

TM,A,B-11

MULTIDRIVE DATA RELIABILITY
MD-11-DZTMH-F

EP-DZTMH-F-DL

COPYRIGHT ©75-77

FICHE 1 OF 1

JAN 1978

digital

MADE IN USA

The image shows a microfiche card with a grid of 150 small, illegible data tables or charts arranged in 10 columns and 15 rows. The text is too small to read, but the layout suggests a structured data presentation, possibly test results or performance metrics for the MD-11-DZTMH-F drive.



PRODUCT CODE	MAINDEC-11-DZTMH-F-D
PRODUCT NAME	TM, A, B-11 MULTIDRIVE DATA RELIABILITY EXERCISER
PRODUCT DATE	15-NOVEMBER-1977
MAINTAINER	DIAGNOSTIC ENGINEERING

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C) 1975, 1977 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	

TABLE OF CONTENTS

PARAGRAPH	SUBJECT	PAGE
1	ABSTRACT	1
2	REQUIREMENTS	1
3	LOADING PROCEDURE	1
4	STARTING PROCEDURE	2
5	DATA PATTERNS	7
6	RANDOMIZATION	8
7	DYNAMIC PARAMETER	9
8	CONSOLE SWITCHES	10
9	ERROR PRINTOUT	14
10	STATISTIC PRINTOUT	20
11	AUTO SEQUENCE	22
12	TESTING PROCEDURES	24
13	LISTING	

1 ABSTRACT

THIS PROGRAM IS DESIGNED TO BE USED BY AN EXPERIENCED ENGINEER /TECHNICIAN FOR EVALUATION AND DEBUGGING OF MAG TAPE DRIVES. THE PROGRAM IS CAPABLE OF EXERCISING ANY TAPE DRIVE THAT CAN BE OPERATED ON A UNIBUS PDP-11 SYSTEM THROUGH THE TM, A, B-11 MAG TAPE CONTROLLER. ANY TYPE OF TAPE DRIVE, 7 OR 9 TRACK MAY BE USED. ANY NUMBER OF DRIVES, SINGLE OR MULTIDRIVE SYSTEMS, UP TO EIGHT (8), MAY BE TESTED BY A SINGLE EXECUTION OF THE PROGRAM. THIS FLEXIBILITY IS POSSIBLE BECAUSE THE PROGRAM HAS NO FIXED PARAMETERS OR TESTING SEQUENCE. THE ENTIRE TEST PLAN, INCLUDING PARAMETERS AND OPERATING SEQUENCE, IS DETERMINED BY THE OPERATOR THROUGH RESPONSES TO TELETYPE REQUESTS AND SETTING OF CONSOLE SWITCHES.

THE PROGRAM PROVIDES FOR TESTING OF ALL TAPE DRIVE FUNCTIONS SUCH AS WRITING, READING, REWINDING, TAPE POSITIONING, EOT - BOT SENSING AND ASSUMES A GOOD CONTROLLER.

HOWEVER, THE CONTROLLER IS TESTED SOMEWHAT INTRINSICALLY DURING THE TEST CYCLE IN ORDER TO PROVIDE FULL INFORMATION ABOUT ANY ERROR CONDITIONS DETECTED.

DURING A TEST CYCLE, CHECKS ARE MADE FOR STATUS ERRORS, DATA ERRORS, POSITION ERRORS, WORD COUNT AND CURRENT MEMORY ADDRESS ERRORS WHEREVER APPLICABLE.

2 REQUIREMENTS (HARDWARE)

- A ANY PDP-11 PROCESSOR
- B 8K OF CORE
- C TELETYPE
- D TM, A, B-11 TAPE CONTROL UNIT
- E 1 TO 8 TSO3 OR TU10, N, W MAG TAPE DRIVES

3 LOADING PROCEDURE

- A USE STANDARD PROCEDURE FOR LOADING BINARY TAPES
- B PROGRAM IS LOADABLE AND CHAINABLE IN 8K OF MEMORY. DEFAULT CHAIN MODE IS A SINGLE PASS ON DRIVE 0 AT 9TRK, 800 BPI, 100 RECORDS OF 200 CHARACTERS EACH WITH PATTERN ONE AND ALL SWITCHES 0.

4
STARTING PROCEDURE

THERE ARE FOUR (4) STARTING ADDRESSES THAT MAY BE USED,
200(8), 204(8), 210(8), AND 240(8)

A 200(8) THIS ADDRESS MUST BE USED ON INITIAL START FROM
LOAD AS ALL PARAMETERS ARE ENTERED FROM HERE.
REQUESTS ARE PRINTED ON THE TELETYPE FOR ENTRY OF
CONTROLLER REGISTER STARTING ADDRESS, VECTOR ADDRESS,
UNIT NUMBER, DENSITY, PARITY, RECORD COUNT, CHARACTER
COUNT, PATTERN NUMBER, TAPE MARK (EOF) OPTION, AND STALL
FOR READ, WRITE, AND TURNAROUND. ALL RESPONSES SHOULD
BE MADE IN OCTAL AND WITHIN THE LIMITS OF THE PARAMETER
A QUESTION MARK (?) WILL BE TYPED IF ANY
CHARACTER ENTERED IS NOT BETWEEN 0 THRU 7 (OCTAL).
THE CHARACTER MAY BE RETYPED FOLLOWING THE QUESTION
MARK IF THE RESPONSE IS NOT WITHIN ITS LIMITS A
QUESTION MARK (?) IS TYPED AND THE ENTIRE RESPONSE
MAY BE REENTERED SOME RESPONSES REQUIRE MORE THAN ONE
(1) CHARACTER, BUT NONE REQUIRES MORE THAN SIX (6)
RESPONSES NEED NOT HAVE
LEADING ZEROS AND SHOULD BE TERMINATED BY A CARRIAGE
RETURN IF LESS THAN THE MAXIMUM NUMBER OF CHARACTERS
S NPUT

B 204(8) THIS ADDRESS SHOULD BE USED ANYTIME A RESTART
OF THE PROGRAM IS NECESSARY AND THE PARAMETERS
ENTERED AT THE INITIAL START OF 200(8) NEED NOT
BE CHANGED ALSO NOTE THAT ANY DATA PATTERN WHICH
HAD BEEN GENERATED BY SETTING THE RANDOM DATA
SWITCH (CONSOLE SWITCH EIGHT) WILL NOT BE OVERWRITTEN
AND THEREFORE IS HELD IN CORE FOR USE UNTIL
CONSOLE SWITCH EIGHT(8) IS AGAIN SET

C 210(8) THIS ADDRESS IS THE SAME AS USING 204(8) IN THAT THE
PREVIOUSLY SET PARAMETERS ARE USED, HOWEVER, THE DATA
PATTERN IS RETURNED TO THE FIXED PATTERN ORIGINALLY
CALLED FOR AT THE 200(8) START ALSO ALL STATISTICS
PREVIOUSLY GATHERED WILL BE CLEARED

E 240(8) THIS IS A SPECIAL ADDRESS WHICH WILL CAUSE THE
PROGRAM TO EXECUTE A PREDETERMINED TEST PLAN ON
ALL AVAILABLE UNITS THE ONLY INPUT REQUIRED
BY THE OPERATOR IS A RESPONSE TO REQUESTS FOR THE
CONTROLLER ADDRESS, VECTOR ADDRESS, AND CONTINUOUS
OPERATION OF THE SEQUENCE

SEE ITEM 11, (PAGE 22) FOR FULL DETAILS

THE FOLLOWING IS AN EXPLANATION OF THE INITIAL START (200 OCTAL) REQUESTS AND RESPONSES.

REGISTER START THE RESPONSE REQUIRED FOR THIS REQUEST IS TO ENTER THE ADDRESS OF THE FIRST CONTROLLER REGISTER (MTS) AS A SIX DIGIT UNIBUS ADDRESS

VECTOR ADDRESS THE RESPONSE FOR THIS REQUEST IS TO ENTER THE INTERRUPT VECTOR ADDRESS USED BY THE CONTROLLER AS A THREE (3) DIGIT ADDRESS

UNIT NUMBER THE UNIT NUMBER IS ENTERED AS ONE (1) OCTAL CHARACTER AND MUST BE WITHIN THE LIMITS OF 0 THROUGH 7. WHEN THE UNIT NUMBER HAS BEEN ENTERED AND IS LEGAL, THE PROGRAM TESTS FOR THE PRESENCE OF A UNIT OF THAT NUMBER. IF THE UNIT IS AVAILABLE A PRINTOUT OF 7 CHANNEL OR 9 CHANNEL WILL BE MADE TO ASSIST THE OPERATOR IN SETTING DENSITY AND PARITY. IF THE UNIT IS NOT AVAILABLE, A MESSAGE STATING SO WILL BE PRINTED AND A NEW UNIT NUMBER REQUEST WILL BE ISSUED. WHEN A GOOD UNIT NUMBER HAS BEEN ENTERED, REQUESTS FOR OPERATING DENSITY AND PARITY ARE MADE FOR THAT UNIT AND SHOULD BE RESPONDED TO ACCORDING TO THAT PARTICULAR UNIT'S NEEDS. AS MANY AS EIGHT (8) UNIT NUMBER REQUESTS MAY BE USED, HOWEVER, AT LEAST ONE MUST BE USED. THE UNIT NUMBER AND THEIR RESPECTIVE DENSITY AND PARITY MAY BE ENTERED IN ANY ORDER. THE INFORMATION FOR EACH UNIT ENTERED IS LOADED INTO A TABLE FOR REFERENCE IN TESTING. IF LESS THAN EIGHT(8) UNITS ARE REQUIRED, THEN RESPONDING TO THE UNIT NUMBER REQUEST WITH A CARRIAGE RETURN WILL TERMINATE THE UNIT ENTRIES AND CONTINUE TO THE NEXT PARAMETER. IT SHOULD BE REMEMBERED THAT AT LEAST ONE UNIT NUMBER REQUEST MUST BE ENTERED. IF THE FIRST REQUEST IS RESPONDED TO BY A CARRIAGE RETURN, THEN THE REQUEST WILL BE REPEATED

DENSITY THE DENSITY REQUEST IS RESPONDED TO BY ONE (1) OCTAL CHARACTER AND MUST BE WITHIN THE LIMITS OF 0 THRU 3. AS EACH UNIT NUMBER IS ENTERED, A REQUEST FOR THE OPERATING DENSITY FOR THAT UNIT IS TYPED. THE RESPONSE MEANINGS ARE AS FOLLOWING.

A 0 = 200BP1, 7 CHANNEL NRZ1
 B 1 = 556BP1, 7 CHANNEL NRZ1
 C 2 = 800BP1, 7 CHANNEL NRZ1
 D 3 = 800BP1, 9 CHANNEL NRZ1

PARITY THE PARITY REQUEST IS RESPONDED TO BY ONE (1) OCTAL CHARACTER AND MUST BE EITHER 0 OR 1

A. 1 = EVEN PARITY
B. 0 = ODD PARITY

RECORD COUNT THIS REQUEST IS RESPONDED TO BY A SIX (6) CHARACTER OCTAL NUMBER FROM 1 TO 177777. REMEMBER LEADING ZEROS ARE NOT REQUIRED AND IF LESS THAN SIX CHARACTERS ARE ENTERED, A CARRIAGE RETURN WILL TERMINATE THE RESPONSE. THE RECORD COUNT IS USED IN CONJUNCTION WITH THE CHARACTER COUNT TO ESTABLISH A BLOCKING FACTOR FOR USE IN READ OR WRITE CYCLES.

CHARACTER COUNT THIS RESPONSE IS ENTERED AS FOUR (4) OCTAL CHARACTERS WITHIN THE LIMITS OF 4 THRU 4000. AGAIN LEADING ZEROS ARE NOT REQUIRED AND A CARRIAGE RETURN TERMINATES A LESS THAN FOUR (4) CHARACTER RESPONSE. THE CHARACTER COUNT IN CONJUNCTION WITH THE RECORD COUNT IS USED TO ESTABLISH THE BLOCK SIZE (CHARACTERS PER RECORD, AND RECORDS PER BLOCK) USED IN READ AND WRITE CYCLES. THE SAME BLOCKING IS USED ON ALL AVAILABLE UNITS.

PATTERN NUMBER THIS RESPONSE IS A TWO (2) CHARACTER OCTAL NUMBER WITHIN THE LIMITS OF 0 THRU 20(8). THE NUMBER ENTERED WILL CAUSE A SPECIFIC DATA PATTERN TO BE USED FOR ALL READING AND WRITING. THIS DATA PATTERN IS NOT CHANGED UNLESS RANDOM DATA IS REQUESTED BY SETTING CONSOLE SWITCH EIGHT (8) TO A ONE. RESETTING OF THE RANDOM DATA SWITCH DOES NOT CAUSE REVERSION TO THE FIXED PATTERN, BUT WILL HOLD THE LAST GENERATED PATTERN UNTIL A RESTART IS DONE FROM LOCATION 210(8) OR 200(8). THE SELECTION OF DATA PATTERN ZERO (0) HAS A SPECIAL USE. PATTERN NUMBER ZERO (0) WILL CAUSE TO BE READ IN AT THE HIGH SPEED PAPER TAPE READER ANY DATA PATTERN DESIRED. THE EXTERNAL INPUT DATA THROUGH THE READER IS DONE BY PREPARING A PAPER TAPE WITH A PROGRAM CALLED DTC. (MAINDEC-11-DZTUF). ANY CONFIGURATION OF BITS AND CHARACTERS MAY BE USED AND A LIMIT OF 377(8) CHARACTERS IS IMPOSED. WHEN EXTERNAL DATA IS INPUT, THE ENTIRE WRITE BUFFER IN CORE IS FILLED WITH THE PATTERN SO THAT ANY SIZE RECORD MAY BE USED. DATA PATTERN ZERO (0) EXTERNAL PAPER TAPE NEED ONLY BE READ ONCE AT INITIAL START OF 200(8), AND NEED NOT BE READ AGAIN UNLESS OVERWRITTEN BY RANDOM DATA. BE SURE TO LOAD THE READER BEFORE PRESSING START.

SEE ITEM 5, (PAGE 7) FOR A DESCRIPTION OF THE DATA PATTERNS.

TAPE MARK

THE TAPE MARK REQUEST IS USED TO DETERMINE IF THE OPERATOR WISHES TO HAVE EACH DATA BLOCK SEPARATED BY A TAPE MARK (OFTEN CALLED EOF FOR END OF FILE) IF RESPONDED TO BY A ONE(1) THE TAPE MARK WILL BE WRITTEN AND WHEN READING WILL BE EXPECTED AT THE END OF EACH DATA BLOCK A ZERO(0) RESPONSE WILL DISALLOW THE TAPE MARK OPTION PLEASE NOTE THAT THE TAPE MARK RECORD INCREASES THE BLOCK SIZE BY ONE(1) RECORD, IN OTHER WORDS, A BLOCK OF 100 RECORDS WILL HAVE THE TAPE MARK AS RECORD 101

SINGLE PASS

IF RESPONDED TO WITH A ONE, THE PROGRAM WILL HALT AND PRINT AN END OF PASS MESSAGE WHEN THE LAST AVAILABLE UNIT REACHES END OF TAPE AND IS REWOUND

STALLS

THE STALL REQUESTS ARE RESPONDED TO BY A SIX (6) CHARACTER OCTAL NUMBER WITHIN THE LIMITS OF 1 THRU 177777 LEADING ZEROS ARE NOT REQUIRED AND AN ENTRY OF LESS THAN SIX (6) CHARACTERS SHOULD BE TERMINATED BY A CARRIAGE RETURN EACH INCREMENT OF THE VALUE ADDS ABOUT 2.6 MICSEC TO THE DELAY

READ THE TIME DELAY BETWEEN EACH RECORD READ
WRITE THE TIME DELAY BETWEEN EACH RECORD WRITTEN
TURN AROUND TIME DELAY BETWEEN CHANGES OF TAPE DIRECT ON (FORWARD, TO REVERSE ETC) AND BETWEEN BLOCKS.

FIXED PARAMETERS IT SHOULD BE NOTED THAT ALL PARAMETERS EXCEPT FOR THE UNIT DESCRIPTION VALUES (UNIT NUMBER, DENSITY, AND PARITY) HAVE NOMINAL VALUES ALREADY STORED IN THE PROGRAM AS EACH PARAMETER REQUEST (PATTERN NUMBER, RECORD COUNT, CHARACTER COUNT, AND STALLS) IS TYPED ITS PRESENT STORED VALUE IS ALSO PRINTED IF THESE VALUES NEED NOT BE CHANGED, SIMPLY TYPE A CARRIAGE RETURN AS RESPONSE AND NO CHANGE WILL BE MADE. EACH START OF THE PROGRAM AT 200(8) WILL SHOW THE CURRENT VALUES OF THESE PARAMETERS AS PER THE LAST ENTRY WHEN A FRESH LOAD OF THE PAPER TAPE IS DONE THE PARAMETERS WILL REFLECT THE FIXED VALUES STORED IN THE PROGRAM

A RECORD COUNT = 100
B CHARACTER COUNT = 200
C PATTERN NUMBER = 1
D READ STALL = 1
E WRITE = 1
F TURN AROUND = 1

SAMPLE START AT 200(8)

THE FOLLOWING IS A SAMPLE OF THE
PRINTED REQUESTS AND THEIR RESPONSES
RESPONSES ARE ENCLOSED IN PARENS FOR
CLARITY ONLY AND (CR) MEANS CARRIAGE RETURN

LOAD ADDRESS 200(8), SET CONSOLE SWITCHES, PRESS START SWITCH

TM, A, B-11 TS03 OR TU10, N, W MULTIDRIVE DATA RELIABILITY EXERCISER

ENTER CONDITIONS IN OCTAL

REGISTER START = 172520 (CR)

VECTOR ADDRESS = 224 (CR)

UNIT NUMBER=(5) 9 TRK

DENSITY=(3)

PARITY=(0)

UNIT NUMBER=(2) 7 TRK

DENSITY=(2)

PARITY=(1)

UNIT NUMBER=(CR)

RECORD COUNT=100 (500)(CR)

CHARACTER COUNT=201 (38)^(7)(CR)

PATTERN NUMBER=1 (22)

?

(6)(CR)

TAPE MARK = 0 (1)(CR)

SINGLE PASS = 0(CR)

ENTER STALLS

READ=1 (CR)

WRITE=1 (CR)

TURN AROUND=1 (3000)(CR)

THE PROGRAM WILL NOW PERFORM THE TEST CYCLE SET IN
THE CONSOLE SWITCHES ON UNIT FIVE (5) THEN TWO (2).
ONE BLOCK ON EACH UNIT PER CYCLE, USING DATA PATTERN
NUMBER SIX (6) WITH A BLOCKING FACTOR OF 37 CHARACTERS
PER RECORD AND 500 RECORDS PER BLOCK. THE DELAYS ARE SET
FOR MINIMUM ON READ AND WRITE, AND APPROXIMATELY 75
SECONDS ON TURN AROUND

DATA PATTERNS

THERE ARE TWENTY DATA PATTERN GENERATORS STORED IN CORE AND ANY ONE OF THESE MAY BE SELECTED. THE ONE UNIQUE CASE IS PATTERN ZERO(0), SELECTION OF PATTERN ZERO(0) REQUIRES THAT A PREVIOUSLY PREPARED PAPER TAPE BE ENTERED AT THE HIGH SPEED READER. THIS TAPE CONTAINS A DATA PATTERN OF NO MORE THAN 377 OCTAL CHARACTERS. THE FIRST CHARACTER READ IN IS THE NUMBER OF ACTUAL DATA CHARACTERS THAT ARE CONTAINED ON THE TAPE. EACH DATA CHARACTER MAY BE ANY COMBINATION OF BITS AND WILL BE LOADED INTO CORE AS THEY APPEAR ON THE TAPE. NO MATTER HOW MANY CHARACTERS ARE ON TAPE, THE ENTIRE WRITE BUFFER (2000 CHARACTERS) WILL BE FILLED WITH THE PATTERN ENTERED SO THAT ANY SIZE RECORD CAN BE USED.

THE FOLLOWING IS A LIST OF THE DATA PATTERNS AVAILABLE

DATA0 EXTERNAL INPUT THRU HIGH SPEED READER (SEE DTC, MAINDEC-11-DZTUF-A)
 DATA1 ALL ONE BITS IN ALL CHARACTERS
 DATA2 ALL ZERO BITS IN ALL CHARACTERS
 DATA3 A ONE BIT WALKING FROM RIGHT TO LEFT IN A FIELD OF ZEROS
 DATA4 A ZERO BIT WALKING FROM RIGHT TO LEFT IN A FIELD OF ONES
 DATA5 ALTERNATING ONE AND ZERO BITS IN EACH CHARACTER
 DATA6 ALTERNATING ZERO AND ONE BITS IN EACH CHARACTER
 DATA7 SAME AS DATA5 BUT WITH EVERY OTHER CHARACTER COMPLEMENTED
 DATA10. SAME AS DATA6 BUT WITH EVERY OTHER CHARACTER COMPLEMENTED
 DATA11 INCREMENTING CHARACTERS (000-377)
 DATA12 DECREMENTING CHARACTERS (377-000)
 DATA13 ALTERNATING CHARACTERS OF ALL ZERO AND ALL ONE BITS
 DATA14 ALTERNATING CHARACTERS OF ALL ONE AND ALL ZERO BITS
 DATA15 SPECIAL PATTERN OF A WALKING ZERO BIT REPEATED 4 TIMES
 DATA16. IBM COMPAT PATTERN 1. RIPPLE
 DATA17 IBM COMPAT PATTERN 2 FIXED (ABCDEF)
 DATA20 IBM COMPAT PATTERN 3 FIXED (J)

b
RANDOMIZATION

THERE ARE THREE (3) VALUES THAT MAY BE GENERATED RANDOMLY, DATA, CHARACTER COUNT, AND RECORD COUNT. THESE ARE NORMALLY SET TO SOME FIXED VALUE BUT MAY BE RANDOMIZED BY SETTING THE APPROPRIATE CONSOLE SWITCHES

- A RANDOM DATA. (CONSOLE SWITCH 8)
GENERATES AN ENTIRE BUFFER, CHARACTER BY CHARACTER, OF RANDOM DATA WHEN SWITCH 8 IS SET TO A ONE. ONCE SET, THE RESETTING OF SWITCH 8 CAUSES THE LAST GENERATED PATTERN TO BE RETAINED IN CORE. A RESTART AT LOCATION 200(8) OR 210(8) WILL CAUSE REVERSION OF THE DATA TO THE FIXED PATTERN REQUESTED INITIALLY. A RESTART AT LOCATION 204(8) WILL HOLD THE LAST GENERATED PATTERN IN CORE UNTIL SWITCH 8 IS AGAIN SET. ALTHOUGH THE DATA IS GENERATED AS RANDOM, THE PROGRESSION OF RANDOM CHARACTERS IS ALWAYS THE SAME FROM THE OUTSET OF RANDOMIZATION. THEREFORE IT IS POSSIBLE TO GENERATE ONE TAPE REEL OF RANDOM DATA ON ONE UNIT, RESTART THE PROGRAM TO RE-ESTABLISH THE OUTSET POINT, AND READ THE RANDOM TAPE REEL ON ANOTHER UNIT FOR COMPATABILITY TESTING. IN MULTIDRIVE SYSTEMS THE SAME BLOCK OF DATA, WHETHER RANDOM OR FIXED, IS WRITTEN OR READ ON EACH AVAILABLE UNIT IN THE ORDER THAT THEY WERE ENTERED, BEFORE BEING CHANGED.
- B RANDOM CHARACTER COUNT. (CONSOLE SWITCH 7)
GENERATES A DIFFERENT NUMBER OF CHARACTERS PER RECORD TO BE WRITTEN ON EACH BLOCK CYCLE. THE SAME NUMBER OF CHARACTERS PER RECORD IS WRITTEN OR READ ON EACH AVAILABLE UNIT BEFORE BEING CHANGED. RESETTING SWITCH 7 HOLDS THE LAST VALUE GENERATED.
- C RANDOM RECORD COUNT. (CONSOLE SWITCH 6)
GENERATES A DIFFERENT NUMBER OF RECORDS FOR EACH BLOCK OF DATA WRITTEN OR READ ON EACH BLOCK CYCLE. THE SAME NUMBER OF RECORDS IS WRITTEN OR READ ON EACH AVAILABLE UNIT BEFORE BEING CHANGED. RESETTING SWITCH 6 HOLDS LAST VALUE GENERATED.

DYNAMIC PARAMETERS

THE THREE (3) STALL VALUES ARE CONSIDERED TO BE DYNAMIC PARAMETERS AS THEY MAY BE CHANGED WHILE THE PROGRAM IS RUNNING BY TYPING A CONTROL C CHARACTER AT THE TELETYPE AS SOON AS THE BUS IS RELEASED BY THE MAG TAPE OPERATION IN PROGRESS, THE PROGRAM WILL RESPOND TO THE CONTROL C INPUT BY TYPING A REQUEST FOR NEW STALL PARAMETERS THE LAST VALUES THAT WERE ENTERED WILL BE PRINTED AS THE STORED VALUES AND MAY BE CHANGED BY ENTERING NEW VALUES OR LEFT UNCHANGED BY TYPING A CARRIAGE RETURN

THIS PROGRAM HAS BEEN MODIFIED TO RUN ON A PROCESSOR WITH OR WITHOUT A HARDWARE SWITCH REGISTER WHEN FIRST EXECUTED THE PROGRAM TESTS THE EXISTENCE OF A HARDWARE SWITCH REGISTER IF NOT FOUND A SOFTWARE SWITCH REGISTER LOCATION (SWREG=LOC 176) IS DEFAULTED TO IF THIS IS THE CASE, UPON EXECUTION THE CONTENTS OF THE SWREG ARE DUMPED IN OCTAL ON THE CONSOLE TTY AND ANY CHANGES ARE REQUESTED

(E) SWR=XXXXXX NEW=

POSSIBLE RESPONSES ARE

- 1 <CR> IF NO CHANGES ARE TO BE MADE
- 2 6 DIGITS 0-7 TO REPRESENT IN OCTAL THE NEW SWITCH REGISTER VALUE ,LAST DIGIT FOLLOWED BY <CR>
- 3 U TO ALLOW REENTERING VALUE IF ERROR IS COMMITTED KEYING IN SWREG VALUE
- 4 <LF> ONLY VALID FOR ACT-11 SYSTEMS-DO NOT USE

BUILT INTO THE PROGRAM IS THE ABILITY TO DYNAMICALLY CHANGE THE CONTENTS OF SWREG DURING PROGRAM EXECUTION. BY STRIKING G (CNTL G) ON CONSOLE TTY THE OPERATOR SETS A REQUEST FLAG TO CHANGE THE CONTENTS OF SWREG, WHICH IS PROCESSED IN KEY AREAS OF THE PROGRAM CODE (IE) ERROR ROUTINES, AFTER HALTS END OF PASS, AND OTHER APPLICABLE AREAS

3 1 CONSOLE SWITCH SETTINGS

THE CONSOLE SWITCHES ARE USED TO SET-UP THE TEST CYCLE DESIRED, TO GENERATE RANDOM VALUES, AND TO CONTROL ERROR RESPONSES. THE SWITCHES SHOULD BE SET IN THE DESIRED MANNER BEFORE PRESSING THE START SWITCH BECAUSE THEY ARE ALL DYNAMIC AND WILL RUN THE PROGRAM IN ANY CONFIGURATION. ALL SWITCHES SET TO ZERO(0) IS NORMAL.

SW15 1=STOP ON ERROR
0=CONTINUE ON ERROR

SW14 1=YOZZLE ON CURRENT BLOCK
0=DO NOT YOZZLE ON BLOCK

SW13 1=DO NOT CHECK DATA ERRORS
0=CHECK DATA ERRORS

SW12 1=DO NOT CHECK WRITE STATUS ERRORS
0=CHECK WRITE STATUS ERRORS

SW11 1=DO NOT CHECK READ STATUS ERRORS
0=CHECK READ STATUS ERRORS

SW10 1=DO NOT PRINT ANY ERRORS
0=PRINT ALL ERRORS

SW9 1=REWIND ALL AVAILABLE TAPES
0=DO NOT REWIND

SW8 1=GENERATE RANDOM DATA
0=USED FIXED DATA

SW7 1=GENERATE RANDOM CHARACTER COUNT
0=USE FIXED CHARACTER COUNT

SW6 1=GENERATE RANDOM RECORD COUNT
0=USED FIXED RECORD COUNT

SW5 1=YOZZLE ON CURRENT RECORD
0=DO NOT YOZZLE ON RECORD

SW4 1=PRINT STATISTICS
0=DO NOT PRINT STATISTICS

SW3 1=DO NOT PEAD
0=READ

SW2 NOT USED

SW1 1=DISABLE WRITE AND READ RETRY OPTION
0=ENABLE WRITE AND READ RETRY OPTION

SW0 1=DO NOT WRITE
0=WRITE

SWITCH EXPLANATION AND EXAMPLES

SW0+SW3

THESE SWITCHES ARE USED TO CONTROL THE SEQUENCE OF MAG TAPE OPERATIONS PERFORMED ON EACH AVAILABLE UNIT. THE BLOCK OF DATA DESCRIBED THROUGH THE RESPONSES TO TELETYPE REQUESTS AT INITIAL START WILL BE EITHER WRITTEN OR READ FROM EACH AVAILABLE UNIT IN THE ORDER THAT THEY WERE ENTERED. THE SEQUENCE OF OPERATIONS IS CALLED A CYCLE, AND WILL BE PERFORMED CONTINUOUSLY UNTIL STOPPED BY THE OPERATOR. WHEN END OF TAPE IS REACHED, THE UNIT WILL BE REWOUND AND FLAGGED AS UNAVAILABLE FOR TEST UNTIL ALL UNITS HAVE REACHED EOT, AT WHICH TIME TESTING IS RESUMED ON ALL AVAILABLE UNITS.

EXAMPLES SW0+SW3

- A SW0=0, SW3=1 WRITE ONLY X RECORDS OF Y CHARACTERS
- B SW0=1, SW3=0 READ ONLY X RECORDS OF Y CHARACTERS
- C SW0=0, SW3=0 WRITE THEN BACKSPACE AND READ X RECORDS

.11

SWITCH ONE (1), WHEN SET TO A ZERO (0), WILL CAUSE ANY DATA RELATED WRITE ERROR TO BE RETRIED. THE RETRY SCHEME CONSISTS OF REWRITING THE RECORD IN THE SAME SPOT ON THE TAPE FOUR (4) TIMES. IF ALL FOUR (4) REPEATS ARE SUCCESSFUL, THE RECORD IS CONSIDERED RECOVERED, AND A TAPE WRITE ERROR IS LOGGED. IF ANY OF THE FOUR (4) REPEATS IS UNSUCCESSFUL, A WRITE WITH EXTENDED INTERCORD GAP IS DONE, A SUSPECTED BAD TAPE SPOT LOGGED AT THIS BLOCK AND RECORD NUMBER, AND A SECOND RETRY OF FOUR REPEATS IS DONE. IF AFTER FOUR (4) RETRIES, THE RECORD CANNOT BE RECOVERED A NOTIFICATION IS PRINTED, AND TESTING IS RESUMED ON THE NEXT RECORD. IF 20(8) BAD TAPE SPOTS ARE FOUND, THE UNIT WILL BE REWOUND AND REMOVED FROM TESTING WITH AN APPROPRIATE MESSAGE PRINTED.

SWITCH ONE (1), WHEN SET TO A ZERO (0), WILL ALSO CAUSE ANY DATA RELATED READ ERROR TO BE RETRIED. THE RETRY SCHEME CONSISTS OF REREADING THE RECORD A MAXIMUM OF FOUR (4) TIMES. IF THE RECORD IS SUCCESSFULLY RECOVERED ON ANY OF THE REREADS IT IS CONSIDERED FOR STATISTICS PURPOSES TO BE A SOFT READ ERROR AND TESTING CONTINUES. IF THE REREADS FAIL TO RECOVER THE RECORD, THE ERROR IS LOGGED AS A HARD READ ERROR.

SW4

SWITCH FOUR (4) WHEN SET WILL PRINT THE STATISTICS GATHERED FOR EACH UNIT. THE NUMBER WILL BE PRINTED AT THE END OF A BLOCK CYCLE.

SEE ITEM 10, PAGE 20 FOR FULL DETAILS

- SW5 SWITCH FIVE (5) WHEN SET DURING A READ OPERATION WILL CAUSE THE PROGRAM TO CONTINUOUSLY READ THE CURRENT RECORD BY SPACING REVERSE OVER THE RECORD AND REREADING THAT RECORD. THIS TAPE MOVEMENT IS CALLED YOZZLING. THERE IS A SOFTWARE DELAY EXECUTED BETWEEN EACH SPACE/READ OF THE RECORD AND IT MAY BE VARIED BY TYPING CONTROL C ON THE TELETYPE DURING THE EXECUTION OF THE YOZZLE AND RESPONDING TO THE PRINTED REQUEST WITH A SIX (6) DIGIT VALUE. THE YOZZLE STALL IS PRESET TO A VALUE OF 1000 IN THE PROGRAM TO PREVENT EXCESSIVE TAPE WEAR, BUT MAY BE SET TO ANY VALUE THROUGH THE TELETYPE.
- SW6-8 THESE THREE (3) SWITCHES CONTROL THE RANDOMIZATION OF DATA AND BLOCK SIZE AND MAY BE SET AND RESET AT ANY TIME. THE ACTUAL CHANGE WILL TAKE PLACE BETWEEN BLOCK CYCLES.
- SW9 SWITCH NINE (9) WHEN SET WILL CAUSE ALL AVAILABLE TAPE UNITS TO BE REWOUND AT THE END OF THE CURRENT BLOCK CYCLE. TESTING WILL BE RESUMED AT A BLOCK COUNT OF ONE (1) WHEN ALL UNITS HAVE REACHED BOT.
- SW10-13 THESE SWITCHES ARE USED TO CONTROL THE ERROR HANDLING TO BE DONE ON THE TAPE OPERATION DESCRIBED BY SWITCHES 0+3.
- A SWITCH TEN (10) WHEN SET TO A ONE WILL DISALLOW ANY ERROR PRINTOUTS MADE ON THE OPERATION IN PROGRESS. CATASTROPHIC FAILURES AND INFORMATION PRINTOUTS WILL STILL OCCUR. IE UNIT NOT AVAILABLE, ILLEGAL BOT, DROP OR PICK OVERFLOW, AND EOT REWIND.
- B SWITCH ELEVEN (11) WHEN SET TO A ONE WILL DISALLOW THE CHECKING FOR STATUS ERRORS ON READ OPERATIONS.
- C SWITCH TWELVE (12) WHEN SET TO A ONE WILL DISALLOW THE CHECKING FOR STATUS ERRORS ON WRITE OPERATIONS.
- D SWITCH THIRTEEN (13) WHEN SET TO A ONE WILL DISALLOW THE CHECKING OF READ DATA. THIS SWITCH HAS NO EFFECT ON STATUS CHECKING.

SW14

SWITCH FOURTEEN (14) IS USED DURING A READ ONLY OPERATION. WHEN SET, THE BLOCK OF DATA BEING READ WILL CONTINUOUSLY BE READ AND SPACED OVER SO THAT TAPE WILL REMAIN AT THE SAME BLOCK. WHEN RESET, THE TAPE WILL BE ALLOWED TO MOVE FORWARD AND DATA BLOCKS WILL BE READ PROGRESSIVELY. THIS IS A BLOCK YOZZLE.

