DOCUMENT INFORMATION

This page provides a sequential record of changes for a multi-page drawing. Each "Revision Description" shall also include the appropriate page number(s). The change on the numbered page(s) shall be indicated with the new revision letter located in the right hand margin of the paragraph that has changed. (The term "Extensive Changes" may be entered if the loss of history is acceptable). All pages of this drawing shall carry the same revision letter as shown on this page.

Title:	795XB / 796XB CONTROLLER UTILITIES ERS
Written By:	MARC LeFEVRE
Model:	795XB / 796XB
Part Numbers:	
Supersedes:	A-07957-60010-6 (B7505XT A-5959-3913-1)
Doc. Users:	ER:48 D/H: C6, ~~15~~, 09, 58, H5, 22

REVISION HISTORY

Rev	Revision Description	Date	Approved by: (name)	MF
A	RENUMBERED FROM A-07957-60010-6 PER C0483042	871208	MO/MARC LEFEVRE MC	M

REV.	DATE THIS ISSUE	DWG. NO	PAGE	
A	12/3/87	A-5959-3913-1	1 OF	28

HEWLETT PACKARD

795XB / 796XB CONTROLLER UTILITIES ERS

MODEL NUMBERS

7957B 7958B 7959B

7961B 7962B 7963B 7963BD

TABLE OF CONTENTS PAGE

1.0 UTILITIES INTERNAL TO THE CONTROLLER

The execution message formats will be similar to the 7945/1X. Differences in content will occur when fields in the execution message are padded with zeros because the required information is not available. The physical sector number, for example, is not available on this product. Hardware differences will cause a redefinition of the fault codes.

Several new utilities and options will be supported in addition to the 7945 utilities. These are:

Read Revision number
Pattern error rate test (cylinder mode & surface mode)
Read Defect List
Read Headers (not described in this document)

Utilities are firmware routines which perform error rate tests, access error logs, access fault logs, and access the spare table. The utilities may be initiated through the CS80 command 'Initiate Utility'.

The specific utility is selected by a one byte micro opcode which may be followed by up to eight bytes of input parameters.

The format for the 'Initiate Utility' command is:

<Initiate Utility> <Micro Opcode> <up to 8 parameter bytes>

Initiate utility = 001100XX
where XX: 00 = no execution message
01 = device will receive execution message text
10 = device will send execution message text

NOTE: This controller does not support utilities which receive an execution message.

2.0 CLASSIFICATION OF UTILITIES

The utilities are classified by the method through which they are invoked. This list includes all of the supported utilities.

CS80 NO EXECUTION MESSAGE

- Clear Logs
- Pattern Error Rate Test (ERT)
- Random Pattern ERT
- Read-Only ERT
- Random Read-Only ER
- Preset

CS80 SEND EXECUTION MESSAGE

- Read Fault Log
- Read Run Time Log
- Read Error Log
- Read Spare Table
- Locate and Read Full Sector
- Servo Test
- Pattern ERT
- Random Pattern ERT
- Read Only Test
- Random Read Only ERT
- Read ROM Revision Number
- Read Defect ESDI List

3.0 LOGS INTERNAL TO THE CONTROLLER

The Run Time (RT) Error log and Fault log contain data error information and fault information respectively. These logs are stored in redundant copies on maintenance tracks in physical cylinder 0 of the disc.

The ERT log is RAM based and contains information on data errors which occur during error rate testing.

3.1 RUN TIME AND FAULT LOGS

During runtime, marginal and recoverable data error information is placed in a RAM-based queue waiting to be posted to the disc-based logs. The same holds true for any drive faults that occur (i.e., there is a fault queue as well as a runtime error queue).

After the Qstat byte has been sent during the report phase of a command, a logs update flag is checked. If there is information to log, all of the RAM-based queues are added to their counterpart logs on the disc. If any of the logs are full, the new entries are written over the oldest entries in the logs, and that older information is lost. Due to the nature of this scheme, the maintenance track overflow bit in status will never be set.

This log update is transparent to the host and takes place in about 120 ms. Consequently, the next host command will be held off for that period of time. The request release function is not used at all with logs.

3.1 RUN TIME AND FAULT LOGS (cont'd)

IMPLEMENTATION:

The RAM-based fault queue is 5 entries long and the runtime error queue is 3 entries long. The disc-based fault log can hold 46 entries, and the runtime error log can hold 50 entries.

