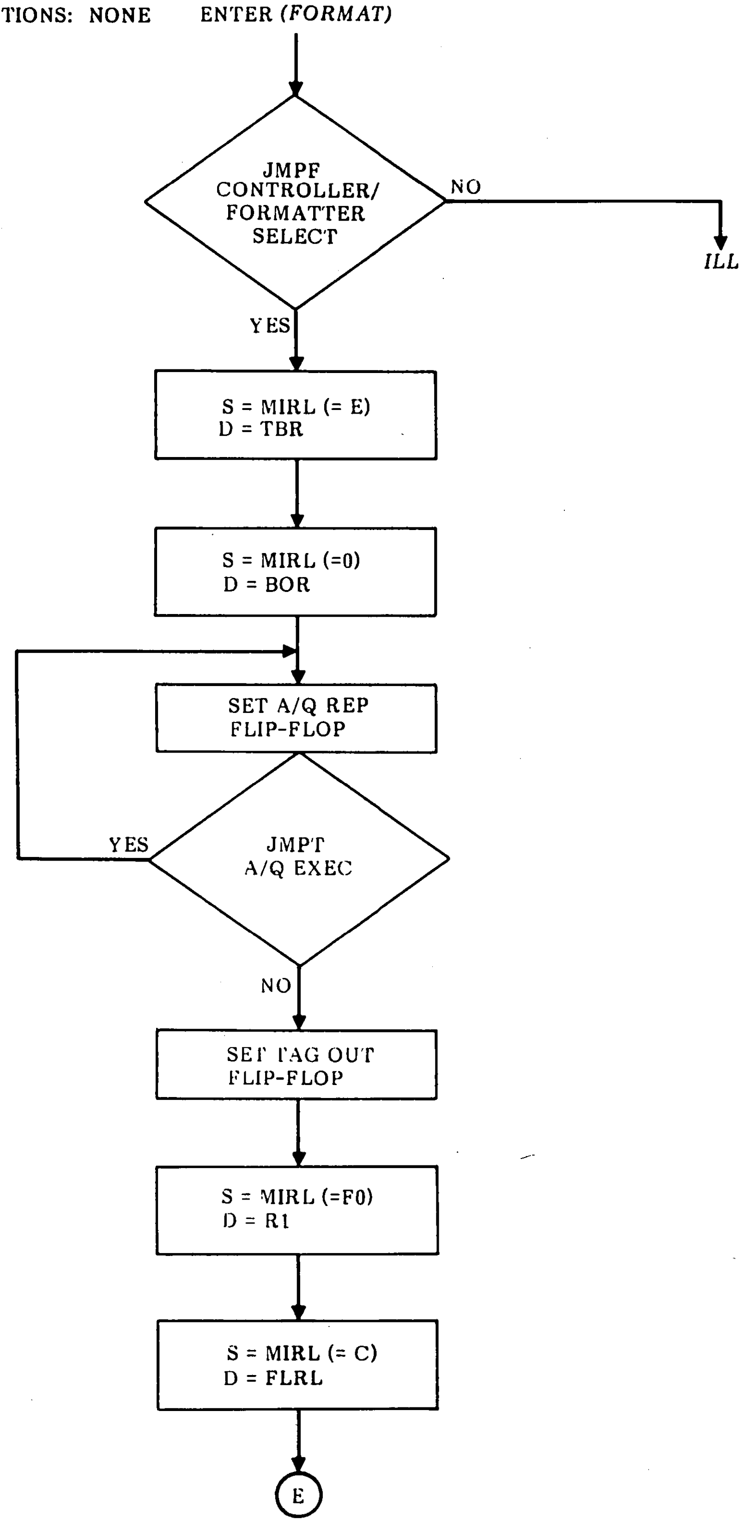


Figure B-1. Storage Module Drive Flow Charts (Sheet 12 of 45)

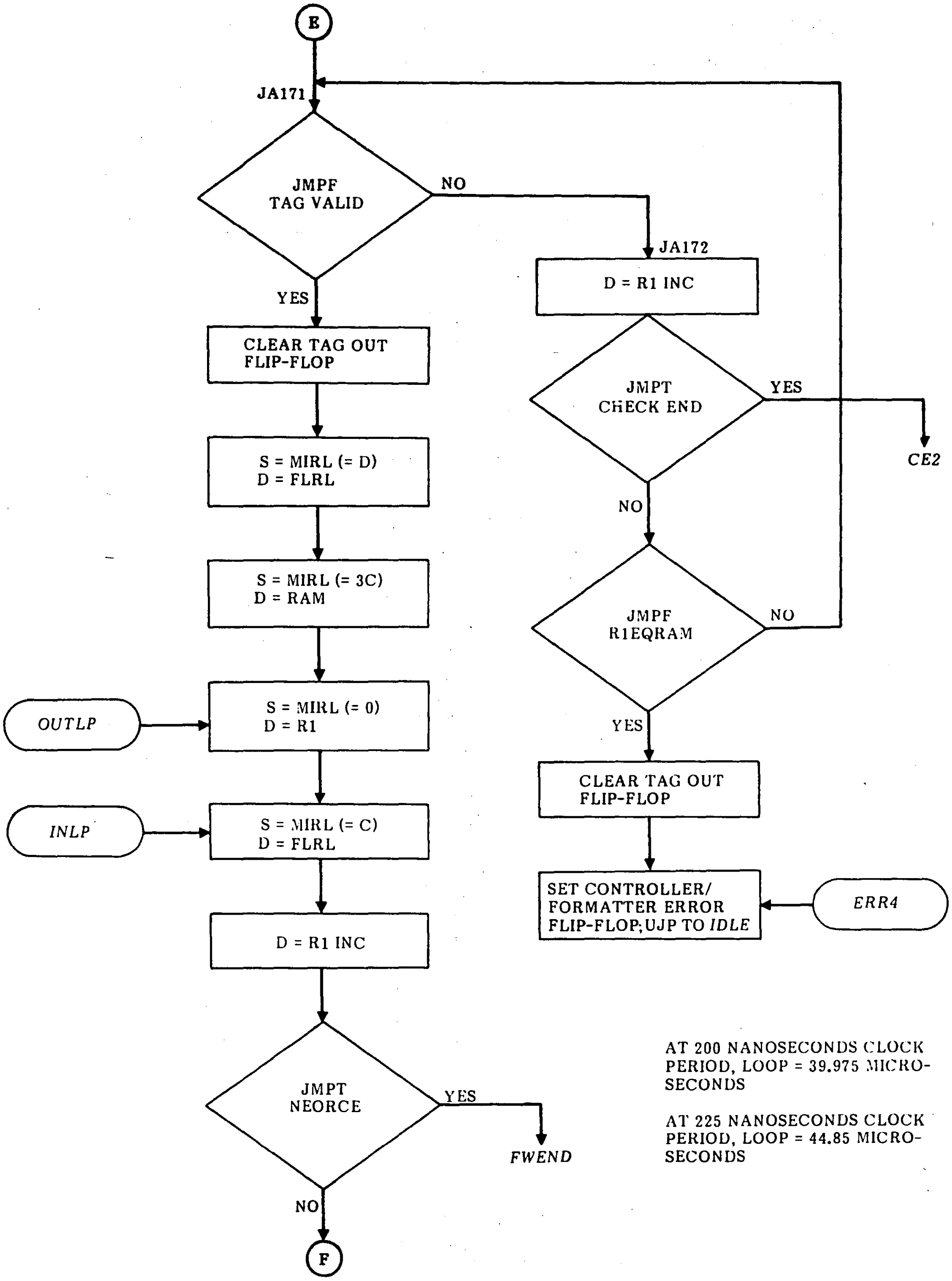
FORMAT WRITE W · 6

ENTRY CONDITIONS: NONE



0497

Figure B-1. Storage Module Drive Flow Charts (Sheet 13 of 45)

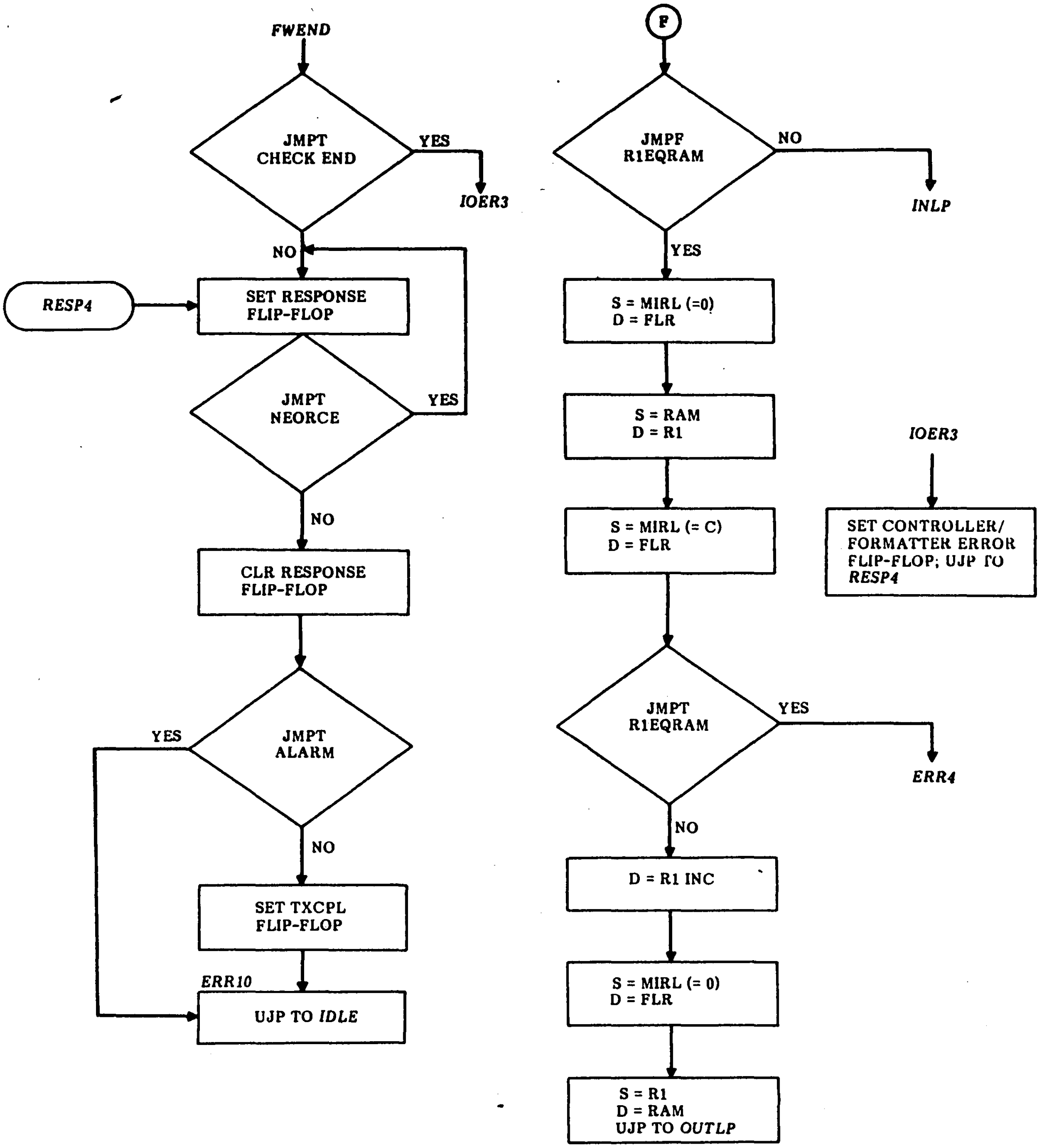


AT 200 NANOSECONDS CLOCK PERIOD, LOOP = 39.975 MICRO-SECONDS

AT 225 NANOSECONDS CLOCK PERIOD, LOOP = 44.85 MICRO-SECONDS

0498

Figure B-1. Storage Module Drive Flow Charts (Sheet 14 of 45)

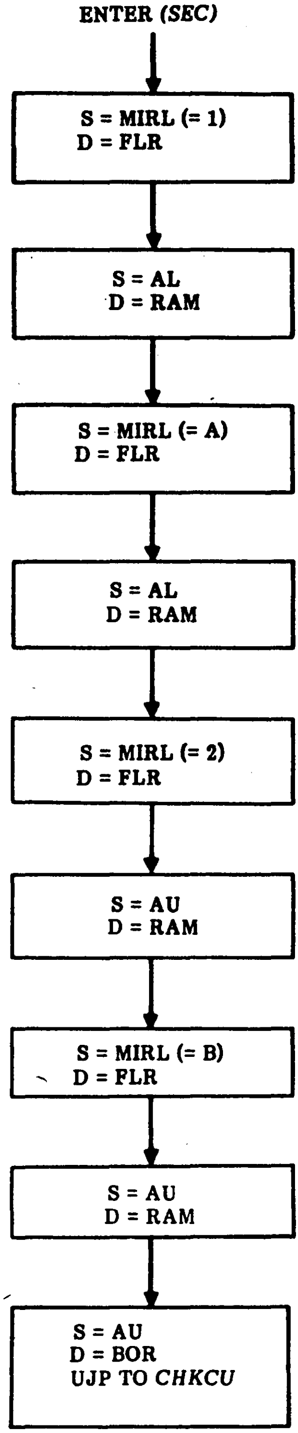


0499

Figure B-1. Storage Module Drive Flow Charts (Sheet 15 of 45)

SET SECTOR AND HEAD W · 7

ENTRY CONDITIONS: THE TAG BUS REGISTER IS SET TO 7.

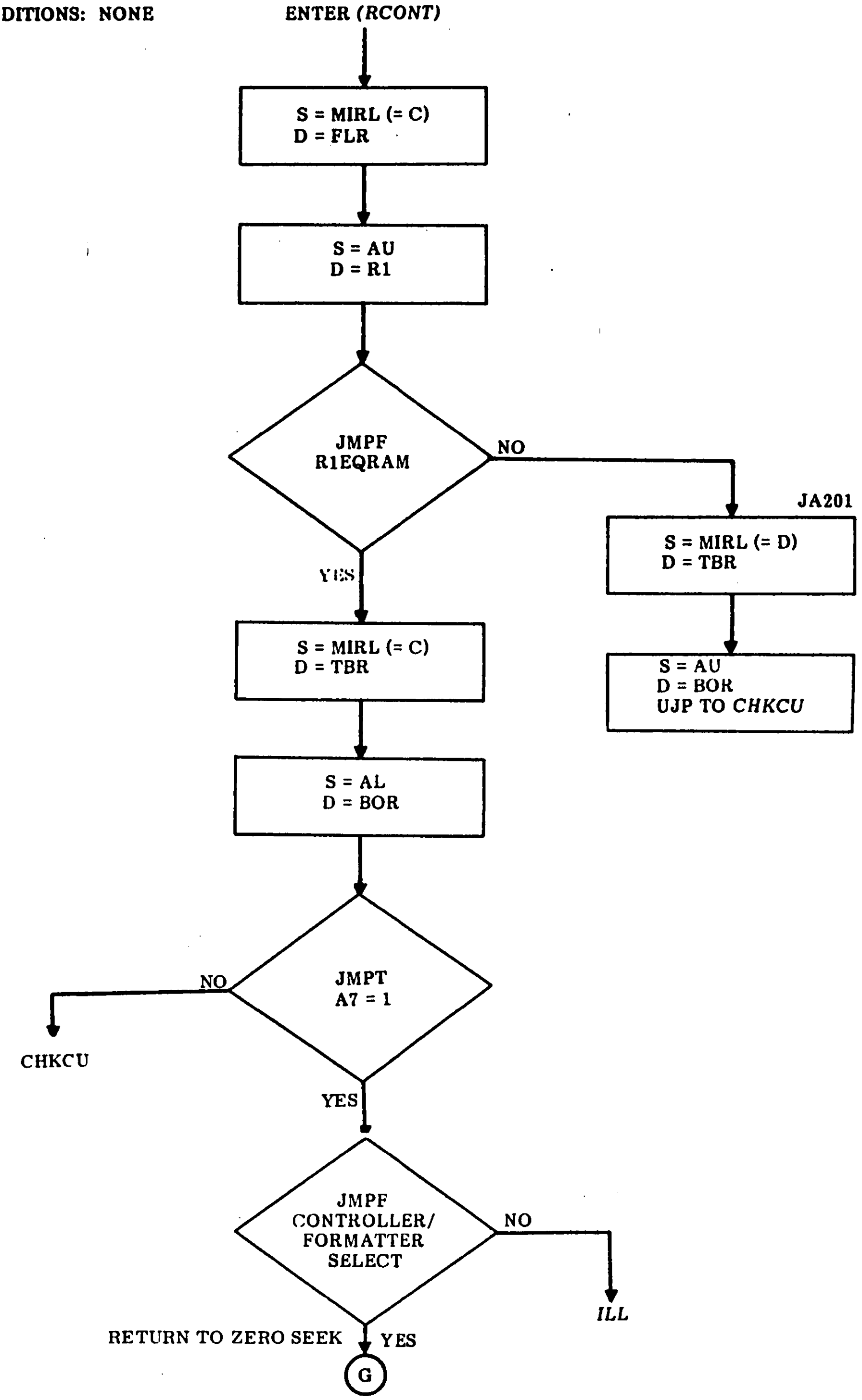


0500

Figure B-1. Storage Module Drive Flow Charts (Sheet 16 of 45)

READ RECOVERY CONTROL W • C

ENTRY CONDITIONS: NONE



0501

Figure B-1. Storage Module Drive Flow Charts (Sheet 17 of 45)

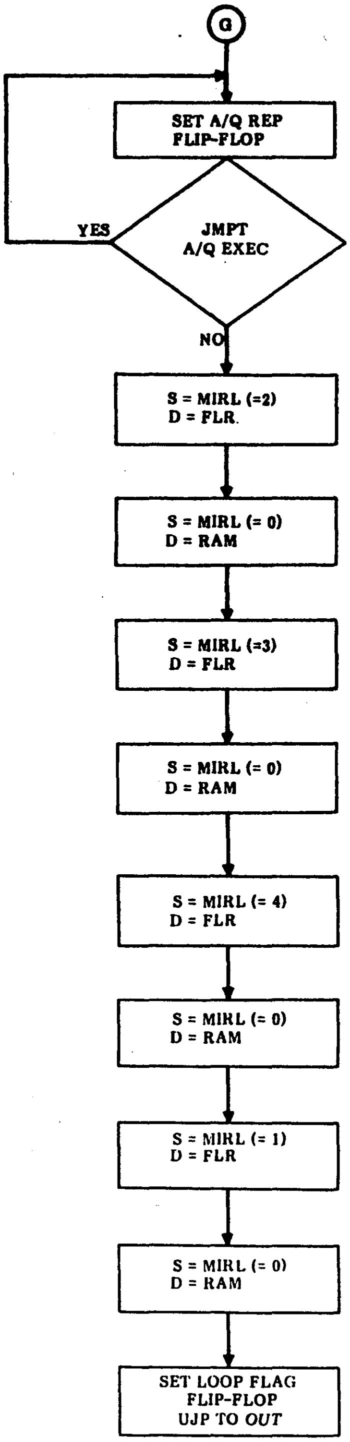
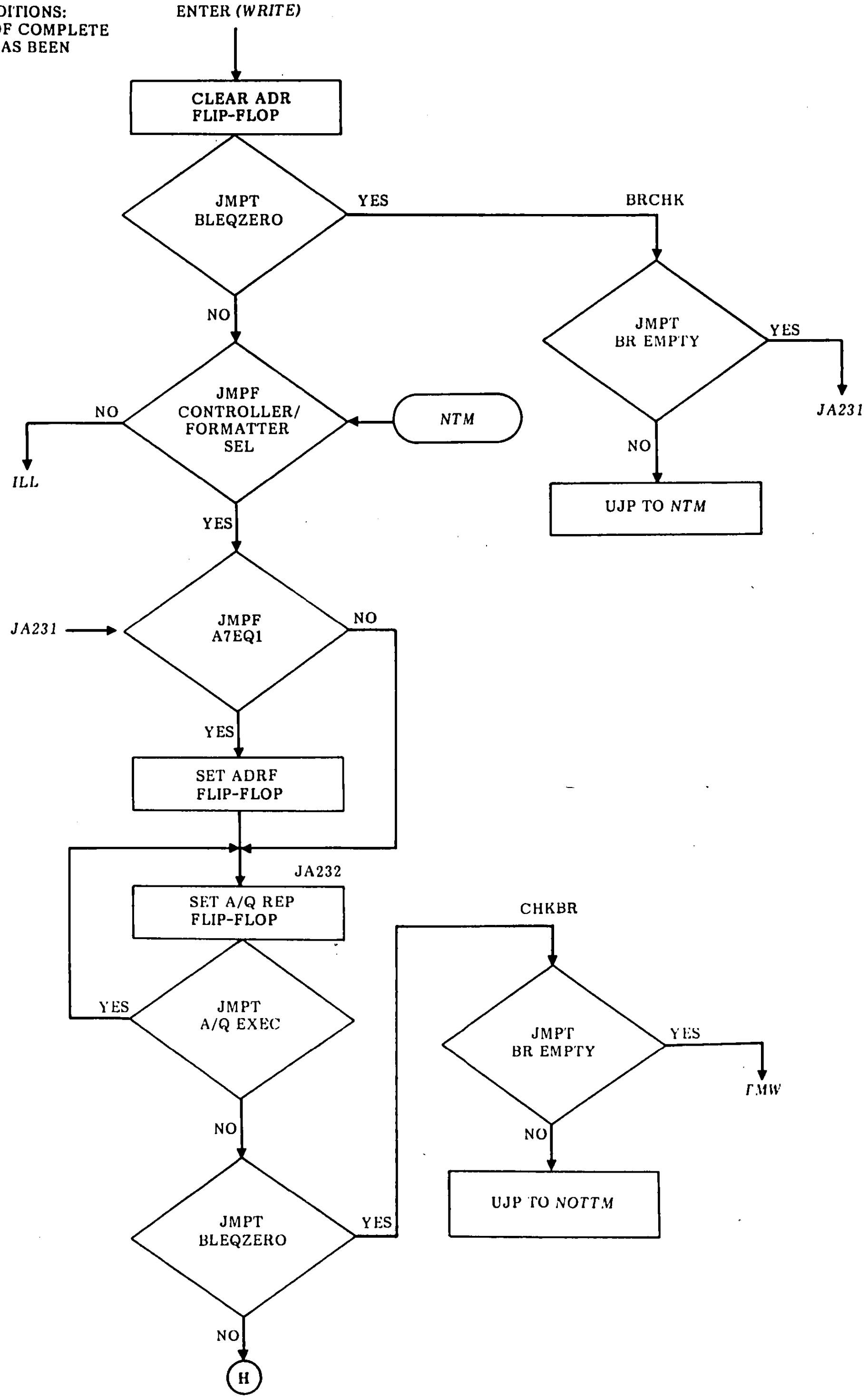


Figure B-1. Storage Module Drive Flow Charts (Sheet 18 of 45)

WRITE W · A

ENTRY CONDITIONS:
TRANSFER OF COMPLETE
FLIP-FLOP HAS BEEN
CLEARED.



0503

Figure B-1. Storage Module Drive Flow Charts (Sheet 19 of 45)

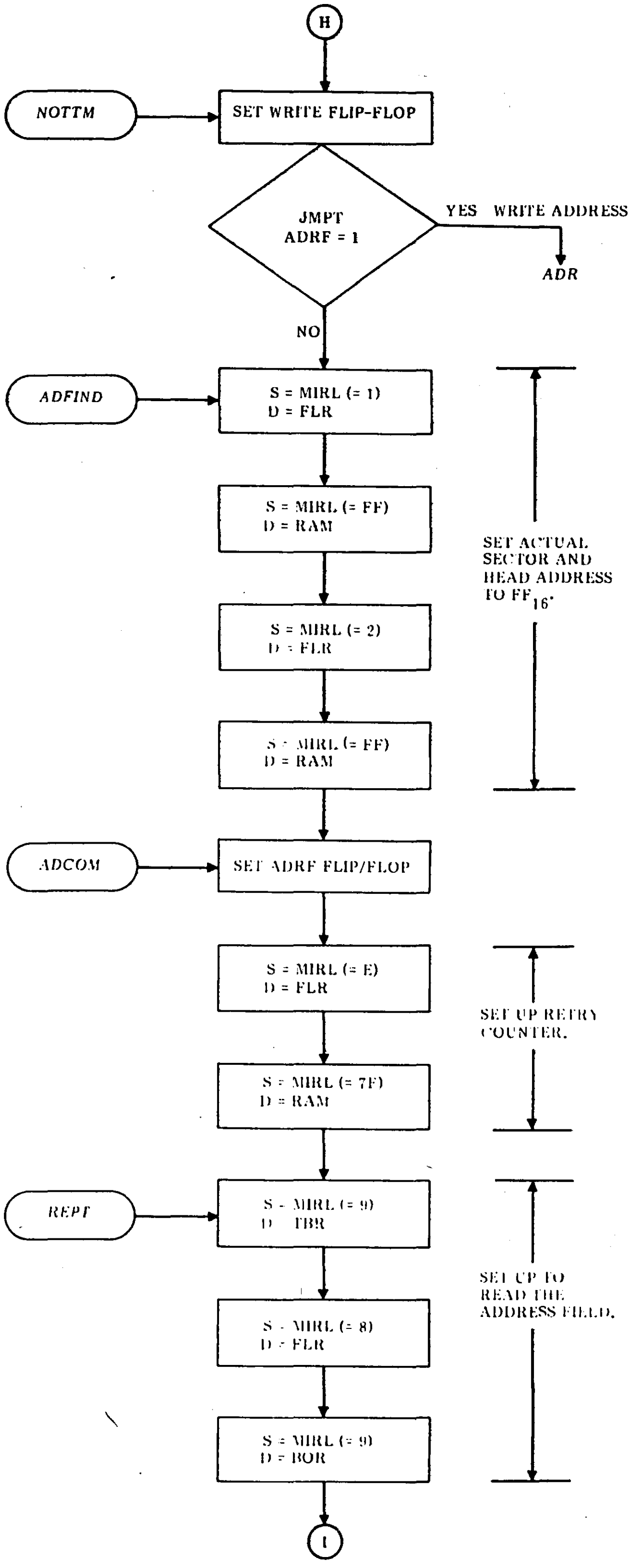


Figure B-1. Storage Module Drive Flow Charts (Sheet 20 of 45)

