TS11 TS11 DATA RELIAB AH E455C MC
CZTSHCO FICHE 1 OF 1
MAR 88
COPYR
MADE IN USA

IDENTIFICATION

PRODUCT CODE: AC-E454C-MC

PRODUCT NAME: CZTSHC0 TS11 DATA RELIAB

PRODUCT DATE: 01-OCT-79

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: ROBERT F. WERY

USER DOCUMENTATION

USER DOCUMENTATION TABLE OF CONTENTS

6.0 DEVICE INFORMATION

GLOSSARY

ACT	AUTOMATED COMPUTER TEST SYSTEM
APT	AUTOMATED PRODUCT TEST SYSTEM
BYTE/RECORD/FILE COUNT BRF	IS STORED IN THE 4TH WORD OF THE COMMAND PACKET AND IT'S USE BY THE TS04 DEPENDS ON THE TYPE OF COMMAND.
CMD	TS04 COMMAND (SEE 2.3.14.1 FOR LIST OF COMMANDS)
COMMAND PACKET CMDPKT	FOUR WORD PACKET IN THE CPU MEMORY WHICH CONTAINS ALL INFORMATION NEEDED BY THE TS04 TO EXECUTE A COMMAND.
EXTENDED STATUS	FOUR WORDS OF TS04 STATUS WHICH ARE TRANSFERRED AS PART OF THE MESSAGE PACKET AT THE COMPLETION OF A COMMAND.
MESSAGE PACKET	SEVEN WORD PACKET IN THE CPU MEMORY INTO WHICH THE TS04 STORES STATUS AT THE COMPLETION OF A COMMAND.
PC	PROGRAM COUNTER
PSW	PROCESSOR STATUS WORD
RESIDUAL FRAME COUNT RFC	THIS COUNT IS PART OF THE MESSAGE PACKET AND CONTAINS THE NUMBER OF BYTES/RECORDS /FILES REMAINING TO BE PROCESSED AT THE COMPLETION OF A COMMAND.
SPECIAL CONDITION SPEC COND	TSS4 BIT15. WHEN SET, INDICATES THAT THE LAST COMMAND DID NOT COMPLETE WITH-OUT INCIDENT.
TERMINATION CLASS CODE TCC	THREE BIT CODE IN THE TSSR WHICH INDI-CATES THE TYPE OF COMMAND TERMINATION.
TSBA	TAPE SYSTEM BUS ADDRESS REGISTER.
TSDB	TAPE SYSTEM DATA BUFFER REGISTER.
TSSR	TAPE SYSTEM STATUS REGISTER.
XST0	EXTENDED STATUS REGISTER 0
XST1	EXTENDED STATUS REGISTER 1
XST2	EXTENDED STATUS REGISTER 2
XST3	EXTENDED STATUS REGISTER 3
XXDP+	XXDP+ IS A ''CATCH-ALL'' NAME FOR A GROUP OF PDP-11 DIAGNOSTIC PACKAGES AVAILABLE ON MULTIMEDIA.

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

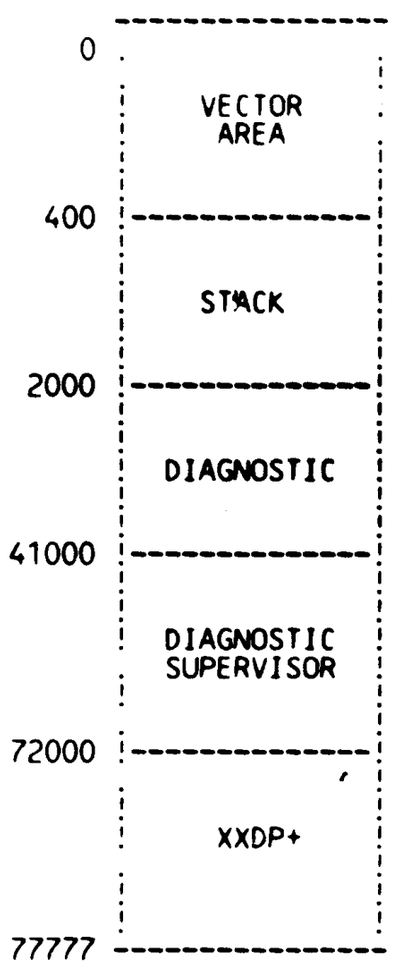
1.1.1 FUNCTIONAL DESCRIPTION

THIS PROGRAM CAN BE USED AS A BASIC FUNCTION TEST, A DATA RELIABILITY TEST, A COMPATABILITY TEST, OR TO EXECUTE A SEQUENCE OF OPERATOR SELECTED COMMANDS.

1.1.2 STRUCTURE OF PROGRAM

THIS DIAGNOSTIC IS A SINGLE PROGRAM FROM THE STANDPOINT OF THE DIAGNOSTIC USER, BUT IT CONTAINS A CONTROL MODULE RELEASED INDEPENDENTLY AS A DIAGNOSTIC SUPERVISOR.

1.1.3 MEMORY MAP



FREE MEMO SPACE FOR WR/RD BFRS OR OTHER PUROSES
IS ALLOCATED BY THE SUPERVISOR ON REQUEST OR CHOOSEN
BY PROGRAMMER TO RESIDE BETWEEN THE DIAG AND THE
SUPERVISOR.

1.1.4 DIAGNOSTIC INFORMATION

1.1.4.1 SCOPE

THIS DIAGNOSTIC CAN TEST UP TO 4 UNITS SIMULTANEOUSLY. THE 4 UNITS ARE ASSIGNED LOGICAL UNIT NUMBERS 0 - 3 BY THE DIAGNOSTIC.

THERE ARE 5 TESTS IN THIS PROGRAM:

TEST 1 - BASIC FUNCTIONS.
TEST 2 - DATA RELIABILITY.
TEST 3 - WRITE COMPATABILITY/WRITE UTILITY.
TEST 4 - READ COMPATABILITY/READ UTILITY.
TEST 5 - OPERATOR SELECTED SEQUENCE UTILITY.

1.1.4.2 ERROR RECOVERY

ERROR RECOVERY IS PERFORMED ON READ, WRITE AND WRITE TAPE MARK ERRORS UNLESS RECOVERY IS INHIBITED BY THE OPERATOR.
THE READ FORWARD/READ REVERSE RETRY LIMIT IS 16 (8 IN THE SAME DIRECTION AND 8 IN THE OPPOSITE DIRECTION). FOR MORE INFORMATION ON ERROR RECOVERY PROCEDURES, SEE SECTION 3.0 (ERROR REPORTING).

1.1.4.3 WRITE ERROR RECOVERY

THERE ARE 2 DISTINCT, SELECTABLE WRITE-ERROR-RECOVERY ALGORITHMS:
1. MEDIA/OPERATIONAL SELECTIVE ALGORITHM
2. OPERATIONAL ALGORITHM

BY DEFAULT THE DIAGNOSTIC SELECTS THE FIRST ALGORITHM TO DISCERN MEDIA RELATED WRITE ERRORS FROM OPERATIONAL ONES.

TO SELECT THE SECOND ALGORITHM:
ANSWER 'Y' TO CHANGE SW (L) ?
ANSWER 'N' TO BAD TAPE SPOT DETECTION (L) Y ?

WHEN ERROR RECOVERY IS INHIBITED, THE LATTER QUESTION IS NOT ASKED AND BOTH ALGORITHMS ARE BYPASSED.

1.1.4.3.1 MEDIA/OPERATIONAL SELECTIVE WRITE-ERROR-RECOVERY ALGORITHM

SCOPE

THE ALGORITHM DISCERNS MEDIA RELATED WRITE ERRORS FROM OPERATIONAL ONES.

ALGORITHM

A WRITE RETRY SUBROUTINE IS CALLED BY THE RECOVERABLE ERROR SUBROUTINE ENTERED UPON DETECTIION OF A WRITE RECOVERABLE ERROR.
THE WRITE RETRY SUBROUTINE REWRITES RECORD IN SAME SPOT ON TAPE:
REPEAT 4 TIMES.
IF ALL 4 REPEATS ARE GOOD, RECORD IS CONSIDERED AS RECOVERED AND A RECOVERABLE WRITE ERROR IS LOGGED AT THAT RECORD NUMBER.
IF ANY OF THE 4 REPEAT FAILS, ERASE BAD RECORD, LOGG SUSPECTED BAD SPOT AT THAT RECORD NUMBER, RETRY AGAIN 3 INCHES FURTHER DOWN TAPE.
RETRY 4 TIMES, UP TO 4 REPEATS EACH.
IF RECORD CANNOT BE WRITTEN WITHOUT RECOVERABLE ERROR AFTER 4 RETRIES, ERASE RECORD, REPORT RETRY FAILED ON BAD SPOT.
THE RECOVERABLE ERROR SUBROUTINE THEN CONTINUES TO CALL THE WRITE RETRY SUBROUTINE, WHICH REISSUES THE GROUP OF 4 RETRIES, UNTIL THE RECORD IS RECOVERED OR 20 BAD SPOTS HAVE BEEN LOGGED .

TWENTY (20) BAD SPOTS MAXIMUM ARE ALLOWED PER TAPE PASS.
WHEN 20 BAD SPOTS HAVE BEEN LOGGED, ON SAME RECORD NUMBER OR NOT,
TAPE IS CONSIDERED DEFECTIVE: A BAD TAPE OVERFLOW MESSAGE IS PRINTED
AND UNIT IS REWOUND, THEN DROPPED.

DURING THE RECOVERY PROCESS, IT IS NECESSARY TO PERFORM SEVERAL TAPE
POSITION OPERATIONS: SPACE REVERSE, ERASE. IF A POSITION ERROR STATUS
IS DETECTED DURING THOSE OPERATIONS, THEN THE RECOVERY ATTEMPT IS ABORTED,
AN APPROPRIATE UNRECOVERABLE MESSAGE IS PRINTED AND UNIT IS DROPPED.

ALL BADLY WRITTEN RECORDS FLAGGED WITH RECOVERABLE ERRORS ARE ERASED
UNTIL RECOVERED, INCLUDING THE RECORD AT THE 20TH BAD SPOT,
SO THAT ALL RECORDS LEFT ON TAPE ARE GOOD WRITTEN RECORDS.
BAD SPOTS ARE ERASED, WITH ERASE GAPS FROM 3 TO 12 INCHES PER RETRY GROUP.
UP TO 20 FEET OF ERASE GAP COULD RESULT WHEN RETRYING TO RECOVER
A SINGLE RECORD,IF NO BAD SPOT WERE PREVIOUSLY DETECTED.
THAT LONG STRETCH OF BAD TAPE WOULD THEN BE FLAGGED WITH 20
BAD SPOTS AT SAME RECORD NUMBER AND THE TAPE CONSIDERED DEFECTIVE.

BAD SPOTS REPORTS

IF THE PRINT OF RECOVERABLE ERRORS IS ENABLED, THE BAD SPOTS ON TAPE ARE
IDENTIFIED AS THEY ARE DETECTED. SINCE THE BAD RECORDS ARE ERASED UNTIL RECOVERED,
THE BAD SPOTS ACTUALLY PRECEDES THE RECORD NUMBER THAT IDENTIFIES THEM.
THE NUMBER OF REPEATS AND RETRIES ATTEMPTED IS PRINTED, FROM WHICH THE
LENGTH OF ERASE GAPS CAN BE DETERMINED: APPROXIMATELY 3 INCHES PER RETRY.

THE STATISTICAL REPORT PRINTED AT THE END OF TEST 2 OR UPON A 'PRINT'' REQUEST,
CONTAINS A SUMMARY OF THE BAD SPOTS LOGGED ON THE CURRENT TAPE PASS.
IN THAT REPORT, ALL COUNTS ARE CUMULATIVE FROM PASS TO PASS, EXCEPT FOR
THE NUMBER OF BAD SPOTS: IT RELATES TO A 'TAPE PASS'' ONLY.
FOR THIS PURPOSE, A 'TAPE PASS'' IS A WRITE PASS FROM BOT TO EOT, OR FROM
BOT TO WHERE THE DIAGNOSTIC IS HALTED BEFORE REACHING EOT.
A PASS IS DEFINED BY THE SUPERVISOR AS A RUN THROUGH ALL THE TESTS REQUESTED
ON ALL UNITS SELECTED. THOSE PASSES ARE IDENTIFIED AS 'PASS'' AND 'EOP''.

THE NUMBER OF WRITE RETRIES, CUMULATIVE FROM PASS TO PASS, IS A GLOBAL
COUNT OF HOW MANY TIMES THE GROUP OF 4 RETRIES HAS BEEN CALLED.

THE NUMBER OF WRITE RECOVERABLE ERRORS EXCLUDES BAD TAPE SPOTS
AND REFLECTS THE SPECIFICATIONS OF THE HARDWARE UNDER TEST.
PER TAPE PASS, THE NUMBER OF WRITE RETRIES EQUALS THE SUM OF THE
NUMBER OF RECOVERABLE WRITE ERRORS AND BAD SPOTS, MOST OF THE TIME.

TO CLEAR CUMULATIVE COUNTS, ANSWER 'Y' TO: CLEAR COUNTERS (L) Y ?.
BAD TAPE SPOTS COUNT IS CLEARED WHEN WRITING FROM BOT.

IF TEST 2 IS HALTED, THEN RESTARTED OR CONTINUED, THE RECORD COUNT
IS RESET TO ZERO AND THE BAD SPOT ID SHALL FOLLOW THAT RESET COUNT.

SINCE ALL WRITTEN RECORDS ARE KNOWN GOOD, THE READ ERRORS CAN
BE ATTRIBUTED TO TRANSIENT NOISE, TRANSIENT ELECTRICAL MALFUNCTIONS,
OR CONTAMINANTS ON TAPE AS OPPOSED TO TAPE DEFECTS.

THE SAME RECORDS MUST BE WRITTEN FORM TAPE PASS TO TAPE PASS
FOR THE BAD SPOTS ID TO REMAIN CONSISTENT IN THOSE TAPE PASSES.

EXAMPLE OF A TAPE PASS PRINTS:

```
C7TSH SFT ERR  00009 ON UNIT 00 TST 002 SUB 000 PC: 012100
RECOVERABLE ERROR
```

```
WRT CMD FAILED - UNIT 0   PASS:    1   RECORD:    6
PREVIOUS CMD WAS  WRT
CMDPKT  TSBA    RFC     TSSR   TCC
100205  002406  000000  100210  4
026600
000000
003107
XST0    XST1    XST2    XST3
000350  000002  100400  000000
SUSPECT BAD SPOT AFTER 1 RETRY, 2 REPEAT
SUSPECT BAD SPOT AFTER 2 RETRY, 1 REPEAT
SUSPECT BAD SPOT AFTER 3 RETRY, 1 REPEAT
SUSPECT BAD SPOT AFTER 4 RETRY, 3 REPEAT
RETRY FAILED ON BAD SPOT...ERASED!
SUSPECT BAD SPOT AFTER 1 RETRY, 1 REPEAT
SUSPECT BAD SPOT AFTER 2 RETRY, 1 REPEAT


CZTSH SFT ERR  00009 ON UNIT 00 TST 002 SUB 000 PC: 012100
RECOVERABLE ERROR
WRT CMD FAILED - UNIT 0   PASS:    1   RECORD:10210
PREVIOUS CMD WAS  WRT
CMDPKT  TSBA    RFC     TSSR   TCC
100205  002406  000000  10C210  4
026600
000000
004000
XST0    XST1    XST2    XST3
000350  000002  100010  000000
RECOVERED ON RETRY # 1
^C
DR>PRI

UNIT 0   PASS:    1   RECORD:10210
BYTES WRITTEN    0,272,279,691
BYTES READ REV   0,301,123,654
BYTES READ REV   0,301,120,381
                        WRT    RDR    RDF
RECOVERABLE ERRORS       1      0      0
UNRECOVERABLE ERRORS     0      0      0
WRITE RETRIES            3

 2 BAD SPOTS THIS TAPE PASS PRECEDING RECORD #:
           6     6
SPEC COND   HARD   FATAL   COMPARE
       2      0       0       0
DR>

THIS EXAMPLE SHOWS:
        RECORD 6 RECOVERED ON 2ND RETRY GROUP
        THE 2 BAD SPOTS RESIDE IN A 18 INCH ERASE GAP BETWEEN RECORDS 5 AND 6
        RECORD 10210 RECOVERED ON 1ST RETRY OF 4 GOOD REPEATS
        3 WRITE GROUP RETRIES ATTEMPTED, RESULTING IN:
                1 RECOVERABLE  WRT ERR FROM RECORD 10210
                2 BAD SPOTS BETWEEN RECORDS 5 AND 6
```

1.1.4.3.2 OPERATIONAL WRITE-ERROR-RECOVERY ALGORITHM

WHEN THIS ALGORITHM IS SELECTED, THE TS11 WRITE RETRY COMMAND IS ISSUED UP TO 16 TIMES OR UNTIL RECORD IS RECOVERED, ON

A WRITE RECOVERABLE ERROR. THE WRITE RETRY COMMAND CONSISTS OF A SPACE REVERSE OVER THE BAD RECORD, THEN AN ERASE OF 3 INCHES OF TAPE AND REWRITE OF THE RECORD. THAT COMPOSITE COMMAND DOES NOT ALLOW TO DETECT BAD SPOTS ON TAPE.
THEREFORE NO BAD TAPE SPOTS STATUS IS PRINTED.

IF RECORD CANNOT BE RECOVERED AFTER 16 WRITE RETRY COMMANDS, A RETRY LIMIT EXCEEDED IS FLAGGED AND UNIT IS DROPPED.

1.1.4.4 DIAGNOSTIC TIMING ADJUSTMENT

A NUMBER OF SUPERVISOR TIMING DELAYS MACROS, KNOWN AS WATCH DOG DELAYS, ARE CALLED BY THE DIAGNOSTIC TO WAIT FOR VARIOUS COMMANDS COMPLETION. THESE DELAYS ARE NOT CALIBRATED AND SIMPLY EXPANDS INTO AN INLINE NESTED LOOP PAIR. THE COUNT FOR THE OUTER LOOP COMES FROM THE VARIABLE ARGUMENT SUPPLIED BY THE DELAY CALLS. THE COUNT FOR THE INNER LOOP COMES FORM THE FIXED ''HEADER'' ELEMENT ''L$DLY''.
AS THE DIAGNOSTIC IS RUN ON DIFFERENT CPU'S, THESE DELAYS WILL VARY IN LENGTH WITH MEMORY SPEED.

IF TIME-OUT OCCURS WHEN NO APPARENT MALFUNCTIONS IN THE TAPE UNIT IS EVIDENT, ALL TIMINGS OF THE DIAGNOSTIC MAY BE ADJUSTED TO MATCH MEMORY SPEED AND NOT RESULT IN TIME-OUTS, BY PATCHING THAT FIXED DELAY ELEMENT ''L$DLY''.

A PRESET COUNT OF 500 RESIDES AT ''L$DLY'' IN LOCATION 2116 OF THE ''HEADER'' SECTION.

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE REQUIREMENTS

PDP-11 PROCESSOR WITH 16K OR MORE OF MEMORY
CONSOLE DEVICE (LA30,LA36,VT50,ETC.)
PROGRAM LOAD DEVICE

1.2.2 SOFTWARE REQUIREMENTS

DIAGNOSTIC SUPERVISOR

1.3 RELATED DOCUMENTS AND STANDARDS

XXDP+ USERS MANUAL MD-11-CHQUS
DIAGNOSTIC SUPERVISOR PROGRAM LISTING
PDP-11 DIAGNOSTIC SUPERVISOR INTERFACE SPECIFICATION.
PDP-11 DIAGNOSTIC SUPERVISOR PROGRAMMER'S GUIDE.
TS11/TS04 PROGRAMMING SPECIFICATION.
TS11/TS04 ENGINEERING SPECIFICATION.
TS11/TS04 COMMAND PACKET SPECIFICATION.

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

ORDER OF HOST CPU DIAGNOSTIC USAGE:

1) CONTROL LOGIC PROGRAM - ALL TESTS.

2) DATA RELIABILITY PROGRAM:

   A) BASIC FUNCTION TEST.
   B) DATA RELIABILITY TEST.

1.5 ASSUMPTIONS

THE HARDWARE OTHER THAN THE SUBSYSTEM BEING TESTED IS ASSUMED TO WORK PROPERLY. FALSE ERRORS MAY BE REPORTED IF THE PROCESSOR, MEMORY, ETC., DO NOT FUNCTION PROPERLY.

1.6 DIAGNOSTIC HISTORY

REVISION A - OCT 1978
- ORIGINAL RELEASE

REVISION B - FEB 1979
- CORRECTED END OF TAPE PROBLEMS IN TESTS 3-5.
- CHANGED DEFAULT VECTOR ADDRESS FROM 150 TO 224.
- DECREASED MAXIMUM RECORD LENGTH FROM 4096 TO 2048 BYTES.

REVISION B - AUG 1979
-DO NOT PRINT RECOVERABLE ERRORS UNLESS REQUESTED BY OPERATOR
-WARN OPERATOR OF UNIT(S) BEING NOT READY OR OFF-LINE.
 DROP UNIT(S) LEFT NOT READY OR OFF-LINE FOR 3.5 MINUTES.
-IMPROVE BEHAVIOR AT EOT
 -IN TEST 2, FREEZE UNITS REACHING EOT UNTIL OTHERS
  CATCH-UP INSTEAD OF ALLOWING THEM TO SHUTTLE AT EOT
 -WHEN ALL UNITS REACH EOT, WRITE ONE RECORD BEYOND EOT.
  READ REV THAT EXTRA RECORD TO POSITION TAPE
  SO THAT THE NEXT COMMAND REQUESTED CAN BE EXECUTED.
  THAT EXTRA RECORD SHALL LEAVE A CLEAN IRG GAP AND A VALID
  RECORD TO READ WHEN SHORTER READ STOP DISTANCE MIGHT CAUSE
  UNIT TO FLAG EOT ON THAT EXTRA RECORD INSTEAD OF THE
  PREVIOUS ONE. THIS SHOULD ELIMINATE MANY READ ERRORS AT
  EOT AND TAPES RUNNING OFF THE WHEELS.
-WRITE RECORD COUNT ON TAPE.
 PRINT RECORD COUNT READ FROM TAPE IN READ ERROR PRINTS TO
 INDICATE IF POSITION WAS LOST.

* CAUTION *

INTERPRET THAT "RECORD READ" COUNT WITH CAUTION.
IF VERY DIFFERENT FROM RECORD COUNT TRACKED BY THE DIAGNOSTIC,
POSITION IS NOT NECESSARELY LOST. ERRORS IN READING THAT
RECORD MIGHT HAVE CAUSED RECORD COUNT TO BE ERRONEOUSLY
READ FROM TAPE.
IN TEST 2, IF DIAGNOSTIC IS RESTARTED OR CONTINUED, RECORD COUNT
IS RESET TO ZERO ALTHOUGH TAPE WAS NOT REWOUND. THIS IS
NECESSARY BECAUSE THERE IS NO ACCURATE WAY TO DETERMINE
ON WHAT RECORD COUNT OF WHAT UNIT THE DIAGNOSTIC WAS HALTED
BEFORE RESTARTING OR CONTINUING.
IT IS SUGGESTED THAT A "PRINT" BE REQUESTED WHEN HALTING DIAG
TO GET A PRINT OF THE RECORD COUNT WHEN HALTED.

-VERIFY RECORD OF 4000 BYTES INSTEAD OF 22 BYTES.
-WHEN COMPARING DATA, CHECK AND PRINT IF NO DATA WAS READ
 OR RECORD WAS LONGER THAN EXPECTED.
-FREEZE TSSR REG WHEN A COMMAND IS COMPLETED TO AVOID DIFFERENCES
 BETWEEN TSSR AND TCC FETCHED AT DIFFERENT TIMES.
-WHEN DROPPING A UNIT, FLAG SECOND PRINT OF EXTENDED STATUS AS BEING
 THE RESULT OF A GET STATUS COMMAND.
 WAIT FOR SSR UP BEFORE PRINTING THAT STATUS.
-ADJUST "PASS" COUNT OF DIAG TO MATCH "EOP" PASS COUNT OF SUPERVISOR.
-INCREASE NUMBER OF SELECTABLE COMMANDS IN TEST 5 FROM
 4 TO 7. DEFAULT COMMAND 6 IS NOW REWIND.
-CONVERT DIAG TO REV C OF SUPERVISOR.
 ADD SEVERAL SECTIONS:
  PROTECT TABLE
  AUTO-DROP CODE
  HARD CODED PARAMETER TABLE

REVISION C - OCT 79
- ADD MEDIA/OPERATIONAL SELECTIVE WRITE-ERROR-RECOVERY ALGORITHM
  TO DETECT BAD SPOTS ON TAPE.

2.0 OPERATING INSTRUCTIONS

FOR OPERATING INSTRUCTIONS, PLEASE SEE CHAPTER 5 OF XXDP+ OPERATOR'S MANUAL.

2.1 HARDWARE PARAMETERS

ON A 'N' RESPONSE TO "CHANGE HW?", THE DIAG SHALL RUN ASSUMING ONE UNIT AT TSSR = 172522 WITH A VECTOR = 224.

ON A 'Y' RESPONSE TO "CHANGE HW?" QUESTION, THEN THE FOLLOWING QUESTIONS WILL BE ASKED ON A START COMMAND. THE VALUE LOCATED TO THE LEFT OF THE QUESTION MARK IS THE DEFAULT VALUE THAT WILL BE TAKEN ON A CARRIAGE RETURN RESPONCE.

TSSR ADDRESS (172522) ?

VECTOR (224) ?

THE VALIDITY OF THESE PARAMETERS CAN BE CHECKED BEFORE RUNNING THE TESTS BY SETTING THE FLAG "ADR" ON A STA, RES OR CON COMMAND. THE SO CALLED AUTO DROP CODE SHALL THEN BE EXECUTED AFTER THE INIT CODE AND BEFORE THE HARDWARE TESTS ARE RUN. THAT CODE FIRST TESTS THE ADRRESS OF THE TSSR(S). IF NO RESPONSE, IT DROPS THE UNIT(S) IMMEDIATELY WITH THE FOLLOWING MESSAGE:

BUS TRAP AT XXXXXX (XXXXXX = TSSR AD)
INTERFACE BAD OR NOT SET TO ABOVE AD.

ON A RESPONSE FROM THE INTERFACE, THE UNITS THAT ARE NOT READY OR NOT ON-LINE ARE DROPPED IMMEDIATELY. THE HARDWARE TESTS SHALL THEN BE RUN ON PESPONDING UNITS.

IF THE "ADR" FLAG IS NOT SET, THE READY AND OFF-LINE STATUS OF THE UNITS ARE CHECKED. A MESSAGE SHALL BE PRINTED EVERY SO OFTEN TO WARN THE OPERATOR OF UNITS BEING NOT READY OR OFF-LINE. THESE UNITS SHALL BE DROPPED AFTER A REASONABLE AMOUNT OF TIME (3 MIN ON A 11/70).

2.2 SOFTWARE PARAMETERS

THE FOLLOWING QUESTIONS ARE ASKED IF REQUESTED ON A START, RESTART, OR CONTINUE. THEY ALLOW FLEXABILITY IN THE WAY THE PROGRAM BEHAVES.

```
CLEAR COUNTERS (L) Y ?

RESET RANDOM VARIABLES (L) N ?

PRINT RECOVERABLE ERRORS  (L) N ?

HALT AFTER EACH CMD (L) N ?

INHIBIT RECOVERY (L) N ?

BAD TAPE SPOT DETECTION (L) Y ?

DISABLE INTERRUPTS (L) N ?

INHIBIT RFC ERROR REPORTS (L) N ?

CHANGE CMD SEQUENCE (L) N ?

NOTE:   THIS QUESTION SHOULD BE ANSWERED (N) UNLESS AN
        OPERATOR SELECTED SEQUENCE IS TO BE EXECUTED.
        IF THIS QUESTION WAS ANSWERED (N), NO MORE
        QUESTIONS WILL BE ASKED.  IF THIS QUESTION WAS
        ANSWERED Y, THE FOLLOWING QUESTIONS MUST BE
        ANSWERED OR DEFAULTED WITH A <CR> ONLY:

CHARACTERISTICS CODE (O) 40 ?   (0,20,40,200) (OCTAL)
CMD/2 (D) 13 ?                  (1-27)    (DECIMAL)
BRF COUNT (D) 1 ?               (1-2K)    (DECIMAL)
# OF OPERATIONS (D) 1 ?         (1-32K)   (DECIMAL)
PATTERN (D) 7 ?                 (0-8)     (DECIMAL)
CMD/3 (D) 4 ?                   (1-27)    (DECIMAL)
BRF COUNT (D) 2048 ?            (1-2K)    (DECIMAL)
# OF OPERATIONS (D) 32000 ?     (1-32K)   (DECIMAL)
PATTERN (D) 7 ?                 (0-8)     (DECIMAL)
CMD/4 (D) 3 ?                   (1-27)    (DECIMAL)
BRF COUNT (D) 2048 ?            (1-2K)    (DECIMAL)
# OF OPERATIONS (D) 32000 ?     (1-32K)   (DECIMAL)
PATTERN (D) 7 ?                 (0-8)     (DECIMAL)
CMD/5 (D) 2 ?                   (1-27)    (DECIMAL)
BRF COUNT (D) 2048 ?            (1-2K)    (DECIMAL)
# OF OPERATIONS (D) 32000 ?     (1-32K)   (DECIMAL)
PATTERN (D) 7 ?                 (0-8)     (DECIMAL)
CMD/6 (D) 13. ?                 (1-27)    (DECIMAL)
BRF COUNT (D) 1 ?               (1-2K)    (DECIMAL)
# OF OPERATIONS (D) 1 ?         (1-32K)   (DECIMAL)
PATTERN (D) 7 ?                 (0-8)     (DECIMAL)
CMD/7 (D) 27. ?                 (1-27)    (DECIMAL)
BRF COUNT (D) 2048 ?            (1-2K)    (DECIMAL)
# OF OPERATIONS (D) 32000 ?     (1-32K)   (DECIMAL)
PATTERN (D) 7 ?                 (0-8)     (DECIMAL)
CMD/8 (D) 27. ?                 (1-27)    (DECIMAL)
BRF COUNT (D) 2048 ?            (1-2K)    (DECIMAL)
# OF OPERATIONS (D) 32000 ?     (1-32K)   (DECIMAL)
PATTERN (D) 7 ?                 (0-8)     (DECIMAL)

NOTE:   THE PROGRAM AUTOMATICALLY INSERTS AN CHARACTERISTIC 40
        AS THE FIRST COMMAND IN THE SEQUENCE TABLE.  IF A
```

DIFFERENT CHARACTERISTIC IS DESIRED, THE OPERATOR SHOULD
ENTER THAT CHARACTERISTIC CODE. A TOTAL OF 7 COMMANDS
MAY BE ENTERED IN ADDITION TO THE SET CHARACTERISTICS
COMMAND. IF THE OPFRATOR WISHES TO USE LESS THAN 7
COMMANDS, AN END COMMAND MUST BE ENTERED AND THEN A
CONTROL Z (^Z) CAN BE ENTERED TO TERMINATE SOFTWARE DIALOGUE.

2.2.1 COMMAND LIST FOR USE IN SOFTWARE DIALOGUE.

CODE	COMMAND	DESCRIPTION
1 =	DRI	DRIVE INITIATE.
2 =	RDF	READ FORWARD.
3 =	RDR	READ REVERSE.
4 =	WRT	WRITE.
5 =	WTV	WRITE/VERIFY. IE. WRITE N RECORDS; READ REVERSE AND CHECK N RECORDS OF DATA; READ FORWARD AND CHECK N RECORDS.
6 =	SRF	SPACE RECORDS FORWARD.
7 =	SRR	SPACE RECORDS REVERSE.
8 =	RNR	READ NEXT REVERSE, IE. SPACE FWD, READ REV.
9 =	RNF	READ NEXT FORWARD, IE. READ FWD, SPACE REV.
10 =	RPF	READ PREVIOUS FWD, IE. SPACE REV, READ FWD.
11 =	RPR	READ PREVIOUS REV, IE. READ REV, SPACE FWD.
12 =	WRR	WRITE RETRY.
13 =	RWD	REWIND.
14 =	MBR	MESSAGE BUFFER RELEASE.
15 =	WTM	WRITE TAPE MARK.
16 =	WTR	WRITE TAPE MARK RETRY.
17 =	SFF	SPACE FILES FORWARD.
18 =	SFR	SPACE FILES REVERSE.
19 =	GES	GET EXTENDED STATUS.
20 =	ERS	ERASE 3 INCHES OF TAPE.
21 =	UNL	UNLOAD.
22 =	CLN	CLEAN TAPE
23 =	SCH	SET DEVICE CHARACTERISTIC. WHERE BRF=200, 40, 20, 0. 200 = ENABLE SKIP TAPE MARKS STOP (STOP AT LOGICAL EOT) 40 = ENABLE ATTENTION INTERRUPTS. 20 = ENABLE MESSAGE BUFFER RELEASE INTERRUPTS. SEE TS11/TS04 PROGRAMMING SPECIFICATION FOR DESCRIPTION.
24 =	DIA	DIAGNOSTICS. SEE TS11/TS04 PROGRAMMING SPECIFICATION FOR DESCRIPTION. ODT MUST BE USED TO LOAD DIAGNOSTIC DATA INTO THE WRITE BUFFER BEFORE THIS CMD IS ISSUED.
25 =	JMP	JUMP TO THE NTH COMMAND IN THE COMMAND SEQUENCE TABLE, WHERE N IS DEFINED IN THE BRF FIELD. THE NUMBER OF JUMPS IS ENTERED IN THE # OF OPERATIONS FIELD
26 =	DLY	DELAY "N" MILISECONDS WHERE N IS DEFINED IN THE # OF OPERATIONS.
27 =	END	END OF COMMAND SEQUENCE.

2.2.2 DATA PATTERN LIST FOR USE IN SOFTWARE DIALOGUE.

PATTERN #	DESCRIPTION.
0	INCREMENTING PATTERN. 0 - 377.
1	ALL "1"'S PATTERN.
2	ALL "0"'S PATTERN.
3	"1" BIT WALKING FROM R TO L IN A FIELD OF "0"'S.
4	"0" BIT WALKING FROM R TO L IF A FIELD OF "1"'S.
5	ALTERNATING "1" AND "0" BITS WITH ALTERNATE BYTES COMPLIMENTED. (125/252)
6	ALTERNATING BYTES OF 000 AND 377.
7	RANDOM DATA PATTERN.
8	NO PATTERN GENERATION.

2.3 EXAMPLES OF SOFTWARE DIALOGUE

2.3.1 BASIC FUNCTION AND DATA RELIABILITY WITH ALL ERROR REPORTING ENABLED

A) RECEIVE PROMPT (DR>)
B) ENTER STA/TES:1-2<CR>
C) ANSWER HARDWARE QUESTIONS.
D) PROCEED WITH THE FOLLOWING DIALOGUE:

CHANGE SW (L) ?	Y<CR>
CLEAR COUNTERS (L) N ?	Y<CR>
RESET RANDOM VARIABLES (L) N ?	N<CR>
PRINT RECOVERABLE ERRORS (L) N ?	Y<CR>
HALT AFTER EACH CMD (L) N ?	N<CR>
INHIBIT RECOVERY (L) N ?	N<CR>
BAD TAPE SPOT DETECTION (L) Y ?	Y<CR>
DISABLE INTERRUPTS (L) N ?	N<CR>
INHIBIT RFC ERROR REPORT (L) N ?	N<CR>
CHANGE CMD SEQUENCE (L) N ?	N<CR>

2.3.2 TO SET UP A SCOPE LOOP FOR A FAILURE IN BASIC FUNCTIONS.

A) RECEIVE PROMPT (DR>)
B) ENTER STA/TES:1/FLA:LOE:IER:ISR:IDU<CR>
C) ANSWER HARDWARE QUESTIONS.
D) PROCEED WITH THE FOLLOWING DIALOGUE:

```
CHANGE SW (L) ?                          Y<CR>
CLEAR COUNTERS (L) N ?                   Y<CR>
RESET RANDOM VARIABLES (L) N ?           N<CR>
PRINT RECOVERABLE ERRORS  (L) N ?        N<CR>
HALT AFTER EACH CMD (L) N ?              N<CR>
INHIBIT RECOVERY (L) N ?                 N<CR>
BAD TAPE SPOT DETECTION (L) Y ?          N<CR>
DISABLE INTERRUPTS (L) N ?               N<CR>
INHIBIT RFC ERROR REPORT (L) N ?         Y<CR>
CHANGE CMD SEQUENCE (L) N ?              N<CR>
```

2.3.3 TO SET UP A SCOPE LOOP FOR A FAILURE IN DATA RELIABILITY

A) RECEIVE PROMPT (DR>)
B) ENTER STA/TES:5/FLA:IER:ISR:IDU/EOP:1000<CR>
C) ANSWER HARDWARE QUESTIONS.
D) PROCEED WITH THE FOLLOWING DIALOGUE:

```
CHANGE SW (L) ?                          Y<CR>
CLEAR COUNTERS (L) N ?                   Y<CR>
RESET RANDOM VARIABLES (L) N ?           N<CR>
PRINT RECOVERABLE ERRORS  (L) N ?        N<CR>
HALT AFTER EACH CMD (L) N ?              N<CR>
INHIBIT RECOVERY (L) N ?                 N<CR>
BAD TAPE SPOT DETECTION (L) Y ?          N<CR>
DISABLE INTERRUPTS (L) N ?               Y<CR>
INHIBIT RFC ERROR REPORT (L) N ?         Y<CR>
CHANGE CMD SEQUENCE (L) N ?              Y<CR>
CHARACTERISTICS CODE (O) 40 ?           40<CR>
CMD/2 (D) 5 ?                           13<CR>   (REWIND) (COULD BE ANY COMMAND)
BRF COUNT (D) 2048 ?                     1<CR>
# OF OPERATIONS (D) 10 ?                1<CR>
PATTERN (D) 7 ?                          1<CR>
CMD/3 (D) 5 ?                            4<CR>   (WRITE) (COULD BE ANY COMMAND)
BRF (D) 2048 ?                        1000<CR>
# OF OPERATIONS (D) 10 ?              10000<CR>
PATTERN (D) 7 ?                          1<CR>
CMD/4 (D) 5 ?                           27<CR>   (END)   (COULD BE ANY COMMAND)
BRF (D) 2048 ?                          <^Z>
```

2.4 EXECUTION TIMES

2.4.1 SYSTEM CONFIGURATION

PDP11/34
MOS MEMORY
LA36
TS11/TS04

2.4.2 TEST EXECUTION TIMES

TEST 1 - BASIC FUNCTIONS - 30 SECONDS PER PASS.
TEST 2 - DATA RELIABILITY - 45 MINUTES PER PASS.
TEST 3 - WRITE COMPATABILITY - 20 MINUTES PER PASS.
TEST 4 - READ COMPATABILITY - 20 MINUTES PER PASS.
TEST 5 - OPERATOR SELECTED SEQUENCE - DEPENDS ON SEQUENCE SELECTED.

NOTE: ALL EXECUTION TIMES ARE SHOWN FOR ONE UNIT OEPRATION.
APPROXIMATELY 10% WILL BE ADDED TO ALL EXECUTION TIMES
FOR EACH ADDITIONAL UNIT.

3.0 ERROR INFORMATION

3.1 ERROR REPORTING

ALL ERROR REPORTS EXCEPT FOR ERRORS #1 AND #17 INCLUDE A DUMP
OF THE FOLLOWING INFORMATION:

ERROR #, TEST #, SUBTEST #, PROGRAM COUNTER, UNIT #,
COMMAND, PREVIOUS COMMAND, PASS COUNT, # OF
RECORDS FROM BOT, RECORD READ COUNT, THE COMMAND PACKET, TSSR,
TCC, TSBA, RFC, AND THE EXTENDED STATUS REGISTERS
(SEE 2.3.14.1 FOR LIST OF COMMANDS).

STANDARD ERROR REPORT FORMAT:

```
CZTSH SFT ERR XXXXX  TST XXX  SUB XXX  PC: XXXXXX
(ASCII ERROR MESSAGE)
XXX CMD FAILED - UNIT X  PASS: XXXXX  RECORD: XXXXX
PREVIOUS CMD WAS  XXX          * RECORD READ: XXXXX *
CMDPKT  TSBA    RFC     TSSR   TCC
XXXXXX  XXXXXX  XXXXXX  XXXXXX  X
XXXXXX
XXXXXX
XXXXXX
XST0    XST1    XST2    XST3
XXXXXX  XXXXXX  XXXXXX  XXXXXX
```

* CAUTION *

INTERPRET THAT 'RECORD READ' COUNT WITH CAUTION.
IF VERY DIFFERENT FROM RECORD COUNT TRACKED BY THE DIAGNOSTIC,
POSITION IS NOT NECESSARELY LOST. ERRORS IN READING THAT
RECORD MIGHT HAVE CAUSED RECORD COUNT TO BE ERRONEOUSLY
READ FROM TAPE.
IN TEST 2, IF DIAGNOSTIC IS RESTARTED OR CONTINUED, RECORD COUNT
IS RESET TO ZERO ALTHOUGH TAPE WAS NOT REWOUND. THIS IS
NECESSARY BECAUSE THERE IS NO ACCURATE WAY TO DETERMINE
ON WHAT RECORD COUNT OF WHAT UNIT THE DIAGNOSTIC WAS HALTED
BEFORE RESTARTING OR CONTINUING.
IT IS SUGGESTED THAT A 'PRINT' BE REQUESTED WHEN HALTING DIAG
TO GET A PRINT OF THE RECORD COUNT WHEN HALTED.

EXAMPLE OF AN ERROR REPORT:

```
CZTSH SFT ERR 00009  TST 002  SUB 000  PC: 010606
RECOVERABLE ERROR
WRT CMD FAILED - UNIT 2  PASS:     2  RECORD:   254
PREVIOUS CMD WAS  WRT
CMDPKT  TSBA    RFC     TSSR   TCC
100005  002324  000000  100210  4
051766
000000
000371
XST0    XST1    XST2    XST3
000350  000002  100004  000000
```

3.1.1 ERROR #1 - COMMAND PACKET ADDRESS NOT ON A MODULO 4 BOUNDRY:

IF THIS ERROR IS REPORTED, THE PROGRAM DID NOT LOAD PROPERLY.
THIS IS A SYSTEM FATAL ERROR AND THE PROGRAM MUST BE RELOADED TO
CORRECT IT.

3.1.2 ERROR #2 - TS04 NOT READY:

BEFORE ANY COMMAND IS ISSUED TO THE TS04, THE SUBSYSTEM READY
BIT IN THE TSS4 IS CHECKED. IF THE SSR IS NOT SET, THE PROGRAM
REPORTS THE NOT READY ERROR. THIS IS A FATAL DEVICE ERROR AND
THE DEVICE WILL BE DROPPED FROM THE TEST SEQUENCE UNLESS THE IDU
OPTION IS USED.

3.1.3 ERROR #3 - NO RESPONSE ERROR:

ONCE THE TSDB IS LOADED, THE TS04 HAS ONE MILLISECOND TO RESPOND
OR THE PROGRAM REPORTS A NO RESPONSE ERROR. THIS IS A FATAL
DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST SEQUENCE
UNLESS THE IDU OPTION IS USED.

3.1.4 ERROR #4 - NO INTERRUPT ERROR:

COMMAND WAS ISSUED AND NO INTERRUPT RECEIVED. THE PROGRAM REPORTS
THAT NO INTERRUPT OCCURRED. THIS IS A FATAL DEVICE ERROR AND
THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU
OPTION IS USED.

3.1.5 SPECIAL CONDITION ERRORS:

IF, DURING EXECUTION, AN INCIDENT OCCURS FORCING THE TSSR
SPECIAL CONDITION BIT TO SET, THE PROGRAM WILL SELECT ONE OF
8 ERROR HANDLING ROUTINES, DEPENDING ON THE TERMINATION CLASS
CODE.

THE TERMINATION CLASS CODES IN THE TSSR ARE PROCESSED AS FOLLOWS
WHEN SPECIAL CONDITION IS SET:

3.1.5.1 ERROR #5 - TERMINATION CLASS CODE 0, UNDEFINED SPECIAL CONDITION

THE ERROR IS REPORTED, A HARD ERROR IS LOGGED
AND THE PROGRAM PROCEEDS NORMALLY.

3.1.5.2 ERROR #6 - TERMINATION CLASS CODE 1, ATTENTION CONDITION

THIS TCC INDICATES THAT THE DRIVE HAS UNDERGONE A STATUS CHANGE
SUCH AS GOING OFFLINE OR COMING ONLINE. THIS IS A FATAL DEVICE
ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS
THE IDU OPTION IS USED.

3.1.5.3 ERROR #7 - TERMINATION CLASS CODE 2, TAPE STATUS ALERT

A STATUS CONDITION HAS BEEN ENCOUNTERED THAT MAY HAVE SIGNIFICANCE
TO THE PROGRAM. BITS OF INTEREST INCLUDE TMK, RLS, LET, RLL, EOT.
ACTION TAKEN DEPENDS ON THE TEST BEING EXECUTED.
IF THE CONDITION IS UNEXPECTED, THE ERROR IS REPORTED AND

A HARD ERROR IS LOGGED. THE PROGRAM PROCEEDS NORMALLY.

3.1.5.4 ERROR #8 - TERMINATION CLASS CODE 3, FUNCTION REJECT

THE SPECIFIED FUNCTION WAS NOT INITIATED. BITS OF INTEREST ARE RMR, OFL, VCK, BOT, ILC, WLE, ILA, AND NBA. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

31.5.5 ERROR #9 - TERMINATION CLASS CODE 4, RECOVERABLE ERROR

TAPE POSITION IS ONE RECORD BEYOND WHAT ITS POSITION WAS WHEN THE FUNCTION WAS INITIATED. RECOVERY PROCEDURE IS TO LOG THE ERROR AND ISSUE THE APPROPRIATE RETRY COMMAND. IF RETRY LIMIT IS REACHED BEFORE THE ERROR IS RECOVERED, RETRY LIMIT EXCEEDED IS REPORTED AS DESCRIBED IN ERROR #14 BELOW.

3.1.5.6 ERROR #10 - TERMINATION CLASS CODE 5, RECOVERABLE ERROR

TAPE POSITION HAS NOT CHANGED. RECOVERY PROCEDURE 'S TO LOG THE ERROR AND RE-ISSUE THE ORIGINAL COMMAND. IF RETRY LIMIT IS REACHED BEFORE THE ERROR IS RECOVERED, RETRY LIMIT EXCEEDED IS REPORTED AS DESCRIBED IN ERROR #14 BELOW.

3.1.5.7 ERROR #11 - TERMINATION CLASS CODE 6, UNRECOVERABLE ERROR

TAPE POSITION HAS BEEN LOST. THE ONLY VALID RECOVERY PROCEDURE IS TO REWIND AND START OVER AT BOT UNLESS THE TAPE HAS LABELS OR SEQUENCE NUMBERS. IF DENSITY CHECK IS SET THIS DIAGNOSTIC WILL REWIND AND RETRY THE COMMAND, OTHERWISE THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.5.8 ERROR #12 - TERMINATION CLASS CODE 7, FATAL SUBSYSTEM ERROR

THE SUBSYSTEM IS INCAPABLE OF PROPERLY PERFORMING COMMANDS OR AT LEAST ITS INTEGRITY IS SERIOUSLY QUESTIONABLE. REFER TO THE FATAL CLASS CODE FIELD IN THE TSSR REGISTER FOR ADDITIONAL INFORMATION ON THE TYPE OF FATAL ERROR. THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.6 ERROR #13 - RFC NON-ZERO ERROR:

IF, AFTER EXECUTION, THE RESIDUAL FRAME COUNT IS NON-ZERO, THE ERROR IS REPORTED AND A HARD ERROR IS LOGGED. THE PROGRAM THEN PROCEEDS NORMALLY. THE REPORTING AND LOGGING OF THESE ERRORS IS OPTIONAL.

3.1.7 ERROR #14 - RETRY LIMIT EXCEEDED:

ON A WRITE COMMAND THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

ON A READ COMMAND THIS ERROR IS LOGGED AS A HARD ERROR AND THE PROGRAM PROCEEDS NORMALLY.

3.1.8 ERROR #15 - TOO MANY INTERRUPTS:

IF MORE THAN ONE INTERRUPT OCCURS PER COMMAND, THIS ERROR IS REPORTED.
THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM
THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.9 ERROR #16 - CAPSTAN RUNAWAY:

CAPSTAN DID NOT STOP WITHIN ACCEPTABLE WINDOW AFTER LAST
COMMAND. THE PROGRAM WILL ISSUE A GET STATUS COMMAND BEFORE REPORTING
THE ERROR SO THAT THE DEAD TRACK FIELD IN EXTENDED STATUS REGISTER 2
WILL CONTAIN THE TACH COUNT WHEN THE TAPE STOPPED.
THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM
THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.10 ERROR #17 - DATA COMPARE ERROR:

IF A DATA VALIDATION ERROR OCCURS DURING A WRITE/VERIFY COMMAND,
THE PROGRAM PRINTS WHAT THE DATA SHOULD HAVE BEEN AND WHAT THE
DATA WAS, AND PRINTS THE BYTE AND RECORD NUMBER THE ERROR OCCURRED
ON. ONLY THE FIRST 10 BYTES IN ERROR PER RECORD ARE PRINTED.
THE TOTAL # OF BYTES IN ERROR PER RECORD IS ALSO PRINTED. A
HARD ERROR IS LOGGED AND THE PROGRAM PROCEEDS NORMALLY.

3.2 ERROR HALTS

ERROR HALTS ARE SUPPORTED PER DESCRIBED IN THE PREVIOUS SECTION
WITH /FLAG:HOE. THERE ARE NO OTHER HALTS.

4.0 PERFORMANCE REPORT

UNIT X PASS:XXXXX RECORD:XXXXX
BYTES WRITTEN XXX,XXX,XXX,XXX
BYTES READ REV XXX,XXX,XXX,XXX
BYTES READ FWD XXX,XXX,XXX,XXX

	WRT	RDR	RDF
RECOVERABLE ERRORS	XXXXX	XXXXX	XXXXX
UNRECOVERABLE ERRORS	XXXXX	XXXXX	XXXXX

SPEC COND	HARD	FATAL	COMPARE
XXXXX	XXXXX	XXXXX	XXXXX

5.0 TEST SUMMARIES

5.1 TEST 1 - BASIC FUNCTIONS.

EXECUTES AND VERIFIES CORRECT COMPLETION OF ALL TS04 FUNCTIONS.

SUBTEST 1 - SET CHAR, DRIVE INIT, GET STATUS.
+ SET CHARACTERISTIC 200.
+ DRIVE INITIATE.
+ SET CHARACTERISTIC 20.
+ GET STATUS
+ SET CHARACTERISTIC 40.
+ PRINT TS04 MICROCODE LEVEL (PASS 1 ONLY)

SUBTEST 2 - REWIND.
+ REWIND.
+ REWIND AT BOT.

SUBTEST 3 - WRITE/VERIFY.
+ WRITE/VERIFY PATTERN 1.
+ WRITE/VERIFY PATTERN 2.
+ WRITE/VERIFY PATTERN 3.
+ WRITE/VERIFY PATTERN 4.
+ WRITE/VERIFY PATTERN 5.
+ WRITE/VERIFY PATTERN 6.
+ WRITE/VERIFY PATTERN 0.

SUBTEST 4 - WRITE TAPE MARK, ERASE.
+ WRITE TAPE MARK.
+ WRITE 10 RECORDS
+ ERASE 10 TIMES
+ WRITE TAPE MARK.
+ WRITE TAPE MARK RETRY.

SUBTEST 5 - SPACE FILES.
+ SPACE 2 FILES REVERSE.
+ SPACE 2 FILES FORWARD.
+ SPACE 2 FILES REVERSE.
+ SPACE 2 FILES FORWARD.

SUBTEST 6 - SPACE RECORDS.
+ REWIND.
+ SPACE 7 RECORDS FORWARD.
+ SPACE 7 RECORDS REVERSE.

```
                    + SPACE 7 RECORDS FORWARD.
                    + SPACE 7 RECORDS REVERSE.

SUBTEST 7  - WRITE RETRY.
                    + REWIND.
                    + WRITE DATA.
                    + WRITE RETRY.

SUBTEST 8  - READ REV RETRY.
                    + READ REVERSE.
                    + READ NEXT REVERSE.
                    + READ NEXT FORWARD.

SUBTEST 9  - READ FWD RETRY.
                    + READ FORWARD.
                    + READ PREVIOUS FORWARD.
                    + READ PREVIOUS REVERSE.

SUBTEST 10 - CLEAN.
                    + CLEAN.
                    + REWIND.

SUBTEST 11  - WRITE/VERIFY SWAPPED DATA BYTES.
                    + WRITE/VERIFY EVEN LENGTH (RECORD 1).
                    + WRITE/VERIFY ODD LENGTH (RECORD 2).
                    + SET DATA BYTE SWAP.
                    + WRITE/VERIFY EVEN LENGTH (RECORD 3).
                    + WRITE/VERIFY ODD LENGTH (RECORD 4).
                    + CLEAR DATA BYTE SWAP.

SUBTEST 12  - READ SWAPPED DATA BYTES.
                    + READ REV RECORD 4.
                    + READ REV RECORD 3.
                    + SET DATA BYTE SWAP.
                    + READ REV RECORD 2.
                    + READ REV RECORD 1.
                    + READ FWD RECORD 1.
                    + READ FWD RECORD 2.
                    + CLEAR DATA BYTE SWAP.
                    + READ FWD RECORD 3.
                    + READ FWD RECORD 4.
```

5.2 TEST 2 - DATA RELIABILITY.

1. THE TAPE IS INITIATED WITH THE FOLLOWING COMMANDS:
   SET CHARACTERISTIC 40
   REWIND
   WRITE/VERIFY 31 RECORDS OF RANDOM LENGTH AND DATA
2. WRITE AND READ COMMANDS ARE SELECTED AT RANDOM AND ARE EXECUTED A RANDOM NUMBER OF TIMES WITH RANDOM LENGTHS AND RANDOM PATTERN UNTIL END OF TAPE IS REACHED.
3. AT THE END OF EACH PASS, A REWIND COMMAND IS ISSUED AND A PERFORMANCE REPORT IS PRINTED.

NOTE: IF A RESTART COMMAND IS USED TO INITIATE TEST 1, THE INITIAL REWIND COMMAND IS NOT ISSUED.

5.3 TEST 3 - WRITE COMPATABILITY/WRITE UTILITY.

REWINDS AND WRITES RECORDS OF RANDOM LENGTHS AND RANDOM DATA FROM BOT TO EOT.

5.4 TEST 4 - READ COMPATABILITY/READ UTILITY.

REWINDS AND READS ENTIRE TAPE, FORWARD AND REVERSE.

5.5 TEST 5 - EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.

THE SEQUENCE OF COMMANDS ENTERED BY THE OPERATOR IS EXECUTED. IF NO COMMANDS WERE ENTERED, A DEFAULT SEQUENCE OF REWIND/WRITE/READ REV/READ FWD/REWIND OF ENTIRE TAPE IS EXECUTED WITH RANDOM PATTERN AND RECORD LENGTH OF 2048 BYTES.

6.0 DEVICE INFORMATION TABLES

6.1 GENERAL

THE TS04 TAPE SUBSYSTEM CONSISTS OF A TS11 UNIBUS TO SERIAL BUS CONTROLLER CONNECTED TO A TS04 DRIVE. FROM A SOFTWARE VIEWPOINT THIS CONFIGURATION IS UNIQUE (FOR A UNIBUS DEVICE) IN A NUMBER OF WAYS:

A. ONLY ONE REGISTER MAY BE WRITTEN - TSDB (TAPE SYSTEM DATA BUFFER),

B. TWO REGISTERS MAY BE READ - TSSR AND TSBA (TAPE SYSTEM STATUS REGISTER AND TAPE SYSTEM BUS ADDRESS REGISTER),

C. COMMANDS ARE NOT WRITTEN TO THE DRIVE; RATHER, COMMAND POINTERS ARE WRITTEN WHICH POINT TO COMMAND PACKETS SOMEWHERE IN CPU MEMORY. THE COMMAND POINTER IS USED BY THE TS04 SUBSYSTEM TO FETCH THE WORD(S) WITHIN THE COMMAND PACKET. THE WORDS WITHIN THE COMMAND PACKET ARE:

1. COMMAND WORD
2. LOW ORDER BUFFER ADDRESS
3. HIGH ORDER BUFFER ADDRESS
4. BYTE COUN

D. THE TSSR CONTAINS ALL THE INFORMATION WHICH WILL BE NECESSARY TO DETERMINE WHETHER:
   1. THE DRIVE IS READY TO ACCEPT ANOTHER COMMAND,
   2. THE PREVIOUS COMMAND WAS EXECUTED WITHOUT ERROR.

