

IDENTIFICATION

Product Code: MAINDEC-15-D3RA-D (D)  
Product Name: TC02 DECTape Random Exerciser  
Date: January 14, 1970  
Maintainer: Diagnostics Group  
Author: Edward P. Steinberger

10

1. ABSTRACT

TC02 DECTape Random Exerciser is a program for use with the TC02 DECTape control and from 1 to 8 TU55 DECTape transports. DECTape functions are exercised by the random determination of function, direction, transport number, number of blocks and data pattern generation. Search, Read Data and Write Data are exercised in both Normal and Continuous modes; Read All is exercised in the Continuous mode; Move is exercised. During the waiting period while DECTape functions are being performed, small instruction test programs are being executed by the central processor to test the proper execution of the Data Break facility.

2. REQUIREMENTS

2.1 Equipment

Standard PDP-15 computer with a TC02 DECTape control and at least one TU55 DECTape transport (up to eight may be tested), and one standard PDP-9 Format DECTape (1100<sub>3</sub> Blocks, 400<sub>8</sub> words each) for each transport.

2.2 Storage

2.2.1 Program - Most of memory from 00000 to 04000.

2.2.2 JMS Test Area - 04000 to 04201.

2.2.3 Data Buffers - 04000 to 05077.

2.3 Preliminary Programs

TC02 Basic Exerciser

3. LOADING PROCEDURE

3.1 Method

The program is loaded from paper tape using the standard binary tape loading procedure.

- a. Place tape in the reader.
- b. Set the ADDRESS switches to 17700.
- c. Set BANK MODE switch to 1.
- d. Depress I/O RESET.
- e. Depress KEY READ IN.
- f. Program will be loaded into memory.

4. STARTING PROCEDURE

4.1 Control Switch Settings

The following is a table of Accumulator Switch settings and their action on the program.

<u>AC Switch</u>	<u>Set As</u>	<u>Action</u>
0	1	HALT ON ERROR
	0	DON'T HALT ON ERROR
1	1	DON'T PRINT ERRORS
	0	PRINT ERRORS
2	1	PRINT ONLY 4 DATA ERRORS
	0	PRINT ALL DATA ERRORS
3	1	HIT END ZONE TWICE FOR 0000 or 1077
	0	HIT END ZONE ONCE FOR 0000 or 1077
10	1	EXERCISE TRANSPORT 8
	0	
11	1	EXERCISE TRANSPORT 1
	0	
12	1	EXERCISE TRANSPORT 2
	0	
13	1	EXERCISE TRANSPORT 3
	0	
14	1	EXERCISE TRANSPORT 4
	0	
15	1	EXERCISE TRANSPORT 5

4.1 Control Switch Settings Cont.

<u>AC Switch</u>	<u>Set As</u>	<u>Action</u>
	0	
16	1	EXERCISE TRANSPORT 6
	0	
17	1	EXERCISE TRANSPORT 7
	0	

4.2 Starting Address

The program's starting address is 00100.

4.3 Operator Action

- a. Load program per 3.1.
- b. Put each TU55 to be tested ON LINE with WRITE ENABLED and a standard PDP-9 DECtape on it.
- c. Set the selector dial(s) to the appropriate number(s).
- d. Set AC switches per 4.1 (Normal setting is 000XXX).
- e. Depress I/O RESET.
- f. Depress START

5. OPERATING PROCEDURE

5.1 Operational Switch Settings

See 4.1.

5.2 Subroutine Abstract

None

### 5.3 Operator Action

"Bad" DECTape transports may be "deselected" at any time by setting to 0 the AC Switch for that transport.

## 6. ERRORS

All DECTape malfunctions detected by that program result in an error typeout (if ACS 1 is 0) and an error halt (if ACS 0 is 1). The halt will not occur until all errors pertaining to the block and operation have been typed.

The first three lines of every typeout indicate the DECTape drive, operation, direction, and mode, and the block in question.

### 6.1 Error Halts

#### 6.1.1 Processor Test Error Halts

Processor test errors cause error halts only. Listed below is the location of these halts and the condition causing each. Consult the program listing for further information. These errors are non-recoverable, the program must be restarted at location 00100.

<u>Address</u>		<u>Test and Cause</u>
02057	ISZ Test	Location 02201 should = 0
02063	ISZ Test	Location 02200 should = 1
02075	ROTATE 1 Test	Link should = 1
02102	ROTATE 1 Test	Data Failure
02113	ROTATE 2 Test	Link should = 0
02120	ROTATE 2 Test	Data failure
02130	SAD TEST	Sad Failure
02170	JMS TEST	JMS Failure

## 6.1.2 DECtape Test Error Halts

### 6.1.2.1 Selection Error

If no transports are selected by ACS 10 to 17, the computer will halt at location 00107 to allow the operator to set the ACS's and depress CONTINUE.

### 6.1.2.2 Read Data Error

A "Read Data" error will cause an error halt at location 00327. To recover, depress "CONTINUE".

### 6.1.2.3 Write Data Error

A "Write Data" error will cause an error halt at location 01064. To recover, depress "CONTINUE".

### 6.1.2.4 Read All Error

A "Read All" status error will cause an error halt at location 01677. To recover, depress "CONTINUE". "Read All" data errors will halt per 6.1.2.2.

### 6.1.2.5 Move Error

A "Move" error will cause an error halt at location 02351. To recover, depress "CONTINUE".

### 6.1.2.6 Search Error

A "Search" error will cause an error halt at location 02627. To recover, depress "CONTINUE".

## 6.2 Error Recovery (Non-Processor)

After an error timeout the processor will halt. To recover, depress "CONTINUE". For all errors, the program will attempt to repeat the same or a similar operation. If an error occurs a second time for the same operation, the timeout and halt will occur again. If "CONTINUE" is depressed this time, the drive in error will rewind and the random selections will commence at block 0 again. If, on the other hand, no error occurs, the program will continue normal operation.

## 6.2 Error Recovery (Non-Processor) Cont.

For read errors ("Read Data" and "Read All"), the direction of the second read operation to the original error's direction. A second error will cause another typeout and halt. Depressing "CONTINUE" will cause the drive to rewind to end zone. If no error occurs on the second pass, the block will be read a third time in the original direction. Any error will cause a typeout and halt. The end result of repeated errors is rewind to end zone and random selection starting with block 0 to prevent the same read sequence.

## 6.3 Error Typeouts

### 6.3.1 Search Errors

"Search" error typeouts contain the following information:

Drive Number

Search Direction and Mode

Block Wanted and Direction

Last Number Put in Memory by TC02

Last Block Number Found if More than Two Were Found

Number of Blocks Found Since Last Startup or Turn-Around

DECtape Status B

Example:

```
DRIVE 4
SEARCH FORWARD
000123 BLOCK WANTED FORWARD
000070 BLOCK FOUND
000073 LAST BLOCK
000004 BLOCKS READ
000100 STATUS B
```

## 6.1.2 DECtape Test Error Halts

### 6.1.2.1 Selection Error

If no transports are selected by ACS 10 to 17, the computer will halt at location 00107 to allow the operator to set the ACS's and depress CONTINUE.

### 6.1.2.2 Read Data Error

A "Read Data" error will cause an error halt at location 00327. To recover, depress "CONTINUE".

### 6.1.2.3 Write Data Error

A "Write Data" error will cause an error halt at location 01064. To recover, depress "CONTINUE".

### 6.1.2.4 Read All Error

A "Read All" status error will cause an error halt at location 01677. To recover, depress "CONTINUE". "Read All" data errors will halt per 6.1.2.2.

### 6.1.2.5 Move Error

A "Move" error will cause an error halt at location 02351. To recover, depress "CONTINUE".

### 6.1.2.6 Search Error

A "Search" error will cause an error halt at location 02627. To recover, depress "CONTINUE".

## 6.2 Error Recovery (Non-Processor)

After an error timeout the processor will halt. To recover, depress "CONTINUE". For all errors, the program will attempt to repeat the same or a similar operation. If an error occurs a second time for the same operation, the timeout and halt will occur again. If "CONTINUE" is depressed this time, the drive in error will rewind and the random selections will commence at block 0 again. If, on the other hand, no error occurs, the program will continue normal operation.



## 6.2 Error Recovery (Non-Processor) Cont.

For read errors ("Read Data" and "Read All"), the direction of the second read operation to the original error's direction. A second error will cause another timeout and halt. Depressing "CONTINUE" will cause the drive to rewind to end zone. If no error occurs on the second pass, the block will be read a third time in the original direction. Any error will cause a timeout and halt. The end result of repeated errors is rewind to end zone and random selection starting with block 0 to prevent the same read sequence.

## 6.3 Error Timeouts

### 6.3.1 Search Errors

"Search" error timeouts contain the following information:

Drive Number

Search Direction and Mode

Block Wanted and Direction

Last Number Put in Memory by TC02

Last Block Number Found if More than Two Were Found

Number of Blocks Found Since Last Startup or Turn-Around

DECtape Status B

Example:

```
DRIVE 4  
  
SEARCH    FORWARD  
  
000123    BLOCK WANTED FORWARD  
  
000070    BLOCK FOUND  
  
000073    LAST BLOCK  
  
000004    BLOCKS READ  
  
000100    STATUS B
```

### 6.3.1 Search Errors Cont.

The above example shows that the control was operating with Drive #4, searching in the forward direction for block #123. It encountered block #70 after block #73 (should have been #74). Up to this time it had read 4 blocks. DEC-tape status was normal. The block sequence was not consecutive, no doubt due to the loss of bit 15 of the block number.

### 6.3.2 Write Data Errors

"Write Data" error typeouts contain the following information:

Drive Number

Direction and Mode

DECtape Status B

Contents of Word Count Register (Location 00030).

Example:

DRIVE 1

WRITE DATA BACKWARD CONTINUOUS

000765 BLOCK

600000 STATUS B

777634 WC

The above example shows that a mark track error occurred while writing data backwards in the continuous mode in block #765.

### 6.3.3 Read Data Errors

"Read Data" error typeouts are of two types - status and data compare.

### 6.3.3.1 Status Error Typeouts

These typeouts contain the following information:

Drive Number

Direction and Mode

Block Being Read

DECtape Status B

Contents of WC if not 0 (Note: No data comparison if WC is not 0).

Example:

DRIVE 7

READ DATA BACKWARD CONTINUOUS

001065 BLOCK

440000 STATUS B

777400 WC

The above example shows that an attempt to read data from block #1065 on drive #7 in the backward direction in continuous mode caused a select error. Since WC is not 0, no data comparison was made.

### 6.3.3.2 Data Compare Errors

These typeouts contain the following information:

Drive Number

Direction

Mode

Block Read

### 6.3.3.2 Data Compare Errors Cont.

Correct Data

Erroneous Data

Address of Incorrect Data

Example:

DRIVE 5	
READ	DATA FORWARD
DATA	ERROR
001074	BLOCK
400000	COR
322667	INC
010374	ADDRS OF INC
000027	COR
067365	INC
010375	ADDRS OF INC

The above example shows that an attempt to read data in the forward direction from block #1074 resulted in data errors. In the first printout: the correct data was 400000, the data from tape was 322667, the incorrect data is in location 010374.

### 6.3.4 Read All Errors

"Read All" error typeouts are of three types - status, data compare, Checksum.

#### 6.3.4.1 Status Error Typeouts

Same format as "Read Data" - see 6.3.3.1.

#### 6.3.4.2 Data Compare Error Typeouts

Same format as "Read Data" - 6.3.3.2.

#### 6.3.4.3 Checksum Error Typeouts

These typeouts contain the following information:

Drive Number

Read All Direction and Mode

Block Number

Reverse Checksum for Tape

Data Checksum Calculated from 6-bit XOR of Data by the Program

LPB Calculated by Program (6-bit XOR of RC, DC, C above).

Example:

```
DRIVE 8
READ ALL    BACKWARD CONTINUOUS
000000     BLOCK
CKSUM      ERROR
000033     REV CHECKSUM
000022     DATA CHECKSUM CALCULATED
670000     CHECKSUM
000076     CALCULATED LPB
```

The above example shows a checksum error. The 6-bit complement XOR of the Reverse Checksum, Calculated Data Checksum and Checksum should be 000077. However, it calculated to 000076.

### 6.3.5 Program Interrupt Errors

The program detects 3 classes of program interrupt errors. After the PI timeout the program forces an operation timeout so that the function being performed may be determined.

#### 6.3.5.1 Program Interrupt and No DECTape Flag or Error Flag Skip

The following timeout will occur:

PI NO DECTAPE SKIP

#### 6.3.5.2 No Program Interrupt or DECTape Flag or Error Flag Skip after Timeout

If no PI or DECTape hardware flags occur after 5 seconds for all functions except "MOVE" (which allows 45 seconds) the following timeout will occur:

NO PI NO DECTAPE SKIP

#### 6.3.5.3 No Program Interrupt but DECTape Flag or Error Flag Skip after Timeout

If no PI occurs, but hardware flags do occur after timeout (5 or 45 seconds) the following timeout will occur:

NO PI DECTAPE SKIP

## 7. RESTRICTIONS

### 7.1 Starting Restrictions

None

### 7.2 Operating Restrictions

Do not "deselect" a transport merely to change a reel of DECTape and then reselect it. If a reel of DECTape must be changed, Stop the program, change the reel of tape, then depress I/O RESET then START with 00100 in the Address Switches.

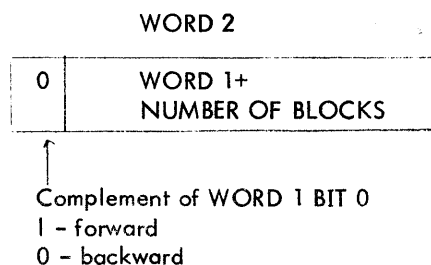
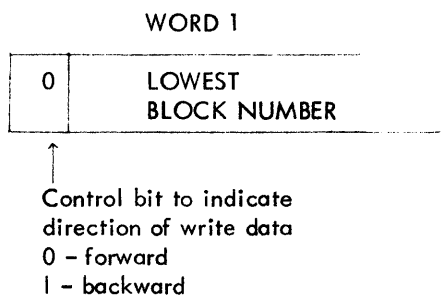
8. MISCELLANEOUS

8.1 Execution Time

Program does not stop unless an error occurs.

8.2 Data Format

The data blocks written by the Random Exerciser are formatted to be bi-directional. Whether written forward or backward, they may be read in either direction on the TC02 with the resultant data looking the same. The first four words of the block contain pertinent information about the block. The first two words of each block are formatted as follows:



WORD 1 indicates the lowest numbered block in the series with bit 0 indicating the written direction of the series of blocks.

WORD 2 is 1 greater than the highest numbered block in the series, bit 0 in WORD 2 is the complement of bit 0 in WORD 1.