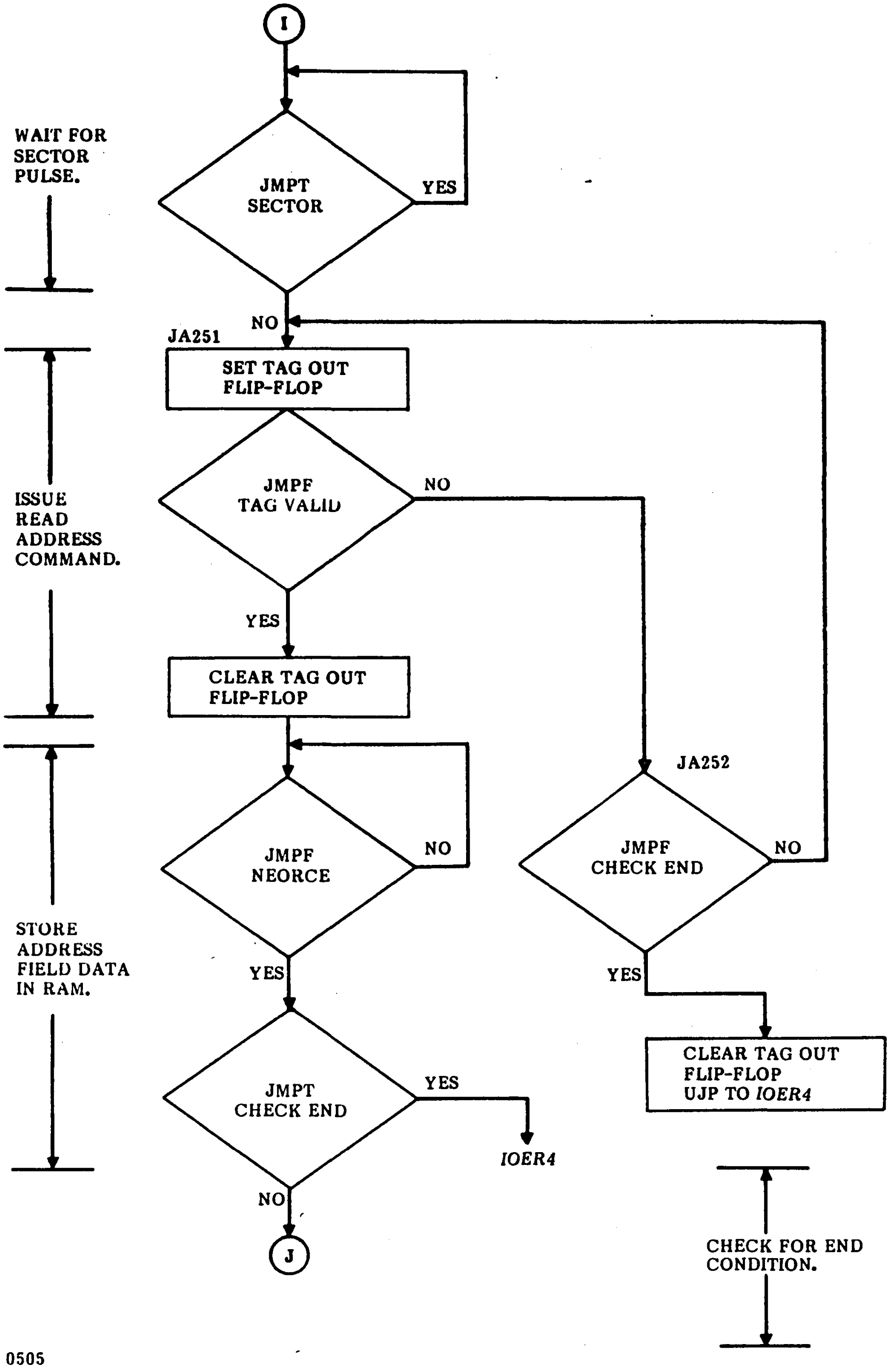
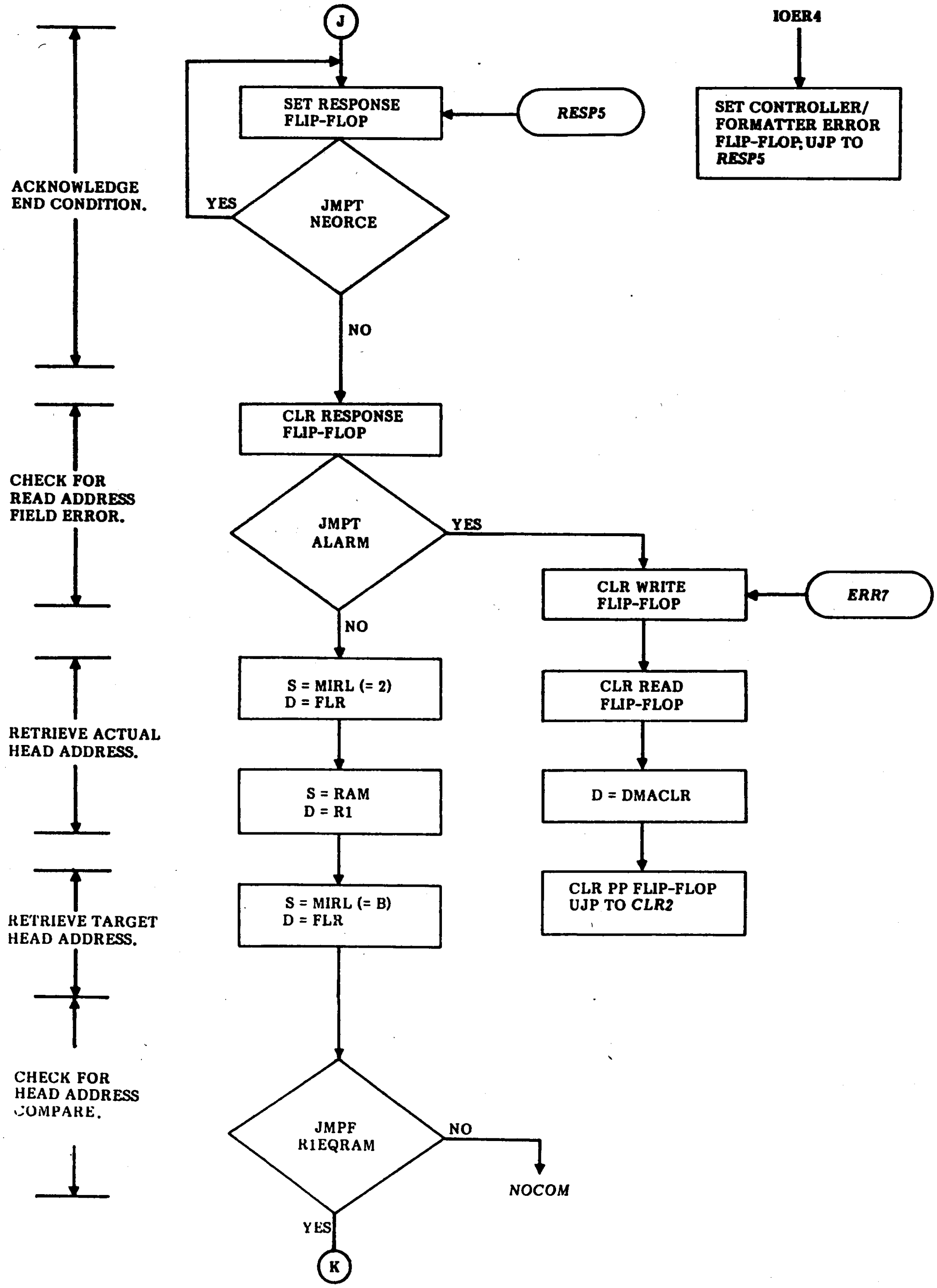


Figure B-1. Storage Module Drive Flow Charts (Sheet 21 of 45)


J
IOER4
SET RESPONSE FLIP-FLOP
RESP5
SET CONTROLLER/ FORMATTER ERROR FLIP-FLOP; UJP TO RESP5
ACKNOWLEDGE END CONDITION.
YES
JMPT NEORCE
NO
CLR RESPONSE FLIP-FLOP
CHECK FOR READ ADDRESS FIELD ERROR.
JMPT ALARM
YES
CLR WRITE FLIP-FLOP
ERR7
NO
S = MIRL (= 2)
D = FLR
CLR READ FLIP-FLOP
RETRIEVE ACTUAL HEAD ADDRESS.
S = RAM
D = R1
D = DMACLR
RETRIEVE TARGET HEAD ADDRESS.
S = MIRL (= B)
D = FLR
CLR PP FLIP-FLOP UJP TO CLR2
CHECK FOR HEAD ADDRESS COMPARE.
JMPF R1EQRAM
NO
NOCOM
YES
K

0506

Figure B-1. Storage Module Drive Flow Charts (Sheet 22 of 45)

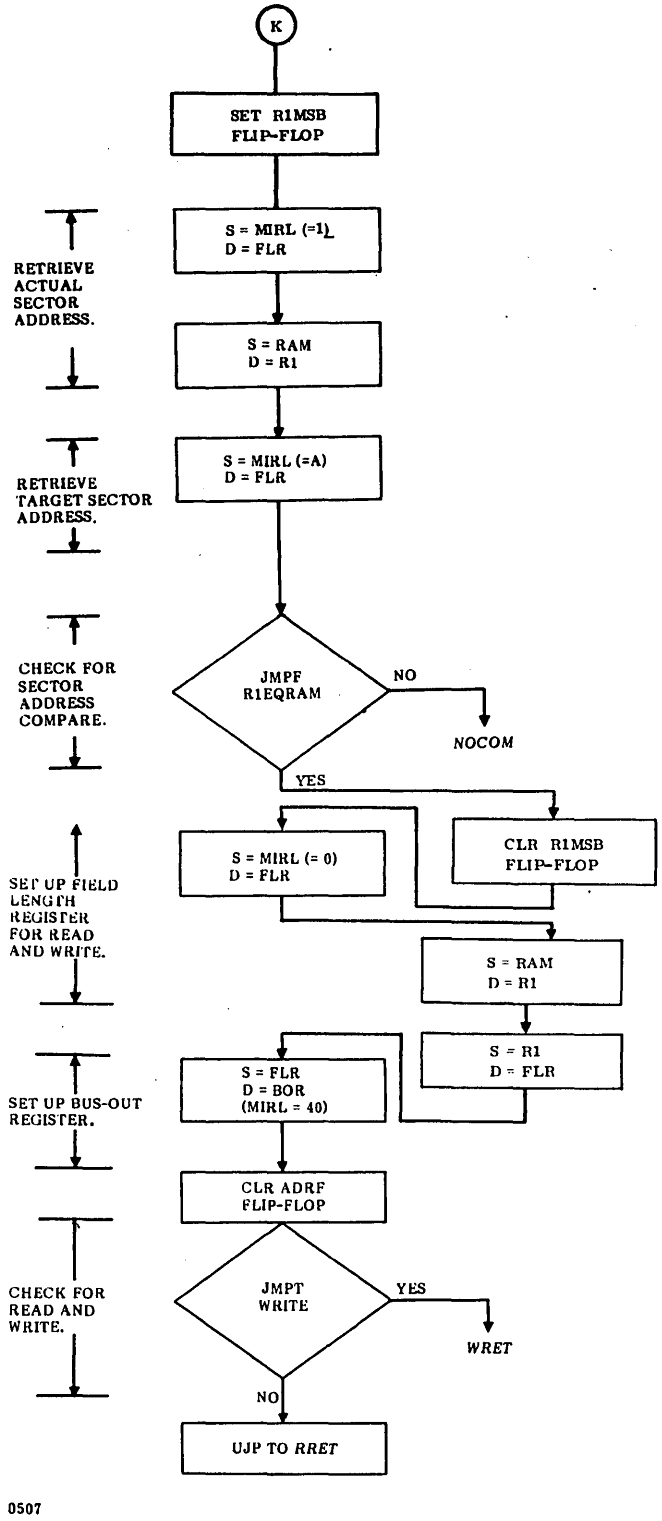
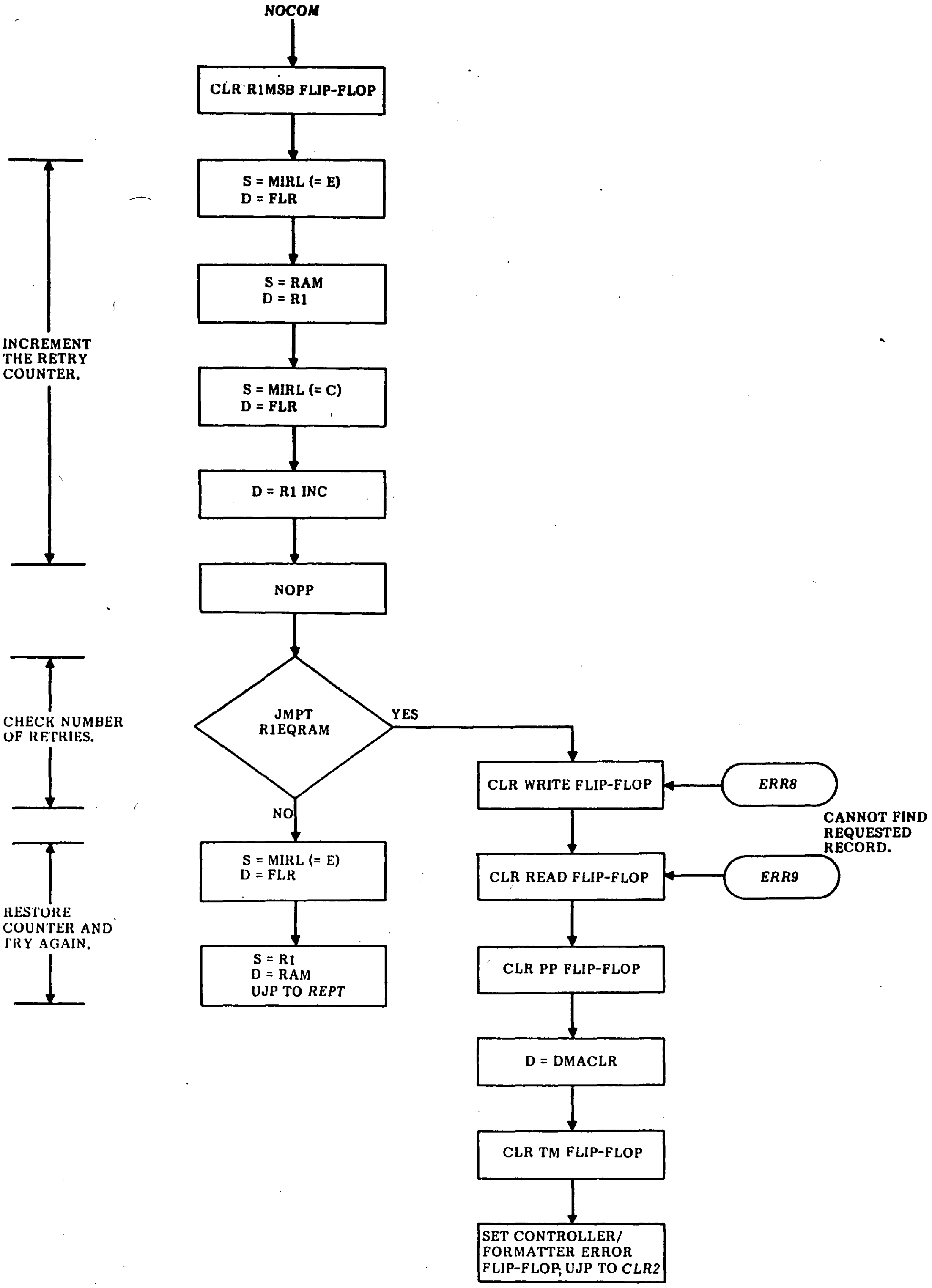


Figure B-1. Storage Module Drive Flow Charts (Sheet 23 of 45)



0508

Figure B-1. Storage Module Drive Flow Charts (Sheet 24 of 45)

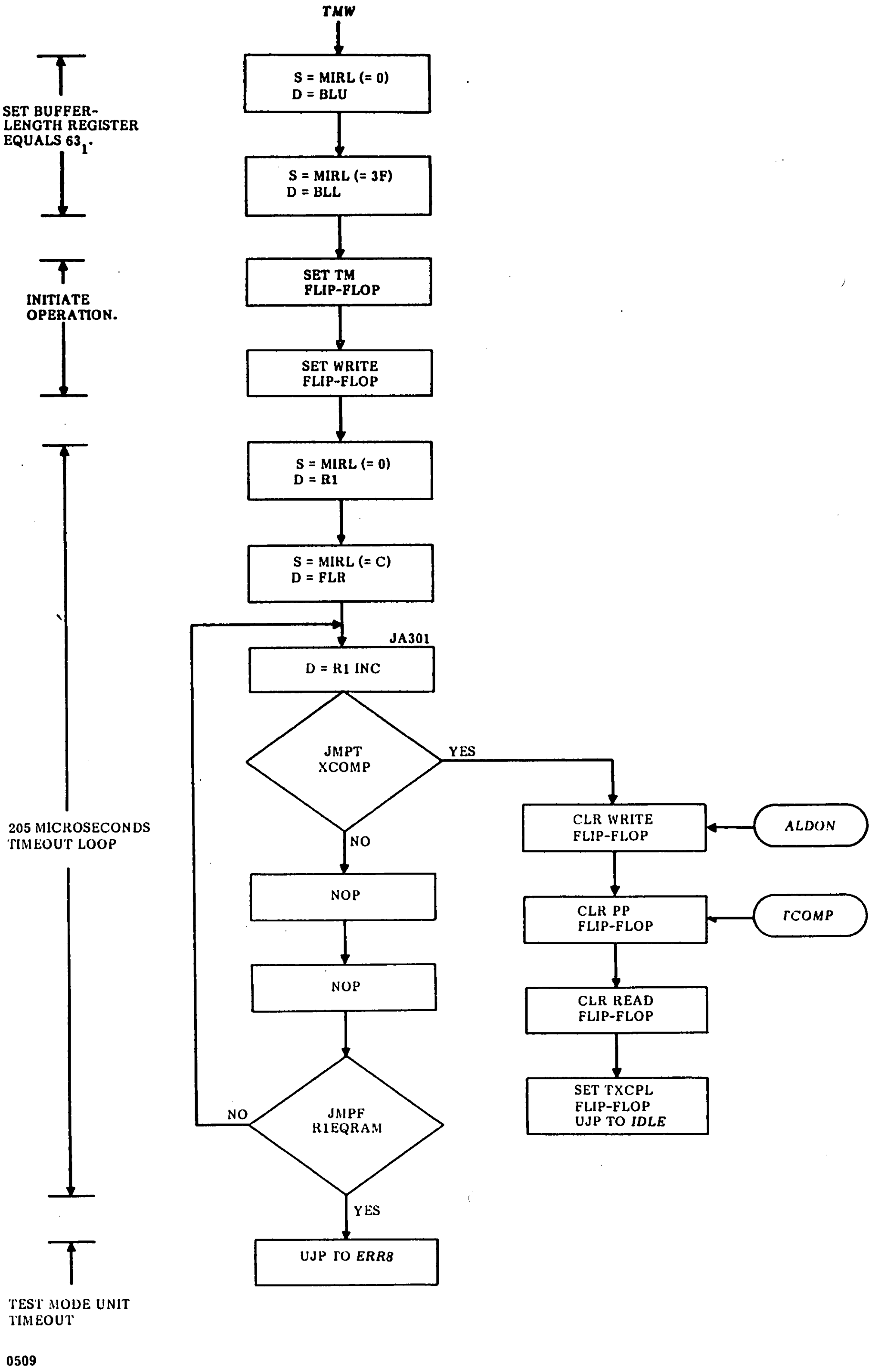
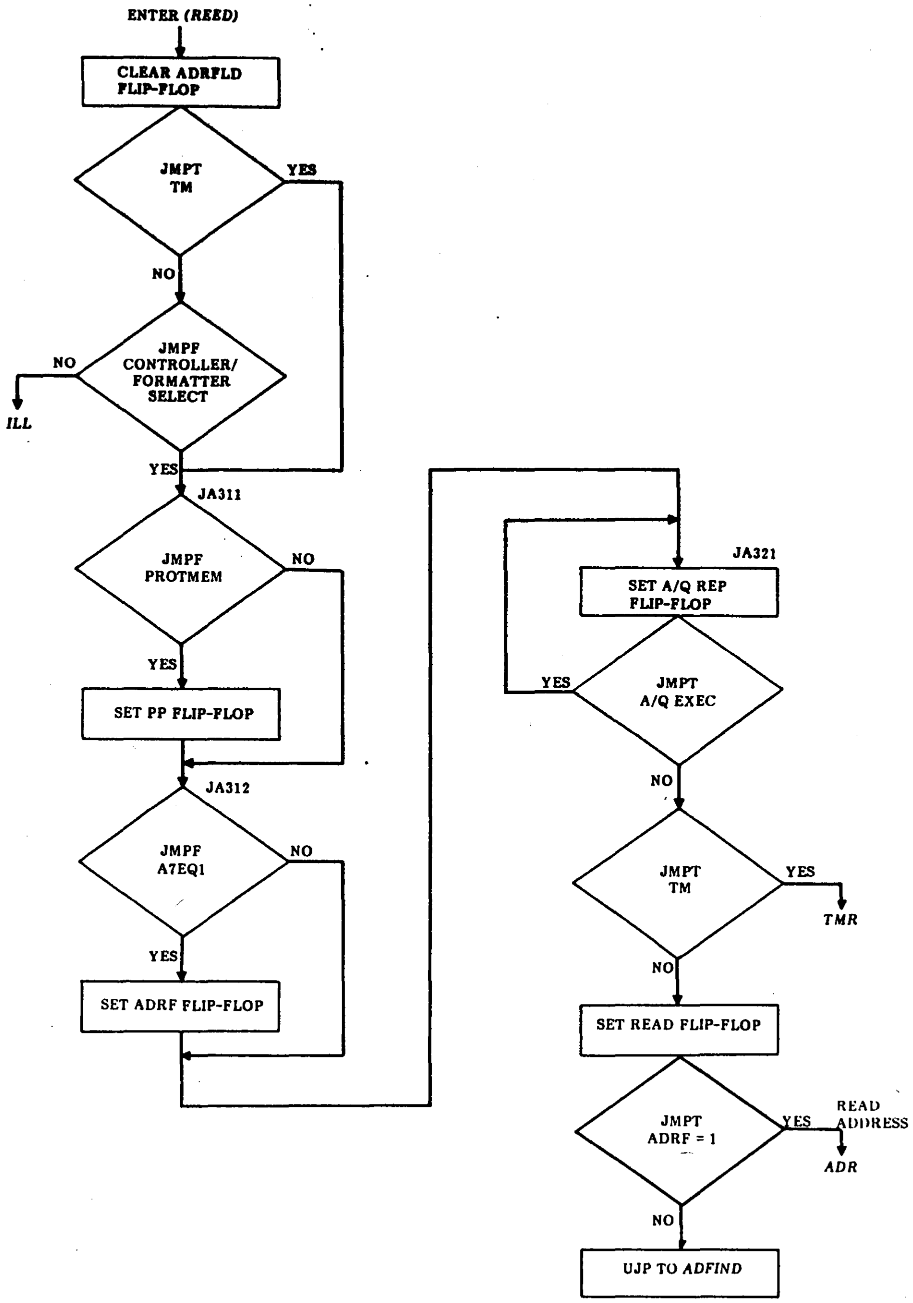


Figure B-1. Storage Module Drive Flow Charts (Sheet 25 of 45)

READ W-S

ENTRY CONDITIONS:
TRANSFER OF COMPLETE
FLIP-FLOP HAS BEEN
CLEARED.



0510

Figure B-1. Storage Module Drive Flow Charts (Sheet 26 of 45)

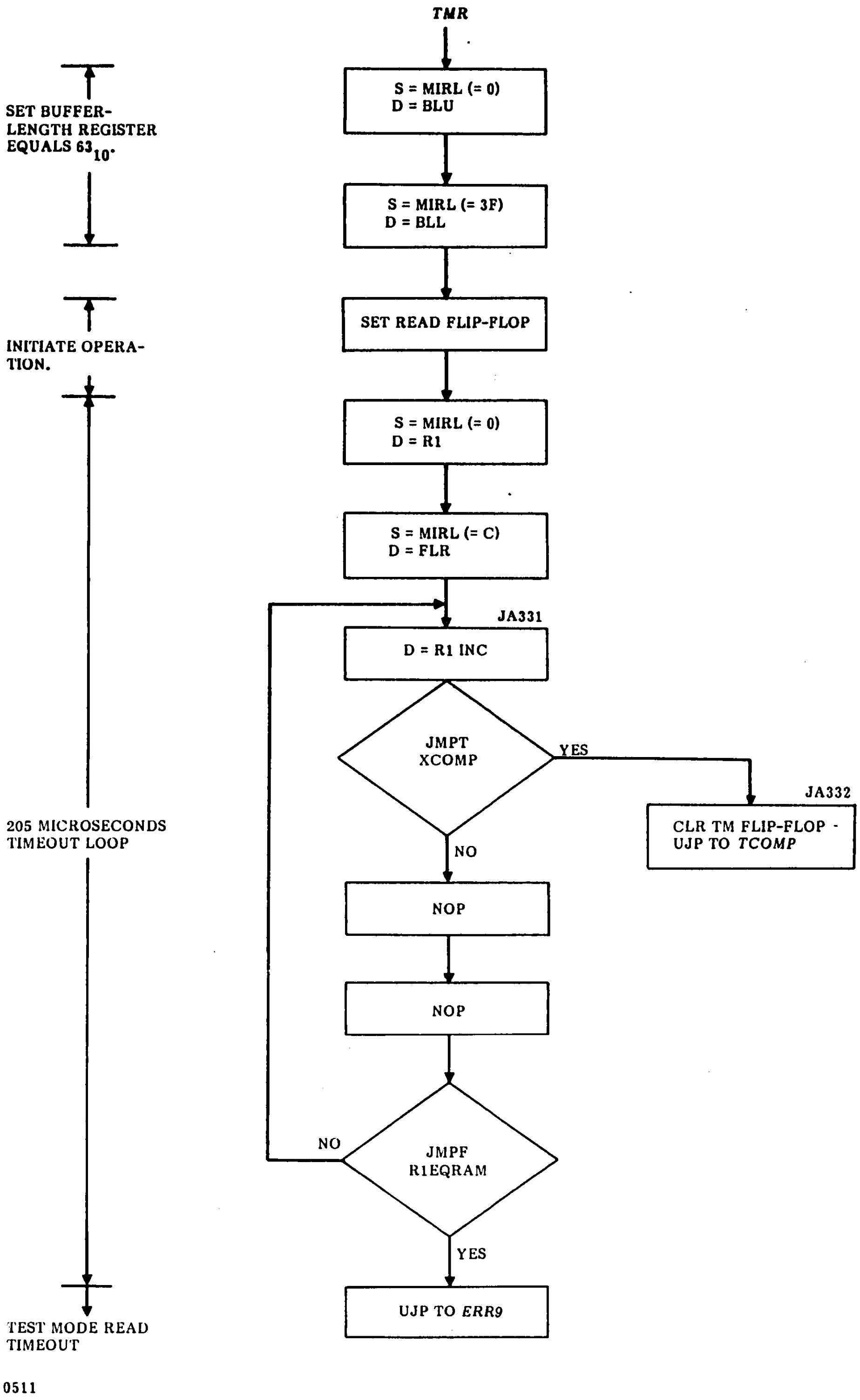
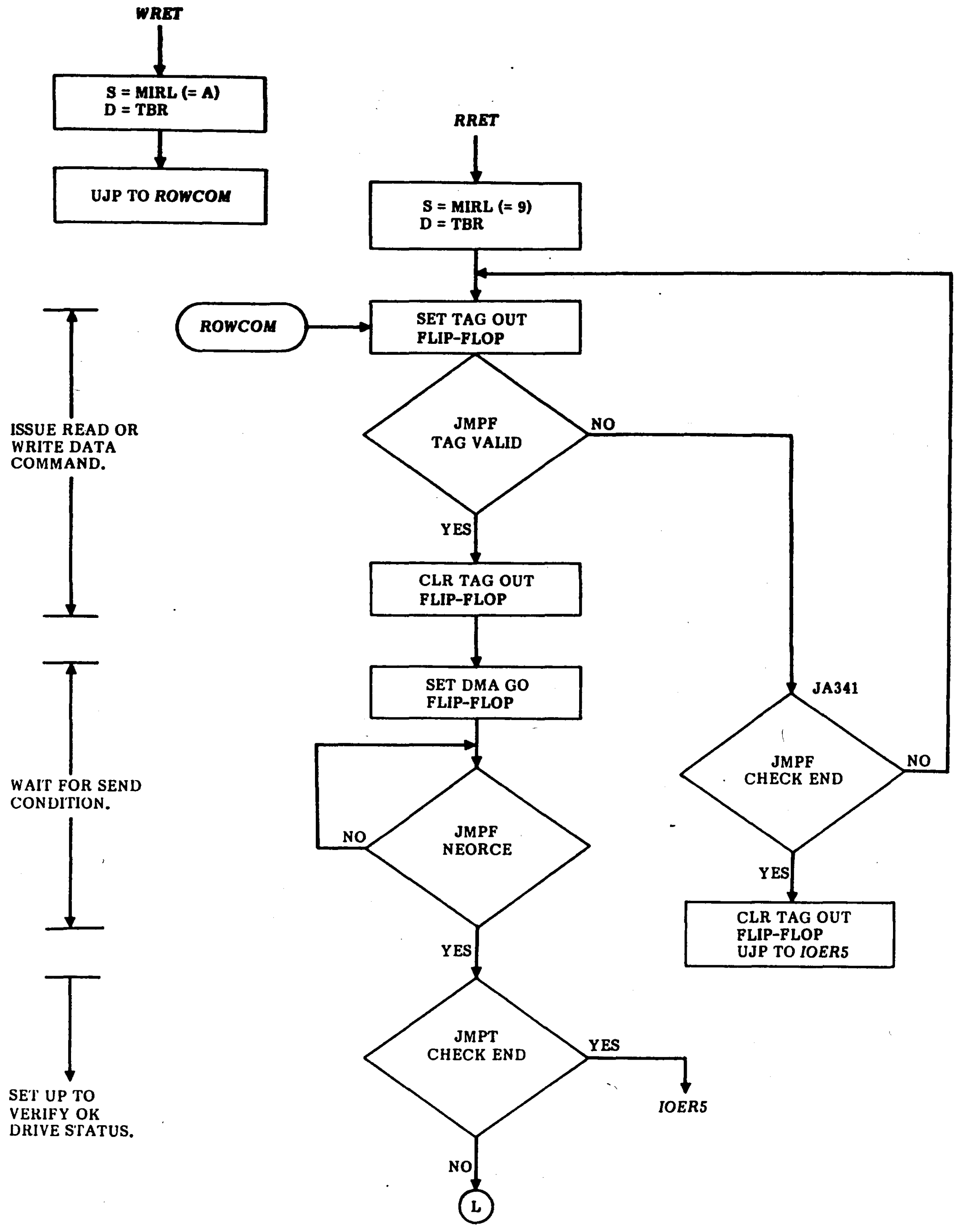
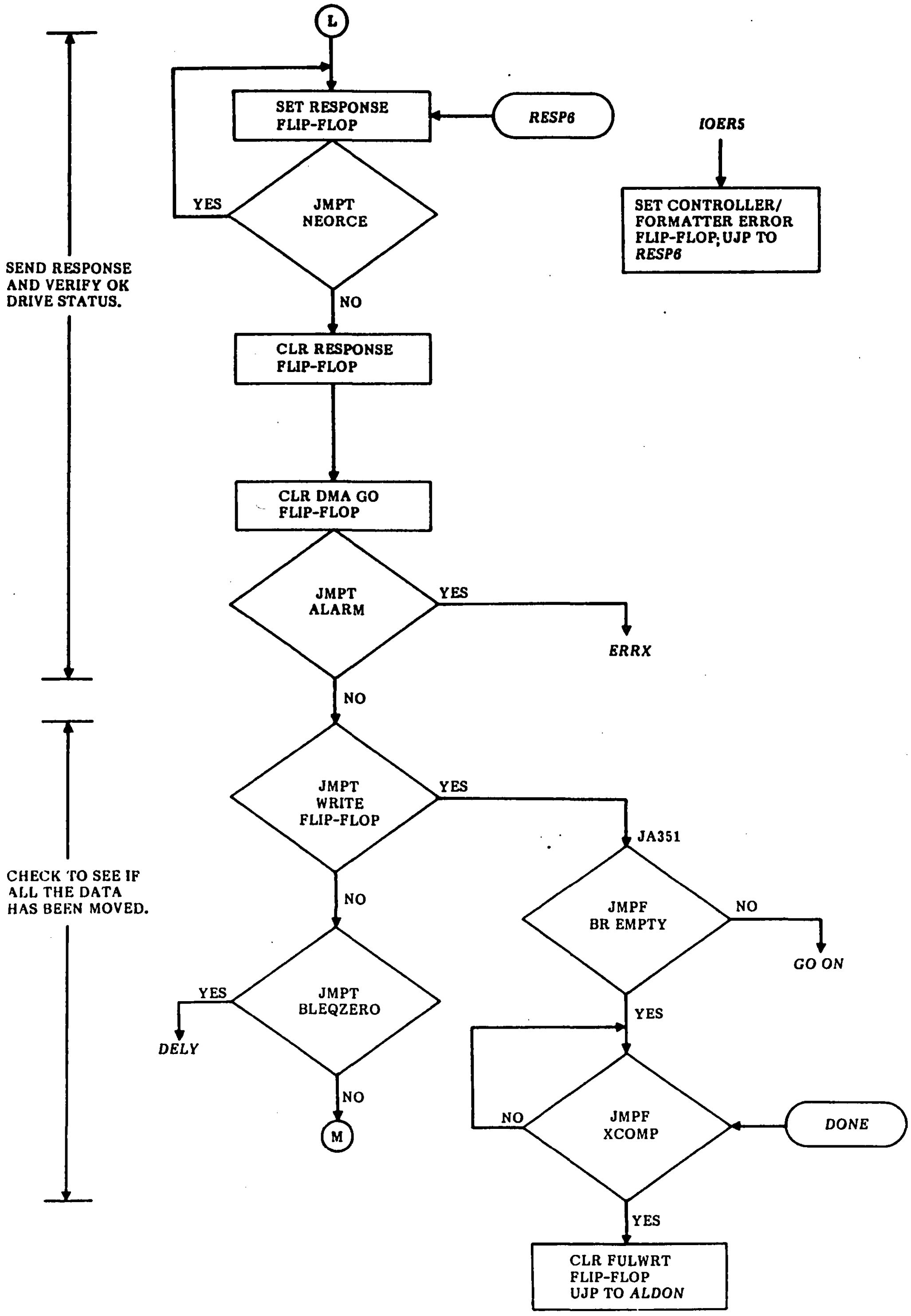