   IF EITHER OF THE ABOVE CONDITIONS IS UNTRUE AT "JOB DONE" OR "COMMAND INITIATION" TIME, IT MAY BE NECESSARY TO GET THE EXTENDED STATUS REGISTERS TO DETERMINE WHAT ACTION IS TO BE TAKEN AND/OR LOG THE ERROR INFORMATION.

E. EXTENDED STATUS REGISTERS ARE NOT READ DIRECTLY FROM DRIVE REGISTERS; RATHER, A "GET STATUS" COMMAND IS ISSUED WHICH WILL CAUSE THE TS04 TO TRANSFER EXTENDED STATUS INFORMATION TO THE MEMORY AREA POINTED TO BY THE BUFFER ADDRESS OF THE "GET STATUS" COMMAND. THERE ARE FOUR EXTENDED STATUS REGISTERS. SEE 6.3.

F. THE TSDB MUST BE WRITTEN WITH A DATO INSTRUCTION TO PROPERLY WRITE THE COMMAND POINTER. A DATOB WILL CAUSE A MAINTENANCE FUNCTION. A DATO TO THE TSSR WILL CAUSE SUBSYSTEM INIT.

G. COMMAND PACKETS MUST RESIDE ON DIVIDE BY FOUR MEMORY BOUNDARIES (AS OPPOSED TO DIVIDE BY 2 OR WORD BOUNDARIES) .

6.2 UNIBUS INTERFACE SPECIFICATIONS

TS11/ TS04	INT. VECTOR	UNIBUS ADDRESS	REGISTER
FIRST	224	772520	TSBA/TSDB
		772522	TSSR
SECOND	154	772524	TSBA/TSDB
		772526	TSSR
THIRD	160	772530	TSBA/TSDB
		772532	TSSR
FOURTH	164	772534	TSBA/TSDB
		772536	TSSR

6.3 BIT DEFINITIONS FOR TS11/TS04 REGISTERS

6.3.1 TS11/TS04 REGISTER SUMMARY

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
TSBA	A15	A14	A13	A12	A11	A10	A09	A08	A07	A06	A05	A04	A03	A02	A01	A00
TSDB	P15	P14	P13	P12	P11	P10	P09	P08	P07	P06	P05	P04	P03	P02	P17	P16
TSSR	SC	UPE	SPE	RMR	NXM	NBA	A17	A16	SSR	OFL	FC1	FC0	TC2	TC1	TC0	
XST0	TMK	RLS	LET	RLL	WLE	NEF	ILC	ILA	MOT	ONL	IE	VCK	PED	WLK	BCT	EOT
XST1	DLT			COR	CRS	TIG	DBF NZO	SCK		IPR	SYN DRP	IPO ITM	IED LCO	POS NZN	POL LRC	UNC CRC	MTE VPE
XST2	OPM	SIP	BPE	CAF			WCF		DTP	DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0
XST3	MICRO DIAGNOSTIC ERROR CODE									LMX	OPI	REV	CRF	DCK	NOI	LXS	RIB

TERMINATION CLASS CODES (TSSR TC0-TC2):

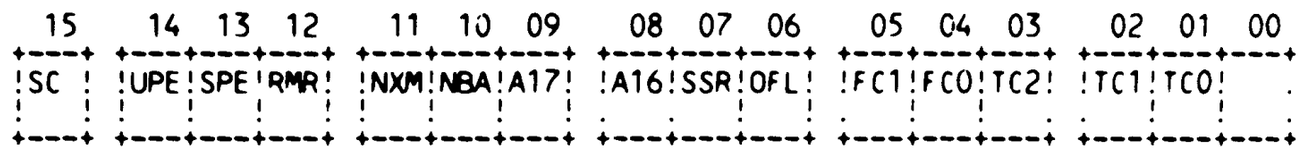
0 = NORMAL TERMINATION
1 = ATTENTION CONDITION
2 = TAPE STATUS ALERT
3 = FUNCTION REJECT
4 = RECOVERABLE ERROR - TAPE POSITION = ONE RECORD DOWN TAPE FROM START OF FUNCTION
5 - RECOVERABLE ERROR - TAPE NOT MOVED
6 = UNRECOVERABLE ERROR - TAPE POSITION LOST
7 = FATAL CONTROLLER ERROR

FATAL CLASS CODES (TSSR FC0-FC1):

0 = MICRO DIAGNOSTIC FAILURE (DISPLAYED IN TS04 OPERATOR PANEL AND XST3).
1 = I/O SEQUENCER CROM PARITY ERROR.
2 = MICROPROCESSOR CROM PARITY ERROR.
SILO PARITY ERROR.
SERIAL BUS PARITY ERROR DETECTED AT TS11 (SPE).
SERIAL BUS PARITY ERROR DETECTED AT TS04 (BPE).
FATAL ERROR HALTS 1750-1777 IN TS04 PROGRAM COUNTER DISPLAY.
3 - LOSS OF AC POWER HAS BEEN DETECTED.

6.3.2 TS11 STATUS REGISTER (TSSR)

UNIBUS ADDRESS + 2 - READ ONLY



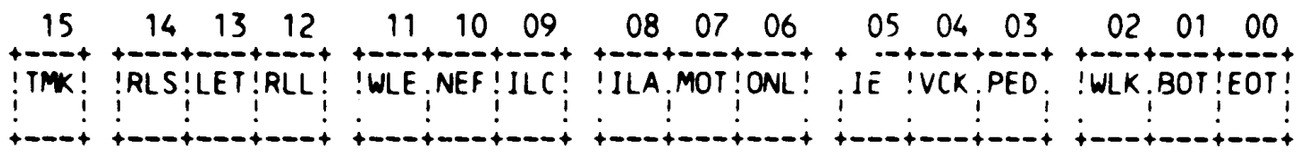
BIT	NAME	TCC	DEFINITION
15	SC	S	SPECIAL CONDITION. WHEN SET, INDICATES THAT THE LAST COMMAND DID NOT COMPLETE WITHOUT INCIDENT. SPECIFICALLY, EITHER AN ERROR WAS DETECTED OR AN EXCEPTION CONDITION OCCURRED. EXCEPTION CONDITIONS CAN BE TAPE MARKS ON READ COMMANDS, REVERSE MOTION AND AT BOT, EOT WHILE WRITING, ETC. MAY ALSO BE SET BY THE ERROR BITS CONTAINED IN THE TSSR REGISTER: UPE, SPE, RMR, AND NXM. THE TERMINATION CLASS BITS ARE SOMETHING OTHER THAN 0 (UNLESS RMR IS THE ONLY ERROR - SEE RMR).
14	UPE	4/5	UNIBUS PARITY ERROR. SET BY THE TS11 WHEN IT DETECTS A PARITY ERROR ON THE UNIBUS DATA WHEN TRANSFERRING TO OR FROM THE CPU'S MEMORY.
13	SPE	7	SERIAL BUS PARITY ERROR. THIS BIT IS SET BY THE TS11 WHEN IT DETECTS A SERIAL BUS PARITY ERROR ON DATA RECEIVED FROM THE TS04.
12	RMR	S	REGISTER MODIFICATION REFUSED. SET BY THE TS11 WHEN A COMMAND POINTER IS LOADED INTO TSDB AND SUB-SYSTEM READY (SSR) IS NOT SET. NOTE THAT THIS BIT CAUSES SPECIAL CONDITION BUT NO TERMINATION CLASS (IN FACT, THE TS04 NEVER SEES THIS ERROR) BECAUSE ON A SYSTEM WITH NO BUGS, THIS BIT MAY COME UP ON AN ATTENTION MESSAGE. IF ATTNS ARE NOT ENABLED, THIS BIT COMING UP IS AN INDICATION OF EITHER A FATAL CONTROLLER ERROR OR A SOFTWARE BUG.
11	NXM	4/5	NON-EXISTENT MEMORY. SET BY THE TS11 WHEN TRYING TO TRANSFER TO OR FROM A MEMORY LOCATION WHICH DOES NOT EXIST. MAY OCCUR WHEN FETCHING THE COMMAND PACKET, FETCHING OR STORING DATA, OR STORING THE MESSAGE PACKET.
10	NBA	S	NEED BUFFER ADDRESS. WHEN SET, INDICATES THAT THE TS04 NEEDS A MESSAGE BUFFER ADDRESS. THIS BIT IS CLEARED DURING THE SET CHARACTERISTICS COMMAND (IF A GOOD ADDRESS WAS GIVEN).
09	A17	S	BUS ADDRESS BIT 17. A17 AND A16 (BIT 08) TRACK THE VALUES OF BITS 17 AND 16 OF THE TSBA RFGISTER.
08	A16	S	BUS ADDRESS BIT 16. E A17 (BIT 09).

07	SSR	S	SUB-SYSTEM READY. WHEN SET, INDICATES THAT THE TS11/TS04 SUBSYSTEM IS NOT BUSY AND IS READY TO ACCEPT A NEW COMMAND POINTER.
06	OFL	S,1,3	OFF-LINE. WHEN SET, INDICATES THAT THE TS04 IS OFF-LINE AND UNAVAILABLE FOR ANY TAPE MOTION COMMANDS. THIS BIT CAN CAUSE A TERMINATION CLASS OF 1 (ON ATTN INTERRUPT) OR 3 (RESULTS IN NEF).
05	FC1	7	FATAL TERMINATION CLASS 01. FC1 AND FC0 (BIT 04) ARE USED TO INDICATE THE TYPE OF FATAL ERROR WHICH HAS OCCURRED ON THE TS04. THESE BITS ARE VALID ONLY WHEN SC IS SET AND THE TERMINATION CLASS CODE BITS ARE ALL SET (111).
04	FC0	7	FATAL TERMINATION CLASS 00. SEE FC1 (BIT 05).
03	TC2	S	TERMINATION CLASS BIT 02. THIS BIT, ALONG WITH THE TC1 AND TC0 BITS, ACT AS AN OFFSET VALUE WHENEVER AN ERROR OR EXCEPTION CONDITION OCCURS ON A COMMAND. EACH OF THE EIGHT POSSIBLE VALUES OF THIS FIELD REPRESENT A PARTICULAR CLASS OF ERRORS OR EXCEPTIONS. THE CONDITIONS IN EACH CLASS HAVE SIMILAR SIGNIFICANCE AND, AS APPLICABLE, RECOVERY PROCEDURES. THE CODE PROVIDED IN THIS FIELD IS EXPECTED TO BE UTILIZED AS AN OFFSET INTO A DISPATCH TABLE FOR HANDLING OF THE CONDITION.
02	TC1	S	TERMINATION CLASS BIT 01. SEE TC2 (BIT 03).
01	TC0	S	TERMINATION CLASS BIT 00. SEE TC2 (BIT 03).
00	-	-	NOT USED.

UNIBUS ADDRESS + 2 - WRITE ONLY

SUBSYSTEM INITIALIZE

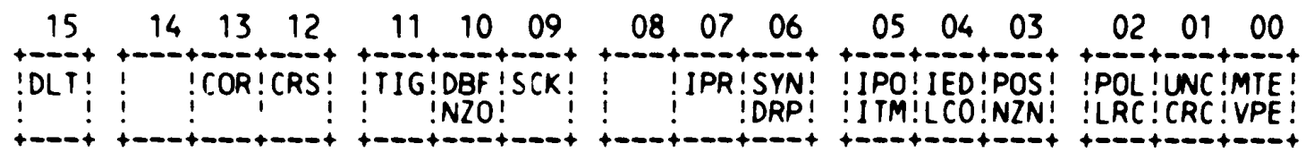
6.3.3 EXTENDED STATUS REGISTER 0 (XSTAT0)



BIT	NAME	TCC	DEFINITION
15	TMK	S,2	TAPE MARK DETECTED. SET WHENEVER A TAPE MARK WAS DETECTED DURING A READ, SPACE, OR SKIP COMMAND AND AS A RESULT OF THE WRITE TAPE MARK OR WITE TAPE MARK RETRY COMMANDS.
14	RLS	2	RECORD LENGTH SHORT. THIS BIT INDICATES THAT EITHER THE RECORD'S LENGTH WAS SHORTER THAN THE BYTE COUNT ON READ OPERATIONS, A SPACE RECORD OPERATION ENCOUNTERED A TAPE MARK OR BOT BEFORE THE POSITION COUNT WAS EXHAUSTED, OR A SKIP TAPE MARKS COMMAND WAS TERMINATED BY ENCOUNTERING BOT OR A DOUBLE TAPE MARK (IF THAT OPERATIONAL MODE IS ENABLED, SEE LET) PRIOR TO EXHAUSTING THE POSITION COUNTER.
13	LET	2	LOGICAL END OF TAPE. SET ONLY ON THE SKIP TAPE MARKS COMMAND WHEN EITHER TWO CONTIGUOUS TAPE MARKS ARE DETECTED OR WHEN MOVING OFF OF BOT AND THE FIRST RECORD ENCOUNTERED IS A TAPE MARK. THE SETTING OF THIS BIT WILL NOT OCCUR UNLESS THIS MODE OF TERMINATION IS ENABLED THROUGH USE OF THE SET CHARACTERISTICS COMMAND.
12	RLL	2	RECORD LENGTH LONG. WHEN SET, THIS BIT INDICATES THAT THE RECORD READ WAS LONGER THAN THE BYTE COUNT SPECIFIED.
11	WLE	3,6	WRITE LOCK ERROR. WHEN SET, INDICATES THAT A WRITE OPERATION WAS ISSUED BUT THE MOUNTED TAPE DID NOT CONTAIN A WRITE ENABLE RING OR THE WRT LOCK SWITCH ACTIVATED DURING THE OPERATION.
10	NEF	3	NON-EXECUTABLE FUNCTION. WHEN SET, INDICATES THAT THE COMMAND COULD NOT BE EXECUTED DUE TO ONE OF THE FOLLOWING CONDITIONS: - THE COMMAND SPECIFIED REVERSE TAPE DIRECTION BUT THE TAPE WAS ALREADY POSITIONED AT BOT. - THE ISSUING OF ANY COMMAND, EXCEPT REWIND,

			UNLOAD, OR A COMMAND WITH THE CLEAR VOLUME CHECK (CVC) BIT SET, WHEN THE VOLUME CHECK BIT IS SET. (I 3) - ANY COMMAND, EXCEPT GET STATUS OR DRIVE INITIALIZE, WHEN THE TS04 IS OFF-LINE. - ANY WRITE COMMAND WHEN THE TAPE DOES NOT CONTAIN A WRITE ENABLE RING (WRITE LOCK STATUS - WLS).
09	ILC	3	ILLEGAL COMMAND. SET WHEN A COMMAND IS ISSUED AND EITHER ITS COMMAND FIELD OR ITS COMMAND MODE FIELD CONTAINS CODES WHICH ARE NOT SUPPORTED BY THE TS04.
08	ILA	3	ILLEGAL ADDRESS. (MORE THAN 18 BITS OR ODD WHEN AN EVEN ADDRESS IS REQUIRED.)
07	MOT	S	TAPE IS MOVING.
06	ONL	S	ON LINE. WHEN SET, INDICATES THAT THE TS04 IS ON-LINE AND OPERABLE.
05	IE	S	INTERRUPT ENABLE. REFLECTS THE STATE OF THE INTERRUPT ENABLE BIT SUPPLIED ON THE LAST COMMAND.
04	VCK	S	VOLUME CHECK. WHEN SET, INDICATES THAT THE DRIVE HAS BEEN EITHER POWERED DOWN OR TURNED OFF-LINE. CLEARED BY THE CLEAR VOLUME CHECK (CVC) BIT IN THE COMMAND HEADER WORD. THIS BIT CAN CAUSE A TERMINATION CLASS OF 3.
03	PED	S	PHASE ENCODED DRIVE. WHEN SET, INDICATES THAT THE TS04 IS CAPABLE OF READING AND WRITING ONLY 1600 BPI PHASE ENCODED DATA. WHEN RESET, INDICATES THAT THE TS04 HAS ONLY 800 BPI NRZI DATA CAPABILITIES.
02	WLK	S,3	WRITE LOCKED. WHEN SET, INDICATES THAT THE MOUNTED REEL OF TAPE DOES NOT HAVE A WRITE-ENABLE RING INSTALLED. THE TAPE IS, THEREFORE, WRITE PROTECTED.
01	BOT	S,3	BEGINNING OF TAPE. WHEN SET, INDICATES THAT THE TAPE IS POSITIONED AT THE LOAD POINT AS DENOTED BY THE BOT REFLECTIVE STRIP ON THE TAPE.
00	EOT	S,2	END OF TAPE. THIS BIT IS SET WHENEVER THE TAPE IS POSITIONED AT OR BEYOND THE END OF TAPE REFLECTIVE STRIP. DOES NOT RESET UNTIL THE TAPE PASSES OVER THE REFLECTIVE STRIP IN THE REVERSE DIRECTION UNDER PROGRAM CONTROL.

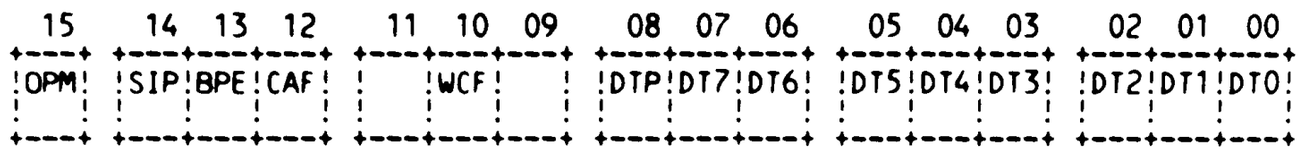
6.3.4 EXTENDED STATUS REGISTER 1 (XSTAT1)



BIT	NAME	TCC	DEFINITION
15	DLT	4	DATA LATE. SET WHEN THE I/O SILO IS FULL ON A READ OR EMPTY ON A WRITE. THESE CONDITIONS OCCUR WHENEVER THE UNIBUS LATENCY EXCEEDS THE DATA TRANSFER RATE OF THE TS04.
14	-	-	NOT USED.
13	COR	S	CORRECTABLE DATA. IN PHASE ENCODED MODE, A CORRECTABLE DATA ERROR HAS BEEN ENCOUNTERED.
12	CRS	4	CREASE DETECTED. FOR NRZI, ALL DATA TRACKS DROPPED OUT FOR MORE THAN THREE CHARACTER TIMES BUT FOR LESS THAN .1 INCHES OF TAPE. FOR PE, EIGHT OUT OF NINE DATA TRACKS WENT DEAD FOR LESS THAN .1 INCHES BEFORE A VALID POSTAMBLE WAS DETECTED.
11	TIG	4	TRASH IN THE GAP. NON-ERASED DATA WAS DETECTED IN A GAP DURING A READ, WRITE, WRITE TAPE MARK, OR ERASE COMMAND.
10	DBF	4	DESKEW BUFFER FAIL. ONE OF THE DESKEW BUFFERS FAILED TO ASSERT 'OUTPUT READY' WITHIN 20 MICROSECONDS AFTER BEING ENABLED. THE DEAD TRACK BITS WILL INDICATE ON WHICH TRACKS THIS FAILURE OCCURRED. THIS ERROR IS PROBABLY A RESULT OF A BROKEN FORMATTER.
	NZO	4	NRZ FIFO OVERRUN.
09	SCK	4	SPEED CHECK. TAPE SPEED WAS OFF BY MORE THAN 5% DURING A WRITE DATA OPERATION. NOTE THAT SPEED AVERAGED OVER 8 TICKS AND THE AVERAGE MUST BE OFF 5% TO CAUSE THIS ERROR.
08	-	-	NOT USED.
07	IPR	S,4	INVALID PREAMBLE. SET ON A PE DRIVE IF THE PREAMBLE APPEARS TO BE SHORTER THAN 36 CHARACTERS OR LONGER THAN 44 CHARACTERS. ALSO SET IF THE PREAMBLE IS INCORRECTLY ENCODED BEYOND THE FIFTEENTH CHARACTER IN READ OR THE TENTH CHARCTER IN READ-AFTER-WRITE.
06	SYN	4	SYNCH FAILURE. SET ON A PE DRIVE IF THE FORMATTER WAS UNABLE TO ACHIEVE SYNCHRONIZATION IN THE PREAMBLE.
	DRP	4	NRZ RECORD DROPPED A CHARACTER (THE NEXT CHARACTER

			WAS TO BE CONSIDERED CRC).
05	IPO	S,4	INVALID POSTAMBLE. SET ON A PE DRIVE DURING READ OR WRITE IF ANY OF THE FIRST 39 CHARACTERS OF THE POSTAMBLE ARE NOT READ CORRECTLY.
	ITM	S,4	ILLEGAL TAPE MARK FOR NRZ.
04	IED	4	INVALID END DATA. FOR PE, EIGHT OUT OF NINE TRACKS WENT DEAD BEFORE THE POSTAMBLE WAS DETECTED.
	LRO	4	FOR NRZI, DATA WAS NOT DETECTED IN EITHER THE LRCC OR CRCC WINDOWS. (LRC WAS ZERO)
03	POS	S,4	POSTAMBLE SHORT. SET ON PE DRIVES DURING A READ OR WRITE WHEN LESS THAN 38 ALL-ZEROES CHARACTERS ARE READ FOLLOWING THE ALL-ONES CHARACTER.
	NZN	S,4	NRZ NOISE RECORD (FEWER THAN 13(10) FRAMES).
02	POL	4	POSTAMBLE LONG. SET ON PE DRIVES DURING READ OR WRITE OPERATIONS WHEN THE POSTAMBLE EXCEEDS 42 CHARACTERS.
	LRC	4	LRC ERROR. SET ON NRZI DRIVES WHEN THE LRCC CHARACTER WAS FOUND IN ERROR.
01	UNC	4	UNCORRECTABLE DATA. SET ON PE DRIVES WHEN A PARITY ERROR OCCURRED WITHOUT A CORRESPONDING DEAD TRACK INDICATION.
	CRC	4	CRC ERROR. SET ON NRZI DRIVES WHEN THE CRC CHARACTER WAS FOUND TO BE IN ERROR.
00	MTE	4	MULTI-TRACK ERROR. SET ON PE DRIVES WHEN MORE THAN ONE DEAD TRACK OCCURRED IN THE PREAMBLE OR IN THE DATA FIELD.
	VPE	4	VERTICAL PARITY ERROR. SET ON NRZI DRIVES WHEN A CHARACTER DID NOT CONTAIN AN ODD NUMBER OF ONE BITS.

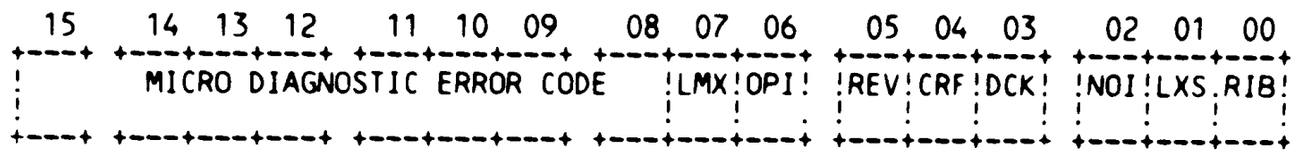
6.3.5 EXTENDED STATUS REGISTER 2 (XSTAT2)



BIT	NAME	TCC	DEFINITION
15	OPM	S	OPERATION IN PROGRESS. (TAPE MOVING)
14	SIP	7,F2	SILO PARITY ERROR. CAUSES FATAL CLASS 2 BECAUSE THE ERROR MIGHT HAVE OCCURRED DURING THE TRANSMISSION OF THE MESSAGE PACKET.
13	BPE	7,F2	SERIAL BUS PARITY ERROR AT DRIVE. SET BY THE TS04 WHEN A PARITY ERROR IS DETECTED ON DATA TRANSMITTED FROM THE TS11 TO THE TS04. CAUSES FATAL CLASS 2 BECAUSE THE ERROR MIGHT HAVE OCCURRED DURING THE TRANSMISSION OF THE MESSAGE PACKET.
12	CAF	7	CAPSTAN ACCELERATION FAIL. AFTER ACCELERATING TAPE FOR .2 INCHES, THE TAPE SPEED WAS CHECKED AND FOUND TO BE OUT OF TOLERANCE BY MORE THAN 10%.
11	-	-	NOT USED.
10	WCF	7	THE WRITE BOARD IS NOT EMPTYING THE I/O SILO AT THE PROPER RATE. THIS ERROR CAN BE THE RESULT OF THE WRITE BOARD CLOCK NOT BEING TURNED ON (BROKEN HARDWARE).
09	-	-	NOT USED.
08	DTP	S	DEAD TRACK PARITY. THE BITS DTP THROUGH DT0 INDICATE WHICH TRACK(S) WENT DEAD, IF ANY, DURING THE LAST DATA TRANSFER OPERATION. IF DESKEW BUFFER FAIL (DBF) IS SET, THESE BITS INDICATE WHICH CHANNEL FAILED.
07	DT7	S	DEAD TRACK 7. SEE DTP.
06	DT6	S	DEAD TRACK 6. SEE DTP.
05	DT5	S	DEAD TRACK 5. SEE DTP.
04	DT4	S	DEAD TRACK 4. SEE DTP.
03	DT3	S	DEAD TRACK 3. SEE DTP.
02	DT2	S	DEAD TPACK 2. SEE DTP.
01	DT1	S	DEAD TRACK 1. SEE DTP.
00	DT0	S	DEAD TRACK 0. SEE DTP.

NOTE: ON A SET CHARACTERISTICS COMMAND, THE UCODE LEVEL IS RETURNED IN DT7 THRU DT0. ON A GET STATUS COMMAND, THE RESIDUAL CAPSTAN TICK COUNT (INTERNALLY R7) IS RETURNED IN DT7 THRU DT0.

6.3.6 EXTENDED STATUS REGISTER 3 (XSTAT3)



BIT	NAME	TCC	DEFINITION
15 TO 08			MICRO DIAGNOSTIC ERROR CODE. (SEE LIST OF CODES BELOW). ALL ERROR CODES IN THE TABLE WILL BE DISPLAYED ON THE TS04 CONTROL PANEL BUT ONLY CODES HIGHER THAN 110 WILL BE AVAILABLE TO CPU DIAGNOSTICS FOR PRINTOUT IN THE MICRO DIAGNOSTIC ERROR CODE FIELD OF XSTAT3. THIS ERROR CODE FIELD IS VALID ONLY WHEN THE TERMINATION CLASS CODE IN THE TSSR EQUALS 7 AND THE FATAL CLASS CODE IN THE TSSR EQUALS 0, INDICATING AN INTERNAL DIAGNOSTIC FAILURE.
07	NTL	6	LIMIT EXCEEDED. SET WHEN THE TAPE TENSION ARMS HAVE EXCEEDED THEIR ALLOWABLE TRAVEL AND HAVE CAUSED THE ACTIVITATION OF THE LIMIT SWITCHES. NO TENSION EXISTS ON THE MOUNTED TAPE.
06	OPI	6	OPERATION INCOMPLETE. SET WHEN A READ, SPACE, OR SKIP OPERATION HAS MOVED 25 FEET OF TAPE WITHOUT DETECTING ANY DATA ON THE TAPE.
05	REV	S	DIRECTION OF CURRENT OPERATION WAS REVERSE (BUT IS 0 IF REWIND OR FORWARD)
04	CRF	7	CAPSTAN RESPONSE FAILURE. A MOTION COMMAND WAS GIVEN TO THE CAPSTAN BUT WE DID NOT GET A TICK BACK WITHIN A REASONABLE AMOUNT OF TIME.
03	DCK	S,6	DENSITY CHECK. SET ON PE DRIVES WHEN A PE IDENTIFICATION BURST WAS NOT DETECTED WHEN MOVING OFF OF BOT. SET ON NRZI DRIVES WHEN A NON-NRZI IDENTIFICATION BURST WAS FOUND WHEN MOVING OFF OF BOT.
02	NOI	6	NOISE RECORD. SET DURING A READ OR SPACE OPERATION WHEN A BURST OF FLUX CHANGES, WHICH DO NOT QUALIFY AS A RECORD (BUT TOO MANY TO IGNORE), ARE DETECTED: NRZI: AT LEAST TWO CHARACTERS IN A ROW BUT LESS THAN TWELVE, FOLLOWED BY A CHARACTER IN EITHER THE CRCC OR LRCC WINDOWS. PE: AT LEAST 24 CHARACTERS IN A ROW THAT DO NOT QUALIFY AS A TAPE MARK OR A DATA PREAMBLE.
01	LXS	S	LIMIT EXCEEDED STATICALLY. THIS BIT IS SET ANY TIME THE LIMIT SWITCHES ARE EXCEEDED. THIS BIT CAN ONLY BE CLEARED BY MANUALLY LOADING THE TAPE.

COMMAND ALREADY IN PROGRESS HAS ENCOUNTERED THE BOT MARKER WHEN MOVING TAPE IN THE REVERSE DIRECTION. TAPE MOTION WILL BE HALTED AT BOT.

MICRO DIAGNOSTIC ERROR CODES

FOLLOWING IS A LIST OF THE ERRORS WHICH ARE DISPLAYED IN THE MICRO DIAGNOSTIC ERROR CODE (XSTAT3 BITS 15 - 08) AND ALSO IN THE LIGHTS ON THE TS04 CONTROL PANEL, DUE TO FAILURES ON THE CAPSTAN BOARD, I/O BOARDS, WRITE BOARD, READ BOARD, OR FORMATTER BOARD. THE MICRO WILL BE IN A TIGHT LOOP IN THE DISPM PROGRAM, WAITING FOR OPERATOR OR CPU INTERVENTION WHILE THE ERROR IS BEING DISPLAYED IN THE CONSOLE LIGHTS. IT IS APPARENT THAT AN ERROR IS BEING DISPLAYED IF THE "UOK" LIGHT IS NOT LIGHTED, THE PROCESSOR IS NOT STOPPED, AND AN OCTAL NUMBER (100-377) IS BEING DISPLAYED IN THE LIGHTS. TO SCOPE LOOP THESE TESTS, ENTER MAINTENANCE MODE (ON-LINE SWITCH TO "OFF" POSITION, MAINTENANCE SWITCH UP, PRESS RESET), ENTER THE OFF-LINE TEST NUMBER (SEE SCOPE LOOP COLUMN BELOW) IN THE OPERATOR CONSOLE LIGHTS (ENTER ONES WITH LEFT-MOST SWITCH, ENTER ZEROES WITH RIGHT-MOST SWITCH), AND PRESS ON-LINE BUTTON. TEST WILL LOOP UNTIL ON-LINE SWITCH IS RETURNED TO OFF-LINE POSITION, ERRORS WILL BE DISPLAYED CONTINUOUSLY.

ERROR (DISPLAY)	PROGRAM	ERROR DESCRIPTION	LIKELY MODULE	SCOPE LOOP
337	OPERATIONAL CODE	CAPSTAN RUNAWAY ERROR (H3.RNY). CAPSTAN DIDN'T STOP WITHIN ACCEPTABLE WINDOW AFTER LAST COMMAND.		
100	IOTSM	BASIC I/O MICRO FAILURE (PARITY ERROR, IOATN, HANDSHAKING, AND DATA WINDOW TEST BETWEEN THE I/O AND MAIN MICROS. NOTE: CAN ALSO BE CAUSED BY THE SERIAL BUS .SHIN (SHIFT IN) STUCK ASSERTED.	M8967	14
101	IOTSM	ERROR IN I/O CONTROL REGISTER TEST	M8966 M8967	15
102	IOTSM	FAILURE OF FRAME COUNTER TEST	M8966	15
103	IOTSM	FAILURE OF I/O SILO NON-PARITY ERROR DATA TEST OR THE WRITE FLAG.	M8966 M8963	16
104	IOTSM	FAILURE OF I/O SILO PARITY ERROR TEST OR DATA LATE TEST.	M8966	17
105	IOTSM	FAILURE OF SHIFT LOOP WITH ZEROES.	M8965	20
106	IOTSM	FAILURE OF SHIFT LOOP WITH ONES.	M8965	21
107	IOTSM	FAILURE OF SHIFT LENGTH MUX.	M8965	22
110	IOTSM	FAILURE TO RECEIVE CORRECT OP-CODE FROM TS11 WHEN WE SENT DATA OVER THE SERIAL BUS.	M8965 TS11 MOTHER BD	47

			C 4	
			SBUS CABLE	
111	CATSM	FAILURE OF 1 KHZ CLOCK TEST. TSTS TAC SYNC FLOP AND ATTN, TOO.	G159 CBUS CABLE M8963	2
112	CATSM	LIGHT REGISTER CHANGED WHEN MOTION REGISTER WAS CLEARED.	G159	3,4
113	CATSM	FWD OR MVG BITS WRONG AFTER 1 TICK OF SIMULATED COMMAND AND TACH PULSES.	G159	3,4
114	CATSM	FAILURE OF SIMULATED CAPSTAN SPEED TEST. THE CAPSTAN SPEED COUNTER WAS OUT OF RANGE WHEN TAPE MOTION AT SPEED WAS SIMULATED.	G159	3,4
115	CATSM	FAILURE OF SIMULATED SLOW CAPSTAN TEST. SPEED COUNTER DID NOT LATCH UP WITH MAX COUNT WHEN SLOW TACH TICKS WERE SIMULATED.	G159	3,4
116	CATSM	FAILURE OF SIMULATED CAPSTAN DECEL TEST. COUNTER NOT ZERO FOR FORWARD OR 377 FOR REVERSE WHILE DECELERATING, OR MVG BIT NOT 1.	G159	3,4
117	CATSM	FAILURE OF MOVING FLOP TO GO TO ZERO AFTER STOPPING (DIRECTION REVERSAL FOR ONE TACH TICK).	G159	3,4
120	PETSM	FAILURE OF WRITE BOARD TO TURN ON AND EMPTY THE SILO, OR DATA LATE BIT DOESN'T WORK.	M8929 M8966	23
121	PETSM	FAILURE OF WRITE BOARD TO EMPTY SILO AT CORRECT SPEED.	M8929	23
124	PETSM	FORMATTER FLAG DOESN'T WORK ON THE M8922.	M8922	24
125	PETSM	FORMATTER SILO FILLING AND DATA ERROR	M8922 M8923 M8924	24
126	PETSM	PEAK SHIFT TEST ERROR	M8922 M8923 M8924	25
127	PETSM	FORMATTER TABLE LOOKUP ROM CHECKSUM TEST ERROR	M8922 M8923 M8924	26