SW15

SWITCH FIFTEEN (15) WHEN SET TO A ONE, WILL CAUSE THE PROGRAM TO HALT ON ANY ERROR DETECTED BY THE OPERATION IN PROGRESS. IF BOTH SWITCH TEN (10) AND FIFTEEN (15) ARE SET, THE ACTUAL ERROR DETECTED WILL NOT BE PRINTED BUT WILL CAUSE A HALT. IF SWITCH TEN (10) IS RESET BEFORE PRESSING CONTINUE, THE ERROR WHICH CAUSED THE HALT WILL BE PRINTED BEFORE TESTING IS RESUMED.

EPPOR PRINTOUTS

THERE ARE THREE TYPES OF ERROR PRINTOUTS MADE BY THE PROGRAM, OPERATION ERRORS, DATA ERRORS, AND CONDITION ERRORS. EACH ERROR MESSAGE PRINTED IS PRECEDED BY A HEADER WHICH CONTAINS THE UNIT NUMBER, BLOCK COUNT NUMBER, BAD RECORD NUMBER PLUS TOTAL NUMBER OF RECORDS, SIZE OF RECORD, AND TYPE OF OPERATION WHICH CAUSED ERROR

A OPERATION ERRORS

THESE ARE ERRORS WHICH CAN OCCUR AS A DIRECT RESULT OF A TAPE OPERATION

- | | | |
|---|--------------------------|--|
| 1 | READ/WRITE STATUS ERRORS | THESE ARE INDICATED BY THE ERROR BIT (BIT 15) OF THE TAPE COMMAND REGISTER BEING SET TO A ONE |
| 2 | RECORD LENGTH ERRORS | THESE ARE INDICATED BY A BYTE COUNT OTHER THAN ZERO (0) OR AN INCORRECT CURRENT MEMORY ADDRESS OR BOTH |
| 3 | TAPE POSITIONING ERRORS | THESE ARE INDICATED BY A SPACE COUNT OTHER THAN ZERO (0), NO BOT FOUND FROM A REWIND, OR NO TAPE UNIT READY AT THE END OF REWIND |

B DATA ERRORS

DATA ERRORS WILL OCCUR WHEN TAPE IS BEING READ AND THE DATA DOES NOT MATCH THE EXPECTED DATA

BECAUSE DATA RECORDS CAN BE UP TO TWO THOUSAND CHARACTERS LONG, AN ERROR CONDITION WHICH WILL CAUSE THE ENTIRE RECORD TO READ INCORRECTLY COULD CAUSE A VERY LENGTHY PRINTOUT. THEREFORE, A COUNTER OF SUCCESSIVE BAD CHARACTERS IS EMPLOYED. IF TEN (10) CHARACTERS IN SUCCESSION ARE BAD, A NOTIFICATION IS PRINTED (BAD RECORD) AND THE NEXT TWENTY (20) CHARACTERS ARE SKIPPED BEFORE CHECKING IS RESUMED. IF THE BAD RECORD CONDITION OCCURS THREE (3) TIMES IN ONE RECORD, THE REST OF THE RECORD IS SKIPPED, DOWN TO THE LAST TEN (10) CHARACTERS, WHICH WILL BE CHECKED. THE SKIPPING AND RESUMPTION OF CHECKING WILL ONLY BE DONE ON RECORDS WHICH ARE LONG ENOUGH TO ALLOW IT.

CONDITION ERRORS THESE ERRORS REFLECT THE STATE OF THE TAPE SYSTEM BEFORE AND AFTER AN OPERATION

- 1 EOT WHEN AN EOT (END OF TAPE) IS ENCOUNTERED DURING EITHER A READ OR A WRITE, THAT UNIT IS FLAGGED AS UNAVAILABLE FOR TESTING AND IS REWOUND UNTIL ALL AVAILABLE UNITS HAVE REACHED EOT. AT WHICH TIME TESTING IS RESUMED ON ALL AVAILABLE UNITS
- 2 ILLEGAL BOT WHEN A UNIT ENCOUNTERS BEGINNING OF TAPE (BOT) DURING A READ OPERATION THE ERROR IS PRINTED AND THE UNIT DROPPED FROM TESTING UNTIL ALL ARE RESTARTED ON THE NEXT PASS
- 3 DROP DRIVE UNIT BECOMES UNAVAILABLE DUE TO LOSE OF SELECT REMOTE, BOT DURING REWIND, OR NO TUR WHEN MAKING INITIAL SELECTION UNIT IS DROPPED, STATISTICS PRINTED, TESTING WILL RESUME AT BEGINNING OF NEXT PASS
- 4 CONTROLLER NOT READY BEFORE ANY OPERATION IS ATTEMPTED THE CONTROLLER IS CHECKED FOR READY IF IT IS NOT READY, AN ERROR WILL BE PRINTED AND THE PROGRAM WILL STOP
- 5 NO INTERRUPT RETURNED EACH TAPE OPERATION SHOULD BE TERMINATED BY SETTING AN INTERRUPT IN THE CPU IF NO INTERRUPT IS RETURNED WITHIN THE APPROPRIATE TIME AN ERROR IS PRINTED
- 6 NO MORE UNITS TO TEST IF ALL UNITS HAVE BEEN DROPPED FOR CATASTROPH C ERRORS THE PROGRAM WILL STOP

E EXAMPLES

GLOSSARY

BN = BLOCK NUMBER
 RN = RECORD NUMBER (X) OF A TOTAL OF (Y)
 RS = RECORD SIZE IN CHARACTERS PER RECORD
 WE = WRITE ERROR
 RE = READ ERROR
 SE = SPACE ERROR
 F - FORWARD
 CR = COMMAND REGISTER
 CS = STATUS REGISTER
 WC = BYTE COUNTER
 CA = CURRENT MEMORY ADDRESS POINTER AND EXPECTED VALUE
 CN = CHARACTER NUMBER
 G = GOOD DATA (SHOWN IN BIT FORMAT AS IN CORE)
 B = BAD DATA (SHOWN IN BIT FORMAT AS IN CORE)
 ERR AMT = NUMBER LEFT TO SPACE
 TM = TAPE MARK (OFTEN CALLED EOF FOR END OF FILE)
 LPC = LONGITUDINAL PARITY CHECK (RECEIVED - EXPECTED)
 PATRN = DATA PATTERN (R=RANDOM)

EXAMPLE 1

EXAMPLE 1 IN THIS EXAMPLE A TAPE VERTICAL PARITY ERROR WAS DETECTED DURING A WRITE OPERATION OF THE TWELVTH (12) RECORD OF THE BLOCK THE WORD COUNT AND CURRENT MEMORY ADDRESS ARE CORRECT THE RETRY OPTION WAS DISABLED

UNIT NO 3 *DEN 1 *PAR 0 *PATTRN 1
 BN 406*RN 12-200*RS 2000*WE
 CCMO 1010001111000100
 STAT 0001000001000001
 WC 0
 CA 14436-14436

EXAMPLE 2

EXAMPLE 2 IN THIS EXAMPLE A RECORD LENGTH ERROR WAS DETECTED WHILE READING THE FIRST RECORD OF THE BLOCK THE RETRY OPTION WAS DISABLED THE WORD COUNT SHOWS A COUNT OF 20 CHARACTERS LEFT TO BE TRANSFERRED THE CURRENT MEMORY ADDRESS REFLECTS THAT A SHORTAGE OF 20 CHARACTERS TRANSFERRED HAD OCCURRED IN THIS EXAMPLE THE STATUS AND COMMAND REGISTERS DO NOT SHOW ANY ERROR, BUT THE LPC IS SHOWN TO BE INCORRECT

UNIT NO 7 *DEN 2 *PAR 0 *PATTRN 6
 BN 10*RN 1-100*RS 50*RE F**
 COMD 0100011111000100
 STAT 0000000001000001
 WC 20
 CA 12466-12506
 LPC 337 147

EXAMPLE 3

EXAMPLE 3 IN THIS EXAMPLE THE TAPE UNIT WAS TRYING TO SPACE OVER THE 15 RECORDS IN THE BLOCK IN ORDER TO ESTABLISH PROPER POSITION TO BEGIN READING THE OPERATION WAS TERMINATED BEFORE THE ENTIRE 15 RECORDS WERE TRAVERSED AND AN ERROR SHOWN BECAUSE THE TAPE IS NOT IN PROPER POSITION TO BEGIN READING

UNIT NO 0 *PATTRN R
 BN 2*RN 15-15*PS 27 *SE
 ERR AMT 4

EXAMPLE 4

EXAMPLE 4 IN THIS EXAMPLE UNIT NUMBER ONE (1) HAD BEEN
REWOUND VIA CONSOLE SWITCH NINE (9) AND AT THE
COMPLETION OF THE OPERATION BOT WAS NOT SET IN
THE STATUS REGISTER

UNIT NO 1 *DEN 3 *PAR 0 *PATTRN R
BN 3002*RN 65-65*RS 10
NC BOT ON REWIND-HALT

EXAMPLE 5

EXAMPLE 5 IN THIS EXAMPLE TWO BAD CHARACTERS WERE
READ FROM TAPE IN THE FORWARD DIRECTION
THE FIRST (0) AND THE THIRTEENTH (13) CHARACTERS
OF THE TOTAL NUMBER OF SIXTEEN (16) CHARACTERS
IN THE BLOCK ARE BAD CHARACTER NUMBER
ZERO (0) HAS DROPPED BIT NUMBER FIVE (5) AND
CHARACTER NUMBER TWELVE (12) HAS PICKED UP
BIT NUMBER SEVEN (7)

UNIT NO 5 *DEN 3 *PAR 0 *PATTRN 5
BN 12*RN 3 10*RS 15*DE-F**
CN 0
G 10101010
B 10001010
CN 12
G 01010101
B 11010101

EXAMPLE 6

EXAMPLE 6 IN THIS EXAMPLE UNIT NUMBER SIX (6) HAS REACHED END OF TAPE (EOT) FOR THE 1ST TIME AND WILL BE REWOUND. TESTING WILL RESTART ON UNIT NUMBER SIX (6) WHEN ALL UNITS HAVE REACHED EOT.

UNIT NO 6 *DEN 3 *PAR 0 *PATRN R
BN 677 *RN 25-600*RS 1566
EOT NO 1
UNIT WILL REWIND AND BE
PESTARTED ON BLOCK ONE
WHEN ALL AVAIL UNITS REACH EOT

EXAMPLE 7

EXAMPLE 7 IN THIS EXAMPLE UNIT NUMBER TWO (2) HAS ENCOUNTERED BEGINNING OF TAPE (BOT) DRIVE WILL BE DROPPED. STATISTICS WILL BE PRINTED, TESTING RESUMED AT BEGINNING OF NEXT PASS.

UNIT NO 2 *DEN 2 *PAR 0 *PATRN 2
BN 56 *RN 2-4 *RS 1200
ILLEGAL BOT

EXAMPLE 8

EXAMPLE 8 IN THIS EXAMPLE THE SELECTED UNIT (NUMBER 0) HAS BECOME UNAVAILABLE. UNIT WILL BE DROPPED. STATISTICS WILL BE PRINTED, TESTING WILL RESUME AT BEGINNING OF NEXT PASS.

UNIT NO 3 *DEN 1 *PAR 0 *PATRN 4
BN 1 *RN 0-200 *RS 66 NOT AVAIL
COR LOST SELECT REMOTE, NO BOT ON REWIND

EXAMPLE 9

EXAMPLE 9 IN THIS EXAMPLE THE WRITE OPERATION EXECUTED ON UNIT NUMBER SIX (6) WAS NOT COMPLETED AND NO INTERRUPT WAS RETURNED.

UNIT NO 6 *DEN 2 *PAR 0 *PATRN P
BN 12 *RN 3-4 *RS 100 *WE
NO INTERRUPT RETURNED

EXAMPLE 10

EXAMPLE 10 THIS EXAMPLE SHOWS A READ ERROR WHICH RECOVERED ON THE SECOND RETRY THIS ERROR WILL BE LOGGED AS A RDERR BUT WILL BE CATEGORIZED AS A SOFT ERROR THE REGISTERS SHOW A PARITY ERROR WAS THE CAUSE OF THE ERROR

UNIT NO 1 *DEN 3 *PAR 1 *PATTRN R
 *BN 10 *RN 2-100 *RS 1117 *RE F***
 COMD 1110100110000010
 STAT 0011000001000001
 WC 0
 LPC 337-147
 ORIGINAL ERROR

UNIT NO 1 *DEN 3 *PAR 0 *PATTRN R
 *BN 10 *RN 2-100 *RS 1117 *RE F***
 COMD 1110100110000010
 STAT 0011000001000001
 WC 0
 LPC 337-147
 READ FAILED--RETRY 1
 PEREAD SUCCESSFUL--RETRY 2

EXAMPLE 11

EXAMPLE 11 THIS EXAMPLE SHOWS A WRITE ERROR WHICH WAS NOT RECOVERED BY SUCCESSFULLY REWRITTING THE RECORD FOUR TIMES AT THAT LOCATION THE RECORD WAS SUCCESSFULLY WRITTEN AFTER 2 INCHES OF TAPE WAS ERASED THIS ERROR WILL BE LOGGED AS A BAD TAPE SPOT

UNIT NO 0 *DEN 3 *PAR 0 *PATTRN R
 *BN 2 *RN 370 -461 *PS 2407 *WE
 COMD 1110000010000100
 STAT 0011000001000001
 WC 0
 CA 25613 -25613
 ORIGINAL ERROR

UNIT NO 0 *DEN 3 *PAR 0 *PATTRN R
 *BN 2 *RN 370 -461 *PS 2407 *WE
 COMD 1110000010000100
 STAT 0011000001000001
 WC 0
 CA 25613 -25613
 SUSPECT BAD TAPE
 RETRY 0
 REPEAT 0
 RECOVERED
 RETRY 1

STATISTICS PRINTOUT

THE PROGRAM GATHERS A VARIETY OF STATISTICS DURING THE COURSE OF ITS TESTING. THE STATISTICS ARE KEPT ON A UNIT BY UNIT BASIS AND ARE SUMMARIZED IN A STATISTICS PRINTOUT. STATISTIC PRINTOUTS CAN BE PRINTED AT THE END OF EACH BLOCK CYCLE BY SETTING SWITCH FOUR (4) TO 1. A STATISTIC PRINTOUT IS AUTOMATICALLY PRINTED WHEN A UNIT REACHES EOT AND IS REWOUND.

HERE IS AN EXPLANATION OF THE STATISTIC SUMMARY

- CPDPS THE NUMBER OF BITS DROPPED ON A PER TRACK BASIS. DROPS ARE COLLECTED DURING THE DATA CHECK ROUTINE.
- DPDPS THE NUMBER OF BITS PICKED ON A PER TRACK BASIS. DROPS ARE COLLECTED DURING THE DATA CHECK ROUTINE.
- WTERR THE NUMBER OF RECORDS IN WHICH A WRITE ERROR OCCURRED. IF WRITE RETRY WAS ENABLED, WTERR WILL CONTAIN ONLY THOSE RECORDS WHICH WERE NOT RECOVERED AFTER ONE RETRY.
- PTRY THE NUMBER OF RETRIES INITIATED UNDER THE WRITE RETRY OPTION (SEE ITEM 8, SW1).
- RDERR THE TOTAL NUMBER OF RECORDS IN WHICH A READ ERROR OCCURRED.
- SOFT THE NUMBER OF READ ERRORS WHICH WERE RECOVERED WITHIN A MAXIMUM OF FOUR REPEATS OF A RECORD UNDER THE READ RETRY OPTION (SEE ITEM 8, SW1).
**NOTE SOFT READ ERRORS ARE ONLY CATEGORIZED FOR THOSE READ ERRORS OCCURRING WHEN CONSOLE SWITCH 1 IS SET TO ZERO.
- HARD THE NUMBER OF READ ERRORS WHICH REMAINED UNRECOVERED UNDER THE READ RETRY SCHEME (SEE ITEM 8, SW1).
**NOTE HARD READ ERRORS ARE ONLY CATEGORIZED FOR THOSE READ ERRORS OCCURRING WHEN CONSOLE SWITCH 1 IS SET TO ZERO.
- DTERR THE NUMBER OF DATA ERRORS FOUND FOR THIS UNIT.
**NOTE DATA ERRORS ARE ONLY FOUND FOR THOSE RECORDS WHICH WERE READ WITH SWITCH 11 RESET TO ZERO.

BAD TAPE SPOTS A COUNT OF THE NUMBER OF TAPE SPOTS
WHERE A RECORD COULD NOT BE REWRITTEN SUCCESSFULLY
UNDER THE WRITE RETRY OPTION (SEE ITEM 8, SW1)
FOLLOWING THE COUNT IS A LIST OF THE BAD TAPE
LOCATIONS IDENTIFIED BY THE BLOCK AND RECORD NUMBER
WHEN THE BAD TAPE SPOT WAS LOGGED

EXAMPLE

DROPS 0 0 0 0 7 0 0 0
PICKS 0 0 0 2 0 0 0 0
WTERR 3
RTY 4
PDEPP 6
SOFT 1
HARD 5
CTEPP 10
1 BAD TAPE SPOTS
2 *EN 16 *PN 41

AUTO SEQUENCE

THE AUTO SEQUENCE (START AT ADDRESS 240) WILL EXECUTE A PREDETERMINED TEST PLAN ON ALL AVAILABLE UNITS. THE ONLY OPERATOR RESPONSE REQUIRED IS TO THE TYPED REQUESTS FOR THE CONTROLLER ADDRESS AND VECTOR AND CONTINUOUS OR SINGLE CYCLE. ALL SWITCHES REMAIN ACTIVE AND MAY BE USED NORMALLY, HOWEVER, THE INTENT IS TO LEAVE ALL SWITCHES DOWN AND ALLOW FULL EXECUTION OF THE TEST PLAN FOR SYSTEM CHECKOUT.

SAMPLE START AT 240(8) AUTO SEQUENCE

LOAD ADDRESS 240(8), SET SWITCHES TO ZERO, PRESS START

TM, A, B-11 AUTO SEQUENCE TEST
ENTER RESPONSES IN OCTAL

REGISTER START = 172520 (CR)
VECTOR = 224 (CR)
AUTO CONT 0 (1)

THIS EXAMPLE SHOWS AN AUTO SEQUENCE START WITH THE CONTROLLER AT BUS ADDRESS 172520 AND A VECTOR OR 224. ALL AVAILABLE UNITS WILL BE TESTED CONTINUOUSLY.

AS EACH PASS IS COMPLETED A DIVIDER LINE OF ASTERISKS WILL BE PRINTED FOLLOWED BY AN END OF PASS MESSAGE INDICATING HOW MANY PASSES HAVE BEEN COMPLETED SINCE THE AUTO SEQUENCE WAS BEGUN. AT THE START OF EACH PASS THE UNITS BEING TESTED ARE PRINTED.

AUTO SEQUENCE TEST PLAN

THE AUTO SEQUENCER WILL EXECUTE A PASS CONSISTING OF THE WRITING, READING, AND CHECKING OF SEVERAL DIFFERENT DATA PATTERNS. EACH PASS WILL START AT BOT AND PROCESS AN ENTIRE MAG TAPE BEFORE REWINDING.

THE UNITS WILL BE SET UP TO WRITE 800 BPI IN NINE TRACK FORMAT. ODD PARITY WILL BE USED AND NO TAPE MARKS WILL BE WRITTEN.

THE DATA PATTERNS WILL BE AS FOLLOWS

THREE FIXED DATA PATTERNS

EACH PATTERN WILL BE USED FOR SIX BLOCKS
EACH BLOCK CONSISTS OF (100) 4000 CHARACTER RECORDS

PATTERN 3 WALKING ONE BIT
PATTERN 7 ALTERNATING ONE AND ZERO BITS
PATTERN 11 INCREMENTING CHARACTERS (000-377)

RANDOM DATA

FOLLOWING THE FIXED DATA PATTERNS, RANDOM
DATA WILL BE WRITTEN IN THE SAME BLOCK STRUCTURE
UNTIL EOT IS REACHED
IT IS IMPORTANT THAT THE TAPE USED FOR THE TEST
BE OF SUFFICIENT LENGTH TO ACCOMODATE ALL OF
THE FIXED DATA PATTERNS AND AT LEAST ONE
RECORD OF RANDOM DATA, OTHERWISE, THE TAPE WILL
BE REWOUND UNTIL ALL OF THE DATA PATTERNS
HAVE BEEN TESTED

12

TESTING PROCEDURES

AS PREVIOUSLY STATED THIS PROGRAM CONTAINS NO FIXED TESTS. THE ENTIRE TEST CYCLE TO BE EXECUTED IS DESCRIBED BY THE OPERATOR THROUGH RESPONSES TO TELETYPE REQUESTS FOR PARAMETERS AND CONSOLE SWITCH SETTINGS FOR OPERATION. THE OPERATION SELECTED WILL BE EXECUTED WITH THE PARAMETERS ENTERED CONTINUOUSLY ON EACH AVAILABLE UNIT. ONE BLOCK AT A TIME, UNTIL STOPPED BY THE OPERATOR. THE OPERATION MAY BE CHANGED DYNAMICALLY BY CHANGING THE CONSOLE SWITCHES AT ANY TIME. THE PROGRAM WILL ATTEMPT TO PERFORM ANY OPERATION SET AND THEREFORE CAUTION SHOULD BE TAKEN TO ASSURE THAT THE UNIT IS CAPABLE OF PERFORMING AS REQUESTED. FOR INSTANCE, ONE SHOULD NOT ATTEMPT TO PERFORM READ OPERATIONS ON A TAPE WHICH HAS NOT BEEN WRITTEN AS THE DATA, IF ANY, IS UNPREDICTABLE. HOWEVER, IF A TAPE HAS BEEN WRITTEN WITH THIS PROGRAM, IT CAN BE READ AS OFTEN AS DESIRED WITHOUT BEING REWRITTEN. THIS IS A GOOD PROCEDURE TO USE FOR TESTING TAPE COMPATABILITY. SCOPING OF TAPE UNITS BECOMES SIMPLE, BY SETTING THE DESIRED OPERATION AND ITS PARAMETER. A UNIT MAY BE CONTINUOUSLY EXERCISED IN ANY MANNER DESIRED BY USING THE VARIOUS ERROR CONTROL SWITCHES AND ENTERING THE NEEDED STALL. ANY FUNCTION CAN BE SCOPED RATHER EASILY. RELIABILITY TESTING CAN BE PERFORMED BY USE OF THE RANDOMIZATION CAPABILITY. PERHAPS A CYCLE OF RANDOM TESTING MIGHT BE SET UP AND ALLOWED TO RUN FOR SOME PERIOD OF TIME, THE STATISTICAL COLLECTION OF DROPS AND PICKS IS THEN SIGNIFICANT. INTERMITTANT PROBLEMS CAN BE FOUND BY SETTING THE DESIRED OPERATION IN MOTION AND DISALLOWING ERROR PRINTOUTS WHILE ALLOWING A HALT ON ERROR. THE ERROR THAT CAUSED THE HALT CAN BE PRINTED BY RESETTING CONSOLE SWITCH TEN AND PRESSING CONTINUE. IF SOME PARTICULAR DATA PATTERN SHOULD BE CAUSING DATA ERROR, USE OF THE YOZZLE SWITCH AND ITS ASSOCIATED STALL CAN BE USED TO ALLOW SCOPING OF THIS PARTICULAR RECORD.

AS YOU SEE, THERE ARE MYRIAD TESTING PROCEDURES WHICH COULD BE PERFORMED. THE PARAMETERS, TAPE OPERATIONS, ERROR EXAMINATION AND REPORTING ARE ALL AT YOUR DISCRETION.

TRY IT. YOU'LL LIKE IT.

13

LISTING

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55

TITLE TM, A, B-11 TSO3 OR TU10, N, W MULTIDRIVE DATA RELIABILITY EXERCISER
, MAINDEC-11-DZTMH-F-D
, 15-NOVEMBER-1977
, R B BARNES/RON PLATUKIS/R SOLER
ENABLE ABS,AMA

, CONSOLE SWITCHES*****

, SW15 1=STOP ON ERROR
, 0=CONTINUE ON ERROR

, SW14 1=YOZZLE ON CURRENT BLOCK
, 0=DO NOT YOZZLE ON BLOCK

, SW13 1=DO NOT CHECK DATA
, 0=CHECK DATA

, SW12 1=DO NOT CHECK WRITE ERRORS
, 0=CHECK WRITE ERRORS

, SW11 1=DO NOT CHECK READ ERRORS
, 0=CHECK READ ERRORS

, SW10 1=DO NOT PRINT ERRORS
, 0=PPINT ERRORS

, SW9 1=REWIND TAPE
, 0=DO NOT REWIND

, SW8 1=USE RANDOM DATA
, 0=USE FIXED DATA PATTERN

, SW7 1=USE RANDOM CHARACTER COUNT
, 0=USE FIXED CHAR COUNT

, SW6 1=USE RANDOM RECORD COUNT
, 0=USE FIXED RECORD COUNT

, SW5 1=YOZZLE ON CURRENT RECORD
, 0=DO NOT YOZZLE ON RECORD

, SW4 1=PRINT DROPS AND PICKS
, 0=DO NOT PRINT DROPS AND PICKS

, SW3 1=DO NOT READ FORWARD
, 0=READ FORWARD

, SW2 NOT USED

, SW1 1=INHIBIT WRITE AND READ RETRY
, 0=ENABLE WRITE AND READ RETRY

, SW0 1=DO NOT WRITE
, 0=WRITE

```

56
57
58
59          000000          RO=%0
60          000001          R1=%1
61          000002          R2=%2
62          000003          R3=%3
63          000004          R4=%4
64          000005          R5=%5
65          000006          SP=%6
66          000007          PC=%7
67          000240          NOP=240
68
69
70
71          000000          ,TRAP CATCHERS*****
72          000042          =0
73          000042          =42
74          SBTTL ACT11 HOOKS
75
76          ,*****
77          ,HOOKS REQUIRED BY ACT11
78          000042          $SVPCL=          ,SAVE PC
79          000046          =46
80          000046          SENDAD          ,1)SET LOC 46 TO ADDRESS OF SENDAD IN SEOP
81          000052          =52
82          000000          WORD 0          ,2)SET LOC 52 TO ZERO
83          000042          =5SVPCL          ,RESTORE PC
84
85          ,TTY INTEPRUPT VECTOR*****
86          000060          =60
87          000060          017404          TTYINT          ,TTY INTEPRUPT HANDLER ADDRESS
88          000062          000000
89
90
91          ,SOFTWARE SWITCH REGISTER LOCATIONS*****
92
93          000174          =174
94          000174          000000          DISPREG 0
95          000176          000000          SWREG. 0
96
97          ,START ADDRESS*****
98
99          000200          =200
100         000200          000137          002772          JMP START          ,ENTER PARAMETERS VIA TTY
101
102          000204          =204
103         000204          000137          003124          JMP STARTA          ,USE FIXED PARAMETERS, HOLD DATA
104
105          000210          =210
106         000210          005037          013304          CHAIN CLR RDFL
107         000214          000137          003142          JMP STARTE          ,USE FIXED PARAMETERS, NEW DATA
108
    
```

```
109  
110          ,MAG TAPE INTERRUPT VECTOR*****  
111  
112          =224  
113 000224 017466 MTINT          ,MAG TAPE INTERRUPT HANDLER ADDRESS  
114 000226 000340 340  
115  
116          ,AUTO SEQUENCE START*****  
117          =240  
118 000240 005237 021632 INC      ASEQF          ,SET AUTO SEQUENCE FLAG  
119 000244 000137 003106 JMP      STAUT          ,GO TO START OF AUTO SEQ
```

120	000600		=600		
121			,CONSTANTS*****		
122					
123	000600	172520	MTS	172520	,TAPE STATUS REGISTER
124	000602	172522	MTC	172522	,TAPE COMMAND REGISTER
125	000604	172524	MWC	172524	,TAPE CHARACTER COUNT REGISTER
126	000606	172526	MDA	172526	,TAPE DATA ADDRESS REGISTER
127	000610	172530	MTD	172530	,TAPE DATA BUFFER
128	000612	172532	MTRD	172532	,TAPE READ LINES
129	000614	000224	VECT	224	, INTERRUPT VECTOR ADDRESS
130	000616	000000	UDES	0	, UNIT DESCRIPTION (PARITY, DENSITY, UNIT, TRACK)
131	000620	000100	RCNT	100	, RECORD COUNTER
132	000622	177600	CARCNT	177600	, NUMBER OF CHAR (2 - 4000) OCTAL IN TWOS COMPLEMENT
133	000624	000001	PATRN	1	, DATA PATTERN SELECTOR (0 - 20) OCTAL
134	000626	000002	RDCMD	2	, READ COMMAND
135	000630	000001	SPFLG	1	, SINGLE PASS FLAG
136	000632	000001	RSTAL	1	, READ STALL
137	000634	000001	WSTAL	1	, WRITE STALL
138	000636	000001	TSTAL	1	, TURN AROUND STAL
139	000640	001000	YSTAL	1000	, YOZZLE STAL
140	000642	000100	RCSAV	100	, RECORD COUNT SAVE
141	000644	177600	CCSAV	-200	, CHARACTER COUNT SAVE
142	000646	000000	TMEX	0	, TAPE MARK FLAG 1=TM 0=NO TM
143	000650	177776	PSW	177776	, PROCESSOR STATUS
144	000652	177570	SWR	177570	, CONSOLE SWITCHES
145	000654	177570	DISPLAY	177570	
146	000656	177560	TAS	177560	, TTY READ STATUS REGISTER
147	000660	177562	TAB	177562	, TTY READ BUFFER
148	000662	177564	TPS	177564	, TTY PUNCH STATUS REGISTER
149	000664	177566	TPB	177566	, TTY PUNCH OUTPUT REGISTER
150	000666	177550	PRS	177550	, H/S READER STATUS REGISTER
151	000670	177552	PRB	177552	, H/S READER BUFFER
152	000672	153624	RANBAS	153624	, RANDOM NUMBER GENERATOR BASE
153	000674	172520	REGST	172520	, STARTING REGISTER ADDRESS
154	000676	032561	RANSAV	032561	, RANDOM NUMBER BUFFER
155					

156				
157				
158				
159	000700	000000	TINF	0
160	000702	000000	TOB.	0
161	000704	000000	TIB	0
162	000706	000000	TEMP1	0
163	000710	000000	TEMP2	0
164	000712	000000	TEMP3	0
165	000714	000000	TEMP4	0
166	000716	000000	EMADDR	0
167	000720	000000	BLCNTR	0
168	000722	000000	BBC	0
169	000724	000000	RTRN	0
170	000726	000000	HDRFL	0
171	000730	000000	STAL	0
172	000732	000000	PFLG	0
173	000734	000000	UNP	0
174	000736	000000	BCNT	0
175	000740	000000	ERSAV	0
176	000742	000000	SERFL	0
177	000744	000000	DERFL	0
178	000746	000000	BTFLG	0
179	000750	000000	RPCNT	0
180	000752	000000	PTCNT	0
181	000754	000000	RTYFL	0
182	000756	000000	TMFLG	0
183	000760	000000	EOTREC	0
184	000762	000000	BTPT	0
185	000764	000000	ERTFL	0
186	000766	000000	BDPP	0
187	000770	000000	BPKP	0
188	000772	000000	BTSTF	0
189	000774	000000	RRTYFL	0
190	000776	000000	SEQCT	0
191	001000	000000	COUNT	0
192	001002	000000	TEMPST	0
193	001004	000000	RDSW	0
194	001306	000000	DUCTR	0
195	001010	000000	STCDFL	0

. FLAGS AND COUNTERS*****

. TTY ENTERY FLAG
 . TTY OUTPUT BUFFER
 . TTY INPUT BUFFER
 . TEMP STORAGE
 . TEMP STORAGE
 . TEMP STORAGE
 . TEMP STORAGE
 . ERROR MSG ADDRESS STORAGE
 . BLOCK COUNTER
 . BAD RECORD COUNTER
 . INTERRUPT RETURN STORAGE
 . HEADER FLAG
 . DELAY STORAGE
 . PRINT FLAG
 . UNIT TABLE POINTER
 . BIT COUNTER
 . STATUS STORAGE
 . STATUS ERROR FLAG
 . DATA ERROR FLAG
 . BAD TAPE FLAG
 . REPEAT COUNTER
 . RETRY COUNTER
 . RETRY FLAG
 . TM FLAG
 . END OF TAPE RECORD
 . BAD TAPE POINTER
 . ERASE TAPE FLAG
 . DROP POINTER
 . PICK POINTER
 . BAD TAPE STATISTICS FLAG
 . READ RETRY FLAG
 . AUTO SEQ PASS COUNT

 . DROPPED UNIT COUNTER
 . TRM COPE DUMP FLAG

197
 198
 199
 200
 201
 202
 203
 204
 205
 206
 207
 208
 209
 210
 211
 212
 213
 214
 215
 216
 217
 218
 219
 220
 221
 222
 223
 224
 225
 226
 227
 228
 229
 230
 231
 232
 233
 234
 235
 236
 237
 238
 239
 240
 241
 242
 243
 244
 245
 246
 247
 248
 249
 250

UNIT ORDER AND DESCRIPTION TABLE *****

UN1	60000	, THIS TABLE IS LOADED
UN2	-1	, WITH UNIT NUMBERS AND
UN3	-1	, THEIR DESCRIPTIONS IN
UN4	-1	, THE ORDER THAT THEY
UN5	-1	, WILL BE TESTED
UN6	-1	
UN7	-1	
UN8	-1	
UNX	-1	

. UNIT DROPS AND PICKS COUNTERS*****

PIK1	BP00
PIK2	BP10
PIK3	BP20
PIK4	BP30
PIK5	BP40
PIK6	BP50
PIK7	BP60
PIK8	BP70
DRP1	BD00
DRP2	BD10
DRP3	BD20
DRP4	BD30
DRP5	BD40
DRP6	BD50
DRP7	BD60
DRP8	BD70

. UNIT WRITE ERRORS*****

WTER1	0
WTER2	0
WTER3	0
WTER4	0
WTER5	0
WTER6	0
WTER7	0
WTER8	0

. UNIT READ ERRORS*****

RDER1	0
RDER2	0
RDER3	0
RDER4	0
RDER5	0
RDER6	0
RDER7	0
RDER8	0

251 .UNIT DATA ERRORS*****
252
253 001134 000000 DATER1 0
254 001136 000000 DATER2 0
255 001140 000000 DATER3 0
256 001142 000000 DATER4 0
257 001144 000000 DATER5 0
258 001146 000000 DATER6 0
259 001150 000000 DATER7 0
260 001152 000000 DATER8 0

261 .UNIT RETRY COUNTERS*****
262
263
264 001154 000000 RTY1 0
265 001156 000000 RTY2 0
266 001160 000000 RTY3 0
267 001162 000000 RTY4 0
268 001164 000000 RTY5 0
269 001166 000000 RTY6 0
270 001170 000000 RTY7 0
271 001172 000000 RTY8 0

272 .UNIT SOFT READ ERRORS*****
273
274
275 001174 000000 GDRTY1 0
276 001176 000000 GDPTY2 0
277 001200 000000 GDRTY3 0
278 001202 000000 GDRTY4 0
279 001204 000000 GDRTY5 0
280 001206 000000 GDRTY6 0
281 001210 000000 GDRTY7 0
282 001212 000000 GDPTY8 0

283 .UNIT HARD READ ERRORS*****
284
285
286 001214 000000 BDRTY1 0
287 001216 000000 BDPTY2 0
288 001220 000000 BDRTY3 0
289 001222 000000 BDRTY4 0
290 001224 000000 BDRTY5 0
291 001226 000000 BDRTY6 0
292 001230 000000 BDRTY7 0
293 001232 000000 BDRTY8 0