Figure B-1. Storage Module Drive Flow Charts (Sheet 27 of 45)



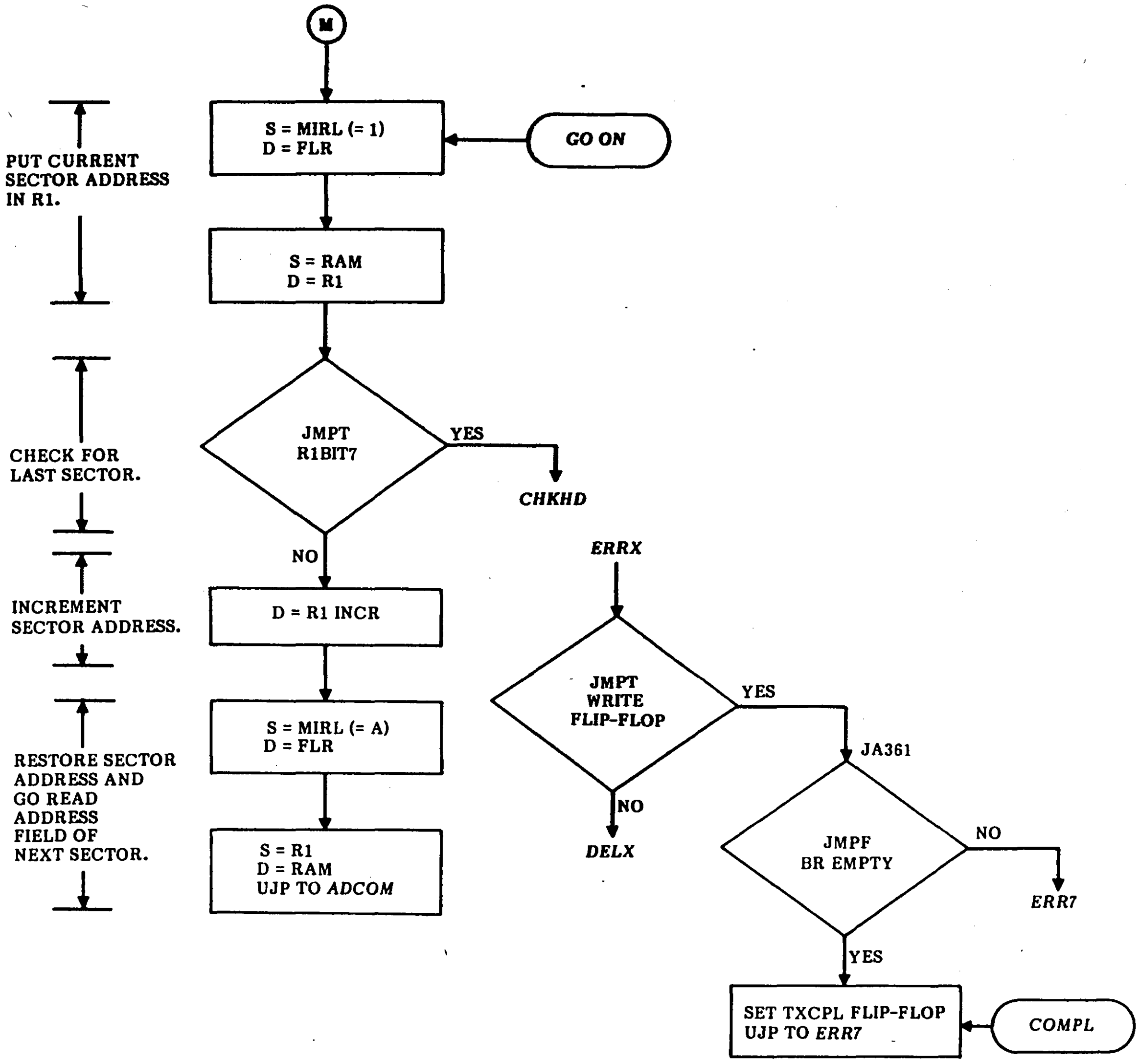
0512

Figure B-1. Storage Module Drive Flow Charts (Sheet 28 of 45)



0513

Figure B-1. Storage Module Drive Flow Charts (Sheet 29 of 45)



0514

Figure B-1. Storage Module Drive Flow Charts (Sheet 30 of 45)

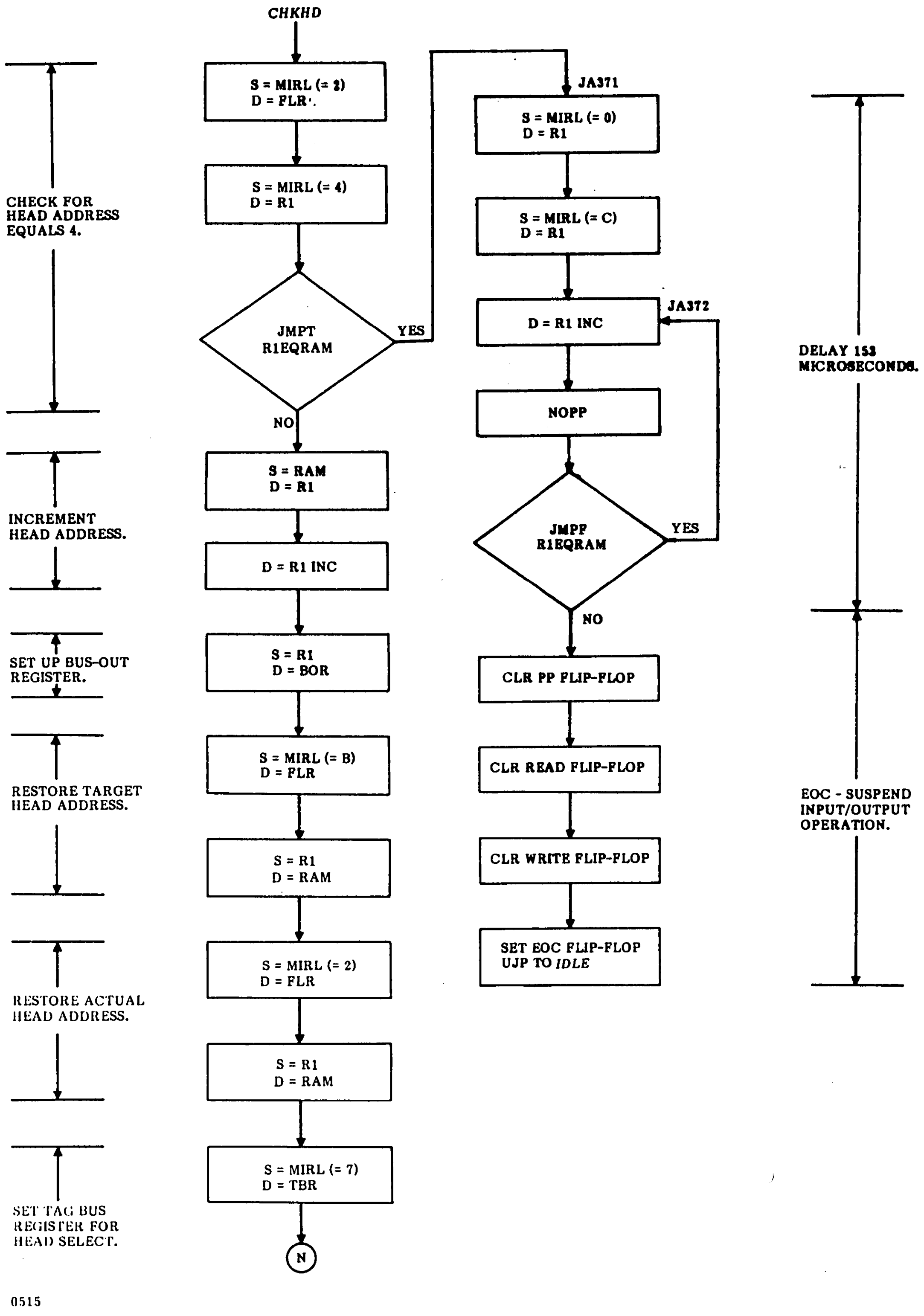
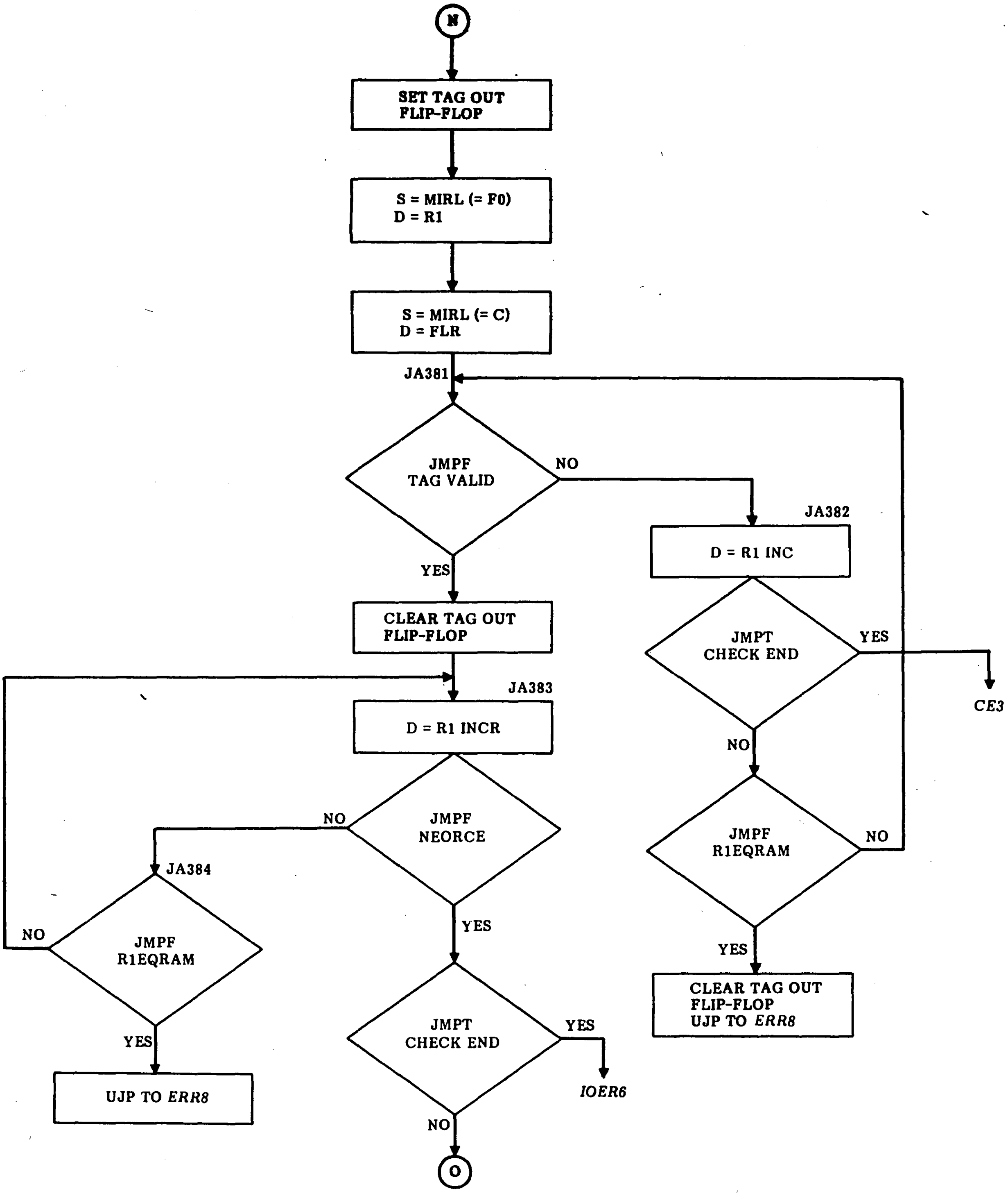
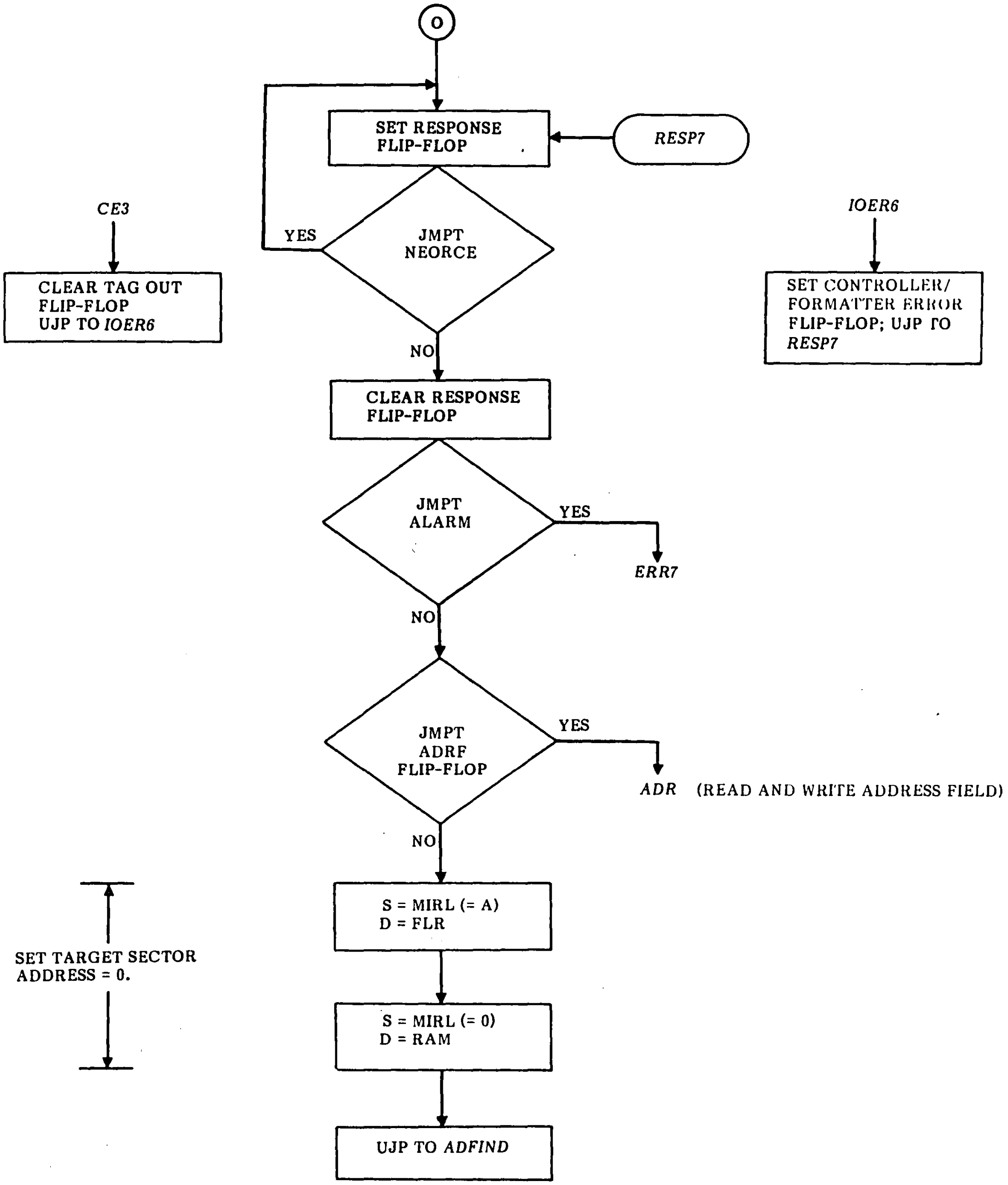


Figure B-1. Storage Module Drive Flow Charts (Sheet 31 of 45)



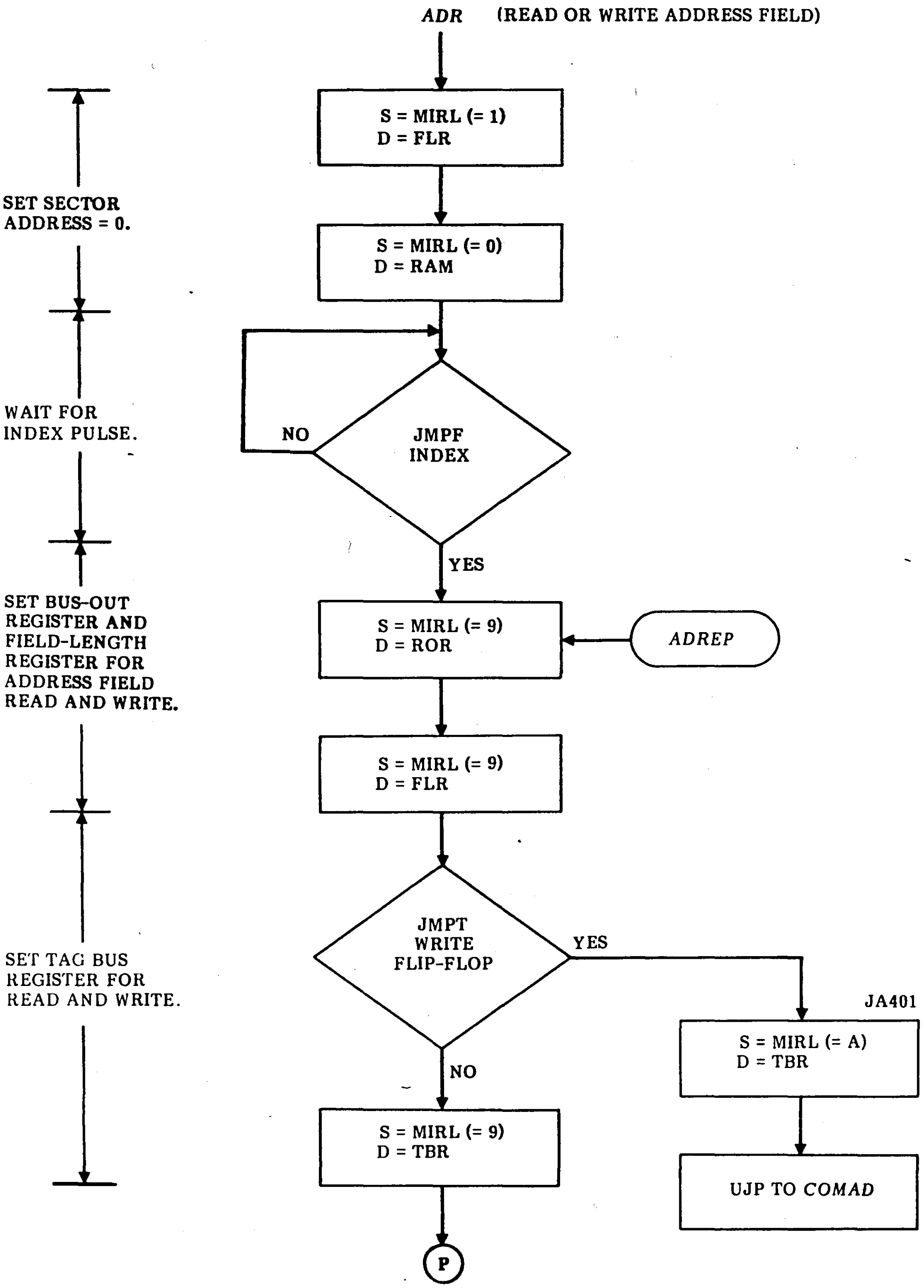
0516

Figure B-1. Storage Module Drive Flow Charts (Sheet 32 of 45)



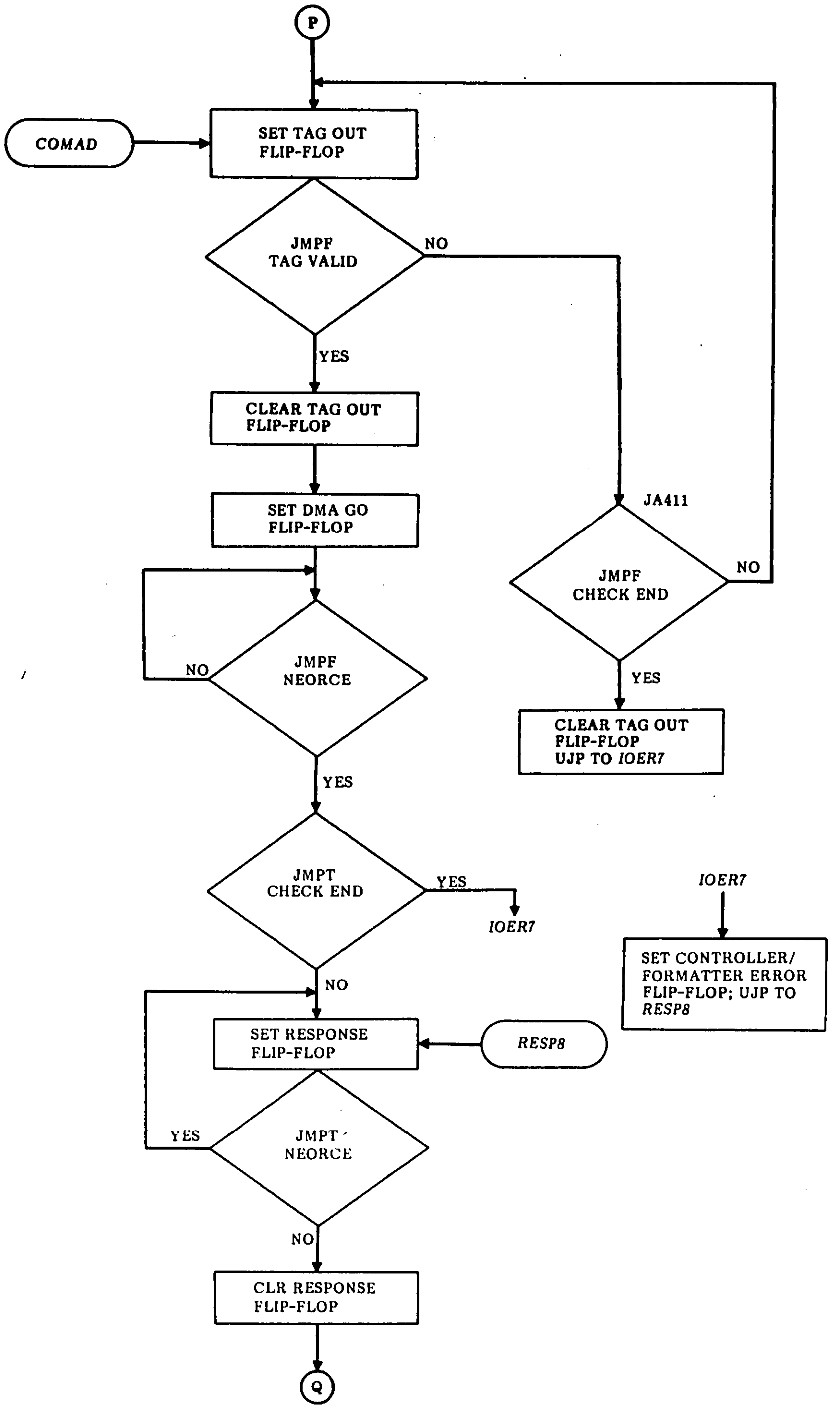
0517

Figure B-1. Storage Module Drive Flow Charts (Sheet 33 of 45)



0518

Figure B-1. Storage Module Drive Flow Charts (Sheet 34 of 45)



0519

Figure B-1. Storage Module Drive Flow Charts (Sheet 35 of 45)