## 8.2 Data Format Cont.

Ignoring bits 0 of the two words:

WORD 2 - WORD 1 = number of blocks in series.

WORD 3 and WORD 4 of the block are random numbers used to generate the rest of the data. WORD 5 of the block is WORD 3 rotated right one position using an 18-bit rotate.

WORD 6 is WORD 4 rotated right one position also using an 18-bit rotate. This process is repeated until WORD 128 is WORD 126 rotated right. WORD 129 is the complement obverse of WORD 128. WORD 130 is the complement obverse of WORD 127, etc., until the last word of the block is complement obverse of the first word of the block.

WORD 1	400617	Block 617 to
WORD 2	000626	625 inclusive were
WORD 3	732654	written backwards using
WORD 4	201356	these two words for data pattern
WORD 5	355326	WORD 3 Right One
WORD 6	100567	WORD 4 Right One, etc.
WORD 253	124675	Complement obverse of WORD 4
WORD 254	321540	Complement obverse of WORD 3
WORD 255	151777	Complement obverse of WORD 2
WORD 256	061773	Complement obverse of WORD 1

## 9. PROGRAM DESCRIPTION

### 9.1 Discussion

TC02 DECTAPE RANDOM EXERCISER will exercise a TC02 DECTape control and any configuration of from 1 to 8 TU55 DECTape Drives. Drive, direction of operation, number of blocks, and data patterns are by random selection. First the program randomly selects a drive, then a number between 1 and 32 (decimal) for the number of blocks, then the direction. There is one possibility in four that the direction will be backward. The number of blocks is added to or subtracted from the last block position of the drive selected. If the block generated has not been written, a write operation is initiated. If the last block table indicates that the block selected has already been written, a read operation is initiated. If READ is selected, the program then generates a random number between 0 and 7. If the number is 0, the block is read in "Read All Continuous" mode; if 4, "Read Data Normal" mode; if any other number, "Read Data Continuous" mode and two blocks are read. If WRITE is selected and the



9.1 Discussion Cont.

number of blocks is a multiple of 3 (3, 6, 9, 12, etc.), the blocks are written in continuous mode. Otherwise, the blocks are written in normal mode.

```

          .TITLE DTRE15
          .ABS
          /TC02 RANDOM EXERCISER - POP-15
004000 RANBFR=4000
004400 BUFFRS=RANBFR+400
000030 WC=30
000031 CA=31
00100 .LOC 100
00100 RANDEX LEM
00101 LAS /READ ACS AND
00102 AND RUBOUT
00103 SZA /SEE IF ANY TRANSPORTS
00104 JMP .+5 /WERE SPECIFIED
00105 LAC (MESS1 /CONTROL COMES HERE
00106 JMS MSPRNT /IF NONE
00107 XX
00110 JMP RANDEX
00111 DAC MSBITS
00112 CLA:CMA
00113 DAC FRSWAT
00114 707554 /CLEAR "B"
00115 JMS REPOSI /PUT ALL DRIVES IN END ZONE
00116 DZM* LSTBLK /CLEAR NUMBER OF BLOCKS
00117 JMS CHNGDR /CHECK TO SEE IF ALL WERE CLEARED
00120 JMP .-2 /NO, THEY WEREN'T
00121 LAC ONE
00122 DAC LSTDRV
00123 RDSWCH LAS /THIS SEQUENCE
00124 AND RUBOUT /ALLOWS THE OPERATOR
00125 SNA /TO DESELECT
00126 JMP RANDEX+5 /" BAD TRANSPORTS"
00127 DAC MSBITS /VIA THE AC SWITCHES
          /RANDOM SELECTION OF OPERATIONS
00130 MOFPRO JMS RANSEL /SELECT A DRIVE
00131 CLA:CMA
00132 DAC PASFLG
00133 LAC CDRIVE /CHECK TO SEE IF
00134 TAD LSTDRV /THIS DRIVE WAS THE
00135 SNA:CLA /LAST ONE SELECTED
00136 JMP SAMDRV /YES, IT WAS!
00137 LAC CDRIVE
00140 CMA
00141 TAD ONE
00142 DAC LSTDRV
00143 CLA:CMA
00144 DAC RSQFLG
          /SELECT NUMBER OF BLOCKS FOR OPERATION (1-32)
00145 MOFPR1 JMS RAND1 /GET A RANDOM NUMBER
00146 AND TERSEV
00147 CMA /MAKE -1 TO -32 (DEC)
00150 DAC NUMBLK
          /SELECT DIRECTION TO GO
00151 JMS RAND2 /GET ANOTHER RANDOM
00152 AND THREE
00153 SNA:CLA

```

```

00154 740001 CMA /SET BACKWARDS
00155 043446 DAC DIRFLG /DIRECTION FLAG
00156 223451 LAC* LSTBLK
00157 755200 SNA!CLA!CLL /HAS DRIVE BEEN WRITTEN ON?
00160 600721 JMP WRITE1 /NO, WRITE
00161 203446 LAC DIRFLG
/GENERATE BLOCK SELECT POSITION + OR - NUMBER
00162 750200 SZA!CLA
00163 744002 CLL!CML
00164 203455 LAC NUMBLK
00165 741400 SZL
00166 600171 JMP .+3
00167 740001 CMA
00170 343401 TAD ONE
00171 363467 TAD* POSITN
00172 043472 DAC RECORD
00173 741100 SPA /- RECORD?
00174 143472 DZM RECORD /MAKE = 0
/HAS BLOCK SELECTED BEEN WRITTEN
/IF NOT, WRITE OPERATION SELECTED
00175 203472 LAC RECORD
00176 740001 CMA
00177 363451 TAD* LSTBLK
00200 741100 SPA /HAS BLOCK BEEN WRITTEN?
00201 600721 JMP WRITE1 /NO
00202 100540 JMS RAND3
00203 503413 AND SEVEN
00204 741200 SNA
00205 601431 JMP RALLTS
00206 503425 AND THREE
00207 750200 SZA!CLA
00210 601221 JMP RDCMOD
00211 100253 RDTAB JMS RDDATA
00212 600272 JMP REREAD
/BLOCK HAS BEEN READ WITHOUT PARITY ERROR
00213 203432 LAC TYTHOU
00214 707544 707544 /STOP TAPE
00215 203472 LAC RECORD
00216 063467 DAC* POSITN /NEW POSITION
00217 203446 LAC DIRFLG
00220 063445 DAC* DIRECT /DIRECTION READ
/MOVE FIRST FOUR WORDS TO REGENERATE DATA PATTERN
00221 100400 JMS PREGEN /GENERATE DATA PATTERN
00222 777400 LAW -400
00223 100626 JMS CODATA /COMPARE 400 WORDS
00224 004000 RANBFR /GOOD DATA HERE
00225 004400 RUFFRS /? DATA HERE
00226 443442 ISZ COCNTR
00227 600325 JMP REREDA /REREAD BLOCK IF ERROR
00230 600123 JMP RDSWCH /GO BACK, READ ACS, SELECT NEW DRIVE
/DRIVE SELECTED SAME AS LAST TIME, TFST FOR SERIES
00231 100516 SAMDRV JMS RAND1 /GET A RANDOM
00232 741100 SPA /READ THIS STRING?
00233 600146 JMP MOFPR1+1 /FIND A NEW BLOCK
00234 443474 ISZ RSQFLG

```

```

00235 600146      JMP MOFPR1+1
00236 503425      AND THREE
00237 751200      SNA:CLA /FORWARD?
00240 740001      CMA /NO, GO BACKWARD
00241 043446      DAC DIRFLG
00242 740001      CMA
00243 343401      TAD ONE
00244 343407      TAD RBUFST /FORWARD GETS
00245 043472      DAC RECORD /FIRST BLOCK
00246 203446      LAC DIRFLG /BACKWARD GETS
00247 363472      TAD* RECORD /LAST BLOCK
00250 503373      AND INDEP /MASK TO 10 BITS
00251 043472      DAC RECORD /AND STORE
00252 601070      JMP RDSEQ /GO READ
/READ DATA SUBROUTINE (1 BLOCK)
/DIRFLG=777777 (BACKWARD), 0 (FORWARD)
RDDATA 0
00253 000000      JMS SEARCH
00254 102353      LAC TRETHO /3000
00255 203426      707544 /CHANGE SEARCH TO READ DATA
00256 707544
00257 777400      LAW -400
00260 040030      DAC WC /INITIALIZE WORD COUNT
00261 203364      LAC BF1LOC
00262 040031      DAC CA /AND CURRENT ADDRESS
00263 101701      JMS WATINT /WAIT FOR COMPLETION
00264 000100      100 /DTF
00265 620253      JMP* RDDATA /STATUS ERROR EXIT
00266 200030      LAC WC
00267 751200      SNA:CLA
00270 440253      ISZ RDDATA
00271 620253      JMP* RDDATA /NORMAL EXIT
/TYPE STATUS ERROR ON READ 1 BLOCK
/TEST FOR DATA ERRORS IF FULL READ
REREAD CLA:CMA
00272 750001      DAC PASFLG
00273 043457      JMS ERSTP /STOP TAPE
00274 102726      JMS TYRDAT
00275 102754      LAC RECORD
00276 203472      JMS OPRINT /TYPE BLOCK NUMBER
00277 102670      LAC (MESS2
00300 203511      JMS MSPRNT /TYPE "BLOCK"
00301 102633      707572 /READ "B"
00302 707572
00303 102670      JMS OPRINT /TYPE CONTENTS OF "B"
00304 203512      LAC (MESS3
00305 102633      JMS MSPRNT /TYPE "STATUS B"
00306 200030      LAC WC
00307 741200      SNA /SHORT BUFFER?
00310 600315      JMP .+5 /NO
00311 102670      JMS OPRINT /TYPE C(WC)
00312 203513      LAC (MESS4
00313 102633      JMS MSPRNT /TYPE "W.C."
00314 600325      JMP REREDA
00315 100400      JMS PREGEN /REGENERATE PATTERN
00316 777405      LAW -373
00317 340031      TAD CA

```

00320	040324	DAC .+4	
00321	777400	LAW -400	
00322	100626	JMS CODATA	
00323	004000	RANBFR	
00324	004400	BUFFRS	
00325	750004	REREDA	LAS
00326	751100	SPA:CLA	/HALT ON ERRORS?
00327	740040	XX	/YES
00330	443457	ISZ PASFLG	
00331	600361	JMP REWDRV	/REWIND DRIVE IF NOT FIRST ERROR
00332	203446	LAC DIRFLG	
00333	740001	CMA	
00334	043446	DAC DIRFLG	/CHANGE TO READ IN OPPOSITE DIRECTION
00335	100253	JMS RDATA	/READ DATA
00336	600274	JMP REREAD+2	/ERROR, AGAIN, TRY AGAIN
00337	100400	JMS PREGEN	/REGENERATE PATTERN
00340	777400	LAW -400	
00341	100626	JMS CODATA	/COMPARE DATA
00342	004000	RANBFR	
00343	004400	BUFFRS	
00344	443442	ISZ COCNTR	/DONE ALL BLOCKS?
00345	600325	JMP REREDA	/NO
00346	203446	LAC DIRFLG	
00347	740001	CMA	
00350	043446	DAC DIRFLG	/CHANGE DIRECTION
00351	100253	JMS RDATA	/READ AGAIN
00352	600274	JMP REREAD+2	/ANOTHER STATUS ERROR
00353	777400	LAW -400	
00354	100626	JMS CODATA	/COMPARE DATA READ
00355	004000	RANBFR	
00356	004400	BUFFRS	
00357	443442	ISZ COCNTR	/DONE ALL BLOCKS?
00360	600325	JMP REREDA	/NO
00361	203410	REWDRV	LAC REWIND
00362	243506	XOR UNFUNC	
00363	707545	707545	/CLEAR AND LOAD "A"
00364	101701	JMS WATINT	
00365	500000	500000	/ERROR, END OF TAPE
00366	600376	JMP .+10	
00367	163451	DZM* LSTBLK	/CLEAR LAST BLOCK
00370	163467	DZM* POSITN	/AND POSITION FOR DRIVE
00371	750001	CLA:CMA	
00372	063445	DAC* DIRECT	/SET DIRECTION FOR DRIVE BACKWARD
00373	203401	LAC ONE	
00374	043452	DAC LSTDRV	/SET LAST DRIVE TO 1
00375	600123	JMP RDSWCH	/GO BACK ALMOST TO BEGINNING
00376	102340	JMS MOVER+2	
00377	600361	JMP REWDRV	
			/MOVE FIRST 4 WORDS OF BLOCK READ
			/AND REGENERATE COMPARE PATTERN
00400	000000	PREGEN	0
00401	203405	LAC RBFLOC	
00402	040010	DAC 10	
00403	777774	LAW -4	
00404	043440	DAC CNTR	

```

00405 220031 LAC* CA
00406 100467 JMS MCOMOB
00407 060010 DAC* 10
00410 750001 CLA!CMA
00411 340031 TAD CA
00412 040031 DAC CA
00413 443440 ISZ CNTR
00414 600405 JMP .-7
00415 100417 JMS PATGEN
00416 620400 JMP* PREGEN
/GENERATE RANDOM DATA PATTERN
/FIRST FOUR WORDS OF PATTERN ARE IN
/RANDOM BUFFER WORDS 1 TO 4
PATGEN 0
00417 000000 LAW -4
00420 777774 DAC CNTR /INITIALIZE A COUNTER
00421 043440 LAC RBUFST
00422 203407 DAC PNTR1 /AND TWO POINTERS
00423 043463 LAC RBUFND
00424 203406 DAC PNTR2 /FOR WORD TRANSFERS
00425 043464 LAC TEM2
00426 203502 DAC TEM1 /MOVE TEM1
00427 043501 LAC* PNTR1
00430 223463 DAC TEM2 /AND TEM2
00431 043502 JMS MCOMOB /FORM COMPLEMENT OBVERSE
00432 100467 DAC* PNTR2 /AND STORE IN RUFFER
00433 063464 ISZ PNTR1 /INCREMENT FORWARD ADDRESS
00434 443463 CLA!CMA
00435 750001 TAD PNTR2 /DECREMENT BACKWARD
00436 343464 DAC PNTR2
00437 043464 ISZ CNTR /DONE 4?
00440 443440 JMP .-13 /NO
00441 600426 LAW -174
00442 777604 DAC CNTR /FOR 174 (OCT) MORE
00443 043440 GNPATR LAC TEM1
00444 203501 CLL!RAR /FORM NEXT WORD
00445 744020 SZL /OF RANDOM
00446 741400 XOR MUCH0 /PATTERN
00447 243400 DAC TEM3
00450 043503 DAC* PNTR1 /STORE IN BUFFER
00451 063463 JMS MCOMOB
00452 100467 DAC* PNTR2 /STORE COMP OBVERSE
00453 063464 LAC TEM2 /MOVE
00454 203502 DAC TEM1 /A
00455 043501 LAC TEM3 /FEW
00456 203503 DAC TEM2 /WORDS
00457 043502 ISZ PNTR1 /INCREMENT FORWARD ADDRESS
00460 443463 CLA!CMA
00461 750001 TAD PNTR2 /DECREMENT BACKWARD ADDRESS
00462 343464 DAC PNTR2
00463 043464 ISZ CNTR /DONE ALL?
00464 443440 JMP GNPATR /NO
00465 600444 JMP* PATGEN
00466 620417 /COMPLEMENT OBVERSE SUBROUTINE
MCOMOB 0
00467 000000