Since the only time the queues get posted to the disc is during report phase, long execution time commands (such as ERTs) run the risk of overflowing the queues. To lessen the effect of this, the queues are managed such that when they get full, the last entry in the queue is overwritten with any new entries as they are posted.

After successful posting of queue information to the disc, the queues are cleared. If the data wasn't successfully posted, the queues remain intact, but the log update flag is cleared. As a result, no more log posting attempts will be made until a new entry is added to one of the queues.

3.1 RUN TIME AND FAULT LOGS (cont'd)

Run Time data errors will take the following format in the logs.

Current physical cylinder	2 bytes
Current physical head	1 byte
Current physical sector	1 byte
Current logical cylinder	2 bytes
Current logical head	1 byte
Current logical sector	1 byte
Error Byte	1 byte
Occurrence Count	1 byte

Faults take the following format in the logs.

Current logical cylinder	2 bytes
Current logical head	1 byte
Current logical sector	1 byte
Target logical cylinder	2 bytes
Target logical head	1 byte
Target logical sector	1 byte
Fault Code	1 byte
Status byte	1 byte
Status byte	1 byte

NOTE: Note these formats are the same as the formats in which the data is returned during the read of the respective logs.

3.1.1 Fault Status Bytes

The contents of the 2 status bytes which are associated with a Fault entry depends on Fault code of that entry. The range of all possible Fault codes (1...255) has been divided into subranges for the purpose of defining appropriate contents for the status bytes.

1-1FH	Faults not logged
20-3FH	Data errors NO fault log entry
40-4FH	Status1=compressed ESDI status, Status2=vendor unique status
50-5FH	Status1=0, Status2=0
60-6FH	Status1=compressed 8466 status, Status2=8466 error reg
70-8FH	Status1=8466 Status reg, Status2=8466 error reg
90-AFH	Status1=compressed ESDI status, Status2=vendor unique status
B0-DFH	Status1=0, Status2=0
E0-FFH	Status1=compressed ESDI status, Status2=0

Compressed ESDI status:

bit 0 = write fault
bit 1 = write gate with track offset
bit 2 = seek fault
bit 3 = invalid or unimplemented command
bit 4 = interface fault
bit 5 = command data parity fault
bit 6 = power on conditions exist
bit 7 = spindle motor stopped

Vendor Unique status:

Only the least significant byte of the first word of vendor unique status is saved. See Micropolis or Coyote product descriptions for a definition of the bits.

3.0 LOGS INTERNAL TO THE CONTROLLER (cont'd)

3.1 RUN TIME AND FAULT LOGS (cont'd)

3.1.2 Physical Address Reporting

Faults will normally be reported with a logical address as is shown in section 5.1 of this document. If the seek preceding the occurrence of the fault was a physical seek then the current logical cylinder and target logical cylinder will be replaced by a current physical cylinder and a target physical cylinder. The exercisers and test systems are informed of the switch by setting the most significant bit of the cylinder address.

NOTE: The physical seek may be to an address which is within the logical address space, but the physical address will still be reported. It is assumed that any operation which performs a physical seek will be primarily interested in the physical address if a fault occurs.

3.0 LOGS INTERNAL TO THE CONTROLLER (cont'd)

3.1 RUN TIME AND FAULT LOGS (cont'd)

3.1.3 Run Time Sectors Read Count

The controller maintains an estimate of the number of sectors read during run time (i.e., not utility or diagnostic) commands. An estimate of the number of sectors read by a head is returned to the host in the header of the read run time log execution message.

The disclaimer 'estimate' is given to the count for several reasons. The count is not kept on an individual head basis as is the case for the ERT sectors read count. The run time sectors read count is a total for all heads, which is divided by the total number of heads, to give an estimate of the number of sectors read by an individual head. This compromise results in lower overhead associated with real time command operation, and also reduces the space in the logs that is required to store that information.

The count is also compromised by the fact that it is initially kept in RAM. The count is only moved to the disc-based logs when a run time marginal/unrecoverable data error occurs or the utility command PRESET is issued. Therefore, if power is lost before the RAM based count is added to the disc-based count, an inaccuracy equal to the RAM based count occurs.

NOTE: Reading the run time error log also updates the disc-based count with the total of the RAM based count and the disc-based count.

3.2 ERROR RATE TEST (ERT) LOG

The RAM based ERT table will store data error information on a maximum of 50 locations.