PARAMETER CODING MACY11 30(1046) 11-OCT-79 14:02 PAGE 1
NLISTF.P11 15-AUG-79 13:01

```
     1
```

```
                        .TITLE PROGRAM HEADER AND TABLES
                        .SBTTL  PROGRAM HEADER

                                .ENABL  AMA
        002000          .       .=      2000
002000                  BGNMOD

                        ;++
                        ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
                        ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
                        ;--

                                POINTER BGNRPT,BGNSW,BGNSFT,BGNAU,BGNDU,BGNSETUP


                                HEADER  CZTSH,C,0,5000,1,#INTPRI
002000                  L$NAME::                ;DIAGNOSTIC NAME
002000  103                                                             .ASCII /C/
002001  132                                                             .ASCII /Z/
002002  124                                                             .ASCII /T/
002003  123                                                             .ASCII /S/
002004  110                                                             .ASCII /H/
002005  000                                                             .BYTE   0
002006  000                                                             .BYTE   0
002007  000                                                             .BYTE   0
002010                  L$REV::                 ;REVISION LEVEL
002010  103                                                             .ASCII  /C/
002011                  L$DEPO::                ;0
002011  060                                                             .ASCII  /0/
002012                  L$UNIT::                ;NUMBER OF UNITS
002012  000001                                                          .WORD   T$PTHV
002014                  L$TIML::                ;LONGEST TEST TIME
002014  005000                                                          .WORD   5000
002016                  L$HPCP::                ;POINTER TO H.W. QUES.
002016  025074                                                          .WORD   L$HARD
002020                  L$SPCP::                ;POINTER TO S.W. QUES.
002020  025146                                                          .WORD   L$SOFT
002022                  L$HPTP::                ;PTR. TO DEF. H.W. PTABLE
002022  002174                                                          .WORD   L$HW
002024                  L$SPTP::                ;PTR. TO S.W. PTABLE
002024  002202                                                          .WORD   L$SW
002026                  L$LADP::                ;DIAG. END ADDRESS
002026  026554                                                          .WORD   L$LAST
002030                  L$STA::                 ;RESERVED FOR APT STATS
002030  000000                                                          .WORD   0
002032                  L$CO::
002032  000000                                                          .WORD   0
002034                  L$DTYP::                ;DIAGNOSTIC TYPE
002034  000001                                                          .WORD   1
002036                  L$APT::                 ;APT EXPANSION
002036  000000                                                          .WORD   0
002040                  L$DTP::                 ;PTR. TO DISPATCH TABLE
002040  002124                                                          .WORD   L$DISPATCH
002042                  L$PRIO::                ;DIAGNOSTIC RUN PRIORITY
002042  000340                                                          .WORD   #INTPRI
002044                  L$EXP1::                ;EXPANSION WORDS
```

```
002044  000000                                                          .WORD   0
002046                  L$EXP2::
002046  000000                                                          .WORD   0
002050                  L$MREV::                ;SVC REV AND EDIT #
002050     003                                                          .BYTE   C$REVISION
002051     002                                                          .BYTE   C$EDIT
002052                  L$EF::                  ;DIAG. EVENT FLAGS
002052  000000                                                          .WORD   0
002054  000000                                                          .WORD   0
002056                  L$SPC::
002056  000000                                                          .WORD   0
002060                  L$DEVP::                ; POINTER TO DEVICE TYPE LIST
002060  002164                                                          .WORD   L$DVTYP
002062                  L$REPP::                ;PTR. TO REPORT CODE
002062  016014                                                          .WORD   L$RPT
002064                  L$EXP4::
002064  000000                                                          .WORD   0
002066                  L$EXP5::
002066  000000                                                          .WORD   0
002070                  L$AUT::                 ;PTR. TO ADD UNIT CODE
002070  021576                                                          .WORD   L$AU
002072                  L$DUT::                 ;PTR. TO DROP UNIT CODE
002072  021524                                                          .WORD   L$DU
002074                  L$LUN::                 ; LUN FOR EXERCISERS TO FILL
002074  000000                                                          .WORD   0
002076                  L$DESP::                ; POINTER TO DIAG. DESCRIPTION
002076  002136                                                          .WORD   L$DESC
002100                  L$LOAD::                ;GENERATE SPECIAL AUTOLOAD EMT
002100  104035                                                          EMT     E$LOAD
002102                  L$ETP::                 ;POINTER TO ERRTBL
002102  000000                                                          .WORD   0
002104                  L$ICP::                 ;PTR. TO INIT CODE
002104  017550                                                          .WORD   L$INIT
002106                  L$CCP::                 ;PTR. TO CLEAN-UP CODE
002106  021462                                                          .WORD   L$CLEAN
002110                  L$ACP::                 ;PTR. TO AUTO CODE
002110  021040                                                          .WORD   L$AUTO
002112                  L$PRT::                 ;PTR. TO PROTECT TABLE
002112  017542                                                          .WORD   L$PROT
002114                  L$TEST::                ;TEST NUMBER
002114  000000                                                          .WORD   0
002116                  L$DLY::                 ;DELAY COUNT
002116  000000                                                          .WORD   0
002120                  L$HIME::                ;PTR. TO HIGH MEM
002120  000000                                                          .WORD   0
```

```
                        .SBTTL  DISPATCH TABLE

                        ;++
                        ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
                        ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
                        ;--

002122                          DISPATCH 5
002122  000005                                                          .WORD   5               ;
002124                  L$DISPATCH::
002124  021672                                                          .WORD   T1
002126  023260                                                          .WORD   T2
002130  023734                                                          .WORD   T3
002132  024100                                                          .WORD   T4
002134  024232                                                          .WORD   T5

                        .SBTTL  DESCRIPTIVE TEXT

                        ;++
                        ;2 LINES OF TEXT PRINTED TO THE OPERATOR TO IDENTIFY THE DIAGNOSTIC AND THE DEVICE UNDER
                        ;--

002136                          DESCRIPT        <DATA RELIABILITY TEST>
002136                  L$DESC::
002136  040504  040524  051040                                          .ASCIZ  /DATA RELIABILIT
002144  046105  040511  044502
002152  044514  054524  052040
002160  051505  000124
                                                                        .EVEN
002164                                  DEVTYP          <TS11>
002164                          L$DVTYP::
002164  051524  030461      000                                         .ASCIZ  /TS11/
        002172                                                          .EVEN
```

```
                                .SBTTL  DEFAULT HARDWARE P-TABLE

                                ;++
                                ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
                                ; THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
                                ; IS IDENTICAL TO THE STRUCTURE OF THE RUN-TIME P-TABLE.
                                ;--

002172                                  BGNHW   DFPTBL
002172  000002                                                                 .WORD   L10000-L$HW/2
002174                          L$HW::
002174                          DFPTBL::

002174  172522                 172522          ;TSSR ADDRESS.
002176  000224                 224             ;VECTOR ADDRESS.

002200                                  ENDHW
002200                          L10000:
```

```
                                .SBTTL  SOFTWARE P-TABLE

                                ;++
                                ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
                                ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
                                ;--

002200                                  BGNSW   SFPTBL
002200  000043                                                                 .WORD   L10001-L$SW/2
002202                          L$SW::
002202                          SFPTBL::

002202  001                     CLRFLG:: .BYTE  1               ;CLEAR COUNTERS FLAG.
002203  000                     RRANV::  .BYTE  0               ;RESET RANDOM VARIABLES EACH PASS FLAG.
002204  000                     HAE::    .BYTE  0               ;HALT AFTER EACH COMMAND FLAG.
002205  000                     ERCVER:: .BYTE  0               ;ENABLE RECOVERABLE ERROR PRINTS FLAG.
002206  001                     BADTSW:: .BYTE  1               ;BAD TAPE SWITCH TO REWRITE ON SAME SPOT & DETECT BAD TA
002207  000                              .BYTE  0               ;SPARE
002210  000                     DINT::   .BYTE  0               ;DISABLE INTERRUPTS FLAG.
002211  000                     IREC::   .BYTE  0               ;INHIBIT ERROR RECOVERY FLAG.
002212  000                     CHGFLG:: .BYTE  0               ;CHANGE CMD SEQ TABLE FLAG.
002213  000                              .BYTE  0               ;SPARE.
002214  000                     PIRE::   .BYTE  0               ;INHIBIT RESIDUAL FRAMECOUNT ERROR REPORT FLAG.
002215  000                              .BYTE  0               ;SPARE.
002216  000040                  CHAR::   CH.EAI                 ;CHARACTERISTICS CODE (DEFAULT = 40).
002220  000015                  CMDD::   .WORD  13.             ;COMMAND 2 (DEFAULT = REWIND`.
002222  000001                           .WORD  1               ;BYTE COUNT
002224  000001                           .WORD  1               ;NUMBER OF OPERATIONS
002226  000007                           .WORD  RANP            ;PATTERN
002230  000004                           .WORD  4               ;COMMAND 3  (DEFAULT = WRITE)
002232  004000                           .WORD  DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
002234  076400                           .WORD  32000.          ;NUMBER OF OPERATIONS (DEFAULT = 32000).
002236  000007                           .WORD  RANP            ;PATTERN (DEFAULT = RANDOM).
002240  000003                           .WORD  3               ;COMMAND 4 (DEFAULT = READ REV).
002242  004000                           .WORD  DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
002244  076400                           .WORD  32000.          ;NUMBER OF OPERATIONS (DEFAULT = 32,000).
002246  000007                           .WORD  RANP            ;PATTERN (DEFAULT = RANDOM).
002250  000002                           .WORD  2               ;COMMAND 5 (DEFAULT = READ FWD).
002252  004000                           .WORD  DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
002254  076400                           .WORD  32000.          ;NUMBER OF OPERATIONS (DEFAULT = 32,000).
002256  000007                           .WORD  RANP            ;PATTERN (DEFAULT = RANDOM).
002260  000015                           .WORD  13.             ;COMMAND 6 (DEFAULT = REWIND).
002262  000001                           .WORD  1               ;BYTE COUNT
002264  000001                           .WORD  1               ;NUMBER OF OPERATIONS
002266  000007                           .WORD  RANP            ;PATTERN
002270  000033                           .WORD  27.             ;END OF CMD SEQ TABLE CODE (DEF) OR CMD 7
002272  004000                           .WORD  DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
002274  076400                           .WORD  32000.          ;NUMBER OF OPERATIONS (DEFAULT = 32000).
002276  000007                           .WORD  RANP            ;PATTERN (DEFAULT = RANDOM).
002300  000033                           .WORD  27.             ;END OF CMD SEQ TABLE CODE (DEF) OR CMD 8
002302  004000                           .WORD  DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
002304  076400                           .WORD  32000.          ;NUMBER OF OPERATIONS (DEFAULT - 32000).
002306  000007                           .WORD  RANP            ;PATTERN (DEFAULT = RANDOM).

002310                                  ENDSW
002310                          L10001:
```

```
  213
  214  002310                              ENDMOD
```

```
                                .TITLE GLOBAL AREAS
                                .SBTTL  GLOBAL EQUATES SECTION

002310                                  BGNMOD

                                ;++
                                ; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
                                ; ARE USED IN MORE THAN ONE TEST.
                                ;--

002310                                  EQUALS
                                ;
                                ; BIT DIFINITIONS
                                ;
        100000                  BIT15== 100000
        040000                  BIT14== 40000
        020000                  BIT13== 20000
        010000                  BIT12== 10000
        004000                  BIT11== 4000
        002000                  BIT10== 2000
        001000                  BIT09== 1000
        000400                  BIT08== 400
        000200                  BIT07== 200
        000100                  BIT06== 100
        000040                  BIT05== 40
        000020                  BIT04== 20
        000010                  BIT03== 10
        000004                  BIT02== 4
        000002                  BIT01== 2
        000001                  BIT00== 1
                                ;
        001000                  BIT9==  BIT09
        000400                  BIT8==  BIT08
        000200                  BIT7==  BIT07
        000100                  BIT6==  BIT06
        000040                  BIT5==  BIT05
        000020                  BIT4==  BIT04
        000010                  BIT3==  BIT03
        000004                  BIT2==  BIT02
        000002                  BIT1==  BIT01
        000001                  BIT0==  BIT00
                                ;
                                ; EVENT FLAG DEFINITIONS
                                ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                ;
        000040                  EF.START==      32.                     ; START COMMAND WAS ISSUED
        000037                  EF.RESTART==    31.                     ; RESTART COMMAND WAS ISSUED
        000036                  EF.CONTINUE==   30.                     ; CONTINUE COMMAND WAS ISSUED
        000035                  EF.NEW==        29.                     ; A NEW PASS HAS BEEN STARTED
        000034                  EF.PWR==        28.                     ; A POWER-FAIL/POWER-UP OCCURRED
                                ;
                                ;
                                ; PRIORITY LEVEL DEFINITIONS
                                ;
        000340                  PRI07== 340
```

```
                                PRI06== 300
                                PRI05== 240
                                PRI04== 200
                                PRI03== 140
                                PRI02== 100
                                PRI01== 40
                                PRI00== 0
                                .
                                ;OPERATOR FLAG BITS
                                ;
000004                          EVL==        4
000010                          LOT==       10
000020                          ADR==       20
000040                          IDU==       40
000100                          ISR--       100
000200                          UAM==       200
000400                          BOE==       400
001000                          PNT==      1000
002000                          PRI==      2000
004000                          IXE==      4000
010000                          IBE==     10000
020000                          IER==     20000
040000                          LOE==     40000
100000                          HOE==    100000


                                ; REGISTER USAGE.
                                ;
                                ;       R0 - PASSES PARAMETERS TO/FROM DIAGNOSTIC SUPERVISOR.
                                ;       R1 - COMMAND SEQUENCE TABLE POINTER.
                                ;       R2 - GENERAL PURPOSE REGISTER.
                                ;       R3 - GENERAL PURPOSE REGISTER.
                                ;       R4 - GENERAL PURPOSE REGISTER.
                                ;       R5 - CURRENT LOGICAL DEVICE NUMBER X 2.
                                ;       R6 - STACK POINTER.
                                ;       R7 - PROGRAM COUNTER.

                                ;THE FOLLOWING ARE BIT DEFINITIONS FOR THE TSSR REGISTERS.

100000          TS.SC==100000                   ;SPECIAL CONDITION BIT.
040000          TS.UPE==40000                   ;UNIBUS PARITY ERROR
020000          TS.SPE==20000                   ;SERIAL BUS PARITY ERROR.
010000          TS.RMR==10000                   ;REGISTER MODIFICATION REFUSED.
004000          TS.NXM==4000                    ;NON-EXISTENT MEMORY.
002000          TS.NBA==2000                    ;NEED BUFFER ADDRESS.
001000          TS.A17==1000                    ;BUS ADDRESS BIT 17.
000400          TS.A16==400                     ;BUS ADDRESS BIT 16.
000200          TS.SSR==200                     ;UNIT READY BIT.
000100          TS.OFL==100                     ;OFF LINE.
177717          TSC.FCC==177717                 ;FATAL CLASS CODE MASK.
177761          TSC.TCC==177761                 ;TERMINATION CLASS CODE MASK.
```

```
                        ;THE FOLLOWING ARE BIT DEFINITIONS FOR THE COMMAND WORD

100000                  ACK.C==100000                   ;ACKNOWLEDGE BIT
040000                  CVC.C==40000                    ;CLEAR VOLUME CHECK.
020000                  OPP.C==20000                    ;OPPOSITE BIT
010000                  SWB.C==10000                    ;SWAP BYTE BIT
004000                  MOD.C3==4000                    ;MODE BIT 3
004000                  BRF.C==4000                     ;BYTE/RECORD/FILE COUNT FLAG BIT.  NOT USED
                                                        ;BY TS04 BUT USED INTERNALLY BY THIS PROGRAM ONLY.
002000                  MOD.C2==2000                    ;MODE BIT 2
001000                  MOD.C1==1000                    ;MODE BIT 1
000400                  MOD.C0==400                     ;MODE BIT 0
000200                  IE.C==200                       ;INTERRUPT ENABLE
000100                  FMT.C1==100                     ;FORMAT BIT 1
000100                  VFY.C==100                      ;WRITE VERIFY FLAG BIT.  INTERNAL USE ONLY.
                                                        ;NOT USED BY TS04.
000040                  FMT.C0==40                      ;FORMAT BIT 0.
000040                  JMP.C==40                       ;JUMP BIT-TO DIRECT THIS PROGRAM TO JUMP TO
                                                        ;A CERTAIN LOCATION IN THE COMMAND SEQUENCE
                                                        ;TABLE.  INTERNAL USE ONLY.
000020                  CMD.C4==20                      ;COMMAND BIT 4
000020                  DLY.C==20                       ;INSERT DELAY. INTERNAL USE ONLY.
000010                  CMD.C3=-10                      ;COMMAND BIT 3
000004                  CMD.C2==4                       ;COMMAND BIT 2
000002                  CMD.C1==2                       ;COMMAND BIT 1
000001                  CMD.C0==1                       ;COMMAND BIT 0

                        ;       BIT DEFINITIONS FOR DEVICE CHARACTERISTICS.

000200                  CH.ESS==200                     ;ENABLE SKIP TAPE MARKS STOP (STOP AT LOGICAL EOT).
000040                  CH.EAI==40                      ;ENABLE ATTENTION INTERRUPTS.
000020                  CH.ERI==20                      ;ENABLE MESSAGE BUFFER RELEASE INTERRUPTS.
000040                  DFTSCH==CH.EAI                  ;DEFAULT CHARACTERISTICS CODE.

                        ;THE FOLLOWING INDICATES THE RELATIVE POSITIONS OF THE STATUS WORDS
                        ;IN THE MESSAGE BUFFER.

000004                  MS.RFC==4                       ;RESIDUAL FRAME COUNT.
000006                  MS.XS0==6                       ;EXT STATUS REG 0
000010                  MS.XS1==10                      ;EXT STATUS REG 1
000012                  MS.XS2==12                      ;EXT STATUS REG 2
000014                  MS.XS3==14                      ;EXT STATUS REG 3

                        ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 0.

100000                  X0.TMK==100000                  ;TAPE MARK.
040000                  X0.RLS==40000                   ;RECORD LENGTH SHORT.
020000                  X0.LET==20000                   ;LOGICAL EOT.
010000                  X0.RLL==10000                   ;RECORD LENGTH LONG.
000100                  X0.ONL==100                     ;ON LINE BIT.
000002                  X0.BOT==2                       ;BOT BIT.
000001                  X0.EOT==1                       ;EOT BIT.

                        ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS RECISTER 2.

100000                  X2.OPM==100000                  ;OPERATION IN PROGRESS, TAPE MOVING
```

```
                                ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 3.

000010                          X3.DCK==10                      ;DENSITY CHECK.
157400                          X3.RNY==157400                  ;CAPSTAN RUNAWAY UDIAG ERROR CODE.

                                ;THE FOLLOWING DEFINITIONS SHOW THE RELATIVE POSITIONS OF THE COMMAND
                                ;PACKET ENTRIES.

000000                          CP.CMD==0                       ;CMDPKT+0==TS04 COMMAND.
000002                          CP.ADL==2                       ;CMDPKT+2==BUFFER ADDRESS LOW.
000004                          CP.ADH==4                       ;CMDPKT+4==BUFFER ADDRESS HIGH.
000006                          CP.CNT==6                       ;CKDPKT+6==BYTE/FILE/RECORD COUNT

                                ;       MISCELLANEOUS DEFINITIONS.

000340                          INTPRI==PRI07                   ;PRIORITY TO BE USED IN INTERRUPT STATE.
002452                          TSBA==TSDB                      ;DATA BUFFER ADDRESS REGISTER.
000010                          SCHCNT==10                      ;ARBITRARY BYTE LENGTH FOR CHARACTERISTIC
                                                                ;BUFFER LENGTH. (EVEN #)
000016                          MSGCNT==16                      ;MESSAGE BUFFER LENGTH IN BYTES  (EVEN #)
003334                          DIABLK==DATAWT                  ;WRITE BUFFER ALSO USED FOR DIAG CMD.
000020                          DIACNT==20                      ;DIAGNOSTIC COMMAND BUFFER EXTENT.
004000                          DATCNT==2048.                   ;MAXIMUM RECORD LENGTH IN BYTES,
                                                                ;THIS COUNT SHOULD BE A MULTIPLE OF 256 TO INSURE
                                                                ;PROPER READ/WRITE BUFFER ALLOCATION BY THE SUPER.
000550                          CNTLEN==CNTEND-CNTBGN           ;LENGTH OF STATISTICAL COUNTER AREA.
177740                          RNOPSC==177740                  ;RANDOM # OF OPERATIONS MASK.
000007                          RANP==7                         ;CODE TO SELECT RANDOM PATTERN.
000020                          RRECL==16.                      ;READ RECOVERY ATTEMPT LIMIT.
000020                          WRECL==16.                      ;WRITE RECOVERY ATTEMPT LIMIT.
153624                          RANBC==153624                   ;CONSTANT USED TO RESET RANDOM # GENERATOR BASE.
032561                          RANSC==32561                    ;CONSTANT USED TO RESET RANDOM # SAVE LOCATION.
177774                          NINUSE==177774                  ;NOT IN USE CODE FOR DEVICE STATE TABLE.
177740                          NCMD.C==ACK.C!CVC.C!OPP.C!SWB.C!MOD.C3!MOD.C2!MOC.C1!MOD.C0!IE.C!FMT.C1!FMT.C0
                                                                ;NOT "COMMAND" BITS.

                                ;THE FOLLOWING DEFINES THE COMMAND WORD FOR EACH TS04 COMMAND.

100013                          DRI==   ACK.C!CMD.C3!CMD.C1!CMD.C0
                                                                ;DRIVE INIT.

104001                          RDF==   ACK.C!BRF.C!CMD.C0
                                                                ;READ FORWARD

104401                          RDR==   ACK.C!BRF.C!MOD.C0!CMD.C0
                                                                ;READ REVERSE

104005                          WRT==   ACK.C!BRF.C!CMD.C0!CMD.C2
                                                                ;WRITE COMMAND

104105                          WTV==   ACK.C!BRF.C!VFY.C!CMD.C0!CMD.C2
                                                                ;WRITE VERIFY

104010                          SRF==   ACK.C!BRF.C!CMD.C3
                                                                ;SPACE RECORD FORWARD
```

```
434
435          104410                  SRR==   ACK.C!BRF.C!MOD.C0!CMD.C3
436                                                          ;SPACE RECORD REVERSE
437
438          105401                  RNR==   ACK.C!BRF.C!MOD.C1!MOD.C0!CMD.C0
439                                          ;READ REV RETRY1 - REREAD NEXT REVERSE, IE. SPACE FWD, READ REVERSE
440
441          125401                  RNF==   ACK.C!BRF.C!OPP.C!MOD.C1!MOD.C0!CMD.C0
442                                          ;READ REV RETRY2 - REREAD NEXT FORWARD, IE.READ FORWARD, SPACE REVERSE
443
444          105001                  RPF==   ACK.C!BRF.C!MOD.C1!CMD.C0
445                                          ;READ FWD RETRY1 - REREAD PREVIOUS FORWARD, IE. SPACE REVERSE, READ FORW
446
447          125001                  RPR==   ACK.C!BRF.C.OPP.C!MOD.C1!CMD.C0
448                                          ;READ FWD RETRY2 - REREAD PREVIOUS REVERSE, IE. READ REVERSE, SPACE FORW
449
450          105005                  WRR-=   ACK.C!MOD.C1!BRF.C!CMD.C2!CMD.C0
451                                                          ;WRITE RETRY
452
453          102010                  RWD==   ACK.C!MOD.C2!CMD.C3
454                                                          ;REWIND COMMAND
455
456          100012                  MBR==   ACK.C!CMD.C3!CMD.C1
457                                                          ;MESSAGE BUFFER RELEASE
458
459          100011                  WTM==   ACK.C!CMD.C3!CMD.C0
460                                                          ;WRITE TAPE MARK.
461
462          101011                  WTR==   ACK.C!MOD.C1!CMD.C3.CMD.C0
463                                                          ;WRITE TAPE MARK RETRY.
464
465          105010                  SFF==   ACK.C!BRF.C.MOD.C1!CMD.C3
466                                                          ;SPACE FILE FORWARD
467
468          105410                  SFR==   ACK.C!BRF.C!MOD.C0!MOD.C1!CMD.C3
469                                                          ;SPACE FILE REVERSE
470
471          100017                  GES==   ACK.C!CMD.C0!CMD.C1!CMD.C2!CMD.C3
472                                                          ;GET EXTENDED STATUS
473
474          100411                  ERS==   ACK.C!MOD.C0!CMD.C3!CMD.C0
475                                                          ;ERASE 3 INCHES OF TAPE
476
477          100412                  UNL==   ACK.C!MOD.C0!CMD.C3!CMD.C1
478                                                          ;UNLOAD COMMAND
479
480          101012                  CLN-=   ACK.C!MOD.C1!CMD.C3!CMD.C1
481                                                          ;ERASE TAPE.
482
483          140004                  SCH==   ACK.C!CVC.C!CMD.C2      ;SET DEVICE CHARACTERISTICS.
484
485          100006                  DIA==   ACK.C!CMD.C2!CMD.C1     ;DIAGNOSTICS.
486
487          000040                  JMP==   JMP.C                   ;JUMP TO 'N'TH COMMAND
488
489          000020                  DLY==   DLY.C                   ;DELAY 'N'' MS.
```

```
490
491          177777                  END==   177777                  ;END OF COMMAND SEQUENCES
```

```
                                .SBTTL  GLOBAL DATA SECTION

                                ;++
                                ; THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
                                ; IN MORE THAN ONE TEST.
                                ;--



                                ;       COMMAND PACKET.

                        002310  .       =       .+3&177774      ;MUST BE ON MOD 4 BOUNDRY.
002310  000000          CMDPKT:: 0                              ;1ST WORD IS TS04 COMMAND.
002312  000000                   0                              ;2ND WORD IS THE BUFFER LOW ADDRESS.
002314  000000                   0                              ;3RD WORD IS THE BUFFER HIGH ADDRESS.
002316  000000                   0                              ;4TH WORD IS THE BYTE/RECORD/FILE COUNT.

                                ;       GET STATUS COMMAND PACKET.

                        002320  .       -       .+3&177774      ;MUST BE ON MOD 4 BOUNDRY.
002320  100017          GSCPK:: .WORD   GES

                                ;       MESSAGE BUFFER RELEASE COMMAND PACKET.

                        002324  .       -       .+3&177774      ;MUST BE ON MOD 4 BOUNDRY.
002324  100012          BRCPK:: .WORD   MBR


                                ;       REWIND COMMAND PACKET (USED IN ERROR RECOVERY ONLY)

                        002330  .               .+3&177774      ;MUST BE ON A MODULE 4 BOUNDARY.
002330  102010          RWCPK:: .WORD   RWD
002332  000001                  .WORD   1

                                ;       WORK AREA FOR ANALYSIS OF MESSAGE PACKET CONTENTS.

002334  000007          MSGPKT:: .BLKW  7               ;1ST WORD:: MESSAGE TYPE.
                                                        ;2ND WORD:: DATA FIELD LENGTH.
                                                        ;3RD WORD:: RESIDUAL FRAME COUNT.
                                                        ;4TH WORD:: XSTAT0
                                                        ;5TH WORD:: XSTAT1
                                                        ;6TH WORD:: XSTAT2
                                                        ;7TH WORD:: XSTAT3
```

```
                                ;       MESSAGE PACKETS.

002352  000007                MSGPK0:: .BLKW  7               ;MESSAGE PACKET FOR DEVICE #0
002370  000007                MSGPK1:: .BLKW  7               ;MESSAGE PACKET FOR DEVICE #1
002406  000007                MSGPK2:: .BLKW  7               ;MESSAGE PACKET FOR DEVICE #2
002424  000007                MSGPK3:: .BLKW  7               ;MESSAGE PACKET FOR DEVICE #3

                                ;       SET CHARACTERISTIC BLOCK.

002442  002352                SCHBK:: MSGPK0                  ;1ST WORD:: MSGPKT ADDR LO(SET UP BY EXCUTE ROUTINE).
002444  000000                        0                       ;2ND WORD:: MSGPKT ADDR HI.
002446  000016                        MSGCNT                  ;3RD WORD:: MSG BUFFER LENGTH (BYTES)
002450  000040                        CH.EAI                  ;4TH WORD:: CHARACTERISTICS WORD(SET BY SETUP ROUTINE).

                                ;       TS04 REGISTER ADDRESSES.

002452  000004                TSDB:: .BLKW    4               ;TS04 DATA BUFFER ADDRESSES.
002462  000004                TSSR:: .BLKW    4               ;TS04 STATUS REGISTER ADDRESSES.
002472  000004                TSVCT:: .BLKW   4               ;TS04 VECTOR ADDRESSES.

                                ;       ADDRESSES OF MESSAGE PACKETS.

002502  002352                MSGPKA:: MSGPK0                 ;DEVICE 0.
002504  002370                        MSGPK1                  ;DEVICE 1.
002506  002406                        MSGPK2                  ;DEVICE 2.
002510  002424                        MSGPK3                  ;DEVICE 3.

                                ;       ADDRESSES OF INTERRUPT HANDLING ROUTINES.

002512  006316                TS4INT:: TS4IN0                 ;DEVICE 0.
002514  006324                        TS4IN1                  ;DEVICE 1.
002516  006332                        TS4IN2                  ;DEVICE 2.
002520  006340                        TS4IN3                  ;DEVICE 3.

                                ;       TS04 CODE LEVELS, WILL BE STORED AFTER SCH CMD IN BASIC FUNCTION TEST

002522  000000                TS4CL:: 0                       ;DEVICE 0
002524  000000                        0                       ;DEVICE 1
002526  000000                        0                       ;DEVICE 2
002530  000000                        0                       ;DEVICE 3

                                ;       UNIT NUMBERS OF ALL DEVICES BEING TESTED(1-4).
                                ;       WHEN DEVICE IS NOT IN USE, IT,S LOCATION WILL - -3.
                                ;       R5 WILL ALWAYS CONTAIN THE PRESENT LOGICAL UNIT NUMBER X 2.

002532  177774                DEVTBL:: .WORD  NINUSE
002534  177774                        .WORD   NINUSE
002536  177774                        .WORD   NINUSE
002540  177774                        .WORD   NINUSE
002542  177777                        .WORD   END


                                ;       BAD TAPE TABLE POINTER: USED BY WRITE RETRY ROUTINE
                                ;       'WRTY'' TO LOG BAD TAPE SPOTS ON UNITS UNDER TEST

002544  002774                BTADDR:: BT0
```

```
    595  002546  003046                          BT1
    596  002550  003120                          BT2
    597  002552  003172                          BT3
```

```
                                        ;       COUNTER AREA.

                    002554              CNTBGN=.
002554  000020      WRBC::  .BLKW   20          ;BYTES WRITTEN.
002614  000020      RRBC::  .BLKW   20          ;BYTES READ REV.
002654  000020      RFBC::  .BLKW   20          ;BYTES READ FWD.
002714  000004      WRREC:: .BLKW   4           ;RECOVERABLE WRITE ERRORS.
002724  000004      WRUNR:: .BLKW   4           ;UNRECOVERABLE WRITE ERRORS.
002734  000004      RRREC:: .BLKW   4           ;RECOVERABLE READ REV ERRORS.
002744  000004      RRUNR:: .BLKW   4           ;UNRECOVERABLE READ REV ERRORS.
002754  000004      RFREC:: .BLKW   4           ;RECOVERABLE READ FWDERRORS.
002764  000004      RFUNR:: .BLKW   4           ;UNRECOVERABLE READ FWD ERRORS.
002774  000025      BT0::   .BLKW   21.         ;UNIT 0 BAT TAPE SPOTS LOG
003046  000025      BT1::   .BLKW   21.         ;UNIT 1 BAT TAPE SPOTS LOG
003120  000025      BT2::   .BLKW   21.         ;UNIT 2 BAT TAPE SPOTS LOG
003172  000025      BT3::   .BLKW   21.         ;UNIT 3 BAT TAPE SPOTS LOG
003244  000004      WRTYCT:: .BLKW  4           ;WRITE RETRY COUNTER
003254  000004      PASCNT:: .BLKW  4           ;PASS COUNT.
003264  000004      SCCNT:: .BLKW   4           ;SPECIAL CONDITION COUNT.
003274  000004      VFYCNT:: .BLKW  4           ;COUNT OF TS04 DATA COMPARE ERRORS.
003304  000004      HRDCNT:: .BLKW  4           ;COUNT OF HARD ERRORS.
003314  000004      FTLCNT:: .BLKW  4           ;COUNT OF FATAL ERRORS.
                    003324              CNTEND=.            ;END OF STATICTICAL COUNTERS.
003324  000004      RECCNT:: .BLKW  4           ;NUMBER OF RECORDS FROM BOT: CLEARED  ON REWIND
                                                ;AND WHEN RESTARTING OR CONTINUING TEST 2.


                                        ;       THE FOLLOWING ARE THE DEFINITIONS OF VARIABLES
                                        ;       USED BY THE PROGRAM.

003334  000000      DATAWT:: .WORD  0           ;WRITE BUFFER ADDRESS.
003336  000000      DATARD:: .WORD  0           ;READ BUFFER ADDRESS.
003340  000000      NCNT::  .WORD   0           ;STORAGE FOR VALUE OF N.
003342  000000      NCNT1:: .WORD   0           ;TEMP STORAGE FOR VALUE OF N.
003344  000000      BRFCNT:: .WORD  0           ;STORAGE FOR BPCR VALUE.
003346  177777      CMDWRD:: .WORD  END         ;CONTAINS COMMAND WORD BEING EXECUTED PRESENTLY.
003350  177777      CMDSAV:: .WORD  END         ;SAVE LOCATION FOR CMD WORD DURING ERROR RECOVERY
003352  177777      PCMDWD:: .WORD  END         ;CONTAINS PREVOUS COMMAND WORD.
003354  000000      CMDLG:: .WORD   0           ;CURRENT COMMAND LOGGING CODE.
003356  000000      LENMSK:: .WORD  0           ;RANDOM WRITE LENGTH MASK, TO BE SET UP BY TESTS
003360  153624      RANB::  .WORD   153624      ;RANDOM # GENERATOR BASE.
003362  032561      RANS::  .WORD   32561       ;RANDOM # SAVE LOCATION.
003364  000000      TIME1:: .WORD   0           ;TIME COUNT 1.
003366  000000      TIME2:: .WORD   0           ;TIME COUNT 2.
003370  000000      JLOOP:: .WORD   0           ;JMP COMMAND LOOP COUNT.
003372  000000      JLOC::  .WORD   0           ;JMP COMMAND LOCATION COUNT.
003374  000000      PATERN:: .WORD  0           ;PATTERN SELECT CODE.
003376  000000      CTCC::  .WORD   0           ;CURRENT TERMINATION CLASS CODE.
003400  000000      R5SAVE:: .WORD  0           ;LOCATION FOR SAVING CURRENT DEVICE POINTER.
003402  000000      TSSREG:: .WORD  0           ;CURRENT STATUS REGISTER.
```

```
                                ;       ERROR FLAG AREA, THESE FLAGS ARE CLEARED DURING INITIALIZATION AND
                                ;       AFTER EACH COMMAND IS COMPLETED.

                        003404  BGNFLG=.
003404  000000          RETRYC:: .WORD  0               ;# OF RECOVERY ATTEMPTS EXECUTED.
003406     000          RPTCNT:: .BYTE  0       ;WRITE REPEAT ON SAME SPOT CNTR. 4 PER WRITE RETRY
003407     000          WRTYFG:: .BYTE  0               ;WRITE RETRY ON SAME SPOT IN PROGRESS FLAG
003410     000          WRTYER:: .BYTE  0               ;WRITE RETRY ON SAME SPOT ERROR FLAG
003411     000          RECLOG:: .BYTE  0               ;RECORD COUNT HAS BEEN UPDATED FOR THIS RECORD.
003412     000          ERLOG:: .BYTE   0               ;DATA BYTES AND ERRORS HAVE BEEN LOGGED FOR THIS RECORD.
003413     000          RWERR:: .BYTE   0               ;READ/WRITE ERROR HAS OCCURED.
003414     000          UNREC:: .BYTE   0               ;UNRECOVERABLE ERROR HAS OCCURED.
003415     000          ERRREC:: .BYTE  0               ;ERROR RECOVERY MODE.
                                .EVEN
                        003416  ENDERF=.

                                ;       ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED DURING INITIALIZATION.

003416  000004          INTFLG:: .BLKW  4               ;INTERRUPT OCCURRED FLAGS FOR EACH DEVICE.
003426  000004          EOTFLG:: .BLKW  4               ;EOT/BOT FLAGS FOR EACH DEVICE (XSTATO).
003436  000000          BTPT::   .WORD  0               ;BAD TAPE SPOT POINTER TO BTO-BT3 VIA BTADDR
003440     000          EXPBOT:: .BYTE  0               ;BOT IS EXPECTED, DO NOT ABORT ON BOT/FUNC RTI.
003441     000          RANDOM:: .BYTE  0               ;RANDOM EVERYTHING FLAG.
003442     000          VFYFLG:: .BYTE  0               ;SET DURING WRITE/VERIFY COMMAND.
003443     000          RPTFLG:: .BYTE  0               ;PERFORMANCE REPORT HAS BEEN REQUESTED.
003444     000          SWBFLG:: .BYTE  0               ;ENABLES SWAP BYTE FUNCTION WHEN NOT EQUAL TO ZERO.
003445     000          IRE::    .BYTE  0               ;INHIBIT RESIDUAL FRAME COUNT ERROR REPORT.
003446     000          DROPED:: .BYTE  0               ;CURRENT UNIT HAS BEEN DROPPED
003447     000          T1SWB::  .BYTE  0               ;TEST1 SWAP BYTES FLAG
003450     000          ALLEOT:: .BYTE  0               ;ALL UNITS @ EOT FLAG
003451     000          ERSFLG:: .BYTE  0               ;ERASE FLAG:  DO ERASE AFTER A SPACE REV TO DELETE
                                                        ;BADLY WRITTEN RECORD.  1 TO 4 ERASES LEAVING
                                                        ;A 3 TO 12 INCH GAP MAY RESULT.
                                .EVEN
                        003452  ENDFLG=.

                                ;       ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED ONLY AFTER BEING CHECKED.

003452     000          STAFLG:: .BYTE  0               ;START FLAG - SET BY INIT CODE IF STARTING.
003453     000          PWRFLG:: .BYTE  0               ;POWER FAILURE FLAG - SET ONLY DURING INIT.
003454     000          TRAPD4:: .BYTE  0               ;TRAPED AT 4 FLAG
003455     000          MISCFG:: .BYTE  0               ;MISCELLANEOUS FLAG

                                ;       OPERATOR FLAG SETTINGS PASSED BY DIAG. SUPERVISOR IN A 16 BIT WORD
                                ;       SEE GLOBAL EQUATES SECTION FOR FLAG BIT LIST

003456  000000          OPFLAG:: .WORD  0       ;READ ONLY OPERATOR FLAG WORD
                                .EVEN
```

```
                                        ;THE FOLLOWING IS THE COMMAND SEQUENCE TABLE.  THE TABLE
                                        ;HAS DEFAULT VALUES AT PROGRAM LOAD AS SHOWN.  THESE VALUES
                                        ;CAN BE UPDATED BY A TEST OR BY OPERATOR INPUT.

003460  140004          CMDSEQ:: .WORD  SCH             ;SET CHARACTERISTICS.
003462  000040                   .WORD  CH.EAI
003464  000001                   .WORD  1
003466  000000                   .WORD  0
003470  102010          CMDSE2:: .WORD  RWD             ;REWIND.
003472  000001                   .WORD  1               ;BYTE COUNT.
003474  000001                   .WORD  1               ;ONCE.
003476  000007                   .WORD  RANP            ;PATTERN.
003500  104005                   .WORD  WRT             ;WRITE.
003502  004000                   .WORD  DATCNT          ;MAX BUFFER LENGTH.
003504  076400                   .WORD  32000.          ;32,000 RECORDS.
003506  000007                   .WORD  RANP            ;RANDOM PATTERN.
003510  104401                   .WORD  RDR             ;READ REV.
003512  004000                   .WORD  DATCNT          ;MAX BUFFER LENGTH.
003514  076400                   .WORD  32000.          ;32,000 RECORDS
003516  000007                   .WORD  RANP            ;RANDOM PATTERN.
003520  104001                   .WORD  RDF             ;READ FWD.
003522  004000                   .WORD  DATCNT          ;MAX BUFFER LENGTH.
003524  076400                   .WORD  32000.          ;32,000 RECORDS.
003526  000007                   .WORD  RANP            ;RANDOM PATTERN.
003530  102010                   .WORD  RWD             ;REWIND.
003532  000001                   .WORD  1               ;BYTE COUNT.
003534  000001                   .WORD  1               ;ONCE.
003536  000007                   .WORD  RANP            ;PATTERN.
003540  000004                   .BLKW  4               ;EXTENSTION TO HOLD 1 MORE CMD.
003550  177777          SEQEND:: .WORD  END             ;SOFT END OF SEQUENCE TABLE.
003552  177777                   .WORD  END
003554  177777                   .WORD  END
003556  177777                   .WORD  END
003560  177777                   .WORD  END             ;HARD END OF SEQUENCE TABLE.
```

```
730                                     ;THE FOLLOWING IS THE TS04 COMMAND TABLE
731
732  003562  100013           CMDTBL:: .WORD  DRI          ;DRIVE INIT.
733  003564  104001                    .WORD  RDF          ;READ FORWARD.
734  003566  104401                    .WORD  RDR          ;READ REVERSE.
735  003570  104005                    .WORD  WRT          ;WRITE
736  003572  104105                    .WORD  WTV          ;WRITE/VERIFY.  (WRITE ALL RECORDS, RDR AND
737                                                        ;CHECK DATA ON ALL RECORDS, RDF AND
738                                                        ;CHECK DATA ON ALL RECORDS.)
739  003574  104010                    .WORD  SRF          ;SPACE 'N'' RECORDS FORWARD.
740  003576  104410                    .WORD  SRR          ;SPACE 'N'' RECORDS REVERSE.
741  003600  105401                    .WORD  RNR          ;READ NEXT REVERSE. I.E., SPACE FWD, READ REVERSE.
742  003602  125401                    .WORD  RNF          ;READ NEXT FORWARD, I.E., READ FORWARD, SPACE REVERSE.
743  003604  105001                    .WORD  RPF          ;READ PREVIOUS FORWARD. I.E., SPACE REVERSE, READ FORWAR
744  003606  125001                    .WORD  RPR          ;READ PREVIOUS REVERSE. I.E., READ REVERSE, SPACE FORWAR
745  003610  105005                    .WORD  WRR          ;WRITE RETRY.
746  003612  102010                    .WORD  RWD          ;REWIND.
747  003614  100012                    .WORD  MBR          ;MESSAGE BUFFER RELEASE
748  003616  100011                    .WORD  WTM          ;WRITE TAPE MARK
749  003620  101011                    .WORD  WTR          ;WRITE TAPE MARK RETRY.
750  003622  105010                    .WORD  SFF          ;SPACE 'N'' FILES FORWARD.
751  003624  105410                    .WORD  SFR          ;SPACE 'N'' FILES REVERSE.
752  003626  100017                    .WORD  GES          ;GET EXTENDED STATUS.
753  0036    100411                    .WORD  ERS          ;ERASE 3 INCHES OF TAPE.
754  003632  100412                    .WORD  UNL          ;REWIND AND UNLOAD.
755  003634  101012                    .WORD  CLN          ;CLEAR TAPE.
756  003636  140004                    .WORD  SCH          ;SET CHARACTERISTICS.
757  003640  100006                    .WORD  DIA          ;DIAGNOSTIC COMMAND.
758  003642  000040                    .WORD  JMP          ;JUMP TO THE NTH COMMAND IN THE SEQUENCE.
759  003644  000020                    .WORD  DLY          ;DELAY 'N'' MS.
760  003646  177777                    .WORD  END          ;END OF COMMAND TABLE
761
```

```
                                ;       THE FOLLOWING TABLE CONTAINS THE ASCII FOR EACH COMMAND.

003650  051104     111         CMDASC:: .ASCII /DRI/          ;DRIVE INIT.
003653     122  043104                  .ASCII  /RDF/          ;READ FORWARD.
003656  042122     122                  .ASCII  /RDR/          ;READ REVERSE.
003661     127  052122                  .ASCII  /WRT/          ;WRITE
003664  052127     126                  .ASCII  /WTV/          ;WRITE/VERIFY.  (WRITE ALL RECORDS, RDR AND CHECK DATA
                                                               ;ON ALL RECORDS, RDF AND CHECK DATA ON ALL RECORDS.)
003667     123  043122                  .ASCII  /SRF/          ;SPACE 'N'' RECORDS FORWARD.
003672  051123     122                  .ASCII  /SRR/          ;SPACE 'N''  RECORDS REVERSE.
003675     122  051116                  .ASCII  /RNR/          ;READ NEXT REVERSE. I.E., SPACE FWD READ REVERSE.
003700  047122     106                  .ASCII  /RNF/          ;READ NEXT FORWARD, I.E., READ FORWARD, SPACE REVERSE.
003703     122  043120                  .ASCII  /RPF/          ;READ PREVIOUS FORWARD. IE., SPACE REVERSE, READ FORWARD
003706  050122     122                  .ASCII  /RPR/          ;READ PREVIOUS REVERSE. IE., READ REVERSE, SPACE FORWARD
003711     127  051122                  .ASCII  /WRR/          ;WRITE RETRY.
003714  053522     104                  .ASCII  /RWD/          ;REWIND.
003717     115  051102                  .ASCII  /MBR/          ;MESSAGE BUFFER RELEASE
003722  052127     115                  .ASCII  /WTM/          ;WRITE TAPE MARK
003725     127  051124                  .ASCII  /WTR/          ;WRITE TAPE MARK RETRY.
003730  043123     106                  .ASCII  /SFF/          ;SPACE 'N'' FILES FORWARD.
003733     123  051106                  .ASCII  /SFR/          ;SPACE 'N'' FILES REVERSE.
003736  042507     123                  .ASCII  /GES/          ;GET EXTENDED STATUS.
003741     105  051522                  .ASCII  /ERS/          ;ERASE 3 INCHES OF TAPE.
003744  047125     114                  .ASCII  /UNL/          ;REWIND AND UNLOAD.
003747     103  047114                  .ASCII  /CLN/          ;CLEAN TAPE.
003752  041523     110                  .ASCII  /SCH/          ;SET CHARACTERISTICS.  WHERE BRF=200, 40, 20, 0.
                                                               ;SEE TS11/TS04 PROGRAMMING SPECIFICATION FOR DESCRIPTION
003755     104  040511                  .ASCII  /DIA/          ;DIAGNOSTICS.  SEE TS11/TS04 PROGRAMMING SPECIFICATION
                                                               ;FOR DESCRIPTION.  ODT MUST BE USED TO LOAD DIAGNOSTIC D
                                                               ;INTO THE WRITE BUFFER BEFORE THIS CMD IS ISSUED.
003760  046512     120                  .ASCII  /JMP/          ;JUMP TO THE NTH COMMAND IN THE COMMAND
                                                               ;SEQUENCE TABLE, WHERE N IS DEFINED IN
                                                               ;THE # OF OPERATIONS.
003763     104  054514                  .ASCII  /DLY/          ;DELAY 'N'' MS, WHERE N IS DEFINED IN
                                                               ;THE # OF OPERATIONS.
003766  047105     104                  .ASCII  /END/          ;END OF COMMAND SEQUENCE.
        003772                          .EVEN
```

```
                                .SBTTL  GLOBAL TEXT SECTION

                                ;++
                                ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
                                ; MESSAGES, AND ASCII INFORMATION THAT ARE USED 'N
                                ; MORE THAN ONE TEST.
                                ;--


                                ;
                                ; FORMAT STATEMENTS USED IN PRINT CALLS
                                ;

                                        .NLIST  BEX

003772 047045 040445 047125  CODELM:: .ASCIZ /%N%AUNIT %D1%A TS11 CODE LEVEL P%03%N%N/
                                      .EVEN
004042 054130 020130 046503  HALTM:: .ASCIZ  /XXX CMD - TYPE <CR> TO CONTINUE/
004102 046503 020104 040520  CMDPKM:: .ASCIZ /CMD PACKET ADR NOT ON MODULO 4 BOUNDARY: RELOAD!/
       004164
                                      .EVEN
004164 040504 040524 041440  WTVERM:: .ASCIZ /DATA COMPARE ERROR/
004207    116 020117 051524  TOERM:: .ASCIZ  /NO TS11 RESPONSE/
004230 047125 042504 044506  SCERM:: .ASCIZ  /UNDEFINED SPEC COND/
004254 043122 020103 047516  RFCERM:: .ASCIZ /RFC NON ZERO/
004271    124 030523 020061  NSSRM:: .ASCIZ  /TS11 NOT READY/
004310 042522 051124 020131  RLEXM:: .ASCIZ  /RETRY LIMIT EXCEEDED/
004335    125 044516 020124  ATTNM:: .ASCIZ  /UNIT OFF LINE/
004353    106 047125 052103  FUNRM:: .ASCIZ  /FUNCTION REJECT/
004373    106 052101 046101  FATSM:: .ASCIZ  /FATAL SUBSYSTEM ERROR/
004421    116 020117 047111  NOINTM:: .ASCIZ /NO INTERRUPT/
004436 040524 042520 051440  TSAM::  .ASCIZ  /TAPE STATUS ALERT/
004460 047524 020117 040515  TOOMM:: .ASCIZ  /TOO MANY INTERRUPTS/
004504 040503 051520 040524  RNYM::  .ASCIZ  /CAPSTAN RUNAWAY-GET STATUS RESULTS:/
004550 042522 047503 042526  RERM::  .ASCIZ  /RECOVERABLE ERROR/
004572 047125 042522 047503  URERM:: .ASCIZ  /UNRECOVERABLE ERROR/
004616 047045 040445 051104  DROPDM:: .ASCIZ /%N%ADROPPED UNIT %D1%N/
004645    045 022516 040501  AUDRPM:: .ASCIZ /%N%AALL UNITS DROPPED%N%N/
004677    045 022516 041101  DTAER2:: .ASCIZ '%N%ABYTE:%D4%S2%AWAS:%B8%S2%AS/B:%B8%N'
004746 042045 022464 020101  DTAER3:: .ASCIZ '%D4%A BYTES IN ERROR OUT OF %D4%N'
005010 040445 047516 042040  DTAER4:: .ASCIZ /%ANO DATA READ%N/
005031    045 051101 041505  DTAER5:: .ASCIZ /%ARECORD TOO LONC: >%04%A BYTES%N/
005073    045 051101 041505  NURTY1:: .ASCIZ /%ARECOVERED ON RETRY #%D2%N/
005127    045 052501 044516  OFLINM:: .ASCIZ /%AUNIT %D1%A OFF LINE%N/
005157    045 043501 052105  GETSTM:: .ASCIZ /%AGET STATUS CMD RESULTS:%N/
005213    045 000116         CRLF::  .ASCIZ  /%N/
005216 047045 051445 000067  CRLFSP:: .ASCIZ /%N%S7/
                                      .LIST   BEX
                                      .EVEN
```

```
                        .SBTTL  GLOBAL ERROR REPORT SECTION

                        ;++
                        ; THE GLOBAL ERROR REPORT SECTION CONTAINS THE PRINTB AND PRINTX CALLS
                        ; THAT ARE USED IN MORE THAN ONE TEST.  IT ALSO INCLUDES THE ASCII MESSAGES
                        ; THAT ARE USED BY THE PRINTB AND PRINTX CALLS..
                        ;--


005224                          BGNMSG  DTAERM
005224                  DTAERM::
005224                          PRINTB  #STAER1,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
005224  016546  003324                                                          MOV     RECCNT(R5),-(SP)
005230  016546  003254                                                          MOV     PASCNT(R5),-(SP)
005234  016546  002532                                                          MOV     DEVTBL(R5),-(SP)
005240  012746  005704                                                          MOV     #STAER1,-(SP)
005244  012746  000004                                                          MOV     #4,-(SP)
005250  010600                                                                  MOV     SP,R0
005252  104414                                                                  TRAP    C$PNTB
005254  062706  000012                                                          ADD     #12,SP
005260                          PRINTB  #STAER7
005260  012746  005776                                                          MOV     #STAER7,-(SP)
005264  012746  000001                                                          MOV     #1,-(SP)
005270  010600                                                                  MOV     SP,R0
005272  104414                                                                  TRAP    C$PNTB
005274  062706  000004                                                          ADD     #4,SP
005300                          LET RECRED := R2                ;SAVE R2
005300  010237  006312                                                          MOV     R2,RECRED
005304                          LET TIME1 := R3                 ;SAVE R3
005304  010337  003364                                                          MOV     R3,TIME1
005310                          LET TIME2 := R4                 ;SAVE R4
005310  010437  003366                                                          MOV     R4,TIME2
005314  004737  006346          JSR PC,RECTAP                   ;RETRIEVE RECORD READ
005320                          LET R2 := RECRED                ;RESTORE R2
005320  013702  006312                                                          MOV     RECRED,R2
005324                          LET RECRED :- R3                ;SAVE RECORD READ
005324  010337  006312                                                          MOV     R3,RECRED
005330                          LET R3 := TIME1                 ;RESTORE R3
005330  013703  003364                                                          MOV     TIME1,R3
005334                          LET R4 := TIME2                 ;RESTORE R4
005334  013704  003366                                                          MOV     TIME2,R4
005340                          PRINTB  #STAER6,RECRED          ;PRINT RECORD READ
005340  013746  006312                                                          MOV     RECRED,-(SP)
005344  012746  006026                                                          MOV     #STAER6,-(SP)
005350  012746  000002                                                          MOV     #2,-(SP)
005354  010600                                                                  MOV     SP,R0
005356  104414                                                                  TRAP    C$PNTB
005360  062706  000006                                                          ADD     #6,SP
005364                          EXIT    MSG
005364  000167                                                                  .WORD   J$JMP
005366  000000                                                                  .WORD   L10002-2-.
                                .EVEN
                                ENDMSG
005370                  L10002:
005370                                                                  TRAP    C$MSG
005370  104423
```

```
873
874  005372                          BGNMSG  STAERM
875  005372                      STAERM::
876  005372                          PRINTB  #STAER1,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
877  005372  016546  003324                                                  MOV     RECCNT(R5),-(SP)
878  005376  016546  003254                                                  MOV     PASCNT(R5),-(SP)
879  005402  016546  002532                                                  MOV     DEVTBL(R5),-(SP)
880  005406  012746  005704                                                  MOV     #STAER1,-(SP)
881  005412  012746  000004                                                  MOV     #4,-(SP)
882  005416  010600                                                          MOV     SP,R0
883  005420  104414                                                          TRAP    C$PNTB
884  005422  062706  000012                                                  ADD     #12,SP
885  005426                          PRINTB  #STAER7
886  005426  012746  005776                                                  MOV     #STAER7,-(SP)
887  005432  012746  000001                                                  MOV     #1,-(SP)
888  005436  010600                                                          MOV     SP,R0
889  005440  104414                                                          TRAP    C$PNTB
890  005442  062706  000004                                                  ADD     #4,SP
891  005446                          LET R2 := CMDPKT CLR.BY #177740
892  005446  013702  002310                                                  MOV     CMDPKT,R2
893  005452  042702  177740                                                  BIC     #177740,R2
894  005456                          LET R2 := R2 - #1
895  005456  005302                                                          DEC     R2
896  005460                          IF R2 EQ #0 THEN                ;IF CMD IS A READ
897  005460  005702                                                          TST     R2
898  005462  001016                                                          BNE     50000$
899  005464  004737  006346           JSR PC,RECTAP                 ;THEN RETRIEVE
900  005470                            LET RECRED := R3              ;AND
901  005470  010337  006312                                                  MOV     R3,RECRED
902  005474                            PRINTB #STAER6,RECRED         ;TYPE RECORD READ
903  005474  013746  006312                                                  MOV     RECRED,-(SP)
904  005500  012746  006026                                                  MOV     #STAER6,-(SP)
905  005504  012746  000002                                                  MOV     #2,-(SP)
906  005510  010600                                                          MOV     SP,R0
907  005512  104414                                                          TRAP    C$PNTB
908  005514  062706  000006                                                  ADD     #6,SP
909  005520                          ENDIF
910  005520                                                          50000$:
911  005520                          PRINTX  #STAER2
912  005520  012746  006062                                                  MOV     #STAER2,-(SP)
913  005524  012746  000001                                                  MOV     #1,-(SP)
914  005530  010600                                                          MOV     SP,R0
915  005532  104415                                                          TRAP    C$PNTX
916  005534  062706  000004                                                  ADD     #4,SP
917  005540                          PRINTX  #STAER3,CMDPKT,@TSDB(R5),MSGPKT+MS.RFC,TSSREG,CTCC
918  005540  013746  003376                                                  MOV     CTCC,-(SP)
919  005544  013746  003402                                                  MOV     TSSREG,-(SP)
920  005550  013746  002340                                                  MOV     MSGPKT+MS.RFC,-(
921  005554  017546  002452                                                  MOV     @TSDB(R5),-(SP)
922  005560  013746  002310                                                  MOV     CMDPKT,-(SP)
923  005564  012746  006141                                                  MOV     #STAER3,-(SP)
924  005570  012746  000006                                                  MOV     #6,-(SP)
925  005574  010600                                                          MOV     SP,R0
926  005576  104415                                                          TRAP    C$PNTX
927  005600  062706  000016                                                  ADD     #16,SP
928  005604                          PRINTX  #STAER4,CMDPKT+2,CMDPKT+4,CMDPKT+6
```

```
  929  005604  013746  002316                                                  MOV     CMDPKT+6,-(SP)
  930  005610  013746  002314                                                  MOV     CMDPKT+4,-(SP)
  931  005614  013746  002312                                                  MOV     CMDPKT+2,-(SP)
  932  005620  012746  006177                                                  MOV     #STAER4,-(SP)
  933  005624  012746  000004                                                  MOV     #4,-(SP)
  934  005630  010600                                                          MOV     SP,R0
  935  005632  104415                                                          TRAP    C$PNTX
  936  005634  062706  000012                                                  ADD     #12,SP
  937  005640                          PRINTX  #STAER5,MSGPKT+MS.XS0,MSGPKT+MS.XS1,MSGPKT+MS.XS2,MSGPKT+MS.XS3
  938  005640  013746  002350                                                  MOV     MSGPKT+MS.XS3,-(
  939  005644  013746  002346                                                  MOV     MSGPKT+MS.XS2,-(
  940  005650  013746  002344                                                  MOV     MSGPKT+MS.XS1,-(
  941  005654  013746  002342                                                  MOV     MSGPKT+MS.XS0,-(
  942  005660  012746  006217                                                  MOV     #STAER5,-(SP)
  943  005664  012746  000005                                                  MOV     #5,-(SP)
  944  005670  010600                                                          MOV     SP,R0
  945  005672  104415                                                          TRAP    C$PNTX
  946  005674  062706  000014                                                  ADD     #14,SP
  947  005700                          EXIT    MSG
  948  005700  000167                                                          .WORD   J$JMP
  949  005702  000410                                                          .WORD   L10003-2-.
  950
  951                                     .NLIST  BEX
       005704  040445  054130  020130  STAER1: .ASCIZ  /%AXXX CMD FAILED - UNIT %D1%S3%APASS:%D5%S3%ARECORD:%D5%N/
                                       .EVEN
       005776  040445  051120  053105  STAER7: .ASCIZ  /%APREVIOUS CMD WAS  XXX/
       006026  051445  030461  040445  STAER6: .ASCIZ  /%S11%A* RECORD READ:%D5%A */
       006062  047045  040445  046503  STAER2: .ASCIZ  /%N%ACMDPKT%S2%ATSBA%S4%ARFC%S5%ATSSR%S3%ATCC%N/
       006141     045  033117  051445  STAER3: .ASCIZ  /%06%S2%06%S2%06%S2%06%S2%D1%N/
       006177     045  033117  047045  STAER4: .ASCII  /%06%N/
       006204  047445  022466     116          .ASCII  /%06%N/
       006211     045  033117  047045          .ASCIZ  /%06%N/
       006217     045  054101  052123  STAER5: .ASCII  /%AXST0%S4%AXST1%S4%AXST2%S4%AXST3%N/
       006262  047445  022466  031123          .ASCIZ  /%06%S2%06%S2%06%S2%06%N/
                                       .LIST   BEX
  952
  953  006312  000000                RECRED: .WORD   0                       ;RECORD READ FROM TAPE
  954
  955  006314                          ENDMSG
  956  006314                      L10003:
  957  006314  104423                                                          TRAP    C$MSG
```

```
                        .SBTTL  GLOBAL SUBROUTINES SECTION

                        ;++
                        ; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
                        ; THAT ARE USED IN MORE THAN ONE TEST.
                        ;--

                        ;       MODULES TO HANDLE TS04 INTERRUPTS.

006316                          BGNSRV  TS4IN0                  ;DEVICE 0.
006316                  TS4IN0::
006316                          LET INTFLG := INTFLG + #1        ;SET INTERRUPT OCCURRED FLAG.
006316  005237  003416                                                  INC     INTFLG
006322                          ENDSRV
006322                  L10004:
006322  000002                                                          RTI

006324                          BGNSRV  TS4IN1                  ;DEVICE 1.
006324                  TS4IN1::
006324                          LET INTFLG+2 := INTFLG+2 + #1    ;SET INTERRUPT OCCURRED FLAG.
006324  005237  003420                                                  INC     INTFLG+2
006330                          ENDSRV
006330                  L10005:
006330  000002                                                          RTI

006332                          BGNSRV  TS4IN2                  ;DEVICE 2.
006332                  TS4IN2::
006332                          LET INTFLG+4 := INTFLG+4 + #1    ;SET INTERRUPT OCCURRED FLAG.
006332  005237  003422                                                  INC     INTFLG+4
006336                          ENDSRV
006336                  L10006:
006336  000002                                                          RTI

006340                          BGNSRV  TS4IN3                  ;DEVICE 3.
006340                  TS4IN3::
006340                          LET INTFLG+6 := INTFLG+6 + #1    ;SET INTERRUPT OCCURRED FLAG.
006340  005237  003424                                                  INC     INTFLG+6
006344                          ENDSRV
006344                  L10007:
006344  000002                                                          RTI
```

```
                                        ;       SUBROUTINE TO RETRIEVE RECORD COUNT READ FROM TAPE FOR ERROR
                                        ;       PRINTS.
                                        ;       INPUTS:
                                        ;       OUTPUTS:  R3 = RECORD COUNT READ
                                        ;       REGISTERS:  R2, R3, R4
                                        ;       CALLS:

006346                                  RECTAP::IF #MOD.C0 SETIN CMDWRD THEN    ;READ REV FETCH
006346  032737  000400  003346                                                          BIT     #MOD.C0,CMDWRD
006354  001430                                                                         BEQ     50001$
006356                                    LET R2 := MSGPKT+MS.RFC + DATARD ;FIND LAST READ AD.
006356  013702  002340                                                                 MOV     MSGPKT+MS.RFC,R2
006362  063702  003336                                                                 ADD     DATARD,R2
006366                                    IF #BIT00 SETIN R2 THEN       ;ODD AD., REASSEMBLE
006366  032702  000001                                                                 BIT     #BIT00,R2
006372  001417                                                                         BEQ     50002$
006374                                      LET R2 := R2 + #1           ;REC COUNT STARTING
006374  005202                                                                       INC     R2
006376                                      LET R3 :B= (R2) CLR.BY #177400  ;WITH UPPER BYTE FETCH
006376  111203                                                                       MOVB    (R2),R3
006400  142703  177400                                                               BICB    #177400,R3
006404                                      LET R3 := SWAP R3           ;
006404  000303                                                                       SWAB    R3
006406                                      LET R2 := R2 - #1           ;LOWER BYTE AD.
006406  005302                                                                       DEC     R2
006410                                      IFB SWBFLG NE #0 THEN
006410  105737  003444                                                               TSTB    SWBFLG
006414  001401                                                                       BEQ     50003$
006416                                        LET R2 := R2 - #1         ;LOWER BYTE AD. ON SWAP
006416  005302                                                                       DEC     R2
006420                                      ENDIF
006420                                                                  50003$:
006420                                      LET R4 :B= (R2) CLR.BY #177400 ;FETCH LOWER BYTE
006420  111204                                                                       MOVB    (R2),R4
006422  142704  177400                                                               BICB    #177400,R4
006426                                      LET R3 := R3 OR R4          ;MERGE BYTES
006426  050403                                                                       BIS     R4,R3
006430                                    ELSE
006430  000401                                                                       BR      50004$
006432                                                                  50002$:
006432                                      LET R3 := (R2)              ;EVEN AD. FETCH
006432  011203                                                                       MOV     (R2),R3
006434                                    ENDIF
006434                                                                  50004$:
006434                                  ELSE
006434  000402                                                                       BR      50005$
006436                                                                  50001$:
006436                                    LET R3 :  @DATARD             ;READ FWD FETCH
006436  017703  174674                                                               MOV     @DATARD,R3
006442                                  ENDIF
006442                                                                  50005$:
006442
006442  000207                          RTS     PC
```

```
1051                                    ;       SUBROUTINE TO STORE A  SET CHARACTERISTIC COMMAND AS
1052                                    ;       THE FIRST ENTRY IN THE SEQUENCE TABLE.
1053                                    ;       INPUTS:
1054                                    ;       OUTPUTS:
1055                                    ;       REGISTERS:
1056                                    ;       CALLS:
1057
1058  006444                            SETCH:: LET R1 := #CMDSEQ               ;INIT COMMAND SEQUENCE TABLE POINTER.
1059  006444  012701  003460                                                                    MOV     #CMDSEQ,R1
1060  006450  012721  140004                   MOV     #SCH,(R1)+              ;THIS CODE SETS UP A SET CHARACTERISTIC
1061  006454  012721  000040                   MOV     #DFTSCH,(R1)+           ;COMMAND AS THE FIRST COMMAND IN THE
1062  006460  012721  000001                   MOV     #1,(R1)+                ;SEQUENCE TABLE.
1063  006464  005721                           TST     (R1)+                   ;SKIP PATTERN LOCATION.
1064  006466  000207                           RTS PC
1065
1066
1067
1068
1069                                    ;       SUBROUTINE TO STORE A REWIND COMMAND IN THE SEQUENCE TABLE
1070                                    ;       INPUTS:
1071                                    ;       OUTPUTS:
1072                                    ;       REGISTERS:
1073                                    ;       CALLS:
1074
1075  006470                            SETRW:: LET (R1)+ := #RWD               ;CMD = REWIND.
1076  006470  012721  102010                                                                    MOV     #RWD,(R1)+
1077  006474                                    LET (R1)+ := #1                 ;BRF.
1078  006474  012721  000001                                                                    MOV     #1,(R1)+
1079  006500                                    LET (R1)+ := #1                 ;# OF OPERATIONS.
1080  006500  012721  000001                                                                    MOV     #1,(R1)+
1081  006504  005721                           TST (R1)+                       ;SKIP PATTERN.
1082  006506  000207                           RTS     PC                      ;RETURN
```

```
1083                                    ;       SUBROUTINE TO EXECUTE ALL COMMANDS IN THE SEQUENCE TABLE ON ALL
1084                                    ;       DEVICES.
1085                                    ;       INPUTS:
1086                                    ;       OUTPUTS:        R2 = TERMINATION INDICATOR (0=END OF TABLE,1=EOT)
1087                                    ;       REGISTERS:
1088                                    ;       CALLS:          CMDAC,SETUP,EXSUB,CKHAE,NEXTU,FIRSTU,VFYDAT.
1089
1090  006510                            EXALL:: LET R1 := #CMDSEQ               ;INIT SEQUENCE TABLE POINTER.
1091  006510  012701  003460                                                            MOV     #CMDSEQ,R1
1092  006514                                    WHILE (R1) NE #END DO             ;WHILE THERE ARE CMDS IN THE SEQUENCE TABLE.
1093  006514                                                                                    50006$:
1094  006514  021127  177777                                                            CMP     (R1),#END
1095  006520  001527                                                                    BEQ     50007$
1096  006522  004737  007452                      JSR   PC,SETUP                    ;GO SETUP THE COMMAND BLOCK.
1097  006526                                      WHILE NCNT LT NCNT1 DO            ;WHILE THERE ARE RECORDS REMAINING:
1098  006526                                                                                    50010$:
1099  006526  023737  003340  003342                                                    CMP     NCNT,NCNT1
1100  006534  002116                                                                    BGE     50011$
1101  006536  004737  007344                        JSR PC,CMDAC                    ;STORE CMD ASCII IN ERROR MESSAGE.
1102  006542                                        IFB RANDOM NE #0 THEN           ;IF IN RANDOM MODE:
1103  006542  105737  003441                                                            TSTB    RANDOM
1104  006546  001435                                                                    BEQ     50012$
1105  006550                                          IF CMDWRD EQ #WRT THEN        ;IF CMD IS A WRITE THEN:
1106  006550  023727  003346  104005                                                    CMP     CMDWRD,#WRT
1107  006556  001031                                                                    BNE     50013$
1108  006560                                            IFB VFYFLG EQ #0 THEN       ;IF DATA IS NOT TO BE VERIFIED THEN:
1109  006560  105737  003442                                                            TSTB    VFYFLG
1110  006564  001026                                                                    BNE     50014$
1111  006566                                              LET RANB := RANB + RANS   ;GENERATE
1112  006566  063737  003362  003360                                                    ADD     RANS,RANB
1113  006574                                              LET RANS := RANS + RANB   ;RANDOM
1114  006574  063737  003360  003362                                                    ADD     RANB,RANS
1115  006602                                              LET BRFCNT := RANS        ;LENGTH
1116  006602  013737  003362  003344                                                    MOV     RANS,BRFCNT
1117  006610                                              LET BRFCNT := BRFCNT CLR.BY LENMSK  ;MASK RANDOM LENGTH.
1118  006610  043737  003356  003344                                                    BIC     LENMSK,BRFCNT
1119  006616                                              IF BRFCNT LT #18. THEN        ;DO NOT ALLOW BYTE COUNT OF LESS THAN 18
1120  006616  023727  003344  000022                                                    CMP     BRFCNT,#18.
1121  006624  002003                                                                    BGE     50015$
1122  006626                                                LET BRFCNT :- #18.  ;CHANGE COUNT OF 0-17 TO 18.
1123  006626  012737  000022  003344                                                    MOV     #18.,BRFCNT
1124  006634                                              ENDIF
1125  006634                                                                                    50015$:
1126  006634                                              LET CMDPKT+CP.CNT := BRFCNT  ;MOVE BRF TO CMD PACKET.
1127  006634  013737  003344  002316                                                    MOV     BRFCNT,CMDPKT+CP
1128  006642                                            ENDIF
1129  006642                                                                                    50014$:
1130  006642                                          ENDIF
1131  006642                                                                                    50013$:
1132  006642                                        ENDIF
1133  006642                                                                                    50012$:
1134  006642  004737  007004                      JSR PC,EXSUB                    ;ISSUE CMD TO ALL,AWAIT INTS,CHECK STATUS.
1135  006646  004737  015724                      JSR PC,CKHAE                    ;CHECK HALT AFTER EACH CMD FLAG.
1136  006652                                      LET R2 := #1                    ;SET ALL UNITS AT BOT/EOT.
1137  006652  012702  000001                                                            MOV     #1,R2
1138  006656  004737  015332                      JSR PC,FIRSTU                   ;FIND FIRST UNIT.
```

```
1139  006662                                  WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE UNITS:
1140  006662                                                                50016$:
1141  006662  026527  002532  177777                                        CMP     DEVTBL(R5),#END
1142  006660  001426                                                        BEQ     50017$
1143  006672                                    IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
1144  006672  032737  000400  003346                                        BIT     #MOD.CO,CMDWRD
1145  006700  001406                                                        BEQ     50020$
1146  006702                                      IF #X0.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT THEN:
1147  006702  032765  000002  003426                                        BIT     #X0.BOT,EOTFLG(R
1148  006710  001001                                                        BNE     50021$
1149  006712                                        LET R2 := #0                    ;CLEAR EOT/BOT FLAG.
1150  006712  005002                                                        CLR     R2
1151  006714                                      ENDIF
1152  006714                                                                50021$:
1153  006714                                    ELSE                        ;ELSE IF CMD IS NOT REVERSE:
1154  006714  000411                                                        BR      50022$
1155  006716                                                                50020$:
1156  006716                                      IF #X0.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
1157  006716  032765  000001  003426                                        BIT     #X0.EOT,EOTFLG(R
1158  006724  001404                                                        BEQ     50023$
1159  006726  032737  000001  003346                                        BIT     #CMD.CO,CMDWRD
1160  006734  001001                                                        BNE     50024$
1161  006736                                                                50023$:
1162                                                                    ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
1163  006736                                        LET R2 :  #0              ;CLEAR EOT/BOT FLAG.
1164  006736  005002                                                        CLR     R2
1165  006740                                      ENDIF
1166  006740                                                                50024$:
1167  006740                                    ENDIF
1168  006740                                                                50022$:
1169  006740  004737  015400                     JSR PC,NEXTU              ;FIND NEXT UNIT
1170  006744                                    ENDDO                       ;
1171  006744  000746                                                        BR      50016$
1172  006746                                                                50017$:
1173  006746                                    IF R2 EQ #1 THEN            ;IF ALL UNIT ARE AT EOT/BOT THEN:
1174  006746  020227  000001                                                CMP     R2,#1
1175  006752  001001                                                        BNE     50025$
1176  006754  000412                             BR        EXARTN          ;RETURN WITH R2 - #1.
1177  006756                                    ENDIF
1178  006756                                                                50025$:
1179  006756                                    LET NCNT := NCNT + #1       ;UPDATE RECORD COUNT.
1180  006756  005237  003340                                                INC     NCNT
1181  006762                                    LET PCMDWD := CMDWRD        ;SAVE PREVIOUS COMMAND WORD.
1182  006762  013737  003346  003352                                        MOV     CMDWRD,PCMDWD
1183  006770                                  ENDDO
1184  006770  000656                                                        BR      50010$
1185  006772                                                                50011$:
1186  006772  004737  014316                   JSR   PC,VFYDAT             ;IF LAST CMD WAS A WRITE VERIFY, THEN GO
1187                                                                        ;VERIFY THE LAST N RECORDS OF DATA.
1188  006776                                ENDDO
1189  006776  000646                                                        BR      50006$
1190  007000                                                                50007$:
1191  007000                                LET R2 := #0                    ;SET NORMAL RETURN INDICATOR.
1192  007000  005002                                                        CLR     R2
1193  007002  000207                 EXARTN: RTS PC                        ;RETURN.
1194
```

```
1195
1196
1197                                    ;       SUBROUTINE TO ISSUE COMMAND TO ALL DEVICES, WAIT FOR
1198                                    ;       ALL INTERRUPTS, AND CHECK ALL STATUS.
1199                                    ;       INPUTS:
1200                                    ;       OUTPUTS:
1201                                    ;       REGISTERS:
1202                                    ;       CALLS:          EXCUTE,GOWAIT,NEXTU,FIRSTU.
1203
1204  007004  004737  015332            EXSUB::     JSR PC,FIRSTU               ;SET UP FOR FIRST UNIT.
1205  007010                              WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE DEVICES:
1206  007010                                                                  50026$:
1207  007010  026527  002532  177777                                              CMP     DEVTBL(R5),#END
1208  007016  001465                                                             BEQ     50027$
1209  007020                                IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
1210  007020  032737  000400  003346                                              BIT     #MOD.CO,CMDWRD
1211  007026  001421                                                             BEQ     50030$
1212  007030                                  IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
1213  007030  032765  000002  003426                                              BIT     #XO.BOT,EOTFLG(R
1214  007036  001014                                                             BNE     50031$
1215  007040                                    IF #XO.EOT SETIN EOTFLG(R5) THEN  ;BUT IF AT EOT
1216  007040  032765  000001  003426                                              BIT     #XO.EOT,EOTFLG(R
1217  007046  001406                                                             BEQ     50032$
1218  007050                                      IFB ALLEOT NE #0 THEN       ;AND ALL OTHERS AT EOT
1219  007050  105737  003450                                                      TSTB    ALLEOT
1220  007054  001402                                                             BEQ     50033$
1221  007056  004737  010326                       JSR PC,EXCUTE              ;THEN EXECUTE REV CMD
1222  007062                                      ENDIF                         ;IF NOT ALL AT EOT, FREEZE UNIT(S) AT EO
1223  007062                                                                  50033$:
1224  007062                                    ELSE                          ;IF NOT AT BOT AND
1225  007062  000402                                                             BR      50034$
1226  007064                                                                  50032$:
1227  007064  004737  010326                     JSR PC,EXCUTE                ;NOT AT EOT, EXEC REV CMD
1228  007070                                    ENDIF
1229  007070                                                                  50034$:
1230  007070                                  ENDIF
1231  007070                                                                  50031$:
1232  007070                              ELSE                        ;ELSE IF CMD IS NOT REVERSE:
1233  007070  000435                                                             BR      50035$
1234  007072                                                                  50030$:
1235  007072                                IF CMDLG EQ #2 AND #XO.BOT SETIN EOTFLG(R5) THEN
1236  007072  023727  003354  000002                                              CMP     CMDLG,#2
1237  007100  001011                                                             BNE     50036$
1238  007102  032765  000002  003426                                              BIT     #XO.BOT,EOTFLG(R
1239  007110  001405                                                             BEQ     50036$
1240                                    ;CLEAR BAD SPOT COUNTS WHEN WRITING FROM BOT
1241  007112                                  LET BTPT := BTADDR(R5)
1242  007112  016537  002544  003436                                              MOV     BTADDR(R5),BTPT
1243  007120                                  LET @BTPT := #0
1244  007120  005077  174312                                                     CLR     @BTPT
1245  007124                                ENDIF
1246  007124                                                                  50036$:
1247  007124                                IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
1248  007124  032765  000001  003426                                              BIT     #XO.EOT,EOTFLG(R
1249  007132  001404                                                             BEQ     50037$
1250  007134  032737  000001  003346                                              BIT     #CMD.CO,CMDWRD
```

```
007142  001003                                                                      BNE     50040$
007144                                                                      50037$:
                                                                ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
007144  004737  010326                          JSR PC,EXCUTE     ;ISSUE CMD TO TS04
007150                                         ELSE
007150  000405                                                                      BR      50041$
007152                                                                      50040$:
007152                                          IFB ALLEOT NE #0 THEN
007152  105737  003450                                                              TSTB    ALLEOT
007156  001402                                                                      BEQ     50042$
007160  004737  010326                           JSR PC,EXCUTE
007164                                          ENDIF
007164                                                                      50042$:
007164                                         ENDIF
007164                                                                      50041$:
007164                                        ENDIF
007164                                                                      50035$:
007164  004737  015400                        JSR       PC,NEXTU        ;FIND NEXT UNIT IN TEST CYCLE.
007170                                       ENDDO
007170  000707                                                                      BR      50026$
007172                                                                      50027$:
007172                                       IFB RPTFLG NE #0 THEN      ;IF REPORT HAS BEEN REQUESTED THEN:
007172  105737  003443                                                              TSTB    RPTFLG
007176  001403                                                                      BEQ     50043$
007200                                        LET RPTFLG :B- #0         ;CLR THE FLAG.
007200  105037  003443                                                              CLRB    RPTFLG
007204                                        DORPT                     ;PRINT THE PERFORMANCE REPORT.
007204  104424                                                                      TRAP    C$DRPT
007206                                       ENDIF                      ;
007206                                                                      50043$:
007206  004737  015332                       JSR PC,FIRSTU              ;SET UP FOR FIRST UNIT.
007212                                       WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE DEVICES:
007212                                                                      50044$:
007212  026527  002532  177777                                                      CMP     DEVTBL(R5),#END
007220  001450                                                                      BEQ     50045$
007222                                        IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
007222  032737  000400  003346                                                      BIT     #MOD.CO,CMDWRD
007230  001421                                                                      BEQ     50046$
007232                                         IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
007232  032765  000002  003426                                                      BIT     #XO.BOT,EOTFLG(R
007240  001014                                                                      BNE     50047$
007242                                          IF #XO.EOT SETIN EOTFLG(R5) THEN     ;BUT IF AT EOT
007242  032765  000001  003426                                                      BIT     #XO.EOT,EOTFLG(R
007250  001406                                                                      BEQ     50050$
007252                                           IFB ALLEOT NE #0 THEN          ;AND ALL OTHERS AT EOT
007252  105737  003450                                                              TSTB    ALLEOT
007256  001402                                                                      BEQ     50051$
007260  004737  010636                            JSR PC,GOWAIT                 ;THEN WAIT FOR CMD END
007264                                           ENDIF                          ;IF NOT ALL AT EOT, DO NOT WAIT
007264                                                                      50051$:
007264                                          ELSE                            ;NOT AT BOT, AND NOT AT EOT
007264  000402                                                                      BR      50052$
007266                                                                      50050$:
007266  004737  010636                           JSR PC,GOWAIT                         ;WAIT FOR INT,CHECK STAT
007272                                          ENDIF
007272                                                                      50052$:
```

```
007272                                    ENDIF
007272                                                                          50047$:
007272                                  ELSE                      ;ELSE IF CMD IS FORWARD:
007272  000420                                                                          BR      50053$
007274                                                                          50046$:
007274                                    IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
007274  032765  000001  003426                                                          BIT     #XO.EOT,EOTFLG(R
007302  001404                                                                          BEQ     50054$
007304  032737  000001  003346                                                          BIT     #CMD.CO,CMDWRD
007312  001003                                                                          BNE     50055$
007314                                                                          50054$:
                                                                  ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
007314  004737  010636                      JSR PC,GOWAIT           ;WAIT FOR INT,CHECK STATUS.
007320                                    ELSE
007320  000405                                                                          BR      50056$
007322                                                                          50055$:
007322                                      IFB ALLEOT NE #0 THEN
007322  105737  003450                                                                  TSTB    ALLEOT
007326  001402                                                                          BEQ     50057$
007330  004737  010636                        JSR PC,GOWAIT
007334                                      ENDIF
007334                                                                          50057$:
007334                                    ENDIF
007334                                                                          50056$:
007334                                  ENDIF
007334                                                                          50053$:
007334  004737  015400                  JSR       PC,NEXTU        ;FIND NEXT UNIT IN TEST CYCLE.
007340                                ENDDO
007340  000724                                                                          BR      50044$
007342                                                                          50045$:
007342  000207                          RTS PC                    ;RETURN.
```

```
                                GLOBAL SUBROUTINES SECTION

                        ;       THIS SUBROUTINE STORES THE ASCII FOR THE CURRENT COMMAND AND PREVIOUS
                        ;       COMMAND IN THE STANDARD ERROR MESSAGE.  ON ENTRY LOCATION CMDWRD
                        ;       CONTAINS CURRENT CMD AND LOCATION PCMDWD CONTAINS PREVIOUS CMD.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:      R3, R4.
                        ;       CALLS:          GCMDA

007344                  CMDAC:: LET R4 := CMDWRD                ;R4 = CMD BINARY.
007344  013704  003346                                                  MOV     CMDWRD,R4
007350  004737  007416          JSR     PC,GCMDA                ;GET CMD ASCII.
007354  112337  005706          MOVB    (R3)+,STAER1+2          ;MOVE CMD ASCII
007360  112337  005707          MOVB    (R3)+,STAER1+3          ;
007364  111337  005710          MOVB    (R3),STAER1+4           ;INTO MSG.
007370                          LET R4 := PCMDWD                ;R4 = PREVIOUS CMD BINARY.
007370  013704  003352                                                  MOV     PCMDWD,R4
007374  004737  007416          JSR     PC,GCMDA                ;GET CMD ASCII.
007400                          LET STAER7+24 :B= (R3)+         ;MOVE CMD ASCII
007400  112337  006022                                                  MOVB    (R3)+,STAER7+24
007404                          LET STAER7+25 :B= (R3)+         ;
007404  112337  006023                                                  MOVB    (R3)+,STAER7+25
007410                          LET STAER7+26 :B= (R3)          ;INTO MSG.
007410  111337  006024                                                  MOVB    (R3),STAER7+26
007414  000207                  RTS     PC                      ;RETURN. GO EXECUTE NEXT FUNCTION.


                        ;       SUBROUTINE TO FIND THE ASCII EQUIVILENT OF THE COMMAND IN R4.
                        ;       ADDRESS OF ASCII 1ST WORD IS RETURNED IN R3.
                        ;       INPUTS:         R4 = PRESENT COMMAND WORD.
                        ;       OUTPUTS:        R3 = ADDRESS OF PRESENT COMMAND ASCII.
                        ;       REGISTERS:
                        ;       CALLS:

007416                  GCMDA:: LET R3 := #0                    ;INIT CMD TBL POINTER.
007416  005003                                                          CLR     R3
007420                          WHILE CMDTBL(R3) NE R4 DO       ;UNTIL CURRENT CMD IS FOUND:
007420                                                                  50060$:
007420  026304  003562                                                  CMP     CMDTBL(R3),R4
007424  001403                                                          BEQ     50061$
007426                            LET R3 := R3 + #2             ;SEARCH CMD TABLE.
007426  062703  000002                                                  ADD     #2,R3
007432                          ENDDO
007432  000772                                                          BR      50060$
007434                                                                  50061$:
007434                          LET R4 := R3
007434  010304                                                          MOV     R3,R4
007436                          LET R3 := R3 SHIFT -1           ;POINT TO ASCII FOR THAT COMMAND
007436  006203                                                          ASR     R3
007440  000240                  NOP
007442  060403                  ADD     R4,R3
007444  062703  003650          ADD     #CMDASC,R3
007450  000207                  RTS     PC                      ;RETURN.
```

```
1391                                    ;       THIS SUBROUTINE LOADS THE TS04 COMMAND PACKET FROM ONE
1392                                    ;       ENTRY IN THE SEQUENCE TABLE.
1393                                    ;       INPUTS:
1394                                    ;       OUTPUTS:
1395                                    ;       REGISTERS:      R2, R3.
1396                                    ;       CALLS:          GENPAT.
1397
1398  007452                            SETUP:: LET CMDLG := #0                 ;CLR CMD LOGGING CODE(DISABLES LOGGING)
1399  007452  005037  003354                                                           CLR     CMDLG
1400  007456  012137  002310                    MOV     (R1)+,CMDPKT           ;LOAD THE COMMAND WORD.
1401  007462  011137  002316                    MOV     (R1),CMDPKT+CP.CNT     ;LOAD THE BYTE/RECORD/FILE COUNT.
1402  007466  011137  003344                    MOV     (R1),BRFCNT            ;SAVE BRF FOR THIS COMMAND.
1403  007472  013702  002310                    MOV     CMDPKT,R2              ;GET CMD.
1404  007476  042702  177740                    BIC     #NCMD.C,R2             ;CLR ALL BUT CMD BITS.
1405  007502  010203                            MOV     R2,R3                  ;SAVE IT TWICE.
1406  007504  162703  000010                    SUB     #CMD.C3,R3             ;POSITION COMMAND?
1407  007510  001003                            BNE     2$                     ;BR IF NOT.
1408  007512  011137  002312                    MOV     (R1),CMDPKT+2          ;MOVE BPCR IN 2ND PKT WORD FOR POSITION CMD.
1409  007516  000461                            BR      3$
1410  007520                            2$:     IF CMDPKT EQ #WTM THEN         ;IF CMD IS A WRITE TAPE MARK THEN:
1411  007520  023727  002310  100011                                                   CMP     CMDPKT,#WTM
1412  007526  001003                                                                  BNE     50062$
1413  007530                                      LET CMDLG := #2               ;WTM LOGGING CODE IS 2.
1414  007530  012737  000002  003354                                                   MOV     #2,CMDLG
1415  007536                                    ENDIF
1416  007536                                                                                  50062$:
1417  007536  010203                            MOV     R2,R3
1418  007540  162703  000001                    SUB     #CMD.C0,R3             ;IS IT A READ?
1419  007544  001017                            BNE     1$                     ;BR IF NOT.
1420  007546  013737  003336  002312            MOV     DATARD,CMDPKT+CP.ADL   ;IF SO, LOAD THE BUFFER ADDR.
1421  007554                                    IF #MOD.C0 SETIN CMDPKT THEN   ;IF CMD IS A READ REV THEN:
1422  007554  032737  000400  002310                                                   BIT     #MOD.C0,CMDPKT
1423  007562  001404                                                                  BEQ     50063$
1424  007564                                      LET CMDLG := #4               ;LOGGING CODE IS 4.
1425  007564  012737  000004  003354                                                   MOV     #4,CMDLG
1426  007572                                    ELSE                           ;ELSE - IF CMD IS A READ FWD:
1427  007572  000403                                                                  BR      50064$
1428  007574                                                                                  50063$:
1429  007574                                      LET CMDLG := #6               ;LOGGING CODE IS 6.
1430  007574  012737  000006  003354                                                   MOV     #6,CMDLG
1431  007602                                    ENDIF
1432  007602                                                                                  50064$:
1433  007602  000427                            BR      3$                     ;CONTINUE.
1434  007604  010203                    1$:     MOV     R2,R3                  ;IS IT
1435  007606  162703  000004                    SUB     #CMD.C2,R3             ;A SET CHARACTERISTICS CMD?
1436  007612  001011                            BNE     4$                     ;BR IF NOT.
1437  007614                                    LET CMDPKT+CP.ADL := #SCHBK    ;SET UP ADR LO FOR SET CHAR.
1438  007614  012737  002442  002312                                                   MOV     #SCHBK,CMDPKT+CP
1439  007622  012737  000010  002316            MOV     #SCHCNT,CMDPKT+CP.CNT  ;SET BUFFER EXTENT
1440  007630                                    LET SCHBK+6 := (R1)            ;STORE CHARACTERISTIC CODE IN SCH BLOCK.
1441  007630  011137  002450                                                           MOV     (R1),SCHBK+6
1442  007634  000412                            BR      3$                     ;CONTINUE.
1443  007636  010203                    4$:     MOV     R2,R3                  ;IS IT
1444  007640  162703  000006                    SUB     #CMD.C1!CMD.C2,R3      ;A DIAGNOSTIC (DIA) CMD?
1445  007644  001006                            BNE     3$                     ;BR IF NOT.
1446  007646  012737  000020  002316            MOV     #DIACNT,CMDPKT+CP.CNT  ;LOAD BUFFER EXTENT.
```

```
1447  007654  012737  003334  002312          MOV     #DIABLK,CMDPKT+CP.ADL   ;LOAD BUFFER ADR LOW.
1448  007662  005721                  3$:     TST     (R1)+                   ;POINT TO N (NUMBER OF TIMES TO EXECUTE THIS INS
1449  007664                                  LET NCNT1 := (R1)+              ;SAVE NUMBER OF OPERATIONS
1450  007664  012137  003342                                                          MOV     (R1)+,NCNT1
1451  007670
1452  007670  005037  003340                  LET NCNT := #0                  ;CLEAR OPERATION COUNTER.
1453  007674  012137  003374                                                          CLR     NCNT
1454  007700  010203                          MOV     (R1)+,PATERN            ;SAVE PATTERN CODE FOR CURRENT CMD.
1455  007702  162703  000005                  MOV     R2,R3                   ;IS IT
1456  007706  001010                          SUB     #CMD.CO!CMD.C2,R3       ;A WRITE?
1457  007710  013737  003334  002312          BNE     5$                      ;BR IF NOT.
1458  007716  004737  010030                  MOV     DATAWT,CMDPKT+CP.ADL    ;LOAD WRITE BUFFER LO ORDER.
1459  007722                                  JSR     PC,GENPAT               ;GO GENERATE THE WRITE PATTERN.
1460  007722  012737  000002  003354          LET CMDLG := #2                 ;WRITE LOGGING CODE IS 2.
1461  007730                                                                          MOV     #2,CMDLG
1462  007730  032737  000100  002310  5$:     IF #VFY.C SETIN CMDPKT THEN     ;IF DATA VERIFICATION IS REQUIRED:
1463  007736  001407                                                                  BIT     #VFY.C,CMDPKT
1464  007740                                                                          BEQ     50065$
1465  007740  112737  000001  003442            LET VFYFLG :B= #1              ;SET VERIFY FLAG.
1466  007746  042737  000100  002310                                                  MOVB    #1,VFYFLG
1467  007754                                    BIC   #VFY.C,CMDPKT           ;CLEAR VERIFY BIT(NOT USED BY HARDWARE).
1468  007754  000402                          ELSE                            ;IF DATA VERIFICATION IS NOT REQUIRED:
1469  007756                                                                          BR      50066$
1470  007756                                                                  50065$:
1471  007756  105037  003442                    LET VFYFLG :B= #0             ;CLR VERIFY FLAG.
1472  007762                                                                          CLRB    VFYFLG
1473  007762                                  ENDIF
1474  007762                                                                  50066$:
1475  007762  013737  003346  003352          LET PCMDWD := CMDWRD             ;SAVE PREVIOUS CMD WORD.
1476  007770                                                                          MOV     CMDWRD,PCMDWD
1477  007770  013737  002310  003346          LET CMDWRD := CMDPKT             ;SAVE PRESENT CMD WORD.
1478  007776                                                                          MOV     CMDPKT,CMDWRD
1479  007776  105737  003444                  IFB SWBFLG NE #0 THEN            ;IF SWAP BYTES IS ENABLED:
1480  010002  001403                                                                 TSTB    SWBFLG
1481  010004                                                                          BEQ     50067$
1482  010004  052737  010000  002310            LET CMDPKT := CMDPKT SET.B' #SWB.C  ;SET SWAP BIT IN COMMAND.
1483  010012                                                                          BIS     #SWB.C,CMDPKT
1484  010012                                  ENDIF
1485  010012  042737  004000  002310                                          50067$:
1486  010020                                  BIC     #BRF.C,CMDPKT           ;CLR BRF BIT (INTERNAL ONLY).
1487  010020  013737  002310  003350          LET CMDSAV := CMDPKT             ;SAVE 1ST WORD OF COMMAND PACKET.
1488  010026  000207                                                                  MOV     CMDPKT,CMDSAV
                                            RTS     PC                      ;RETURN.
```

```
