TU80

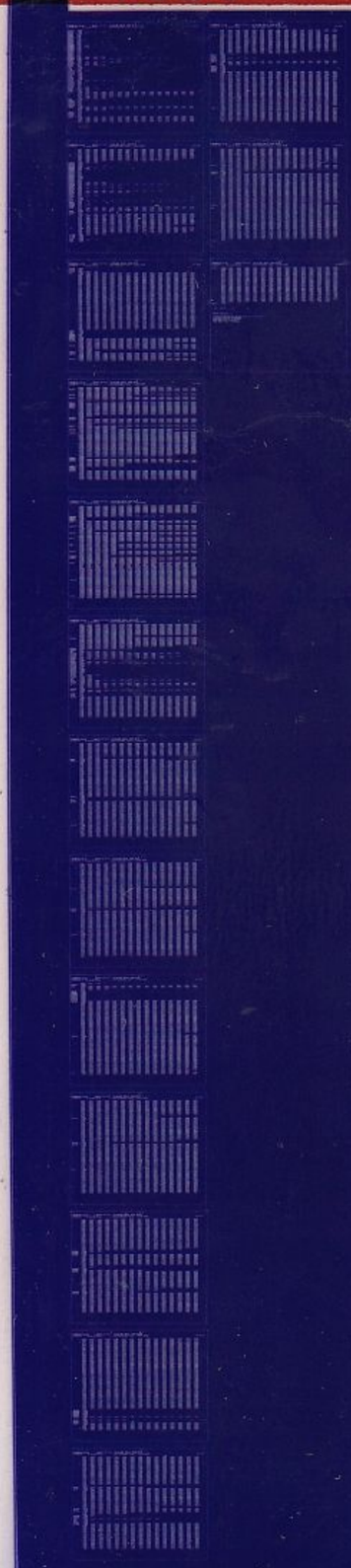
TU80 DATA RELIAB
CZTUVBO

AH-T264B-MC
FICHE 1 OF 2

TU80

TU80 DATA RELIAB
CZTUVBO

AH-T264B-MC
FICHE 2 OF 2



```
.REM &

                    IDENTIFICATION
                    --------------

PRODUCT CODE:   AC-T263B-MC

PRODUCT NAME:   CZTUVB0 TUc0 DATA RELIABILITY TEST

PRODUCT DATE:   11 - JULY - 1983

MAINTAINER:     TAPE DIAGNOSTIC ENGINEERING

AUTHOR:         ROBERT F. WERY/JACK RICHARDSON/TERRENCE REILLY
```

USER DOCUMENTATION TABLE OF CONTENTS

```
                3.0     ERROR INFORMATION

                        3.1     ERROR REPORTING

                                3.1.1   ERROR #1  - COMMAND PACKET ADDRESS IS NOT ON A M
                                3.1.2   ERROR #2  - TU80 NOT READY
                                3.1.3   ERROR #3  - NO RESPONSE ERRORS
                                3.1.4   ERROR #4  - NO INTERRUPT ERROR
                                3.1.5   SPECIAL CONDITION ERRORS

                                        3.1.5.1  ERROR #5 - TCC0, UNDEFINED SPECIAL COND
                                        3.1.5.2  ERROR #6 - TCC1, ATTENTION CONDITION
                                        3.1.5.3  ERROR #7 - TCC2, TAPE STATUS ALERT
                                        3.1.5.4  ERROR #8 - TCC3, FUNCTION REJECT
                                        3.1.5.5  ERROR #9 - TCC4, RECOVERABLE ERROR
                                        3.1.5.6  ERROR #10- TCC5, RECOVERABLE ERROR
                                        3.1.5.7  ERROR #11- TCC6, UNRECOVERABLE ERROR
                                        3.1.5.8  ERROR #12- TCC7, FATAL SUBSYSTEM ERROR

                                3.1.6   ERROR #13 - RFC NON-ZERO ERROR
                                3.1.7   ERROR #14 - RETRY LIMIT EXCEEDED
                                3.1.8   ERROR #15 - TOO MANY INTERRUPTS
                                3.1.9   ERROR #16 - CAPSTAN RUNAWAY
                                3.1.10  ERROR #17 - DATA COMPARE ERRORS

                        3.2     ERROR HALTS

                4.0     PERFORMANCE REPORT

                5.0     TEST SUMMARIES

                        5.1     TEST 1 - BASIC FUNCTIONS
                        5.2     TEST 2 - DATA RELIABILITY
                        5.3     TEST 3 - WRITE AND READ STREAMING TEST
                        5.4     TEST 4 - WRITE COMPATIBILITY/WRITE UTILITY
                        5.5     TEST 4 - READ COMPATABILITY/READ UTILITY
                        5.6     TEST 5 - EXECUTE OPERATOR SELECTED COMMAND SEQUENCE

                6.0     DEVICE INFORMATION

                        6.1     GENERAL
                        6.2     UNIBUS INTERFACE SPECIFICATIONS
                        6.3     BIT DEFINITIONS FOR TU80 REGISTERS

                                6.3.1   TU80 REGISTER SUMMARY
                                6.3.2   TU80 STATUS REGISTER (TSSR)
                                6.3.3   EXTENDED STATUS REGISTER 0 (XSTAT0)
                                6.3.4   EXTENDED STATUS REGISTER 1 (XSTAT1)
                                6.3.5   EXTENDED STATUS REGISTER 2 (XSTAT2)
                                6.3.6   EXTENDED STATUS REGISTER 3 (XSTAT3)
```

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 TU80 DEPENDS ON THE TYPE OF COMMAND.
CMD	TU80 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 TU80 TO EXECUTE A COMMAND.
EXTENDED STATUS	FOUR WORDS OF TU80 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 TU80 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 Or 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-1 DIAGNOSTIC PACKAGES AVAILABLE ON MULTIMEDIA.

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

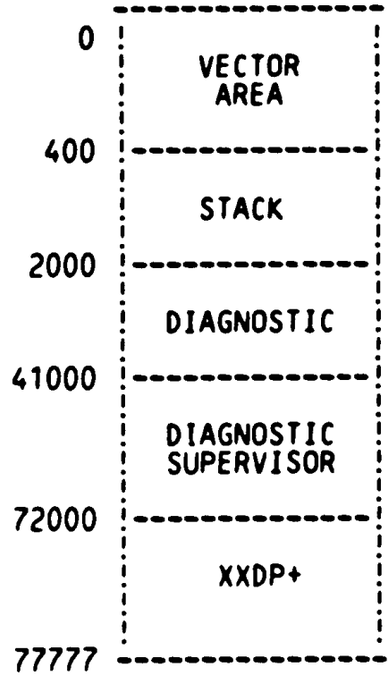
1.1.1 FUNCTIONAL DESCRIPTION

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 6 TESTS IN THIS PROGRAM:

TEST 1 - BASIC FUNCTIONS.
TEST 2 - DATA RELIABILITY.
TEST 3 - WRITE AND READ STREAMING TEST.
TEST 4 - WRITE COMPATABILITY/WRITE UTILITY.
TEST 5 - READ COMPATABILITY/READ UTILITY.
TEST 6 - 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 RECOVER 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.

```
314                     IF ANY OF THE 4 REPEAT FAILS, ERASE BAD RECORD, LOG SUSPECTED
315                     BAD SPOT AT THAT RECORD NUMBER, RETRY AGAIN 3 INCHES FURTHER DOWN TAPE.
316                     RETRY 4 TIMES, UP TO 4 REPEATS EACH.
317                     IF RECORD CANNOT BE WRITTEN WITHOUT RECOVERABLE ERROR  AFTER 4 RETRIES,
318                     ERASE RECORD, REPORT RETRY FAILED ON BAD SPOT.
319                     THE RECOVERABLE ERROR SUBROUTINE THEN CONTINUES TO CALL THE WRITE
320                     RETRY SUBROUTINE, WHICH REISSUES THE GROUP OF 4 RETRIES,
321                     UNTIL THE RECORD IS RECOVERED OR 20 BAD SPOTS HAVE BEEN  LOGGED .
322
323                     TWENTY (20) BAD SPOTS MAXIMUM ARE ALLOWED PER TAPE PASS.
324                     WHEN 20 BAD SPOTS HAVE BEEN LOGGED, ON SAME RECORD NUMBER OR NOT,
325                     TAPE IS CONSIDERED DEFECTIVE:  A BAD TAPE OVERFLOW MESSAGE IS PRINTED
326                     AND UNIT IS REWOUND, THEN DROPPED.
327
328                     DURING THE RECOVERY PROCESS, IT IS NECESSARY TO PERFORM SEVERAL TAPE
329                     POSITION OPERATIONS: SPACE REVERSE, ERASE.  IF A POSITION ERROR STATUS
330                     IS DETECTED DURING THOSE OPERATIONS, THEN THE RECOVERY ATTEMPT IS ABORTE
331                     AN APPROPRIATE UNRECOVERABLE MESSAGE IS PRINTED AND UNIT IS DROPPED.
332
333                     ALL BADLY WRITTEN RECORDS FLAGGED WITH RECOVERABLE ERRORS ARE ERASED
334                     UNTIL RECOVERED, INCLUDING THE RECORD AT THE 20TH BAD SPOT,
335                     SO THAT ALL RECORDS LEFT ON TAPE ARE GOOD WRITTEN RECORDS.
336                     BAD SPOTS ARE ERASED, WITH ERASE GAPS FROM 3 TO 12 INCHES  PER RETRY GRO
337                     UP TO 20 FEET OF ERASE GAP COULD RESULT WHEN RETRYING TO RECOVER
338                     A SINGLE RECORD,IF NO BAD SPOT WERE PREVIOUSLY DETECTED.
339                     THAT LONG STRETCH OF  BAD TAPE WOULD THEN BE FLAGGED WITH 20
340                     BAD SPOTS AT SAME RECORD NUMBER AND THE TAPE CONSIDERED DEFECTIVE.
341
342                     BAD SPOTS REPORTS
343
344                     IF THE PRINT OF RECOVERABLE ERRORS IS ENABLED, THE BAD SPOTS ON TAPE ARE
345                     IDENTIFIED AS THEY ARE DETECTED.  SINCE THE BAD RECORDS ARE ERASED UNTIL
346                     THE BAD SPOTS ACTUALLY PRECEDES THE RECORD NUMBER THAT IDENTIFIES THEM.
347                     THE NUMBER OF REPEATS AND RETRIES ATTEMPTED IS PRINTED, FROM WHICH THE
348                     LENGTH OF ERASE GAPS CAN BE DETERMINED:  APPROXIMATELY 3 INCHES PER RETR
349
350                     THE STATISTICAL REPORT PRINTED AT THE END OF TEST 2 OR UPON A "PRINT" RE
351                     CONTAINS A SUMMARY OF THE BAD SPOTS LOGGED ON THE CURRENT TAPE PASS.
352                     IN THAT REPORT, ALL COUNTS ARE CUMULATIVE FROM PASS TO PASS, EXCEPT FOR
353                     THE NUMBER OF BAD SPOTS: IT RELATES TO A "TAPE PASS" ONLY.
354                     FOR THIS PURPOSE, A "TAPE PASS" IS A WRITE PASS FROM BOT TO EOT, OR FROM
355                     BOT TO WHERE THE DIAGNOSTIC IS HALTED BEFORE REACHING EOT.
356                     A PASS IS DEFINED BY THE SUPERVISOR AS A RUN THROUGH ALL THE TESTS REQUE
357                     ON ALL UNITS SELECTED. THOSE PASSES ARE IDENTIFIED AS "PASS" AND "EOP".
358
359                     THE NUMBER OF WRITE RETRIES, CUMULATIVE FROM PASS TO PASS, IS A GLOBAL
360                     COUNT OF HOW MANY TIMES THE GROUP OF 4 RETRIES HAS BEEN CALLED.
361
362                     THE NUMBER OF WRITE RECOVERABLE ERRORS EXCLUDES BAD TAPE SPOTS
363                     AND REFLECTS THE SPECIFICATIONS OF THE HARDWARE UNDER TEST.
364                     PER TAPE PASS, THE NUMBER OF WRITE RETRIES EQUALS THE SUM OF THE
365                     NUMBER OF RECOVERABLE WRITE ERRORS AND BAD SPOTS, MOST OF THE TIME.
366
367                     TO CLEAR CUMULATIVE COUNTS, ANSWER 'Y' TO:   CLEAR COUNTERS (L) Y ?.
368                     BAD TAPE SPOTS COUNT IS CLEARED WHEN WRITING FROM BOT.
369
```

```
                        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:
                        CZTUV 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

                        CZTUV 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  100210  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
                                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 TU80 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 000000 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
                TU80 DRIVE AND UNIBUS ADAPTER MODULE
```

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.
TU80 PROGRAMMING SPECIFICATION.
TU80 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 - 23-MAR-83 - ORIGINAL RELEASE
REVISION B - 17-APR-83 - FIX TEST 3 FOR NEW SPEED ALGORITHM

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 ADRRES 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 RESPONDING 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)

```
622
623
624
625
626
627
628
629
630
631
632
633
634
```

```
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 OPERATOR 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 DIAL
```

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 CHEC 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 TU80 PROGRAMMING SPECIFICATION FOR DESCRIPTION.
24 =	NOT USED	
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 FI
26 =	DLY	DELAY "N" MILISECONDS WHERE N IS DEFINED IN THE # OF OPERATIONS. THIS DELAY IS USED BETWEEN EACH EXECUTABLE COMMAND.
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 COMPL
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>
```

```
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
```

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
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 B
BRF (D) 2048 ?                        1000<CR>
# OF OPERATIONS (D) 10 ?              10000<CR>
PATTERN (D) 7 ?                          1<CR>
CMD/4 (D) 5 ?                           27<CR>   (END)   (COULD B
BRF (D) 2048 ?                          <^Z>
```

2.4 EXECUTION TIMES

2.4.1 SYSTEM CONFIGURATION

PDP11/34
MOS MEMORY
LA36
TU80

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/READ STREAMING TEST - 15 MINUTES PER UNIT.
TEST 4 - WRITE COMPATABILITY - 20 MINUTES PER PASS.
TEST 5 - READ COMPATABILITY - 20 MINUTES PER PASS.
TEST 6 - 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 EXCEPT WITH TEST 3, WHERE EACH
ADDITIONAL UNIT ADDS 8 MINUTES TO THE EXECUTION TIME.

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:

```
CZTUV 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 DIAGNOST
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
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 DI
TO GET A PRINT OF THE RECORD COUNT WHEN HALTED.

EXAMPLE OF AN ERROR REPORT:

```
CZTUV 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 - TU80 NOT READY:

BEFORE ANY COMMAND IS ISSUED TO THE TU80 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 TU80 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 IS 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 INFOR-
MATION 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.

```
1008                                    4.0 PERFORMANCE REPORT
1009                                        ------------------
1010
1011                                        UNIT X   PASS:XXXXX  RECORD:XXXXX
1012                                        BYTES WRITTEN  XXX,XXX,XXX,XXX
1013                                        BYTES READ REV XXX,XXX,XXX,XXX
1014                                        BYTES READ FWD XXX,XXX,XXX,XXX
1015                                                          WRT       RDR     RDF
1016                                        RECOVERABLE ERRORS    XXXXX     XXXXX   XXXXX
1017                                        UNRECOVERABLE ERRORS  XXXXX     XXXXX   XXXXX
1018
1019                                        SPEC COND   HARD   FATAL   COMPARE
1020                                           XXXXX   XXXXX   XXXXX   XXXXX
1021
1022
1023
1024
1025
1026                                    5.0 TEST SUMMARIES
1027                                        --------------
1028
1029                                    5.1     TEST 1 -        BASIC FUNCTIONS.
1030
1031                                                            EXECUTES AND VERIFIES CORRECT COMPLETION OF ALL TUBO FUN
1032
1033                                                            SUBTEST 1  - SET CHAR, DRIVE INIT, GET STATUS.
1034                                                                         + SET CHARACTERISTIC 200.
1035                                                                         + DRIVE INITIATE.
1036                                                                         + SET CHARACTERISTIC  20.
1037                                                                         + GET STATUS
1038                                                                         + SET CHARACTERISTIC  40.
1039
1040                                                            SUBTEST 2  - REWIND.
1041                                                                         + REWIND.
1042                                                                         + REWIND AT BOT.
1043
1044                                                            SUBTEST 3  - WRITE/VERIFY.
1045                                                                         + WRITE/VERIFY PATTERN 1.
1046                                                                         + WRITE/VERIFY PATTERN 2.
1047                                                                         + WRITE/VERIFY PATTERN 3.
1048                                                                         + WRITE/VERIFY PATTERN 4.
1049                                                                         + WRITE/VERIFY PATTERN 5.
1050                                                                         + WRITE/VERIFY PATTERN 6.
1051                                                                         + WRITE/VERIFY PATTERN 0.
1052
1053                                                            SUBTEST 4  - WRITE TAPE MARK, ERASE.
1054                                                                         + WRITE TAPE MARK.
1055                                                                         + WRITE 10 RECORDS
1056                                                                         + ERASE 10 TIMES
1057                                                                         + WRITE TAPE MARK.
1058                                                                         + WRITE TAPE MARK RETRY.
1059
1060                                                            SUBTEST 5  - SPACE FILES.
1061                                                                         + SPACE 2 FILES REVERSE.
1062                                                                         + SPACE 2 FILES FORWARD.
1063                                                                         + 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.
```

```
1112                                    5.2     TEST 2 -        DATA RELIABILITY.
1113
1114                                                              1. THE TAPE IS INITIATED WITH THE FOLLOWING COMMANDS:
1115                                                                    SET CHARACTERISTIC 40
1116                                                                    REWIND
1117                                                                    WRITE/VERIFY 31 RECORDS OF RANDOM LENGTH AND DAT
1118                                                              2. WRITE AND READ COMMANDS ARE SELECTED AT RANDOM AND
1119                                                                 EXECUTED A RANDOM NUMBER OF TIMES WITH RANDOM
1120                                                                 LENGTHS AND RANDOM PATTERN UNTIL END OF TAPE IS REA
1121                                                              3. AT THE END OF EACH PASS, A REWIND COMMAND IS ISSUED
1122                                                                 A PERFORMANCE REPORT IS PRINTED.
1123
1124                                                                    NOTE:   IF A RESTART COMMAND IS USED TO INITIATE
1125                                                                            TEST 1, THE INITIAL REWIND COMMAND IS NO
1126
1127                                    5.3     TEST 3 -        WRITE AND READ STREAMING TEST.
1128
1129                                                              1. REWINDS ALL UNITS, THEN ON EACH UNIT:
1130
1131                                                                           >>>  REPEAT TWICE  <<<
1132
1133                                                              ************************************************************
1134
1135                                                              2. WRITE PATTERN 5 FOR 7000 - 1 KBYTE RECORDS.
1136                                                                 THE SPEED ALGORITHM IN THE TU80 WILL ENABLE 100 IPS S
1137                                                              3. READ REVERSE FOR 3500 RECORDS AT 100 IPS STREAMING.
1138                                                              4. READ REVERSE FOR 3500 RECORDS AT 25 IPS STREAMING.
1139                                                              5. READ FORWARD FOR 3500 RECORDS AT 100 IPS STREAMING.
1140                                                              6. READ FORWARD FOR 3500 RECORDS AT 25 IPS STREAMING.
1141                                                              7. WRITE A TAPE MARK
1142
1143                                                              ************************************************************
1144
1145                                                                           >>>  END REPEAT  <<<
1146
1147                                                              NOTE: 7000(10) RECORDS OF 1000(10) BYTES ARE WRITTEN AND
1148                                                              0/. EACH ITERATION.
1149
1150                                    5.4     TEST 4 -        WRITE COMPATABILITY/WRITE UTILITY.
1151
1152                                                              REWINDS AND WRITES RECORDS OF RANDOM LENGTHS
1153                                                              AND RANDOM DATA FROM BOT TO EOT.
1154
1155
1156                                    5.5     TEST 5 -        READ COMPATABILITY/READ UTILITY.
1157
1158                                                              REWINDS AND READS ENTIRE TAPE, FORWARD AND REVERSE.
1159
1160
1161                                    5.6     TEST 6 -        EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.
1162
1163                                                              THE SEQUENCE OF COMMANDS ENTERED BY THE OPERATOR
1164                                                              IS EXECUTED.  IF NO COMMANDS WERE ENTERED, A
1165                                                              DEFAULT SEQUENCE OF REWIND/WRITE/READ REV/READ FWD/REWIN
1166                                                              OF ENTIRE TAPE IS EXECUTED WITH RANDOM PATTERN
1167                                                              AND RECORD LENGTH OF 2048 BYTES.
```

6.0 DEVICE INFORMATION TABLES

6.1 GENERAL

THE TU80 SUBSYSTEM IS A REEL-TO-REEL TAPE TRANSPORT WITH AN INTEGRATED FORMATTER. AN ADAPTER MODULE CONNECTS THE DRIVE TO ANY PROCESSOR WITH A UNIBUS. THE SUBSYSTEM READS AND WRITES AT EITHER 25 OR 100 IPS AND RECORDS AT 1600 BPI, PHASE ENCODED, ANSI STANDARD FORMAT. TAPE SPEEDS OF 25 OR 100 IPS ARE AUTOMATICALLY SELECTED BY THE CONTROLLER ACCORDING TO A SPEED ALGORITHM WITHIN THE CONTROLLERS' FIRMWARE. THE HOST SOFTWARE MAY SELECT THE 100 IPS SPEED; THE 25 IPS SPEED IS NOT USER-SELECTABLE. IN ADDITION, THE DRIVE IS MICROPROCESSOR CONTROLLED AND HAS SELF-CONTAINED LOGIC AND FUNCTIONAL DIAGNOSTICS. THE ADAPTER MODULE ENABLES THE TU80 TO BE 100% PROTOCOL COMPATIBLE WITH THE TS11.

OPERATIONALLY, THE TU80 IS UNIQUE 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 COUNT

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 TU80 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

TU80	INT. VECTOR	UNIBUS ADDRESS	REGISTER
FIRST	224	772520	TSBA/TSDB
		772522	TSSR
SECOND	FLOAT	772524	TSBA/TSDB
		772526	TSSR
THIRD	FLOAT	772530	TSBA/TSDB
		772532	TSSR
FOURTH	FLOAT	772534	TSBA/TSDB
		772536	TSSR

6.3 BIT DEFINITIONS FOR TU80 REGISTERS

6.3.1 TU80 REGISTER SUMMARY

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01
TSBA	A15	A14	A13	A12	A11	A10	A09	A08	A07	A06	A05	A04	A03	A02	A0
TSDB	P15	P14	P13	P12	P11	P10	P09	P08	P07	P06	P05	P04	P03	P02	P1
TSSR	SC			RMR	NXM	NBA	A17	A16	SSR	OFL	FC1	FC0	TC2	TC1	TC
XST0	TMK	RLS	LET	RLL	WLE	NEF	ILC	ILA		ONL	IE	VCK	PED	WLK	BO
XST1	DLT			COR	CRS		DBF								UN CR
XST2	OPM	SIP	BPE	CAF		WCF		DTP	DT7	DT6	DT5	DT4	DT3	DT2	DT
XST3	MICRO DIAGNOSTIC ERROR CODE									OPI	REV		DCK		LX

TERMINATION CLASS CODES (TSSR TC0-TC2):

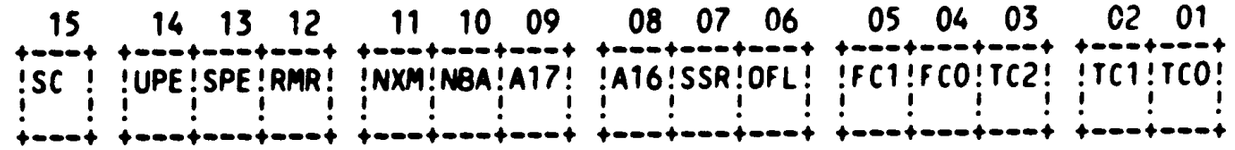
0 = NORMAL TERMINATION
1 = ATTENTION CONDITION
2 = TAPE STATUS ALERT
3 = FUNCTION REJECT
4 = RECOVERABLE ERROR - TAPE POSITION = ONE RECORD
DOWN TAPE FROM START OF FUNCTION
5 = RECOVERABLE ERROR - TAPE NOT MOVED
6 = UNRECOVERABLE ERROR - TAPE POSITION LOST
7 = FATAL CONTROLLER ERROR

FATAL CLASS CODES (TSSR FC0-FC1):

0 = MICRO DIAGNOSTIC FAILURE (DISPLAYED IN TU80 OPERATOR PANEL AND
1 = I/O SEQUENCER CROM PARITY ERROR.
2 = MICROPROCESSOR CROM PARITY ERROR.
SILO PARITY ERROR.
SERIAL BUS PARITY ERROR DETECTED AT TU80 (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 TU80 STATUS REGISTER (TSSR)

UNIBUS ADDRESS + 2 - READ ONLY



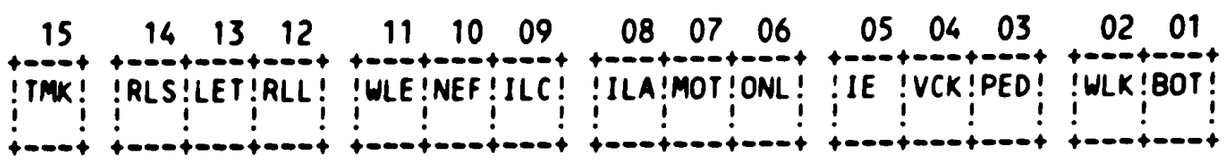
BIT	NAME	TCC	DEFINITION
15	SC	S	SPECIAL CONDITION. WHEN SET, INDICATES THAT THE LAST COMMAND DID NOT COMPLETE WITHOUT INCIDENT. SPECIFICALLY, EITHER AN ERROR WAS DETECTED OR AN EXCEPTION CONDITION OCCURRED. EXCEPTION CONDITIONS CAN BE TAPE MARKS ON READ COMMANDS, REVERSE MOTION AND AT BOT, EOT WHILE 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 SOMET OTHER THAN 0 (UNLESS RMR IS THE ONLY ERROR - SEE RM
14	NOT USED		
13	NOT USED		
12	RMR	S	REGISTER MODIFICATION REFUSED. SET BY THE TU80 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 TERMINATIO 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 TU80 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 REGISTER.
	08	A16	S	BUS ADDRESS BIT 16. SEE A17 (BIT 09).
	07	SSR	S	SUB-SYSTEM READY. WHEN SET, INDICATES THAT THE TU80 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

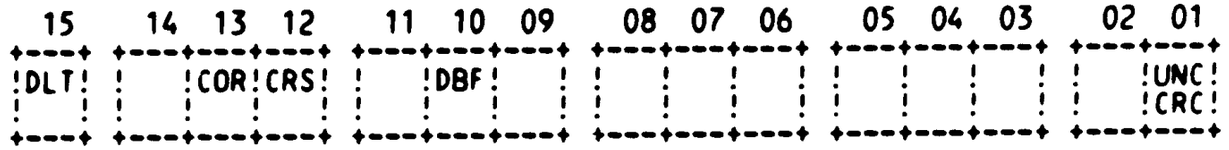
6.3.3 EXTENDED STATUS REGISTER 0 (XSTAT0)



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.
				- 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	NOT USED		
	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	NOT USED		
10	DBF	4	DESKEW BUFFER FAIL: THIS BIT IS SET WHEN ONE OF THE DESKEW BUFFERS FAILS TO SET OUTPUT READY WITHIN 20 MICROSECONDS AFTER BEING ENABLED. THE DEAD TRACK BITS INDICATE ON WHICH TRACKS THIS FAILURE OCCURRED.
09	NOT USED		
08	NOT USED		
07	NOT USED		
06	NOT USED		
05	NOT USED		
04	NOT USED		
03	NOT USED		
02	NOT USED		

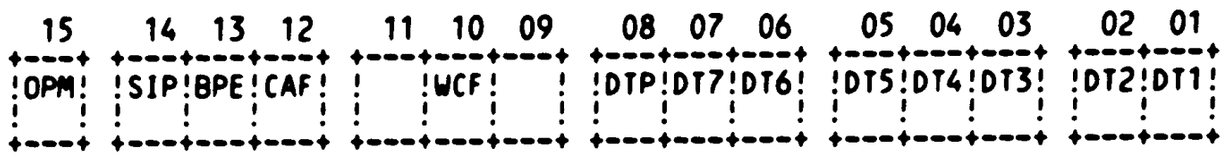
```
                                01   UNC     4       UNCORRECTABLE DATA.  SET ON PE  DRIVES  WHEN  A
                                                     PARITY  ERROR  OCCURRED WITHOUT A CORRESPONDING
                                                     DEAD TRACK INDICATION.

                                     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.
```

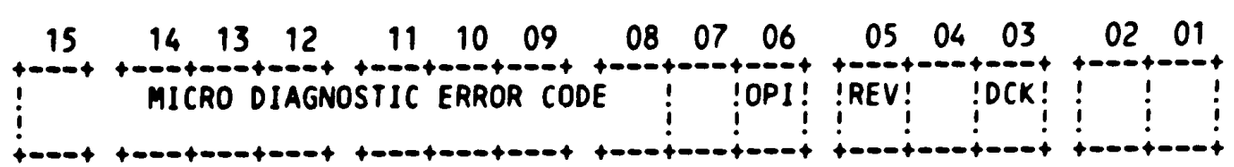
6.3.5 EXTENDED STATUS REGISTER 2 (XSTAT2)



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 TU80 WHEN A PARITY ERROR IS DETECTED ON DATA TRANSMITTED FROM THE ADAPTOR CARD TO THE TU80. CAUSES FA 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.

```
1649
1650                                    02   DT2    S       DEAD TRACK 2.  SEE DTP.
1651
1652                                    01   DT1    S       DEAD TRACK 1.  SEE DTP.
1653
1654                                    00   DT0    S       DEAD TRACK 0.  SEE DTP.
1655
1656                                NOTE:  ON A SET CHARACTERISTICS COMMAND, THE UCODE LEVEL  IS  RETURNED
1657                                       IN DT7 THRU DT0.  ON A GET STATUS COMMAND, THE RESIDUAL CAPSTAN
1658                                       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		7	MICRO DIAGNOSTIC ERROR CODE. THERE IS ONE OPERATIONAL ERROR. 337(8) OR LEFT JUSTIFIED TO 157400(8) IN A 16-BIT REGISTER (CAPSTAN RUNAWAY), WHICH IS DISPLAYED HERE. THIS MEANS THAT THE CAPSTAN WAS COMMANDED TO STOP BUT EXCEEDED THE ALLOWABLE STOPPING WINDOW. DRIVE MUST BE INITIALIZED TO BE USED FOR TAPE MOTION AGAIN.
07	NOT USED		
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	NOT USED		.
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	NOT USED		
01	NOT USED		
00	RIB	2	REVERSE INTO BOT. A READ, SPACE, OR SKIP COMMAND ALREADY IN PROGRESS HAS ENCOUNTERED THE BOT MARKER WHEN MOVING TAPE IN THE REVERSE DIRECTION. TAPE MOTION WILL BE HALTED AT BOT.

&

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

                                        .ENABL  ABS,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  CZTUV,B,0,5000,1,INTPRI
002000                          L$NAME::                ;DIAGNOSTIC NAME
002000  103                                                             .ASCII /C/
002001  132                                                             .ASCII /Z/
002002  124                                                             .ASCII /T/
002003  125                                                             .ASCII /U/
002004  126                                                             .ASCII /V/
002005  000                                                             .BYTE   0
002006  000                                                             .BYTE   0
002007  000                                                             .BYTE   0
002010                          L$REV::                 ;REVISION LEVEL
002010  102                                                             .ASCII  /B/
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$HPCP::                ;POINTER TO H.W. QUES.
002016  027202                                                          .WORD   L$HARD
002020                          L$SPCP::                ;POINTER TO S.W. QUES.
002020  027254                                                          .WORD   L$SOFT
002022                          L$HPTP::                ;PTR. TO DEF. H.W. PTABLE
002022  002176                                                          .WORD   L$HW
002024                          L$SPTP::                ;PTR. TO S.W. PTABLE
002024  002204                                                          .WORD   L$SW
002026                          L$LADP::                ;DIAG. END ADDRESS
002026  030670                                                          .WORD   L$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$DTP::                 ;PTR. TO DISPATCH TABLE
002040  002124                                                          .WORD   L$DISPAT
002042                          L$PRIO.:                ;DIAGNOSTIC RUN PRIORITY
002042  000340                                                          .WORD   INTPRI
002044                          L$ENVI::                ;FLAGS DESCRIBE HOW IT WAS SETUP
```

```
(4)  002044  000000                                                  .WORD   0
(5)  002046                  L$EXP1::          ;EXPANSION WORD
(4)  002046  000000                                                  .WORD   0
(5)  002050                  L$MREV::          ;SVC REV AND EDIT #
(4)  002050     003                                                  .BYTE   C$REVISI
(3)  002051     003                                                  .BYTE   C$EDIT
(5)  002052                  L$EF::            ; IAG. EVENT FLAGS
(4)  002052  000000                                                  .WORD   0
(5)  002054  000000                                                  .WORD   0
(5)  002056                  L$SPC::
(4)  002056  000000                                                  .WORD   0
(5)  002060                  L$DEVP::          ; POINTER TO DEVICE TYPE LIST
(4)  002060  002166                                                  .WORD   L$DVTYP
(5)  002062                  L$REPP::          ;PTR. TO REPORT CODE
(4)  002062  017366                                                  .WORD   L$RPT
(5)  002064                  L$EXP4::
(4)  002064  000000                                                  .WORD   0
(5)  002066                  L$EXP5::
(4)  002066  000000                                                  .WORD   0
(5)  002070                  L$AUT::           ;PTR. TO ADD UNIT CODE
(4)  002070  023246                                                  .WORD   L$AU
(5)  002072                  L$DUT::           ;PTR. TO DROP UNIT CODE
(4)  002072  023202                                                  .WORD   L$DU
(5)  002074                  L$LUN::           ;LUN FOR EXERCISERS TO FILL
(4)  002074  000000                                                  .WORD   0
(5)  002076                  L$DESP::          ;POINTER TO DIAG. DESCRIPTION
(4)  002076  002140                                                  .WORD   L$DESC
(5)  002100                  L$LOAD::          ;GENERATE SPECIAL AUTOLOAD EMT
(4)  002100  104035                                                  EMT     E$LOAD
(5)  002102                  L$ETP::           ;POINTER TO ERRTBL
(4)  002102  000000                                                  .WORD   0
(5)  002104                  L$ICP::           ;PTR. TO INIT CODE
(4)  002104  021222                                                  .WORD   L$INIT
(5)  002106                  L$CCP::           ;PTR. TO CLEAN-UP CODE
(4)  002106  023140                                                  .WORD   L$CLEAN
(5)  002110                  L$ACP::           ;PTR. TO AUTO CODE
(4)  002110  022516                                                  .WORD   L$AUTO
(5)  002112                  L$PRT::           ;PTR. TO PROTECT TABLE
(4)  002112  021214                                                  .WORD   L$PROT
(5)  002114                  L$TEST::          ;TEST NUMBER
(4)  002114  000000                                                  .WORD   0
(5)  002116                  L$DLY::           ;DELAY COUNT
(4)  002116  000000                                                  .WORD   0
(5)  002120                  L$HIME::          ;PTR. TO HIGH MEM
(4)  002120  000000                                                  .WORD   0
1776
```

```
                                        .SBTTL  DISPATCH TABLE

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

                                                DISPATCH 6              ; SIX TESTS
                                                                                        .WORD   6
                                        L$DISPATCH::
                                                                                        .WORD   T1
                                                                                        .WORD   T2
                                                                                        .WORD   T3
                                                                                        .WORD   T4
                                                                                        .WORD   T5
                                                                                        .WORD   T6


                                        .SBTTL  DESCRIPTIVE TEXT

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

                                                DESCRIPT        <DATA RELIABILITY TEST>
                                        L$DESC::
                                                                                        .ASCIZ  /DATA RE

                                                                                        .EVEN

                                                        DEVTYP          <TU80>
                                                L$DVTYP::
                                                                                        .ASCIZ  /TU80/
                                                                                        .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.
                                        ;--

002174                                          BGNHW   DFPTBL
(3) 002174  000002                                                      .WORD   L10000-L
(3) 002176                              L$HW::
(3) 002176                              DFPTBL::

002176  172522                          172522          ;TSSR ADDRESS.
002200  000224                          224             ;VECTOR ADDRESS.

002202                                          ENDHW
(3) 002202                              L10000:
```

```
                                .SBTTL  SOFTWARE P-TABLE

                                ;++
                                ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
                                ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
                                ;--
                                        BGNSW   SFPTBL
 1832
 1833
 1834
 1835
 1836
 1837
 1838  002202                                                                 .WORD   L10001-L
  (3)  002202  000043
  (3)  002204                   L$SW::
  (3)  002204                   SFPTBL::
 1845  002204      001          CLRFLG::.BYTE   1               ;CLEAR COUNTERS FLAG.
 1846  002205      000          RRANV:: .BYTE   0               ;RESET RANDOM VARIABLES EACH PASS FLAG.
 1847  002206      000          HAE::   .BYTE   0               ;HALT AFTER EACH COMMAND FLAG.
 1848  002207      000          ERCVER::.BYTE   0               ;ENABLE RECOVERABLE ERROR PRINTS FLAG.
 1849  002210      000          IREC::  .BYTE   0               ;INHIBIT ERROR RECOVERY FLAG.
 1850  002211      001          BADTSW::.BYTE   1               ;BAD TAPE SWITCH TO REWRITE ON SAME SPOT & DETEC
 1851  002212      000          DINT::  .BYTE   0               ;DISABLE INTERRUPTS FLAG.
 1852  002213      000          PIRE::  .BYTE   0               ;INHIBIT RESIDUAL FRAMECOUNT ERROR REPORT FLAG.
 1853  002214      000          RAMWRT::.BYTE   0               ;ENABLE OPTIONAL RAM DUMP
 1854  002215      000          CHGFLG::.BYTE   0               ;CHANGE CMD SEQ TABLE FLAG.
 1855  002216      000                  .BYTE   0               ;SPARE
 1856  002217      000                  .BYTE   0               ;SPARE
 1857                                    .EVEN
 1858  002220  000040           CHAR::  CH.EAI                  ;CHARACTERISTICS CODE (DEFAULT = 40).
 1859  002222  000015           CMDD::  .WORD   13.             ;COMMAND 2 (DEFAULT = REWIND).
 1860  002224  000001                   .WORD   1               ;BYTE COUNT
 1861  002226  000001                   .WORD   1               ;NUMBER OF OPERATIONS
 1862  002230  000007                   .WORD   RANP            ;PATTERN
 1863  002232  000004                   .WORD   4               ;COMMAND 3  (DEFAULT = WRITE)
 1864  002234  004000                   .WORD   DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
 1865  002236  076400                   .WORD   32000.          ;NUMBER OF OPERATIONS (DEFAULT = 32000).
 1866  002240  000007                   .WORD   RANP            ;PATTERN (DEFAULT = RANDOM).
 1867  002242  000003                   .WORD   3               ;COMMAND 4 (DEFAULT = READ REV).
 1868  002244  004000                   .WORD   DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
 1869  002246  076400                   .WORD   32000.          ;NUMBER OF OPERATIONS (DEFAULT = 32,000).
 1870  002250  000007                   .WORD   RANP            ;PATTERN (DEFAULT = RANDOM).
 1871  002252  000002                   .WORD   2               ;COMMAND 5 (DEFAULT = READ FWD).
 1872  002254  004000                   .WORD   DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
 1873  002256  076400                   .WORD   32000.          ;NUMBER OF OPERATIONS (DEFAULT = 32,000).
 1874  002260  000007                   .WORD   RANP            ;PATTERN (DEFAULT = RANDOM).
 1875  002262  000015                   .WORD   13.             ;COMMAND 6 (DEFAULT = REWIND).
 1876  002264  000001                   .WORD   1.              ;BYTE COUNT
 1877  002266  000001                   .WORD   1               ;NUMBER OF OPERATIONS
 1878  002270  000007                   .WORD   RANP            ;PATTERN
 1879  002272  000033                   .WORD   27.             ;END OF CMD SEQ TABLE CODE (DEF) OR CMD 7
 1880  002274  004000                   .WORD   DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
 1881  002276  076400                   .WORD   32000.          ;NUMBER OF OPERATIONS (DEFAULT = 32000).
 1882  002300  000007                   .WORD   RANP            ;PATTERN (DEFAULT = RANDOM).
 1883  002302  000033                   .WORD   27.             ;END OF CMD SEQ TABLE CODE (DEF) OR CMD 8
 1884  002304  004000                   .WORD   DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
 1885  002306  076400                   .WORD   32000.          ;NUMBER OF OPERATIONS (DEFAULT = 32000).
 1886  002310  000007                   .WORD   RANP            ;PATTERN (DEFAULT = RANDOM).
 1887  002312                           ENDSW
  (3)  002312                   L10001:
 1888  002312                           ENDMOD
```

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

002312                                          BGNMOD

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

002312                                          EQUALS
(1)                                     ;
(1)                                     ; BIT DIFINITIONS
(1)                                     ;
(1)             100000                  BIT15== 100000
(1)             040000                  BIT14== 40000
(1)             020000                  BIT13== 20000
(1)             010000                  BIT12== 10000
(1)             004000                  BIT11== 4000
(1)             002000                  BIT10== 2000
(1)             001000                  BIT09== 1000
(1)             000400                  BIT08== 400
(1)             000200                  BIT07== 200
(1)             000100                  BIT06== 100
(1)             000040                  BIT05== 40
(1)             000020                  BIT04== 20
(1)             000010                  BIT03== 10
(1)             000004                  BIT02== 4
(1)             000002                  BIT01== 2
(1)             000001                  BIT00== 1
(1)                                     ;
(1)             001000                  BIT9==  BIT09
(1)             000400                  BIT8==  BIT08
(1)             000200                  BIT7==  BIT07
(1)             000100                  BIT6==  BIT06
(1)             000040                  BIT5==  BIT05
(1)             000020                  BIT4==  BIT04
(1)             000010                  BIT3==  BIT03
(1)             000004                  BIT2==  BIT02
(1)             000002                  BIT1==  BIT01
(1)             000001                  BIT0==  BIT00
(1)                                     ;
(1)                                     ; EVENT FLAG DEFINITIONS
(1)                                     ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
(1)                                     ;
(1)             000040                  EF.START==      32.                     ; START COMMAND WAS ISSUED
(1)             000037                  EF.RESTART==    31.                     ; RESTART COMMAND WAS ISSUED
(1)             000036                  EF.CONTINUE==   30.                     ; CONTINUE COMMAND WAS ISSUED
(1)             000035                  EF.NEW==        29.                     ; A NEW PASS HAS BEEN STARTED
(1)             000034                  EF.PWR==        28.                     ; A POWER-FAIL/POWER-UP OCCURRED
(1)                                     ;
(1)                                     ;
(1)                                     ; PRIORITY LEVEL DEFINITIONS
(1)                                     ;
(1)             000340                  PRI07== 340
```

```
    (1)          000300                    PRI06== 300
    (1)          000240                    PRI05== 240
    (1)          000200                    PRI04== 200
    (1)          000140                    PRI03== 140
    (1)          000100                    PRI02== 100
    (1)          000040                    PRI01== 40
    (1)          000000                    PRI00== 0
    (1)                                    ;
    (1)                                    ;OPERATOR FLAG BITS
    (1)                                    ;
    (1)          000004                    EVL==        4
    (1)          000010                    LOT==       10
    (1)          000020                    ADR==       20
    (1)          000040                    IDU==       40
    (1)          000100                    ISR==      100
    (1)          000200                    UAM==      200
    (1)          000400                    BOE==      400
    (1)          001000                    PNT==     1000
    (1)          002000                    PRI==     2000
    (1)          004000                    IXE==     4000
    (1)          010000                    IBE==    10000
    (1)          020000                    IER==    20000
    (1)          040000                    LOE==    40000
    (1)          100000                    HOE==   100000
 1921
 1929
 1930                                      ; REGISTER USAGE.
 1931                                      ;
 1932                                      ;       R0 - PASSES PARAMETERS TO/FROM DIAGNOSTIC SUPERVISOR.
 1933                                      ;       R1 - COMMAND SEQUENCE TABLE POINTER.
 1934                                      ;       R2 - GENERAL PURPOSE REGISTER.
 1935                                      ;       R3 - GENERAL PURPOSE REGISTER.
 1936                                      ;       R4 - GENERAL PURPOSE REGISTER.
 1937                                      ;       R5 - CURRENT LOGICAL DEVICE NUMBER X 2.
 1938                                      ;       R6 - STACK POINTER.
 1939                                      ;       R7 - PROGRAM COUNTER.
 1940
 1941                                      ;THE FOLLOWING ARE BIT DEFINITIONS FOR THE TSSR REGISTERS.
 1942
 1943          100000                    TS.SC==100000                ;SPECIAL CONDITION BIT.
 1944          040000                    TS.UPE==40000                ;UNIBUS PARITY ERROR
 1945          020000                    TS.SPE==20000                ;SERIAL BUS PARITY ERROR.
 1946          010000                    TS.RMR==10000                ;REGISTER MODIFICATION REFUSED.
 1947          004000                    TS.NXM==4000                 ;NON-EXISTENT MEMORY.
 1948          002000                    TS.NBA==2000                 ;NEED BUFFER ADDRESS.
 1949          001000                    TS.A17==1000                 ;BUS ADDRESS BIT 17.
 1950          000400                    TS.A16==400                  ;BUS ADDRESS BIT 16.
 1951          000200                    TS.SSR==200                  ;UNIT READY BIT.
 1952          000100                    TS.OFL==100                  ;OFF LINE.
 1953          177717                    TSC.FCC==177717              ;FATAL CLASS CODE MASK.
 1954          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 TU80 BUT USED INTERNALLY BY THIS PROGRAM ONL
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 TU80.
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 EO
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 REGISTER 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==TU80 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.
000010                  SCHCNT==10                      ;ARBITRARY BYTE LENGTH FOR CHARACTERISTIC
                                                        ;BUFFER LENGTH. (EVEN #)
000016                  MSGCNT==16                      ;MESSAGE BUFFER LENGTH IN BYTES. (EVEN #)
000020                  DIACNT==20                      ;DIAGNOSTIC COMMAND BUFFER EXTENT.
004000                  DATCNT==2048.                   ;MAXIMUM RECORD LENGTH IN BYTES,
                                                        ;THIS COUNT SHOULD BE A MULTIPLE OF 256 TO INSUR
                                                        ;PROPER READ/WRITE BUFFER ALLOCATION BY THE SUPE
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!MOD.C1!MOD.C0!IE.C!FMT.C1!FMT.C0
                                                        ;NOT "COMMAND" BITS.