The log is allocated in the following way:

sectors read head 0	5 bytes
sectors read head 1	5 bytes
.	.
.	.
.	.
sectors read head 15	5 bytes
	80 bytes

entry:	physical cylinder	2 bytes
	physical head	1 byte
	physical sector *	1 byte
	logical cylinder	2 bytes
	logical head	1 byte
	logical sector	1 byte
	error byte	1 byte
	occurrence count	1 byte
		10 bytes

50 entries ===> 500 bytes

* equals logical sector

If the error rate tests are initiated via CS80 using the 'send execution message' option, errors are reported immediately to the host and are not logged to the ERT log.

Both 'send execution' and 'no execution' ERT utilities keep a count of the sectors successfully read by individual heads. The sectors read count is read by the host through the read ERT log command.

3.3 ERROR BYTE DESCRIPTION

The error byte is used by error rate test utilities and by the run time log to report the location of a data error and its severity to the host.

error byte format:

```
bit 7 = Byte sync (No Data Sync or Sector Overrun)
bit 6 = error in header field
bit 5 = error in data field

bit 4 = unrecoverable data error  (all retries exhausted)
bit 3 = ECC marginal data error
bit 2 = retry marginal data error (more than 1 retry required)
bit 1 = recoverable on the first retry
bit 0 = track offset invoked
```

bits 5-7 describe the location
bits 1-4 describe the severity

Bit 0 of the error byte is set when a radial track offset was used to recover data during the ERT. The recovery algorithm includes doing 12 retries on track before an offset is attempted. As a result, the error status must be at least marginal for bit 0 to be set. Consequently, bits 0 and 1 will never be set together.

4.0 CS80 NO EXECUTION MESSAGE UTILITIES

Error rate tests run with 'no execution message' store error information in the ERT table and faults in the fault log. See Section 3.0 for a description of these logs.

4.1 CLEAR LOGS

```
MICRO OPCODE : 0CDH (205)
PARAMETERS   : 0 = clear all logs (run time/Fault log and ERT log)
               1 = clear ERT log only
               2 = clear disc-based logs (fault, run time data error)
```

Clearing the ERT log clears the ERT sectors read count. Clearing the disc-based logs clears the run time sectors read estimate.

4.2 PATTERN ERROR RATE TEST (ERT)

```
MICRO OPCODE : 0C8H  (200)
PARAMETERS   : LOOP     (0-255) 255 implies infinite loop
             : OFFSET   (XXXXXXXX)
             : REPORT   (XXXXXXXX)
             : TEST AREA        0 = SECTOR
                                1 = TRACK
                                2 = CYLINDER
                                3 = SURFACE
                                4 = VOLUME (the whole disc)
             : PATTERN SELECT   0 = change pattern with each loop *
                                1 = 39CE7H
                                2 = C30H
                                3 = 30E61CC3987H
                                4 = B8F32E3CCH
                                5 = CCH
                                6 = DB6H
                                7 = 33F94CFE5H
                                8 = random data
```

*Includes random data

This utility will store the addresses of data errors in the ERT log. The sector count is incremented for each head at the end of a read operation.

The first loop of the test begins at the target address which should be determined by the complementary command 'Set Address'. If the target address is in the middle of the test area, the first pass will consist of writing from the target address to the end of the test area. The first read operation will also be from the target address to the end of the test area. The second loop, and each loop thereafter, will be from the start of the test area to the end of the test area.

Each loop consists of writing and reading the test area. If the loop count is set to 255, the test will continue indefinitely until a CANCEL or CLEAR command is sent from the host.

4.3 RANDOM PATTERN ERROR RATE TEST

```
MICRO OPCODE : 0CBH  (203)
PARAMETERS   : LOOP    (0-255) 255 implies infinite loop
             : OFFSET  (XXXXXXXX)
             : REPORT  (XXXXXXXX)
             : PATTERN SELECT  0 = change pattern with each loop *
                               1 = 39CE7H
                               2 = C30H
                               3 = 30E61CC3987H
                               4 = B8F32E3CCH
                               5 = CCH
                               6 = DB6H
                               7 = 33F94CFE5H
                               8 = random data
```

* Includes random data

The Random Pattern ERT functions similarly to the non-random Pattern ERT except that the test area is randomly generated. This is done by generating a random starting address and then generating a random transfer length between 1 sector and 63 sectors. It is assumed that random error rate test is allowed to read and write anywhere in the logical data space of the disc.