294 .UNIT EGT COUNTERS*****
295
296
297 001234 000000 EOTCT1 0
298 001236 000000 EOTCT2 0
299 001240 000000 EOTCT3 0
300 001242 000000 EOTCT4 0
301 001244 000000 EOTCT5 0
302 001246 000000 EOTCT6 0
303 001250 000000 EOTCT7 0
304 001252 000000 EOTCT8 0
305

306				
307				
308				
309	001254	000000	BP00	0
310		001274		= +16
311	001274	000000	BP10	0
312		001314		= +16
313	001314	000000	BP20	0
314		001334		= +16
315	001334	000000	BP30	0
316		001354		= +16
317	001354	000000	BP40	0
318		001374		= +16
319	001374	000000	BP50	0
320		001414		= +16
321	001414	000000	BP60	0
322		001434		= +16
323	001434	000000	BP70	0
324		001454		= +16
325	001454	000000	B000	0
326		001474		= +16
327	001474	000000	B010	0
328		001514		= +16
329	001514	000000	B020	0
330		001534		= +16
331	001534	000000	B030	0
332		001554		= +16
333	001554	000000	B040	0
334		001574		= +16
335	001574	000000	B050	0
336		001614		= +16
337	001614	000000	B060	0
338		001634		= +16
339	001634	000000	B070	0
340		001654		= +16
341				
342				
343				
344	001654	000000	BT00	0
345		001760		= +102
346	001760	000000	BT01	0
347		002064		= +102
348	002064	000000	BT02	0
349		002170		= +102
350	002170	000000	BT03	0
351		002274		= +102
352	002274	000000	BT04	0
353		002400		= +102
354	002400	000000	BT05	0
355		002504		= +102
356	002504	000000	BT06	0
357		002610		= +102
358	002610	000000	BT07	0
359		002714		= +102

360
 361
 362
 363
 364
 365
 366
 367
 368
 369
 370
 371
 372
 373
 374
 375
 376
 377
 378
 379
 380
 381
 382
 383
 384
 385
 386
 387
 388
 389
 390

. UNIT BAD TAPE POINTERS*****

BTADDR BT00
 BT01
 BT02
 BT03
 BT04
 BT05
 BT06
 BT07

002714 001654
 002716 001760
 002720 002064
 002722 002170
 002724 002274
 002726 002400
 002730 002504
 002732 002610

. DATA PATTERN GENERATORS*****

DATBL
 DATA0 DAT0
 DATA1 DAT1
 DATA2 DAT2
 DATA3 DAT3
 DATA4 DAT4
 DATA5 DAT5
 DATA6 DAT6
 DATA7 DAT7
 DATA10 DAT10
 DATA11 DAT11
 DATA12 DAT12
 DATA13 DAT13
 DATA14 DAT14
 DATA15 DAT15

. ENTRY TABLE
 . EXTERNAL INPUT FROM H/S READER
 . ALL ONES
 . ALL ZEROS
 . WALKING ONE
 . WALKING ZERO
 . ALTERNATING ONE/ZERO
 . ALTERNATING ZERO/ONE
 . ALTERNATING ONE/ZERO IN ALTERNATING CHARACTERS
 . ALTERNATING ZERO/ONE IN ALTERNATING CHARACTERS
 . ALL BITS 0-377
 . ALL BITS 377-0
 . ALTERNATING CHARACTERS 0 AND 377
 . ALTERNATING CHARACTERS 377 AND 0
 . WALKING ZERO REPEATED FOUR TIMES

002734 002734
 002736 012552
 002740 012754
 002742 012776
 002744 013004
 002746 013032
 002750 013044
 002752 013054
 002754 013064
 002756 013074
 002760 013104
 002762 013126
 002764 013152
 002766 013162
 002770 013172

```

391
392
393
394          EVEN
395          ,*****
396          ,PROGRAM START AND SEQUENCE FORMATTER
397
398          ,THIS ROUTINE IS USED TO PERFORM ALL HOUSEKEEPING,
399          ,DECIDE WHICH TRANSPORT TO TEST AND ITS AVAILABILITY,
400          ,LOAD THE WRITE BUFFER WITH THE SELECTED DATA PATTERN,
401          ,GENERATE ANY RANDOM NUMBER AND THEN EXECUTE
402          ,THE TEST CYCLE REQUESTED BY THE SWITCH SETTING
403          ,AT THE END OF THE TEST CYCLE THE NEXT UNIT IS SELECTED
404          ,AND CHECKED FOR AVAILABILITY AND THE TEST CYCLE IS
405          ,EXECUTED ON IT
406          ,THE NUMBER OF BITS DROPPED OR PICKED MAY BE PRINTED
407          ,AT THE END OF EACH TEST CYCLE VIA CONSOLE SWITCH FOUR (4)
408          ,*****
409
410
411 002772 005037 021632          START CLR      ASEQF      ,CLEAR AUTO SEQ FLAG
412 002776 012737 177570 000652 MOV      #177570,SWR  ,PRESET FOR CONSOLE SWITCHES
413 003004 005737 000042          TST      @#42      ,SEE IF CHAIN MODE
414 003010 001436          BEQ      STAUT     ,IF NOT BR
415 003012 012706 000500          MOV      #500,SP   ,SET UP STACK POINTER
416 003016 012704 023152          MOV      #MSG31,R4
417 003022 004737 020536          JSR      PC,TTOUT  ,PRINT TITLE
418 003026 122737 000004 000041 CMPB     #4,@#41    ,SEE IF LOAD MEDIUM
419 003034 001006          BNE     1$        ,IF NOT BR
420 003036 012704 026124          MOV      #MSG97,R4
421 003042 004737 020536          JSR      PC,TTOUT  ,PRINT NO TEST LOAD MEDIUM
422 003046 000137 004622          JMP      REOT8     ,END TEST
423 003052 012737 000176 000652 1$ MOV      #176,SWR   ,SET FOR SOFTWARE SWITCHES
424 003060 012700 001014          MOV      #UN2,RO   ,SET UNIT POINTER
425 003064 022720 177777          2$ CMP      #-1,(RO)+ ,SEE IF END OF UNITS
426 003070 001404          BEQ     3$        ,IF SO BR
427 003072 062737 000401 004716 ADD      #401,REOTC ,ELSE BUMP UNIT EOT COUNTER
428 003100 000771          BR      2$
429 003102 000137 000210          3$ JMP      CHAIN     ,GO DO CHAIN START
430 003106 012737 000001 000700 STAUT   MOV      #1,TINF ,SET TTY ENTRY FLAG
431 003114 005037 013304          CLR      RDFL     ,CLEAR RANDOM DATA FLAG
432 003120 000137 003146          JMP      STARTB
433 003124 005037 000700          STARTA CLR      TINF   ,CLEAR TTY ENTRY FLAG
434 003130 012706 000500          MOV      #500,SP  ,SET STACK POINTER
435 003134 004737 022046          JSR      PC,SUSWR ,CHECK FOR SOFTSWR
436 003140 000451          BR      STAUTO
437 003142 005037 000700          STARTE CLR      TINF  ,CLEAR INPUT FLAG
438 003146 012700 000702          STARTB MOV      #TOB,RO
439 003152 012701 000044          MOV      #44,R1
440 003156 005020          STARTO CLR      (RO)+  ,CLEAR FLAGS AND COUNTERS
441 003160 005301          DEC     R1
442 003162 001375          BNE     STARTO
443 003164 012700 000510          MOV      #510,RO  ,SET SIZE OF TABLE
444 003170 012701 001074          MOV      #WTER1,R1 ,SET START OF TABLE
445 003174 005021          STARTC CLR      R1)+   ,CLEAR STATISTICS TABLES
446 003176 005300          DEC     RO
    
```

L 3

447	003200	001375			BNE	STARTC	, CLEAR ALL
448	003202	012737	177777	012750	MOV	#-1, PATS	, RESET PATTERN
449	003210	012737	177777	012752	MOV	#-1, PARS	, RESET PARITY
450	003216	012737	000001	000720	MOV	#1, BLCNTR	, PRESET BLOCK COUNTER
451	003224	005077	175352		CLR	DMTC	
452	003230	052777	010000	175344	BIS	#10000, DMTC	, POWER CLEAR CONTROLLER
453	003236	012706	000500		STARTD	MOV	#500, SP
454	003242	004737	022046		JSR	PC, SUSWR	, CHECK FOR SORTSWR
455	003246	012777	000340	175374	15.	MOV	#340, APSW
456	003254	004737	010722		JSR	PC, TINP	, GO GET PARAMETERS FROM TTY
457	003260	004737	004110		JSR	PC, RANSET	, GO RESET BASE
458	003264	005000			STAUTO	CLR	RO
459	003266	022737	000176	000652		CMR	#SWREG, SWR
460	003274	001005				BNE	STAROA
461	003276	005737	000042			TST	DM42
462	003302	001002				BNE	STAROA
463	003304	004737	022172			JSR	PC, CNTLU
464	003310	005160	001012		STAROA	COM	UN1(RO)
465	003314	001411				BEQ	STAROB
466	003316	005160	001012			COM	UN1(RO)
467	003322	042760	100200	001012		BIC	#100200, UN1(RO)
468	003330	062700	000002			ADD	#2, RO
469	003334	000137	003310			JMP	STAROA
470	003340	005160	001012		STAROB	COM	UN1(RO)
471	003344	013703	004716			MOV	REOTC, R3
472	003350	000303				SWAB	R3
473	003352	110337	004716			MOVB	R3, REOTC
474	003356	012777	000100	175272	START1	MOV	#100, DMKS
475	003364	013700	000734			MOV	UNP, RO
476	003370	005160	001012			COM	UN1(RO)
477	003374	001407				BEQ	STAR1B
478	003376	005160	001012			COM	UN1(RO)
479	003402	016037	001012	000616	STAR1A	MOV	UN1(RO), UDES
480	003410	000137	003542			JMP	START4
481	003414	005237	000720		STAR1B	INC	BLCNTR
482	003420	005737	021632			TST	ASEOF
483	003424	001414				BEQ	STAR1C
484	003426	023737	000720	021630		CMR	BLCNTR, ABLCNT
485	003434	001010				BNE	STAR1C
486	003436	005160	001012			COM	UN1(RO)
487	003442	012737	000001	000720		MOV	#1, BLCNTR
488	003450	005037	000734			CLR	UNP
489	003454	000207				RTS	PC
490	003456	005037	000734		STAR1C	CLR	UNP
491	003462	005160	001012			COM	UN1(RO)
492	003466	005000				CLR	RO
493	003470	016037	001012	000616		MOV	UN1(RO), UDES
494	003476	032777	000200	175146		BIT	#200, DMWR
495	003504	001402				BEQ	START2
496	003506	004737	010644			JSR	PC, CCNTR
497	003512	032777	000400	175132	START2	BIT	#400, DMWR
498	003520	001402				BEQ	START3
499	003522	004737	013236			JSR	PC, DATR
500	003526	032777	000100	175116	START3	BIT	#100, DMWR
501	003534	001402				BEQ	START4
502	003536	004737	010676			JSR	PC, RCNTR

, RESET PATTERN
 , RESET PARITY
 , PRESET BLOCK COUNTER
 , POWER CLEAR CONTROLLER
 , CHECK FOR SORTSWR
 , GO GET PARAMETERS FROM TTY
 , GO RESET BASE
 , POINT TO FIRST ENTRY
 , TEST FOR SOFTSWR
 , SEE IF CHAIN MODE
 , IF SO: BR
 , ASK FOR CONTROL SETTINGS
 , SEE IF LAST ENTRY
 , IF SO: BR
 , CLEAR EOT/DROPPED FLAG
 , POINT TO NEXT UNIT ENTRY
 , CONTINUE CLEARING
 , RESTORE EOT CNTR
 , SET TTY INTERRUPT ENABLE
 , RO = UNIT TABLE POINTER
 ; IF LAST UNIT IN STRING BR
 , LOAD NEXT UNIT DESCRIPTION
 , BUMP BLOCK COUNTER
 , SEE IF AUTO SEQ
 ; IF NOT BR
 , SEE IF DONE SEQ
 ; IF NOT BR
 , RESET UNIT TABLE TERMINATOR
 , RESET BLOCK COUNTER
 , RESET UNIT POINTER
 , RETURN TO AUTO SEQ
 , LOAD FIRST UNIT DESCRIPTION
 , SEE IF RANDOM RECORD SIZE
 ; IF NOT BR
 , GO GENERATE RANDOM CHAR COUNT
 , SEE IF RANDOM DATA
 ; IF NOT BR
 , GO GENERATE RANDOM DATA
 , SEE IF RANDOM RECORD COUNT
 ; IF NOT BR
 , GO GENERATE RANDOM RECORD COUNT

503	003542	032760	100000	001012	START4	BIT	#100000, UN1(RO)	, SEE IF UNIT REACHED EOT OR DROPPED
504	003550	001404				BEQ	STAR40	, IF NOT BR
505	003552	062737	000002	000734		ADD	#2, UNP	, POINT TO NEXT UNIT
506	003560	000676				BR	START1	
507	003562	013777	000616	175012	STAR40	MOV	UDES, @MTC	, SET UNIT NUMBER
508	003570	004737	021234			JSR	PC, STDLY	, GO AWAIT ASSURED STATUS
509	003574	032777	000001	174776		BIT	#1, @MTC	, SEE IF TUR
510	003602	001030				BNE	STAR46	, IF SO. BR
511	003604	032777	000002	174766		BIT	#2, @MTC	, SEE IF REWINDING
512	003612	001414				BEQ	STAR45	, IF NOT BR
513	003614	004737	017506			JSR	PC, PAPRT	, PRINT HEADER
514	003620	012704	025503			MOV	#MSG89, R4	
515	003624	004737	020536			JSR	PC, TTOUT	, PRINT REWIND MSG
516	003630	032777	000001	174742	STAR44	BIT	#1, @MTC	
517	003636	001774				BEQ	STAR44	, AWAIT REWIND DONE
518	003640	000137	003664			JMP	STAR46	
519	003644	004737	017506		STAR45	JSR	PC, PAPRT	, PRINT HEADER
520	003650	012704	023707			MOV	#MSG49, R4	
521	003654	004737	020536			JSR	PC, TTOUT	, PRINT NOT AVAIL
522	003660	000137	020050			JMP	DRPDRV	, GO DROP DRIVE
523	003664	005037	001010		STAR46	CLR	STCDFL	, CLEAR 7 TRK CORE DUMP FLAG
524	003670	032777	000020	174702		BIT	#20, @MTC	, SEE IF 7 TRK
525	003676	001411				BEQ	1\$, IF NOT: BR
526	003700	013704	000616			MOV	UDES, R4	, GET UNIT DESCRIPTION
527	003704	042704	117777			BIC	#117777, R4	, MASK DENSITY
528	003710	022704	060000			CMP	#60000, R4	, SEE IF CORE DUMP
529	003714	001002				BNE	1\$, IF NOT. BR
530	003716	005237	001010			INC	STCDFL	, ELSE SET FLAG
531	003722	004737	012370		1\$	JSR	PC, DSUP	, GO SET UP WRITE DATA
532	003726	004737	004720			JSR	PC, RWND	, REWIND
533	003732	004737	005254			JSR	PC, WRITE	, WRITE
534	003736	013737	000636	000730		MOV	TSTAL, STAL	, SET TURN AROUND DELAY
535	003744	004737	010634			JSR	PC, STALL	, DELAY
536	003750	004737	006626			JSR	PC, RSEQ	, GO TO READ SEQUENCER
537	003754	013737	000636	000730		MOV	TSTAL, STAL	, SET TURN AROUND DELAY
538	003762	004737	010634			JSR	PC, STALL	, DELAY
539	003766	032777	000020	174656		BIT	#20, @SWR	, SEE IF SHOULD PRINT DROPS AND PICK
540	003774	001410				BEQ	START5	, IF NOT. BR
541	003776	012700	000001			MOV	#1, RO	, SET RECORD COUNTER TO 1
542	004002	005237	000772			INC	BTSTF	, SET FOR STAT PRINT ONLY
543	004006	004737	015370			JSR	PC, PRSTAT	, PRINT STATISTICS
544	004012	005037	000772			CLR	BTSTF	, CLEAR FLAG
545	004016	017700	174630		START5	MOV	@SWR, RO	, LOAD SWR
546	004022	042700	177762			BIC	#177762, RO	, MASK READ/WRITE SWITCHES
547	004026	022700	000015			CMP	#15, RO	, SEE IF HAVE READ OR WRITE
548	004032	001424				BEQ	START8	, IF NOT BR
549	004034	032777	000001	174536	START6	BIT	#1, @MTC	, SEE IF HAVE UNIT READY
550	004042	001013				BNE	START7	, IF SO BR
551	004044	005337	000730			DEC	STAL	
552	004050	001371				BNE	START6	, DELAY FOR TUR
553	004052	004737	017506			JSR	PC, PAPRT	, PRINT HEADER
554	004056	012704	023707			MOV	#MSG49, R4	
555	004062	004737	020536			JSR	PC, TTOUT	, PRINT NOT AVAIL
556	004066	000137	020050			JMP	DRPDRV	, GO DROP DRIVE
557	004072	062737	000002	000734	START7	ADD	#2, UNP	, POINT TO NEXT UNIT
558	004100	005077	174476			CLR	@MTC	

```
559 004104 000137 003356      START8 JMP      START1      ;CONTINUE
560
561                                ,RANDOM BASE RESET*****
562
563 004110 012737 153624 000672 RANSET MOV      #153624,RANBAS ;RESET BASE
564 004116 012737 032561 000676      MOV      #32561,RANSAV ;RESET BUFFER
565 004124 013737 000642 000620      MOV      RCSAV,RCNT   ;RESET RECORD COUNT
566 004132 013737 000644 000622      MOV      CCSAV,CARCNT ;RESET CHAR COUNT
567 004140 000207                                RTS      PC
```

```

568 ,*****
569 ,REWIND FROM EOT:
570 ,
571 ,WHEN ANY TRANSPORT BEING TESTED REACHES END OF TAPE
572 ,DURING A READ OR WRITE OPERATION, IT WILL BE REWOUND
573 ,AND FLAGGED AS UNAVAILABLE UNTIL ALL AVAILABLE UNITS
574 ,HAVE REACHED EOT AT WHICH TIME ALL TESTING WILL BE RESUMED
575 ,AT A BLOCK COUNT OF ONE (1). A MESSAGE WILL BE
576 ,PRINTED ON THE SUPERVISORS CONSOLE AS EACH UNIT REACHES
577 ,EOT AND IS REWOUND.
578 ,*****
579
580 004142 013777 000616 174432 REOT MOV UDES,@MTC ,LOAD COMMAND REGISTEP
581 004150 032777 000010 174422 REOT1 BIT #10,@MTC
582 004156 001374 BNE REOT1 ,AWAIT SETTLE DOWN RESET
583 004160 052777 000017 174414 BIS #17,@MTC ,START REWIND
584 004166 004737 017506 JSR PC,PAPRT ,PRINT HEADER
585 004172 032737 000004 000746 BIT #4,BTFLG ,ERROR DURING RETRY?
586 004200 001405 BEQ 1$ ,IF NOT BR
587 004202 012704 025474 MOV #MSG88,R4
588 004206 004737 020536 JSR PC,TTOUT ,PRINT RETRY
589 004212 000404 BR 2$
590 004214 032737 000002 000746 1$ BIT #2,BTFLG ,BACKSPACE ERROR
591 004222 001405 BEQ REOT1C ,IF NOT BR
592 004224 012704 024312 2$ MOV #MSG61,R4 ,POINT TO BACKSPACE ERROR MESSG
593 004230 005037 000746 CLR BTFLG ,CLEAR BAD TAPE FLAG
594 004234 000437 BR REOT1B
595 004236 005737 000746 REOT1C TST BTFLG ,TEST BAD TAPE FLAG
596 004242 001405 BEQ REOT1D ,IF NOT BR
597 004244 012704 024122 MOV #MSG59,R4 ,SET UP BAD TAPE MESSAGE
598 004250 005037 000746 CLR BTFLG ,CLEAR BAD TAPE FLAG
599 004254 000427 BR REOT1B
600 004256 005737 021632 REOT1D TST ASEQF ,IS IT AUTO SEQ?
601 004262 001406 BEQ REOT1A ,IF NOT BR
602 004264 005737 000624 TST PATRN ,IS IT RANDOM DATA?
603 004270 100403 BMI REOT1A ,IF SO BR
604 004272 012704 025340 MOV #MSG87,R4 ,PRINT EARLY ASEQ EOT MESSG
605 004276 000416 BR REOT1B
606 004300 012704 022600 REOT1A MOV #MSG20,R4
607 004304 004737 020536 JSR PC,TTOUT ,PRINT EOT MESSAGE
608 004310 013704 000734 MOV UNP,R4
609 004314 005264 001234 INC EOTCT1(R4) ,BUMP EOT COUNTER
610 004320 016403 001234 MOV EOTCT1(R4),R3
611 004324 004737 020724 JSR PC,OCTP ,PRINT EOT COUNT
612 004330 012704 022614 MOV #MSG20A,R4
613 004334 004737 020536 REOT1B JSR PC,TTOUT ,PRINT REWIND MSG
614 004340 004737 015400 JSR PC,PRSTA2 ,PRINT STATS WITHOUT HEADER
615 004344 032777 000200 174230 REOT2 BIT #200,@MTC
616 004352 001774 BEQ REOT2 ,AWAIT CUR
617 004354 105337 004716 DECB REOTC ,SEE IF LAST UNIT TO REACH EOT
618 004360 001410 BEQ REOT3 ,IF SO BR
619 004362 013700 000734 MOV UNP,R0
620 004366 052760 100000 001012 BIS #100000,UN1(R0) ,SET EOT FLAG
621 004374 005726 TST (SP)+
622 004376 000137 004072 JMP START7 ,GO TO NEXT UNIT
623 004402 000337 004716 REOT3 SWAB REOTC
    
```

624	004406	013700	004716			MOV	REOTC,RO	
625	004412	000337	004716			SWAB	REOTC	
626	004416	110037	004716			MOVB	RO,REOTC	,RESTORE EOT UNIT COUNTER
627	004422	005037	000734			CLR	UNP	
628	004426	013700	000734			MOV	UNP,RO	,POINT TO FIRST UNIT
629	004432	016037	001012	000616	REOT4	MOV	UN1(RO),UDES	,LOAD UNIT DESCRIPTION
630	004440	032737	000200	000616		BIT	#200,UDES	,SEE IF UNIT IS DROPPED
631	004446	001034				BNE	REOT6A	,IF SO BR
632	004450	013777	000616	174124		MOV	UDES,@MTC	,LOAD COMMAND REGISTER
633	004456	032777	000002	174114	REOT5	BIT	#2,@MTC	
634	004464	001374				BNE	REOT5	,AWAIT RWS RESET
635	004466	032777	000040	174104		BIT	#40,@MTC	,SEE IF HAVE BOT
636	004474	001012				BNE	REOT6	,IF SO BR
637	004476	012700	000001			MOV	#1,RO	
638	004502	004737	017506			JSR	PC,PAPRT	,PRINT HEADER
639	004506	012704	023661			MOV	#MSG48,R4	
640	004512	004737	020536			JSR	PC,TTOUT	,PRINT BOT ERROR
641	004516	000137	020050			JMP	DRPDRV	,GO DROP DRIVE
642	004522	032777	000010	174050	REOT6	BIT	#10,@MTC	,SEE IF SWDN IS RESET
643	004530	001374				BNE	REOT6	,IF NOT AWAIT SWDN RESET
644	004532	042760	100200	001012		BIC	#100200,UN1(RO)	,CLEAR EOT/DROPPED FLAG
645	004540	062737	000002	000734	REOT6A	ADD	#2,UNP	
646	004546	013700	000734			MOV	UNP,RO	,POINT TO NEXT UNIT
647	004552	005160	001012			COM	UN1(RO)	,SEE IF LAST UNIT
648	004556	001404				BEQ	REOT7	,IF SO BP
649	004560	005160	001012			COM	UN1(RO)	
650	004564	000137	004432			JMP	REOT4	,DO NEXT UNIT
651	004570	005160	001012		REOT7	COM	UN1(RO)	
652	004574	012737	000001	000720		MOV	#1,BLCNTR	,SET TO BLOCK COUNT 1
653	004602	005037	000734			CLP	UNP	
654	004606	005000				CLR	RO	,SET TO RESTART WITH FIRST UNIT
655	004610	005726				TST	(SP)+	,RESET STACK
656	004612	005737	021632			TST	ASEQF	,SEE IF AUTO SEQ
657	004616	001401				BEQ	REOT8	,IF NOT BR
658	004620	000207				RTS	PC	,RETURN
659	004622	012704	023501		REOT8	MOV	#MSG39,R4	
660	004626	004737	020536			JSR	PC,TTOUT	,PRINT END OF PASS
661	004632	005737	000630			TST	SPFLG	,SEE IF SINGLE PASS
662	004636	001412				BEQ	REOTX	,IF NOT BR
663	004640	013704	000042		REOT9	MOV	@#42,R4	
664	004644	001405				BEQ	HERE	IF NOT CHAIN MODE BP
665	004646	000005				RESET		
666	004650	004714			SENDAD	JSR	PC,(R4)	
667	004652	000240				NOP		
668	004654	000240				NOP		
669	004656	000240				NOP		
670	004660	000240			HERE	NOP		
671	004662	000000			REOT10	HALT		
672	004664	012706	000500		REOTX	MOV	#500,SP	,RESET STACK
673	004670	004737	004110			JSR	PC,RANSET	,GO RESET RANDOM BASE
674	004674	012737	177777	012750		MOV	#-1,PATS	,PRESET PATTERN
675	004702	005037	013304			CLR	RDFL	,CLEAR RANDOM DATA FLAG
676	004706	005037	001006			CLR	DUCTP	,CLEAR DROPPED UNITER COUNTER
677	004712	000137	003264			JMP	STAUTO	,RESTART AT BLOCK NUMBER ONE
678	004716	000401			REOTC	401		,EOT UNIT COUNTER(DEFAULT TO ONE UNIT)

```

679 ,*****
680 ,REWIND ALL AVAIL TAPES
681 ,
682 ,THIS ROUTINE, ENTERED VIA CONSOLE SWITCH NINE (9),
683 ,WILL REWIND ALL AVAILABLE TAPES TO BOT NO MATTER
684 ,WHERE THEY ARE CURRENTLY POSITIONED AND RESUME TESTING
685 ,AT A BLOCK COUNT OF ONE (1)
686 ,*****
687
688 004720 032777 001000 173724 RWND BIT #1000, @SWR ,SEE IF SHOULD REWIND
689 004726 001001 BNE RWND4 ,IF SO BR
690 004730 000207 RTS PC ,ELSE EXIT
691 004732 005037 000734 RWND4 CLR UNP ,CLEAR POINTER
692 004736 000337 004716 SWAB REOTC
693 004742 013700 004716 MOV REOTC, RO
694 004746 000337 004716 SWAB REOTC
695 004752 110037 004716 MOV8 RO, REOTC ,RESTORE EOT UNIT COUNTER
696 004756 013700 000734 RWND0 MOV UNP, RO ,POINT TO UNIT ENTRY
697 004762 005160 001012 COM UN1(RO) ,SEE IF LAST ENTRY
698 004766 001424 BEQ RWND2 ,IF SO BR
699 004770 005160 001012 COM UN1(RO)
700 004774 016037 001012 000616 MOV UN1(RO), UDES ,SET UNIT DESCRIPTION
701 005002 013777 000616 173572 MOV UDES, @MTC ,LOAD COMMAND REGISTER
702 005010 052777 000017 173564 BIS #17, @MTC ,START REWIND
703 005016 032777 000200 173556 RWND1 BIT #200, @MTC
704 005024 001774 BEQ RWND1 ,AWAIT CUP
705 005026 062737 000002 000734 ADD #2, UNP ,BUMP POINTER
706 005034 000137 004756 JMP RWND0 ,DO NEXT UNIT
707 005040 005160 001012 RWND2 COM UN1(RO)
708 005044 005037 000734 CLR UNP ,CLEAR POINTER
709 005050 013700 000734 RWND3 MOV UNP, RO ,POINT TO UNIT ENTRY
710 005054 005160 001012 COM UN1(RO) ,SEE IF LAST ENTRY
711 005060 001452 BEQ RWNDX ,IF SO BR
712 005062 005160 001012 COM UN1(RO)
713 005066 016037 001012 000616 MOV UN1(RO), UDES ,SET UNIT DESCRIPTION
714 005074 032737 000200 000616 BIT #200, UDES ,SEE IF UNIT IS DROPPED
715 005102 001403 BEQ 15 ,IF NOT BR
716 005104 005337 004716 DEC REOTC ,ELSE DECREMENT EOT UNIT CNTR
717 005110 000417 BR RWND5
718 005112 013777 000616 173462 15 MOV UDES, @MTC LOAD COMMAND REGISTER
719 005120 032777 000002 173452 RWND4 BIT #2, @MTC
720 005126 001374 BNE RWND4 ,AWAIT RWS RESET
721 005130 032777 000040 173442 BIT #40, @MTC ,SEE IF HAVE BOT
722 005136 001411 BEQ RWND6 ,IF NOT BR
723 005140 032777 000010 173432 15 BIT #10, @MTC ,SEE IF SDWN SET
724 005146 001374 BNE 15 ,IF SO AWAIT RESET
725 005150 062737 000002 000734 RWND5 ADD #2, UNP BUMP POINTER
726 005156 000137 005050 JMP RWND3 ,DO NEXT UNIT
727 005162 012700 000001 RWND6 MOV #1, RO
728 005166 004737 017506 JSR PC, PAPRT ,PRINT HEADER
729 005172 012704 023661 MOV #MSG48, R4
730 005176 004737 020536 JSR PC, TOUT ,PRINT NO BOT
731 005202 000137 020050 JMP DRPDRV GO DROP DRIVE
732 005206 005160 001012 RWNDX COM UN1(RO)
733 005212 005000 CLR RO
734 005214 010037 000734 15 MOV RO, UNP
    
```

735	005220	016037	001012	000616		MOV	UN1(RO), UDES	
736	005226	032737	100200	000616		BIT	#100200, UDES	, SEE IF UNIT DROPPED
737	005234	001403				BEQ	2\$, IF NOT BR
738	005236	062700	000002			ADD	#2, RO	
739	005242	000764				BR	1\$	
740	005244	012737	000001	000720	2\$	MOV	#1, BLCNTR	
741	005252	000207				RTS	PC	

742
 743
 744
 745
 746
 747
 748
 749
 750
 751
 752
 753
 754
 755
 756
 757
 758
 759
 760
 761
 762
 763

 ,WRITE ROUTINE
 ,
 ,THIS ROUTINE IS USED TO WRITE ONTO TAPE THE BLOCK
 ,OF DATA DESCRIBED BY THE OPERATOR AND SET UP
 ,IN THE SEQUENCE FORMATTER THE TAPE UNIT TO BE USED
 ,HAS BEEN ASSIGNED BY THE SEQUENCE FORMATTER AND
 ,ITS PARAMETERS SET IN A UNIT DESCRIPTION WORD
 ,AS EACH RECORD OF THE BLOCK IS WRITTEN, IT IS CHECKED
 ,FOR STATUS ERRORS, WORD COUNT ZERO, AND CORRECT CURRENT
 ,MEMORY ADDRESS. IF THE WRITE OPERATION CAUSES THE SELECTED
 ,UNIT TO REACH END OF TAPE (EOT), THE UNIT IS REWOUND
 ,AND FLAGGED AS UNAVAILABLE FOR TESTING UNTIL ALL AVAILABLE
 ,UNITS HAVE REACHED EOT AT WHICH TIME ALL UNITS WILL
 ,BE RESTARTED AT A BLOCK COUNT OF ONE (1)
 ,ERROR CHECKING MAY BE DISALLOWED VIA CONSOLE SWITCH
 ,TWELVE (12)
 ,WRITING TO TAPE MAY BE DISALLOWED VIA CONSOLE SWITCH
 ,ZERO (0)

764	005254	032777	000001	173370	WRITE	BIT	#1, @SWR	,SEE IF SHOULD WRITE
765	005262	001076				BNE	W3A0	,IF NOT BR
766	005264	012737	022460	000716		MOV	#MSG5, EMADDR	SET ERROR MSG ADDRESS
767	005272	005077	173304			CLR	@MTC	
768	005276	005077	173276			CLR	@MTS	
769	005302	005037	000760			CLR	EOTREC	,CLEAR EOT FLAG
770	005306	013700	000620			MOV	RCNT, R0	,R0=RECORD COUNT
771	005312	013777	000622	173264	W0	MOV	CARCNT, @MWC	,LOAD CHAR COUNT
772	005320	012777	026204	173260		MOV	#WDATA, @MDA	,SET DATA ADDR
773	005326	005737	000764			TST	ERTFL	,SEE IF SHOULD ERASE
774	005332	001406				BEQ	W0A	,IF NOT BR
775	005334	112777	000014	173240		MOVB	#14, @MTC	,SET OP-CODE WRITE W/EXTENDED IRG
776	005342	005037	000764			CLR	ERTFL	,CLEAR ERASE FLAG
777	005346	000403				BR	W0B	
778	005350	112777	000004	173224	W0A	MOVB	#4, @MTC	,SET WRITE OP COMMAND
779	005356	012737	005370	000724	W0B	MOV	#W1A, RTRN	,SET RETURN ADDRESS
780	005364	000137	017006			JMP	TAPG	,GO EXECUTE COMMAND
781	005370	005737	017402		W1A	TST	WEOTF	,SEE IF EOT FOUND
782	005374	001413				BEQ	W1	,IF NOT BR
783	005376	005037	017402			CLR	WEOTF	,CLEAR WRITE EOT FLAG
784	005402	013701	000620			MOV	RCNT, P1	,BUILD SHORTENED RECORD COUNT
785	005406	160001				SUB	R0, R1	
786	005410	005201				INC	R1	
787	005412	010137	000760			MOV	R1, EOTREC	
788	005416	052737	100000	000760		BIS	#100000, EOTREC	,SET EOT FLAG
789	005424	032777	010000	173220	W1	BIT	#10000, @SWR	,SEE IF SHOULD CHECK ERPOPS
790	005432	001002				BNE	W3	,IF NOT BR
791	005434	004737	016036			JSR	PC, ERCHK	,GO CHECK ERRORS
792	005440	013737	000634	000730	W3	MOV	WSTAL, STAL	,SET DELAY
793	005446	004737	010634			JSR	PC, STALL	,DELAY
794	005452	005737	000754			TST	RTYFL	,SEE IF RETRY
795	005456	001401				BEQ	W3A	,IF NOT BR
796	005460	000207			W3A0	RTS	PC	,ELSE RETURN TO RETRY ROUTINE
797	005462	005737	000742		W3A	TST	SERFL	,SEE IF WRITE ERROR