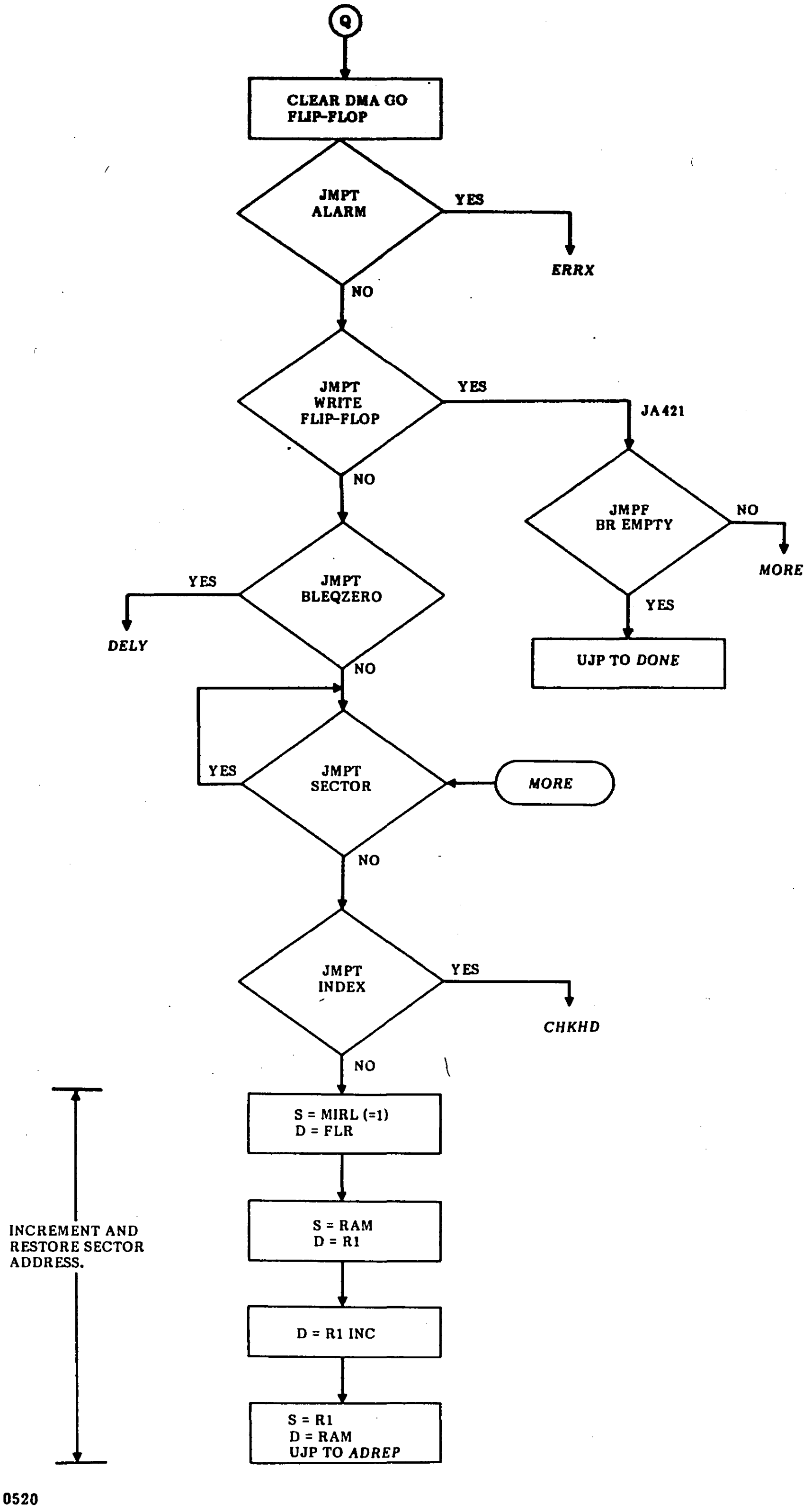
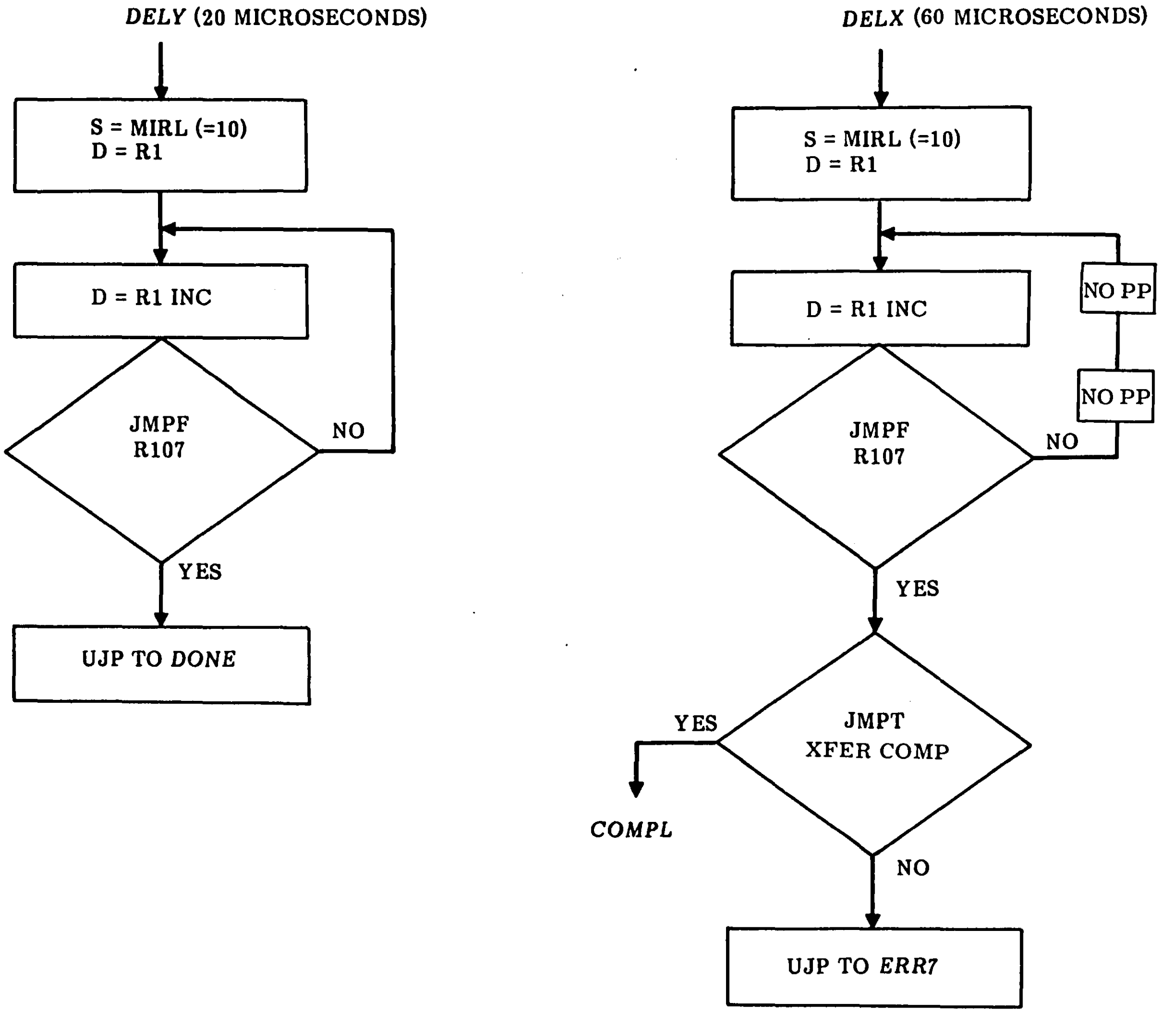


Figure B-1. Storage Module Drive Flow Charts (Sheet 36 of 45)



0521

Figure B-1. Storage Module Drive Flow Charts (Sheet 37 of 45)

AUTOLOAD

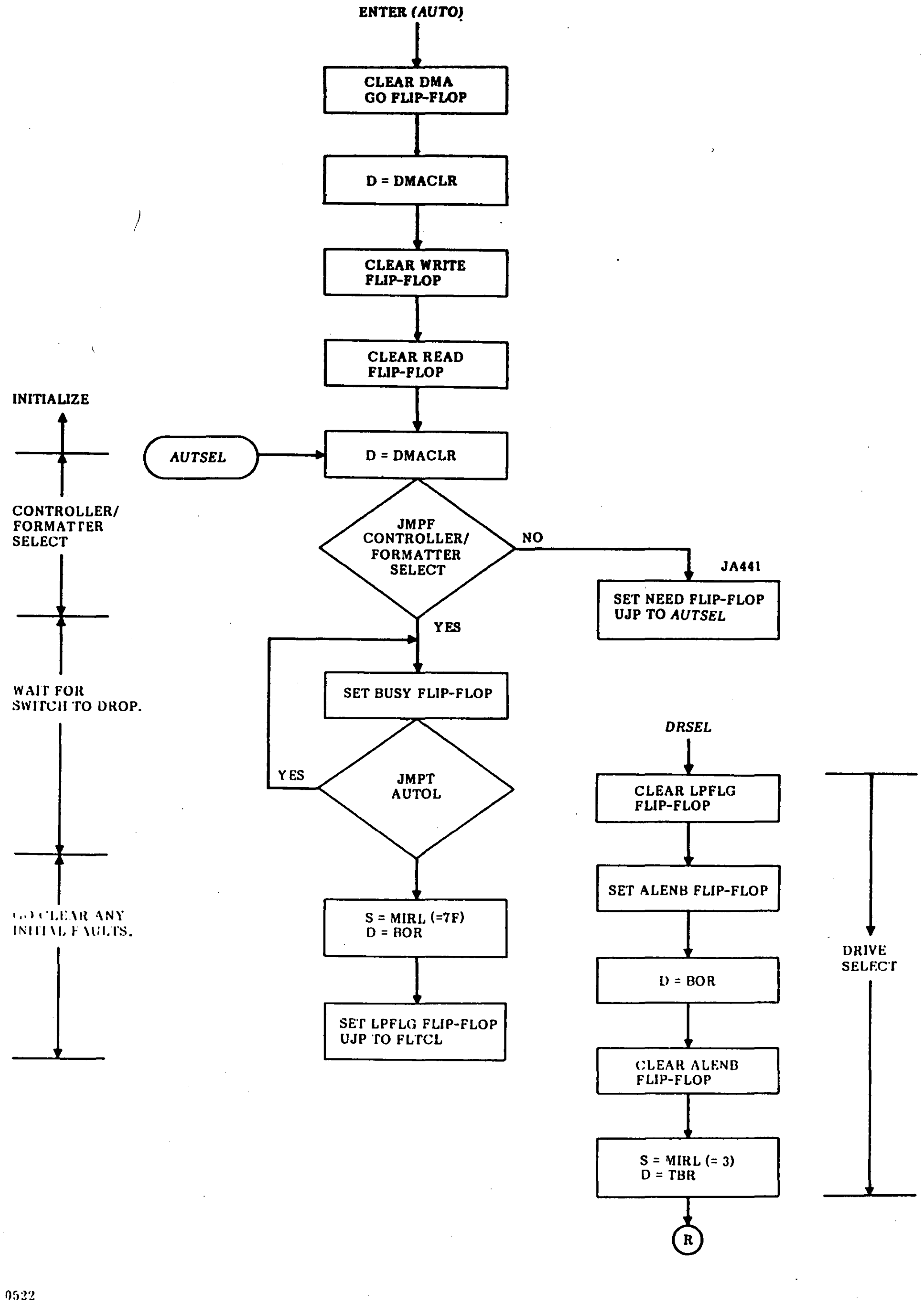
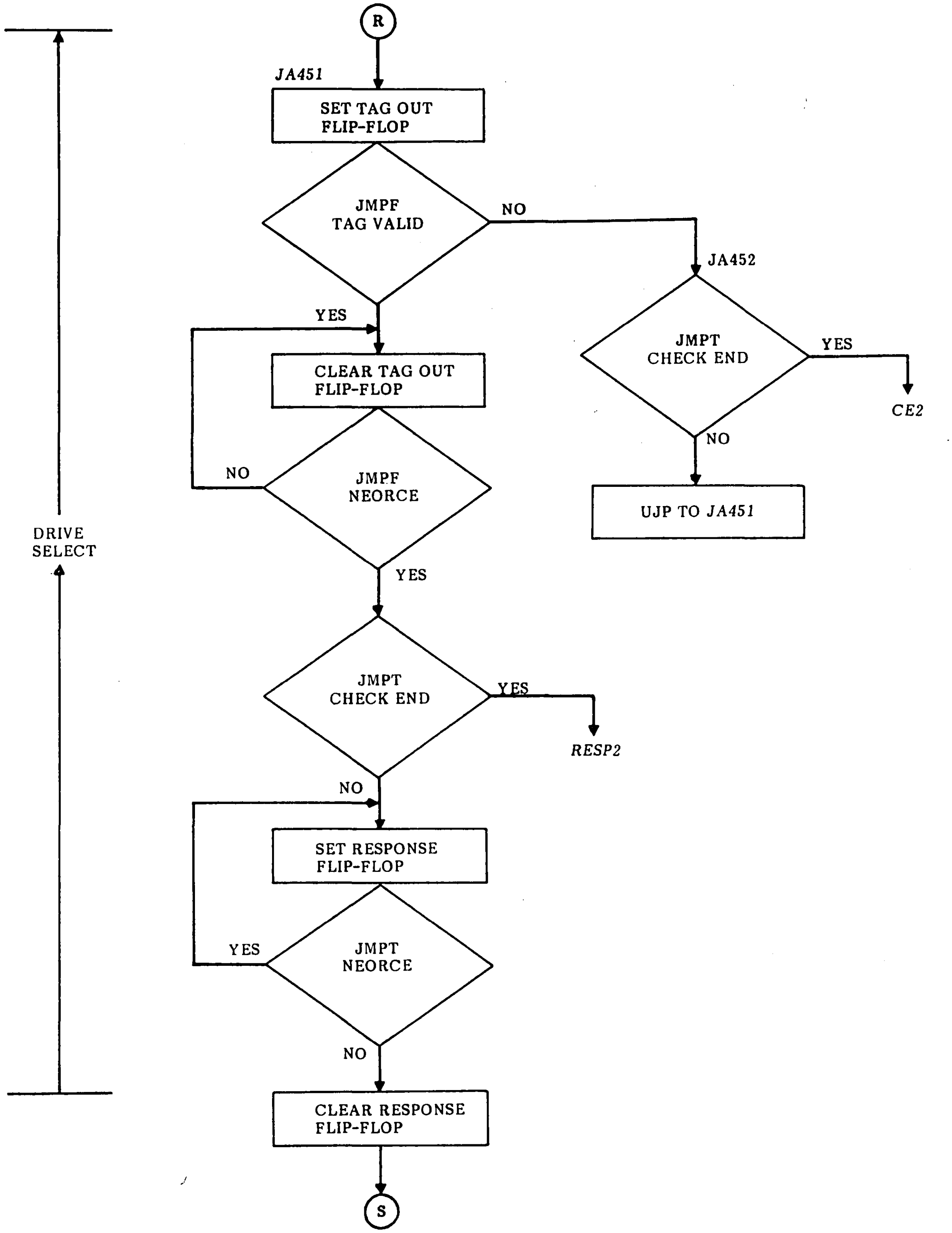
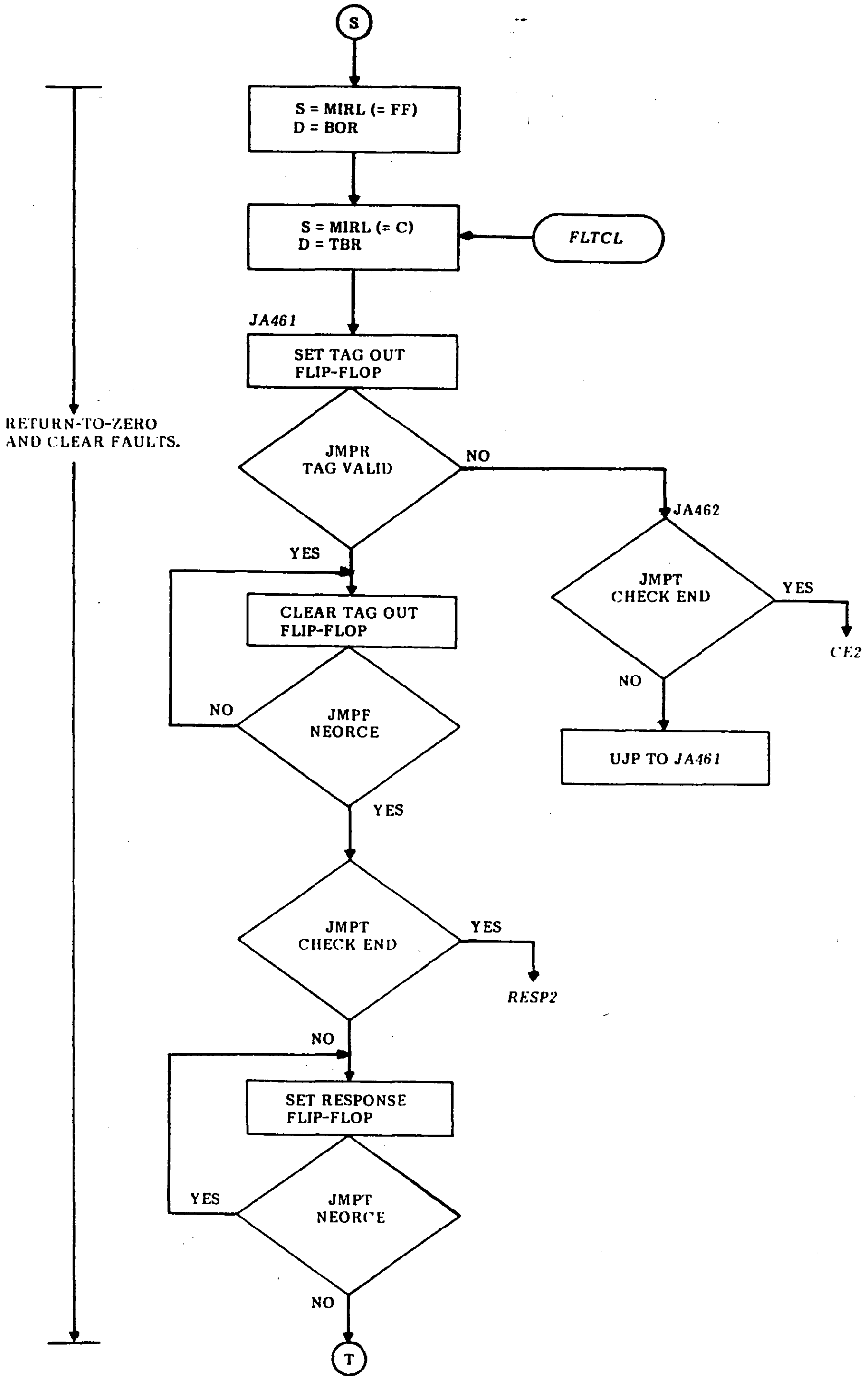


Figure B-1. Storage Module Drive Flow Charts (Sheet 38 of 45)



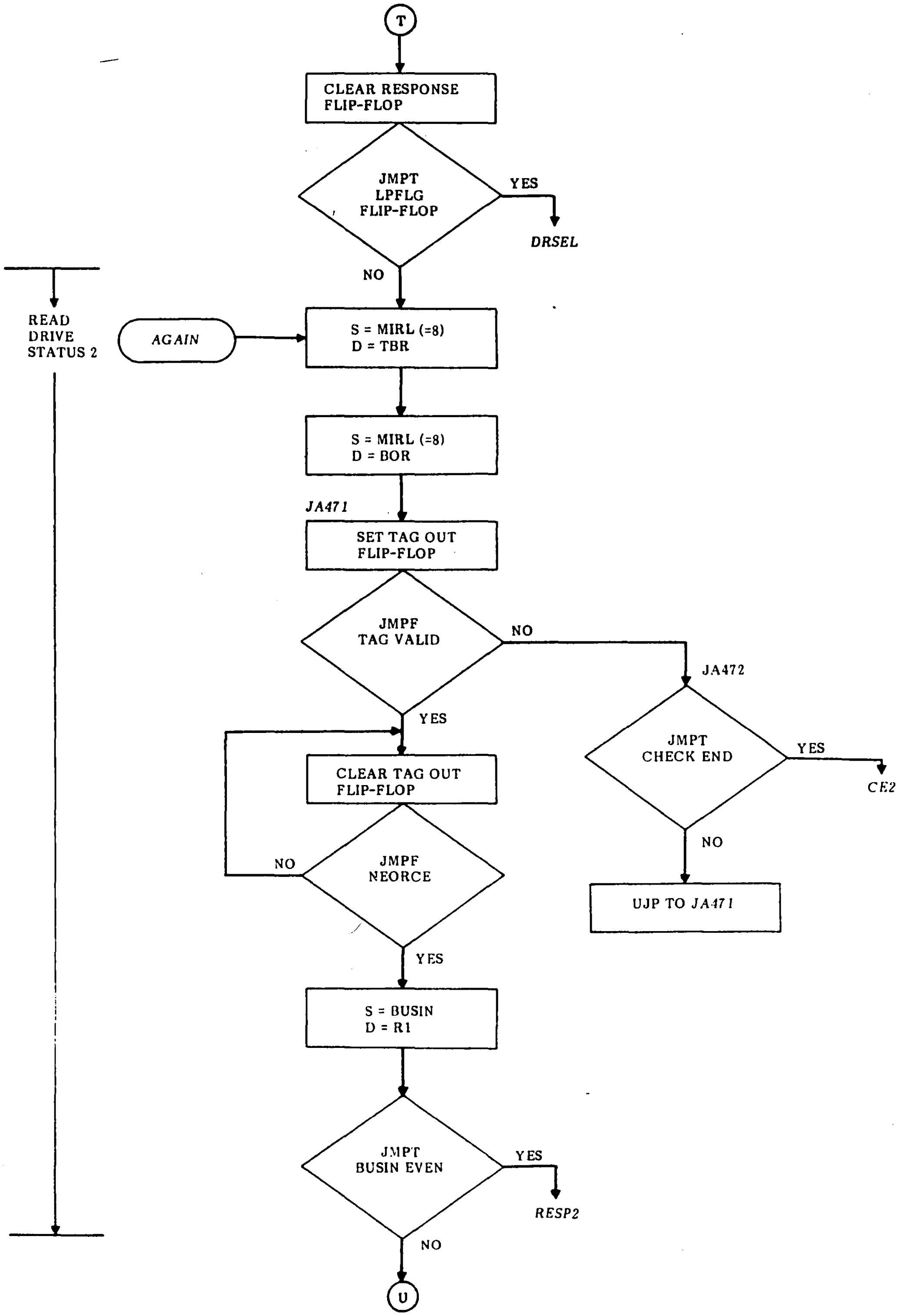
0523

Figure B-1. Storage Module Drive Flow Charts (Sheet 39 of 45)



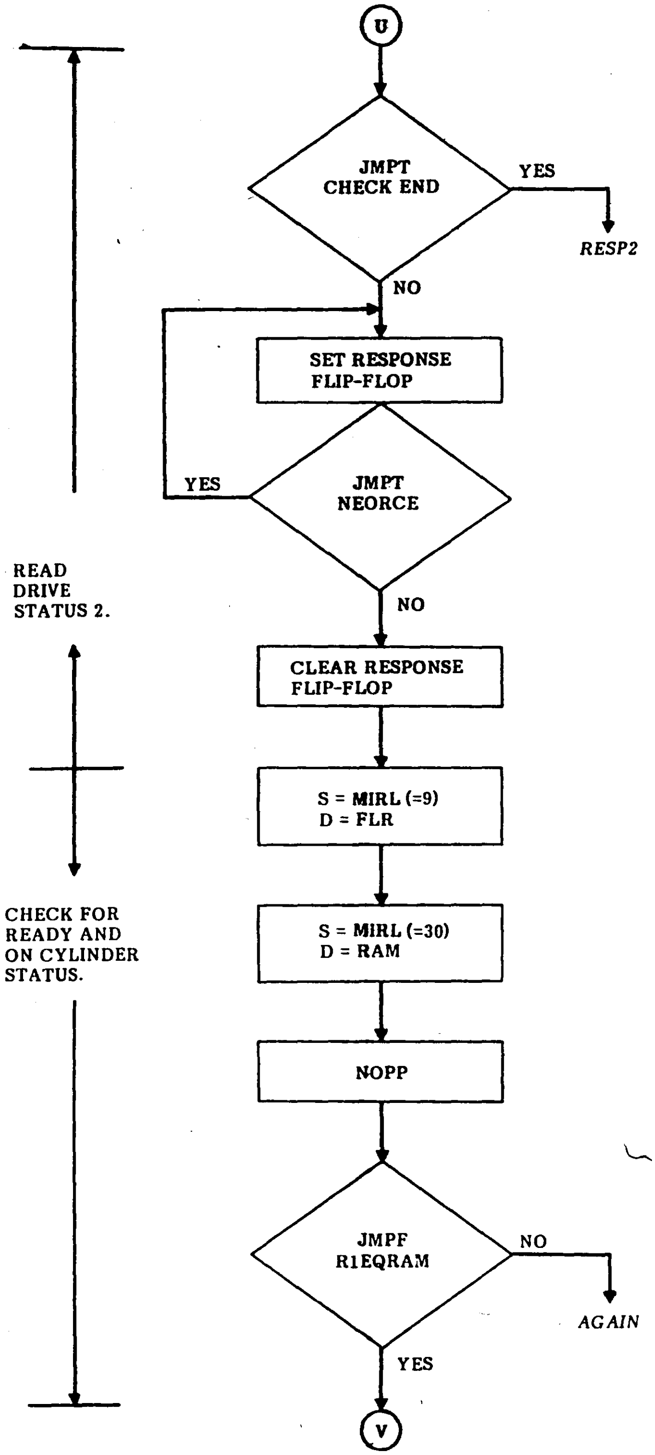
0524

Figure B-1. Storage Module Drive Flow Charts (Sheet 40 of 45)



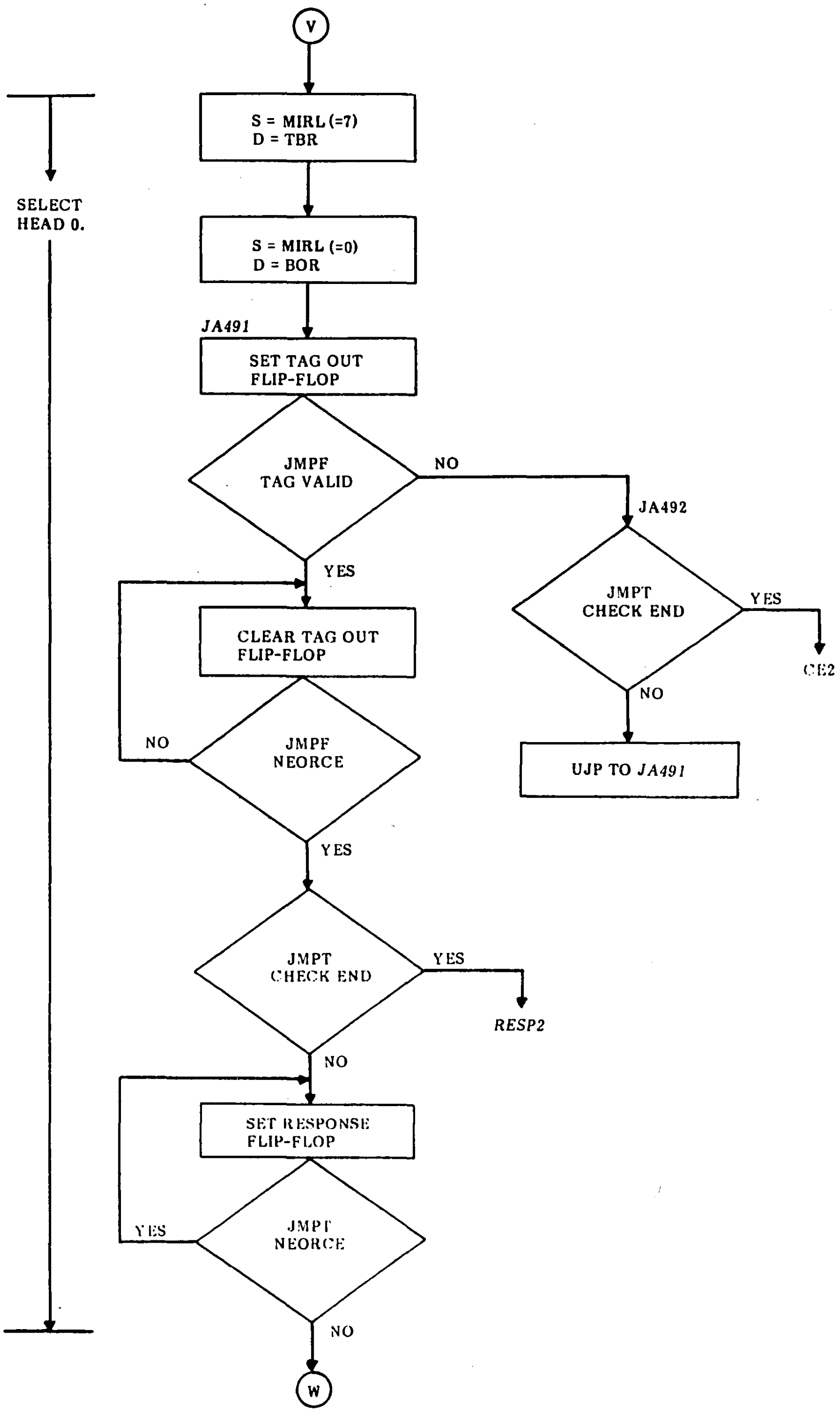
0525

Figure B-1. Storage Module Drive Flow Charts (Sheet 41 of 45)