One host-specified loop consists of 256 random writes with the starting address and length generated as described above. This is followed by 256 random reads whose starting addresses and lengths are identical to those used for the previous 256 writes.

This test increments the sector count for each head.

4.0 CS80 NO EXECUTION MESSAGE UTILITIES (cont'd)

4.4 READ ONLY ERROR RATE TEST

```
MICRO OPCODE : 0C9H  (201)
PARAMETERS   : LOOP    (0-255) 255 implies infinite loop
             : OFFSET  (XXXXXXXX)
             : REPORT  (XXXXXXXX)
             : TEST AREA       0 = SECTOR
                               1 = TRACK
                               2 = CYLINDER
                               3 = SURFACE
                               4 = VOLUME (the whole disc)
```

This utility will store the addresses of data errors in the ERT log and fault descriptions in the fault log. The sector count is updated for each head. As the name implies, this test does not write data on the disc. Therefore, data written previous to calling this utility will not be destroyed.

The first loop of the test begins at the target address which should be determined by the complementary command 'Set Address'. If the target address is in the middle of the test area, the first pass will consist of reading from the target address to the end of the test area. Each loop, thereafter, will begin at the start of the test area.

4.5 RANDOM READ ONLY ERROR RATE TEST

```
MICRO OPCODE : 0CCH  (204)
PARAMETERS   : LOOP    (0-255) 255 implies infinite loop
```

This routine functions like the Read Only Error Rate test. The one exception is that the TEST AREA is generated randomly. This is done by generating a random starting address and then generating a random length between 1 sector and 63 sectors.

One host-specified loop consists of 256 random reads where the starting address and length are randomly generated for each read.

4.6 PRESET

MICRO OPCODE : 0CEH (206)
PARAMETERS : none

This utility posts any data in the fault and runtime queues as well as the runtime sectors read count to the disc. It is usually not necessary since this posting takes place as necessary during the Report phase of a command.

4.7 FAULTS AND ERRORS DURING ERROR RATE TESTING

Faults: If a fault is encountered, the ERT is halted and the state of the machine is saved. Qstat is set and the status will indicate the fault. The host may continue the test by sending the ERT command sequence with the loop count set to zero.

Note that, due to the way the fault queue is maintained, if you execute a no-execution message ERT, the maximum number of log-only faults you will observe is 5. This is due to the fact that the queue only gets posted to the disc during report phase, and since the test executes for a long time without going to report phase, the queue will fill up and the last entry will be continually overwritten.

A fault is generated during error rate testing if the ERT log overflows. The host is allowed to read the ERT log, send the clear ERT log command and then continue the error rate test by sending the ERT command sequence with the loop count set to zero.

Errors: If a data error occurs, it will be logged in the ERT log and the test will continue at the next block. Qstat will be "pass" if only data errors occur.

NOTE: If no errors have occurred since the last restart of a send execution message test, then one byte (containing a zero) tagged with EOI will be returned to the host upon completion of the test.

5.0 CS80 SEND EXECUTION MESSAGE UTILITIES

5.1 READ FAULT LOG

MICRO OPCODE : 0C7H (199)
PARAMETERS : none

The utility reads the data from the disc-based fault log and returns the entries to the host in chronological order. That is, the first faults returned are the oldest, and the last one returned is the most recent.

NOTE: Performing an error rate test does not clear the fault log, so unless the log is specifically cleared before beginning a test, the faults which are returned may have occurred during previous tests or run time activities.

Format:

header:

# of entries	1 byte	

(Max. number of entries: 46)
entry:

Current logical cylinder	2 bytes	32 - 33
Current logical head	1 byte	34
Logical sector	1 byte	35
Target logical cylinder	2 bytes	36 - 37
Target logical head	1 byte	38
Logical sector	1 byte	39
Fault code	1 byte	40
Status	1 byte	41
Status	1 byte	42

See Section 3.11 for a description of the contents of the status bytes. See Section 3.12 for a description of physical address reporting for faults.

NOTE: In the case where a read of the fault log fails, the Qstat returned will be 1, but the execution message will contain any fault information that was present in the queue.

Also, if the queue contains data when the fault log is read, that data is assumed to be new, relevant data and is appended to the disc- based information. As a result, the maximum number of faults that can be returned is 46 + 5, or 51.

5.2 READ RUN TIME LOG

MICRO OPCODE : 0C5H (197)
PARAMETERS : head # (1 byte)

The utility reads the disc-based runtime error log and sorts out those errors that pertain to the selected head. These errors are then returned to the host in chronological order.