798	005466	001453		BEQ	W3D	. IF NOT BR
799	005470	013704	000734	MOV	UNP,R4	. BUMP WRITE ERROR
800	005474	005264	001074	INC	WTER1(R4)	
801	005500	005037	000742	CLR	SERFL	. CLEAR STATUS ERROR FLAG
802	005504	032777	000002	BIT	#2,@SWR	. SEE IF RETRY -- SW1
803	005512	001041		BNE	W3D	. IF NOT BR
804	005514	042737	072521	BIC	#072521,ERSAV	. MASK UNRECOVERABLE ERROR
805	005522	005737	000740	TST	ERSAV	. SEE IF RETRYABLE ERROR
806	005526	001411		BEQ	W3B	. IF SO BR
807	005530	012704	023741	MOV	#MSG52,R4	
808	005534	004737	020536	JSR	PC,TTOUT	. PRINT NON-RETRYABLE ERROR FLAG
809	005540	012704	022460	MOV	#MSG5,R4	
810	005544	004737	020536	JSR	PC,TTOUT	. PRINT WRITE ERROR TAG
811	005550	000422		BR	W3D	
812	005552	013704	000734	MOV	UNP,R4	
813	005556	005264	001154	INC	RTY1(R4)	. BUMP RETRY CNTR
814	005562	032777	002000	BIT	#2000,@SWR	. SEE IF PRINT ERRORS
815	005570	001004		BNE	W3C	. IF NOT BR
816	005572	012704	023761	MOV	#MSG53,R4	
817	005576	004737	020536	JSR	PC,TTOUT	. PRINT ORIGINAL ERROR TAG
818	005602	005037	000752	CLR	RTCNT	. CLEAR RETRY NUMBER
819	005606	005037	000750	CLR	RPCNT	. CLEAR REPEAT COUNTER
820	005612	004737	006142	JSR	PC,WRTY	. GO RETRY WRITE ERROR
821	005616	005037	000754	CLR	RTYFL	. CLEAR RETRY FLAG
822	005622	005737	000760	TST	EOTPEC	. WAS EOT REACHED?
823	005626	100403		BMI	WEX	. IF SO BR
824	005630	005300		DEC	RO	. SEE IF DONE ALL
825	005632	001227		BNE	W0	. IF NOT BR
826	005634	005200		INC	RO	. ADJUST FOR REC NO IN HEADER
827	005636	005737	000646	TST	TMEX	. SEE IF TM
828	005642	001402		BEQ	WEX1	. IF NOT BR
829	005644	004737	005712	JSP	PC,WTM	. WRITE TM
830	005650	005037	000754	CLR	RTYFL	. CLEAR RETRY FLAG
831	005654	005037	000756	CLR	TMFLG	. CLEAR TM FLAG
832	005660	005737	000760	TST	EOTREC	. TEST FOR EOT
833	005664	100401		BMI	W4	. IF SO BR
834	005666	000207		RTS	PC	. EXIT
835	005670	017704	172756	MOV	@SWR,R4	
836	005674	042704	177767	BIC	#177767,R4	. CHECK IF WRITE ONLY
837	005700	022704	000010	CMP	#10,R4	
838	005704	001370		BNE	WEX2	. IF NOT BR
839	005706	000137	004142	JMP	PECT	. GO REWIND ALL AMAIL TAPES

840

```

841      ,*****
842      ,WRITE TAPE MARK
843
844      ,THIS ROUTINE, ENABLED THRU TELETYPE RESPONSE
845      ,AT PROGRAM START-UP, WILL WRITE A TAPE MARK
846      ,FOLLOWING THE WRITING OF EACH BLOCK OF DATA
847      ,THIS OPTION INCREASES THE BLOCK SIZE BY ONE RECORD,
848      ,A BLOCK OF 100 RECORDS WILL HAVE A TAPE MARK
849      ,WRITTEN AS RECORD 101
850      ,*****
851
852 005712 012737 024463 000716 WTM  MOV  #MSG62,EMADDR  ,POINT TO TM ERROR MSG
853 005720 005300          DEC  RO
854 005722 005237 000756          INC  TMFLG      ,SET TM FLAG
855 005726 005077 172652          CLR  @MWC      ,CLEAR BYTE COUNTER
856 005732 012777 026204 172646  MOV  #WDATA,@MDA
857 005740 012777 000006 172634  MOV  #6,@MTC   ,SET TM OP CODE
858 005746 012737 005760 000724  MOV  #WTMO,PTRN ,SAVE RETURN ADDRESS
859 005754 000137 017006          JMP  TAPG      ,EXECUTE TM COMMAND
860 005760 032777 010000 172664  WTM0  BIT  #10000,@SWR ,SEE IF SHOULD CHECK EPRORS
861 005766 001062          BNE  WTM4      ,IF NOT: BR
862 005770 004737 016036          JSR  PC,ERCHK  ,CHECK FOR ERRORS
863 005774 005737 000742          TST  SERFL     ,SEE IF STATUS ERROR
864 006000 001455          BEQ  WTM4      ,IF NOT BR
865 006002 005737 000754          TST  RTYFL     ,SEE IF RETRY
866 006006 001401          BEQ  WTM1      ,IF NOT BR
867 006010 000207          RTS  PC        ,ELSE RETURN TO RETPY ROUTINE
868 006012 013704 000734          WTM1  MOV  UNP,R4
869 006016 005264 001074          INC  WTER1(R4) ,BUMP WRITE ERROR
870 006022 032777 000002 172622  BIT  #2,@SWR   ,SEE IF SHOULD RETRY
871 006030 001041          BNE  WTM4      ,IF NOT BR
872 006032 042737 147377 000740  BIC  #147377,ERSAV ,MASK UNRECOVERABLE ERROR
873 006040 005737 000740          TST  ERSAV     ,SEE IF RECOVERABLE
874 006044 001411          BEQ  WTM2      ,IF SO BR
875 006046 012704 023741          MOV  #MSG52,R4
876 006052 004737 020536          JSR  PC,TTOUT  ,PRINT UNRETRYABLE TAG
877 006056 012704 024463          MOV  #MSG62,R4
878 006062 004737 020536          JSR  PC,TTOUT  ,PRINT TM ERROR TAG
879 006066 000207          RTS  PC
880 006070 005037 000750          WTM2  CLR  RPCNT    ,CLEAR REPEAT CNTP
881 006074 013704 000734          MOV  UNP,R4
882 006100 005264 001154          INC  RTY1(R4)  ,BUMP RETPY CNTR
883 006104 005037 000752          CLR  RTCNT     ,CLEAR RETRY LOOP CNTP
884 006110 032777 002000 172534  BIT  #2000,@SWP ,SEE IF PRINT EPPORS
885 006116 001004          BNE  WTM3      ,IF NOT BR
886 006120 012704 023761          MOV  #MSG53,P4
887 006124 004737 020536          JSR  PC,TTOUT  ,PRINT ORIGINAL ERROR TAG
888 006130 004737 006142          WTM3  JSR  PC,WRTY   ,GO DO RETRY
889 006134 005037 000756          WTM4  CLR  TMFLG   ,CLEAR TM FLAG
890 006140 000207          RTS  PC        ,EXT
SP1
  
```

```

892 ,*****
893 WRITE ERROR RETRY
894 ,*****
895
896
897 006142 012737 000001 000754 WRTY MOV #1,RTYFL ,SET RETRY FLAG
898 006150 004737 006504 WRTYO JSR PC,WRTSB ,GO SPACE BACK FOR REPEAT
899 006154 005737 000756 TST TMFLG ,SEE IF A TM
900 006160 001003 BNE WRTYTM ,IF SO. BR
901 006162 004737 005312 JSR PC,W0 ,REWRITE RECORD
902 006166 000402 BR WRTYR ,CONTINUE
903 006170 004737 005712 WRTYTM JSR PC,WTM ,GO WRITE TM AGAIN
904 006174 005737 000742 WRTYR TST SERFL ,REWRITE GOOD?
905 006200 001027 BNE WRTY2 ,IF NOT BR
906 006202 005237 000750 INC RPCNT ,BUMP REPEAT COUNTER
907 006206 022737 000004 000750 CMP #4,RPCNT ,SEE IF FOUR GOOD REPEATS
908 006214 001355 BNE WRTYO ,IF NOT DO ANOTHER
909 006216 032777 002000 172426 B T #2000,@SWR ,SEE IF PRINT
910 006224 001014 BNE WRTY1 ,IF NOT BR
911 006226 012704 024007 MOV #MSG54,R4
912 006232 004737 020536 JSR PC,TTOUT ,PRINT RECOVERED MESSAGE
913 006236 012704 024022 MOV #MSG55,R4
914 006242 004737 020536 JSR PC,TTOUT ,PRINT RETRY TAG
915 006246 013703 000752 MOV RTCNT,R3
916 006252 004737 020724 JSR PC,OCTP ,PRINT RETRY NUMBER
917 006256 000207 WRTY1 RTS PC ,RESUME TESTING
918 006260 032777 002000 172364 WRTY2 BIT #2000,@SWP ,SEE IF PRINT
919 006266 001024 BNE WRTY3 ,IF NOT BR
920 006270 012704 024033 MOV #MSG56,R4
921 006274 004737 020536 JSR PC,TTOUT ,PRINT BAD TAPE SUSPECT
922 006300 012704 024022 MOV #MSG55,R4
923 006304 004737 020536 JSP PC,TTOUT ,PRINT RETRY TAG
924 006310 013703 000752 MOV RTCNT,R3
925 006314 004737 020724 JSR PC,OCTP ,PRINT RETRY NUMBER
926 006320 012704 024055 MOV #MSG57,R4
927 006324 004737 020536 JSR PC,TTOUT ,PRINT REPEAT TAG
928 006330 013703 000750 MOV RPCNT,R3
929 006334 004737 020724 JSR PC,OCTP ,PRINT REPEAT NUMBER
930 006340 005737 000752 WPTY3 TST RTCNT ,SEE IF FIRST RETRY
931 006344 001004 BNE WRTY3A ,IF NOT BR
932 006346 013704 000734 MOV UNP,R4
933 006352 005364 001074 WRTY3A DEC WTER1(R4) ,DECREMENT WRITE ERROR CNTP
934 006356 013704 000734 MOV UNP,R4 ,GET UNIT NUMBER
935 006362 016437 002714 000762 MOV BTADDR(R4),BTPT ,GET ADDRESS OF UNIT BAD TAPE CNTP
936 006370 017704 172366 MOV @BTPT,R4 ,GET COUNTER
937 006374 005724 TST (R4)+ ,SET POINTER OFFSET
938 006376 010477 172360 MOV R4,@BTPT
939 006402 013703 000762 MOV BTPT,R3
940 006406 060304 ADD R3,R4 ,SET ABSOLUTE POINTER
941 006410 013714 000720 MOV BLCNTR,(R4) ,SET BLOCK NUMBER
942 006414 062704 000040 ADD #40,R4 ,ADD RCNT OFFSET
943 006420 013714 000620 MOV RCNT,(R4)
944 006424 160014 SUB R0,(R4) ,SET RECORD NUMBER
945 006426 005214 INC (R4) ,CORRECT RECORD NUMBER
946 006430 022777 000040 172324 CMP #40,@BTPT ,SEE IF TOO MANY BAD SPOTS
947 006436 001002 BNE WRTY4 ,IF NOT BR
    
```

```

948 006440 000137 006570          JMP      BTOV      , ELSE GO TO BAD TAPE OVERFLOW
949 006444 005237 000752          INC      RTCNT     , BUMP RETRY COUNTER
950 006450 022737 000004 000752      CMP      #4, RTCNT , SEE IF DONE 4 RETRIES
951 006456 001410          BEQ      WRTY5     , IF SO BR
952 006460 013704 000734          MOV      UNP, R4
953 006464 005264 001154          INC      RTY1(R4) , BUMP RETRY COUNTER
954 006470 005237 000764          INC      ERTFL     , SET ERASE FLAG
955 006474 000137 006150          JMP      WRTY0     , DO NEXT RETRY
956 006500 000137 006614          JMP      BTUR      , ELSE GO TO BAD TAPE UNRECOVERABLE
957
958
959
          , WR TE RETRY BACKSPACE-ERASE SUBROUTINE
960 006504 005037 000742          WRTSB   CLR      SERFL     ; CLEAR FLAG
961 006510 012777 177777 172066      MOV      #-1, @MWC , SET FOR 1 RECORD
962 006516 012737 024557 000716      MOV      #MSG69, EMADDR
963 006524 004737 010466          JSR      PC, SPBK  ; DO SPACE BACK
964 006530 012737 022460 000716      MOV      #MSG5, EMADDR
965 006536 032737 000002 000746      BIT      #2, BTFLG , SEE IF ERROR ON BACKSPACE
966 006544 001410          BEQ      WRTSBO    , IF NOT BR
967 006546 005037 000754          CLR      RTYFL
968 006552 022626          CMP      (SP)+, (SP)+ , RESET STACK
969 006554 052737 000004 000746      BIS      #4, BTFLG , MARK RETRY ERROR
970 006562 000137 004142          JMP      REOT      ; REWIND AND REMOVE FROM TESTING
971 006566 000207          WRTSBO  RTS      PC      , RETURN
972
973
          , BAD TAPE OVERFLOW SUBROUTINE*****
974
975 006570 013704 000734          BTOV   MOV      UNP, R4
976 006574 005264 001154          INC      RTY1(R4) , BUMP RETRY COUNTER
977 006600 012737 000001 000746      MOV      #1, BTFLG , SET BAD TAPE OVERFLOW FLAG
978 006606 005726          TST      (SP)+    , RESET STACK
979 006610 000137 004142          JMP      REOT      , GO REWIND AND REMOVE FROM TESTING
980
          , BAD TAPE UNRECOVERABLE SUBROUTINE*****
981
982
983 006614 012704 024067          BTUR   MOV      #MSG58, R4
984 006620 004737 020536          JSR      PC, TTOUT , PRINT UNRECOVERABLE BAD SPOT MSG
985 006624 000207          RTS      PC      , RESUME TESTING
    
```

```

986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001 006626 032777 000010 172016 RSEQ BIT #10, @SWR ; SEE IF SHOULD READ FORWARD
1002 006634 001031 BNE RSEX ; IF NOT: BR
1003 006636 032777 000001 172006 BIT #1, @SWR ; SEE IF WRITE
1004 006644 001404 BEQ RSFROA ; IF SO: BR
1005 006646 032777 040000 171776 BIT #40000, @SWR ; SEE IF SHOULD REMAIN IN PLACE
1006 006654 001410 BEQ RSFRO ; IF NOT: BR
1007 006656 004737 010132 RSFROA JSR PC, BKSP ; GO BACKSPACE TO START
1008 006662 032737 000002 000746 BIT #2, BTFLG ; ERROR ON BACKSPACE?
1009 006670 001402 BEQ RSFRO ; IF NOT: BR
1010 006672 000137 004142 JMP REOT ; REWIND AND REMOVE FROM TESTING
1011 006676 012737 000002 000626 RSFRO MOV #2, RDCMD ; LOAD READ FORWARD COMMAND
1012 006704 004737 006722 JSR PC, READ ; GO READ FORWARD
1013 006710 032777 040000 171734 BIT #40000, @SWR ; SEE IF SHOULD READ SAME BLOCK
1014 006716 001357 BNE RSFROA ; IF SO: BR
1015 006720 000207 RSEX PTS ; EXIT
1016
    
```

```

1017 , *****
1018 , READ ROUTINE
1019 ,
1020 , THIS ROUTINE PERFORMS THE READ OPERATION DETERMINED
1021 , BY THE READ SEQUENCE ROUTINE ONE RECORD AT A TIME
1022 , AT THE END OF EACH READ OPERATION THE STATUS REGISTER
1023 , IS SCANNED FOR EITHER END OF TAPE OR BEGINNING OF TAPE
1024 , IF EOT WAS REACHED, CONTROL WILL BE PASSED TO
1025 , THE EOT SUBROUTINE TO REWIND THE UNIT AND FLAG IT
1026 , UNAVAILABLE UNTIL ALL UNITS HAVE REACHED EOT
1027 , IF BOT WAS REACHED AND ERROR IS PRINTED AND THE
1028 , PROGRAM WILL HALT. TESTING MAY BE RESUMED BY PRESSING
1029 , THE CONTINUE SWITCH TWICE.
1030 , CONSOLE SWITCHES ELEVEN (11) AND THIRTEEN (13) DETERMINE WHETHER
1031 , OR NOT TO CHECK FOR STATUS ERRORS (11) OR DATA ERRORS (13),
1032 , CONSOLE SWITCH FIVE (5) IS USED TO CAUSE A CONTINUOUS
1033 , READ AND SPACE (FORWARD OR REVERSE) OF THE CURRENT
1034 , RECORD ON TAPE (YOZZLE).
1035 , *****
1036
1037 006722 013700 000620 READ MOV RCNT, R0 ; LOAD REC CNTR
1038 006726 2737 022465 000716 MOV #MSG6, EMADDR ; SET ERROR MSG ADDRESS
1039 006734 005037 000756 CLR TMFLG ; CLEAR TM FLAG
1040 006740 052777 040000 171644 BIS #40000, @MTRD ; SET TO READ LPC ON READ
1041 006746 005077 171630 RDO CLR @MTC
1042 006752 005077 171622 CLR @MTC
1043 006756 013777 000622 171620 RD1 MOV CARCNT, @MWC ; LOAD CHAR CNTR
1044 006764 012777 032216 171614 RD1A MOV #RDATA, @MDA ; LOAD DATA ADDR
1045 006772 053777 000626 171602 BIS RDCMD, @MTC ; LOAD READ OP COMMAND
1046 007000 012737 007012 000724 MOV #RD2, RTRN ; SET INTERRUPT RETURN ADDRESS
1047 007006 000137 017006 JMP TAPG ; GO EXECUTE TAPE COMMAND
1048 007012 032777 002000 171560 RD2 BIT #2000, @MTC ; SEE IF AT EOT
1049 007020 001405 BEQ RD3 ; IF NOT BR
1050 007022 052737 100000 000760 BIS #100000, EOTREC ; MARK EOT FOUND
1051 007030 000137 007350 JMP RDEX ; GO REWIND
1052 007034 032777 000040 171536 RD3 BIT #40, @MTC ; SEE IF AT LOAD POINT
1053 007042 001411 BEQ RD4 ; IF NOT BR
1054 007044 004737 017506 JSR PC, PAPRT ; PRINT CYCLE NUMBER
1055 007050 012704 022733 MOV #MSG22, R4
1056 007054 004737 020536 JSR PC, TTOUT ; PRINT BOT ERROR
1057 007060 000240 NOP
1058 007062 000137 020050 JMP DRPDRV ; DROP DRIVE
1059 007066 032777 004000 171556 PD4 BIT #4000, @SWR ; SEE IF SHOULD CHECK ERRORS
1060 007074 001037 BNE RD5 ; IF NOT BR
1061 007076 004737 016036 JSR PC, ERCHK ; GO CHECK ERRORS
1062 007102 005737 000742 TST SERFL ; SEE IF STATUS ERROR
1063 007106 001432 BEQ RD5 ; IF NOT BR
1064 007110 013704 000734 MOV UNP, R4
1065 007114 005264 001114 INC RDER1(R4) ; BUMP READ ERROR
1066 007120 032777 000002 171524 BIT #2, @SWR ; SEE IF SHOULD DO READ RETRY
1067 007126 001022 BNE RD5 ; IF NOT BR
1068 007130 017737 171444 000740 MOV @MTC, ERSAV
1069 007136 042737 073525 000740 BIC #073525, ERSAV ; MASK NON-RETRYABLE ERRORS
1070 007144 001411 BEQ RD4A ; IF RETRYABLE BR
1071 007146 012704 023741 MOV #MSG52, R4
1072 007152 004737 020536 JSR PC, TTOUT ; PRINT NON-RETRYABLE MESSG
    
```

1073	007156	012704	022465			MOV	#MSG6, R4	
1074	007162	004737	020536			JSR	PC, TTOUT	, PRINT READ ERROR TAG
1075	007166	000402				BR	RD5	
1076	007170	004737	007376		RD4A	JSR	PC, RRTY	, DO RETRY
1077	007174	032777	020000	171450	RD5	BIT	#20000, @SWR	, SEE IF SHOULD DO DATA CHECK
1078	007202	001007				BNE	RD6	, IF NOT: BR
1079	007204	005737	000756			TST	TMFLG	, IS IT TM?
1080	007210	001004				BNE	RD6	, IF SO: BR
1081	007212	004737	013712			JSR	PC, DCHK	, GO CHECK DATA
1082	007216	005037	000742			CLR	SERFL	, CLEAR STATUS ERROR FLAG
1083	007222	004737	012516		RD6	JSR	PC, DS3	, CLEAR BUFFER
1084	007226	032777	000040	171416		BIT	#40, @SWR	, SEE IF SHOULD YOZZLE
1085	007234	001402				BEQ	RD7	, IF NOT: BR
1086	007236	004737	007600			JSR	PC, YOZ	, ELSE GO YOZZLE
1087	007242	013737	000632	000730	RD7	MOV	RSTAL, STAL	, SET DELAY
1088	007250	004737	010634			JSR	PC, STALL	, STALL
1089	007254	005737	000756			TST	TMFLG	, JUST DONE TM?
1090	007260	001033				BNE	RDEX	, IF SO: BR
1091	007262	005737	000760			TST	EOTREC	, WAS EOT REACHED
1092	007266	100430				BMI	RDEX	, IF SO: BR
1093	007270	005300				DEC	RO	
1094	007272	01225				BNE	RDO	, IF NOT DONE ALL: BR
1095	007274	005200				INC	RO	, ADJUST FOR REC NO IN HEADER
1096	007276	005737	000646		RD10	TST	TMEX	, EXPECT A TAPE MARK?
1097	007302	001422				BEQ	RDEX	, IF NOT: BR
1098	007304	005300				DEC	RO	, ELSE READ TM
1099	007306	012777	177776	171270		MOV	#-2, @MWC	, SET BYTE COUNT
1100	007314	005737	001010			TST	STCOFL	, SEE IF 7 TRK CORE DUMP
1101	007320	001402				BEQ	1\$, IF NOT: BR
1102	007322	005277	171256			INC	@MWC	, SET TO ONE CHAR
1103	007326	005237	000756		1\$	INC	TMFLG	, SET TM FLAG
1104	007332	012737	024567	000716		MOV	#MSG70, EMADDR	
1105	007340	042777	040000	171244		BIC	#40000, @MTRD	, SET TO READ LPC ON READ TM
1106	007346	000606				BR	RD1A	, GO READ
1107	007350	005037	000756		RDEX	C'R	TMFLG	
1108	007354	005737	000760			TST	EOTREC	, WAS EOT REACHED
1109	007360	100005				BPL	RDEXX	, IF NOT: BR
1110	007362	005726				TST	(SP)+	, RESET STACK
1111	007364	005037	000760			CLR	EOTREC	, CLEAR EOT IND.
1112	007370	000137	004142			JMP	REOT	, GO REWIND
1113	007374	000207			RDEXX	RTS	PC	, EXIT
1114								
1115								
1116								
1117								
1118								
1119								
1120								
1121								
1122	007376	005237	000774		RRTY	INC	RRTYFL	, SET READ RETRY FLAG
1123	007402	032777	002000	171242		BIT	#2000, @SWR	, SEE IF PRINT?
1124	007410	001004				BNE	RRTYO	, IF NOT: BR
1125	007412	012704	023761			MOV	#MSG53, R4	
1126	007416	004737	020536			JSR	PC, TTOUT	, PRINT ORIGINAL ERROR MESSG
1127	007422	005037	000752		RRTYO	CLR	RTCNT	, CLEAR RETRY COUNT
1128	007426	004737	007600		RRTY1	JSR	PC, YOZ	, GO REREAD

 , READ ERROR RETRY
 , *****

1129	007432	005237	000752		INC	RTCNT	, BUMP RETRY COUNT
1130	007436	005737	000742		TST	SERFL	, SEE IF ERROR?
1131	007442	001431			BEQ	RRTY4	, IF NOT BR
1132	007444	032777	002000	171200	BIT	#2000, @SWR	, SEE IF PRINT?
1133	007452	001010			BNE	RRTY2	, IF NOT BR
1134	007454	012704	024576		MOV	#MSG71, R4	
1135	007460	004737	020536		JSR	PC, TTOUT	, PRINT FAILED RETRY MESSG
1136	007464	013703	000752		MOV	RTCNT, R3	
1137	007470	004737	020724		JSR	PC, OCTP	, PRINT RETRY NUMBER
1138	007474	022737	000004	000752	RRTY2	CMP	#4, RTCNT
1139	007502	001351			BNE	RRTY1	, IF NOT BR
1140	007504	012704	024624		MOV	#MSG72, R4	
1141	007510	004737	020536		JSR	PC, TTOUT	, PRINT SUSPECT HARD ERROR MESSG
1142	007514	013704	000734		RRTY3	MOV	UNP, R4
1143	007520	005264	001214		INC	BDRTY1(R4)	, BUMP HARD ERROR COUNT
1144	007524	000420			BR	RRTYX	
1145	007526	032777	002000	171116	RRTY4	BIT	#2000, @SWR
1146	007534	001010			BNE	RRTY5	, SEE IF SHOULD PRINT?
1147	007536	012704	024650		MOV	#MSG73, R4	, IF NOT BR
1148	007542	004737	020536		JSR	PC, TTOUT	, TYPE SUCCESSFUL RETRY MESSAGE
1149	007546	013703	000752		MOV	RTCNT, R3	
1150	007552	004737	020724		JSR	PC, OCTP	, PRINT RETRY COUNT
1151	007556	013704	000734		RRTY5	MOV	UNP, R4
1152	007562	005264	001174		INC	GDRTY1(R4)	, INCREASE SOFT ERROR COUNT
1153	007566	005037	000774		RRTYX	CLR	RRTYFL
1154	007572	004737	022126		JSR	PC, CKSWR	, CLEAR RETRY FLAG
1155	007576	000207			RTS	PC	, GO CHECK FOR G
1156							, RETURN

```

1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171 007600 012777 000001 171050 YOZ MOV #1, @TKS ; SET TTY ENABLE
1172 007606 013737 000640 000730 MOV YSTAL, STAL
1173 007614 004737 010634 JSR PC, STALL ; DO YOZZLE STALL
1174 007620 012777 177777 170756 YOZO MOV #-1, @MWC ; SET TO 1 RECORD SPACING
1175 007626 112777 000012 170746 YOZA MOV #12, @MTC ; SET TO SPACE REVERSE
1176 007634 012737 007654 000724 YOZB MOV #YOZC, RTRN ; SET RETURN ADDRESS
1177 007642 012737 177775 000730 MOV #177775, STAL ; SET TIME MULTIPLIER
1178 007650 000137 017006 JMP TAPG ; GO YOZZLE
1179 007654 013737 000640 000730 YOZC MOV YSTAL, STAL
1180 007662 004737 010634 JSR PC, STALL ; DO YOZZLE STALL
1181 007666 113777 000626 170706 MOV #R0CMD, @MTC ; SET READ COMMAND F OR P
1182 007674 012777 032216 170704 MOV #RDATA, @MOR ; SET READ ADDRESS
1183 007702 013777 000622 170674 MOV #CARCNT, @MWC ; SET CHARACTER COUNT
1184 007710 005737 000756 TST TMFLG ; IS IT A TM?
1185 007714 001410 BEQ YOZC1 ; IF NOT: BR
1186 007716 012777 177776 170660 MOV #-2, @MWC ; SET FOR TM
1187 007724 005737 001010 TST STCDFL ; SEE IF 7 TRK CORE DUMP
1188 007730 001402 BEQ YOZC1 ; IF NOT: BR
1189 007732 005277 170646 INC @MWC ; SET TO ONE CHARACTER
1190 007736 012737 007750 000724 YOZC1 MOV #YOZD, RTRN ; SET RETURN ADDRESS
1191 007744 000137 017006 JMP TAPG ; GO YOZZLE
1192 007750 032777 004000 170674 YOZD BIT #4000, @SWR ; SEE IF ERROR CHECK
1193 007756 001002 BNE YOZE ; IF NOT: BR
1194 007760 004737 016036 JSR PC, ERCHK ; ELSE GO CHECK ERRORS
1195 007764 005737 000774 YOZE TST RRTYFL ; IS IT A READ RETRY?
1196 007770 001401 BEQ YOZE1 ; IF NOT: BR
1197 007772 000207 RTS PC
1198 007774 032777 020000 170650 YOZE1 BIT #20000, @SWR ; SEE IF SHOULD CHECK DATA
1199 010002 001002 BNE YOZF ; IF NOT: BR
1200 010004 004737 013712 JSP PC, DCHK ; ELSE GO CHECK DATA
1201 010010 004737 012516 YOZF JSR PC, DS3 ; GO CLEAR DATA AREA
1202 010014 105777 170636 TSTB @TKS ; SEE IF HAVE NEW STALL VALUE
1203 010020 100034 BPL YOZG ; IF NOT: BR
1204 010022 122777 000203 170630 CMPB #203, @TKB ; SEE IF CONT C
1205 010030 001030 BNE YOZG ; IF NOT: BR
1206 010032 012704 023606 MOV #MSG44, R4
1207 010036 004737 020536 JSR PC, TTOUT ; PRINT YSTALL REQUEST
1208 010042 013703 000640 MOV YSTAL, R3
1209 010046 004737 020724 JSR PC, OCTP ; PRINT PRESENT STALL
1210 010052 010037 000712 MOV R0, TEMP3 ; SAVE R0(REC CNT)
1211 010056 012705 000640 MOV #YSTAL, R5 ; SET ADDRESS OF YSTL
1212 010062 012701 000006 MOV #6, R1 ; SET NUMBER OF CHAR TO INPUT
    
```

1213	010066	012702	177777			MOV	#-1,R2	,SET MAXIMUM LIMIT
1214	010072	012703	001000			MOV	#1000,R3	,SET MINIMUM LIMIT
1215	010076	004737	020272			JSR	PC,TTR	,GO GET VALUE
1216	010102	013700	000712			MOV	TEMP3,RO	,RESTORE RO(REC CNTR)
1217	010106	000137	007600			JMP	YOZ	,RESTART YOZZLE
1218	010112	032777	000040	170532	YOZG	BIT	#40,@SWR	,SEE IF SHOULD CONTINUE YOZZLE
1219	010120	001227				BNE	YOZ	,IF SO, BR
1220	010122	012777	000100	170526		MOV	#100,@TKS	,SET TTY INTERRUPT ENABLE
1221	010130	000207				RTS	PC	,EXIT
1222								

1223
 1224
 1225
 1226
 1227
 1228
 1229
 1230
 1231
 1232
 1233
 1234
 1235
 1236
 1237
 1238
 1239
 1240
 1241
 1242
 1243
 1244
 1245
 1246
 1247
 1248
 1249
 1250
 1251
 1252
 1253
 1254
 1255
 1256
 1257
 1258
 1259
 1260
 1261
 1262
 1263
 1264
 1265
 1266
 1267
 1268
 1269
 1270
 1271
 1272
 1273
 1274
 1275
 1276
 1277
 1278

010132 005037 000726
 010136 013700 000620
 010142 005100
 010144 005200
 010146 005737 000760
 010152 001407
 010154 013700 000760
 010160 042700 100000
 010164 005400
 010166 005037 000760
 010172 010037 000714
 010176 005737 000646
 010202 001520
 010204 012737 024472 000716
 010212 012777 177777 170364
 010220 013700 000620
 010224 063700 000714
 010230 004737 010466
 010234 032737 000002 000746
 010242 001401
 010244 000207
 010246 017737 170326 000740
 010254 032737 040000 000740
 010262 001053
 010264 005737 000726
 010270 001014
 010272 032777 002000 170352
 010300 001040
 010302 004737 017506
 010306 013704 000716
 010312 004737 020536
 010316 012704 022756
 010322 004737 020536
 010326 017703 170250
 010332 005037 000712
 010336 000303
 010340 004737 021152
 010344 000303
 010346 004737 021152
 010352 005737 000712
 010356 001011

BKSP
 BO

 BK0

 BK1

 BK1C

 BK2

 BK1A

CLR HDRFL
 MOV RCNT,RO
 COM RO
 INC RO
 TST EOTREC
 BEQ BK0
 MOV EOTREC,RO
 BIC #100000,RO
 NEG RO
 CLR EOTPEC
 MOV RO,TEMP4
 TST TMEX
 BEQ BDR
 MOV #MSG63,EMADDR
 MOV #-1,AMWC
 MOV RCNT,RO
 ADD TEMP4,RO
 JSR PC,SPBK
 BIT #2,BTFLG
 BEQ BK1
 RTS PC
 MOV AMTS,ERSAV
 BIT #40000,ERSAV
 BNE BK3
 TST HDRFL
 BNE BK2
 BIT #2000,ASWR
 BNE BK1B
 JSP PC,PAPRT
 MOV EMADDR,R4
 JSR PC,TTOUT
 MOV #MSG23,R4
 JSR PC,TTOUT
 MOV AMTC,R3
 CLR TEMP3
 SWAB R3
 JSR PC,DOUT
 SWAB R3
 JSR PC,DOUT
 TST TEMP3
 BNE BK1B

 BACKSPACE SUBROUTINE

 THIS SUBROUTINE IS USED TO PERFORM THE
 BACKSPACE OPERATION REQUIRED BY THE READ
 ROUTINE EITHER FOR READ FORWARD AFTER WRITING.
 OR FOR CONTINUOUS READING OF A DATA BLOCK
 WHEN IN READ ONLY MODE WITH SWITCH FOURTEEN (14)
 SET TO A ONE.
 A CHECK FOR RECORD COUNT ZERO IS MADE AT THE
 END OF THE SPACE OPERATION TO ASSURE THAT PROPER
 TAPE POSITIONING WAS DONE.