```

```

00470 744001 CMA:CLL
00471 043454 DAC NUMBER
00472 143456 DZM OBVERS
00473 777772 LAW -6
00474 043477 DAC TALLY
00475 203514 LAC (SEVEN
00476 043465 DAC PNTR3
00477 203454 LAC NUMBER
00500 741000 SKP
00501 742010 LOOP RTL
00502 742010 RTL
00503 742010 RTL
00504 043454 DAC NUMBER
00505 523465 AND* PNTR3
00506 243456 XOR OBVERS
00507 043456 DAC OBVERS
00510 443465 ISZ PNTR3
00511 203454 LAC NUMBER
00512 443477 ISZ TALLY
00513 600501 JMP LOOP
00514 203456 LAC OBVERS
00515 620467 JMP* MCOMOR
/RANDOM NUMBER GENERATORS
00516 000000 RAND1 0
00517 200525 LAC R1
00520 744010 RAL:CLL
00521 741400 SZL
00522 340526 TAD R1+1
00523 040525 DAC R1
00524 620516 JMP* RAND1
00525 000171 R1 171
00526 000003 3
00527 000000 RAND2 0
00530 200536 LAC R2
00531 744010 RAL:CLL
00532 741400 SZL
00533 340537 TAD R2+1
00534 040536 DAC R2
00535 620527 JMP* RAND2
00536 000015 R2 15
00537 000003 3
00540 000000 RAND3 0
00541 200547 LAC R3
00542 744010 RAL:CLL
00543 741400 SZL
00544 340550 TAD R3+1
00545 040547 DAC R3
00546 620540 JMP* RAND3
00547 000031 R3 31
00550 000003 3
/RANDOMLY SELECT A DRIVE
/STAYS IN THIS ROUTINE UNTIL A
/DRIVE IS FOUND AND DIRECTION AND
/POSITION POINTERS ARE GENERATED
00551 000000 RANSEL 0

```

```

00552 100527 JMS RAND2 /GFT A RANDOM NUMBER
00553 503413 AND SEVEN /MASK OFF UNWANTED BITS
00554 043436 DAC CDRIVE /SAVE
00555 744020 CLL:RAR
00556 742020 RTR
00557 740020 RAR
00560 043506 DAC UNFUNC /DRIVE NUMBER
00561 203436 LAC CDRIVE
00562 740001 CMA
00563 043440 DAC CNTR
00564 203430 LAC TWOHUN /FIND POSITION
00565 443440 ISZ CNTR /OF
00566 741000 SKP /BIT
00567 600572 JMP EXIST /WITH RESPECT
00570 744020 CLL:RAR /TO
00571 600565 JMP .-4 /ACS
00572 503453 EXIST AND MSBITS /ASK WITH ACS
00573 741200 SNA /DOES UNIT EXIST (TEST UNIT)?
00574 600552 JMP RANSEL+1 /NO
00575 203436 LAC CDRIVE /YES
00576 343403 TAD POSTBL
00577 043467 DAC POSITN /FORM POSITION POINTER
00600 203436 LAC CDRIVE
00601 343366 TAD DIRTBL
00602 043445 DAC DIRECT /FORM DIRECTION POINTER
00603 203436 LAC CDRIVE
00604 343374 TAD LSTTBL
00605 043451 DAC LSTBLK /LAST BLOCK WRITTEN POINTER
00606 620551 JMP* RANSEL
/MOVE DATA SUBROUTINE
/# OF WORDS IN AC, ORIGIN JMS+1
/DESTINATION JMS+2
MVDATA 0
00607 000000 DAC CNTR
00610 043440 CLA:CMA
00611 750001 TAD* MVDATA
00612 360607 DAC 10
00613 040010 ISZ MVDATA
00614 440607 CLA:CMA
00615 750001 TAD* MVDATA
00616 360607 DAC 11
00617 040011 ISZ MVDATA
00620 440607 LAC* 10
00621 220010 DAC* 11
00622 060011 ISZ CNTR
00623 443440 JMP .-3
00624 600621 JMP* MVDATA
00625 620607 /COMPARE DATA SUBROUTINE
/# OF WORDS IN AC, CORRECT ADDRESS IN JMS+1
/QUESTIONABLE ADDRESS IN JMS+2
CODATA 0
00626 000000 DAC CNTR
00627 043440 LAC* CODATA /GET CORRECT ADDRESS
00630 220626 DAC PNTR1
00631 043463 ISZ CODATA
00632 440626

```



```

00633 220626          LAC* CODATA          /GET QUESTIONABLE ADDRESS
00634 043464          DAC PNTR2
00635 750001          CLA!CMA
00636 043442          DAC COCNTR          /SET NO ERROR FLAG
00637 440626          ISZ CODATA
00640 223463          COLOOP            LAC* PNTR1
00641 740001          CMA
00642 343401          TAD ONE
00643 363464          TAD* PNTR2
00644 740200          SZA
00645 600653          JMP COERRO
00646 443463          ISZ PNTR1
00647 443464          ISZ PNTR2
00650 443440          ISZ CNTR
00651 600640          JMP COLOOP
00652 620626          JMP* CODATA
00653 443442          COERRO            ISZ COCNTR          /FIRST ERROR?
00654 600673          JMP COERR1        /NO
00655 102726          JMS ERSTP /STOP TAPE
00656 707552          707552 /READ "A"
00657 503402          AND ONETHO
00660 741200          SNA /"ALL" MODE?
00661 600664          JMP .+3 /NO
00662 102765          JMS TYRALL        /YES, GO TYPE
00663 741000          SKP
00664 102754          JMS TYRDAT
00665 203515          LAC (MESS5        /TYPE "DATA ERROR"
00666 102633          JMS MSPRNT
00667 203472          LAC RECORD
00670 102670          JMS OPRINT        /TYPE BLOCK NUMBER
00671 203516          LAC (MESS2
00672 102633          JMS MSPRNT        /TYPE "BLOCK"
00673 750004          COERR1            LAS
00674 742010          RTL
00675 740100          SMA /SHOULD ONLY 4 BE TYPED
00676 600703          JMP .+5 /NO, TYPE ALL ERRORS
00677 777774          LAW -4 /YES HAVE
00700 343442          TAD COCNTR        /4 BEEN
00701 740100          SMA /TYPED?
00702 600646          JMP COLOOP+6     /YES, DELETE REST OF TEST
00703 102714          JMS CRLF
00704 223463          LAC* PNTR1
00705 102670          JMS OPRINT        /TYPE STANDARD
00706 203517          LAC (MESS6
00707 102633          JMS MSPRNT        /AND "COR"
00710 223464          LAC* PNTR2
00711 102670          JMS OPRINT        /TYPE INCORRECT
00712 203520          LAC (MESS7
00713 102633          JMS MSPRNT        /AND "INC"
00714 203464          LAC PNTR2
00715 102670          JMS OPRINT        /TYPE ADDRESS OF INCORRECT
00716 203521          LAC (MESS8
00717 102633          JMS MSPRNT        /TYPE "ADDR"
00720 600646          JMP COLOOP+6
/INITIATE WRITE OPERATIONS

```

```

/GENERATE PATTERN WORDS AND
/BLOCK NUMBERS
WRITE1      LAC RBFLOC
DAC 10      /SET UP 10 FOR STORAGE
LAC* LSTBLK
DAC* 10     /SET UP FIRST WORD
LAC NUMBLK
CMA
TAD ONE
TAD* LSTBLK      /LAST BLOCK+1
DAC RECORD
LAW -1100
TAD RECORD
SMA          /REWIND UNIT?
JMP REWCK /YES
LAC RECORD
DAC* 10
JMS RAND3
DAC* 10     /FIRST RANDOM WORD
JMS RAND1
DAC* 10     /SECOND RANDOM WORD
JMS RAND2
AND THREE
SNA:CLA     /FORM DIRECTION
CMA
DAC DIRFLG      /AND STORE
TAD RBUFST
DAC 10
DAC 11        /FORM FIRST WORD
CLA:CLL:CML     /INDICATE
RAR          /BACKWARD
TAD* 10       /OR 2ND WORD
DAC* 11      /INDICATE FORWARD
JMS PATGEN     /GENERATE 256 WORD PATTERN
LAC NUMBLK
DAC SAVNUM     /SAVE NUMBER OF BLOCKS
CORT1R      LAC NUMBLK
CMA
DAC RECORD
LAC DIRFLG
SZA          /BACKWARD IS
LAC RECORD     /LAST BLOCK + NUMBER OF BLOCKS
TAD* LSTBLK
DAC RECORD     /TO FIND FIRST BLOCK TO BE WRITTEN
/TEST FOR WRITE DATA CONTINUOUS MODE
/IF NUMBER OF BLOCKS = INC OF 3
JMP WDCMOD
JMS SEARCH
WRT1A      LAC FIVTHO      /SEARCH TO WRITE DATA
707544     /XOR INTO "A"
LAW -400
DAC WC      /SET UP WC
LAC RBFLOC
DAC CA      /AND CA
JMS WATINT     /WAIT
00721      203405
00722      040210
00723      223451
00724      060010
00725      203455
00726      740001
00727      343401
00730      363451
00731      043472
00732      776700
00733      343472
00734      740100
00735      601034
00736      203472
00737      060010
00740      100540
00741      060010
00742      100516
00743      060010
00744      100527
00745      503425
00746      751200
00747      740001
00750      043446
00751      343407
00752      040010
00753      040011
00754      754002
00755      740020
00756      360010
00757      060011
00760      100417
00761      203455
00762      043475
00763      203455
00764      740001
00765      043472
00766      203446
00767      740200
00770      203472
00771      363451
00772      043472
00773      601332
00774      102353
00775      203370
00776      707544
00777      777400
01000      040030
01001      203405
01002      040031
01003      101701

```

01004	000100	100	/DTF
01005	601042	JMP	WSTERR
01006	200030	LAC	WC
01007	744200	SZA:CLL	/WC = 0?
01010	601042	JMP	WSTERR /NO
01011	203446	LAC	DIRFLG
01012	343472	TAD	RECORD
01013	741100	SPA	
01014	754002	CLA:CLL:CML	
01015	043472	DAC	RECORD
01016	750400	SNL:CLA	/RECORD -1?
01017	443472	ISZ	RECORD /NO +1
01020	443455	ISZ	NUMBLK /DONE ALL BLOCKS?
01021	600776	JMP	WRT1A+1 /NO, DO ANOTHER
01022	203472	WDINC	LAC RECORD
01023	063467	DAC*	POSITION /NEW POSITION
01024	203446	LAC	DIRFLG
01025	063445	DAC*	DIRECT /NEW DIRECTION
01026	204001	LAC	RANBFR+1
01027	503373	AND	INDEP
01030	063451	DAC*	LSTBLK /NEW LAST BLOCK
01031	203432	LAC	TYTHOU
01032	707544	707544	/XOR INTO "A" (STOP TAPE)
01033	600123	JMP	RDSWCH /GET C(SR) THEN SELECT AGAIN
01034	100540	REWCK	JMS RAND3 /GET RANDOM NUMBER
01035	741100	SPA	/+ OR - ?
01036	600361	JMP	REWDRV /(-)
01037	143472	DZM	RECORD /READ BLOCK 0
01040	143446	DZM	DIRFLG /FORWARD
01041	600211	JMP	RDTAB
01042	102726	WSTERR	JMS ERSTP /STOP TAPE - LEAVE FLAGS
01043	102776	JMS	TYWDAT
01044	203472	LAC	RECORD
01045	102670	JMS	OPRINT /TYPE BLOCK NUMBER
01046	203522	LAC	(MESS2
01047	102633	JMS	MSPRNT /TYPE "BLOCK"
01050	203475	LAC	SAVNUM
01051	043455	DAC	NUMBLK
01052	707572	707572	/READ "B"
01053	102670	JMS	OPRINT /PRINT C(B)
01054	203523	LAC	(MESS3
01055	102633	JMS	MSPRNT /TYPE "STATUS B"
01056	200030	LAC	WC
01057	102670	JMS	OPRINT /PRINT C(WC)
01060	203524	LAC	(MESS4
01061	102633	JMS	MSPRNT /TYPE "WC"
01062	750004	LAS	
01063	751100	SPA:CLA	/STOP ON ERROR?
01064	740040	XX	/YES
01065	443457	ISZ	PASFLG /2ND ERROR?
01066	600361	JMP	REWDRV /YES, REWIND
01067	600763	JMP	CORT1R /NO
			/READ ENTIRE SEQUENCE OF BLOCKS
			/AS ORIGINALLY WRITTEN
01070	204000	RDSEQ	LAC RANBFR /GET LOWEST BLOCK

01071	503373	AND INDEP	
01072	043472	DAC RECORD	
01073	204001	LAC RANBFR+1	/GET LAST BLOCK+1
01074	503373	AND INDEP	
01075	043455	DAC NUMBLK	/MAKE NUMBER OF BLOCKS
01076	203472	LAC RECORD	
01077	740001	CMA	
01100	343455	TAD NUMBLK	
01101	043455	DAC NUMBLK	/SET UP NUMBLK
01102	100516	JMS RAND1	/GET A RANDOM
01103	755100	SPA:CLA:CLL	/READ FORWARD?
01104	740003	CMA:CML	/NO BACKWARD, AC= -0, L=1
01105	043446	DAC DIRFLG	
01106	203472	LAC RECORD	
01107	741400	SZL	
01110	343455	TAD NUMBLK	
01111	043472	DAC RECORD	
01112	203455	LAC NUMBLK	/MAKE NUMBER OF BLOCKS NEGATIVE
01113	740001	CMA	
01114	043455	DAC NUMBLK	
01115	100253	JMS RDDATA	/READ FIRST BLOCK
01116	600272	JMP REREAD	/STATUS ERROR IN READ
01117	443455	RDRUF1 ISZ NUMBLK	/READ ALL BLOCKS?
01120	601123	JMP .+3	/NO
01121	203432	LAC TYTHOU	/GET CONSTANT TO STOP TAPE
01122	601130	JMP .+6	
01123	203365	LAC BF2LOC	
01124	040031	DAC CA	/SET UP CA
01125	777400	LAW -400	
01126	040030	DAC WC	/AND WC
01127	750000	CLA	
01130	707544	707544	/XOR INTO "A"
01131	777400	LAW -400	
01132	100626	JMS CODATA	/COMPARE DATA
01133	004000	RANBFR	
01134	004400	BUFFRS	
01135	443442	ISZ COCNTR	/ANY DATA ERRORS?
01136	600325	JMP REREDA	/YES, READ OTHER DIRECTION
01137	203455	LAC NUMBLK	
01140	741200	SNA	/DONE COMPLETE SERIES
01141	601214	JMP NDOFRD	/YES
01142	203446	LAC DIRFLG	/NO
01143	741200	SNA	
01144	343401	TAD ONE	
01145	343472	TAD RECORD	/LAST BLOCK +1 OR -1
01146	043472	DAC RECORD	
01147	101701	JMS WATINT	/WAIT
01150	000100	100	/DTF
01151	600272	JMP REREAD	/STATUS ERROR, REREAD
01152	200030	LAC WC	
01153	740200	SZA	/WC = 0?
01154	600272	JMP REREAD	/NO, REREAD
01155	443455	ISZ NUMBLK	/DONE ALL READS?
01156	601161	JMP .+3	/NO
01157	203432	LAC TYTHOU	/YES, GET READ TO