                                        GLOBAL SUBROUTINES SECTION

                                ;       THIS SUBROUTINE SETS UP AND CALLS THE APPROPRIATE SUBROUTINE TO GENERATE
                                ;       THE DESIRED PATTERN FOR THE WRITE AND WRITE/VERIFY COMMANDS.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2, R3, R4.
                                ;       CALLS:          PATR0 - PATR7

010030                          GENPAT:: LET R3 := PATERN SHIFT 1       ;SETUP PATTERN ROUTINE POINTER
010030  013703  003374                                                                  MOV     PATERN,R3
010034  006303                                                                        ASL     R3
010036                                  LET R4 := BRFCNT + #1   ;SET LENGTH OF WRITE BFR
010036  013704  003344                                                                  MOV     BRFCNT,R4
010042  005204                                                                        INC     R4
010044                                  LET R4 := R4 CLR.BY #1  ;ROUNDED UP TO NEXT WORD
010044  042704  000001                                                                  BIC     #1,R4
010050                                  LET R4 :- R4 - #2       ;WITH FIRST WORD RESERVED
010050  162704  000002                                                                  SUB     #2,R4
010054                                  LET R2 := DATAWT + #2   ;FOR RECORD COUNT
010054  013702  003334                                                                  MOV     DATAWT,R2
010060  062702  000002                                                                  ADD     #2,R2
010064  004773  010072                  JSR     PC,@PATTBL(R3)  ;GO GENERATE THE APPROPRIATE PATTERN.
010070  000207                          RTS     PC              ;RETURN TO SETUP SUBROUTINE.

                                        ;TS04 WRITE PATTERN LOOKUP TABLE.  USED TO JSR TO THE
                                        ;CORRECT DATA PATTERN GENERATING ROUTINE.

010072  010114                  PATTBL: PATR0
010074  010152                          PATR1
010076  010172                          PATR2
010100  010202                          PATR3
010102  010226                          PATR4
010104  010240                          PATR5
010106  010252                          PATR6
010110  010272                          PATR7
010112  010324                          PATR8


                                        ;INCREMENTING PATTERN.  0 - 377.

010114                          PATR0:: LET R3 := #400
010114  012703  000400                                                                  MOV     #400,R3
010120                          1$:     LET R4 := R4 - #2       ;DECREMENT WORD COUNT.
010120  162704  000002                                                                  SUB     #2,R4
010124  100411                          BMI     2$              ;BR IF DONE.
010126                                  LET (R2)+ := R3         ;STORE DATA WORD.
010126  010322                                                                        MOV     R3,(R2)+
010130                                  LET R3 := R3 + #1002    ;UPDATE PATTERN.
010130  062703  001002                                                                  ADD     #1002,R3
010134                                  IF R3 EQ #1000 THEN     ;IF PATTERN HAS WRAPPED AROUND THEN:
010134  020327  001000                                                                  CMP     R3,#1000
010140  001002                                                                        BNE     50070$
010142                                    LET R3 := #400          ;INIT THE PATTERN AGAIN.
010142  012703  000400                                                                  MOV     #400,R3
010146                                  ENDIF
010146                                                                          50070$:
010146  000764                          BR      1$              ;DO IT AGAIN.
```

```
1545  010150  000207                2$:     RTS     PC              ;RETURN.
1546
1547                                        ;ALL ONE'S PATTERN.
1548
1549  010152  012703  177777        PATR1:: MOV     #-1,R3          ;ALL ONES PATTERN;.
1550  010156                        ZROPAT: LET R4 := R4 - #2       ;DECREMENT BYTE COUNT.
1551  010156  162704  000002                                                 SUB     #2,R4
1552  010162  100402                        BMI     1$              ;DONE?,BR IF YES.
1553  010164  010322                        MOV     R3,(R2)+        ;IF NOT LOAD NEXT BYTE WITH PATTERN.
1554  010166  000773                        BR      ZROPAT          ;DO IT AGAIN.
1555
1556  010170  000207                1$:     RTS     PC              ;RETURN.
```

```
                                        ;ALL ZEROES PATTERN.

010172  005003          PATR2:: CLR     R3              ;CLR PATTERN REGISTER.
010174  004737  010156          JSR     PC,ZROPAT       ;GO GENERATE IT.
010200  000207                  RTS     PC              ;RETURN.

                                        ;ONE BIT WALKING FROM R TO L IN A FIELD OF ZEROES.

010202  012703  000401  PATR3:: MOV     #401,R3         ;INIT PATTERN REGISTER.
010206          WLKZRO: LET R4 := R4 - #2               ;DECREMENT WORD COUNT.
010206  162704  000002                                                   SUB     #2,R4
010212  100404                  BMI     1$              ;BR IF DONE.
010214  010322                  MOV     R3,(R2)+        ;LOAD DATA.
010216  006303                  ASL     R3              ;SHIFT PATTERN.
010220  005503                  ADC     R3              ;ADD CARRY BACK INTO PATTERN.
010222  000771                  BR      WLKZRO          ;DO IT AGAIN.
010224  000207          1$:     RTS     PC              ;RETURN.

                                        ;ZERO BIT WALKING FROM R TO L IN A FIELD OF 1'S.

010226  012703  177376  PATR4:: MOV     #177376,R3      ;INIT PATTERN REGISTER.
010232  004737  010206          JSR     PC,WLKZRO       ;GO GENERATE ;IT.
010236  000207                  RTS     PC              ;RETURN.

                                        ;ALTERNATING ONE AND ZERO BITS WITH ALTERNATE BYTES
                                        ;COMPLEMENTED.

010240  012703  125125  PATR5:: MOV     #125125,R3      ;INIT PATTERN REGISTER.
010244  004737  010156          JSR     PC,ZROPAT       ;GO GENERATE IT.
010250  000207                  RTS     PC              ;RETURN.

                                        ;ALTERNATING BYTES OF 000 AND 377.

010252  012703  177400  PATR6:: MOV     #177400,R3      ;INIT PATTERN REGISTER.
010256          1$:     LET R4 := R4 - #2               ;DECREMENT WORD COUNT.
010256  162704  000002                                                   SUB     #2,R4
010262  100402                  BMI     2$              ;BR IF DONE.
010264  010322                  MOV     R3,(R2)+        ;LOAD DATA.
010266  000773                  BR      1$              ;DO IT AGAIN.
010270  000207          2$:     RTS     PC              ;RETURN.

                                        ;RANDOM PATTERN GENERATOR

010272          PATR7:: LET R4 :- R4 - #2               ;DECREMENT WORD COUNT
010272  162704  000002                                                   SUB     #2,R4
010276  100411                  BMI     GIT             ;BR IF DONE.
010300  063737  003362  003360          ADD     RANS,RANB
010306  063737  003360  003362          ADD     RANB,RANS       ;GET NEW #.
010314  013722  003362                  MOV     RANS,(R2)+      ;SAVE #.
010320  000764                  BR      PATR7           ;CONTINUE.
010322  000207          GIT:    RTS     PC              ;RETURN

                        ;       NO PATTERN GENERATION.

010324  000207          PATR8:: RTS     PC              ;RETURN.
```

```
                                ;       THIS SUBROUTINE INITIATES TS04 COMMAND EXECUTION
                                ;       AND CHECKS FOR TS04 RESPONSE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2, R3.
                                ;       CALLS:          DROPU, MOVMSG, FIRSTU, NEXTU, WSSR.

010326                          EXCUTE:: LET TIME1 :- #-1               ;INIT TIMEOUT COUNTER.
010326  012737  177777  003364                                                  MOV     #-1,TIME1
010334                          REPEAT                                  ;WAIT -
010334                                                                                  50071$:
010334                            LET TIME1 := TIME1 - #1               ;UPDATE TIMEOUT COUNTER.
010334  005337  003364                                                          DEC     TIME1
010340                            IF TIME1 EQ #0 THEN                   ;IF TIMED OUT:
010340  005737  003364                                                          TST     TIME1
010344  001011                                                                BNE     50072$
010346  004737  011206                JSR PC,MOVMSG                       ;MOVE CURRENT PACKET MSG.
010352                                ERRDF #2,NSSRM,STAERM               ;REPORT TS04 NOT READY
010352  104455                                                                  TRAP    C$ERDF
010354  000002                                                                  .WORD   2
010356  004271                                                                  .WORD   NSSRM
010360  005372                                                                  .WORD   STAERM
010362  004737  015430                JSR PC,DROPU                        ;DROP THE UNIT.
010366  000522                        BR  EXCRTN                          ;RETURN.
010370                            ENDIF
010370                                                                                  50072$:
010370                          UNTIL #TS.SSR SETIN @TSSR(R5)           ;WAIT UNTIL DEVICE IS READY.
010370  032775  000200  002462                                                  BIT     #TS.SSR,@TSSR(R5
010376  001756                                                                  BEQ     50071$
010400                          IF CMDWRD EQ #SCH THEN                  ;IF WE ARE DOING A SET CHAR CMD THEN:
010400  023727  003346  140004                                                  CMP     CMDWRD,#SCH
010406  001022                                                                BNE     50073$
010410                            LET R5SAVE := R5                        ;SAVE CURRENT DEVICE POINTER.
010410  010537  003400                                                          MOV     R5,R5SAVE
010414  004737  015332                JSR   PC,FIRSTU                     ;FIND FIRST UNIT.
010420                            WHILE DEVTBL(R5) NE #END DO
010420                                                                                  50074$:
010420  026527  002532  177777                                                  CMP     DEVTBL(R5),#END
010426  001405                                                                BEQ     50075$
010430  004737  011152                JSR PC,WSSR                           ;WAIT FOR UNIT READY OR TIME OUT.
010434  004737  015400                JSR PC,NEXTU                          ;FIND NEXT UNIT.
010440                            ENDDO
010440  000767                                                                  BR      50074$
010442                                                                                  50075$:
010442                            LET R5 := R5SAVE                        ;RESTORE CURRENT DEVICE POINTER.
010442  013705  003400                                                          MOV     R5SAVE,R5
010446                            LET SCHBK := MSGPKA(R5)                 ;SET UP ADR OF MSG PKT IN SCH BLOCK.
010446  016537  002502  002442                                                  MOV     MSGPKA(R5),SCHBK
010454                          ENDIF
010454                                                                                  50073$:
010454                          LET R3 := MSGPKA(R5)                    ;ADR OF THIS UNIT'S MSG PACKET.
010454  016503  002502                                                          MOV     MSGPKA(R5),R3
010460                          LET R2 := #0                            ;CLR COUNTER.
010460  005002                                                                  CLR     R2
010462                          WHILE R2 NE #MSGCNT DO                  ;WHILE THERE ARE MORE LOCATIONS:
010462                                                                                  50076$:
```

```
1668  010462  020227  000016                                                        CMP     R2,#MSGCNT
1669  010466  001405                                                              BEQ     50077$
1670  010470                          LET (R3)+ := #-1            ;INIT THE MSG PACKET WITH ALL 1'S
1671  010470  012723  177777                                                      MOV     #-1,(R3)+
1672  010474                          LET R2 := R2 + #2           ;UPDATE COUNTER.
1673  010474  062702  000002                                                      ADD     #2,R2
1674  010500                        ENDDO
1675  010500  000770                                                              BR      50076$
1676  010502                                                                  50077$:
1677  010502  105737  002210        TSTB    DINT                ;ARE INTERRUPTS DISABLED.
1678  010506  001023                BNE     1$                  ;BR IF YES.
1679  010510                        IFB INTFLG(R5) GT #1 THEN   ;IF MORE THAN ONE INTERRUPT HAS OCCURED:
1680  010510  126527  003416  000001                                              CMPB    INTFLG(R5),#1
1681  010516  003412                                                              BLE     50100$
1682  010520                          LET TSSREG := @TSSR(R5)           ;FREEZE THE CURRENT STATUS REG FOR PRINT
1683  010520  017537  002462  003402                                              MOV     @TSSR(R5),TSSREG
1684  010526                          ERRDF #15,TOOMM,STAERM    ;REPORT TOO MANY INTERRUPTS.
1685  010526  104455                                                              TRAP    C$ERDF
1686  010530  000017                                                              .WORD   15
1687  010532  004460                                                              .WORD   TOOMM
1688  010534  005372                                                              .WORD   STAERM
1689  010536  004737  015430          JSR PC,DROPU              ;DROP THE UNIT
1690  010542  000434                  BR  EXCRTN                ;RETURN - UNIT HAS BEEN DROPPED.
1691  010544                        ENDIF
1692  010544                                                                  50100$:
1693  010544                        LET INTFLG(R5) := #0          ;CLR INTERRUPT FLAG FOR THIS DEV.
1694  010544  005065  003416                                                      CLR     INTFLG(R5)
1695  010550  052737  000200  002310  BIS     #IE.C,CMDPKT        ;SET INT ENABLE BIT.
1696  010556                  1$:     IFB ERRREC EQ #0 THEN       ;IF NOT RETRYING
1697  010556  105737  003415                                                      TSTB    ERRREC
1698  010562  001005                                                              BNE     50101$
1699  010564                          LET RECCNT(R5) := RECCNT(R5) + #1
1700  010564  005265  003324                                                      INC     RECCNT(R5)
1701  010570                          LET @DATAWT := RECCNT(R5)   ;THEN UPDATE REC COUNT TO WRITE IT ON TAPE
1702  010570  016577  003324  172536                                              MOV     RECCNT(R5),@DATA
1703  010576                        ENDIF
1704  010576                                                                  50101$:
1705  010576  012775  002310  002452  MOV     #CMDPKT,@TSDB(R5)   ;LOAD TSDB WITH CMDPKT ADDRESS
1706                                                              ;THIS INITIATES COMMAND EXECUTION.
1707  010604                        IF #TS.SSR SETIN @TSSR(R5) THEN ;IF READY DID NOT DROP THEN:
1708  010604  032775  000200  002462                                              BIT     #TS.SSR,@TSSR(R5
1709  010612  001410                                                              BEQ     50102$
1710  010614  004737  011206          JSR   PC,MOVMSG             ;MOVE CURRENT MESSAGE PACKET TO COMMON.
1711  010620                          ERRDF #3,TOERM,STAERM       ;REPORT NO TS04 RESPONSE.
1712  010620  104455                                                              TRAP    C$ERDF
1713  010622  000003                                                              .WORD   3
1714  010624  004207                                                              .WORD   TOERM
1715  010626  005372                                                              .WORD   STAERM
1716  010630  004737  015430          JSR PC,DROPU              ;DROP THE UNIT
1717  010634                        ENDIF
1718  010634                                                                  50102$:
1719  010634  000207          EXCRTN: RTS     PC                ;RETURN.
```

```
                        ;       THIS SUBROUTINE WAITS FOR THE TS04 INERRUPT OR DONE BIT TO SET AND ALLOWS THE
                        ;       OPERATOR TO TRANSFER CONROL TO THE SUPERVISOR.
                        ;       UPON APPEARANCE OF THE INTERRUPT OR DONE, CHECK TSSR FOR STATUS ERRORS,
                        ;       LOG BYTES AND ERRORS AND PERFORM ERROR RECOVERY IF NESSASARY.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:      R2, R3.
                        ;       CALLS:          DROPU, MOVMSG, RECUD, CHKERR, LOG, CLRERR.

010636                  GOWAIT:: LET TIME1 :  #-1               ;INIT TIME OUT COUNTER.
010636  012737  177777  003364                                          MOV     #-1,TIME1
010644                          REPEAT                          ;REPEAT UNTIL INTERRUPT OCCURES:
010644                                                                  50103$:
010644                            BREAK                         ;GO TO THE SUPER TO ALLOW TTY INPUT.
010644  104422                                                          TRAP    C$BRK
010646                            IF CMDWRD EQ #RWD THEN        ;IF COMMAND WAS REWIND THEN:
010646  023727  003346  102010                                          CMP     CMDWRD,#RWD
010654  001014                                                          BNE     50104$
010656                              DELAY 10.                   ;WAIT  EXTRA MSECS EACH LOOP.
010656  012727  000012                                                  MOV     #10.,(PC)+
010662  000000                                                          .WORD   0
010664  013727  002116                                                  MOV     L$DLY,(PC)+
010670  000000                                                          .WORD   0
010672  005367  177772                                                  DEC     -6(PC)
010676  001375                                                          BNE     .-4
010700  005367  177756                                                  DEC     -22(PC)
010704  001367                                                          BNE     .-20
010706                            ENDIF
010706                                                                  50104$:
010706                          IF CMDWRD EQ #SFF OR CMDWRD EQ #SFR THEN
010706  023727  003346  105010                                          CMP     CMDWRD,#SFF
010714  001404                                                          BEQ     50105$
010716  023727  003346  105410                                          CMP     CMDWRD,#SFR
010724  001014                                                          BNE     50106$
010726                                                                  50105$:
010726                            DELAY 12.                   ;ADD DELAY FOR SPACE TAPE MARK COMMANDS
010726  012727  000014                                                  MOV     #12.,(PC)+
010732  000000                                                          .WORD   0
010734  013727  002116                                                  MOV     L$DLY,(PC)+
010740  000000                                                          .WORD   0
010742  005367  177772                                                  DEC     -6(PC)
010746  001375                                                          BNE     .-4
010750  005367  177756                                                  DEC     -22(PC)
010754  001367                                                          BNE     .-20
010756                          ENDIF
010756                                                                  50106$:
010756                          IFB DINT EQ #0 THEN             ;IF INTERRUPTS ARE ENABLED.
010756  105737  002210                                                  TSTB    DINT
010762  001003                                                          BNE     50107$
010764                            LET R2 := INTFLG(R5)          ;FETCH INTERRUPT OCCURRED FLAG.
010764  016502  003416                                                  MOV     INTFLG(R5),R2
010770                          ELSE                            ;IF IN BRUTUS MODE:
010770  000406                                                          BR      50110$
010772                                                                  50107$:
010772                            LET R3 := COMP #TS.SSR        ;SET UP A MASK FOR THE DONE BIT.
010772  012703  000200                                                  MOV     #TS.SSR,R3
```

```
010776  005103                                                                            COM     R3
011000                                  LET R2 := @TSSR(R5) CLR.BY R3  ;FETCH DONE BIT.
011000  017502  002462                                                                    MOV     @TSSR(R5),R2
011004  040302                                                                            BIC     R3,R2
011006                                ENDIF
011006                                                                            50110$:
011006                                LET TIME1 := TIME1 - #1        ;UPDATE TIMEOUT COUNTER.
011006  005337  003364                                                                    DEC     TIME1
011012                              UNTIL R2 NE #0 OR TIME1 EQ #0    ;REPEAT UNTIL INTERRUPT OR READY OCCURES.
011012  005702                                                                            TST     R2
011014  001003                                                                            BNE     50111$
011016  005737  003364                                                                    TST     TIME1
011022  001310                                                                            BNE     50103$
011024                                                                            50111$:
011024                              IF TIME1 EQ #0 THEN              ;IF TIME OUT HAS OCCURRED:
011024  005737  003364                                                                    TST     TIME1
011030  001022                                                                            BNE     50112$
011032                                LET @DATAWT := RECCNT(R5) - #1 ;RE-ADJUST REC COUNT DOWN
011032  016577  003324  172274                                                            MOV     RECCNT(R5),@DATA
011040  005377  172270                                                                    DEC     @DATAWT
011044  004737  011206                JSR   PC,MOVMSG                ;MOVE CURRENT MSG PACKET TO COMMON AREA.
011050                                ERRDF #4,NOINTM,STAERM         ;REPORT NO INTERRUPT.
011050  104455                                                                            TRAP    C$ERDF
011052  000004                                                                            .WORD   4
011054  004421                                                                            .WORD   NOINTM
011056  005372                                                                            .WORD   STAERM
011060  004737  015430                JSR PC,DROPU           ;DROP THE UNIT.
011064                                LET R3 :- #ENDERF
011064  012703  003416                                                                    MOV     #ENDERF,R3
011070  004737  011136                JSR PC,CLRERR                  ;CLEAR ALL ERROR FLAGS
011074                              ELSE
011074  000417                                                                            BR      50113$
011076                                                                            50112$:
011076  004737  011206                JSR   PC,MOVMSG                ;MOVE CURRENT MSG. PACKET TO COMMON AREA.
011102  004737  011272                JSR   PC,RECUD                 ;UPDATE THE RECORD COUNT.
011106  004737  011440                JSR   PC,CHKERR                ;CHECK FOR STATUS ERRORS.
011112                                IFB WRTYFG EQ #0 THEN          ;
011112  105737  003407                                                                    TSTB    WRTYFG
011116  001006                                                                            BNE     50114$
011120  004737  014016                  JSR PC,LOG                   ;LOG BYTES AND ERRORS.
011124                                  LET R3 := #ENDERF
011124  012703  003416                                                                    MOV     #ENDERF,R3
011130  004737  011136                  JSR PC,CLRERR                        ;CLEAR ALL ERROR FLAGS
011134                                ENDIF
011134                                                                            50114$:
011134                              ENDIF
011134                                                                            50113$:
011134  000207                      RTS     PC                       ;RETURN IF DONE.
```

```
1824                                    ;       SUBROUTINE TO CLEAR FLAGS.
1825                                    ;       INPUTS:         R3 = LWA TO BE CLEARED + 2.
1826                                    ;       OUTPUTS:
1827                                    ;       REGISTERS:      R2
1828                                    ;       CALLS:
1829
1830  0111_                             CLRERR:: LET R2 := #BGNFLG
1831  011136                                                                          MOV     #BGNFLG,R2
1832  01114?                                    REPEAT
1833  011142                                                                  50115$:
1834  011142                                      LET (R2)+ := #0
1835  011142  005c.                                                                   CLR     (R2)+
1836  011144                                    UNTIL R2 EQ R3
1837  011144  020203                                                                 CMP     R2,R3
1838  011146  001375                                                                 BNE     50115$
1839  011150  000207                           RTS PC
1840
1841
1842
1843                                    SUBROUTINE TO WAIT UNTIL CURRENT UNIT IS READY OR UNTIL TIME OUT.
1844                                    INPUTS:
1845                                    OUTPUTS:
1846                                    REGISTERS:
1847                                    CALLS:
1848
1849  011152                            ,     ::  LET TIME1 := #-1                ;INIT TIMEOUT COUNTER.
1850  011152  012737  177777  003364                                                 MOV     #-1,TIME1
1851  011160                                    REPEAT                            ;REPEAT UNTIL DEV READY OR TIMEOUT:
1852  011160                                                                  50116$:
1853  011160                                      BREAK                             ;BREAK TO THE SUPERVISOR.
1854  011160  104422                                                                 TRAP    C$BRK
1855  011162                                      LET TIME1 := TIME1 - #1           ;UPDATE TIMEOUT COUNTER.
1856  011162  005337  003364                                                        DEC     TIME1
1857  011166                                  UN IL #TS.SSR SETIN @TSSR(R5) OR TIME1 EQ #0
1858  011166  032775  000200  002462                                               BIT     #TS.SSR,@TSSR(R5
1859  011174  001003                                                                BNE     50117$
1860  011176  005737  003364                                                       TST     TIME1
1861  011202  001366                                                                BNE     50116$
1862  011204                                                                  50117$:
1863                                                                    ;REPEAT UNTIL DEV READY OR TIMEOUT.
1864  011204  000207                           RTS PC                  ;RETURN.
1865
1866
```

```
1867
1868                                    ;       SUBROUTINE TO MOVE THE CURRENT MESSAGE PACKET TO THE COMMON AREA AND
1869                                    ;       TO UPDATE THE CURRENT TERMINATION CLASS CODE.
1870                                    ;       INPUTS:
1871                                    ;       OUTPUTS:
1872                                    ;       REGISTERS:      R2, R3.
1873                                    ;       CALLS:
1874
1875  011206                            MOVMSG:. LET TSSREG := @TSSR(R5)        ;FREEZE THE STATUS REG CONTENTS
1876  011206  017537  002462  003402                                                MOV     @TSSR(R5),TSSREG
1877  011214                                    LET R2 := TSSREG CLR.BY #TSC.TCC ;EXTRACT THE TERMINATION CLASS CODE.
1878  011214  013702  003402                                                       MOV     TSSREG,R2
1879  011220  042702  177761                                                       BIC     #TSC.TCC,R2
1880  011224                                    LET CTCC := R2 SHIFT -1         ;AND SAVE IT
1881  011224  010237  003376                                                       MOV     R2,CTCC
1882  011230  006237  003376                                                       ASR     CTCC
1883  011234                                    LET R3 := MSGPKA(R5)            ;ADR OF THIS DEVICE'S MSG.
1884  011234  016503  002502                                                       MOV     MSGPKA(R5),R3
1885  011240                                    LET R2 := #0                    ;CLR COUNTER.
1886  011240  005002                                                               CLR     P2
1887  011242                                    WHILE R2 NE #MSGCNT DO          ;WHILE THERE ARE MORE LOCATIONS:
1888  011242                                                                    50120$:
1889  011242  020227  000016                                                       CMP     R2,#MSGCNT
1890  011246  001405                                                               BEQ     50121$
1891  011250                                      LET MSGPKT(R2) := (R3)+       ;MOVE MSG TO COMMON AREA.
1892  011250  012362  002334                                                       MOV     (R3)+,MSGPKT(R2)
1893  011254                                      LET R2 := R2 + #2             ;UPDATE COUNTER.
1894  011254  062702  000002                                                       ADD     #2,R2
1895  011260                                    ENDDO
1896  011260  000770                                                               BR      50120$
1897  011262                                                                    50121$:
1898  011262                                    LET EOTFLG(R5) := MSGPKT+MS.XS0 ;MOVE XSTATO TO EOT FLAG.
1899  011262  013765  002342  003426                                                MOV     MSGPKT+MS.XS0,EO
1900  011270  000207                                    RTS     PC
```

```
1901                                    ;       SUBROUTINE TO ADJUST THE RECORD COUNT.
1902                                    ;       INPUTS:
1903                                    ;       OUTPUTS:
1904                                    ;       REGISTERS:
1905                                    ;       CALLS:
1906
1907  011272                            RECUD:: IFB RECLOG EQ #0 THEN           ;IF RECORD HAS NOT BEEN LOGGED:
1908  011272  105737  003411                                                              TSTB    RECLOG
1909  011276  001057                                                                      BNE     50122$
1910  011300                              LET RECCNT(R5) := RECCNT(R5) - #1
1911  011300  005365  003324                                                              DEC     RECCNT(R5)
1912  011304                              IF #BIT0 NOTSETIN CTCC AND #X2.OPM SETIN MSGPKT+MS.XS2 THEN   ;IF TAPE MOVED T
1913  011304  032737  000001  003376                                                      BIT     #BIT0,CTCC
1914  011312  001046                                                                      BNE     50123$
1915  011314  032737  100000  002346                                                      BIT     #X2.OPM,MSGPKT+M
1916  011322  001442                                                                      BEQ     50123$
1917  011324                                LET RECLOG :B= RECLOG + #1  ;SET RECORD LOGGED,
1918  011324  105237  003411                                                              INCB    RECLOG
1919  011330                                IF CMDWRD EQ #RWD THEN      ;IF THIS IS A REWIND CMD:
1920  011330  023727  003346  102010                                                      CMP     CMDWRD,#RWD
1921  011336  001003                                                                      BNE     50124$
1922  011340                                  LET RECCNT(R5) :  #0      ;CLEAR RECORD COUNT,
1923  011340  005065  003324                                                              CLR     RECCNT(R5)
1924  011344                                ELSE
1925  011344  000431                                                                      BR      50125$
1926  011346                                                                      50124$:
1927  011346                                  IF #BRF.C SETIN CMDWRD THEN       ;IF BRF USED, UPDATE RECORD COUNT.
1928  011346  032737  004000  003346                                                      BIT     #BRF.C,CMDWRD
1929  011354  001425                                                                      BEQ     50126$
1930  011356                                  IF #MOD.CO NOTSETIN CMDWRD THEN  ;IF A FORWARD CMD:
1931  011356  032737  000400  003346                                                      BIT     #MOD.CO,CMDWRD
1932  011364  001007                                                                      BNE     50127$
1933  011366                                    IF #MOD.CO NOTSETIN PCMDWD THEN  ;IF PREV CMD WAS A FWD ALSO:
1934  011366  032737  000400  003352                                                      BIT     #MOD.CO,PCMDWD
1935  011374  001002                                                                      BNE     50130$
1936  011376                                      LET RECCNT(R5) := RECCNT(R5) + #1  ;INCREMENT RECORD COUNT.
1937  011376  005265  003324                                                              INC     RECCNT(R5)
1938  011402                                    ENDIF
1939  011402                                                                      50130$:
1940  011402                                  ELSE                      ;IF REVERSE CMD:
1941  011402  000412                                                                      BR      50131$
1942  011404                                                                      50127$:
1943  011404                                    IF #MOD.CO SETIN PCMDWD THEN  ;IF PREVIOUS CMD WAS A REV ALSO:
1944  011404  032737  000400  003352                                                      BIT     #MOD.CO,PCMDWD
1945  011412  001406                                                                      BEQ     50132$
1946  011414                                      IF #X0.BOT NOTSETIN EOTFLG(R5) THEN ;WHEN NOT AT BOT THEN
1947  011414  032765  000002  003426                                                      BIT     #X0.BOT,EOTFLG(R
1948  011422  001002                                                                      BNE     50133$
1949  011424                                        LET RECCNT(R5) := RECCNT(R5) - #1  ;DECREMENT RECORD COUNT.
1950  011424  005365  003324                                                              DEC     RECCNT(R5)
1951  011430                                      ENDIF
1952  011430                                                                      50133$:
1953  011430                                    ENDIF
1954  011430                                                                      50132$:
1955  011430                                  ENDIF
1956  011430                                                                      50131$:
```

```
1957 011430                              ENDIF
1958 011430                                                              50126$:
1959 011430                            ENDIF
1960 011430                                                              50125$:
1961 011430                          ENDIF
1962 011430                                                              50123$:
1963 011430                          LET @DATAWT := RECCNT(R5)
1964 011430 016577 003324 171676                                                  MOV     RECCNT(R5),@DATA
1965 011436                        ENDIF
1966 011436                                                              50122$:
1967 011436 000207                 RTS     PC                  ;RETURN.
```

```
1968                                    ;       THIS IS THE ERROR CHECK SUBROUTINE.  AFTER INTERRUPT THIS
1969                                    ;       SUBROUTINE IS CALLED TO CHECK THE TS04 STATUS.
1970                                    ;       IF SPECIAL COND IS SET THEN THE TCC HANDLING SUBROUTINE IS ENTERED.
1971                                    ;       IF THE RFC IS NON ZERO FOR A COMMAND REQUIRING A BPCR,
1972                                    ;       THEN AN ERROR RFC IS REPORTED,
1973                                    ;       INPUTS:
1974                                    ;       OUTPUTS:
1975                                    ;       REGISTERS:      R2, R4.
1976                                    ;       CALLS:          TCC0-TCC7.
1977
1978  011440                            CHKERR:: IF #TS.SC SETIN TSSREG THEN            ;IF SPECIAL COND STATUS IS SET THEN:
1979  011440  032737  100000  003402                                                            BIT     #TS.SC,TSSREG
1980  011446  001441                                                                           BEQ     50134$
1981  011450                              IF CTCC NE #2 THEN              ;IF TCC IS NOT 2 THEN:
1982  011450  023727  003376  000002                                                            CMP     CTCC,#2
1983  011456  001405                                                                           BEQ     50135$
1984  011460                                IFB ERRREC EQ #0 THEN           ;IF NOT IN ERROR RECOVERY:
1985  011460  105737  003415                                                                   TSTB    ERRREC
1986  011464  001002                                                                           BNE     50136$
1987  011466  005265  003264                 INC       SCCNT(R5)             ;INC SC COUNTER.
1988  011472                                ENDIF
1989  011472                                                                  50136$:
1990  011472                              ENDIF
1991  011472                                                                  50135$:
1992  011472                              IF #TS.NXM SETIN TSSREG OR #TS.UPE SETIN TSSREG THEN  ;WHEN NON-EXISTANT MEMO
1993  011472  032737  004000  003402                                                            BIT     #TS.NXM,TSSREG
1994  011500  001004                                                                           BNE     50137$
1995  011502  032737  040000  003402                                                            BIT     #TS.UPE,TSSREG
1996  011510  001412                                                                           BEQ     50140$
1997  011512                                                                  50137$:
1998  011512                                IF #X2.OPM NOTSETIN MSGPKT+MS.XS2 THEN                  ;AND TAPE NOT MOVED
1999  011512  032737  100000  002346                                                            BIT     #X2.OPM,MSGPKT+M
2000  011520  001003                                                                           BNE     50141$
2001  011522                                  LET R2 :- #5                  ;SET TCC5 INDEX
2002  011522  012702  000005                                                                   MOV     #5,R2
2003  011526                                ELSE
2004  011526  000402                                                                           BR      50142$
2005  011530                                                                  50141$:
2006  011530                                  LET R2 := #4                  ;TAPE MOVED, SET TCC4 INDEX
2007  011530  012702  000004                                                                   MOV     #4,R2
2008  011534                                ENDIF
2009  011534                                                                  50142$:
2010  011534                              ELSE
2011  011534  000402                                                                           BR      50143$
2012  011536                                                                  50140$:
2013  011536                                  LET R2 := CTCC                ;SET DETECTED TCC INDEX
2014  011536  013702  003376                                                                   MOV     CTCC,R2
2015  011542                              ENDIF
2016  011542                                                                  50143$:
2017  011542                              LET R2 := R2 SHIFT 1  ;CURRENT TCC X 2.
2018  011542  006302                                                                           ASL     R2
2019  011544  004772  011644             JSR   PC,@TCCRA(R2)           ;GO TO THE TCC HANDLING SUBROUTINE.
2020  011550                            ELSE
2021  011550  000426                                                                           BR      50144$
2022  011552                                                                  50134$:
2023  011552                              IF #BRF.C SETIN CMDWRD THEN   ;IF BRF IS USED IN THIS CMD THEN:
```

```
2024  011552  032737  004000  003346                                                          BIT     #BRF.C,CMDWRD
2025  011560  001422                                                                          BEQ     50145$
2026  011562                                  IF MSGPKT+MS.RFC NE #0 THEN   ;IF THERE IS AN RFC THEN:
2027  011562  005737  002340                                                                  TST     MSGPKT+MS.RFC
2028  011566  001417                                                                          BEQ     50146$
2029  011570                                    IFB RANDOM EQ #0 ORB VFYFLG NE #0 THEN
2030  011570  105737  003441                                                                  TSTB    RANDOM
2031  011574  001403                                                                          BEQ     50147$
2032  011576  105737  003442                                                                  TSTB    VFYFLG
2033  011602  001411                                                                          BEQ     50150$
2034  011604                                                                        50147$:
2035                                                                          ;IF NOT IN RANDOM OR IF CMD IS WTV:
2036  011604                                      IFB IRE EQ #0 THEN            ;IF RFC ERROR REPORTS ARE ALLOWED:
2037  011604  105737  003445                                                                  TSTB    IRE
2038  011610  001006                                                                          BNE     50151$
2039  011612                                        LET HRDCNT(R5) := HRDCNT(R5) + #1  ;UPDATE HARD ERROR COUNT
2040  011612  005265  003304                                                                  INC     HRDCNT(R5)
2041  011616                                        ERRHRD #13,RFCERM,STAERM      ;REPORT RFC ERROR
2042  011616  104456                                                                          TRAP    C$ERHRD
2043  011620  000015                                                                          .WORD   13
2044  011622  004254                                                                          .WORD   RFCERM
2045  011624  005372                                                                          .WORD   STAERM
2046  011626                                      ENDIF
2047  011626                                                                        50151$:
2048  011626                                    ENDIF
2049  011626                                                                        50150$:
2050  011626                                  ENDIF
2051  011626                                                                        50146$:
2052  011626                                  ENDIF
2053  011626                                                                        50145$:
2054  011626                                ENDIF
2055  011626                                                                        50144$:
2056  011626                                IFB RWERR NE #0 THEN          ;IF A READ/WRITE ERROR HAS OCCURRED THEN:
2057  011626  105737  003413                                                                  TSTB    RWERR
2058  011632  001403                                                                          BEQ     50152$
2059  011634                                  LET CMDPKT := CMDSAV          ;RESTORE CMD PACKET AFTER ERROR RECOV.
2060  011634  013737  003350  002310                                                          MOV     CMDSAV,CMDPKT
2061  011642                                ENDIF
2062  011642                                                                        50152$:
2063  011642  000207                       RTS     PC                    ;RETURN.
2064
2065                                  ;     ADDRESSES OF TCC HANDLING ROUTINES FOR TERMINATION CLASS CODES 0 - 7.
2066
2067  011644  011664                  TCCRA:  TCC0
2068  011646  011702                          TCC1
2069  011650  011720                          TCC2
2070  011652  012030                          TCC3
2071  011654  012046                          TCC4
2072  011656  012462                          TCC5
2073  011660  012560                          TCC6
2074  011662  012722                          TCC7
```

```
                                ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 0, UNDEFINED SPECIAL
                                ;       CONDITION ERROR.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:

011664                          TCC0::  LET HRDCNT(R5) := HRDCNT(R5) + #1  ;UPDATE HARD ERROR COUNT.
011664  005265  003304                                                          INC     HRDCNT(R5)
011670                                  ERRHRD #5,SCERM,STAERM          ;REPORT SPECIAL CONDITION ERROR.
011670  104456                                                                  TRAP    C$ERHRD
011672  000005                                                                  .WORD   5
011674  004230                                                                  .WORD   SCERM
011676  005372                                                                  .WORD   STAERM
011700  000207                          RTS PC                          ;RETURN.


                                ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 1, ATTENTION CONDITION.
                                ;       THIS TCC INDICATES THAT THE DRIVE HAS UNDERGONE A STATUS CHANGE
                                ;       SUCH AS GOING OFFLINE OR COMING ONLINE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2,R4
                                ;       CALLS:          DROPU

011702                          TCC1::  ERRDF #6,ATTNM,STAERM           ;REPORT ATTENTION-UNIT OFF LINE.
011702  104455                                                                  TRAP    C$ERDF
011704  000006                                                                  .WORD   6
011706  004335                                                                  .WORD   ATTNM
011710  005372                                                                  .WORD   STAERM
011712  004737  015430                  JSR PC,DROPU                    ;DROP THE UNIT.
011716  000207                          RTS PC                          ;RETURN.
```

```
                                        ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 2, TAPE STATUS ALERT.
                                        ;       A STATUS CONDITION HAS BEEN ENCOUNTERED THAT MAY HAVE SIGNIFICANCE
                                        ;       TO THE PROGRAM.  BITS OF INTEREST INCLUDE TMK, RLS, LET, RLL, BOT, EOT.
                                        ;       INPUTS:
                                        ;       OUTPUTS:
                                        ;       REGISTERS:
                                        ;       CALLS:

011720                                  TCC2::    IF #X0.BOT SETIN MSGPKT+MS.XS0 ANDB EXPBOT NE #0 THEN
011720  032737  000002  002342                                                          BIT     #X0.BOT,MSGPKT+M
011726  001404                                                                        BEQ     50153$
011730  105737  003440                                                                TSTB    EXPBOT
011734  001401                                                                      BEQ     50153$
                                                                        ;IF AT BOT AND BOT IS EXPECTED:
011736  000433                                    BR TC2RTN                           ;RETURN-TCC2 CAUSED BY EXPECTED BOT.
011740                                          ENDIF
011740                                                                          50153$:
011740                                          IF #X0.RLS!X0.RLL!X0.TMK!X0.LET!X0.BOT SETIN MSGPKT+MS.XS0 THEN
011740  032737  170002  002342                                                          BIT     #X0.RLS!X0.RLL!X
011746  001427                                                                      BEQ     50154$
                                                                        ;IF TCC2 CAUSED BY ANYTHING BUT EOT:
011750                                            IFB RANDOM EQ #0 ORB VFYFLG NE #0 THEN
011750  105737  003441                                                              TSTB    RANDOM
011754  001403                                                                    BEQ     50155$
011756  105737  003442                                                              TSTB    VFYFLG
011762  001421                                                                    BEQ     50156$
011764                                                                          50155$:
                                                                        ;IF NOT IN RANDOM OR IF CMD IS WTV:
011764                                              IFB IRE EQ #0 THEN          ;IF RFC ERROR REPORTS ARE ALLOWED:
011764  105737  003445                                                              TSTB    IRE
011770  001016                                                                    BNE     50157$
011772                                                IFB ERRREC NE #0 THEN       ;IF WE ARE IN ERROR RECOVERY THEN:
011772  105737  003415                                                              TSTB    ERRREC
011776  001403                                                                    BEQ     50160$
012000                                                  LET UNREC :B= UNREC + #1   ;SET UNRECOVERABLE FLAG FOR LOG.
012000  105237  003414                                                              INCB    UNREC
012004                                                ELSE                        ;ELSE - IF NOT IN ERROR RECOVERY:
012004  000402                                                                    BR      50161$
012006                                                                          50160$:
012006                                                  LET SCCNT(R5) := SCCNT(R5) + #1 ;INCREMENT THE SPEC COND COUNTER.
012006  005265  003264                                                              INC     SCCNT(R5)
012012                                                ENDIF
012012                                                                          50161$:
012012                                                  LET HRDCNT(R5) := HRDCNT(R5) + #1   ;UPDATE HARD ERROR COUNT.
012012  005265  003304                                                              INC     HRDCNT(R5)
012016                                                  ERRHRD #7,TSAM,STAERM       ;REPORT TAPE STATUS ALERT.
012016  104456                                                                      TRAP    C$ERHRD
012020  000007                                                                      .WORD   7
012022  004436                                                                      .WORD   TSAM
012024  005372                                                                      .WORD   STAERM
012026                                            ENDIF
012026                                                                          50157$:
012026                                          ENDIF
012026                                                                          50156$:
012026                                      ENDIF
012026                                                                          50154$:
```

```
012026  000207                  TC2RTN:  RTS PC                                    ;RETURN.




                                ;        SUBROUTINE TO HANDLE TERMINATION CLASS CODE 3, FUNCTION REJECT.
                                ;        THE SPECIFIED FUNCTION WAS NOT INITIATED.  BITS OF INTEREST ARE
                                ;        RMR, OFL, VCK, BOT, ILC, WLE, ILA, AND NBA.
                                ;        INPUTS:
                                ;        OUTPUTS:
                                ;        REGISTERS:      R2,R4
                                ;        CALLS:          DROPU

012030                          TCC3::   ERRDF #8,FUNRM,STAERM            ;REPORT FUNCTION REJECT.
012030  104455                                                                           TRAP    C$ERDF
012032  000010                                                                           .WORD   8
012034  004353                                                                           .WORD   FUNRM
012036  005372                                                                           .WORD   STAERM
012040  004737  015430                   JSR PC,DROPU                     ;DROP THE UNIT.
012044  000207                           RTS PC                           ;RETURN.
```

```
2187                                    ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 4, RECOVERABLE ERROR.
2188                                    ;       TAPE POSITION IS ONE RECORD BEYOND WHAT ITS POSITION WAS WHEN
2189                                    ;       THE FUNCTION WAS INITIATED.  RECOVERY PROCEDURE IS TO LOG THE
2190                                    ;       ERROR AND ISSUE THE APPROPRIATE RETRY COMMAND.
2191                                    ;       2 WRITE-ERROR-RECOVERY ALGORITHMS CAN BE SELECTED:
2192                                    ;       THE FIRST ONE, VIA BADTSW SWITCH,  DOES DETECT BAD SPOTS ON TAPE.
2193                                    ;       IT CALLS A WRITE RETRY SUBR UNTIL THE RECORD IS RECOVERED
2194                                    ;       OR 20 BAD SPOTS HAVE BEEN LOGGED. ON REACHING 20 BAD
2195                                    ;       SPOTS LOGGED, A BAD TAPE OVERFLOW MSG IS PRINTED AND THE
2196                                    ;       UNIT DROPPED.
2197                                    ;       THE SECOND ALGORITHM ISSUES THE TS11 WRITE RETRY COMMAND
2198                                    ;       UP TO 16 TIMES BEFORE DROPPING THE UNIT OR PROCEEDING
2199                                    ;       WITH THE NEXT RECORD ON RECOVERY.
2200                                    ;       INPUTS:
2201                                    ;       OUTPUTS:
2202                                    ;       REGISTERS:      R2,R4.
2203                                    ;       CALLS:          RTLE, EXCUTE, GOWAIT, DROPU, WRTY
2204
2205  012046                            TCC4::  IF CMDLG EQ #2 ANDB BADTSW NE #0 THEN
2206  012046  023727  003354  000002                                                          CMP     CMDLG,#2
2207  012054  001125                                                                         BNE     50162$
2208  012056  105737  002206                                                                TSTB    BADTSW
2209  012062  001522                                                                         BEQ     50162$
2210  012064                              IFB ERRREC EQ #0 ANDB ERCVER NE #0 THEN
2211  012064  105737  003415                                                                TSTB    ERRREC
2212  012070  001007                                                                         BNE     50163$
2213  012072  105737  002205                                                                TSTB    ERCVER
2214  012076  001404                                                                         BEQ     50163$
2215  012100                                ERRSOFT #9,RERM,STAERM      ;
2216  012100  104457                                                                        TRAP    C$ERSOFT
2217  012102  000011                                                                        .WORD   9
2218  012104  004550                                                                        .WORD   RERM
2219  012106  005372                                                                        .WORD   STAERM
2220  012110                          ENDIF
2221  012110                                                                      50163$:
2222  012110                          IFB IREC EQ #0 THEN         ;
2223  012110  105737  002211                                                                TSTB    IREC
2224  012114  001102                                                                         BNE     50164$
2225  012116                            LET ERRREC :B- ERRREC + #1  ;RETRY FLAG FOR EXCUTE SUBR: DON'T UPDATE REC CN
2226  012116  105237  003415                                                                INCB    ERRREC
2227  012122                            LET WRTYER :B  WRTYER + #1  ;REWRITE ERROR FLAG FOR WRTY SUBR
2228  012122  105237  003410                                                                INCB    WRTYER
2229  012126                            IFB WRTYFG EQ #0 THEN       ;FIRST RETRY ON THIS RECORD: SUBSEQUENT
2230  012126  105737  003407                                                                TSTB    WRTYFG
2231  012132  001072                                                                         BNE     50165$
2232                                                                ;RETRIES WITH TCC4 + WORS BY-PASS THIS SECTION
2233  012134                              LET WTYWRD := CMDWRD      ;SAVE WRITE COMMAND PACKET
2234  012134  013737  003346  013336                                                        MOV     CMDWRD,WTYWRD
2235  012142                            LET WTYCMD :- CMDPKT        ;
2236  012142  013737  002310  013334                                                        MOV     CMDPKT,WTYCMD
2237  012150                            LET WTYBRF :  CMDPKT+CP.CNT       ;
2238  012150  013737  002316  013340                                                        MOV     CMDPKT+CP.CNT,WT
2239  012156                            LET RWERR :B= RWERR + #1   ;LOG SUBR FLAG: COUNT WRT ERRORS
2240  012156  105237  003413                                                                INCB    RWERR
2241  012162                            LET WRTYFG :B- WRTYFG + #1         ;RETRY IN PROGRESS FLAG
2242  012162  105237  003407                                                                INCB    WRTYFG
```

```
2243  012166                                    REPEAT
2244  012166                                                                  50166$:
2245  012166                                      LET WRTYCT(R5) := WRTYCT(R5) + #1        ;COUNT GLOBAL WRITE RETRIES
2246  012166  005265  003244                                                       INC     WRTYCT(R5)
2247  012172                                      LET RETRYC := #0                 ;CLEAR # OF RETRIES PER RECORD
2248  012172  005037  003404                                                       CLR     RETRYC
2249  012176                                      LET RPTCNT :B= #0                ;CLEAR # OF REPEATS
2250  012176  105037  003406                                                       CLRB    RPTCNT
2251  012202  004737  013066                      JSR PC,WRTY              ;CALL WRITE RETRY
2252  012206                                    UNTILB WRTYER EQ #0 OR @BTPT HIS #40.     ;REPEAT RETRIES ON SAME RECORD
2253  012206  105737  003410                                                       TSTB    WRTYER
2254  012212  001404                                                             BEQ     50167$
2255  012214  027727  171216  000050                                             CMP     @BTPT,#40.
2256  012222  103761                                                             BLO     50166$
2257  012224                                                                  50167$:
2258                                            ;UNTIL RECOVERED OR 20 BAD SPOTS
2259  012224                                    IF @BTPT HIS #40. THEN   ;WHEN 20 BAD SPOTS LOGGED
2260  012224  027727  171206  000050                                             CMP     @BTPT,#40.
2261  012232  103423                                                             BLO     50170$
2262  012234                                      PRINTB #BTMSG2           ;PRINT BAD TAPE OVERFLOW MSG
2263  012234  012746  013427                                                       MOV     #BTMSG2,-(SP)
2264  012240  012746  000001                                                       MOV     #1,-(SP)
2265  012244  010600                                                             MOV     SP,R0
2266  012246  104414                                                             TRAP    C$PNTB
2267  012250  062706  000004                                                       ADD     #4,SP
2268  012254  004737  013546                      JSR PC,BORERS            ;ERASE BAD RECORD
2269  012260                                      LET RECCNT(R5) := RECCNT(R5) - #1       ;
2270  012260  005365  003324                                                       DEC     RECCNT(R5)
2271  012264  004737  015430                      JSR PC,DROPU             ;DROP UNIT
2272  012270                                      LET RECCNT(R5) :   #0    ;
2273  012270  005065  003324                                                       CLR     RECCNT(R5)
2274  012274                                      LET @TSDB(R5) :- #RWCPK ;REWIND UNIT
2275  012274  012775  002330  002452                                             MOV     #RWCPK,@TSDB(R5)
2276  012302                                        ENDIF
2277  012302                                                                  50170$:
2278  012302                                        LET WRTYFG :B= #0          ;RETRY COMPLETE FLAG
2279  012302  105037  003407                                                       CLRB    WRTYFG
2280  012306                                        LET MISCFG :B= MISCFG + #1         ;DO NOT HALT ON THIS CMD FLG
2281  012306  105237  003455                                                       INCB    MISCFG
2282  012312                                        LET PCMDWD :- WTYWRD       ;RESTORE ORIGINAL WRT CMD AFTER RECOVERY
2283  012312  013737  013336  003352                                             MOV     WTYWRD,PCMDWD
2284  012320                                      ENDIF
2285  012320                                                                  50165$:
2286  012320                                    ELSE
2287  012320  000402                                                             BR      50171$
2288  012322                                                                  50164$:
2289  012322                                      LET UNREC :B  UNREC + #1    ;
2290  012322  105237  003414                                                       INCB    UNREC
2291  012326                                    ENDIF
2292  012326                                                                  50171$:
2293  012326                                  ELSE
2294  012326  000454                                                             BR      50172$
2295  012330                                                                  50162$:
2296  012330  004737  012740                      JSR PC,RTLE              ;CHECK FOR RETRY LIMIT EXCEEDED.
2297  012334                                      IF CMDLG GT #2 THEN      ;IF READ CMD THEN:
2298  012334  023727  003354  000002                                             CMP     CMDLG,#2
```

```
012342  003411                                                                         BLE     50173$
012344                              LET R2 := #RRECL SHIFT -1    ;R2=READ RETRY COUNT LIMIT / 2
012344  012702  000024                                                                 MOV     #RRECL,R2
012350  006202                                                                         ASR     R2
012352                              IF RETRYC GE R2 THEN                 ;IF RETRY COUNT IS MORE THAN HALF LIMIT:
012352  023702  003404                                                                 CMP     RETRYC,R2
012356  002403                                                                         BLT     50174$
012360                                LET CMDPKT := CMDPKT SET.BY #OPP.C        ;SET OPPOSITE BIT FOR RETRY2.
012360  052737  020000  002310                                                         BIS     #OPP.C,CMDPKT
012366                              ENDIF
012366                                                                     50174$:
012366                            ENDIF
012366                                                                     50173$:
012366                            IF RETRYC EQ #0 ANDB ERCVER NE #0 THEN  ;IF THIS IS THE ORIGINAL ERROR THEN:
012366  005737  003404                                                                 TST     RETRYC
012372  001007                                                                         BNE     50175$
012374  105737  002205                                                                 TSTB    ERCVER
012400  001404                                                                         BEQ     50175$
012402                              ERRSOFT #9,RERM,STAERM      ;REPORT RECOVERABLE ERROR
012402  104457                                                                         TRAP    C$ERSOFT
012404  000011                                                                         .WORD   9
012406  004550                                                                         .WORD   RERM
012410  005372                                                                         .WORD   STAERM
012412                            ENDIF                         ;PROVIDED OPERATOR HAS ENABLED THE REPORT
012412                                                                     50175$:
012412                            LET RETRYC := RETRYC + #1     ;UPDATE RETRY COUNT.
012412  005237  003404                                                                 INC     RETRYC
012416                            LET CMDPKT := CMDPKT SET.BY #MOD.C1   ;SET RETRY BIT IN CMD PACKET.
012416  052737  001000  002310                                                         BIS     #MOD.C1,CMDPKT
012424                            IFB IREC EQ #0 THEN           ;IF ERROR RECOVERY ENABLED:
012424  105737  002211                                                                 TSTB    IREC
012430  001011                                                                         BNE     50176$
012432                              LET ERRREC :B= ERRREC + #1  ;SET ERROR RECOVERY FLAG.
012432  105237  003415                                                                 INCB    ERRREC
012436                              POP R2,R2                   ;POP 2 RTN ADRS FROM STACK.
012436  012602                                                                         MOV     (SP)+,R2
012440  012602                                                                         MOV     (SP)+,R2
012442  004737  010326              JSR PC,EXCUTE                       ;GO EXECUTE THE RETRY COMMAND.
012446  000137  010636              JMP GOWAIT                  ;GO WAIT FOR INTERRUPT + CHECK STATUS.
012452                            ELSE                          ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
012452  000402                                                                         BR      50177$
012454                                                                     50176$:
012454                              LET UNREC :B= UNREC + #1    ;SET UNRECOVERABLE ERROR FLAG.
012454  105237  003414                                                                 INCB    UNREC
012460                            ENDIF
012460                                                                     50177$:
012460                          ENDIF
012460                                                                     50172$:
012460  000207                  RTS PC                          ;RETURN
```

```
                        ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 5, RECOVERABLE ERROR.
                        ;       TAPE POSITION HAS NOT CHANGED.  RECOVERY PROCEDURE IS TO LOG THE
                        ;       ERROR AND RE-ISSUE THE ORIGINAL COMMAND.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:      R2,R4.
                        ;       CALLS:          RTLE, EXCUTE, GOWAIT, DROPU.

012462  004737  012740  TCC5::  JSR PC,RTLE                      ;CHECK FOR RETRY LIMIT EXCEEDED
012466                  IF RETRYC EQ #0 THEN                    ;IF THIS IS THE ORIGINAL ERROR THEN:
012466  005737  003404                                                  TST     RETRYC
012472  001004                                                          BNE     50200$
012474
012474  104457            ERRSOFT #10,RERM,STAERM               ;REPORT RECOVERABLE ERROR.
012476  000012                                                          TRAP    C$ERSOFT
012500  004550                                                          .WORD   10
012502  005372                                                          .WORD   RERM
                                                                        .WORD   STAERM
012504                  ENDIF
012504                                                                50200$:
012504                  LET RETRYC := RETRYC + #1                ;UPDATE RETRY COUNTER.
012504  005237  003404                                                  INC     RETRYC
012510                  IFB IREC EQ #0 THEN                      ;IF ERROR RECOVERY IS ENABLED:
012510  105737  002211                                                  TSTB    IREC
012514  001016                                                          BNE     50201$
012516
012516  105237  003415        LET ERRREC :B= ERRREC + #1           ;SET ERROR RECOVERY FLAG.
012522                                                                  INCB    ERRREC
012522  005265  003324        LET RECCNT(R5) := RECCNT(R5) + #1        ;UPDATE REC COUNT
                                                                        INC     RECCNT(R5)
012526                      LET @DATAWT := RECCNT(R5)                ;AND INSERT IT INTO WRT BFR
012526  016577  003324  170600                                        MOV     RECCNT(R5),@DATA
012534                      POP R2,R2                           ;POP 2 RTN ADRS FROM STACK.
012534  012602                                                          MOV     (SP)+,R2
012536  012602                                                          MOV     (SP)+,R2
012540  004737  010326        JSR PC,EXCUTE                       ;GO RE-ISSUE THE COMMAND.
012544  000137  010636        JMP GOWAIT                          ;GO WAIT FOR INTERRUPT + CHECK STATUS.
012550                  ELSE                                    ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
012550  000402                                                          BR      50202$
012552                                                                50201$:
012552                      LET UNREC :B= UNREC + #1             ;SET UNRECOVERABLE ERROR FLAG.
012552  105237  003414                                                  INCB    UNREC
012556                  ENDIF
012556                                                                50202$:
012556  000207          RTS PC                                  ;RETURN.
```

```
                        ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 6, UNRECOVERABLE ERROR.
                        ;       TAPE POSITION HAS BEEN LOST.  THE ONLY VALID RECOVERY PROCEDURE
                        ;       IS TO REWIND AND START OVER AT BOT UNLESS THE TAPE HAS LABELS OR
                        ;       SEQUENCE NUMBERS.  THIS DIAGNOSTIC WILL REWIND AND RETRY THE
                        ;       COMMAND ONLY IF DENSITY CHECK IS SET, OTHERWISE THE UNIT WILL BE
                        ;       DROPPED FROM THE TEST SEQUENCE.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:      R2, R4
                        ;       CALLS:          RTLE, WSSR, EXCUTE, GOWAIT, DROPU

012560                  TCC6::  IF X3.DCK NOTSETIN MSGPKT+MS.XS3 THEN
012560  033737  000010  002350                                          BIT     X3.DCK,MSGPKT+MS
012566  001016                                                          BNE     50203$
                                                              ;IF THERE IS NO DENSITY CHECK THEN:
012570                            IF CMDLG NE #0 THEN           ;IF CMD IS A READ OR WRITE THEN:
012570  005737  003354                                                  TST     CMDLG
012574  001404                                                          BEQ     50204$
012576
012576  105237  003413                    LET RWERR :B= RWERR + #1       ;SET RD/WR ERROR FLAG.
012602                                                                  INCB    RWERR
012602  105237  003414                    LET UNREC :B- UNREC + #1       ;SET UNRECOVERABLE ERROR FLAG.
012606                                                                  INCB    UNREC
012606                            ENDIF                         ;
012606                                                                50204$:
012606                            ERRDF #11,URERM,STAERM        ;REPORT UNRECOVERABLE ERROR.
012606  104455                                                          TRAP    C$ERDF
012610  000013                                                          .WORD   11
012612  004572                                                          .WORD   URERM
012614  005372                                                          .WORD   STAERM
012616  004737  015430                JSR PC,DROPU                  ;REPORT ERROR + DROP UNIT.
012622                            ELSE                          ;ELSE-IF THERE IS DENSITY CHECK:
012622  000436                                                          BR      50205$
012624                                                                50203$:
012624  004737  012740                JSR PC,RTLE                   ;CHECK FOR RETRY LIMIT EXCEEDED.
012630                            IF RETRYC EQ #0 THEN          ;IF THIS IS THE ORIGINAL ERROR THEN:
012630  005737  003404                                                  TST     RETRYC
012634  001004                                                          BNE     50206$
012636                                ERRSOFT #11,URERM,STAERM      ;REPORT DENSITY CHECK ERROR.
012636  104457                                                          TRAP    C$ERSOFT
012640  000013                                                          .WORD   11
012642  004572                                                          .WORD   URERM
012644  005372                                                          .WORD   STAERM
012646                            ENDIF
012646                                                                50206$:
012646                            LET RETRYC := RETRYC + #1      ;UPDATE RETRY COUNT.
012646  005237  003404                                                  INC     RETRYC
012652                            IFB IRE EQ #0 THEN            ;IF FRROR RECOVERY IS ENABLED THEN:
012652  105737  003445                                                  TSTB    IRE
012656  001016                                                          BNE     50207$
012660
012660  105237  003415                    LET ERRREC :B= ERRREC + #1      ;SET ERROR RECOVERY FLAG.
012664                                                                  INCB    ERRREC
012664  012775  002330  002452          LET @TSDB(R5) := #RWCPK        ;ISSUE A REWIND COMMAND.
012672  004737  011152                    JSR PC,WSSR                   ;WAIT FOR SUBSYSTEM READY.
012676                                    POP R2,R2                     ;POP 2 RTN ADR'S FROM STACK.
012676  012602                                                          MOV     (SP)+,R2
```

```
 2449  012700  012602                                                                     MOV     (SP)+,R2
 2450  012702  004737  010326              JSR PC,EXCUTE                      ;REISSUE THE COMMAND,
 2451  012706  000137  010636              JMP GOWAIT                         ;WAIT FOR INTERRUPT
 2452  012712                            ELSE                               ;ELSE-IF ERR REC DISABLED:
 2453  012712  000402                                                                     BR      50210$
 2454  012714                                                                    50207$:
 2455  012714                              LET UNREC :B- UNREC + #1           ;SET UNRECOVERABLE ERROR FLAG.
 2456  012714  105237  003414                                                             INCB    UNREC
 2457  012720                            ENDIF                              ;
 2458  012720                                                                    50210$:
 2459  012720                          ENDIF                              ;
 2460  012720                                                                    50205$:
 2461  012720  000207                  RTS PC                             ;RETURN
```

```
                                ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 7, FATAL SUBSYSTEM
                                ;       ERROR.  THE SUBSYSTEM IS INCAPABLE OF PROPERLY PERFORMING
                                ;       COMMANDS OR AT LEAST ITS INTEGRITY IS SERIOUSLY QUESTIONABLE.
                                ;       REFER TO THE FATAL CLASS CODE FIELD IN THE TSSR REGISTER FOR
                                ;       ADDITIONAL INFORMATION ON THE TYPE OF FATAL ERROR.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2, R4
                                ;       CALLS:

012722                          TCC7::  ERRDF #12,FATSM,STAERM          ;REPORT FATAL SUBSYSTEM ERROR.
012722  104455                                                                  TRAP    C$ERDF
012724  000014                                                                  .WORD   12
012726  004373                                                                  .WORD   FATSM
012730  005372                                                                  .WORD   STAERM
012732  004737  015430                  JSR PC,DROPU                    ;DROP THE UNIT.
012736  000207                          RTS PC                          ;RETURN.


                                ;       SUBROUTINE TO CHECK FOR RETRY LIMIT EXCEEDED.  PRINTS ERROR MESSAGE
                                ;       IF EXCEEDED AND DROP UNIT UNLESS COMMAND IS A READ.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2, R4.
                                ;       CALLS:          DROPU

012740                          RTLE::  IF CMDLG EQ #0 THEN             ;IF CMD IS NOT A READ OR WRITE THEN:
012740  005737  003354                                                          TST     CMDLG
012744  001010                                                                  BNE     50211$
012746
012746  104455                    ERRDF #11,URERM,STAERM          ;REPORT UNRECOVERABLE ERROR.
012750  000013                                                                  TRAP    C$ERDF
012752  004572                                                                  .WORD   11
012754  005372                                                                  .WORD   URERM
012756  004737  015430                                                          .WORD   STAERM
012762                          JSR PC,DROPU                    ;DROP THE UNIT.
012762  012602                  POP R2                                          MOV     (SP)+,R2
012764  000437                  BR RTLRTN                       ;AND RETURN.
012766                          ENDIF
012766                                                                  50211$:
012766                          LET RWERR :B= RWERR + #1        ;SET READ/WRITE ERROR FLAG.
012766  105237  003413                                                          INCB    RWERR
012772                          IF CMDLG EQ #2 THEN             ;IF CMD IS A WRT OR WTM:
012772  023727  003354  000002                                                  CMP     CMDLG,#2
013000  001016                                                                  BNE     50212$
013002                          IF RETRYC EQ #WRECL THEN        ;IF RETRY COUNT HAS REACHED LIMIT:
013002  023727  003404  000020                                                  CMP     RETRYC,#WRECL
013010  001011                                                                  BNE     50213$
013012                            LET UNREC :B- UNREC + #1      ;SET UNRECOVERABLE FLAG
013012  105237  003414                                                          INCB    UNREC
013016                            ERRDF #14,RLEXM,STAERM        ;REPORT RETRY LIMIT EXCEEDED.
013016  104455                                                                  TRAP    C$ERDF
013020  000016                                                                  .WORD   14
013022  004310                                                                  .WORD   RLEXM
013024  005372                                                                  .WORD   STAERM
```

```
2518  013026  004737  015430                    JSR PC,DROPU                ;DROP THE UNIT.
2519  013032                                    POP R2
2520  013032  012602                                                                MOV     (SP)+,R2
2521  013034                                  ENDIF
2522  013034                                                                  50213$:
2523  013034                                ELSE                            ;ELSE - CMD IS A READ:
2524  013034  000413                                                                BR      50214$
2525  013036                                                                  50212$:
2526  013036                                  IF RETRYC EQ #RRECL THEN      ;IF RETRY COUNT HAS REACHED LIMIT:
2527  013036  023727  003404  000020                                                CMP     RETRYC,#RRECL
2528  013044  001007                                                                BNE     50215$
2529  013046                                    LET UNREC :B= UNREC + #1    ;SET UNRECOVERABLE FLAG
2530  013046  105237  003414                                                      INCB    UNREC
2531  013052                                    ERRHRD #14,RLEXM,STAERM     ;REPORT RECOVERABLE ERROR.
2532  013052  104456                                                                TRAP    C$ERHRD
2533  013054  000016                                                                .WORD   14
2534  013056  004310                                                                .WORD   RLEXM
2535  013060  005372                                                                .WORD   STAERM
2536  013062                                    POP R2
2537  013062  012602                                                                MOV     (SP)+,R2
2538  013064                                  ENDIF
2539  013064                                                                  50215$:
2540  013064                                ENDIF
2541  013064                                                                  50214$:
2542  013064  000207                  RTLRTN: RTS PC                          ;RETURN
```

```
2543                                    ;       SUBR TO REWRITE A BAD, BUT RECOVERABLE WRITTEN RECORD.
2544                                    ;       REWRITE RECORD ON SAME SPOT: REPEAT 4 TIMES.
2545                                    ;       IF ALL 4 REPEATS GOOD, RECORD IS RECOVERED
2546                                    ;       AND A RECOVERABLE WRITE ERROR IS LOGGED.
2547                                    ;       IF ANY OF 4 REPEATS BAD, ERASE BAD RECORD, LOG SUSPECTED
2548                                    ;       BAD SPOT, RETRY AGAIN. RETRY 4 TIMES, UP TO 4 REPEATS EACH.
2549                                    ;       IF RECORD NOT GOOD AFTER 4 RETRIES, ERASE IT, EXIT WITH
2550                                    ;       ERROR FLAG WRTYER SET, PRINTING RETRY FAILED.
2551                                    ;       THIS ALL SCHEME IS REENTERED 20 T.MES MAX, IE 20 BAD
2552                                    ;       SPOTS MAX ARE ALLOWED.
2553
2554                                    ;       INPUTS:
2555                                    ;       OUTPUTS:
2556                                    ;       REGISTERS:      R3,R4
2557                                    ;       CALLS:          BORERS, REWRT
2558
2559  013066                            WRTY::  BEGIN RETRY
2560  013066                                      REPEAT
2561  013066                                                                                  50217$:
2562  013066                                          BEGIN REPEAT
2563  013066                                            REPEAT
2564  013066                                                                                  50221$:
2565  013066  004737  013546                           JSR PC,BORERS                 ;BACKSPACE/ERASE ONE RECORD
2566  013072                                            LET WRTYER :B= #0             ;CLEAR WRITE RETRY ERROR
2567  013072  105037  003410                                                                  CLRB    WRTYER
2568  013076  004737  013722                           JSR PC,REWRT                  ;REWRITE RECORD ON SAME SPOT
2569  013102                                            LET RPTCNT :B= RPTCNT + #1     ;COUNT REPEATS
2570  013102  105237  003406                                                                  INCB    RPTCNT
2571  013106                                          UNTILB RPTCNT EQ #4 ORB WRTYER NE #0      ;LIMIT: 4 REPEATS OR RECOVERED
2572  013106  123727  003406  000004                                                          CMPB    RPTCNT,#4
2573  013114  001403                                                                         BEQ     50222$
2574  013116  105737  003410                                                                 TSTB    WRTYER
2575  013122  001761                                                                         BEQ     50221$
2576  013124                                                                                  50222$:
2577  013124                                          END REPEAT                    ;
2578  013124                                                                                  50220$:
2579  013124                                          LET RETRYC := RETRYC + #1      ;COUNT RETRIES
2580  013124  005237  003404                                                                  INC     RETRYC
2581  013130                                          IFB WRTYER EQ #0 THEN         ;
2582  013130  105737  003410                                                                 TSTB    WRTYER
2583  013134  001001                                                                         BNE     50223$
2584  013136                                            LEAVE RETRY                 ;EXIT RETRY LOOP IF RECOVERED
2585  013136  000457                                                                         BR      50216$
2586  013140                                          ELSE                          ;
2587  013140                                                                                  50223$:
2588  013140                                            IFB ERCVER NE #0 THEN       ;
2589  013140  105737  002205                                                                 TSTB    ERCVER
2590  013144  001415                                                                         BEQ     50225$
2591  013146                                              PRINTB #BTMSG1,RETRYC,<B,RPTCNT>      ;PRINT SUSPECTED BAD SPOT
2592  013146  005046                                                                         CLR     -(SP)
2593  013150  153716  003406                                                                 BISB    RPTCNT,(SP)
2594  013154  013746  003404                                                                 MOV     RETRYC,-(SP)
2595  013160  012746  013342                                                                 MOV     #BTMSG1,-(SP)
2596  013164  012746  000003                                                                 MOV     #3,-(SP)
2597  013170  010600                                                                         MOV     SP,R0
2598  013172  104414                                                                         TRAP    C$PNTB
```

```
013174  062706  000010                                                             ADD     #10,SP
013200                          ENDIF                              ;
013200                                                                       50225$:
013200                          IF RETRYC EQ #1 THEN         ;ON FIRST RETRY, LOGG BAD SPOT
013200  023727  003404  000001                                                     CMP     RETRYC,#1
013206  001021                                                                     BNE     50226$
013210                            LET BTPT :- BTADDR(R5)           ;BTPT IS BOTH THE BAD SPOT COUNTER
013210  016537  002544  003436                                                     MOV     BTADDR(R5),BTPT
013216                            LET R4 := @BTPT + #2             ;AND THE LOGGING INDEX
013216  017704  170214                                                             MOV     @BTPT,R4
013222  062704  000002                                                             ADD     #2,R4
013226                            LET @BTPT := R4                  ;
013226  010477  170204                                                             MOV     R4,@BTPT
013232                            IF R4 LOS #40. THEN              ;
013232  020427  000050                                                             CMP     R4,#40.
013236  101005                                                                     BHI     50227$
013240                              LET R3 := BTPT                 ;STORE FIRST 20 BAD SPOTS
013240  013703  003436                                                             MOV     BTPT,R3
013244                              LET R4 := R4 + R3              ;
013244  060304                                                                     ADD     R3,R4
013246                              LET (R4) := RECCNT(R5)         ;
013246  016514  003324                                                             MOV     RECCNT(R5),(R4)
013252                            ENDIF
013252                                                                       50227$:
013252                          ENDIF                              ;
013252                                                                       50226$:
013252                          LET ERSFLG :B= ERSFLG + #1         ;ERASE FLAG TO ERASE BAD RECORD
013252  105237  003451                                                             INCB    ERSFLG
013256                          LET RWERR :B= #0                   ;CANCELL 'LOG'' ERROR FLAG ON FAILING RET
013256  105037  003413                                                             CLRB    RWERR
013262                          LET RPTCNT :B= #0                  ;CLEAR REPEAT COUNT FOR NEXT RETRY
013262  105037  003406                                                             CLRB    RPTCNT
013266                        ENDIF                                ;
013266                                                                       50224$:
013266                      UNTIL RETRYC EQ #4                     ;LIMIT: 4 RETRIES
013266  023727  003404  000004                                                     CMP     RETRYC,#4
013274  001274                                                                     BNE     50217$
013276                    END RETRY                                ;
013276                                                                       50216$:
013276                    IFB WRTYER NE #0 THEN                    ;
013276  105737  003410                                                             TSTB    WRTYER
013302  001413                                                                     BEQ     50230$
013304                      IFB ERCVER NE #0 THEN                  ;
013304  105737  002205                                                             TSTB    ERCVER
013310  001410                                                                     BEQ     50231$
013312                        PRINTB #BTMSG3                       ;PRINT RETRY FAILED
013312  012746  013477                                                             MOV     #BTMSG3,-(SP)
013316  012746  000001                                                             MOV     #1,-(SP)
013322  010600                                                                     MOV     SP,R0
013324  104414                                                                     TRAP    C$PNTB
013326  062706  000004                                                             ADD     #4,SP
013332                      ENDIF
013332                                                                       50231$:
013332                    ENDIF
013332                                                                       50230$:
013332  000207                    RTS PC
```

```
2655
2656
2657
2658
2659
2660
2661  013334  000000                    WTYCMD: .WORD   0               ;STORAGE FOR WRITE CMD WHILE RETRYING
2662  013336  000000                    WTYWRD: .WORD   0               ;STORAGE FOR WRITE CMD WORD WHILE RETRYING
2663  013340  000000                    WTYBRF: .WORD   0               ;STORAGE FOR WRITE BPCR WHILE RETRYING
2664
2665
2666  013342  040445  052523  050123  BTMSG1: .ASCIZ  /%ASUSPECT BAD SPOT AFTER %D1%A RETRY, %D1%A REPEAT%N/
2667  013350  041505  020124  040502
2668  013356  020104  050123  052117
2669  013364  040440  052106  051105
2670  013372  022440  030504  040445
2671  013400  051040  052105  054522
2672  013406  020054  042045  022461
2673  013414  020101  042522  042520
2674  013422  052101  047045     000
2675  013427     045  022516  041101  BTMSG2: .ASCIZ  /%N%ABAD TAPE OVERFLOW: CHANGE TAPE!%N%N/
2676  013434  042101  052040  050101
2677  013442  020105  053117  051105
2678  013450  046106  053517  020072
2679  013456  044103  047101  042507
2680  013464  052040  050101  020505
2681  013472  047045  047045     000
2682  013477     045  051101  052105  BTMSG3: .ASCIZ  /%ARETRY FAILED ON BAD SPOT...ERASED!%N/
2683  013504  054522  043040  044501
2684  013512  042514  020104  047117
2685  013520  041040  042101  051440
2686  013526  047520  027124  027056
2687  013534  051105  051501  042105
2688  013542  022441  000116
2689                                  .EVEN
```

```
                        ;       SUBR TO BACSPACE ONE RECORD
                        ;       IF THE ERASE FLAG IS SET, THEN ERASE THAT RECORD
                        ;       INPUTS:         ERSFLG  1 = DO ERASE
                        ;       OUTPUTS:
                        ;       REGISTERS:
                        ;       CALLS:          EXCUTE, GOWAIT, CKHAE

013546                  BORERS:: LET PCMDWD := CMDWRD   ;SET COMMAND  TO SPACE REV
013546  013737  003346  003352                                                    MOV     CMDWRD,PCMDWD
013554                          LET CMDWRD := #SRR      ;
013554  012737  104410  003346                                                    MOV     #SRR,CMDWRD
013562                          LET CMDPKT := CMDWRD CLR.BY #BRF.C      ;
013562  013737  003346  002310                                                    MOV     CMDWRD,CMDPKT
013570  042737  004000  002310                                                    BIC     #BRF.C,CMDPKT
013576                          LET CMDSAV := CMDPKT    ;
013576  013737  002310  003350                                                    MOV     CMDPKT,CMDSAV
013604                          LET CMDPKT+CP.ADL := #1 ;
013604  012737  000001  002310                                                    MOV     #1,CMDPKT+CP.ADL
013612                          LET CMDLG := #0         ;
013612  005037  003354                                                            CLR     CMDLG
013616  004737  007344                  JSR PC,CMDAC            ;
013622  004737  010326                  JSR PC,EXCUTE           ;
013626  004737  010636                  JSR PC,GOWAIT           ;
013632  004737  015724                  JSR PC,CKHAE            ;
013636                          IFB ERSFLG NE #0 THEN   ;WHEN ERASE FLAG IS SET, DO ERASE
013636  105737  003451                                                            TSTB    ERSFLG
013642  001426                                                                    BEQ     50232$
013644                            LET PCMDWD :- CMDWRD  ;
013644  013737  003346  003352                                                    MOV     CMDWRD,PCMDWD
013652                            LET CMDWRD := #ERS    ;
013652  012737  100411  003346                                                    MOV     #ERS,CMDWRD
013660                            LET CMDPKT := CMDWRD  ;
013660  013737  003346  002310                                                    MOV     CMDWRD,CMDPKT
013666                            LET CMDSAV := CMDPKT  ;
013666  013737  002310  003350                                                    MOV     CMDPKT,CMDSAV
013674  004737  007344                    JSR PC,CMDAC            ;
013700  004737  010326                    JSR PC,EXCUTE
013704  004737  010636                    JSR PC,GOWAIT
013710  004737  015724                    JSR PC,CKHAE
013714                                    LET ERSFLG :B- #0
013714  105037  003451                                                            CLRB    ERSFLG
013720                          ENDIF
013720                                                                  50232$:
013720  000207                  RTS PC
                        ;       SUBR TO REWRITE A BADLY WRITTEN RECORD

013722                  REWRT:  LET PCMDWD :- CMDWRD    ;RESTORE WRITE COMMAND PACKET
013722  013737  003346  003352                                                    MOV     CMDWRD,PCMDWD
013730                          LET CMDWRD := WTYWRD    ;
013730  013737  013336  003346                                                    MOV     WTYWRD,CMDWRD
013736                          LET CMDPKT :  WTYCMD    ;
013736  013737  013334  002310                                                    MOV     WTYCMD,CMDPKT
013744                          LET CMDSAV :  CMDPKT    ;
013744  013737  002310  003350                                                    MOV     CMDPKT,CMDSAV
013752                          LET CMDPKT+CP.ADL :  DATAWT     ;
013752  013737  003334  002312                                                    MOV     DATAWT,CMDPKT+CP
```

```
2746  013760                                  LET CMDPKT+CP.CNT :- WTYBRF     ;
2747  013760  013737  013340  002316                                          MOV     WTYBRF,CMDPKT+CP
2748  013766                                  LET CMDLG :- #2                 ;
2749  013766  012737  000002  003354                                          MOV     #2,CMDLG
2750  013774  004737  007344                 JSR PC,CMDAC
2751  014000  004737  010326                 JSR PC,EXCUTE                   ;RE-WRITE RECORD
2752  014004  004737  010636                 JSR PC,GOWAIT                   ;
2753  014010  004737  015724                 JSR PC,CKHAE                    ;
2754  014014  000207                         RTS PC
```

```
2755                                    ;       SUBROUTINE TO LOG BYTES READ/WRITTEN.
2756                                    ;       ALSO UPDATES READ/WRITE ERROR COUNTERS.
2757                                    ;       INPUTS:
2758                                    ;       OUTPUTS:
2759                                    ;       REGISTERS:      R2, R3, R4.
2760                                    ;       CALLS:
2761
2762  014016                            LOG::   IFB ERLOG EQ #0 THEN            ;IF DATA AND ERRORS HAVE NOT BEEN LOGGED THEN:
2763  014016  105737  003412                                                                    TSTB    ERLOG
2764  014022  001126                                                                             BNE     50233$
2765  014024                                    LET ERLOG :B= ERLOG + #1       ;SET LOG DONE FLAG.
2766  014024  105237  003412                                                                    INCB    ERLOG
2767  014030                                    LET R4 := CMDLG                ;GET CURRENT CMD LOGGING CODE.
2768  014030  013704  003354                                                                    MOV     CMDLG,R4
2769  014034                                    IF R4 NE #0 THEN               ;IF THERE IS A CODE THEN:
2770  014034  005704                                                                            TST     R4
2771  014036  001520                                                                            BEQ     50234$
2772  014040                                      LET R4 := R4 - #2            ;ADJUST THE CODE FOR TABLE INDEX.
2773  014040  162704  000002                                                                   SUB     #2,R4
2774  014044                                      LET R2 := R5 + BINC(R4) + #CNTBGN  ;R2 = ADR OF BYTE COUNT LSW.
2775  014044  010502                                                                            MOV     R5,R2
2776  014046  066402  014302                                                                   ADD     BINC(R4),R2
2777  014052  062702  002554                                                                   ADD     #CNTBGN,R2
2778  014056                                      LET (R2) := (R2) + BRFCNT    ;ADD BRF TO LSW.
2779  014056  063712  003344                                                                   ADD     BRFCNT,(R2)
2780  014062                                      IF MSGPKT+MS.RFC LOS BRFCNT THEN  ;IF THE RFC IS LOWER OR THE SAME AS BRF THEN
2781  014062  023737  002340  003344                                                           CMP     MSGPKT+MS.RFC,BR
2782  014070  101002                                                                           BHI     50235$
2783  014072                                        LET (R2) := (R2) - MSGPKT+MS.RFC  ;SUBTRACT RFC FROM EXPECTED BRF.
2784  014072  163712  002340                                                                   SUB     MSGPKT+MS.RFC,(R
2785  014076                                      ENDIF
2786  014076                                                                          50235$:
2787  014076                                    LET R3 := R2 + #10             ;R3 = ADR OF 2ND WORD.
2788  014076  010203                                                                            MOV     R2,R3
2789  014100  062703  000010                                                                   ADD     #10,R3
2790  014104                                    WHILE (R2) GT #999. DO
2791  014104                                                                          50236$:
2792  014104  021227  001747                                                                   CMP     (R2),#999.
2793  014110  003404                                                                           BLE     50237$
2794  014112                                      LET (R2) := (R2) - #1000.    ;UPDATE BYTE COUNT
2795  014112  162712  001750                                                                   SUB     #1000.,(R2)
2796  014116                                      LET (R3) := (R3) + #1        ;2ND WORD.
2797  014116  005213                                                                           INC     (R3)
2798  014120                                    ENDDO
2799  014120  000771                                                                           BR      50236$
2800  014122                                                                          50237$:
2801  014122                                    LET R2 := R3 + #10             ;R2 = ADR OF 3RD WORD.
2802  014122  010302                                                                            MOV     R3,R2
2803  014124  062702  000010                                                                   ADD     #10,R2
2804  014130                                    WHILE (R3) GT #999. DO
2805  014130                                                                          50240$:
2806  014130  021327  001747                                                                   CMP     (R3),#999.
2807  014134  003404                                                                           BLE     50241$
2808  014136                                      LET (R3) := (R3) - #1000.    ;UPDATE BYTE COUNT
2809  014136  162713  001750                                                                   SUB     #1000.,(R3)
2810  014142                                      LET (R2) := (R2) + #1        ;3RD WORD.
```

```
014142  005212                                                             INC     (R2)
014144                          ENDDO
014144  000771                                                             BR      50240$
014146                                                          50241$:
014146                          LET R3 := R2 + #10          ;R3 - ADR OF 4TH WORD.
014146  010203                                                             MOV     R2,R3
014150  062703  000010                                                     ADD     #10,R3
014154                          WHILE (R2) GT #999. DO
014154                                                          50242$:
014154  021227  001747                                                     CMP     (R2),#999.
014160  003404                                                             BLE     50243$
014162                            LET (R2) := (R2) - #1000.   ;UPDATE BYTE COUNT
014162  162712  001750                                                     SUB     #1000.,(R2)
014166                            LET (R3) := (R3) + #1       ;4TH WORD.
014166  005213                                                             INC     (R3)
014170                          ENDDO
014170  000771                                                             BR      50242$
014172                                                          50243$:
014172                          IFB RWERR NE #0 THEN        ;IF R/W ERROR, UPDATE ERROR COUNT.
014172  105737  003413                                                     TSTB    RWERR
014176  001440                                                             BEQ     50244$
014200                            LET R2 := R5 + EINC(R4) + #WRREC  ;R2   ADR OF COUNTER.
014200  010502                                                             MOV     R5,R2
014202  066402  014310                                                     ADD     EINC(R4),R2