                        ;THE FOLLOWING DEFINES THE COMMAND WORD FOR EACH TU80 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

104410                  SRR==   ACK.C!BRF.C!MOD.C0!CMD.C3
                                                        ;SPACE RECORD REVERSE
```

```
2068
2069    105401                  RNR==   ACK.C!BRF.C!MOD.C1!MOD.C0!CMD.C0
2070                                            ;READ REV RETRY1 - REREAD NEXT REVERSE, IE. SPACE FWD, READ REVE
2071
2072    125401                  RNF==   ACK.C!BRF.C!OPP.C!MOD.C1!MOD.C0!CMD.C0
2073                                            ;READ REV RETRY2 - REREAD NEXT FORWARD, IE.READ FORWARD, SPACE R
2074
2075    105001                  RPF==   ACK.C!BRF.C!MOD.C1!CMD.C0
2076                                            ;READ FWD RETRY1 - REREAD PREVIOUS FORWARD, IE. SPACE REVERSE, R
2077
2078    125001                  RPR==   ACK.C!BRF.C!OPP.C!MOD.C1!CMD.C0
2079                                            ;READ FWD RETRY2 - REREAD PREVIOUS REVERSE, IE. READ REVERSE, SP
2080
2081    105005                  WRR==   ACK.C!MOD.C1!BRF.C!CMD.C2!CMD.C0
2082                                                    ;WRITE RETRY
2083
2084    102010                  RWD==   ACK.C!MOD.C2!CMD.C3
2085                                                    ;REWIND COMMAND
2086
2087    100012                  MBR==   ACK.C!CMD.C3!CMD.C1
2088                                                    ;MESSAGE BUFFER RELEASE
2089
2090    100011                  WTM==   ACK.C!CMD.C3!CMD.C0
2091                                                    ;WRITE TAPE MARK.
2092
2093    101011                  WTR==   ACK.C!MOD.C1!CMD.C3!CMD.C0
2094                                                    ;WRITE TAPE MARK RETRY.
2095
2096    105010                  SFF==   ACK.C!BRF.C!MOD.C1!CMD.C3
2097                                                    ;SPACE FILE FORWARD
2098
2099    105410                  SFR==   ACK.C!BRF.C!MOD.C0!MOD.C1!CMD.C3
2100                                                    ;SPACE FILE REVERSE
2101
2102    100017                  GES==   ACK.C!CMD.C0!CMD.C1!CMD.C2!CMD.C3
2103                                                    ;GET EXTENDED STATUS
2104
2105    100411                  ERS==   ACK.C!MOD.C0!CMD.C3!CMD.C0
2106                                                    ;ERASE 3 INCHES OF TAPE
2107
2108    100412                  UNL==   ACK.C!MOD.C0!CMD.C3!CMD.C1
2109                                                    ;UNLOAD COMMAND
2110
2111    101012                  CLN==   ACK.C!MOD.C1!CMD.C3!CMD.C1
2112                                                    ;ERASE TAPE.
2113
2114    140004                  SCH==   ACK.C!CVC.C!CMD.C2      ;SET DEVICE CHARACTERISTICS.
2115
2116    100006                  DIA==   ACK.C!CMD.C2!CMD.C1     ;DIAGNOSTICS.
2117
2118    000040                  JMP==   JMP.C                   ;JUMP TO 'N'TH COMMAND
2119
2120    000020                  DLY==   DLY.C                   ;DELAY 'N'' MS.
2121
2122    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.

                002314                  .       =       .+3&177774      ;MUST BE ON MOD 4 BOUNDRY.
002314  000000                  CMDPKT:: 0                              ;1ST WORD IS TU80 COMMAND.
002316  000000                          0                               ;2ND WORD IS THE BUFFER LOW ADDRESS.
002320  000000                          0                               ;3RD WORD IS THE BUFFER HIGH ADDRESS.
002322  000000                          0                               ;4TH WORD IS THE BYTE/RECORD/FILE COUNT.

                                        ;       GET STATUS COMMAND PACKET.

                002324                  .       =       .+3&177774      ;MUST BE ON MOD 4 BOUNDRY.
002324  100017                  GSCPK:: .WORD   GES

                                        ;       MESSAGE BUFFER RELEASE COMMAND PACKET.

                002330                  .       =       .+3&177774      ;MUST BE ON MOD 4 BOUNDRY.
002330  100012                  BRCPK:: .WORD   MBR

                                        ;       REWIND COMMAND PACKET (USED IN ERROR RECOVERY ONLY)

                002334                  .       =       .+3&177774      ;MUST BE ON A MODULE 4 BOUNDARY.
002334  102010                  RWCPK:: .WORD   RWD
002336  000001                          .WORD   1

                                        ;       WORK AREA FOR ANALYSIS OF MESSAGE PACKET CONTENTS.

002340  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.

002356  000007                  MSGPK0:: .BLKW  7                       ;MESSAGE PACKET FOR DEVICE #0
002374  000007                  MSGPK1:: .BLKW  7                       ;MESSAGE PACKET FOR DEVICE #1
002412  000007                  MSGPK2:: .BLKW  7                       ;MESSAGE PACKET FOR DEVICE #2
002430  000007                  MSGPK3:: .BLKW  7                       ;MESSAGE PACKET FOR DEVICE #3
```

```
2173                                    ;       SET CHARACTERISTIC BLOCK.
2174
2175  002446  002356                    SCHBK:: MSGPK0                  ;1ST WORD:: MSGPKT ADDR LO(SET UP BY EXCUTE ROUT
2176  002450  000000                            0                       ;2ND WORD:: MSGPKT ADDR HI.
2177  002452  000016                            MSGCNT                  ;3RD WORD:: MSG BUFFER LENGTH (BYTES)
2178  002454  000040                            CH.EAI                  ;4TH WORD:: CHARACTERISTICS WORD(SET BY SETUP RO
2179
2180                                    ;       TU80 REGISTER ADDRESSES.
2181
2182  002456  000004                    TSDB:: .BLKW    4               ;TU80 DATA BUFFER ADDRESSES.
2183  002466  000004                    TSSR:: .BLKW    4               ;TU80 STATUS REGISTER ADDRESSES.
2184  002476  000004                    TSVCT:: .BLKW   4               ;TU80 VECTOR ADDRESSES.
2185          002456                    TSBA==TSDB                      ;DATA BUFFER ADDRESS REGISTER.
2186
2187                                    ;       ADDRESSES OF MESSAGE PACKETS.
2188
2189  002506  002356                    MSGPKA:: MSGPK0                 ;DEVICE 0.
2190  002510  002374                            MSGPK1                  ;DEVICE 1.
2191  002512  002412                            MSGPK2                  ;DEVICE 2.
2192  002514  002430                            MSGPK3                  ;DEVICE 3.
2193
2194                                    ;       ADDRESSES OF INTERRUPT HANDLING ROUTINES.
2195
2196  002516  006566                    TS4INT:: TS4IN0                 ;DEVICE 0.
2197  002520  006574                            TS4IN1                  ;DEVICE 1.
2198  002522  006602                            TS4IN2                  ;DEVICE 2.
2199  002524  006610                            TS4IN3                  ;DEVICE 3.
2200
2201                                    ;       TU80 CODE LEVELS, WILL BE STORED AFTER SCH CMD IN BASIC FUNCTION TEST
2202
2203  002526  000000                    TS4CL:: 0                       ;DEVICE 0
2204  002530  000000                            0                       ;DEVICE 1
2205  002532  000000                            0                       ;DEVICE 2
2206  002534  000000                            0                       ;DEVICE 3
2207
2208                                    ;       UNIT NUMBERS OF ALL DEVICES BEING TESTED(1-4).
2209                                    ;       WHEN DEVICE IS NOT IN USE, IT,S LOCATION WILL = -3.
2210                                    ;       R5 WILL ALWAYS CONTAIN THE PRESENT LOGICAL UNIT NUMBER X 2.
2211
2212  002536  177774                    DEVTBL:: .WORD  NINUSE
2213  002540  177774                            .WORD   NINUSE
2214  002542  177774                            .WORD   NINUSE
2215  002544  177774                            .WORD   NINUSE
2216  002546  177777                            .WORD   END
2217
2218                                    ;       BAD TAPE TABLE POINTER: USED BY WRITE RETRY ROUTINE
2219                                    ;       'WRTY' TO LOG BAD TAPE SPOTS ON UNITS UNDER TEST
2220
2221  002550  003000                    BTADDR:: BT0
2222  002552  003052                             BT1
2223  002554  003124                             BT2
2224  002556  003176                             BT3
```

```
2226                                    ;       COUNTER AREA.
2227
2228            002560                  CNTBGN=.
2229  002560    000020                  WRBC:: .BLKW    20              ;BYTES WRITTEN.
2230  002620    000020                  RRBC:: .BLKW    20              ;BYTES READ REV.
2231  002660    000020                  RFBC:: .BLKW    20              ;BYTES READ FWD.
2232  002720    000004                  WRREC:: .BLKW   4               ;RECOVERABLE WRITE ERRORS.
2233  002730    000004                  WRUNR:: .BLKW   4               ;UNRECOVERABLE WRITE ERRORS.
2234  002740    000004                  RRREC:: .BLKW   4               ;RECOVERABLE READ REV ERRORS.
2235  002750    000004                  RRUNR:: .BLKW   4               ;UNRECOVERABLE READ REV ERRORS.
2236  002760    000004                  RFREC:: .BLKW   4               ;RECOVERABLE READ FWDERRORS.
2237  002770    000004                  RFUNR:: .BLKW   4               ;UNRECOVERABLE READ FWD ERRORS.
2238  003000    000025                  BT0::   .BLKW   21.             ;UNIT 0 BAT TAPE SPOTS LOG
2239  003052    000025                  BT1::   .BLKW   21.             ;UNIT 1 BAT TAPE SPOTS LOG
2240  003124    000025                  BT2::   .BLKW   21.             ;UNIT 2 BAT TAPE SPOTS LOG
2241  003176    000025                  BT3::   .BLKW   21.             ;UNIT 3 BAT TAPE SPOTS LOG
2242  003250    000004                  WRTYCT:: .BLKW  4               ;WRITE RETRY COUNTER
2243  003260    000004                  PASCNT:: .BLKW  4               ;PASS COUNT.
2244  003270    000004                  SCCNT:: .BLKW   4               ;SPECIAL CONDITION COUNT.
2245  003300    000004                  VFYCNT:: .BLKW  4               ;COUNT OF TU80 DATA COMPARE ERRORS.
2246  003310    000004                  HRDCNT:: .BLKW  4               ;COUNT OF HARD ERRORS.
2247  003320    000004                  FTLCNT:: .BLKW  4               ;COUNT OF FATAL ERRORS.
2248            003330                  CNTEND=.                        ;END OF STATICTICAL COUNTERS.
2249  003330    000004                  RECCNT:: .BLKW  4               ;NUMBER OF RECORDS FROM BOT: CLEARED  ON REWIND
                                                                        ;AND WHEN RESTARTING OR CONTINUING TEST 2.
2250
2251            000550                  CNTLEN==CNTEND-CNTBGN           ;LENGTH OF STATISTICAL COUNTER AREA.
2252
2253                                    ;       THE FOLLOWING ARE THE DEFINITIONS OF VARIABLES
2254                                    ;       USED BY THE PROGRAM.
2255
2256  003340      000                   DOAGIN:: .BYTE  0               ;TEST 3 LOOP CONTROL
2257  003341      000                   HERE::  .BYTE   0               ;THIS IS A 'BEEN HERE BEFORE'' MARKER
2258                                    .EVEN
2259  003342    000000                  RAMHLD:  .WORD  0               ;RAM ADDR HOLDER 1ST ADDRESS
2260  003344    000000                  RAMR5H:  .WORD  0               ;HOLDS R5 FOR LATER
2261  003346    000020                  RAMDATA::.BLKW  16.             ;DATA READ FROM RAM PACKET OR MESSAGE BUF AREA
2262  003406    000000                  RAMSIZ:: .WORD  0               ;RAM DATA SIZE FOR PRAMPKT ROUTINE
2263  003410    000000                  CMPDAT:: .WORD  0               ;COUNTS # OF READS (TEST 3) BEFORE ALLOWING A DA
2264  003412    000000                  DATRAT:: .WORD  0               ;CONTROLS THE DATA COMPARE RATIO
2265  003414    000000                  STTIM::  .WORD  0               ;BTL
2266  003416    000000                  DATAWT:: .WORD  0               ;WRITE BUFFER ADDRESS.
2267  003420    000000                  DATARD:: .WORD  0               ;READ BUFFER ADDRESS.
2268  003422    000000                  NCNT::   .WORD  0               ;STORAGE FOR VALUE OF N.
2269  003424    000000                  NCNT1::  .WORD  0               ;TEMP STORAGE FOR VALUE OF N.
2270  003426    000000                  BRFCNT:: .WORD  0               ;STORAGE FOR BPCR VALUE.
2271  003430    177777                  CMDWRD:: .WORD  END             ;CONTAINS COMMAND WORD BEING EXECUTED PRESENTLY.
2272  003432    177777                  CMDSAV:: .WORD  END             ;SAVE LOCATION FOR CMD WORD DURING ERROR RECOVER
2273  003434    177777                  PCMDWD:: .WORD  END             ;CONTAINS PREVOUS COMMAND WORD.
2274  003436    000000                  CMDLG::  .WORD  0               ;CURRENT COMMAND LOGGING CODE.
2275  003440    000000                  LENMSK:: .WORD  0               ;RANDOM WRITE LENGTH MASK, TO BE SET UP BY TESTS
2276  003442    153624                  RANB::   .WORD  153624          ;RANDOM # GENERATOR BASE.
2277  003444    032561                  RANS::   .WORD  32561           ;RANDOM # SAVE LOCATION.
2278  003446    000000                  TIME1::  .WORD  0               ;TIME COUNT 1.
2279  003450    000000                  TIME2::  .WORD  0               ;TIME COUNT 2.
2280  003452    000000                  JLOOP::  .WORD  0               ;JMP COMMAND LOOP COUNT.
2281  003454    000000                  JLOC::   .WORD  0               ;JMP COMMAND LOCATION COUNT.
```

```
  2282  003456  000000                    PATERN:: .WORD  0           ;PATTERN SELECT CODE.
```

```
2284  003460  000000                  CTCL::   .WORD  0              ;CURRENT TERMINATION CLASS CODE.
2285  003462  000000                  R5SAVE:: .WORD  0              ;LOCATION FOR SAVING CURRENT DEVICE POINTER.
2286  003464  000000                  TSSREG:: .WORD  0              ;CURRENT STATUS REGISTER.
2287          003416                  DIABLK==DATAWT                 ;WRITE BUFFER ALSO USED FOR DIAG CMD.
2288
2289                                  ;       ERROR FLAG AREA, THESE FLAGS ARE CLEARED DURING INITIALIZATION AND
2290                                  ;       AFTER EACH COMMAND IS COMPLETED.
2291
2292          003466                  BGNFLG=.
2293  003466  000000                  RETRYC:: .WORD  0              ;# OF RECOVERY ATTEMPTS EXECUTED.
2294  003470     000                  RPTCNT:: .BYTE  0       ;WRITE REPEAT ON SAME SPOT CNTR: 4 PER WRITE RETRY
2295  003471     000                  WRTYFG:: .BYTE  0              ;WRITE RETRY ON SAME SPOT IN PROGRESS FLAG
2296  003472     000                  WRTYER:: .BYTE  0              ;WRITE RETRY ON SAME SPOT ERROR FLAG
2297  003473     000                  RECLOG:: .BYTE  0              ;RECORD COUNT HAS BEEN UPDATED FOR THIS RECORD.
2298  003474     000                  ERLOG::  .BYTE  0              ;DATA BYTES AND ERRORS HAVE BEEN LOGGED FOR THIS
2299  003475     000                  RWERR::  .BYTE  0              ;READ/WRITE ERROR HAS OCCURED.
2300  003476     000                  UNREC::  .BYTE  0              ;UNRECOVERABLE ERROR HAS OCCURED.
2301  003477     000                  ERRREC:: .BYTE  0              ;ERROR RECOVERY MODE.
2302                                          .EVEN
2303          003500                  ENDERF=.
2304
2305                                  ;       ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED DURING INITIALIZATION.
2306
2307  003500  000004                  INTFLG:: .BLKW  4              ;INTERRUPT OCCURRED FLAGS FOR EACH DEVICE.
2308  003510  000004                  EOTFLG:: .BLKW  4              ;EOT/BOT FLAGS FOR EACH DEVICE (XSTAT0).
2309  003520  000000                  BTPT::   .WORD  0              ;BAD TAPE SPOT POINTER TO BT0-BT3 VIA BTADDR
2310  003522     000                  EXPBOT:: .BYTE  0              ;BOT IS EXPECTED, DO NOT ABORT ON BOT/FUNC RTI.
2311  003523     000                  RANDOM:: .BYTE  0              ;RANDOM EVERYTHING FLAG.
2312  003524     000                  VFYFLG:: .BYTE  0              ;SET DURING WRITE/VERIFY COMMAND.
2313  003525     000                  RPTFLG:: .BYTE  0              ;PERFORMANCE REPORT HAS BEEN REQUESTED.
2314  003526     000                  SWBFLG:: .BYTE  0              ;ENABLES SWAP BYTE FUNCTION WHEN NOT EQUAL TO ZE
2315  003527     000                  IRE::    .BYTE  0              ;INHIBIT RESIDUAL FRAME COUNT ERROR REPORT.
2316  003530     000                  DROPED:: .BYTE  0              ;CURRENT UNIT HAS BEEN DROPPED
2317  003531     000                  T1SWB::  .BYTE  0              ;TEST1 SWAP BYTES FLAG
2318  003532     000                  ALLEOT:: .BYTE  0              ;ALL UNITS @ EOT FLAG
2319  003533     000                  STREAM:: .BYTE  0              ;INDICATES TEST ONE UNIT AT A TIME, COMPLETELY.
2320  003534     000                  ERSFLG:: .BYTE  0              ;ERASE FLAG:  DO ERASE AFTER A SPACE REV TO DELE
2321                                                                 ;BADLY WRITTEN RECORD.  1 TO 4 ERASES LEAVING
2322                                                                 ;A 3 TO 12 INCH GAP MAY RESULT.
2323          003536                          .EVEN
2324          003536                  ENDFLG=.
2325
2326                                  ;       ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED ONLY AFTER BEING CHECKED.
2327
2328  003536     000                  STAFLG:: .BYTE  0              ;START FLAG - SET BY INIT CODE IF STARTING.
2329  003537     000                  PWRFLG:: .BYTE  0              ;POWER FAILURE FLAG - SET ONLY DURING INIT.
2330  003540     000                  TRAPD4:: .BYTE  0              ;TRAPED AT 4 FLAG
2331  003541     000                  MISCFG:: .BYTE  0              ;MISCELLANEOUS FLAG
2332
```

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

003542  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.

003544  140004                          CMDSEQ:: .WORD  SCH             ;SET CHARACTERISTICS.
003546  000040                                   .WORD  CH.EAI
003550  000001                                   .WORD  1
003552  000000                                   .WORD  0
003554  102010                          CMDSE2:: .WORD  RWD             ;REWIND.
003556  000001                                   .WORD  1               ;BYTE COUNT.
003560  000001                                   .WORD  1               ;ONCE.
003562  000007                                   .WORD  RANP            ;PATTERN.
003564  104005                                   .WORD  WRT             ;WRITE.
003566  004000                                   .WORD  DATCNT          ;MAX BUFFER LENGTH.
003570  076400                                   .WORD  32000.          ;32,000 RECORDS.
003572  000007                                   .WORD  RANP            ;RANDOM PATTERN.
003574  104401                                   .WORD  RDR             ;READ REV.
003576  004000                                   .WORD  DATCNT          ;MAX BUFFER LENGTH.
003600  076400                                   .WORD  32000.          ;32,000 RECORDS
003602  000007                                   .WORD  RANP            ;RANDOM PATTERN.
003604  104001                                   .WORD  RDF             ;READ FWD.
003606  004000                                   .WORD  DATCNT          ;MAX BUFFER LENGTH.
003610  076400                                   .WORD  32000.          ;32,000 RECORDS.
003612  000007                                   .WORD  RANP            ;RANDOM PATTERN.
003614  102010                                   .WORD  RWD             ;REWIND.
003616  000001                                   .WORD  1               ;BYTE COUNT.
003620  000001                                   .WORD  1               ;ONCE.
003622  000007                                   .WORD  RANP            ;PATTERN.
003624  000004                                   .BLKW  4               ;EXTENSTION TO HOLD 1 MORE CMD.
003634  177777                          SEQEND:: .WORD  END             ;SOFT END OF SEQUENCE TABLE.
003636  177777                                   .WORD  END
003640  177777                                   .WORD  END
003642  177777                                   .WORD  END
003644  177777                                   .WORD  END             ;HARD END OF SEQUENCE TABLE.
```

```
2375                                        ;THE FOLLOWING IS THE TU80 COMMAND TABLE
2376
2377  003646  100013          CMDTBL:: .WORD  DRI          ;DRIVE INIT.
2378  003650  104001                   .WORD  RDF          ;READ FORWARD.
2379  003652  104401                   .WORD  RDR          ;READ REVERSE.
2380  003654  104005                   .WORD  WRT          ;WRITE
2381  003656  104105                   .WORD  WTV          ;WRITE/VERIFY.  (WRITE ALL RECORDS, RDR AND
2382                                                       ;CHECK DATA ON ALL RECORDS, RDF AND
2383                                                       ;CHECK DATA ON ALL RECORDS.)
2384  003660  104010                   .WORD  SRF          ;SPACE 'N'' RECORDS FORWARD.
2385  003662  104410                   .WORD  SRR          ;SPACE 'N'' RECORDS REVERSE.
2386  003664  105401                   .WORD  RNR          ;READ NEXT REVERSE. I.E., SPACE FWD, READ REVERS
2387  003666  125401                   .WORD  RNF          ;READ NEXT FORWARD, I.E., READ FORWARD, SPACE RE
2388  003670  105001                   .WORD  RPF          ;READ PREVIOUS FORWARD. I.E., SPACE REVERSE, REA
2389  003672  125001                   .WORD  RPR          ;READ PREVIOUS REVERSE. I.E., READ REVERSE, SPAC
2390  003674  105005                   .WORD  WRR          ;WRITE RETRY.
2391  003676  102010                   .WORD  RWD          ;REWIND.
2392  003700  100012                   .WORD  MBR          ;MESSAGE BUFFER RELEASE
2393  003702  100011                   .WORD  WTM          ;WRITE TAPE MARK
2394  003704  101011                   .WORD  WTR          ;WRITE TAPE MARK RETRY.
2395  003706  105010                   .WORD  SFF          ;SPACE 'N'' FILES FORWARD.
2396  003710  105410                   .WORD  SFR          ;SPACE 'N'' FILES REVERSE.
2397  003712  100017                   .WORD  GES          ;GET EXTENDED STATUS.
2398  003714  100411                   .WORD  ERS          ;ERASE 3 INCHES OF TAPE.
2399  003716  100412                   .WORD  UNL          ;REWIND AND UNLOAD.
2400  003720  101012                   .WORD  CLN          ;CLEAR TAPE.
2401  003722  140004                   .WORD  SCH          ;SET CHARACTERISTICS.
2402  003724  100006                   .WORD  DIA          ;DIAGNOSTIC COMMAND.
2403  003726  000040                   .WORD  JMP          ;JUMP TO THE NTH COMMAND IN THE SEQUENCE.
2404  003730  000020                   .WORD  DLY          ;DELAY 'N'' MS.
2405  003732  177777                   .WORD  END          ;END OF COMMAND TABLE
2406
```

```
2408                                    ;       THE FOLLOWING TABLE CONTAINS THE ASCII FOR EACH COMMAND.
2409
2410  003734  051104     111      CMDASC:: .ASCII /DRI/          ;DRIVE INIT.
2411  003737     122  043104              .ASCII  /RDF/          ;READ FORWARD.
2412  003742  042122     122              .ASCII  /RDR/          ;READ REVERSE.
2413  003745     127  052122              .ASCII  /WRT/          ;WRITE
2414  003750  052127     126              .ASCII  /WTV/          ;WRITE/VERIFY.  (WRITE ALL RECORDS, RDR AND CHEC
2415                                                             ;ON ALL RECORDS, RDF AND CHECK DATA ON ALL RECOR
2416  003753     123  043122              .ASCII  /SRF/          ;SPACE 'N' RECORDS FORWARD.
2417  003756  051123     122              .ASCII  /SRR/          ;SPACE 'N'  RECORDS REVERSE.
2418  003761     122  051116              .ASCII  /RNR/          ;READ NEXT REVERSE. I.E., SPACE FWD READ REVERSE
2419  003764  047122     106              .ASCII  /RNF/          ;READ NEXT FORWARD, I.E., READ FORWARD, SPACE RE
2420  003767     122  043120              .ASCII  /RPF/          ;READ PREVIOUS FORWARD. IE., SPACE REVERSE, READ
2421  003772  050122     122              .ASCII  /RPR/          ;READ PREVIOUS REVERSE. IE., READ REVERSE, SPACE
2422  003775     127  051122              .ASCII  /WRR/          ;WRITE RETRY.
2423  004000  053522     104              .ASCII  /RWD/          ;REWIND.
2424  004003     115  051102              .ASCII  /MBR/          ;MESSAGE BUFFER RELEASE
2425  004006  052127     115              .ASCII  /WTM/          ;WRITE TAPE MARK
2426  004011     127  051124              .ASCII  /WTR/          ;WRITE TAPE MARK RETRY.
2427  004014  043123     106              .ASCII  /SFF/          ;SPACE 'N' FILES FORWARD.
2428  004017     123  051106              .ASCII  /SFR/          ;SPACE 'N' FILES REVERSE.
2429  004022  042507     123              .ASCII  /GES/          ;GET EXTENDED STATUS.
2430  004025     105  051522              .ASCII  /ERS/          ;ERASE 3 INCHES OF TAPE.
2431  004030  047125     114              .ASCII  /UNL/          ;REWIND AND UNLOAD.
2432  004033     103  047114              .ASCII  /CLN/          ;CLEAN TAPE.
2433  004036  041523     110              .ASCII  /SCH/          ;SET CHARACTERISTICS.  WHERE BPI=200, 40, 20, 0.
2434                                                             ;SEE TU80 PROGRAMMING SPECIFICATION FOR DESCRIPT
2435  004041     104  040511              .ASCII  /DIA/          ;DIAGNOSTICS.  SEE TU80 PROGRAMMING SPECIFICATIO
2436                                                             ;FOR DESCRIPTION.  ODT MUST BE USED TO LOAD DIAG
2437                                                             ;INTO THE WRITE BUFFER BEFORE THIS CMD IS ISSUED
2438  004044  046512     120              .ASCII  /JMP/          ;JUMP TO THE NTH COMMAND IN THE COMMAND
2439                                                             ;SEQUENCE TABLE, WHERE N IS DEFINED IN
2440                                                             ;THE # OF OPERATIONS.
2441  004047     104  054514              .ASCII  /DLY/          ;DELAY 'N' MS, WHERE N IS DEFINED IN
2442                                                             ;THE # OF OPERATIONS.
2443  004052  047105     104              .ASCII  /END/          ;END OF COMMAND SEQUENCE.
2444          004056                      .EVEN
2445
2446
2447
```

```
                                        .SBTTL  GLOBAL TEXT SECTION

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



                                        ;
                                        ; FORMAT STATEMENTS USED IN PRINT CALLS
                                        ;

                                                .NLIST  BEX

004056  047045  040445  047125          CODELM:: .ASCIZ /%N%AUNIT %D1%A TU80 CODE LEVEL P%03%N%N/
                                                .EVEN
004126  054130  020130  046503          HALTM:: .ASCIZ  /XXX CMD - TYPE <CR> TO CONTINUE/
004166  046503  020104  040520          CMDPKM:: .ASCIZ /CMD PACKET ADR NOT ON MODULO 4 BOUNDARY: RELOAD!/
        004250                          .EVEN
004250  040504  040524  041440          WTVERM:: .ASCIZ /DATA COMPARE ERROR/
004273     116  020117  052524          TOERM:: .ASCIZ  /NO TU80 RESPONSE/
004314  047125  042504  044506          SCERM:: .ASCIZ  /UNDEFINED SPEC COND/
004340  043122  020103  047516          RFCERM:: .ASCIZ /RFC NON ZERO/
004355     124  034125  020060          NSSRM:: .ASCIZ  /TU80 NOT READY/
004374  042522  051124  020131          RLEXM:: .ASCIZ  /RETRY LIMIT EXCEEDED/
004421     125  044516  020124          ATTNM:: .ASCIZ  /UNIT OFF LINE/
004437     106  047125  0521C3          FUNRM:: .ASCIZ  /FUNCTION REJECT/
004457     106  052101  046101          FATSM:: .ASCIZ  /FATAL SUBSYSTEM ERROR/
004505     116  020117  047111          NOINTM:: .ASCIZ /NO INTERRUPT/
004522  040524  042520  051440          TSAM::  .ASCIZ  /TAPE STATUS ALERT/
004544  047524  020117  040515          TOOMM:: .ASCIZ  /TOO MANY INTERRUPTS/
004570  040503  051520  040524          RNYM::  .ASCIZ  /CAPSTAN RUNAWAY-GET STATUS RESULTS:/
004634  042522  047503  042526          RERM::  .ASCIZ  /RECOVERABLE ERROR/
004656  047125  042522  047503          URERM:: .ASCIZ  /UNRECOVERABLE ERROR/
004702  047045  040445  051104          DROPDM:: .ASCIZ /%N%ADROPPED UNIT %D1%N/
004731     045  022516  040501          AUDRPM:: .ASCIZ /%N%AALL UNITS DROPPED%N%N/
004763     045  022516  041101          DTAER2:: .ASCIZ '%N%ABYTE:%D4%S2%AWAS:%B8%S2%AS/B:%B8%N'
005032  042045  022464  020101          DTAER3:: .ASCIZ '%D4%A BYTES IN ERROR OUT OF %D4%N'
005074  040445  047516  042040          DTAER4:: .ASCIZ /%ANO DATA READ%N/
005115     045  051101  041505          DTAER5:: .ASCIZ /%ARECORD TOO LONG: >%04%A BYTES%N/
005157     045  051101  041505          NURTY1:: .ASCIZ /%ARECOVERED ON RETRY #%D2%N/
005213     045  052501  044516          OFLINM:: .ASCIZ /%AUNIT %D1%A OFF LINE%N/
005243     045  043501  052105          GETSTM:: .ASCIZ /%AGET STATUS CMD RESULTS:%N/
005277     045  000116                  CRLF::  .ASCIZ  /%N/
005302  047045  051445  000067          CRLFSP:: .ASCIZ /%N%S7/
005310  047045  040445  025040          RAMFHR: .ASCIZ  '%N%A ********** SPECIAL M7454 RAM MEMORY DUMP **********'
005401     045  022516  020101          RAMIOP: .ASCIZ  '%N%A RAM ADDRESS (OCTAL) = %03%A - %03%N'
005452  040445  022440  031517          RAMPD:  .ASCIZ  '%A %03%A '
005464  047045  047045  047045          RAMLIN: .ASCIZ  '%N%N%N'
                                        .LIST   BEX
        005474                          .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..
                                ;--


                                        BGNMSG  DTAERM
                                DTAERM::
                                        PRINTB  #STAER1,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
2518  005474                                                                            MOV     RECCNT(R
 (3)  005474                                                                            MOV     PASCNT(R
2524  005474                                                                            MOV     DEVTBL(R
 (10) 005474  016546  003330                                                           MOV     #STAER1,
 (9)  005500  016546  003260                                                           MOV     #4,-(SP)
 (8)  005504  016546  002536                                                           MOV     SP,R0
 (7)  005510  012746  006154                                                           TRAP    C$PNTB
 (6)  005514  012746  000004                                                           ADD     #12,SP
 (3)  005520  010600
 (4)  005522  104414
 (4)  005524  062706  000012
2525  005530                                    PRINTB  #STAER7
 (7)  005530  012746  006246                                                           MOV     #STAER7,
 (6)  005534  012746  000001                                                           MOV     #1,-(SP)
 (3)  005540  010600                                                                   MOV     SP,R0
 (4)  005542  104414                                                                   TRAP    C$PNTB
 (4)  005544  062706  000004                                                           ADD     #4,SP
2526  005550                                    LET RECRED := R2              ;SAVE R2
 (4)  005550  010237  006562                                                           MOV     R2,RECRE
2527  005554                                    LET TIME1 := R3               ;SAVE R3
 (4)  005554  010337  003446                                                           MOV     R3,TIME1
2528  005560                                    LET TIME2 := R4               ;SAVE R4
 (4)  005560  010437  003450                                                           MOV     R4,TIME2
2529  005564  004737  006616                   JSR PC,RECTAP                 ;RETRIEVE RECORD READ
2530  005570                                    LET R2 := RECRED              ;RESTORE R2
 (4)  005570  013702  006562                                                           MOV     RECRED,R
2531  005574                                    LET RECRED := R3              ;SAVE RECORD READ
 (4)  005574  010337  006562                                                           MOV     R3,RECRE
2532  005600                                    LET R3 := TIME1               ;RESTORE R3
 (4)  005600  013703  003446                                                           MOV     TIME1,R3
2533  005604                                    LET R4 := TIME2               ;RESTORE R4
 (4)  005604  013704  003450                                                           MOV     TIME2,R4
2534  005610                                    PRINTB  #STAER6,RECRED        ;PRINT RECORD READ
 (8)  005610  013746  006562                                                           MOV     RECRED,-
 (7)  005614  012746  006276                                                           MOV     #STAER6,
 (6)  005620  012746  000002                                                           MOV     #2,-(SP)
 (3)  005624  010600                                                                   MOV     SP,R0
 (4)  005626  104414                                                                   TRAP    C$PNTB
 (4)  005630  062706  000006                                                           ADD     #6,SP
2535  005634                                    EXIT    MSG
 (4)  005634  000167                                                                   .WORD   J$JMP
 (3)  005636  000000                                                                   .WORD   L10002-2
2536                                    .EVEN
2537
2538  005640                            ENDMSG
 (3)  005640                    L10002:
 (3)  005640  104423                                                                   TRAP    C$MSG
```

```
2539
2540  005642                                  BGNMSG  STAERM
 (3)  005642                          STAERM::
2541  005642                                  PRINTB  #STAER1,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
 (10) 005642  016546  003330                                                    MOV     RECCNT(R
 (9)  005646  016546  003260                                                    MOV     PASCNT(R
 (8)  005652  016546  002536                                                    MOV     DEVTBL(R
 (7)  005656  012746  006154                                                    MOV     #STAER1,
 (6)  005662  012746  000004                                                    MOV     #4,-(SP)
 (3)  005666  010600                                                            MOV     SP,R0
 (4)  005670  104414                                                            TRAP    C$PNTB
 (4)  005672  062706  000012                                                    ADD     #12,SP
2542  005676                                  PRINTB  #STAER7
 (7)  005676  012746  006246                                                    MOV     #STAER7,
 (6)  005702  012746  000001                                                    MOV     #1,-(SP)
 (3)  005706  010600                                                            MOV     SP,R0
 (4)  005710  104414                                                            TRAP    C$PNTB
 (4)  005712  062706  000004                                                    ADD     #4,SP
2543  005716                                  LET R2 := CMDPKT CLR.BY #177740
 (4)  005716  013702  002314                                                    MOV     CMDPKT,R
 (6)  005722  042702  177740                                                    BIC     #177740,
2544  005726                                  LET R2 := R2 - #1
 (6)  005726  005302                                                            DEC     R2
2545  005730                                  IF R2 EQ #0 THEN                ;IF CMD IS A READ
 (6)  005730  005702                                                            TST     R2
 (9)  005732  001016                                                            BNE     50000$
2546  005734  004737  006616                    JSR PC,RECTAP                 ;THEN RETRIEVE
2547  005740                                    LET RECRED := R3              ;AND
 (4)  005740  010337  006562                                                    MOV     R3,RECRE
2548  005744                                    PRINTB #STAER6,RECRED         ;TYPE RECORD READ
 (8)  005744  013746  006562                                                    MOV     RECRED,-
 (7)  005750  012746  006276                                                    MOV     #STAER6,
 (6)  005754  012746  000002                                                    MOV     #2,-(SP)
 (3)  005760  010600                                                            MOV     SP,R0
 (4)  005762  104414                                                            TRAP    C$PNTB
 (4)  005764  062706  000006                                                    ADD     #6,SP
2549  005770                                  ENDIF
 (4)  005770                                                          50000$:
2550  005770                                  PRINTX  #STAER2
 (7)  005770  012746  006332                                                    MOV     #STAER2,
 (6)  005774  012746  000001                                                    MOV     #1,-(SP)
 (3)  006000  010600                                                            MOV     SP,R0
 (4)  006002  104415                                                            TRAP    C$PNTX
 (4)  006004  062706  000004                                                    ADD     #4,SP
2551  006010                                  PRINTX  #STAER3,CMDPKT,@TSDB(R5),MSGPKT+MS.RFC,TSSREG,CTCC
 (12) 006010  013746  003460                                                    MOV     CTCC,-(S
 (11) 006014  013746  003464                                                    MOV     TSSREG,-
 (10) 006020  013746  002344                                                    MOV     MSGPKT+M
 (9)  006024  017546  002456                                                    MOV     @TSDB(R5
 (8)  006030  013746  002314                                                    MOV     CMDPKT,-
 (7)  006034  012746  006411                                                    MOV     #STAER3,
 (6)  006040  012746  000006                                                    MOV     #6,-(SP)
 (3)  006044  010600                                                            MOV     SP,R0
 (4)  006046  104415                                                            TRAP    C$PNTX
 (4)  006050  062706  000016                                                    ADD     #16,SP
2552  006054                                  PRINTX  #STAER4,CMDPKT+2,CMDPKT+4,CMDPKT+6
```

```
 (10) 006054  013746  002322                                                    MOV     CMDPKT+6
 (9)  006060  013746  002320                                                    MOV     CMDPKT+4
 (8)  006064  013746  002316                                                    MOV     CMDPKT+2
 (7)  006070  012746  006447                                                    MOV     #STAER4,
 (6)  006074  012746  000004                                                    MOV     #4,-(SP)
 (3)  006100  010600                                                            MOV     SP,R0
 (4)  006102  104415                                                            TRAP    C$PNTX
 (4)  006104  062706  000012                                                    ADD     #12,SP
2553  006110                        PRINTX  #STAER5,MSGPKT+MS.XS0,MSGPKT+MS.XS1,MSGPKT+MS.XS2,MSGPKT+MS.XS3
 (11) 006110  013746  002354                                                    MOV     MSGPKT+M
 (10) 006114  013746  002352                                                    MOV     MSGPKT+M
 (9)  006120  013746  002350                                                    MOV     MSGPKT+M
 (8)  006124  013746  002346                                                    MOV     MSGPKT+M
 (7)  006130  012746  006467                                                    MOV     #STAER5,
 (6)  006134  012746  000005                                                    MOV     #5,-(SP)
 (3)  006140  010600                                                            MOV     SP,R0
 (4)  006142  104415                                                            TRAP    C$PNTX
 (4)  006144  062706  000014                                                    ADD     #14,SP
2554  006150                        EXIT    MSG
 (4)  006150  000167                                                            .WORD   J$JMP
 (3)  006152  000410                                                            .WORD   L10003-2
2555
2556                                    .NLIST  BEX
2557  006154  040445  054130  020130  STAER1: .ASCIZ  /%AXXX CMD FAILED - UNIT %D1%S3%APASS:%D5%S3%ARECORD:%D5%N/
2558                                    .EVEN
2559  006246  040445  051120  053105  STAER7: .ASCIZ  /%APREVIOUS CMD WAS  XXX/
2560  006276  051445  030461  040445  STAER6: .ASCIZ  /%S11%A* RECORD READ:%D5%A */
2561  006332  047045  040445  046503  STAER2: .ASCIZ  /%N%ACMDPKT%S2%ATSBA%S4%ARFC%S5%ATSSR%S3%ATCC%N/
2562  006411     045  033117  051445  STAER3: .ASCIZ  /%06%S2%06%S2%06%S2%06%S2%D1%N/
2563  006447     045  033117  047045  STAER4: .ASCII  /%06%N/
2564  006454  047445  022466     116          .ASCII  /%06%N/
2565  006461     045  033117  047045          .ASCIZ  /%06%N/
2566  006467     045  054101  052123  STAER5: .ASCII  /%AXST0%S4%AXST1%S4%AXST2%S4%AXST3%N/
2567  006532  047445  022466  031123          .ASCIZ  /%06%S2%06%S2%06%S2%06%N/
2568                                    .LIST   BEX
2569                                    .EVEN
2570  006562  000000                  RECRED: .WORD   0                       ;RECORD READ FROM TAPE
2571
2572  006564                            ENDMSG
 (3)  006564                        L10003:
 (3)  006564  104423                                                            TRAP    C$MSG
```

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

                                        ;       MODULES TO HANDLE TU80 INTERRUPTS.

2582  006566                                    BGNSRV  TS4IN0                  ;DEVICE 0.
                                        TS4IN0::
2583  006566                                    LET INTFLG := INTFLG + #1        ;SET INTERRUPT OCCURRED FLAG.
 (6)  006566  005237  003500                                                     INC     INTFLG
2584  006572                                    ENDSRV
                                        L10004:
 (2)  006572  000002                                                            RTI

2586  006574                                    BGNSRV  TS4IN1                  ;DEVICE 1.
                                        TS4IN1::
2587  006574                                    LET INTFLG+2 := INTFLG+2 + #1    ;SET INTERRUPT OCCURRED FLAG.
 (6)  006574  005237  003502                                                     INC     INTFLG+2
2588  006600                                    ENDSRV
                                        L10005:
 (2)  006600  000002                                                            RTI

2590  006602                                    BGNSRV  TS4IN2                  ;DEVICE 2.
                                        TS4IN2::
2591  006602                                    LET INTFLG+4 := INTFLG+4 + #1    ;SET INTERRUPT OCCURRED FLAG.
 (6)  006602  005237  003504                                                     INC     INTFLG+4
2592  006606                                    ENDSRV
                                        L10006:
 (2)  006606  000002                                                            RTI

2594  006610                                    BGNSRV  TS4IN3                  ;DEVICE 3.
                                        TS4IN3::
2595  006610                                    LET INTFLG+6 := INTFLG+6 + #1    ;SET INTERRUPT OCCURRED FLAG.
 (6)  006610  005237  003506                                                     INC     INTFLG+6
2596  006614                                    ENDSRV
                                        L10007:
 (2)  006614  000002                                                            RTI
```

```
2598
2599                                    ;       SUBROUTINE TO RETRIEVE RECORD COUNT READ FROM TAPE FOR ERROR
2600                                    ;       PRINTS.
2601                                    ;       INPUTS:
2602                                    ;       OUTPUTS:  R3 = RECORD COUNT READ
2603                                    ;       REGISTERS:  R2, R3, R4
2604                                    ;       CALLS:
2605
2606  006616                            RECTAP::IF #MOD.CO SETIN CMDWRD THEN    ;READ REV FETCH
 (6)  006616  032737  000400  003430                                                      BIT     #MOD.CO,
 (9)  006624  001430                                                                     BEQ     50001$
2607  006626                                      LET R2 := MSGPKT+MS.RFC + DATARD ;FIND LAST READ AD.
 (4)  006626  013702  002344                                                            MOV     MSGPKT+M
 (6)  006632  063702  003420                                                            ADD     DATARD,R
2608  006636                                      IF #BIT00 SETIN R2 THEN     ;ODD AD., REASSEMBLE
 (6)  006636  032702  000001                                                            BIT     #BIT00,R
 (9)  006642  001417                                                                    BEQ     50002$
2609  006644                                        LET R2 := R2 + #1         ;REC COUNT STARTING
 (6)  006644  005202                                                                    INC     R2
2610  006646                                        LET R3 :B= (R2) CLR.BY #177400  ;WITH UPPER BYTE FETCH
 (4)  006646  111203                                                                    MOVB    (R2),R3
 (6)  006650  142703  177400                                                            BICB    #177400,
2611  006654                                        LET R3 := SWAP R3         ;
 (6)  006654  000303                                                                    SWAB    R3
2612  006656                                        LET R2 := R2 - #1         ;LOWER BYTE AD.
 (6)  006656  005302                                                                    DEC     R2
2613  006660                                        IFB SWBFLG NE #0 THEN
 (6)  006660  105737  003526                                                            TSTB    SWBFLG
 (9)  006664  001401                                                                    BEQ     50003$
2614  006666                                          LET R2 := R2 - #1       ;LOWER BYTE AD. ON SWAP
 (6)  006666  005302                                                                    DEC     R2
2615  006670                                        ENDIF
 (4)  006670                                                                  50003$:
2616  006670                                        LET R4 :B= (R2) CLR.BY #177400 ;FETCH LOWER BYTE
 (4)  006670  111204                                                                    MOVB    (R2),R4
 (6)  006672  142704  177400                                                            BICB    #177400,
2617  006676                                        LET R3 := R3 OR R4        ;MERGE BYTES
 (6)  006676  050403                                                                    BIS     R4,R3
2618  006700                                      ELSE
 (4)  006700  000401                                                                    BR      50004$
 (3)  006702                                                                  50002$:
2619  006702                                        LET R3 := (R2)            ;EVEN AD. FETCH
 (4)  006702  011203                                                                    MOV     (R2),R3
2620  006704                                      ENDIF
 (4)  006704                                                                  50004$:
2621  006704                                    ELSE
 (4)  006704  000402                                                                    BR      50005$
 (3)  006706                                                                  50001$:
2622  006706                                      LET R3 := @DATARD           ;READ FWD FETCH
 (4)  006706  017703  174506                                                            MOV     @DATARD,
2623  006712                                    ENDIF
 (4)  006712                                                                  50005$:
2624
2625  006712  000207                    RTS     PC
```

```
                        ;       SUBROUTINE TO STORE A  SET CHARACTERISTIC COMMAND AS
                        ;       THE FIRST ENTRY IN THE SEQUENCE TABLE.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:
                        ;       CALLS:

006714                  SETCH:: LET R1 := #CMDSEQ               ;INIT COMMAND SEQUENCE TABLE POINTER.
 (4) 006714  012701  003544                                                     MOV     #CMDSEQ,
006720  012721  140004          MOV     #SCH,(R1)+              ;THIS CODE SETS UP A SET CHARACTERISTIC
006724  012721  000040          MOV     #DFTSCH,(R1)+           ;COMMAND AS THE FIRST COMMAND IN THE
006730  012721  000001          MOV     #1,(R1)+                ;SEQUENCE TABLE.
006734  005721                  TST     (R1)+                   ;SKIP PATTERN LOCATION.
006736  000207                  RTS PC


                        ;       SUBROUTINE TO STORE A REWIND COMMAND IN THE SEQUENCE TABLE
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:
                        ;       CALLS:

006740                  SETRW:: LET (R1)+ := #RWD               ;CMD = REWIND.
 (4) 006740  012721  102010                                                     MOV     #RWD,(R1
006744                          LET (R1)+ := #1                 ;BRF.
 (4) 006744  012721  000001                                                    MOV     #1,(R1)+
006750                          LET (R1)+ := #1                 ;# OF OPERATIONS.
 (4) 006750  012721  000001                                                    MOV     #1,(R1)+
006754  005721                  TST (R1)+                       ;SKIP PATTERN.
006756  000207                  RTS     PC                      ;RETURN
```

```
                        ;       SUBROUTINE TO EXECUTE ALL COMMANDS IN THE SEQUENCE TABLE ON ALL
                        ;       DEVICES.
                        ;       INPUTS:
                        ;       OUTPUTS:        R2 = TERMINATION INDICATOR (0=END OF TABLE,1=EOT)
                        ;       REGISTERS:
                        ;       CALLS:          CMDAC,SETUP,EXSUB,CKHAE,NEXTU,FIRSTU,VFYDAT.
                        ;
2664  006760            EXALL:: LET R1 := #CMDSEQ               ;INIT SEQUENCE TABLE POINTER.
 (4)  006760  012701  003544                                                    MOV     #CMDSEQ,
                                WHILE (R1) NE #END DO           ;WHILE THERE ARE CMDS IN THE SEQUENCE TA
2665  006764                                                                   50006$:
 (4)  006764                                                                   CMP     (R1),#EN
 (6)  006764  021127  177777                                                   BEQ     50007$
 (9)  006770  001527
2666  006772  004737  007722            JSR   PC,SETUP              ;GO SETUP THE COMMAND BLOCK.
2667  006776                            WHILE NCNT LT NCNT1 DO      ;WHILE THERE ARE RECORDS REMAINING:
 (4)  006776                                                                   50010$:
 (6)  006776  023737  003422  003424                                           CMP     NCNT,NCN
 (9)  007004  002116                                                           BGE     50011$
2668  007006  004737  007614              JSR PC,CMDAC              ;STORE CMD ASCII IN ERROR MESSAGE.
2669  007012                              IFB RANDOM NE #0 THEN     ;IF IN RANDOM MODE:
 (6)  007012  105737  003523                                                   TSTB    RANDOM
 (9)  007016  001435                                                           BEQ     50012$
2670  007020                                IF CMDWRD EQ #WRT THEN  ;IF CMD IS A WRITE THEN:
 (6)  007020  023727  003430  104005                                           CMP     CMDWRD,#
 (9)  007026  001031                                                           BNE     50013$
2671  007030                                  IFB VFYFLG EQ #0 THEN ;IF DATA IS NOT TO BE VERIFIED THEN:
 (6)  007030  105737  003524                                                   TSTB    VFYFLG
 (9)  007034  001026                                                           BNE     50014$
2672  007036                                    LET RANB := RANB + RANS  ;GENERATE
 (6)  007036  063737  003444  003442                                           ADD     RANS,RAN
2673  007044                                    LET RANS := RANS + RANB  ;RANDOM
 (6)  007044  063737  003442  003444                                           ADD     RANB,RAN
2674  007052                                    LET BRFCNT := RANS       ;LENGTH
 (4)  007052  013737  003444  003426                                           MOV     RANS,BRF
2675  007060                                    LET BRFCNT := BRFCNT CLR.BY LENMSK  ;MASK RANDOM LENGTH.
 (6)  007060  043737  003440  003426                                           BIC     LENMSK,B
2676  007066                                    IF BRFCNT LT #18. THEN        ;DO NOT ALLOW BYTE COUNT OF LESS
 (6)  007066  023727  003426  000022                                           CMP     BRFCNT,#
 (9)  007074  002003                                                           BGE     50015$
2677  007076                                      LET BRFCNT := #18.  ;CHANGE COUNT OF 0-17 TO 18.
 (4)  007076  012737  000022  003426                                           MOV     #18.,BRF
2678  007104                                    ENDIF
 (4)  007104                                                                 50015$:
2679  007104                                    LET CMDPKT+CP.CNT := BRFCNT  ;MOVE BRF TO CMD PACKET.
 (4)  007104  013737  003426  002322                                           MOV     BRFCNT,C
2680  007112                                  ENDIF
 (4)  007112                                                                 50014$:
2681  007112                                ENDIF
 (4)  007112                                                                 50013$:
2682  007112                            ENDIF
 (4)  007112                                                                 50012$:
2683  007112  004737  007254            JSR PC,EXSUB                ;ISSUE CMD TO ALL,AWAIT INTS,CHECK STATU
2684  007116  004737  017222            JSR PC,CKHAE                ;CHECK HALT AFTER EACH CMD FLAG.
2685  007122                            LET R2 := #1                ;SET ALL UNITS AT BOT/EOT.
 (4)  007122  012702  000001                                                   MOV     #1,R2
2686  007126  004737  016614            JSR PC,FIRSTU               ;FIND FIRST UNIT.
```