NOTE: Since an entry contains an occurrence count, the total number of errors may exceed the number of entries.

format:

header:

# of entries on the selected head	1 byte
estimate sectors read by this head	5 bytes
not used = 0	2 bytes
# of entries on the selected head	1 byte

entry: (selected head only)

Current physical cylinder	2 bytes
Current physical head	1 byte
Logical sector	1 byte
Current logical cylinder	2 bytes
Current logical head	1 byte
Logical sector	1 byte
'Error byte'	1 byte
Occurrence count	1 byte

The error byte gives information about the location in the sector where the error occurred and the severity of the error (i.e.: recoverable, nonrecoverable).

error byte format:

bit 7 = Byte sync (No Data Sync or Sector Overrun)
bit 6 = error in header field
bit 5 = error in data field

bit 4 = unrecoverable data error (all retries exhausted)
bit 3 = ECC marginal data error
bit 2 = retry marginal data error (more than 1 retry required)
bit 1 = recoverable on the first retry
bit 0 = track offset invoked

5.3 READ ERROR (ERT) LOG

```
MICRO OPCODE : 0C6H  (198)
PARAMETERS   : head #    (1 byte)
```

This utility returns the data error information stored in the RAM based ERT log to the host. Only those data errors which occurred on the specified head are sent to the host.

format:

header:

# of entries on the selected head	1 byte
# of sectors read by this head	5 bytes
not used = 0	2 bytes
# of entries on the selected head	1 byte

entry: (selected head only)

current phys cyl	2 bytes
current phys head	1 byte
current phys sector *	1 byte
current log cyl	2 bytes
current log head	1 byte
current log sector	1 byte
'Error byte'	1 byte
occurrence count	1 byte

* equals logical sector

The error byte gives information about the location in the sector where the error occurred and the severity of the error (i.e., recoverable, nonrecoverable).

error byte format:

```
bit 7 = Byte sync (No Data Sync or Sector Overrun)
bit 6 = error in header field
bit 5 = error in data field

bit 4 = unrecoverable data error  (all retries exhausted)
bit 3 = ECC marginal data error
bit 2 = retry marginal data error (more than 1 retry required)
bit 1 = recoverable on the first retry
bit 0 = track offset invoked
```

If a data error occurs more than once at the same location (i.e., same physical cylinder, physical head, and logical sector) on the disc, the error byte is 'OR'ed to give a cumulative report, and the occurrence count is incremented.

5.0 CS80 SEND EXECUTION MESSAGE UTILITIES (cont'd)

5.4 READ SPARE TABLE

MICRO OPCODE : 0C4H (196)
PARAMETERS : Table = 1 (spare table)

The spare table is the only accessible table in this controller. Since the spare table is not stored on maintenance tracks, it must be built on power up. This is done by going to the spare track area and seeing which have been allocated. The first time the spare table is read after power on will take up to 2 minutes to execute because each track must be checked to count the number of sector spares. Subsequent reads do not require this step and will execute much faster.

HEADER		
Head #	1 byte	
# of spare operations	2 bytes	(MSB first)
# of spare tracks used	1 byte	
# of logical spared tracks	1 byte	

ENTRY		
Cylinder High	1 byte	
Cylinder low	1 byte	
Scalar #	1 byte	(MSB set indicates a factory spare track)

The head # is the head address. There will be one header for each head in the drive.

The '# of spare operations' is the number of sectors on this head which have been spared.

The '# of spare tracks used' is the total number of spare tracks on this head that have been used for factory and field sparing operations.

The '# of logical spared tracks' is the number of tracks on this head which have been determined to be defective and were spared. The number of logical spared tracks will be equal to the number of entries.

The scalar # indicates which of the spare tracks on a surface was used when the defective track was spared. For example, if there are 6 (0 thru 5) spare tracks per surface, and the second spare track was used by this entry, then the scalar # would be 1.

5.0 CS80 SEND EXECUTION MESSAGE UTILITIES (cont'd)

5.5 LOCATE AND READ FULL SECTOR

```
MICRO OPCODE : 0C0H  (192)
PARAMETERS   :   physical cylinder     (2 bytes)
                 physical head         (1 byte)
                 physical sector       (1 byte)
```

The utility will read a physical sector and return the 256 data bytes from that sector and the 6 ECC bytes. The total execution message will be 262 bytes long. Faults will show up in the status, but data errors are neither detected nor reported.