014206  062702  002714                                                     ADD     #WRREC,R2
014212                            IFB UNREC NE #0 THEN        ;IS THE ERROR UNRECOVERABLE?
014212  105737  003414                                                     TSTB    UNREC
014216  001404                                                             BEQ     50245$
014220                              LET R2 := R2 + #10        ;YES, POINT TO NEXT COUNTER.
014220  062702  000010                                                     ADD     #10,R2
014224                              LET (R2) := (R2) + #1     ;UPDATE THE ERROR COUNTER
014224  005212                                                             INC     (R2)
014226                            ELSE                        ;ELSE - IF ERROR IS RECOVERABLE:
014226  000424                                                             BR      50246$
014230                                                          50245$:
014230                              LET (R2) := (R2) + #1     ;UPDATE THE ERROR COUNTER
014230  005212                                                             INC     (R2)
014232                              IFB IREC EQ #0 THEN       ;IF ERROR RECOVERY IS ENABLED:
014232  105737  002211                                                     TSTB    IREC
014236  001020                                                             BNE     50247$
014240                                IFB DROPED EQ #0 ANDB ERCVER NE #0 THEN ;IF UNIT HAS NOT BEEN DROPPED:
014240  105737  003446                                                     TSTB    DROPED
014244  001015                                                             BNE     50250$
014246  105737  002205                                                     TSTB    ERCVER
014252  001412                                                             BEQ     50250$
014254                                  PRINTB #NURTY1,RETRYC    ;PRINT # OF RETRIES TO RECOVER
014254  013746  003404                                                     MOV     RETRYC,-(SP)
014260  012746  005073                                                     MOV     #NURTY1,-(SP)
014264  012746  000002                                                     MOV     #2,-(SP)
014270  010600                                                             MOV     SP,R0
014272  104414                                                             TRAP    C$PNTB
014274  062706  000006                                                     ADD     #6,SP
014300                                ENDIF                      ;PROVIDED PRINT HAS BEEN ENABLED
014300                                                          50250$:
014300                              ENDIF
014300                                                          50247$:
```

```
014300                                  ENDIF
014300                                                                          50246$:
014300                                ENDIF
014300                                                                          50244$:
014300                              ENDIF
014300                                                                          50234$:
014300                              ENDIF
014300                                                                          50233$:
014300  000207                      RTS PC
                            ;       INDEXES TO BYTE COUNTERS.
014302  000000              BINC:   0                               ;WRITE.
014304  000040                      40                              ;READ REV.
014306  000100                      100                             ;READ FWD.
                            ;       INDEXES TO READ/WRITE ERROR COUNTERS.
014310  000000              EINC:   0                               ;WRITE.
014312  000020                      20                              ;READ REV.
014314  000040                      40                              ;READ FWD.




                            ;       IF A WRITE/VERIFY COMMAND IS ISSUED, CONTROL IS THEN
                            ;       TRANSFERRED TO THIS SUBROUTINE TO READ REVERSE, CHECK DATA,
                            ;       READ FORWARD, CHECK DATA, THEN CONTINUE TO NEXT COMMAND.
                            ;       INPUTS:
                            ;       OUTPUTS:
                            ;       REGISTERS:
                            ;       CALLS:          VFEXC.

014316                      VFYDAT:: IFB VFYFLG NE #0 THEN          ;IF DATA IS TO BE VERIFIED:
014316  105737  003442                                                      TSTB    VFYFLG
014322  001426                                                              BEQ     50251$
014324                                LET PCMDWD := CMDWRD          ;SAVE THE PREVIOUS COMMAND WORD.
014324  013737  003346  003352                                              MOV     CMDWRD,PCMDWD
014332                                LET CMDWRD :- #RDR            ;COMMAND IS READ REV.
014332  012737  104401  003346                                              MOV     #RDR,CMDWRD
014340                                LET CMDLG := #4               ;SET UP CMD LOGGING INDEX.
014340  012737  000004  003354                                              MOV     #4,CMDLG
014346  004737  014402                JSR   PC,VFEXC                ;GO READ ALL THE RECORDS REV.
014352                                LET PCMDWD := CMDWRD          ;SAVE THE PREVIOUS COMMAND WORD.
014352  013737  003346  003352                                              MOV     CMDWRD,PCMDWD
014360                                LET CMDWRD ·  #RDF            ;COMMAND IS READ FWD.
014360  012737  104001  003346                                              MOV     #RDF,CMDWRD
014366                                LET CMDLG := #6               ;SET UP CMD LOGGING INDEX.
014366  012737  000006  003354                                              MOV     #6,CMDLG
014374  004737  014402                JSR   PC,VFEXC                ;GO READ ALL RECORDS FWD.
014400                              ENDIF
014400                                                                          50251$:
014400  000207                      RTS     PC      ;RETURN.
```

```
                                ;       SUBROUTINE TO EXECUTE THE READ AND VERIFY, FORWARD OR REVERSE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2
                                ;       CALLS:          CMDAC, FIRSTU, VFISU, NEXTU, CKHAE.

014402                          VFEXC:: LET CMDPKT :- CMDWRD CLR.BY #BRF.C  ;COMMAND PACKET - READ REV OR FWD.
014402  013737  003346  002310                                                    MOV     CMDWRD,CMDPKT
014410  042737  004000  002310                                                    BIC     #BRF.C,CMDPKT
014416                          IFB SWBFLG NE #0 THEN           ;IF BYTES ARE TO BE SWAPPED:
014416  105737  003444                                                            TSTB    SWBFLG
014422  001403                                                                  BEQ     50252$
014424                            LET CMDPKT :- CMDPKT SET.BY #SWB.C  ;SET SWAB BIT IN CMD PACKET.
 14424  052737  010000  002310                                                    BIS     #SWB.C,CMDPKT
014432                          ENDIF
014432                                                                  50252$:
014432                          LET CMDSAV :  CMDPKT            ;SAVE COMMAND PACKET 1ST WORD.
014432  013737  002310  003350                                                    MOV     CMDPKT,CMDSAV
014440  013737  003336  002312  MOV     DATARD,CMDPKT+CP.ADL    ;SAVE BUFFER START ADDRESS.
014446                          LET NCNT :- #0                  ;CLEAR NUMBER OF OPERATIONS.
014446  005037  003340                                                            CLR     NCNT
014452                          WHILE NCNT LT NCNT1 DO          ;WHILE THERE ARE RECORDS REMAINING:
014452                                                                  50253$:
014452  023737  003340  003342                                                    CMP     NCNT,NCNT1
014460  002062                                                                  BGE     50254$
014462  004737  007344            JSR PC,CMDAC                    ;STORE CMD ASCII IN ERROR MSG.
014466  004737  015332            JSR   PC,FIRSTU                 ;SET UP FOR FIRST UNIT.
014472                            WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE DEVICES REMAINING:
014472                                                                  50255$:
014472  026527  002532  177777                                                    CMP     DEVTBL(R5),#END
014500  001442                                                                  BEQ     50256$
014502                        IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
014502  032737  000400  003346                                                    BIT     #MOD.CO,CMDWRD
014510  001421                                                                  BEQ     50257$
014512                          IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
014512  032765  000002  003426                                                    BIT     #XO.BOT,EOTFLG(R
014520  001014                                                                  BNE     50260$
014522                            IF #XO.EOT SETIN EOTFLG(R5) THEN   ;BUT IF AT EOT
014522  032765  000001  003426                                                    BIT     #XO.EOT,EOTFLG(R
014530  001406                                                                  BEQ     502'1$
014532                              IFB ALLEOT NE #0 THEN           ;AND ALL OTHERS AT EO
014532  105737  003450                                                            TSTB    ALLEOT
014536  001402                                                                  BEQ     50262$
014540  004737  014630                JSR PC,VFISU                ;THEN READ VERIFY
014544                              ENDIF                         ;IF NOT ALL AT EOT, FREEZE UNIT(S)
014544                                                                  50262$:
014544                            ELSE                            ;IF NOT AT BOT AND
014544  000402                                                                  BR      50263$
014546                                                                  50261$:
014546  004737  014630                JSR PC,VFISU                ;NOT AT EOT, READ VFY
014552                              ENDIF
014552                                                                  50263$:
014552                          ENDIF
014552                                                                  50260$:
014552                        ELSE                        ;ELSE IF CMD IS NOT REVERSE:
014552  000412                                                                  BR      50264$
```

```
014554                                                                        50257$:
014554                                          IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
014554  032765  000001  003426                                                              BIT     #XO.EOT,EOTFLG(R
014562  001404                                                                              BEQ     50265$
014564  032737  000001  003346                                                              BIT     #CMD.CO,CMDWRD
014572  001002                                                                              BNE     50266$
014574                                                                        50265$:
                                                              ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
014574  004737  014630                     JSR PC,VFISU         ;ISSUE CMD, CHECK STATUS AND DATA.
014600                                    ENDIF
014600                                                                        50266$:
014600                                 ENDIF
014600                                                                        50264$:
014600  004737  015400                 JSR PC,NEXTU           ;GO FIND THE NEXT UNIT.
014604                              ENDDO
014604  000732                                                                              BR      50255$
014606                                                                        50256$:
014606  004737  015724              JSR    PC,CKHAE          ;CHECK FOR HALT AFTER EACH CMD.
014612                              LET NCNT := NCNT + #1    ;UPDATE THE RECORD COUNT.
014612  005237  003340                                                                      INC     NCNT
014616                              LET PCMDWD :  CMDWRD     ;SAVE PREVIOUS COMMAND WORD.
014616  013737  003346  003352                                                              MOV     CMDWRD,PCMDWD
014624                           ENDDO
014624  000712                                                                              BR      50253$
014626                                                                        50254$:
014626  000207                   RTS     PC                  ;RETURN.
```

```
                        ;       SUBROUTINE TO ISSUE COMMAND, AWAIT INTERRUPT,
                        ;       CHECK STATUS, CHECK DATA.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:      R2
                        ;       CALLS:          EXCUTE, GOWAIT, CKDATA.

014630                  VFISU::     LET R2 := DATARD + #8.      ;INIT READ BUFFER POINTER.
014630  013702  003336                                                    MOV     DATARD,R2
014634  062702  000010                                                    ADD     #8.,R2
014640                              WHILE R2 NE DATARD DO       ;UNTIL 8 BYTES HAVE BEEN SET,
014640                                                                  50267$:
014640  020237  003336                                                    CMP     R2,DATARD
014644  001403                                                          BEQ     50270$
014646                                LET -(R2) := #-1          ;INIT READ BUFFER.
014646  012742  177777                                                    MOV     #-1,-(R2)
014652                              ENDDO
014652  000772                                                          BR      50267$
014654                                                                  50270$:
014654  004737  010326                    JSR PC,EXCUTE               ;GO EXECUTE THE COMMAND.
014660                              IFB DROPED EQ #0 THEN       ;IF UNIT HAS NOT BEEN DROPPED THEN:
014660  105737  003446                                                    TSTB    DROPED
014664  001002                                                          BNE     50271$
014666  004737  010636                      JSR       PC,GOWAIT       ;GO WAIT FOR DONE BIT.
014672                              ENDIF
014672                                                                  50271$:
014672                              IFB DROPED EQ #0 THEN       ;IF UNIT HAS NOT BEEN DROPPED THEN:
014672  105737  003446                                                    TSTB    DROPED
014676  001006                                                          BNE     50272$
014700                                IF #X0.BOT NOTSETIN EOTFLG(R5) THEN     ;WHEN NOT REVERSED INTO BOT, THEN
014700  032765  000002  003426                                            BIT     #X0.BOT,EOTFLG(R
014706  001002                                                          BNE     50273$
014710  004737  014716                      JSR     PC,CKDATA       ;GO VERIFY DATA.
014714                                  ENDIF
014714                                                                  50273$:
014714                              ENDIF
014714                                                                  50272$:
014714  000207                      RTS PC
```

```
                        ;       SUBROUTINE TO COMPARE DATA BETWEEN READ AND WRITE BUFFERS
                        ;       AND PRINT ERROR MESSAGE ON MISCOMPARE.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:      R2, R3, R4.
                        ;       CALLS:          GCMDA

014716                  CKDATA:: LET R3 := BRFCNT - MSGPKT+MS.RFC ;COMPUTE REC LENGTH READ
014716  013703  003344                                                  MOV     BRFCNT,R3
014722  163703  002340                                                  SUB     MSGPKT+MS.RFC,R3
014726                          IF R3 EQ #0 THEN                ;WHEN NO DATA RECEIVED
014726  005703                                                          TST     R3
014730  001015                                                          BNE     50274$
014732                            ERRHRD 17,WTVERM,DTAERM               ;PRINT ERROR AND EXIT
014732  104456                                                          TRAP    C$ERHRD
014734  000021                                                          .WORD   17
014736  004164                                                          .WORD   WTVERM
014740  005224                                                          .WORD   DTAERM
014742                            PRINTB #DTAER4                ;COMPARE ROUTINE
014742  012746  005010                                                  MOV     #DTAER4,-(SP)
014746  012746  000001                                                  MOV     #1,-(SP)
014752  010600                                                          MOV     SP,R0
014754  104414                                                          TRAP    C$PNTB
014756  062706  000004                                                  ADD     #4,SP
014762                          ELSE
014762  000560                                                          BR      50275$
014764                                                          50274$:
014764                            IF R3 HI BRFCNT THEN          ;WHEN REC READ IS LONGER
014764  020337  003344                                                  CMP     R3,BRFCNT
014770  101417                                                          BLOS    50276$
014772                              ERRHRD 17,WTVERM,DTAERM             ;THAN EXPECTED, PRINT
014772  104456                                                          TRAP    C$ERHRD
014774  000021                                                          .WORD   17
014776  004164                                                          .WORD   WTVERM
015000  005224                                                          .WORD   DTAERM
015002                              PRINTB #DTAER5,CMDPKT+CP.CNT        ;AN ERROR MESSAGE
015002  013746  002316                                                  MOV     CMDPKT+CP.CNT,-(
015006  012746  005031                                                  MOV     #DTAER5,-(SP)
015012  012746  000002                                                  MOV     #2,-(SP)
015016  010600                                                          MOV     SP,R0
015020  104414                                                          TRAP    C$PNTB
015022  062706  000006                                                  ADD     #6,SP
015026                          ELSE                            ;AND EXIT ROUTINE
015026  000536                                                          BR      50277$
015030                                                          50276$:
015030                              LET CKDCNT := R3 - #1               ;SAVE VERIFICATION LENGTH - 1.
015030  010337  015326                                                  MOV     R3,CKDCNT
015034  005337  015326                                                  DEC     CKDCNT
015040  005037  015330                              CLR CKDFF           ;CLEAR # OF BYTES IN ERROR COUNTER.
015044  005002                                      CLR R2              ;INIT BYTE COUNTER
015046                                              LET R3 := DATAWT    ;GET WRITE BUFFER ADDRESS.
015046  013703  003334                                                  MOV     DATAWT,R3
015052                                              LET R4 := DATARD    ;GET READ BUFFER ADDRESS.
015052  013704  003336                                                  MOV     DATARD,R4
015056                                              IFB T1SWB NE #0 THEN ;WHEN RUNNING TEST1-SUB 12,
015056  105737  003447                                                  TSTB    T1SWB
```

```
3093 015062 001401                                                                 BEQ     50300$
3094 015064 000313                        SWAB (R3)             ;SWAP FIRST WORD OF WRT BFR
3095 015066                               ENDIF                 ;WHICH CONTAINS THE RECORD COUNT
3096 015066                                                                 50300$:
3097 015066                               REPEAT                ;REPEAT UNTIL ALL DATA IS COMPARED:
3098 015066                                                                 50301$:
3099 015066                                 IF R2 EQ CKDCNT THEN    ;IF THIS IS THE LAST BYTE THEN:
3100 015066 020237 015326                                                       CMP     R2,CKDCNT
3101 015072 001011                                                              BNE     50302$
3102 015074                                   IFB SWBFLG NE #0 THEN  ;IF BYTE SWAPPING IS ENABLED THEN:
3103 015074 105737 003444                                                       TSTB    SWBFLG
3104 015100 001406                                                              BEQ     50303$
3105 015102                                     IF #BIT00 NOTSETIN CKDCNT THEN      ;IF RECORD LENGTH IS ODD THEN:
3106 015102 032737 000001 015326                                                BIT     #BIT00,CKDCNT
3107 015110 001002                                                              BNE     50304$
3108 015112 105723                                 TSTB      (R3)+          ;LAST BYTE WILL BE IN
3109 015114 105724                                 TSTB      (R4)+          ;THE UPPER BYTE.
3110 015116                                     ENDIF
3111 015116                                                                 50304$:
3112 015116                                   ENDIF
3113 015116                                                                 50303$:
3114 015116                                 ENDIF
3115 015116                                                                 50302$:
3116 015116 121314                           CMPB (R3),(R4)        ;ARE THEY EQUAL.
3117 015120 001452                           BEQ       3$                ;BR IF SO.
3118 015122 005737 015330                    TST CKDFF             ;1 ST TIME THRU?
3119 015126 001010                           BNE       2$                ;BR IF NOT.
3120 015130 005265 003274                    INC       VFYCNT(R5)        ;INC THE VERIFY ERROR COUNTER.
3121 015134 005265 003304                    INC HRDCNT(R5)        ;INC THE HARD ERROR COUNT.
3122 015140                                  ERRHRD #17,WTVERM,DTAERM  ;REPORT WRITE/VERIFY ERROR.
3123 015140 104456                                                              TRAP    C$ERHRD
3124 015142 000021                                                              .WORD   17
3125 015144 004164                                                              .WORD   WTVERM
3126 015146 005224                                                              .WORD   DTAERM
3127 015150                          2$:     LET CKDFF := CKDFF + #1   ;INCREMENT # OF BYTES IN ERROR.
3128 015150 005237 015330                                                       INC     CKDFF
3129 015154 111437 003364                    MOVB (R4),TIME1       ;SAVE WAS DATA FOR TYPOUT.
3130 015160 042737 177400 003364             BIC       #177400,TIME1     ;CLEAR GARBAGE.
3131 015166 111337 003366                    MOVB (R3),TIME2       ;SAVE SHOULD BE DATA FOR TYPOUT.
3132 015172 042737 177400 003366             BIC       #177400,TIME2     ;CLEAR GARBAGE.
3133 015200                                  IF CKDFF LT #11. THEN    ;IF ERROR BYTE COUNT IS LESS THAN 11:
3134 015200 023727 015330 000013                                                CMP     CKDFF,#11.
3135 015206 002017                                                              BGE     50305$
3136 015210                                    PRINTX #DTAER2,R2,<B,TIME1>,<B,TIME2>  ;PRINT EXP + ACT DATA.
3137 015210 005046                                                              CLR     -(SP)
3138 015212 153716 003366                                                       BISB    TIME2,(SP)
3139 015216 005046                                                              CLR     -(SP)
3140 015220 153716 003364                                                       BISB    TIME1,(SP)
3141 015224 010246                                                              MOV     R2,-(SP)
3142 015226 012746 004677                                                       MOV     #DTAER2,-(SP)
3143 015232 012746 000004                                                       MOV     #4,-(SP)
3144 015236 010600                                                              MOV     SP,R0
3145 015240 104415                                                              TRAP    C$PNTX
3146 015242 062706 000012                                                       ADD     #12,SP
3147 015246                                  ENDIF
3148 015246                                                                 50305$:
```

```
015246  105723                  3$:       TSTB (R3)+                        ;UPDATE WRITE BUFFER ADDRESS.
015250  105724                            TSTB (R4)+                        ;UPDATE READ BUFFER ADDRESS.
015252  105722                            TSTB (R2)+                        ;UPDATE BYTE COUNTER.
015254                                  UNTIL R2 GT CKDCNT          ;END OF DATA COMPARE REPEAT LOOP.
015254  020237  015326                                                                  CMP     R2,CKDCNT
015260  003702                                                                          BLE     50301$
015262                                  LET CKDCNT := CKDCNT + #1   ,CKDCNT EQUALS RECORD LENGTH.
015262  005237  015326                                                                  INC     CKDCNT
015266                                  IF CKDFF NE #0 THEN         ;IF COMPARE ERROR HAS OCCURED THEN:
015266  005737  015330                                                                  TST     CKDFF
015272  001414                                                                          BEQ     50306$
015274                                    PRINTB #DTAER3,CKDFF,CKDCNT       ;PRINT # OF BYTES IN ERROR.
015274  013746  015326                                                                  MOV     CKDCNT,-(SP)
015300  013746  015330                                                                  MOV     CKDFF,-(SP)
015304  012746  004746                                                                  MOV     #DTAER3,-(SP)
015310  012746  000003                                                                  MOV     #3,-(SP)
015314  010600                                                                          MOV     SP,R0
015316  104414                                                                          TRAP    C$PNTB
015320  062706  000010                                                                  ADD     #10,SP
015324                                  ENDIF
015324                                                                          50306$:
015324                                ENDIF
015324                                                                          50277$:
015324                              ENDIF
015324                                                                          50275$:
015324  000207                      RTS     PC                      ;OTHERWISE, RETURN.

015326  000000              CKDCNT: .WORD   0                       ;# OF BYTES TO BE VERIFIED -1.
015330  000000              CKDFF:  .WORD   0                       ;# OF BYTES IN ERROR COUNTER.
```

```
                                        GLOBAL SUBROUTINES SECTION

                                ;       SUBROUTINE TO FIND THE FIRST DEVICE IN THE TEST SEQUENCE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:

015332                          FIRSTU:: LET DROPED :B= #0              ;CLR UNIT DROPPED FLAG
015332  105037  003446                                                          CLRB    DROPED
                                        LET R5 := #0                    ;CLR DEVICE POINTER.
015336                                                                          CLR     R5
005005
                                        WHILE DEVTBL(R5) EQ #NINUSE DO  ;WHILE DEVICES ARE NOT IN USE:
015340                                                                  50307$:
015340  026527  002532  177774                                                  CMP     DEVTBL(R5),#NINU
015346  001003                                                                  BNE     50310$
                                          LET R5 := R5 + #2             ;POINT TO NEXT DEVICE.
015350                                                                          ADD     #2,R5
015350  062705  000002
                                        ENDDO
015354                                                                          BR      50307$
000771
                                                                        50310$:
015356                                  IF DEVTBL(R5) EQ #END THEN      ;IF ALL UNITS HAVE BEEN DROPPED THEN:
015356  026527  002532  177777                                                  CMP     DEVTBL(R5),#END
015364  001001                                                                  BNE     50311$
                                          DOCLN                           ;DO CLEAN CODE AND TERMINATE PASS.
015366                                                                          TRAP    C$DCLN
104444
                                        ENDIF
015370                                                                  50311$:
                                        LET L$LUN := DEVTBL(R5)         ;SET UNIT # IN 'HEADER' FOR ERROR REPORT
015370  016537  002532  002074                                                  MOV     DEVTBL(R5),L$LUN
015376  000207                        RTS     PC                      ;RETURN WITH 1ST DEVICE IN R5.
```

```
                                ;       SUBROUTINE TO FIND THE NEXT UNIT IN THE TEST CYCLE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:

015400                          NEXTU:: LET DROPED :B= #0               ;CLR UNIT DROPPED FLAG
015400  105037  003446                                                          CLRB    DROPED
                                        REPEAT                          ;REPEAT UNTIL THE NEXT DEVICE IS FOUND.
015404                                                                  50312$:
015404                                    LET R5 := R5 + #2             ;UPDATE DEVICE TABLE POINTER.
015404  062705  000002                                                          ADD     #2,R5
                                        UNTIL DEVTBL(R5) NE #NINUSE
015410                                                                          CMP     DEVTBL(R5),#NINU
015410  026527  002532  177774                                                  BEQ     50312$
015416  001772                        LET L$LUN := DEVTBL(R5)         ;SET UNIT # IN 'HEADER' FOR ERROR REPORT
015420                                                                          MOV     DEVTBL(R5),L$LUN
015420  016537  002532  002074
015426  000207                        RTS     PC                      ;RETURN.
```

```
                                        ;       SUBROUTINE TO DROP A DEVICE FROM THE TEST SEQUENCE.
                                        ;       INPUTS:
                                        ;       OUTPUTS:
                                        ;       REGISTERS:
                                        ;       CALLS:          MOVMSG, PRXST, LOG

                                        DROPU:: LET FTLCNT(R5) := FTLCNT(R5) + #1  ;INCREMENT THE FATAL ERROR COUNT.
015430  005265  003314                                                              INC     FTLCNT(R5)
                                                LET R4 := MSGPKT+MS.XS3 CLR.BY #377 ;GET UDIAG ERROR CODE FROM XSTAT3.
015434  013704  002350                                                              MOV     MSGPKT+MS.XS3,R4
015440  042704  000377                                                              BIC     #377,R4
                                                LET R3 := MSGPKA(R5)            ;ADR OF THIS UNIT'S MSG PACKET.
015444  016503  002502                                                              MOV     MSGPKA(R5),R3
                                                LET R2 := #0                    ;CLR COUNTER.
015450  005002                                                                    CLR     R2
                                                WHILE R2 NE #MSGCNT DO          ;WHILE THERE ARE MORE LOCATIONS:
015452                                                                          50313$:
015452  020227  000016                                                          CMP     R2,#MSGCNT
015456  001405                                                                  BEQ     50314$
                                                  LET (R3)+ := #-1              ;INIT THE MSG PACKET WITH ALL 1'S
015460  012723  177777                                                          MOV     #-1,(R3)+
                                                  LET R2 := R2 + #2             ;UPDATE COUNTER.
015464  062702  000002                                                          ADD     #2,R2
                                                ENDDO
015470  000770                                                                  BR      50313$
015472                                                                          50314$:
                                                LET @TSDB(R5) := #GSCPK         ;INITIATE A GET STATUS COMMAND.
015472  012775  002320  002452                                                  MOV     #GSCPK,@TSDB(R5)
015500  004737  011152                                                          JSR PC,WSSR                     ;WAIT A WHILE FOR SSR=1
015504  004737  011206                                                          JSR PC,MOVMSG                   ;MOVE MSG PACKET TO COMMON AREA.
                                                IF R4 EQ #X3.RNY THEN           ;IF WE HAVE A CAPSTAN RUNAWAY THEN:
015510  020427  157400                                                          CMP     R4,#X3.RNY
015514  001005                                                                  BNE     50315$
                                                  ERRDF #16,RNYM,STAERM           ;REPORT CAPSTAN RUNAWAY WITH TACH CNT.
015516  104455                                                                  TRAP    C$ERDF
015520  000020                                                                  .WORD   16
015522  004504                                                                  .WORD   RNYM
015524  005372                                                                  .WORD   STAERM
                                                ELSE                            ;ELSE-IF NOT A RUNNAWAY:
015526  000402                                                                  BR      50316$
015530                                                                          50315$:
015530  004737  015642                                                          JSR PC,PRXST                    ;PRINT EXTENDED STATUS REGISTERS.
                                                ENDIF
015534                                                                          50316$:
                                                IFB RECLOG NE #0 THEN           ;IF THE RECORD HAS BEEN LOGGED THEN:
015534  105737  003411                                                          TSTB    RECLOG
015540  001404                                                                  BEQ     50317$
                                                  LET DROPED :B= DROPED + #1    ;SET UNIT DROPPED FLAG.
015542  105237  003446                                                          INCB    DROPED
015546  004737  014016                                                          JSR PC,LOG                      ;LOG DATA BYTES + RD/WR ERRORS.
                                                ENDIF
015552                                                                          50317$:
                                                DORPT                           ;PRINT PERFORMANCE REPORT
015552  104424                                                                  TRAP    C$DRPT
015554                                  DROPUA: IF PASCNT(R5) NE #0 THEN
015554  005765  003254                                                          TST     PASCNT(R5)
```

```
015560  001402                                                                           BEQ     50320$
015562                                    LET PASCNT(R5) := PASCNT(R5) - #1
015562  005365  003254                                                                   DEC     PASCNT(R5)
015566                                  ENDIF
015566                                                                          50320$:
015566                                  LET DROPN := DEVTBL(R5)         ;SAVE # OF UNIT TO BE DROPPED.
015566  016537  002532  015640                                                           MOV     DEVTBL(R5),DROPN
015574                                  LET R0 := R5 SHIFT -1           ;R0=LOGICAL DEVICE NUMBER
015574  010500                                                                           MOV     R5,R0
015576  006200                                                                           ASR     R0
015600                                  DODU R0                   ;DROP THE UNIT: EXEC BGNDU-ENDDU CODE IF IDU = 0
015600  104451                                                                           TRAP    C$DODU
015602                                  IF DEVTBL(R5) NE #NINUSE THEN   ;IF UNIT NOT DROPPED
015602  026527  002532  177774                                                           CMP     DEVTBL(R5),#NINU
015610  001410                                                                           BEQ     50321$
015612                                    IFB IREC EQ #0 THEN             ;IF RECOVERY IS ENABLED THEN:
015612  105737  002211                                                                   TSTB    IREC
015616  001005                                                                           BNE     50322$
015620  000240                              NOP
015622  000240                              NOP
015624  000240                              NOP
015626                                      LET STAFLG :B- STAFLG + #1      ;SET START FLAG TO ENABLE REWIND.
015626  105237  003452                                                                   INCB    STAFLG
015632                                    ENDIF
015632                                                                          50322$:
015632                                  ENDIF
015632                                                                          50321$:
015632                          DRORTN: LET DROPED :B= DROPED + #1      ;SET UNIT DROPPED FLAG.
015632  105237  003446                                                                   INCB    DROPED
015636  000207                          RTS     PC                      ;RETURN.

015640  000000                  DROPN:  .WORD   0                       ;# OF UNIT TO BE DROPPED
```

```
                                        ;       SUBROUTINE TO PRINT EXTENDED STATUS REGISTERS.
                                        ;       INPUTS:
                                        ;       OUTPUTS:
                                        ;       REGISTERS:
                                        ;       CALLS:

015642                                  PRXST:: PRINTX #GETSTM
015642  012746  005157                                                  MOV     #GETSTM,-(SP)
015646  012746  000001                                                  MOV     #1,-(SP)
015652  010600                                                          MOV     SP,R0
015654  104415                                                          TRAP    C$PNTX
015656  062706  000004                                                  ADD     #4,SP
015662                                          PRINTX  #STAER5,MSGPKT+MS.XS0,MSGPKT+MS.XS1,MSGPKT+MS.XS2,MSGPKT+MS.XS3
015662  013746  002350                                                  MOV     MSGPKT+MS.XS3,-(
015666  013746  002346                                                  MOV     MSGPKT+MS.XS2,-(
015672  013746  002344                                                  MOV     MSGPKT+MS.XS1,-(
015676  013746  002342                                                  MOV     MSGPKT+MS.XS0,-(
015702  012746  006217                                                  MOV     #STAER5,-(SP)
015706  012746  000005                                                  MOV     #5,-(SP)
015712  010600                                                          MOV     SP,R0
015714  104415                                                          TRAP    C$PNTX
015716  062706  000014                                                  ADD     #14,SP
015722  000207                                          RTS PC


                                        ;       SUBROUTINE TO HALT AFTER EACH COMMAND.
                                        ;       INPUTS:
                                        ;       OUTPUTS:
                                        ;       REGISTERS:      R3, R4
                                        ;       CALLS:

015724                                  CKHAE:: IFB HAE NE #0 THEN              ;IF HALT FLAG IS SET:
015724  105737  002204                                                          TSTB    HAE
015730  001430                                                                  BEQ     50323$
015732                                            IFB MISCFG EQ #0 THEN         ;
015732  105737  003455                                                          TSTB    MISCFG
015736  001023                                                                  BNE     50324$
015740                                              MANUAL                      ;IS MANUAL INTERVENTION ALLOWED?
015740  104450                                                                  TRAP    C$MANI
015742                                              BNCOMPLETE CKHRTN           ;BR IF NOT.
015742  103023                                                                  BCC     CKHRTN
015744                                              LET R4 := CMDWRD            ;COMMAND WORD.
015744  013704  003346                                                          MOV     CMDWRD,R4
015750  004737  007416                              JSR PC,GCMDA                ;FETCH ADR OF CMD ASCII.
015754                                              LET HALTM :B= (R3)+         ;MOVE CMD ASCII
015754  112337  004042                                                          MOVB    (R3)+,HALTM
015760                                              LET HALTM+1 :B- (R3)+
015760  112337  004043                                                          MOVB    (R3)+,HALTM+1
015764                                              LET HALTM+2 :B  (R3)                ;INTO MESSAGE.
015764  111337  004044                                                          MOVB    (R3),HALTM+2
015770                                              GMANIL HALTM,TIME1,1,YES    ;HALT - WAIT FOR AN OEPRATOR INPUT.
015770  104443                                                                  TRAP    C$GMAN
015772  000404                                                                  BR      10000$
015774  003364                                                                  .WORD   TIME1
```

```
3377  015776  000130                                                                  .WORD   T$CODE
3378  016000  004042                                                                  .WORD   HALTM
3379  016002  000001                                                                  .WORD   1
3380  016004                        10000$:
3381  016004                                 ELSE
3382  016004  000402                                                                  BR      50325$
3383  016006                                                                  50324$:
3384  016006                                   LET MISCFG :B= #0             ;
3385  016006  105037  003455                                                         CLRB    MISCFG
3386  016012                                 ENDIF
3387  016012                                                                  50325$:
3388  016012                               ENDIF
3389  016012                                                                  50323$:
3390  016012  000207                 CKHRTN: RTS     PC                      ;RETURN
3391                                        .EVEN
3392
3393  016014                               ENDMOD
```

```
3394
3395                                    .TITLE MISCELLANEOUS SECTIONS
3396                                    .SBTTL  REPORT CODING SECTION
3397
3398  016014                                    BGNMOD
3399
3400                                    ;++
3401                                    ; THE REPORT CODING SECTION CONTAINS THE
3402                                    ; 'PRINTS'' CALLS THAT GENERATE STATISTICAL REPORTS.
3403                                    ;--
3404
3405  016014                                    BGNRPT
3406  016014                            L$RPT::
3407
3408
3409  016014                                    LET     R5SAVE := R5            ;SAVE CURRENT DEVICE POINTER.
3410  016014  010537  003400                                                            MOV     R5,R5SAVE
3411  016020  004737  015332                    JSR     PC,FIRSTU               ;FIND THE FIRST UNIT.
3412  016024                                    WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE MORE DEVICES:
3413  016024                                                                                      50326$:
3414  016024  026527  002532  177777                                                          CMP     DEVTBL(R5),#END
3415  016032  001562                                                                        BEQ     50327$
3416  016034                                      PRINTS        #RPT1A,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
3417  016034  016546  003324                                                              MOV     RECCNT(R5),-(SP)
3418  016040  016546  003254                                                              MOV     PASCNT(R5),-(SP)
3419  016044  016546  002532                                                              MOV     DEVTBL(R5),-(SP)
3420  016050  012746  016656                                                              MOV     #RPT1A,-(SP)
3421  016054  012746  000004                                                              MOV     #4,-(SP)
3422  016060  010600                                                                      MOV     SP,R0
3423  016062  104416                                                                      TRAP    C$PNTS
3424  016064  062706  000012                                                              ADD     #12,SP
3425  016070                                PRINTS        #RPT1B,WRBC+30(R5),WRBC+20(R5),WRBC+10(R5),WRBC(R5)
3426  016070  016546  002554                                                              MOV     WRBC(R5),-(SP)
3427  016074  016546  002564                                                              MOV     WRBC+10(R5),-(SP
3428  016100  016546  002574                                                              MOV     WRBC+20(R5),-(SP
3429  016104  016546  002604                                                              MOV     WRBC+30(R5),-(SP
3430  016110  012746  016733                                                              MOV     #RPT1B,-(SP)
3431  016114  012746  000005                                                              MOV     #5,-(SP)
3432  016120  010600                                                                      MOV     SP,R0
3433  016122  104416                                                                      TRAP    C$PNTS
3434  016124  062706  000014                                                              ADD     #14,SP
3435  016130                                PRINTS        #RPT1C,RRBC+30(R5)                  C+10(R5),RRBC(R5)
3436  016130  016546  002614                                                              MOV     RRBC(R5),-(SP)
3437  016134  016546  002624                                                              MOV     RRBC+10(R5),-(SP
3438  016140  016546  002634                                                              MOV     RRBC+20(R5),-(SP
3439  016144  016546  002644                                                              MOV     RRBC+30(R5),-(SP
3440  016150  012746  017004                                                              MOV     #RPT1C,-(SP)
3441  016154  012746  000005                                                              MOV     #5,-(SP)
3442  016160  010600                                                                      MOV     SP,R0
3443  016162  104416                                                                      TRAP    C$PNTS
3444  016164  062706  000014                                                              AD:     #14,SP
3445  016170                                PRINTS        #RPT1D,RFBC+                        (R5),RFBC(R5)
3446  016170  016546  002654                                                              MOV     RFBC(R5),-(SP)
3447  016174  016546  002664                                                              MOV     RFBC+10(R5),-(SP
3448  016200  016546  002674                                                              MOV     RFBC+20(R5),-(SP
3449  016204  016546  002704                                                              MOV     RFBC+30(R5),-(SP
```

```
3450  016210  012746  017055                                                  MOV     #RPT1D,-(SP)
3451  016214  012746  000005                                                  MOV     #5,-(SP)
3452  016220  010600                                                         MOV     SP,R0
3453  016222  104416                                                         TRAP    C$PNTS
3454  016224  062706  000014                                                 ADD     #14,SP
3455  016230                        PRINTS        #RPT1F,WRREC(R5),RRREC(R5),RFREC(R5)
3456  016230  016546  002754                                                  MOV     RFREC(R5),-(SP)
3457  016234  016546  002734                                                  MOV     RRREC(R5),-(SP)
3458  016240  016546  002714                                                  MOV     WRREC(R5),-(SP)
3459  016244  012746  017161                                                  MOV     #RPT1F,-(SP)
3460  016250  012746  000004                                                  MOV     #4,-(SP)
3461  016254  010600                                                         MOV     SP,R0
3462  016256  104416                                                         TRAP    C$PNTS
3463  016260  062706  000012                                                 ADD     #12,SP
3464  016264                        PRINTS        #RPT1G,WRUNR(R5),RRUNR(R5),RFUNR(R5)
3465  016264  016546  002764                                                  MOV     RFUNR(R5),-(SP)
3466  016270  016546  002744                                                  MOV     RRUNR(R5),-(SP)
3467  016274  016546  002724                                                  MOV     WRUNR(R5),-(SP)
3468  016300  012746  017232                                                  MOV     #RPT1G,-(SP)
3469  016304  012746  000004                                                  MOV     #4,-(SP)
3470  016310  010600                                                         MOV     SP,R0
3471  016312  104416                                                         TRAP    C$PNTS
3472  016314  062706  000012                                                 ADD     #12,SP
3473  016320                        IFB BADTSW NE #0 THEN         ;
3474  016320  105737  002206                                                  TSTB    BADTSW
3475  016324  001402                                                         BEQ     50330$
3476  016326  004737  016410          JSR PC,BTRPT          ;GO PRINT BAD TAPE SPOTS WHEN  ENABLED
3477  016332                        ENDIF
3478  016332                                                        50330$:
3479  016332                        PRINTS        #RPT1I,SCCNT(R5),HRDCNT(R5),FTLCNT(R5),VFYCNT(R5)
3480  016332  016546  003274                                                  MOV     VFYCNT(R5),-(SP)
3481  016336  016546  003314                                                  MOV     FTLCNT(R5),-(SP)
3482  016342  016546  003304                                                  MOV     HRDCNT(R5),-(SP)
3483  016346  016546  003264                                                  MOV     SCCNT(R5),-(SP)
3484  016352  012746  017427                                                  MOV     #RPT1I,-(SP)
3485  016356  012746  000005                                                  MOV     #5,-(SP)
3486  016362  010600                                                         MOV     SP,R0
3487  016364  104416                                                         TRAP    C$PNTS
3488  016366  062706  000014                                                 ADD     #14,SP
3489  016372  004737  015400          JSR   PC,NEXTU              ;FIND THE NEXT UNIT.
3490  016376                        ENDDO
3491  016376  000612                                                         BR      50326$
3492  016400                                                        50327$:
3493  016400                        LET R5 := R5SAVE            ;RESTORE CURRENT DEVICE POINTER.
3494  016400  013705  003400                                                  MOV     R5SAVE,R5
3495  016404                        EXIT    RPT
3496  016404  000167                                                         .WORD   J$JMP
3497  016406  001130                                                         .WORD   L10010-2-.
3498
3499
3500
3501
3502                          ;       SUBR TO PRINT BAD TAPES SPOTS DURING THE REPORT PRINTS
3503                          ;       WRITE RETRIES: CUMULATIVE COUNT
3504                          ;       BAD TAPE SPOTS: COUNT PER TAPE PASS ONLY, NOT CUMULATIVE.
3505                          ;       COUNT OF RECOVERABLE WRITE ERRORS EXCLUDES BAD TAPE SPOTS.
```

```
3506
3507
3508  016410                                BTRPT:  PRINTS  #RPT1E,WRTYCT(R5)       ;PRINT GLOBAL WRITE RETRY COUNT
3509  016410  016546  003244                                                                MOV     WRTYCT(R5),-(SP)
3510  016414  012746  017303                                                                MOV     #RPT1E,-(SP)
3511  016420  012746  000002                                                                MOV     #2,-(SP)
3512  016424  010600                                                                        MOV     SP,R0
3513  016426  104416                                                                        TRAP    C$PNTS
3514  016430  062706  000006                                                                ADD     #6,SP
3515  016434                                        LET BTPT := BTADDR(R5)  ;BTPT IS BOTH THE BAD TAPE SPOT COUNTER
3516  016434  016537  002544  003436                                                        MOV     BTADDR(R5),BTPT
3517  016442                                        LET R3 := @BTPT SHIFT -1        ;AND THE LOGGING INDEX
3518  016442  017703  164770                                                                MOV     @BTPT,R3
3519  016446  006203                                                                        ASR     R3
3520  016450                                        PRINTS  #RPT1J,R3       ;PRINT # OF BAD TAPE SPOTS
3521  016450  010346                                                                        MOV     R3,-(SP)
3522  016452  012746  017333                                                                MOV     #RPT1J,-(SP)
3523  016456  012746  000002                                                                MOV     #2,-(SP)
3524  016462  010600                                                                        MOV     SP,R0
3525  016464  104416                                                                        TRAP    C$PNTS
3526  016466  062706  000006                                                                ADD     #6,SP
3527  016472                                        IF R3 NE #0 THEN                ;PRINT RECORD # IF BAD SPOTS DETECTED
3528  016472  005703                                                                        TST     R3
3529  016474  001457                                                                        BEQ     50331$
3530  016476                                          IF R3 HI #20. THEN            ;
3531  016476  020327  000024                                                                CMP     R3,#20.
3532  016502  101402                                                                        BLOS    50332$
3533  016504                                            LET R3 := #20.              ;20 BAD SPOTS IS THE LIMIT
3534  016504  012703  000024                                                                MOV     #20.,R3
3535  016510                                          ENDIF
3536  016510                                                                  50332$:
3537  016510                                        PRINTS          #CRLFSP         ;
3538  016510  012746  005216                                                                MOV     #CRLFSP,-(SP)
3539  016514  012746  000001                                                                MOV     #1,-(SP)
3540  016520  010600                                                                        MOV     SP,R0
3541  016522  104416                                                                        TRAP    C$PNTS
3542  016524  062706  000004                                                                ADD     #4,SP
3543  016530                                        LET R4 :- BTPT + #2     ;FETCH A BAD SPOT ID
3544  016530  013704  003436                                                                MOV     BTPT,R4
3545  016534  062704  000002                                                                ADD     #2,R4
3546  016540                                        LET R2 := #0            ;R2 = PRINT COUNT PER LINE: 10 MAX
3547  016540  005002                                                                        CLR     R2
3548  016542                                        REPEAT                          ;
3549  016542                                                                  50333$:
3550  016542                                          PRINTS        #RPT1K,(R4)     ;PRINT A BAD SPOT ID
3551  016542  011446                                                                        MOV     (R4),-(SP)
3552  016544  012746  017420                                                                MOV     #RPT1K,-(SP)
3553  016550  012746  000002                                                                MOV     #2,-(SP)
3554  016554  010600                                                                        MOV     SP,R0
3555  016556  104416                                                                        TRAP    C$PNTS
3556  016560  062706  000006                                                                ADD     #6,SP
3557  016564                                          LET R2 := R2 + #1     ;COUNT PRINTS
3558  016564  005202                                                                        INC     R2
3559  016566                                          LET R4 := R4 + #2     ;NEXT
3560  016566  062704  000002                                                                ADD     #2,R4
3561  016572                                          IF R2 EQ #10. THEN    ;
```

```
3562  016572  020227  000012                                                          CMP     R2,#10.
3563  016576  001014                                                                  BNE     50334$
3564  016600                          PRINTS #CRLFSP          ;GO TO NEXT PRINT LINE PAST 10 PRINTS
3565  016600  012746  005216                                                          MOV     #CRLFSP,-(SP)
3566  016604  012746  000001                                                          MOV     #1,-(SP)
3567  016610  010600                                                                  MOV     SP,R0
3568  016612  104416                                                                  TRAP    C$PNTS
3569  016614  062706  000004                                                          ADD     #4,SP
3570  016620                          LET R3 := R3 - #10.     ;ADJUST BAD SPOT COUNT
3571  016620  162703  000012                                                          SUB     #10.,R3
3572  016624                          LET R2 := R2 - #10.     ;ADJUST PRINT COUNT
3573  016624  162702  000012                                                          SUB     #10.,R2
3574  016630                        ENDIF                     ;
3575  016630                                                                  50334$:
3576  016630                      UNTIL R2 EQ R3              ;LIMIT: # OF BAD SPOTS
3577  016630  020203                                                                  CMP     R2,R3
3578  016632  001343                                                                  BNE     50333$
3579  016634                    ENDIF                         ;
3580  016634                                                                  50331$:
3581  016634                    PRINTS  #CRLF                 ;
3582  016634  012746  005213                                                          MOV     #CRLF,-(SP)
3583  016640  012746  000001                                                          MOV     #1,-(SP)
3584  016644  010600                                                                  MOV     SP,R0
3585  016646  104416                                                                  TRAP    C$PNTS
3586  016650  062706  000004                                                          ADD     #4,SP
3587  016654  000207                    R*S PC
3588
3589
3590                                    .NLIST  BEX
      016656  047045  047045  040445  RPT1A:  .ASCIZ  /%N%N%AUNIT %D1%S3%APASS:%D5%S3%ARECORD:%D5%N/
      016733     045  041101  052131  RPT1B:  .ASCIZ  /%ABYTES WRITTEN  %D3%A,%Z3%A,%Z3%A,%Z3%N/
      017004  040445  054502  042524  RPT1C:  .ASCIZ  /%ABYTES READ REV %D3%A,%Z3%A,%Z3%A,%Z3%N/
      017055     045  041101  052131  RPT1D:  .ASCII  /%ABYTES READ FWD %D3%A,%Z3%A,%Z3%A,%Z3%N/
      017125     045  031123  022463          .ASCIZ  /%S23%AWRT%S4%ARDR%S4%ARDF%N/
      017161     045  051101  041505  RPT1F:  .ASCIZ  /%ARECOVERABLE ERRORS   %D5%S2%D5%S2%D5%N/
      017232  040445  047125  042522  RPT1G:  .ASCIZ  /%AUNRECOVERABLE ERRORS %D5%S2%D5%S2%D5%N/
      017303     045  053501  044522  RPT1E:  .ASCIZ  /%AWRITE RETRIES%S8%D5%N/
      017333     045  022516  031104  RPT1J:  .ASCIZ  /%N%D2%A BAD SPOTS THIS TAPE PASS PRECEDING RECORD #:/
      017420  042045  022465  030523  RPT1K:  .ASCIZ  /%D5%S1/
      017427     045  051501  042520  RPT1I:  .ASCII  '%ASPEC COND%S3%AHARD%S3%AFATAL%S3%ACOMPARE%N''
      017503     045  031523  042045          .ASCIZ  /%S3%D5%S3%D5%S3%D5%S3%D5%N%N/
                                              .LIST   BEX
3591                                          .EVEN
3592
3593  017540                                  ENDRPT
3594  017540                          L10010:
3595  017540  104425                                                          TRAP    C$RPT
3596
3597                                  .SBTTL  LOAD DEVICE PROTECTION TABLE
3598
3599                                  ;++
3600                                  ;TABLE FOR SUPERVISOR TO IDENTIFY THE P-TBL FOR THE LOAD DEV
3601                                  ;THE SUPERVISOR USES THE TBL TO WARN THE OPERATOR WHEN HE TRIES TO TEST THE LOAD DEV
3602                                  ;--
3603
3604  017542                                  BGNPROT
```

```
                                LOAD DEVICE PROTECTION TABLE

  3605  017542                  L$PROT::
  3606  017542  000000                .WORD 0             ;P-TBL OFFSET OF TSSR, THE TS11 CSR
  3607  017544  177777                .WORD -1            ;P-TBL OFFSET OF MASS BUS UNIT #: -1 = NOT A MASS BUS DE
  3608  017546  177777                .WORD -1            ;P-TBL OFFSET OF DRIVE #: -1 - NONE, ONE DRIVE PER UNIBU
  3609  017550                      ENDPRCT
```

```
                                        INITIALIZE SECTION

                                        .SBTTL  INITIALIZE SECTION

                                        ;++
                                        ; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
                                        ; AT THE BEGINNING OF EACH PASS.
                                        ;--

                                                BGNINIT
                                        L$INIT::

                                        INIT10: IF #BIT0!BIT1 SETIN #CMDPKT THEN  ;IF CMD PACKET IS NOT ON MODULO 4 BOUNDRY:
017550  032727  000003  002310                                                    BIT     #BIT0!BIT1,#CMDP
017556  001421                                                                    BEQ     50335$
                                                  ERRSF #1,CMDPKM               ;PRINT ERROR MSG,
017560  104454                                                                    TRAP    C$ERSF
017562  000001                                                                    .WORD   1
017564  004102                                                                    .WORD   CMDPKM
017566  000000                                                                    .WORD   0
                                                  DELAY 2000.                   ;GO TO SUPERVISOR, WAIT 2 SECONDS.
017570  012727  003720                                                            MOV     #2000.,(PC)+
017574  000000                                                                    .WORD   0
017576  013727  002116                                                            MOV     L$DLY,(PC)+
017602  000000                                                                    .WORD   0
017604  005367  177772                                                            DEC     -6(PC)
017610  001375                                                                    BNE     .-4
017612  005367  177756                                                            DEC     -22(PC)
017616  001367                                                                    BNE     .-20
017620  000753                                  BR      INIT10                ;
017622                                          ENDIF
017622                                                                                  50335$:
017622
017622  105737  002202                      IFB CLRFLG NE #0 THEN             ;IF CLR COUNTERS FLAG SET:
017626  001413                                                                    TSTB    CLRFLG
017630  105037  002202                        CLRB  CLRFLG                    ;INIT CLR FLAG.
017634                                        LET R2 := #0
017634  005002                                                                    CLR     R2
017636                                        WHILE R2 NE #CNTLEN DO
017636                                                                                  50337$:
017636  020227  000550                                                            CMP     R2,#CNTLEN
017642  001405                                                                    BEQ     50340$
017644                                          LET WRBC(R2) := #0              ;CLR ALL STATISTICAL COUNTERS.
017644  005062  002554                                                            CLR     WRBC(R2)
017650                                          LET R2 :- R2 + #2
017650  062702  000002                                                            ADD     #2,R2
017654                                        ENDDO
017654  000770                                                                    BR      50337$
017656                                                                                  50340$:
017656                                  ENDIF
017656                                                                                  50336$:
017656
017656  105737  002203                      IFB RRANV NE #0 THEN              ;IF RESET RANDOM VARIABLE FLAG IS SET THEN:
017662  001406                                                                    TSTB    RRANV
017664                                                                            BEQ     50341$
017664  012737  153624  003360                LET RANB :- #RANBC                ;RESET RANDOM BASE #.
                                                                                  MOV     #RANBC,RANB
```

```
3666  017672                                  LET RANS := #RANSC              ,RESET RANDOM SAVE LOCATION.
3667  017672  012737  032561  003362                                                  MOV     #RANSC,RANS
3668  017700                                  ENDIF
3669  017700                                                                50341$:
3670  017700                                  READEF  #EF.START               ;READ START COMMAND EVENT FLAG.
3671  017700  012700  000040                                                  MOV     #EF.START,R0
3672  017704  104447                                                          TRAP    C$REFG
3673  017706                                  BNCOMPLETE      INIT15          ;BRANCH IF NOT STARTING.
3674  017706  103026                                                          BCC     INIT15
3675  017710                                  LET STAFLG :B= STAFLG + #1       ;SET START COMMAND FLAG.
3676  017710  105237  003452                                                  INCB    STAFLG
3677  017714                                  LET R5 := #6
3678  017714  012705  000006                                                  MOV     #6,R5
3679  017720                                  REPEAT                          ;INITIATE UNIT NUMBER TABLE
3680  017720                                                                50342$:
3681  017720                                    LET DEVTBL(R5) := #NINUSE     ;BY STORING NOT IN USE IN EACH LOCATION.
3682  017720  012765  177774  002532                                          MOV     #NINUSE,DEVTBL(R
3683  017726                                    LET R5 := R5 - #2
3684  017726  162705  000002                                                  SUB     #2,R5
3685  017732                                  UNTIL R5 EQ #0
3686  017732  005705                                                          TST     R5
3687  017734  001371                                                          BNE     50342$
3688  017736                                  LET R5 := L$UNIT SHIFT 1
3689  017736  013705  002012                                                  MOV     L$UNIT,R5
3690  017742  006305                                                          ASL     R5
3691  017744                                  REPEAT                          ;STORE ALL UNIT
3692  017744                                                                50343$:
3693  017744                                    LET R5 := R5 - #2             ;NUMBERS IN DEVTBL.
3694  017744  162705  000002                                                  SUB     #2,R5
3695  017750                                    LET DEVTBL(R5) := R5 SHIFT -1
3696  017750  010565  002532                                                  MOV     R5,DEVTBL(R5)
3697  017754  006265  002532                                                  ASR     DEVTBL(R5)
3698  017760                                  UNTIL R5 EQ #0
3699  017760  005705                                                          TST     R5
3700  017762  001370                                                          BNE     50343$
3701
3702  017764                                  INIT15: READEF #EF.PWR          ;HAS THERE BE A POWER FAILURE?
3703  017764  012700  000034                                                  MOV     #EF.PWR,R0
3704  017770  104447                                                          TRAP    C$REFG
3705  017772                                  BNCOMPLETE INIT16               ;BRANCH IF NOT.
3706  017772  103004                                                          BCC     INIT16
3707  017774                                    LET STAFLG :B= STAFLG + #1      ;IF SO - SET THE START FLAG.
3708  017774  105237  003452                                                  INCB    STAFLG
3709  020000                                    LET PWRFLG :B= PWRFLG + #1     ;IF SO - SET THE POWER FAIL FLAG.
3710  020000  105237  003453                                                  INCB    PWRFLG
3711
3712  020004                                  INIT16: RFLAGS  OPFLAG          ;READ AND STORE FLAGS SET BY OPERATOR
3713  020004  104421                                                          TRAP    C$RFLA
3714  020006  010037  003456                                                  MOV     R0,OPFLAG
3715  020012                                  LET R3 := #0                    ;CLEAR EVENT FLAG
3716  020012  005003                                                          CLR     R3
3717  020014                                      IFB PWRFLG EQ #0 THEN       ;IF POWER FAIL HAS NOT OCCURRED THEN:
3718  020014  105737  003453                                                  TSTB    PWRFLG
3719  020020  001020                                                          BNE     50344$
3720  020022                                        READEF #EF.NEW            ;UPDATE PASS COUNT WHEN
3721  020022  012700  000035                                                  MOV     #EF.NEW,R0
```

```
020026  104447                                                                      TRAP    C$REFG
020030                                   IFCOND CS THEN          ;SUPERVISOR IS IN NEW PASS
020030  103014                                                                      BCC     50345$
020032                                     IFB STAFLG EQ #0 THEN  ;AND DIAG WAS NEITHER STARTED
020032  105737  003452                                                              TSTB    STAFLG
020036  001010                                                                      BNE     50346$
020040                                       READEF #EF.RES      ;NOR
020040  012700  000037                                                              MOV     #EF.RES,R0
020044  104447                                                                      TRAP    C$REFG
020046                                       IFCOND CC THEN      ;RESTARTED
020046  103402                                                                      BCS     50347$
020050                                         LET R3 := COMP R3 ;DO IT
020050  005103                                                                      COM     R3
020052                                       ELSE
020052  000401                                                                      BR      50350$
020054                                                                      50347$:
020054                                         LET R3 := R3 + #1 ;SET 1ST PASS IF NEW PASS AND
020054  005203                                                                      INC     R3
020056                                       ENDIF               ;RESTARTING
020056                                                                      50350$:
020056                                     ELSE
020056  000401                                                                      BR      50351$
020060                                                                      50346$:
020060                                       LET R3 := R3 + #1   ;SET 1ST PASS IF NEW PASS AND
020060  005203                                                                      INC     R3
020062                                     ENDIF                 ;STARTING
020062                                                                      50351$:
020062                                   ENDIF                   ;DO NOT UPDATE IT ON CONTINUE
020062                                                                      50345$:
020062                                 ENDIF                     ;OR ON POWER FAIL
020062                                                                      50344$:
020062  004737  015332              JSR     PC,FIRSTU            ;INIT DEVICE POINTER.
020066                              LET R2 := #0                 ;INIT DEVICE COUNTER.
020066  005002                                                                      CLR     R2
020070                              WHILE DEVTBL(R5) NE #END DO
020070                                                                      50352$:
020070  026527  002532  177777                                                      CMP     DEVTBL(R5),#END
020076  001450                                                                      BEQ     50353$
020100                                LET R2 := R2 + #1
020100  005202                                                                      INC     R2
020102                                LET R0 := R5 SHIFT -1
020102  010500                                                                      MOV     R5,R0
020104  006200                                                                      ASR     R0
020106                                GPHARD  R0,R0              ;GET HARDWARE P TABLE FROM SUPER.
020106  104442                                                                      TRAP    C$GPHRD
020110                                IFCOND CS THEN
020110  103036                                                                      BCC     50354$
020112                                  LET TSSR(R5) := (R0)     ;SAVE TSSR ADDRESS.
020112  011065  002462                                                              MOV     (R0),TSSR(R5)
020116                                  LET TSDB(R5) := (R0)+ - #2  ;SAVE TSDB ADDRESS.
020116  012065  002452                                                              MOV     (R0)+,TSDB(R5)
020122  162765  000002  002452                                                      SUB     #2,TSDB(R5)
020130                                  LET TSVCT(R5) := (R0)    ;SAVE INTERRUPT VECTOR ADDRESS.
020130  011065  002472                                                              MOV     (R0),TSVCT(R5)
020134                                  SETVEC TSVCT(R5),TS4INT(R5),#INTPRI ;SET UP INTERUPT PROCESSING CONDITIONS.
020134  012746  000340                                                              MOV     #INTPRI,-(SP)
```

```
3778  020140  016546  002512                                                          MOV     TS4INT(R5),-(SP)
3779  020144  016546  002472                                                          MOV     TSVCT(R5),-(SP)
3780  020150  012746  000003                                                          MOV     #3,-(SP)
3781  020154  104437                                                                 TRAP    C$SVEC
3782  020156  062706  000010                                                          ADD     #10,SP
3783  020162                        IF R3 NE #0 THEN          ;ACTUAL PASSCOUNT UPDATE PER R3
3784  020162  005703                                                                 TST     R3
3785  020164  001410                                                                 BEQ     50355$
3786  020166                          IF R3 LT #0 THEN
3787  020166  005703                                                                 TST     R3
3788  020170  002003                                                                 BGE     50356$
3789  020172                            LET PASCNT(R5) := PASCNT(R5) + #1
3790  020172  005265  003254                                                        INC     PASCNT(R5)
3791  020176                          ELSE
3792  020176  000403                                                                 BR      50357$
3793  020200                                                                50356$:
3794  020200                            LET PASCNT(R5) := #1
3795  020200  012765  000001  003254                                               MOV     #1,PASCNT(R5)
3796  020206                          ENDIF
3797  020206                                                                50357$:
3798  020206                        ENDIF
3799  020206                                                                50355$:
3800  020206                      ENDIF
3801  020206                                                                50354$:
3802  020206                      LET RECCNT(R5) := #0      ;CLEAR RECORD COUNT
3803  020206  005065  003324                                                        CLR     RECCNT(R5)
3804  020212  004737  015400             JSR   PC,NEXTU    ;DO IT FOR ALL DEVICES.
3805  020216                      ENDDO
3806  020216  000724                                                                BR      50352$
3807  020220                                                                50353$:
3808
3809  020220                      IF R2 EQ #0 THEN          ;IF THERE ARE NO UNITS:
3810  020220  005702                                                                 TST     R2
3811  020222  001026                                                                 BNE     50360$
3812  020224                        PRINTF #AUDR 1          ;PRINT ALL UNITS DROPPED,
3813  020224  012746  004645                                                        MOV     #AUDRPM,-(SP)
3814  020230  012746  000001                                                        MOV     #1,-(SP)
3815  020234  010600                                                                MOV     SP,R0
3816  020236  104417                                                                TRAP    C$PNTF
3817  020240  062706  000004                                                        ADD     #4,SP
3818  020244                        DELAY 2000.             ;GO TO SUPERVISOR, WAIT 2 SECONDS.
3819  020244  012727  003720                                                        MOV     #2000.,(PC)+
3820  020250  000000                                                                .WORD   0
3821  020252  013727  002116                                                        MOV     L$DLY,(PC)+
3822  020256  000000                                                                .WORD   0
3823  020260  005367  177772                                                        DEC     -6(PC)
3824  020264  001375                                                                BNE     .-4
3825  020266  005367  177756                                                        DEC     -22(PC)
3826  020272  001367                                                                BNE     .-20
3827  020274
3828  020274  104422                        BREAK             ;GO TO SUPERVISOR, CHECK TTY.
3829  020276                                                                        TRAP    C$BRK
3830  020276  104444                      DOCLN               ;DO CLEAN CODE + ABORT PASS.
3831  020300                                                                        TRAP    C$DCLN
3832  020300                      ENDIF
3833                                                                        50360$:
```

```
3834  020300                                  SETPRI #PRI00                   ;LOWER CPU PRIORITY TO 0
3835  020300  012700  000000                                                          MOV     #PRI00,R0
3836  020304  104441                                                                  TRAP    C$SPRI
3837  020306                                  IFB IREC EQ #0 AND #ADR NOTSETIN OPFLAG THEN    ;IF ERROR RECOVERY IS ENABLED
3838  020306  105737  002211                                                          TSTB    IREC
3839  020312  001145                                                                  BNE     50361$
3840  020314  032737  000020  003456                                                  BIT     #ADR,OPFLAG
3841  020322  001141                                                                  BNE     50361$
3842  020324  004737  015332                    JSR PC,FIRSTU                   ;AND AUTO-DROP NOT CALLED, THEN SET UP FOR FIRST
3843  020330                                    WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE MORE DEVICES:
3844  020330                                                                  50362$:
3845  020330  026527  002532  177777                                                  CMP     DEVTBL(R5),#END
3846  020336  001533                                                                  BEQ     50363$
3847  020340                                      BEGIN COUNTER                 ;START 3.5 MINUTE COUNTER
3848  020340                                        INCR TIME1 FROM #1 TO #25 BY #1
3849  020340  012737  000001  003364                                                  MOV     #1,TIME1
3850  020346  000402                                                                  BR      50365$
3851  020350                                                                  50366$:
3852  020350  005237  003364                                                          INC     TIME1
3853  020354                                                                  50365$:
3854  020354  023727  003364  000025                                                  CMP     TIME1,#25
3855  020362  003106                                                                  BGT     50367$
3856  020364                                          LE      ,DB(R5) := #GSCPK ;AND GET UNITS STATUS
3857  020364  012775  002320  002452                                                  MOV     #GSCPK,@TSDB(R5)
3858  020372                                          DEL     1               ;WAIT
3859  020372  012727  000001                                                          MOV     #1,(PC)+
3860  020376  000000                                                                  .WORD   0
3861  020400  013727  002116                                                          MOV     L$DLY,(PC)+
3862  020404  000000                                                                  .WORD   0
3863  020406  005367  177772                                                          DEC     -6(PC)
3864  020412  001375                                                                  BNE     .-4
3865  020414  005367  177756                                                          DEC     -22(PC)
3866  020420  001367                                                                  BNE     .-20
3867  020422                                  IF #TS.SSR SETIN @TSSR(R5) THEN
3868  020422  032775  000200  002462                                                  BIT     #TS.SSR,@TSSR(R5
3869  020430  001420                                                                  BEQ     50370$
3870  020432                                    IF #TS.OFL NOTSETIN @TSSR(R5) THEN
3871  020432  032775  000100  002462                                                  BIT     #TS.OFL,@TSSR(R5
3872  020440  001001                                                                  BNE     50371$
3873  020442                                      LEAVE COUNTER               ;EXIT COUNTER WHEN UNIT ON LINE
3874  020442  000456                                                                  BR      50364$
3875  020444                                    ELSE
3876  020444                                                                  50371$:
3877  020444                                      PRINTF #OFLINM,DEVTBL(R5) ;PRINT UNIT OFF LINE EVERY 10 SEC
3878  020444  016546  002532                                                          MOV     DEVTBL(R5),-(SP)
3879  020450  012746  005127                                                          MOV     #OFLINM,-(SP)
3880  020454  012746  000002                                                          MOV     #2,-(SP)
3881  020460  010600                                                                  MOV     SP,R0
3882  020462  10441?                                                                  TRAP    C$PNTF
3883  020464  062706  000006                                                          ADD     #6,SP
3884  020470                                    ENDIF
3885  020470                                                                  50372$:
3886  020470                                  ELSE
3887  020470  000412                                                                  BR      50373$
3888  020472                                                                  50370$:
3889  020472                                    PRINTF #NRDYM,DEVTBL(R5)
```

```
3890  020472  016546  002532                                                        MOV     DEVTBL(R5),-(SP)
3891  020476  012746  021424                                                        MOV     #NRDYM,-(SP)
3892  020502  012746  000002                                                        MOV     #2,-(SP)
3893  020506  010600                                                               MOV     SP,R0
3894  020510  104417                                                               TRAP    C$PNTF
3895  020512  062706  000006                                                       ADD     #6,SP
3896  020516                        ENDIF
3897  020516                                                                50373$:
3898  020516                        INCR TIME2 FROM #1 TO #13 BY #1
3899  020516  012737  000001  003366                                              MOV     #1,TIME2
3900  020524  000402                                                               BR      50374$
3901  020526                                                                50375$:
3902  020526  005237  003366                                                       INC     TIME2
3903  020532                                                                50374$:
3904  020532  023727  003366  000013                                              CMP     TIME2,#13
3905  020540  003016                                                               BGT     50376$
3906  020542                          DELAY 1000.           ;WAIT FOR UNIT TO BE SET ON-LINE
3907  020542  012727  001750                                                       MOV     #1000.,(PC)+
3908  020546  000000                                                               .WORD   0
3909  020550  013727  002116                                                       MOV     L$DLY,(PC)+
3910  020554  000000                                                               .WORD   0
3911  020556  005367  177772                                                       DEC     -6(PC)
3912  020562  001375                                                               BNE     .-4
3913  020564  005367  177756                                                       DEC     -22(PC)
3914  020570  001367                                                               BNE     .-20
3915  020572                          BREAK           ;ALLOW TERMINAL INTERRUPT
3916  020572  104422                                                               TRAP    C$BRK
3917  020574                        ENDINC
3918  020574  000754                                                               BR      50375$
3919  020576                                                                50376$:
3920  020576                          ENDINC
3921  020576  000664                                                               BR      50366$
3922  020600                                                                50367$:
3923  020600                        END COUNTER
3924  020600                                                                50364$:
3925  020600                        IF TIME1 GT #25 THEN          ;IF OFF LINE FOR 3.5 MINUTES
3926  020600  023727  003364  000025                                              CMP     TIME1,#25
3927  020606  003404                                                               BLE     50377$
3928  020610  004737  011206          JSR PC,MOVMSG             ;GET MESSAGE PACKET
3929  020614  004737  011702          JSR PC,TCC1               ;PRINT ERROR AND DROP OFF LINE UNIT
3930  020620                        ENDIF
3931  020620                                                                50377$:
3932                                  ;REPEAT UNTIL ON LINE OR TIMED OUT.
3933  020620  004737  015400        JSR PC,NEXTU              ;SET UP FOR NEXT UNIT.
3934  020624                      ENDDO
3935  020624  000641                                                               BR      50362$
3936  020626                                                                50363$:
3937  020626                    ENDIF
3938  020626                                                                50361$.
3939  020626                    IFB PWRFLG EQ #0 THEN
3940  020626  105737  003453                                                       TSTB    PWRFLG
3941  020632  001026                                                               BNE     50400$
3942  020634                      MEMORY DATAWT               ;REQUEST MEMORY FROM SUPER FOR RD/WR BUFFERS.
3943  020634  104431                                                               TRAP    C$MEM
3944  020636  010037  003334                                                       MOV     R0,DATAWT
3945  020642                      LET DATARD := DATAWT + #DATCNT          ;SET RD BFR AD
```

```
3946  020642  013737  003334  003336                                                  MOV     DATAWT,DATARD
3947  020650  062737  004000  003336                                                  ADD     #DATCNT,DATARD
3948  020656                          IF @DATAWT LT #DATCNT THEN    ;WHEN NOT ENOUGH FREE MEMO AVAILABLE
3949  020656  027727  162452  004000                                                  CMP     @DATAWT,#DATCNT
3950  020664  002011                                                                  BGE     50401$
3951  020666                            PRINTF #MEMOM               ,WARN OPERATOR
3952  020666  012746  020734                                                          MOV     #MEMOM,-(SP)
3953  020672  012746  000001                                                          MOV     #1,-(SP)
3954  020676  010600                                                                  MOV     SP,R0
3955  020700  104417                                                                  TRAP    C$PNTF
3956  020702  062706  000004                                                          ADD     #4,SP
3957  020706                            DOCLN                       ;AND ABORT PASS
3958  020706  104444                                                                  TRAP    C$DCLN
3959  020710                          ENDIF         ;DIAG MUST BE RE-LOADED IN A CPU WITH LARGER MEMO
3960  020710                                                                50401$:
3961  020710                        ENDIF
3962  020710                                                                50400$:
3963
3964  020710                        LET CHGFLG :B= #0                ;CLR CHANGE CMD SEQ TBL FLAG.
3965  020710  105037  002212                                                          CLRB    CHGFLG
3966  020714                        LET R3 := #ENDFLG
3967  020714  012703  003452                                                          MOV     #ENDFLG,R3
3968  020720  004737  011136        JSR     PC,CLRERR                ;CLEAR ALL FLAGS.
3969  020724                        LET PWRFLG :B= #0                ;CLEAR THE POWER FAIL FLAG.
3970  020724  105037  003453                                                          CLRB    PWRFLG
3971
3972  020730                        EXIT    INIT
3973  020730  104432                                                                  TRAP    C$EXIT
3974  020732  000104                                                                  .WORD   L10012-.
3975  020734  040445  051106  042505  MEMOM:  .ASCII  /%AFREE MEMO TOO SMALL FOR RD-WR BFRS%N/
3976  020742  046440  046505  020117
3977  020750  047524  020117  046523
3978  020756  046101  020114  047506
3979  020764  020122  042122  053455
3980  020772  020122  043102  051522
3981  021000  047045
3982  021002  040445  042522  046055          .ASCIZ  /%ARE-LOAD IN LARGER MEMO%N/
3983  021010  040517  020104  047111
3984  021016  046040  051101  042507
3985  021024  020122  042515  047515
3986  021032  047045     000
3987          021036                  .EVEN
3988
3989  021036                          ENDINIT
3990  021036                  L10012:
3991  021036  104411                                                                  TRAP    C$INIT
```

```
                                        AUTO DROP SECTION

  3992                                  .SBTTL  AUTO DROP SECTION
  3993
  3994                                  ;++
  3995                                  ;SECTION EXECUTED AFTER THE INIT CODE WHEN 'ADR'' FLAG IS SET BY OPERATOR
  3996                                  ;SECTION CHEKS FOR A VALID INTERFACE LOCATION.  DROPS UNIT IF NO RESPONSE
  3997                                  ;FROM INTERFACE
  3998                                  ;--
  3999
  4000  021040                          BGNAUTO
  4001  021040                        L$AUTO::
  4002
  4003  021040  004737  015332                JSR PC,FIRSTU                   ;FIND FIRST UNIT
  4004  021044                                WHILE DEVTBL(R5) NE #END DO     ;
  4005  021044                                                                                50402$:
  4006  021044  026527  002532  177777                                                        CMP     DEVTBL(R5),#END
  4007  021052  001525                                                                       BEQ     50403$
  4008  021054                                  LET TRAPD4 :B= #0             ;
  4009  021054  105037  003454                                                              CLRB    TRAPD4
  4010  021060                                  SETVEC #4,#TRAP4,#PRIO7       ;SET VECTOR 4
  4011  021060  012746  000340                                                              MOV     #PRIO7,-(SP)
  4012  021064  012746  021454                                                              MOV     #TRAP4,-(SP)
  4013  021070  012746  000004                                                              MOV     #4,-(SP)
  4014  021074  012746  000003                                                              MOV     #3,-(SP)
  4015  021100  104437                                                                       TRAP    C$SVEC
  4016  021102  062706  000010                                                              ADD     #10,SP
  4017  021106                                  LET R2 := @TSSR(R5)           ;ADDRESS TS11 INTERFACE
  4018  021106  017502  002462                                                              MOV     @TSSR(R5),R2
  4019  021112                                  CLRVEC #4                     ;CLEAR VECTOR AT 4
  4020  021112  012700  000004                                                              MOV     #4,R0
  4021  021116  104436                                                                       TRAP    C$CVEC
  4022  021120                                  IFB TRAPD4 NE #0 THEN
  4023  021120  105737  003454                                                              TSTB    TRAPD4
  4024  021124  001423                                                                       BEQ     50404$
  4025  021126                                    LET FTLCNT(R5) := FTLCNT(R5) + #1
  4026  021126  005265  003314                                                              INC     FTLCNT(R5)
  4027  021132                                    PRINTF #AUTODM,TSSR(R5)     ;PRINT ERROR
  4028  021132  016546  002462                                                              MOV     TSSR(R5),-(SP)
  4029  021136  012746  021330                                                              MOV     #AUTODM,-(SP)
  4030  021142  012746  000002                                                              MOV     #2,-(SP)
  4031  021146  010600                                                                       MOV     SP,R0
  4032  021150  104417                                                                       TRAP    C$PNTF
  4033  021152  062706  000006                                                              ADD     #6,SP
  4034  021156                                    LET DROPN := DEVTBL(R5)             ;SAVE # OF UNIT TO BE DROPPED.
  4035  021156  016537  002532  015640                                                      MOV     DEVTBL(R5),DROPN
  4036  021164                                    LET R0 := R5 SHIFT -1               ;R0 LOGICAL DEVICE NUMBER
  4037  021164  010500                                                                       MOV     R5,R0
  4038  021166  006200                                                                       ASR     R0
  4039  021170                                    DODU R0                     ;DROP THE UNIT: EXEC BGNDU-ENDDU CODE IF IDU   0
  4040  021170  104451                                                                       TRAP    C$DODU
  4041  021172                                  ELSE
  4042  021172  000452                                                                       BR      50405$
  4043  021174                                                                      50404$:
  4044  021174                                    LET @TSDB(R5) :- #GSCPK     ;SEND GET STATUS COMMAND
  4045  021174  012775  002320  002452                                                      MOV     #GSCPK,@TSDB(R5)
  4046  021202  004737  011152                  JSR PC,WSSR                 ;WAIT
  4047  021206                                    IF #TS.SSR SETIN @TSSR(R5) THEN
```

```
 4048  021206  032775  000200  002462                                                    BIT     #TS.SSR,@TSSR(R5
 4049  021214  001423                                                                    BEQ     50406$
 4050  021216                                IF #TS.OFL SETIN @TSSR(R5) THEN
 4051  021216  032775  000100  002462                                                    BIT     #TS.OFL,@TSSR(R5
 4052  021224  001416                                                                    BEQ     50407$
 4053  021226                                  LET FTLCNT(R5) := FTLCNT(R5) + #1
 4054  021226  005265  003314                                                          INC     FTLCNT(R5
 4055  021232                                  PRINTF #OFLINM,DEVTBL(R5)
 4056  021232  016546  002532                                                          MOV     DEVTBL(R5),-(SP)
 4057  021236  012746  005127                                                          MOV     #OFLINM,-(SP)
 4058  021242  012746  000002                                                          MOV     #2,-(SP)
 4059  021246  010600                                                                  MOV     SP,R0
 4060  021250  104417                                                                  TRAP    C$PNTF
 4061  021252  062706  000006                                                          ADD     #6,SP
 4062  021256  004737  015554                                  JSR PC,DROPUA
 4063  021262                                  ENDIF
 4064  021262                                                                50407$:
 4065  021262                                ELSE
 4066  021262  000416                                                                  BR      50410$
 4067  021264                                                                50406$:
 4068  021264                                  LET FTLCNT(R5) := FTLCNT(R5) + #1
 4069  021264  005265  003314                                                          INC     FTLCNT(R5)
 4070  021270                                  PRINTF #NRDYM,DEVTBL(R5)
 4071  021270  016546  002532                                                          MOV     DEVTBL(R5),-(SP)
 4072  021274  012746  021424                                                          MOV     #NRDYM,-(SP)
 4073  021300  012746  000002                                                          MOV     #2,-(SP)
 4074  021304  010600                                                                  MOV     SP,R0
 4075  021306  104417                                                                  TRAP    C$PNTF
 4076  021310  062706  000006                                                          ADD     #6,SP
 4077  021314  004737  015554                                  JSR PC,DROPUA
 4078  021320                                  ENDIF
 4079  021320                                                                50410$:
 4080  021320                                ENDIF
 4081  021320                                                                50405$:
 4082  021320  004737  015400                                JSR PC,NEXTU
 4083  021324                                ENDDO
 4084  021324  000647                                                                  BR      50402$
 4085  021326                                                                50403$:
 4086
 4087  021326                                ENDAUTO
 4088  021326                          L10013:
 4089  021326  104461                                                                  TRAP    C$AUTO
 4090
 4091  021330  040445  052502  020123  AUTODM: .ASCII  /%ABUS TRAP AT %O6%N/
 4092  021336  051124  050101  040440
 4093  021344  020124  047445  022466
 4094  021352     116
 4095  021353     045  044501  052116          .ASCIZ  /%AINTERFACE BAD OR NOT SET TO ABOVE AD%N/
 4096  021360  051105  040506  042503
 4097  021366  041040  042101  047440
 4098  021374  020122  047516  020124
 4099  021402  042523  020124  047520
 4100  021410  040440  047502  042526
 4101  021416  040440  022504  000116
 4102  021424  040445  0471    052111  NRDYM:  .ASCIZ  /%AUNIT %D1%A NOT RDY%N/
 4103  021432  022440  030504  040445
```

```
 4104  021440  047040  052117  051040
 4105  021446  054504  047045     000
 4106          021454                          .EVEN
 4107
 4108                                  ;       DEVICE BUS TRAP HANDLER
 4109                                  ;       OUTPUT: TRAPD4 BYTE  1: TRAPED AT 4
 4110                                  ;                            0: NO TRAP
 4111
 4112  021454                          TRAP4:: LET TRAPD4 :B= TRAPD4 + #1
 4113  021454  105237  003454                                                    INCB    TRAPD4
 4114  021460  000002                          RTI
 4115
 4116
 4117
```

```
                                        .SBTTL  CLEANUP CODING SECTION

                                        ;++
                                        ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
                                        ; AT THE END OF EACH PASS.
                                        ;--

021462                                          BGNCLN
021462                                  L$CLEAN::

021462  004737  015332                          JSR     PC,FIRSTU               ;FIND FIRST UNIT.
021466                                          WHILE DEVTBL(R5) NE #END DO
021466                                                                                  50411$:
021466  026527  002532  177777                                                            CMP     DEVTBL(R5),#END
021474  001410                                                                          BEQ     50412$
021476  004737  011152                            JSR PC,WSSR                   ;WAIT FOR UNIT READY OR TIMEOUT,
021502                                            CLRVEC        TSVCT(R5)       ;RELEASE INTERRUPT VECTORS FOR ALL DEV.
021502  016500  002472                                                            MOV     TSVCT(R5),R0
021506  104436                                                                  TRAP    C$CVEC
021510  004737  015400                            JSR   PC,NEXTU                ;FIND NEXT UNIT.
021514                                          ENDDO
021514  000764                                                                  BR      50411$
021516                                                                                  50412$:
                                        EXIT    CLN
021516                                                                          TRAP    C$EXIT
021516  104432                                                                  .WORD   L10014-.
021520  000002
                                        .EVEN

021522                                  ENDCLN
021522                                  L10014:
021522  104412                                                                  TRAP    C$CLEAN
```