```
2687  007132                              WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE UNITS:
 (4)  007132                                                                   50016$:
 (6)  007132  026527  002536  177777                                                     CMP     DEVTBL(R
 (9)  007140  001426                                                                     BEQ     50017$
2688  007142                                IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
 (6)  007142  032737  000400  003430                                                     BIT     #MOD.CO,
 (9)  007150  001406                                                                     BEQ     50020$
2689  007152                                  IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT THEN:
 (6)  007152  032765  000002  003510                                                     BIT     #XO.BOT,
 (9)  007160  001001                                                                     BNE     50021$
2690  007162                                    LET R2 := #0                 ;CLEAR EOT/BOT FLAG.
 (4)  007162  005002                                                                     CLR     R2
2691  007164                                  ENDIF
 (4)  007164                                                                   50021$:
2692  007164                                ELSE                          ;ELSE IF CMD IS NOT REVERSE:
 (4)  007164  000411                                                                     BR      50022$
 (3)  007166                                                                   50020$:
2693  007166                                  IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
 (6)  007166  032765  000001  003510                                                     BIT     #XO.EOT,
 (8)  007174  001404                                                                     BEQ     50023$
 (6)  007176  032737  000001  003430                                                     BIT     #CMD.CO,
 (9)  007204  001001                                                                     BNE     50024$
 (6)  007206                                                                   50023$:
2694                                                                  ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
2695  007206                                    LET R2 := #0                ;CLEAR EOT/BOT FLAG.
 (4)  007206  005002                                                                     CLR     R2
2696  007210                                  ENDIF
 (4)  007210                                                                   50024$:
2697  007210                                ENDIF
 (4)  007210                                                                   50022$:
2698  007210  004737  016662                JSR PC,NEXTU              ;FIND NEXT UNIT
2699  007214                              ENDDO                       ;
 (4)  007214  000746                                                                     BR      50016$
 (3)  007216                                                                   50017$:
2700  007216                              IF R2 EQ #1 THEN            ;IF ALL UNIT ARE AT EOT/BOT THEN:
 (6)  007216  020227  000001                                                             CMP     R2,#1
 (9)  007222  001001                                                                     BNE     50025$
2701  007224  000412                        BR        EXARTN          ;RETURN WITH R2 = #1.
2702  007226                              ENDIF
 (4)  007226                                                                   50025$:
2703  007226                              LET NCNT := NCNT + #1       ;UPDATE RECORD COUNT.
 (6)  007226  005237  003422                                                             INC     NCNT
2704  007232                              LET PCMDWD := CMDWRD        ;SAVE PREVIOUS COMMAND WORD.
 (4)  007232  013737  003430  003434                                                     MOV     CMDWRD,P
2705  007240                            ENDDO
 (4)  007240  000656                                                                     BR      50010$
 (3)  007242                                                                   50011$:
2706  007242  004737  015370            JSR    PC,VFYDAT            ;IF LAST CMD WAS A WRITE VERIFY, THEN GO
2707                                                                ;VERIFY THE LAST N RECORDS OF DATA.
2708  007246                          ENDDO
 (4)  007246  000646                                                                     BR      50006$
 (3)  007250                                                                   50007$:
2709  007250                          LET R2 := #0                  ;SET NORMAL RETURN INDICATOR.
 (4)  007250  005002                                                                     CLR     R2
2710  007252  000207          EXARTN: RTS PC                        ;RETURN.
2711
```

```
2712
2713
2714                                    ;       SUBROUTINE TO ISSUE COMMAND TO ALL DEVICES, WAIT FOR
2715                                    ;       ALL INTERRUPTS, AND CHECK ALL STATUS.
2716                                    ;       INPUTS:
2717                                    ;       OUTPUTS:
2718                                    ;       REGISTERS:
2719                                    ;       CALLS:          EXCUTE,GCWAIT,NEXTU,FIRSTU.
2720
2721  007254  004737  016614            EXSUB::     JSR PC,FIRSTU               ;SET UP FOR FIRST UNIT.
2722  007260                              WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE DEVICES:
 (4)  007260                                                                     50026$:
 (6)  007260  026527  002536  177777                                              CMP     DEVTBL(R
 (9)  007266  001465                                                            BEQ     50027$
2723  007270                                IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
 (6)  007270  032737  000400  003430                                            BIT     #MOD.CO,
 (9)  007276  001421                                                            BEQ     50030$
2724  007300                                  IF #X0.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
 (6)  007300  032765  000002  003510                                            BIT     #X0.BOT,
 (9)  007306  001014                                                            BNE     50031$
2725  007310                                    IF #X0.EOT SETIN EOTFLG(R5) THEN  ;BUT IF AT EOT
 (6)  007310  032765  000001  003510                                            BIT     #X0.EOT,
 (9)  007316  001406                                                            BEQ     50032$
2726  007320                                      IFB ALLEOT NE #0 THEN       ;AND ALL OTHERS AT EOT
 (6)  007320  105737  003532                                                    TSTB    ALLEOT
 (9)  007324  001402                                                            BEQ     50033$
2727  007326  004737  010576                        JSR PC,EXCUTE             ;THEN EXECUTE REV CMD
2728  007332                                      ENDIF                       ;IF NOT ALL AT EOT, FREEZE UNIT(
 (4)  007332                                                                     50033$:
2729  007332                                    ELSE                          ;IF NOT AT BOT AND
 (4)  007332  000402                                                            BR      50034$
 (3)  007334                                                                    50032$:
2730  007334  004737  010576                        JSR PC,EXCUTE             ;NOT AT EOT, EXEC REV CM
2731  007340                                      ENDIF
 (4)  007340                                                                    50034$:
2732  007340                                  ENDIF
 (4)  007340                                                                    50031$:
2733  007340                              ELSE                        ;ELSE IF CMD IS NOT REVERSE:
 (4)  007340  000435                                                            BR      50035$
 (3)  007342                                                                    50030$:
2734  007342                                IF CMDLG EQ #2 AND #X0.BOT SETIN EOTFLG(R5) THEN
 (6)  007342  023727  003436  000002                                            CMP     CMDLG,#2
 (9)  007350  001011                                                            BNE     50036$
 (6)  007352  032765  000002  0035:0                                            BIT     #X0.BOT,
 (9)  007360  001405                                                            BEQ     50036$
2735                                                        ;CLEAR BAD SPOT COUNTS WHEN WRITING FROM
2736  007362                                  LET BTPT := BTADDR(R5)
 (4)  007362  016537  002550  003520                                            MOV     BTADDR(R
2737  007370                                  LET @BTPT := #0
 (4)  007370  005077  174124                                                    CLR     @BTPT
2738  007374                                ENDIF
 (4)  007374                                                                    50036$:
2739  007374                                IF #X0.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
 (6)  007374  032765  000001  003510                                            BIT     #X0.EOT,
 (8)  007402  001404                                                            BEQ     50037$
 (6)  007404  032737  000001  003430                                            BIT     #CMD.CO,
```

```
  (9)  007412  001003                                                          BNE     50040$
  (6)  007414                                                          50037$:
2740                                                       ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
2741   007414  004737  010576            JSR PC,EXCUTE       ;ISSUE CMD TO TU80.
2742   007420                            ELSE
  (4)  007420  000405                                                          BR      50041$
  (3)  007422                                                          50040$:
2743   007422                            IFB ALLEOT NE #0 THEN
  (6)  007422  105737  003532                                                  TSTB    ALLEOT
  (9)  007426  001402                                                          BEQ     50042$
2744   007430  004737  010576              JSR PC,EXCUTE
2745   007434                            ENDIF
  (4)  007434                                                          50042$:
2746   007434                          ENDIF
  (4)  007434                                                          50041$:
2747   007434                         ENDIF
  (4)  007434                                                          50035$:
2748   007434  004737  016662         JSR       PC,NEXTU       ;FIND NEXT UNIT IN TEST CYCLE.
2749   007440                        ENDDO
  (4)  007440  000707                                                          BR      50026$
  (3)  007442                                                          50027$:
2750   007442                        IFB RPTFLG NE #0 THEN      ;IF REPORT HAS BEEN REQUESTED THEN:
  (6)  007442  105737  003525                                                  TSTB    RPTFLG
  (9)  007446  001403                                                          BEQ     50043$
2751   007450                          LET RPTFLG :B= #0        ;CLR THE FLAG.
  (4)  007450  105037  003525                                                  CLRB    RPTFLG
2752   007454                          DORPT                    ;PRINT THE PERFORMANCE REPORT.
  (3)  007454  104424                                                          TRAP    C$DRPT
2753   007456                        ENDIF                      ;
  (4)  007456                                                          50043$:
2754   007456  004737  016614        JSR PC,FIRSTU              ;SET UP FOR FIRST UNIT.
2755   007462                        WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE DEVICES:
  (4)  007462                                                          50044$:
  (6)  007462  026527  002536  177777                                          CMP     DEVTBL(R
  (9)  007470  001450                                                          BEQ     50045$
2756   007472                          IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
  (6)  007472  032737  000400  003430                                          BIT     #MOD.CO,
  (9)  007500  001421                                                          BEQ     50046$
2757   007502                            IF #X0.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
  (6)  007502  032765  000002  003510                                          BIT     #X0.BOT,
  (9)  007510  001014                                                          BNE     50047$
2758   007512                              IF #X0.EOT SETIN EOTFLG(R5) THEN      ;BUT IF AT EOT
  (6)  007512  032765  000001  003510                                          BIT     #X0.EOT,
  (9)  007520  001406                                                          BEQ     50050$
2759   007522                                IFB ALLEOT NE #0 THEN              ;AND ALL OTHERS AT EOT
  (6)  007522  105737  003532                                                  TSTB    ALLEOT
  (9)  007526  001402                                                          BEQ     50051$
2760   007530  004737  011166                  JSR PC,GOWAIT                    ;THEN WAIT FOR CMD END
2761   007534                                ENDIF                              ;IF NOT ALL AT EOT, DO N
  (4)  007534                                                          50051$:
2762   007534                              ELSE                                 ;NOT AT BOT, AND NOT AT
  (4)  007534  000402                                                          BR      50052$
  (3)  007536                                                          50050$:
2763   007536  004737  011166                JSR PC,GOWAIT                            ;WAIT FOR INT,CH
2764   007542                              ENDIF
  (4)  007542                                                          50052$:
```

```
2765  007542                                                        ENDIF
 (4)  007542                                                                                      50047$:
2766  007542                                                      ELSE                      ;ELSE IF CMD IS FORWARD:
 (4)  007542  000420                                                                                     BR      50053$
 (3)  007544                                                                                      50046$:
2767  007544                                                        IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
 (6)  007544  032765  000001  003510                                                                     BIT     #XO.EOT,
 (8)  007552  001404                                                                                     BEQ     50054$
 (6)  007554  032737  000001  003430                                                                     BIT     #CMD.CO,
 (9)  007562  001003                                                                                     BNE     50055$
 (6)  007564                                                                                      50054$:
2768                                                                                        ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
2769  007564  004737  011166                                          JSR PC,GOWAIT           ;WAIT FOR INT,CHECK STATUS.
2770  007570                                                        ELSE
 (4)  007570  000405                                                                                     BR      50056$
 (3)  007572                                                                                      50055$:
2771  007572                                                          IFB ALLEOT NE #0 THEN
 (6)  007572  105737  003532                                                                             TSTB    ALLEOT
 (9)  007576  001402                                                                                     BEQ     50057$
2772  007600  004737  011166                                            JSR PC,GOWAIT
2773  007604                                                          ENDIF
 (4)  007604                                                                                      50057$:
2774  007604                                                        ENDIF
 (4)  007604                                                                                      50056$:
2775  007604                                                      ENDIF
 (4)  007604                                                                                      50053$:
2776  007604  004737  016662                                      JSR         PC,NEXTU        ;FIND NEXT UNIT IN TEST CYCLE.
2777  007610                                                    ENDDO
 (4)  007610  000724                                                                                     BR      50044$
 (3)  007612                                                                                      50045$:
2778  007612  000207                                            RTS PC                      ;RETURN.
```

```
2780                                    ;       THIS SUBROUTINE STORES THE ASCII FOR THE CURRENT COMMAND AND PREVIOUS
2781                                    ;       COMMAND IN THE STANDARD ERROR MESSAGE.  ON ENTRY LOCATION CMDWRD
2782                                    ;       CONTAINS CURRENT CMD AND LOCATION PCMDWD CONTAINS PREVIOUS CMD.
2783                                    ;       INPUTS:
2784                                    ;       OUTPUTS:
2785                                    ;       REGISTERS:      R3, R4.
2786                                    ;       CALLS:          GCMDA
2787
2788  007614                            CMDAC:: LET R4 := CMDWRD                ;R4 = CMD BINARY.
 (4)  007614  013704  003430                                                            MOV     CMDWRD,R
2789  007620  004737  007670                    JSR     PC,GCMDA                ;GET CMD ASCII.
2790  007624  112337  006156                    MOVB    (R3)+,STAER1+2          ;MOVE CMD ASCII
2791  007630  112337  006157                    MOVB    (R3)+,STAER1+3          ;
2792  007634  111337  006160                    MOVB    (R3),STAER1+4           ;INTO MSG.
2793  007640                                    LET R4 := PCMDWD                ;R4 = PREVIOUS CMD BINARY.
 (4)  007640  013704  003434                                                            MOV     PCMDWD,R
2794  007644  004737  007670                    JSR     PC,GCMDA                ;GET CMD ASCII.
2795  007650  000240                            NOP
2796  007652                                    LET STAER7+24 :B= (R3)+         ;MOVE CMD ASCII
 (4)  007652  112337  006272                                                            MOVB    (R3)+,ST
2797  007656                                    LET STAER7+25 :B= (R3)+         ;
 (4)  007656  112337  006273                                                            MOVB    (R3)+,ST
2798  007662                                    LET STAER7+26 :B= (R3)          ;INTO MSG.
 (4)  007662  111337  006274                                                            MOVB    (R3),STA
2799  007666  000207                            RTS     PC                      ;RETURN. GO EXECUTE NEXT FUNCTION
2800
2801
2802
2803                                    ;       SUBROUTINE TO FIND THE ASCII EQUIVILENT OF THE COMMAND IN R4.
2804                                    ;       ADDRESS OF ASCII 1ST WORD IS RETURNED IN R3.
2805                                    ;       INPUTS:         R4 = PRESENT COMMAND WORD.
2806                                    ;       OUTPUTS:        R3 = ADDRESS OF PRESENT COMMAND ASCII.
2807                                    ;       REGISTERS:
2808                                    ;       CALLS:
2809
2810  007670                            GCMDA:: LET R3 := #0                    ;INIT CMD TBL POINTER.
 (4)  007670  005003                                                                    CLR     R3
2811  007672                                    WHILE CMDTBL(R3) NE R4 DO       ;UNTIL CURRENT CMD IS FOUND:
 (4)  007672                                                                    50060$:
 (6)  007672  026304  003646                                                            CMP     CMDTBL(R
 (9)  007676  001403                                                                    BEQ     50061$
2812  007700                                      LET R3 := R3 + #2             ;SEARCH CMD TABLE.
 (6)  007700  062703  000002                                                            ADD     #2,R3
2813  007704                                    ENDDO
 (4)  007704  000772                                                                    BR      50060$
 (3)  007706                                                                    50061$:
2814  007706                                    LET R4 := R3
 (4)  007706  010304                                                                    MOV     R3,R4
2815  007710                                    LET R3 := R3 SHIFT -1           ;POINT TO ASCII FOR THAT COMMAND
 (7)  007710  006203                                                                    ASR     R3
2816  007712  060403                            ADD     R4,R3
2817  007714  062703  003734                    ADD     #CMDASC,R3
2818  007720  000207                            RTS     PC                      ;RETURN.
```

```
                        ;       THIS SUBROUTINE LOADS THE TU80 COMMAND PACKET FROM ONE
                        ;       ENTRY IN THE SEQUENCE TABLE.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:      R2, R3.
                        ;       CALLS:          GENPAT.

007722                  SETUP:: LET CMDLG := #0                 ;CLR CMD LOGGING CODE(DISABLES LOGGING)
(4) 007722 005037 003436                                                        CLR     CMDLG
007726 012137 002314            MOV     (R1)+,CMDPKT            ;LOAD THE COMMAND WORD.
007732 011137 002322            MOV     (R1),CMDPKT+CP.CNT      ;LOAD THE BYTE/RECORD/FILE COUNT.
007736 011137 003426            MOV     (R1),BRFCNT             ;SAVE BRF FOR THIS COMMAND.
007742 013702 002314            MOV     CMDPKT,R2               ;GET CMD.
007746 042702 177740            BIC     #NCMD.C,R2              ;CLR ALL BUT CMD BITS.
007752 010203                   MOV     R2,R3                   ;SAVE IT TWICE.
007754 162703 000010            SUB     #CMD.C3,R3              ;POSITION COMMAND?
007760 001003                   BNE     2$                      ;BR IF NOT.
007762 011137 002316            MOV     (R1),CMDPKT+2           ;MOVE BPCR IN 2ND PKT WORD FOR POSITION
007766 000461                   BR      3$
007770                  2$:     IF CMDPKT EQ #WTM THEN          ;IF CMD IS A WRITE TAPE MARK THEN:
(6) 007770 023727 002314 100011                                                 CMP     CMDPKT,#
(9) 007776 001003                                                               BNE     50062$
010000                            LET CMDLG := #2                 ;WTM LOGGING CODE IS 2.
(4) 010000 012737 000002 003436                                                 MOV     #2,CMDLG
010006                          ENDIF
(4) 010006                                                                    50062$:
010006 010203                   MOV     R2,R3
010010 162703 000001            SUB     #CMD.C0,R3              ;IS IT A READ?
010014 001017                   BNE     1$                      ;BR IF NOT.
010016 013737 003420 002316     MOV     DATARD,CMDPKT+CP.ADL    ;IF SO, LOAD THE BUFFER ADDR.
010024                          IF #MOD.C0 SETIN CMDPKT THEN    ;IF CMD IS A READ REV THEN:
(6) 010024 032737 000400 002314                                                 BIT     #MOD.C0,
(9) 010032 001404                                                               BEQ     50063$
010034                            LET CMDLG := #4                 ;LOGGING CODE IS 4.
(4) 010034 012737 000004 003436                                                 MOV     #4,CMDLG
010042                          ELSE                            ;ELSE - IF CMD IS A READ FWD:
(4) 010042 000403                                                               BR      50064$
(3) 010044                                                                    50063$:
010044                            LET CMDLG := #6                 ;LOGGING CODE IS 6.
(4) 010044 012737 000006 003436                                                 MOV     #6,CMDLG
010052                          ENDIF
(4) 010052                                                                    50064$:
010052 000427                   BR      3$                      ;CONTINUE.
010054 010203           1$:     MOV     R2,R3                   ;IS IT
010056 162703 000004            SUB     #CMD.C2,R3              ;A SET CHARACTERISTICS CMD?
010062 001011                   BNE     4$                      ;BR IF NOT.
010064                          LET CMDPKT+CP.ADL := #SCHBK     ;SET UP ADR LO FOR SET CHAR.
(4) 010064 012737 002446 002316                                                 MOV     #SCHBK,C
010072 012737 000010 002322     MOV     #SCHCNT,CMDPKT+CP.CNT   ;SET BUFFER EXTENT
010100                          LET SCHBK+6 := (R1)             ;STORE CHARACTERISTIC CODE IN SCH BLOCK.
(4) 010100 011137 002454                                                        MOV     (R1),SCH
010104 000412                   BR      3$                      ;CONTINUE.
010106 010203           4$:     MOV     R2,R3                   ;IS IT
010110 162703 000006            SUB     #CMD.C1!CMD.C2,R3       ;A DIAGNOSTIC (DIA) CMD?
010114 001006                   BNE     3$                      ;BR IF NOT.
010116 012737 000020 002322     MOV     #DIACNT,CMDPKT+CP.CNT   ;LOAD BUFFER EXTENT.
```

```
2862  010124  012737  003416  002316          MOV     #DIABLK,CMDPKT+CP.ADL   ;LOAD BUFFER ADR LOW.
2863  010132  005721                  3$:     TST     (R1)+                   ;POINT TO N (NUMBER OF TIMES TO EXECUTE
2864  010134                                  LET NCNT1 := (R1)+              ;SAVE NUMBER OF OPERATIONS
 (4)  010134  012137  003424                                                          MOV     (R1)+,NC
2865  010140                                  LET NCNT := #0                  ;CLEAR OPERATION COUNTER.
 (4)  010140  005037  003422                                                          CLR     NCNT
2866  010144  012137  003456                  MOV     (R1)+,PATERN            ;SAVE PATTERN CODE FOR CURRENT CMD.
2867  010150  010203                          MOV     R2,R3                   ;IS IT
2868  010152  162703  000005                  SUB     #CMD.CO!CMD.C2,R3       ;A WRITE?
2869  010156  001010                          BNE     5$                      ;BR IF NOT.
2870  010160  013737  003416  002316          MOV     DATAWT,CMDPKT+CP.ADL    ;LOAD WRITE BUFFER LO ORDER.
2871  010166  004737  010300                  JSR     PC,GENPAT               ;GO GENERATE THE WRITE PATTERN.
2872  010172                                  LET CMDLG := #2                 ;WRITE LOGGING CODE IS 2.
 (4)  010172  012737  000002  003436                                                  MOV     #2,CMDLG
2873  010200                          5$:     IF #VFY.C SETIN CMDPKT THEN     ;IF DATA VERIFICATION IS REQUIRED:
 (6)  010200  032737  000100  002314                                                  BIT     #VFY.C,C
 (9)  010206  001407                                                                  BEQ     50065$
2874  010210                                    LET VFYFLG :B= #1             ;SET VERIFY FLAG.
 (4)  010210  112737  000001  003524                                                  MOVB    #1,VFYFL
2875  010216  042737  000100  002314           BIC   #VFY.C,CMDPKT           ;CLEAR VERIFY BIT(NOT USED BY HARDWARE).
2876  010224                                  ELSE                            ;IF DATA VERIFICATION IS NOT REQUIRED:
 (4)  010224  000402                                                                  BR      50066$
 (3)  010226                                                                  50065$:
2877  010226                                    LET VFYFLG :B= #0             ;CLR VERIFY FLAG.
 (4)  010226  105037  003524                                                          CLRB    VFYFLG
2878  010232                                  ENDIF
 (4)  010232                                                                  50066$:
2879  010232                                  LET PCMDWD := CMDWRD            ;SAVE PREVIOUS CMD WORD.
 (4)  010232  013737  003430  003434                                                  MOV     CMDWRD,P
2880  010240                                  LET CMDWRD := CMDPKT            ;SAVE PRESENT CMD WORD.
 (4)  010240  013737  002314  003430                                                  MOV     CMDPKT,C
2881  010246                                  IFB SWBFLG NE #0 THEN           ;IF SWAP BYTES IS ENABLED:
 (6)  010246  105737  003526                                                          TSTB    SWBFLG
 (9)  010252  001403                                                                  BEQ     50067$
2882  010254                                    LET CMDPKT := CMDPKT SET.BY #SWB.C  ;SET SWAP BIT IN COMMAND.
 (6)  010254  052737  010000  002314                                                  BIS     #SWB.C,C
2883  010262                                  ENDIF
 (4)  010262                                                                  50067$:
2884  010262  042737  004000  002314          BIC     #BRF.C,CMDPKT           ;CLR BRF BIT (INTERNAL ONLY).
2885  010270                                  LET CMDSAV := CMDPKT            ;SAVE 1ST WORD OF COMMAND PACKET.
 (4)  010270  013737  002314  003432                                                  MOV     CMDPKT,C
2886  010276  000207                          RTS     PC                      ;RETURN.
```

```
2888                            ;       THIS SUBROUTINE SETS UP AND CALLS THE APPROPRIATE SUBROUTINE TO GENERAT
2889                            ;       THE DESIRED PATTERN FOR THE WRITE AND WRITE/VERIFY COMMANDS.
2890                            ;       INPUTS:
2891                            ;       OUTPUTS:
2892                            ;       REGISTERS:      R2, R3, R4.
2893                            ;       CALLS:          PATR0 - PATR7
2894
2895  010300                    GENPAT:: LET R3 := PATERN SHIFT 1       ;SETUP PATTERN ROUTINE POINTER
 (4)  010300  013703  003456                                                    MOV     PATERN,R
 (7)  010304  006303                                                            ASL     R3
2896  010306                            LET R4 := BRFCNT + #1   ;SET LENGTH OF WRITE BFR
 (4)  010306  013704  003426                                                    MOV     BRFCNT,R
 (6)  010312  005204                                                            INC     R4
2897  010314                            LET R4 := R4 CLR.BY #1  ;ROUNDED UP TO NEXT WORD
 (6)  010314  042704  000001                                                    BIC     #1,R4
2898  010320                            LET R4 := R4 - #2       ;WITH FIRST WORD RESERVED
 (6)  010320  162704  000002                                                    SUB     #2,R4
2899  010324                            LET R2 := DATAWT + #2   ;FOR RECORD COUNT
 (4)  010324  013702  003416                                                    MOV     DATAWT,R
 (6)  010330  062702  000002                                                    ADD     #2,R2
2900  010334  004773  010342           JSR     PC,@PATTBL(R3)  ;GO GENERATE THE APPROPRIATE PATTERN.
2901  010340  000207                   RTS     PC              ;RETURN TO SETUP SUBROUTINE.
2902
2903                            ;       TU80 WRITE PATTERN LOOKUP TABLE.  USED TO JSR TO THE
2904                            ;       CORRECT DATA PATTERN GENERATING ROUTINE.
2905
2906  010342  010364            PATTBL: PATR0                   ; INCREMENTING PATTERN, 0 - 377
2907  010344  010422                    PATR1                   ; ALL ONES PATTERN
2908  010346  010442                    PATR2                   ; ALL ZEROES PATTERN
2909  010350  010452                    PATR3                   ; '1' BIT SHIFT, RIGHT TO LEFT
2910  010352  010476                    PATR4                   ; '0' BIT SHIFT, RIGHT TO LEFT
2911  010354  010510                    PATR5                   ; ALTERNATE '0' & '1' WITH ALT. BYTES COMPL.
2912  010356  010522                    PATR6                   ; ALTERNATE BYTES OF 000 AND 377
2913  010360  010542                    PATR7                   ; RANDOM PATTERN.
2914  010362  010574                    PATR8                   ; DUMMY.  NO PATTERN, JUST EXITS.
2915
2916
2917                            ;       INCREMENTING PATTERN.  0 - 377.
2918
2919  010364                    PATR0:: LET R3 := #400
 (4)  010364  012703  000400                                                    MOV     #400,R3
2920  010370                    1$:     LET R4 := R4 - #2       ;DECREMENT WORD COUNT.
 (6)  010370  162704  000002                                                    SUB     #2,R4
2921  010374  100411                   BMI     2$              ;BR IF DONE.
2922  010376                            LET (R2)+ := R3         ;STORE DATA WORD.
 (4)  010376  010322                                                            MOV     R3,(R2)+
2923  010400                            LET R3 := R3 + #1002    ;UPDATE PATTERN.
 (6)  010400  062703  001002                                                    ADD     #1002,R3
2924  010404                            IF R3 EQ #1000 THEN     ;IF PATTERN HAS WRAPPED AROUND THEN:
 (6)  010404  020327  001000                                                    CMP     R3,#1000
 (9)  010410  001002                                                            BNE     50070$
2925  010412                              LET R3 := #400          ;INIT THE PATTERN AGAIN.
 (4)  010412  012703  000400                                                    MOV     #400,R3
2926  010416                            ENDIF
 (4)  010416                                                            50070$:
2927  010416  000764                   BR      1$              ;DO IT AGAIN.
```

```
 2928  010420  000207                    2$:     RTS     PC              ;RETURN.
 2929
 2930                                            ;ALL ONE'S PATTERN.
 2931
 2932  010422  012703  177777            PATR1:: MOV     #-1,R3          ;ALL ONES PATTERN;.
 2933  010426                            ZROPAT: LET R4 := R4 - #2       ;DECREMENT BYTE COUNT.
   (6) 010426  162704  000002                                                    SUB     #2,R4
 2934  010432  100402                            BMI     1$              ;DONE?,BR IF YES.
 2935  010434  010322                            MOV     R3,(R2)+        ;IF NOT LOAD NEXT BYTE WITH PATTERN.
 2936  010436  000773                            BR      ZROPAT          ;DO IT AGAIN.
 2937
 2938  010440  000207                    1$:     RTS     PC              ;RETURN.
```

```
                                        ;ALL ZEROES PATTERN.
                                
010442  005003                  PATR2:: CLR     R3              ;CLR PATTERN REGISTER.
010444  004737  010426                  JSR     PC,ZROPAT       ;GO GENERATE IT.
010450  000207                          RTS     PC              ;RETURN.
                                
                                        ;ONE BIT WALKING FROM R TO L IN A FIELD OF ZEROES.
                                
010452  012703  000401          PATR3:: MOV     #401,R3         ;INIT PATTERN REGISTER.
010456                          WLKZRO: LET R4 := R4 - #2       ;DECREMENT WORD COUNT.
(6)     010456  162704  000002                                                   SUB     #2,R4
010462  100404                          BMI     1$              ;BR IF DONE.
010464  010322                          MOV     R3,(R2)+        ;LOAD DATA.
010466  006303                          ASL     R3              ;SHIFT PATTERN.
010470  005503                          ADC     R3              ;ADD CARRY BACK INTO PATTERN.
010472  000771                          BR      WLKZRO          ;DO IT AGAIN.
010474  000207                  1$:     RTS     PC              ;RETURN.
                                
                                        ;ZERO BIT WALKING FROM R TO L IN A FIELD OF 1'S.
                                
010476  012703  177376          PATR4:: MOV     #177376,R3      ;INIT PATTERN REGISTER.
010502  004737  010456                  JSR     PC,WLKZRO       ;GO GENERATE ;IT.
010506  000207                          RTS     PC              ;RETURN.
                                
                                        ;ALTERNATING ONE AND ZERO BITS WITH ALTERNATE BYTES
                                        ;COMPLEMENTED.
                                
010510  012703  125125          PATR5:: MOV     #125125,R3      ;INIT PATTERN REGISTER.
010514  004737  010426                  JSR     PC,ZROPAT       ;GO GENERATE IT.
010520  000207                          RTS     PC              ;RETURN.
                                
                                        ;ALTERNATING BYTES OF 000 AND 377.
                                
010522  012703  177400          PATR6:: MOV     #177400,R3      ;INIT PATTERN REGISTER.
010526                          1$:     LET R4 := R4 - #2       ;DECREMENT WORD COUNT.
(6)     010526  162704  000002                                                   SUB     #2,R4
010532  100402                          BMI     2$              ;BR IF DONE.
010534  010322                          MOV     R3,(R2)+        ;LOAD DATA.
010536  000773                          BR      1$              ;DO IT AGAIN.
010540  000207                  2$:     RTS     PC              ;RETURN.
                                
                                        ;RANDOM PATTERN GENERATOR
                                
010542                          PATR7:: LET R4 := R4 - #2       ;DECREMENT WORD COUNT
(6)     010542  162704  000002                                                   SUB     #2,R4
010546  100411                          BMI     GIT             ;BR IF DONE.
010550  063737  003444  003442          ADD     RANS,RANB
010556  063737  003442  003444          ADD     RANB,RANS       ;GET NEW #.
010564  013722  003444                  MOV     RANS,(R2)+      ;SAVE #.
010570  000764                          BR      PATR7           ;CONTINUE.
010572  000207                  GIT:    RTS     PC              ;RETURN
                                
                                ;       NO PATTERN GENERATION.
                                
010574  000207                  PATR8:: RTS     PC              ;RETURN.
```

```
                        ;       THIS SUBROUTINE INITIATES TU80 COMMAND EXECUTION
                        ;       AND CHECKS FOR TU80 RESPONSE.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:      R2, R3.
                        ;       CALLS:          DROPU, MOVMSG, FIRSTU, NEXTU, WSSR.

                        EXCUTE::LET TIME1 := #-1                ;INIT TIMEOUT COUNTER.
                                                                                MOV     #-1,TIMF
                                REPEAT                          ;WAIT -
                                                                            50071$:
                                  LET TIME1 := TIME1 - #1       ;UPDATE TIMEOUT COUNTER.
                                                                                DEC     TIME1
                                  IF TIME1 EQ #0 THEN           ;IF TIMED OUT:
                                                                                TST     TIME1
                                                                                BNE     50072$
                                    JSR PC,MOVMSG               ;MOVE CURRENT PACKET MSG.
                                    ERRDF 2,'SSRM,STAERM        ;REPORT TU80 NOT READY
                                                                                TRAP    C$ERDF
                                                                                .WORD   2
                                                                                .WORD   NSSRM
                                                                                .WORD   STAERM
                                    JSR PC,DROPU                ;DROP THE UNIT.
                                    BR  EXCRTN                  ;RETURN.
                                  ENDIF
                                                                            50072$:
                                UNTIL #TS.SSR SETIN @TSSR(R5)   ;WAIT UNTIL DEVICE IS READY.
                                                                                BIT     #TS.SSR,
                                                                                BEQ     50071$
                                IF CMDWRD EQ #SCH THEN          ;IF WE ARE DOING A SET CHAR CMD THEN:
                                                                                CMP     CMDWRD,#
                                                                                BNE     50073$
                                  LET R5SAVE := R5              ;SAVE CURRENT DEVICE POINTER.
                                                                                MOV     R5,R5SAV
                                  JSR   PC,FIRSTU               ;FIND FIRST UNIT.
                                  WHILE DEVTBL(R5) NE #END DO
                                                                            50074$:
                                                                                CMP     DEVTBL(R
                                                                                BEQ     50075$
                                    JSR PC,WSSR                   ;WAIT FOR UNIT READY OR TIME OUT.
                                    JSR PC,NEXTU                  ;FIND NEXT UNIT.
                                  ENDDO
                                                                                BR      50074$
                                                                            50075$:
                                  LET R5 := R5SAVE              ;RESTORE CURRENT DEVICE POINTER.
                                                                                MOV     R5SAVE,R
                                  LET SCHBK := MSGPKA(R5)       ;SET UP ADR OF MSG PKT IN SCH BLOCK.
                                                                                MOV     MSGPKA(R
                                ENDIF
                                                                            50073$:
                                LET R3 := MSGPKA(R5)            ;ADR OF THIS UNIT'S MSG PACKET.
                                                                                MOV     MSGPKA(R
                                LET R2 := #0                    ;CLR COUNTER.
                                                                                CLR     R2
                                WHILE R2 NE #MSGCNT DO          ;WHILE THERE ARE MORE LOCATIONS:
                                                                            50076$:
```

```
   (6)  010732  020227  000016                                                          CMP     R2,#MSGC
   (9)  010736  001405                                                                 BEQ     50077$
3023    010740                        LET (R3)+ := #-1              ;INIT THE MSG PACKET WITH ALL 1'S
   (4)  010740  012723  177777                                                         MOV     #-1,(R3)
3024    010744                        LET R2 := R2 + #2             ;UPDATE COUNTER.
   (6)  010744  062702  000002                                                         ADD     #2,R2
3025    010750                     ENDDO
   (4)  010750  000770                                                                 BR      50076$
   (3)  010752                                                                50077$:
3026    010752  105737  002212     TSTB    DINT                     ;ARE INTERRUPTS DISABLED.
3027    010756  001023              BNE     1$                       ;BR IF YES.
3028    010760                     IFB INTFLG(R5) GT #1 THEN         ;IF MORE THAN ONE INTERRUPT HAS OCCURED:
   (6)  010760  126527  003500  000001                                                 CMPB    INTFLG(R
   (9)  010766  003412                                                                 BLE     50100$
3029    010770                        LET TSSREG := @TSSR(R5)                ;FREEZE THE CURRENT STATUS REG F
   (4)  010770  017537  002466  003464                                                MOV     @TSSR(R5
3030    010776                        ERRDF 15,TOOMM,STAERM   ;REPORT TOO MANY INTERRUPTS.
   (4)  010776  104455                                                                 TRAP    C$ERDF
   (5)  011000  000017                                                                 .WORD   15
   (5)  011002  004544                                                                 .WORD   TOOMM
   (5)  011004  005642                                                                 .WORD   STAERM
3031    011006  004737  016716        JSR PC,DROPU                  ;DROP THE UNIT
3032    011012  000464                BR  EXCRTN                    ;RETURN - UNIT HAS BEEN DROPPED.
3033    011014                     ENDIF
   (4)  011014                                                                50100$:
3034    011014                     LET INTFLG(R5) := #0              ;CLR INTERRUPT FLAG FOR THIS DEV.
   (4)  011014  005065  003500                                                         CLR     INTFLG(R
3035    011020  052737  000200  002314     BIS     #IE.C,CMDPKT     ;SET INT ENABLE BIT.
3036    011026                  1$:        IFB ERRREC EQ #0 THEN     ;IF NOT RETRYING
   (6)  011026  105737  003477                                                         TSTB    ERRREC
   (9)  011032  001005                                                                 BNE     50101$
3037    011034                        LET RECCNT(R5) := RECCNT(R5) + #1
   (6)  011034  005265  003330                                                         INC     RECCNT(R
3038    011040                        LET @DATAWT := RECCNT(R5)     ;THEN UPDATE REC COUNT TO WRITE IT ON TA
   (4)  011040  016577  003330  172350                                                MOV     RECCNT(R
3039    011046                     ENDIF
   (4)  011046                                                                50101$:
3040    011046                     IF L$TEST EQ #3 AND CMDWRD EQ #WRT THEN
   (6)  011046  023727  002114  000003                                                 CMP     L$TEST,#
   (9)  011054  001024                                                                 BNE     50102$
   (6)  011056  023727  003430  104005                                                 CMP     CMDWRD,#
   (9)  011064  001020                                                                 BNE     50102$
3041    011066                        IF NCNT GT #3500. THEN
   (6)  011066  023727  003422  006654                                                 CMP     NCNT,#35
   (9)  011074  003414                                                                 BLE     50103$
3042    011076                        DELAY 15.
   (2)  011076  012727  000017                                                         MOV     #15.,(PC
   (2)  011102  000000                                                                 .WORD   0
   (2)  011104  013727  002116                                                         MOV     L$DLY,(P
   (2)  011110  000000                                                                 .WORD   0
   (2)  011112  005367  177772                                                         DEC     -6(PC)
   (2)  011116  001375                                                                 BNE     .-4
   (2)  011120  005367  177756                                                         DEC     -22(PC)
   (2)  011124  001367                                                                 BNE     .-20
3043    011126                        ENDIF
   (4)  011126                                                                50103$:
```

```
3044  011126                                     ENDIF
 (4)  011126                                                                              50102$:
3045  011126  012775  002314  002456             MOV     #CMDPKT,@TSDB(R5)      ;LOAD TSDB WITH CMDPKT ADDRESS
3046                                                                            ;THIS INITIATES COMMAND EXECUTION.
3047  011134                                     IF #TS.SSR SETIN @TSSR(R5) THEN ;IF READY DID NOT DROP THEN:
 (6)  011134  032775  000200  002466                                                              BIT     #TS.SSR,
 (9)  011142  001410                                                                              BEQ     50104$
3048  011144  004737  011552                       JSR   PC,MOVMSG                ;MOVE CURRENT MESSAGE PACKET TO COMMON
3049  011150                                       ERRDF 3,TOERM,STAERM           ;REPORT NO TU80 RESPONSE.
 (4)  011150  104455                                                                              TRAP    C$ERDF
 (5)  011152  000003                                                                              .WORD   3
 (5)  011154  004273                                                                              .WORD   TOERM
 (5)  011156  005642                                                                              .WORD   STAERM
3050  011160  004737  016716                      JSR PC,DROPU                    ;DROP THE UNIT
3051  011164                                     ENDIF
 (4)  011164                                                                              50104$:
3052  011164  000207                     EXCRTN: RTS     PC                     ;RETURN.
```

```
                        ;       THIS SUBROUTINE WAITS FOR THE TU80 INERRUPT OR DONE BIT TO SET AND ALLOW
                        ;       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.