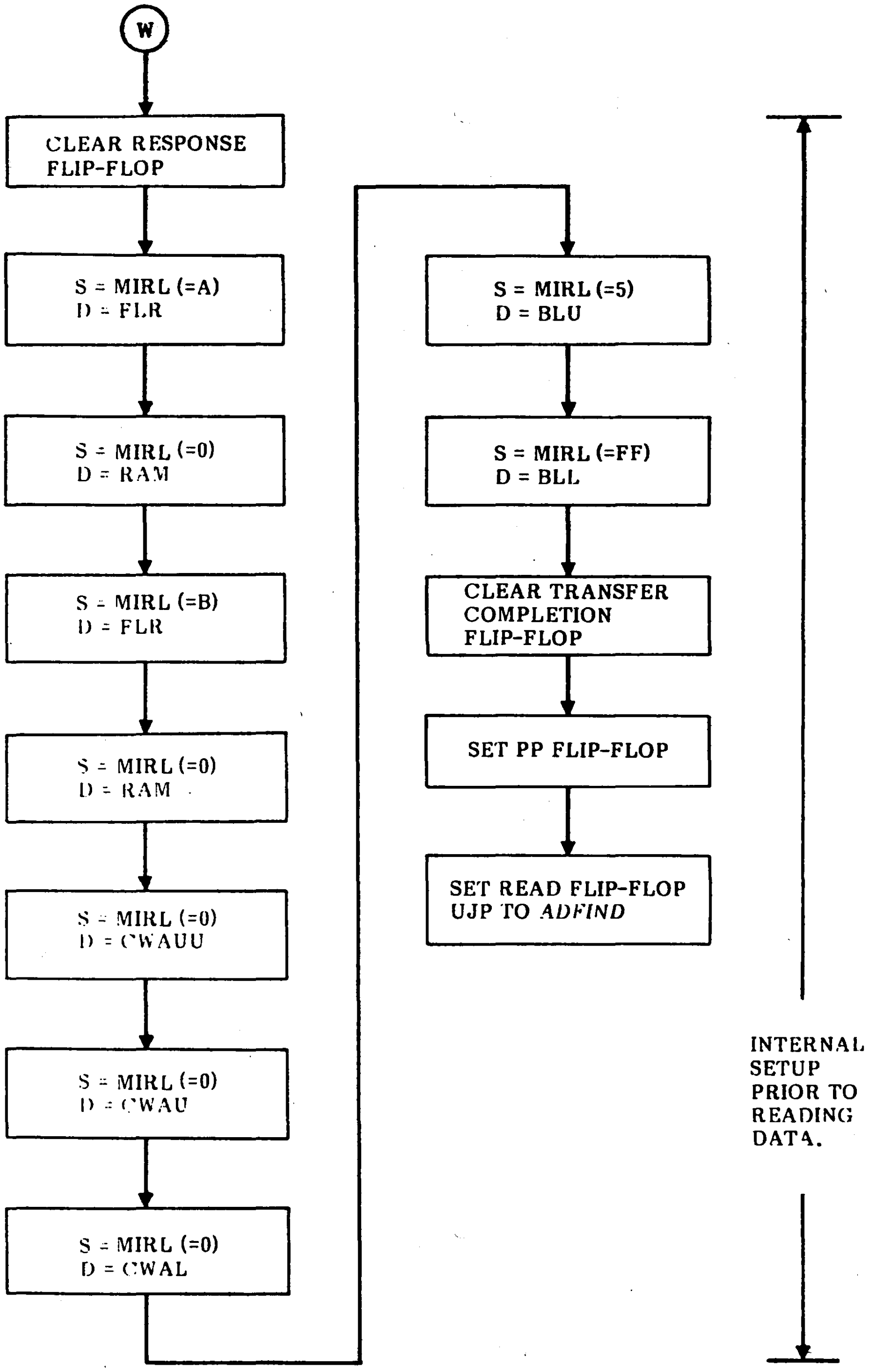
0526

Figure B-1. Storage Module Drive Flow Charts (Sheet 42 of 45)



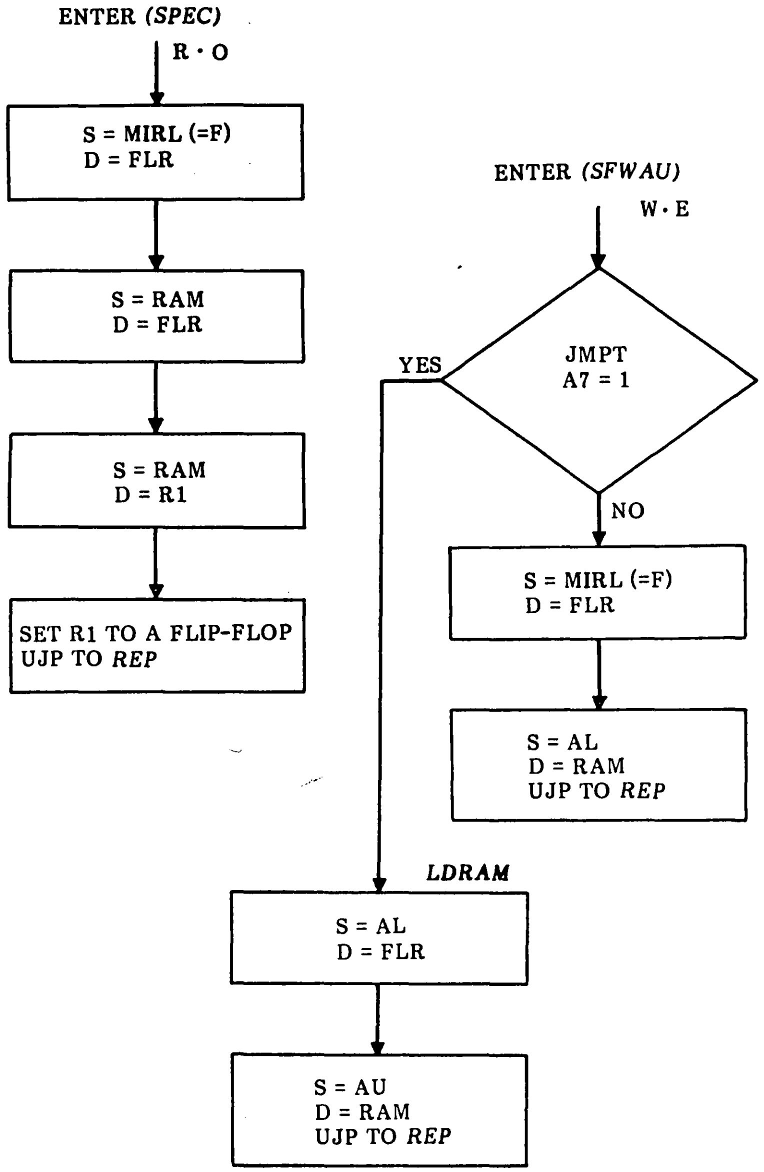
0527

Figure B-1. Storage Module Drive Flow Charts (Sheet 43 of 45)



0528

Figure B-1. Storage Module Drive Flow Charts (Sheet 44 of 45)



0529

Figure B-1. Storage Module Drive Flow Charts (Sheet 45 of 45)