 CLR HDRFL .CLEAR HEADER FLAG
 MOV RCNT,RO
 COM RO .BUILD SPACE AMOUNT
 INC RO
 TST EOTREC .SEE IF EOT WAS REACHED
 BEQ BK0 .IF NOT BR
 MOV EOTREC,RO .GET SHORTENED BLOCK COUNT
 BIC #100000,RO
 NEG RO
 CLR EOTPEC .CLEAR EOT FLAG
 MOV RO,TEMP4 .SAVE BACKSPACE COUNT
 TST TMEX .IS THERE A TM?
 BEQ BDR .IF NOT: BR
 MOV #MSG63,EMADDR .POINT TO TM SP ERROR MESSG
 MOV #-1,AMWC .SET FOR 1 RECORD
 MOV RCNT,RO .RO=RECORD COUNT
 ADD TEMP4,RO .RO=RCNT-BACKSPACE CNT FOR HEADER
 JSR PC,SPBK .BACKSPACE OVER TM
 BIT #2,BTFLG .WAS THERE AN ERROR
 BEQ BK1 .IF NOT BR
 RTS PC .ELSE RETURN WITH EPROP FLAG SET
 MOV AMTS,ERSAV .GET STATUS
 BIT #40000,ERSAV .IS TM SET
 BNE BK3 .IF SET BR
 TST HDRFL .ALREADY PRINTED HEADER?
 BNE BK2 .IF SO BR
 BIT #2000,ASWR .SHOULD PRINT?
 BNE BK1B .IF NOT BR
 JSP PC,PAPRT .PRINT HEADER
 MOV EMADDR,R4 .POINT TO TM SP ERROR
 JSR PC,TTOUT .PRINT ERROR
 MOV #MSG23,R4
 JSR PC,TTOUT .PRINT COMMAND HEADER
 MOV AMTC,R3
 CLR TEMP3
 SWAB R3 .POSITION MOST SIGNIFICANT
 JSR PC,DOUT .PRINT
 SWAB R3 .POSITION LEAST SIGNIFICANT
 JSR PC,DOUT .PRINT
 TST TEMP3 .SEE IF PRINTED STATUS
 BNE BK1B .IF SO BR

1279	010360	005237	000712			INC	TEMP3	, SET FLAG
1280	010364	012704	023143			MOV	#MSG30, R4	, PRINT STATUS HEADER
1281	010370	004737	020536			JSR	PC, TTOUT	
1282	010374	017703	170200			MOV	@MTS, R3	, LOAD STATUS
1283	010400	000756				BR	BK1A	, GO PRINT STATUS
1284	010402	052737	000002	000746	BK1B	BIS	#2, BTFLG	, SET BT FLAG (POSITION ERROR)
1285	010410	060207				RTS	PC	, RETURN
1286	010412	042737	142121	000740	BK3	BIC	#142121, ERSV	, LOOK FOR NON-TM ERRORS
1287	010420	001407				BEQ	BOB	, IF NOT BR
1288	010422	005737	000760			TST	EOTREC	, WAS EOT REACHED
1289	010426	001716				BEQ	BK1C	, IF NOT BR
1290	010430	042737	002000	000740		B C	#2000, ERSV	, CHECK FOR NON-EOT ERRORS
1291	010436	001312				BNE	BK1C	, IF ANY. BR
1292								
1293	010440	163700	000620		BOB	SUB	RCNT, RO	, AGAIN RO=BACKSPACE COUNT
1294	010444	012737	022477	000716	BOA	MOV	#MSG10, EMADDR	, POINT TO SE MESSG
1295	010452	005200				INC	RO	, RO=BACKSPACE COUNT+1
1296	010454	063700	000620			ADD	RCNT, RO	, RO=RCNT-BACKSPACE CNT + 1 FOR HEADER
1297	010460	013777	000714	170116		MOV	TEMP4, @MWC	
1298	010466	013737	000636	000730	SPBK	MOV	TSTAL, STAL	
1299	010474	004737	010634			JSR	PC, STALL	, DO STALL
1300	010500	005077	170074			CLR	@MTS	
1301	010504	105077	170072			CLRB	@MTC	
1302	010510	052777	000012	170064		BIS	#12, @MTC	, SET BACKSPACE OP
1303	010516	012737	010536	000724		MOV	#B1, RTRN	, SET RETURN ADDRESS
1304	010524	012737	177377	000730		MOV	#177377, STAL	, SET INTERRUPT TIME MULTIPLIER
1305	010532	000137	017006			JMP	TAPG	, GO DO SPACE
1306	010536	017701	170042		B1	MOV	@MWC, R1	, LOAD SPACE COUNTER
1307	010542	001426				BEQ	B2	, IF COUNT IS ZERO BR
1308	010544	032777	002000	170100		BIT	#2000, @SWR	, SEE IF PRINT
1309	010552	001017				BNE	B1A	, IF NOT BR
1310	010554	004737	017506			JSR	PC, PAPRT	, ELSE PRINT SPACE ERROR
1311	010560	013704	000716			MOV	EMADDR, R4	
1312	010564	004737	020536			JSR	PC, TTOUT	
1313	010570	012704	023635			MOV	#MSG45, R4	
1314	010574	004737	020536			JSR	PC, TTOUT	, PRINT SPACE COUNT HEADER
1315	010600	005301				DEC	R1	
1316	010602	005101				COM	R1	
1317	010604	010103				MOV	R1, R3	
1318	010606	004737	020724			JSR	PC, OCTP	, PRINT NUMBER OF RECORDS LEFT TO SPACE
1319	010612	012737	000002	000746	B1A	MOV	#2, BTFLG	, SET BAD TAPE FLAG
1320	010620	013737	000636	000730	B2	MOV	TSTAL, STAL	, DO STALL
1321	010626	004737	010634			JSR	PC, STALL	, STALL
1322	010632	000207				RTS	PC	, EXIT
1323								

```

1324
1325 ,*****
1326 ,STALL ROUTINE
1327
1328 ,THIS ROUTINE IS USED TO PROVIDE SOFTWARE DELAYS
1329 ,DURING READ, WRITE, TURN AROUND, AND YOZZLE
1330 ,THE DELAY TIMES MAY BE SET BY THE OPERATOR AT
1331 ,INITIAL START FROM 200(8) OR MAY BE MODIFIED
1332 ,AT ANY TIME BY ENTERING CNTRL C ON THE TTY AND
1333 ,INSERTING NEW VALUES IN RESPONSE TO THE REQUEST
1334 ,PRINTED
1335 ,THE READ STALL AND THE WRITE STALL ARE DELAYS
1336 ,EXECUTED BETWEEN EACH RECORD OF THE DATA BLOCK
1337 ,THE TURN AROUND STALL IS EXECUTED EACH TIME
1338 ,THE DIRECTION OF TAPE MOVEMENT IS CHANGED AND
1339 ,ALSO EACH TIME THE TAPE OPERATION CHANGES FROM
1340 ,WRITE TO READ OR READ TO WRITE.
1341 ,THE YOZZLE STALL IS EXECUTED ONLY DURING THE
1342 ,YOZZLE ROUTINE
1343 ,*****
    
```

```

1344
1345 010634 005337 000730 STALL DEC STAL
1346 010640 001375 BNE STALL ,DELAY
1347 010642 000207 PTS PC ,EXIT
    
```

```

1348
1349 ,*****
1350 ,RANDOM CHARACTER COUNT GENERATOR
1351
1352 ,THIS ROUTINE ENTERED VIA CONSOLE SWITCH
1353 ,SEVEN (7) IS USED TO GENERATE A RANDOM
1354 ,CHARACTER COUNT FOR EACH DATA BLOCK.
1355 ,ALL RECORDS WITHIN A GIVEN BLOCK WILL BE
1356 ,THE SAME, BUT EACH BLOCK WILL VARY
1357 ,THE LIMITS ARE TWENTY (20) TO TWO THOUSAND
1358 ,(2000) OCTAL CHARACTERS PER RECORD
1359 ,*****
    
```

```

1360
1361 010644 012701 177760 CCNTR MOV #-20,R1 ;SET HIGH LIMIT
1362 010650 012702 174000 MOV #-4000,R2 ;SET LOW LIMIT
1363 010654 004737 020240 JSR PC,RANG ;GO GENERATE NUMBER
1364 010660 013737 000676 000622 MOV RANSAN,CARCNT ;SET CHAR COUNT
1365 010666 012737 177777 012750 MOV #-1,PATS ;PRESET DATA PATTERN
1366 010674 000207 PTS PC ,EXIT
1367
    
```

```
1368  
1369  
1370  
1371  
1372  
1373  
1374  
1375  
1376  
1377  
1378  
1379 010676 012702 000001 RCNTR MOV #1,R2 ,SET LOW LIMIT  
1380 010702 012701 000500 MOV #500,R1 ,SET HIGH LIMIT  
1381 010706 004737 020240 JSR PC,RANG ,GO GENERATE NUMBER  
1382 010712 013737 000676 000620 MOV RANSV,RCNT ,SET RECORD COUNT  
1383 010720 000207 RTS PC ,EXIT  
1384  
1385  
1386  
1387  
1388  
1389  
1390  
1391  
1392  
1393  
1394  
1395  
1396  
1397  
1398  
1399  
1400  
1401  
1402  
1403  
1404  
1405  
1406  
1407  
1408  
1409  
1410  
1411  
1412 010722 005737 000700 TINP TST TINF ,SEE IF SHOULD INPUT FROM TTY  
1413 010726 001001 BNE TINPA ,IF SO BR  
1414 010730 000207 RTS PC ,EXIT  
1415 010732 005037 000734 TINPA CLR UNP ,CLEAR TABLE POINTER  
1416 010736 005037 004716 CLR REOTC ,CLEAR EOT UNIT COUNTER  
1417 010742 012700 000010 MOV #10,R0 ,SET SIZE OF TABLE  
1418 010746 012701 001010 MOV #UN1,R1 ,SET START OF TABLE  
1419 010752 005021 *TINPB CLR (R1)+ ,CLEAR TABLE  
1420 010754 005300 DEC R0 ,SEE IF DONE  
1421 010756 001375 BNE TINPB ,IF NOT BR  
1422 010760 005737 001630 TST ASEQF ,SEE IF AUTO SEQUENCE  
1423 010764 001405 BEQ TINPB1 ,IF NOT BR
```

1424	010766	012704	024737		MOV	#MSG77,R4	
1425	010772	004737	020536		JSR	PC,TTOUT	,PRINT AUTO SEQ PROGRAM NAME
1426	010776	000410			BR	TINPO	
1427	011000	012704	023152	TINPB1	MOV	#MSG31,R4	
1428	011004	004737	020536		JSR	PC,TTOUT	,PRINT PROGRAM NAME
1429	011010	012704	023267		MOV	#MSG31A,R4	
1430	011014	004737	020536		JSR	PC,TTOUT	,PRINT REST OF TITLE
1431	011020	122737	000004	000041	TINPO	CMPB	#4,#41
1432	011026	001006			BNE	1\$,SEE IF LOAD MEDIUM
1433	011030	012704	026124		MOV	#MSG97,R4	,IF NOT BR
1434	011034	004737	020536		JSR	PC,TTOUT	;ELSE PRINT NO TEST
1435	011040	000137	004662		JMP	REOT10	;END TEST
1436	011044	012704	025261	1\$	MOV	#MSG84,R4	
1437	011050	004737	020536		JSR	PC,TTOUT	,REQUEST STARTING REGISTER ADDRESS
1438	011054	013703	000600		MOV	MTS,R3	
1439	011060	004737	020724		JSR	PC,OCTP	,PRINT CURRINT REGISTER START
1440	011064	013705	000674		MOV	REGST,R5	,SAVE ADDRESS LOCATION
1441	011070	012701	000006		MOV	#6,R1	,SET SIZE OF ENTRY
1442	011074	012702	177770		MOV	#177770,R2	,SET UPPER LIMIT
1443	011100	012703	170000		MOV	#170000,R3	,SET LOWER LIMIT
1444	011104	004737	020272		JSR	PC,TTR	,GO GET RESPONSE
1445	011110	012705	000602		MOV	#MTC,R5	,SET TABLE BASE
1446	011114	013704	000600		MOV	MTS,R4	,GET INITIAL ADDRESS
1447	011120	062704	000002	2\$	ADD	#2,R4	,BUMP ADDRESS
1448	011124	010425			MOV	R4,(R5)+	,FILL TABLE
1449	011126	020527	000614		CMP	R5,#MTRD+2	,DONE?
1450	011132	001372			BNE	2\$,IF NOT BR
1451	011134	012704	025304		MOV	#MSG85,R4	
1452	011140	004737	020536		JSR	PC,TTOUT	,REQUEST VECTOR ADDR
1453	011144	013703	000614		MOV	VECT,R3	
1454	011150	004737	020724		JSR	PC,OCTP	,PRINT CURRENT VECTOR
1455	011154	012705	000614		MOV	#VECT,R5	,SET SAVE LOCATION
1456	011160	012701	000003		MOV	#3,R1	,SET SIZE OF RESPONSE
1457	011164	012702	000476		MOV	#476,R2	,SET UPPER LIMIT
1458	011170	012703	000060		MOV	#60,R3	,SET LOWER LIMIT
1459	011174	004737	020272		JSR	PC,TTR	,GO GET RESPONSE
1460	011200	013700	000614		MOV	VECT,R0	,GET VECTOR ADDRESS
1461	011204	012720	017466		MOV	#MTINT,(R0)+	,LOAD VECTOR WITH HANDLER ADDR
1462	011210	012710	000340		MOV	#340,(R0)	,LOAD PRIORITY LEVEL
1463	011214	005737	021632		TST	ASEQF	,SEE IF AUTO SEQ
1464	011220	001403			BEQ	TINPO0	,IF NOT BR
1465	011222	005726			TST	(SP)+	,RESET STACK
1466	011224	000137	021260		JMP	ASEQ	,GO TO AUTO SEQ
1467	011230	012704	023322	TINPO0	MOV	#MSG32,R4	
1468	011234	004737	020536		JSR	PC,TTOUT	,PRINT UNIT NUMBER REQUEST
1469	011240	005037	000710		CLR	TEMP2	,CLEAR BUFFER
1470	011244	012705	000710		MOV	#TEMP2,R5	,SET UNIT DESCRIPTION BUFFER ADDRESS
1471	011250	012701	000001		MOV	#1,R1	,SET NUMBER OF CHARACTERS TO INPUT
1472	011254	012702	000007		MOV	#7,R2	,SET MAXIMUM LIMIT
1473	011260	012703	000000		MOV	#0,R3	,SET MINIMUM LIMIT
1474	011264	004737	020272		JSR	PC,TTR	,GO GET UNIT NUMBER
1475	011270	005737	000706		TST	TEMP1	,SEE IF HAVE NEW PARAMETER
1476	011274	001014			BNE	TINPOB	,IF SO BR
1477	011276	005737	000734		TST	UNP	,SEE IF FIRST ENTRY
1478	011302	001002			BNE	TINPOA	,IF NOT BR
1479	011304	000137	011230		JMP	TINPO0	,ELSE PENTRY

1536	011660	004737	020536		JSR	PC, TTOUT	. PRINT RECORD COUNT REQUEST
1537	011664	013703	000620		MOV	RCNT, R3	
1538	011670	004737	020724		JSR	PC, OCTP	. PRINT RECORD COUNT
1539	011674	012705	000620		MOV	#RCNT, R5	. SET RECORD COUNT ADDRESS
1540	011700	012701	000006		MOV	#6, R1	. SET NUMBER OF CHARACTERS TO INPUT
1541	011704	012702	177777		MOV	#-1, R2	. SET MAXIMUM LIMIT
1542	011710	012703	000001		MOV	#1, R3	. SET MINIMUM LIMIT
1543	011714	004737	020272		JSR	PC, TTR	. GO GET RECORD COUNT
1544	011720	013737	000620	000642	MOV	RCNT, RCSAV	. SAVE RECORD COUNT
1545	011726	012704	023412		MOV	#MSG36, R4	
1546	011732	004737	020536		JSR	PC, TTOUT	. PRINT CHARACTER COUNT REQUEST
1547	011736	005437	000622		NEG	CARCNT	
1548	011742	013703	000622		MOV	CARCNT, R3	
1549	011746	004737	020724		JSR	PC, OCTP	. PRINT CHAR COUNT
1550	011752	012705	000622		MOV	#CARCNT, R5	. SET CHARACTER COUNT ADDRESS
1551	011756	012701	000006		MOV	#6, R1	. SET NUMBER OF CHARACTERS TO INPUT
1552	011762	012702	004000		MOV	#4000, R2	. SET MAXIMUM LIMIT
1553	011766	012703	000004		MOV	#4, R3	. SET MINIMUM LIMIT
1554	011772	004737	020272		JSR	PC, TTR	. GO GET CHARACTER COUNT
1555	011776	005437	000622		NEG	CARCNT	. SET TO TWO'S COMPLEMENT
1556	012002	013737	000622	000644	MOV	CARCNT, CCSAV	. SAVE CHAR COUNT
1557	012010	012704	023436		MOV	#MSG37, R4	. PRINT PATTERN NUMBER REQUEST
1558	012014	004737	020536		JSR	PC, TTOUT	
1559	012020	013703	000624		MOV	PATRN, R3	
1560	012024	004737	020724		JSR	PC, OCTP	. PRINT PATTERN
1561	012030	005037	012746		CLR	DOFL	. CLEAR EXTERNAL DATA FLAG
1562	012034	012705	000624		MOV	#PATRN, R5	. SET PATTERN NUMBER ADDRESS
1563	012040	012701	000002		MOV	#2, R1	. SET NUMBER OF CHARACTERS TO INPUT
1564	012044	012702	000015		MOV	#15, R2	. SET MAXIMUM LIMIT
1565	012050	012703	000000		MOV	#0, R3	. SET MINIMUM LIMIT
1566	012054	004737	020272		JSR	PC, TTR	. GO GET PATTERN NUMBER
1567	012060	012704	024274		MOV	#MSG60, R4	. PRINT TM REQUEST
1568	012064	004737	020536		JSR	PC, TTOUT	
1569	012070	013703	000646		MOV	TMEX, R3	
1570	012074	004737	020724		JSR	PC, OCTP	. PRINT TMEX VALUE
1571	012100	012705	000646		MOV	#TMEX, R5	. SET TMEX ADDRESS
1572	012104	012701	000001		MOV	#1, R1	. SET NUMBER OF CHARACTERS TO INPUT
1573	012110	010102			MOV	R1, R2	. SET MAXIMUM LIMIT
1574	012112	005003			CLR	R3	. SET MINIMUM LIMIT
1575	012114	004737	020272		JSR	PC, TTR	. GO GET RESPONSE
1576	012120	012704	023461		MOV	#MSG38, R4	
1577	012124	004737	020536		JSR	PC, TTOUT	. PRINT SINGLE PASS REQUEST
1578	012130	013703	000630		MOV	SPFLG, R3	
1579	012134	004737	020724		JSR	PC, OCTP	. PRINT CURRENT FLAG SETTING
1580	012140	012705	000630		MOV	#SPFLG, R5	. GET ADDRESS OF FLAG
1581	012144	012701	000001		MOV	#1, R1	. SET SIZE OF RESPONSE
1582	012150	012702	000001		MOV	#1, R2	. SET UPPER LIMIT
1583	012154	012703	000000		MOV	#0, R3	. SET LOWER LIMIT
1584	012160	004737	020272		JSR	PC, TTR	. GO GET RESPONSE
1585	012164	012704	023521	TINP4	MOV	#MSG40, R4	
1586	012170	004737	020536		JSR	PC, TTOUT	. PRINT READ STALL REQUEST
1587	012174	013703	000632		MOV	RSTAL, R3	
1588	012200	004737	020724		JSR	PC, OCTP	. PRINT READ STALL
1589	012204	012705	000632		MOV	#RSTAL, R5	. SET READ STALL ADDRESS
1590	012210	012701	000006		MOV	#6, R1	. SET NUMBER OF CHARACTERS TO INPUT
1591	012214	012702	177777		MOV	#-1, R2	. SET MAXIMUM LIMIT

1592	012220	012703	000001		MOV	#1, R3	, SET MINIMUM LIM T
1593	012224	004737	020272		JSR	PC, TTR	, GO GET READ STALL
1594	012230	012704	023550		MOV	#MSG41, R4	
1595	012234	004737	020536		JSR	PC, TTOUT	, PRINT WRITE STALL PEQUEST
1596	012240	013703	000634		MOV	WSTAL, R3	
1597	012244	004737	020724		JSR	PC, OCTP	, PRINT READ STALL
1598	012250	012705	000634		MOV	#WSTAL, R5	, SET WRITE STALL ADDRESS
1599	012254	012701	000006		MOV	#6, R1	, SET NUMBER OF CHARACTERS TO INPUT
1600	012260	012702	177777		MOV	#-1, R2	, SET MAXIMUM LIMIT
1601	012264	012703	000001		MOV	#1, R3	, SET MINIMUM LIMIT
1602	012270	004737	020272		JSR	PC, TTR	, GO GET WRITE STALL
1603	012274	012704	023562		MOV	#MSG42, R4	
1604	012300	004737	020536		JSR	PC, TTOUT	, PRINT TURN AROUND STALL REQUEST
1605	012304	013703	000636		MOV	TSTAL, R3	
1606	012310	004737	020724		JSR	PC, OCTP	, PRINT TA STALL
1607	012314	012705	000636		MOV	#TSTAL, R5	, SET TURN AROUND STALL ADDRESS
1608	012320	012701	000006		MOV	#6, R1	, SET NUMBER OF CHARACTERS TO INPUT
1609	012324	012702	177777		MOV	#-1, R2	, SET MAXIMUM LIMIT
1610	012330	012703	000001		MOV	#1, R3	, SET MINIMUM LIMIT
1611	012334	004737	020272		JSR	PC, TTR	, GO GET TURN AROUND STALL
1612	012340	000207			RTS	PC	, EXIT
1613							
1614							, UNIT DESCRIPTION POSITIONING SUBROUTINE*****
1615							
1616	012342	000241		TPOS	CLC		
1617	012344	006137	000710		ROL	TEMP2	, POSITION CHARACTER
1618	012350	005303			DEC	R3	, SEE IF DONE
1619	012352	001373			BNE	TPOS	, IF NOT BR
1620	012354	013700	000734		MOV	UNP, R0	, LOAD UNIT POINTER
1621	012360	053760	000710	001012	BIS	TEMP2, UN1(R0)	, LOAD CHARACTER INTO UN1(PO
1622	012366	000207			RTS	PC	, EXIT
1623							

1624
 1625
 1626
 1627
 1628
 1629
 1630
 1631
 1632
 1633
 1634
 1635
 1636
 1637
 1638
 1639
 1640
 1641
 1642
 1643
 1644
 1645
 1646
 1647
 1648
 1649
 1650
 1651
 1652
 1653
 1654
 1655
 1656
 1657
 1658
 1659
 1660
 1661
 1662
 1663
 1664
 1665
 1666
 1667
 1668
 1669
 1670
 1671
 1672
 1673
 1674
 1675
 1676
 1677
 1678

```

, *****
, DATA SETUP ROUTINE.
,
, THIS ROUTINE IS USED TO GENERATE INTO THE ENTIRE
, WRITE BUFFER (4000 OCTAL CHARACTERS) THE DATA PATTERN
, SELECTED BY THE OPERATOR. THESE ARE 20 (8) FIXED
, DATA PATTERNS AVAILABLE AND ONE SELECTION (DATA PATTERN 0)
, WHICH WILL READ ANY PATTERN PRESENTED AT THE
, HIGH SPEED PAPER TAPE READER THIS TAPE MUST BE PREPARED
, BY USING THE PROGRAM CALLED DTC
, RANDOM DATA MAY ALSO BE USED VIA CONSOLE
, SWITCH EIGHT (8)
, THIS ROUTINE IS ALSO USED TO CLEAR OUT THE
, READ BUFFER (4000 OCTAL CHARACTERS) BEFORE EACH
, RECORD IS READ
, *****
1642 DSUP TST RDFL , SEE IF DID RANDOM DATA
1643 BNE DS1 , IF SO: BR
1644 DSO TST ASEQF , SEE IF AUTO SEQ
1645 BEQ DSOA , IF NOT: BR
1646 TST PATRN , SEE IF AUTO RANDOM
1647 BPL DSOA , IF NOT: BR
1648 JSR PC, DATR , ELSE GO GENERATE RANDOM DATA
1649 RTS PC , RETURN
1650 DSOR CMP PATRN, PATS , NEW PATTERN?
1651 BNE DSOC , IF SO: BR
1652 MOV UDES, R3 , GET UNIT DESCRIPTION
1653 BIC #173777, R3 , MASK PARITY
1654 CMP PARS, R3 , SEE IF SAME AS LAST TIME
1655 BEQ DSOB , IF SO, BR
1656 MOV R3, PARS , SAVE PARITY
1657 JSR PC, CRCLRC , GO GENERATE EXPECTED CRC/LRC
1658 DSOB RTS PC
1659 DSOC
1660 MOV #WDATA, R3 , R3 = ADDRS OF WRITE BUFFER
1661 MOV PATRN, R1 , R1 = PATTERN SELECTOR
1662 MOV R1, PATS
1663 ADD #1, R1 , BUMP POINTER
1664 CLC
1665 ROL R1 , MAKE PATTERN SELECTOR EVEN
1666 JMP @DATBL(R1) , GO GENERATE PATTERN
1667 DS1 NOP
1668 JSR PC, CRCLRC
1669 DS3 MOV #2002, R2 , R2=BUFFER SIZE +2
1670 MOV #RDATA, R1 , R1=READ DATA START
1671 DS4 CLR (R1)+ , CLEAR BUFFER
1672 DEC R2 , SEE IF DONE ALL
1673 BNE DS4 , IF NOT BR
1674 MOV UDES, PARS , SET PARITY
1675 BIC #173777, PARS
1676 RTS PC , EXIT
    
```

```

1679
1680      .EXTERNAL DATA INPUT FROM H/S READER (256 CHARACTER MAXIMUM)
1681
1682 012552 005737 012746      DATO      TST      DOFL      .SEE IF SHOULD DO EXTERNAL INPUT
1683 012556 001354              BNE      DS1      .IF NOT: BR
1684 012560 012737 000001 012746      MOV      #1,DOFL  .SET EXTERNAL FLAG
1685 012566 005077 166076              CLR      @PRB    .CLEAR READER BUFFER
1686 012572 005077 166070              CLR      @PRS    .CLEAR READER STATUS
1687 012576 005037 000706              CLR      TEMP1   .CLEAR FOR USE AS CHARACTER FLAG
1688 012602 052777 000001 166056  DATOA    BIS      #1,@PRS   .START READER
1689 012610 005037 000714              CLR      TEMP4
1690 012614 012704 000004              MOV      #4,R4   .SET UP READER DONE DELAY
1691 012620 032777 000200 166040  DATOB    BIT      #200,@PRS .SEE IF DONE
1692 012626 001006              BNE      1$      .IF SO BR
1693 012630 005337 000714              DEC      TEMP4
1694 012634 001371              BNE      DATOB   .DELAY FOR READER DONE
1695 012636 005304              DEC      R4
1696 012640 001367              BNE      DATOB   .CONTINUE DELAY
1697 012642 000722              BR       DS1     .IF READER NEVER DONE BR
1698 012644 005001              CLR      R1     .CLEAR SAVE LOCATION
1699 012646 117701 166016      MOVB    @PRB,R1  .SAVE CHARACTER
1700 012652 005737 000706      TST     TEMP1   .SEE IF HAVE FOUND START CHARACTER
1701 012656 001012              BNE      DATOC   .IF SO BR
1702 012660 105701              TSTB    R1     .SEE IF CHARACTER IS 0
1703 012662 001747              BEQ     DATOA    .IF SO BR
1704 012664 012737 000001 000706      MOV     #1,TEMP1 .ELSE SET CHARACTER FOUND FLAG
1705 012672 010137 000710      MOV     R1,TEMP2 .SAVE DATA SIZE
1706 012676 010102              MOV     R1,R2   .SAVE DATA SIZE
1707 012700 000137 012602              JMP     DATOA    .GO GET FIRST DATA CHAR
1708 012704 110123      DATOC   MOVB    R1,(R3)+ .LOAD BUFFER
1709 012706 005302              DEC     R2     .SEE IF READ ALL
1710 012710 001334              BNE     DATOA   .IF NOT BR
1711 012712 012701 026204      DATOD   MOV     #WDATA,R1 .R1 = START OF WRITE BUFFER
1712 012716 013707 000710      DATOE   MOV     TEMP2,R2  .R2 = SIZE OF DATA FIELD
1713 012722 112123      DATOE   MOVB    (R1)+,(R3)+ .REPEAT LOAD OF DATA FIELD
1714 012724 022703 032216      CMP     #RDATA,R3 .SEE IF DONE
1715 012730 003002              BGT     DATOF   .IF NOT BR
1716 012732 000137 012510      DATOF   JMP     DS1     .EXIT
1717 012736 005302      DATOF   DEC     R2     .SEE IF AT END OF DATA FIELD
1718 012740 001370              BNE     DATOE   .IF NOT BR
1719 012742 000137 0127.2              JMP     DATOD   .ELSE RESTART FILL
1720 012746 000000      DOFL    0       .EXTERNAL DATA FLAG=1 IF ALREADY DONE
1721 012750 177777      PATS    -1
1722 012752 177777      PARS    -1
1723
    
```

```

1724
1725          , ALL ONES*****
1726
1727 012754 012701 177777 DAT1  MOV    #-1,R1      ,R1=DATA
1728 012760 012702 002002 DAT1A MOV    #2002,R2    ,R2=WORD COUNT +2
1729 012764 010123          DAT1B MOV    R1,(R3)+    ,LOAD BUFFER
1730 012766 005302          DEC    R2            ,SEE IF DONE
1731 012770 001375          BNE   DAT1B         ,IF NOT BR
1732 012772 000137 012510          JMP    DS1          ,RETURN
1733
1734          , ALL ZEROS*****
1735
1736 012776 005001          DAT2  CLR    R1        ,R1=DATA
1737 013000 000137 012760          JMP    DAT1A       ,LOAD BUFFER
1738
1739          ; WALKING ONE*****
1740
1741 013004 012701 000001 DAT3.  MOV    #1,R1      ,R1=DATA
1742 013010 000241          CLC
1743 013012 012702 004C04 DAT3A. MOV    #4004,R2    ;R2=CHARACTER COUNT+4
1744 013016 110123          DAT3B MOVB   R1,(R3)+    ;LOAD BUFFER
1745 013020 106101          ROLB  R1            ,SET NEXT CHARACTER
1746 013022 005302          DEC    R2            ,SEE IF DONE
1747 013024 001374          BNE   DAT3B         ,IF NOT BR
1748 013026 000137 012510          JMP    DS1          ,RETURN
1749
1750          ; WALKING ZERO*****
1751
1752 013032 012701 000376 DAT4   MOV    #376,R1     ,R1=START OF DATA
1753 013036 000261          SEC
1754 013040 000137 013012          JMP    DAT3A       ,LOAD BUFFER
1755
1756          , ALTERNATING ONE/ZERO*****
1757
1758
1759 013044 012701 052525 DAT5   MOV    #52525,R1    ,R1=DATA
1760 013050 000137 012760          JMP    DAT1A       ,LOAD BUFFER
1761
1762          , ALTERNATING ZERO/ONE*****
1763
1764 013054 012701 125252 DAT6   MOV    #125252,R1   ,R1=DATA
1765 013060 000137 012760          JMP    DAT1A       ,LOAD BUFFER
1766
1767          , ONE/ZERO IN ALTERNATING CHARACTERS*****
1768
1769 013064 012701 125125 DAT7   MOV    #125125,R1   ,R1=DATA
1770 013070 000137 012760          JMP    DAT1A       ,LOAD BUFFER
1771
1772          , ZERO/ONE IN ALTERNATING CHARACTERS*****
1773
1774 013074 012701 052652 DAT10  MOV    #52652,R1   ,R1=DATA
1775 013100 000137 012760          JMP    DAT1A       ,LOAD BUFFER
1776
    
```

```

1777
1778
1779
1780 013104 005001          DAT11  CLR      R1          ;R1=STARTING DATA
1781 013106 012702 004004  MOV      #4004,R2        ;R2=CHARACTER COUNT+4
1782 013112 110127          DAT11A MOVB     R1,(R3)+       ;LOAD BUFFER
1783 013114 105201          INCB     R1              ;BUMP DATA
1784 013116 005302          DEC      R2              ;SEE IF DONE
1785 013120 001374          BNE      DAT11A         ;IF NOT BR
1786 013122 000137 012510  JMP      DS1             ;RETURN
1787
1788
1789
1790 013126 012701 000377  DAT12  MOV      #377,R1      ;R1=STARTING DATA
1791 013132 012702 004004  MOV      #4004,R2        ;R2=CHARACTER COUNT+4
1792 013136 110123          DAT12A MOVB     R1,(R3)+       ;LOAD BUFFER
1793 013140 105301          DECB     R1              ;BUMP DATA
1794 013142 005302          DEC      R2              ;SEE IF DONE
1795 013144 001374          BNE      DAT12A         ;IF NOT BR
1796 013146 000137 012510  JMP      DS1             ;RETURN
1797
1798
1799
1800 013152 012701 000377  DAT13  MOV      #377,R1      ;R1 = DATA
1801 013156 000137 012760  JMP      DAT1A           ;LOAD BUFFER
1802
1803
1804
1805 013162 012701 177400  DAT14  MOV      #177400,R1     ;R1 = DATA
1806 013166 000137 012760  JMP      DAT1A           ;LOAD BUFFER
1807
1808
1809
1810 013172 012702 002002  DAT15  MOV      #2002,R2        ;SET NUMBER OF WORDS
1811 013176 012701 177376  DAT15R MOV      #177376,R1     ;SET START OF DATA
1812 013202 012704 000002  DAT15A MOV      #2,R4          ;SET NUMBER OF REPEATS
1813 013206 010123          DAT15B MOV      R1,(R3)+       ;LOAD DATA
1814 013210 005302          DEC      R2              ;SEE IF DONE
1815 013212 001002          BNE      DAT15C         ;IF NOT BR
1816 013214 000137 012510  JMP      DS1             ;RETURN
1817 013220 005304          DAT15C DEC      R4          ;SEE IF DONE REPEATS
1818 013222 001371          BNE      DAT15B         ;IF NOT BR
1819 013224 000261          SEC
1820 013226 006101          ROL      R1              ;SET NEXT PATTERN
1821 013230 103764          BCS      DAT15A         ;SEE IF SHOULD RESTART
1822 013232 000137 013176  JMP      DAT15R         ;IF SO BR
1823

```

```

1824
1825
1826
1827 013236 013704 000622      DATR  MOV  CARCNT,R4      ,SET SIZE OF RECORD
1828 013242 012703 026204      MOV  #WDATA,R3      ;SET ADDRESS OF START OF BUFFER
1829 013246 012701 177777      MOV  #-1,R1         ,SET HIGH LIMIT
1830 013252 005002              CLR  R2             ,SET LOW LIMIT
1831 013254 004737 020240      DATRO JSR  PC,RANG      ,GO GENERATE NUMBER
1832 013260 013723 000676      MOV  RANSV,(R3)+    ,LOAD BUFFER
1833 013264 005204              INC  R4             ,SEE IF DONE ALL
1834 013266 001372              BNE  DATRO          ,IF NOT, BR
1835 013270 004737 012510      JSR  PC,DS1         ,GO CHECK FOR 7 CH
1836 013274 012737 000001 013304  MOV  #1,RDFL        ,SET RANDOM DATA FLAG
1837 013302 000207              RTS  PC             ,EXIT
1838 013304 000000      RDFL  0             ,RANDOM DATA SELECT FLAG
    
```

1

```

1839
1840 ,*****
1841 ,CRC/LRC CHARACTER BUILD,
1842
1843 ,THIS ROUTINE WILL CONSTRUCT AND SAVE THE EXPECTED
1844 ,CRC AND LRC CHARACTERS ACCORDING TO DATA AND
1845 ,RECORD SIZE IF OPERATING IN NRZ MODE
1846 ,*****
1847
1848 013306 000240 CRCLRC NOP
1849 013310 013700 000622 CRLR MOV CARCNT,RO ,SET RECORD SIZE
1850 013314 005400 NEG RO
1851 013316 012701 026204 MOV #WDATA,R1 ,SET START OF BUFFER
1852 013322 005037 013674 CLR XORS
1853 013326 111104 CLO MOVB (R1),R4 ,GET CHARACTER
1854 013330 004737 013522 JSR PC,CLP ,GO GET PARITY OF CHARACTER
1855 013334 004737 013650 JSR PC,XOR ,XOR CHARACTER
1856 013340 000241 CLC
1857 013342 006004 ROR R4 ,ROTATE 1 RIGHT
1858 013344 103014 BCC CL2 ,IF NO CARRY BR
1859 013346 052704 000400 BIS #400,R4 ,SET BIT NINE
1860 013352 000241 CLC
1861 013354 010405 CL1 MOV R4,R5 ,SAVE CHARACTER
1862 013356 042705 177703 BIC #177703,R5
1863 013362 005105 COM R5
1864 013364 042705 177703 BIC #177703,P5
1865 013370 042704 000074 BIC #74,R4
1866 013374 050504 BIS R5,R4 ,COMPLEMENT BITS 2,3,4,5
1867 013376 010437 013674 CL2 MOV R4,XORS
1868 013402 005300 DEC RO
1869 013404 001402 BEQ CLLAST ,IF LAST CHARACTER BP
1870 013406 000137 013326 JMP CLO ,GET NEXT
1871 013412 013704 013674 CLLAST MOV XORS,R4
1872 013416 005137 013674 COM XORS
1873 013422 042737 177050 013674 BIC #177050,XORS
1874 013430 042704 177727 BIC #177727,R4 ,COMPLEMENT ALL BUT BITS 3&5
1875 013434 050437 013674 BIS R4,XORS
1876 013440 013737 013674 013676 MOV XORS,EXCRC ,SAVE EXPECTED CPC
1877 013446 013700 000622 MOV CARCNT,RO
1878 013452 005400 NEG RO
1879 013454 012701 026204 MOV #WDATA,R1 ,DO EXPT LRC
1880 013460 005037 013674 CLR XORS
1881 013464 111104 CL3 MOVB (R1),R4
1882 013466 004737 013522 JSR PC,CLP ,GET PARITY
1883 013472 004737 013650 JSR PC,XOR ,XOR CHARACTER
1884 013476 005300 DEC RO
1885 013500 001371 BNE CL3 ,DO ALL FOR LRC
1886 013502 013704 013676 MOV EXCRC,R4
1887 013506 004737 013650 JSR PC,XOR ,XOR CRC TO DATA
1888 013512 013737 013674 013700 MOV XORS,EXLPC ,SAVE EXPT LRC
1889 013520 000207 RTS PC ,RETURN
1890 013522 005704 CLP TST R4 ,SEE IF 0 CHAR
1891 013524 001010 BNE CLPE ,IF NOT BR
1892 013526 032737 004000 000616 BIT #4000,UDES ,SEE IF EVEN PARITY
1893 013534 001404 BEQ CLPE ,IF NOT BR
1894 013536 012704 000420 MOV #420,R4 ,SET 0 CHAR EVEN PARITY
    
```