011166                  GOWAIT::IF DEVTBL(R5) EQ #NINUSE THEN
(6) 011166  026527  002536  177774                                       CMP     DEVTBL(R
(9) 011174  001002                                                      BNE     50105$
011176  000540                BR      1$
011200                  ELSE
(4) 011200  000400                                                      BR      50106$
(3) 011202                                                      50105$:
011202                  ENDIF
(4) 011202                                                      50106$:
011202                  LET TIME1 := #-1                ;INIT TIME OUT COUNTER.
(4) 011202  012737  177777  003446                                      MOV     #-1,TIME
011210                  REPEAT                          ;REPEAT UNTIL INTERRUPT OCCURES:
(3) 011210                                                      50107$:
011210                    BREAK                         ;GO TO THE SUPER TO ALLOW TTY INPUT.
(3) 011210  104422                                                      TRAP    C$BRK
011212                    IF CMDWRD EQ #RWD THEN        ;IF COMMAND WAS REWIND THEN:
(6) 011212  023727  003430  102010                                      CMP     CMDWRD,#
(9) 011220  001014                                                      BNE     50110$
011222                      DELAY 10.                   ;WAIT  EXTRA 10 MSECS EACH LOOP.
(2) 011222  012727  000012                                              MOV     #10.,(PC
(2) 011226  000000                                                      .WORD   0
(2) 011230  013727  002116                                              MOV     L$DLY,(P
(2) 011234  000000                                                      .WORD   0
(2) 011236  005367  177772                                              DEC     -6(PC)
(2) 011242  001375                                                      BNE     .-4
(2) 011244  005367  177756                                              DEC     -22(PC)
(2) 011250  001367                                                      BNE     .-20
011252                  ENDIF
(4) 011252                                                      50110$:
011252                  IF CMDWRD EQ #SFF OR CMDWRD EQ #SFR THEN
(6) 011252  023727  003430  105010                                      CMP     CMDWRD,#
(8) 011260  001404                                                      BEQ     50111$
(6) 011262  023727  003430  105410                                      CMP     CMDWRD,#
(9) 011270  001014                                                      BNE     50112$
(6) 011272                                                      50111$:
011272                    DELAY 12.                   ;ADD DELAY FOR SPACE TAPE MARK COMMANDS
(2) 011272  012727  000014                                              MOV     #12.,(PC
(2) 011276  000000                                                      .WORD   0
(2) 011300  013727  002116                                              MOV     L$DLY,(P
(2) 011304  000000                                                      .WORD   0
(2) 011306  005367  177772                                              DEC     -6(PC)
(2) 011312  001375                                                      BNE     .-4
(2) 011314  005367  177756                                              DEC     -22(PC)
(2) 011320  001367                                                      BNE     .-20
011322                  ENDIF
(4) 011322                                                      50112$:
011322                  IFB DINT EQ #0 THEN             ;IF INTERRUPTS ARE ENABLED.
```

```
   (6)  011322  105737  002212                                                    TSTB    DINT
   (9)  011326  001003                                                            BNE     50113$
3077    011330
   (4)  011330  016502  003500        LET R2 := INTFLG(R5)        ;FETCH INTERRUPT OCCURRED FLAG.
                                                                                  MOV     INTFLG(R
3078    011334                      ELSE                        ;IF IN BRUTUS MODE:
   (4)  011334  000406                                                            BR      50114$
   (3)  011336                                                                 50113$:
3079    011336        LET R3 := COMP #TS.SSR      ;SET UP A MASK FOR THE DONE BIT.
   (6)  011336  012703  000200                                                    MOV     #TS.SSR,
   (6)  011342  005103                                                            COM     R3
3080    011344        LET R2 := @TSSR(R5) CLR.BY R3  ;FETCH DONE BIT.
   (4)  011344  017502  002466                                                    MOV     @TSSR(R5
   (6)  011350  040302                                                            BIC     R3,R2
3081    011352                      ENDIF
   (4)  011352                                                                 50114$:
3082    011352        LET TIME1 := TIME1 - #1       ;UPDATE TIMEOUT COUNTER.
   (6)  011352  005337  003446                                                    DEC     TIME1
3083    011356      UNTIL R2 NE #0 OR TIME1 EQ #0   ;REPEAT UNTIL INTERRUPT OR READY OCCURES
   (4)  011356  005702                                                            TST     R2
   (6)  011360  001003                                                            BNE     50115$
   (4)  011362  005737  003446                                                    TST     TIME1
   (7)  011366  001310                                                            BNE     50107$
   (4)  011370                                                                 50115$:
3084    011370      IF TIME1 EQ #0 THEN             ;IF TIME OUT HAS OCCURRED:
   (6)  011370  005737  003446                                                    TST     TIME1
   (9)  011374  001022                                                            BNE     50116$
3085    011376        LET @DATAWT := RECCNT(R5) - #1 ;RE-ADJUST REC COUNT DOWN
   (4)  011376  016577  003330  172012                                           MOV     RECCNT(R
   (6)  011404  005377  172006                                                    DEC     @DATAWT
3086    011410  004737  011552        JSR   PC,MOVMSG             ;MOVE CURRENT MSG PACKET TO COMMON AREA.
3087    011414        ERRDF 4,NOINTM,STAERM ;REPORT NO INTERRUPT.
   (4)  011414  104455                                                            TRAP    C$ERDF
   (5)  011416  000004                                                            .WORD   4
   (5)  011420  004505                                                            .WORD   NOINTM
   (5)  011422  005642                                                            .WORD   STAERM
3088    011424  004737  016716          JSR PC,DROPU          ;DROP THE UNIT.
3089    011430        LET R3 := #ENDERF
   (4)  011430  012703  003500                                                    MOV     #ENDERF,
3090    011434  004737  011502          JSR PC,CLRERR                 ;CLEAR ALL ERROR FLAGS
3091    011440                      ELSE
   (4)  011440  000417                                                            BR      50117$
   (3)  011442                                                                 50116$:
3092    011442  004737  011552          JSR   PC,MOVMSG               ;MOVE CURRENT MSG. PACKET TO COMMON AREA
3093    011446  004737  011636          JSR   PC,RECUD                ;UPDATE THE RECORD COUNT.
3094    011452  004737  012004          JSR   PC,CHKERR               ;CHECK FOR STATUS ERRORS.
3095    011456          IFB WRTYFG EQ #0 THEN         ;
   (6)  011456  105737  003471                                                    TSTB    WRTYFG
   (9)  011462  001006                                                            BNE     50120$
3096    011464  004737  014664            JSR PC,LOG                  ;LOG BYTES AND ERRORS.
3097    011470          LET R3 := #ENDERF
   (4)  011470  012703  003500                                                    MOV     #ENDERF,
3098    011474  004737  011502            JSR PC,CLRERR                       ;CLEAR ALL ERROR FLAGS
3099    011500          ENDIF
   (4)  011500                                                                 50120$:
3100    011500      ENDIF
   (4)  011500                                                                 50117$:
```

```
  3101  011500  000207                    1$:     RTS     PC                  ;RETURN IF DONE.
```

```
                        ;       SUBROUTINE TO CLEAR FLAGS.
                        ;       INPUTS:         R3 = LWA TO BE CLEARED + 2.
                        ;       OUTPUTS:
                        ;       REGISTERS:      R2
                        ;       CALLS:

011502                  CLRERR:: LET R2 := #BGNFLG
(4)  011502 012702 003466                                       MOV     #BGNFLG,
                                REPEAT
(3)  011506                                                  50121$:
                                  LET (R2)+ := #0
(4)  011506 005022                                              CLR     (R2)+
                                UNTIL R2 EQ R3
(3)  011510 020203                                              CMP     R2,R3
(6)  011512 001375                                              BNE     50121$
                                RTS PC

                        ;       SUBROUTINE TO WAIT UNTIL CURRENT UNIT IS READY OR UNTIL TIME OUT.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:
                        ;       CALLS:

011516                  WSSR::  LET TIME1 := #-1                ;INIT TIMEOUT COUNTER.
(4)  011516 012737 177777 003446                                MOV     #-1,TIME
                                REPEAT                          ;REPEAT UNTIL DEV READY OR TIMEOUT:
(3)  011524                                                  50122$:
                                  BREAK                           ;BREAK TO THE SUPERVISOR.
(3)  01.524 104422                                              TRAP    C$BRK
                                  LET TIME1 := TIME1 - #1       ;UPDATE TIMEOUT COUNTER.
(6)  011526 005337 003446                                       DEC     TIME1
                                UNTIL #TS.SSR SETIN @TSSR(R5) OR TIME1 EQ #0
(4)  011532 032775 000200 002466                                BIT     #TS.SSR,
(6)  011540 001003                                              BNE     50123$
(4)  011542 005737 003446                                       TST     TIME1
(7)  011546 001366                                              BNE     50122$
(4)  011550                                                  50123$:
                                                            ;REPEAT UNTIL DEV READY OR TIMEOUT.
011550 000207           RTS PC                              ;RETURN.
```

```
3133
3134                            ;       SUBROUTINE TO MOVE THE CURRENT MESSAGE PACKET TO THE COMMON AREA AND
3135                            ;       TO UPDATE THE CURRENT TERMINATION CLASS CODE.
3136                            ;       INPUTS:
3137                            ;       OUTPUTS:
3138                            ;       REGISTERS·      R2, R3.
3139                            ;       CALLS:
3140
3141  011552                    MOVMSG:: LET TSSREG := @TSSR(R5)        ;FREEZE THE STATUS REG CONTENTS
 (4)  011552  017537  002466  003464                                                MOV     @TSSR(R5
3142  011560                           LET R2 := TSSREG CLR.BY #TSC.TCC ;EXTRACT THE TERMINATION CLASS CODE.
 (4)  011560  013702  003464                                                        MOV     TSSREG,R
 (6)  011564  042702  177761                                                        BIC     #TSC.TCC
3143  011570                           LET CTCC := R2 SHIFT -1          ;AND SAVE IT
 (4)  011570  010237  003460                                                        MOV     R2,CTCC
 (7)  011574  006237  003460                                                        ASR     CTCC
3144  011600                           LET R3 := MSGPKA(R5)             ;ADR OF THIS DEVICE'S MSG.
 (4)  011600  016503  002506                                                        MOV     MSGPKA(R
3145  011604                           LET R2 := #0                     ;CLR COUNTER.
 (4)  011604  005002                                                                CLR     R2
3146  011606                           WHILE R2 NE #MSGCNT DO           ;WHILE THERE ARE MORE LOCATIONS:
 (4)  011606                                                                50124$:
 (6)  011606  020227  000016                                                        CMP     R2,#MSGC
 (9)  011612  001405                                                                BEQ     50125$
3147  011614                             LET MSGPKT(R2) := (R3)+         ;MOVE MSG TO COMMON AREA.
 (4)  011614  012362  002340                                                        MOV     (R3)+,MS
3148  011620                             LET R2 := R2 + #2               ;UPDATE COUNTER.
 (6)  011620  062702  000002                                                        ADD     #2,R2
3149  011624                           ENDDO
 (4)  011624  000770                                                                BR      50124$
 (3)  011626                                                                50125$:
3150  011626                           LET EOTFLG(R5) := MSGPKT+MS.XS0 ;MOVE XSTATO TO EOT FLAG.
 (4)  011626  013765  002346  003510                                                MOV     MSGPKT+M
3151  011634  000207                  RTS     PC
```

```
                        ;       SUBROUTINE TO ADJUST THE RECORD COUNT.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:
                        ;       CALLS:

011636                  RECUD:: IFB RECLOG EQ #0 THEN           ;IF RECORD HAS NOT BEEN LOGGED:
011636  105737  003473                                                  TSTB    RECLOG
011642  001057                                                          BNE     50126$
011644                    LET RECCNT(R5) := RECCNT(R5) - #1
011644  005365  003330                                                  DEC     RECCNT(R
011650                    IF #BIT0 NOTSETIN CTCC AND #X2.OPM SETIN MSGPKT+MS.XS2 THEN   ;IF TAPE
011650  032737  000001  003460                                          BIT     #BIT0,CT
011656  001046                                                          BNE     50127$
011660  032737  100000  002352                                          BIT     #X2.OPM,
011666  001442                                                          BEQ     50127$
011670                      LET RECLOG :B= RECLOG + #1  ;SET RECORD LOGGED,
011670  105237  003473                                                  INCB    RECLOG
011674                      IF CMDWRD EQ #RWD THEN      ;IF THIS IS A REWIND CMD:
011674  023727  003430  102010                                          CMP     CMDWRD,#
011702  001003                                                          BNE     50130$
011704                        LET RECCNT(R5) := #0      ;CLEAR RECORD COUNT,
011704  005065  003330                                                  CLR     RECCNT(R
011710                      ELSE
011710  000431                                                          BR      50131$
011712                                                          50130$:
011712                        IF #BRF.C SETIN CMDWRD THEN       ;IF BRF USED, UPDATE RECORD COUN
011712  032737  004000  003430                                          BIT     #BRF.C,C
011720  001425                                                          BEQ     50132$
011722                          IF #MOD.CO NOTSETIN CMDWRD THEN  ;IF A FORWARD CMD:
011722  032737  000400  003430                                          BIT     #MOD.CO,
011730  001007                                                          BNE     50133$
011732                        IF #MOD.CO NOTSETIN PCMDWD THEN  ;IF PREV CMD WAS A FWD ALSO:
011732  032737  000400  003434                                          BIT     #MOD.CO,
011740  001002                                                          BNE     50134$
011742                          LET RECCNT(R5) := RECCNT(R5) + #1  ;INCREMENT RECORD COUNT.
011742  005265  003330                                                  INC     RECCNT(R
011746                      ENDIF
011746                                                          50134$:
011746                    ELSE                      ;IF REVERSE CMD:
011746  000412                                                          BR      50135$
011750                                                          50133$:
011750                      IF #MOD.CO SETIN PCMDWD THEN  ;IF PREVIOUS CMD WAS A REV ALSO:
011750  032737  000400  003434                                          BIT     #MOD.CO,
011756  001406                                                          BEQ     50136$
011760                        IF #X0.BOT NOTSETIN EOTFLG(R5) THEN ;WHEN NOT AT BOT THEN
011760  032765  000002  003510                                          BIT     #X0.BOT,
011766  001002                                                          BNE     50137$
011770                          LET RECCNT(R5) := RECCNT(R5) - #1  ;DECREMENT RECORD COUNT
011770  005365  003330                                                  DEC     RECCNT(R
011774                        ENDIF
011774                                                          50137$:
011774                      ENDIF
011774                                                          50136$:
011774                    ENDIF
011774                                                          50135$:
```

```
 3178  011774                                            ENDIF
  (4)  011774                                                                            50132$:
 3179  011774                                          ENDIF
  (4)  011774                                                                            50131$:
 3180  011774                                        ENDIF
  (4)  011774                                                                            50127$:
 3181  011774                                        LET @DATAWT := RECCNT(R5)
  (4)  011774  016577  003330  171414                                                           MOV     RECCNT(R
 3182  012002                                      ENDIF
  (4)  012002                                                                            50126$:
 3183  012002  000207                              RTS     PC                  ;RETURN.
```

```
3185                                    ;       THIS IS THE ERROR CHECK SUBROUTINE.  AFTER INTERRUPT THIS
3186                                    ;       SUBROUTINE IS CALLED TO CHECK THE TU80 STATUS.
3187                                    ;       IF SPECIAL COND IS SET THEN THE TCC HANDLING SUBROUTINE IS ENTERED.
3188                                    ;       IF THE RFC IS NON ZERO FOR A COMMAND REQUIRING A BPCR,
3189                                    ;       THEN AN ERROR RFC IS REPORTED.
3190                                    ;       INPUTS:
3191                                    ;       OUTPUTS:
3192                                    ;       REGISTERS:      R2, R4.
3193                                    ;       CALLS:          TCC0-TCC7.
3194
3195  012004                            CHKERR::IF DEVTBL(R5) EQ #NINUSE THEN
 (6)  012004  026527  002536  177774                                          CMP     DEVTBL(R
 (9)  012012  001003                                                          BNE     50140$
3196  012014  000137  012230                    JMP     1$
3197  012020                            ELSE
 (4)  012020  000400                                                          BR      50141$
 (3)  012022                                                        50140$:
3198  012022                            ENDIF
 (4)  012022                                                        50141$:
3199  012022                            IF #TS.SC SETIN TSSREG THEN     ;IF SPECIAL COND STATUS IS SET THEN:
 (6)  012022  032737  100000  003464                                          BIT     #TS.SC,T
 (9)  012030  001441                                                          BEQ     50142$
3200  012032                              IF CTCC NE #2 THEN            ;IF TCC IS NOT 2 THEN:
 (6)  012032  023727  003460  000002                                          CMP     CTCC,#2
 (9)  012040  001405                                                          BEQ     50143$
3201  012042                                IFB ERRREC EQ #0 THEN       ;IF NOT IN ERROR RECOVERY:
 (6)  012042  105737  003477                                                  TSTB    ERRREC
 (9)  012046  001002                                                          BNE     50144$
3202  012050  005265  003270                    INC     SCNT(R5)        ;INC SC COUNTER.
3203  012054                                  ENDIF
 (4)  012054                                                        50144$:
3204  012054                            ENDIF
 (4)  012054                                                        50143$:
3205  012054                            IF #TS.NXM SETIN TSSREG OR #TS.UPE SETIN TSSREG THEN  ;WHEN NON-EXISTA
 (6)  012054  032737  004000  003464                                          BIT     #TS.NXM,
 (8)  012062  001004                                                          BNE     50145$
 (6)  012064  032737  040000  003464                                          BIT     #TS.UPE,
 (9)  012072  001412                                                          BEQ     50146$
 (6)  012074                                                        50145$:
3206  012074                              IF #X2.OPM NOTSETIN MSGPKT+MS.XS2 THEN        ;AND TAPE NOT MO
 (6)  012074  032737  100000  002352                                          BIT     #X2.OPM,
 (9)  012102  001003                                                          BNE     50147$
3207  012104                                LET R2 := #5                  ;SET TCC5 INDEX
 (4)  012104  012702  000005                                                  MOV     #5,R2
3208  012110                              ELSE
 (4)  012110  000402                                                          BR      50150$
 (3)  012112                                                        50147$:
3209  012112                                LET R2 := #4                  ;TAPE MOVED, SET TCC4 INDEX
 (4)  012112  012702  000004                                                  MOV     #4,R2
3210  012116                              ENDIF
 (4)  012116                                                        50150$:
3211  012116                            ELSE
 (4)  012116  000402                                                          BR      50151$
 (3)  012120                                                        50146$:
3212  012120                              LET R2 := CTCC                ;SET DETECTED TCC INDEX
 (4)  012120  013702  003460                                                  MOV     CTCC,R2
```

```
3213  012124                                          ENDIF
 (4)  012124                                                                       50151$:
3214  012124                                          LET R2 := R2 SHIFT 1  ;CURRENT TCC X 2.
 (7)  012124  006302                                                                       ASL     R2
3215  012126  004772  012232                          JSR    PC,@TCCRA(R2)          ;GO TO THE TCC HANDLING SUBROUTINE.
3216  012132                                        ELSE
 (4)  012132  000430                                                                       BR      50152$
 (3)  012134                                                                       50142$:
3217  012134                                          IF #BRF.C SETIN CMDWRD THEN   ;IF BRF IS USED IN THIS CMD THEN:
 (6)  012134  032737  004000  003430                                                       BIT     #BRF.C,C
 (9)  012142  001424                                                                       BEQ     50153$
3218  012144                                            IF MSGPKT+MS.RFC NE #0 THEN   ;IF THERE IS AN RFC THEN:
 (6)  012144  005737  002344                                                               TST     MSGPKT+M
 (9)  012150  001421                                                                       BEQ     50154$
3219  012152                                              IFB RANDOM EQ #0 ORB VFYFLG NE #0 THEN
 (6)  012152  105737  003523                                                               TSTB    RANDOM
 (8)  012156  001403                                                                       BEQ     50155$
 (6)  012160  105737  003524                                                               TSTB    VFYFLG
 (9)  012164  001413                                                                       BEQ     50156$
 (6)  012166                                                                       50155$:
3220                                                                      ;IF NOT IN RANDOM OR IF CMD IS W
3221  012166                                                IFB IRE EQ #0 THEN          ;IF RFC ERROR REPORTS ARE ALLOWE
 (6)  012166  105737  003527                                                               TSTB    IRE
 (9)  012172  001010                                                                       BNE     50157$
3222  012174                                                LET HRDCNT(R5) := HRDCNT(R5) + #1  ;UPDATE HARD ERROR COUNT
 (6)  012174  005265  003310                                                               INC     HRDCNT(R
3223  012200                                                ERRHRD 13,RFCERM,STAERM ;REPORT RFC ERROR
 (4)  012200  104456                                                                       TRAP    C$ERHRD
 (5)  012202  000015                                                                       .WORD   13
 (5)  012204  004340                                                                       .WORD   RFCERM
 (5)  012206  005642                                                                       .WORD   STAERM
3224  012210  004737  013360                            JSR     PC,RAMDUM              ;GO DO RAM DUMP
3225  012214                                                ENDIF
 (4)  012214                                                                       50157$:
3226  012214                                              ENDIF
 (4)  012214                                                                       50156$:
3227  012214                                            ENDIF
 (4)  012214                                                                       50154$:
3228  012214                                          ENDIF
 (4)  012214                                                                       50153$:
3229  012214                                        ENDIF
 (4)  012214                                                                       50152$:
3230  012214                                        IFB RWERR NE #0 THEN          ;IF A READ/WRITE ERROR HAS OCCURRED THEN
 (6)  012214  105737  003475                                                               TSTB    RWERR
 (9)  012220  001403                                                                       BEQ     50160$
3231  012222                                          LET CMDPKT := CMDSAV          ;RESTORE CMD PACKET AFTER ERROR RECOV.
 (4)  012222  013737  003432  002314                                                       MOV     CMDSAV,C
3232  012230                                        ENDIF
 (4)  012230                                                                       50160$:
3233  012230  000207                        1$:     RTS     PC                     ;RETURN.
3234
```

```
3236                                    ;       ADDRESSES OF TCC HANDLING ROUTINES FOR TERMINATION CLASS CODES 0 - 7.
3237
3238  012232  012252                    TCCRA:  TCC0
3239  012234  012274                            TCC1
3240  012236  012312                            TCC2
3241  012240  012460                            TCC3
3242  012242  012502                            TCC4
3243  012244  013200                            TCC5
3244  012246  013302                            TCC6
3245  012250  013336                            TCC7
```

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

                                TCC0::  LET HRDCNT(R5) := HRDCNT(R5) + #1  ;UPDATE HARD ERROR COUNT.
 (6)  012252  005265  003310                                                        INC     HRDCNT(R
                                        ERRHRD 5,SCERM,STAERM           ;REPORT SPECIAL CONDITION ERROR.
 (4)  012256  104456                                                                TRAP    C$ERHRD
 (5)  012260  000005                                                                .WORD   5
 (5)  012262  004314                                                                .WORD   SCERM
 (5)  012264  005642                                                                .WORD   STAERM
 012266  004737  013360                JSR     PC,RAMDUM               ;GO DO RAM DUMP
 012272  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

                                TCC1::  ERRDF 6,ATTNM,STAERM            ;REPORT ATTENTION-UNIT OFF LINE.
 (4)  012274  104455                                                                TRAP    C$ERDF
 (5)  012276  000006                                                                .WORD   6
 (5)  012300  004421                                                                .WORD   ATTNM
 (5)  012302  005642                                                                .WORD   STAERM
 012304  004737  013360                JSR     PC,RAMDUM               ;GO DO RAM DUMP
 012310  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:

012312                                  TCC2::    IF #X0.BOT SETIN MSGPKT+MS.XS0 ANDB EXPBOT NE #0 THEN
(6) 012312  032737  000002  002346                                                          BIT     #X0.BOT,
(9) 012320  001404                                                                          BEQ     50161$
(6) 012322  105737  003522                                                                  TSTB    EXPBOT
(9) 012326  001401                                                                          BEQ     50161$
                                                                          ;IF AT BOT AND BOT IS EXPECTED:
012330  000452                              BR TC2RTN                       ;RETURN-TCC2 CAUSED BY EXPECTED BOT.
012332                                    ENDIF
(4) 012332                                                                          50161$:
012332                                  IF L$TEST EQ #3 THEN
(6) 012332  023727  002114  000003                                                          CMP     L$TEST,#
(9) 012340  001011                                                                          BNE     50162$
012342                                   IF PCMDWD EQ #WTM AND CMDWRD EQ #SRR THEN
(6) 012342  023727  003434  100011                                                          CMP     PCMDWD,#
(9) 012350  001005                                                                          BNE     50163$
(6) 012352  023727  003430  104410                                                          CMP     CMDWRD,#
(9) 012360  001001                                                                          BNE     50163$
012362  000435                            BR    TC2RTN
012364                                    ENDIF
(4) 012364                                                                          50163$:
012364                                  ENDIF
(4) 012364                                                                          50162$:
012364                                    IF #X0.RLS!X0.RLL!X0.TMK!X0.LET!X0.BOT SETIN MSGPKT+MS.XS0 THEN
(6) 012364  032737  170002  002346                                                          BIT     #X0.RLS!
(9) 012372  001431                                                                          BEQ     50164$
                                                                          ;IF TCC2 CAUSED BY ANYTHING BUT EOT:
012374                                      IFB RANDOM EQ #0 ORB VFYFLG NE #0 THEN
(6) 012374  105737  003523                                                                  TSTB    RANDOM
(8) 012400  001403                                                                          BEQ     50165$
(6) 012402  105737  003524                                                                  TSTB    VFYFLG
(9) 012406  001423                                                                          BEQ     50166$
(6) 012410                                                                          50165$:
                                                                        ;IF NOT IN RANDOM OR IF CMD IS WTV:
012410                                        IFB IRE EQ #0 THEN              ;IF RFC ERROR REPORTS ARE ALLOWED:
(6) 012410  105737  003527                                                                  TSTB    IRE
(9) 012414  001020                                                                          BNE     50167$
012416                                        IFB ERRREC NE #0 THFN           ;IF WE ARE IN ERROR RECOVERY THE
(6) 012416  105737  003477                                                                  TSTB    ERRREC
(9) 012422  001403                                                                          BEQ     50170$
012424                                          LET UNREC :B= UNREC + #1        ;SET UNRECOVERABLE FLAG FOR LO
(6) 012424  105237  003476                                                                  INCB    UNREC
012430                                        ELSE                            ;ELSE - IF NOT IN ERROR RECOVERY
(4) 012430  000402                                                                          BR      50171$
(3) 012432                                                                          50170$:
012432                                          LET SCCNT(R5) := SCCNT(R5) + #1 ;INCREMENT THE SPEC COND COUNT
(6) 012432  005265  003270                                                                  INC     SCCNT(R5
012436                                        ENDIF
(4) 012436                                                                          50171$:
```

```
                                                        LET HRDCNT(R5) := HRDCNT(R5) + #1   ;UPDATE HARD ERROR COUNT.
                                                                                                INC     HRDCNT(R
                                                        ERRHRD 7,TSAM,STAERM                ;REPORT TAPE STATUS ALERT.
                                                                                                TRAP    C$ERHRD
                                                                                                .WORD   7
                                                                                                .WORD   TSAM
                                                                                                .WORD   STAERM
                                                            JSR     PC,RAMDUM               ;GO DO RAM DUMP
                                                        ENDIF
                                                                                            50167$:
                                                    ENDIF
                                                                                            50166$:
                                                ENDIF
                                                                                            50164$:
                                TC2RTN:     RTS PC                                      ;RETURN.


                                ;       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

                                TCC3::  ERRDF 8,FUNRM,STAERM                            ;REPORT FUNCTION REJECT.
                                                                                                TRAP    C$ERDF
                                                                                                .WORD   8
                                                                                                .WORD   FUNRM
                                                                                                .WORD   STAERM
                                        JSR     PC,RAMDUM                               ;GO DO RAM DUMP
                                        JSR     PC,DROPU                                ;DROP THE UNIT.
                                        RTS PC                                          ;RETURN.
```

```
3327                                    ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 4, RECOVERABLE ERROR.
3328                                    ;       TAPE POSITION IS ONE RECORD BEYOND WHAT ITS POSITION WAS WHEN
3329                                    ;       THE FUNCTION WAS INITIATED.  RECOVERY PROCEDURE IS TO LOG THE
3330                                    ;       ERROR AND ISSUE THE APPROPRIATE RETRY COMMAND.
3331                                    ;       2 WRITE-ERROR-RECOVERY ALGORITHMS CAN BE SELECTED:
3332                                    ;       THE FIRST ONE, VIA BADTSW SWITCH,  DOES DETECT BAD SPOTS ON TAPE.
3333                                    ;       IT CALLS A WRITE RETRY SUBR UNTIL THE RECORD IS RECOVERED
3334                                    ;       OR 20 BAD SPOTS HAVE BEEN LOGGED. ON REACHING 20 BAD
3335                                    ;       SPOTS LOGGED, A BAD TAPE OVERFLOW MSG IS PRINTED AND THE
3336                                    ;       UNIT DROPPED.
3337                                    ;       THE SECOND ALGORITHM ISSUES THE TU80 WRITE RETRY COMMAND
3338                                    ;       UP TO 16 TIMES BEFORE DROPPING THE UNIT OR PROCEEDING
3339                                    ;       WITH THE NEXT RECORD ON RECOVERY.
3340                                    ;       INPUTS:
3341                                    ;       OUTPUTS:
3342                                    ;       REGISTERS:      R2,R4.
3343                                    ;       CALLS:          RTLE, EXCUTE, GOWAIT, DROPU, WRTY
3344
3345  012502                            TCC4::  IF DEVTBL(R5) EQ #NINUSE THEN
 (6)  012502  026527  002536  177774                                                  CMP     DEVTBL(R
 (9)  012510  001003                                                                  BNE     50172$
3346  012512  000137  013176                    JMP     3$
3347  012516                            ELSE
 (4)  012516  000400                                                                  BR      50173$
 (3)  012520                                                                50172$:
3348  012520                            ENDIF
 (4)  012520                                                                50173$:
3349  012520                            IF CMDLG EQ #2 ANDB BADTSW NE #0 THEN
 (6)  012520  023727  003436  000002                                                  CMP     CMDLG,#2
 (9)  012526  001137                                                                  BNE     50174$
 (6)  012530  105737  002211                                                          TSTB    BADTSW
 (9)  012534  001534                                                                  BEQ     50174$
3350  012536                              IFB ERRREC EQ #0 ANDB ERCVER NE #0 THEN
 (6)  012536  105737  003477                                                          TSTB    ERRREC
 (9)  012542  001011                                                                  BNE     50175$
 (6)  012544  105737  002207                                                          TSTB    ERCVER
 (9)  012550  001406                                                                  BEQ     50175$
3351  012552                              ERRSOFT 9,RERM,STAERM ;
 (4)  012552  104457                                                                  TRAP    C$ERSOFT
 (5)  012554  000011                                                                  .WORD   9
 (5)  012556  004634                                                                  .WORD   RERM
 (5)  012560  005642                                                                  .WORD   STAERM
3352  012562  004737  013360              JSR   PC,RAMDUM               ;GO DO RAM DUMP
3353  012566                            ENDIF
 (4)  012566                                                                50175$:
3354  012566                            IFB IREC EQ #0 THEN             ;
 (6)  012566  105737  002210                                                          TSTB    IREC
 (9)  012572  001112                                                                  BNE     50176$
3355  012574                              LET ERRREC :B= ERRREC + #1   ;RETRY FLAG FOR EXCUTE SUBR: DON'T UPDAT
 (6)  012574  105237  003477                                                          INCB    ERRREC
3356  012600                              LET WRTYER :B= WRTYER + #1   ;REWRITE ERROR FLAG FOR WRTY SUBR
 (6)  012600  105237  003472                                                          INCB    WRTYER
3357  012604                              IFB WRTYFG EQ #0 THEN        ;FIRST RETRY ON THIS RECORD: SUBSEQUENT
 (6)  012604  105737  003471                                                          TSTB    WRTYFG
 (9)  012610  001102                                                                  BNE     50177$
3358                                                                    ;RETRIES WITH TCC4 ERRORS BY-PASS THIS S
```

```
3359  012612
 (4)  012612  013737  003430  014164
3360  012620
 (4)  012620  013737  002314  014162
3361  012626
 (4)  012626  013737  002322  014166
3362  012634
 (6)  012634  105237  003475
3363  012640
 (6)  012640  105237  003471
3364  012644
 (3)  012644
3365  012644
 (6)  012644  005265  003250
3366  012650
 (4)  012650  005037  003466
3367  012654
 (4)  012654  105037  003470
3368  012660  004737  013650
3369  012664
 (6)  012664  026527  002536  177774
 (9)  012672  001002
3370  012674  000540
3371  012676
 (4)  012676  000400
 (3)  012700
3372  012700
 (4)  012700
3373  012700
 (4)  012700  105737  003472
 (6)  012704  001404
 (4)  012706  027727  170606  000050
 (7)  012714  103753
 (4)  012716
3374
3375  012716
 (6)  012716  027727  170576  000050
 (9)  012724  103425
3376  012726
 (7)  012726  012746  014255
 (6)  012732  012746  000001
 (3)  012736  010600
 (4)  012740  104414
 (4)  012742  062706  000004
3377  012746  004737  014374
3378  012752
 (6)  012752  005365  003330
3379  012756  004737  013360
3380  012762  004737  016716
3381  012766
 (4)  012766  005065  003330
3382  012772
 (4)  012772  012775  002334  002456
3383  013000
 (4)  013000
3384  013000
```

```
LET WTYWRD := CMDWRD      ;SAVE WRITE COMMAND PACKET
                                        MOV     CMDWRD,W
LET WTYCMD := CMDPKT      ;
                                        MOV     CMDPKT,W
LET WTYBRF := CMDPKT+CP.CNT       ;
                                        MOV     CMDPKT+C
LET RWERR :B= RWERR + #1  ;LOG SUBR FLAG: COUNT WRT ERRORS
                                        INCB    RWERR
LET WRTYFG :B= WRTYFG + #1        ;RETRY IN PROGRESS FLAG
                                        INCB    WRTYFG
REPEAT
                                    50200$:
  LET WRTYCT(R5) := WRTYCT(R5) + #1       ;COUNT GLOBAL WRITE RETR
                                        INC     WRTYCT(R
  LET RETRYC := #0                ;CLEAR # OF RETRIES PER RECORD
                                        CLR     RETRYC
  LET RPTCNT :B= #0               ;CLEAR # OF REPEATS
                                        CLRB    RPTCNT
  JSR PC,WRTY             ;CALL WRITE RETRY
        IF DEVTBL(R5) EQ #NINUSE THEN
                                        CMP     DEVTBL(R
                                        BNE     50201$
  BR 3$
        ELSE
                                        BR      50202$
                                    50201$:
        ENDIF
                                    50202$:
UNTILB WRTYER EQ #0 OR @BTPT HIS #40.     ;REPEAT RETRIES ON SAME
                                        TSTB    WRTYER
                                        BEQ     50203$
                                        CMP     @BTPT,#4
                                        BLO     50200$
                                    50203$:
                          ;UNTIL RECOVERED OR 20 BAD SPOTS
IF @BTPT HIS #40. THEN    ;WHEN 20 BAD SPOTS LOGGED
                                        CMP     @BTPT,#4
                                        BLO     50204$
  PRINTB #BTMSG2          ;PRINT BAD TAPE OVERFLOW MSG
                                        MOV     #BTMSG2,
                                        MOV     #1,-(SP)
                                        MOV     SP,R0
                                        TRAP    C$PNTB
                                        ADD     #4,SP
  JSR PC,BORERS           ;ERASE BAD RECORD
  LET RECCNT(R5) := RECCNT(R5) - #1       ;
                                        DEC     RECCNT(R
  JSR     PC,RAMDUM       ;GO DO RAM DUMP
  JSR     PC,DROPU        ;DROP THE UNIT.
  LET RECCNT(R5) :=  #0   ;
                                        CLR     RECCNT(R
  LET @TSDB(R5) := #RWCPK ;REWIND UNIT
                                        MOV     #RWCPK,@
ENDIF
                                    50204$:
LET WRTYFG :B= #0         ;RETRY COMPLETE FLAG
```

```
  (4)  013000  105037  003471                                                                   CLRB    WRTYFG
 3385  013004                              LET MISCFG :B= MISCFG + #1        ;DO NOT HALT ON THIS CMD FLG
  (6)  013004  105237  003541                                                                   INCB    MISCFG
 3386  013010                              LET PCMDWD := WTYWRD      ;RESTORE ORIGINAL WRT CMD AFTER RECOVERY
  (4)  013010  013737  014164  003434                                                           MOV     WTYWRD,P
 3387  013016                            ENDIF
  (4)  013016                                                                          50177$:
 3388  013016                          ELSE
  (4)  013016  000402                                                                           BR      50205$
  (3)  013020                                                                          50176$:
 3389  013020                            LET UNREC :B= UNREC + #1     ;
  (6)  013020  105237  003476                                                                   INCB    UNREC
 3390  013024                          ENDIF
  (4)  013024                                                                          50205$:
 3391  013024                        ELSE
  (4)  013024  000464                                                                           BR      50206$
  (3)  013026                                                                          50174$:
 3392  013026  004737  013512            JSR PC,RTLE                     ;CHECK FOR RETRY LIMIT EXCEEDED.
 3393  013032                            IF CMDLG GT #2 THEN             ;IF READ CMD THEN:
  (6)  013032  023727  003436  000002                                                           CMP     CMDLG,#2
  (9)  013040  003411                                                                           BLE     50207$
 3394  013042                              LET R2 := #RRECL SHIFT -1   ;R2=READ RETRY COUNT LIMIT / 2
  (4)  013042  012702  000020                                                                   MOV     #RRECL,R
  (7)  013046  006202                                                                           ASR     R2
 3395  013050                              IF RETRYC GE R2 THEN               ;IF RETRY COUNT IS MORE THAN HAL
  (6)  013050  023702  003466                                                                   CMP     RETRYC,R
  (9)  013054  002403                                                                           BLT     50210$
 3396  013056                                LET CMDPKT := CMDPKT SET.BY #OPP.C        ;SET OPPOSITE BIT FOR RE
  (6)  013056  052737  020000  002314                                                           BIS     #OPP.C,C
 3397  013064                              ENDIF
  (4)  013064                                                                          50210$:
 3398  013064                            ENDIF
  (4)  013064                                                                          50207$:
 3399  013064                            IF RETRYC EQ #0 ANDB ERCVER NE #0 THEN  ;IF THIS IS THE ORIGINAL ERROR
  (6)  013064  005737  003466                                                                   TST     RETRYC
  (9)  013070  001011                                                                           BNE     50211$
  (6)  013072  105737  002207                                                                   TSTB    ERCVER
  (9)  013076  001406                                                                           BEQ     50211$
 3400  013100                            ERRSOFT 9,RERM,STAERM          ;REPORT RECOVERABLE ERROR
  (4)  013100  104457                                                                           TRAP    C$ERSOFT
  (5)  013102  000011                                                                           .WORD   9
  (5)  013104  004634                                                                           .WORD   RERM
  (5)  013106  005642                                                                           .WORD   STAERM
 3401  013110  004737  013360            JSR    PC,RAMDUM               ;GO DO RAM DUMP
 3402  013114                            ENDIF                           ;PROVIDED OPERATOR HAS ENABLED THE REPOR
  (4)  013114                                                                          50211$:
 3403  013114                            LET RETRYC := RETRYC + #1      ;UPDATE RETRY COUNT.
  (6)  013114  005237  003466                                                                   INC     RETRYC
 3404  013120                            LET CMDPKT := CMDPKT SET.BY #MOD.C1   ;SET RETRY BIT IN CMD PACKET.
  (6)  013120  052737  001000  002314                                                           BIS     #MOD.C1,
 3405  013126                            IFB IREC EQ #0 THEN            ;IF ERROR RECOVERY ENABLED:
  (6)  013126  105737  002210                                                                   TSTB    IREC
  (9)  013132  001017                                                                           BNE     50212$
 3406  013134                                          IF DEVTBL(R5) EQ #NINUSE THEN
  (6)  013134  026527  002536  177774                                                           CMP     DEVTBL(R
  (9)  013142  001002                                                                           BNE     50213$
```

```
3407  013144  000414                                    BR 3$
3408  013146                                                  ELSE
 (4)  013146  000400                                                    BR      50214$
 (3)  013150                                                          50213$:
3409  013150                                                  ENDIF
 (4)  013150                                                          50214$:
3410  013150                                      LET ERRREC :B= ERRREC + #1  ;SET ERROR RECOVERY FLAG.
 (6)  013150  105237  003477                                                  INCB    ERRREC
3411  013154                                      POP R2,R2                  ;POP 2 RTN ADRS FROM STACK.
 (2)  013154  012602                                                          MOV     (SP)+,R2
 (3)  013156  012602                                                          MOV     (SP)+,R2
3412  013160  004737  010576                     JSR PC,EXCUTE                    ;GO EXECUTE THE RETRY COMMAND.
3413  013164  000137  011166                     JMP GOWAIT                 ;GO WAIT FOR INTERRUPT + CHECK STATUS.
3414  013170                                    ELSE                       ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
 (4)  013170  000402                                                          BR      50215$
 (3)  013172                                                          50212$:
3415  013172                                      LET UNREC :B= UNREC + #1    ;SET UNRECOVERABLE ERROR FLAG.
 (6)  013172  105237  003476                                                  INCB    UNREC
3416  013176                                    ENDIF
 (4)  013176                                                          50215$:
3417  013176                                  ENDIF
 (4)  013176                                                          50206$:
3418  013176  000207                    3$:    RTS PC                     ;RETURN
```

```
3420                                    ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 5, RECOVERABLE ERROR.
3421                                    ;       TAPE POSITION HAS NOT CHANGED.  RECOVERY PROCEDURE IS TO LOG THE
3422                                    ;       ERROR AND RE-ISSUE THE ORIGINAL COMMAND.
3423                                    ;       INPUTS:
3424                                    ;       OUTPUTS:
3425                                    ;       REGISTERS:      R2,R4.
3426                                    ;       CALLS:          RTLE, EXCUTE, GOWAIT, DROPU.
3427
3428  013200  004737  013512              TCC5::  JSR PC,RTLE                     ;CHECK FOR RETRY LIMIT EXCEEDED
3429  013204                                      IF RETRYC EQ #0 THEN            ;IF THIS IS THE ORIGINAL ERROR THEN:
 (6)  013204  005737  003466                                                                TST     RETRYC
 (9)  013210  001006                                                                        BNE     50216$
3430  013212                                        ERRSOFT 10,RERM,STAERM        ;REPORT RECOVERABLE ERROR.
 (4)  013212  104457                                                                        TRAP    C$ERSOFT
 (5)  013214  000012                                                                        .WORD   10
 (5)  013216  004634                                                                        .WORD   RERM
 (5)  013220  005642                                                                        .WORD   STAERM
3431  013222  004737  013360                      JSR     PC,RAMDUM               ;GO DO RAM DUMP
3432  013226                                      ENDIF
 (4)  013226                                                                                50216$:
3433  013226                                      LET RETRYC := RETRYC + #1        ;UPDATE RETRY COUNTER.
 (6)  013226  005237  003466                                                                INC     RETRYC
3434  013232                                      IFB IREC EQ #0 THEN              ;IF ERROR RECOVERY IS ENABLED:
 (6)  013232  105737  002210                                                                TSTB    IREC
 (9)  013236  001016                                                                        BNE     50217$
3435  013240                                        LET ERRREC :B= ERRREC + #1     ;SET ERROR RECOVERY FLAG.
 (6)  013240  105237  003477                                                                INCB    ERRREC
3436  013244                                        LET RECCNT(R5) := RECCNT(R5) + #1      ;UPDATE REC COUNT
 (6)  013244  005265  003330                                                                INC     RECCNT(R
3437  013250                                        LET @DATAWT := RECCNT(R5)              ;AND INSERT IT INTO WRT BFR
 (4)  013250  016577  003330  170140                                                        MOV     RECCNT(R
3438  013256                                        POP R2,R2                      ;POP 2 RTN ADRS FROM STACK.
 (2)  013256  012602                                                                        MOV     (SP)+,R2
 (3)  013260  012602                                                                        MOV     (SP)+,R2
3439  013262  004737  010576                       JSR PC,EXCUTE                 ;GO RE-ISSUE THE COMMAND.
3440  013266  000137  011166                       JMP GOWAIT                    ;GO WAIT FOR INTERRUPT + CHECK STATUS.
3441  013272                                      ELSE                             ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
 (4)  013272  000402                                                                        BR      50220$
 (3)  013274                                                                                50217$:
3442  013274                                        LET UNREC :B= UNREC + #1       ;SET UNRECOVERABLE ERROR FLAG.
 (6)  013274  105237  003476                                                                INCB    UNREC
3443  013300                                      ENDIF
 (4)  013300                                                                                50220$:
3444  013300  000207                       RTS PC                                ;RETURN.
3445
3446
```

```
                                        ;       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

013302                                  TCC6::  LET @TSDB(R5) := #RWCPK         ;ISSUE A REWIND COMMAND,
 (4) 013302  012775  002334  002456                                                     MOV     #RWCPK,@
013310  004737  011516                          JSR PC,WSSR                     ;WAIT FOR SUBSYSTEM READY,
013314                                          ERRDF 11,URERM,STAERM           ;REPORT UNRECOVERABLE ERROR.
 (4) 013314  104455                                                                     TRAP    C$ERDF
 (5) 013316  000013                                                                     .WORD   11
 (5) 013320  004656                                                                     .WORD   URERM
 (5) 013322  005642                                                                     .WORD   STAERM
013324  004737  013360                          JSR     PC,RAMDUM               ;GO DO RAM DUMP
013330  004737  016716                          JSR     PC,DROPU                ;DROP THE UNIT.
013334  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 TYPF OF FATAL ERROR.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:      R2, R4
                        ;       CALLS:

013336                  TCC7::  ERRDF 12,FATSM,STAERM           ;REPORT FATAL SUBSYSTEM ERROR.
013336  104455                                                          TRAP    C$ERDF
013340  000014                                                          .WORD   12
013342  004457                                                          .WORD   FATSM
013344  005642                                                          .WORD   STAERM
013346  004737  013360                JSR     PC,RAMDUM               ;GO DO RAM DUMP
013352  004737  016716                JSR     PC,DROPU                ;DROP THE UNIT.
013356  000207                        RTS PC                          ;RETURN.


013360                  RAMDUM::IFB RAMWRT NE #0 THEN
013360  105737  002214                                                  TSTB    RAMWRT
013364  001452                                                          BEQ     50221$
013366                          PRINTX  #RAMFHR
013366  012746  005310                                                  MOV     #RAMFHR,
013372  012746  000001                                                  MOV     #1,-(SP)
013376  010600                                                          MOV     SP,R0
013400  104415                                                          TRAP    C$PNTX
013402  062706  000004                                                  ADD     #4,SP
013406  012737  000010  003406        MOV     #8.,RAMSIZ      ;RAM FIELD IS 8 BYTES LONG
013414  012737  000020  003342        MOV     #20,RAMHLD      ;FIELD STARTS AT 20 OCTAL (10 HEX)
013422  004737  015164                JSR     PC,RAMER        ;READ AND PRINT THEM
013426  012737  000040  003342        MOV     #40,RAMHLD      ;FIELD STARTS AT 40 OCTAL (20 HEX)
013434  004737  015164                JSR     PC,RAMER        ;READ AND PRINT THEM
013440  012737  000060  003342        MOV     #60,RAMHLD      ;FIELD STARTS AT 60 OCTAL (30 HEX)
013446  004737  015164                JSR     PC,RAMER        ;READ AND PRINT THEM
013452  012737  000020  003406        MOV     #16.,RAMSIZ     ;RAM FIELD IS SIXTEEN BYTES LONG
013460  012737  000100  003342        MOV     #100,RAMHLD     ;FIELD STARTS AT 100 OCTAL (40 HEX)
013466  004737  015164                JSR     PC,RAMER        ;READ AND PRINT THEM
013472                          PRINTX  #RAMLIN
013472  012746  005464                                                  MOV     #RAMLIN,
013476  012746  000001                                                  MOV     #1,-(SP)
013502  010600                                                          MOV     SP,R0
013504  104415                                                          TRAP    C$PNTX
013506  062706  000004                                                  ADD     #4,SP
013512                  ENDIF
013512                                                          50221$:


                        ;       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.
```

```
3505                                    ;       CALLS:          DROPU
3506
3507  013512                            RTLE::  IF CMDLG EQ #0 THEN             ;IF CMD IS NOT A READ OR WRITE THEN:
 (6)  013512  005737  003436                                                            TST     CMDLG
 (9)  013516  001012                                                                    BNE     50222$
3508  013520                                    ERRDF 11,URERM,STAERM             ;REPORT UNRECOVERABLE ERROR.
 (4)  013520  104455                                                                    TRAP    C$ERDF
 (5)  013522  000013                                                                    .WORD   11
 (5)  013524  004656                                                                    .WORD   URERM
 (5)  013526  005642                                                                    .WORD   STAERM
3509  013530  004737  013360                    JSR     PC,RAMDUM               ;GO DO RAM DUMP
3510  013534  004737  016716                    JSR     PC,DROPU                ;DROP THE UNIT.
3511  013540                                    POP R2
 (2)  013540  012602                                                                    MOV     (SP)+,R2
3512  013542  000441                           BR RTLRTN                       ;AND RETURN.
3513  013544                                    ENDIF
 (4)  013544                                                                    50222$:
3514  013544                                    LET RWERR :B= RWERR + #1        ;SET READ/WRITE ERROR FLAG.
 (6)  013544  105237  003475                                                            INCB    RWERR
3515  013550                                    IF CMDLG EQ #2 THEN             ;IF CMD IS A WRT OR WTM:
 (6)  013550  023727  003436  000002                                                    CMP     CMDLG,#2
 (9)  013556  001020                                                                    BNE     50223$
3516  013560                                      IF RETRYC EQ #WRECL THEN      ;IF RETRY COUNT HAS REACHED LIMIT:
 (6)  013560  023727  003466  000020                                                    CMP     RETRYC,#
 (9)  013566  001013                                                                    BNE     50224$
3517  013570                                       LET UNREC :B= UNREC + #1     ;SET UNRECOVERABLE FLAG
 (6)  013570  105237  003476                                                            INCB    UNREC
3518  013574                                       ERRDF 14,RLEXM,STAERM        ;REPORT RETRY LIMIT EXCEEDED.
 (4)  013574  104455                                                                    TRAP    C$ERDF
 (5)  013576  000016                                                                    .WORD   14
 (5)  013600  004374                                                                    .WORD   RLEXM
 (5)  013602  005642                                                                    .WORD   STAERM
3519  013604  004737  013360                       JSR  PC,RAMDUM               ;GO DO RAM DUMP
3520  013610  004737  016716                       JSR  PC,DROPU                ;DROP THE UNIT.
3521  013614                                       POP  R2
 (2)  013614  012602                                                                    MOV     (SP)+,R2
3522  013616                                     ENDIF
 (4)  013616                                                                    50224$:
3523  013616                                   ELSE                            ;ELSE - CMD IS A READ:
 (4)  013616  000403                                                                    BR      50225$
 (3)  013620                                                                    50223$:
3524  013620                                     IF RETRYC EQ #RRECL THEN      ;IF RETRY COUNT HAS REACHED LIMIT:
 (6)  013620  023727  003466  000020                                                    CMP     RETRYC,#
 (9)  013626  001007                                                                    BNE     50226$
3525  013630                                       LET UNREC :B= UNREC + #1     ;SET UNRECOVERABLE FLAG
 (6)  013630  105237  003476                                                            INCB    UNREC
3526  013634                                       ERRHRD 14,RLEXM,STAERM       ;REPORT RECOVERABLE ERROR.
 (4)  013634  104456                                                                    TRAP    C$ERHRD
 (5)  013636  000016                                                                    .WORD   14
 (5)  013640  004374                                                                    .WORD   RLEXM
 (5)  013642  005642                                                                    .WORD   STAERM
3527  013644                                       POP R2
 (2)  013644  012602                                                                    MOV     (SP)+,R2
3528  013646                                     ENDIF
 (4)  013646                                                                    50226$:
3529  013646                                   ENDIF
```

```
 (4)  013646                                                          50225$:
3530  013646  000207               RTLRTN: RTS PC                ;RETURN
```

```
3532                                    ;       SUBR TO REWRITE A BAD, BUT RECOVERABLE WRITTEN RECORD.
3533                                    ;       REWRITE RECORD ON SAME SPOT: REPEAT 4 TIMES.
3534                                    ;       IF ALL 4 REPEATS GOOD, RECORD IS RECOVERED
3535                                    ;       AND A RECOVERABLE WRITE ERROR IS LOGGED.
3536                                    ;       IF ANY OF 4 REPEATS BAD, ERASE BAD RECORD, LOG SUSPECTED
3537                                    ;       BAD SPOT, RETRY AGAIN. RETRY 4 TIMES, UP TO 4 REPEATS EACH.
3538                                    ;       IF RECORD NOT GOOD AFTER 4 RETRIES, ERASE IT, EXIT WITH
3539                                    ;       ERROR FLAG WRTYER SET, PRINTING RETRY FAILED.
3540                                    ;       THIS ALL SCHEME IS REENTERED 20 TIMES MAX, IE 20 BAD
3541                                    ;       SPOTS MAX ARE ALLOWED.
3542                                    ;
3543                                    ;       INPUTS:
3544                                    ;       OUTPUTS:
3545                                    ;       REGISTERS:      R3,R4
3546                                    ;       CALLS:          BORERS, REWRT
3547
3548  013650                            WRTY:: IF DEVTBL(R5) NE #NINUSE THEN            ;IF DRIVE NOT DROPPED
 (6)  013650  026527  002536  177774                                                        CMP     DEVTBL(R
 (9)  013656  001540                                                                      BEQ     50227$
3549  013660                                  BEGIN RETRY
3550  013660                                    REPEAT
 (3)  013660                                                                                50231$:
3551  013660                                      BEGIN REPEAT
3552  013660                                        REPEAT
 (3)  013660                                                                                50233$:
3553  013660  004737  014374                          JSR PC,BORERS                       ;BACKSPACE/ERASE ONE RECORD
3554  013664                                          LET WRTYER :B= #0                   ;CLEAR WRITE RETRY ERROR
 (4)  013664  105037  003472                                                                CLRB    WRTYER
3555  013670  004737  014550                          JSR PC,REWRT                        ;REWRITE RECORD ON SAME SPOT
3556  013674                                          IF DEVTBL(R5) EQ #NINUSE THEN
 (6)  013674  026527  002536  177774                                                        CMP     DEVTBL(R
 (9)  013702  001004                                                                      BNE     50234$
3557  013704                                            LET RPTCNT :B= #3
 (4)  013704  112737  000003  003470                                                      MOVB    #3,RPTCN
3558  013712                                          ELSE
 (4)  013712  000400                                                                      BR      50235$
 (3)  013714                                                                                50234$:
3559  013714                                          ENDIF
 (4)  013714                                                                                50235$:
3560  013714                                          LET RPTCNT :B= RPTCNT + #1      ;COUNT REPEATS
 (6)  013714  105237  003470                                                                INCB    RPTCNT
3561  013720                                        UNTILB RPTCNT EQ #4 ORB WRTYER NE #0  ;LIMIT: 4 REPEATS OR REC
 (4)  013720  123727  003470  000004                                                        CMPB    RPTCNT,#
 (6)  013726  001403                                                                      BEQ     50236$
 (4)  013730  105737  003472                                                              TSTB    WRTYER
 (7)  013734  001751                                                                      BEQ     50233$
 (4)  013736                                                                                50236$:
3562  013736                                      END REPEAT                          ;
 (3)  013736                                                                                50232$:
3563  013736                                      LET RETRYC := RETRYC + #1            ;COUNT RETRIES
 (6)  013736  005237  003466                                                                INC     RETRYC
3564  013742                                        IF DEVTBL(R5) EQ #NINUSE THEN
 (6)  013742  026527  002536  177774                                                        CMP     DEVTBL(R
 (9)  013750  001002                                                                      BNE     50237$
3565  013752  000502                                  BR      3$
3566  013754                                          ELSE
```

```
  (4) 013754  000400                                                                              BR      50240$
  (3) 013756                                                                              50237$:
 3567 013756                                         ENDIF
  (4) 013756                                                                              50240$:
 3568 013756                                 IFB WRTYER EQ #0 THEN                 ;
  (6) 013756  105737  003472                                                                      TSTB    WRTYER
  (9) 013762  001001                                                                              BNE     50241$
 3569 013764                                     LEAVE RETRY                       ;EXIT RETRY LOOP IF RECOVERED
  (4) 013764  000457                                                                              BR      50230$
 3570 013766                                 ELSE                                  ;
  (3) 013766                                                                              50241$:
 3571 013766                                     IFB ERCVER NE #0 THEN             ;
  (6) 013766  105737  002207                                                                      TSTB    ERCVER
  (9) 013772  001415                                                                              BEQ     50243$
 3572 013774                                         PRINTB #BTMSG1,RETRYC,<B,RPTCNT>        ;PRINT SUSPECTED BAD SPO
  (9) 013774  005046                                                                              CLR     -(SP)
  (9) 013776  153716  003470                                                                      BISB    RPTCNT,(
  (8) 014002  013746  003466                                                                      MOV     RETRYC,-
  (7) 014006  012746  014170                                                                      MOV     #BTMSG1,
  (6) 014012  012746  000003                                                                      MOV     #3,-(SP)
  (3) 014016  010600                                                                              MOV     SP,R0
  (4) 014020  104414                                                                              TRAP    C$PNTB
  (4) 014022  062706  000010                                                                      ADD     #10,SP
 3573 014026                                     ENDIF                             ;
  (4) 014026                                                                              50243$:
 3574 014026                                     IF RETRYC EQ #1 THEN      ;ON FIRST RETRY, LOGG BAD SPOT
  (6) 014026  023727  003466  000001                                                              CMP     RETRYC,#
  (9) 014034  001021                                                                              BNE     50244$
 3575 014036                                         LET BTPT := BTADDR(R5)        ;BTPT IS BOTH THE BAD SPOT COUNT
  (4) 014036  016537  002550  003520                                                              MOV     BTADDR(R
 3576 014044                                         LET R4 := @BTPT + #2          ;AND THE LOGGING INDEX
  (4) 014044  017704  167450                                                                      MOV     @BTPT,R4
  (6) 014050  062704  000002                                                                      ADD     #2,R4
 3577 014054                                         LET @BTPT := R4               ;
  (4) 014054  010477  167440                                                                      MOV     R4,@BTPT
 3578 014060                                         IF R4 LOS #40. THEN           ;
  (6) 014060  020427  000050                                                                      CMP     R4,#40.
  (9) 014064  101005                                                                              BHI     50245$
 3579 014066                                             LET R3 := BTPT            ;STORE FIRST 20 BAD SPOTS
  (4) 014066  013703  003520                                                                      MOV     BTPT,R3
 3580 014072                                             LET R4 := R4 + R3         ;
  (6) 014072  060304                                                                              ADD     R3,R4
 3581 014074                                             LET (R4) := RECCNT(R5)    ;
  (4) 014074  016514  003330                                                                      MOV     RECCNT(R
 3582 014100                                         ENDIF
  (4) 014100                                                                              50245$:
 3583 014100                                     ENDIF                             ;
  (4) 014100                                                                              50244$:
 3584 014100                                     LET ERSFLG :B= ERSFLG + #1        ;ERASE FLAG TO ERASE BAD RECORD
  (6) 014100  105237  003534                                                                      INCB    ERSFLG
 3585 014104                                     LET RWERR :B= #0                  ;CANCELL ''LOG'' ERROR FLAG ON FAI
  (4) 014104  105037  003475                                                                      CLRB    RWERR
 3586 014110                                     LET RPTCNT :B= #0                 ;CLEAR REPEAT COUNT FOR NEXT RET
  (4) 014110  105037  003470                                                                      CLRB    RPTCNT
 3587 014114                                 ENDIF                                 ;
  (4) 014114                                                                              50242$:
```

```
3588  014114                                  UNTIL RETRYC EQ #4          ;LIMIT: 4 RETRIES
 (3)  014114  023727  003466  000004                                              CMP     RETRYC,#
 (6)  014122  001256                                                             BNE     50231$
3589  014124                                  END RETRY                   ;
 (3)  014124                                                                  50230$:
3590  014124                                  IFB WRTYER NE #0 THEN       ;
 (6)  014124  105737  003472                                                     TSTB    WRTYER
 (9)  014130  001413                                                             BEQ     50246$
3591  014132                                    IFB ERCVER NE #0 THEN     ;
 (6)  014132  105737  002207                                                     TSTB    ERCVER
 (9)  014136  001410                                                             BEQ     50247$
3592  014140                                      PRINTB #BTMSG3          ;PRINT RETRY FAILED
 (7)  014140  012746  014325                                                     MOV     #BTMSG3,
 (6)  014144  012746  000001                                                     MOV     #1,-(SP)
 (3)  014150  010600                                                             MOV     SP,R0
 (4)  014152  104414                                                             TRAP    C$PNTB
 (4)  014154  062706  000004                                                     ADD     #4,SP
3593  014160                                    ENDIF
 (4)  014160                                                                  50247$:
3594  014160                                  ENDIF
 (4)  014160                                                                  50246$:
3595  014160                                ENDIF
 (4)  014160                                                                  50227$:
3596  014160  000207                     3$:     RTS PC
3597
3598
3599
3600
3601
3602
3603  014162  000000                     WTYCMD: .WORD   0                   ;STORAGE FOR WRITE CMD WHILE RETRYING
3604  014164  000000                     WTYWRD: .WORD   0                   ;STORAGE FOR WRITE CMD WORD WHILE RETRYING
3605  014166  000000                     WTYBRF: .WORD   0                   ;STORAGE FOR WRITE BPCR WHILE RETRYING
3606
3607
3608  014170  040445  052523  050123    BTMSG1: .ASCIZ  /%ASUSPECT BAD SPOT AFTER %D1%A RETRY, %D1%A REPEAT%N/
      014176  041505  020124  040502
      014204  020104  050123  052117
      014212  040440  052106  051105
      014220  022440  030504  040445
      014226  051040  052105  054522
      014234  020054  042045  022461
      014242  020101  042522  042520
      014250  052101  047045     000
3609  014255     045  022516  041101    BTMSG2: .ASCIZ  /%N%ABAD TAPE OVERFLOW: CHANGE TAPE!%N%N/
      014262  042101  052040  050101
      014270  020105  053117  051105
      014276  046106  053517  020072
      014304  044103  047101  042507
      014312  052040  050101  020505
      014320  047045  047045     000
3610  014325     045  051101  052105    BTMSG3: .ASCIZ  /%ARETRY FAILED ON BAD SPOT...ERASED!%N/
      014332  054522  043040  044501
      014340  042514  020104  047117
      014346  041040  042101  051440
      014354  047520  027124  027056
```

```
        014362  051105  051501  042105
        014370  022441  000116
  3611                                  .EVEN
```

```
3613                            ;       SUBR TO BACSPACE ONE RECORD
3614                            ;       IF THE ERASE FLAG IS SET, THEN ERASE THAT RECORD
3615                            ;       INPUTS:         ERSFLG  1 = DO ERASE
3616                            ;       OUTPUTS:
3617                            ;       REGISTERS:
3618                            ;       CALLS:          EXCUTE, GOWAIT, CKHAE
3619
3620  014374                    BORERS:: LET PCMDWD := CMDWRD   ;SET COMMAND  TO SPACE REV
 (4)  014374  013737  003430  003434                                                    MOV     CMDWRD,P
3621  014402                           LET CMDWRD := #SRR      ;
 (4)  014402  012737  104410  003430                                                    MOV     #SRR,CMD
3622  014410                           LET CMDPKT := CMDWRD CLR.BY #BRF.C      ;
 (4)  014410  013737  003430  002314                                                    MOV     CMDWRD,C
 (6)  014416  042737  004000  002314                                                    BIC     #BRF.C,C
3623  014424                           LET CMDSAV := CMDPKT    ;
 (4)  014424  013737  002314  003432                                                    MOV     CMDPKT,C
3624  014432                           LET CMDPKT+CP.ADL := #1 ;
 (4)  014432  012737  000001  002316                                                    MOV     #1,CMDPK
3625  014440                           LET CMDLG := #0         ;
 (4)  014440  005037  003436                                                            CLR     CMDLG
3626  014444  004737  007614          JSR PC,CMDAC            ;
3627  014450  004737  010576          JSR PC,EXCUTE           ;
3628  014454  004737  011166          JSR PC,GOWAIT           ;
3629  014460  004737  017222          JSR PC,CKHAE            ;
3630  014464                           IFB ERSFLG NE #0 THEN   ;WHEN ERASE FLAG IS SET, DO ERASE
 (6)  014464  105737  003534                                                            TSTB    ERSFLG
 (9)  014470  001426                                                                    BEQ     50250$
3631  014472                             LET PCMDWD := CMDWRD  ;
 (4)  014472  013737  003430  003434                                                    MOV     CMDWRD,P
3632  014500                             LET CMDWRD := #ERS    ;
 (4)  014500  012737  100411  003430                                                    MOV     #ERS,CMD
3633  014506                             LET CMDPKT := CMDWRD  ;
 (4)  014506  013737  003430  002314                                                    MOV     CMDWRD,C
3634  014514                             LET CMDSAV := CMDPKT  ;
 (4)  014514  013737  002314  003432                                                    MOV     CMDPKT,C
3635  014522  004737  007614            JSR PC,CMDAC          ;
3636  014526  004737  010576            JSR PC,EXCUTE
3637  014532  004737  011166            JSR PC,GOWAIT
3638  014536  004737  017222            JSR PC,CKHAE
3639  014542                             LET ERSFLG :B= #0
 (4)  014542  105037  003534                                                            CLRB    ERSFLG
3640  014546                           ENDIF
 (4)  014546                                                                    50250$:
3641  014546  000207                   RTS PC
3642                            ;       SUBR TO REWRITE A BADLY WRITTEN RECORD
3643
3644  014550                    REWRT:  IF DEVTBL(R5) NE #NINUSE THEN   ;IF DRIVE NOT DROPPED
 (6)  014550  026527  002536  177774                                                    CMP     DEVTBL(R
 (9)  014556  001441                                                                    BEQ     50251$
3645  014560                           LET PCMDWD := CMDWRD    ;RESTORE WRITE COMMAND PACKET
 (4)  014560  013737  003430  003434                                                    MOV     CMDWRD,P
3646  014566                           LET CMDWRD := WTYWRD    ;
 (4)  014566  013737  014164  003430                                                    MOV     WTYWRD,C
3647  014574                           LET CMDPKT := WTYCMD    ;
 (4)  014574  013737  014162  002314                                                    MOV     WTYCMD,C
3648  014602                           LET CMDSAV := CMDPKT    ;
```

```
   (4)  014602  013737  002314  003432                                                      MOV     CMDPKT,C
  3649  014610                                  LET CMDPKT+CP.ADL := DATAWT     ;
   (4)  014610  013737  003416  002316                                                      MOV     DATAWT,C
  3650  014616                                  LET CMDPKT+CP.CNT := WTYBRF     ;
   (4)  014616  013737  014166  002322                                                      MOV     WTYBRF,C
  3651  014624                                  LET CMDLG := #2                 ;
   (4)  014624  012737  000002  003436                                                      MOV     #2,CMDLG
  3652  014632  004737  007614                  JSR PC,CMDAC
  3653  014636  004737  010576                  JSR PC,EXCUTE                   ;RE-WRITE RECORD
  3654  014642                                    IF DEVTBL(R5) NE #NINUSE THEN
   (6)  014642  026527  002536  177774                                                      CMP     DEVTBL(R
   (9)  014650  001404                                                                      BEQ     50252$
  3655  014652  004737  011166                  JSR PC,GOWAIT                   ;
  3656  014656  004737  017222                  JSR PC,CKHAE                    ;
  3657  014662                                    ENDIF
   (4)  014662                                                                  50252$:
  3658  014662                                  ENDIF
   (4)  014662                                                                  50251$:
  3659  014662  000207                          RTS PC
```

```
3661                                    ;       SUBROUTINE TO LOG BYTES READ/WRITTEN.
3662                                    ;       ALSO UPDATES READ/WRITE ERROR COUNTERS.
3663                                    ;       INPUTS:
3664                                    ;       OUTPUTS:
3665                                    ;       REGISTERS:      R2, R3, R4.
3666                                    ;       CALLS:
3667
3668  014664                            LOG::   IFB ERLOG EQ #0 THEN            ;IF DATA AND ERRORS HAVE NOT BEEN LOGGED
 (6)  014664  105737  003474                                                            TSTB    ERLOG
 (9)  014670  001126                                                                    BNE     50253$
3669  014672                                    LET ERLOG :B= ERLOG + #1           ;SET LOG DONE FLAG.
 (6)  014672  105237  003474                                                            INCB    ERLOG
3670  014676                                    LET R4 := CMDLG                    ;GET CURRENT CMD LOGGING CODE.
 (4)  014676  013704  003436                                                            MOV     CMDLG,R4
3671  014702                                    IF R4 NE #0 THEN                   ;IF THERE IS A CODE THEN:
 (6)  014702  005704                                                                    TST     R4
 (9)  014704  001520                                                                    BEQ     50254$
3672  014706                                      LET R4 := R4 - #2                ;ADJUST THE CODE FOR TABLE INDEX.
 (6)  014706  162704  000002                                                            SUB     #2,R4
3673  014712                                      LET R2 := R5 + BINC(R4) + #CNTBGN  ;R2 = ADR OF BYTE COUNT LSW.
 (4)  014712  010502                                                                    MOV     R5,R2
 (6)  014714  066402  015150                                                            ADD     BINC(R4)
 (7)  014720  062702  002560                                                            ADD     #CNTBGN,
3674  014724                                      LET (R2) := (R2) + BRFCNT      ;ADD BRF TO LSW.
 (6)  014724  063712  003426                                                            ADD     BRFCNT,(
3675  014730                                      IF MSGPKT+MS.RFC LOS BRFCNT THEN  ;IF THE RFC IS LOWER OR THE SAME AS
 (6)  014730  023737  002344  003426                                                    CMP     MSGPKT+M
 (9)  014736  101002                                                                    BHI     50255$
3676  014740                                        LET (R2) := (R2) - MSGPKT+MS.RFC  ;SUBTRACT RFC FROM EXPECTED BRF.
 (6)  014740  163712  002344                                                            SUB     MSGPKT+M
3677  014744                                      ENDIF
 (4)  014744                                                                  50255$:
3678  014744                                      LET R3 := R2 + #10             ;R3 = ADR OF 2ND WORD.
 (4)  014744  010203                                                                    MOV     R2,R3
 (6)  014746  062703  000010                                                            ADD     #10,R3
3679  014752                                      WHILE (R2) GT #999. DO
 (4)  014752                                                                  50256$:
 (6)  014752  021227  001747                                                            CMP     (R2),#99
 (9)  014756  003404                                                                    BLE     50257$
3680  014760                                        LET (R2) := (R2) - #1000.    ;UPDATE BYTE COUNT
 (6)  014760  162712  001750                                                            SUB     #1000.,(
3681  014764                                        LET (R3) := (R3) + #1        ;2ND WORD.
 (6)  014764  005213                                                                    INC     (R3)
3682  014766                                      ENDDO
 (4)  014766  000771                                                                    BR      50256$
 (3)  014770                                                                  50257$:
3683  014770                                      LET R2 := R3 + #10             ;R2 = ADR OF 3RD WORD.
 (4)  014770  010302                                                                    MOV     R3,R2
 (6)  014772  062702  000010                                                            ADD     #10,R2
3684  014776                                      WHILE (R3) GT #999. DO
 (4)  014776                                                                  50260$:
 (6)  014776  021327  001747                                                            CMP     (R3),#99
 (9)  015002  003404                                                                    BLE     50261$
3685  015004                                        LET (R3) := (R3) - #1000.    ;UPDATE BYTE COUNT
 (6)  015004  162713  001750                                                            SUB     #1000.,(
3686  015010                                        LET (R2) := (R2) + #1        ;3RD WORD.
```

```
  (6) 015010 005212                                                                    INC     (R2)
 3687 015012                              ENDDO
  (4) 015012 000771                                                                    BR      50260$
  (3) 015014                                                                   50261$:
 3688 015014                              LET R3 := R2 + #10          ;R3 = ADR OF 4TH WORD.
  (4) 015014 010203                                                                    MOV     R2,R3
  (6) 015016 062703 000010                                                             ADD     #10,R3
 3689 015022                              WHILE (R2) GT #999. DO
  (4) 015022                                                                   50262$:
  (6) 015022 021227 001747                                                             CMP     (R2),#99
  (9) 015026 003404                                                                    BLE     50263$
 3690 015030                                LET (R2) := (R2) - #1000.  ;UPDATE BYTE COUNT
  (6) 015030 162712 001750                                                             SUB     #1000.,(
 3691 015034                                LET (R3) := (R3) + #1      ;4TH WORD.
  (6) 015034 005213                                                                    INC     (R3)
 3692 015036                              ENDDO
  (4) 015036 000771                                                                    BR      50262$
  (3) 015040                                                                   50263$:
 3693 015040                              IFB RWERR NE #0 THEN        ;IF R/W ERROR, UPDATE ERROR COUNT.
  (6) 015040 105737 003475                                                             TSTB    RWERR
  (9) 015044 001440                                                                    BEQ     50264$
 3694 015046                                LET R2 := R5 + EINC(R4) + #WRREC  ;R2 = ADR OF COUNTER.
  (4) 015046 010502                                                                    MOV     R5,R2
  (6) 015050 066402 015156                                                             ADD     EINC(R4)
  (7) 015054 062702 002720                                                             ADD     #WRREC,R
 3695 015060                                IFB UNREC NE #0 THEN       ;IS THE ERROR UNRECOVERABLE?
  (6) 015060 105737 003476                                                             TSTB    UNREC
  (9) 015064 001404                                                                    BEQ     50265$
 3696 015066                                  LET R2 := R2 + #10       ;YES, POINT TO NEXT COUNTER.
  (6) 015066 062702 000010                                                             ADD     #10,R2
 3697 015072                                  LET (R2) := (R2) + #1    ;UPDATE THE ERROR COUNTER
  (6) 015072 005212                                                                    INC     (R2)
 3698 015074                                ELSE                       ;ELSE - IF ERROR IS RECOVERABLE:
  (4) 015074 000424                                                                    BR      50266$
  (3) 015076                                                                   50265$:
 3699 015076                                  LET (R2) := (R2) + #1    ;UPDATE THE ERROR COUNTER
  (6) 015076 005212                                                                    INC     (R2)
 3700 015100                                  IFB IREC EQ #0 THEN      ;IF ERROR RECOVERY IS ENABLED:
  (6) 015100 105737 002210                                                             TSTB    IREC
  (9) 015104 001020                                                                    BNE     50267$
 3701 015106                                    IFB DROPED EQ #0 ANDB ERCVER NE #0 THEN ;IF UNIT HAS NOT BEEN DR
  (6) 015106 105737 003530                                                             TSTB    DROPED
  (9) 015112 001015                                                                    BNE     50270$
  (6) 015114 105737 002207                                                             TSTB    ERCVER
  (9) 015120 001412                                                                    BEQ     50270$
 3702 015122                                      PRINTB #NURTY1,RETRYC    ;PRINT # OF RETRIES TO RECOVER
  (8) 015122 013746 003466                                                             MOV     RETRYC,-
  (7) 015126 012746 005157                                                             MOV     #NURTY1,
  (6) 015132 012746 000002                                                             MOV     #2,-(SP)
  (3) 015136 010600                                                                    MOV     SP,R0
  (4) 015140 104414                                                                    TRAP    C$PNTB
  (4) 015142 062706 000006                                                             ADD     #6,SP
 3703 015146                                    ENDIF                      ;PROVIDED PRINT HAS BEEN ENABLED
  (4) 015146                                                                   50270$:
 3704 015146                                  ENDIF
  (4) 015146                                                                   50267$:
```

```
 3705  015146                                          ENDIF
  (4)  015146                                                                                   50266$:
 3706  015146                                        ENDIF
  (4)  015146                                                                                   50264$:
 3707  015146                                      ENDIF
  (4)  015146                                                                                   50254$:
 3708  015146                                      ENDIF
  (4)  015146                                                                                   50253$:
 3709  015146  000207                              RTS PC
 3710                                      ;       INDEXES TO BYTE COUNTERS.
 3711  015150  000000                      BINC:   0                               ;WRITE.
 3712  015152  000040                              40                              ;READ REV.
 3713  015154  000100                              100                             ;READ FWD.
 3714                                      ;       INDEXES TO READ/WRITE ERROR COUNTERS.
 3715  015156  000000                      EINC:   0                               ;WRITE.
 3716  015160  000020                              20                              ;READ REV.
 3717  015162  000040                              40                              ;READ FWD.
 3718
 3719
 3720
```

```
                                        .SBTTL RAMER - READ AND DISPLAY SELECTED RAM
                                ;+
                                ;ROUTINE TO READ THE SELECTED RAM LOCATIONS
                                ;-
                                RAMER:: MOV     R5,-(SP)
                                        MOV     R4,-(SP)
                                        MOV     R3,-(SP)
                                        MOV     R2,-(SP)
                                        MOV     R1,-(SP)
                                        MOV     #RAMDATA,R1     ;ADDRESS TO SAVE THE RAM DATA
                                        MOV     RAMHLD,R2       ;BYTE ADDRESS OF THE FIRST RAM DATA
                                        MOV     RAMSIZ,R3       ;SET THE SIZE OF THE READ UP
                                        MOV     TSDB(R5),R4     ;MOV THE TSDB ADDRESS INTO R4
                                        INC     R4              ;ADD 1 TO IT
                                10$:    NOP
                                        JSR     PC,WSSR         ;WAIT FOR THE SSR TO SET
                                        MOVB    R2,(R4)         ;SELECT NEXT RAM ADDRESS
                                        JSR     PC,WSSR         ;WAIT FOR SSR TO SET
                                        MOVB    @TSBA(R5),(R1)+ ;READ THE RAM DATA
                                20$:    ADD     #1,R2           ;ADDRESS OF THE NEXT RAM LOCATION
                                        SOB     R3,10$          ;NUMBER OF LOCATIONS COUNTER
                                        MOV     RAMSIZ,R4       ;GET THE RAM SIZE
                                        MOV     RAMHLD,R2       ;GET THE STARTING RAM ADDRESS
                                        ADD     R2,R4           ;CALCULATE THE END ADDRESS
                                        SUB     #1,R4           ;CORRECT VALUE OF PRINTOUT
                                        PRINTX  #RAMIOP,R2,R4   ;RAM ADDRESS = 10 - 17, ETC.
  (9) 015264  010446                                                                    MOV     R4,-(SP)
  (8) 015266  010246                                                                    MOV     R2,-(SP)
  (7) 015270  012746  005401                                                            MOV     #RAMIOP,
  (6) 015274  012746  000003                                                            MOV     #3,-(SP)
  (3) 015300  010600                                                                    MOV     SP,R0
  (4) 015302  104415                                                                    TRAP    C$PNTX
  (4) 015304  062706  000010                                                            ADD     #10,SP
3748  015310  012701  003346                       MOV     #RAMDATA,R1     ;ADDRESS OF WHERE RAM DATA IS
3749  015314  013703  003406                       MOV     RAMSIZ,R3       ;THE SIZE OF THE RAM FIELD READ
3750  015320  005004                       30$:    CLR     R4              ;NO EXTRA DATA LEFT OVER
3751  015322  112104                               MOVB    (R1)+,R4        ;PICK UP BYTE OF RAM DATA
3752  015324  042704  177400                       BIC     #177400,R4      ;GET RID OF SIGN EXTEND
3753  015330                                       PRINTX  #RAMPD,R4       ;"010 211 111 222 377 000 123 134 ETC."
  (8) 015330  010446                                                                    MOV     R4,-(SP)
  (7) 015332  012746  005452                                                            MOV     #RAMPD,-
  (6) 015336  012746  000002                                                            MOV     #2,-(SP)
  (3) 015342  010600                                                                    MOV     SP,R0
  (4) 015344  104415                                                                    TRAP    C$PNTX
  (4) 015346  062706  000006                                                            ADD     #6,SP
3754  015352  077316                               SOB     R3,30$          ;LOOP UNTIL ALL PRINTED
3755  015354  012601                               MOV     (SP)+,R1
3756  015356  012602                               MOV     (SP)+,R2
3757  015360  012603                               MOV     (SP)+,R3
3758  015362  012604                               MOV     (SP)+,R4
3759  015364  012605                               MOV     (SP)+,R5
3760  015366  000207                       50$:    RTS     PC              ;RETURN
3761                                          ;
3762                                          ;       IF A WRITE/VERIFY COMMAND IS ISSUED, CONTROL IS THEN
3763                                          ;       TRANSFERRED TO THIS SUBROUTINE TO READ REVERSE, CHECK DATA,
3764                                          ;       READ FORWARD, CHECK DATA, THEN CONTINUE TO NEXT COMMAND.
                                        ;       INPUTS:
```

```
                        ;       OUTPUTS:
                        ;       REGISTERS:
                        ;       CALLS:          VFEXC.
                        ;
015370                  VFYDAT::  LET CMPDAT := #0              ;BTL
(4) 015370  005037  003410                                              CLR     CMPDAT
                                  LET STTIM  := #0              ;BTL
(4) 015374  005037  003414                                              CLR     STTIM
                                  IFB VFYFLG NE #0 THEN         ;IF DATA IS TO BE VERIFIED:
(6) 015400  105737  003524                                              TSTB    VFYFLG
(9) 015404  001436                                                      BEQ     50271$
                                  LET PCMDWD := CMDWRD          ;SAVE THE PREVIOUS COMMAND WORD.
(4) 015406  013737  003430  003434                                       MOV     CMDWRD,P
                                  LET CMDWRD := #RDR            ;COMMAND IS READ REV.
(4) 015414  012737  104401  003430                                       MOV     #RDR,CMD
                                  LET CMDLG  := #4              ;SET UP CMD LOGGING INDEX.
(4) 015422  012737  000004  003436                                       MOV     #4,CMDLG
015430  004737  015504            JSR   PC,VFEXC                ;GO READ ALL THE RECORDS REV.
                                  IF DEVTBL(R5) NE #NINUSE THEN ;IF DRIVE HAS BEEN DROPPED EXIT
(6) 015434  026527  002536  177774                                       CMP     DEVTBL(R
(9) 015442  001417                                                      BEQ     50272$
                                  LET CMPDAT := #0              ;BTL
(4) 015444  005037  003410                                              CLR     CMPDAT
                                  LET STTIM  := #0              ;BTL
(4) 015450  005037  003414                                              CLR     STTIM
                                  LET PCMDWD := CMDWRD          ;SAVE THE PREVIOUS COMMAND WORD.
(4) 015454  013737  003430  003434                                       MOV     CMDWRD,P
                                  LET CMDWRD := #RDF            ;COMMAND IS READ FWD.
(4) 015462  012737  104001  003430                                       MOV     #RDF,CMD
                                  LET CMDLG  := #6              ;SET UP CMD LOGGING INDEX.
(4) 015470  012737  000006  003436                                       MOV     #6,CMDLG
015476  004737  015504            JSR   PC,VFEXC                ;GO READ ALL RECORDS FWD.
015502                            ENDIF
(4) 015502                                                              50272$:
                                ENDIF
(4) 015502                                                              50271$:
015502  000207                  RTS     PC                    ;RETURN.
```