DRIVE INTERFACE MICRO PROGRAM LISTING

```
      DISK                    PAGE   1            DATE: 03/05/77

                        NAM   DISK
                 SEQSD  MAC   S,D,DD
                        VFD   X8/*/2-V,N7/0,N1/1,
                        VFD   X3/'S',,X4/'D',,X9/'DD'
                        EMC
                 SEQFF  MAC   M,F
                        VFD   X8/*/2-V,N6/0,N1/1,X1/'M',
                        VFD   N2/0,X5/'F',,N9/0
                        EMC
                 UJP    MAC   JA
                        VFD   X8/*/2-V,N1/1,N7/0,
                        VFD   N7/0,X9/'JA'/2-V
                        EMC
                 UJPSD  MAC   S,D,JA
                        VFD   X8/*/2-V,N1/1,N6/0,N1/1,
                        VFD   X3/'S',,X4/'D',,X9/'JA'/2-V
                        EMC
                 UJPFF  MAC   M,F,JA
                        VFD   X8/*/2-V,N1/1,N5/0,N1/1,X1/'M',
                        VFD   N2/0,X5/'F',,X9/'JA'/2-V
                        EMC
                 JMPF   MAC   JC,JA
                        VFD   X8/*/2-V,N1/0,X5/'JC',,N2/0,
                        VFD   N7/0,X9/'JA'/2-V
                        EMC
                 JMPFSD MAC   JC,S,D,JA
                        VFD   X8/*/2-V,N1/0,X5/'JC',,N1/0,N1/1,
                        VFD   X3/'S',,X4/'D',,X9/'JA'/2-V
                        EMC
                 JMPFFF MAC   JC,M,F,JA
                        VFD   X8/*/2-V,N1/0,X5/'JC',,N1/1,X1/'M',
                        VFD   N2/0,X5/'F',,X9/'JA'/2-V
                        EMC
                 JMPT   MAC   JC,JA
                        VFD   X5/*/2-V,N1/1,X5/'JC',,N2/0,
                        VFD   N7/0,X9/'JA'/2-V
                        EMC
                 JMPTSD MAC   JC,S,D,JA
                        VFD   X8/*/2-V,N1/1,X5/'JC',,N1/0,N1/1,
                        VFD   X3/'S',,X4/'D',,X9/'JA'/2-V
                        EMC
                 JMPTFF MAC   JC,M,F,JA
                        VFD   X8/*/2-V,N1/1,X5/'JC',,N1/1,X1/'M',
                        VFD   N2/0,X5/'F',,X9/'JA'/2-V
                        EMC
                 NOPP   MAC
                        VFD   X8/*/2-V,N8/0,
                        VFD   N16/0
                        EMC
```

```
                  *   S FIELD (SOURCE) EQUATES
                  *
       0000              EQU   BUSIN(0)        BUS IN
       0001              EQU   MIR(1)          MIR BITS 0-7
       0002              EQU   SR1(2)          R1 REGISTER
       0003              EQU   SRAM(3)         8 X 16 FILE
       0004              EQU   SFLR(4)         FLR BITS 0-3 AND MIR BITS 4-7
       0005              EQU   Q(5)            Q BITS 0-3
       0006              EQU   AL(6)           A REGISTER BITS 0-7
       0007              EQU   AU(7)           A REGISTER BITS 8-15
                  *
                  *
                  *
                  *   D FIELD (DESTINATION) EQUATES
                  *
       0000              EQU   R1(00)          R1 REGISTER
       0001              EQU   FLR(01)         FIELD LENGTH REGISTER
       0003              EQU   BLL(03)         BUFFER LENGTH LOWER (0-7)
       0004              EQU   BLU(04)         BUFFER LENGTH UPPER (8-15)
       0005              EQU   CWAL(05)        CURRENT WORD ADDR REG (0-7)
       0006              EQU   CWAU(06)        CURRENT WORD ADDR REG (8-15)
       0007              EQU   CWAUU(07)       CURRENT WORD ADDR REG (16-17)
       0008              EQU   TBR(08)         TAG BUS REGISTER
       0009              EQU   BOR(09)         BUS OUT REGISTER
       000A              EQU   RAM(10)         8 X 16 FILE
       000B              EQU   R1INC(11)       INC R1 REG (STROBE)
       000C              EQU   BLDEC(12)       BUFFER LENGTH REG DEC (STROBE)
       000D              EQU   DMACLR(13)      DMA CLEAR (STROBE)
                  *
                  *
                  *
                  *   F FIELD (FLIP FLOPS) EQUATES
                  *
       0000              EQU   REPLY(00)       REPLY F/F (SET ONLY)
       0001              EQU   REJECT(01)      REJECT F/F (SET ONLY)
       0004              EQU   CUERR(04)       CU ERROR F/F (SET ONLY)
       0005              EQU   NEED(05)        NEED F/F (SET ONLY)
       0006              EQU   XFER(06)        TRANSFER COMPLETE F/F
       0007              EQU   EOC(07)         END OF CYLINDER F/F
       0008              EQU   TAGOUT(08)      TAG OUT F/F
       0009              EQU   RESP(09)        RESPONSE F/F
       000A              EQU   LPFLG(10)       LOOP FLAG F/F
       000B              EQU   TM(11)          TEST MODE F/F
       000C              EQU   WRT(12)         WRITE F/F
       000D              EQU   ADRFLD(13)      ADDRESS FIELD F/F
       000E              EQU   READ(14)        READ F/F
       000F              EQU   ALENB(15)       AUTOLOAD ENABLE F/F
       0010              EQU   BUSY(16)        BUSY F/F
       0011              EQU   DMAGO(17)       DMA GO F/F
       0012              EQU   PP(18)          PROGRAM PROTECT F/F
       0013              EQU   FULWRT(19)      FULL RITE ENABLE F/F
       0014              EQU   R1TOA(20)       R1 TO A ENABLE F/F
       0016              EQU   R1MSB(22)       R1 MSB INHIBIT F/F
```

```
              *
              *
              *
              *    JC FIELD (JUMP CONDITION) EQUATES
              *
    0001              EQU  AQEXEC(01)     DA AND A/Q READ OR WRITE
    0002              EQU  A7(02)         A REG BIT 7
    0003              EQU  ADFLD(03)      ADDRESS FIELD F/F
    0004              EQU  WRTFF(04)      WRITE F/F
    0005              EQU  LFLAG(05)      LOOP FLAG F/F
    0006              EQU  TSTMD(06)      TEST MODE F/F
    0007              EQU  R107(07)       R1 REG BIT 7
    0008              EQU  PROT(08)       X-FER TO PROT MEM OK
    0009              EQU  BIEVEN(09)     BUS IN PARITY
    000A              EQU  CE(10)         CHECKEND
    000C              EQU  NEORCE(12)     NORMAL END OR CHECK END
    000D              EQU  XCOMP(13)      DMA TRANSFER COMPLETE
    000E              EQU  ALARM(14)      ALARM CONDITION
    000F              EQU  EMPTY(15)      FIFO EMPTY
    0010              EQU  CUSEL(16)      CU SELECTED
    0011              EQU  TAGVAL(17)     TAG VALID
    0012              EQU  SECTOR(18)     SECTOR PULSE (INVERTED)
    0013              EQU  INDEX(19)      INDEX PULSE
    0015              EQU  BLEQZ(21)      BUFFER LENGTH EQUAL ZERO
    0016              EQU  R1EQRM(22)     R1 REG EQUALS RAM OUTPUT
    0017              EQU  AUTOL(23)      AUTOLOAD SWITCH
              *
              *
              *
              *    M FIELD (SET OR CLR F/F) EQUATES
              *
    0000              EQU  CLR(0)         CLR SELECTED F/F
    0001              EQU  SET(1)         SET SELECTED F/F
              *
              *
              *
              *    DD FIELD (DIRECT DATA) EQUATES
              *
    0000              EQU  ZERO(0)
    0001              EQU  ONE(1)
    0002              EQU  TWO(2)
    0003              EQU  THRE(3)
    0004              EQU  FOUR(4)
    0005              EQU  FIVE(5)
    0006              EQU  SIX(6)
    0007              EQU  SEVEN(7)
    0008              EQU  EIGHT(8)
    0009              EQU  NINE(9)
    000A              EQU  HEXA($A)
    000B              EQU  HEXB($B)
    000C              EQU  HEXC($C)
    000D              EQU  HEXD($D)
    000E              EQU  HEXE($E)
```

```
0157        000F        EQU  HEXF($F)
0158        0010        EQU  HEX10($10)
0159        0030        EQU  HEX30($30)
0160        003C        EQU  HEX3C($3C)
0161        003F        EQU  HEX3F($3F)
0162        0040        EQU  HEX40($40)
0163        005F        EQU  HEX5F($5F)
0164        007F        EQU  HEX7F($7F)
0165        00F0        EQU  HEXF0($F0)
0166        00FF        EQU  HEXFF($FF)
```

```
0168              ***********************************************************************
0169              *                                                                     *
0170              *                   REVISION 28    11/01/76    RELEASED               *
0171              *                                                                     *
0172              ***********************************************************************
0173              V      NOPP
0173 P0000 0000
0173 P0001 0000
0174                     SEQSD    MIR,FLR,HEXC
0174 P0002 0101
0174 P0003 220C
0175                     SEQSD    MIR,RAM,ZERO
0175 P0004 0201
0175 P0005 3400
0176                     SEQSD    ZERO,DMACLR,ZERO
0176 P0006 0301
0176 P0007 1A00
0177                     SEQFF    CLR,FULWRT
0177 P0008 0402
0177 P0009 2600
0178              CLR2   SEQSD    ZERO,DMACLR,ZERO
0178 P000A 0501
0178 P000B 1A00
0179              IDLE   JMPTFF   AQEXEC,CLR,BUSY,TFORM   IDLE LOOP
0179 P000C 0686
0179 P000D 2009
0180                     JMPF     AUTOL,IDLE
0180 P000E 075C
0180 P000F 0006
0181                     UJPFF    SET,FULWRT,AUTO
0181 P0010 0883
0181 P0011 2797
0182              TFORM  UJPFF    SET,BUSY,HIGH
0182 P0012 0983
0182 P0013 21E0
0183              ILL    JMPTFF   AQEXEC,SET,REJECT,*      A/Q REJECT
0183 P0014 0A87
0183 P0015 020A
0184                     UJP      IDLE
0184 P0016 0880
0184 P0017 0006
0185              WAIT   JMPT     AQEXEC,*
0185 P0018 0C84
0185 P0019 000C
0186                     UJP      IDLE
0186 P001A 0D80
0186 P001B 0006
0187              REP    JMPTFF   AQEXEC,SET,REPLY,*       A/Q REPLY
0187 P001C 0E87
0187 P001D 000E
0188                     UJPFF    CLR,R1TOA,IDLE
0188 P001E 0F82
0188 P001F 2806
```

```
0189              CUOP   SEQSD    AL,BOR,ZERO
0189 P0020 1001
0189 P0021 0200
0190              CHKCU  JMPF     CUSEL,ILL
0190 P0022 1140
0190 P0023 000A
0191                     SEQFF    SET,TAGOUT
0191 P0024 1203
0191 P0025 1000
0192                     SEQSD    MIR,R1,HEXF0
0192 P0026 1301
0192 P0027 20F0
0193                     SEQSD    MIR,FLR,HEXC
0193 P0028 1401
0193 P0029 220C
0194              JA41   JMPF     TAGVAL,JA42
0194 P002A 1544
0194 P002B 001E
0195                     SEQFF    CLR,TAGOUT
0195 P002C 1602
0195 P002D 1000
0196              JA43   JMPFSD   NEORCE,ZERO,R1INC,JA44
0196 P002E 1731
0196 P002F 1622
0197                     SEQSD    BUSIN,R1,ZERO
0197 P0030 1801
0197 P0031 0000
0198                     JMPT     BIEVEN,IOER1
0198 P0032 19A4
0198 P0033 0025
0199                     JMPT     CE,IOER1
0199 P0034 1AA8
0199 P0035 0025
0200              RESP1  JMPTFF   NEORCE,SET,RESP,*
0200 P0036 1BB3
0200 P0037 121B
0201                     SEQFF    SET,R1TOA
0201 P0038 1C03
0201 P0039 2800
0202                     UJPFF    CLR,RESP,REP
0202 P003A 1D82
0202 P003B 120E
0203              JA42   JMPTSD   CE,ZERO,R1INC,CE1
0203 P003C 1EA9
0203 P003D 1624
0204                     JMPF     R1EQRM,JA41
0204 P003E 1F58
0204 P003F 0015
0205                     SEQFF    CLR,TAGOUT
0205 P0040 2002
0205 P0041 1000
0206              ERR1   UJPFF    SET,CUERR,REP
0206 P0042 2183
```

```
0206 P0043 080E
0207                JA44   JMPF       R1EQRM,JA43
0207 P0044 2258
0207 P0045 0017
0208                       UJP        ERR1
0208 P0046 2380
0208 P0047 0021
0209                CE1    SEQFF      CLR.TAGOUT
0209 P0048 2402
0209 P0049 1000
0210                IOER1  UJPFF      SET.CUERR.RESP1
0210 P004A 2583
0210 P004B 081B
0211                CSS    SEQSD      MIR,BOR,ZERO          CURRENT SECTOR STATUS
0211 P004C 2601
0211 P004D 3200
0212                       SEQSD      MIR.TBR.FOUR
0212 P004E 2701
0212 P004F 3004
0213                       UJP        CHKCU
0213 P0050 2880
0213 P0051 0011
0214                PATT   SEQSD      MIR.BOR.ONE           ECC PATTERN
0214 P0052 2901
0214 P0053 3201
0215                       UJP        SETB
0215 P0054 2A80
0215 P0055 002E
0216                DFS    SEQSD      MIR.BOR.ZERO          DRIVE FAULT STATUS
0216 P0056 2B01
0216 P0057 3200
0217                       UJPSD      Q.TBR.CHKCU
0217 P0058 2C81
0217 P0059 B011
0218                COND   SEQSD      MIR.BOR.TWO           ECC CONDITIONS
0218 P005A 2D01
0218 P005B 3202
0219                SETB   SEQSD      MIR.TBR.HEXB
0219 P005C 2E01
0219 P005D 300B
0220                       UJP        CHKCU
0220 P005E 2F80
0220 P005F 0011
0221                PHYS   SEQSD      MIR.BOR.FOUR          PHYSICAL UNIT NO.
0221 P0060 3001
0221 P0061 3204
0222                       UJP        SET8
0222 P0062 3180
0222 P0063 0037
0223                CUSTA  SEQSD      MIR,BOR.TWO           CU STATUS
0223 P0064 3201
0223 P0065 3202
0224                       UJP        SET8
```

```
0224 P0066 3380
0224 P0067 0037
0225                DRST1  SEQSD   MIR,BOR,ONE             DRIVE STATUS ONE
0225 P0068 3401
0225 P0069 3201
0226                       UJP     SET8
0226 P006A 3580
0226 P006B 0037
0227                DRST2  SEQSD   MIR,BOR,EIGHT           DRIVE STATUS TWO
0227 P006C 3601
0227 P006D 3208
0228                SET8   SEQSD   MIR,TBR,EIGHT
0228 P006E 3701
0228 P006F 3008
0229                       UJP     CHKCU
0229 P0070 3860
0229 P0071 0011
0230                CYLST  SEQSD   MIR,FLR,FOUR            CYL ADR STATUS
0230 P0072 3901
0230 P0073 2204
0231                       SEQSD   SRAM,R1,ZERO
0231 P0074 3A01
0231 P0075 6000
0232                       SEQSD   MIR,FLR,THRE
0232 P0076 3B01
0232 P0077 2203
0233                       UJP     REP
0233 P0078 3C80
0233 P0079 000E
0234                SECST  SEQSD   MIR,FLR,TWO             SEC/HEAD STATUS
0234 P007A 3D01
0234 P007B 2202
0235                       SEQSD   SRAM,R1,ZERO
0235 P007C 3E01
0235 P007D 6000
0236                       SEQSD   MIR,FLR,ONE
0236 P007E 3F01
0236 P007F 2201
0237                       UJP     REP
0237 P0080 4080
0237 P0081 000E
0238                SFWAL  UJPSD   AL,CWAL,REP             SET FWA LOWER
0238 P0082 4181
0238 P0083 CA0E
0239                BUFFL  SEQSD   AL,BLL,ZERO             BUFFER LENGTH
0239 P0084 4201
0239 P0085 C600
0240                       UJPSD   ZERO,BLDEC,REP
0240 P0086 4381
0240 P0087 160E
0241                SEL    JMPF    CUSEL,JA91              UNIT SELECT
0241 P0088 4440
0241 P0089 0046
```

```
0242                        UJP     CUOP
0242 P008A 4580
0242 P008B 0010
0243                 JA91   UJPFF   SET,NEED,REP
0243 P008C 4683
0243 P008D 0A0E
0244                 DECHO  JMPF    A7,ILL                   DRIVE ECHO OUTPUT
0244 P008E 4708
0244 P008F 000A
0245                        UJP     CUOP
0245 P0090 4880
0245 P0091 0010
0246                 SEEK   JMPF    CUSEL,ILL                LOAD ADDR. (SEEK)
0246 P0092 4940
0246 P0093 000A
0247                        SEQSD   MIR,FLR,THRE
0247 P0094 4A01
0247 P0095 2203
0248                        SEQSD   AL,RAM,ZERO
0248 P0096 4B01
0248 P0097 D400
0249                        SEQSD   AU,BOR,ZERO
0249 P0098 4C01
0249 P0099 F200
0250                        SEQSD   MIR,FLR,FOUR
0250 P009A 4D01
0250 P009B 2204
0251                        SEQSD   AU,RAM,ZERO
0251 P009C 4E01
0251 P009D F400
0252                        SEQSD   MIR,TBR,SIX
0252 P009E 4F01
0252 P009F 3006
0253                        JMPTFF  AQEXEC,SET,REPLY,*
0253 P00A0 5067
0253 P00A1 0050
0254                 OUT    SEQFF   SET,TAGOUT
0254 P00A2 5103
0254 P00A3 1000
0255                        SEQSD   MIR,R1,HEXF0
0255 P00A4 5201
0255 P00A5 20F0
0256                        SEQSD   MIR,FLR,HEXC
0256 P00A6 5301
0256 P00A7 220C
0257                 JA111  JMPF    TAGVAL,JA112
0257 P00A8 5444
0257 P00A9 005E
0258                        SEQFF   CLR,TAGOUT
0258 P00AA 5502
0258 P00AB 1000
0259                 JA113  JMPFSD  NEORCE,ZERO,R1INC,JA114
0259 P00AC 5631
```

```
0259 P00AD 1663
0260                     JMPT      CE,RESP2
0260 P00AE 57A8
0260 P00AF 0067
0261                     JMPTFF    NEORCE,SET,RESP,*
0261 P00B0 58B3
0261 P00B1 1258
0262                     JMPTFF    LFLAG,CLR,RESP,JA121
0262 P00B2 5996
0262 P00B3 1266
0263                     SEQSD     MIR,TBR,FIVE
0263 P00B4 5A01
0263 P00B5 3005
0264                     SEQSD     MIR,FLR,THRE
0264 P00B6 5B01
0264 P00B7 2203
0265                     SEQSD     SRAM,BOR,ZERO
0265 P00B8 5C01
0265 P00B9 7200
0266                     UJPFF     SET,LPFLG,OUT
0266 P00BA 5D83
0266 P00BB 1451
0267              JA112  JMPTSD    CE,ZERO,R1INC,CE2
0267 P00BC 5EA9
0267 P00BD 1665
0268                     JMPF      R1EQRM,JA111
0268 P00BE 5F58
0268 P00BF 0054
0269                     SEQFF     CLR,TAGOUT
0269 P00C0 6002
0269 P00C1 1000
0270              ERR2   SEQFF     CLR,LPFLG
0270 P00C2 6102
0270 P00C3 1400
0271                     UJPFF     SET,CUERR,IDLE
0271 P00C4 6283
0271 P00C5 0806
0272              JA114  JMPF      R1EQRM,JA113
0272 P00C6 6358
0272 P00C7 0056
0273                     UJP       ERR2
0273 P00C8 6480
0273 P00C9 0061
0274              CE2    UJPFF     CLR,TAGOUT,RESP2
0274 P00CA 6582
0274 P00CB 1067
0275              JA121  UJPFF     CLR,LPFLG,IDLE
0275 P00CC 6682
0275 P00CD 1406
0276              RESP2  JMPTFF    NEORCE,SET,RESP,*
0276 P00CE 67B3
0276 P00CF 1267
0277                     UJPFF     CLR,RESP,ERR2
```

```
0277 P00D0 6882
0277 P00D1 1261
0278                POLL   JMPF      CUSEL,ILL                   INITIATE POLL
0278 P00D2 6940
0278 P00D3 000A
0279                       SEQSD     AL,BOR,ZERO
0279 P00D4 6A01
0279 P00D5 D200
0280                       JMPTFF    AQEXEC,SET,REPLY,*
0280 P00D6 6B87
0280 P00D7 006B
0281                       SEQFF     SET,TAGOUT
0281 P00D8 6C03
0281 P00D9 1000
0282                       SEQSD     MIR,R1,HEXF0
0282 P00DA 6D01
0282 P00DB 20F0
0283                       SEQSD     MIR,FLR,HEXC
0283 P00DC 6E01
0283 P00DD 220C
0284                JA141  JMPF      TAGVAL,JA142
0284 P00DE 6F44
0284 P00DF 0076
0285                       SEQFF     CLR,TAGOUT
0285 P00E0 7002
0285 P00E1 1000
0286                       SEQSD     MIR,R1,HEX5F
0286 P00E2 7101
0286 P00E3 205F
0287                JA143  JMPFSD    NEORCE,ZERO,R1INC,JA144
0287 P00E4 7231
0287 P00E5 167C
0288                       SEQSD     BUSIN,R1.ZERO
0288 P00E6 7301
0288 P00E7 0000
0289                       JMPT      BIEVEN,IOER2
0289 P00E8 74A4
0289 P00E9 007E
0290                       JMPT      CE.IOER2
0290 P00EA 75A6
0290 P00EB 007E
0291                RESP3  JMPTFF    NEORCE,SET,RESP,*
0291 P00EC 76B3
0291 P00ED 1276
0292                       UJPFF     CLR,RESP,IDLE
0292 P00EE 7782
0292 P00EF 1206
0293                JA142  JMPTSD    CE,ZERO,R1INC,CE2
0293 P00F0 78A9
0293 P00F1 1665
0294                       JMPF      R1EQRM,JA141
0294 P00F2 7958
0294 P00F3 006F
```

```
0295                        SEQFF    CLR,TAGOUT
0295 P00F4 7A02
0295 P00F5 1000
0296                 ERR3   UJPFF    SET,CUERR,IDLE
0296 P00F6 7B83
0296 P00F7 0806
0297                 JA144  JMPF      R1EQRM,JA143
0297 P00F8 7C58
0297 P00F9 0072
0298                        UJP      ERR3
0298 P00FA 7D80
0298 P00FB 007B
0299                 IOER2  UJPFF    SET,CUERR,RESP3
0299 P00FC 7E83
0299 P00FD 0876
0300                 FORMAT JMPF     CUSEL,ILL                 FORMAT WRITE
0300 P00FE 7F40
0300 P00FF 000A
0301                        SEQSD    MIR,TBR,HEXE
0301 P0100 8001
0301 P0101 300E
0302                        SEQSD    MIR,BOR,ZERO
0302 P0102 8101
0302 P0103 3200
0303                        JMPTFF   AQEXEC,SET,REPLY,*
0303 P0104 8287
0303 P0105 0082
0304                        SEQFF    SET,TAGOUT
0304 P0106 8303
0304 P0107 1000
0305                        SEQSD    MIR,R1,HEXF0
0305 P0108 8401
0305 P0109 20F0
0306                        SEQSD    MIR,FLR,HEXC
0306 P010A 8501
0306 P010B 220C
0307                 JA171  JMPF     TAGVAL,JA172
0307 P010C 8644
0307 P010D 0096
0308                        SEQFF    CLR,TAGOUT
0308 P010E 8702
0308 P010F 1000
0309                        SEQSD    MIR,FLR,HEXD
0309 P0110 8801
0309 P0111 220D
0310                        SEQSD    MIR,RAM,HEX3C
0310 P0112 8901
0310 P0113 343C
0311                 OUTLP  SEQSD    MIR,R1,ZERO
0311 P0114 8A01
0311 P0115 2000
0312                 INLP   SEQSD    MIR,FLR,HEXC
0312 P0116 8B01
```

```
0312 P0117 220C
0313                          SEQSD     ZERO.R1INC.ZERO
0313 P0118 8C01
0313 P0119 1600
0314                          JMPT      NEORCE.FWEND
0314 P011A 80B0
0314 P011B 009A
0315                          JMPF      R1EQRM.INLP
0315 P011C 8E58
0315 P011D 008B
0316                          SEQSD     MIR.FLR.HEXD
0316 P011E 8F01
0316 P011F 220D
0317                          SEQSD     SRAM.R1.ZERO
0317 P0120 9001
0317 P0121 6000
0318                          SEQSD     MIR.FLR.HEXC
0318 P0122 9101
0318 P0123 220C
0319                          JMPT      R1EQRM.ERR4
0319 P0124 92D8
0319 P0125 0099
0320                          SEQSD     ZERO.R1INC.ZERO
0320 P0126 9301
0320 P0127 1600
0321                          SEQSD     MIR.FLR.HEXD
0321 P0128 9401
0321 P0129 220D
0322                          UJPSD     SR1.RAM.OUTLP
0322 P012A 9581
0322 P012B 548A
0323                   JA172  JMPTSD    CE.ZERO.R1INC.CE2
0323 P012C 96A9
0323 P012D 1665
0324                          JMPF      R1EQRM.JA171
0324 P012E 9758
0324 P012F 0086
0325                          SEQFF     CLR.TAGOUT
0325 P0130 9802
0325 P0131 1000
0326                   ERR4   UJPFF     SET.CUERR.IDLE
0326 P0132 9983
0326 P0133 0806
0327                   FWEND  JMPT      CE.IOER3
0327 P0134 9AA8
0327 P0135 009F
0328                   RESP4  JMPTFF    NEORCE.SET.RESP.*
0328 P0136 9B63
0328 P0137 1298
0329                          JMPTFF    ALARM.CLR.RESP.ERR10
0329 P0138 9CBA
0329 P0139 129E
0330                          SEQFF     SET.XFER
```

```
0330 P013A 9D03
0330 P013B 0C00
0331              ERR10  UJP      IDLE
0331 P013C 9E80
0331 P013D 0006
0332              IOER3  UJPFF    SET,CUERR,RESP4
0332 P013E 9F83
0332 P013F 089B
0333              SEC    SEQSD    MIR,FLR,ONE                 SET SECTOR/HEAD
0333 P0140 A001
0333 P0141 2201
0334                     SEQSD    AL,RAM,ZERO
0334 P0142 A101
0334 P0143 D400
0335                     SEQSD    MIR,FLR,HEXA
0335 P0144 A201
0335 P0145 220A
0336                     SEQSD    AL,RAM,ZERO
0336 P0146 A301
0336 P0147 D400
0337                     SEQSD    MIR,FLR,TWO
0337 P0148 A401
0337 P0149 2202
0338                     SEQSD    AU,RAM,ZERO
0338 P014A A501
0338 P014B F400
0339                     SEQSD    MIR,FLR,HEXB
0339 P014C A601
0339 P014D 220B
0340                     SEQSD    AU,RAM,ZERO
0340 P014E A701
0340 P014F F400
0341                     UJPSD    AU,ROR,CHKCU
0341 P0150 A881
0341 P0151 F211
0342              RCONT  SEQSD    MIR,FLR,HEXC                READ RECOVERY CONTROL
0342 P0152 A901
0342 P0153 220C
0343                     SEQSD    AU,R1,ZERO
0343 P0154 AA01
0343 P0155 E000
0344                     JMPF     R1EQRM,JA201
0344 P0156 AB58
0344 P0157 00BA
0345                     SEQSD    MIR,THR,HEXC
0345 P0158 AC01
0345 P0159 300C
0346                     SEQSD    AL,ROR,ZERO
0346 P015A AD01
0346 P015B D200
0347                     JMPF     A7,CHKCU
0347 P015C AE08
0347 P015D 0011
```

```
0348                          JMPF      CUSEL,ILL                 RETURN TO ZERO SEEK
0348 P015E AF40
0348 P015F 000A
0349                          JMPTFF    AQEXEC,SET,REPLY,*
0349 P0160 B087
0349 P0161 0080
0350                          SEQSD     MIR,FLR,TWO
0350 P0162 B101
0350 P0163 2202
0351                          SEQSD     MIR,RAM,ZERO
0351 P0164 B201
0351 P0165 3400
0352                          SEQSD     MIR,FLR,THRE
0352 P0166 B301
0352 P0167 2203
0353                          SEQSD     MIR,RAM,ZERO
0353 P0168 B401
0353 P0169 3400
0354                          SEQSD     MIR,FLR,FOUR
0354 P016A B501
0354 P016B 2204
0355                          SEQSD     MIR,RAM,ZERO
0355 P016C B601
0355 P016D 3400
0356                          SEQSD     MIR,FLR,ONE
0356 P016E B701
0356 P016F 2201
0357                          SEQSD     MIR,RAM,ZERO
0357 P0170 B801
0357 P0171 3400
0358                          UJPFF     SET,LPFLG,OUT
0358 P0172 B983
0358 P0173 1451
0359                  JA201   SEQSD     MIR,THR,HEXD
0359 P0174 BA01
0359 P0175 300D
0360                          UJPSD     AU,HOR,CHKCD
0360 P0176 B881
0360 P0177 F211
0361                  WRITE   JMPTFF    BLEQZ,CLR,ADRFLD,ERCHK
0361 P0178 BCD6
0361 P0179 1B93
0362                  NTM     JMPF      CUSEL,ILL
0362 P017A BD40
0362 P017B 000A
0363                  JA231   JMPF      A7,JA232
0363 P017C BF0B
0363 P017D 00C0
0364                          SEQFF     SET,ADRFLD
0364 P017E BF03
0364 P017F 1A00
0365                  JA232   JMPTFF    AQEXEC,SET,REPLY,*
0365 P0180 C087
```

```
0365 P0181 00C0
0366                     JMPT     BLEQZ,CHKBR
0366 P0182 C1D4
0366 P0183 0195
0367              NOTTM  JMPTFF   ADFLD,SET,WRT,ADR
0367 P0184 C28F
0367 P0185 196E
0368              ADFIND SEQSD    MIR,FLR,ONE
0368 P0186 C301
0368 P0187 2201
0369                     SEQSD    MIR,RAM,HEXFF
0369 P0188 C401
0369 P0189 34FF
0370                     SEQSD    MIR,FLR,TWO
0370 P018A C501
0370 P018B 2202
0371                     SEQSD    MIR,RAM,HEXFF
0371 P018C C601
0371 P018D 34FF
0372              ADCOM  SEQFF    SET,ADRFLD
0372 P018E C703
0372 P018F 1A00
0373                     SEQSD    MIR,FLR,HEXF
0373 P0190 C801
0373 P0191 220E
0374                     SEQSD    MIR,RAM,HEX7F
0374 P0192 C901
0374 P0193 347F
0375              REPT   SEQSD    MIR,THR,NINE
0375 P0194 CA01
0375 P0195 3009
0376                     SEQSD    MIR,FLR,EIGHT
0376 P0196 CB01
0376 P0197 2208
0377                     SEQSD    MIR,BOR,NINE
0377 P0198 CC01
0377 P0199 3209
0378                     JMPT     SECTOR,*
0378 P019A CDC8
0378 P019B 00CD
0379              JA251  JMPFFF   TAGVAL,SET,TAGOUT,JA252
0379 P019C CE47
0379 P019D 10E4
0380                     SEQFF    CLR,TAGOUT
0380 P019E CF02
0380 P019F 1000
0381                     JMPF     NEORCE,*
0381 P01A0 D030
0381 P01A1 00D0
0382                     JMPT     CE,IOER4
0382 P01A2 D1A8
0382 P01A3 00E6
0383              RESP5  JMPTFF   NEORCE,SET,RESP,*
```

```
0383 P01A4 D230
0383 P01A5 00D2
0384                      JMPT    CE,IOER4
0384 P01A6 D3A8
0384 P01A7 00E6
0385           RESP5      JMPTFF  NEORCE,SET,RESP,*
0385 P01A8 D4B3
0385 P01A9 12D4
0386                      JMPTFF  ALARM,CLR,RESP,ERR7
0386 P01AA D5BA
0386 P01AB 12E8
0387                      SEQSD   MIR,FLR,TWO
0387 P01AC D601
0387 P01AD 2202
0388                      SEQSD   SRAM,R1,ZERO
0388 P01AE D701
0388 P01AF 6000
0389                      SEQSD   MIR,FLR,HEXB
0389 P01B0 D801
0389 P01B1 220B
0390                      JMPFFF  R1EQRM,SET,R1MSB,NOCOM
0390 P01B2 D95B
0390 P01B3 2CEC
0391                      SEQSD   MIR,FLR,ONE
0391 P01B4 DA01
0391 P01B5 2201
0392                      SEQSD   SRAM,R1,ZERO
0392 P01B6 DB01
0392 P01B7 6000
0393                      SEQSD   MIR,FLR,HEXA
0393 P01B8 DC01
0393 P01B9 220A
0394                      JMPFFF  R1EQRM,CLR,R1MSB,NOCOM
0394 P01BA DD5A
0394 P01BB 2CEC
0395                      SEQSD   MIR,FLR,ZERO
0395 P01BC DE01
0395 P01BD 2200
0396                      SEQSD   SRAM,FLR,ZERO
0396 P01BE DF01
0396 P01BF 6200
0397                      SEQSD   SFLR,BOR,HEX40
0397 P01C0 E001
0397 P01C1 9240
0398                      JMPTFF  WRTFF,CLR,ADRFLD,WRET
0398 P01C2 E192
0398 P01C3 1B24
0399                      UJP     RRET
0399 P01C4 E280
0399 P01C5 0126
0400           JA252      JMPT    ALARM,ERR6
0400 P01C6 E3B6
0400 P01C7 00E7
```

```
0401                     JMPF     CE,JA251
0401 P01C8 E428
0401 P01C9 00D0
0402                     SEQFF    CLR,TAGOUT
0402 P01CA E502
0402 P01CB 1000
0403              IOER4  UJPFF    SET,CUERR,RESP5
0403 P01CC E683
0403 P01CD 08D4
0404              ERR6   SEQFF    CLR,TAGOUT
0404 P01CE E702
0404 P01CF 1000
0405              ERR7   SEQFF    CLR,WRT
0405 P01D0 E802
0405 P01D1 1800
0406                     SEQFF    CLR,READ
0406 P01D2 E902
0406 P01D3 1C00
0407                     SEQSD    ZERO,DMACLR,ZERO
0407 P01D4 EA01
0407 P01D5 1A00
0408                     UJPFF    CLR,PP,CLP2
0408 P01D6 E882
0408 P01D7 2405
0409              NOCOM  SEQFF    CLR,R1MSB
0409 P01D8 EC02
0409 P01D9 2C00
0410                     SEQSD    MIR,FLR,HEXE
0410 P01DA ED01
0410 P01DB 220E
0411                     SEQSD    SRAM,R1,ZERO
0411 P01DC EE01
0411 P01DD 6000
0412                     SEQSD    MIR,FLR,HEXC
0412 P01DE EF01
0412 P01DF 220C
0413                     NOPP
0413 P01E0 F000
0413 P01E1 0000
0414                     JMPT     R1EQRM,ERR8
0414 P01E2 F1D8
0414 P01E3 00FA
0415                     SEQSD    ZERO,R1INC,ZERO
0415 P01E4 F201
0415 P01E5 1600
0416                     SEQSD    MIR,FLR,HEXF
0416 P01E6 F301
0416 P01E7 220E
0417                     UJPSD    SR1,RAM,REPT
0417 P01E8 F481
0417 P01E9 54CC
0418              SFWAU  JMPT     A7,LDRAM
0418 P01EA F588
```

```
0418 P01EB 220F
0419                         UJPSD     AL,RAM,REP
0419 P01EC F681
0419 P01ED D40E
0420                 LDRAM   SEQSD     AL,FLR
0420 P01EE F701
0420 P01EF C200
0421                         UJPSD     AU,RAM,REP
0421 P01F0 F881
0421 P01F1 F40E
0422                 ERR8    SEQFF     CLR,WRT
0422 P01F2 F902
0422 P01F3 1800
0423                 ERR9    SEQFF     CLR,READ
0423 P01F4 FA02
0423 P01F5 1C00
0424                         SEQFF     CLR,PP
0424 P01F6 FB02
0424 P01F7 2400
0425                         SEQSD     ZERO,DMACLR,ZERO
0425 P01F8 FC01
0425 P01F9 1A00
0426                         SEQFF     CLR,TM
0426 P01FA FD02
0426 P01FB 1600
0427                         UJPFF     SET,CUERR,CLR2
0427 P01FC FE83