01160	601165	JMP .+6 /STOP TAPE
01161	203364	LAC BF1LOC
01162	040031	DAC CA /SET UP CA
01163	777400	LAW -400
01164	040030	DAC WC /AND WC
01165	750000	CLA
01166	707544	707544 /XOR INTO A
01167	777400	LAW -400
01170	100626	JMS CODATA /COMPARE DATA
01171	004000	RANBFR
01172	005001	BUFFRS+401
01173	443442	ISZ COCNTR /ANY COMPARE ERRORS
01174	600325	JMP REREDA /YES, READ OPPOSITE DIRECTION
01175	203455	LAC NUMBLK
01176	741200	SNA /READ AND COMPARE ALL?
01177	601214	JMP NDOFRD /YES
01200	203446	LAC DIRFLG /NO
01201	741200	SNA
01202	343401	TAD ONE
01203	343472	TAD RECORD /LAST BLOCK +1 OR -1
01204	043472	DAC RECORD
01205	101701	JMS WATINT /WAIT
01206	000100	100 /DTF
01207	600272	JMP REREAD /STATUS ERROR, REREAD
01210	200030	LAC WC
01211	740200	SZA /WC = 0?
01212	600272	JMP REREAD /NO, REREAD
01213	601117	JMP RDBUF1 /COMPARE FIRST BUFFER
01214	203472	NDOFRD LAC RECORD
01215	063467	DAC* POSITN /NEW POSITION
01216	203446	LAC DIRFLG
01217	063445	DAC* DIRECT /NEW DIRECTION
01220	600123	JMP RDSWCH /GO BACK FOR ANOTHER
		/READ DATA CONTINUOUS MODE
		/TWO BLOCKS AND COMPARE DATA READ
01221	203446	RDCMOD LAC DIRFLG
01222	740200	SZA /FORWARD?
01223	601232	JMP .+7 /NO
01224	203472	LAC RECORD
01225	740001	CMA
01226	363451	TAD* LSTBLK /BLOCK +1 WRITTEN?
01227	740200	SZA
01230	601235	JMP .+5
01231	600211	JMP RDTAB /BLOCK NOT WRITTEN, READ ONLY 1
01232	203472	LAC RECORD
01233	741200	SNA
01234	600211	JMP RDTAB /BLOCK 0 BACKWARDS, READ ONLY 1
01235	102353	JMS SEARCH /FIND FIRST BLOCK
01236	203427	LAC TRTETH /13000
01237	707544	707544 /XOR INTO "A" (CHANGED TO READ CONT)
01240	777000	LAW -1000
01241	040030	DAC WC /SET UP WC AND
01242	203364	LAC BF1LOC
01243	040031	DAC CA /CA FOR 2 BLOCKS
01244	101701	JMS WATINT /WAIT FOR END OF TRANSFER

01245	002100	100	/DTF
01246	601314	JMP RDCERR	/ERROR, SEE WHICH BLOCK
01247	200030	LAC WC	
01250	740200	SZA	/WC = 0?
01251	600272	JMP REREAD	/NO
01252	203432	LAC TYTHOU	/GET "STOP"
01253	707544	707544	/XOR INTO "A"
01254	203446	LAC DIRFLG	
01255	741200	SNA	
01256	343401	TAD ONE	
01257	343472	TAD RECORD	
01260	043472	DAC RECORD	
01261	100400	JMS PREGEN	/VERIFY LAST
01262	777400	LAW -400	/BLOCK
01263	100626	JMS CODATA	/FIRST
01264	004000	RANBFR	
01265	005000	RUFFRS+400	
01266	443442	ISZ COCNTR	/ANY DATA ERRORS
01267	600325	JMP REREDA	/YES, REREAD
01270	750001	CLA!CMA	
01271	343365	TAD BF2LOC	
01272	040031	DAC CA	
01273	100400	JMS PREGEN	
01274	203446	LAC DIRFLG	
01275	741200	SNA	
01276	343401	TAD ONE	
01277	740001	CMA	
01300	343401	TAD ONE	
01301	343472	TAD RECORD	
01302	043472	DAC RECORD	
01303	777400	LAW -400	
01304	100626	JMS CODATA	
01305	004000	RANBFR	
01306	004400	RUFFRS	
01307	443442	ISZ COCNTR	/ANY DATA ERRORS?
01310	600325	JMP REREDA	/YES, REREAD
01311	203472	LAC RECORD	/NO, SET NEW
01312	063467	DAC* POSITN	/POSITION
01313	600123	JMP RDSWCH	/GO BACK FOR ANOTHER JOB
01314	200030	RDCERR	
01315	741200	LAC WC	
01316	601324	SNA	/ERROR IN 2ND BLOCK?
01317	740001	JMP .+6	/YES, COUNT BLOCK
01320	343401	CMA	
01321	340777	TAD ONE	
01322	040030	TAD WRT1A+2	/-400 (OCT)
01323	600272	DAC WC	
01324	203446	JMP REREAD	/TYPE STATUS ERROR
01325	741200	LAC DIRFLG	
01326	343401	SNA	/BACKWARD (-1)
01327	343472	TAD ONE	/FORWARDS (+1)
01330	043472	TAD RECORD	
01331	600272	DAC RECORD	
		JMP REREAD	

/WRITE DATA CONTINUOUS MODE IF  
/NUMBER OF BLOCKS IS AN INCREMENT OF 3

01332	143505	WDCMOD	DZM TRECTR
01333	203455	LAC NUMBLK	
01334	343425	TAD THREE	
01335	741200	SNA	/AN INCREMENT OF 3?
01336	601343	JMP SWCMOD	/YES, WRITE CONT.
01337	740100	SMA	/GONE PAST 0?
01340	600774	JMP WRT1A-1	/YES, NOT INC. OF 3
01341	443505	ISZ TRECTR	/NO, COUNT 1
01342	601334	JMP WDCMOD+2	/AND REPEAT
01343	203505	SWCMOD	LAC TRECTR
01344	740001	CMA	
01345	043505	DAC TRECTR	
01346	777000	LAW -1000	/MAKE PATTERN
01347	100607	JMS MVDATA	/3 BUFFERS
01350	004000	RANBFR	/LONG
01351	004400	RUFFRS	
01352	102353	JMS SEARCH	
01353	203367	LAC FIFTHO	/15000 (SEARCH TO WRITE CONT)
01354	707544	SWCMDL	707544 /XOR INTO "A"
01355	776400	LAW -1400	/SET UP WC
01356	040030	DAC WC	/FOR 3 BLOCKS
01357	203405	LAC RBFLOC	
01360	040031	DAC CA	/AND CA
01361	101701	JMS WATINT	/WAIT
01362	000100	100	/DTF
01363	601405	JMP WDCERR	/NOT NORMAL INTERRUPT
01364	200030	LAC WC	
01365	740200	SZA	/WC = 0?
01366	601405	JMP WDCERR	/NO
01367	203446	LAC DIRFLG	
01370	754200	SZA:CLA:CLL	/FORWARD?
01371	740002	CML	/NO, SET LINK
01372	343425	TAD THREE	/GET 3
01373	740400	SNL	/BACKWARD
01374	601377	JMP .+3	/NO
01375	740001	CMA	/YES, -3
01376	343401	TAD ONE	
01377	343472	TAD RECORD	
01400	043472	DAC RECORD	
01401	750000	CLA	
01402	443505	ISZ TRECTR	/DONE ALL
01403	601354	JMP SWCMDL	/NO, DO NEXT 3
01404	601022	JMP WDINC	/YES
01405	143505	WDCERR	DZM TRECTR
01406	200030	LAC WC	
01407	740001	CMA	
01410	343401	TAD ONE	/ -WC
01411	340777	TAD WRT1A+2	/ -400 (OCT)
01412	741100	SPA	
01413	601416	JMP .+3	
01414	443505	ISZ TRECTR	
01415	601411	JMP .-4	
01416	203446	LAC DIRFLG	
01417	754200	SZA:CLA:CLL	/FORWARD?
01420	740002	CML	/NO, BACKWARD, MAKE -1 OR -2

```

01421 203505 LAC TRECTR
01422 740400 SNL
01423 601426 JMP .+3
01424 740001 CMA
01425 343401 TAD ONE
01426 343472 TAD RECORD
01427 043472 DAC RECORD /BLOCK IN ERROR
01430 601042 JMP WSTERR /TYPE OUT WRITE ERROR
/READ ALL ROUTINE
/REV CKSM, DATA, AND CKSM ARE READ
/CKSMS ARE GENERATED AND TESTED
01431 203446 RALLTS LAC DIRFLG
01432 741200 SNA /BACKWARDS?
01433 343401 TAD ONE /NO, FORWARD
01434 740001 CMA
01435 343401 TAD ONE /+1 FOR BACKWARD, -1 FOR FORWARD
01436 343472 TAD RECORD
01437 043472 DAC RECORD
01440 741100 SPA
01441 600123 JMP RDSWCH /CAN'T DO 0
01442 343376 TAD MBLOCK
01443 750100 SMA:CLA
01444 600145 JMP MOFPR1 /OR 1100
/FIND THE BLOCK AND CHANGE TO READ ALL
/CLEAR READ ALL ERROR FLAG
01445 143471 DZM RAEFLG /CLEAR READ ALL ERROR FLAG
01446 102353 JMS SEARCH /FIND BLOCK
01447 707554 707554 /CLEAR DTF AND EF
01450 203446 LAC DIRFLG
01451 741200 SNA
01452 343401 TAD ONE
01453 343472 TAD RECORD
01454 043472 DAC RECORD
01455 707561 707561 /SKIP ON ERROR FLAG
01456 741000 SKP
01457 601556 JMP RASERR /DECTAPE ERROR
01460 707601 707601 /SKIP ON DECTAPE FLAG
01461 601455 JMP .-4 /NO FLAGS, CHECK AGAIN
01462 203435 LAC BLKFND /GET CURRENT DECTAPE BLOCK NUMBER
01463 740001 CMA
01464 343401 TAD ONE
01465 343472 TAD RECORD
01466 750200 SZA:CLA /IS IT THE ONE WE ARE LOOKING FOR?
01467 601556 JMP RASERR /NO, ERROR
/CHANGE TO READ ALL CONTINUOUS
01470 203431 LAC TWVTHO /SEARCH TO READ ALL CONT
01471 707544 707544 /XOR INTO "A"
01472 777374 LAW -404
01473 040030 DAC WC /SET WC FOR 404 WORDS
01474 203364 LAC BF1LOC
01475 040031 DAC CA /SET CA
01476 101701 JMS WATINT /WAIT
01477 000100 100 /DTF
01500 601657 JMP ERRRAR /READ ALL STATUS ERROR
01501 200030 LAC WC

```



```

01502 740200      SZA          /WC = 0?
01503 601657      JMP ERRRAR          /NO
01504 203432      LAC TYTHOU          /STOP TAPE
01505 707544      707544          /XOR INTO "A"
/GENERATE CKSMS AND TEST SUM = 00
01506 203404      LAC RADLOC
01507 040010      DAC 10           /SETUP 10
01510 777400      LAW -400
01511 043440      DAC CNTR        /AND COUNTER FOR
01512 143437      DZM CKSUMR      /PROCESSING, ZERO CHECKSUM
01513 204402      LAC RUFFRS+2
01514 503421      AND SEVSEV
01515 043473      DAC REVCHK      /SAVE REVERSE CHECK
01516 220010      LAC* 10         /GET NEXT DATAWORD
01517 101560      JMS XORSUM      /XOR INTO CHECKSUM
01520 443440      ISZ CNTR        /DONE?
01521 601516      JMP .-3         /NO
01522 203437      LAC CKSUMR      /YES, GET FINAL
01523 503421      AND SEVSEV      /MASK
01524 043444      DAC DATASM      /AND SAVE
01525 770000      LAW 10000       /LOAD AC WITH 770000
01526 520010      AND* 10         /COMBINE WITH FORWARD CKSM
01527 043450      DAC FWDCHK      /AND SAVE
01530 101560      JMS XORSUM      /COMBINE FWDCHK AND DATASM
01531 203473      LAC REVCHK
01532 101560      JMS XORSUM      /COMBINE WITH REVCHK
01533 203437      LAC CKSUMR
01534 740001      CMA
01535 503421      AND SEVSEV
01536 740200      SZA          /IS RESULT 0?
01537 601602      JMP ERRCKS      /NO, ERROR
RADCHK 203472      LAC RECORD
01541 063467      DAC* POSITN     /UPDATE POSITION TABLE
01542 750001      CLA!CMA
01543 340031      TAD CA
01544 040031      DAC CA          /SUBTRACT 1 FROM CA
01545 100400      JMS PREGEN      /GENERATE PATTERN
01546 777400      LAW -400        /COMPARE AGAINST
01547 100626      JMS CODATA      /PATTERN OBTAINED
01550 004000      RANBFR          /FROM TAPE
01551 004403      RUFFRS+3
01552 443471      ISZ RAEFLG      /ANY READ ALL ERRORS
01553 443442      ISZ COCNTR      /NO, ANY DATA ERRORS
01554 600325      JMP REREDA      /YES, YES
01555 600121      JMP RDSWCH-2    /NO, GO BACK
RASERR 043457      DZM PASFLG
01557 602562      JMP SRHERR
/TC02 RANDOM EXERCISER - TAPE 2
/FORM 6-BIT XOR OF AC AND CKSUMR
/SAVE RESULTS IN CKSUMR
XORSUM 0
01560 000000
01561 740001      CMA
01562 043501      DAC TEM1
01563 777775      LAW -3
01564 043477      DAC TALLY