DROP UNIT SECTION

```
                                .SBTTL  DROP UNIT SECTION

                                ;++
                                ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
                                ; TO NO LONGER BE TESTED.  THAT CODE SHALL BE EXECUTED WHEN DODU
                                ;MACRO IS CALLED WHILE IDU FLAG IS NOT SET BY OPERATOR
                                ;--

021524                                  BGNDU
021524                          L$DU::

021524                                  LET R5 := R0 SHIFT 1            ;R5 = LOGICAL DEVICE NUMBER X 2.
021524  010005                                                                  MOV     R0,R5
021526  006305                                                                  ASL     R5
021530                                  LET DEVTBL(R5) := #NINUSE       ;SET NOT IN USE FLAG FOR THE DEVICE.
021530  012765  177774  002532                                                  MOV     #NINUSE,DEVTBL(R
021536                                  CLRVEC TSVCT(R5)                ;RELEASE THE INTERRUPT VECTOR.
021536  016500  002472                                                          MOV     TSVCT(R5),R0
021542  104436                                                                  TRAP    C$CVEC
021544                                  PRINTF  #DROPDM,DROPN           ;PRINT DROP DEVICE MESSAGE
021544  013746  01564C                                                          MOV     DROPN,-(SP)
021550  012746  004616                                                          MOV     #DROPDM,-(SP)
021554  012746  000002                                                          MOV     #2,-(SP)
021560  010600                                                                  MOV     SP,R0
021562  104417                                                                  TRAP    C$PNTF
021564  062706  000006                                                          ADD     #6,SP

021570                                  EXIT    DU
021570  000167                                                                  .WORD   J$JMP
021572  000000                                                                  .WORD   L10015
                                        .EVEN

021574                                  ENDDU
021574                          L10015:
021574  104453                                                                  TRAP    C$DU
```

```
                                        .SBTTL  ADD UNIT SECTION

                                        ;++
                                        ; THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
                                        ;  TO BE (A) TESTED FOR THE FIRST TIME, OR (B) RESUMED IN TESTING.  IF
                                        ; 'EF.AUNIT' IS SET, THE UNIT WILL BE TESTED AS A NEW UNIT.
                                        ;--

021576                                          BGNAU
021576                                  L$AU::

                                                LET R5 := R0 SHIFT 1            ;R5 = LOGICAL DEVICE NUMBER X 2.
021576  010005                                                                         MOV     R0,R5
021600  006305                                                                         ASL     R5
021602                                          LET DEVTBL(R5) := R0    ;STORE UNIT # IN DEVICE TABLE.
021602  010065  002532                                                                 MOV     R0,DEVTBL(R5)
021606                                          GPHARD  R0,R0                   ;GET HARDWARE P TABLE FROM SUPER.
021606  104442                                                                         TRAP    C$GPHRD
021610                                          LET TSSR(R5) := (R0)            ;SAVE TSSR ADDRESS.
021610  011065  002462                                                                 MOV     (R0),TSSR(R5)
021614                                          LET TSDB(R5) := (R0)+ - #2      ;SAVE TSDB ADDRESS.
021614  012065  002452                                                                 MOV     (R0)+,TSDB(R5)
021620  162765  000002  002452                                                         SUB     #2,TSDB(R5)
021626                                          LET TSVCT(R5) := (R0)           ;SAVE INTERRUPT VECTOR ADDRESS.
021626  011065  002472                                                                 MOV     (R0),TSVCT(R5)
021632                                          SETVEC TSVCT(R5),TS4INT(R5),#INTPRI     ;SET UP INTERUPT PROCESSING CONDITIONS.
021632  012746  000340                                                                 MOV     #INTPRI,-(SP)
021636  016546  002512                                                                 MOV     TS4INT(R5),-(SP)
021642  016546  002472                                                                 MOV     TSVCT(R5),-(SP)
021646  012746  000003                                                                 MOV     #3,-(SP)
021652  104437                                                                         TRAP    C$SVEC
021654  062706  000010                                                                 ADD     #10,SP
021660                                          LET INTFLG(R5) :- #0            ;CLEAR INTERRUPT FLAGS.
021660  005065  003416                                                                 CLR     INTFLG(R5)
021664                                          EXIT    AU
021664  000167                                                                         .WORD   J$JMP
021666  000000                                                                         .WORD   L10016-2-.

                                                .EVEN

021670                                          ENDAU
021670                                  L10016:
021670  104452                                                                         TRAP    C$AU

021672                                          ENDMOD
```

```
                                .TITLE HARDWARE TESTS

                                .SBTTL  TEST 1:  BASIC FUNCTIONS.

                                ;++
                                ; TEST TO EXECUTE ALL TS04 FUNCTIONS.
                                ;--

                                        BGNMOD

                                        BGNTST
                                T1::

                                        LET RANDOM :B- #0               ;CLR THE RANDOM OPERATIONS FLAG.
                                                                                CLRB    RANDOM
                                        LET EXPBOT :B- #0               ;CLR EXPECT BOT FLAG.
                                                                                CLRB    EXPBOT

                                        BGNSUB                          ;SUBTEST 1 - SET CHAR, DRIVE INIT, GET STATUS.
                                T1.1:
                                                                                TRAP    C$BSUB

                                        LET R2 := #BFSEQ0               ;ADR.OF CMD SEQ.
                                                                                MOV     #BFSEQ0,R2
                                        JSR     PC,BFSEQ                ;SET UP CMD SEQ.
                                        JSR     PC,EXALL                ;EXECUTE CMD SEQ ON ALL DEVICES.
                                        JSR     PC,FIRSTU               ;FIND THE FIRST UNIT.
                                        WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE MORE DEVICES:
                                                                                50413$:
                                                                                CMP     DEVTBL(R5),#END
                                                                                BEQ     50414$
  LET R2 := MSGPKA(R5)                  ;GET MSG PACKET ADR,
                                                                                MOV     MSGPKA(R5),R2
  LET R2 := R2 + #12                    ;GET XSTAT2 ADR,
                                                                                ADD     #12,R2
  LET TS4CL(R5) := (R2) CLR.BY #177400  ;STORE CODE LEVEL FROM DTR BYTE,
                                                                                MOV     (R2),TS4CL(R5)
                                                                                BIC     #177400,TS4CL(R5
  IF PASCNT(R5) EQ #1 THEN              ;IF THIS IS PASS 1 THEN:
                                                                                CMP     PASCNT(R5),#1
                                                                                BNE     50415$
    PRINTF #CODELM,DEVTBL(R5),TS4CL(R5) ;PRINT THE TS04 MICROCODE LEVEL.
                                                                                MOV     TS4CL(R5),-(SP)
                                                                                MOV     DEVTBL(R5),-(SP)
                                                                                MOV     #CODELM,-(SP)
                                                                                MOV     #3,-(SP)
                                                                                MOV     SP,R0
                                                                                TRAP    C$PNTF
                                                                                ADD     #10,SP
  ENDIF
                                                                        50415$:
  JSR   PC,NEXTU                        ;FIND NEXT UNIT.
ENDDO
                                                                                BR      50413$
                                                                        50414$:
```

```
4290  022024                                    ENDSUB
4291  022024                              L10020:
4292  022024  104403                                                    TRAP    C$FSUB
4293
4294  022026                                    BGNSUB                  ;SUBTEST 2 - REWIND.
4295  022026                              T1.2:
4296  022026  104402                                                    TRAP    C$BSUB
4297
4298  022030                                    LET R2 := #BFSEQ1       ;ADR OF CMD SEQ.
4299  022030  012702  022602                                           MOV     #BFSEQ1,R2
4300  022034  004737  022504                    JSR     PC,BFSEQ        ;SET UP CMD SEQ.
4301  022040  004737  006510                    JSR     PC,EXALL        ;EXECUTE CMD SEQ ON ALL DEVICES.
4302  022044                                    LET STAFLG :B= #0       ;CLEAR START FLAG
4303  022044  105037  003452                                           CLRB    STAFLG
4304  022050                                    ENDSUB
4305  022050                              L10021:
4306  022050  104403                                                    TRAP    C$ESUB
4307
4308  022052                                    BGNSUB                  ;SUBTEST 3 - WRITE/VERIFY.
4309  022052                              T1.3:
4310  022052  104402                                                    TRAP    C$BSUB
4311
4312  022054                                    LET R2 := #BFSEQ2       ;ADR OF CMD SEQ.
4313  022054  012702  022614                                           MOV     #BFSEQ2,R2
4314  022060  004737  022504                    JSR     PC,BFSEQ        ;SET UP CMD SEQ.
4315  022064  004737  006510                    JSR     PC,EXALL        ;EXECUTE CMD SEQ ON ALL DEVICES.
4316  022070                                    ENDSUB
4317  022070                              L10022:
4318  022070  104403                                                    TRAP    C$ESUB
4319
4320  022072                                    BGNSUB                  ;SUBTEST 4 - WRITE TAPE MARK, ERASE.
4321  022072                              T1.4:
4322  022072  104402                                                    TRAP    C$BSUB
4323
4324  022074                                    LET R2 := #BFSEQ3       ;ADR OF CMD SEQ.
4325  022074  012702  022706                                           MOV     #BFSEQ3,R2
4326  022100  004737  022504                    JSR     PC,BFSEQ        ;SET UP CMD SEQ.
4327  022104  004737  006510                    JSR     PC,EXALL        ;EXECUTE CMD SEQ ON ALL DEVICES.
4328  022110                                    ENDSUB
4329  022110                              L10023:
4330  022110  104403                                                    TRAP    C$ESUB
4331
4332  022112                                    BGNSUB                  ;SUBTEST 5 - SPACE FILES.
4333  022112                              T1.5:
4334  022112  104402                                                    TRAP    C$BSUB
4335
4336  022114                                    LET R2 := #BFSEQ4       ;ADR OF CMD SEQ.
4337  022114  012702  022760                                           MOV     #BFSEQ4,R2
4338  022120  004737  022504                    JSR     PC,BFSEQ        ;SET UP CMD SEQ.
4339  022124  004737  006510                    JSR     PC,EXALL        ;EXECUTE CMD SEQ ON ALL DEVICES.
4340  022130                                    ENDSUB
4341  022130                              L10024:
4342  022130  104403                                                    TRAP    C$ESUB
4343
4344  022132                                    BGNSUB                  ;SUBTEST 6 - SPACE RECORDS.
4345  022132                              T1.6:
```

```
4346  022132  104402                                                          TRAP    C$BSUB
4347
4348  022134                          LET R2 := #BFSEQ5            ;ADR OF CMD SEQ.
4349  022134  012702  023022                                                  MOV     #BFSEQ5,R2
4350  022140  004737  022504          JSR     PC,BFSEQ             ;SET UP CMD SEQ.
4351  022144  004737  006510          JSR     PC,EXALL             ;EXECUTE CMD SEQ ON ALL DEVICES.
4352  022150                          ENDSUB
4353  022150                  L10025:
4354  022150  104403                                                          TRAP    C$ESUB
4355
4356  022152                          BGNSUB                       ;SUBTEST 7 - WRITE RETRY.
4357  022152                  T1.7:
4358  022152  104402                                                          TRAP    C$BSUB
4359
4360  022154                          LET R2 := #BFSEQ6            ;ADR OF CMD SEQ.
4361  022154  012702  023074                                                  MOV     #BFSEQ6,R2
4362  022160  004737  022504          JSR     PC,BFSEQ             ;SET UP CMD SEQ.
4363  022164  004737  006510          JSR     PC,EXALL             ;EXECUTE CMD SEQ ON ALL DEVICES.
4364  022170                          ENDSUB
4365  022170                  L10026:
4366  022170  104403                                                          TRAP    C$ESUB
4367
4368  022172                          BGNSUB                       ;SUBTEST 8 - READ REV RETRY.
4369  022172                  T1.8:
4370  022172  104402                                                          TRAP    C$BSUB
4371
4372  022174                          LET R2 := #BFSEQ7            ;ADR OF CMD SEQ.
4373  022174  012702  023126                                                  MOV     #BFSEQ7,R2
4374  022200  004737  022504          JSR     PC,BFSEQ             ;SET UP CMD SEQ.
4375  022204  004737  006510          JSR     PC,EXALL             ;EXECUTE CMD SEQ ON ALL DEVICES.
4376  022210                          ENDSUB
4377  022210                  L10027:
4378  022210  104403                                                          TRAP    C$ESUB
4379
4380  022212                          BGNSUB                       ;SUBTEST 9 - READ FWD RETRY.
4381  022212                  T1.9:
4382  022212  104402                                                          TRAP    C$BSUB
4383
4384  022214                          LET R2 := #BFSEQ8            ;ADR OF CMD SEQ.
4385  022214  012702  023160                                                  MOV     #BFSEQ8,R2
4386  022220  004737  022504          JSR     PC,BFSEQ             ;SET UP CMD SEQ.
4387  022224  004737  006510          JSR     PC,EXALL             ;EXECUTE CMD SEQ ON ALL DEVICES.
4388  022230                          ENDSUB
4389  022230                  L10030:
4390  022230  104403                                                          TRAP    C$ESUB
4391
4392  022232                          BGNSUB                       ;SUBTEST 10- CLEAN.
4393  022232                  T1.10:
4394  022232  104402                                                          TRAP    C$BSUB
4395
4396  022234                          LET R2 := #BFSEQ9            ;ADR OF CMD SEQ.
4397  022234  012702  023212                                                  MOV     #BFSEQ9,R2
4398  022240  004737  022504          JSR     PC,BFSEQ             ;SET UP CMD SEQ.
4399  022244  004737  006510          JSR     PC,EXALL             ;EXECUTE CMD SEQ ON ALL DEVICES.
4400  022250                          ENDSUB
4401  022250                  L10031:
```

```
                        TEST 1:  BASIC FUNCTIONS.

4402  022250  104403                                                    TRAP    C$ESUB
4403
4404  022252                        BGNSUB                              ;SUBTEST 11 - WTV SWAPPED DATA BYTES.
4405  022252                T1.11:
4406  022252  104402                                                    TRAP    C$BSUB
4407
4408  022254                        LET R2 := #BFSE10                   ;ADR OF CMD SEQ.
4409  022254  012702  023234                                           MOV     #BFSE10,R2
4410  022260  004737  022504        JSR     PC,BFSEQ                    ;SET UP CMD SEQ.
4411  022264  004737  006510        JSR     PC,EXALL                    ;WRITE/VERIFY RECORDS 1 AND 2.
4412  022270                        LET SWBFLG :B= #1                   ;ENABLE BYTE SWAPPING.
4413  022270  112737  000001  003444                                    MOVB    #1,SWBFLG
4414  022276  004737  006510        JSR     PC,EXALL                    ;WRITE/VERIFY RECORDS 3 AND 4.
4415  022302                        LET SWBFLG :B= #0                   ;DISABLE BYTE SWAPPING.
4416  022302  105037  003444                                           CLRB    SWBFLG
4417  022306                        ENDSUB
4418  022306                L10032:
4419  022306  104403                                                    TRAP    C$ESUB
4420
4421  022310                        LET R2 := DATAWT + #10.             ;INIT WRITE BUFFER POINTER.
4422  022310  013702  003334                                           MOV     DATAWT,R2
4423  022314  062702  000012                                           ADD     #10.,R2
4424  022320                        WHILE R2 NE DATAWT DO   ;UNTIL 10 BYTES HAVE BEEN SWAPPED.
4425  022320                                                      50416$:
4426  022320  020237  003334                                           CMP     R2,DATAWT
4427  022324  001402                                                    BEQ     50417$
4428  022326  000342                          SWAB -(R2)                ;SWAP DATA BYTES IN WRITE BUFFER.
4429  022330                        ENDDO
4430  022330  000773                                                    BR      50416$
4431  022332                                                      50417$:
4432  022332                        LET T1SWB :B= T1SWB + #1            ;SET T1 SWAP BYTES FLAG FOR ''CKDATA'' SUBR
4433  022332  105237  003447                                           INCB    T1SWB
4434
4435  022336                        BGNSUB                              ;SUBTEST 12 - READ SWAPPED DATA BYTES.
4436  022336                T1.12:
4437  022336  104402                                                    TRAP    C$BSUB
4438
4439  022340                        LET CMDWRD := #RDR                  ;CMD IS READ REV.
4440  022340  012737  104401  003346                                    MOV     #RDR,CMDWRD
4441  022346  004737  014402        JSR     PC,VFEXC                    ;VERIFY ODD LENGTH SWAP (RECORD 4).
4442  022352                        LET CMDPKT+CP.CNT := #12            ;CHANGE BYTE COUNT TO 10.
4443  022352  012737  000012  002316                                    MOV     #12,CMDPKT+CP.CN
4444  022360  004737  014402        JSR     PC,VFEXC                    ;VERIFY EVEN LENGTH SWAP (RECORD 3).
4445  022364                        LET SWBFLG :B= #1                   ;ENABLE BYTE SWAPPING.
4446  022364  112737  000001  003444                                    MOVB    #1,SWBFLG
4447  022372                        LET CMDPKT+CP.CNT := #11                ;CHANGE BYTE COUNT TO 9.
4448  022372  012737  000011  002316                                    MOV     #11,CMDPKT+CP.CN
4449  022400  004737  014402        JSR     PC,VFEXC                    ;VERIFY ODD LENGTH SWAP (RECORD 2).
4450  022404                        LET CMDPKT+CP.CNT := #12            ;CHANGE BYTE COUNT TO 10.
4451  022404  012737  000012  002316                                    MOV     #12,CMDPKT+CP.CN
4452  022412  004737  014402        JSR     PC,VFEXC                    ;VERIFY EVEN LENGTH SWAP (RECORD 1).
4453  022416                        LET CMDWRD := #RDF                  ;CMD IS READ FWD.
4454  022416  012737  104001  003346                                    MOV     #RDF,CMDWRD
4455  022424  004737  014402        JSR     PC,VFEXC                    ;VERIFY EVEN LENGTH SWAP (RECORD 1).
4456  022430                        LET CMDPKT+CP.CNT := #11                ;CHANGE BYTE COUNT TO 9.
4457  022430  012737  000011  002316                                    MOV     #11,CMDPKT+CP.CN
```

```
4458  022436  004737  014402                  JSR     PC,VFEXC                ;VERIFY ODD LENGTH SWAP (RECORD 2).
4459  022442                                  LET SWBFLG :B= #0               ;DISABLE BYTE SWAPPING.
4460  022442  105037  003444                                                  CLRB    SWBFLG
4461  022446                                  LET CMDPKT+CP.CNT := #12        ;CHANGE BYTE COUNT TO 10.
4462  022446  012737  000012  002316                                          MOV     #12,CMDPKT+CP.CN
4463  022454  004737  014402                  JSR     PC,VFEXC                ;VERIFY EVEN LENGTH SWAP (RECORD 3).
4464  022460                                  LET CMDPKT+CP.CNT := #11        ;CHANGE BYTE COUNT TO 9.
4465  022460  012737  000011  002316                                          MOV     #11,CMDPKT+CP.CN
4466  022466  004737  014402                  JSR     PC,VFEXC                ;VERIFY ODD LENGTH SWAP (RECORD 4).
4467
4468  022472                                  ENDSUB
4469  022472                          L10033:
4470  022472  104403                                                          TRAP    CSESUB
4471
4472  022474                                  LET T1SWB :B= #0                ;CLEAR T1 SWAP BYTES FLAG
4473  022474  105037  003447                                                  CLRB    T1SWB
4474
4475
4476  022500                                  EXIT    TST
4477  022500  104432                                                          TRAP    CSEXIT
4478  022502  000554                                                          .WORD   L10017-.
```

```
                                ;       SUBROUTINE TO MOVE A COMMAND SEQUENCE TO THE SEQUENCE TABLE.
                                ;       INPUTS:         R2 = FWA OF COMMAND SEQUENCE.
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:

022504                          BFSEQ:  LET R1 := #CMDSEQ               ;INIT SEQ TABLE ADDRESS.
022504  012701  003460                                                          MOV     #CMDSEQ,R1
022510                                  WHILE (R2) NE #END DO           ;WHILE THERE ARE MORE COMMANDS:
022510                                                                  50420$:
022510  021227  177777                                                          CMP     (R2),#END
022514  001402                                                                  BEQ     50421$
022516                                    LET (R1)+ := (R2)+            ;MOVE COMMANDS TO SEQ TABLE.
022516  012221                                                                  MOV     (R2)+,(R1)+
022520                                  ENDDO
022520  000773                                                                  BR      50420$
022522                                                                  50421$:
022522                                  LET (R1) := #END                ;STORE END OF SEQUENCE CODE.
022522  012711  177777                                                          MOV     #END,(R1)
022526  000207                          RTS     PC                      ;RETURN.


                                ;       BASIC FUNCTION COMMAND SEQUENCE

022530  140004                  BFSEQ0: .WORD   SCH                     ;SET CHAR. 200.         (1)
022532  000200                                  200
022534  000001                                  1
022536  000000                                  0
022540  100013                                  DRI                     ;DRIVE INIT.            (2)
022542  000001                                  1
022544  000001                                  1
022546  000000                                  0
022550  140004                                  SCH                     ;SET CHAR. 20           (3)
022552  000020                                  20
022554  000001                                  1
022556  000000                                  0
022560  100017                                  GES                     ;GET STATUS.            (4)
022562  000001                                  1
022564  000001                                  1
022566  000000                                  0
022570  140004                                  SCH                     ;SET CHAR. 40.          (5)
022572  000040                                  40
022574  000001                                  1
022576  000000                                  0
022600  177777                  .WORD   END

022602  102010                  BFSEQ1:         RWD                     ;REWIND TWICE.          (6)
022604  000001                                  1
022606  000002                                  2
022610  000000                                  0
022612  177777                  .WORD   END

022614  104105                  BFSEQ2:         WTV                     ;WRITE/VERIFY PAT 1.    (7)
022616  004000                                  DATCNT
022620  000001                                  1
```

```
4535  022622  000001                          1
4536  022624  104105                          WTV             ;WTV PAT 2.          (8)
4537  022626  004000                          DATCNT
4538  022630  000001                          1
4539  022632  000002                          2
4540  022634  104105                          WTV             ;WTV PAT 3.          (9)
4541  022636  004000                          DATCNT
4542  022640  000001                          1
4543  022642  000003                          3
4544  022644  104105                          WTV             ;WTV PAT 4.          (10)
4545  022646  004000                          DATCNT
4546  022650  000001                          1
4547  022652  000004                          4
4548  022654  104105                          WTV             ;WTV PAT 5.          (11)
4549  022656  004000                          DATCNT
4550  022660  000001                          1
4551  022662  000005                          5
4552  022664  104105                          WTV             ;WTV PAT 6.          (12)
4553  022666  004000                          DATCNT
4554  022670  000001                          1
4555  022672  000006                          6
4556  022674  104105                          WTV             ;WTV PAT 0.          (13)
4557  022676  004000                          DATCNT
4558  022700  000001                          1
4559  022702  000000                          0
4560  022704  177777              .WORD       END
4561
4562  022706  100011          BFSEQ3:         WTM             ;WRITE TAPE MARK.    (14)
4563  022710  000001                          1
4564  022712  000001                          1
4565  022714  000000                          0
4566  022716  104005                          WRT             ;WRITE 10 RECORDS.   (15)
4567  022720  004000                          DATCNT
4568  022722  000010                          10
4569  022724  000001                          1
4570  022726  100411                          ERS             ;ERASE 10 TIMES.     (16)
4571  022730  000001                          1
4572  022732  000010                          10
4573  022734  000000                          0
4574  022736  100011                          WTM             ;WRITE TAPE MARK.    (17)
4575  022740  000001                          1
4576  022742  000001                          1
4577  022744  000000                          0
4578  022746  101011                          WTR             ;WTM RETRY           (18)
4579  022750  000001                          1
4580  022752  000001                          1
4581  022754  000000                          0
4582  022756  177777              .WORD       END
4583
4584  022760  105410          BFSEQ4:         SFR             ;SPACE 2 FILES REV.  (19)
4585  022762  000002                          2
4586  022764  000001                          1
4587  022766  000000                          0
4588  022770  105010                          SFF             ;SPACE 2 FILES FWD.  (20)
4589  022772  000002                          2
4590  022774  000001                          1
```

```
4591  022776  000000                              0
4592  023000  105410                              SFR             ;SPACE 2 FILES REV.     (21)
4593  023002  000001                              1
4594  023004  000002                              2
4595  023006  000000                              0
4596  023010  105010                              SFF             ;SPACE 2 FILES FWD.     (22)
4597  023012  000001                              1
4598  023014  000002                              2
4599  023016  000000                              0
4600  023020  177777                      .WORD   END
4601
4602  023022  102010          BFSEQ5:             RWD             ;REWIND.                (23)
4603  023024  000001                              1
4604  023026  000001                              1
4605  023030  000000                              0
4606  023032  104010                              SRF             ;SPACE 7 RECORDS FWD.   (24)
4607  023034  000007                              7
4608  023036  000001                              1
4609  023040  000000                              0
4610  023042  104410                              SRR             ;SPACE 7 RECORDS REV.   (25)
4611  023044  000007                              7
4612  023046  000001                              1
4613  023050  000000                              0
4614  023052  104010                              SRF             ;SPACE 7 RECORDS FWD.   (26)
4615  023054  000001                              1
4616  023056  000007                              7
4617  023060  000000                              0
4618  023062  104410                              SRR             ;SPACE 7 RECORDS REV.   (27)
4619  023064  000001                              1
4620  023066  000007                              7
4621  023070  000000                              0
4622  023072  177777                      .WORD   END
4623
4624  023074  102010          BFSEQ6:             RWD             ;REWIND.                (28)
4625  023076  000001                              1
4626  023100  000001                              1
4627  023102  000000                              0
4628  023104  104005                              WRT             ;WRITE.                 (29)
4629  023106  004000                              DATCNT
4630  023110  000001                              1
4631  023112  000001                              1
4632  023114  105005                              WRR             ;WRITE RETRY.           (30)
4633  023116  004000                              DATCNT
4634  023120  000001                              1
4635  023122  000001                              1
4636  023124  177777                      .WORD   END
4637
4638  023126  104401          BFSEQ7:             RDR             ;READ REV.              (31)
4639  023130  004000                              DATCNT
4640  023132  000001                              1
4641  023134  000001                              1
4642  023136  105401                              RNR             ;READ NEXT REV.         (32)
4643  023140  004000                              DATCNT
4644  023142  000001                              1
4645  023144  000001                              1
4646  023146  125401                              RNF             ;READ NEXT FWD.         (33)
```

```
4647  023150  004000                          DATCNT
4648  023152  000001                          1
4649  023154  000001                          1
4650  023156  177777                  .WORD   END
4651
4652  023160  104001          BFSEQ8:         RDF             ;READ FWD.              (34)
4653  023162  004000                          DATCNT
4654  023164  000001                          1
4655  023166  000001                          1
4656  023170  105001                          RPF             ;READ PREVIOUS FWD.     (35)
4657  023172  004000                          DATCNT
4658  023174  000001                          1
4659  023176  000001                          1
4660  023200  125001                          RPR             ;READ PREVIOUS REV.     (36)
4661  023202  004000                          DATCNT
4662  023204  000001                          1
4663  023206  000001                          1
4664  023210  177777                  .WORD   END
4665
4666  023212  101012          BFSEQ9: .WORD   CLN             ;CLEAN.                 (37)
4667  023214  000001                          1
4668  023216  000001                          1
4669  023220  000000                          0
4670  023222  102010                          RWD             ;REWIND                 (38)
4671  023224  000001                          1
4672  023226  000001                          1
4673  023230  000000                          0
4674  023232  177777                  .WORD   END             ;END OF SEQUENCE.
4675
4676  023234  104105          BFSE10:         WTV             ;WRITE/VERIFY EVEN LENGTH.      (39)
4677  023236  000012                          12
4678  023240  000001                          1
4679  023242  000000                          0
4680  023244  104105                          WTV             ;WRITE/VERIFY ODD LENGTH.       (40)
4681  023246  000011                          11
4682  023250  000001                          1
4683  023252  000000                          0
4684  023254  177777                  .WORD   END
4685                                  .EVEN
4686
4687  023256                          ENDTST
4688  023256          L10^17:
4689  023256  104401                                          TRAP    C$ETST
```

```
                                        TEST 2: DATA RELIABILITY.

 4690                                   .SBTTL  TEST 2: DATA RELIABILITY.
 4691
 4692                                   ;++
 4693                                   ; TEST TO CHECK THE DATA RELIABILITY OF THE TS04.
 4694                                   ;--
 4695  023260                                   BGNTST
 4696  023260                           T2::
 4697
 4698  023260                                   LET RANDOM :B  #1              ;SET THE RANDOM OPERATIONS FLAG.
 4699  023260  112737  000001  003441                                                  MOVB    #1,RANDOM
 4700  023266                                   LET EXPBOT :B= #0              ;CLEAR EXPECT BOT FLAG.
 4701  023266  105037  003440                                                  CLRB    EXPBOT
 4702  023272                                   LET R2 := #DATCNT - #1         ;SET UP THE RECORD LENGTH MASK.
 4703  023272  012702  004000                                                  MOV     #DATCNT,R2
 4704  023276  005302                                                          DEC     R2
 4705  023300                                   LET LENMSK :  COMP R2          ;ALLOW MAXIMUM BUFFER.
 4706  023300  010237  003356                                                  MOV     R2,LENMSK
 4707  023304  005137  003356                                                  COM     LENMSK
 4708  023310  004737  006444                 JSR PC,SETCH                   ;CMD 1 - SET CHARACTERISTIC.
 4709  023314                                   IFB STAFLG NE #0 THEN          ;IF STARTING THEN:
 4710  023314  105737  003452                                                  TSTB    STAFLG
 4711  023320  001404                                                          BEQ     50422$
 4712  023322  004737  006470                     JSR   PC,SETRW             ;CMD2=REWIND
 4713  023326                                     LET STAFLG :B- #0            ;C_R START FLAG.
 4714  023326  105037  003452                                                  CLRB    STAFLG
 4715  023332                                   ENDIF
 4716  023332                                                          50422$:
 4717  023332                                   LET (R1)+ := #WTV              ;CMD3 = WRITE/ VERIFY.
 4718  023332  012721  104105                                                  MOV     #WTV,(R1)+
 4719  023336                                   LET (R1)+ := #DATCNT           ;SET BRF TO MAX FOR PATTERN GENERATION.
 4720  023336  012721  004000                                                  MOV     #DATCNT,(R1)+
 4721  023342                                   LET R2 :  COMP #RNOPSC
 4722  023342  012702  177740                                                  MOV     #RNOPSC,R2
 4723  023346  005102                                                          COM     R2
 4724  023350                                   LET (R1)+ :- R2                ;31 OPERATIONS.
 4725  023350  010221                                                          MOV     R2,(R1)+
 4726  023352                                   LET (R1)+ := #RANP             ;RANDOM PATTERN.
 4727  023352  012721  000007                                                  MOV     #RANP,(R1)+
 4728  023356                                   REPEAT                         ;REPEAT TO EOT:
 4729  023356                                                          50423$:
 4730  023356                                     WHILE R1 LT #SEQEND DO       ;FILL SEQ TBL WITH RANDOM CMDS.
 4731  023356                                                          50424$:
 4732  023356  020127  003550                                                  CMP     R1,#SEQEND
 4733  023362  002012                                                          BGE     50425$
 4734  023364
 4735  023364  063737  003360  003362             LET RANS := RANS + RANB
                                                                              ADD     RANB,RANS
 4736  023372                                     LET R2 := RANS CLR.BY #177741  ;R2 - RANDOM # (0 - 36).
 4737  023372  013702  003362                                                  MOV     RANS,R2
 4738  023376  042702  177741                                                  BIC     #177741,R2
 4739  023402  004772  023540                     JSR PC,@RANCMD(R2)         ;SET UP A RANDOM CMD + BRF.
 4740  023406                                     ENDDO
 4741  023406  000763                                                          BR      50424$
 4742  023410                                                          50425$:
 4743  023410                                   LET (R1) := #END               ;STORE END OF SEQUENCE CODE IN TABLE.
 4744  023410  012711  177777                                                  MOV     #END,(R1)
 4745  023414  004737  006510                   JSR   PC,EXALL                 ;GO EXECUTE ALL CMDS IN SEQUENCE TABLE.
```

```
023420                              LET R1 := #CMDSEQ                ;INIT CMD SEQ TBL POINTER,
023420  012701  003460                                                                 MOV     #CMDSEQ,R1
023424                              UNTIL R2 NE #0                   ;REPEAT UNTIL EOT IS REACHED
023424  005702                                                                         TST     R2
023426  001753                                                                         BEQ     50423$
023430                              LET ALLEOT :B= ALLEOT + #1       ;FLAG ALL UNITS @ EOT
023430  105237  003450                                                                 INCB    ALLEOT
023434  000240                      NOP
023436  000240                      NOP
023440  000240                      NOP
023442  004737  024764              JSR     PC,T5WE01                ;WRITE ONE RECORD BEYOND EOT ON ALL UNITS
                                                                     ;SO THAT SHORTER READ STOP DISTANCE
                                                                     ;SHALL POSITION HEAD IN CLEAN IRG GAP
                                                                     ;READ REV THAT EXTRA REC TO RE-POSITION THE TAPE
023446  004737  023600              JSR     PC,RANRD                 ;SET UP READ REV/FWD CMDS,
023452                              LET CMDSEQ+4 :- COMP #RNOPSC     ;# OF RECORDS FOR READ REV.
023452  012737  177740  003464                                                         MOV     #RNOPSC,CMDSEQ+4
023460  005137  003464                                                                 COM     CMDSEQ+4
023464                              LET CMDSEQ+14 := CMDSEQ+4        ;# OF RECORDS FOR READ FORWARD.
023464  013737  003464  003474                                                         MOV     CMDSEQ+4,CMDSEQ+
023472                              LET (R1) :  #END                 ;STORE END OF SEQUENSE CODE IN SEQ TABLE.
023472  012711  177777                                                                 MOV     #END,(R1)
023476  004737  006510              JSR     PC,EXALL                 ;GO EXECUTE READ REV/FWD OF LAST N RECORDS.
023502                              LET ALLEOT :B= #0                ;CLEAR ALL UNITS @ EOT FLAG
023502  105037  003450                                                                 CLRB    ALLEOT
023506                              LET RPTFLG :B= #1                ;REQUEST PERFORMANCE REPORT DURING REWIND.
023506  112737  000001  003443                                                         MOVB    #1,RPTFLG
023514                              LET R1 := #CMDSEQ                ;INIT SEQ TBL POINTER,
023514  012701  003460                                                                 MOV     #CMDSEQ,R1
023520  004737  006470              JSR PC,SETRW                     ;STORE REWIND IN SEQ TBL,
023524                              LET (R1) := #END                 ;STORE END IN SEQ TBL,
023524  012711  177777                                                                 MOV     #END,(R1)
023530  004737  006510              JSR PC,EXALL                     ;EXECUTE REWIND CMD ON ALL UNITS

023534                              EXIT    TST
023534  104432                                                                         TRAP    C$EXIT
023536  000174                                                                         .WORD   L10034-.