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

015504                  VFEXC:: LET CMDPKT := CMDWRD CLR.BY #BRF.C  ;COMMAND PACKET = READ REV OR FWD.
(4)  015504  013737  003430  002314                                          MOV     CMDWRD,C
(6)  015512  042737  004000  002314                                          BIC     #BRF.C,C
015520                          IFB SWBFLG NE #0 THEN           ;IF BYTES ARE TO BE SWAPPED:
(6)  015520  105737  003526                                                 TSTB    SWBFLG
(9)  015524  001403                                                         BEQ     50273$
015526                            LET CMDPKT := CMDPKT SET.BY #SWB.C  ;SET SWAB BIT IN CMD PACKET.
(6)  015526  052737  010000  002314                                          BIS     #SWB.C,C
015534                          ENDIF
(4)  015534                                                                 50273$:
015534                          LET CMDSAV := CMDPKT            ;SAVE COMMAND PACKET 1ST WORD.
(4)  015534  013737  002314  003432                                          MOV     CMDPKT,C
015542  013737  003420  002316  MOV     DATARD,CMDPKT+CP.ADL    ;SAVE BUFFER START ADDRESS.
015550                          LET NCNT := #0                  ;CLEAR NUMBER OF OPERATIONS.
(4)  015550  005037  003422                                                 CLR     NCNT
015554                          WHILE NCNT LT NCNT1 DO          ;WHILE THERE ARE RECORDS REMAINING:
(4)  015554                                                                 50274$:
(6)  015554  023737  003422  003424                                          CMP     NCNT,NCN
(9)  015562  002101                                                         BGE     50275$
015564  004737  007614            JSR PC,CMDAC                    ;STORE CMD ASCII IN ERROR MSG.
015570  105737  003533            TSTB  STREAM                    ;CHECK IF WE ARE STREAMING
015574  001006                    BNE   1$                        ;BRANCH OVER DEVTBL CHECK. THIS ENABLES
                                                        ;US TO TEST ONE DRIVE AT A TIME.
015576  004737  016614            JSR   PC,FIRSTU                 ;SET UP FOR FIRST UNIT.
015602                            WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE DEVICES REMAINING:
(4)  015602                                                                 50276$:
(6)  015602  026527  002536  177777                                          CMP     DEVTBL(R
(9)  015610  001445                                                         BEQ     50277$
015612                  1$:             IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
(6)  015612  032737  000400  003430                                          BIT     #MOD.CO,
(9)  015620  001421                                                         BEQ     50300$
015622                              IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
(6)  015622  032765  000002  003510                                          BIT     #XO.BOT,
(9)  015630  001014                                                         BNE     50301$
015632                                IF #XO.EOT SETIN EOTFLG(R5) THEN   ;BUT IF AT EOT
(6)  015632  032765  000001  003510                                          BIT     #XO.EOT,
(9)  015640  001406                                                         BEQ     50302$
015642                                  IFB ALLEOT NE #0 THEN           ;AND ALL OTHERS AT EOT
(6)  015642  105737  003532                                                 TSTB    ALLEOT
(9)  015646  001402                                                         BEQ     50303$
015650  004737  015770                    JSR PC,VFISU                  ;THEN READ VERIFY
015654                                  ENDIF                           ;IF NOT ALL AT EOT, FREEZE
(4)  015654                                                                 50303$:
015654                                ELSE                            ;IF NOT AT BOT AND
(4)  015654  000402                                                         BR      50304$
(3)  015656                                                                 50302$:
015656  004737  015770                    JSR PC,VFISU                  ;NOT AT EOT, READ VFY
015662                                  ENDIF
(4)  015662                                                                 50304$:
015662                              ENDIF
```

```
 (4)  015662                                                                      50301$:
3817  015662                                        ELSE                     ;ELSE IF CMD IS NOT REVERSE:
 (4)  015662  000412                                                                      BR      50305$
 (3)  015664                                                                      50300$:
3818  015664                                          IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
 (6)  015664  032765  000001  003510                                                      BIT     #XO.EOT,
 (8)  015672  001404                                                                      BEQ     50306$
 (6)  015674  032737  000001  003430                                                      BIT     #CMD.CO,
 (9)  015702  001002                                                                      BNE     50307$
 (6)  015704                                                                      50306$:
3819                                                                     ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
3820  015704  004737  015770                            JSR PC,VFISU           ;ISSUE CMD, CHECK STATUS AND DATA.
3821  015710                                          ENDIF
 (4)  015710                                                                      50307$:
3822  015710                                        ENDIF
 (4)  015710                                                                      50305$:
3823  015710  105737  003533                      TSTB STREAM              ;CHECK FOR TEST OF ON UNIT AT A TIME.
3824  015714  001003                              BNE 2$                   ;BRANCH, IF STREAMING TESTS.
3825  015716  004737  016662                      JSR PC,NEXTU             ;GO FIND THE NEXT UNIT.
3826  015722                                    ENDDO
 (4)  015722  000727                                                                      BR      50276$
 (3)  015724                                                                      50277$:
3827  015724  004737  017222              2$:   JSR   PC,CKHAE             ;CHECK FOR HALT AFTER EACH CMD.
3828  015730                                    IF DEVTBL(R5) EQ #NINUSE THEN ;IF DRIVES BEEN DROPPED EXIT
 (6)  015730  026527  002536  177774                                                      CMP     DEVTBL(R
 (9)  015736  001005                                                                      BNE     50310$
3829  015740                                    LET NCNT := NCNT1 - #1
 (4)  015740  013737  003424  003422                                                      MOV     NCNT1,NC
 (6)  015746  005337  003422                                                              DEC     NCNT
3830  015752                                    ENDIF
 (4)  015752                                                                      50310$:
3831  015752                                    LET NCNT := NCNT + #1        ;UPDATE THE RECORD COUNT.
 (6)  015752  005237  003422                                                              INC     NCNT
3832  015756                                    LET PCMDWD := CMDWRD         ;SAVE PREVIOUS COMMAND WORD.
 (4)  015756  013737  003430  003434                                                      MOV     CMDWRD,P
3833
3834  015764                                  ENDDO
 (4)  015764  000673                                                                      BR      50274$
 (3)  015766                                                                      50275$:
3835  015766  000207                          RTS     PC                   ;RETURN.
```

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

                        VFISU::
                            LET R2 := DATARD + #8.      ;INIT READ BUFFER POINTER.
                                                                        MOV     DATARD,R
                                                                        ADD     #8.,R2
                            WHILE R2 NE DATARD DO       ;UNTIL 8 BYTES HAVE BEEN SET,
                                                                50311$:
                                                                        CMP     R2,DATAR
                                                                        BEQ     50312$
                            LET -(R2) := #-1            ;INIT READ BUFFER.
                                                                        MOV     #-1,-(R2
                            ENDDO
                                                                        BR      50311$
                                                                50312$:
                            JSR PC,EXCUTE               ;GO EXECUTE THE COMMAND.
                            IFB DROPED EQ #0 THEN       ;IF UNIT HAS NOT BEEN DROPPED THEN:
                                                                        TSTB    DROPED
                                                                        BNE     50313$
                            JSR PC,GOWAIT       ;GO WAIT FOR DONE BIT.
                            ENDIF
                                                                50313$:
                            IFB DROPED EQ #0 THEN       ;IF UNIT HAS NOT BEEN DROPPED THEN:
                                                                        TSTB    DROPED
                                                                        BNE     50314$
                              IF #X0.BOT NOTSETIN EOTFLG(R5) THEN       ;WHEN NOT REVERSED INTO BOT, THEN
                                                                        BIT     #X0.BOT,
                                                                        BNE     50315$
                                IF L$TEST NE #3 THEN
                                                                        CMP     L$TEST,#
                                                                        BEQ     50316$
                                  JSR   PC,CKDATA       ;GO VERIFY DATA.
                                  ELSE
                                                                        BR      50317$
                                                                50316$:
                                   IF NCNT LT #3500. THEN
                                                                        CMP     NCNT,#35
                                                                        BGE     50320$
                                    LET DATRAT := #56.  ;1.7% DATA COMPARISONS
                                                                        MOV     #56.,DAT
                                   ELSE
                                                                        BR      50321$
                                                                50320$:
                                    IF DATRAT NE #15. THEN
                                                                        CMP     DATRAT,#
                                                                        BEQ     50322$
                                    LET DATRAT := #15.  ;6.67% DATA COMPARISONS
                                                                        MOV     #15.,DAT
                                    LET CMPDAT := #0
                                                                        CLR     CMPDAT
                                    ENDIF
```

```
  (4)  016130                                                              50322$:
 3865  016130                                ENDIF
  (4)  016130                                                              50321$:
 3866  016130  005237  003410               INC    CMPDAT          ;ONE MORE XFER BEFORE A COMPARISON
 3867  016134                               IF CMPDAT EQ DATRAT THEN
  (6)  016134  023737  003410  003412                                              CMP     CMPDAT,D
  (9)  016142  001015                                                              BNE     50323$
 3868  016144                                 IF STTIM NE #0 THEN ;BTL
  (6)  016144  005737  003414                                                      TST     STTIM
  (9)  016150  001405                                                              BEQ     50324$
 3869  016152  004737  016200                 JSR PC,CKDATA
 3870  016156                                 LET CMPDAT := #0
  (4)  016156  005037  003410                                                      CLR     CMPDAT
 3871  016162                                 ELSE
  (4)  016162  000405                                                              BR      50325$
  (3)  016164                                                              50324$:
 3872  016164                                 LET STTIM  := #1           ;BTL
  (4)  016164  012737  000001  003414                                              MOV     #1,STTIM
 3873  016172                                 LET CMPDAT := #0           ;BTL
  (4)  016172  005037  003410                                                      CLR     CMPDAT
 3874  016176                                 ENDIF                      ;BTL
  (4)  016176                                                              50325$:
 3875  016176                               ENDIF
  (4)  016176                                                              50323$:
 3876  016176                             ENDIF
  (4)  016176                                                              50317$:
 3877  016176                           ENDIF
  (4)  016176                                                              50315$:
 3878  016176                         ENDIF
  (4)  016176                                                              50314$:
 3879  016176  000207                 RTS PC
 3880
```

```
3882                                    ;       SUBROUTINE TO COMPARE DATA BETWEEN READ AND WRITE BUFFERS
3883                                    ;       AND PRINT ERROR MESSAGE ON MISCOMPARE.
3884                                    ;       INPUTS:
3885                                    ;       OUTPUTS:
3886                                    ;       REGISTERS:      R2, R3, R4.
3887                                    ;       CALLS:          GCMDA
3888
3889  016200                            CKDATA:: LET R3 := BRFCNT - MSGPKT+MS.RFC ;COMPUTE REC LENGTH READ
 (4)  016200  013703  003426                                                    MOV     BRFCNT,R
 (6)  016204  163703  002344                                                    SUB     MSGPKT+M
3890  016210                                    IF R3 EQ #0 THEN                ;WHEN NO DATA RECEIVED
 (6)  016210  005703                                                            TST     R3
 (9)  016212  001015                                                            BNE     50326$
3891  016214                                      ERRHRD 17,WTVERM,DTAERM               ;PRINT ERROR AND EXIT
 (4)  016214  104456                                                            TRAP    C$ERHRD
 (5)  016216  000021                                                            .WORD   17
 (5)  016220  004250                                                            .WORD   WTVERM
 (5)  016222  005474                                                            .WORD   DTAERM
3892  016224                                      PRINTB #DTAER4                ;COMPARE ROUTINE
 (7)  016224  012746  005074                                                    MOV     #DTAER4,
 (6)  016230  012746  000001                                                    MOV     #1,-(SP)
 (3)  016234  010600                                                            MOV     SP,R0
 (4)  016236  104414                                                            TRAP    C$PNTB
 (4)  016240  062706  000004                                                    ADD     #4,SP
3893  016244                                    ELSE
 (4)  016244  000560                                                            BR      50327$
 (3)  016246                                                                50326$:
3894  016246                                      IF R3 HI BRFCNT THEN          ;WHEN REC READ IS LONGER
 (6)  016246  020337  003426                                                    CMP     R3,BRFCN
 (9)  016252  101417                                                            BLOS    50330$
3895  016254                                        ERRHRD 17,WTVERM,DTAERM             ;THAN EXPECTED, PRINT
 (4)  016254  104456                                                            TRAP    C$ERHRD
 (5)  016256  000021                                                            .WORD   17
 (5)  016260  004250                                                            .WORD   WTVERM
 (5)  016262  005474                                                            .WORD   DTAERM
3896  016264                                        PRINTB #DTAER5,CMDPKT+CP.CNT      ;AN ERROR MESSAGE
 (8)  016264  013746  002322                                                    MOV     CMDPKT+C
 (7)  016270  012746  005115                                                    MOV     #DTAER5,
 (6)  016274  012746  000002                                                    MOV     #2,-(SP)
 (3)  016300  010600                                                            MOV     SP,R0
 (4)  016302  104414                                                            TRAP    C$PNTB
 (4)  016304  062706  000006                                                    ADD     #6,SP
3897  016310                                    ELSE                            ;AND EXIT ROUTINE
 (4)  016310  000536                                                            BR      50331$
 (3)  016312                                                                50330$:
3898  016312                                      LET CKDCNT := R3 - #1                 ;SAVE VERIFICATION LENGTH - 1.
 (4)  016312  010337  016610                                                    MOV     R3,CKDCN
 (6)  016316  005337  016610                                                    DEC     CKDCNT
3899  016322  005037  016612                     CLR CKDFF                     ;CLEAR # OF BYTES IN ERROR COUNTER.
3900  016326  005002                             CLR R2                        ;INIT BYTE COUNTER
3901  016330                                    LET R3 := DATAWT                ;GET WRITE BUFFER ADDRESS.
 (4)  016330  013703  003416                                                    MOV     DATAWT,R
3902  016334                                    LET R4 := DATARD                ;GET READ BUFFER ADDRESS.
 (4)  016334  013704  003420                                                    MOV     DATARD,R
3903  016340                                    IFB T1SWB NE #0 THEN            ;WHEN RUNNING TEST1-SUB 12,
 (6)  016340  105737  003531                                                    TSTB    T1SWB
```

```
  (9)  016344  001401                                                                    BEQ     50332$
 3904  016346  000313                          SWAB (R3)                  ;SWAP FIRST WORD OF WRT BFR
 3905  016350                                 ENDIF                      ;WHICH CONTAINS THE RECORD COUNT
  (4)  016350                                                                    50332$:
 3906  016350                                 REPEAT                     ;REPEAT UNTIL ALL DATA IS COMPARED:
  (3)  016350                                                                    50333$:
 3907  016350                                   IF R2 EQ CKDCNT THEN     ;IF THIS IS THE LAST BYTE THEN:
  (6)  016350  020237  016610                                                    CMP     R2,CKDCN
  (9)  016354  001011                                                            BNE     50334$
 3908  016356                                     IFB SWBFLG NE #0 THEN  ;IF BYTE SWAPPING IS ENABLED THEN:
  (6)  016356  105737  003526                                                    TSTB    SWBFLG
  (9)  016362  001406                                                            BEQ     50335$
 3909  016364                                       IF #BIT00 NOTSETIN CKDCNT THEN     ;IF RECORD LENGTH IS ODD
  (6)  016364  032737  000001  016610                                            BIT     #BIT00,C
  (9)  016372  001002                                                            BNE     50336$
 3910  016374  105723                                     TSTB      (R3)+            ;LAST BYTE WILL BE IN
 3911  016376  105724                                     TSTB      (R4)+            ;THE UPPER BYTE.
 3912  016400                                       ENDIF
  (4)  016400                                                                    50336$:
 3913  016400                                     ENDIF
  (4)  016400                                                                    50335$:
 3914  016400                                   ENDIF
  (4)  016400                                                                    50334$:
 3915  016400  121314                           CMPB (R3),(R4)             ;ARE THEY EQUAL.
 3916  016402  001452                           BEQ       3$                       ;BR IF SO.
 3917  016404  005737  016612                   TST CKDFF                  ;1 ST TIME THRU?
 3918  016410  001010                           BNE       2$                       ;BR IF NOT.
 3919  016412  005265  003300                   INC       VFYCNT(R5)               ;INC THE VERIFY ERROR COUNTER.
 3920  016416  005265  003310                   INC HRDCNT(R5)             ;INC THE HARD ERROR COUNT.
 3921  016422                                   ERRHRD 17,WTVERM,DTAERM    ;REPORT WRITE/VERIFY ERROR.
  (4)  016422  104456                                                            TRAP    C$ERHRD
  (5)  016424  000021                                                            .WORD   17
  (5)  016426  004250                                                            .WORD   WTVERM
  (5)  016430  005474                                                            .WORD   DTAERM
 3922  016432                          2$:      LET CKDFF := CKDFF + #1    ;INCREMENT # OF BYTES IN ERROR.
  (6)  016432  005237  016612                                                    INC     CKDFF
 3923  016436  111437  003446                   MOVB (R4),TIME1            ;SAVE WAS DATA FOR TYPOUT.
 3924  016442  042737  177400  003446           BIC       #177400,TIME1            ;CLEAR GARBAGE.
 3925  016450  111337  003450                   MOVB (R3),TIME2            ;SAVE SHOULD BE DATA FOR TYPOUT.
 3926  016454  042737  177400  003450           BIC       #177400,TIME2            ;CLEAR GARBAGE.
 3927  016462                                   IF CKDFF LT #11. THEN      ;IF ERROR BYTE COUNT IS LESS THAN 11:
  (6)  016462  023727  016612  000013                                            CMP     CKDFF,#1
  (9)  016470  002017                                                            BGE     50337$
 3928  016472                                     PRINTX #DTAER2,R2,<B,TIME1>,<B,TIME2>  ;PRINT EXP + ACT DATA.
 (10)  016472  005046                                                            CLR     -(SP)
 (10)  016474  153716  003450                                                    BISB    TIME2,(S
  (9)  016500  005046                                                            CLR     -(SP)
  (9)  016502  153716  003446                                                    BISB    TIME1,(S
  (8)  016506  010246                                                            MOV     R2,-(SP)
  (7)  016510  012746  004763                                                    MOV     #DTAER2,
  (6)  016514  012746  000004                                                    MOV     #4,-(SP)
  (3)  016520  010600                                                            MOV     SP,R0
  (4)  016522  104415                                                            TRAP    C$PNTX
  (4)  016524  062706  000012                                                    ADD     #12,SP
 3929  016530                                   ENDIF
  (4)  016530                                                                    50337$:
```

```
3930  016530  105723                  3$:     TSTB (R3)+                      ;UPDATE WRITE BUFFER ADDRESS.
3931  016532  105724                          TSTB (R4)+                      ;UPDATE READ BUFFER ADDRESS.
3932  016534  105722                          TSTB (R2)+                      ;UPDATE BYTE COUNTER.
3933  016536                                UNTIL R2 GT CKDCNT          ;END OF DATA COMPARE REPEAT LOOP.
 (3)  016536  020237  016610                                                    CMP     R2,CKDCN
 (6)  016542  003702                                                            BLE     50333$
3934  016544                                LET CKDCNT := CKDCNT + #1   ;CKDCNT EQUALS RECORD LENGTH.
 (6)  016544  005237  016610                                                    INC     CKDCNT
3935  016550                                IF CKDFF NE #0 THEN         ;IF COMPARE ERROR HAS OCCURED THEN:
 (6)  016550  005737  016612                                                    TST     CKDFF
 (9)  016554  001414                                                            BEQ     50340$
3936  016556                                  PRINTB #DTAER3,CKDFF,CKDCNT       ;PRINT # OF BYTES IN ERROR.
 (9)  016556  013746  016610                                                    MOV     CKDCNT,-
 (8)  016562  013746  016612                                                    MOV     CKDFF,-(
 (7)  016566  012746  005032                                                    MOV     #DTAER3,
 (6)  016572  012746  000003                                                    MOV     #3,-(SP)
 (3)  016576  010600                                                            MOV     SP,R0
 (4)  016600  104414                                                            TRAP    C$PNTB
 (4)  016602  062706  000010                                                    ADD     #10,SP
3937  016606                                  ENDIF
 (4)  016606                                                          50340$:
3938  016606                                ENDIF
 (4)  016606                                                          50331$:
3939  016606                              ENDIF
 (4)  016606                                                          50327$:
3940  016606  000207                      RTS     PC                  ;OTHERWISE, RETURN.
3941
3942  016610  000000              CKDCNT: .WORD   0                   ;# OF BYTES TO BE VERIFIED -1.
3943  016612  000000              CKDFF:  .WORD   0                   ;# OF BYTES IN ERROR COUNTER.
```