```

01565 203501  
 01566 744010  
 01567 742010  
 01570 742010  
 01571 742010  
 01572 043501  
 01573 503421  
 01574 243437  
 01575 043437  
 01576 203501  
 01577 443477  
 01600 601567  
 01601 621560

LAC TEM1  
 RAL:CLL  
 RTL  
 RTL  
 RTL  
 DAC TEM1  
 AND SEVSEV  
 XOR CKSUMR  
 DAC CKSUMR  
 LAC TEM1  
 ISZ TALLY  
 JMP .-11  
 JMP\* XORSUM

/CHECKSUM ERROR TYPEOUT  
 ERRCKS JMS TYRALL /TYPE OUT HEADER

01602 102765  
 01603 203472  
 01604 102670  
 01605 203525  
 01606 102633  
 01607 203526  
 01610 102633  
 01611 102714  
 01612 203473  
 01613 102670  
 01614 203527  
 01615 102633  
 01616 101647  
 01617 102714  
 01620 203444  
 01621 102670  
 01622 760240  
 01623 102661  
 01624 103030  
 01625 101647  
 01626 101653  
 01627 102714  
 01630 203450  
 01631 102670  
 01632 101647  
 01633 102714  
 01634 203437  
 01635 740001  
 01636 503421  
 01637 043437  
 01640 102670  
 01641 101653  
 01642 203530  
 01643 102633  
 01644 750001  
 01645 043471  
 01646 601540  
 01647 000000  
 01650 203531  
 01651 102633  
 01652 621647

LAC RECORD  
 JMS OPRINT /TYPE OUT BLOCK NUMBER  
 LAC (MESS2  
 JMS MSPRNT /TYPE "BLOCK"  
 LAC (MESS9  
 JMS MSPRNT /TYPE "CKSUM ERROR"  
 JMS CRLF  
 LAC REVCHK  
 JMS OPRINT /TYPE OUT REVCHK  
 LAC (MESS10  
 JMS MSPRNT /TYPE "REV"  
 JMS CKSTYP /TYPE "CHKSUM"  
 JMS CRLF  
 LAC DATASM  
 JMS OPRINT /TYPE DATA SUM  
 LAW 240  
 JMS TYPE  
 JMS TYDATA  
 JMS CKSTYP /TYPE "CHKSUM"  
 JMS CALCTY  
 JMS CRLF  
 LAC FWDCHK  
 JMS OPRINT /TYPE FORWARD CHECKSUM  
 JMS CKSTYP /TYPE "CHKSUM"  
 JMS CRLF  
 LAC CKSUMR  
 CMA  
 AND SEVSEV  
 DAC CKSUMR  
 JMS OPRINT /TYPE CALCULATED CKSM  
 JMS CALCTY  
 LAC (MESS11  
 JMS MSPRNT /TYPE "L.P.B."  
 CLA:CMA  
 DAC RAEFLG  
 JMP RADCHK  
 CKSTYP 0  
 LAC (MESS12  
 JMS MSPRNT  
 JMP\* CKSTYP

```

01653 000000 CALCTY 0
01654 203532 LAC (MESS13
01655 102633 JMS MSPRNT
01656 621653 JMP* CALCTY
/READ ALL STATUS ERROR TIMEOUT
ERRRAR JMS ERSTP /STOP TAPE
01657 102726 JMS TYRALL /TYPE HEADER
01660 102765 LAC RECORD
01661 203472 JMS OPRINT /TYPE OUT BLOCK NUMBER
01662 102670 LAC (MESS2
01663 203533 JMS MSPRNT /TYPE "BLOCK"
01664 102633 707572 /READ "B"
01665 707572 JMS OPRINT /TYPE OUT B
01666 102670 LAC (MESS3
01667 203534 JMS MSPRNT /TYPE "STATUS B"
01670 102633 LAC WC
01671 200030 JMS OPRINT /TYPE OUT WORD COUNT
01672 102670 LAC (MESS4
01673 203535 JMS MSPRNT /TYPE "W.C."
01674 102633 LAS
01675 750004 SPA:CLA /STOP ON ERROR?
01676 751100 XX /YES
01677 740040 JMP REWDRV
01700 600361 /WAIT FOR INTERRUPT ROUTINE
/SYNCHRONIZE PROCESSOR TESTS
WATINT 0
01701 000000 LAC (JMP IRECD
01702 203536 DAC 1
01703 040001 LAW -4
01704 777774 DAC WTCNTR /INITIALIZE COUNTER FOR 5 SECONDS
01705 043507 707552 /READ "A"
01706 707552 AND SEVEN+3 /7000
01707 503416 SZA /IS FUNCTION "MOVE"
01710 740200 JMP .+3 /NO, 5 SEC. OK
01711 601714 LAW -30 /CHANGE COUNTER
01712 777750 DAC WTCNTR /TO 45 SEC.
01713 043507 ISZ FRSWAT /FIRST PROGRAM WAIT?
01714 443447 JMP RESETO /NO, RESTORE AC AND L
01715 601721 JMS SETIDX /YES, SET UP
01716 102015 ION /TURN ON PI
01717 700042 JMP ISZTST /GO TO FIRST TEST
01720 602046 RESETO DZM FRSWAT /INDICATE NOT FIRST WAIT
01721 143447 ISZ PICNTR /COUNT PI COUNTER
01722 443460 SKP /NOT ZERO
01723 741000 JMS SETIDX /ZERO
01724 102015 LAC 0
01725 200000 DAC* PIDEX1 /SAVE OLD PC
01726 063461 ISZ PIDEX1
01727 443461 LAC 0
01730 200000 RAL /RESTORE LINK
01731 740010 LAC ACCUM /RESTORE AC
01732 203433 ION /TURN ON PI
01733 700042 JMP* 0 /EXIT
01734 620000 /INTERRUPT RETURNS HERE
IRECD DAC ACCUM /SAVE AC
01735 043433

```

01736	707601	707601	/SKIP ON DTF
01737	741000	SKP	/NONE
01740	601743	JMP .+3	/DECTAPE FLAG
01741	707561	707561	/SKIP ON ERROR FLAG
01742	601762	JMP NDTSKP	/NO DECTAPE FLAGS
01743	707572	707572	/READ "B"
01744	043476	DAC SBRECD	/SAVE B
01745	221701	LAC* WATINT	
01746	740001	CMA	/COMPARE FLAGS SET
01747	503476	AND SBRECD	/AGAINST FLAGS ALLOWED
01750	741200	SNA	/ANY ILLEGAL?
01751	441701	ISZ WATINT	/NO, INCREMENT TWICE
01752	441701	ISZ WATINT	/YES, INCREMENT ONCE
01753	201701	LAC WATINT	
01754	063462	DAC* PIDEK2	/SAVE C(WATINT)
01755	443462	ISZ PIDEK2	/MORE POINTER
01756	203435	LAC BLKFND	/CLEAN UP
01757	503373	AND INDEP	/BLOCK NUMBER
01760	043435	DAC BLKFND	/FROM TAPE
01761	621701	JMP* WATINT	/EXIT
		.EJECT	

```

01762 102726 NDTSKP JMS ERSTP /STOP TAPE
01763 203537 LAC (MESS14
01764 102633 JMS MSPRNT /TYPE "PI"
01765 602005 JMP PTSTNS
01766 000000 PTSTND 0
01767 443507 ISZ WTCNTR /WAITED LONG ENOUGH?
01770 621766 JMP* PTSTND /NO, DO NEXT TEST
01771 700012 IOF+10 /YES, TURN OFF PI
01772 707601 707601 /SKIP ON DTF
01773 741000 SKP
01774 601777 JMP .+3
01775 707561 707561 /SKIP ON ERROR FLAG
01776 740001 CMA /NO FLAGS, STORE -1
01777 043476 DAC SBRECD /OR 0 IF ANY FLAG
02000 102726 JMS ERSTP /STOP TAPE
02001 203540 LAC (MESS15
02002 102633 JMS MSPRNT /TYPE "NO PI"
02003 443476 ISZ SBRECD /ANY DECTAPE FLAGS
02004 602007 JMP PTSTNS+2 /YES
02005 203541 PTSTNS LAC (MESS16
02006 102633 JMS MSPRNT /TYPE "NO"
02007 203542 LAC (MESS17
02010 102633 JMS MSPRNT /TYPE "DECTAPE SKIP"
02011 441701 ISZ WATINT
02012 750001 CLA!CMA
02013 043447 DAC FRSWAT
02014 621701 JMP* WATINT
02015 000000 SETIDX 0
02016 777772 LAW -6
02017 043460 DAC PICNTR
02020 202025 LAC PITBL
02021 043461 DAC PIDX1
02022 202045 LAC PIRTBL
02023 043462 DAC PIDX2
02024 622015 JMP* SETIDX
02025 002026 PITBL .+1
02045 .LOC PITBL+20
02045 002035 PIRTBL .-10
/PROCESSOR TESTS, RUN WHILE WAITING FOR INTERRUPT
/ISZ TEST - 2.6 SECONDS
02046 750001 ISZTST CLA!CMA
02047 042204 DAC TEST5
02050 142201 DZM TEST2
02051 142200 DZM TEST1
02052 442200 ISZ TEST1
02053 442201 ISZ TEST2
02054 602052 JMP .-2
02055 202201 LAC TEST2
02056 740200 SZA
02057 740040 XX
02060 750001 CLA!CMA
02061 342200 TAD TEST1
02062 740200 SZA
02063 740040 XX
02064 442204 ISZ TEST5

```

02065	741000	SKP
02066	602051	JMP ISZTST+3
02067	101766	JMS PTSTND
		/ROTATE 1 TEST - 3.9 SECONDS
02070	202201	ROT1TS LAC TEST2
02071	744002	CLL!CML
02072	740020	RAR
02073	740010	RAL
02074	740400	SNL
02075	740040	XX
02076	740001	CMA
02077	343401	TAD ONE
02100	342201	TAD TEST2
02101	740200	SZA
02102	740040	XX
02103	442201	ISZ TEST2
02104	602070	JMP ROT1TS
02105	101766	JMS PTSTND
		/ROTATE 2 TEST - 3.9 SECONDS
02106	202201	ROT2TS LAC TEST2
02107	744000	CLL
02110	742010	RTL
02111	742020	RTR
02112	741400	SZL
02113	740040	XX
02114	740001	CMA
02115	343401	TAD ONE
02116	342201	TAD TEST2
02117	740200	SZA
02120	740040	XX
02121	442201	ISZ TEST2
02122	602106	JMP ROT2TS
02123	101766	JMS PTSTND
		/SAD TEST - 2.1 SECONDS
02124	142203	SADTST DZM TEST4
02125	202203	LAC TEST4
02126	542203	SAD TEST4
02127	741000	SKP
02130	740040	XX
02131	442203	ISZ TEST4
02132	602125	JMP SADTST+1
02133	101766	JMS PTSTND
		/JMS TEST - 3 SECONDS
02134	776200	JMSTST LAW -1600
02135	042200	DAC TEST1 /1600 LOOPS
02136	777600	LAW -200
02137	042201	DAC TEST2 /200 JMS .
02140	203543	LAC (6000
02141	042202	DAC TEST3 /FIRST ADDRESS OF JMS
02142	202205	LAC JMSTST
02143	042203	DAC TEST4
02144	745000	SKP!CLL
02145	202203	LAC TEST4
02146	062202	DAC* TEST3
02147	442203	ISZ TEST4

```

02150 442202          ISZ TEST3
02151 442201          ISZ TEST2
02152 602145          JMP .-5
02153 202206          LAC RETJMP
02154 062202          DAC* TEST3
02155 606000          JMP 6000
02156 777600          JMPRET          LAW -200
02157 042201          DAC TEST2
02160 402140          XCT JMSTST+4
02161 042202          DAC TEST3
02162 741000          SKP
02163 202202          LAC TEST3
02164 740001          CMA
02165 362202          TAD* TEST3
02166 503375          AND MASK
02167 740200          SZA
02170 740040          XX
02171 442202          ISZ TEST3
02172 442201          ISZ TEST2
02173 602163          JMP .-10
02174 442200          ISZ TEST1
02175 602136          JMP JMSTST+2
02176 101766          JMS PTSTND
02177 602046          JMP ISZTST
02200 000000          TEST1 0
02201 000000          TEST2 0
02202 000000          TEST3 0
02203 000000          TEST4 0
02204 000000          TEST5 0
02205 106000          JMSSCON          JMS 6000
02206 602156          RETJMP          JMP JMPRET
          /REWIND ALL SELECTED DRIVES TO REVERSE END ZONE
02207 000000          REPOSI          0
02210 102225          JMS RSFORV          /RESET POINTERS FOR FIRST DRIVE
02211 203377          LAC MOVBAK          /GET MORE BACKWARDS
02212 243506          XOR UNFUNC          /COMBINE WITH UNIT NUMBER
02213 707545          707545          /CLEAR AND LOAD A
02214 101701          JMS WATINT          /WAIT
02215 500000          500000          /EXPECT END ZONE AND EF
02216 602336          JMP MOVER          /MOVE ERROR (INCORRECT STATUS)
02217 750001          CLA!CMA
02220 063467          DAC* POSITN          /INDICATE END ZONE
02221 063445          DAC* DIRECT          /AND BACKWARDS
02222 102244          JMS CHNGDR          /SET UP NEXT DRIVE
02223 602211          JMP REPOSI+2          /REWIND NEXT DRIVE
02224 622207          JMP* REPOSI          /ALL DRIVES IN END ZONE, EXIT
          /RESET CURRENT DRIVE POINTER TO FIRST DRIVE SELECTED
02225 000000          RSFORV          0
02226 143436          DZM CORIVE          /ZERO CURRENT DRIVE
02227 203430          LAC TWOHUN
02230 043443          DAC COMBIT          /SET UP COMBIT
02231 203453          LAC MSBITS
02232 503443          AND COMBIT
02233 750200          SZA!CLA          /IS THIS DRIVE SELECTED
02234 602242          JMP RSFOR1          /YES, SET POINTER