1895	013542	005201			INC	R1	,BUMP POINTER
1896	013544	000207			RTS	PC	,RETURN
1897	013546	005037	013710		CLPE	CLR	,CLEAR BIT COUNTER
1898	013552	012703	000010			MOV	,SET NUMBER OF BITS
1899	013556	032704	000001		CLPO	BIT	,SEE IF ONE BIT
1900	013562	001402				BEQ	,IF NOT BR
1901	013564	005237	013710			INC	,BUMP COUNTER
1902	013570	000241			CLP1	CLC	
1903	013572	006004				ROR	,ROTATE TO NEXT BIT
1904	013574	005303				DEC	
1905	013576	001367				BNE	,CONTINUE FOR ALL BITS
1906	013600	112104				MOVB	(R1)+,R4
1907	013602	042704	177400			BIC	#177400,R4
1908	013606	032737	000001	013710		BIT	,SEE IF ODD NUMBER OF ONE BITS
1909	013614	001005				BNE	,IF SO BR
1910	013616	032737	004000	000616		BIT	,SEE IF SHOULD BE EVEN PARITY
1911	013624	001406				BEQ	,IF NOT BR
1912	013626	000207				RTS	,ELSE EXIT
1913	013630	032737	004000	000616	CLP2	BIT	,SEE IF SHOULD BE ODD PARITY
1914	013636	001001				BNE	,IF NOT BR
1915	013640	000207				RTS	,ELSE EXIT
1916	013642	052704	000400		CLP3	BIS	,SET PARITY BIT
1917	013646	000207				RTS	
1918	013650	010446			XOR	MOV	R4 -(SP)
1919	013652	043716	013674			BIC	XORS,(SP)
1920	013656	040437	013674			BIC	R4,XORS
1921	013662	052637	013674			BIS	(SP)+,XORS
1922	013666	013704	013674			MOV	XORS,R4
1923	013672	000207				RTS	PC
1924							
1925	013674	000000			XORS	0	,XOR SAVE
1926	013676	000000			EXCPC	0	,EXPECTED CRC
1927	013700	000000			EXLRC	0	,EXPECTED LRC
1928	013702	000000			ACTLRC	0	,ACTUAL LRC
1929	013704	000000			LRC5V	0	,LRC SAVE
1930	013706	000000			LRCPT	0	,CRC PRINT FLAG
1931	013710	000000			PARCNT	0	,PARITY COUNTER
1932							

```

1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947 013712 005037 000722          DCHK  CLR  BBC          , CLEAR BAD RECORD CNTR
1948 013716 005037 000744          CLR  DERFL        , CLEAR DATA ERROR FLAG
1949 013722 005037 000726          CLR  HDRFL        , CLEAR HEADER FLAG
1950 013726 013705 000622          MOV  CARCNT, R5   , LOAD CHAR COUNT
1951 013732 012701 026204          MOV  #WDATA, R1  , SET WRITE DATA ADDR
1952 013736 012702 032216          MOV  #RDATA, R2  , SET READ DATA ADDR
1953 013742 032737 004000 000616  BIT  #4000, UDES  , SEE IF EVEN PARITY
1954 013750 001435                    BEQ  DFO          , IF NOT BR
1955 013752 005737 001010          TST  STCDFL      , SEE IF 7 TRK CORE DUMP
1956 013756 001032                    BNE  DFO          , IF SO BR
1957 013760 012703 000377          MOV  #377, R3
1958 013764 042703 177400          BIC  #177400, P3 , BACKGROUND DATA MASK
1959 013770 032777 000020 164602  BIT  #20, DMTS   , SEE IF 7 TRK DRIVE(NORMAL)
1960 013776 001402                    BEQ  DFA          , IF NOT BR
1961 014000 042703 000300          BIC  #300, R3    , MASK FOR 7 TRK NORMAL DATA
1962 014004 130311                    BITB R3, (R1)    , SEE IF ZERO CHARACTER
1963 014006 001404                    BEQ  DFC          ,
1964 014010 005201                    INC  R1          , BUMP POINTER
1965 014012 005205                    INC  P5          , SEE IF DONE
1966 014014 001373                    BNE  DFA          , IF NOT BR
1967 014016 000406                    BR   DFO
1968 014020 112721 000020          MOVB #20, (R1)+  , REPLACE 0 WITH 20
1969 014024 012737 177777 012750  MOV  #-1, PATS   , SET TO GENERATE NEW PATTERN
1970 014032 000767                    BR   DFB
1971 014034 013705 000622          MOV  CARCNT, R5  , RESET COUNT
1972 014040 012701 026204          MOV  #WDATA, R1  , RESET ADDRESS
1973 014044 032777 000020 164526  BIT  #20, DMTS   , SEE IF 7 TRACK
1974 014052 001403                    BEQ  DF9          , IF NOT BR
1975 014054 005737 001010          TST  STCDFL      , SEE IF 7 TRK CORE DUMP
1976 014060 001417                    BEQ  DF7          , IF NOT BR
1977 014062 122122                    CMPB (R1)+, (R2)+ , SEE IF DATA IS GOOD
1978 014064 001003                    BNE  DF91        , IF NOT BR
1979 014066 105037 000722          CLRB BBC         , ELSE CLEAR BAD RECORD COUNTER
1980 014072 000407                    BR   DF92
1981 014074 004737 014656          JSR  PC, DRPKF   , GO DO DROPS AND PICKS
1982 014100 004737 014242          JSR  PC, DERR    , GO PRINT ERROR
1983 014104 012737 000001 000744  MOV  #1, DERFL   , SET DATA ERROR FLAG
1984 014112 005205                    INC  R5          , SEE IF DONE ALL CHARACTERS
1985 014114 001362                    BNE  DF9         , IF NOT DO ALL
1986 014116 000432                    BR   DF3
1987 014120 000240                    NOP
1988 014122 010137 014240          MOV  R1, STAS    , SAVE CHARACTER ADDRESSES
    
```

1989	014126	117737	000106	014236		MOVB	@STAS, STCS	, SAVE CHARACTER
1990	014134	142711	000300			BICB	#300, (R1)	, MASK FOR 7 TRACK DRIVE
1991	014140	122122				CMPB	(R1)+, (R2)+	, SEE IF DATA IS GOOD
1992	014142	001003				BNE	DF71	, IF NOT BR
1993	014144	105037	000722			CLRB	BBC	, CLEAR BAD RECORD COUNTER
1994	014150	000407				BR	DF72	
1995	014152	004737	014656		DF71	JSR	PC, DPPKF	, GO DO DROPS AND PICKS
1996	014156	004737	014242			JSR	PC, DERR	, GC PRINT ERROR
1997	014162	012737	000001	000744		MOV	#1, DERFL	, SET DATA ERROR FLAG
1998	014170	000240			DF72	NOP		
1999	014172	153777	014236	000040		BISB	STCS, @STAS	, RESET DATA
2000	014200	005205				INC	R5	, SEE IF DONE ALL
2001	014202	001346				BNE	DF7	, IF NOT: DO ALL
2002	014204	005737	000744		DF3	TST	DERFL	, SEE IF HAD DATA ERROR
2003	014210	001411				BEQ	DFX	, IF NOT BR
2004	014212	005737	000742			TST	SERFL	
2005	014216	001006				BNE	DFX	, IF NOT DATA ERROR ONLY BR
2006	014220	013704	000734			MOV	UNP, R4	
2007	014224	005264	001134			INC	DATER1(R4)	, BUMP DATA ERROR COUNTER
2008	014230	004737	022126			SR	PC, CKSWP	, CHECK FOR G
2009	014234	000207			DFX	PTS	PL	, EXIT
2010	014236	000000			STCS	0		, 7 TRACK DATA SAVE
2011	014240	000000			STAS	0		, 7 TRACK ADDRESS SAVE

```

2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040 014242 032777 002000 16440 DEPR BIT #2000,DSWR ,SEE IF SHOULD PRINT ERRORS
2041 014250 001402 BEQ DERR0 ,IF SO BR
2042 014252 000137 014400 JMP DERR4 ,ELSE SKIP PRINT
2043 014256 005237 000732 DEPRD INC PFLG ,SET PRINT FLAG
2044 014262 005737 000726 TST HDRFL ,SEE IF HAVE PRINTED HEADER
2045 014266 001013 BNE DERR0A ,IF SO BR
2046 014270 005737 000742 TST SERFL ,ALREADY PRINTED HEADER?
2047 014274 001010 BNE DERR0A ,IF SO BR
2048 014276 004737 017506 JSR PC,PAPRT ,PRINT CYCLE NUMBER
2049 014302 012704 022434 MOV #MSG1,R4 ,LOAD ERROR MSG ADDR
2050 014306 004737 020536 JSR PC,TTOUT ,PRINT ERROR
2051 014312 004737 016742 JSR PC,FRPRT ,PRINT F OR R
2052 014316 012704 022453 DEPRDA MOV #MSG4,R4
2053 014322 004737 020536 JSR PC,TTOUT PRINT CHAR NO HEADER
2054 014326 010203 MOV R2,R3
2055 014330 162703 032216 SUB #RDATA,R3 ,POINT TO CHAR
2056 014334 005303 DEC R3
2057 014336 004737 020724 DEPRDB JSR PC,OCTP ,PRINT CHAR NUMBER
2058 014342 012704 022441 MOV #MSG2,R4
2059 014346 004737 020536 JSR PC,TTOUT ,PRINT EXPECTED DATA
2060 014352 114103 MOVB -(R1),R3 ,LOAD EXPECTED DATA
2061 014354 004737 021152 JSR PC,DOUT ,GO PRINT CHAR
2062 014360 012704 022446 MOV #MSG3,R4
2063 014364 004737 020536 JSR PC,TTOUT ,PRINT RECEIVED DATA
2064 014370 114203 DEPR1 MOVB -(R2),R3
2065 014372 004737 021152 DEPR2 JSR PC,DOUT ,PRINT BAD CHAR
2066 014376 122122 DEPR3 CMPB (R1)+,(R2)+ ,RESET POINTERS
2067 014400 105237 000722 DEPR4 INCB BBC ,BUMP BAD RECORD CNT
    
```

Line	Address	Offset	Hex	Label	Op	Opnd	Comment
2068	014404	122737	000010	000722	CMPB	#10,BBC	,SEE IF BLD BTH
2069	014412	001076			BNE	DEREX	,IF NOT BR
2070	014414	032777	002000	164230	BIT	#2000,@SWR	,SEE IF PRINT INHIBIT
2071	014422	001004			BNE	15	,IF SO BR
2072	014424	012704	022534		MOV	#MSG15,R4	
2073	014430	004737	020536		JSR	PC,TTOUT	,PRINT BLD BTH
2074	014434	105037	000722	15	CLRB	BBC	,RESET BAD RECORD CNTR
2075	014440	000337	000722		SWAB	BBC	,PCPOSITION BLD BTH AMOUNT
2076	014444	105237	000722		INCB	BBC	,BUMP AMOUNT
2077	014450	122737	000003	000722	CMPB	#3,BBC	,SEE IF HAD 3 BLD BTHS
2078	014456	101037			BHI	DERR4B	,IF NOT BR
2079	014460	000337	000722		SWAB	BBC	,REPOSITION BBC
2080	014464	022705	177767		CMF	#177767,R5	,SEE IF ON LAST EIGHT CHARS
2081	014470	101445			BLOS	DERR6	,IF SO BR
2082	014472	012705	177767		MOV	#177767,R5	,SET CHAR CNTR TO 8
2083	014476	013737	000622	000706	MOV	CARCNT,TEMP1	,LOAD CHAR COUNT
2084	014504	005137	000706		COM	TEMP1	
2085	014510	005237	000706		INC	TEMP1	
2086	014514	162737	000010	000706	SUB	#10,TEMP1	,POINT TO BUFFER -8
2087	014522	013701	000706		MOV	TEMP1,R1	,POINT TO NEXT CHAR
2088	014526	062701	026204		ADD	#WDATA,R1	,POINT TO NEXT WRITE CHAR
2089	014532	013702	000706		MOV	TEMP1,R2	,POINT TO END OF READ DATA -8 FORWARD
2090	014536	062702	032216		ADD	#RDATA,R2	,POINT TO NEXT CHAR
2091	014542	000422			BR	DEREX	,EXIT
2092	014544	012702	000010	DERR4A	MOV	#10,R2	,POINT TO THE END OF READ DATA -8 PEVEPSE
2093	014550	062702	032216		ADD	#RDATA,R2	,POINT TO THE NEXT CHAR
2094	014554	000415			BR	DEREX	,EXIT
2095	014556	000337	000722	DERR4B	SWAB	BBC	,REPOSITION BBC
2096	014562	000241			CLC		
2097	014564	062705	000024		ADD	#24,R5	,SKIP 20 CHARS
2098	014570	103405			BCS	DERR6	,IF EXCEED RECORD SIZE BP
2099	014572	062701	000024		ADD	#24,R1	,SKIP 20 CHARS
2100	014576	062702	000024	DERR5	ADD	#24,R2	SKIP FORWARD 20 CHARS
2101	014602	070402			BR	DEREX	
2102	014604	012705	177777	DERR6	MOV	#-1,R5	,SET TO EOR
2103	014610	032777	100000	164034	DEREX	#100000,@SWR	,SEE IF SHOULD HALT ON EPROP
2104	014616	001412			BEQ	DEREX1	,IF NOT BR
2105	014620	000000			HALT		
2106	014622	005737	000732		TST	PFLG	,SEE IF PRINTED
2107	014626	001006			BNE	DEREX1	,IF SO BR
2108	014630	032777	002000	164014	BIT	#2000,@SWR	,SEE IF SHOULD PRINT
2109	014636	001002			BNE	DEREX1	,IF NOT BR
2110	014640	000137	014256		JMP	DERRO	,ELSE PRINT
2111	014644	004737	022126	DEREX1	JSR	PC,CKSWP	,TEST FOR G
2112	014650	005037	000732		CLR	PFLG	,CLEAR FLAG
2113	014654	000207			RTS	PC	,RETURN
2114							

2115
 2116
 2117
 2118
 2119
 2120
 2121
 2122
 2123
 2124
 2125
 2126
 2127
 2128
 2129
 2130
 2131
 2132

 DROPS AND PICKS SUBROUTINE
 THIS SUBROUTINE IS USED TO ACCUMULATE FROM
 EACH BAD DATA CHARACTER FOUND THE NUMBER
 OF BITS WHICH WERE EITHER DROPPED OR PICKED UP
 TWO COUNTERS ARE USED TO ACCUMULATE THIS
 INFORMATION AND CAN STORE UP TO 32K DROPS
 OR PICKS BEFORE OVERFLOWING IF OVERFLOW IS
 ABOUT TO OCCUR, THESE ACCUMULATORS ARE
 PRINTED IN OCTAL AND RESET TO ZERO
 THE CONTENTS OF THE ACCUMULATORS MAY BE
 DISPLAYED AT ANY TIME BY SETTING CONSOLE
 SWITCH FOURTEEN TO A ONE (1) THE PRINTOUT WILL OCCUR
 AT THE END OF THE CURRENT BLOCK CYCLE

2133	014656	005037	000706	DRPKF	CLR	TEMP1		
2134	014662	005037	000710		CLR	TEMP2		
2135	014666	005037	000712		CLR	TEMP3		
2136	014672	013704	000734		MOV	UNP,R4		
2137	014676	016437	001034	000770	MOV	PIK1(R4),BPKP		
2138	014704	016437	001054	000766	MOV	DRP1(R4),BOPP		
2139	014712	124142			CMPB	-(R1),-(R2)	.POINT TO CHAR	
2140	014714	112137	000706		MOVB	(R1)+,TEMP1	.LOAD GOOD CHAR	
2141	014720	112237	000710		MOVB	(R2)+,TEMP2	.LOAD BAD CHAR	
2142	014724	004737	014736	DRPK	JSR	PC,DROP	.GET DROPS	
2143	014730	004737	015156		JSR	PC,PICK	.GET PICKS	
2144	014734	000207			RTS	PC	.EXIT	
2145	014736	113703	000706	DROP	MOVB	TEMP1,R3	.R3 = GOOD CHAR	
2146	014742	113704	000710		MOVB	TEMP2,R4	.R4 = BAD CHAR	
2147	014746	140403		DPC	BICB	R4,R3	.GET DROPS/PICKS	
2148	014750	001001			BNE	DPCG	.IF SOME BR	
2149	014752	000207			RTS	PC	.RETURN	
2150	014754	012737	000010	000736	DPCG	MOV	#10,BCNT	.SET NUMBER TO CHECK
2151	014762	132703	000001	DPCO	BITB	#1,R3	.SEE IF DROPPED OR PICKED THIS BIT	
2152	014766	001455			BEQ	DPC2	.IF NOT BR	
2153	014770	105737	000712		TSTB	TEMP3	.SEE IF ON PICKS	
2154	014774	001016			BNE	DPC1	.IF SO BR	
2155	014776	005277	163764		INC	@BOPP	.BUMP DROP CNTR	
2156	015002	005777	163760		TST	@BOPP		
2157	015006	100045			BPL	DPC2	.IF NO OVERFLOW BR	
2158	015010	032777	002000	163634	BIT	#2000,@SWR	.SEE IF HAVE PRINTED DATA	
2159	015016	001402			BEQ	DPCOA	.IF SO BR	
2160	015020	004737	017506		JSR	PC,PAPRT	.PRINT CYCLE NUMBER	
2161	015024	004737	015222	DPCOA	JSR	PC,DPPRT	.PRINT DROPS AND PICKS	
2162	015030	000415			BR	DPC2A		
2163	015032	005277	163732	DPC1	INC	@BPKP	.BUMP PICK CNTR	
2164	015036	005777	163726		TST	@BPKP	.SEE IF OVERFLOW	
2165	015042	100027			BPL	DPC2	.IF NOT: BR	
2166	015044	032777	002000	163600	BIT	#2000,@SWR	.SEE IF HAVE PRINTED DATA	
2167	015052	001402			BEQ	DPC1A	.IF SO BR	
2168	015054	004737	017506		JSR	PC,PAPRT	.PRINT CYCLE NUMBER	
2169	015060	004737	015222	DPC1A	JSR	PC,DPPRT	.PRINT DROPS AND PICKS	
2170	015064	013704	000734	DPC2A	MOV	UNP,R4		

2171	015070	016403	001054		MOV	DRP1(R4),R3	, SET DROP POINTER
2172	015074	016404	001034		MOV	PIK1(R4),R4	, SET PICK POINTER
2173	015100	012737	000010	000736	MOV	#10,BCNT	, SET NUMBER OF BITS
2174	015106	005023			CLR	(R3)+	, CLEAR DROPS
2175	015110	005024			CLR	(R4)+	, CLEAR PICK
2176	015112	005337	000736		DEC	BCNT	, SEE IF DONE
2177	015116	001373			BNE	DPC2B	, IF NOT BR
2178	015120	000207			RTS	PC	, EXIT
2179	015122	000241			CLC		
2180	015124	106003			RORB	R3	, GET NEXT BIT
2181	015126	005337	000736		DEC	BCNT	, SEE IF DONE
2182	015132	001410			BEQ	DPC3	
2183	015134	062737	000002	000770	ADD	#2,BPKP	
2184	015142	062737	000002	000766	ADD	#2,BDPP	
2185	015150	000137	014762		JMP	DPC0	, CONTINUE
2186	015154	000207			RTS	PC	, RETLRN
2187	015156	013704	000734		MOV	UNP,R4	, SET UNIT POINTER
2188	015162	016437	001034	000770	MOV	PIK1(R4),BPKP	, SET PICK POINTER
2189	015170	016437	001054	000766	MOV	DRP1(R4),BDPP	, SET DROP POINTER
2190	015176	113704	000706		MOVB	TEMP1,R4	, R4 = GOOD CHAR
2191	015202	113703	000710		MOVB	TEMP2,R3	, R3 = BAD CHAR
2192	015206	112737	000001	000712	MOVB	#1,TEMP3	, SET PICK FLAG
2193	015214	004737	014746		JSR	PC,DPC	, GO CHECK PICKS
2194	015220	000207			RTS	PC	, EXIT
2195	015222	012704	023121		MOV	#MSG26,R4	
2196	015226	004737	020536		JSR	PC,TTOUT	, PRINT DROP HEADER
2197	015232	013704	000734		MOV	UNP,R4	
2198	015236	016437	001054	000766	MOV	DRP1(R4),BDPP	, SET DROP POINTER
2199	015244	016437	001034	000770	MOV	PIK1(R4),BPKP	, SET PICK POINTER
2200	015252	062737	000016	000766	ADD	#16,BDPP	
2201	015260	062737	000016	000770	ADD	#16,BPKP	
2202	015266	012737	000010	000736	MOV	#10,BCNT	, SET NUMBER TO PRINT
2203	015274	017703	163466		MOV	@BDPP,R3	
2204	015300	004737	020724		JSR	PC,OCTP	, PRINT DROPS
2205	015304	005337	000736		DEC	BCNT	, SEE IF DONE
2206	015310	001404			BEQ	DPPRT1	, IF NOT BR
2207	015312	162737	000002	000766	SUB	#2,BDPP	, BUMP POINTER
2208	015320	000765			BR	DPPRTO	, CONTINUE FOR ALL 8 BITS
2209	015322	012737	000010	000736	MOV	#10,BCNT	, SET NUMBER TO PRINT
2210	015330	012704	023132		MOV	#MSG27,R4	
2211	015334	004737	020536		JSR	PC,TTOUT	, PRINT PICK HEADER
2212	015340	017703	163424		MOV	@BPKP,R3	
2213	015344	004737	020724		JSR	PC,OCTP	, PRINT PICKS
2214	015350	005337	000736		DEC	BCNT	, SEE IF DONE
2215	015354	001404			BEQ	DPPRTX	, IF SO BR
2216	015356	162737	000002	000770	SUB	#2,BPKP	, BUMP POINTER
2217	015364	000765			BR	DPPRT2	, CONTINUE FOR ALL 8 BITS
2218	015366	000207			RTS	PC	, RETURN

```
2219 , *****  
2220 , STATISTICS PRINT  
2221 ,  
2222 , THIS SUBROUTINE PRINTS THE ACCUMULATED  
2223 , ERROR STATISTICS FOR EACH DRIVE.  
2224 , THE ROUTINE CAN BE CALLED TO PRINT  
2225 , AT THE END OF EACH BLOCK BY SELECTING  
2226 , SW14=1. THE SUMMARY IS AUTOMATICALLY  
2227 , PRINTED FOR A DRIVE WHENEVER A TAPE  
2228 , IS REWOUND FROM EOT OR DROPPED  
2229 , *****  
2230  
2231 015370 012700 000001 PRSTAT MOV #1, R0 ; SET RECORD COUNTER TO 1  
2232 015374 004737 017506 JSR PC, PAPRT ; PRINT CYCLE NUMBER  
2233 015400 004737 015222 PRSTA2 JSR PC, DPPRT ; PRINT DROPS AND PICKS  
2234 015404 012704 024501 MOV #MSG64, R4  
2235 015410 004737 020536 JSR PC, TTOUT ; PRINT WRITE ERROR TAG  
2236 015414 013704 000734 MOV UNP, R4  
2237 015420 016403 001074 MOV WTER1(R4), R3  
2238 015424 004737 020724 JSR PC, OCTP ; PRINT WRITE ERRORS  
2239 015430 012704 024726 MOV #MSG76, R4  
2240 015434 004737 020536 JSR PC, TTOUT ; PRINT RETRY TOTAL  
2241 015440 013704 000734 MOV UNP, R4  
2242 015444 016403 001154 MOV RTY1(R4), R3  
2243 015450 004737 020724 JSR PC, OCTP ; PRINT RETRIES  
2244 015454 012704 024512 MOV #MSG65, R4  
2245 015460 004737 020536 JSR PC, TTOUT ; PRINT READ ERROR TAG  
2246 015464 013704 000734 MOV UNP, R4  
2247 015470 016403 001114 MOV RDER1(R4), R3  
2248 015474 004737 020724 JSR PC, OCTP ; PRINT REPD ERRORS  
2249 015500 012704 024704 MOV #MSG74, R4  
2250 015504 004737 020536 JSR PC, TTOUT ; PRINT SOFT ERROR MESSAGE  
2251 015510 013704 000734 MOV UNP, R4  
2252 015514 016403 001174 MOV GORTY1(R4), R3  
2253 015520 004737 020724 JSR PC, OCTP ; PRINT SOFT ERROR NUMBER  
2254 015524 012704 024715 MOV #MSG75, R4  
2255 015530 004737 020536 JSR PC, TTOUT ; PRINT HARD RD ERROR MESSG  
2256 015534 013704 000734 MOV UNP, R4  
2257 015540 016403 001214 MOV BDRTY1(R4), R3  
2258 015544 004737 020724 JSR PC, OCTP ; PRINT HARD RD ERROR COUNT  
2259 015550 012704 024523 MOV #MSG66, R4  
2260 015554 004737 020536 JSR PC, TTOUT ; PRINT DATA ERROR TAG  
2261 015560 013704 000734 MOV UNP, R4  
2262 015564 016403 001134 MOV DATER1(R4), R3  
2263 015570 004737 020724 JSR PC, OCTP ; PRINT DATA ERROR NUMBER  
2264 015574 004737 015606 JSR PC, BTPRT ; PRINT BAD TAPE STATS  
2265 015600 004737 022126 JSR PC, CKSWR ; CHECK FOR G  
2266 015604 000207 RTS PC ; RETURN  
2267
```

```

2268
2269 ;BAD TAPE STATISTICS PRINT*****
2270
2271 015606 005037 000754 BTPRT CLR RTYFL
2272 015612 012704 024534 MOV #MSG67, R4
2273 015616 004737 020536 JSR PC, TTOUT ; DO CR/LF
2274 015622 013704 000734 MOV UNP, R4
2275 015626 016437 002714 000762 MOV BTADDR(R4), BTPT ; SET TABLE POINTER
2276 015634 017703 163122 MOV @BTPT, R3
2277 015640 000241 CLC
2278 015642 006003 ROR R3 ; CORRECT NUMBER
2279 015644 004737 020724 JSR PC, OCTP ; PRINT NUMBER OF BAD SPOTS
2280 015650 012704 024536 MOV #MSG68, R4
2281 015654 004737 020536 JSR PC, TTOUT ; PRINT BAD TAPE TAG
2282 015660 005777 163076 TST @BTPT ; SEE IF ANY BAD SPOTS
2283 015664 001001 BNE BTOVD ; IF SO BR
2284 015666 000207 RTS PC
2285 015670 013701 000762 BTOVD MOV BTPT, R1 ; SET TABLE POINTER
2286 015674 005721 TST (R1)+
2287 015676 005000 CLR RO
2288 015700 010003 BTOV1 MOV RO, R3
2289 015702 000241 CLC
2290 015704 006003 ROR R3 ; R3=R3/2 FOR CORRECT NUMBER
2291 015706 004737 020724 JSR PC OCTP ; PRINT ENTRY NUMBER
2292 015712 012704 022521 MOV #MSG13, R4
2293 015716 105724 TSTB (R4)+ ; SKIP CR/LF
2294 015720 004737 020536 JSR PC, TTOUT ; PRINT BLOCK NUMBER TAG
2295 015724 011103 MOV (R1), R3
2296 015726 004737 020724 JSR PC, OCTP ; PRINT BLOCK NUMBER
2297 015732 012704 022527 MOV #MSG14, R4
2298 015736 004737 020536 JSR PC, TTOUT ; PRINT RECORD NUMBER TAG
2299 015742 062701 000040 ADD #40, R1
2300 015746 012103 MOV (R1)+, R3
2301 015750 004737 020724 JSR PC, OCTP ; PRINT RECORD NUMBER
2302 015754 162701 000040 SUB #40, R1 ; RESET POINTER FOR BLOCK NUMBER
2303 015760 005720 TST (RO)+
2304 015762 020077 162774 CMP RO, @BTPT ; SEE IF DONE
2305 015766 001405 BEQ BTOV2
2306 015770 012704 024534 MOV #MSG67, R4
2307 015774 004737 020536 JSR PC, TTOUT ; DO CR/LF
2308 016000 000737 BR BTOV1 ; CONTINUE
2309 016002 005737 000772 BTOV2 TST BTSTF ; SEE IF STAT ONLY PRINT
2310 016006 001002 BNE BTOVX ; IF SO BR
2311 016010 004737 016016 JSR PC, BTCLR ; CLEAR TABLE
2312 016014 000207 BTOVX RTS PC ; RETURN
2313
2314 ; CLEAR BAD TAPE TABLE
2315
2316 016016 012703 000041 BTCLR MOV #41, R3 ; SET SIZE OF TABLE
2317 016022 013704 000762 MOV BTPT, R4 ; SET POINTER
2318 016026 005024 BTCLR4 CLR (R4)+ ; CLEAR ENTRY
2319 016030 005303 DEC R3 ; DONE?
2320 016032 001375 BNE BTCLR1 ; IF NOT BR
2321 016034 000207 RTS PC ; RETURN
    
```

```

2322 ;*****
2323 ;READ/WRITE STATUS CHECK SUBROUTINE.
2324 ;
2325 ;THIS SUBROUTINE IS USED TO PERFORM A CHECK
2326 ;OF THE TAPE STATUS REGISTER FOR ERRORS AND
2327 ;TO ASSURE A CORRECT CURRENT MEMORY ADDRESS
2328 ;AND CHARACTER COUNT AT THE END OF EACH TAPE
2329 ;OPERATION (READ OR WRITE).
2330 ;IF A STATUS ERROR IS INDICATED BY BIT FIFTEEN (15)
2331 ;OF THE COMMAND REGISTER BEING SET, THEN AN ERROR
2332 ;HEADER CONSISTING OF UNIT NUMBER, BLOCK NUMBER,
2333 ;RECORD NUMBER, RECORD SIZE, AND TYPE OF ERROR
2334 ;WILL BE PRINTED FOLLOWED BY THE CONTENTS OF
2335 ;THE COMMAND REGISTER AND STATUS REGISTER PLUS
2336 ;THE CURRENT MEMORY ADDRESS AND CHARACTER COUNT.
2337 ;IF NO STATUS ERROR IS INDICATED, THE CHARACTER COUNT
2338 ;AND CURRENT MEMORY ADDRESS ARE BOTH CHECKED AND
2339 ;THE ENTIRE PRINT OUT IS DONE IF EITHER IS IN ERROR
2340 ;ERROR PRINT OUTS MAY BE DISALLOWED BY SETTING CONSOLE
2341 ;SWITCH TEN (10) TO A ONE (1).
2342 ;THE PROGRAM MAY BE HALTED ON ANY ERROR BY SETTING
2343 ;CONSOLE SWITCH FIFTEEN TO A ONE (1)
2344 ;*****
2345
2346 016036 013703 000622 ERCHK MOV CARCNT, R3 ;GET CHARACTER COUNT
2347 016042 004737 022126 JSR PC, CKSWR ;CHECK FOR G
2348 016046 005037 000742 CLR SERFL ;CLEAR STATUS ERROR FLAG
2349 016052 005403 NEG R3
2350 016054 005737 000756 TST TMFLG ;A TM OPERATION?
2351 016060 001413 BEQ EROA ;IF NOT BR
2352 016062 012703 000002 MOV #2, R3
2353 016066 005737 001010 TST STCDL ;SEE IF 7 TRK CORE DUMP
2354 016072 001401 BEQ 15 ;IF NOT BR
2355 016074 005303 DEC R3 ;SET TO ONE CHARACTER
2356 016076 032777 000004 162476 15 BIT #4, @MTC ;SEE IF A WRITE TM?
2357 016104 001401 BEQ EROA ;IF NOT BR
2358 016106 005003 CLR R3 ;ELSE CLEAR R3
2359 016110 032777 000004 162464 EROA BIT #4, @MTC ;SEE IF WRITE OP
2360 016116 001404 BEQ ERO
2361 016120 062703 026204 ADD #WDATA, R3
2362 016124 000137 016134 JMP ER1
2363 016130 062703 032216 ERO ADD #RDATA, R3 ;ADD START OF BUFFER
2364 016134 010337 016740 ER1 MOV R3, CADER ;SAVE EXPT ADDRESS
2365 016140 020377 162442 CMP R3, @MDA ;SEE IF ADDRESS OK
2366 016144 001105 BNE ER2 ;IF NOT BR
2367 016146 017703 162432 MOV @MWC, R3 ;GET CHARACTER COUNT
2368 016152 001102 BNE ER2 ;IF NOT ZERO BR
2369 016154 005037 013706 CLR LPCPT ;CLEAR LPC PRINT FLAG
2370 016160 032777 000004 162414 BIT #4, @MTC ;A WRITE OP?
2371 016166 001045 BNE ER1B ;IF SO BR
2372 016170 032777 000020 162402 BIT #20, @MTC ;SEE IF SEVEN TRACK DRIVE
2373 016176 001041 BNE ER1B ;IF SO BR
2374 016200 013737 013700 013704 MOV EXLRC, LRCSV ;SET FOR EXPECTED LPC
2375 016206 005737 000756 TST TMFLG ;IS IT A TM?
2376 016212 001404 BEQ 15 ;IF NOT BR
2377 016214 000432 BR ER1B
    
```