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

                                        FIRSTU::LET DROPED :B= #0               ;CLR UNIT DROPPED FLAG
                                                                                        CLRB    DROPED
                                                LET R5 := #0                    ;CLR DEVICE POINTER.
                                                                                        CLR     R5
                                                WHILE DEVTBL(R5) EQ #NINUSE DO  ;WHILE DEVICES ARE NOT IN USE:
                                                                                50341$:
                                                                                        CMP     DEVTBL(R
                                                                                        BNE     50342$
                                                  LET R5 := R5 + #2             ;POINT TO NEXT DEVICE.
                                                                                        ADD     #2,R5
                                                ENDDO
                                                                                        BR      50341$
                                                                                50342$:
                                                IF DEVTBL(R5) EQ #END THEN      ;IF ALL UNITS HAVE BEEN DROPPED THEN:
                                                                                        CMP     DEVTBL(R
                                                                                        BNE     50343$
                                                  DOCLN                           ;DO CLEAN CODE AND TERMINATE PASS.
                                                                                        TRAP    C$DCLN
                                                ENDIF
                                                                                50343$:
                                                LET L$LUN := DEVTBL(R5)         ;SET UNIT # IN 'HEADER' FOR ERROR REPORT
                                                                                        MOV     DEVTBL(R
                                                RTS     PC                      ;RETURN WITH 1ST DEVICE IN R5.
```

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

                                        NEXTU:: LET DROPED :B= #0               ;CLR UNIT DROPPED FLAG
                                                                                        CLRB    DROPED
                                                BIC     #177770,R5
                                                REPEAT                          ;REPEAT UNTIL THE NEXT DEVICE IS FOUND.
                                                                                50344$:
                                                  LET R5 := R5 + #2             ;UPDATE DEVICE TABLE POINTER.
                                                                                        ADD     #2,R5
                                                UNTIL DEVTBL(R5) NE #NINUSE
                                                                                        CMP     DEVTBL(R
                                                                                        BEQ     50344$
                                                LET L$LUN := DEVTBL(R5)         ;SET UNIT # IN 'HEADER' FOR ERROR REPORT
                                                                                        MOV     DEVTBL(R
                                                RTS     PC                      ;RETURN.
```

```
3983                                    ;       SUBROUTINE TO DROP A DEVICE FROM THE TEST SEQUENCE.
3984                                    ;       INPUTS:
3985                                    ;       OUTPUTS:
3986                                    ;       REGISTERS:
3987                                    ;       CALLS:          MOVMSG, PRXST, LOG
3988
3989  016716                            DROPU:: LET R5SAVE := R5
 (4)  016716  010537  003462                                                    MOV     R5,R5SAV
3990  016722                                    LET FTLCNT(R5) := FTLCNT(R5) + #1  ;INCREMENT THE FATAL ERROR COUNT.
 (6)  016722  005265  003320                                                    INC     FTLCNT(R
3991  016726                                    LET R4 := MSGPKT+MS.XS3 CLR.BY #377 ;GET UDIAG ERROR CODE FROM XSTAT3.
 (4)  016726  013704  002354                                                    MOV     MSGPKT+M
 (6)  016732  042704  000377                                                    BIC     #377,R4
3992  016736                                    LET R3 := MSGPKA(R5)            ;ADR OF THIS UNIT'S MSG PACKET.
 (4)  016736  016503  002506                                                    MOV     MSGPKA(R
3993  016742                                    LET R2 := #0                    ;CLR COUNTER.
 (4)  016742  005002                                                            CLR     R2
3994  016744                                    WHILE R2 NE #MSGCNT DO          ;WHILE THERE ARE MORE LOCATIONS:
 (4)  016744                                                                50345$:
 (6)  016744  020227  000016                                                    CMP     R2,#MSGC
 (9)  016750  001405                                                            BEQ     50346$
3995  016752                                      LET (R3)+ := #-1              ;INIT THE MSG PACKET WITH ALL 1'S
 (4)  016752  012723  177777                                                    MOV     #-1,(R3)
3996  016756                                      LET R2 := R2 + #2             ;UPDATE COUNTER.
 (6)  016756  062702  000002                                                    ADD     #2,R2
3997  016762                                    ENDDO
 (4)  016762  000770                                                            BR      50345$
 (3)  016764                                                                50346$:
3998  016764                            LET @TSDB(R5) := #GSCPK                 ;INITIATE A GET STATUS COMMAND.
 (4)  016764  012775  002324  002456                                            MOV     #GSCPK,@
3999  016772  004737  011516           JSR PC,WSSR                             ;WAIT A WHILE FOR SSR=1
4000  016776  004737  011552           JSR PC,MOVMSG                           ;MOVE MSG PACKET TO COMMON AREA.
4001  017002                            IF R4 EQ #X3.RNY THEN                   ;IF WE HAVE A CAPSTAN RUNAWAY THEN:
 (6)  017002  020427  157400                                                    CMP     R4,#X3.R
 (9)  017006  001005                                                            BNE     50347$
4002  017010                              ERRDF 16,RNYM,STAERM                    ;REPORT CAPSTAN RUNAWAY WITH TACH CNT.
 (4)  017010  104455                                                            TRAP    C$ERDF
 (5)  017012  000020                                                            .WORD   16
 (5)  017014  004570                                                            .WORD   RNYM
 (5)  017016  005642                                                            .WORD   STAERM
4003  017020                            ELSE                                    ;ELSE-IF NOT A RUNNAWAY:
 (4)  017020  000402                                                            BR      50350$
 (3)  017022                                                                50347$:
4004  017022  004737  017140             JSR PC,PRXST                          ;PRINT EXTENDED STATUS REGISTERS.
4005  017026                            ENDIF
 (4)  017026                                                                50350$:
4006  017026                            IFB RECLOG NE #0 THEN                   ;IF THE RECORD HAS BEEN LOGGED THEN:
 (6)  017026  105737  003473                                                    TSTB    RECLOG
 (9)  017032  001404                                                            BEQ     50351$
4007  017034                              LET DROPED :B= DROPED + #1            ;SET UNIT DROPPED FLAG.
 (6)  017034  105237  003530                                                    INCB    DROPED
4008  017040  004737  014664             JSR PC,LOG                            ;LOG DATA BYTES + RD/WR ERRORS.
4009  017044                            ENDIF
 (4)  017044                                                                50351$:
4010  017044                            DORPT                                   ;PRINT PERFORMANCE REPORT
 (3)  017044  104424                                                            TRAP    C$DRPT
```

```
4011  017046                    DROPUA: IF PASCNT(R5) NE #0 THEN
 (6)  017046  005765  003260                                        TST     PASCNT(R
 (9)  017052  001402                                                BEQ     50352$
4012  017054
 (6)  017054  005365  003260              LET PASCNT(R5) := PASCNT(R5) - #1
                                                                    DEC     PASCNT(R
4013  017060                          ENDIF
 (4)  017060                                                    50352$:
4014  017060                          LET DROPN := DEVTBL(R5)          ;SAVE # OF UNIT TO BE DROPPED.
 (4)  017060  016537  002536  017136                                 MOV     DEVTBL(R
4015  017066                          LET R0 := R5 SHIFT -1            ;R0=LOGICAL DEVICE NUMBER
 (4)  017066  010500                                                MOV     R5,R0
 (7)  017070  006200                                                ASR     R0
4016  017072
 (3)  017072  104451                      DODU R0                ;DROP THE UNIT: EXEC BGNDU-ENDDU CODE IF IDU = 0
                                                                    TRAP    C$DODU
4017  017074                          IF DEVTBL(R5) NE #NINUSE THEN    ;IF UNIT NOT DROPPED
 (6)  017074  026527  002536  177774                                 CMP     DEVTBL(R
 (9)  017102  001410                                                BEQ     50353$
4018  017104
 (6)  017104  105737  002210              IFB IREC EQ #0 THEN              ;IF RECOVERY IS ENABLED THEN:
                                                                    TSTB    IREC
 (9)  017110  001005                                                BNE     50354$
4019  017112  000240                            NOP
4020  017114  000240                            NOP
4021  017116  000240                            NOP
4022  017120                                   LET STAFLG :B= STAFLG + #1      ;SET START FLAG TO ENABLE REWIND,
 (6)  017120  105237  003536                                                   INCB    STAFLG
4023  017124                              ENDIF
 (4)  017124                                                    50354$:
4024  017124                          ENDIF
 (4)  017124                                                    50353$:
4025  017124                          DRORTN: LET DROPED :B= DROPED + #1      ;SET UNIT DROPPED FLAG.
 (6)  017124  105237  003530                                                   INCB    DROPED
4026  017130                                  LET R5 := R5SAVE
 (4)  017130  013705  003462                                                   MOV     R5SAVE,R
4027  017134  000207                         RTS     PC                      ;RETURN.
4028
4029  017136  000000                  DROPN:  .WORD   0                       ;# OF UNIT TO BE DROPPED
```

```
                                        ;       SUBROUTINE TO PRINT EXTENDED STATUS REGISTERS.
                                        ;       INPUTS:
                                        ;       OUTPUTS:
                                        ;       REGISTERS:
                                        ;       CALLS:
                                        PRXST:: PRINTX #GETSTM
                                                                                MOV     #GE1STM,
                                                                                MOV     #1,-(SP)
                                                                                MOV     SP,R0
                                                                                TRAP    C$PNTX
                                                                                ADD     #4,SP
                                                PRINTX  #STAER5,MSGPKT+MS.XS0,MSGPKT+MS.XS1,MSGPKT+MS.XS2,MSGPKT+MS.XS3
                                                                                MOV     MSGPKT+M
                                                                                MOV     MSGPKT+M
                                                                                MOV     MSGPKT+M
                                                                                MOV     MSGPKT+M
                                                                                MOV     #STAER5,
                                                                                MOV     #5,-(SP)
                                                                                MOV     SP,R0
                                                                                TRAP    C$PNTX
                                                                                ADD     #14,SP
                                                RTS PC


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

                                        CKHAE:: IFB HAE NE #0 THEN              ;IF HALT FLAG IS SET:
                                                                                TSTB    HAE
                                                                                BEQ     50355$
                                                  IFB MISCFG EQ #0 THEN         ;
                                                                                TSTB    MISCFG
                                                                                BNE     50356$
                                                    MANUAL                      ;IS MANUAL INTERVENTION ALLOWED?
                                                                                TRAP    C$MANI
                                                    BNCOMPLETE CKHRTN           ;BR IF NOT.
                                                                                BCC     CKHRTN
                                                    LET R4 := CMDWRD            ;COMMAND WORD.
                                                                                MOV     CMDWRD,R
                                                    JSR PC,GCMDA                ;FETCH ADR OF CMD ASCII.
                                                    LET HALTM :B= (R3)+         ;MOVE CMD ASCII
                                                                                MOVB    (R3)+,HA
                                                    LET HALTM+1 :B= (R3)+
                                                                                MOVB    (R3)+,HA
                                                    LET HALTM+2 :B= (R3)                ;INTO MESSAGE.
                                                                                MOVB    (R3),HAL
                                                    GMANIL HALTM,TIME1,1,YES    :HALT - WAIT FOR AN OEPRATOR INPUT.
                                                                                TRAP    C$GMAN
                                                                                BR      10000$
                                                                                .WORD   TIME1
```

```
  (5)  017274  000130                                                                    .WORD   T$CODE
  (5)  017276  004126                                                                    .WORD   HALTM
  (5)  017300  000001                                                                    .WORD   1
  (3)  017302                                10000$:
4060   017302                                          ELSE
  (4)  017302  000402                                                                    BR      50357$
                                                                                50356$:
  (3)  017304
4061   017304                                            LET MISCFG :B= #0          ;
  (4)  017304  105037  003541                                                           CLRB    MISCFG
4062   017310                                          ENDIF
  (4)  017310                                                                   50357$:
4063   017310                                        ENDIF
  (4)  017310                                                                   50355$:
4064   017310  000207                      CKHRTN: RTS     PC                  ;RETURN
```

```
 4066                                   ;       SUBROUTINE TO CREATE THE SEQUENCE FOR A WRITE TAPE MARK
 4067                                   ;       COMMAND.  WILL EXECUTE COMMAND TO UUT.
 4068                                   ;       INPUTS:
 4069                                   ;       OUTPUTS: CMDSEQ
 4070                                   ;       CALLS:  SETUP, CMDAC, EXCUTE, GOWAIT
 4071
 4072  017312                           WRITEM::
 4073  017312                                   LET R1 := #CMDSEQ
  (4)  017312  012701  003544                                                      MOV     #CMDSEQ,
 4074  017316                                   LET (R1)+ := #WTM       ;COMMAND
  (4)  017316  012721  100011                                                      MOV     #WTM,(R1
 4075  017322                                   LET (R1)+ := #1         ;BRF
  (4)  017322  012721  000001                                                      MOV     #1,(R1)+
 4076  017326                                   LET (R1)+ := #1         ;ITERATIONS
  (4)  017326  012721  000001                                                      MOV     #1,(R1)+
 4077  017332  005721                           TST (R1)+               ;PATTERN
 4078  017334                                   LET (R1)+ := #END       ;TERMINATOR
  (4)  017334  012721  177777                                                      MOV     #END,(R1
 4079  017340                                   LET R1 := #CMDSEQ       ;TOP OF BUFFER
  (4)  017340  012701  003544                                                      MOV     #CMDSEQ,
 4080  017344  004737  007722                   JSR PC, SETUP           ;SET UP THE TABLE
 4081  017350  004737  007614                   JSR PC, CMDAC           ;LOAD THE ASCII
 4082  017354  004737  010576                   JSR PC, EXCUTE          ;ISSUE THE WTM COMMAND
 4083  017360  004737  011166                   JSR PC, GOWAIT          ;WAIT FOR THE COMMAND TO FINISH
 4084  017364  000207                           RTS PC                  ;RETURN TO CALLER
 4085                                           .EVEN
 4086
 4087  017366                                   ENDMOD
```

```
                                .TITLE MISCELLANEOUS SECTIONS
                                .SBTTL  REPORT CODING SECTION

017366                                  BGNMOD

                                ;++
                                ; THE REPORT CODING SECTION CONTAINS THE
                                ; 'PRINTS'' CALLS THAT GENERATE STATISTICAL REPORTS.
                                ;--

017366                                  BGNRPT
017366                          L$RPT::

                                        LET     R5SAVE := R5            ;SAVE CURRENT DEVICE POINTER.
017366  010537  003462                                                          MOV     R5,R5SAV
017372  004737  016614                  JSR     PC,FIRSTU               ;FIND THE FIRST UNIT.
017376                                  WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE MORE DEVICES:
017376                                                                          50360$:
017376  026527  002536  177777                                                  CMP     DEVTBL(R
017404  001562                            PRINTS      #RPT1A,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
017406                                                                          BEQ     50361$
017406  016546  003330                                                          MOV     RECCNT(R
017412  016546  003260                                                          MOV     PASCNT(R
017416  016546  002536                                                          MOV     DEVTBL(R
017422  012746  020230                                                          MOV     #RPT1A,-
017426  012746  000004                                                          MOV     #4,-(SP)
017432  010600                                                                  MOV     SP,R0
017434  104416                                                                  TRAP    C$PNTS
017436  062706  000012                                                          ADD     #12,SP
017442                          PRINTS      #RPT1B,WRBC+30(R5),WRBC+20(R5),WRBC+10(R5),WRBC(R5)
017442  016546  002560                                                          MOV     WRBC(R5)
017446  016546  002570                                                          MOV     WRBC+10(
017452  016546  002600                                                          MOV     WRBC+20(
017456  016546  002610                                                          MOV     WRBC+30(
017462  012746  020305                                                          MOV     #RPT1B,-
017466  012746  000005                                                          MOV     #5,-(SP)
017472  010600                                                                  MOV     SP,R0
017474  104416                                                                  TRAP    C$PNTS
017476  062706  000014                                                          ADD     #14,SP
017502                          PRINTS      #RPT1C,RRBC+30(R5),RRBC+20(R5),RRBC+10(R5),RRBC(R5)
017502  016546  002620                                                          MOV     RRBC(R5)
017506  016546  002630                                                          MOV     RRBC+10(
017512  016546  002640                                                          MOV     RRBC+20(
017516  016546  002650                                                          MOV     RRBC+30(
017522  012746  020356                                                          MOV     #RPT1C,-
017526  012746  000005                                                          MOV     #5,-(SP)
017532  010600                                                                  MOV     SP,R0
017534  104416                                                                  TRAP    C$PNTS
017536  062706  000014                                                          ADD     #14,SP
017542                          PRINTS      #RPT1D,RFBC+30(R5),RFBC+20(R5),RFBC+10(R5),RFBC(R5)
017542  016546  002660                                                          MOV     RFBC(R5)
017546  016546  002670                                                          MOV     RFBC+10(
017552  016546  002700                                                          MOV     RFBC+20(
017556  016546  002710                                                          MOV     RFBC+30(
017562  012746  020427                                                          MOV     #RPT1D,-
```

```
   (6)  017566  012746  000005                                                                  MOV     #5,-(SP)
   (3)  017572  010600                                                                          MOV     SP,R0
   (4)  017574  104416                                                                          TRAP    C$PNTS
   (4)  017576  062706  000014                                                                  ADD     #14,SP
 4132   017602                    PRINTS        #RPT1F,WRREC(R5),RRREC(R5),RFREC(R5)
  (10)  017602  016546  002760                                                                  MOV     RFREC(R5
  (9)   017606  016546  002740                                                                  MOV     RRREC(R5
  (8)   017612  016546  002720                                                                  MOV     WRREC(R5
  (7)   017616  012746  020533                                                                  MOV     #RPT1F,-
  (6)   017622  012746  000004                                                                  MOV     #4,-(SP)
  (3)   017626  010600                                                                          MOV     SP,R0
  (4)   017630  104416                                                                          TRAP    C$PNTS
  (4)   017632  062706  000012                                                                  ADD     #12,SP
 4133   017636                    PRINTS        #RPT1G,WRUNR(R5),RRUNR(R5),RFUNR(R5)
  (10)  017636  016546  002770                                                                  MOV     RFUNR(R5
  (9)   017642  016546  002750                                                                  MOV     RRUNR(R5
  (8)   017646  016546  002730                                                                  MOV     WRUNR(R5
  (7)   017652  012746  020604                                                                  MOV     #RPT1G,-
  (6)   017656  012746  000004                                                                  MOV     #4,-(SP)
  (3)   017662  010600                                                                          MOV     SP,R0
  (4)   017664  104416                                                                          TRAP    C$PNTS
  (4)   017666  062706  000012                                                                  ADD     #12,SP
 4134   017672                    IFB BADTSW NE #0 THEN         ;
  (6)   017672  105737  002211                                                                  TSTB    BADTSW
  (9)   017676  001402                                                                          BEQ     50362$
 4135   017700  004737  017762      JSR PC,BTRPT        ;GO PRINT BAD TAPE SPOTS WHEN  ENABLED
 4136   017704                    ENDIF
  (4)   017704                                                                    50362$:
 4137   017704                    PRINTS        #RPT1I,SCCNT(R5),HRDCNT(R5),FTLCNT(R5),VFYCNT(R5)
  (11)  017704  016546  003300                                                                  MOV     VFYCNT(R
  (10)  017710  016546  003320                                                                  MOV     FTLCNT(R
  (9)   017714  016546  003310                                                                  MOV     HRDCNT(R
  (8)   017720  016546  003270                                                                  MOV     SCCNT(R5
  (7)   017724  012746  021001                                                                  MOV     #RPT1I,-
  (6)   017730  012746  000005                                                                  MOV     #5,-(SP)
  (3)   017734  010600                                                                          MOV     SP,R0
  (4)   017736  104416                                                                          TRAP    C$PNTS
  (4)   017740  062706  000014                                                                  ADD     #14,SP
 4138   017744  004737  016662        JSR   PC,NEXTU                ;FIND THE NEXT UNIT.
 4139   017750                    ENDDO
  (4)   017750  000612                                                                          BR      50360$
  (3)   017752                                                                    50361$:
 4140   017752                    LET R5 := R5SAVE              ;RESTORE CURRENT DEVICE POINTER.
  (4)   017752  013705  003462                                                                  MOV     R5SAVE,R
 4141   017756                    EXIT    RPT
  (4)   017756  000167                                                                          .WORD   J$JMP
  (3)   017760  001230                                                                          .WORD   L10010-2
 4142
 4143                       ;      SUBR TO PRINT BAD TAPES SPOTS DURING THE REPORT PRINTS
 4144                       ;      WRITE RETRIES: CUMULATIVE COUNT
 4145                       ;      BAD TAPE SPOTS: COUNT PER TAPE PASS ONLY, NOT CUMULATIVE.
 4146                       ;      COUNT OF RECOVERABLE WRITE ERRORS EXCLUDES BAD TAPE SPOTS.
 4147
 4148   017762             BTRPT:  PRINTS  #RPT1E,WRTYCT(R5)       ;PRINT GLOBAL WRITE RETRY COUNT
  (8)   017762  016546  003250                                                                  MOV     WRTYCT(R
  (7)   017766  012746  020655                                                                  MOV     #RPT1E,-
```

```
   (6)  017772  012746  000002                                                                    MOV     #2,-(SP)
   (3)  017776  010600                                                                            MOV     SP,R0
   (4)  020000  104416                                                                            TRAP    C$PNTS
   (4)  020002  062706  000006                                                                    ADD     #6,SP
 4149   020006
   (4)  020006  016537  002550  003520  LET BTPT := BTADDR(R5)  ;BTPT IS BOTH THE BAD TAPE SPOT COUNTER
                                                                                                  MOV     BTADDR(R
 4150   020014
   (4)  020014  017703  163500  LET R3 := @BTPT SHIFT -1        ;AND THE LOGGING INDEX
                                                                                                  MOV     @BTPT,R3
   (7)  020020  006203                                                                            ASR     R3
 4151   020022
   (8)  020022  010346  PRINTS  #RPT1J,R3       ;PRINT # OF BAD TAPE SPOTS
   (7)  020024  012746  020705                                                                    MOV     R3,-(SP)
   (6)  020030  012746  000002                                                                    MOV     #RPT1J,-
   (3)  020034  010600                                                                            MOV     #2,-(SP)
   (4)  020036  104416                                                                            MOV     SP,R0
   (4)  020040  062706  000006                                                                    TRAP    C$PNTS
 4152   020044                                                                                   ADD     #6,SP
   (6)  020044  005703  IF R3 NE #0 THEN                ;PRINT RECORD # I" BAD SPOTS DETECTED
   (9)  020046  001457                                                                            TST     R3
 4153   020050                                                                                   BEQ     50363$
   (6)  020050  020327  000024    IF R3 HI #20. THEN           ;
   (9)  020054  101402                                                                            CMP     R3,#20.
 4154   020056                                                                                   BLOS    50364$
   (4)  020056  012703  000024      LET R3 := #20.             ;20 BAD SPOTS IS THE LIMIT
                                                                                                  MOV     #20.,R3
 4155   020062    ENDIF
   (4)  020062                                                                        50364$:
 4156   020062    PRINTS        #CRLFSP          ;
   (7)  020062  012746  005302                                                                    MOV     #CRLFSP,
   (6)  020066  012746  000001                                                                    MOV     #1,-(SP)
   (3)  020072  010600                                                                            MOV     SP,R0
   (4)  020074  104416                                                                            TRAP    C$PNTS
   (4)  020076  062706  000004                                                                    ADD     #4,SP
 4157   020102
   (4)  020102  013704  003520  LET R4 := BTPT + #2    ;FETCH A BAD SPOT ID
   (6)  020106  062704  000002                                                                    MOV     BTPT,R4
 4158   020112                                                                                   ADD     #2,R4
   (4)  020112  005002  LET R2 := #0          ;R2 = PRINT COUNT PER LINE: 10 MAX
 4159   020114                                                                                   CLR     R2
   (3)  020114  REPEAT                              ;
 4160   020114                                                                         50365$:
   (8)  020114  011446    PRINTS      #RPT1K,(R4)      ;PRINT A BAD SPOT ID
   (7)  020116  012746  020772                                                                    MOV     (R4),-(S
   (6)  020122  012746  000002                                                                    MOV     #RPT1K,-
   (3)  020126  010600                                                                            MOV     #2,-(SP)
   (4)  020130  104416                                                                            MOV     SP,R0
   (4)  020132  062706  000006                                                                    TRAP    C$PNTS
 4161   020136                                                                                   ADD     #6,SP
   (6)  020136  005202    LET R2 := R2 + #1    ;COUNT PRINTS
 4162   020140                                                                                   INC     R2
   (6)  020140  062704  000002    LET R4 := R4 + #2    ;NEXT
 4163   020144                                                                                   ADD     #2,R4
   (6)  020144  020227  000012    IF R2 EQ #10. THEN  ;
   (9)  020150  001014                                                                            CMP     R2,#10.
 4164   020152                                                                                   BNE     50366$
   (7)  020152  012746  005302      PRINTS #CRLFSP             ;GO TO NEXT PRINT LINE PAST 10 PRINTS
   (6)  020156  012746  000001                                                                    MOV     #CRLFSP,
                                                                                                  MOV     #1,-(SP)
```

```
   (3)  020162  010600                                                                          MOV     SP,R0
   (4)  020164  104416                                                                          TRAP    C$PNTS
   (4)  020166  062706  000004                                                                  ADD     #4,SP
 4165   020172                          LET R3 := R3 - #10.       ;ADJUST BAD SPOT COUNT
   (6)  020172  162703  000012                                                                  SUB     #10.,R3
 4166   020176                          LET R2 := R2 - #10.       ;ADJUST PRINT COUNT
   (6)  020176  162702  000012                                                                  SUB     #10.,R2
 4167   020202                        ENDIF                       ;
   (4)  020202                                                                      50366$:
 4168   020202                       UNTIL R2 EQ R3               ;LIMIT: # OF BAD SPOTS
   (3)  020202  020203                                                                          CMP     R2,R3
   (6)  020204  001343                                                                          BNE     50365$
 4169   020206                      ENDIF                        ;
   (4)  020206                                                                      50363$:
 4170   020206                      PRINTS  #CRLF                ;
   (7)  020206  012746  005277                                                                  MOV     #CRLF,-(
   (6)  020212  012746  000001                                                                  MOV     #1,-(SP)
   (3)  020216  010600                                                                          MOV     SP,R0
   (4)  020220  104416                                                                          TRAP    C$PNTS
   (4)  020222  062706  000004                                                                  ADD     #4,SP
 4171   020226  000207                      RTS PC
 4172
 4184                                      .NLIST  BEX
 4185   020230  047045  047045  040445  RPT1A:  .ASCIZ  /%N%N%AUNIT %D1%S3%APASS:%D5%S3%ARECORD:%D5%N/
 4186   020305     045  041101  052131  RPT1B:  .ASCIZ  /%ABYTES WRITTEN  %D3%A,%Z3%A,%Z3%A,%Z3%N/
 4187   020356  040445  054502  042524  RPT1C:  .ASCIZ  /%ABYTES READ REV %D3%A,%Z3%A,%Z3%A,%Z3%N/
 4188   020427     045  041101  052131  RPT1D:  .ASCII  /%ABYTES READ FWD %D3%A,%Z3%A,%Z3%A,%Z3%N/
 4189   020477     045  031123  022463          .ASCIZ  /%S23%AWRT%S4%ARDR%S4%ARDF%N/
 4190   020533     045  051101  041505  RPT1F:  .ASCIZ  /%ARECOVERABLE ERRORS   %D5%S2%D5%S2%D5%N/
 4191   020604  040445  047125  042522  RPT1G:  .ASCIZ  /%AUNRECOVERABLE ERRORS %D5%S2%D5%S2%D5%N/
 4192   020655     045  053501  044522  RPT1E:  .ASCIZ  /%AWRITE RETRIES%SB%D5%N/
 4193   020705     045  022516  031104  RPT1J:  .ASCIZ  /%N%D2%A BAD SPOTS THIS TAPE PASS PRECEDING RECORD #:/
 4194   020772  042045  022465  030523  RPT1K:  .ASCIZ  /%D5%S1/
 4195   021001     045  051501  042520  RPT1I:  .ASCII  '%ASPEC COND%S3%AHARD%S3%AFATAL%S3%ACOMPARE%N'
 4196   021055     045  031523  042045          .ASCIZ  /%S3%D5%S3%D5%S3%D5%S3%D5%N%N/
 4197   021112  047045  040445  040524  TAPCAP: .ASCII  /%N%ATAPE LENGTH MUST BE 2400 FT. TO /
 4198   021156  047503  051122  041505          .ASCIZ  /CORRECTLY RUN THIS TEST!!%N/
 4199                                      .LIST   BEX
 4200                                      .EVEN
 4201   021212                           ENDRPT
   (3)  021212                   L10010:
   (3)  021212  104425                                                                          TRAP    C$RPT
 4202
 4203                                    ;LOAD DEVICE PROTECTION TABLE
 4204                                    ;TABLE FOR SUPERVISOR TO IDENTIFY THE P-TBL FOR THE LOAD DEV
 4205                                    ;THE SUPERVISOR USES THE TBL TO WARN THE OPERATOR WHEN HE TRIES TO TEST THE LOAD
 4206
 4207   021214                           BGNPROT
   (3)  021214                   L$PROT::
 4208   021214  000000                     .WORD 0                 ;P-TBL OFFSET OF TSSR, THE TU80 CSR
 4209   021216  177777                     .WORD -1                ;P-TBL OFFSET OF MASS BUS UNIT #: -1 = NOT A MAS
 4210   021220  177777                     .WORD -1                ;P-TBL OFFSET OF DRIVE #: -1 = NONE, ONE DRIVE P
 4211   021222                           ENDPROT
```

```
                                        .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 BOUN
                                                                                                BIT     #BIT0!BI
                                                                                                BEQ     50367$
                                                ERRSF 1,CMDPKM                  ;PRINT ERROR MSG,
                                                                                                TRAP    C$ERSF
                                                                                                .WORD   1
                                                                                                .WORD   CMDPKM
                                                                                                .WORD   0
                                                DELAY 20.                       ;GO TO SUPERVISOR, WAIT 2 SECONDS.
                                                                                                MOV     #20.,(PC
                                                                                                .WORD   0
                                                                                                MOV     L$DLY,(P
                                                                                                .WORD   0
                                                                                                DEC     -6(PC)
                                                                                                BNE     .-4
                                                                                                DEC     -22(PC)
                                                                                                BNE     .-20
                                                  BR    INIT10                  ;
                                                ENDIF
                                                                                        50367$:

                                        IFB CLRFLG NE #0 THEN           ;IF CLR COUNTERS FLAG SET:
                                                                                                TSTB    CLRFLG
                                                                                                BEQ     50370$
                                          CLRB  CLRFLG
                                          LET R2 := #0                  ;INIT CLR FLAG.
                                                                                                CLR     R2
                                          WHILE R2 NE #CNTLEN DO
                                                                                        50371$:
                                                                                                CMP     R2,#CNTL
                                                                                                BEQ     50372$
                                            LET WRBC(R2) := #0          ;CLR ALL STATISTICAL COUNTERS.
                                                                                                CLR     WRBC(R2)
                                            LET R2 := R2 + #2
                                                                                                ADD     #2,R2
                                          ENDDO
                                                                                                BR      50371$
                                                                                        50372$:
                                        ENDIF
                                                                                        50370$:

                                        IFB RRANV NE #0 THEN            ;IF RESET RANDOM VARIABLE FLAG IS SET TH
                                                                                                TSTB    RRANV
                                                                                                BEQ     50373$
                                          LET RANB := #RANBC              ;RESET RANDOM BASE #.
                                                                                                MOV     #RANBC,R
```

```
4248  021344
 (4)  021344  012737  032561  003444
4249  021352
 (4)  021352
4250  021352
 (3)  021352  012700  000040
 (3)  021356  104447
4251  021360
 (2)  021360  103030
4252  021362
 (6)  021362  105237  003536
4253  021366
 (4)  021366  012705  000006
4254  021372
 (4)  021372  105037  003341
4255  021376
 (3)  021376
4256  021376
 (4)  021376  012765  177774  002536
4257  021404
 (6)  021404  162705  000002
4258  021410
 (3)  021410  005705
 (6)  021412  001371
4259  021414
 (4)  021414  013705  002012
 (7)  021420  006305
4260  021422
 (3)  021422
4261  021422
 (6)  021422  162705  000002
4262  021426
 (4)  021426  010565  002536
 (7)  021432  006265  002536
4263  021436
 (3)  021436  005705
 (6)  021440  001370
4264
4265  021442
 (3)  021442  012700  000034
 (3)  021446  104447
4266  021450
 (2)  021450  103004
4267  021452
 (6)  021452  105237  003536
4268  021456
 (6)  021456  105237  003537
4269
4270  021462
 (3)  021462  104421
 (3)  021464  010037  003542
4271  021470
 (4)  021470  005003
4272  021472
 (6)  021472  105737  003537
 (9)  021476  001020
```

```
        LET RANS := #RANSC                  ;RESET RANDOM SAVE LOCATION.
                                                    MOV     #RANSC,R
ENDIF
                                                50373$:
READEF  #EF.START                   ;READ START COMMAND EVENT FLAG.
                                                    MOV     #EF.STAR
                                                    TRAP    C$REFG
BNCOMPLETE      INIT15              ;BRANCH IF NOT STARTING.
                                                    BCC     INIT15
LET STAFLG :B= STAFLG + #1          ;SET START COMMAND FLAG.
                                                    INCB    STAFLG
LET R5 := #6
                                                    MOV     #6,R5
LET HERE :B= #0                     ;CLEAR THE 'BEEN HERE BEFORE'' FLAG
                                                    CLRB    HERE
REPEAT                              ;INITIATE UNIT NUMBER TABLE
                                                50374$:
  LET DEVTBL(R5) := #NINUSE         ;BY STORING NOT IN USE IN EACH LOCATION.
                                                    MOV     #NINUSE,
  LET R5 := R5 - #2
                                                    SUB     #2,R5
UNTIL R5 EQ #0
                                                    TST     R5
                                                    BNE     50374$
LET R5 := L$UNIT SHIFT 1
                                                    MOV     L$UNIT,R
                                                    ASL     R5
REPEAT                              ;STORE ALL UNIT
                                                50375$:
  LET R5 := R5 - #2                 ;NUMBERS IN DEVTBL.
                                                    SUB     #2,R5
  LET DEVTBL(R5) := R5 SHIFT -1
                                                    MOV     R5,DEVTB
                                                    ASR     DEVTBL(R
UNTIL R5 EQ #0
                                                    TST     R5
                                                    BN      50375$
INIT15: READEF #EF.PWR              ;HAS THERE BE A POWER FAILURE?
                                                    MOV     #EF.PWR,
                                                    TRAP    C$REFG
        BNCOMPLETE INIT16           ;BRANCH IF NOT.
                                                    BCC     INIT16
  LET STAFLG :B= STAFLG + #1          ;IF SO - SET THE START FLAG.
                                                    INCB    STAFLG
  LET PWRFLG :B= PWRFLG + #1        ;IF SO - SET THE POWER FAIL FLAG.
                                                    INCB    PWRFLG
INIT16: RFLAGS  OPFLAG              ;READ AND STORE FLAGS SET BY OPERATOR
                                                    TRAP    C$RFLA
                                                    MOV     R0,OPFLA
        LET R3 := #0                ;CLEAR EVENT FLAG
                                                    CLR     R3
            IFB PWRFLG EQ #0 THEN   ;IF POWER FAIL HAS NOT OCCURRED THEN:
                                                    TSTB    PWRFLG
                                                    BNE     50376$
```

```
4273  021500
 (3)  021500  012700  000035          READEF #EF.NEW            ;UPDATE PASS COUNT WHEN
 (3)  021504  104447                                                    MOV     #EF.NEW,
                                                                        TRAP    C$REFG
4274  021506
 (6)  021506  103014                  IFCOND CS THEN            ;SUPERVISOR IS IN NEW PASS
                                                                        BCC     50377$
4275  021510
 (6)  021510  105737  003536            IFB STAFLG EQ #0 THEN   ;AND DIAG WAS NEITHER STARTED
 (9)  021514  001010                                                    TSTB    STAFLG
                                                                        BNE     50400$
4276  021516
 (3)  021516  012700  000037              READEF #EF.RES        ;NOR
 (3)  021522  104447                                                    MOV     #EF.RES,
                                                                        TRAP    C$REFG
4277  021524
 (6)  021524  103402                      IFCOND CC THEN        ;RESTARTED
                                                                        BCS     50401$
4278  021526
 (6)  021526  005103                        LET R3 := COMP R3   ;DO IT
                                                                        COM     R3
4279  021530
 (4)  021530  000401                      ELSE
 (3)  021532                                                            BR      50402$
                                                                50401$:
4280  021532
 (6)  021532  005203                        LET R3 := R3 + #1   ;SET 1ST PASS IF NEW PASS AND
                                                                        INC     R3
4281  021534                              ENDIF                 ;RESTARTING
 (4)  021534                                                    50402$:
4282  021534
 (4)  021534  000401                      ELSE
 (3)  021536                                                            BR      50403$
                                                                50400$:
4283  021536
 (6)  021536  005203                        LET R3 := R3 + #1   ;SET 1ST PASS IF NEW PASS AND
                                                                        INC     R3
4284  021540                              ENDIF                 ;STARTING
 (4)  021540                                                    50403$:
4285  021540                          ENDIF                     ;DO NOT UPDATE IT ON CONTINUE
 (4)  021540                                                    50377$:
4286  021540                        ENDIF                       ;OR ON POWER FAIL
 (4)  021540                                                    50376$:
4287  021540  004737  016614  JSR     PC,FIRSTU                 ;INIT DEVICE POINTER.
4288  021544                  LET R2 := #0                      ;INIT DEVICE COUNTER.
 (4)  021544  005002                                                    CLR     R2
4289  021546                  WHILE DEVTBL(R5) NE #END DO
 (4)  021546                                                    50404$:
 (6)  021546  026527  002536  177777                                    CMP     DEVTBL(R
 (9)  021554  001450                                                    BEQ     50405$
4290  021556                    LET R2 := R2 + #1
 (6)  021556  005202                                                    INC     R2
4291  021560                    LET R0 := R5 SHIFT -1
 (4)  021560  010500                                                    MOV     R5,R0
 (7)  021562  006200                                                    ASR     R0
4292  021564                    GPHARD  R0,R0                   ;GET HARDWARE P TABLE FROM SUPER.
 (3)  021564  104442                                                    TRAP    C$GPHRD
4293  021566                    IFCOND CS THEN
 (6)  021566  103036                                                    BCC     50406$
4294  021570                      LET TSSR(R5) := (R0)          ;SAVE TSSR ADDRESS.
 (4)  021570  011065  002466                                            MOV     (R0),TSS
4295  021574                      LET TSDB(R5) := (R0)+ - #2    ;SAVE TSDB ADDRESS.
 (4)  021574  012065  002456                                            MOV     (R0)+,TS
 (6)  021600  162765  000002  002456                                    SUB     #2,TSDB(
4296  021606                      LET TSVCT(R5) := (R0)         ;SAVE INTERRUPT VECTOR ADDRESS.
 (4)  021606  011065  002476                                            MOV     (R0),TSV
```

```
4297  021612
 (7)  021612  012746  000340                SETVEC TSVCT(R5),TS4INT(R5),#INTPRI ;SET UP INTERUPT PROCESSING COND
 (6)  021616  016546  002516                                                        MOV     #INTPRI,
 (5)  021622  016546  002476                                                        MOV     TS4INT(R
 (4)  021626  012746  000003                                                        MOV     TSVCT(R5
 (3)  021632  104437                                                                MOV     #3,-(SP)
 (2)  021634  062706  000010                                                        TRAP    C$SVEC
4298  021640                                                                       ADD     #10,SP
 (6)  021640  005703                      IF R3 NE #0 THEN           ;ACTUAL PASSCOUNT UPDATE PER R3
 (9)  021642  001410                                                                TST     R3
4299  021644                                                                       BEQ     50407$
 (6)  021644  005703                        IF R3 LT #0 THEN
 (9)  021646  002003                                                                TST     R3
4300  021650                                                                       BGE     50410$
 (6)  021650  005265  003260                  LET PASCNT(R5) := PASCNT(R5) + #1
4301  021654                                                                       INC     PASCNT(R
 (4)  021654  000403                        ELSE
 (3)  021656                                                                       BR      50411$
4302  021656                                                                50410$:
 (4)  021656  012765  000001  003260          LET PASCNT(R5) := #1
4303  021664                                                                       MOV     #1,PASCN
 (4)  021664                                ENDIF
4304  021664                                                                50411$:
 (4)  021664                              ENDIF
4305  021664                                                                50407$:
 (4)  021664                            ENDIF
4306  021664                                                                50406$:
 (4)  021664  005065  003330              LET RECCNT(R5) := #0          ;CLEAR RECORD COUNT
4307  021670  004737  016662                                                        CLR     RECCNT(R
4308  021674                            JSR   PC,NEXTU                ;DO IT FOR ALL DEVICES.
 (4)  021674  000724                  ENDDO
 (3)  021676                                                                       BR      50404$
4309                                                                          50405$:
4310  021676                      IF R2 EQ #0 THEN                ;IF THERE ARE NO UNITS:
 (6)  021676  005702                                                               TST     R2
 (9)  021700  001026                                                               BNE     50412$
4311  021702                        PRINTF #AUDRPM                ;PRINT ALL UNITS DROPPED,
 (7)  021702  012746  004731                                                       MOV     #AUDRPM,
 (6)  021706  012746  000001                                                       MOV     #1,-(SP)
 (3)  021712  010600                                                               MOV     SP,R0
 (4)  021714  104417                                                               TRAP    C$PNTF
 (4)  021716  062706  000004                                                       ADD     #4,SP
4312  021722                        DELAY 20.                     ;GO TO SUPERVISOR, WAIT 2 SECONDS.
 (2)  021722  012727  000024                                                       MOV     #20.,(PC
 (2)  021726  000000                                                               .WORD   0
 (2)  021730  013727  002116                                                       MOV     L$DLY,(P
 (2)  021734  000000                                                               .WORD   0
 (2)  021736  005367  177772                                                       DEC     -6(PC)
 (2)  021742  001375                                                               BNE     .-4
 (2)  021744  005367  177756                                                       DEC     -22(PC)
 (2)  021750  001367                                                               BNE     .-20
4313  021752                        BREAK                         ;GO TO SUPERVISOR, CHECK TTY.
 (3)  021752  104422                                                               TRAP    C$BRK
4314  021754                      DOCLN                           ;DO CLEAN CODE + ABORT PASS.
 (3)  021754  104444                                                               TRAP    C$DCLN
4315  021756                      ENDIF
```

```
  (4)  021756                                                                   50412$:
4316
4317   021756                                  SETPRI #PRI00                 ;LOWER CPU PRIORITY TO 0
  (3)  021756  012700  000000                                                        MOV     #PRI00,R
  (3)  021762  104441                                                                TRAP    C$SPRI
4318   021764                                  IFB IREC EQ #0 AND #ADR NOTSETIN OPFLAG THEN    ;IF ERROR RECOVERY IS EN
  (6)  021764  105737  002210                                                        TSTB    IREC
  (9)  021770  001145                                                                BNE     50413$
  (6)  021772  032737  000020  003542                                                BIT     #ADR,OPF
  (9)  022000  001141                                                                BNE     50413$
4319   022002  004737  016614                    JSR PC,FIRSTU               ;AND AUTO-DROP NOT CALLED, THEN SET UP F
4320   022006                                    WHILE DEVTBL(R5) NE #END DO  ;WHILE THERE ARE MORE DEVICES:
  (4)  022006                                                                   50414$:
  (6)  022006  026527  002536  177777                                                CMP     DEVTBL(R
  (9)  022014  001533                                                                BEQ     50415$
4321   022016                                      BEGIN COUNTER             ;START 3.5 MINUTE COUNTER
4322   022016                                        INCR TIME1 FROM #1 TO #25 BY #1
  (4)  022016  012737  000001  003446                                                MOV     #1,TIME1
  (5)  022024  000402                                                                BR      50417$
  (4)  022026                                                                   50420$:
  (7)  022026  005237  003446                                                        INC     TIME1
  (5)  022032                                                                   50417$:
  (5)  022032  023727  003446  000025                                                CMP     TIME1,#2
  (7)  022040  003106                                                                BGT     50421$
4323   022042                                          LET @TSDB(R5) := #GSCPK ;AND GET UNITS STATUS
  (4)  022042  012775  002324  002456                                                MOV     #GSCPK,@
4324   022050                                          DELAY 32.             ;WAIT 3.2 MSEC.
  (2)  022050  012727  000040                                                        MOV     #32.,(PC
  (2)  022054  000000                                                                .WORD   0
  (2)  022056  013727  002116                                                        MOV     L$DLY,(P
  (2)  022062  000000                                                                .WORD   0
  (2)  022064  005367  177772                                                        DEC     -6(PC)
  (2)  022070  001375                                                                BNE     .-4
  (2)  022072  005367  177756                                                        DEC     -22(PC)
  (2)  022076  001367                                                                BNE     .-20
4325   022100                                          IF #TS.SSR SETIN @TSSR(R5) THEN
  (6)  022100  032775  000200  002466                                                BIT     #TS.SSR,
  (9)  022106  001420                                                                BEQ     50422$
4326   022110                                            IF #TS.OFL NOTSETIN @TSSR(R5) THEN
  (6)  022110  032775  000100  002466                                                BIT     #TS.OFL,
  (9)  022116  001001                                                                BNE     50423$
4327   022120                                              LEAVE COUNTER         ;EXIT COUNTER WHEN UNIT ON LINE
  (4)  022120  000456                                                                BR      50416$
4328   022122                                            ELSE
  (3)  022122                                                                   50423$:
4329   022122                                              PRINTF #OFLINM,DEVTBL(R5) ;PRINT UNIT OFF LINE EVERY 10 SEC
  (8)  022122  016546  002536                                                        MOV     DEVTBL(R
  (7)  022126  012746  005213                                                        MOV     #OFLINM,
  (6)  022132  012746  000002                                                        MOV     #2,-(SP)
  (3)  022136  010600                                                                MOV     SP,R0
  (4)  022140  104417                                                                TRAP    C$PNTF
  (4)  022142  062706  000006                                                        ADD     #6,SP
4330   022146                                            ENDIF
  (4)  022146                                                                   50424$:
4331   022146                                          ELSE
  (4)  022146  000412                                                                BR      50425$
```

```
  (3)  022150
 4332  022150
  (8)  022150  016546  002536
  (7)  022154  012746  023102
  (6)  022160  012746  000002
  (3)  022164  010600
  (4)  022166  104417
  (4)  022170  062706  000006
 4333  022174
  (4)  022174
 4334  022174
  (4)  022174  012737  000001  003450
  (5)  022202  000402
  (4)  022204
  (7)  022204  005237  003450
  (5)  022210
  (5)  022210  023727  003450  000013
  (7)  022216  003016
 4335  022220
  (2)  022220  012727  000310
  (2)  022224  000000
  (2)  022226  013727  002116
  (2)  022232  000000
  (2)  022234  005367  177772
  (2)  022240  001375
  (2)  022242  005367  177756
  (2)  022246  001367
 4336  022250
  (3)  022250  104422
 4337  022252
  (4)  022252  000754
  (3)  022254
 4338  022254
  (4)  022254  000664
  (3)  022256
 4339  022256
  (3)  022256
 4340  022256
  (6)  022256  023727  003446  000025
  (9)  022264  003404
 4341  022266  004737  011552
 4342  022272  004737  012274
 4343  022276
  (4)  022276
 4344
 4345  022276  004737  016662
 4346  022302
  (4)  022302  000641
  (3)  022304
 4347  022304
  (4)  022304
 4348  022304
  (6)  022304  105737  003537
  (9)  022310  001026
 4349  022312
  (3)  022312  104431
```

```
                PRINTF #NRDYM,DEVTBL(R5)

                ENDIF
                INCR TIME2 FROM #1 TO #13 BY #1

                  DELAY 200.            ;WAIT FOR UNIT TO BE SET ON-LINE

                  BREAK          ;ALLOW TERMINAL INTERRUPT

                ENDINC

              ENDINC

                END COUNTER

                IF TIME1 GT #25 THEN        ;IF OFF LINE FOR 3.5 MINUTES

                  JSR PC,MOVMSG             ;GET MESSAGE PACKET
                  JSR PC,TCC1               ;PRINT ERROR AND DROP OFF LINE UNIT
                ENDIF

                  JSR PC,NEXTU              ;SET UP FOR NEXT UNIT.
              ENDDO

            ENDIF
            IFB PWRFLG EQ #0 THEN

              MEMORY DATAWT                 ;REQUEST MEMORY FROM SUPER FOR RD/WR BUF
```

```
50422$:
                MOV     DEVTBL(R
                MOV     #NRDYM,-
                MOV     #2,-(SP)
                MOV     SP,R0
                TRAP    C$PNTF
                ADD     #6,SP
50425$:
                MOV     #1,TIME2
                BR      50426$
50427$:
                INC     TIME2
50426$:
                CMP     TIME2,#1
                BGT     50430$
                MOV     #200.,(P
                .WORD   0
                MOV     L$DLY,(P
                .WORD   0
                DEC     -6(PC)
                BNE     .-4
                DEC     -22(PC)
                BNE     .-20
                TRAP    C$BRK
                BR      50427$
50430$:
                BR      50420$
50421$:
50416$:
                CMP     TIME1,#2
                BLE     50431$
50431$:
                BR      50414$
50415$:
50413$:
                TSTB    PWRFLG
                BNE     50432$
                TRAP    C$MEM
```

```
  (3)  022314  010037  003416                                                                MOV     RO,DATAW
 4350  022320                                  LET DATARD := DATAWT + #DATCNT        ;SET RD BFR AD
  (4)  022320  013737  003416  003420                                                        MOV     DATAWT,D
  (6)  022326  062737  004000  003420                                                        ADD     #DATCNT,
                                               IF @DATAWT LT #DATCNT THEN    ;WHEN NOT ENOUGH FREE MEMO AVAILABLE
 4351  022334
  (6)  022334  027727  161056  004000                                                        CMP     @DATAWT,
  (9)  022342  002011                                                                        BGE     50433$
 4352  022344                                    PRINTF #MEMOM                ;WARN OPERATOR
  (7)  022344  012746  022412                                                                MOV     #MEMOM,-
  (6)  022350  012746  000001                                                                MOV     #1,-(SP)
  (3)  022354  010600                                                                        MOV     SP,RO
  (4)  022356  104417                                                                        TRAP    C$PNTF
  (4)  022360  062706  000004                                                                ADD     #4,SP
 4353  022364                                    DOCLN                        ;AND ABORT PASS
  (3)  022364  104444                                                                        TRAP    C$DCLN
 4354  022366                                  ENDIF         ;DIAG MUST BE RE-LOADED IN A CPU WITH LARGER MEMO
  (4)  022366                                                                         50433$:
 4355  022366                                ENDIF
  (4)  022366                                                                         50432$:
 4356
 4357  022366                                LET CHGFLG :B= #0              ;CLR CHANGE CMD SEQ TBL FLAG.
  (4)  022366  105037  002215                                                                CLRB    CHGFLG
 4358  022372                                LET R3 := #ENDFLG
  (4)  022372  012703  003536                                                                MOV     #ENDFLG,
 4359  022376  004737  011502                JSR     PC,CLRERR              ;CLEAR ALL FLAGS.
 4360  022402                                LET PWRFLG :B= #0              ;CLEAR THE POWER FAIL FLAG.
  (4)  022402  105037  003537                                                                CLRB    PWRFLG
 4361
 4362  022406                                EXIT    INIT
  (3)  022406  104432                                                                        TRAP    C$EXIT
  (3)  022410  000104                                                                        .WORD   L10012-.
 4374  022412  040445  051106  042505   MEMOM:  .ASCII  /%AFREE MEMO TOO SMALL FOR RD-WR BFRS%N/
       022420  046440  046505  020117
       022426  047524  020117  046523
       022434  046101  020114  047506
       022442  020122  042122  053455
       022450  020122  043102  051522
       022456  047045
 4375  022460  040445  042522  046055           .ASCIZ  /%ARE-LOAD IN LARGER MEMO%N/
       022466  040517  020104  047111
       022474  046040  051101  042507
       022502  020122  042515  047515
       022510  047045     000
 4376          022514                           .EVEN
 4377
 4378  022514                                   ENDINIT
  (3)  022514                           L10012:
  (3)  022514  104411                                                                        TRAP    C$INIT
```

```
                                       .SBTTL  AUTO DROP SECTION

                                       ;++
                                       ;SECTION EXECUTED AFTER THE INIT CODE WHEN "ADR" FLAG IS SET BY OPERATOR
                                       ;SECTION CHEKS FOR A VALID INTERFACE LOCATION.  DROPS UNIT IF NO RESPONSE
                                       ;FROM INTERFACE
                                       ;--

      022516                                    BGNAUTO
 (3)  022516                           L$AUTO::

      022516  004737  016614                    JSR PC,FIRSTU                 ;FIND FIRST UNIT
      022522                                    WHILE DEVTBL(R5) NE #END DO   ;
 (4)  022522                                                                               50434$:
 (6)  022522  026527  002536  177777                                                               CMP     DEVTBL(R
 (9)  022530  001525                                                                               BEQ     50435$
      022532                                      LET TRAPD4 :B= #0           ;
 (4)  022532  105037  003540                                                                       CLRB    TRAPD4
      022536                                      SETVEC #4,#TRAP4,#PRI07     ;SET VECTOR 4
 (7)  022536  012746  000340                                                                       MOV     #PRI07,-
 (6)  022542  012746  023132                                                                       MOV     #TRAP4,-
 (5)  022546  012746  000004                                                                       MOV     #4,-(SP)
 (4)  022552  012746  000003                                                                       MOV     #3,-(SP)
 (3)  022556  104437                                                                               TRAP    C$SVEC
 (2)  022560  062706  000010                                                                       ADD     #10,SP
      022564                                      LET R2 := @TSSR(R5)         ;ADDRESS TU80 INTERFACE
 (4)  022564  017502  002466                                                                       MOV     @TSSR(R5
      022570                                      CLRVEC #4                   ;CLEAR VECTOR AT 4
 (3)  022570  012700  000004                                                                       MOV     #4,R0
 (3)  022574  104436                                                                               TRAP    C$CVEC
      022576                                      IFB TRAPD4 NE #0 THEN
 (6)  022576  105737  003540                                                                       TSTB    TRAPD4
 (9)  022602  001423                                                                               BEQ     50436$
      022604                                        LET FTLCNT(R5) := FTLCNT(R5) + #1
 (6)  022604  005265  003320                                                                       INC     FTLCNT(R
      022610                                        PRINTF #AUTODM,TSSR(R5)   ;PRINT ERROR
 (8)  022610  016546  002466                                                                       MOV     TSSR(R5)
 (7)  022614  012746  023006                                                                       MOV     #AUTODM,
 (6)  022620  012746  000002                                                                       MOV     #2,-(SP)
 (3)  022624  010600                                                                               MOV     SP,R0
 (4)  022626  104417                                                                               TRAP    C$PNTF
 (4)  022630  062706  000006                                                                       ADD     #6,SP
      022634                                        LET DROPN := DEVTBL(R5)           ;SAVE # OF UNIT TO BE DROPPED.
 (4)  022634  016537  002536  017136                                                               MOV     DEVTBL(R
      022642                                        LET R0 := R5 SHIFT -1             ;R0=LOGICAL DEVICE NUMBER
 (4)  022642  010500                                                                               MOV     R5,R0
 (7)  022644  006200                                                                               ASR     R0
      022646                                        DODU R0                   ;DROP THE UNIT: EXEC BGNDU-ENDDU CODE IF
 (3)  022646  104451                                                                               TRAP    C$DODU
      022650                                      ELSE
 (4)  022650  000452                                                                               BR      50437$
 (3)  022652                                                                               50436$:
      022652                                        LET @TSDB(R5) := #GSCPK   ;SEND GET STATUS COMMAND
 (4)  022652  012775  002324  002456                                                               MOV     #GSCPK,@
      022660  004737  011516                        JSR PC,WSSR               ;WAIT
      022664                                        IF #TS.SSR SETIN @TSSR(R5) THEN
```

```
   (6)  022664  032775  000200  002466
   (9)  022672  001423
 4406   022674
   (6)  022674  032775  000100  002466
   (9)  022702  001416
 4407   022704
   (6)  022704  005265  003320
 4408   022710
   (8)  022710  016546  002536
   (7)  022714  012746  005213
   (6)  022720  012746  000002
   (3)  022724  010600
   (4)  022726  104417
   (4)  022730  062706  000006
 4409   022734  004737  017046
 4410   022740
   (4)  022740
 4411   022740
   (4)  022740  000416
   (3)  022742
 4412   022742
   (6)  022742  005265  003320
 4413   022746
   (8)  022746  016546  002536
   (7)  022752  012746  023102
   (6)  022756  012746  000002
   (3)  022762  010600
   (4)  022764  104417
   (4)  022766  062706  000006
 4414   022772  004737  017046
 4415   022776
   (4)  022776
 4416   022776
   (4)  022776
 4417   022776  004737  016662
 4418   023002
   (4)  023002  000647
   (3)  023004
 4419
 4420   023004
   (3)  023004
   (3)  023004  104461
 4421
 4422   023006  040445  052502  020123
        023014  051124  050101  040440
        023022  020124  047445  022466
        023030     116
 4423   023031     045  044501  052116
        023036  051105  040506  042503
        023044  041040  042101  047440
        023052  020122  047516  020124
        023060  042523  020124  047524
        023066  040440  047502  042526
        023074  040440  022504  000116
 4424   023102  040445  047125  052111
        023110  022440  030504  040445
```

```
                    IF #TS.OFL SETIN @TSSR(R5) THEN

                      LET FTLCNT(R5) := FTLCNT(R5) + #1

                      PRINTF #OFLINM,DEVTBL(R5)

                        JSR PC,DROPUA
                      ENDIF

                    ELSE

                      LET FTLCNT(R5) := FTLCNT(R5) + #1

                      PRINTF #NRDYM,DEVTBL(R5)

                        JSR PC,DROPUA
                      ENDIF

                    ENDIF

                      ENDIF

                      JSR PC,NEXTU
                    ENDDO

                    ENDAUTO
L10013:

AUTODM: .ASCII  /%ABUS TRAP AT %O6%N/

        .ASCIZ  /%AINTERFACE BAD OR NOT SET TO ABOVE AD%N/

NRDYM:  .ASCIZ  /%AUNIT %D1%A NOT RDY%N/
```



```
                    BIT     #TS.SSR,
                    BEQ     50440$

                    BIT     #TS.OFL,
                    BEQ     50441$

                    INC     FTLCNT(R

                    MOV     DEVTBL(R
                    MOV     #OFLINM,
                    MOV     #2,-(SP)
                    MOV     SP,R0
                    TRAP    C$PNTF
                    ADD     #6,SP

50441$:

                    BR      50442$
50440$:

                    INC     FTLCNT(R

                    MOV     DEVTBL(R
                    MOV     #NRDYM,-
                    MOV     #2,-(SP)
                    MOV     SP,R0
                    TRAP    C$PNTF
                    ADD     #6,SP

50442$:

50437$:

                BR      50434$
50435$:

                TRAP    C$AUTO
```

```
        023116  047040  052117  051040
        023124  054504  047045     000
 4425           023132                          .EVEN
 4426
 4427                                   ;       DEVICE BUS TRAP HANDLER
 4428                                   ;       OUTPUT: TRAPD4 BYTE  1: TRAPED AT 4
 4429                                   ;                            0: NO TRAP
 4430
 4431   023132                          TRAP4:: LET TRAPD4 :B= TRAPD4 + #1
  (6)   023132  105237  003540                                                        INCB    TRAPD4
 4432   023136  000002                          RTI
 4433
 4434
 4435
```

```
                                        .SBTTL  CLEANUP CODING SECTION

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

                                                BGNCLN
                                        L$CLEAN::

                                                JSR     PC,FIRSTU               ;FIND FIRST UNIT.
                                                WHILE DEVTBL(R5) NE #END DO
                                                                                        50443$:
                                                                                                CMP     DEVTBL(R
                                                                                                BEQ     50444$
                                                  JSR PC,WSSR                   ;WAIT FOR UNIT READY OR TIMEOUT,
                                                  CLRVEC        TSVCT(R5)       ;RELEASE INTERRUPT VECTORS FOR ALL DEV.
                                                                                                MOV     TSVCT(R5
                                                                                                TRAP    C$CVEC
                                                  JSR   PC,NEXTU                ;FIND NEXT UNIT.
                                                ENDDO
                                                                                                BR      50443$
                                                                                        50444$:

                                                EXIT    CLN
                                                                                                TRAP    C$EXIT
                                                                                                .WORD   L10014-.

                                                .EVEN

                                                ENDCLN
                                        L10014: ENDCLN
                                                                                                TRAP    C$CLEAN
```

```
                                        .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
                                        ;--

4484  023202                                    BGNDU
 (3)  023202                            L$DU::
4485
4491  023202                                    LET R5 := R0 SHIFT 1          ;R5 = LOGICAL DEVICE NUMBER X 2.
 (4)  023202  010005                                                                MOV     R0,R5
 (7)  023204  006305                                                                ASL     R5
4492  023206                                    LET DEVTBL(R5) := #NINUSE     ;SET NOT IN USE FLAG FOR THE DEVICE.
 (4)  023206  012765  177774  002536                                               MOV     #NINUSE,
4493  023214                                    PRINTF  #DROPDM,DROPN         ;PRINT DROP DEVICE MESSAGE
 (8)  023214  013746  017136                                                        MOV     DROPN,-(
 (7)  023220  012746  004702                                                        MOV     #DROPDM,
 (6)  023224  012746  000002                                                        MOV     #2,-(SP)
 (3)  023230  010600                                                                MOV     SP,R0
 (4)  023232  104417                                                                TRAP    C$PNTF
 (4)  023234  062706  000006                                                        ADD     #6,SP
4494  023240                                    EXIT    DU
 (4)  023240  000167                                                                .WORD   J$JMP
 (3)  023242  000000                                                                .WORD   L10015-2
4506
4507
4508  023244                                    .EVEN
 (3)  023244                                    ENDDU
 (3)  023244  104453                    L10015:                                     TRAP    C$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.
                                        ;--

 4511                                           BGNAU
 4512                                   L$AU::
 4513
 4514                                           LET R5 := R0 SHIFT 1            ;R5 = LOGICAL DEVICE NUMBER X 2.
 4515                                                                                   MOV     R0,R5
 4516                                                                                   ASL     R5
 4517                                           LET DEVTBL(R5) := R0    ;STORE UNIT # IN DEVICE TABLE.
 4518                                                                                   MOV     R0,DEVTB
 4519  023246                                   GPHARD  R0,R0                   ;GET HARDWARE P TABLE FROM SUPER.
  (3)  023246                                                                          TRAP    C$GPHRD
 4520                                           LET TSSR(R5) := (R0)            ;SAVE TSSR ADDRESS.
 4526                                                                                   MOV     (R0),TSS
 4527  023246                                   LET TSDB(R5) := (R0)+ - #2      ;SAVE TSDB ADDRESS.
  (4)  023246  0100C5                                                                   MOV     (R0)+,TS
  (7)  023250  006305                                                                   SUB     #2,TSDB(
 4528  023252                                   LET TSVCT(R5) := (R0)           ;SAVE INTERRUPT VECTOR ADDRESS.
  (4)  023252  010065  002536                                                           MOV     (R0),TSV
 4529  023256                                   SETVEC TSVCT(R5),TS4INT(R5),#INTPRI     ;SET UP INTERUPT PROCESSING COND
  (3)  023256  104442                                                                   MOV     #INTPRI,
 4530  023260                                                                          MOV     TS4INT(R
  (4)  023260  011065  002466                                                           MOV     TSVCT(R5
 4531  023264                                                                          MOV     #3,-(SP)
  (4)  023264  012065  002456                                                           TRAP    C$SVEC
  (6)  023270  162765  000002  002456                                                   ADD     #10,SP
 4532  023276
  (4)  023276  011065  002476                   LET INTFLG(R5) := #0             ;CLEAR INTERRUPT FLAGS.
 4533  023302                                                                          CLR     INTFLG(R
  (7)  023302  012746  000340
  (6)  023306  016546  002516                   EXIT    AU
  (5)  023312  016546  002476                                                           .WORD   J$JMP
  (4)  023316  012746  000003                                                           .WORD   L10016-2
  (3)  023322  104437
  (2)  023324  062706  000010
 4534  023330
  (4)  023330  005065  003500                   .EVEN
 4535
 4536  023334                                   ENDAU
  (4)  023334  000167                   L10016:
  (3)  023336  000000                                                                   TRAP    C$AU
 4548
 4549                                           ENDMOD
 4550
 4551  023340
  (3)  023340
  (3)  023340  104452
 4552
 4553  023342
 4554
```

```
                                        .TITLE HARDWARE TESTS

                                        .SBTTL  TEST 1:  BASIC FUNCTIONS.

                                        ;++
                                        ; TEST TO EXECUTE ALL TU80 FUNCTIONS.
                                        ;--

     023342                                     BGNMOD

     023342                                     BGNTST
(3)  023342                             T1::

     023342                                     LET RANDOM :B= #0               ;CLR THE RANDOM OPERATIONS FLAG.
(4)  023342  105037  003523                                                                    CLRB    RANDOM
     023346                                     LET EXPBOT :B= #0               ;CLR EXPECT BOT FLAG.
(4)  023346  105037  003522                                                                    CLRB    EXPBOT

     023352                                     BGNSUB                          ;SUBTEST 1 - SET CHAR, DRIVE INIT, GET S
(3)  023352                             T1.1:
(3)  023352  104402                                                                            TRAP    C$BSUB

     023354                                     LET R2 := #BFSEQ0               ;ADR OF CMD SEQ.
(4)  023354  012702  024200                                                                    MOV     #BFSEQ0,
     023360  004737  024154                     JSR     PC,BFSEQ                ;SET UP CMD SEQ.
     023364  004737  006760                     JSR     PC,EXALL                ;EXECUTE CMD SEQ ON ALL DEVICES.
     023370  004737  016614                     JSR     PC,FIRSTU               ;FIND THE FIRST UNIT.
     023374                                     WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE MORE DEVICES:
(4)  023374                                                                          50445$:
(6)  023374  026527  002536  177777                                                            CMP     DEVTBL(R
(9)  023402  001434                                                                            BEQ     50446$
     023404                                       LET R2 := MSGPKA(R5)          ;GET MSG PACKET ADR.
(4)  023404  016502  002506                                                                    MOV     MSGPKA(R
     023410                                       LET R2 := R2 + #12            ;GET XSTAT2 ADR.
(6)  023410  062702  000012                                                                    ADD     #12,R2
     023414                                       LET TS4CL(R5) := (R2) CLR.BY #177400  ;STORE CODE LEVEL FROM DTR BYTE.
(4)  023414  011265  002526                                                                    MOV     (R2),TS4
(6)  023420  042765  177400  002526                                                            BIC     #177400,
     023426                                       IF PASCNT(R5) EQ #1 THEN      ;IF THIS IS PASS 1 THEN:
(6)  023426  026527  003260  000001                                                            CMP     PASCNT(R
(9)  023434  001014                                                                            BNE     50447$
     023436                                         PRINTF #CODELM,DEVTBL(R5),TS4CL(R5) ;PRINT THE TU80 MICROCODE LEVEL.
(9)  023436  016546  002526                                                                    MOV     TS4CL(R5
(8)  023442  016546  002536                                                                    MOV     DEVTBL(R
(7)  023446  012746  004056                                                                    MOV     #CODELM,
(6)  023452  012746  000003                                                                    MOV     #3,-(SP)
(3)  023456  010600                                                                            MOV     SP,R0
(4)  023460  104417                                                                            TRAP    C$PNTF
(4)  023462  062706  000010                                                                    ADD     #10,SP
     023466                                       ENDIF
(4)  023466                                                                          50447$:
     023466  004737  016662                       JSR   PC,NEXTU                ;FIND NEXT UNIT.
     023472                                     ENDDO
(4)  023472  000740                                                                            BR      50445$
(3)  023474                                                                          50446$:
```

```
4598  023474                                        ENDSUB
 (3)  023474                                L10020:
 (3)  023474  104403                                                    TRAP    C$ESUB
4599
4600  023476                                        BGNSUB                          ;SUBTEST 2 - REWIND.
 (3)  023476                                T1.2:
 (3)  023476  104402                                                    TRAP    C$BSUB
4601
4602  023500                                        LET R2 := #BFSEQ1               ;ADR OF CMD SEQ.
 (4)  023500  012702  024252                                           MOV     #BFSEQ1,
4603  023504  004737  024154                       JSR     PC,BFSEQ                ;SET UP CMD SEQ.
4604  023510  004737  006760                       JSR     PC,EXALL                ;EXECUTE CMD SEQ ON ALL DEVICES.
4605  023514                                        LET STAFLG :B= #0               ;CLEAR START FLAG
 (4)  023514  105037  003536                                           CLRB    STAFLG
4606  023520                                        ENDSUB
 (3)  023520                                L10021:
 (3)  023520  104403                                                    TRAP    C$ESUB
4607
4608  023522                                        BGNSUB                          ;SUBTEST 3 - WRITE/VERIFY.
 (3)  023522                                T1.3:
 (3)  023522  104402                                                    TRAP    C$BSUB
4609
4610  023524                                        LET R2 := #BFSEQ2               ;ADR OF CMD SEQ.
 (4)  023524  012702  024264                                           MOV     #BFSEQ2,
4611  023530  004737  024154                       JSR     PC,BFSEQ                ;SET UP CMD SEQ.
4612  023534  004737  006760                       JSR     PC,EXALL                ;EXECUTE CMD SEQ ON ALL DEVICES.
4613  023540                                        ENDSUB
 (3)  023540                                L10022:
 (3)  023540  104403                                                    TRAP    C$ESUB
4614
4615  023542                                        BGNSUB                          ;SUBTEST 4 - WRITE TAPE MARK, ERASE.
 (3)  023542                                T1.4:
 (3)  023542  104402                                                    TRAP    C$BSUB
4616
4617  023544                                        LET R2 := #BFSEQ3               ;ADR OF CMD SEQ.
 (4)  023544  012702  024356                                           MOV     #BFSEQ3,
4618  023550  004737  024154                       JSR     PC,BFSEQ                ;SET UP CMD SEQ.
4619  023554  004737  006760                       JSR     PC,EXALL                ;EXECUTE CMD SEQ ON ALL DEVICES.
4620  023560                                        ENDSUB
 (3)  023560                                L10023:
 (3)  023560  104403                                                    TRAP    C$ESUB
4621
4622  023562                                        BGNSUB                          ;SUBTEST 5 - SPACE FILES.
 (3)  023562                                T1.5:
 (3)  023562  104402                                                    TRAP    C$BSUB
4623
4624  023564                                        LET R2 := #BFSEQ4               ;ADR OF CMD SEQ.
 (4)  023564  012702  024430                                           MOV     #BFSEQ4,
4625  023570  004737  024154                       JSR     PC,BFSEQ                ;SET UP CMD SEQ.
4626  023574  004737  006760                       JSR     PC,EXALL                ;EXECUTE CMD SEQ ON ALL DEVICES.
4627  023600                                        ENDSUB
 (3)  023600                                L10024:
 (3)  023600  104403                                                    TRAP    C$ESUB
4628
4629  023602                                        BGNSUB                          ;SUBTEST 6 - SPACE RECORDS.
 (3)  023602                                T1.6:
```

```
 (3)  023602  104402                                                              TRAP    C$BSUB

      023604                         LET R2 := #BFSEQ5            ;ADR OF CMD SEQ.
 (4)  023604  012702  024472                                                      MOV     #BFSEQ5,
      023610  004737  024154         JSR     PC,BFSEQ             ;SET UP CMD SEQ.
      023614  004737  006760         JSR     PC,EXALL             ;EXECUTE CMD SEQ ON ALL DEVICES.
      023620                         ENDSUB
 (3)  023620                 L10025:
 (3)  023620  104403                                                              TRAP    C$ESUB

      023622                         BGNSUB                       ;SUBTEST 7 - WRITE RETRY.
 (3)  023622                 T1.7:
 (3)  023622  104402                                                              TRAP    C$BSUB

      023624                         LET R2 := #BFSEQ6            ;ADR OF CMD SEQ.
 (4)  023624  012702  024544                                                      MOV     #BFSEQ6,
      023630  004737  024154         JSR     PC,BFSEQ             ;SET UP CMD SEQ.
      023634  004737  006760         JSR     PC,EXALL             ;EXECUTE CMD SEQ ON ALL DEVICES.
      023640                         ENDSUB
 (3)  023640                 L10026:
 (3)  023640  104403                                                              TRAP    C$ESUB

      023642                         BGNSUB                       ;SUBTEST 8 - READ REV RETRY.
 (3)  023642                 T1.8:
 (3)  023642  104402                                                              TRAP    C$BSUB

      023644                         LET R2 := #BFSEQ7            ;ADR OF CMD SEQ.
 (4)  023644  012702  024576                                                      MOV     #BFSEQ7,
      023650  004737  024154         JSR     PC,BFSEQ             ;SET UP CMD SEQ.
      023654  004737  006760         JSR     PC,EXALL             ;EXECUTE CMD SEQ ON ALL DEVICES.
      023660                         ENDSUB
 (3)  023660                 L10027:
 (3)  023660  104403                                                              TRAP    C$ESUB

      023662                         BGNSUB                       ;SUBTEST 9 - READ FWD RETRY.
 (3)  023662                 T1.9:
 (3)  023662  104402                                                              TRAP    C$BSUB

      023664                         LET R2 := #BFSEQ8            ;ADR OF CMD SEQ.
 (4)  023664  012702  024630                                                      MOV     #BFSEQ8,
      023670  004737  024154         JSR     PC,BFSEQ             ;SET UP CMD SEQ.
      023674  004737  006760         JSR     PC,EXALL             ;EXECUTE CMD SEQ ON ALL DEVICES.
      023700                         ENDSUB
 (3)  023700                 L10030:
 (3)  023700  104403                                                              TRAP    C$ESUB

      023702                         BGNSUB                       ;SUBTEST 10- CLEAN.
 (3)  023702                 T1.10:
 (3)  023702  104402                                                              TRAP    C$BSUB

      023704                         LET R2 := #BFSEQ9            ;ADR OF CMD SEQ.
 (4)  023704  012702  024662                                                      MOV     #BFSEQ9,
      023710  004737  024154         JSR     PC,BFSEQ             ;SET UP CMD SEQ.
      023714  004737  006760         JSR     PC,EXALL             ;EXECUTE CMD SEQ ON ALL DEVICES.
      023720                         ENDSUB
 (3)  023720                 L10031:
```

```
  (3)  023720  104403                                                                       TRAP    C$ESUB
 4663
 4664  023722                                    BGNSUB                  ;SUBTEST 11 - WTV SWAPPED DATA BYTES.
  (3)  023722                           T1.11:
  (3)  023722  104402                                                                       TRAP    C$BSUB
 4665
 4666  023724                                    LET R2 := #BFSE10       ;ADR OF CMD SEQ.
  (4)  023724  012702  024704                                                               MOV     #BFSE10,
 4667  023730  004737  024154                    JSR     PC,BFSEQ        ;SET UP CMD SEQ.
 4668  023734  004737  006760                    JSR     PC,EXALL        ;WRITE/VERIFY RECORDS 1 AND 2.
 4669  023740                                    LET SWBFLG :B= #1       ;ENABLE BYTE SWAPPING.
  (4)  023740  112737  000001  003526                                                       MOVB    #1,SWBFL
 4670  023746  004737  006760                    JSR     PC,EXALL        ;WRITE/VERIFY RECORDS 3 AND 4.
 4671  023752                                    LET SWBFLG :B= #0       ;DISABLE BYTE SWAPPING.
  (4)  023752  105037  003526                                                               CLRB    SWBFLG
 4672  023756                                    ENDSUB
  (3)  023756                           L10032:
  (3)  023756  104403                                                                       TRAP    C$ESUB
 4673
 4674  023760                                    LET R2 := DATAWT + #10.         ;INIT WRITE BUFFER POINTER.
  (4)  023760  013702  003416                                                               MOV     DATAWT,R
  (6)  023764  062702  000012                                                               ADD     #10.,R2
 4675  023770                                    WHILE R2 NE DATAWT DO   ;UNTIL 10 BYTES HAVE BEEN SWAPPED.
  (4)  023770                                                                      50450$:
  (6)  023770  020237  003416                                                               CMP     R2,DATAW
  (9)  023774  001402                                                                       BEQ     50451$
 4676  023776  000342                              SWAB -(R2)            ;SWAP DATA BYTES IN WRITE BUFFER.
 4677  024000                                    ENDDO
  (4)  024000  000773                                                                       BR      50450$
  (3)  024002                                                                      50451$:
 4678  024002                                    LET T1SWB :B= T1SWB + #1        ;SET T1 SWAP BYTES FLAG FOR ''CKDATA'' SUB
  (6)  024002  105237  003531                                                               INCB    T1SWB
 4679
 4680  024006                                    BGNSUB                  ;SUBTEST 12 - READ SWAPPED DATA BYTES.
  (3)  024006                           T1.12:
  (3)  024006  104402                                                                       TRAP    C$BSUB
 4681
 4682  024010                                    LET CMDWRD := #RDR              ;CMD IS READ REV.
  (4)  024010  012737  104401  003430                                                       MOV     #RDR,CMD
 4683  024016  004737  015504                    JSR     PC,VFEXC        ;VERIFY ODD LENGTH SWAP (RECORD 4).
 4684  024022                                    LET CMDPKT+CP.CNT := #12        ;CHANGE BYTE COUNT TO 10.
  (4)  024022  012737  000012  002322                                                       MOV     #12,CMDP
 4685  024030  004737  015504                    JSR     PC,VFEXC        ;VERIFY EVEN LENGTH SWAP (RECORD 3).
 4686  024034                                    LET SWBFLG :B= #1       ;ENABLE BYTE SWAPPING.
  (4)  024034  112737  000001  003526                                                       MOVB    #1,SWBFL
 4687  024042                                    LET CMDPKT+CP.CNT := #11                ;CHANGE BYTE COUNT TO 9.
  (4)  024042  012737  000011  002322                                                       MOV     #11,CMDP
 4688  024050  004737  015504                    JSR     PC,VFEXC        ;VERIFY ODD LENGTH SWAP (RECORD 2).
 4689  024054                                    LET CMDPKT+CP.CNT := #12        ;CHANGE BYTE COUNT TO 10.
  (4)  024054  012737  000012  002322                                                       MOV     #12,CMDP
 4690  024062  004737  015504                    JSR     PC,VFEXC        ;VERIFY EVEN LENGTH SWAP (RECORD 1).
 4691  024066                                    LET CMDWRD := #RDF      ;CMD IS READ FWD.
  (4)  024066  012737  104001  003430                                                       MOV     #RDF,CMD
 4692  024074  004737  015504                    JSR     PC,VFEXC        ;VERIFY EVEN LENGTH SWAP (RECORD 1).
 4693  024100                                    LET CMDPKT+CP.CNT := #11                ;CHANGE BYTE COUNT TO 9.
  (4)  024100  012737  000011  002322                                                       MOV     #11,CMDP
```

```
 4694  024106  004737  015504                      JSR     PC,VFEXC              ;VERIFY ODD LENGTH SWAP (RECORD 2).
 4695  024112                                      LET SWBFLG :B= #0             ;DISABLE BYTE SWAPPING.
  (4)  024112  105037  003526                                                                CLRB    SWBFLG
 4696  024116                                      LET CMDPKT+CP.CNT := #12      ;CHANGE BYTE COUNT TO 10.
  (4)  024116  012737  000012  002322                                                       MOV     #12,CMDP
 4697  024124  004737  015504                      JSR     PC,VFEXC              ;VERIFY EVEN LENGTH SWAP (RECORD 3).
 4698  024130                                      LET CMDPKT+CP.CNT := #11      ;CHANGE BYTE COUNT TO 9.
  (4)  024130  012737  000011  002322                                                       MOV     #11,CMDP
 4699  024136  004737  015504                      JSR     PC,VFEXC              ;VERIFY ODD LENGTH SWAP (RECORD 4).
 4700
 4701  024142                                      ENDSUB
  (3)  024142                              L10033:
  (3)  024142  104403                                                                       TRAP    C$ESUB
 4702
 4703  024144                                      LET T1SWB :B= #0              ;CLEAR T1 SWAP BYTES FLAG
  (4)  024144  105037  003531                                                               CLRB    T1SWB
 4704
 4705
 4706  024150                                      EXIT    TST
  (3)  024150  104432                                                                       TRAP    C$EXIT
  (3)  024152  000554                                                                       .WORD   L10017-.
```