```

```

02235 203443 LAC COMBIT
02236 744020 CLL!RAR /MOVE DRIVE BIT RIGHT
02237 043443 DAC COMBIT
02240 443436 ISZ CDRIVE /INCREMENT DRIVE NUMBER
02241 602231 JMP RSFDRV+4
/HAVE FOUND FIRST DRIVE SELECTED
02242 102263 RSFDR1 JMS GNPTRS /GENERATE CONTROL POINTERS
02243 622225 JMP* RSFDRV /EXIT
/SELECT NEXT DRIVE OR RESET TO FIRST AND SKIP
02244 000000 CHNGDR 0
02245 203443 LAC COMBIT /GET DRIVE BIT
02246 744020 CLL!RAR /MOVE RIGHT
02247 740400 SNL /CHECKED ALL DRIVES
02250 602254 JMP .+4 /NO
02251 102225 JMS RSFDRV /YES RESET TO FIRST
02252 442244 ISZ CHNGDR /INCREMENT
02253 622244 JMP* CHNGDR /AND EXIT
02254 043443 DAC COMBIT
02255 443436 ISZ CDRIVE
02256 503453 AND MSBITS
02257 751200 SNA!CLA /THIS DRIVE SELECTED?
02260 602245 JMP CHNGDR+1 /NO
02261 102263 JMS GNPTRS /GENERATE DRIVE POINTERS
02262 622244 JMP* CHNGDR /EXIT
/GENERATE LAST RECMD, DIRECTION AND UNIT NUMBER
/POINTERS FOR DECTAPE FUNCTIONS
02263 000000 GNPTRS 0
02264 203436 LAC CDRIVE /GET DRIVE NUMBER
02265 744020 CLL!RAR
02266 740020 RAR
02267 742020 RTR /REPOSITION FOR STORAGE
02270 043506 DAC UNFUNC /IN UNIT FUNCTION REGISTER
02271 203436 LAC CDRIVE
02272 342303 TAD PNTRS
02273 043467 DAC POSITN /SET UP POSITION
02274 203436 LAC CDRIVE
02275 342314 TAD PNTRS+11
02276 043445 DAC DIRECT /DIRECTION AND
02277 203436 LAC CDRIVE
02300 342325 TAD PNTRS+22
02301 043451 DAC LSTBLK /LAST BLOCK BLOCK POINTERS
02302 622263 JMP* GNPTRS
02303 002304 PNTRS .+1
02314 .LOC .+10
02314 002315 .+1
02325 .LOC .+10
02325 002326 .+1
02336 .LOC .+10
/MOVE ERROR
02336 102340 MOVER JMS .+2
02337 602211 JMP REPOSI+2
02340 000000 0
02341 102726 JMS ERSTP /STOP TAPE
02342 102734 JMS TYMOVE /TYPE OUT HEADER
02343 707572 /READ "B"

```



```

02344 102670 JMS OPRINT /TYPE OUT B
02345 203544 LAC (MESS3
02346 102633 JMS MSPRNT /TYPE "STATUS R"
02347 750004 LAS
02350 751100 SPA:CLA
02351 740040 XX
02352 622340 JMP* MOVER+2
/SEARCH ROUTINE - FIND BLOCK INDICATED
/BY THE CONTENTS OF "RECORD" IN THE DIRECTION
/INDICATED BY "DIRFLG", 0-FWD, -1-BKWD
02353 000000 SEARCH 0
02354 203446 LAC DIRFLG
02355 754200 SZA:CLA:CLL /FORWARD?
02356 740002 CML /NO, BACKWARD
02357 203425 LAC THREE /COMPUTE
02360 741400 SZL /TURN
02361 602364 JMP .+3 /AROUND
02362 740001 CMA /POINT
02363 343401 TAD ONE /FOR
02364 343472 TAD RECORD /DECTAPE
02365 740001 CMA /SEARCH
02366 343401 TAD ONE /AND
02367 043500 DAC TAPONT /STORE
02370 203446 LAC DIRFLG
02371 751200 SNA:CLA /FWD OR BKWD?
02372 203371 LAC FTYTHO /BKWD (40,000)
02373 343412 TAD SEAFOR /SEARCH FORWARD
02374 343506 TAD UNFUNC /COMBINE IN UNIT NUMBER
02375 707545 707545 /CLEAR AND LOAD "A"
02376 203372 LAC IDCON /BLKFND
02377 040031 DAC CA /SET UP CA
02400 750001 CLA:CMA
02401 043434 DAC BLKFLG /SET UP BLKFLG
02402 101701 JMS WATINT
02403 000100 100 /DTF
02404 602503 JMP SREZTS /ERROR, SEE IF END ZONE
02405 102441 JMS SRCONT
02406 602412 JMP SRTARN /BLOCK = BLOCK FOUND
02407 602412 JMP SRTARN /GONE PAST BLOCK
02410 707554 707544+10 /CLEAR AC, XOR A
02411 602402 JMP .-7 /HAVEN'T REACHED BLOCK YET
02412 707552 SRTARN 707552 /READ "A"
02413 740001 CMA
02414 503432 AND TYTHOU /MOTION BIT
02415 343371 TAD FTYTHO /40,000
02416 707544 707544 /XOR "A" (TURN AROUND)
02417 203472 LAC RECORD
02420 740001 CMA
02421 343401 TAD ONE
02422 043500 DAC TAPONT /SET UP TAPONT
02423 750001 CLA:CMA
02424 043434 DAC BLKFLG /AND BLKFLG
02425 101701 JMS WATINT /WAIT
02426 000100 100 /DTF
02427 602562 JMP SRHERR /ERROR

```

```

02430 102441 JMS SRCONT /CHECK RELATION OF BLOCK
02431 622353 JMP* SEARCH /FOUND BLOCK, EXIT
02432 602562 JMP SRHERR
02433 750001 CLA:CMA
02434 343434 TAD BLKFLG
02435 751200 SNA:CLA /READ 2 BLOCK NUMBERS
02436 602533 JMP SRCMOD /YES, CHANGE TO CONTINUOUS MODE
02437 707544 707544 /XOR INTO "A" (CLEAR FLAGS)
02440 602425 JMP .-13
/FIND RELATION OF BLOCK FOUND TO BLOCK
/SOUGHT AND TEST BLOCK NUMBERS TO BE CONSECUTIVE
SRCONT 0
02441 000000 ISZ BLKFLG /FIRST BLOCK NUMBER?
02442 443434 SKP /NOT FIRST
02443 741000 JMP SRFBLK /FIRST, JUST SAVE IT
02444 602461 707552 /READ "A"
02445 707552 AND FTYTHO /40,000
02446 503371 SZA:CLA /BACKWARDS
02447 750200 CMA /YES, -1
02450 740001 SNA
02451 741200 TAD ONE /FORWARD, +1
02452 343401 TAD PREBLK /LAST BLOCK +1 OR -1
02453 343470 CMA
02454 740001 TAD ONE
02455 343401 TAD BLKFND
02456 343435 SZA:CLA /BLOCKS CONSECUTIVE
02457 750200 JMP SRHERR /NO, ERROR
02460 602562 SRFBLK LAC BLKFND
02461 203435 DAC PREBLK
02462 043470 TAD TAPONT
02463 343500 SNA /FOUND BLOCK?
02464 741200 JMP* SRCONT /YES
02465 622441 ISZ SRCONT
02466 442441 SPA:CLA:CLL /BLKFND GREATER
02467 755100 CML /NO, LESS
02470 740002 707552 /READ A
02471 707552 AND FTYTHO /40,000
02472 503371 SZL
02473 741400 JMP .+4
02474 602500 SZA:CLA /IF FORWARD HAVE GONE PAST
02475 750200 ISZ SRCONT /BACKWARD
02476 442441 JMP* SRCONT
02477 622441 SNA:CLA
02500 751200 ISZ SRCONT
02501 442441 JMP* SRCONT
02502 622441 SREZTS 707572 /READ "B"
02503 707572 RTL /MOVE EZ BIT INTO 0
02504 742010 SMA:CLA:CLL /END ZONE INTERRUPT?
02505 754100 JMP SRHERR /NO, SOME OTHER
02506 602562 LAC TAPONT
02507 203500 SMA /BLOCK 0 OR 1
02510 740100 JMP .+4
02511 602515 TAD MBLOCK
02512 343376 SMA:CLA /BLOCK 1077 OR 1100?
02513 750100 JMP .+14 /NO, TURN AROUND
02514 602530

```

02515	750104	I AS
02516	742010	RTL
02517	740010	RAL
02520	750100	SMA:CLA /NEW FORMAT TAPE?
02521	602412	JMP SRTARN /YES, TURN AROUND
02522	203432	LAC TYTHOU /20,000
02523	707544	707544 /XOR INTO "A" (GO AGAIN)
02524	101701	JMS WATINT /WAIT FOR EZ
02525	500000	500000 /ERROR, END ZONE
02526	602562	JMP SRHERR
02527	602412	JMP SRTARN
02530	443434	ISZ BLKFLG /NOT FIRST INTERRUPT
02531	602562	JMP SRHERR /EZ IS ERROR
02532	602412	JMP SRTARN /TURN AROUND
02533	203435	SRCMOD LAC BLKFND /FIND DIFFERENCE
02534	740001	CMA /IN NUMBER
02535	343401	TAD ONE /OF
02536	343472	TAD RECORD /BLOCKS
02537	741100	SPA
02540	602543	JMP .+3
02541	740001	CMA /MAKE NEGATIVE
02542	343401	TAD ONE
02543	040030	DAC WC /FOR WORD COUNT
02544	203423	LAC TENTHO /10,000 (CONTINUOUS)
02545	707544	707544 /XOR INTO "A"
02546	101701	JMS WATINT
02547	000100	100 /DTF
02550	602562	JMP SRHERR /NOT NORMAL INTERRUPT
02551	203435	LAC BLKFND
02552	740001	CMA
02553	343401	TAD ONE
02554	343472	TAD RECORD
02555	750200	SZA:CLA /CORRECT BLOCK?
02556	602562	JMP SRHERR /NO, ERROR
02557	203423	LAC TENTHO /10,000
02560	707544	707544 /XOR INTO "A" (CLEAR CONT)
02561	622353	JMP* SEARCH
		/SEARCH ERROR TYPEOUT
02562	102726	SRHERR JMS ERSTP /STOP TAPE
02563	102743	JMS TYSRCH /TYPE HEADER
02564	203472	LAC RECORD
02565	102670	JMS OPRINT /TYPE BLOCK NUMBER
02566	203545	LAC (MESS18
02567	102633	JMS MSPRNT /TYPE "BLOCK WANTED
02570	203446	LAC DIRFLG
02571	751200	SNA:CLA
02572	602575	JMP .+3
02573	103100	JMS TYBKW /TYPE BACKWARDS
02574	741000	SKP
02575	103104	JMS TYFWD /TYPE FORWARD
02576	102714	JMS CRLF
02577	203435	LAC BLKFND
02600	102670	JMS OPRINT /TYPE BLOCK FOUND
02601	203546	LAC (MESS19
02602	102633	JMS MSPRNT /TYPE "BLOCK FOUND!

02603	750001	CLA: CMA	
02604	343434	TAD BLKFLG	
02605	751100	SPA: CLA	
02606	602613	JMP .+5	
02607	203470	LAC PREBLK	
02610	102670	JMS OPRINT	/TYPE PREVIOUS BLOCK
02611	203547	LAC (MESS20	
02612	102633	JMS MSPRNT	/TYPE "LAST BLOCK"
02613	443434	ISZ BLKFLG	
02614	740000	NOP	
02615	203434	LAC BLKFLG	
02616	102670	JMS OPRINT	/TYPE OUT QUANTITY OF BLOCKS
02617	203550	LAC (MESS21	
02620	102633	JMS MSPRNT	/TYPE "BLOCKS READ"
02621	707572	707572 /READ "B"	
02622	102670	JMS OPRINT	/TYPE OUT B
02623	203551	LAC (MESS3	
02624	102633	JMS MSPRNT	/TYPE "STATUS B"
02625	750004	LAS	
02626	751100	SPA: CLA /HALT ON ERROR?	
02627	740040	XX /YES	
02630	443457	ISZ PASFLG	/SECOND ERROR?
02631	600361	JMP REWDRV	/YES, REWIND DRIVE
02632	602354	JMP SEARCH+1	/NO, SEARCH AGAIN

/TYPE TEXT ROUTINE  
MSPRNT 0

02633	000000	AND (7777	
02634	503552	DAC PNTR4	
02635	043466	LAS	
02636	750004	RAL	
02637	740010	SPA: CLA	
02640	751100	JMP* MSPRNT	
02641	622633	LAC* PNTR4	
02642	223466	RTR	
02643	742020	RTR	
02644	742020	RTR	
02645	742020	RTR	
02646	742020	RTR	
02647	740020	RAR	
02650	102661	JMS TYPE	
02651	543411	SAD RUBOUT	
02652	622633	JMP* MSPRNT	
02653	223466	LAC* PNTR4	
02654	102661	JMS TYPE	
02655	543411	SAD RUBOUT	
02656	622633	JMP* MSPRNT	
02657	443466	ISZ PNTR4	
02660	602636	JMP MSPRNT+3	
02661	000000	TYPE 0	
02662	503411	AND RUBOUT	
02663	700406	TLS	
02664	700401	TSF	
02665	602664	JMP .-1	
02666	700402	TCF	
02667	622661	JMP* TYPE	

/TYPE OUT CONTENTS OF AC IN OCTAL

```

02670 000000 OPRINT 0
02671 043524 DAC TEM4
02672 750004 LAS
02673 740010 RAL
02674 751100 SPA:CLA
02675 622670 JMP* OPRINT
02676 777772 LAW -6
02677 043441 DAC CNTR1
02700 203504 LAC TEM4
02701 744010 RAL:CLL
02702 740010 RAL
02703 742010 RTL
02704 043524 DAC TEM4
02705 503413 AND SEVEN
02706 243363 XOR ASCII
02707 102661 JMS TYPE
02710 203504 LAC TEM4
02711 443441 ISZ CNTR1
02712 602702 JMP .-10
02713 622670 JMP* OPRINT
02714 000000 CRLF 0
02715 750004 LAS
02716 740010 RAL
02717 751100 SPA:CLA
02720 622714 JMP* CRLF
02721 760215 LAW 215
02722 102661 JMS TYPE
02723 760212 LAW 212
02724 102661 JMS TYPE
02725 622714 JMP* CRLF
/STOP TAPE ON ERROR, LEAVE FLAGS SET
02726 000000 ERSTP 0
02727 707552 /READ "A"
02730 503432 AND TYTHOU /STOP
02731 343411 TAD RUBOUT /DON'T CLEAR DTF AND EF
02732 707544 /XOR INTO "A"
02733 622726 JMP* ERSTP
/TYPE "MOVE" AND DIRECTION
02734 000000 TYMOVE 0
02735 103040 JMS TYDRV
02736 203553 LAC (MESS22
02737 102633 JMS MSPRNT
02740 103056 JMS TYDIR
02741 102714 JMS CRLF
02742 622734 JMP* TYMOVE
/TYPE SEARCH DIRECTION AND MODE
02743 000000 TYSRCH 0
02744 103040 JMS TYDRV
02745 102714 JMS CRLF
02746 203554 LAC (MESS23
02747 102633 JMS MSPRNT
02750 103056 JMS TYDIR
02751 103070 JMS TYMODE
02752 102714 JMS CRLF
02753 622743 JMP* TYSRCH