```

```
4784                                    ;       ADDRESSES OF SUBROUTINES USED TO SET UP RANDOM OPERATIONS IN
4785                                    ;       THE DATA RELIABILITY TEST.
4786
4787  023540  023666                    RANCMD: RANWV                   ;WRITE/VERIFY.
4788  023542  023654                            RANWR                   ;WRITE.
4789  023544  023654                            RANWR                   ;WRITE.
4790  023546  023654                            RANWR                   ;WRITE.
4791  023550  023654                            RANWR                   ;WRITE.
4792  023552  023654                            RANWR                   ;WRITE.
4793  023554  023654                            RANWR                   ;WRITE.
4794  023556  023654                            RANWR                   ;WRITE.
4795  023560  023600                            RANRD                   ;READ.
4796  023562  023600                            RANRD                   ;READ.
4797  023564  023600                            RANRD                   ;READ.
4798  023566  023600                            RANRD                   ;READ.
4799  023570  023600                            RANRD                   ;READ.
4800  023572  023600                            RANRD                   ;READ.
4801  023574  023600                            RANRD                   ;READ.
4802  023576  023600                            RANRD                   ;READ.
4803
4804
4805
4806
4807
4808                                    ;       SUBROUTINE TO SET UP READ COMMANDS IN SEQUENCE TABLE.
4809                                    ;       INPUTS:
4810                                    ;       OUTPUTS: .
4811                                    ;       REGISTERS:      R2
4812                                    ;       CALLS:
4813
4814  023600                            RANRD:  LET (R1)+ := #RDR               ;STORE READ REV CMD.
4815  023600  012721  104401                                                            MOV     #RDR,(R1)+
4816  023604                                    LET (R1)+ := #DATCNT          ;SET BRF TO MAX FOR READ RANDOM LENGTHS.
4817  023604  012721  004000                                                            MOV     #DATCNT,(R1)+
4818  023610                                    LET RANB := RANB + RANS
4819  023610  063737  003362  003360                                                    ADD     RANS,RANB
4820  023616                                    LET R2 := RANB CLR.BY #RNOPSC
4821  023616  013702  003360                                                            MOV     RANB,R2
4822  023622  042702  177740                                                            BIC     #RNOPSC,R2
4823  023626                                    LET (R1)+ := R2               ;SET RANDOM # OF OPERATIONS.
4824  023626  010221                                                            MOV     R2,(R1)+
4825  023630                                    LET (R1)+ := #RANP            ;RANDOM PATTERN.
4826  023630  012721  000007                                                            MOV     #RANP,(R1)+
4827  023634                                    LET (R1)+ := #RDF             ;STORE READ FWD CMD.
4828  023634  012721  104001                                                            MOV     #RDF,(R1)+
4829  023640                                    LET (R1)+ := #DATCNT          ;SET BRF TO MAX TO READ RANDOM LENGTHS.
4830  023640  012721  004000                                                            MOV     #DATCNT,(R1)+
4831  023644                                    LET (R1)+ := R2               ;SET RANDOM # OF OPERATIONS.
4832  023644  010221                                                            MOV     R2,(R1)+
4833  023646                                    LET (R1)+ := #RANP            ;RANDOM PATTERN.
4834  023646  012721  000007                                                            MOV     #RANP,(R1)+
4835  023652  000207                           RTS PC
```

```
4836                                    ;       SUBROUTINE TO SET UP A WRITE COMMAND IN THE SEQUENCE TABLE.
4837                                    ;       INPUTS:
4838                                    ;       OUTPUTS:
4839                                    ;       REGISTERS:
4840                                    ;       CALLS:
4841
4842  023654                            RANWR:  LET (R1)+ := #WRT               ;STORE WRITE CMD.
4843  023654  012721  104005                                                            MOV     #WRT,(R1)+
4844  023660  004737  023700                   JSR PC,RANW                     ;STORE BRF, # OF OPERATIONS, PATTERN.
4845  023664  000207                           RTS PC
4846
4847
4848
4849
4850
4851                                    ;       SUBROUTINE TO SET UP A WRITE/VERIFY COMMAND IN THE SEQUENCE TABLE.
4852                                    ;       INPUTS:
4853                                    ;       OUTPUTS:
4854                                    ;       REGISTERS:
4855                                    ;       CALLS:
4856
4857  023666                            RANWV:  LET (R1)+ := #WTV               ;STORE WRITE/VERIFY CMD.
4858  023666  012721  104105                                                            MOV     #WTV,(R1)+
4859  023672  004737  023700                   JSR PC,RANW                     ;STORE BRF, # OF OPERATIONS, PATTERN.
4860  023676  000207                           RTS     PC
4861
4862
4863
4864
4865
4866                                    ;       SUBROUTINE TO STORE BRF, # OF OPERATIONS, PATTERN IN COMMAND
4867                                    ;       SEQUENCE TABLE FOR WRITE AND WRITE/VERIFY COMMANDS.
4868                                    ;       INPUTS:
4869                                    ;       OUTPUTS:
4870                                    ;       REGISTERS:      R2
4871                                    ;       CALLS:
4872
4873  023700                            RANW:   LET (R1)+ := #DATCNT            ;SET BRF TO MAX FOR PATTERN GENERATION.
4874  023700  012721  004000                                                            MOV     #DATCNT,(R1)+
4875                                                                            ;RANDOM BRF WILL BE GENERATED FOR EACH RECORD.
4876  023704                                  LET RANB :  RANB + RANS
4877  023704  063737  003362  003360                                                    ADD     RANS,RANB
4878  023712                                  LET R2 := RANB CLR.BY #RNOPSC
4879  023712  013702  003360                                                            MOV     RANB,R2
4880  023716  042702  177740                                                            BIC     #RNOPSC,R2
4881  023722                                  LET (R1)+ := R2                 ;SET RANDOM # OF OPERATIONS.
4882  023722  010221                                                                    MOV     R2,(R1)+
4883  023724                                  LET (R1)+ :- #RANP              ;RANDOM PATTERN.
4884  023724  012721  000007                                                            MOV     #RANP,(R1)+
4885  023730  000207                           RTS PC                          ;RETURN.
4886
4887                                          .EVEN
4888
4889  023732                                  ENDTST
4890  023732                          L10034:
4891  023732  104401                                                            TRAP    C$ETST
```