0427 P01FD 0805
0428                         NOPP
0428 P01FE FF00
0428 P01FF 0000
0429                 TMW     SEQSD     MIR,ALU,ZERO                 TEST MODE WRITE
0429 P0200 0001
0429 P0201 2800
0430                         SEQSD     MIR,ALL,HEX3F
0430 P0202 0101
0430 P0203 263F
0431                         SEQFF     SET,TM
0431 P0204 0203
0431 P0205 1600
0432                         SEQFF     SET,WRT
0432 P0206 0303
0432 P0207 1800
0433                         SEQSD     MIR,R1,ZERO
0433 P0208 0401
0433 P0209 2000
0434                         SEQSD     MIR,FLR,HEXC
0434 P020A 0501
0434 P020B 220C
0435                 JA301   JMPTSD    XCOMP,ZERO,R1INC,ALDON
0435 P020C 0685
0435 P020D 170B
0436                         NOPP
```

```
0436 P020E 0700
0436 P020F 0000
0437                          NOPP
0437 P0210 0800
0437 P0211 0000
0438                          JMPF      R1EQRM,JA301
0438 P0212 0958
0438 P0213 0106
0439                          UJP       ERRH
0439 P0214 0A80
0439 P0215 00F9
0440                  ALDON   SEQFF     CLR,WRT
0440 P0216 0B02
0440 P0217 1800
0441                  TCOMP   SEQFF     CLR,PP
0441 P0218 0C02
0441 P0219 2400
0442                          SEQFF     CLR.READ
0442 P021A 0D02
0442 P021B 1C00
0443                          UJPFF     SET.XFER.IDLE
0443 P021C 0E83
0443 P021D 0C06
0444                  REED    JMPTFF    TSTMD,CLR,ADRFLD.JA311
0444 P021E 0F9A
0444 P021F 1B11
0445                          JMPF      CUSEL.ILL
0445 P0220 1040
0445 P0221 000A
0446                  JA311   JMPF      PROT.JA312
0446 P0222 1120
0446 P0223 0113
0447                          SEQFF     SET,PP
0447 P0224 1203
0447 P0225 2400
0448                  JA312   JMPF      A7,JA321
0448 P0226 1308
0448 P0227 0115
0449                          SEQFF     SET,ADRFLD
0449 P0228 1403
0449 P0229 1A00
0450                  JA321   JMPTFF    AQEXEC,SET,REPLY,*
0450 P022A 1587
0450 P022B 0115
0451                          JMPT      TSTMD.TMR
0451 P022C 169B
0451 P022D 0119
0452                          JMPTFF    ADFLD.SET,READ,ADR
0452 P022E 178F
0452 P022F 1D6E
0453                          UJP       ADFIND
0453 P0230 18B0
0453 P0231 00C3
```

```
0454                TMR    SEQSD   MIR,BLU,ZERO                  TEST MODE READ
0454 P0232 1901
0454 P0233 2800
0455                       SEQSD   MIR,BLL,HEX3F
0455 P0234 1A01
0455 P0235 263F
0456                       SEQFF   SET,READ
0456 P0236 1B03
0456 P0237 1C00
0457                       SEQSD   MIR,R1,ZERO
0457 P0238 1C01
0457 P0239 2000
0458                       SEQSD   MIR,FLR,HEXC
0458 P023A 1001
0458 P023B 220C
0459                JA331  JMPTSD  XCOMP,ZERO,R1INC,JA332
0459 P023C 1EB5
0459 P023D 1723
0460                       NOPP
0460 P023E 1F00
0460 P023F 0000
0461                       NOPP
0461 P0240 2000
0461 P0241 0000
0462                       JMPF    R1EQRM,JA331
0462 P0242 2158
0462 P0243 011E
0463                       UJP     ERR9
0463 P0244 2280
0463 P0245 00FA
0464                JA332  UJPFF   CLR,TM,TCOMP
0464 P0246 23B2
0464 P0247 170C
0465                WRET   SEQSD   MIR,TBR,HEXA
0465 P0248 2401
0465 P0249 300A
0466                       UJP     ROWCOM
0466 P024A 2580
0466 P024B 0127
0467                RRET   SEQSD   MIR,TBR,NINE
0467 P024C 2601
0467 P024D 3009
0468                ROWCOM JMPFFF  TAGVAL,SET,TAGOUT,JA341
0468 P024E 2747
0468 P024F 1140
0469                       SEQFF   CLR,TAGOUT
0469 P0250 2802
0469 P0251 1000
0470                       SEQFF   SET,DMAGO
0470 P0252 2903
0470 P0253 2200
0471                       JMPF    NEORCE,*
0471 P0254 2A30
```

```
0471 P0255 012A
0472                        JMPT      CE,IOERS
0472 P0256 28A8
0472 P0257 0142
0473                RESP6   JMPTFF    NEORCE,SET,RESP,*
0473 P0258 2CB3
0473 P0259 132C
0474                        SEQFF     CLR,RESP
0474 P025A 2D02
0474 P025B 1200
0475                        JMPTFF    ALARM,CLR,DMAGO,ERRX
0475 P025C 2EBA
0475 P025D 2337
0476                        JMPT      WRTFF,JA351
0476 P025E 2F90
0476 P025F 0143
0477                        JMPT      RLEQZ,DELY
0477 P0260 30D4
0477 P0261 018C
0478                GOON    SEQSD     MIR,FLR,ONE
0478 P0262 3101
0478 P0263 2201
0479                        SEQSD     SRAM,R1,ZERO
0479 P0264 3201
0479 P0265 6000
0480                        JMPT      R107,CHKHD
0480 P0266 339C
0480 P0267 0146
0481                        SEQSD     ZERO,R1INC,ZERO
0481 P0268 3401
0481 P0269 1600
0482                        SEQSD     MIR,FLR,HEXA
0482 P026A 3501
0482 P026B 220A
0483                        UJPSD     SR1,RAM,ADCOM
0483 P026C 3681
0483 P026D 54C7
0484                ERRX    JMPT      WRTFF,JA361
0484 P026E 3790
0484 P026F 013E
0485                        SEQSD     MIR,R1,HEX10
0485 P0270 3801
0485 P0271 2010
0486                JA431   NOPP
0486 P0272 3900
0486 P0273 0000
0487                        NOPP
0487 P0274 3A00
0487 P0275 0000
0488                        JMPFSD    R107,ZERO,R1INC,JA431
0488 P0276 3B1D
0488 P0277 1739
0489                        JMPT      XCOMP,COMPL
```

```
0489 P0278 3CB4
0489 P0279 013F
0490                    UJP      ERR7
0490 P027A 3D80
0490 P027B 00E7
0491             JA361  JMPF     EMPTY,ERR7
0491 P027C 3E3C
0491 P027D 00E7
0492             COMPL  UJPFF    SET,XFER,ERR7
0492 P027E 3F83
0492 P027F 0CF7
0493             JA341  JMPF     CE,ROWCOM
0493 P0280 402B
0493 P0281 0127
0494                    SEQFF    CLR,TAGOUT
0494 P0282 4102
0494 P0283 1000
0495             IOER5  UJPFF    SET,CUERR,RESP6
0495 P0284 42B3
0495 P0285 092C
0496             JA351  JMPF     EMPTY,GOON
0496 P0286 433C
0496 P0287 0131
0497             DONE   JMPF     XCOMP,*
0497 P0288 4434
0497 P0289 0144
0498                    UJPFF    CLR,FULWRT,ALUON
0498 P028A 4582
0498 P028B 270B
0499             CHKHD  SEQSD    MIR,FLR,TWO
0499 P028C 4601
0499 P028D 2202
0500                    SEQSD    MIR,R1,FOUR
0500 P028E 4701
0500 P028F 2004
0501                    JMPT     P1FORM,JA371
0501 P0290 4BDE
0501 P0291 015F
0502                    SEQSD    SRAM,R1,ZERO
0502 P0292 4901
0502 P0293 6000
0503                    SEQSD    ZERO,R1INC,ZERO
0503 P0294 4A01
0503 P0295 1600
0504                    SEQSD    SR1,HOR,ZERO
0504 P0296 4B01
0504 P0297 5200
0505                    SEQSD    MIR,FLR,HEXB
0505 P0298 4C01
0505 P0299 220B
0506                    SEQSD    SR1,RAM,ZERO
0506 P029A 4D01
0506 P029B 5400
```

```
0507                      SEQSD    MIR,FLR,TWO
0507 P029C 4E01
0507 P029D 2202
0508                      SEQSD    SR1,RAM,ZERO
0508 P029E 4F01
0508 P029F 5400
0509                      SEQSD    MIR,THR,SEVEN
0509 P02A0 5001
0509 P02A1 3007
0510                      SEQFF    SET,TAGOUT
0510 P02A2 5103
0510 P02A3 1000
0511                      SEQSD    MIR,R1,HEXF0
0511 P02A4 5201
0511 P02A5 20F0
0512                      SEQSD    MIR,FLR,HEXC
0512 P02A6 5301
0512 P02A7 220C
0513               JA381  JMPF     TAGVAL,JA382
0513 P02A8 5444
0513 P02A9 0167
0514                      SEQFF    CLR,TAGOUT
0514 P02AA 5502
0514 P02AB 1000
0515               JA383  JMPFSD   NFORCE,ZERO,R1INC,JA384
0515 P02AC 5631
0515 P02AD 176A
0516                      JMPT     CF,IOERR
0516 P02AE 57A8
0516 P02AF 016D
0517               RESP7  JMPTFF   NFORCE,SET,RESP,*
0517 P02B0 58B3
0517 P02B1 1358
0518                      JMPTFF   ALARM,CLR,RESP,ERR7
0518 P02B2 59BA
0518 P02B3 12F7
0519                      JMPT     ADFLD,ADR
0519 P02B4 5A8C
0519 P02B5 016E
0520                      SEQSD    MIR,FLR,HEXA
0520 P02B6 5B01
0520 P02B7 220A
0521                      SEQSD    MIR,RAM,ZERO
0521 P02B8 5C01
0521 P02B9 3400
0522                      UJP      ADFIND
0522 P02BA 5D80
0522 P02BB 00C3
0523               JA371  SEQSD    MIR,R1,ZERO
0523 P02BC 5E01
0523 P02BD 2000
0524                      SEQSD    MIR,FLR,HEXC
0524 P02BE 5F01
```

```
0524 P02BF 220C
0525                JA372  SEQSD    ZERO,R1INC,ZERO
0525 P02C0 6001
0525 P02C1 1600
0526                       NOPP
0526 P02C2 6100
0526 P02C3 0000
0527                       JMPF     R1EQRM,JA372
0527 P02C4 6258
0527 P02C5 0160
0528                       SEQFF    CLR,PP
0528 P02C6 6302
0528 P02C7 2400
0529                       SEQFF    CLR,READ
0529 P02C8 6402
0529 P02C9 1C00
0530                       SEQFF    CLR,WRT
0530 P02CA 6502
0530 P02CB 1800
0531                       UJPFF    SET,EOC,IDLE
0531 P02CC 6683
0531 P02CD 0E06
0532                JA382  JMPTSD   CE,ZERO,R1INC,CE3
0532 P02CE 67A9
0532 P02CF 176C
0533                       JMPF     R1EQRM,JA381
0533 P02D0 6858
0533 P02D1 0154
0534                       UJPFF    CLR,TAGOUT,ERRB
0534 P02D2 6982
0534 P02D3 10F9
0535                JA384  JMPF     R1EQRM,JA383
0535 P02D4 6A58
0535 P02D5 0156
0536                       UJP      ERRB
0536 P02D6 6880
0536 P02D7 00F9
0537                CE3    SEQFF    CLR,TAGOUT
0537 P02D8 6C02
0537 P02D9 1600
0538                IOER6  UJPFF    SET,CUERR,RESP7
0538 P02DA 6083
0538 P02DB 0958
0539                ADR    SEQSD    MIR,FLR,ONE                 READ OR WRITE ADR FIELD
0539 P02DC 6E01
0539 P02DD 2201
0540                       SEQSD    MIR,RAM,ZERO
0540 P02DE 6F01
0540 P02DF 3400
0541                       JMPF     INDEX,*
0541 P02E0 704C
0541 P02E1 0170
0542                ADREP  SEQSD    MIR,BOR,NINE
```

```
0542 P02E2 7101
0542 P02E3 3209
0543                        SEQSD    MIR,FLR,NINE
0543 P02E4 7201
0543 P02E5 2209
0544                        JMPT     WRTFF,JA401
0544 P02E6 7390
0544 P02E7 0185
0545                        SEQSD    MIR,THR,NINE
0545 P02E8 7401
0545 P02E9 3009
0546                COMAD   JMPFFF   TAGVAL,SET,TAGOUT,JA411
0546 P02EA 7547
0546 P02EB 1187
0547                        SEQFF    CLR,TAGOUT
0547 P02EC 7602
0547 P02ED 1000
0548                        SEQFF    SET,DMAGO
0548 P02EE 7703
0548 P02EF 2200
0549                        JMPF     NEORCE,*
0549 P02F0 7830
0549 P02F1 017B
0550                        JMPT     CE,IOER7
0550 P02F2 79A8
0550 P02F3 0189
0551                RESP8   JMPTFF   NEORCE,SET,RESP,*
0551 P02F4 7AB3
0551 P02F5 137A
0552                        SEQFF    CLR,RESP
0552 P02F6 7B02
0552 P02F7 1200
0553                        JMPTFF   ALARM,CLR,DMAGO,ERRX
0553 P02F8 7C9A
0553 P02F9 2337
0554                        JMPT     WRTFF,JA421
0554 P02FA 7D90
0554 P02FB 018A
0555                        JMPT     GLEQZ,DELY
0555 P02FC 7ED4
0555 P02FD 018C
0556                MORE    JMPT     SECTOR,*
0556 P02FE 7FC8
0556 P02FF 017F
0557                        JMPT     INDEX,CHKHD
0557 P0300 80CC
0557 P0301 0146
0558                        SEQSD    MIR,FLR,ONE
0558 P0302 8101
0558 P0303 2201
0559                        SEQSD    SRAM,R1,ZERO
0559 P0304 8201
0559 P0305 6000
```

```
0560                   SEQSD     ZERO,R1INC,ZERO
0560 P0306 8301
0560 P0307 1600
0561                   UJPSD     SR1,RAM,ADRFP
0561 P0308 8481
0561 P0309 5571
0562            JA401  SEQSD     MIR,THR,HEXA
0562 P030A 8501
0562 P030B 300A
0563                   UJP       COMAD
0563 P030C 8680
0563 P030D 0175
0564            JA411  JMPF      CE,COMAD
0564 P030E 8728
0564 P030F 0175
0565                   SEQFF     CLR,TAGOUT
0565 P0310 8802
0565 P0311 1000
0566            IOER7  UJPFF     SET,CUERR,RESPB
0566 P0312 8983
0566 P0313 097A
0567            JA421  JMPF      EMPTY,MORE
0567 P0314 8A3C
0567 P0315 017F
0568                   UJP       DONE
0568 P0316 8880
0568 P0317 0144
0569            DELY   SEQSD     MIR,R1,HEX10
0569 P0318 8C01
0569 P0319 2010
0570                   JMPFSD    R107,ZERO,R1INC,*
0570 P031A 8D1D
0570 P031B 17B0
0571                   UJP       DONE
0571 P031C 8E80
0571 P031D 0144
0572            SPEC   SEQSD     MIR,FLR,HEXF
0572 P031E 8F01
0572 P031F 220F
0573                   SEQSD     SRAM,FLR,ZERO
0573 P0320 9001
0573 P0321 6200
0574                   SEQSD     SRAM,R1,ZERO
0574 P0322 9101
0574 P0323 6000
0575                   UJPFF     SET,R1TOA,FEP
0575 P0324 9283
0575 P0325 280E
0576            RRCHK  JMPT      EMPTY,JA231
0576 P0326 93BC
0576 P0327 00BE
0577                   UJP       NTM
0577 P0328 9480
```

```
0577 P0329 008D
0578              CHKBR  JMPT     EMPTY,TMW
0578 P032A 95BC
0578 P032B 0100
0579                     UJP      NOTTM
0579 P032C 9680
0579 P032D 00C2
0580              AUTO   SEQFF    CLR,DMAGO                   *****  AUTOLOAD  *****
0580 P032E 9702
0580 P032F 2200
0581                     SEQSD    ZERO,DMACLR,ZERO
0581 P0330 9801
0581 P0331 1A00
0582                     SEQFF    CLR,WRT
0582 P0332 9902
0582 P0333 1800
0583                     SEQFF    CLR,READ
0583 P0334 9A02
0583 P0335 1C00
0584              AUTSEL JMPFSD   CUSEL,ZERO,DMACLR,JA441
0584 P0336 9B41
0584 P0337 1BD0
0585                     JMPTFF   AUTOL,SET,BUSY,*
0585 P0338 9CDF
0585 P0339 219C
0586                     SEQSD    MIR,BOR,HEX7F
0586 P033A 9D01
0586 P033B 327F
0587                     UJPFF    SET,LPFLG,FLTCL
0587 P033C 9E83
0587 P033D 15AA
0588              DRSEL  SEQFF    CLR,LPFLG
0588 P033E 9F02
0588 P033F 1400
0589                     SEQFF    SET,ALENB
0589 P0340 A003
0589 P0341 1E00
0590                     SEQSD    ZERO,BOR,ZERO
0590 P0342 A101
0590 P0343 1200
0591                     SEQFF    CLR,ALENB
0591 P0344 A202
0591 P0345 1E00
0592                     SEQSD    MIR,THR,THRE
0592 P0346 A301
0592 P0347 3003
0593              JA451  JMPFFF   TAGVAL,SET,TAGOUT,JA452
0593 P0348 A447
0593 P0349 11D1
0594                     JMPFFF   NEORCE,CLR,TAGOUT,*
0594 P034A A532
0594 P034B 11A5
0595                     JMPT     CE,RESP2
```

```
0595 P034C A6A8
0595 P034D 0067
0596                    JMPTFF    NEORCE,SET,RESP,*
0596 P034E A7B3
0596 P034F 13A7
0597                    SEQFF     CLR,RESP
0597 P0350 A802
0597 P0351 1200
0598                    SEQSD     MIR,BOR,HEXFF
0598 P0352 A901
0598 P0353 32FF
0599            FLTCL   SEQSD     MIR,TBR,HEXC
0599 P0354 AA01
0599 P0355 300C
0600            JA461   JMPFFF    TAGVAL,SET,TAGOUT,JA462
0600 P0356 AB47
0600 P0357 11D3
0601                    JMPFFF    NEORCE,CLR,TAGOUT,*
0601 P0358 AC32
0601 P0359 11AC
0602                    JMPT      CE,RESP2
0602 P035A ADA8
0602 P035B 0067
0603                    JMPTFF    NEORCE,SET,RESP,*
0603 P035C AEB3
0603 P035D 13AE
0604                    JMPTFF    LFLAG,CLR,RESP,DRSEL
0604 P035E AF96
0604 P035F 139F
0605            AGAIN   SEQSD     MIR,TBR,EIGHT
0605 P0360 B001
0605 P0361 3008
0606                    SEQSD     MIR,BOR,EIGHT
0606 P0362 B101
0606 P0363 3208
0607            JA471   JMPFFF    TAGVAL,SET,TAGOUT,JA472
0607 P0364 B247
0607 P0365 1105
0608                    JMPFFF    NEORCE,CLR,TAGOUT,*
0608 P0366 B332
0608 P0367 11B3
0609                    SEQSD     BUSIN,R1,ZERO
0609 P0368 B401
0609 P0369 0000
0610                    JMPT      BIEVEN,RESP2
0610 P036A B5A4
0610 P036B 0067
0611                    JMPT      CE,RESP2
0611 P036C B6A8
0611 P036D 0067
0612                    JMPTFF    NEORCE,SET,RESP,*
0612 P036E B7B3
0612 P036F 13B7
```

```
0613                      SEQFF    CLR,RESP
0613 P0370 B802
0613 P0371 1200
0614                      SEQSD    MIR,FLR,NINE
0614 P0372 B901
0614 P0373 2209
0615                      SEQSD    MIR,RAM,HEX30
0615 P0374 BA01
0615 P0375 3430
0616                      NOPP
0616 P0376 BB00
0616 P0377 0000
0617                      JMPF     RIEQRM,AGAIN
0617 P0378 BC58
0617 P0379 01B0
0618                      SEQSD    MIR,TBR,SEVEN
0618 P037A BD01
0618 P037B 3007
0619                      SEQSD    MIR,BOR,ZERO
0619 P037C BE01
0619 P037D 3200
0620               JA491  JMPFFF   TAGVAL,SET,TAGOUT,JA492
0620 P037E BF47
0620 P037F 11D7
0621                      JMPFFF   NEORCE,CLR,TAGOUT,*
0621 P0380 C032
0621 P0381 11C0
0622                      JMPT     CE,RESP2
0622 P0382 C1A8
0622 P0383 0067
0623                      JMPTFF   NEORCE,SET,RESP,*
0623 P0384 C2B3
0623 P0385 13C2
0624                      SEQFF    CLR,RESP
0624 P0386 C302
0624 P0387 1200
0625                      SEQSD    MIR,FLR,HEXA
0625 P0388 C401
0625 P0389 220A
0626                      SEQSD    MIR,RAM,ZERO
0626 P038A C501
0626 P038B 3400
0627                      SEQSD    MIR,FLR,HEXB
0627 P038C C601
0627 P038D 220B
0628                      SEQSD    MIR,RAM,ZERO
0628 P038E C701
0628 P038F 3400
0629                      SEQSD    MIR,CWAUU,ZERO
0629 P0390 C801
0629 P0391 2E00
0630                      SEQSD    MIR,CWAU,ZERO
0630 P0392 C901
```

```
0630 P0393 2C00
0631                      SEQSD    MIR,CWAL,ZERO
0631 P0394 CA01
0631 P0395 2A00
0632                      SEQSD    MIR,BLU,FIVE
0632 P0396 CB01
0632 P0397 2805
0633                      SEQSD    MIR,BLL,HEXFF
0633 P0398 CC01
0633 P0399 26FF
0634                      SEQFF    CLR,XFER
0634 P039A CD02
0634 P039B 0C00
0635                      SEQFF    SET,PP
0635 P039C CE03
0635 P039D 2400
0636                      UJPFF    SET,READ,ADFIND
0636 P039E CF83
0636 P039F 1CC5
0637               JA441  UJPFF    SET,NEED,AUTSEL
0637 P03A0 D083
0637 P03A1 0898
0638               JA452  JMPT     CE,CE2
0638 P03A2 D1A8
0638 P03A3 0065
0639                      UJP      JA451
0639 P03A4 D280
0639 P03A5 01A4
0640               JA462  JMPT     CE,CE2
0640 P03A6 D3A8
0640 P03A7 0065
0641                      UJP      JA461
0641 P03A8 D480
0641 P03A9 01AB
0642               JA472  JMPT     CE.CE2
0642 P03AA D5A6
0642 P03AB 0065
0643                      UJP      JA471
0643 P03AC D680
0643 P03AD 01B2
0644               JA492  JMPT     CE.CE2
0644 P03AE D7A8
0644 P03AF 0065
0645                      UJP      JA491
0645 P03B0 D880
0645 P03B1 01BF
0646                      NOPP
0646 P03B2 D900
0646 P03B3 0000
0647                      NOPP
0647 P03B4 DA00
0647 P03B5 0000
0648                      NOPP
```

```
0648 P03B6 DB00
0648 P03B7 0000
0649                     NOPP
0649 P03B8 DC00
0649 P03B9 0000
0650                     NOPP
0650 P03BA DD00
0650 P03BB 0000
0651                     NOPP
0651 P03BC DE00
0651 P03BD 0000
0652                     NOPP
0652 P03BE DF00
0652 P03BF 0000
0653              *************************************************************
0654              *                                                           *
0655              *                   TRANSFORM JUMP TABLE                    *
0656              *                                                           *
0657              *************************************************************
0658              *
0659              HIGH   UJP      SPEC                      SPECIAL FOR TESTING
0659 P03C0 E080
0659 P03C1 018F
0660                     UJP      PHYS                      PHYSICAL UNIT NUMBER
0660 P03C2 E180
0660 P03C3 0030
0661                     UJPFF    SET•R1TOA•REP             POLL STATUS
0661 P03C4 E283
0661 P03C5 280E
0662                     UJPFF    SET•R1TOA•REP             SEL ACK STATUS
0662 P03C6 E383
0662 P03C7 280E
0663                     UJPFF    SET•R1TOA•REP             DRIVE ECHO INPUT
0663 P03C8 E483
0663 P03C9 280E
0664                     UJP      CYLST                     CYL ADDR STATUS
0664 P03CA E580
0664 P03CB 0039
0665                     UJP      CSS                       CURRENT SECTOR STATUS
0665 P03CC E680
0665 P03CD 0026
0666                     UJP      SECST                     SECTOR AND HEAD STATUS
0666 P03CE E780
0666 P03CF 003D
0667                     UJPFF    CLR•BUSY•WAIT             DA STATUS
0667 P03D0 E882
0667 P03D1 200C
0668                     UJP      DRST2                     DRIVE STATUS 2
0668 P03D2 E980
0668 P03D3 0036
0669                     UJP      PATT                      ECC PATTERN
0669 P03D4 EA80
0669 P03D5 0029
```

```
0670                     UJP          COND                         ECC CONDITIONS
0670 P03D6 EB80
0670 P03D7 002D
0671                     UJP          DFS                          DRIVE FAULT STATUS
0671 P03D8 EC80
0671 P03D9 002B
0672                     UJP          CUSTA                        CU STATUS
0672 P03DA ED80
0672 P03DB 0032
0673                     UJP          DRST1                        DRIVE STATUS 1
0673 P03DC EE80
0673 P03DD 0034
0674                     UJPFF        SET,R1TOA,REP                CU ECHO INPUT
0674 P03DE EF83
0674 P03DF 280E
0675                     UJPSD        AU,BLU,BUFFL                 SET BUFFER LENGTH
0675 P03E0 F081
0675 P03E1 E842
0676                     UJPSD        Q,TBR,CUOP                   DRIVE REQUEST
0676 P03E2 F181
0676 P03E3 B010
0677                     UJPSD        Q,TBR,POLL                   INITIATE POLL
0677 P03E4 F281
0677 P03E5 B069
0678                     UJPSD        Q,TBR,SEL                    UNIT SELECT
0678 P03E6 F381
0678 P03E7 B044
0679                     UJPSD        Q,TBR,DECHO                  DRIVE ECHO OUTPUT
0679 P03E8 F481
0679 P03E9 B047
0680                     UJP          SEEK                         LOAD CYL ADDRESS
0680 P03EA F580
0680 P03EB 0049
0681                     UJP          FORMAT                       FORMAT WRITE
0681 P03EC F680
0681 P03ED 007F
0682                     UJPSD        Q,TBR,SEC                    LOAD SEC AND HEAD ADDR
0682 P03EE F781
0682 P03EF B0A0
0683                     UJP          WAIT                         DIRECTOR FUNCTION
0683 P03F0 F880
0683 P03F1 000C
0684                     UJPFF        CLR,XFER,REFD                READ FROM DISK
0684 P03F2 F982
0684 P03F3 000F
0685                     UJPFF        CLR,XFER,WRITE               WRITE TO DISK
0685 P03F4 FA82
0685 P03F5 0CBC
0686                     UJPSD        Q,TBR,CUOP                   ECC CONTROL
0686 P03F6 FB81
0686 P03F7 B010
0687                     UJP          RCONT                        READ RECOV/STATUS CNTL
0687 P03F8 FC80
```

```
0687 P03F9 00A9
0688                     UJPSD     AU,CWAU,SFWAL             SET FWA LOWER
0688 P03FA FD81
0688 P03FB EC41
0689                     UJPSD     AL,CWAUU,SFWAU            SET FWA UPPER
0689 P03FC FE81
0689 P03FD CEF4
0690                     UJPSD     Q,TBR,CUOP                CU ECHO OUTPUT
0690 P03FE FF81
0690 P03FF B010
0691            *
0692            *
0693            *
0694            *
0695                     END

PGM=  0400 (  1024)   COM = 0000 (     0)   DAT = 0000 (     0)
```

```
E Q U I V A L E N C E S
-----------------------

 DEF.LINE  NAME    VALUE              REFERENCED AT LINE NUMBER

   0000    I       00FF   (000255)
   0053    BUSIN   0000   (000000)  0197, 0288, 0609
   0054    MIR     0001   (000001)  0174, 0175, 0192, 0193, 0211, 0212, 0214, 0216, 0218, 0219, 0221, 0223, 0225, 0227, 0228, 0230
                                    0232, 0234, 0236, 0247, 0250, 0252, 0255, 0256, 0263, 0264, 0282, 0283, 0286, 0301, 0302, 0305
                                    0306, 0309, 0310, 0311, 0312, 0316, 0318, 0321, 0333, 0335, 0337, 0339, 0342, 0345, 0350, 0351
                                    0352, 0353, 0354, 0355, 0356, 0357, 0359, 0368, 0369, 0370, 0371, 0373, 0374, 0375, 0376, 0377
                                    0385, 0387, 0390, 0392, 0395, 0409, 0411, 0415, 0418, 0429, 0430, 0433, 0434, 0454, 0455, 0457
                                    0458, 0465, 0467, 0478, 0482, 0485, 0499, 0500, 0505, 0507, 0509, 0511, 0512, 0520, 0521, 0523
                                    0524, 0539, 0540, 0542, 0543, 0545, 0558, 0562, 0569, 0572, 0586, 0592, 0598, 0599, 0605, 0606
                                    0614, 0615, 0618, 0619, 0625, 0626, 0627, 0628, 0629, 0630, 0631, 0632, 0633
   0055    SR1     0002   (000002)  0322, 0397, 0416, 0483, 0504, 0506, 0508, 0561
   0056    SRAM    0003   (000003)  0231, 0235, 0265, 0317, 0386, 0391, 0396, 0410, 0479, 0502, 0559, 0573, 0574
   0057    SFLR    0004   (000004)  0398
   0058    Q       0005   (000005)  0217, 0676, 0677, 0678, 0679, 0682, 0686, 0690
   0059    AL      0006   (000006)  0189, 0238, 0239, 0248, 0279, 0334, 0336, 0346, 0419, 0420, 0689
   0060    AU      0007   (000007)  0249, 0251, 0338, 0340, 0341, 0343, 0360, 0421, 0675, 0688
   0066    R1      0000   (000000)  0192, 0197, 0231, 0235, 0255, 0282, 0286, 0288, 0305, 0311, 0317, 0343, 0386, 0391, 0396, 0410
                                    0433, 0457, 0479, 0485, 0500, 0502, 0511, 0523, 0559, 0569, 0574, 0609
   0067    FLR     0001   (000001)  0174, 0193, 0230, 0232, 0234, 0236, 0247, 0250, 0256, 0264, 0283, 0306, 0309, 0312, 0316, 0318
                                    0321, 0333, 0335, 0337, 0339, 0342, 0350, 0352, 0354, 0356, 0368, 0370, 0373, 0376, 0385, 0387
                                    0390, 0392, 0395, 0397, 0409, 0411, 0415, 0418, 0420, 0434, 0458, 0478, 0482, 0499, 0505, 0507
                                    0512, 0520, 0524, 0539, 0543, 0558, 0572, 0573, 0614, 0625, 0627
   0068    BLL     0003   (000003)  0239, 0430, 0455, 0633
   0069    BLU     0004   (000004)  0429, 0454, 0632, 0675
   0070    CWAL    0005   (000005)  0238, 0631
   0071    CWAU    0006   (000006)  0630, 0688
   0072    CWAUU   0007   (000007)  0629, 0689
   0073    TBR     0008   (000008)  0212, 0217, 0219, 0228, 0252, 0263, 0301, 0345, 0359, 0375, 0465, 0467, 0509, 0545, 0562, 0592
                                    0599, 0605, 0618, 0676, 0677, 0678, 0679, 0682, 0686, 0690
   0074    BOR     0009   (000009)  0189, 0211, 0214, 0216, 0218, 0221, 0223, 0225, 0227, 0249, 0265, 0279, 0302, 0341, 0346, 0360
                                    0377, 0398, 0504, 0542, 0586, 0590, 0598, 0606, 0619
   0075    RAM     000A   (000010)  0175, 0248, 0251, 0310, 0322, 0334, 0336, 0338, 0340, 0351, 0353, 0355, 0357, 0369, 0371, 0374
                                    0416, 0419, 0421, 0483, 0506, 0508, 0521, 0540, 0561, 0615, 0626, 0628
   0076    R1INC   000B   (000011)  0196, 0203, 0259, 0267, 0287, 0293, 0313, 0320, 0323, 0414, 0435, 0459, 0481, 0488, 0503, 0515
                                    0525, 0532, 0560, 0570
   0077    BLDEC   000C   (000012)  0240
   0078    DMACLR  000D   (000013)  0176, 0178, 0406, 0425, 0581, 0584
   0084    REPLY   0000   (000000)  0187, 0253, 0280, 0303, 0349, 0365, 0450
```

```
REJECT   0001   (000001)   0183
CUERR    0004   (000004)   0206, 0210, 0271, 0296, 0299, 0326, 0332, 0403, 0427, 0495, 0538, 0566
NEED     0005   (000005)   0243, 0637
XFER     0006   (000006)   0330, 0443, 0492, 0634, 0684, 0685
EOC      0007   (000007)   0531
TAGOUT   0008   (000008)   0191, 0195, 0205, 0209, 0254, 0258, 0269, 0274, 0281, 0285, 0295, 0304, 0308, 0325, 0379, 0380
                           0402, 0468, 0469, 0494, 0510, 0514, 0534, 0537, 0546, 0547, 0565, 0593, 0594, 0600, 0601, 0607
                           0608, 0620, 0621
RESP     0009   (000009)   0200, 0202, 0261, 0262, 0276, 0277, 0291, 0292, 0326, 0329, 0383, 0384, 0473, 0474, 0517, 0518
                           0551, 0552, 0596, 0597, 0603, 0604, 0612, 0613, 0623, 0624
LPFLG    000A   (000010)   0266, 0270, 0275, 0356, 0587, 0588
TM       000B   (000011)   0426, 0431, 0464
WRT      000C   (000012)   0367, 0404, 0422, 0432, 0440, 0530, 0582
ADRFLD   000D   (000013)   0361, 0364, 0372, 0399, 0444, 0449
READ     000E   (000014)   0405, 0423, 0442, 0452, 0456, 0529, 0583, 0636
ALENB    000F   (000015)   0589, 0591
BUSY     0010   (000016)   0179, 0182, 0585, 0667
DMAGO    0011   (000017)   0470, 0475, 0546, 0553, 0580
PP       0012   (000018)   0407, 0424, 0441, 0447, 0528, 0635
FULWRT   0013   (000019)   0177, 0181, 0498
RITOA    0014   (000020)   0188, 0201, 0575, 0661, 0662, 0663, 0674
RIMSB    0016   (000022)   0389, 0394, 0408
AQEXEC   0001   (000001)   0179, 0183, 0185, 0187, 0253, 0280, 0303, 0349, 0365, 0450
A7       0002   (000002)   0244, 0347, 0363, 0417, 0448
ADFLD    0003   (000003)   0367, 0452, 0519
WRTFF    0004   (000004)   0399, 0476, 0484, 0544, 0554
LFLAG    0005   (000005)   0262, 0604
TSTMD    0006   (000006)   0444, 0451
R107     0007   (000007)   0480, 0488, 0570
PROT     0008   (000008)   0446
BIEVEN   0009   (000009)   0198, 0289, 0610
CE       000A   (000010)   0199, 0203, 0260, 0267, 0290, 0293, 0323, 0327, 0382, 0401, 0472, 0493, 0516, 0532, 0550, 0564
                           0595, 0602, 0611, 0622, 0638, 0640, 0642, 0644
NEORCE   000C   (000012)   0196, 0200, 0259, 0261, 0276, 0287, 0291, 0314, 0328, 0381, 0383, 0471, 0473, 0515, 0517, 0549
                           0551, 0594, 0596, 0601, 0603, 0608, 0612, 0621, 0623
XCOMP    000D   (000013)   0435, 0459, 0489, 0497
ALARM    000E   (000014)   0329, 0384, 0475, 0518, 0553
EMPTY    000F   (000015)   0491, 0496, 0567, 0576, 0578
CUSEL    0010   (000016)   0190, 0241, 0246, 0278, 0300, 0348, 0362, 0445, 0584
TAGVAL   0011   (000017)   0194, 0257, 0284, 0307, 0379, 0468, 0513, 0546, 0593, 0600, 0607, 0620
SECTOR   0012   (000018)   0378, 0556
INDEX    0013   (000019)   0541, 0557
RLEQZ    0015   (000021)   0361, 0366, 0477, 0555
RIEORM   0016   (000022)   0204, 0207, 0268, 0272, 0294, 0297, 0315, 0319, 0324, 0344, 0388, 0393, 0413, 0438, 0462, 0501
                           0527, 0533, 0535, 0617
AUTOL    0017   (000023)   0180, 0585
CLR      0000   (000000)   0177, 0179, 0188, 0195, 0202, 0205, 0209, 0258, 0262, 0269, 0270, 0274, 0275, 0277, 0285, 0292
                           0295, 0308, 0325, 0329, 0361, 0380, 0384, 0394, 0399, 0402, 0404, 0405, 0407, 0408, 0422, 0423
                           0424, 0426, 0440, 0441, 0442, 0444, 0464, 0469, 0474, 0475, 0494, 0498, 0514, 0518, 0528, 0529
                           0530, 0534, 0537, 0547, 0552, 0553, 0565, 0580, 0582, 0583, 0588, 0591, 0594, 0597, 0601, 0604
                           0608, 0613, 0621, 0624, 0634, 0667, 0684, 0685
SET      0001   (000001)   0181, 0182, 0183, 0187, 0191, 0200, 0201, 0206, 0210, 0243, 0253, 0254, 0261, 0266, 0271, 0276