```

```

      /TYPE "READ DATA" DIRECTION AND MODE
02754 000000 TYRDAT 0
02755 103040 JMS TYDRV
02756 102714 JMS CRLF
02757 103020 JMS TYREAD
02760 103030 JMS TYDATA
02761 103056 JMS TYDIR
02762 103070 JMS TYMODE
02763 102714 JMS CRLF
02764 622754 JMP* TYRDAT
      /TYPE "READ ALL" DIRECTION AND MODE
02765 000000 TYRALL 0
02766 103040 JMS TYDRV
02767 102714 JMS CRLF
02770 103020 JMS TYREAD
02771 103034 JMS TYALL
02772 103056 JMS TYDIR
02773 103070 JMS TYMODE
02774 102714 JMS CRLF
02775 622765 JMP* TYRALL
      /TYPE "WRITE DATA" DIRECTION AND MODE
02776 000000 TYWDAT 0
02777 103040 JMS TYDRV
03000 102714 JMS CRLF
03001 103024 JMS TYWRIT
03002 103030 JMS TYDATA
03003 103056 JMS TYDIR
03004 103070 JMS TYMODE
03005 102714 JMS CRLF
03006 622776 JMP* TYWDAT
      /TYPE "WRITE ALL" DIRECTION AND MODE
03007 000000 TYWALL 0
03010 103040 JMS TYDRV
03011 102714 JMS CRLF
03012 103024 JMS TYWRIT
03013 103034 JMS TYALL
03014 103056 JMS TYDIR
03015 103070 JMS TYMODE
03016 102714 JMS CRLF
03017 623007 JMP* TYWALL
      /TYPE "READ"
03020 000000 TYREAD 0
03021 203555 LAC (MESS24
03022 102633 JMS MSPRNT
03023 623020 JMP* TYREAD
      /TYPE "WRITE"
03024 000000 TYWRIT 0
03025 203556 LAC (MESS25
03026 102633 JMS MSPRNT
03027 623024 JMP* TYWRIT
      /TYPE "DATA"
03030 000000 TYDATA 0
03031 203557 LAC (MESS26
03032 102633 JMS MSPRNT
03033 623030 JMP* TYDATA

```

```

      /TYPE "ALL"
03034 000000 TYALL 0
03035 203560 LAC (MESS27
03036 102633 JMS MSPRNT
03037 623034 JMP* TYALL
      /TYPE DRIVE AND NUMBER
03040 000000 TYDRV 0
03041 750004 LAS
03042 740010 RAL
03043 751100 SPA:CLA
03044 623040 JMP* TYDRV
03045 203561 LAC (MESS28
03046 102633 JMS MSPRNT
03047 203436 LAC CDRIVE
03050 741200 SNA
03051 343422 TAD TEN
03052 243363 XOR ASCII
03053 102661 JMS TYPE
03054 102714 JMS CRLF
03055 623040 JMP* TYDRV
      /TYPE "FORWARD" OR "BACKWARD"
03056 000000 TYDIR 0
03057 707552 /READ "A"
03060 503371 AND FTYTHO /400000
03061 741200 SNA
03062 603065 JMP .+3
03063 203562 LAC (MESS29 /BACKWARD
03064 741000 SKP
03065 203563 LAC (MESS30 /FORWARD
03066 102633 JMS MSPRNT
03067 623056 JMP* TYDIR
      /TYPE "CONTINUOUS" IF NOT NORMAL MODE
03070 000000 TYMODE 0
03071 707552 /READ "A"
03072 503423 AND TENTHO /10000
03073 741200 SNA
03074 623070 JMP* TYMODE
03075 203564 LAC (MESS31
03076 102633 JMS MSPRNT /TYPE "CONTINUOUS"
03077 623070 JMP* TYMODE
      /TYPE "BACKWARD"
03100 000000 TYBKW 0
03101 203565 LAC (MESS29
03102 102633 JMS MSPRNT
03103 623100 JMP* TYBKW
      /TYPE "FORWARD"
03104 000000 TYFWD 0
03105 203566 LAC (MESS30
03106 102633 JMS MSPRNT
03107 623104 JMP* TYFWD
MESS1 215212 /CR,LF
03110 215212 316317 /N,O
03111 316317 240324 /SP,T
03112 240324 322301 /R,A
03113 322301 316323 /N,S
03114 316323

```

03115	320317		320317	/P,O
03116	322324		322324	/R,T
03117	323240		323240	/S,SP
03120	323305		323305	/S,E
03121	314305		314305	/L,E
03122	303324		303324	/C,T
03123	305304		305304	/E,D
03124	215212		215212	/CR,LF
03125	377000		377000	/R,O.
03126	240302	MESS2	240302	/SP,B
03127	314317		314317	/L,O
03130	303313		303313	/C,K
03131	215212		215212	/CR,LF
03132	377000		377000	/R,O.
03133	240323	MESS3	240323	/SP,S
03134	324301		324301	/T,A
03135	324325		324325	/T,U
03136	323240		323240	/S,SP
03137	302215		302215	/B,CR
03140	212377		212377	/LF,R.O.
03141	240327	MESS4	240327	/SP,W
03142	256303		256303	/,C
03143	256215		256215	/,CR
03144	212377		212377	/LF,RO
03145	304301	MESS5	304301	/D,A
03146	324301		324301	/T,A
03147	240305		240305	/SP,E
03150	322322		322322	/R,R
03151	317322		317322	/O,R
03152	215212		215212	/CR,LF
03153	377000		377000	/RO
03154	240303	MESS6	240303	/SP,C
03155	317322		317322	/O,R
03156	215212		215212	/CR,LF
03157	377000		377000	/RO
03160	240311	MESS7	240311	/SP,1
03161	316303		316303	/N,C
03162	215212		215212	/CR,LF
03163	377000		377000	/RO
03164	240301	MESS8	240301	/SP,A
03165	304304		304304	/D,D
03166	322323		322323	/R,S
03167	240317		240317	/SP,O
03170	306240		306240	/F,SP
03171	311316		311316	/1,N
03172	303215		303215	/C,CR
03173	212377		212377	/LF,RO
03174	303313	MESS9	303313	/C,K
03175	323325		323325	/S,U
03176	315240		315240	/M,SP
03177	305322		305322	/E,R
03200	322317		322317	/R,O
03201	322215		322215	/R,CR
03202	212377		212377	/LF,RO
03203	240322	MESS10	240322	/SP,R



03204	305326		305326	/E,V	
03205	240377		240377	/SP,RO	
03206	240314	MESS11	240314	/SP,L	
03207	256320		256320	/.,P	
03210	256302		256302	/.,B	
03211	256215		256215	/.,CR	
03212	212377		212377	/LF,RO	
03213	240303	MESS12	240303	/SP,C	
03214	310305		310305	/H,E	
03215	303313		303313	/C,K	
03216	240323		240323	/SP,S	
03217	325315		325315	/U,M	
03220	377000		377000	/RO	
03221	240303	MESS13	240303	/SP,C	
03222	301314		301314	/A,L	
03223	303325		303325	/C,U	
03224	314301		314301	/L,A	
03225	324305		324305	/T,E	
03226	304377		304377	/D,RO	
03227	215212	MESS14	215212	/CR,LF	
03230	320256		320256	/P,.	
03231	311256		311256	/I,.	
03232	240377		240377	/SP,RO	
03233	215212	MESS15	215212	/CR,LF	
03234	316317		316317	/N,O	
03235	240320		240320	/SP,P	
03236	256311		256311	/.,I	
03237	256377		256377	/.,RO	
03240	240316	MESS16	240316	/SP,N	
03241	317377		317377	/O,RO	
03242	240304	MESS17	240304	/SP,D	
03243	305303		305303	/E,C	
03244	324301		324301	/T,A	
03245	320305		320305	/P,E	
03246	240323		240323	/SP,S	
03247	313311		313311	/K,I	
03250	320215		320215	/P,CR	
03251	212377		212377	/LF,RO	
03252	240302	MESS18	240302	/SP,B	
03253	314317		314317	/L,O	
03254	303313		303313	/C,K	
03255	240327		240327	/SP,W	
03256	301316		301316	/A,N	
03257	324305		324305	/T,E	
03260	304377		304377	/D,RO	
03261	240302	MESS19	240302	/SP,B	
03262	314317		314317	/L,O	
03263	303313		303313	/C,K	
03264	240306		240306	/SP,F	
03265	317325		317325	/O,U	
03266	316304		316304	/N,D	
03267	215212		215212	/CR,LF	
03270	377000		377000	/RO	
03271	240314	MESS20	240314	/SP,L	
03272	301323		301323	/A,S	

03273	324240	324240	/T,SP	
03274	302314	302314	/B,C	
03275	317303	317303	/O,C	
03276	313215	313215	/K,CR	
03277	212377	212377	/LF,CR	
03300	240302	MESS21 240302	/SP,R	
03301	314317	314317	/L,O	
03302	303313	303313	/C,K	
03303	323240	323240	/S,SP	
03304	322305	322305	/R,E	
03305	301304	301304	/A,D	
03306	215212	215212	/CR,LF	
03307	377000	377000	/RO	
03310	215212	MESS22 215212	/CR,LF	
03311	315317	315317	/M,O	
03312	326305	326305	/V,E	
03313	240377	240377	/SP,RO	
03314	323305	MESS23 323305	/S,E	
03315	301322	301322	/A,R	
03316	303310	303310	/C,H	
03317	240377	240377	/SP,RO	
03320	322305	MESS24 322305	/R,E	
03321	301304	301304	/A,D	
03322	240377	240377	/SP,RO	
03323	327322	MESS25 327322	/W,R	
03324	311324	311324	/I,T	
03325	305240	305240	/E,SP	
03326	377000	377000	/RO	
03327	304301	MESS26 304301	/D,A	
03330	324301	324301	/T,A	
03331	240377	240377	/SP,RO	
03332	301314	MESS27 301314	/A,L	
03333	314240	314240	/L,SP	
03334	377000	377000	/RO	
03335	215212	MESS28 215212	/CR,LF	
03336	215212	215212	/CR,LF	
03337	304322	304322	/D,R	
03340	311326	311326	/I,V	
03341	305240	305240	/E,SP	
03342	240377	240377	/SP,RO	
03343	302301	MESS29 302301	/B,A	
03344	303313	303313	/CK	
03345	327301	327301	/W,A	
03346	322304	322304	/R,D	
03347	240377	240377	/SP,RO	
03350	306317	MESS30 306317	/F,O	
03351	322327	322327	/R,W	
03352	301322	301322	/A,R	
03353	304240	304240	/D,SP	
03354	377000	377000	/RO	
03355	303317	MESS31 303317	/C,O	
03356	316324	316324	/N,T	
03357	311316	311316	/I,N	
03360	325317	325317	/U,O	
03361	325323	325323	/U,S	

03362	240377	240377	/SP,RO
		/CONSTANTS	
03363	000260	ASCII 260	
03364	004377	BF1LOC	RUFFRS-1 /CONSTANT FOR DATA STORE
03365	005000	BF2LOC	RUFFRS+400 /ADDRESS OF SECOND BUFFER
03366	002315	DIRTBL	PNTRS+12 /DIRECTION TABLE
03367	015000	FIFTHO	15000 /SEARCH TO WRITE CONT
03370	005000	FIVTHO	5000
03371	040000	FTYTHO	40000 /BACKWARD
03372	003435	IDCON BLKFND	
03373	001777	INDEP 1777	/MASK FOR BLOCK NUMBER
03374	002326	LSTTBL	PNTRS+23 /LAST BLOCK WRITTEN TABLE
03375	017777	MASK 17777	
03376	776700	MBLOCK	-1-1100+1 /-NUMBER OF BLOCKS
03377	060400	MOVBK	060400 /MOVE BACKWARDS
03400	400000	MUCH0 400000	
03401	000001	ONE 1	
03402	001000	ONETHO	1000
03403	002304	POSTBL	PNTRS+1 /DRIVE POSITION TABLE
03404	004402	RADLOC	RUFFRS+2 /LOCATION -1 OF FIRST DATA WORD (READ ALL)
03405	003777	RBFLOC	RANBFR-1
03406	004377	RBUFND	RANBFR+377
03407	004000	RBUFST	RANBFR
03410	060400	REWIND	060400
03411	000377	RUR0UT	377 /MASK FOR 10-17
03412	021400	SEAFOR	021400
03413	000007	SEVEN 7	
03414	000070	70	
03415	000700	700	
03416	007000	7000	
03417	070000	70000	
03420	700000	700000	
03421	000077	SEVSEV	77 /MASK
03422	000010	TEN 10	
03423	010000	TENTHO	10000 /CONTINUOUS
03424	000037	TERSEV	37 /MASK
03425	000003	THREE 3	
03426	003000	TRETHO	3000 /SEARCH TO READ
03427	013000	TRTETH	13000 /SEARCH TO READ CONTINUOUS.
03430	000200	TWOHUN	200
03431	012000	TWVTHO	12000 /SEARCH TO READ ALL CONTINUOUS
03432	020000	TYTHOU	20000 /GO, STOP
		/VARIABLES	
03433	000000	ACCUM 0	/AC AT P.I
03434	000000	BLKFLG	0
03435	000000	BLKFND	0 /NUMBER OF BLOCK FOUND
03436	000000	CDRIVE	0 /CURRENT DRIVE NUMBER
03437	000000	CKSUMR	0 /CALCULATED CHECKSUM
03440	000000	CNTR 0	/EXTRANEIOUS COUNTER
03441	000000	CNTR1 0	/COUNTER FOR OPRINT
03442	000000	COCNTR	0 /COMPARE ERROR FLAG
03443	000000	COMBIT	0 /BIT POSITION FOR TEST AGAINST S.R.
03444	000000	DATASM	0 /SAVED DATA SUM
03445	000000	DIRECT	0 /CONTAINS DIRECTION READ
03446	000000	DIRFLG	0 /INDICATES DIRECTION TO GO