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

                                BFSEQ:  LET R1 := #CMDSEQ               ;INIT SEQ TABLE ADDRESS.
 024154                                                                                 MOV     #CMDSEQ,
                                        WHILE (R2) NE #END DO           ;WHILE THERE ARE MORE COMMANDS:
 024160                                                                                 50452$:
 024160 021227 177777                                                                   CMP     (R2),#EN
 024164 001402                                                                          BEQ     50453$
                                          LET (R1)+ := (R2)+            ;MOVE COMMANDS TO SEQ TABLE.
 024166                                                                                 MOV     (R2)+,(R
                                        ENDDO
 024170                                                                                 BR      50452$
 024172                                                                         50453$:
                                        LET (R1) := #END                ;STORE END OF SEQUENCE CODE.
 024172 012711 177777                                                                   MOV     #END,(R1
                                        RTS     PC                      ;RETURN.


                                ;       BASIC FUNCTION COMMAND SEQUENCE

                                BFSEQ0: .WORD   SCH                     ;SET CHAR. 200.         (1)
 024200 140004                          200
 024202 000200                          1
 024204 000001                          0
 024206 000000                          DRI                     ;DRIVE INIT.            (2)
 024210 100013                          1
 024212 000001                          1
 024214 000001                          0
 024216 000000                          SCH                     ;SET CHAR. 20           (3)
 024220 140004                          20
 024222 000020                          1
 024224 000001                          0
 024226 000000                          GES                     ;GET STATUS.            (4)
 024230 100017                          1
 024232 000001                          1
 024234 000001                          0
 024236 000000                          SCH                     ;SET CHAR. 40.          (5)
 024240 140004                          40
 024242 000040                          1
 024244 000001                          0
 024246 000000                  .WORD   END
 024250 177777
                                BFSEQ1:         RWD                     ;REWIND TWICE.          (6)
 024252 102010                          1
 024254 000001                          2
 024256 000002                          0
 024260 000000                  .WORD   END
 024262 177777

                                BFSEQ2:         WTV                     ;WRITE/VERIFY PAT 1.    (7)
 024264 104105                          DATCNT
 024266 004000                          1
 024270 000001
```

```
 4756  024272  000001                                    1
 4757  024274  104105                                    WTV             ;WTV PAT 2.         (8)
 4758  024276  004000                                    DATCNT
 4759  024300  000001                                    1
 4760  024302  000002                                    2
 4761  024304  104105                                    WTV             ;WTV PAT 3.         (9)
 4762  024306  004000                                    DATCNT
 4763  024310  000001                                    1
 4764  024312  000003                                    3
 4765  024314  104105                                    WTV             ;WTV PAT 4.         (10)
 4766  024316  004000                                    DATCNT
 4767  024320  000001                                    1
 4768  024322  000004                                    4
 4769  024324  104105                                    WTV             ;WTV PAT 5.         (11)
 4770  024326  004000                                    DATCNT
 4771  024330  000001                                    1
 4772  024332  000005                                    5
 4773  024334  104105                                    WTV             ;WTV PAT 6.         (12)
 4774  024336  004000                                    DATCNT
 4775  024340  000001                                    1
 4776  024342  000006                                    6
 4777  024344  104105                                    WTV             ;WTV PAT 0.         (13)
 4778  024346  004000                                    DATCNT
 4779  024350  000001                                    1
 4780  024352  000000                                    0
 4781  024354  177777                        .WORD       END
 4782
 4783  024356  100011                BFSEQ3:             WTM             ;WRITE TAPE MARK.   (14)
 4784  024360  000001                                    1
 4785  024362  000001                                    1
 4786  024364  000000                                    0
 4787  024366  104005                                    WRT             ;WRITE 10 RECORDS.  (15)
 4788  024370  004000                                    DATCNT
 4789  024372  000010                                    10
 4790  024374  000001                                    1
 4791  024376  100411                                    ERS             ;ERASE 10 TIMES.    (16)
 4792  024400  000001                                    1
 4793  024402  000010                                    10
 4794  024404  000000                                    0
 4795  024406  100011                                    WTM             ;WRITE TAPE MARK.   (17)
 4796  024410  000001                                    1
 4797  024412  000001                                    1
 4798  024414  000000                                    0
 4799  024416  101011                                    WTR             ;WTM RETRY          (18)
 4800  024420  000001                                    1
 4801  024422  000001                                    1
 4802  024424  000000                                    0
 4803  024426  177777                        .WORD       END
 4804
 4805  024430  105410                BFSEQ4:             SFR             ;SPACE 2 FILES REV. (19)
 4806  024432  000002                                    2
 4807  024434  000001                                    1
 4808  024436  000000                                    0
 4809  024440  105010                                    SFF             ;SPACE 2 FILES FWD. (20)
 4810  024442  000002                                    2
 4811  024444  000001                                    1
```

```
4812  024446  000000                                    0
4813  024450  105410                                    SFR             ;SPACE 2 FILES REV.     (21)
4814  024452  000001                                    1
4815  024454  000002                                    2
4816  024456  000000                                    0
4817  024460  105010                                    SFF             ;SPACE 2 FILES FWD.     (22)
4818  024462  000001                                    1
4819  024464  000002                                    2
4820  024466  000000                                    0
4821  024470  177777                          .WORD     END
4822
4823  024472  102010                BFSEQ5:             RWD             ;REWIND.                (23)
4824  024474  000001                                    1
4825  024476  000001                                    1
4826  024500  000000                                    0
4827  024502  104010                                    SRF             ;SPACE 7 RECORDS FWD.   (24)
4828  024504  000007                                    7
4829  024506  000001                                    1
4830  024510  000000                                    0
4831  024512  104410                                    SRR             ;SPACE 7 RECORDS REV.   (25)
4832  024514  000007                                    7
4833  024516  000001                                    1
4834  024520  000000                                    0
4835  024522  104010                                    SRF             ;SPACE 7 RECORDS FWD.   (26)
4836  024524  000001                                    1
4837  024526  000007                                    7
4838  024530  000000                                    0
4839  024532  104410                                    SRR             ;SPACE 7 RECORDS REV.   (27)
4840  024534  000001                                    1
4841  024536  000007                                    7
4842  024540  000000                                    0
4843  024542  177777                          .WORD     END
4844
4845  024544  102010                BFSEQ6:             RWD             ;REWIND.                (28)
4846  024546  000001                                    1
4847  024550  000001                                    1
4848  024552  000000                                    0
4849  024554  104005                                    WRT             ;WRITE.                 (29)
4850  024556  004000                                    DATCNT
4851  024560  000001                                    1
4852  024562  000001                                    1
4853  024564  105005                                    WRR             ;WRITE RETRY.           (30)
4854  024566  004000                                    DATCNT
4855  024570  000001                                    1
4856  024572  000001                                    1
4857  024574  177777                          .WORD     END
4858
4859  024576  104401                BFSEQ7:             RDR             ;READ REV.              (31)
4860  024600  004000                                    DATCNT
4861  024602  000001                                    1
4862  024604  000001                                    1
4863  024606  105401                                    RNR             ;READ NEXT REV.         (32)
4864  024610  004000                                    DATCNT
4865  024612  000001                                    1
4866  024614  000001                                    1
4867  024616  125401                                    RNF             ;READ NEXT FWD.         (33)
```

```
4868  024620  004000                              DATCNT
4869  024622  000001                              1
4870  024624  000001                              1
4871  024626  177777                    .WORD     END
4872
4873  024630  104001          BFSEQ8:             RDF                 ;READ FWD.              (34)
4874  024632  004000                              DATCNT
4875  024634  000001                              1
4876  024636  000001                              1
4877  024640  105001                              RPF                 ;READ PREVIOUS FWD.     (35)
4878  024642  004000                              DATCNT
4879  024644  000001                              1
4880  024646  000001                              1
4881  024650  125001                              RPR                 ;READ PREVIOUS REV.     (36)
4882  024652  004000                              DATCNT
4883  024654  000001                              1
4884  024656  000001                              1
4885  024660  177777                    .WORD     END
4886
4887  024662  101012          BFSEQ9:   .WORD     CLN                 ;CLEAN.                 (37)
4888  024664  000001                              1
4889  024666  000001                              1
4890  024670  000000                              0
4891  024672  102010                              RWD                 ;REWIND                 (38)
4892  024674  000001                              1
4893  024676  000001                              1
4894  024700  000000                              0
4895  024702  177777                    .WORD     END                 ;END OF SEQUENCE.
4896
4897  024704  104105          BFSE10:             WTV                 ;WRITE/VERIFY EVEN LENGTH.  (39)
4898  024706  000012                              12
4899  024710  000001                              1
4900  024712  000000                              0
4901  024714  104105                              WTV                 ;WRITE/VERIFY ODD LENGTH.   (40)
4902  024716  000011                              11
4903  024720  000001                              1
4904  024722  000000                              0
4905  024724  177777                    .WORD     END
4906                                    .EVEN
4907
4908  024726                          ENDTST
 (3)  024726          L10017:
 (3)  024726  104401                                                 TRAP    C$ETST
```

```
                                        .SBTTL  TEST 2: DATA RELIABILITY.