4892

```
                                        TEST 3: WRITE COMPATABILITY/WRITE UTILITY.

                                        .SBTTL  TEST 3: WRITE COMPATABILITY/WRITE UTILITY.

                                        ;++
                                        ; TEST TO WRITE RECORDS FROM BOT TO EOT.
                                        ;--

                                                BGNTST
                                        T3::

                                                LET RANDOM :B= #1               ;SET THE RANDOM OPERATIONS FLAG.
023734  112737  000001  003441                                                          MOVB    #1,RANDOM
                                                LET EXPBOT :B- #0               ;CLEAR EXPECT BOT FLAG.
023742  105037  003440                                                                 CLRB    EXPBOT
                                                LET R2 := #DATCNT - #1          ;SET UP THE RECORD LENGTH MASK.
023746  012702  004000                                                                 MOV     #DATCNT,R2
023752  005302                                                                         DEC     R2
                                                LET LENMSK := COMP R2           ;ALLOW MAXIMUM BUFFER.
023754  010237  003356                                                                 MOV     R2,LENMSK
023760  005137  003356                                                                 COM     LENMSK
                                                JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
023764  004737  006444
                                                JSR PC,SETRW                    ;CMD2=REWIND
023770  004737  006470
                                                LET STAFLG :B- #0               ;CLEAR START FLAG
023774  105037  003452                                                                 CLRB    STAFLG
                                                REPEAT                          ;REPEAT TO EOT.
024000                                                                                  50426$:
                                                  WHILE R1 LT #SEQEND DO        ;WHILE THERE IS MORE ROOM IN SEQ TABLE:
024000                                                                                  50427$:
024000  020127  003550                                                                 CMP     R1,#SEQEND
024004  002003                                                                         BGE     50430$
                                                    JSR PC,RANWR                  ;STORE A WRITE CMD IN SEQUENCE TABLE.
024006  004737  023654
                                                  ENDDO
024012                                                                                  BR      50427$
024012  000772                                                                  50430$:
                                                  LET (R1) := #END              ;STORE END OF SEQUENCE CODE IN TABLE.
024014                                                                                  MOV     #END,(R1)
024014  012711  177777
                                                  JSR   PC,EXALL                ;EXECUTE ALL CMDS IN SEQ TBL ON UNITS.
024020  004737  006510
                                                  LET R1 := #CMDSEQ             ;INIT SEQ TBL POINTER,
024024                                                                                  MOV     #CMDSEQ,R1
024024  012701  003460
                                                UNTIL R2 NE #0                  ;REPEAT UNTIL EOT IS REACHED
024030                                                                                  TST     R2
024030  005702                                                                         BEQ     50426$
024032  001762
                                                LET ALLEOT :B- ALLEOT + #1      ;SET ALL UNITS @ EOT FLAG
024034                                                                                  INCB    ALLEOT
024034  105237  003450
                                                NOP
024040  000240
                                                NOP
024042  000240
                                                NOP
024044  000240
                                                JSR     PC,T5WEOT               ;WRITE ONE RECORD BEYOND EOT ON ALL UNITS
024046  004737  024764                                                            ;SO THAT SHORTER READ STOP DISTANCE
                                                                                ;SHALL POSITION HEAD IN CLEAN IRG GAP
                                                                                ;READ REV THAT EXTRA REC TO RE-POSITION TAPE
                                                LET ALLEOT :B= #0               ;CLEAR ALL UNITS @ EOT FLAG
024052                                                                                  CLRB    ALLEOT
024052  105037  003450
                                                JSR PC,SETRW                    ;STORE REWIND IN SEQ TBL,
024056  004737  006470
                                                LET (R1) := #END                ;STORE END IN SEQ TBL,
024062                                                                                  MOV     #END,(R1)
024062  012711  177777
                                                JSR PC,EXALL                    ;EXECUTE REWIND CMD ON ALL UNITS
024066  004737  006510
```

```
4949
4950  024072                                  EXIT    TST
4951  024072  104432                                                    TRAP    C$EXIT
4952  024074  000002                                                    .WORD   L10035-.
4953
4954                                          .EVEN
4955
4956  024076                                  ENDTST
4957  024076                            L10035:
4958  024076  104401                                                    TRAP    C$ETST
4959
```

```
                                        TEST 3: WRITE COMPATABILITY/WRITE UTILITY.


                                        .SBTTL  TEST 4:  READ COMPATABILITY/READ UTILITY.

                                        ;++
                                        ; TEST TO READ ENTIRE TAPE FORWARD AND REVERSE.
                                        ;--

                                                BGNTST
                                        T4::

                                                LET RANDOM :B= #1               ;SET THE RANDOM OPERATIONS FLAG.
                                                                                        MOVB    #1,RANDOM
                                                LET EXPBOT :B= #1               ;SET EXPECT BOT FLAG.
                                                                                        MOVB    #1,EXPBOT
                                                JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
                                                JSR PC,SETRW                    ;CMD2=REWIND.
                                                LET STAFLG :B= #0               ;CLEAR START FLAG
                                                                                        CLRB    STAFLG
                                                LET (R1)+ := #RDF               ;CMD3 = READ FORWARD.
                                                                                        MOV     #RDF,(R1)+
                                                LET (R1)+ := #DATCNT            ;SET LENGTH TO MAX FOR UNKNOWN LENGTHS.
                                                                                        MOV     #DATCNT,(R1)+
                                                LET (R1)+ := #77777             ;SET RECORD COUNT TO MAX FOR WHOLE TAPE.
                                                                                        MOV     #77777,(R1)+
                                                LET (R1)+ := #RANP              ;PATTERN = RANDOM.
                                                                                        MOV     #RANP,(R1)+
                                                LET (R1) := #END                ;STORE END OF SEQUENCE CODE IN TABLE.
                                                                                        MOV     #END,(R1)
                                                JSR     PC,EXALL                ;EXECUTE ALL CMDS IN SEQ TBL ON ALL UNITS.
                                                LET ALLEOT :B= ALLEOT + #1      ;FLAG TO ALLOW ALL UNITS AT EOT TO READ REV
                                                                                        INCB    ALLEOT
                                                LET R1 := #CMDSEQ               ;INIT CMD SEQ TBL POINTER.
                                                                                        MOV     #CMDSEQ,R1
                                                LET (R1)+ := #RDR               ;CMD1 = READ REVERSE.
                                                                                        MOV     #RDR,(R1)+
                                                LET (R1)+ := #DATCNT            ;SET LENGTH TO MAX FOR UNKNOWN LENGTHS.
                                                                                        MOV     #DATCNT,(R1)+
                                                LET (R1)+ := #77777             ;RECORD COUNT = MAX FOR WHOLE TAPE.
                                                                                        MOV     #77777,(R1)+
                                                LET (R1)+ := #RANP              ;PATTERN = RANDOM.
                                                                                        MOV     #RANP,(R1)+
                                                LET (R1) := #END                ;STORE END OF SEQUENCE CODE IN TABLE.
                                                                                        MOV     #END,(R1)
                                                JSR     PC,EXALL                ;GO EXECUTE READ REV. OF ENTIRE TAPE.
                                                LET ALLEOT :B= #0               ;CLEAR ALL UNITS @ EOT FLAG
                                                                                        CLRB    ALLEOT

                                                EXIT    TST

                                                                                        TRAP    C$EXIT
                                                                                        .WORD   L10036-.


                                                .EVEN

                                                ENDTST
                                        L10036:

                                                                                        TRAP    C$ETST
```

```
                                        TEST 5: EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.

                                        .SBTTL  TEST 5: EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.

                                        ;++
                                        ; TEST TO EXECUTE  OPERATOR SELECTED COMMAND SEQUENCE.
                                        ;--

                                                BGNTST
                                        T5::

                                                LET RANDOM :B= #0               ;CLEAR RAMDOM MODE FLAG.
                                                                                        CLRB    RANDOM
                                                LET EXPBOT :B= #1               ;SET EXPECT BOT FLAG.
                                                                                        MOVB    #1,EXPBOT
                                                LET IRE :B= PIRE                ;MOVE INHIBIT RFC ERROR REPORT FLAG.
                                                                                        MOVB    PIRE,IRE
                                                JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
                                                LET CMDSEQ+2 := CHAR            ;MOVE CHAR CODE FROM P TBL TO SEQ TBL.
                                                                                        MOV     CHAR,CMDSEQ+2
                                                LET R2 := #CMDD                 ;R2 POINTS TO CMD2 IN SOFT P TABLE.
                                                                                        MOV     #CMDD,R2
                                                JSR     PC,PTCMDS               ;MOVE CMD 2 FROM P TBL TO SEQ TBL.
                                                JSR     PC,PTCMDS               ;MOVE CMD 3 FROM P TBL TO SEQ TBL.
                                                JSR     PC,PTCMDS               ;MOVE CMD 4 FROM P TBL TO SEQ TBL.
                                                JSR     PC,PTCMDS               ;MOVE CMD 5 FROM P TBL TO SEQ TBL.
                                                JSR     PC,PTCMDS               ;MOVE CMD 6 FROM P TBL TO SEQ TBL.
                                                JSR     PC,PTCMDS               ;MOVE CMD 7 FROM P TBL TO SEQ TBL.
                                                JSR     PC,PTCMDS               ;MOVE END CMD FROM P TBL TO SEQ TBL.
                                                LET JLOOP := #0                 ;CLEAR JMP CMD LOOP COUNT.
                                                                                        CLR     JLOOP
                                                LET STAFLG :B= #0               ;CLEAR START FLAG
                                                                                        CLRB    STAFLG
                                                LET R1 := #CMDSEQ               ;INIT SEQUENCE TABLE POINTER.
                                                                                        MOV     #CMDSEQ,R1
                                        3$:     WHILE (R1) NE #END DO           ;WHILE THERE ARE CMDS LEFT IN SEQUENCE TBLE:
                                                                                50431$:
                                                                                        CMP     (R1),#END
                                                                                        BEQ     50432$
                                                  CMP   #JMP.C,(R1)             ;IS THIS A JUMP CMD?
                                                  BNE   6$                      ;BR IF NOT.
                                                  LET R1 := R1 + #2             ;POINT TO BRF.
                                                                                        ADD     #2,R1
                                                  MOV   (R1)+,JLOC              ;SAVE BRF (LOCATION).
                                                  CMP (R1)+,JLOOP               ;HAS LOOP COUNT BE SATISFIED?
                                                  BNE   1$                      ;IF NOT, JMP AGAIN.
                                                  LET R1 := R1 + #2             ;IF SO, ADJUST SEQ POUNTER
                                                                                        ADD     #2,R1
                                                  BR    3$                      ;AND GO TO NEXT COMMAND.
                                        1$:       LET JLOOP := JLOOP + #1       ;UPDATE THE LOOP COUNT.
                                                                                        INC     JLOOP
                                                  LET R1 := #CMDSEQ             ;INIT CMD SEQ TABLE POINTER.
                                                                                        MOV     #CMDSEQ,R1
                                        2$:       DEC   JLOC                    ;DECR LOCATION COUNTER.
                                                  BEQ   3$                      ;IF THIS IS THE RIGHT LOCATION TO JMP TO, GO SET
                                                  LET R1 := R1 + #10            ;IF NOT, UPDATE SEQ POINTER TO NEXT CMD.
                                                                                        ADD     #10,R1
                                                  BR    2$                      ;DO IT AGAIN.
```

```
5072  024424  022711  000020          6$:     CMP   #DLY.C,(R1)             ;DELAY?
5073  024430  001026                          BNE   4$                      ;BR IF NOT.
5074  024432                                  LET R1 := R1 + #4              ;R1 = LOCATION OF N COUNT.
5075  024432  062701  000004                                                          ADD     #4,R1
5076  024436                                  LET TIME2 := (R1)              ;SAVE N COUNT.
5077  024436  011137  003366                                                          MOV     (R1),TIME2
5078  024442                          7$:     DELAY 1                        ;GO TO SUPER-WAIT 1 MSEC.
5079  024442  012727  000001                                                          MOV     #1,(PC)+
5080  024446  000000                                                          .WORD   0
5081  024450  013727  002116                                                          MOV     L$DLY,(PC)+
5082  024454  000000                                                          .WORD   0
5083  024456  005367  177772                                                          DEC     -6(PC)
5084  024462  001375                                                          BNE     .-4
5085  024464  005367  177756                                                          DEC     -22(PC)
5086  024470  001367                                                          BNE     .-20
5087  024472  005337  003366                  DEC   TIME2
5088  024476  001361                          BNE   7$
5089  024500                                  LET R1 := R1 + #4              ;POINT TO NEXT CMD.
5090  024500  062701  000004                                                          ADD     #4,R1
5091  024504  000715                          BR    3$                       ;GO CHECK NEXT CMD.
5092  024506  004737  007452          4$:     JSR   PC,SETUP                 ;GO SETUP THE COMMAND BLOCK.
5093  024512                                  WHILE NCNT LT NCNT1 DO         ;WHILE THERE ARE RECORDS REMAINING:
5094  024512                                                                                  50433$:
5095  024512  023737  003340  003342                                                          CMP     NCNT,NCNT1
5096  024520  002103                                                          BGE     50434$
5097  024522  004737  007344                    JSR PC,CMDAC                 ;STORE CMD ASC'I IN ERROR MSG.
5098  024526  004737  007004                    JSR PC,EXSUB                 ;ISSUE CMD TO ALL,AWAIT INTS,CHECK STATUS.
5099  024532                                    IF CMDWRD EQ #GES THEN       ;IF CMD IS GET STATUS THEN:
5100  024532  023727  003346  100017                                                          CMP     CMDWRD,#GES
5101  024540  001002                                                          BNE     50435$
5102  024542  004737  015642                      JSR PC,PRXST               ;PRINT EXTENDED STATUS REGISTERS.
5103  024546                                    ENDIF
5104  024546                                                                                  50435$:
5105  024546  004737  015724                    JSR PC,CKHAE                 ;CHECK HALT AFTER EACH CMD FLAG.
5106  024552                                    LET R2 := #1                 ;SET ALL UNITS AT BOT/EOT.
5107  024552  012702  000001                                                          MOV     #1,R2
5108  024556  004737  015332                    JSR PC,FIRSTU                ;FIND FIRST UNIT.
5109  024562                                    WHILE DEVTBL(R5) NE #END DO  ;WHILE THERE ARE MORE UNITS:
5110  024562                                                                                  50436$:
5111  024562  026527  002532  177777                                                          CMP     DEVTBL(R5),#END
5112  024570  001426                                                          BEQ     50437$
5113  024572                                      IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
5114  024572  032737  000400  003346                                                          BIT     #MOD.CO,CMDWRD
5115  024600  001406                                                          BEQ     50440$
5116  024602                                        IF #X0.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT THEN:
5117  024602  032765  000002  003426                                                          BIT     #X0.BOT,EOTFLG(R
5118  024610  001001                                                          BNE     50441$
5119  024612                                          LET R2 := #0                 ;CLEAR EOT/BOT FLAG.
5120  024612  005002                                                          CLR     R2
5121  024614                                        ENDIF
5122  024614                                                                                  50441$:
5123  024614                                      ELSE                         ;ELSE IF CMD IS NOT REVERSE:
5124  024614  000411                                                          BR      50442$
5125  024616                                                                                  50440$:
5126  024616                                        IF #X0.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
5127  024616  032765  000001  003426                                                          BIT     #X0.EOT,EOTFLG(R
```

```
5128  024624  001404                                                          BEQ     50443$
5129  024626  032737  000001  003346                                          BIT     #CMD.CO,CMDWRD
5130  024634  001001                                                          BNE     50444$
5131  024636                                                          50443$:
5132                                                          ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
5133  024636                        LET R2 := #0                    ;CLEAR EOT/BOT FLAG.
5134  024636  005002                                                          CLR     R2
5135  024640                      ENDIF
5136  024640                                                          50444$:
5137  024640                    ENDIF
5138  024640                                                          50442$:
5139  024640  004737  015400      JSR PC,NEXTU                    ;FIND NEXT UNIT
5140  024644                    ENDDO                             ;
5141  024644  000746                                                          BR      50436$
5142  024646                                                          50437$:
5143  024646                    IF R2 EQ #1 THEN                ;IF ALL UNIT ARE AT EOT/BOT THEN:
5144  024646  020227  000001                                                  CMP     R2,#1
5145  024652  001016                                                          BNE     50445$
5146  024654                      LET NCNT1 := NCNT + #1        ;FORCE TERMINATION OF COMMAND.
5147  024654  013737  003340  003342                                          MOV     NCNT,NCNT1
5148  024662  005237  003342                                                  INC     NCNT1
5149  024666                      LET ALLEOT :B= ALLEOT + #1 ;FLAG ALL UNITS AT EOT/BOT TO ALLOW VERIFY OF D
5150  024666  105237  003450                                                  INCB    ALLEOT
5151  024672                      IF CMDLG EQ #2 THEN           ;WHEN WRITING IS CURRENT COMMAND
5152  024672  023727  003354  000002                                          CMP     CMDLG,#2
5153  024700  001002                                                          BNE     50446$
5154  024702  004737  024764        JSR PC,T5WEOT               ;GO WRITE/READ REV ONE RECORD BEYOND EOT
5155  024706                      ENDIF
5156  024706                                                          50446$:
5157  024706                    ELSE
5158  024706  000402                                                          BR      50447$
5159  024710                                                          50445$:
5160  024710                      LET ALLEOT :B= #0 ;WHEN NOT ALL @EOT, CLEAR FLAG
5161  024710  105037  003450                                                  CLRB    ALLEOT
5162  024714                      ENDIF
5163  024714                                                          50447$:
5164  024714                      LET NCNT := NCNT + #1         ;UPDATE RECORD COUNT.
5165  024714  005237  003340                                                  INC     NCNT
5166  024720                      LET PCMDWD := CMDWRD          ;SAVE PREVIOUS COMMAND WORD.
5167  024720  013737  003346  003352                                          MOV     CMDWRD,PCMDWD
5168  024726                    ENDDO
5169  024726  000671                                                          BR      50433$
5170  024730                                                          50434$:
5171  024730  004737  014316    JSR   PC,VFYDAT                  ;IF LAST CMD WAS A WRITE VERIFY, THEN GO
5172                                                          ;VERIFY THE LAST N RECORDS OF DATA.
5173  024734                  ENDDO
5174  024734  000601                                                          BR      50431$
5175  024736                                                          50432$:
5176
5177  024736                  EXIT    TST
5178  024736  104432                                                          TRAP    C$EXIT
5179  024740  000130                                                          .WORD   L10037-.
5180
5181
5182
5183
```

```
                                ;       SUBROUTINE TO MOVE A COMMAND FROM THE SOFTWARE P TABLE TO
                                ;       THE COMMAND SEQUENCE TABLE.
                                ;       INPUTS:         R2 = POINTER TO SOFT 'P'' TABLE
                                ;       OUTPUTS:
                                ;       REGISTERS:      R3.
                                ;       CALLS:

024742                          PTCMDS: LET R3 := (R2)+ - #1 SHIFT +1   ;R3 = COMMAND TABLE INDEX.
024742  012203                                                                  MOV     (R2)+,R3
024744  005303                                                                  DEC     R3
024746  006303                                                                  ASL     R3
024750                                  LET (R1)+ := CMDTBL(R3)         ;MOVE COMMAND WORD.
024750  016321  003562                                                          MOV     CMDTBL(R3),(R1)+
024754                                  LET (R1)+ := (R2)+              ;MOVE # OF BYTES.
024754  012221                                                                  MOV     (R2)+,(R1)+
024756                                  LET (R1)+ := (R2)+              ;MOVE # OF OPERATIONS.
024756  012221                                                                  MOV     (R2)+,(R1)+
024760                                  LET (R1)+ := (R2)+              ;MOVF PATTERN CODE.
024760  012221                                                                  MOV     (R2)+,(R1)+
024762  000207                          RTS PC

                                ;       SUBROUTINE TO WRITE THEN READ REVERSE ONE RECORD BEYOND EOT
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:          CMDAC,EXSUB,CKHAE

024764  000240                  T5WEOT: NOP
024766  000240                          NOP
024770  004737  007004                  JSR PC,EXSUB            ;WRITE ONE RECORD BEYOND EOT
024774  004737  015724                  JSR PC,CKHAE            ;SO THAT READ SHORTER STOP DISTANCE
                                                                ;SHALL POSITION HEAD IN CLEAN IRG GAP
025000                                  LET PCMDWD :- CMDWRD    ;REPOSITION TAPE
025000  013737  003346  003352                                                  MOV     CMDWRD,PCMDWD
025006                                  LET CMDWRD := #RDR      ;BEFORE EXTRA RECORD
025006  012737  104401  003346                                                  MOV     #RDR,CMDWRD
025014                                  LET CMDLG :  #4         ;BY READING REVERSE
025014  012737  000004  003354                                                  MOV     #4,CMDLG
025022                                  LET CMDPKT := CMDWRD CLR.BY #BPF.C
025022  013737  003346  002310                                                  MOV     CMDWRD,CMDPKT
025030  042737  004000  002310                                                  BIC     #BPF.C,CMDPKT
025036                                  LET CMDSAV :- CMDPKT    ;THAT RECORD TO ALLOW
025036  013737  002310  003350                                                  MOV     CMDPKT,CMDSAV
025044                                  LET CMDPKT+CP.ADL :- DATARD     ;NEXT COMMAND IN THE
025044  013737  003336  002312                                                  MOV     DATARD,CMDPKT+CP
025052  004737  007344                  JSR PC,CMDAC            ;TABLE TO BE EXECUTED
025056  004737  007004                  JSR PC,EXSUB
025062  004737  015724                  JSR PC,CKHAE
025066  000207                          RTS PC


                                        .EVEN

025070                                  ENDTST
```

```
 5240  025070                          L10037:
 5241  025070  104401                                                  TRAP    C$ETST
 5242
 5243  025072                                  ENDMOD
```

```
                                .TITLE PARAMETER CODING

                                .SBTTL  HARDWARE PARAMETER CODING SECTION

025072                                  BGNMOD

                                ;++
                                ; THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
                                ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                ; WITH THE OPERATOR.
                                ;--

025072                                  BGNHRD
025072  000024                                                          .WORD L10040-L$HARD/2
025074                          L$HARD::

025074                                  GPRMA   TS4ADR,0,0,160002,177564,YES
025074  000031                                                          .WORD   T$CODE
025076  025120                                                          .WORD   TS4ADR
025100  160002                                                          .WORD   T$LOLIM
025102  177564                                                          .WORD   T$HILIM
025104                                  GPRMD   TS4VCT,2,0,777,60,776,YES
025104  001032                                                          .WORD   T$CODE
025106  025135                                                          .WORD   TS4VCT
025110  000777                                                          .WORD   777
025112  000060                                                          .WORD   T$LOLIM
025114  000776                                                          .WORD   T$HILIM

025116                                  EXIT HRD
025116  013004                                                          .WORD   T$CODE

                                        .NLIST  BEX
025120  051524  051123  040440  TS4ADR: .ASCIZ  /ISSR ADDRESS/
025135     126  041505  047524  TS4VCT: .ASCIZ  /VECTOR/
                                        .LIST   BEX
                                        .EVEN

025144                                  ENDHRD

                                                                        .EVEN
025144                          L10040:
```

```
                                .SBTTL  SOFTWARE PARAMETER CODING SECTION

                                ;++
                                ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
                                ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES  THE
                                ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                ; WITH THE OPERATOR.
                                ;--

025144                                  BCNSFT
025144  000501                                                    .WORD L10041-L$SOFT/2
025146                          L$SOFT::

025146                                  GPRML   CLRM,0,1,YES
025146  000130                                                    .WORD   T$CODE
025150  025710                                                    .WORD   CLRM
025152  000001                                                    .WORD   1
025154                                  GPRML   RRVM,0,400,YES
025154  000130                                                    .WORD   T$CODE
025156  025727                                                    .WORD   RRVM
025160  000400                                                    .WORD   400
025162                                  GPRML   RCVERM,2,400,YES
025162  001130                                                    .WORD   T$CODE
025164  026032                                                    .WORD   RCVERM
025166  000400                                                    .WORD   400
025170                                  GPRML   HAEM,2,1,YES
025170  001130                                                    .WORD   T$CODE
025172  025756                                                    .WORD   HAEM
025174  000001                                                    .WORD   1
025176                                  GPRML   IRECM,6,400,YES
025176  003130                                                    .WORD   T$CODE
025200  026106                                                    .WORD   IRECM
025202  000400                                                    .WORD   400
025204                                  XFERT   NEXTSP
025204  004024                                                    .WORD   T$CODE
025206                                  GPRML   BADTM,4,1,YES
025206  002130                                                    .WORD   T$CODE
025210  026002                                                    .WORD   BADTM
025212  000001                                                    .WORD   1
025214                          NEXTSP: GPRML   DINTM,6,1,YES
025214  003130                                                    .WORD   T$CODE
025216  026063                                                    .WORD   DINTM
025220  000001                                                    .WORD   1
025222                                  GPRML   IREM,12,1,YES
025222  005130                                                    .WORD   T$CODE
025224  026153                                                    .WORD   IREM
025226  000001                                                    .WORD   1
025230                                  GPRML   CHGM,10,1,YES
025230  004130                                                    .WORD   T$CODE
025232  026127                                                    .WORD   CHGM
025234  000001                                                    .WORD   1
025236                                  XFER-F  ENDSP1
025236  127044                                                    .WORD   T$CODE
```

```
5340  025240
5341  025240  006032                GPRMD   CHARM,14,O,377,0,777 YES
5342  025242  026204                                                  .WORD   T$CODE
5343  025244  000377                                                  .WORD   CHARM
5344  025246  000000                                                  .WORD   377
5345  025250  000777                                                  .WORD   T$LOLIM
5346  025252                                                          .WORD   T$HILIM
5347  025252  007052                GPRMD   CMD2M,16,D,37,1,33,YES
5348  025254  026231                                                  .WORD   T$CODE
5349  025256  000037                                                  .WORD   CMD2M
5350  025260  000001                                                   WORD   37
5351  025262  000033                                                  .WORD   T$LOLIM
5352  025264                                                          .WORD   T$HILIM
5353  025264  010052                GPRMD   BPCRM,20,D,-1,1,DATCNT,YES
5354  025266  026237                                                  .WORD   T$CODE
5355  025270  177777                                                  .WORD   BPCRM
5356  025272  000001                                                  .WORD   -1
5357  025274  004000                                                  .WORD   T$LOLIM
5358  025276                                                          .WORD   T$HILIM
5359  025276  011052                GPRMD   NUMBM,22,D,-1,1,77777,YES
5360  025300  026251                                                  .WORD   T$CODE
5361  025302  177777                                                  .WORD   NUMBM
5362  025304  000001                                                  .WORD   -1
5363  025306  077777                                                  .WORD   T$LOLIM
5364  025310                                                          .WORD   T$HILIM
5365  025310  012052                GPRMD   PATTM,24,D,17,0,10,YES
5366  025312  026271                                                  .WORD   T$CODE
5367  025314  000017                                                  .WORD   PATTM
5368  025316  000000                                                  .WORD   17
5369  025320  000010                                                  .WORD   T$LOLIM
5370  025322                                                          .WORD   T$HILIM
5371  025322  013052                GPRMD   CMD3M,26,D,37,1,33,YES
5372  025324  026304                                                  .WORD   T$CODE
5373  025326  000037                                                  .WORD   CMD3M
5374  025330  000001                                                  .WORD   37
5375  025332  000033                                                  .WORD   T$LOLIM
5376  025334                                                          .WORD   T$HILIM
5377  025334  014052                GPRMD   BPCRM,30,D,-1,1,DATCNT,YES
5378  025336  026237                                                  .WORD   T$CODE
5379  025340  177777                                                  .WORD   BPCRM
5380  025342  000001                                                  .WORD   -1
5381  025344  004000                                                  .WORD   T$LOLIM
5382  025346                                                          .WORD   T$HILIM
5383  025346  015052                GPRMD   NUMBM,32,D,-1,1,77777,YES
5384  025350  026251                                                  .WORD   T$CODE
5385  025352  177777                                                  .WORD   NUMBM
5386  025354  000001                                                  .WORD   -1
5387  025356  077777                                                  .WORD   T$LOLIM
5388  025360                                                          .WORD   T$HILIM
5389  025360  016052                GPRMD   PATTM,34,D,17,0,10,YES
5390  025362  026271                                                  .WORD   T$CODE
5391  025364  000017                                                  .WORD   PATTM
5392  025366  000000                                                  .WORD   17
5393  025370  000010                                                  .WORD   T$LOLIM
5394  025372                                                          .WORD   T$HILIM
5395  025372  017052                GPRMD   CMD4M,36,D,37,1,33,YES
                                                                      .WORD   T$CODE
```

```
5396  025374  026312                                                    .WORD   CMD4M
5397  025376  000037                                                    .WORD   37
5398  025400  000001                                                    .WORD   T$LOLIM
5399  025402  000033                                                    .WORD   T$HILIM
5400  025404                        GPRMD   BPCRM,40,D,-1,1,DATCNT,YES
5401  025404  020052                                                    .WORD   T$CODE
5402  025406  026237                                                    .WORD   BPCRM
5403  025410  177777                                                    .WORD   -1
5404  025412  000001                                                    .WORD   T$LOLIM
5405  025414  004000                                                    .WORD   T$HILIM
5406  025416                        GPRMD   NUMBM,42,D,-1,1,77777,YES
5407  025416  021052                                                    .WORD   T$CODE
5408  025420  026251                                                    .WORD   NUMBM
5409  025422  177777                                                    .WORD   -1
5410  025424  000001                                                    .WORD   T$LOLIM
5411  025426  077777                                                    .WORD   T$HILIM
5412  025430                        GPRMD   PATTM,44,D,17,0,10,YES
5413  025430  022052                                                    .WORD   T$CODE
5414  025432  026271                                                    .WORD   PATTM
5415  025434  000017                                                    .WORD   17
5416  025436  000000                                                    .WORD   T$LOLIM
5417  025440  000010                                                    .WORD   T$HILIM
5418  025442                        GPRMD   CMD5M,46,D,37,1,33,YES
5419  025442  023052                                                    .WORD   T$CODE
5420  025444  026320                                                    .WORD   CMD5M
5421  025446  000037                                                    .WORD   37
5422  025450  000001                                                    .WORD   T$LOLIM
5423  025452  000033                                                    .WORD   T$HILIM
5424  025454                        GPRMD   BPCRM,50,D,-1,1,DATCNT,YES
5425  025454  024052                                                    .WORD   T$CODE
5426  025456  026237                                                    .WORD   BPCRM
5427  025460  177777                                                    .WORD   -1
5428  025462  000001                                                    .WORD   T$LOLIM
5429  025464  004000                                                    .WORD   T$HILIM
5430  025466                        GPRMD   NUMBM,52,D,-1,1,77777,YES
5431  025466  025052                                                    .WORD   T$CODE
5432  025470  026251                                                    .WORD   NUMBM
5433  025472  177777                                                    .WORD   -1
5434  025474  000001                                                    .WORD   T$LOLIM
5435  025476  077777                                                    .WORD   T$HILIM
5436  025500                        GPRMD   PATTM,54,D,17,0,10,YES
5437  025500  026052                                                    .WORD   T$CODE
5438  025502  026271                                                    .WORD   PATTM
5439  025504  000017                                                    .WORD   17
5440  025506  000000                                                    .WORD   T$LOLIM
5441  025510  000010                                                    .WORD   T$HILIM
5442  025512                        XFER    ENDSP2
5443  025512  002004                                                    .WORD   T$CODE
5444  025514                ENDSP1: XFER    ENDSP
5445  025514  075004                                                    .WORD   T$CODE
5446  025516                ENDSP2: GPRMD   CMD6M,56,D,37,1,33,YES
5447  025516  027052                                                    .WORD   T$CODE
5448  025520  026326                                                    .WORD   CMD6M
5449  025522  000037                                                    .WORD   37
5450  025524  000001                                                    .WORD   T$LOLIM
5451  025526  000033                                                    .WORD   T$HILIM
```

```
5452  025530                    GPRMD   BPCRM,60,D,-1,1,DATCNT,YES
5453  025530  030052                                              .WORD   T$CODE
5454  025532  026237                                              .WORD   BPCRM
5455  025534  177777                                              .WORD   -1
5456  025536  000001                                              .WORD   T$LOLIM
5457  025540  004000                                              .WORD   T$HILIM
5458  025542                    GPRMD   NUMBM,62,D,-1,1,77777,YES
5459  025542  031052                                              .WORD   T$CODE
5460  025544  026251                                              .WORD   NUMBM
5461  025546  177777                                              .WORD   -1
5462  025550  000001                                              .WORD   T$LOLIM
5463  025552  077777                                              .WORD   T$HILIM
5464  025554                    GPRMD   PATTM,64,D,17,0,10,YES
5465  025554  032052                                              .WORD   T$CODE
5466  025556  026271                                              .WORD   PATTM
5467  025560  000017                                              .WORD   17
5468  025562  000000                                              .WORD   T$LOLIM
5469  025564  000010                                              .WORD   T$HILIM
5470  025566                    GPRMD   CMD7M,66,D,37,1,33,YES
5471  025566  033052                                              .WORD   T$CODE
5472  025570  026334                                              .WORD   CMD7M
5473  025572  000037                                              .WORD   37
5474  025574  000001                                              .WORD   T$LOLIM
5475  025576  000033                                              .WORD   T$HILIM
5476  025600                    GPRMD   BPCRM,70,D,-1,1,DATCNT,YES
5477  025600  034052                                              .WORD   T$CODE
5478  025602  026237                                              .WORD   BPCRM
5479  025604  177777                                              .WORD   -1
5480  025606  000001                                              .WORD   T$LOLIM
5481  025610  004000                                              .WORD   T$HILIM
5482  025612                    GPRMD   NUMBM,72,D,-1,1,77777,YES
5483  025612  035052                                              .WORD   T$CODE
5484  025614  026251                                              .WORD   NUMBM
5485  025616  177777                                              .WORD   -1
5486  025620  000001                                              .WORD   T$LOLIM
5487  025622  077777                                              .WORD   T$HILIM
5488  025624                    GPRMD   PATTM,74,D,17,0,10,YES
5489  025624  036052                                              .WORD   T$CODE
5490  025626  026271                                              .WORD   PATTM
5491  025630  000017                                              .WORD   17
5492  025632  000000                                              .WORD   T$LOLIM
5493  025634  000010                                              .WORD   T$HILIM
5494  025636                    GPRMD   CMD8M,76,D,37,1,33,YES
5495  025636  037052                                              .WORD   T$CODE
5496  025640  026342                                              .WORD   CMD8M
5497  025642  000037                                              .WORD   37
5498  025644  000001                                              .WORD   T$LOLIM
5499  025646  000033                                              .WORD   T$HILIM
5500  025650                    GPRMD   BPCRM,100,D,-1,1,DATCNT,YES
5501  025650  040052                                              .WORD   T$CODE
5502  025652  026237                                              .WORD   BPCRM
5503  025654  177777                                              .WORD   -1
5504  025656  000001                                              .WORD   T$LOLIM
5505  025660  004000                                              .WORD   T$HILIM
5506  025662                    GPRMD   NUMBM,102,D,-1,1,77777,YES
5507  025662  041052                                              .WORD   T$CODE
```

```
 5508  025664  026251                                                    .WORD   NUMBM
 5509  025666  177777                                                    .WORD   -1
 5510  025670  000001                                                    .WORD   T$LOLIM
 5511  025672  077777                                                    .WORD   T$HILIM
 5512  025674                        GPRMD   PATTM,104,D,17,0,10,YES
 5513  025674  042052                                                    .WORD   T$CODE
 5514  025676  026271                                                    .WORD   PATTM
 5515  025700  000017                                                    .WORD   17
 5516  025702  000000                                                    .WORD   T$LOLIM
 5517  025704  000010                                                    .WORD   T$HILIM
 5518  025706                  ENDSP:
 5519  025706                        XFER    JMPMSG
 5520  025706  176004                                                    .WORD   T$CODE
```

```
5521
5522                                                    .NLIST  BEX
     025710 046103 040505 020122 CLRM:    .ASCIZ  /CLEAR COUNTERS/
     025727    122 051505 052105 RRVM:    .ASCIZ  /RESET RANDOM VARIABLES/
     025756 040510 052114 040440 HAEM:    .ASCIZ  /HALT AFTER EACH CMD/
     026002 040502 020104 040524 BADTM:   .ASCIZ  /BAD TAPE SPOT DETECTION/
     026032 051120 047111 020124 RCVERM:  .ASCIZ  /PRINT RECOVERABLE ERRORS/
     026063    104 051511 041101 DINTM:   .ASCIZ  /DISABLE INTERRUPTS/
     026106 047111 044510 044502 IRECM:   .ASCIZ  /INHIBIT RECOVERY/
     026127    103 040510 043516 CHGM:    .ASCIZ  /CHANGE CMD SEQUENCE/
     026153    111 044116 041111 IREM:    .ASCIZ  /INHIBIT RFC ERROR REPORT/
     026204 044103 051101 041501 CHARM:   .ASCIZ  /CHARACTERISTICS CODE/
     026231    103 042115 031057 CMD2M:   .ASCIZ  ''CMD/2''
     026237    102 043122 041440 BPCRM:   .ASCIZ  /BRF COUNT/
     026251    043 047440 020106 NUMBM:   .ASCIZ  /# OF OPERATIONS/
     026271    120 052101 042524 PATTM:   .ASCIZ  /PATTERN/
                                          .LIST   BEX
5523           026302                     .EVEN
5524
5525 026302                          JMPMSG:
5526 026302                                  EXIT SFT
5527 026302 023004                                                      .WORD   T$CODE
5528
5529                                                    .NLIST  BEX
     026304 046503 027504 000063 CMD3M:   .ASCIZ  ''CMD/3''
     026312 046503 027504 000064 CMD4M:   .ASCIZ  ''CMD/4''
     026320 046503 027504 000065 CMD5M:   .ASCIZ  ''CMD/5''
     026326 046503 027504 000066 CMD6M:   .ASCIZ  ''CMD/6''
     026334 046503 027504 000067 CMD7M:   .ASCIZ  ''CMD/7''
     026342 046503 027504 000070 CMD8M:   .ASCIZ  ''CMD/8''
                                          .LIST   BEX
5530                                      .EVEN
5531
5532 026350                                  ENDSFT
5533                                                                    .EVEN
5534 026350                          L10041:
5535
5536                                 ;*****************************************************************************
5537                                 ;*****************************************************************************
5538                                 ;       PATCH AREA
5539
5540 026350 000100                   PATCH:: .BLKW   64.
5541                                 ;*****************************************************************************
5542                                 ;*****************************************************************************
5543
5544 026550                                  LASTAD
5545                                                                    .EVEN
5546 026550 026572                                                      .WORD T$FREE
5547 026552 000007                                                      .WORD T$SIZE
5548 026554                          L$LAST::
5549 026554                                  ENDMOD
```

```
 5550                                   .SBTTL  HARD CODED P-TBL
 5551
 5552                                   ;++
 5553                                   ;DIAG IS PRE-PARAMETERIZED PER TBL
 5554                                   ;--
 5555
 5556  026554                                   BGNSETUP 1
 5557  026554                                     BGNPTAB
 5558  026554  000000                                                        .WORD   0
 5559  026556  000000                                                        .WORD   0
 5560  026560  000000                                                        .WORD   0
 5561  026562  000000                                                        .WORD   0
 5562  026564  000002                                                        .WORD   L10044-./2-1
 5563  026566                         L10042:
 5564  026566  172522                             172522
 5565  026570  000224                             224
 5566  026572                                   ENDPTAB
 5567  026572                         L10044:
 5568  026572                                   ENDSETUP
 5569
 5570          000001                   .END
```

```
ACK.C = 100000 G
ADR   = 000020 G
ALLEOT  003450 G
ASSEMB= 000010
ATTNM   004335 G
AUDRPM  004645 G
AUTODM  021330
BADTM   026002
BADTSW  002206 G
BFSEQ   022504
BFSEQ0  022530
BFSEQ1  022602
BFSEQ2  022614
BFSEQ3  022706
BFSEQ4  022760
BFSEQ5  023022
BFSEQ6  023074
BFSEQ7  023126
BFSEQ8  023160
BFSEQ9  023212
BFSE10  023234
BGNFLG= 003404
BINC    014302
BIT0  = 000001 G
BIT00 = 000001 G
BIT01 = 000002 G
BIT02 = 000004 G
BIT03 = 000010 G
BIT04 = 000020 G
BIT05 = 000040 G
BIT06 = 000100 G
BIT07 = 000200 G
BIT08 = 000400 G
BIT09 = 001000 G
BIT1  = 000002 G
BIT10 = 002000 G
BIT11 = 004000 G
BIT12 = 010000 G
BIT13 = 020000 G
BIT14 = 040000 G
BIT15 = 100000 G
BIT2  = 000004 G
BIT3  = 000010 G
BIT4  = 000020 G
BIT5  = 000040 G
BIT6  = 000100 G
BIT7  = 000200 G
BIT8  = 000400 G
BIT9  = 001000 G
BOE   = 000400 G
BORERS  013546 G
BPCRM   026237
BRCPK   002324 G
BRFCNT  003344 G
BRF.C = 004000 G
BTADDR  002544 G
BTMSG1  013342
BTMSG2  013427
BTMSG3  013477
BTPT    003436 G
BTRPT   016410
BT0     002774 G
BT1     003046 G
BT2     003120 G
BT3     003172 G
CHAR    002216 G
CHARM   026204
CHGFLG  002212 G
CHGM    026127
CHKERR  011440 G
CH.EAI= 000040 G
CH.ERI= 000020 G
CH.ESS= 000200 G
CKDATA  014716 G
CKDCNT  015326
CKDFF   015330
CKHAE   015724 G
CKHRTN  016012
CLN   = 101012 G
CLRERR  011136 G
CLRFLG  002202 G
CLRM    025710
CMDAC   007344 G
CMDASC  003650 G
CMDD    002220 G
CMDLG   003354 G
CMDPKM  004102 G
CMDPKT  002310 G
CMDSAV  003350 G
CMDSEQ  003460 G
CMDSE2  003470 G
CMDTBL  003562 G
CMDWRD  003346 G
CMD.C0= 000001 G
CMD.C1= 000002 G
CMD.C2= 000004 G
CMD.C3= 000010 G
CMD.C4= 000020 G
CMD2M   026231
CMD3M   026304
CMD4M   026312
CMD5M   026320
CMD6M   026326
CMD7M   026334
CMD8M   026342
CNTBGN= 002554
CNTEND= 003324
CNTLEN= 000550 G
CODELM  003772 G
COUNTE= 050364
CP.ADH= 000004 G
CP.ADL= 000002 G
CP.CMD= 000000 G
CP.CNT= 000006 G
CRLF    005213 G
CRLFSP  005216 G
CTCC    003376 G
CVC.C = 040000 G
C$AU  = 000052
C$AUTO= 000061
C$BRK = 000022
C$BSEG= 000004
C$BSUB= 000002
C$CEFG= 000045
C$CLCK= 000062
C$CLEA= 000012
C$CLOS= 000035
C$CLP1= 000006
C$CVEC= 000036
C$DCLN= 000044
C$DODU= 000051
C$DRPT= 000024
C$DU  = 000053
C$EDIT= 000002
C$ERDF= 000055
C$ERHR= 000056
C$ERRO= 000060
C$ERSF= 000054
C$ERSO= 000057
C$ESCA= 000010
C$ESEG= 000005
C$ESUB= 000003
C$ETST= 000001
C$EXIT= 000032
C$GETB= 000026
C$GETW= 000027
C$GMAN= 000043
C$GPHR= 000042
C$GPLO= 000030
C$GPRI= 000040
C$INIT= 000011
C$INVP= 000020
C$MANI= 000050
C$MEM = 000031
C$MSG = 000023
C$OPEN= 000034
C$PNTB= 000014
C$PNTF= 000017
C$PNTS= 000016
C$PNTX= 000015
C$QIO = 000377
C$RDBU= 000007
C$REFG= 000047
C$RESE= 000033
C$REVI= 000003
C$RFLA= 000021
C$RPT = 000025
C$SEFG= 000046
C$SPRI= 000041
C$SVEC= 000037
C$TPRI= 000013
DATARD  003336 G
DATAWT  003334 G
DATCNT= 004000 G
DEVTBL  002532 G
DFPTBL  002174 G
DFTSCH= 000040 G
DIA   = 100006 G
DIABLK= 003334 G
DIACNT= 000020 G
DIAGMC= 000000
DINT    002210 G
DINTM   026063
DLY   = 000020 G
DLY.C = 000020 G
DRI   = 100013 G
DROPDM  004616 G
DROPED  003446 G
DROPN   015640
DROPU   015430 G
DROPUA  015554
DRORTN  015632
DTAERM  005224 G
DTAER2  004677 G
DTAER3  004746 G
DTAER4  005010 G
DTAER5  005031 G
EF.CON= 000036 G
EF.NEW= 000035 G
EF.PWR= 000034 G
EF.RES= 000037 G
EF.STA= 000040 G
EINC    014310
END   = 177777 G
ENDERF= 003416
ENDFLG= 003452
ENDSP   025706
ENDSP1  025514
ENDSP2  025516
EOTFLG  003426 G
ERCVER  002205 G
ERLOG   003412 G
ERRREC  003415 G
ERS   = 100411 G
ERSFLG  003451 G
EVL   = 000004 G
EXALL   006510 G
EXARTN  007002
EXCRTN  010634
EXCUTE  010326 G
EXPBOT  003440 G
EXSUB   007004 G
E$END = 002100
E$LOAD= 000035
FATSM   004373 G
FIRSTU  015332 G
FMT.C0= 000000 G
FMT.C1= 000100 G
FTLCNT  003314 G
FUNRM   004353 G
F$AU  = 000015
F$AUTO= 000020
F$BGN = 000040
F$CLEA= 000007
F$DU  = 000016
F$END = 000041
F$HARD= 000004
F$HW  = 000013
F$INIT= 000006
F$JMP = 000050
F$MOD = 000000
F$MSG = 000011
F$PROT= 000021
F$PWR = 000017
F$RPT = 000012
F$SEG = 000003
F$SOFT= 000005
F$SRV = 000010
F$SUB = 000002
F$SW  = 000014
F$TEST= 000001
GCMDA   007416 G
GENPAT  010030 G
GES   = 100017 G
GETSTM  005157 G
GIT     010322
GOWAIT  010636 G
GSCPK   002320 G
G$CNTO= 000200
G$DISP= 000003
G$EXCP= 000400
G$HILI= 000002
G$LOLI= 000001
G$NO  = 000000
G$OFFS= 000400
```

```
G$OFSI= 000376          JMPMSG  026302          L$SPCP  002020 G        MSGPK0  002352 G        PRI02 = 000100 G
G$PRMA= 000001          JMP.C = 000040 G        L$SPTP  002024 G        MSGPK1  002370 G        PRI03 = 000140 G
G$PRMD= 000002          J$JMP = 000167          L$STA   002030 G        MSGPK2  002406 G        PRI04 = 000200 G
G$PRML= 000000          LENMSK  003356 G       L$SW    002202 G        MSGPK3  002424 G        PRI05 = 000240 G
G$RADA= 000140          LOE   = 040000 G        L$TEST  002114 G        MS.RFC= 000004 G        PRI06 = 000300 G
G$RADB= 000000          LOG     014016 G        L$TIML  002014 G        MS.XS0= 000006 G        PRI07 = 000340 G
G$RADD= 000040          LOT   = 000010 G        L$UNIT  002012 G        MS.XS1= 000010 G        PRXST   015642 G
G$RADL= 000120          L$ACP   002110 G        L10000  002200          MS.XS2= 000012 G        PTCMDS  024742
G$RADO= 000020          L$APT   002036 G        L10001  002310          MS.XS3= 000014 G        PURFLG  003453 G
G$XFER= 000004          L$AU    021576 G        L10002  005370          NCMD.C= 177740 G        RANB    003360 G
G$YES = 000010          L$AUT   002070 G        L10003  006314          NCNT    003340 G        RANBC = 153624 G
HAE     002204 G        L$AUTO  021040 G        L10004  006322          NCNT1   003342 G        RANCMD  023540
HAEM    025756          L$CCP   002106 G        L10005  006330          NEXTSP  025214          RANDOM  003441 G
HALTM   004042 G        L$CLEA  021462 G        L10006  006336          NEXTU   015400 G        RANP  = 000007 G
HELP  = 000000          L$CO    002032 G        L10007  006344          NINUSE= 177774 G        RANRD   023600
HOE   = 100000 G        L$DEPO  002011 G        L10010  017540          NOINTM  004421 G        RANS    003362 G
HRDCNT  003304 G        L$DESC  002136 G        L10012  021036          NRDYM   021424          RANSC = 032561 G
IBE   = 010000 G        L$DESP  002076 G        L10013  021326          NSSRM   004271 G        RANW    023700
IDU   = 000040 G        L$DEVP  002060 G        L10014  021522          NUMBM   026251          RANWR   023654
IER   = 020000 G        L$DISP  002124 G        L10015  021574          NURTY1  005073 G        RANWV   023666
IE.C  = 000200 G        L$DLY   002116 G        L10016  021670          OFLINM  005127 G        RCVERM  026032
INIT10  017550          L$DTP   002040 G        L10017  023256          ONEFIL= 000001          RDF   = 104001 G
INIT15  017764          L$DTYP  002034 G        L10020  022024          OPFLAG  003456 G        RDR   = 104401 G
INIT16  020004          L$DU    021524 G        L10021  022050          OPP.C = 020000 G        RECCNT  003324 G
INTFLG  003416 G        L$DUT   002072 G        L10022  022070          O$APTS= 000000          RECLOG  003411 G
INTPRI= 000340 G        L$DVTY  002164 G        L10023  022110          O$AU  = 000001          RECRED  006312
IRE     003445 G        L$EF    002052 G        L10024  022130          O$BGNR= 000001          RECTAP  006346 G
IREC    002211 G        L$ETP   002102 G        L10025  022150          O$BGNS= 000001          RECUD   011272 G
IRECM   026106          L$EXP1  002044 G        L10026  022170          O$DU  = 000001          REPEAT= 050220
IREM    026153          L$EXP2  002046 G        L10027  022210          O$ERRT= 000000          RERM    004550 G
ISR   = 000100 G        L$EXP4  002064 G        L10030  022230          O$GNSW= 000001          RETRY = 050216
IXE   = 004000 G        L$EXP5  002066 G        L10031  022250          O$POIN= 000001          RETRYC  003404 G
I$AU  = 000041          L$HARD  025074 G        L10032  022306          O$SETU= 000001          REWRT   013722
I$AUTO= 000041          L$HIME  002120 G        L10033  022472          PASCNT  003254 G        RFBC    002654 G
I$CLN = 000041          L$HPCP  002016 G        L10034  023732          PATCH   026350 G        RFCERM  004254 G
I$DU  = 000041          L$HPTP  002022 G        L10035  024076          PATERN  003374 G        RFREC   002754 G
I$HRD = 000041          L$HW    002174 G        L10036  024230          PATR0   010114 G        RFUNR   002764 G
I$INIT= 000041          L$ICP   002104 G        L10037  025070          PATR1   010152 G        RLEXM   004310 G
I$MOD = 000041          L$INIT  017550 G        L10040  025144          PATR2   010172 G        RNF   = 125401 G
I$MSG = 000041          L$LADP  002026 G        L10041  026350          PATR3   010202 G        RNOPSC= 177740 G
I$PROT= 000040          L$LAST  026554 G        L10042  026566          PATR4   010226 G        RNR   = 105401 G
I$PTAB= 000041          L$LOAD  002100 G        L10044  026572          PATR5   010240 G        RNYM    004504 G
I$PWR = 000041          L$LUN   002074 C        MBR   = 100012 G        PATR6   010252 G        RPF   = 105001 G
I$RPT = 000041          L$MREV  002050 G        MEMOM   020734          PATR7   010272 G        RPR   = 125001 G
I$SEG = 000041          L$NAME  002000 G        MISCFG  003455 G        PATR8   010324 G        RPTCNT  003406 G
I$SETU= 000041          L$PRIO  002042 G        MOD.C0= 000400 G        PATTBL  010072          RPTFLG  003443 G
I$SFT = 000041          L$PROT  017542 G        MOD.C1= 001000 G        PATTM   026271          RPT1A   016656
I$SRV = 000041          L$PRT   002112 G        MOD.C2= 002000 G        PCMDWD  003352 G        RPT1B   016733
I$SUB = 000041          L$REPP  002062 G        MOD.C3= 004000 G        PIRE    002214 G        RPT1C   017004
I$ST  = 000041          L$REV   002010 G        MOVMSG  011206 G        PNT   = 001000 G        RPT1D   017055
JLOC    003372 G        L$RPT   016014 G        MSGCNT= 000016 G        PRI   = 002000 G        RPT1E   017303
JLOOP   003370 G        L$SOFT  025146 G        MSGPKA  002502 G        PRI00 = 000000 G        RPT1F   017161
JMP   = 000040 G        L$SPC   002056 G        MSGPKT  002334 G        PRI01 = 000040 G        RPT1G   017232
```

```
RPT1I   017427          TCC6    012560 G        T$PTHV= 000001          VFEXC   014402 G        $FSNAM= 000160
RPT1J   017333          TCC7    012722 G        T$PTNU= 000001          VFISU   014630 G        $FSNO = 000403
RPT1K   017420          TC2RTN  012026          T$SAVL= 177777          VFYCNT  003274 G        $FSOR = 000320
RRANV   002203 G        TIME1   003364 G        T$SEGL= 177777          VFYDAT  014316 G        $FSRTN= 000300
RRBC    002614 G        TIME2   003366 G        T$SIZE= 000007          VFYFLG  003442 G        $FSSEL= 000140
RRECL = 000020 G        TOERM   004207 G        T$SUBN= 000000          VFY.C = 000100 G        $FSTHE= 000330
RRREC   002734 G        TOOMM   004460 G        T$TAGL= 177777          WLKZRO  010206          $FSTRU= 000404
RRUNR   002744 G        TRAPD4  003454 G        T$TAGN= 010045          WRBC    002554 G        $FSUNT= 000130
RRVM    025727          TRAP4   021454 G        T$TEMP= 000000          WRECL = 000020 G        $FSWHI= 000120
RTLE    012740 G        TSAM    004436 G        T$TEST= 000005          WRR   = 105005 G        $FSYES= 000402
RTLRTN  013064          TSBA  = 002452 G        T$TSTM= 177777          WRREC   002714 G        $IFLEV= 177777
RWCPK   002330 G        TSC.FC= 177717 G        T$TSTS= 000001          WRT   = 104005 G        $ISK0 = 000001
RWD   = 102010 G        TSC.TC= 177761 G        T$$AU = 010016          WRTY    013066 G        $ISK1 = 000001
RWERR   003413 G        TSDB    002452 G        T$$AUT= 010013          WRTYCT  003244 G        $ISK2 = 000001
R5SAVE  003400 G        TSSR    002462 G        T$$CLE= 010014          WRTYER  003410 G        $ISK3 = 000001
SCCNT   003264 G        TSSREG  003402 G        T$$DAT= 010044          WRTYFG  003407 G        $ISK4 = 000001
SCERM   004230 G        TSVCT   002472 G        T$$DU = 010015          WRUNR   002724 G        $ISK5 = 000001
SCH   = 140004 G        TS.A16= 000400 G        T$$HAR= 010040          WSSR    011152 G        $ISK6 = 000001
SCHBK   002442 G        TS.A17= 001000 G        T$$HW = 010000          WTM   = 100011 G        $LOCTA= 177777
SCHCNT= 000010 G        TS.NBA= 002000 G        T$$INI= 010012          WTR   = 101011 G        $LSTCN= 177777
SEQEND  003550 G        TS.NXM= 004000 G        T$$MSG= 010003          WTV   = 104105 G        $LSTIN= 000001
SETCH   006444 G        TS.OFL= 000100 G        T$$PC = 000001          WTVERM  004164 G        $LSTST= 177777
SETRW   006470 G        TS.RMR= 010000 G        T$$PRO= 010011          WTYBRF  013340          $LSTTA= 000001
SETUP   007452 G        TS.SC = 100000 G        T$$PTA= 010043          WTYCMD  013334          $MCALL= 000000
SFF   = 105010 G        TS.SPE= 020000 G        T$$RPT= 010010          WTYWRD  013336          $NESTL= 177777
SFPTBL  002202 G        TS.SSR= 000200 G        T$$SOF= 010041          X%ALWA= 000000          $NSK0 = 000120
SFR   = 105410 G        TS.UPE= 040000 G        T$$SRV= 010007          X$FALS= 000040          $NSK1 = 000120
SRF   = 104010 G        TS4ADR  025120          T$$SUB= 010033          X$OFFS= 000400          $NSK2 = 000110
SRR   = 104410 G        TS4CL   002522 G        T$$SW = 010001          X$TRUE= 000020          $NSK3 = 000110
STAERM  005372 G        TS4INT  002512 G        T$$TES= 010037          X0.BOT= 000002 G        $NSK4 = 000110
STAER1  005704          TS4IN0  006316 G        T1      021672 G        X0.EOT= 000001 G        $NSK5 = 000110
STAER2  006062          TS4IN1  006324 G        T1SWB   003447 G        X0.LET= 020000 G        $NSK6 = 000110
STAER3  006141          TS4IN2  006332 G        T1.1    021702          X0.ONL= 000100 G        $SAVLE= 177777
STAER4  006177          TS4IN3  006340 G        T1.10   022232          X0.RLL= 010000 G        $SSK0 = 050432
STAER5  006217          TS4VCT  025135          T1.11   022252          X0.RLS= 040000 G        $TAGLE= 177777
STAER6  006026          T$ARGC= 000003          T1.12   022336          X0.TMK= 100000 G        $TAGNU= 050450
STAER7  005776          T$CODE= 023004          T1.2    022026          X2.OPM= 100000 G        $TEMP = 000402
STAFLG  003452 G        T$ERRN= 000001          T1.3    022052          X3.DCK= 000010 G        $TSK0 = 050431
SVCGBL= 000000          T$EXCP= 000000          T1.4    022072          X3.RNY= 157700 G        $TSK1 = 050432
SVCINS= 000001          T$FLAG= 000041          T1.5    022112          ZROPAT  010156          $TSK2 = 050433
SVCSUB= 000000          T$FREE= 026572          T1.6    022132          $BGNLE= 177777          $TSK3 = 050434
SVCTAG= 000000          T$GMAN= 000000          T1.7    022152          $ERFLG= 000400          $TSK4 = 050447
SVCTST= 000000          T$HILI= 000010          T1.8    022172          $FSAND= 000310          $TSK5 = 050446
SWBFLG  003444 G        T$LAST= 000001          T1.9    022212          $FSBAD= 000401          $TSK6 = 050442
SWB.C = 010000 G        T$LOLI= 000000          T2      023260 G        $FSBLA= 000170          $TSK7 = 050444
S$LSYM= 010000          T$LSYM= 010000          T3      023734 G        $FSCAS= 000150          $$ARGC= 000000
TCCRA   011644          T$LTNO= 000005          T4      024100 G        $FSDEC= 000220          $$BYTE= 000403
TCC0    011664 G        T$NEST= 177777          T5      024232 G        $FSDO = 000340          $$CASE= 000000
TCC1    011702 G        T$NS0 = 000000          T5WEOT  024764          $FSFAL= 000405          $$DST = 000000
TCC2    011720 G        T$NS1 = 000005          UAM   = 000200 G        $FSGOO= 000400          $$ELOC= 000402
TCC3    012030 G        T$NS2 = 000002          UNL   = 100412 G        $FSIF = 000110          $$ERFL= 000000
TCC4    012046 G        T$PCNT= 000000          UNREC   003414 G        $FSINC= 000210          $$FLAG= 000001
TCC5    012462 G        T$PTAB= 010043          URERM   004572 G        $FSLOO= 000200          $$FROM= 000000
```

```
$$LOC = 024700          $$RETU= 000000          $$SRC = 000000          $$TGS2= 000000          .     = 026572
$$LOCN= 000000          $$RTN1= 000000          $$TGSV= 000000          $$TO  = 000000
$$REG = 177777          $$RTN2= 000000          $$TGS1= 000000          $$$TAG= 050000

. ABS.  026572     000


 ERRORS DETECTED:  0

 CZTSHC,CZTSHC/SOL/EQ:ONEFILE=NLISTF.P11,SVC.SML,SPMAC.SML,LISTF.P11,CZTSHC.P11
 RUN-TIME: 149 156 .9 SECONDS
 RUN-TIME RATIO: 530/307=1.7
 CORE USED:  28K  (56 PAGES)
```