Headers are not read by the read full sector command.

5.6 SERVO TEST

```
MICRO OPCODE : 0BFH  (191)
PARAMETERS   : Loop count
```

To mimic the Falstaff servo test, this utility will perform seeks to the following tracks: 0, 1, 2, 4, 8, 16, 32,... n, 0, n, n-1, n-2, n-4, n-8, n-16, n-32, ... 0. (n = max log track) This sequence of seeks is repeated once for each head.

A seek failure or a timeout will be reflected in the status as well as the first byte of the execution message. A seek error will only be reflected in the first byte of the execution message. A seek error results if the header is readable, and it shows that we are not on the target track.

Servo test can be cancelled or cleared for early termination.

The test will halt and return the execution message as soon as the first failure occurs.

execution message:

```
   byte 1:  0 = pass                    Qstat 0
     1 = seek failure          Qstat 1 (timeout, header read failed)
     2 = seek error            Qstat 0

   byte 2, 3:  Number of seeks completed in the last loop.
```

5.7 PATTERN ERROR RATE TEST (ERT)

```
MICRO OPCODE : 0C8H  (200)
PARAMETERS   : LOOP     (0-255) 255 implies infinite loop
             : OFFSET   (XXXXXXXX)
             : REPORT   (XXXXXXXX)
             : TEST AREA        0 = SECTOR
                                1 = TRACK
                                2 = CYLINDER
                                3 = SURFACE
                                4 = VOLUME (the whole disc)
             : PATTERN SELECT   0 = change pattern with each loop *
                                1 = 39CE7H
                                2 = C30H
                                3 = 30E61CC3987H
                                4 = B8F32E3CCH
                                5 = CCH
                                6 = DB6H
                                7 = 33F94CFE5H
                                8 = random data
```

* includes random data

This utility will report data errors to the host immediately after they occur. The errors are not stored in the ERT log.

The sector count is updated for each head.

The first loop of the test begins at the target address which should be determined by the complementary command 'Set Address'. If the target address is in the middle of the test area, the first pass will consist of writing from the target address to the end of the test area. The first read operation will also be from the target address to the end of the test area. The second loop and each loop thereafter will be from the start of the test area to the end of the test area.

Each loop consists of writing and reading the test area. If the loop count is set to 255, the test will continue indefinitely until a CANCEL or CLEAR command is sent from the host.

5.7 PATTERN ERROR RATE TEST (ERT) (cont'd)

If a fault occurs, the test will halt; Qstat is set; status will indicate the fault.

If a data error occurs, the execution message has the following format:

(no header is sent)
entry:

current phys cyl	2 bytes
current phys head	1 byte
current phys sector	1 byte *
current logical cyl	2 bytes
current logical head	1 byte
current logical sector	1 byte
error byte	1 byte
loop count when error occurred	1 byte

* equals Logical sector

error byte format:
bit 7 = Byte sync (No Data Sync or Sector Overrun)
bit 6 = error in header field
bit 5 = error in data field

bit 4 = unrecoverable data error (all retries exhausted)
bit 3 = ECC marginal data error
bit 2 = retry marginal data error (more than 1 retry required)
bit 1 = recoverable on the first retry
bit 0 = track offset invoked

The state of the test is saved when an error or fault occurs. After reporting the error/fault, the test may be continued by re-sending the ERT command sequence with the loop count set to zero.

At the end of the test (loop count exhausted), one byte (containing a zero for Falstaff I compatibility) tagged with EOI will be sent to the host.

5.8 RANDOM PATTERN ERROR RATE TEST

```
MICRO OPCODE : 0CBH  (203)
PARAMETERS   : LOOP    (0-255) 255 implies infinite loop
             : OFFSET  (XXXXXXXX)
             : REPORT  (XXXXXXXX)
             : PATTERN SELECT  0 = change pattern with each loop *
                               1 = 39CE7H
                               2 = C30H
                               3 = 30E61CC3987H
                               4 = B8F32E3CCH
                               5 = CCH
                               6 = DB6H
                               7 = 33F94CFE5H
                               8 = random data
```

* includes random data

The Random Pattern ERT functions like the non-random Pattern ERT with the exception that the test area is randomly generated. This is done by generating a random starting address and then generating a random transfer length between 1 sector and 63 sectors. It is assumed that random error rate test is allowed to read and write anywhere in the logical data space of the disc.