03447	000000	FRSWAT	0	
03450	000000	FWCHK	0	/FORWARD CHECKSUM
03451	000000	LSTBLK	0	/LAST BLOCK WRITTEN POINTER
03452	000000	LSTDRV	0	/NEGATIVE OF LAST DRIVE SELECTED
03453	000000	MSBITS	0	/HOLD UNIT BITS
03454	000000	NUMBER	0	/STORE FOR MCOMOB
03455	000000	NUMBLK	0	/NUMBER OF BLOCKS WITH WHICH TO OPERATE
03456	000000	OBVERS	0	/ORVERSE NUMBER
03457	000000	PASFLG	0	
03460	000000	PICNTR	0	/PI COUNTER
03461	000000	PIDEX1	0	/PI POINTER
03462	000000	PIDEX2	0	/PI POINTER
03463	000000	PNTR1 0	0	/EXTRANEIOUS POINTER
03464	000000	PNTR2 0	0	/EXTRANEIOUS POINTER
03465	000000	PNTR3 0	0	/POINTER FOR MCOMOB
03466	000000	PNTR4 0	0	/POINTER FOR MSPRNT
03467	000000	POSITN	0	
03470	000000	PREBLK	0	
03471	000000	RAEFLG	0	/READ ALL ERROR FLAG
03472	000000	RECORD	0	
03473	000000	REVCHK	0	/REVERSE CHECK FROM TAPE
03474	000000	RSQFLG	0	
03475	000000	SAVNUM	0	/STORE NUMBER OF BLOCKS
03476	000000	SBRECD	0	/STATUS OF B
03477	000000	TALLY 0	0	/TALLY FOR MCOMOB
03500	000000	TAPONT	0	/TURN AROUND POINT FOR SEARCH
03501	000000	TEM1 0	0	/TEMP STORAGE
03502	000000	TEM2 0	0	/FOR PATGEN
03503	000000	TEM3 0	0	/SUBROUTINE
03504	000000	TEM4 0	0	/STORAGE FOR OPRINT
03505	000000	TRECTR	0	/COUNTER FOR WRITE DATA CONTINUOUS
03506	000000	UNFUNC	0	/UNIT NUMBER
03507	000000	WTCNTR	0	/WAIT COUNTER

000000		.END
--------	--	------

03510	003110	*L
03511	003126	*L
03512	003133	*L
03513	003141	*L
03514	003413	*L
03515	003145	*L
03516	003126	*L
03517	003154	*L
03520	003160	*L
03521	003164	*L
03522	003126	*L
03523	003133	*L
03524	003141	*L
03525	003126	*L
03526	003174	*L
03527	003203	*L
03530	003206	*L
03531	003213	*L
03532	003221	*L
03533	003126	*L
03534	003133	*L

03535	003141	*L
03536	001735	*L
03537	003227	*L
03540	003233	*L
03541	003240	*L
03542	003242	*L
03543	006000	*L
03544	003133	*L
03545	003252	*L
03546	003261	*L
03547	003271	*L
03550	003300	*L
03551	003133	*L
03552	007777	*L
03553	003310	*L
03554	003314	*L
03555	003320	*L
03556	003323	*L
03557	003327	*L
03560	003332	*L
03561	003335	*L
03562	003343	*L
03563	003350	*L
03564	003355	*L
03565	003343	*L
03566	003350	*L

SIZE=03567

NO ERROR LINES

ACCUM	03433
ASCII	03363
BF1LOC	03364
BF2LOC	03365
BLKFLG	03434
BLKFND	03435
RUFFRS	004400
CA	000031
CALCTY	01653
CDRIVE	03436
CHNGDR	02244
CKSTYP	01647
CKSUMR	03437
CLOF	700004
CLON	700044
CLSF	700001
CNTR	03440
CNTR1	03441
COCNTR	03442
CODATA	00626
COERRO	00653
COERR1	00673
COLOOP	00640
COMBIT	03443
CORT1R	00763
CRLF	02714
DATASM	03444
DIRECT	03445
DIRFLG	03446
DIRTBL	03366
EEM	707702
ERRCKS	01602
ERRRAR	01657
ERSTP	02726
EXIST	00572
FIFTHO	03367
FIVTHO	03370
FRSWAT	03447
FTYTHO	03371
FWDCHK	03450
GNPATR	00444
GNPTRS	02263
IDCON	03372
INDEP	03373
IRECO	01735
ISZTST	02046
JMPRET	02156
JMSCON	02205
JMSTST	02134
KRR	700312
KSF	700301
LEM	707704
LOOP	00501
LSTBLK	03451
LSTDRV	03452

LSTTRL	03374
MASK	03375
MBLOCK	03376
MCOMOR	00467
MESS1	03110
MESS10	03203
MESS11	03206
MESS12	03213
MESS13	03221
MESS14	03227
MESS15	03233
MESS16	03240
MESS17	03242
MESS18	03252
MESS19	03261
MESS2	03126
MESS20	03271
MESS21	03300
MESS22	03310
MESS23	03314
MESS24	03320
MESS25	03323
MESS26	03327
MESS27	03332
MESS28	03335
MESS29	03343
MESS3	03133
MESS30	03350
MESS31	03355
MESS4	03141
MESS5	03145
MESS6	03154
MESS7	03160
MESS8	03164
MESS9	03174
MOFPRO	00130
MOFPR1	00145
MOVBAK	03377
MOVER	02336
MSBITS	03453
MSPRNT	02633
MUCH0	03400
MVDATA	00607
NDOFRD	01214
NDTSKP	01762
NUMBER	03454
NUMBLK	03455
OBVERS	03456
ONE	03401
ONETHO	03402
OPRINT	02670
PASFLG	03457
PATGEN	00417
PCF	700202
PICNTR	03460

PIDEX1	03461
PIDEX2	03462
PIRTRL	02045
PITBL	02025
PNTRS	02303
PNTR1	03463
PNTR2	03464
PNTR3	03465
PNTR4	03466
POSITN	03467
POSTRL	03403
PREBLK	03470
PREGEN	00400
PSA	700204
PSB	700244
PSF	700201
PTSTND	01766
PTSTNS	02005
RADCHK	01540
RADLOC	03404
RAEFLG	03471
RALLTS	01431
RANBFR	004000
RANDEX	00100
RAND1	00516
RAND2	00527
RAND3	00540
RANSEL	00551
RASERR	01556
RBFLOC	03405
RBUFND	03406
RBUFST	03407
RCF	700102
RDBUF1	01117
RDCERR	01314
RDCMOD	01221
RDDATA	00253
ROSEQ	01070
RDSWCH	00123
RDTAB	00211
RECORD	03472
REPOSI	02207
REREAD	00272
REREDA	00325
RESETD	01721
RETJMP	02206
REVCHK	03473
REWCK	01034
REWDRV	00361
REWIND	03410
ROT1TS	02070
ROT2TS	02106
RRB	700112
RSA	700104
RSB	700144



RSF	700101
RSFDRV	02225
RSFDR1	02242
RSQFLG	03474
RUROUT	03411
R1	00525
R2	00536
R3	00547
SADTST	02124
SAMDRV	00231
SAVNUM	03475
SBRECD	03476
SEAFOR	03412
SEARCH	02353
SETIDX	02015
SEVEN	03413
SEVSEV	03421
SRCMOD	02533
SRCONT	02441
SREZTS	02503
SRFBLK	02461
SRHERR	02562
SRTARN	02412
SWCMDL	01354
SWCMOD	01343
TALLY	03477
TAPONT	03500
TCF	700402
TEM1	03501
TEM2	03502
TEM3	03503
TEM4	03504
TEN	03422
TENTHO	03423
TERSEV	03424
TEST1	02200
TEST2	02201
TEST3	02202
TEST4	02203
TEST5	02204
THREE	03425
TLS	700406
TRECTR	03505
TRETHO	03426
TRTETH	03427
TSF	700401
TWOHUN	03430
TWVTHO	03431
TYALL	03034
TYBKW	03100
TYDATA	03030
TYDIR	03056
TYDRV	03040
TYFWD	03104
TYMODE	03070

TYMOVE	02734
TYPE	02661
TYRALL	02765
TYRDAT	02754
TYREAD	03020
TYSRCH	02743
TYTHOU	03432
TYWALL	03007
TYWDAT	02776
TYWRIT	03024
UNFUNC	03506
WATINT	01701
WC	000030
WDCERR	01405
WDCMOD	01332
WDINC	01022
WRITE1	00721
WRT1A	00775
WSTERR	01042
WTCNTR	03507
XORSUM	01560
.EOT	00000

.EOT	00000
WC	000030
CA	000031
RANDEX	00100
RDSWCH	00123
MOFPR0	00130
MOFPR1	00145
ROTAR	00211
SAMDRV	00231
RDDATA	00253
PEREAD	00272
PEREDA	00325
REWDRV	00361
PREGEN	00400
PATGEN	00417
GNPATR	00444
MCOMOB	00467
LOOP	00501
RAND1	00516
R1	00525
RAND2	00527
R2	00536
RAND3	00540
R3	00547
RANSEL	00551
EXIST	00572
MVDATA	00607
CODATA	00626
COLOOP	00640
COERRO	00653
COERR1	00673
WRITE1	00721
CORT1R	00763
WRT1A	00775
WDINC	01022
REWCK	01034
WSTERR	01042
RDSEQ	01070
RDBUF1	01117
NDOFRD	01214
ROCMOD	01221
RDCERR	01314
WDCMOD	01332
SWCMOD	01343
SWCMDL	01354
WDCERR	01405
RALLTS	01431
RADCHK	01540
RASERR	01556
XORSUM	01560
ERRCKS	01602
CKSTYP	01647
CALCTY	01653
ERRRAR	01657
WATINT	01701

RESETD	01721
IRECD	01735
NDTSKP	01762
PTSTND	01766
PTSTNS	02005
SETIDX	02015
PITBL	02025
PIRTBL	02045
ISZTST	02046
ROT1TS	02070
ROT2TS	02106
SADTST	02124
JMSTST	02134
JMPRET	02156
TEST1	02200
TEST2	02201
TEST3	02202
TEST4	02203
TEST5	02204
JMSCON	02205
RETJMP	02206
REPOSI	02207
RSFDRV	02225
RSFDR1	02242
CHNGDR	02244
GNPTRS	02263
PNTRS	02303
MOVER	02336
SEARCH	02353
SRTARN	02412
SRCONT	02441
SRFBLK	02461
SREZTS	02503
SRCMOD	02533
SRHERR	02562
MSPRNT	02633
TYPE	02661
OPRINT	02670
CRLF	02714
ERSTP	02726
TYMOVE	02734
TYSRCH	02743
TYRDAT	02754
TYRALL	02765
TYWDAT	02776
TYWALL	03007
TYREAD	03020
TYWRIT	03024
TYDATA	03030
TYALL	03034
TYDRV	03040
TYDIR	03056
TYMODE	03070
TYBKW	03100
TYFWD	03104

MESS1	03110
MESS2	03126
MESS3	03133
MESS4	03141
MESS5	03145
MESS6	03154
MESS7	03160
MESS8	03164
MESS9	03174
MESS10	03203
MESS11	03206
MESS12	03213
MESS13	03221
MESS14	03227
MESS15	03233
MESS16	03240
MESS17	03242
MESS18	03252
MESS19	03261
MESS20	03271
MESS21	03300
MESS22	03310
MESS23	03314
MESS24	03320
MESS25	03323
MESS26	03327
MESS27	03332
MESS28	03335
MESS29	03343
MESS30	03350
MESS31	03355
ASCII	03363
RF1LOC	03364
RF2LOC	03365
DIRTBL	03366
FIFTHO	03367
FIVTHO	03370
FTYTHO	03371
IDCON	03372
INDEP	03373
LSTTBL	03374
MASK	03375
MBLOCK	03376
MOVBAK	03377
MUCH0	03400
ONE	03401
ONETHO	03402
POSTBL	03403
RADLOC	03404
RBFL0C	03405
RBUFND	03406
RBUFST	03407
REWIND	03410
RUBOUT	03411
SEAFOR	03412

SEVEN	03413
SEVSEV	03421
TEN	03422
TENTHO	03423
TERSEV	03424
THREE	03425
TRETHO	03426
TRTETH	03427
TWOHUN	03430
TWVTHO	03431
TYTHOU	03432
ACCUM	03433
BLKFLG	03434
BLKFND	03435
CDRIVE	03436
CKSUMR	03437
CNTR	03440
CNTR1	03441
COCNTR	03442
COMBIT	03443
DATASM	03444
DIRECT	03445
DIRFLG	03446
FRSWAT	03447
FWCHK	03450
LSTBLK	03451
LSTDRV	03452
MSBITS	03453
NUMBER	03454
NUMBLK	03455
OBVERS	03456
PASFLG	03457
PICNTR	03460
PIDEX1	03461
PIDEX2	03462
PNTR1	03463
PNTR2	03464
PNTR3	03465
PNTR4	03466
POSITN	03467
PREBLK	03470
RAEFLG	03471
RECORD	03472
REVCHK	03473
RSQFLG	03474
SAVNUM	03475
SBRECD	03476
TALLY	03477
TAPONT	03500
TEM1	03501
TEM2	03502
TEM3	03503
TEM4	03504
TRECTR	03505
UNFUNC	03506

WTCNTR	03507
RANBFR	004000
BUFFRS	004400
CLSF	700001
CLOF	700004
CLOM	700044
RSF	700101
RCF	700102
RSA	700104
RRB	700112
RSB	700144
PSF	700201
PCF	700202
PSA	700204
PSB	700244
KSF	700301
KRB	700312
TSF	700401
TCF	700402
TLS	700406
EEM	707702
LEM	707704

MAINDEC EVALUATION REQUEST

After sufficient familiarization with the operation and documentation of this MAINDEC, please indicate your assessment of the following areas and return this form to Digital Equipment Corporation.

IDENTIFICATION: MAINDEC NO. \_\_\_\_\_ Program Title \_\_\_\_\_

USAGE: Used by: Field Service  Production  Other \_\_\_\_\_

Frequency of Usage: Daily  Weekly  Monthly

SUGGESTIONS FOR IMPROVEMENT

1. Are the program loading and operating instructions: clear? , incomplete? , difficult to follow?
2. Do the error reports and program documentation provide sufficient diagnostic information. in all cases? , in most cases? , in very few cases? . Suggestions for improvement:

---

---

---

3. Is the program effective in isolating malfunctions: in all cases? , in most cases? , in very few cases? . Would additional Scope loops or Switch Register control be helpful? \_\_\_\_\_  
Suggestions for improvement:

---

---

---

4. Does the program ever fail to detect malfunctions exposed by other software? \_\_\_\_\_  
Were Margins used? \_\_\_\_\_ Please describe malfunction in detail:

---

---

---

5. Does the program ever report non-existent malfunctions? \_\_\_\_\_  
Please indicate erroneous report and any pertinent operating conditions:

---

---

---

6. Does this MAINDEC ever expose malfunctions in the Central Processor or other peripheral units not detected by the appropriate MAINDEC? \_\_\_\_\_  
Please describe malfunction and MAINDEC(S) used:

---

---

---

7. Does the document provide a general understanding of the functional programming requirements of the system? Good , Fair , None . Would a general description of programming requirements increase the effectiveness of this MAINDEC? \_\_\_\_\_

Remarks:

---

---

---



..... Fold Here .....

..... Do Not Tear - Fold Here and Staple .....

FIRST CLASS  
PERMIT NO. 33  
MAYNARD, MASS.

BUSINESS REPLY MAIL  
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

**digital**

Digital Equipment Corporation  
Diagnostic Programming Group  
146 Main Street, Building 12  
Maynard, Massachusetts 01754