2378	016216	012737	000023	013704		MOV	#23, LRCSV	.USE TM LPC
2379	016224	013704	000616		15	MOV	UDES, R4	.GET UNIT DESCRIPTION
2380	016230	042704	117777			BIC	#117777, R4	.MASK DENSITY
2381	016234	022704	060000			CMP	#60000, R4	.SEE IF9 TRK DENSITY AT 800 BPI
2382	016240	001020				BNE	ER1B	.IF NOT: BR
2383	016242	017737	162342	013702		MOV	@MTD, ACTLRC	.GET ACTUAL LPC
2384	016250	032777	020000	162374		BIT	#20000, @SWR	.SEE IF NO DATA CHECK
2385	016256	001011				BNE	ER1B	.IF NOT BR (ALLOW READ OF UNKNOWN TAPES)
2386	016260	005237	013706			INC	LRCPT	.SET LPC PRINT FLAG
2387	016264	042737	177000	013702		BIC	#177000, ACTLRC	.JUST 9 BITS
2388	016272	023737	013702	013704		CMP	ACTLRC, LRCSV	.DOES ACTUAL AGREE WITH EXPECTED?
2389	016300	001027				BNE	ER2	.IF NOT: BR
2390	016302	032777	100000	162272	ER1B	BIT	#100000, @MTC	.SEE IF HAVE ERROR
2391	016310	001002				BNE	15	.IF SO BR
2392	016312	000137	016720			JMP	EREX	
2393	016316	017737	162256	000740	15	MOV	@MTS, ERSAV	.GET STATUS
2394	016324	005737	000756			TST	TMFLG	.A TM OPERATION?
2395	016330	001404				BEQ	ER1A	.IF NOT BR
2396	016332	042737	042125	000740		BIC	#42125, ERSAV	.IGNORE TM INDICATOR AND WPL
2397	016340	001567				BEQ	EREX	.IF NO OTHER ERRORS BP
2398	016342	005737	000760		ER1A	TST	EOTREC	.IS IT EOT
2399	016346	100004				BPL	ER2	.IF NOT: BR
2400	016350	042737	032125	000740		BIC	#32125, ERSAV	.IGNORE EOT INDICATOR
2401	016356	001560				BEQ	EREX	.IF NO OTHER ERRORS BR
2402	016360	005237	000742		ER2	INC	SERFL	.SET STATUS ERROR FLAG
2403	016364	032777	002000	162260		BIT	#2000, @SWR	.SEE IF SHOULD PRINT EPROPS
2404	016372	001411				BEQ	ER3	.IF SO BR
2405	016374	005737	000774			TST	RRTYFL	.SEE IF READ RETRY
2406	016400	001404				BEQ	ER2A	.IF NOT BR
2407	016402	022737	000003	000752		CMP	#3, RTCNT	.SEE IF LAST RETRY
2408	016410	001402				BEQ	ER3	.IF SO BR
2409	016412	000137	016664		ER2A	JMP	EREXO	.ELSE EXIT
2410	016416	005237	000732		ER3	INC	PFLG	.SET PRINT FLAG
2411	016422	004737	017506			JSR	PC, PAPRT	.PRINT HEADEP
2412	016426	013704	000716		ER3A	MOV	EMADDR, R4	
2413	016432	004737	020536		ER3B	JSR	PC, TTOUT	.PRINT ERROR HEADER
2414	016436	004737	016742			JSR	PC, FRPRT	.PRINT F OR R
2415	016442	005037	000712			CLR	TEMP3	
2416	016446	012704	022756			MOV	#MSG23, R4	
2417	016452	004737	020536			JSR	PC, TTOUT	.PRINT COMMAND HEADER
2418	016456	017703	162120			MOV	@MTC, R3	
2419	016462	000303			EP7	SWAB	R3	.POSITION MOST SIGNIFICANT
2420	016464	004737	021152			JSR	PC, DOUT	.PRINT REGISTER
2421	016470	000303				SWAB	R3	.POSITION LEAST SIGNIFICANT
2422	016472	004737	021152			JSR	PC, DOUT	.PRINT REGISTER
2423	016476	005737	000712			TST	TEMP3	.SEE IF PRINTED STATUS REGISTER
2424	016502	001012				BNE	ER10	.IF SO: BR
2425	016504	005237	000712			INC	TEMP3	.SET FLAG
2426	016510	012704	023143			MOV	#MSG30, R4	
2427	016514	004737	020536			JSR	PC, TTOUT	.PRINT STATUS HEADER
2428	016520	017703	162054			MOV	@MTS, R3	.LOAD STATUS REGISTER
2429	016524	000137	016462			JMP	ER7	.GO PRINT STATUS
2430	016530	012704	023647		ER10	MOV	#MSG46, R4	
2431	016534	004737	020536			JSR	PC, TTOUT	.PRINT CHARACTER COUNT HEADEP
2432	016540	017703	162040			MOV	@MWC, R3	
2433	016544	005403				NEG	R3	.SET TO TRUE VALUE

2434	016546	004737	020724			JSR	PC, OCTP	, PRINT CHARACTER COUNT
2435	016552	012704	023654			MOV	#MSG47, R4	
2436	016556	004737	020536			JSR	PC, TTOUT	, PRINT ADDRESS HEADER
2437	016562	017703	162020			MOV	@MDA, R3	
2438	016566	004737	020724			JSR	PC, OCTP	, PRINT ADDRESS
2439	016572	012737	000255	000702		MOV	#255, TOB	
2440	016600	004737	020676			JSR	PC, TOG	, PRINT /
2441	016604	013703	016740			MOV	CADER, R3	
2442	016610	004737	020724			JSR	PC, OCTP	, PRINT EXPT ADDRESS
2443	016614	005737	013706			TST	LRCPT	, WAS LPC CHECKED?
2444	016620	001421				BEQ	EREXO	, IF NOT BR
2445	016622	012704	025253			MOV	#MSG83, R4	
2446	016626	004737	020536			JSR	PC, TTOUT	, PRINT LPC TAG
2447	016632	013703	013702			MOV	ACTLRC, R3	
2448	016636	004737	020724			JSR	PC, OCTP	, PRINT ACTUAL LPC
2449	016642	012737	000255	000702		MOV	#255, TOB	
2450	016650	004737	020676			JSR	PC, TOG	, PRINT -
2451	016654	013703	013704			MOV	LRCSV, R3	
2452	016660	004737	020724			JSR	PC, OCTP	, PRINT EXPECTED LPC
2453	016664	032777	100000	161760	EREXO	BIT	#100000, @SWR	, SEE IF STOP ON ERROR
2454	016672	001412				BEQ	EREX	, IF NOT BR
2455	016674	000000				HALT		
2456	016676	005737	000732			TST	PFLG	, SEE IF PRINT
2457	016702	001006				BNE	EREX	, IF SO: BR
2458	016704	032777	002000	161740		BIT	#2000, @SWR	, SEE IF SHOULD PRINT
2459	016712	001002				BNE	EREX	, IF NOT BR
2460	016714	000137	016416			JMP	ER3	, PRINT ERROR
2461	016720	004737	022126		EREX	JSR	PC, CKSWR	, GO TEST FOR G
2462	016724	005037	000732			CLR	PFLG	, CLEAR FLAG
2463	016730	017737	161644	000740		MOV	@MTS, ERSV	, SAVE STATUS REGISTER
2464	016736	000207				RTS	PC	, RETURN
2465	016740	000000			CADER	D		, EXPT ADDRESS SAVE LOCATION
2466								
2467								, *****
2468								, F FOR FORWARD/R FOR REVERSE PRINT SUBROUTINE
2469								
2470								, THIS SUBROUTINE IS USED TO PRINT OUT THE
2471								, TAPE DIRECTION USED WHEN ANY ERROR IS
2472								, DETECTED IN STATUS OF READ OR WRITE, DATA, OR
2473								, SPACING OPERATIONS.
2474								, *****
2475								
2476	016742	032777	000004	161632	FRPRT	BIT	#4, @MTC	, SEE IF WRITE COMMAND
2477	016750	001015				BNE	FREX	, IF SO BR
2478	016752	032737	010000	000626		BIT	#10000, RDCMD	, SEE IF READ REVERSE
2479	016760	001405				BEQ	FRO	, IF NOT BR
2480	016762	012704	022572			MOV	#MSG17, R4	
2481	016766	004737	020536			JSR	PC, TTOUT	, PRINT R
2482	016772	000404				BR	FREX	
2483	016774	012704	022564		FRO	MOV	#MSG16, R4	
2484	017000	004737	020536			JSR	PC, TTOUT	, PRINT F
2485	017004	000207			FREX	RTS	PC	, EXIT

```

2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515 017006 005037 000706          TAPG  CLR      TEMP1
2516 017012 013704 000602          MOV      MTC, R4          ;GET CMD REGISTER ADDRESS
2517 017016 005204          INC      R4              ;BUMP TO HIGH BYTE
2518 017020 113714 000617          MOV      UDES+1, (R4)    ;LOAD UNIT DESCRIPTION
2519 017024 032777 000200 161550 TAPGO  BIT      #200, @MTC      ;SEE IF HAVE READY
2520 017032 001035          BNE     TAPG3           ;IF SO. BR
2521 017034 005237 000706          INC      TEMP1          ;SEE IF TIMED OUT
2522 017040 001371          BNE     TAPGO          ;WAIT FOR READY
2523 017042 004737 017506          JSR     PC, PAPRT      ;PRINT CYCLE NUMBER
2524 017046 032777 000004 161526          BIT      #4, @MTC       ;SEE IF WRITE OP
2525 017054 001405          BEQ     TAPG1          ;IF NOT. BR
2526 017056 012704 022460          MOV      #MSG5, R4
2527 017062 004737 020536          JSR     PC, TTOUT      ;PRINT WRITE ERP
2528 017066 000406          BR      TAPG2
2529 017070 012704 022465          TAPG1  MOV      #MSG6, R4
2530 017074 004737 020536          JSR     PC, TTOUT      ;PRINT READ ERR
2531 017100 004737 016742          JSR     PC, FRPRT      ;PRINT F OR R
2532 017104 012704 023015          TAPG2  MOV      #MSG25, R4
2533 017110 004737 020536          JSR     PC, TTOUT      ;PRINT NO READY ERP
2534 017114 000000          TAPG2A HALT
2535 017116 005037 000734          CLR     UNP            ;RESET UNIT POINTER
2536 017122 000137 003264          JMP     STAUTO         ;RESTART
2537 017126 000240          TAPG3  NOP
2538 017130 000240          NOP
2539 017132 005037 000706          CLR     TEMP1          ;SET DELAY
2540 017136 032777 000100 161434 15          BIT     #100, @MTS     ;SEE IF SELP
2541 017144 001013          BNE     25             F SO BR
    
```



```

2589
2590      , TTY INTERRUPT HANDLER*****
2591
2592 017404 012777 000340 161236 TTINT MOV    #340, @PSW      , RESET PSW
2593 017412 005077 161240          CLR    @TKS        , CLEAR TTY STATUS
2594 017416 122777 000203 161234 CMPB   #203, @TKB   , SEE IF CONT C
2595 017424 001404          BEQ    TTINTO      , IF SO BR
2596 017426 004737 022126          JSR    PC, CKSWR   , GC CHECK FOR G
2597 017432 000240          NOP
2598 017434 000002          RTI        , ELSE RETURN
2599 017436 010037 000712          TTINTO MOV   R0, TEMP3   , SAVE R0(REC CNTR)
2600 017442 004737 012164          JSR    PC, TINP4   , GO GET STALL VALUES
2601 017446 013700 000712          MOV   TEMP3, R0    , RESTORE R0(REC CNTR)
2602 017452 005077 161202          CLR    @TKB        , CLEAR TTY BUFFER
2603 017456 012777 000100 161172          MOV   #100, @TKS  , RESET INTERRUPT ENABLE
2604 017464 000002          RTI        , RETURN
2605
2606      , MAG TAPE INTERRUPT HANDLER*****
2607
2608 017466 022626          MTINT CMP   (SP)+, (SP)+ , RESET STACK POINTER
2609 017470 042777 000100 161104          BIC   #100, @MTC  , RESET INTERRUPT ENABLE
2610 017476 000240          NOP
2611 017500 000240          NOP
2612 017502 000177 161216          JMP   @RTRN      , RETURN
2613
2614      , *****
2615      , ERROR HEADER PRINT SUBROUTINE
2616      ,
2617      , THIS ROUTINE IS USED TO PRINT OUT A HEADER
2618      , WITH EACH ERROR MESSAGE THE PRINT IS IN TWO
2619      , LINES AND CONTAINS THE FOLLOWING INFORMATION
2620      , LINE 1 UNIT NUMBER, DATA PATTERN NUMBER
2621      , LINE 2. CURRENT BLOCK NUMBER, RECORD NUMBER IN
2622      , WHICH THE ERROR OCCURED PLUS THE TOTAL NUMBER
2623      , OF RECORDS IN THIS BLOCK, THE RECORD SIZE (NUMBER
2624      , OF CHARACTERS), AND THE ERROR TYPE (READ, WRITE, SPACE, ETC)
2625      , PLUS THE TAPE DIRECTION (FORWARD OR REVERSE)
2626      , ALL NUMBERS ARE IN OCTAL
2627      , *****
2628
2629 017506 012704 022504          PAPRT MOV   #MSG11, R4
2630 017512 004737 020536          JSR   PC, TTOUT   , PRINT UNIT HEADER
2631 017516 013703 000616          MOV   UDES, P3
2632 017522 000303          SWAB  R3
2633 017524 042703 177770          BIC   #177770, R3
2634 017530 004737 020724          JSR   PC, OCTP    , PRINT UNIT NUMBER
2635 017534 012704 025564          MOV   #MSG90, R4
2636 017540 004737 020536          JSR   PC, TTOUT   , PRINT DENSITY TAG
2637 017544 005003          CLR   R3
2638 017546 032737 020000 000616          BIT   #20000, UDES , SEE IF BIT 1 OF DENSITY=1
2639 017554 001401          BEQ   1$         , IF NOT BR
2640 017556 005203          INC   R3         , ELSE SET BIT 1
2641 017560 032737 040000 000616 1$          BIT   #40000, UDES , SEE IF BIT 2 OF DENSITY=1
2642 017566 001402          BEQ   2$         , IF NOT BR
2643 017570 052703 000002          BIS   #2, R3     , ELSE SET BIT 2
2644 017574 004737 020724          JSR   PC, OCTP    , PRINT DENSITY SETTING
    
```

2645	017600	012704	025572		MOV	#MSG91,R4	
2646	017604	004737	020536		JSR	PC,TTOUT	,PRINT PARITY TAG
2647	017610	005003			CLR	R3	
2648	017612	032737	004000	000616	BIT	#4000,UDES	,SEE IF EVEN PARITY
2649	017620	001401			BEQ	3\$,IF NOT BR
2650	017622	005203			INC	R3	,ELSE SET TO A ONE
2651	017624	004737	020724	3\$	JSR	PC,OCTP	,PRINT PARITY
2652	017630	012704	025327		MOV	#MSG86,R4	
2653	017634	004737	020536		JSR	PC,TTOUT	,PRINT PATTRN TAG
2654	017640	032777	000400	161004	BIT	#400,DSWR	,SEE IF RANDOM DATA
2655	017646	001406			BEQ	PAPRTB	,IF NOT BR
2656	017650	012737	000122	000702	PAPRTA	MOV	#122,TOB
2657	017656	004737	020676		JSR	PC,TOG	,PRINT R
2658	017662	000412			BR	PAPRTD	
2659	017664	005737	021632		PAPRTB	TST	,SEE IF AUTO SEQ
2660	017670	001403			BEQ	PAPRTC	,IF NOT BR
2661	017672	005737	000624		TST	PATRN	,SEE IF AUTO RANDOM
2662	017676	100764			BMI	PAPRTA	,IF SO BR
2663	017700	013703	000624		PAPRTC	MOV	PATRN,R3
2664	017704	004737	020724		JSR	PC,OCTP	,PRINT PATTRN NUMBER
2665	017710	012704	022521		PAPRTD	MOV	#MSG13,R4
2666	017714	004737	020536		JSR	PC,TTOUT	,PRINT BLOCK NO HEADER
2667	017720	013703	000720		MOV	BLCNTR,R3	
2668	017724	004737	020724		JSR	PC,OCTP	,PRINT NUMBER
2669	017730	012704	022527		MOV	#MSG14,R4	
2670	017734	004737	020536		JSR	PC,TTOUT	,PRINT REC NO HEADER
2671	017740	010003			MOV	RO,R3	
2672	017742	032777	000004	160632	BIT	#4,DMTC	,SEE IF WRITE OPEATION
2673	017750	001000			BNE	PAPRT1	,IF SO BR
2674	017752	013703	000620		PAPRT1	MOV	RCNT,R3
2675	017756	160003			SUB	RO,R3	,GET RECORD NUMBER
2676	017760	005203			INC	R3	
2677	017762	004737	020724		PAPRT2	JSR	PC,OCTP
2678	017766	012737	000055	000702	MOV	#55,TOB	,PRINT RECORD NUMBER
2679	017774	004737	020676		JSR	PC,TOG	,LOAD DASH (-)
2680	020000	013703	000620		MOV	RCNT,R3	,PRINT DASH (-)
2681	020004	004737	020724		JSR	PC,OCTP	,PRINT RECORD COUNT
2682	020010	012704	022472		MOV	#MSG7,R4	
2683	020014	004737	020536		JSR	PC,TTOUT	,PRINT RECORD SIZE HEADER
2684	020020	013703	000622		MOV	CARCNT,R3	,GET CHARACTER COUNT
2685	020024	005303			DEC	R3	
2686	020026	005103			COM	R3	,REMOVE TWOS COMPLEMENT
2687	020030	004737	020724		JSR	PC,OCTP	,PRINT RECORD SIZE
2688	020034	012737	000001	000726	MOV	#1,HDRFL	,SET HEADER FLAG
2689	020042	004737	022126		JSR	PC,CKSWR	,TEST FOR G
2690	020046	000207			PTS	PC	RETURN
2691							

```

2692
2693
2694
2695 020050 000240 DRPDRV NOP
2696 020052 012777 010000 160522 MOV #10000, @MTC ,POWER CLEAR CONTROLLER
2697 020060 012704 025600 MOV #MSG92, R4
2698 020064 004737 020536 JSR PC, TTOUT ;PRINT UNIT DROPPED
2699 020070 013703 000616 MOV UDES, R3 ;GET UNIT DESCRIPTION
2700 020074 000303 SWAB R3
2701 020076 042703 177770 BIC #177770, R3 ;MASK UNIT NUMBER
2702 020102 004737 020724 JSR PC, OCTP ;PRINT DROPPED UNIT NUMBER
2703 020106 012704 025624 MOV #MSG93, R4
2704 020112 004737 020536 JSR PC, TTOUT ;PRINT REST OF MSG
2705 020116 013700 000734 MOV UNP, R0 ;SET UNIT POINTER
2706 020122 052760 100200 001012 BIS #100200, UN1(R0) ;SET DROPPED FLAG
2707 020130 005337 004716 DEC REOTC ;DECREMENT EOT UNIT COUNTER
2708 020134 004737 015370 JSR PC, PRSTAT ;PRINT CURRENT STATS
2709 020140 005237 001006 INC DUCTR ;BUMP DROPPED UNIT COUNTER
2710 020144 123737 001006 004717 CMPB DUCTR, REOTC+1 ;SEE IF DROPPED ALL UNITS
2711 020152 103406 BLO 1$ ;IF NOT BR
2712 020154 012704 026053 MOV #MSG96, R4
2713 020160 004737 020536 JSR PC, TTOUT ;PRINT ALL DROPPED STOP
2714 020164 000137 004640 JMP REOT9 ;GO TO END ROUTINE
2715 020170 000240 1$ NOP
2716 020172 005000 CLR R0
2717 020174 032760 100200 001012 2$ BIT #100200, UN1(R0) ;SEE IF ANY DRIVES LEFT IN THIS PASS
2718 020202 001414 BEQ 3$ ;IF SO BR
2719 020204 062700 000002 ADD #2, R0 ;BUMP POINTER
2720 020210 022760 177777 001012 CMP #-1, UN1(R0) ;SEE IF LAST ENTRY
2721 020216 001366 BNE 2$ ;IF NOT BR
2722 020220 012704 025716 MOV #MSG94, R4
2723 020224 004737 020536 JSR PC, TTOUT ;PRINT NO MORE UNITS
2724 020230 000137 004622 JMP REOT8 ;GO TO END OF PASS ROUTINE
2725 020234 000137 004072 3$ JMP START7 ;GO TO NEXT UNIT
2726
2727 ,*****
2728 ,RANDOM NUMBER GENERATOR SUBROUTINE
2729
2730 THIS SUBROUTINE IS USED TO GENERATE THE RANDOM
2731 NUMBERS REQUIRED FOR USE AS PANDOM DATA,
2732 RECORD COUNT, AND CHARACTER COUNT
2733 ,*****
2734
2735 020240 063737 000676 000672 RANG. ADD RANSV, RANBAS
2736 020246 063737 000672 000676 ADD RANBAS, RANSV ;GET NEW NUMBER
2737 020254 023701 000676 CMP RANSV, R1 ;SEE IF NUMBER TOO BIG
2738 020260 101367 BHI RANG ;IF SO BR
2739 020262 020237 000676 CMP R2, RANSV ;SEE IF NUMBER TOO SMALL
2740 020266 101364 BHI PANG ;IF SO BR
2741 020270 000207 RTS PC ;EXIT
2742
    
```

2743
 2744
 2745
 2746
 2747
 2748
 2749
 2750
 2751
 2752
 2753
 2754
 2755
 2756
 2757
 2758
 2759
 2760
 2761
 2762
 2763
 2764
 2765
 2766
 2767
 2768
 2769
 2770
 2771
 2772
 2773
 2774
 2775
 2776
 2777
 2778
 2779
 2780
 2781
 2782
 2783
 2784
 2785
 2786
 2787
 2788
 2789
 2790
 2791
 2792
 2793
 2794
 2795

 .TTY ENTRY SUBROUTINE
 .
 .THIS SUBROUTINE IS USED BY THE TEST CONDITION
 .ENTRY ROUTINE TO READ THE RESPONSE ENTERED
 .AT THE TTY AND CHECK THEM FOR LEGALITY AND
 .LIMITS ALL RESPONSE MUST BE TYPED IN OCTAL
 .(0-7) AND MUST FALL WITHIN THE LIMITS SET BY
 .THE CALLING ROUTINE
 .IF AN ENTRY IS ILLEGAL OR OUTSIDE THE LIMITS,
 .A QUESTION MARK IS TYPED (?) AND THE RESPONSE
 .MAY BE REENTERED
 .ENTRIES MAY NOT EXCEED SIX (6) CHARACTERS AND
 .MAY BE TERMINATED AT LESS THAN SIX BY TYPING A
 .CARRIAGE RETURN

```

2761 020272 005037 000706      TTR   CLR   TEMP1      .CLEAR FIRST CHARACTER FLAG
2762 020276 005000              CLR   RO
2763 020300 004737 020464      TTR0  JSR   PC,TTIN   .GO READ CHARACTER
2764 020304 042737 177600 000704  BIC   #177600,T1B   .STRIP GARBAGE
2765 020312 122737 000015 000704  CMPB  #15,T1B       .SEE IF CR
2766 020320 001005              BNE   TTR1         .IF NOT BR
2767 020322 005737 000706      TST   TEMP1       .SEE IF FIRST CHARACTER
2768 020326 001446              BEQ   TTR5         .IF SO BR
2769 020330 000137 020422      JMP   TTR2         .ELSE GO LOAD VALUE
2770 020334 122737 000060 000704  TTR1  CMPB  #60,T1B    .SEE IF CHAR IS LESS THAN 0
2771 020342 101402              BLOS  TTR1A        .IF NOT BR
2772 020344 000137 020446      JMP   T1NER        .ELSE GO TO ERPOP
2773 020350 122737 000070 000704  TTR1A CMPB  #70,T1B    .SEE IF CHAR IS GREATER THAN 7
2774 020356 101002              BHI   TTR1B        .IF NOT BR
2775 020360 000137 020446      JMP   T1NER        .ELSE GO TO EPROP
2776 020364 005237 000706      TTR1B INC   TEMP1     .SET FIRST CHARACTER FLAG
2777 020370 000241              CLC
2778 020372 006100              ROL   RO
2779 020374 000241              CLC
2780 020376 006100              ROL   RO           .SHIFT 3 LEFT
2781 020400 000241              CLC
2782 020402 006100              ROL   RO
2783 020404 042737 177770 000704  BIC   #177770,T1B  .STRIP ASCII
2784 020412 053700 000704      BIS   T1B,RO       .LOAD CHARACTER
2785 020416 005301              DEC   R1           .SEE IF DONE
2786 020420 001327              BNE   TTP0         .IF NOT BR
2787 020422 020002              TTP2  CMP   RO,R2     .SEE IF EXCEEDED MAXIMUM LIMIT
2788 020424 101402              BLOS  TTR3         .IF NOT BR
2789 020426 000137 020446      JMP   T1NER        .ELSE GO TO ERROR
2790 020432 020300              TTP3  CMP   R3,RO   .SEE IF BELOW MINIMUM LIMIT
2791 020434 101402              BLOS  TTR4         .IF NOT BR
2792 020436 000137 020446      JMP   T1NER        .ELSE GO TO ERROR
2793 020442 010015              TTP4  MOV   RO,(R5) .LOAD VALUE
2794 020444 000207              TTP5  RTS   PC      .EXIT
    
```

2796
 2797
 2798
 2799
 2800
 2801
 2802
 2803
 2804
 2805
 2806
 2807
 2808
 2809
 2810
 2811
 2812
 2813
 2814
 2815
 2816
 2817
 2818
 2819
 2820
 2821
 2822
 2823
 2824
 2825
 2826
 2827
 2828
 2829
 2830
 2831
 2832
 2833
 2834
 2835
 2836
 2837
 2838
 2839
 2840
 2841
 2842
 2843
 2844
 2845
 2846
 2847
 2848
 2849
 2850
 2851

020446 012704 023602
 020452 004737 020536
 020456 162716 000020
 020462 000207

 020464 005077 160166
 020470 005077 160164
 020474 005037 000704
 020500 005277 160152
 020504 105777 160146
 020510 100375
 020512 017737 160142 000704
 020520 105777 160136
 020524 100375
 020526 113777 000704 160130
 020534 000207

 020536 112437 000702
 020542 122737 000043 000702
 020550 001460
 020552 122737 000045 000702
 020560 001407
 020562 122737 000041 000702
 020570 001434
 020572 004737 020676
 020576 000757
 020600 112737 000015 000702 TCRLF
 020606 004737 020676
 020612 012703 000004
 020616 005037 000702 TCRLFA
 020622 004737 020676
 020626 005303
 020630 001372
 020632 112737 000012 000702
 020640 004737 020676
 020644 105737 001004
 020650 100401
 020652 000731
 020654 005037 001004 15
 020660 000414
 020662 112737 000007 000702 TBELL
 020670 004737 020676
 020674 000720
 020676 105777 157760 TOG
 020702 100375
 020704 113777 000702 157752
 020712 000207

.TTY ENTRY ERROR SUBROUTINE*****

```
TINER  MOV  #MSG43,R4
        JSR  PC,TTOUT      ,PRINT?
        SUB  #20,(SP)      ,RESET SP TO START OF VALUE ROUTINE
        RTS  PC           ,REDO VALUE ENTRY
```

.TTY READ SUBROUTINE*****

```
TTIN   CLR  @TKS
        CLR  @TKB
        CLR  TIB
        INC  @TKS
TTIN1  TSTB @TKS
        BPL TTIN1
        MOV  @TKB,TIB
TTIN2  TSTB @TPS
        BPL TTIN2
        MOVB TIB,@TPB
        RTS  PC
```

.TTY OUTPUT SUBROUTINE*****

```
TTOUT  MOVB (R4)+,TOB
        CMPB #43,TOB
        BEQ  TEX
        CMPB #45,TOB
        BEQ  TCRLF
        CMPB #41,TOB
        BEQ  TBELL
        JSR  PC,TOG
        BR   TTOUT
TCRLF  MOVB #15,TOB
        JSR  PC,TOG
        MOV  #4,R3
TCRLFA CLR  TOB
        JSR  PC,TOG
        DEC  R3
        BNE TCRLFA      ,DO F LLEPC
        MOVB #12,TOB
        JSR  PC,TOG
        TSTB RDSW
        BMI  15
        BR   TTOUT
        CLR  RDSW
        BR   TEX
TBELL  MOVB #7,TOB
        JSR  PC,TOG
        BR   TTOUT
TOG    TSTB @TPS
        BPL TOG
        MOVB TOB,@TPB
        RTS  PC
TEX    RTS
```

```

2852          ,OCTAL OUTPUT SUBROUTINE*****
2853
2854 020714 012737 000001 021150 OCTPE MOV #1, OFL
2855 020722 000402          BR OCTPE1
2856 020724 005037 021150 OCTP CLR OFL          , CLEAR FLAG FOR LEADING ZERO
2857 020730 010304          OCTPE1 MOV R3, R4
2858 020732 001007          BNE OCTPD          , IF NOT ZERO BR
2859 020734 005737 021150 TST OFL
2860 020740 001004          BNE OCTPD
2861 020742 004737 021130 JSR PC, OCTPG1          , ELSE PRINT ZERO
2862 020746 000137 021072 JMP OCTP3          , SPACE AND EXIT
2863 020752 032704 100000 OCTPD BIT #100000, R4          , SEE IF MSD = 1
2864 020756 001406          BEQ OCTP1          , IF NOT BR
2865 020760 012704 000001 MOV #1, R4
2866 020764 004737 021106 JSR PC, OCTPG          , PRINT 1
2867 020770 000137 021002 JMP OCTP2
2868 020774 005004          OCTP1 CLR R4
2869 020776 004737 021106 JSR PC, OCTPG          , PRINT 0
2870 021002 010304          OCTP2 MOV R3, R4
2871 021004 006004          ROR R4
2872 021006 006004          ROR R4
2873 021010 006004          ROR R4          , POSITION DIGIT
2874 021012 006004          ROR R4
2875 021014 000304          SWAB R4
2876 021016 004737 021106 JSR PC, OCTPG          , PRINT DIGIT 2
2877 021022 010304          MOV R3, R4
2878 021024 006004          ROR R4
2879 021026 000304          SWAB R4
2880 021030 004737 021106 JSR PC, OCTPG          , PRINT DIGIT 3
2881 021034 010304          MOV R3, R4
2882 021036 006104          ROL R4
2883 021040 006104          ROL R4
2884 021042 000304          SWAB R4
2885 021044 004737 021106 JSR PC, OCTPG          , PRINT DIGIT 4
2886 021050 010304          MOV R3, R4
2887 021052 006004          ROR R4
2888 021054 006004          ROR R4
2889 021056 006004          ROR R4
2890 021060 004737 021106 JSR PC, OCTPG
2891 021064 010304          MOV R3, R4
2892 021066 004737 021106 JSR PC, OCTPG          , PRINT DIGIT 5
2893 021072 012737 000240 000702 OCTP3 MOV #240, TOB
2894 021100 004737 020676 JSR PC, TOG          , PRINT SPACE
2895 021104 000207          RTS PC          , EXIT
2896 021106 042704 177770 OCTPG BIC #177770, R4
2897 021112 001004          BNE OCTPG0
2898 021114 005737 021150 TST OFL
2899 021120 001001          BNE OCTPG0
2900 021122 000207          RTS PC
2901 021124 005237 021150 OCTPG0 INC OFL
2902 021130 052704 000260 OCTPG1 BIS #260, R4
2903 021134 010437 000702 MOV R4, TOB
2904 021140 004737 020676 JSR PC, TOG
2905 021144 010304          MOV R3, R4
2906 021146 000207          RTS PC
2907 021150 000000          OFL 0          , FIRST CHAR FLAG
    
```

```
2908  
2909  
2910          .DATA CHARACTER OUTPUT SUBROUTINE*****  
2911  
2912 021152 005037 000702          DOUT  CLR      TOB  
2913 021156 012704 000010          MOV      #10, R4          .SET NUMBER TO PRINT  
2914 021162 110337 000702          MOVB     R3, TOB  
2915 021166 105777 157470          DOUT1   TSTB     @TPS  
2916 021172 100375          BPL      DOUT1  
2917 021174 132737 000200 000702   BITB     #200, TOB  
2918 021202 001404          BEQ      DOUT2  
2919 021204 012777 000061 157452   MOV      #061, @TPB  
2920 021212 000403          BR       DOUT3  
2921 021214 012777 000060 157442   DOUT2.  MOV      #060, @TPB  
2922 021222 006137 000702   DOUT3.  ROL      TOB  
2923 021226 005304          DEC      R4  
2924 021230 001356          BNE     DOUT1  
2925 021232 000207          RTS      PC  
2926  
2927          .ASSURE VALID STATUS DELAY SUBROUTINE*****  
2928  
2929 021234 005777 157352          STDLY   TST      @MTRD  
2930 021240 100775          BMI     STDLY          .AWAIT TIMER = 0  
2931 021242 005777 157344          1$     TST      @MTRD  
2932 021246 100375          BPL     1$          .AWAIT TIMER = 1  
2933 021250 005777 157336          2$     TST      @MTRD  
2934 021254 100775          BMI     2$          .AWAIT TIMER = 0  
2935 021256 000207          PTS     PC          .EXIT
```

```

, AUTO SEQUENCE TEST ROUTINE*****
2936
2937
2938 021260 012704 025042      ASEQ  MOV  #MSG78,R4
2939 021264 004737 020536      JSR  PC,TTOUT      ;PRINT CONT REQUEST
2940 021270 013703 021634      MOV  ASEQCF,R3
2941 021274 004737 020724      JSR  PC,OCTP       ;PRINT CURRENT VALUE
2942 021300 012705 021634      MOV  #ASEQCF,R5    ;SET ENTRY ADDRESS
2943 021304 012701 000001      MOV  #1,R1         ;SET SIZE OF ENTRY
2944 021310 012702 000001      MOV  #1,R2         ;SET UPPER LIMIT
2945 021314 005003              CLR  R3            ;SET LOWER LIMIT
2946 021316 004737 020272      JSR  PC,TTR        ;GET INPUT
2947
2948 021322 004737 021636      ASEQ0 JSR  PC,HRDS     ;SELECT HARDWARE CONFIGURATION
2949 021326 012704 025057      MOV  #MSG79,R4
2950 021332 004737 020536      JSR  PC,TTOUT     ;PRINT DIVIDER
2951 021336 012704 025125      MOV  #MSG80,R4
2952 021342 004737 020536      JSR  PC,TTOUT     ;PRINT UNITS NUMBER MESSG
2953 021346 012700 001012      MOV  #UN1,R0      ;POINT TOP OF DRIVE TABLE
2954 021352 005710              ASEQ2 TST  (R0)     ;SEE IF END
2955 021354 100424              BMI  AMOD1        ;IF SO BR
2956 021356 011037 000706      MOV  (R0),TEMP1   ;GET UNIT DESCRIPTION
2957 021362 113703 000707      MOVB TEMP1+1,R3   ;POSITION AND
2958 021366 042703 177770      BIC  #177770,R3   ; MASK UNIT NUMBER
2959 021372 004737 020724      JSR  PC,OCTP      ;PRINT DRIVE TABLE
2960 021376 012704 023732      MOV  #MSG51,R4    ;PRESET FOR 9 TRK MSG
2961 021402 032710 020000      BIT  #20000,(R0)  ;SEC IF 7 TRK
2962 021406 001002              BNE  15           ;IF NOT BR
2963 021410 012704 023723      MOV  #MSG50,R4    ;SET TO 7 TRK MSG
2964 021414 004737 020536      15. JSR  PC,TTOUT   ;PRINT TRK MSG
2965 021420 062700 000002      ADD  #2,R0        ;BUMP POINTER
2966 021424 000752              BR   ASEQ2        ;DO ALL
2967 021426 005037 000720      AMOD1 CLR  BLCNTR
2968
2969 021432 004737 004732      AMOD1B JSR  PC,RWINDA    ;GO REWIND ALL DRIVES
2970 021436 012737 000006 021630  MOV  #6,ABLCNT    ;SET NUMBER OF BLOCKS
2971 021444 012737 174000 000622  MOV  #-4000,CARCNT ;SET RECORD SIZE
2972 021452 012737 000100 000620  MOV  #100,RCNT    ;SET RECORD COUNT
2973 021460 012737 000003 000624  MOV  #3,PATRN     ;SELECT PATTERN 3
2974 021466 005037 000646      CLR  TMEX         ;ASSURE NO TM
2975 021472 004737 003264      JSR  PC,STAUTO    ;GO DO THIS PATTERN
2976 021476 012737 000007 000624  MOV  #7,PATRN     ;SELECT PATTERN 7
2977 021504 004737 003264      JSR  PC,STAUTO    ;GO DO THIS PATTERN
2978 021510 012737 000011 000624  MOV  #11,PATRN    ;SELECT PATTERN 11
2979 021516 004737 003264      JSR  PC,STAUTO    ;GO DO THIS PATTERN
2980 021522 012737 177777 021630  MOV  #-1,ABLCNT   ;FORCE TO END OF TAPE
2981 021530 012737 177777 000624  MOV  #-1,PATRN    ;SELECT AUTO RANDOM DATA
2982 021536 012737 152634 000672  MOV  #152634,RANBAS ;RESET RANDOM DATA BASE
2983 021544 012737 032561 000676  MOV  #32561,RANSAV ;GO DO RANDOM
2984 021552 004737 003264      JSR  PC,STAUTO
2985 021556 012704 025057      MOV  #MSG79,R4
2986 021562 004737 020536      JSR  PC,TTOUT     ;PRINT DIVIDER
2987 021566 012704 025151      ASEQX MOV  #MSG81,R4
2988 021572 004737 020536      JSR  PC,TTOUT
2989 021576 005737 021634      TST  ASEQCF       ;SEE IF CONTINUOUS AUTO SEQ
2990 021602 001003              BNE  ASEQXX       ;IF SO BR
2991 021604 000000              HALT
    
```

```

2992 021606 004737 022126          JSR    PC,CKSWR      ,TEST FOR G
2993 021612 005237 000776      ASEQXX INC    SEQCT      ,BUMP PASS COUNT
2994 021616 013703 000776          MOV    SEQCT,R3
2995 021622 004737 020724          JSR    PC,OCTP      ,PRINT PASS COUNT
2996 021626 000635          BR     ASEQO
2997 021630 000000          ABLCNT O
2998 021632 000000          ASEQF  O
2999 021634 000000          ASEQCF O
3000
3001          ,SUBROUTINE TO SELECT AUTO SEQ HARDWARE*****
3002
3003 021636 005003          HRDS  CLR    R3          ,CLEAR TABLE POINTER
3004 021640 005037 000706          CLR    TEMP1        ,CLEAR UNIT DESCRIPTION HOLDER
3005 021644 005037 000712          CLR    TEMP3        ,UNIT COUNT
3006 021650 005037 004716          CLR    REOTC        ,CLEAR EOT COUNTER
3007 021654 005037 000710          CLR    TEMP2        ,CLEAR UNIT INCREMENT
3008 021660 012777 010000 156714      MOV    #10000,@MTC   ,POWER CLEAR CONTROLLER
3009 021666 113737 000710 000707      HRDS1 MOVB  TEMP2,TEMP1+1 ,POSITION UNIT NUMBER
3010 021674 013777 000706 156700      MOV    TEMP1,@MTC   ,SELECT DRIVE
3011 021702 004737 021234          JSR    PC,STOLY     ,GO ASSURE VALID STATUS
3012 021706 032777 000001 156664      BIT    #1,@MTC      ,SEE IF AVAIL
3013 021714 001421          BEQ    HRDS2        ,IF NOT: BR
3014 021716 052737 060000 000706      BIS    #60000,TEMP1 ,SET DENSITY AND PARITY
3015 021724 032777 000020 156646      BIT    #20,@MTC    ,SEE IF 7 TRK
3016 021732 001403          BEQ    1$          ,IF NOT BR
3017 021734 042737 020000 000706      BIC    #20000,TEMP1 ,ELSE SET TO 7 TRK NORMAL DENSITY
3018 021742 013763 000706 001012 1$      MOV    TEMP1,UN1(R3) ,PUT IN TABLE
3019 021750 062703 000002          ADD    #2,R3
3020 021754 005237 000712          INC    TEMP3        ,INCREMENT COUNT
3021
3022 021760 005237 000710          HRDS2 INC    TEMP2        ,SET FOR NEXT UNIT
3023 021764 022737 000010 000710      CMP    #10,TEMP2   ,DONE?
3024 021772 001335          BNE    HRDS1        ,IF NOT BR
3025 021774 005703          TST    R3          ,FOUND A UNIT?
3026 021776 001007          BNE    HRDSX        ,IF SO BR
3027 022000 012704 025177          MOV    #MSG82,R4
3028 022004 004737 020536          JSR    PC,TTOUT     ,TYPE NO UNIT AVAILABLE
3029 022010 000000          HALT
3030 022012 000137 003106          JMP    STAUT        ,START AUTO SEQ AGAIN
3031 022016 012763 177777 001012      HRDSX MOV    #-1,UN1(R3) ,MARK END OF TABLE
3032 022024 013737 000712 004716      MOV    TEMP3,REOTC ,SET NUMBER OF UNITS
3033 022032 000337 000712          SWAB  TEMP3
3034 022036 053737 000712 004716      BIS    TEMP3,REOTC ,SET EOT CNTR
3035 022044 000207          RTS    PC          ,RETURN
3036
3037
3038 022046 013746 000006          SUSWR MOV    @#6,-(SP)    ,SAVE VECTORS
3039 022052 013746 000004          MOV    @#4,-(SP)
3040 022056 012737 022076 000004      MOV    #15,@#4     ,SET UP FOR TIMEOUT
3041 022064 022777 177777 156560      CMP    #-1,@SWR    ,REFERENCE HARDWARE SWITCH REGISTER
3042 022072 001402          BEQ    2$
3043 022074 000407          BR     3$
3044 022076 022626          1$      CMP    (SP)+,(SP)+ ,ADJUST STACK
3045 022100 012737 000176 000652 2$      MOV    #SWREG,SWR  ,POINT TO SOFTWARE SWITCH REG
3046 022106 012737 000174 000654      MOV    #DISPREG,DISPLAY ,POINT TO SOFT DISPLAY REG
3047 022114 012637 000004          3$      MOV    (SP)+,@#4   ,RESTORE VECTORS
  