One loop of the test is performed by writing 256 random test areas and then reading the same 256 random test areas. The random number generator is reinitialized to generate the same sequence of random test areas each time the Random pattern Error rate test command is received. (NOTE: A test may be continued after a data error has been reported. This is done by re-issuing the command with the loop count set to zero. Continuing a test does not reinitialize the random number generator to the start of the sequence.)

The sector count is updated for each head.

See Section 5.7 for details on execution message.

5.9 READ ONLY ERROR RATE TEST

```
MICRO OPCODE : 0C9H  (201)
PARAMETERS   : LOOP    (0-255) 255 implies infinite loop
             : OFFSET  (XXXXXXXX)
             : REPORT  (XXXXXXXX)
             : TEST AREA       0 = SECTOR
                               1 = TRACK
                               2 = CYLINDER
                               3 = SURFACE
                               4 = VOLUME (the whole disc)
```

This utility will report data errors to the host as soon as they occur. The sector count is updated for each head and is available by reading the ERT log. As the name implies, this test does not write data on the disc. Therefore, data written previous to calling this utility will not be destroyed.

The first loop of the test begins at the target address which should be determined by the complementary command 'Set Address'. If the target address is in the middle of the test area, the first pass will consist of reading from the target address to the end of the test area. Each loop, thereafter, will begin at the start of the test area.

If a fault occurs, the test will halt; Qstat is set; status will indicate the fault.

If a data error occurs, the execution message has the following format:

(no header is sent)
entry:

current phys cyl	2 bytes
current phys head	1 byte
current phys sector	1 byte *
current logical cyl	2 bytes
current logical head	1 byte
current logical sector	1 byte
error byte	1 byte
loop count when error occurred	1 byte

* equals Logical sector

5.11 READ REVISION NUMBERS

MICRO OPCODE : 0C3H (195)
PARAMETERS : none

This utility will read the firmware ROM revision numbers and return them in an execution message. The first byte specifies the number of revision number bytes that will follow.

byte #
- 0 number of bytes to follow
- 1 revision # for ROM 1
- 2 revision # for ROM 2 (if installed)

The format for the revision byte is for the most significant nibble to contain the main revision number, and for the least significant nibble to contain a secondary revision number.

5.12 READ DEFECT LIST

MICRO OPCODE : 0D4H (212)
PARAMETERS : 2 byte physical cylinder
1 byte physical head

This routine attempts to read the ESDI defect list stored on the track selected by the input parameter. The disc controller chip is configured for the proper format and an ignore header - read data operation is performed. The CRC is used to check the integrity of the data. A Qstat of 1 is returned if the hard disc controller chip detects any problem or a timeout occurred while waiting for the operation to complete. The Qstat will also be set to 1 if the seek to the target track fails. No retries are attempted by this utility.

The execution message will always consist of 256 bytes. The execution message is initialized to 99H and is overwritten by the data as physical sector zero of the target track is read. The execution message data will be in the format specified by ESDI for the defect list data field.

This routine does input parameter checking and allows only three possible values for the cylinder address. Those values are the maximum cylinder (for that particular mechanism), maximum cylinder -8, or cylinder 4095 (a special ESDI provision for 'hiding' another defect list). Any head between 0 and the maximum head for that mechanism is allowed.

5.9 READ ONLY ERROR RATE TEST (cont'd)

error byte format:
bit 7 = Byte sync (No Data Sync or Sector Overrun)
bit 6 = error in header field
bit 5 = error in data field

bit 4 = unrecoverable data error (all retries exhausted)
bit 3 = ECC marginal data error
bit 2 = retry marginal data error (more than 1 retry required)
bit 1 = recoverable on the first retry
bit 0 = track offset invoked

The state of the test is saved when an error or fault occurs. After reporting the error/fault, the test may be continued by re-sending the ERT command sequence with the loop count set to zero.

At the end of the test (loop count exhausted), one byte (containing a one) tagged with EOI will be sent to the host.

5.10 RANDOM READ ONLY ERROR RATE TEST

MICRO OPCODE : 0CCH (204)
PARAMETERS : LOOP (0-255) 255 implies infinite loop

This routine functions like the Read Only Error Rate test. The one exception is that the TEST AREA is generated randomly. This is done by generating a random starting address and then generating a random length between 1 sector and 63 sectors.

See Section 5.9 for details on the execution message.