                           0280, 0281, 0291, 0296, 0299, 0303, 0304, 0326, 0328, 0330, 0332, 0349, 0358, 0364, 0365, 0367
```

```
                                   0372, 0379, 0383, 0389, 0403, 0427, 0431, 0432, 0443, 0447, 0449, 0450, 0452, 0456, 0468, 0470
                                   0473, 0492, 0495, 0510, 0517, 0531, 0538, 0546, 0548, 0551, 0566, 0575, 0585, 0587, 0589, 0593
                                   0596, 0600, 0603, 0607, 0612, 0620, 0623, 0635, 0636, 0637, 0661, 0662, 0663, 0674
0142     ZERO     0000    (000000) 0175, 0176, 0176, 0178, 0178, 0189, 0196, 0197, 0203, 0211, 0216, 0231, 0235, 0239, 0240, 0248
                                   0249, 0251, 0259, 0265, 0267, 0279, 0287, 0288, 0293, 0302, 0311, 0313, 0313, 0317, 0320, 0320
                                   0323, 0334, 0336, 0338, 0340, 0343, 0346, 0351, 0353, 0355, 0357, 0386, 0391, 0395, 0396, 0397
                                   0406, 0406, 0410, 0414, 0414, 0425, 0425, 0429, 0433, 0435, 0454, 0457, 0459, 0479, 0481, 0481
                                   0488, 0502, 0503, 0503, 0504, 0506, 0508, 0515, 0521, 0523, 0525, 0525, 0532, 0540, 0559, 0560
                                   0560, 0570, 0573, 0574, 0581, 0581, 0584, 0590, 0590, 0609, 0619, 0626, 0628, 0629, 0630, 0631
0143     ONE      0001    (000001) 0214, 0225, 0236, 0333, 0356, 0368, 0390, 0476, 0539, 0558
0144     TWO      0002    (000002) 0218, 0223, 0234, 0337, 0350, 0370, 0385, 0499, 0507
0145     THRE     0003    (000003) 0232, 0247, 0264, 0352, 0592
0146     FOUR     0004    (000004) 0212, 0221, 0230, 0250, 0354, 0500
0147     FIVE     0005    (000005) 0263, 0632
0148     SIX      0006    (000006) 0252
0149     SEVEN    0007    (000007) 0509, 0618
0150     EIGHT    0008    (000008) 0227, 0228, 0376, 0605, 0606
0151     NINE     0009    (000009) 0375, 0377, 0467, 0542, 0543, 0545, 0614
0152     HEXA     000A    (000010) 0335, 0392, 0465, 0482, 0520, 0562, 0625
0153     HEXB     000B    (000011) 0219, 0339, 0387, 0505, 0627
0154     HEXC     000C    (000012) 0174, 0193, 0256, 0283, 0306, 0312, 0318, 0342, 0345, 0411, 0434, 0456, 0512, 0524, 0599
0155     HEXD     000D    (000013) 0309, 0316, 0321, 0359
0156     HEXE     000E    (000014) 0301, 0373, 0409, 0415
0157     HEXF     000F    (000015) 0418, 0572
0158     HEX10    0010    (000016) 0485, 0569
0159     HEX30    0030    (000048) 0615
0160     HEX3C    003C    (000060) 0310
0161     HEX3F    003F    (000063) 0430, 0455
0162     HEX40    0040    (000064) 0398
0163     HEX5F    005F    (000095) 0286
0164     HEX7F    007F    (000127) 0374, 0586
0165     HEXF0    00F0    (000240) 0192, 0255, 0282, 0305, 0511
0166     HEXFF    00FF    (000255) 0369, 0371, 0598, 0633
```

```
S Y M B O L S
-------------

 DEF.LINE  NAME     ADDRESS          REFERENCED AT LINE NUMBER

   0173     V        0000              0173, 0174, 0175, 0176, 0177, 0178, 0179, 0179, 0180, 0180, 0181, 0181, 0182, 0182, 0183, 0183
                                       0184, 0184, 0185, 0185, 0186, 0186, 0187, 0187, 0188, 0188, 0189, 0190, 0190, 0191, 0192, 0193
                                       0194, 0194, 0195, 0196, 0196, 0197, 0198, 0198, 0199, 0199, 0200, 0200, 0201, 0202, 0202, 0203
                                       0203, 0204, 0204, 0205, 0206, 0206, 0207, 0207, 0208, 0208, 0209, 0210, 0210, 0211, 0212, 0213
                                       0213, 0214, 0215, 0215, 0216, 0217, 0217, 0218, 0219, 0220, 0220, 0221, 0222, 0222, 0223, 0224
                                       0224, 0225, 0226, 0226, 0227, 0228, 0229, 0229, 0230, 0231, 0232, 0233, 0233, 0234, 0235, 0236
                                       0237, 0237, 0238, 0238, 0239, 0240, 0240, 0241, 0241, 0242, 0242, 0243, 0243, 0244, 0244, 0245
                                       0245, 0246, 0246, 0247, 0248, 0249, 0250, 0251, 0252, 0253, 0253, 0254, 0255, 0256, 0257, 0257
                                       0258, 0259, 0259, 0260, 0260, 0261, 0261, 0262, 0262, 0263, 0264, 0265, 0266, 0266, 0267, 0267
                                       0268, 0268, 0269, 0270, 0271, 0271, 0272, 0272, 0273, 0273, 0274, 0274, 0275, 0275, 0276, 0276
                                       0277, 0277, 0278, 0278, 0279, 0280, 0280, 0281, 0282, 0283, 0284, 0284, 0285, 0286, 0287, 0287
                                       0288, 0289, 0289, 0290, 0290, 0291, 0291, 0292, 0292, 0293, 0293, 0294, 0294, 0295, 0296, 0296
                                       0297, 0297, 0298, 0298, 0299, 0299, 0300, 0300, 0301, 0302, 0303, 0303, 0304, 0305, 0306, 0307
                                       0307, 0308, 0309, 0310, 0311, 0312, 0313, 0314, 0314, 0315, 0315, 0316, 0317, 0318, 0319, 0319
                                       0320, 0321, 0322, 0322, 0323, 0323, 0324, 0324, 0325, 0326, 0326, 0327, 0327, 0328, 0328, 0329
                                       0329, 0330, 0331, 0331, 0332, 0332, 0333, 0334, 0335, 0336, 0337, 0338, 0339, 0340, 0341, 0341
                                       0342, 0343, 0344, 0344, 0345, 0346, 0347, 0347, 0348, 0348, 0349, 0349, 0350, 0351, 0352, 0353
                                       0354, 0355, 0356, 0357, 0358, 0358, 0359, 0360, 0360, 0361, 0361, 0362, 0362, 0363, 0363, 0364
                                       0365, 0365, 0366, 0366, 0367, 0367, 0368, 0369, 0370, 0371, 0372, 0373, 0374, 0375, 0376, 0377
                                       0378, 0378, 0379, 0379, 0380, 0381, 0381, 0382, 0382, 0383, 0383, 0384, 0384, 0385, 0386, 0387
                                       0388, 0388, 0389, 0390, 0391, 0392, 0393, 0393, 0394, 0395, 0396, 0397, 0398, 0399, 0399, 0400
                                       0400, 0401, 0401, 0402, 0403, 0403, 0404, 0405, 0406, 0407, 0407, 0408, 0409, 0410, 0411, 0412
                                       0413, 0413, 0414, 0415, 0416, 0416, 0417, 0417, 0418, 0419, 0419, 0420, 0421, 0421, 0422, 0423
                                       0424, 0425, 0426, 0427, 0427, 0428, 0429, 0430, 0431, 0432, 0433, 0434, 0435, 0435, 0436, 0437
                                       0438, 0438, 0439, 0439, 0440, 0441, 0442, 0443, 0443, 0444, 0444, 0445, 0445, 0446, 0446, 0447
                                       0448, 0448, 0449, 0450, 0450, 0451, 0451, 0452, 0452, 0453, 0453, 0454, 0455, 0456, 0457, 0458
                                       0459, 0459, 0460, 0461, 0462, 0462, 0463, 0463, 0464, 0464, 0465, 0466, 0466, 0467, 0468, 0468
                                       0469, 0470, 0471, 0471, 0472, 0472, 0473, 0473, 0474, 0475, 0475, 0476, 0476, 0477, 0477, 0478
                                       0479, 0480, 0480, 0481, 0482, 0483, 0483, 0484, 0484, 0485, 0486, 0487, 0488, 0488, 0489, 0489
                                       0490, 0490, 0491, 0491, 0492, 0492, 0493, 0493, 0494, 0495, 0495, 0496, 0496, 0497, 0497, 0498
                                       0498, 0499, 0500, 0501, 0501, 0502, 0503, 0504, 0505, 0506, 0507, 0508, 0509, 0510, 0511, 0512
                                       0513, 0513, 0514, 0515, 0515, 0516, 0516, 0517, 0517, 0518, 0518, 0519, 0519, 0520, 0521, 0522
                                       0522, 0523, 0524, 0525, 0526, 0527, 0527, 0528, 0529, 0530, 0531, 0531, 0532, 0532, 0533, 0533
                                       0534, 0534, 0535, 0535, 0536, 0536, 0537, 0538, 0538, 0539, 0540, 0541, 0541, 0542, 0543, 0544
                                       0544, 0545, 0546, 0546, 0547, 0548, 0549, 0549, 0550, 0550, 0551, 0551, 0552, 0553, 0553, 0554
                                       0554, 0555, 0555, 0556, 0556, 0557, 0557, 0558, 0559, 0560, 0561, 0561, 0562, 0563, 0563, 0564
                                       0564, 0565, 0566, 0566, 0567, 0567, 0568, 0568, 0569, 0570, 0570, 0571, 0571, 0572, 0573, 0574
                                       0575, 0575, 0576, 0576, 0577, 0577, 0578, 0578, 0579, 0579, 0580, 0581, 0582, 0583, 0584, 0584
                                       0585, 0585, 0586, 0587, 0587, 0588, 0589, 0590, 0591, 0592, 0593, 0593, 0594, 0594, 0595, 0595
                                       0596, 0596, 0597, 0598, 0599, 0600, 0600, 0601, 0601, 0602, 0602, 0603, 0603, 0604, 0604, 0605
                                       0606, 0607, 0607, 0608, 0608, 0609, 0610, 0610, 0611, 0611, 0612, 0612, 0613, 0614, 0615, 0616
```

```
                                  0617, 0617, 0618, 0619, 0620, 0620, 0621, 0621, 0622, 0622, 0623, 0623, 0624, 0625, 0626, 0627
                                  0628, 0629, 0630, 0631, 0632, 0633, 0634, 0635, 0636, 0636, 0637, 0637, 0638, 0638, 0639, 0639
                                  0640, 0640, 0641, 0641, 0642, 0642, 0643, 0643, 0644, 0644, 0645, 0645, 0646, 0647, 0648, 0649
                                  0650, 0651, 0652, 0659, 0659, 0660, 0660, 0661, 0661, 0662, 0662, 0663, 0663, 0664, 0664, 0665
                                  0665, 0666, 0666, 0667, 0667, 0668, 0668, 0669, 0669, 0670, 0670, 0671, 0671, 0672, 0672, 0673
                                  0673, 0674, 0674, 0675, 0675, 0676, 0676, 0677, 0677, 0678, 0678, 0679, 0679, 0680, 0680, 0681
                                  0681, 0682, 0682, 0683, 0683, 0684, 0684, 0685, 0685, 0686, 0686, 0687, 0687, 0688, 0688, 0689
                                  0689, 0690, 0690
0178    CLR2    000A              0407, 0427
0179    IDLE    000C              0180, 0184, 0186, 0188, 0271, 0275, 0292, 0296, 0326, 0331, 0443, 0531
0182    TFORM   0012              0179
0183    ILL     0014              0190, 0244, 0246, 0278, 0300, 0348, 0362, 0445
0185    WAIT    0018              0667, 0683
0187    REP     001C              0202, 0206, 0233, 0237, 0238, 0240, 0243, 0419, 0421, 0575, 0661, 0662, 0663, 0674
0189    CUOP    0020              0242, 0245, 0676, 0686, 0690
0190    CHKCU   0022              0213, 0217, 0220, 0229, 0341, 0347, 0360
0194    JA41    002A              0204
0196    JA43    002E              0207
0200    RESP1   0036              0210
0203    JA42    003C              0194
0206    ERR1    0042              0208
0207    JA44    0044              0196
0209    CE1     0048              0203
0210    IOER1   004A              0198, 0199
0211    CSS     004C              0665
0214    PATT    0052              0669
0216    DFS     0056              0671
0218    COND    005A              0670
0219    SETB    005C              0215
0221    PHYS    0060              0660
0223    CUSTA   0064              0672
0225    DRST1   0068              0673
0227    DRST2   006C              0668
0228    SETA    006E              0222, 0224, 0226
0230    CYLST   0072              0664
0234    SECST   007A              0666
0238    SFWAL   0082              0688
0239    BUFFL   0084              0675
0241    SEL     0088              0678
0243    JA91    008C              0241
0244    DECHO   008E              0679
0246    SEEK    0092              0680
0254    OUT     00A2              0266, 0358
0257    JA111   00A8              0268
0259    JA113   00AC              0272
0267    JA112   00BC              0257
0270    ERR2    00C2              0273, 0277
0272    JA114   00C6              0259
0274    CE2     00CA              0267, 0293, 0323, 0638, 0640, 0642, 0644
0275    JA121   00CC              0262
0276    RESP2   00CE              0260, 0274, 0595, 0602, 0610, 0611, 0622
0278    POLL    00D2              0677
0284    JA141   00DE              0294
```

```
0287     JA143    00E4                 0297
0291     RESP3    00EC                 0299
0293     JA142    00F0                 0284
0296     ERR3     00F6                 0298
0297     JA144    00F8                 0287
0299     IOER2    00FC                 0289, 0290
0300     FORMAT   00FE                 0681
0307     JA171    010C                 0324
0311     OUTLP    0114                 0322
0312     INLP     0116                 0315
0323     JA172    012C                 0307
0326     ERR4     0132                 0319
0327     FWEND    0134                 0314
0328     RESP4    0136                 0332
0331     ERR10    013C                 0329
0332     IOER3    013E                 0327
0333     SEC      0140                 0682
0342     RCONT    0152                 0687
0359     JA201    0174                 0344
0361     WRITE    0178                 0685
0362     NTM      017A                 0577
0363     JA231    017C                 0576
0365     JA232    0180                 0363
0367     NOTTM    0184                 0579
0368     ADFIND   0186                 0453, 0522, 0636
0372     ADCOM    018E                 0483
0375     REPT     0194                 0416
0379     JA251    019C                 0401
0383     RESP5    01A4                 0403
0401     JA252    01C8                 0379
0403     IOER4    01CC                 0382
0404     ERR7     01CE                 0384, 0490, 0491, 0492, 0518
0408     NOCOM    01D6                 0388, 0393
0417     SFWAU    01E8                 0689
0420     LDRAM    01EE                 0417
0422     ERR8     01F2                 0413, 0439, 0534, 0536
0423     ERR9     01F4                 0463
0429     TMW      0200                 0578
0435     JA301    020C                 0438
0440     ALDON    0216                 0435, 0498
0441     TCOMP    0218                 0464
0444     PEED     021E                 0684
0446     JA311    0222                 0444
0448     JA312    0226                 0446
0450     JA321    022A                 0448
0454     TMR      0232                 0451
0459     JA331    023C                 0462
0464     JA332    0246                 0459
0465     WRET     0248                 0399
0467     RRET     024C                 0400
0468     RDWCOM   024E                 0466, 0493
0473     RESP6    0258                 0495
0478     GOON     0262                 0496
```

```
0484     ERRX     026E                0475, 0553
0486     JA431    0272                0468
0491     JA361    027C                0484
0492     COMPL    027E                0489
0493     JA341    0280                0468
0495     IOER5    0284                0472
0496     JA351    0286                0476
0497     DONE     0288                0568, 0571
0499     CHKHD    028C                0480, 0557
0513     JA381    02A8                0533
0515     JA383    02AC                0535
0517     RESP7    02B0                0538
0523     JA371    02BC                0501
0525     JA372    02C0                0527
0532     JA382    02CE                0513
0535     JA384    02D4                0515
0537     CE3      02D8                0532
0538     IOER6    02DA                0516
0539     ADR      02DC                0367, 0452, 0519
0542     ADREP    02E2                0561
0546     COMAD    02EA                0563, 0564
0551     RESP6    02F4                0566
0556     MORE     02FE                0567
0562     JA401    030A                0544
0564     JA411    030E                0546
0566     IOER7    0312                0550
0567     JA421    0314                0554
0569     DELY     0318                0477, 0555
0572     SPEC     031E                0659
0576     BRCHK    0326                0361
0578     CHKBR    032A                0366
0580     AUTO     032E                0181
0584     AUTSEL   0336                0637
0588     DRSEL    033E                0604
0593     JA451    0348                0639
0599     FLTCL    0354                0587
0600     JA461    0356                0641
0605     AGAIN    0360                0617
0607     JA471    0364                0643
0620     JA491    037E                0645
0637     JA441    03A0                0584
0638     JA452    03A2                0593
0640     JA462    03A6                0600
0642     JA472    03AA                0607
0644     JA492    03AE                0620
0659     HIGH     03C0                0182
```

```
*** A L P H A B E T I C A L   S O R T   O F   S Y M B O L S ***

A7       0110  ADCOM    0372  ADFIND   0368  ADFLD    0111  ADR      0539  ADREP    0542  ADRFLD   0095  AGAIN    0605  AL       0059
ALARM    0121  ALDON    0440  ALENB    0097  AQEXEC   0109  AU       0060  AUTO     0580  AUTOL    0129  AUTSEL   0584  BIEVEN   0117
BLDEC    0077  BLEQZ    0127  BLL      0068  BLU      0069  BOR      0074  BRCHK    0576  BUFFL    0239  BUSIN    0053  BUSY     0096
CE       0118  CE1      0209  CE2      0274  CE3      0537  CHKBR    0578  CHKCU    0190  CHKHD    0499  CLR      0135  CLR2     0178
COMAD    0546  COMPL    0492  COND     0218  CSS      0211  CUERR    0086  CUOP     0169  CUSEL    0123  CUSTA    0223  CWAL     0070
CWAU     0071  CWAUU    0072  CYLST    0230  DECHO    0244  DELY     0569  DFS      0216  DMACLR   0078  DMAGO    0099  DONE     0497
DRSEL    0588  DRST1    0225  DRST2    0227  EIGHT    0150  EMPTY    0122  EOC      0089  ERR1     0206  ERR10    0331  ERR2     0270
ERR3     0296  ERR4     0326  ERR7     0404  ERR8     0422  ERR9     0423  ERRX     0484  FIVE     0147  FLR      0067  FLTCL    0599
FORMAT   0300  FOUR     0146  FULWRT   0101  FWEND    0327  GOON     0478  HEX10    0158  HEX30    0159  HEX3C    0160  HEX3F    0161
HEX40    0162  HEX5F    0163  HEX7F    0164  HEXA     0152  HEXB     0153  HEXC     0154  HEXD     0155  HEXE     0156  HEXF     0157
HEXF0    0165  HEXFF    0166  HIGH     0659  I        0000  IDLE     0179  ILL      0183  INDEX    0126  INLP     0312  IOER1    0210
IOER2    0299  IOER3    0332  IOER4    0403  IOER5    0495  IOER6    0538  IOER7    0566  JA111    0257  JA112    0267  JA113    0259
JA114    0272  JA121    0275  JA141    0284  JA142    0293  JA143    0287  JA144    0297  JA171    0307  JA172    0323  JA201    0359
JA231    0363  JA232    0365  JA251    0379  JA252    0401  JA301    0435  JA311    0446  JA312    0448  JA321    0450  JA331    0459
JA332    0464  JA341    0493  JA351    0496  JA361    0491  JA371    0523  JA372    0525  JA381    0513  JA382    0532  JA383    0515
JA384    0535  JA401    0562  JA41     0194  JA411    0564  JA42     0203  JA421    0567  JA43     0196  JA431    0466  JA44     0207
JA441    0637  JA451    0593  JA452    0638  JA461    0600  JA462    0640  JA471    0607  JA472    0642  JA491    0620  JA492    0644
JA91     0243  LDRAM    0420  LFLAG    0113  LPFLG    0092  MIR      0054  MORE     0556  NEED     0087  NEORCE   0119  NINE     0151
NOCOM    0408  NOTTM    0367  NTM      0362  ONE      0143  OUT      0254  OUTLP    0311  PATT     0214  PHYS     0221  POLL     0278
PP       0100  PROT     0116  Q        0058  R1       0066  R107     0115  R1EQRM   0128  R1INC    0076  R1MSB    0103  R1TOA    0102
RAM      0075  RCONT    0342  READ     0096  REED     0444  REJECT   0085  REP      0187  REPLY    0084  REPT     0375  RESP     0091
RESP1    0200  RESP2    0276  RESP3    0291  RESP4    0328  RESP5    0363  RESP6    0473  RESP7    0517  RESP8    0551  ROWCOM   0466
RRET     0467  SEC      0333  SECST    0234  SECTOR   0125  SEEK     0246  SEL      0241  SET      0136  SET8     0228  SETB     0219
SEVEN    0149  SFLR     0057  SFWAL    0238  SFWAU    0417  SIX      0148  SPEC     0572  SR1      0055  SRAM     0056  TAGOUT   0090
TAGVAL   0124  TBR      0073  TCOMP    0441  TFORM    0182  THRE     0145  TM       0093  TMR      0454  TMW      0429  TSTMD    0114
TWO      0144  V        0173  WAIT     0185  WRET     0465  WRITE    0361  WRT      0094  WRTFF    0112  XCOMP    0120  XFER     0086
ZERO     0142
```

```
0206     ERR1     0042          0208
0207     JA44     0044          0196
0209     CE1      0048          0203
0210     IOER1    004A          0198, 0199
0211     CSS      004C          0665
0214     PATT     0052          0669
0216     DFS      0056          0671
0218     COND     005A          0670
0219     SETB     005C          0215
0221     PHYS     0060          0660
0223     CUSTA    0064          0672
0225     DRST1    0068          0673
0227     DRST2    006C          0668
0228     SET8     006E          0222, 0224, 0226
0230     CYLST    0072          0664
0234     SECST    007A          0666
0238     SFWAL    0082          0688
0239     BUFFL    0084          0675
0241     SEL      0088          0678
0243     JA91     008C          0241
0244     DECHO    008E          0679
0246     SEEK     0092          0680
0254     OUT      00A2          0266, 0362
0257     JA111    00A8          0268
0259     JA113    00AC          0272
0267     JA112    00BC          0257
0270     ERR2     00C2          0273, 0277
0272     JA114    00C6          0259
0274     CE2      00CA          0267, 0293, 0323, 0638, 0640, 0642
                                0644
0275     JA121    00CC          0262
0276     RESP2    00CE          0260, 0274, 0595, 0602, 0610, 0611
                                0622
0278     POLL     00D2          0677
0284     JA141    00DE          0294
0287     JA143    00E4          0297
0291     RESP3    00EC          0299
0293     JA142    00F0          0284
0296     ERR3     00F6          0298
0297     JA144    00F8          0287
0299     IOER2    00FC          0289, 0290
0300     FORMAT   00FE          0681
0307     JA171    010C          0324
0311     OUTLP    0114          0322
0312     INLP     0116          0315
0323     JA172    012C          0307
0326     ERR4     0132          0319
0327     FWEND    0134          0314
0328     RESP4    0136          0332
0331     ERR10    013C          0329
0332     IOER3    013E          0327
0333     SEC      0140          0682
0342     RCONT    0152          0687
```

```
0348    JA201   015E             0344
0350    RTZS    0162             0345
0363    WRITE   017C             0685
0364    NTM     017E             0577
0365    JA231   0180             0576
0367    JA232   0184             0365
0369    NOTTM   0188             0579
0370    ADFIND  018A             0453, 0521, 0636
0374    ADCOM   0192             0483
0377    REPT    0198             0417
0381    JA251   01A0             0401
0385    RESP5   01A8             0403
0400    JA252   01C6             0381
0403    IOER4   01CC             0384
0404    ERR6    01CE             0400, 0491, 0563
0405    ERR7    01D0             0386, 0488. 0489, 0490. 0517
0409    NOCOM   01D8             0390, 0394
0418    SFWAU   01EA             0689
0421    LDRAM   01F0             0418
0423    ERR8    01F4             0414. 0439. 0533, 0535
0424    ERR9    01F6             0463
0429    TMW     0200             0578
0435    JA301   020C             0438
0440    ALDON   0216             0435, 0497
0441    TCOMP   0218             0464
0444    REED    021E             0684
0446    JA311   0222             0444
0448    JA312   0226             0446
0450    JA321   022A             0448
0454    TMR     0232             0451
0459    JA331   023C             0462
0464    JA332   0246             0459
0465    WRET    0248             0398
0467    RRET    024C             0399
0468    ROWCOM  024E             0466, 0492
0473    RESP6   0258             0494
0478    GOON    0262             0495
0484    ERRX    026E             0475, 0552
0489    JA361   0278             0484
0490    COMPL   027A             0487
0491    JA341   027C             0468
0494    IOER5   0282             0472
0495    JA351   0284             0476
0496    DONE    0286             0568, 0571
0498    CHKHD   028A             0480, 0556
0512    JA381   02A6             0532
0514    JA383   02AA             0534
0516    RESP7   02AE             0537
0522    JA371   02BA             0500
0524    JA372   02BF             0526
0531    JA382   02CC             0512
0534    JA384   02D2             0514
0536    CE3     02D6             0531
```

```
0537     IOER6    02D8             0515
0538     ADR      02DA             0369, 0452, 0518
0541     ADREP    02E0             0560
0545     COMAD    02E8             0562, 0564
0550     RESP8    02F2             0566
0555     MORE     02FC             0567
0561     JA401    0308             0543
0563     JA411    030C             0545
0566     IOER7    0312             0549
0567     JA421    0314             0553
0569     DELY     0318             0477, 0554
0572     SPEC     031E             0659
0576     BRCHK    0326             0363
0578     CHKBR    032A             0368
0580     AUTO     032E             0181
0584     AUTSEL   0336             0637
0588     DRSEL    033E             0604
0593     JA451    0348             0639
0599     FLTCL    0354             0587
0600     JA461    0356             0641
0605     AGAIN    0360             0617
0607     JA471    0364             0643
0620     JA491    037E             0645
0637     JA441    03A0             0584
0638     JA452    03A2             0593
0640     JA462    03A6             0600
0642     JA472    03AA             0607
0644     JA492    03AE             0620
0659     HIGH     03C0             0182
```

```
*** A L P H A B E T I C A L   S O R T   O F   S Y M B O L S ***

 A7        0110  ADCOM     0374  ADFIND    0370  ADFLD     0111  ADR       0538
 ADREP     0541  ADRFLD    0095  AGAIN     0605  AL        0059  ALARM     0121
 ALDON     0440  ALENB     0097  AQEXEC    0109  AU        0060  AUTO      0580
 AUTOL     0129  AUTSEL    0584  BIEVEN    0117  BLDEC     0077  BLEQZ     0127
 BLL       0068  BLU       0069  BOR       0074  BRCHK     0576  BUFFL     0239
 BUSIN     0053  BUSY      0098  CE        0118  CE1       0209  CE2       0274
 CE3       0536  CHKBR     0578  CHKCU     0190  CHKHD     0498  CLR       0135
 CLR2      0178  COMAD     0545  COMPL     0490  COND      0218  CSS       0211
 CUERR     0086  CUOP      0189  CUSEL     0123  CUSTA     0223  CWAL      0070
 CWAU      0071  CWAUU     0072  CYLST     0230  DECHO     0244  DELY      0569
 DFS       0216  DMACLR    0078  DMAGO     0099  DONE      0496  DRSEL     0588
 DRST1     0225  DRST2     0227  EIGHT     0150  EMPTY     0122  EOC       0089
 ERR1      0206  ERR10     0331  ERR2      0270  ERR3      0296  ERR4      0326
 ERR6      0404  ERR7      0405  ERR8      0423  ERR9      0424  ERRX      0484
 FIVE      0147  FLR       0067  FLTCL     0599  FORMAT    0300  FOUR      0146
 FULWRT    0101  FWEND     0327  GOON      0478  HEX10     0158  HEX30     0159
 HEX3C     0160  HEX3F     0161  HEX40     0162  HEX5F     0163  HEX7F     0164
 HEXA      0152  HEXB      0153  HEXC      0154  HEXD      0155  HEXE      0156
 HEXF      0157  HEXF0     0165  HEXFF     0166  HIGH      0659  I         0000
 IDLE      0179  ILL       0183  INDEX     0126  INLP      0312  IOER1     0210
 IOER2     0299  IOER3     0332  IOER4     0403  IOER5     0494  IOER6     0537
 IOER7     0566  JA111     0257  JA112     0267  JA113     0259  JA114     0272
 JA121     0275  JA141     0284  JA142     0293  JA143     0287  JA144     0297
 JA171     0307  JA172     0323  JA201     0348  JA231     0365  JA232     0367
 JA251     0381  JA252     0400  JA301     0435  JA311     0446  JA312     0448
 JA321     0450  JA331     0459  JA332     0464  JA341     0491  JA351     0495
 JA361     0489  JA371     0522  JA372     0524  JA381     0512  JA382     0531
 JA383     0514  JA384     0534  JA401     0561  JA41      0194  JA411     0563
 JA42      0203  JA421     0567  JA43      0196  JA44      0207  JA441     0637
 JA451     0593  JA452     0638  JA461     0600  JA462     0640  JA471     0607
 JA472     0642  JA491     0620  JA492     0644  JA91      0243  LDRAM     0421
 LFLAG     0113  LPFLG     0092  MIR       0054  MORE      0555  NEED      0087
 NFORCE    0119  NINE      0151  NOCOM     0409  NOTTM     0369  NTM       0364
 ONE       0143  OUT       0254  OUTLP     0311  PATT      0214  PHYS      0221
 POLL      0278  PP        0100  PROT      0116  Q         0058  R1        0066
 R107      0115  R1EQRM    0128  R1INC     0076  R1MSB     0103  R1TUA     0102
 RAM       0075  RCONT     0342  READ      0096  REED      0444  REJECT    0085
 REP       0187  REPLY     0084  REPT      0377  RESP      0091  RESP1     0200
 RESP2     0276  RESP3     0291  RESP4     0328  RESP5     0385  RESP6     0473
 RESP7     0516  RESP8     0550  ROWCOM    0468  RRET      0467  RTZS      0350
 SEC       0333  SECST     0234  SECTOR    0125  SEEK      0246  SEL       0241
 SET       0136  SET8      0228  SETB      0219  SEVEN     0149  SFLR      0057
 SFWAL     0238  SFWAU     0418  SIX       0148  SPEC      0572  SPI       0055
 SRAM      0056  TAGOUT    0090  TAGVAL    0124  TBR       0073  TCOMP     0441
 TFORM     0182  THRE      0145  TM        0093  TMR       0454  TMW       0429
 TSTMD     0114  TWO       0144  V         0173  WAIT      0185  WRET      0465
 WRITE     0363  WRT       0094  WRTFF     0112  XCOMP     0120  XFER      0088
 ZERO      0142
```

COMMENT SHEET

MANUAL TITLE CDC® Storage Module Drive Interface

Hardware Reference/Maintenance Manual

PUBLICATION NO. 96761300 REVISION B

FROM NAME: ______________________________

BUSINESS
ADDRESS: ______________________________

COMMENTS: This form is not intended to be used as an order blank. Your evaluation of this manual will be welcomed by Control Data Corporation. Any errors, suggested additions or deletions, or general comments may be made below. Please include page number.

CUT ALONG LINE

STAPLE

STAPLE

FOLD

FIRST CLASS
PERMIT NO. 333

LA JOLLA CA.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION
PUBLICATIONS AND GRAPHICS DIVISION
4455 EASTGATE MALL
LA JOLLA, CALIFORNIA 92037

CUT ALONG LINE

FOLD

STAPLE

STAPLE

CORPORATE HEADQUARTERS, P.O. BOX 0, MINNEAPOLIS, MINNESOTA 55440
SALES OFFICES AND SERVICE CENTERS IN MAJOR CITIES THROUGHOUT THE WORLD

LITHO IN U.S.A

CONTROL DATA CORPORATION