```

3048	022120	012637	000006		MOV	(SP)+, @#6	
3049	022124	000207			RTS	PC	, RETURN
3050							
3051	022126	022737	000176	000652	CKSWR	CMP	#SWREG, SWR
3052	022134	001036			BNE	OUT	, SOFTWARE SWITCH REG PRESENT
3053	022136	017737	156516	000704	MOV	@TKB, TIB	, NO, GET OUT
3054	022144	042737	177600	000704	BIC	#177600, TIB	, AND STRIP OFF
3055	022152	022737	000007	000704	CMP	#7, TIB	, THE GARBAGE
3056	022160	001024			BNE	OUT	, IS IT A < G >
3057	022162	012704	026160		MOV	#SCNTG, R4	
3058	022166	004737	020536		JSR	PC, TTOUT	
3059	022172	012704	026164		MOV	#MSWR, R4	
3060	022176	004737	020536		JSR	PC, TTOUT	
3061	022202	017703	156444		MOV	@SWR, R3	
3062	022206	004737	020714		JSR	PC, OCTPE	
3063	022212	012704	026174		MOV	#MNEW, R4	
3064	022216	004737	020536		JSR	PC, TTOUT	
3065	022222	005037	001002		CLR	@TEMPST	
3066	022226	004737	022234		JSR	PC, \$READ	, GO READ A LINE
3067	022232	000207			OUT	RTS	PC
3068							, RETURN TO MAIN BODY OF PROGRAM
3069	022234	005037	001002		\$READ	CLR	TEMPST
3070	022240	012737	000007	001000	MOV	#7, COUNT	
3071	022246	004737	020464		15	JSR	PC TTIN
3072	022252	042737	177600	000704	BIC	#177600, TIB	, GO READ A CHARACTER
3073	022260	122737	000025	000704	CMPB	#25, TIB	, STRIP OFF GARBAGE
3074	022266	001002			BNE	25	, IS IT A U?
3075	022270	005726			35	TST	(SP)+
3076	022272	000737			BR	CNTLU	, POP THE STACK
3077	022274	122737	000015	000704	25	CMPB	#15, TIB
3078	022302	001013			BNE	45	, START OVER
3079	022304	012737	000200	001004	MOV	#200, RDSW	, IS IT A < CR > ?
3080	022312	004737	020600		JSR	PC, TCRLF	, BRANCH IF NOT
3081	022316	022737	000007	001000	CMP	#7, COUNT	, ECHO IT WITH < LF >
3082	022324	001037			BNE	75	, WAS IT FIRST CHARACTER
3083	022326	005726			85	TST	(SP)+
3084	022330	000740			BR	OUT	, CHANGE SWR IF NOT FIRST ONE
3085	022332	122737	000060	000704	45	CMPB	#60, TIB
3086	022340	003004			BGT	55	, POP THE STACK
3087	022342	122737	000067	000704	CMPB	#67, TIB	, GET OUT
3088	022350	002005			BGE	65	
3089	022352	012704	023602		55	MOV	#MSG43, R4
3090	022356	004737	020536		JSR	PC, TTOUT	
3091	022362	000742			BR	35	, START OVER IF NOT LEGAL CHARACTER
3092	022364	006337	001002		65	ASL	TEMPST
3093	022370	006337	001002		ASL	TEMPST	
3094	022374	006337	001002		ASL	TEMPST	
3095	022400	142737	000060	000704	BICB	#60, TIB	, GET NITTY-GRITTY
3096	022406	153737	000704	001002	BISB	TIB, TEMPST	
3097	022414	005337	001000		DEC	COUNT	, ONLY WANT 6 DIGITS
3098	022420	001754			BEQ	55	
3099	022422	000711			BR	15	
3100	022424	013777	001002	156220	75	MOV	TEMPST, @SWR
3101	022432	006735			BR	85	, CHANGE SWITCH REGISTER CONTENTS
3102							
3103							

```

3104
3105          , ERROR MESSAGES*****
3106
3107 022434 042052 020105 043 MSG1  ASCII /*DE #/
3108
3109 022441 045 035507 021440 MSG2  ASCII /*G, #/
3110
3111 022446 041045 020073 043 MSG3  ASCII /*B; #/
3112
3113 022453 045 047103 021440 MSG4  ASCII /*CN #/
3114
3115 022460 053452 020105 043 MSG5  ASCII /*WE #/
3116
3117 022465 052 042522 021440 MSG6  ASCII /*RE #/
3118
3119 022472 051052 020123 043 MSG7  ASCII /*RS #/
3120
3121 022477 052 042523 021440 MSG10 ASCII /*SE #/
3122
3123 022504 022445 052445 044516 MSG11 ASCII /*%%UNIT NO #/
3124 022512 020124 047516 020056
3125 022520 043
3126
3127 022521 045 041052 020116 MSG13 ASCII /**BN #/
3128 022526 043
3129
3130 022527 052 047122 021440 MSG14 ASCII /*RN #/
3131
3132 022534 020045 020040 020040 MSG15 ASCII /*          BAD RECORD%%#/
3133 022542 020040 020040 041040
3134 022550 042101 051040 041505
3135 022556 051117 022504 021445
3136
3137 022564 043040 025052 021452 MSG16 ASCII /* F***#/
3138
3139 022572 051040 025052 021452 MSG17 ASCII /* R***#/
3140
3141 022600 042445 052117 020040 MSG20 ASCII /*EOT NO #/
3142 022606 047040 027117 021440
3143 022614 052445 044516 020124 MSG20A ASCII /*UNIT WILL REWIND AND BE:
3144 022622 044527 046114 051040
3145 022630 053505 047111 020104
3146 022636 047101 020104 042502
3147 022644 045
3148 022645 122 051505 040524 ASCII /*RESTARTED ON BLOCK ONE:/
3149 022652 052122 042105 047440
3150 022660 020116 046102 041517
3151 022666 020113 047117 022505
3152 022674 044127 047105 040440 ASCII /*WHEN ALL AVAIL UNITS REACH EOT#/
3153 022702 046114 040440 040526
3154 022710 046111 052440 044516
3155 022716 051524 051040 040505
3156 022724 044103 042440 052117
3157 022732 043
3158
3159
    
```

3160	022733	045	020441	044441	MSG22	ASCII	/%!!!ILLEGAL BOT%%#/
3161	022740	046114	043505	046101			
3162	022746	041040	052117	022445			
3163	022754	021445					
3164							
3165	022756	041445	046517	020104	MSG23	ASCII	/%COMD #/
3166	022764	043					
3167							
3168	022765	045	047516	044440	MSG24	ASCII	/%NO INTERRUPT RETURNED%/
3169	022772	052116	051105	052522			
3170	023000	052120	051040	052105			
3171	023006	051125	042516	022504			
3172	023014	043					
3173							
3174	023015	045	020441	047041	MSG25	ASCII	/%!!!NO CONTROLLER READY !!! STOP %/
3175	023022	020117	047503	052116			
3176	023030	047522	046114	051105			
3177	023036	051040	040505	054504			
3178	023044	020440	020441	051440			
3179	023052	047524	035120	045			
3180	023057	120	042522	051523		ASCII	/%PRESS CONTINUE TO RESUME TESTING%/
3181	023064	041440	047117	044524			
3182	023072	052516	020105	047524			
3183	023100	051040	051505	046525			
3184	023106	020105	042524	052123			
3185	023114	047111	022507	043			
3186							
3187	023121	045	051104	050117	MSG26	ASCII	/%DROPS #/
3188	023126	035123	021440				
3189							
3190	023132	050045	041511	051513	MSG27	ASCII	/%PICKS: #/
3191	023140	020072	043				
3192							
3193	023143	045	052123	052101	MSG30	ASCII	/%STAT #/
3194	023150	021440					
3195							
3196	023152	022445	046524	040454	MSG31	ASCII	/%TM,A,B-11 TS03 OR TU10,N,W MULTIDRIVE DATA RELIABILITY EXERCISER (DZTM
3197	023160	041054	030455	035061			
3198	023166	051524	031460	047440			
3199	023174	020122	052524	030061			
3200	023202	047054	053454	046440			
3201	023210	046125	044524	051104			
3202	023216	053111	020105	040504			
3203	023224	040524	051040	046105			
3204	023232	040511	044502	052114			
3205	023240	020131	054105	051105			
3206	023246	044503	042523	020122			
3207	023254	042050	052132	044115			
3208	023262	043055	022451	043			
3209	023267	105	052116	051105	MSG31A	ASCII	/%ENTER CONDITIONS IN OCTAL%/
3210	023274	041440	047117	044504			
3211	023302	044524	047117	020123			
3212	023310	047111	047440	052103			
3213	023316	046101	021445				
3214							
3215	023322	052445	044516	020124	MSG32	ASCII	/%UNIT NUMBER = #/

3216	023330	052516	041115	051105			
3217	023336	036440	021440				
3218							
3219	023342	042045	047105	044523	MSG33	ASCII	/%DENSITY = #/
3220	023350	054524	036440	021440			
3221							
3222	023356	050045	051101	052111	MSG34	ASCII	/%PARITY = #/
3223	023364	020131	020075	043			
3224							
3225	023371	045	042522	047503	MSG35	ASCII	/%RECORD COUNT = #/
3226	023376	042122	041440	052517			
3227	023404	052116	036440	021440			
3228							
3229	023412	041445	040510	040522	MSG36	ASCII	/%CHARACTER COUNT = #/
3230	023420	052103	051105	041440			
3231	023426	052517	052116	036440			
3232	023434	021440					
3233							
3234	023436	050045	052101	042524	MSG37	ASCII	/%PATTERN NUMBER = #/
3235	023444	047122	047040	046525			
3236	023452	042502	020122	020075			
3237	023460	043					
3238							
3239	023461	045	044523	043516	MSG38	ASCII	/%SINGLE PASS = #/
3240	023466	043514	050040	051501			
3241	023474	020123	020075	043			
3242	023501	041	020441	042445	MSG39	ASCII	/%END OF PASS#
3243	023506	042116	047440	020106			
3244	023514	040520	051523	043			
3245	023521	045	042445	052116	MSG40	ASCII	/%ENTER STALLS:READ = #
3246	023526	051105	051440	040524			
3247	023534	046114	022523	042522			
3248	023542	042101	036440	021440			
3249							
3250	023550	053445	044522	042524	MSG41	ASCII	/%WRITE = #/
3251	023556	036440	021440				
3252							
3253	023562	052045	051125	020116	MSG42	ASCII	/%TUPN AROUND = #
3254	023570	051101	052517	042116			
3255	023576	036440	021440				
3256							
3257	023602	037445	021445		MSG43	ASCII	/%#
3258							
3259	023606	042445	052116	051105	MSG44	ASCII	/%ENTER NOZZLE STALL = #
3260	023614	054440	055117	046132			
3261	023622	020105	052123	046101			
3262	023630	020114	020075	043			
3263							
3264	023635	045	051105	020122	MSG45	ASCII	/%ERR AMT #/
3265	023642	046501	020124	043			
3266							
3267	023647	045	041527	021440	MSG46	ASCII	/%WC #/
3268							
3269	023654	041445	020101	043	MSG47	ASCII	/%CA #/
3270							
3271	023661	045	020441	047041	MSG48	ASCII	/%TUPN BOT ON REWIND #

3272	023666	020117	047502	020124			
3273	023674	047117	051040	053505			
3274	023702	047111	035174	043			
3275							
3276	023707	040	047516	020124	MSG49	ASCII	/ NOT AVAIL #/
3277	023714	053101	044501	020114			
3278	023722	043					
3279	023723	055	052067	045522	MSG50	ASCII	/-7TRK #/
3280	023730	021440					
3281	023732	034455	051124	020113	MSG51	ASCII	/-9TRK #/
3282	023740	043					
3283	023741	045	047516	035116	MSG52	ASCII	/%NON. RETRYABLE #/
3284	023746	042522	051124	040531			
3285	023754	046102	020105	043			
3286	023761	045	025052	047452	MSG53	ASCII	/%***ORIGINAL ERROR***#/
3287	023766	044522	044507	040516			
3288	023774	020114	051105	047522			
3289	024002	025122	025052	043			
3290	024007	045	042522	047503	MSG54	ASCII	/%RECOVERED#/
3291	024014	042526	042522	021504			
3292	024022	051045	052105	054522	MSG55	ASCII	/%RETRY #/
3293	024030	020072	043				
3294	024033	045	052523	050123	MSG56	ASCII	/%SUSPECT BAD TAPE#/
3295	024040	041505	020124	040502			
3296	024046	020104	040524	042520			
3297	024054	043					
3298	024055	045	042522	042520	MSG57	ASCII	/%REPEAT #/
3299	024062	052101	020072	043			
3300	024067	045	020441	052441	MSG58	ASCII	/%''''UNRECOVERABLE BAD SPOT#/
3301	024074	051116	041505	053117			
3302	024102	051105	041101	042514			
3303	024110	041040	042101	051440			
3304	024116	047520	021524				
3305							
3306	024122	020445	020441	040502	MSG59	ASCII	/%''''BAD TAPE OVERFLOW/
3307	024130	020104	040524	042520			
3308	024136	047440	042526	043122			
3309	024144	047514	127				
3310	024147	045	040524	042520		ASCII	/%TAPE WILL BE REWOUND AND REMOVED FROM/
3311	024154	053440	046111	020114			
3312	024162	042502	051040	053505			
3313	024170	052517	042116	040440			
3314	024176	042116	051040	046505			
3315	024204	053117	042105	043040			
3316	024212	047522	115				
3317	024215	045	042524	052123		ASCII	/%TESTING UNTIL ALL ARE PESTARTED AT BLOCK ONE #'
3318	024222	047111	020107	047125			
3319	024230	044524	020114	046101			
3320	024236	020114	051101	020105			
3321	024244	042522	052123	051101			
3322	024252	042524	020104	052101			
3323	024260	041040	047514	045503			
3324	024266	047440	042516	021456			
3325	024274	052045	050101	020105	MSG60	ASCII	/%TAPE MARK = #'
3326	024302	040515	045522	036440			
3327	024310	021440					

3328								
3329	024312	020445	020441	040502	MSG61	ASCII	/%''BACKSPACE ERROR/	
3330	024320	045503	050123	041501				
3331	024326	020105	051105	047522				
3332	024334	122						
3333	024335	045	040524	042520		ASCII	/%TAPE WILL BE REWOUND AND REMOVED FROM /	
3334	024342	053440	046111	020114				
3335	024350	042502	051040	053505				
3336	024356	052517	042116	040440				
3337	024364	042116	051040	046505				
3338	024372	053117	042105	043040				
3339	024400	047522	020115					
3340	024404	052045	051505	044524		ASCII	/%TESTING UNTIL ALL ARE RESTARTED AT BLOCK ONE #/	
3341	024412	043516	052440	052116				
3342	024420	046111	040440	046114				
3343	024426	040440	042522	051040				
3344	024434	051505	040524	052122				
3345	024442	042105	040440	020124				
3346	024450	046102	041517	020113				
3347	024456	047117	027105	043				
3348	024463	052	042527	052040	MSG62	ASCII	/*WE TM*/	
3349	024470	021515						
3350	024472	051452	020105	046524	MSG63	ASCII	/*SE TM*/	
3351	024500	043						
3352	024501	045	052127	051105	MSG64	ASCII	/%WTERR. #/	
3353	024506	035122	021440					
3354	024512	051045	042504	051122	MSG65	ASCII	/%RDERR: #/	
3355	024520	020072	043					
3356	024523	045	052104	051105	MSG66	ASCII	/%DTERR: #/	
3357	024530	035122	021440					
3358	024534	021445			MSG67	ASCII	/%#/	
3359	024536	041040	042101	052040	MSG68	ASCII	/'BAD TAPE SPOTS%#/'	
3360	024544	050101	020105	050123				
3361	024552	052117	022523	043				
3362	024557	052	042523	051040	MSG69	ASCII	/*SE RTY#/'	
3363	024564	054524	043					
3364	024567	052	042522	052040	MSG70	ASCII	/*RE TM#/'	
3365	024574	021515						
3366	024576	051045	040505	020104	MSG71	ASCII	/%PEAD FAILED--RETRY. #	
3367	024604	040506	046111	042105				
3368	024612	026455	042522	051124				
3369	024620	035131	021440					
3370	024624	020445	020441	040510	MSG72	ASCII	/%''HARD READ EPROP#	
3371	024632	042122	051040	040505				
3372	024640	020104	051105	047522				
3373	024646	021522						
3374	024650	051045	051105	040505	MSG73	ASCII	/%REPEAD SUCCESSFUL--PETR: #	
3375	024656	020104	052523	041503				
3376	024664	051505	043123	046125				
3377	024672	026455	042522	051124				
3378	024700	035131	021440					
3379	024704	020045	047523	052106	MSG74	ASCII	/% SOFT #/'	
3380	024712	020072	043					
3381	024715	045	044040	051101	MSG75	ASCII	/% HARD: #/'	
3382	024722	035104	021440					
3383	024726	020045	052122	044522	MSG76	ASCII	/% RTRY #/'	

3384	024734	020072	043			
3385	024737	045	052045	026115	MSG77	ASCII /%TM,A,B-11 AUTO SEQUENCE TEST (DZTMH-F)%/
3386	024744	026101	026502	030461		
3387	024752	040440	052125	020117		
3388	024760	042523	052521	047105		
3389	024766	042503	052040	051505		
3390	024774	020124	042050	052132		
3391	025002	044115	043055	022451		
3392	025010	047105	042524	020122	ASCII	/ENTER RESPONSES IN OCTAL%#/
3393	025016	042522	050123	047117		
3394	025024	042523	020123	047111		
3395	025032	047440	052103	046101		
3396	025040	021445				
3397	025042	040445	052125	020117	MSG78	ASCII /%AUTO CONT #/
3398	025050	047503	052116	020072		
3399	025056	043				
3400	025057	045	025045	025052	MSG79	ASCII /:*****//
3401	025064	025052	025052	025052		
3402	025072	025052	025052	025052		
3403	025100	025052				
3404	025102	025052	025052	025052	ASCII	/*****%#/
3405	025110	025052	025052	025052		
3406	025116	025052	025052	022451		
3407	025124	043				
3408	025125	125	044516	051524	MSG80	ASCII /UNITS TO BE TESTED%#/
3409	025132	052040	020117	042502		
3410	025140	052040	051505	042524		
3411	025146	022504	043			
3412	025151	105	042116	047440	MSG81	ASCII /END OF SEQUENCE NO. #/
3413	025156	020106	042523	052521		
3414	025164	047105	042503	020040		
3415	025172	047516	020056	043		
3416	025177	045	020441	047041	MSG82	ASCII /NO DRIVES AVAILABLE FOR AUTO SEQ--HALT%#/
3417	025204	020117	051104	053111		
3418	025212	051505	040440	040526		
3419	025220	046111	041101	042514		
3420	025226	043040	051117	040440		
3421	025234	052125	020117	042523		
3422	025242	026521	044055	046101		
3423	025250	022524	043			
3424	025253	045	050114	020103	MSG83	ASCII /%LPC #/
3425	025260	043				
3426	025261	045	042522	044507	MSG84	ASCII /%REGISTER START = #
3427	025266	052123	051105	051440		
3428	025274	040524	052122	036440		
3429	025302	021440				
3430	025304	053045	041505	047524	MSG85	ASCII /%VECTOR ADDRESS = #
3431	025312	020122	042101	051104		
3432	025320	051505	020123	020075		
3433	025326	043				
3434	025327	052	040520	052124	MSG86	ASCII /%PATTRN #/
3435	025334	047122	021440			
3436	025340	050045	042522	040515	MSG87	ASCII /%PREMATURE EOT IN AUTO SEQ
3437	025346	052524	042522	042440		
3438	025354	052117	044440	020116		
3439	025362	052501	047524	051440		

3440	025370	050505					
3441	025372	052045	050101	020105		ASCII	/TAPE WILL BE REWOUND AND AUTO SEQUENCE/
3442	025400	044527	046114	041040			
3443	025406	020105	042522	047527			
3444	025414	047125	020104	047101			
3445	025422	020104	052501	047524			
3446	025430	051440	050505	042525			
3447	025436	041516	105				
3448	025441	045	044527	046114		ASCII	/WILL CONTINUE ON THIS UNIT#/
3449	025446	041440	047117	047111			
3450	025454	042525	047440	020116			
3451	025462	044124	051511	052440			
3452	025470	044516	021524				
3453	025474	051040	052105	054522	MSG88	ASCII	/RETRY#/
3454	025502	043					
3455							
3456	025503	045	020441	052441	MSG89	ASCII	/UNIT IS REWINDING, TEST WILL START WHEN DONE#/
3457	025510	044516	020124	051511			
3458	025516	051040	053505	047111			
3459	025524	044504	043516	020073			
3460	025532	042524	052123	053440			
3461	025540	046111	020114	052123			
3462	025546	051101	020124	044127			
3463	025554	047105	042040	047117			
3464	025562	021505					
3465	025564	042052	047105	021440	MSG90	ASCII	/DEN #/
3466	025572	050052	051101	021440	MSG91	ASCII	/PAR #/
3467	025600	020441	022441	042045	MSG92	ASCII	/DROPPED UNIT #/
3468	025606	047522	050120	042105			
3469	025614	052440	044516	035124			
3470	025622	021440					
3471	025624	040445	052124	046505	MSG93	ASCII	/ATTEMPT TO RESTART UNIT WILL BE/
3472	025632	052120	052040	020117			
3473	025640	042522	052123	051101			
3474	025646	020124	047125	052111			
3475	025654	053440	046111	020114			
3476	025662	042502					
3477	025664	046445	042101	020105		ASCII	/MADE AT END OF PASS #/
3478	025672	052101	042440	042116			
3479	025700	047440	020106	040520			
3480	025706	051523	020441	022441			
3481	025714	021445					
3482	025716	020441	022441	047045	MSG94	ASCII	/NO MORE UNITS TO TEST IN THIS PASS/
3483	025724	020117	047515	042522			
3484	025732	052440	044516	051524			
3485	025740	052040	020117	042524			
3486	025746	052123	044440	020116			
3487	025754	044124	051511	050040			
3488	025762	051501	123				
3489	025765	045	046101	020114		ASCII	/ALL ARE DROPPED OR REWOUND #/
3490	025772	051101	020105	051104			
3491	026000	050117	042520	020104			
3492	026006	051117	051040	053505			
3493	026014	047525	042116	022456			
3494	026022	021445					
3495	026024	020441	022441	047514	MSG95	ASCII	/LOST SELECT REMOTE#/

3496	026032	052123	051440	046105		
3497	026040	041505	020124	042522		
3498	026046	047515	042524	043		
3499	026053	041	020441	040445	MSG96	ASCII /!!!%ALL ARE DROPPED END OF PASS STOP !!!/
3500	026060	046114	040440	042522		
3501	026066	042040	047522	050120		
3502	026074	042105	020072	047105		
3503	026102	020104	043117	050040		
3504	026110	051501	020123	052123		
3505	026116	050117	020441	021441		
3506	026124	020441	041445	047101	MSG97	ASCII /!!!%CANNOT TEST LOAD MEDIUM%#/
3507	026132	047516	020124	042524		
3508	026140	052123	046040	040517		
3509	026146	020104	042515	044504		
3510	026154	046525	021445			
3511	026160	057045	021507		SCNTG	ASCII /% G#/
3512	026164	022445	053523	036522	SMSWR	ASCII /%:SWR= #/
3513	026172	021440				
3514	026174	020040	042516	036527	SMNEW	ASCII / NEW= #
3515	026202	021440				
3516						
3517	026204	000000			WDATA	EVEN 0 WRITE BUFFER
3518						
3519		032216				= +4010
3520	032216	000000			RDATA	0 READ BUFFER
3521						
3522		000001				ENC

DATER1	001134	253#	2007*	2262						
DATER2	001136	254#								
DATER3	001140	255#								
DATER4	001142	256#								
DATER5	001144	257#								
DATER6	001146	258#								
DATER7	001150	259#								
DATER8	001152	260#								
DATR	013236	499	1648	1827#						
DATRO	013254	1831#	1834							
DATO	012552	375	1682#							
DATOA	012602	1688#	1703	1707	1710					
DATOB	012620	1691#	1694	1696						
DATOC	012704	1701	1708#							
DATOD	012712	1711#	1719							
DATOE	012722	1713#	1718							
DATOF	012736	1715	1717#							
DAT1	012754	376	1727#							
DAT1A	012760	1728#	1737	1760	1765	1770	1775	1801	1806	
DAT1B	012764	1729#	1731							
DAT10	013074	383	1774#							
DAT11	013104	384	1780#							
DAT11A	013112	1782#	1785							
DAT12	013126	385	1790#							
DAT12A	013136	1792#	1795							
DAT13	013152	386	1800#							
DAT14	013162	387	1805#							
DAT15	013172	388	1810#							
DAT15A	013202	1812#	1821							
DAT15B	013206	1813#	1818							
DAT15C	013220	1815	1817#							
DAT15R	013176	1811#	1822							
DAT2	012776	377	1736#							
DAT3	013004	378	1741#							
DAT3A	013012	1743#	1754							
DAT3B	013016	1744#	1747							
DAT4	013032	379	1752#							
DAT5	013044	380	1759#							
DAT6	013054	381	1764#							
DAT7	013064	382	1769#							
DCHK	013712	1081	1200	1947#						
DEREX	014610	2069	2091	2094	2101	2103#				
DEREX1	014644	2104	2107	2109	2111#					
DERFL	000744	177#	1948*	1983*	1997*	2002				
DERR	014242	1982	1996	2040#						
DERRO	014256	2041	2043#	2110						
DERROA	014316	2045	2047	2052#						
DERROB	014336	2057#								
DERR1	014370	2064#								
DERR2	014372	2065#								
DERR3	014376	2066#								
DERR4	014400	2042	2067#							
DERR4A	014544	2092#								
DERR4B	014556	2078	2095#							
DERR5	014576	2100#								
DERR6	014604	2081	2098	2102#						

MSG15	022534	2072	3132#			
MSG16	022564	2483	3137#			
MSG17	022572	2480	3139#			
MSG2	022441	2058	3109#			
MSG20	022600	606	3141#			
MSG20A	022614	612	3143#			
MSG22	022733	1055	3160#			
MSG23	022756	1269	2416	3165#		
MSG24	022765	2568	3168#			
MSG25	023015	2532	3174#			
MSG26	023121	2195	3187#			
MSG27	023132	2210	3190#			
MSG3	022446	2062	3111#			
MSG30	023143	1280	2426	3193#		
MSG31	023152	416	1427	3196#		
MSG31A	023267	1429	3209#			
MSG32	023322	1467	3215#			
MSG33	023342	1502	3219#			
MSG34	023356	1514	3222#			
MSG35	023371	1535	3225#			
MSG36	023412	1545	3229#			
MSG37	023436	1557	3234#			
MSG38	023461	1576	3239#			
MSG39	023501	659	3242#			
MSG4	022453	2052	3113#			
MSG40	023521	1585	3245#			
MSG41	023550	1594	3250#			
MSG42	023562	1603	3253#			
MSG43	023602	2799	3089	3257#		
MSG44	023606	1206	3259#			
MSG45	023635	1313	3264#			
MSG46	023647	2430	3267#			
MSG47	023654	2435	3269#			
MSG48	023661	639	729	3271#		
MSG49	023707	520	554	1493	3276#	
MSG5	022460	766	809	964	2526	3115#
MSG50	023723	1498	2963	3279#		
MSG51	023732	1500	2960	3281#		
MSG52	023741	807	875	1071	3283#	
MSG53	023761	816	886	1125	3286#	
MSG54	024007	911	3290#			
MSG55	024022	913	922	3292#		
MSG56	024033	920	3294#			
MSG57	024055	926	3298#			
MSG58	024067	983	3300#			
MSG59	024122	597	3306#			
MSG6	022465	1038	1073	2529	3117#	
MSG60	024274	1567	3325#			
MSG61	024312	592	3329#			
MSG62	024463	852	877	3348#		
MSG63	024472	1251	3350#			
MSG64	024501	2234	3352#			
MSG65	024512	2244	3354#			
MSG66	024523	2259	3356#			
MSG67	024534	2272	2306	3358#		
MSG68	024536	2280	3359#			

_OMMEN 1#
 ENDCOM 1#
 ESCAPE 1#
 GETPRI 1#
 GETSWR 1#
 MULT 1#
 NEWTST 1#
 POP 1#
 PUSH 1#
 PEPORT 1#
 SETPRI 1#
 SETUP 1#
 SKIP 1#
 SLASH 1#
 STARS 1#
 SWRSU 1#
 TYPB'N 1#
 TYPDEC 1#
 TYPNAM 1#
 TYPNUM 1#
 TYPOCS 1#
 TYPOCT 1#
 TYPTXT 1#
 \$\$ESCA 1#
 \$\$NEWT 1#
 \$\$\$SKIP 1#
 EQUAT 1#
 HEADE 1#
 KT11 1#
 SETUP 1#
 SWRHI 1#
 \$ACT1 1#
 \$APT8 1#
 \$APTH 1#
 \$APTY 1#
 \$ASTA 1#
 \$CATC 1#
 \$CMTA 1#
 \$DB20 1#
 \$DB20 1#
 \$DIV 1#
 \$EOP 1#
 \$ERRO 1#
 \$ERRT 1#
 \$MULT 1#
 \$POWE 1#
 \$RAND 1#
 \$RDDE 1#
 \$RDOC 1#
 \$READ 1#
 \$R2AZ 1#
 \$SAVE 1#
 \$SB20 1#
 \$SB20 1#
 \$SCOP 1#
 \$S'ZE 1#

75

6# 73

SSUPR 1#
STRAP 1#
STYPB 1#
STYPD 1#
STYPE 1#
STYPO 1#
S4OCR 1#
1170 1#

95 032220 000

ERROPS DETECTED 0

DZTMHF BIN, DZTMHF LST/CRF/SOL/NL TOC=DZTMHF SML, DZTMHF P11
RUN-TIME 9 13 1 SECONDS
RUN-TIME RATIO 267/24=10.7
CORE USED 32K (63 PAGES)

ACO

L