                                        ;++
                                        ; TEST TO CHECK THE DATA RELIABILITY OF THE TU80.
                                        ;--
                                                BGNTST
                                        T2::

                                                LET RANDOM :B= #1               ;SET THE RANDOM OPERATIONS FLAG.
                                                                                        MOVB    #1,RANDO
                                                LET EXPBOT :B= #0               ;CLEAR EXPECT BOT FLAG.
                                                                                        CLRB    EXPBOT
                                                LET R2 := #DATCNT - #1          ;SET UP THE RECORD LENGTH MASK.
                                                                                        MOV     #DATCNT,
                                                                                        DEC     R2
                                                LET LENMSK := COMP R2           ;ALLOW MAXIMUM BUFFER.
                                                                                        MOV     R2,LENMS
                                                                                        COM     LENMSK
                                                JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
                                                IFB STAFLG NE #0 THEN           ;IF STARTING THEN:
                                                                                        TSTB    STAFLG
                                                                                        BEQ     50454$

                                                  JSR   PC,SETRW                ;CMD2=REWIND
                                                  LET STAFLG :B= #0             ;CLR START FLAG.
                                                                                        CLRB    STAFLG
                                                ENDIF
                                                                                50454$:
                                                LET (R1)+ := #WTV               ;CMD3 = WRITE/ VERIFY.
                                                                                        MOV     #WTV,(R1
                                                LET (R1)+ := #DATCNT            ;SET BRF TO MAX FOR PATTERN GENERATION.
                                                                                        MOV     #DATCNT,
                                                LET R2 := COMP #RNOPSC
                                                                                        MOV     #RNOPSC,
                                                                                        COM     R2
                                                LET (R1)+ := R2                 ;31 OPERATIONS.
                                                                                        MOV     R2,(R1)+
                                                LET (R1)+ := #RANP              ;RANDOM PATTERN.
                                                                                        MOV     #RANP,(R
                                                REPEAT                          ;REPEAT TO EOT:
                                                                                50455$:
                                                  WHILE R1 LT #SEQEND DO        ;FILL SEQ TBL WITH RANDOM CMDS.
                                                                                50456$:
                                                                                        CMP     R1,#SEQE
                                                                                        BGE     50457$
                                                    LET RANS := RANS + RANB
                                                                                        ADD     RANB,RAN
                                                    LET R2 := RANS CLR.BY #177741  ;R2 = RANDOM # (0 - 36).
                                                                                        MOV     RANS,R2
                                                                                        BIC     #177741,
                                                    JSR PC,@RANCMD(R2)          ;SET UP A RANDOM CMD + BRF.
                                                  ENDDO
                                                                                        BR      50456$
                                                                                50457$:
                                                  LET (R1) := #END              ;STORE END OF SEQUENCE CODE IN TABLE.
                                                                                        MOV     #END,(R1
                                                  JSR   PC,EXALL                ;GO EXECUTE ALL CMDS IN SEQUENCE TABLE.
```

4910					
4911					
4912					
4913					
4914					
4915	024730				
(3)	02473C				
4916					
4917	024730				
(4)	024730	112737	000001	003523	
4918	024736				
(4)	024736	105037	003522		
4919	024742				
(4)	024742	012702	004000		
(6)	024746	005302			
4920	024750				
(6)	024750	010237	003440		
(6)	024754	005137	003440		
4921	024760	004737	006714		
4922	024764				
(6)	024764	105737	003536		
(9)	024770	001404			
4923	024772	004737	006740		
4924	024776				
(4)	024776	105037	003536		
4925	025002				
(4)	025002				
4926	025002				
(4)	025002	012721	104105		
4927	025006				
(4)	025006	012721	004000		
4928	025012				
(6)	025012	012702	177740		
(6)	025016	005102			
4929	025020				
(4)	025020	010221			
4930	025022				
(4)	025022	012721	000007		
4931	025026				
(3)	025026				
4932	025026				
(4)	025026				
(6)	025026	020127	003634		
(9)	025032	002012			
4933	025034				
(6)	025034	063737	003442	003444	
4934	025042				
(4)	025042	013702	003444		
(6)	025046	042702	177741		
4935	025052	004772	025210		
4936	025056				
(4)	025056	000763			
(3)	025060				
4937	025060				
(4)	025060	012711	177777		
4938	025064	004737	006760		

```
4939  025070
 (4)  025070  012701  003544                    LET R1 := #CMDSEQ              ;INIT CMD SEQ TBL POINTER,
                                                                                        MOV     #CMDSEQ,
4940  025074
 (3)  025074  005702                            UNTIL R2 NE #0                 ;REPEAT UNTIL EOT IS REACHED
 (6)  025076  001753                                                                    TST     R2
                                                                                        BEQ     50455$
4941  025100
 (6)  025100  105237  003532                    LET ALLEOT :B= ALLEOT + #1      ;FLAG ALL UNITS @ EOT
                                                                                        INCB    ALLEOT
4942  025104  000240                            NOP
4943  025106  000240                            NOP
4944  025110  000240                            NOP
4945  025112  004737  027072                    JSR     PC,T5WEOT              ;WRITE ONE RECORD BEYOND EOT ON ALL UNIT
4946                                                                            ;SO THAT SHORTER READ STOP DISTANCE
4947                                                                            ;SHALL POSITION HEAD IN CLEAN IRG GAP
4948                                                                            ;READ REV THAT EXTRA REC TO RE-POSITION
4949  025116  004737  025250                    JSR     PC,RANRD               ;SET UP READ REV/FWD CMDS,
4950  025122                                    LET CMDSEQ+4 := COMP #RNOPSC   ;# OF RECORDS FOR READ REV.
 (6)  025122  012737  177740  003550                                                 MOV     #RNOPSC,
 (6)  025130  005137  003550                                                        COM     CMDSEQ+4
4951  025134
 (4)  025134  013737  003550  003560            LET CMDSEQ+14 := CMDSEQ+4      ;# OF RECORDS FOR READ FORWARD.
                                                                                        MOV     CMDSEQ+4
4952  025142                                    LET (R1) := #END               ;STORE END OF SEQUENSE CODE IN SEQ TABLE
 (4)  025142  012711  177777                                                            MOV     #END,(R1
4953  025146  004737  006760                    JSR     PC,EXALL               ;GO EXECUTE READ REV/FWD OF LAST N RECOR
4954  025152                                    LET ALLEOT :B= #0              ;CLEAR ALL UNITS @ EOT FLAG
 (4)  025152  105037  003532                                                            CLRB    ALLEOT
4955  025156                                    LET RPTFLG :B= #1              ;REQUEST PERFORMANCE REPORT DURING REWIN
 (4)  025156  112737  000001  003525                                                    MOVB    #1,RPTFL
4956  025164                                    LET R1 := #CMDSEQ              ;INIT SEQ TBL POINTER,
 (4)  025164  012701  003544                                                            MOV     #CMDSEQ,
4957  025170  004737  006740                    JSR PC,SETRW                   ;STORE REWIND IN SEQ TBL,
4958  025174                                    LET (R1) := #END               ;STORE END IN SEQ TBL,
 (4)  025174  012711  177777                                                            MOV     #END,(R1
4959  025200  004737  006760                    JSR PC,EXALL                   ;EXECUTE REWIND CMD ON ALL UNITS
4960
4961  025204                                    EXIT    TST
 (3)  025204  104432                                                            TRAP    CSEXIT
 (3)  025206  000174                                                            .WORD   L10034-.
4962
```

```
                                        ;       ADDRESSES OF SUBROUTINES USED TO SET UP RANDOM OPERATIONS IN
                                        ;       THE DATA RELIABILITY TEST.
                                        ;
4964
4965
4966
4967  025210  025336                    RANCMD: RANWV                   ;WRITE/VERIFY.
4968  025212  025324                            RANWR                   ;WRITE.
4969  025214  025324                            RANWR                   ;WRITE.
4970  025216  025324                            RANWR                   ;WRITE.
4971  025220  025324                            RANWR                   ;WRITE.
4972  025222  025324                            RANWR                   ;WRITE.
4973  025224  025324                            RANWR                   ;WRITE.
4974  025226  025324                            RANWR                   ;WRITE.
4975  025230  025250                            RANRD                   ;READ.
4976  025232  025250                            RANRD                   ;READ.
4977  025234  025250                            RANRD                   ;READ.
4978  025236  025250                            RANRD                   ;READ.
4979  025240  025250                            RANRD                   ;READ.
4980  025242  025250                            RANRD                   ;READ.
4981  025244  025250                            RANRD                   ;READ.
4982  025246  025250                            RANRD                   ;READ.
4983
4984
4985
4986
4987
                                        ;       SUBROUTINE TO SET UP READ COMMANDS IN SEQUENCE TABLE.
                                        ;       INPUTS:
                                        ;       OUTPUTS:
                                        ;       REGISTERS:      R2
                                        ;       CALLS:
4992
4993
4994  025250                            RANRD:  LET (R1)+ := #RDR               ;STORE READ REV CMD.
 (4)  025250  012721  104401                                                            MOV     #RDR,(R1
4995  025254                                    LET (R1)+ := #DATCNT            ;SET BRF TO MAX FOR READ RANDOM LENGTHS.
 (4)  025254  012721  004000                                                            MOV     #DATCNT,
4996  025260                                    LET RANB := RANB + RANS
 (6)  025260  063737  003444  003442                                                    ADD     RANS,RAN
4997  025266                                    LET R2 := RANB CLR.BY #RNOPSC
 (4)  025266  013702  003442                                                            MOV     RANB,R2
 (6)  025272  042702  177740                                                            BIC     #RNOPSC,
4998  025276                                    LET (R1)+ := R2                 ;SET RANDOM # OF OPERATIONS.
 (4)  025276  010221                                                                    MOV     R2,(R1)+
4999  025300                                    LET (R1)+ := #RANP              ;RANDOM PATTERN.
 (4)  025300  012721  000007                                                            MOV     #RANP,(R
5000  025304                                    LET (R1)+ := #RDF               ;STORE READ FWD CMD.
 (4)  025304  012721  104001                                                            MOV     #RDF,(R1
5001  025310                                    LET (R1)+ := #DATCNT            ;SET BRF TO MAX TO READ RANDOM LENGTHS.
 (4)  025310  012721  004000                                                            MOV     #DATCNT,
5002  025314                                    LET (R1)+ := R2                 ;SET RANDOM # OF OPERATIONS.
 (4)  025314  010221                                                                    MOV     R2,(R1)+
5003  025316                                    LET (R1)+ := #RANP              ;RANDOM PATTERN.
 (4)  025316  012721  000007                                                            MOV     #RANP,(R
5004  025322  000207                           RTS PC
```

```
5006                                    ;       SUBROUTINE TO SET UP A WRITE COMMAND IN THE SEQUENCE TABLE.
5007                                    ;       INPUTS:
5008                                    ;       OUTPUTS:
5009                                    ;       REGISTERS:
5010                                    ;       CALLS:
5011
5012  025324                            RANWR:  LET (R1)+ := #WRT               ;STORE WRITE CMD.
 (4)  025324  012721  104005                                                    MOV     #WRT,(R1
5013  025330  004737  025350                    JSR PC,RANW                     ;STORE BRF, # OF OPERATIONS, PATTERN.
5014  025334  000207                            RTS PC
5015
5016
5017
5018
5019
5020                                    ;       SUBROUTINE TO SET UP A WRITE/VERIFY COMMAND IN THE SEQUENCE TABLE.
5021                                    ;       INPUTS:
5022                                    ;       OUTPUTS:
5023                                    ;       REGISTERS:
5024                                    ;       CALLS:
5025
5026  025336                            RANWV:  LET (R1)+ := #WTV               ;STORE WRITE/VERIFY CMD.
 (4)  025336  012721  104105                                                    MOV     #WTV,(R1
5027  025342  004737  025350                    JSR PC,RANW                     ;STORE BRF, # OF OPERATIONS, PATTERN.
5028  025346  000207                            RTS     PC
5029
5030
5031
5032
5033
5034                                    ;       SUBROUTINE TO STORE BRF, # OF OPERATIONS, PATTERN IN COMMAND
5035                                    ;       SEQUENCE TABLE FOR WRITE AND WRITE/VERIFY COMMANDS.
5036                                    ;       INPUTS:
5037                                    ;       OUTPUTS:
5038                                    ;       REGISTERS:      R2
5039                                    ;       CALLS:
5040
5041  025350                            RANW:   LET (R1)+ := #DATCNT            ;SET BRF TO MAX FOR PATTERN GENERATION.
 (4)  025350  012721  004000                                                    MOV     #DATCNT,
5042 -                                                                          ;RANDOM BRF WILL BE GENERATED FOR EACH R
5043  025354                                    LET RANB := RANB + RANS
 (6)  025354  063737  003444  003442                                           ADD     RANS,RAN
5044  025362                                    LET R2 := RANB CLR.BY #RNOPSC
 (4)  025362  013702  003442                                                    MOV     RANB,R2
 (6)  025366  042702  177740                                                    BIC     #RNOPSC,
5045  025372                                    LET (R1)+ := R2               ;SET RANDOM # OF OPERATIONS.
 (4)  025372  010221                                                            MOV     R2,(R1)+
5046  025374                                    LET (R1)+ := #RANP            ;RANDOM PATTERN.
 (4)  025374  012721  000007                                                    MOV     #RANP,(R
5047  025400  000207                            RTS PC                          ;RETURN.
5048
5049                                            .EVEN
5050
5051  025402                                    ENDTST
```

```
   (3)  025402                          L10034:
   (3)  025402  104401                                    TRAP    C$ETST
  5052
```

```
                                        .SBTTL  TEST 3: WRITE AND READ STREAMING TEST.

                                        ;++
                                        ;
                                        ;       THIS TEST STREAM WRITES 7000 RECORDS OF 1000 BYTES EACH.
                                        ;       DATA IS THEN VERIFIED BY READ REVERSE 7000 RECORDS, FOLLOWED
                                        ;       BY READ FORWARD THE 7000 RECORDS.  THIS PARTICULAR SEQUENCE IS
                                        ;       PERFORMED TWICE ON EACH UNIT UNDER TEST.  ALL STREAMING SPEEDS
                                        ;       ARE TESTED FOR BOTH READ AND WRITE OPERATIONS.
                                        ;
                                        ;--

025404                                  BGNTST
025404                                  T3::
025404                                  IFB HERE EQ #0 THEN
025404  105737  003341                                                  TSTB    HERE
025410  001013                                                          BNE     50460$
025412                                   PRINTF #TAPCAP
025412  012746  021112                                                  MOV     #TAPCAP,
025416  012746  000001                                                  MOV     #1,-(SP)
025422  010600                                                          MOV     SP,R0
025424  104417                                                          TRAP    C$PNTF
025426  062706  000004                                                  ADD     #4,SP
025432                                   LET HERE :B= #255.
025432  112737  000377  003341                                          MOVB    #255.,HE
025440                                  ENDIF
025440                                                          50460$:
                                        ;
025440                                  LET DOAGIN :B= #0               ;FIRST OF TWO ITERATONS
025440  105037  003340                                                  CLRB    DOAGIN
025444                                  LET RANDOM :B= #0               ;CLEAR THE RANDOM OPERATIONS FLAG.
025444  105037  003523                                                  CLRB    RANDOM
025450                                  LET EXPBOT :B= #0               ;CLEAR THE EXPECT BOT FLAG.
025450  105037  003522                                                  CLRB    EXPBOT
025454  004737  006714                  JSR PC, SETCH                   ;SET CHARACTERISTICS.
025460  004737  006740                  JSR PC, SETRW                   ;SET REWIND COMMAND IN BUFFER.
025464                                  LET STAFLG :B= #0               ;CLEAR THE START FLAG.
025464  105037  003536                                                  CLRB    STAFLG
025470                                  LET (R1) := #END                ;PLACE END FLAG IN SEQUENCE TABLE.
025470  012711  177777                                                  MOV     #END,(R1
025474  004737  006760                  JSR PC, EXALL                   ;REWIND ALL UNITS.
                                        ;
025500  004737  016614                  JSR PC, FIRSTU                  ;FIND THE FIRST UNIT TO TEST (UUT)
                                        ; ***************************************************************
                                        ;
                                        ;WRITE AND READ EACH UNIT IN TURN BEFORE GOING ON TO THE NEXT.
                                        ;
                                        ; ***************************************************************
                                        ;
025504                                  WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE MORE DEVICES:
025504                                                          50461$:
025504  026527  002536  177777                                          CMP     DEVTBL(R
025512  001546                                                          BEQ     50462$
                                        ;
025514                                  LET BTPT := BTADDR(R5)          ;CLEAR BAD SPOT COUNTER
```

```
  (4)  025514  016537  002550  003520                                                             MOV     BTADDR(R
 5092  025522                                  LET @BTPT := #0              ;START FROM BOT
  (4)  025522  005077  155772                                                                     CLR     @BTPT
 5093  025526                                  LET STREAM :B= #255.         ;SET FLAG - WE'RE GOING TO STREAM
  (4)  025526  112737  000377  003533                                                             MOVB    #255.,ST
 5094  025534                        1$:         LET R1 := #CMDSEQ          ;SETUP SEQUENCE TABLE ADDRESS
  (4)  025534  012701  003544                                                                     MOV     #CMDSEQ,
 5095  025540                                    LET (R1)+ := #WRT          ;WRITE COMMAND
  (4)  025540  012721  104005                                                                     MOV     #WRT,(R1
 5096  025544                                    LET (R1)+ := #1000.        ;1000-BYTE RECORD LENGTH.
  (4)  025544  012721  001750                                                                     MOV     #1000.,(
 5097  025550                                    LET (R1)+ := #7000.        ;WRITE 7000 RECORDS.
  (4)  025550  012721  015530                                                                     MOV     #7000.,(
 5098  025554                                    LET (R1)+ := #5            ;GENERATE AND WRITE PATTERN 5.
  (4)  025554  012721  000005                                                                     MOV     #5,(R1)+
 5099  025560                                    LET (R1)  := #END          ;SET END OF SEQUENCE TABLE.
  (4)  025560  012711  177777                                                                     MOV     #END,(R1
 5100
 5101  025564                                    LET R1 := #CMDSEQ          ;SEQ. TABLE ADDRESS FOR SUBR. 'SETUP'.
  (4)  025564  012701  003544                                                                     MOV     #CMDSEQ,
 5102  025570  004737  007722                    JSR PC, SETUP              ;SETUP THE COMMAND TABLE
 5103
 5104  025574                                      WHILE NCNT LT NCNT1 DO   ;WHILE MORE RECORDS SHOULD BE WRITTEN:
  (4)  025574                                                                          50463$:
  (6)  025574  023737  003422  003424                                                             CMP     NCNT,NCN
  (9)  025602  002022                                                                             BGE     50464$
 5105
 5106  025604  004737  007614                      JSR PC, CMDAC            ;SAVE ASCII COMMAND FOR ERROR MESSAGE
 5107  025610  004737  010576                      JSR PC, EXCUTE           ;ISSUE COMMAND TO UNIT.
 5108  025614  004737  011166                      JSR PC, GOWAIT           ;GO WAIT FOR DONE TO SET
 5109  025620                                  IF DEVTBL(R5) EQ #NINUSE THEN
  (6)  025620  026527  002536  177774                                                             CMP     DEVTBL(R
  (9)  025626  001005                                                                             BNE     50465$
 5110  025630                                  LET NCNT := NCNT1 - #1
  (4)  025630  013737  003424  003422                                                             MOV     NCNT1,NC
  (6)  025636  005337  003422                                                                     DEC     NCNT
 5111  025642                                  ENDIF
  (4)  025642                                                                          50465$:
 5112  025642                                      LET NCNT := NCNT + #1    ;UPDATE THE RECORD COUNT
  (6)  025642  005237  003422                                                                     INC     NCNT
 5113  025646                                    ENDDO                      ;END OF RECORD 'DO' LOOP
  (4)  025646  000752                                                                             BR      50463$
  (3)  025650                                                                          50464$:
 5114  025650                                  IF DEVTBL(R5) NE #NINUSE THEN ;IF DRIVE DROPPED EXIT
  (6)  025650  026527  002536  177774                                                             CMP     DEVTBL(R
  (9)  025656  001453                                                                             BEQ     50466$
 5115  025660  004737  017312                  JSR PC, WRITEM               ;WRITE A TAPE MARK NOW
 5116  025664                                  LET ERSFLG :B= #0            ;DON'T ERASE AFTER BACKSPACE
  (4)  025664  105037  003534                                                                     CLRB    ERSFLG
 5117  025670  004737  014374                  JSR PC, BORERS               ;BACKSPACE OVER THE TAPE MARK
 5118  025674                                  LET R1 := #CMDSEQ            ;RELOAD THE COMMAND TABLE
  (4)  025674  012701  003544                                                                     MOV     #CMDSEQ,
 5119  025700                                  LET (R1)+ := #WRT            ;NOT REALLY NECESSARY FOR WHAT FOLLOWS
  (4)  025700  012721  104005                                                                     MOV     #WRT,(R1
 5120  025704                                  LET (R1)+ := #1000.          ;1000-BYTE RECORD LENGTH
  (4)  025704  012721  001750                                                                     MOV     #1000.,(
```

```
5121  025710
 (4)  025710  012721  015530
5122  025714
 (4)  025714  012721  000005
5123  025720
 (4)  025720  012711  177777
5124  025724
 (4)  025724  012701  003544
5125  025730  004737  007722
5126  025734
 (4)  025734  112737  000001  003524
5127  025742
 (4)  025742  010537  003462
5128  025746  004737  015370
5129  025752
 (4)  025752  013705  003462
5130  025756
 (6)  025756  026527  002536  177774
 (9)  025764  001410
5131  025766  105137  003340
5132  025772  001405
5133  025774
 (4)  025774  005037  003422
5134  026000  004737  017312
5135  026004  000653
5136  026006
 (4)  026006
5137  026006
 (4)  026006
5138  026006
 (4)  026006  005037  003422
5139  026012
 (4)  026012  105037  003524
5140  026016
 (4)  026016  105037  003522
5141  026022  004737  016662
5142
5143  026026
 (4)  026026  000626
 (3)  026030
5144
5145  026030
 (4)  026030  105037  003533
5146
5147  026034
 (3)  026034  104432
 (3)  026036  000002
5148
5149
5150  026040
 (3)  026040
 (3)  026040  104401
```

```
                LET (R1)+ := #7000.             ;FOR 7000 ITERATIONS
                LET (R1)+ := #5                 ;DATA PATTERN NUMBER 5
                LET (R1) := #END                ;TABLE TERMINATOR
                LET R1 := #CMDSEQ               ;TOP OF THE TABLE AGAIN
                JSR PC, SETUP                   ;SET UP THE COMMAND TABLE
                LET VFYFLG :B= #1               ;ALLOW THE DATA VERIFY
                LET R5SAVE := R5                ;SAVE R5
                JSR PC, VFYDAT                  ;GO OFF AND CHECK REV AND FWD.
                LET R5 := R5SAVE                ;RESTORE R5
                  IF DEVTBL(R5) NE #NINUSE THEN ;IF DRIVE DROPPED EXIT
                COMB    DOAGIN                  ;LOOP CONTROL
                BEQ     2$                      ;IF ZERO, DONE
                LET NCNT := #0                  ;KEEP CONTROL HERE
                JSR PC, WRITEM                  ;WRITE ANOTHER TAPE MARK
                BR      1$                      ;AND LOOP
                  ENDIF
                ENDIF
2$:             LET NCNT := #0                  ;CLEAR RECORD COUNT
                LET VFYFLG :B= #0               ;CLEAR VERIFY FLAG
                LET EXPBOT :B= #0               ;CLEAR EXPECT BOT FLAG.
                JSR PC, NEXTU                   ;GET NEXT UNIT TO TEST (UUT).
                ENDDO                           ;END OF UUT LOOP
                LET STREAM :B= #0               ;CLEAR STREAMING FLAG FOR OTHER TESTS.
                EXIT    TST                     ;EXIT TEST
                .EVEN                           ;JUST IN CASE.
                ENDTST
L10035:
```

```
                        MOV     #7000.,(
                        MOV     #5,(R1)+
                        MOV     #END,(R1
                        MOV     #CMDSEQ,
                        MOVB    #1,VFYFL
                        MOV     R5,R5SAV
                        MOV     R5SAVE,R
                        CMP     DEVTBL(R
                        BEQ     50467$
                        CLR     NCNT
                50467$:
                50466$:
                        CLR     NCNT
                        CLRB    VFYFLG
                        CLRB    EXPBOT
                        BR      50461$
                50462$:
                        CLRB    STREAM
                        TRAP    C$EXIT
                        .WORD   L10035-.
                        TRAP    C$ETST
```

```
                                .SBTTL  TEST 4: WRITE COMPATABILITY/WRITE UTILITY.

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

                                        BGNTST
                                T4::

                                        LET RANDOM :B= #1               ;SET THE RANDOM OPERATIONS FLAG.
                                                                                MOVB    #1,RANDO
                                        LET EXPBOT :B= #0               ;CLEAR EXPECT BOT FLAG.
                                                                                CLRB    EXPBOT
                                        LET R2 := #DATCNT - #1          ;SET UP THE RECORD LENGTH MASK.
                                                                                MOV     #DATCNT,
                                                                                DEC     R2
                                        LET LENMSK := COMP R2           ;ALLOW MAXIMUM BUFFER.
                                                                                MOV     R2,LENMS
                                                                                COM     LENMSK
                                        JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
                                        JSR PC,SETRW                    ;CMD2=REWIND
                                        LET STAFLG :B= #0               ;CLEAR START FLAG
                                                                                CLRB    STAFLG
                                        REPEAT                          ;REPEAT TO EOT.
                                                                        50470$:
                                          WHILE R1 LT #SEQEND DO        ;WHILE THERE IS MORE ROOM IN SEQ TABLE:
                                                                        50471$:
                                                                                CMP     R1,#SEQE
                                                                                BGE     50472$
                                            JSR PC,RANWR                  ;STORE A WRITE CMD IN SEQUENCE TABLE.
                                          ENDDO
                                                                                BR      50471$
                                                                        50472$:
                                          LET (R1) := #END              ;STORE END OF SEQUENCE CODE IN TABLE.
                                                                                MOV     #END,(R1
                                          JSR   PC,EXALL                ;EXECUTE ALL CMDS IN SEQ TBL ON UNITS.
                                          LET R1 := #CMDSEQ             ;INIT SEQ TBL POINTER,
                                                                                MOV     #CMDSEQ,
                                        UNTIL R2 NE #0                  ;REPEAT UNTIL EOT IS REACHED
                                                                                TST     R2
                                                                                BEQ     50470$
                                        LET ALLEOT :B= ALLEOT + #1      ;SET ALL UNITS @ EOT FLAG
                                                                                INCB    ALLEOT
                                        NOP
                                        NOP
                                        NOP
                                        JSR     PC,T5WEOT               ;WRITE ONE RECORD BEYOND EOT ON ALL UNIT
                                                                        ;SO THAT SHORTER READ STOP DISTANCE
                                                                        ;SHALL POSITION HEAD IN CLEAN IRG GAP
                                                                        ;READ REV THAT EXTRA REC TO RE-POSITION
                                        LET ALLEOT :B= #0               ;CLEAR ALL UNITS @ EOT FLAG
                                                                                CLRB    ALLEOT
                                        JSR PC,SETRW                    ;STORE REWIND IN SEQ TBL,
                                        LET (R1) := #END                ;STORE END IN SEQ TBL,
                                                                                MOV     #END,(R1
```

```
HARDWARE TESTS  MACY11 30(1046)  12-JUL-83  09:44  PAGE 77-6
CZTUVB.P11    12-JUL-83 09:26          TEST 4: WRITE COMPATABILITY/WRITE UTILITY.                                    SEQ 0157

  5187  026174  004737  006760                    JSR PC,EXALL                ;EXECUTE REWIND CMD ON ALL UNITS
  5188
  5189  026200                                    EXIT    TST
   (3)  026200  104432                                                        TRAP    C$EXIT
   (3)  026202  000002                                                        .WORD   L10036-.
  5190
  5191                                            .EVEN
  5192
  5193  026204                                    ENDTST
   (3)  026204                            L10036:
   (3)  026204  104401                                                        TRAP    C$ETST
  5194
```

```
5196
5197                                    .SBTTL  TEST 5:  READ COMPATABILITY/READ UTILITY.
5198
5199                                    ;++
5200                                    ; TEST TO READ ENTIRE TAPE FORWARD AND REVERSE.
5201                                    ;--
5202
5203  026206                                    BGNTST
 (3)  026206                            T5::
5204
5205  026206                                    LET RANDOM :B= #1             ;SET THE RANDOM OPERATIONS FLAG.
 (4)  026206  112737  000001  003523                                                MOVB    #1,RANDO
5206  026214                                    LET EXPBOT :B= #1             ;SET EXPECT BOT FLAG.
 (4)  026214  112737  000001  003522                                                MOVB    #1,EXPBO
5207  026222  004737  006714                    JSR PC,SETCH                  ;CMD 1 = SET CHARACTERISTIC.
5208  026226  004737  006740                    JSR PC,SETRW                  ;CMD2=REWIND.
5209  026232                                    LET STAFLG :B= #0             ;CLEAR START FLAG
 (4)  026232  105037  003536                                                CLRB    STAFLG
5210  026236                                    LET (R1)+ := #RDF             ;CMD3 = READ FORWARD.
 (4)  026236  012721  104001                                                MOV     #RDF,(R1
5211  026242                                    LET (R1)+ := #DATCNT          ;SET LENGTH TO MAX FOR UNKNOWN LENGTHS.
 (4)  026242  012721  004000                                                MOV     #DATCNT,
5212  026246                                    LET (R1)+ := #77777           ;SET RECORD COUNT TO MAX FOR WHOLE TAPE.
 (4)  026246  012721  077777                                                MOV     #77777,(
5213  026252                                    LET (R1)+ := #RANP            ;PATTERN = RANDOM.
 (4)  026252  012721  000007                                                MOV     #RANP,(R
5214  026256                                    LET (R1) := #END              ;STORE END OF SEQUENCE CODE IN TABLE.
 (4)  026256  012711  177777                                                MOV     #END,(R1
5215  026262  004737  006760                    JSR     PC,EXALL              ;EXECUTE ALL CMDS IN SEQ TBL ON ALL UNIT
5216  026266                                    LET ALLEOT :B= ALLEOT + #1    ;FLAG TO ALLOW ALL UNITS AT EOT TO READ
 (6)  026266  105237  003532                                                INCB    ALLEOT
5217  026272                                    LET R1 := #CMDSEQ             ;INIT CMD SEQ TBL POINTER.
 (4)  026272  012701  003544                                                MOV     #CMDSEQ,
5218  026276                                    LET (R1)+ := #RDR             ;CMD1 = READ REVERSE.
 (4)  026276  012721  104401                                                MOV     #RDR,(R1
5219  026302                                    LET (R1)+ := #DATCNT          ;SET LENGTH TO MAX FOR UNKNOWN LENGTHS.
 (4)  026302  012721  004000                                                MOV     #DATCNT,
5220  026306                                    LET (R1)+ := #77777           ;RECORD COUNT = MAX FOR WHOLE TAPE.
 (4)  026306  012721  077777                                                MOV     #77777,(
5221  026312                                    LET (R1)+ := #RANP            ;PATTERN = RANDOM.
 (4)  026312  012721  000007                                                MOV     #RANP,(R
5222  026316                                    LET (R1) := #END              ;STORE END OF SEQUENCE CODE IN TABLE.
 (4)  026316  012711  177777                                                MOV     #END,(R1
5223  026322  004737  006760                    JSR     PC,EXALL              ;GO EXECUTE READ REV. OF ENTIRE TAPE.
5224  026326                                    LET ALLEOT :B= #0             ;CLEAR ALL UNITS @ EOT FLAG
 (4)  026326  105037  003532                                                CLRB    ALLEOT
5225
5226  026332                                    EXIT    TST
 (3)  026332  104432                                                        TRAP    C$EXIT
 (3)  026334  000002                                                        .WORD   L10037-.
5227
5228                                    .EVEN
5229
5230  026336                                    ENDTST
 (3)  026336                            L10037:
 (3)  026336  104401                                                        TRAP    C$ETST
```

```
5232
5233                                    .SBTTL  TEST 6: EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.
5234
5235                                    ;++
5236                                    ; TEST TO EXECUTE  OPERATOR SELECTED COMMAND SEQUENCE.
5237                                    ;--
5238
5239  026340                                    BGNTST
 (3)  026340                            T6::
5240
5241  026340                                    LET RANDOM :B= #0               ;CLEAR RAMDOM MODE FLAG.
 (4)  026340  105037  003523                                                            CLRB    RANDOM
5242  026344                                    LET EXPBOT :B= #1               ;SET EXPECT BOT FLAG.
 (4)  026344  112737  000001  003522                                                    MOVB    #1,EXPBO
5243  026352                                    LET IRE :B= PIRE                ;MOVE INHIBIT RFC ERROR REPORT FLAG.
 (4)  026352  113737  002213  003527                                                    MOVB    PIRE,IRE
5244  026360  004737  006714                   JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
5245  026364                                    LET CMDSEQ+2 := CHAR            ;MOVE CHAR CODE FROM P TBL TO SEQ TBL.
 (4)  026364  013737  002220  003546                                                    MOV     CHAR,CMD
5246  026372                                    LET R2 := #CMDD                 ;R2 POINTS TO CMD2 IN SOFT P TABLE.
 (4)  026372  012702  002222                                                            MOV     #CMDD,R2
5247  026376  004737  027050                   JSR     PC,PTCMDS               ;MOVE CMD 2 FROM P TBL TO SEQ TBL.
5248  026402  004737  027050                   JSR     PC,PTCMDS               ;MOVE CMD 3 FROM P TBL TO SEQ TBL.
5249  026406  004737  027050                   JSR     PC,PTCMDS               ;MOVE CMD 4 FROM P TBL TO SEQ TBL.
5250  026412  004737  027050                   JSR     PC,PTCMDS               ;MOVE CMD 5 FROM P TBL TO SEQ TBL.
5251  026416  004737  027050                   JSR     PC,PTCMDS               ;MOVE CMD 6 FROM P TBL TO SEQ TBL.
5252  026422  004737  027050                   JSR     PC,PTCMDS               ;MOVE CMD 7 FROM P TBL TO SEQ TBL.
5253  026426  004737  027050                   JSR     PC,PTCMDS               ;MOVE END CMD FROM P TBL TO SEQ TBL.
5254  026432                                    LET JLOOP := #0                 ;CLEAR JMP CMD LOOP COUNT.
 (4)  026432  005037  003452                                                            CLR     JLOOP
5255  026436                                    LET STAFLG :B= #0               ;CLEAR START FLAG
 (4)  026436  105037  003536                                                            CLRB    STAFLG
5256  026442                                    LET R1 := #CMDSEQ               ;INIT SEQUENCE TABLE POINTER.
 (4)  026442  012701  003544                                                            MOV     #CMDSEQ,
5257
5258         000000                     $BRJMP=0       ;ENABLE JMP SUBSTITUTION FOR BR, IF NECESSARY.
5259
5260  026446                            3$:     WHILE (R1) NE #END DO           ;WHILE THERE ARE CMDS LEFT IN SEQUENCE T
 (4)  026446                                                                    50473$:
 (6)  026446  021127  177777                                                            CMP     (R1),#EN
 (9)  026452  001574                                                                    BEQ     50474$
5261  026454  022711  000040                       CMP   #JMP.C,(R1)               ;IS THIS A JUMP CMD?
5262  026460  001024                               BNE   6$                        ;BR IF NOT.
5263  026462                                      LET R1 := R1 + #2               ;POINT TO BRF.
 (6)  026462  062701  000002                                                            ADD     #2,R1
5264  026466  012137  003454                       MOV   (R1)+,JLOC                ;SAVE BRF (LOCATION).
5265  026472  022137  003452                       CMP (R1)+,JLOOP                 ;HAS LOOP COUNT BE SATISFIED?
5266  026476  001003                               BNE   1$                        ;IF NOT, JMP AGAIN.
5267  026500                                      LET R1 := R1 + #2               ;IF SO, ADJUST SEQ POUNTER
 (6)  026500  062701  000002                                                            ADD     #2,R1
5268  026504  000760                             BR    3$                        ;AND GO TO NEXT COMMAND.
5269  026506                            1$:     LET JLOOP := JLOOP + #1         ;UPDATE THE LOOP COUNT.
 (6)  026506  005237  003452                                                            INC     JLOOP
5270  026512                                    LET R1 := #CMDSEQ               ;INIT CMD SEQ TABLE POINTER.
 (4)  026512  012701  003544                                                            MOV     #CMDSEQ,
5271  026516  005337  003454           2$:     DEC   JLOC                      ;DECR LOCATION COUNTER.
```

```
5272  026522  001751                          BEQ   3$                        ;IF THIS IS THE RIGHT LOCATION TO JMP TO
5273  026524                                  LET R1 := R1 + #10                ;IF NOT, UPDATE SEQ POINTER TO NEXT CMD.
 (6)  026524  062701  000010                                                            ADD     #10,R1
5274  026530  000772                         BR    2$                          ;DO IT AGAIN.
5275  026532  022711  000020          6$:    CMP   #DLY.C,(R1)                 ;DELAY?
5276  026536  001026                         BNE   4$                          ;BR IF NOT.
5277  026540                                  LET R1 := R1 + #4                 ;R1 = LOCATION OF N COUNT.
 (6)  026540  062701  000004                                                            ADD     #4,R1
5278  026544                                  LET TIME2 := (R1)                 ;SAVE N COUNT.
 (4)  026544  011137  003450                                                            MOV     (R1),TIM
5279  026550                          7$:     DELAY 10.                         ;GO TO SUPER-WAIT 1 MSEC.
 (2)  026550  012727  000012                                                            MOV     #10.,(PC
 (2)  026554  000000                                                                    .WORD   0
 (2)  026556  013727  002116                                                            MOV     L$DLY,(P
 (2)  026562  000000                                                                    .WORD   0
 (2)  026564  005367  177772                                                            DEC     -6(PC)
 (2)  026570  001375                                                                    BNE     .-4
 (2)  026572  005367  177756                                                            DEC     -22(PC)
 (2)  026576  001367                                                                    BNE     .-20
5280  026600  005337  003450                  DEC   TIME2
5281  026604  001361                         BNE   7$
5282  026606                                  LET R1 := R1 + #4                 ;POINT TO NEXT CMD.
 (6)  026606  062701  000004                                                            ADD     #4,R1
5283  026612  000715                         BR    3$                          ;GO CHECK NEXT CMD.
5284  026614  004737  007722          4$:    JSR   PC,SETUP                    ;GO SETUP THE COMMAND BLOCK.
5285  026620                                  WHILE NCNT LT NCNT1 DO            ;WHILE THERE ARE RECORDS REMAINING:
 (4)  026620                                                                    50475$:
 (6)  026620  023737  003422  003424                                                    CMP     NCNT,NCN
 (9)  026626  002103                                                                    BGE     50476$
5286  026630  004737  007614                  JSR PC,CMDAC                      ;STORE CMD ASCII IN ERROR MSG.
5287  026634  004737  007254                  JSR PC,EXSUB                      ;ISSUE CMD TO ALL,AWAIT INTS,CHECK STATU
5288  026640                                  IF CMDWRD EQ #GES THEN            ;IF CMD IS GET STATUS THEN:
 (6)  026640  023727  003430  100017                                                    CMP     CMDWRD,#
 (9)  026646  001002                                                                    BNE     50477$
5289  026650  004737  017140                    JSR PC,PRXST                    ;PRINT EXTENDED STATUS REGISTERS.
5290  026654                                  ENDIF
 (4)  026654                                                                    50477$:
5291  026654  004737  017222                  JSR PC,CKHAE                      ;CHECK HALT AFTER EACH CMD FLAG.
5292  026660                                  LET R2 := #1                      ;SET ALL UNITS AT BOT/EOT.
 (4)  026660  012702  000001                                                            MOV     #1,R2
5293  026664  004737  016614                  JSR PC,FIRSTU                     ;FIND FIRST UNIT.
5294  026670                                  WHILE DEVTBL(R5) NE #END DO       ;WHILE THERE ARE MORE UNITS:
 (4)  026670                                                                    50500$:
 (6)  026670  026527  002536  177777                                                    CMP     DEVTBL(R
 (9)  026676  001426                                                                    BEQ     50501$
5295  026700                                    IF #MOD.CO SETIN CMDWRD THEN    ;IF CMD IS REVERSE THEN:
 (6)  026700  032737  000400  003430                                                    BIT     #MOD.CO,
 (9)  026706  001406                                                                    BEQ     50502$
5296  026710                                      IF #X0.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT THEN:
 (6)  026710  032765  000002  003510                                                    BIT     #X0.BOT,
 (9)  026716  001001                                                                    BNE     50503$
5297  026720                                        LET R2 := #0                ;CLEAR EOT/BOT FLAG.
 (4)  026720  005002                                                                    CLR     R2
5298  026722                                      ENDIF
 (4)  026722                                                                    50503$:
5299  026722                                    ELSE                            ;ELSE IF CMD IS NOT REVERSE:
```

```
  (4)  026722  000411                                                                        BR      50504$
  (3)  026724                                                                        50502$:
 5300  026724                                  IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
  (6)  026724  032765  000001  003510                                                        BIT     #XO.EOT,
  (8)  026732  001404                                                                        BEQ     50505$
  (6)  026734  032737  000001  003430                                                        BIT     #CMD.CO,
  (9)  026742  001001                                                                        BNE     50506$
  (6)  026744                                                                        50505$:
 5301                                                         ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
 5302  026744                                    LET R2 := #0             ;CLEAR EOT/BOT FLAG.
  (4)  026744  005002                                                                        CLR     R2
 5303  026746                                  ENDIF
  (4)  026746                                                                        50506$:
 5304  026746                                 ENDIF
  (4)  026746                                                                        50504$:
 5305  026746  004737  016662                  JSR PC,NEXTU               ;FIND NEXT UNIT
 5306  026752                                ENDDO                        ;
  (4)  026752  000746                                                                        BR      50500$
  (3)  026754                                                                        50501$:
 5307  026754                                IF R2 EQ #1 THEN             ;IF ALL UNIT ARE AT EOT/BOT THEN:
  (6)  026754  020227  000001                                                                CMP     R2,#1
  (9)  026760  001016                                                                        BNE     50507$
 5308  026762                                 LET NCNT1 := NCNT + #1      ;FORCE TERMINATION OF COMMAND.
  (4)  026762  013737  003422  003424                                                        MOV     NCNT,NCN
  (6)  026770  005237  003424                                                                INC     NCNT1
 5309  026774                                 LET ALLEOT :B= ALLEOT + #1 ;FLAG ALL UNITS AT EOT/BOT TO ALLOW VER
  (6)  026774  105237  003532                                                                INCB    ALLEOT
 5310  027000                                 IF CMDLG EQ #2 THEN         ;WHEN WRITING IS CURRENT COMMAND
  (6)  027000  023727  003436  000002                                                        CMP     CMDLG,#2
  (9)  027006  001002                                                                        BNE     50510$
 5311  027010  004737  027072                  JSR PC,T5WEOT              ;GO WRITE/READ REV ONE RECORD BEYOND EOT
 5312  027014                                 ENDIF
  (4)  027014                                                                        50510$:
 5313  027014                                ELSE
  (4)  027014  000402                                                                        BR      50511$
  (3)  027016                                                                        50507$:
 5314  027016                                 LET ALLEOT :B= #0 ;WHEN NOT ALL @EOT, CLEAR FLAG
  (4)  027016  105037  003532                                                                CLRB    ALLEOT
 5315  027022                                ENDIF
  (4)  027022                                                                        50511$:
 5316  027022                                LET NCNT := NCNT + #1        ;UPDATE RECORD COUNT.
  (6)  027022  005237  003422                                                                INC     NCNT
 5317  027026                                LET PCMDWD := CMDWRD         ;SAVE PREVIOUS COMMAND WORD.
  (4)  027026  013737  003430  003434                                                        MOV     CMDWRD,P
 5318  027034                               ENDDO
  (4)  027034  000671                                                                        BR      50475$
  (3)  027036                                                                        50476$:
 5319  027036  004737  015370               JSR    PC,VFYDAT              ;IF LAST CMD WAS A WRITE VERIFY, THEN GO
 5320                                                                     ;VERIFY THE LAST N RECORDS OF DATA.
 5321
 5322  027042                              ENDDO
  (4)  027042  000601                                                                        BR      50473$
  (3)  027044                                                                        50474$:
 5323
 5324          177777               $BRJMP=-1      ;TURN OFF JMP SUBSTITUTION (SPMACJ CONTROL, ONLY).
 5325
```

```
5326  027044                                  EXIT    TST
 (3)  027044  104432                                        TRAP    C$EXIT
 (3)  027046  000130                                        .WORD   L10040-.
5327
```

```
                        ;       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:

                        PTCMDS: LET R3 := (R2)+ - #1 SHIFT +1   ;R3 = COMMAND TABLE INDEX.
027050  012203                                                          MOV     (R2)+,R3
027052  005303                                                          DEC     R3
027054  006303                                                          ASL     R3
                                LET (R1)+ := CMDTBL(R3)         ;MOVE COMMAND WORD.
027056  016321  003646                                                  MOV     CMDTBL(R
                                LET (R1)+ := (R2)+              ;MOVE # OF BYTES.
027062  012221                                                          MOV     (R2)+,(R
                                LET (R1)+ := (R2)+              ;MOVE # OF OPERATIONS.
027064  012221                                                          MOV     (R2)+,(R
                                LET (R1)+ := (R2)+              ;MOVE PATTERN CODE.
027066  012221                                                          MOV     (R2)+,(R
027070  000207                  RTS PC

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

027072  000240          T5WEOT: NOP
027074  000240                  NOP
027076  004737  007254          JSR PC,EXSUB            ;WRITE ONE RECORD BEYOND EOT
027102  004737  017222          JSR PC,CKHAE            ;SO THAT READ SHORTER STOP DISTANCE
                                                        ;SHALL POSITION HEAD IN CLEAN IRG GAP
                                LET PCMDWD := CMDWRD    ;REPOSITION TAPE
027106  013737  003430  003434                                          MOV     CMDWRD,P
                                LET CMDWRD := #RDR      ;BEFORE EXTRA RECORD
027114  012737  104401  003430                                          MOV     #RDR,CMD
                                LET CMDLG := #4         ;BY READING REVERSE
027122  012737  000004  003436                                          MOV     #4,CMDLG
                                LET CMDPKT := CMDWRD CLR.BY #BRF.C
027130  013737  003430  002314                                          MOV     CMDWRD,C
027136  042737  004000  002314                                          BIC     #BRF.C,C
                                LET CMDSAV := CMDPKT    ;THAT RECORD TO ALLOW
027144  013737  002314  003432                                          MOV     CMDPKT,C
                                LET CMDPKT+CP.ADL := DATARD     ;NEXT COMMAND IN THE
027152  013737  003420  002316                                          MOV     DATARD,C
027160  004737  007614          JSR PC,CMDAC            ;TABLE TO BE EXECUTED
027164  004737  007254          JSR PC,EXSUB
027170  004737  017222          JSR PC,CKHAE
027174  000207                  RTS PC
                                .EVEN
                                ENDTST
027176                  L10040:
027176                                                          TRAP    C$ETST
027200                  ENDMOD
```

```
                                        .TITLE PARAMETER CODING

                                        .SBTTL  HARDWARE PARAMETER CODING SECTION

                                                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.
                                        ;--

                                                BGNHRD
                                                                                .WORD L10041-L$H
                                        L$HARD::

                                                GPRMA   TS4ADR,0,0,160002,177564,YES
                                                                                .WORD   T$CODE
                                                                                .WORD   TS4ADR
                                                                                .WORD   T$LOLIM
                                                                                .WORD   T$HILIM
                                                GPRMD   TS4VCT,2,0,777,60,776,YES
                                                                                .WORD   T$CODE
                                                                                .WORD   TS4VCT
                                                                                .WORD   777
                                                                                .WORD   T$LOLIM
                                                                                .WORD   T$HILIM

                                                EXIT HRD
                                                                                .WORD   T$CODE

                                                .NLIST  BEX
027226 051524 051123 040440             TS4ADR: .ASCIZ  /TSSR ADDRESS/
027243    126 041505 047524             TS4VCT: .ASCIZ  /VECTOR/
                                                .LIST   BEX
                                                .EVEN

                                                ENDHRD
                                                                                .EVEN
                                        L10041:
```

```
                                        .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.
                                        ;--

5440
5441  027252                                    BGNSFT
 (3)  027252  000504                                                    .WORD L10042-L$S
 (3)  027254                            L$SOFT::
5442
5449
5450  027254                                    GPRML   CLRM,0,1,YES
 (4)  027254  000130                                                    .WORD   T$CODE
 (4)  027256  030024                                                    .WORD   CLRM
 (4)  027260  000001                                                    .WORD   1
5451  027262                                    GPRML   RRVM,0,400,YES
 (4)  027262  000130                                                    .WORD   T$CODE
 (4)  027264  030043                                                    .WORD   RRVM
 (4)  027266  000400                                                    .WORD   400
5452  027270                                    GPRML   HAEM,2,1,YES
 (4)  027270  001130                                                    .WORD   T$CODE
 (4)  027272  030072                                                    .WORD   HAEM
 (4)  027274  000001                                                    .WORD   1
5453  027276                                    GPRML   RCVERM,2,400,YES
 (4)  027276  001130                                                    .WORD   T$CODE
 (4)  027300  030116                                                    .WORD   RCVERM
 (4)  027302  000400                                                    .WORD   400
5454  027304                                    GPRML   IRECM,4,1,YES
 (4)  027304  002130                                                    .WORD   T$CODE
 (4)  027306  030140                                                    .WORD   IRECM
 (4)  027310  000001                                                    .WORD   1
5455  027312                                    XFERT   NEXTSP
 (5)  027312  004024                                                    .WORD   T$CODE
5456  027314                                    GPRML   BADTM,4,400,YES
 (4)  027314  002130                                                    .WORD   T$CODE
 (4)  027316  030161                                                    .WORD   BADTM
 (4)  027320  000400                                                    .WORD   400
5457  027322                            NEXTSP: GPRML   DINTM,6,1,YES
 (4)  027322  003130                                                    .WORD   T$CODE
 (4)  027324  030206                                                    .WORD   DINTM
 (4)  027326  000001                                                    .WORD   1
5458  027330                                    GPRML   IREM,6,400,YES
 (4)  027330  003130                                                    .WORD   T$CODE
 (4)  027332  030231                                                    .WORD   IREM
 (4)  027334  000400                                                    .WORD   400
5459  027336                                    GPRML   RAMM,10,1,YES
 (4)  027336  004130                                                    .WORD   T$CODE
 (4)  027340  030262                                                    .WORD   RAMM
 (4)  027342  000001                                                    .WORD   1
5460  027344                                    GPRML   CHGM,10,400,YES
 (4)  027344  004130                                                    .WORD   T$CODE
```

```
   (4)  027346  030301                                                    .WORD   CHGM
   (4)  027350  000400                                                    .WORD   400
 5461   027352                  XFERF   ENDSP1
   (5)  027352  127044                                                    .WORD   T$CODE
 5462   027354                  GPRMD   CHARM,14,0,377,0,777,YES
   (4)  027354  006032                                                    .WORD   T$CODE
   (4)  027356  030320                                                    .WORD   CHARM
   (4)  027360  000377                                                    .WORD   377
   (4)  027362  000000                                                    .WORD   T$LOLIM
   (4)  027364  000777                                                    .WORD   T$HILIM
 5463   027366                  GPRMD   CMD2M,16,D,37,1,33,YES
   (4)  027366  007052                                                    .WORD   T$CODE
   (4)  027370  030345                                                    .WORD   CMD2M
   (4)  027372  000037                                                    .WORD   37
   (4)  027374  000001                                                    .WORD   T$LOLIM
   (4)  027376  000033                                                    .WORD   T$HILIM
 5464   027400                  GPRMD   BPCRM,20,D,-1,1,DATCNT,YES
   (4)  027400  010052                                                    .WORD   T$CODE
   (4)  027402  030353                                                    .WORD   BPCRM
   (4)  027404  177777                                                    .WORD   -1
   (4)  027406  000001                                                    .WORD   T$LOLIM
   (4)  027410  004000                                                    .WORD   T$HILIM
 5465   027412                  GPRMD   NUMBM,22,D,-1,1,77777,YES
   (4)  027412  011052                                                    .WORD   T$CODE
   (4)  027414  030365                                                    .WORD   NUMBM
   (4)  027416  177777                                                    .WORD   -1
   (4)  027420  000001                                                    .WORD   T$LOLIM
   (4)  027422  077777                                                    .WORD   T$HILIM
 5466   027424                  GPRMD   PATTM,24,D,17,0,10,YES
   (4)  027424  012052                                                    .WORD   T$CODE
   (4)  027426  030405                                                    .WORD   PATTM
   (4)  027430  000017                                                    .WORD   17
   (4)  027432  000000                                                    .WORD   T$ JLIM
   (4)  027434  000010                                                    .WORD   T$HILIM
 5467   027436                  GPRMD   CMD3M,26,D,37,1,33,YES
   (4)  027436  013052                                                    .WORD   T$CODE
   (4)  027440  030420                                                    .WORD   CMD3M
   (4)  027442  000037                                                    .WORD   37
   (4)  027444  000001                                                    .WORD   T$LOLIM
   (4)  027446  000033                                                    .WORD   T$HILIM
 5468   027450                  GPRMD   BPCRM,30,D,-1,1,DATCNT,YES
   (4)  027450  014052                                                    .WORD   T$CODE
   (4)  027452  030353                                                    .WORD   BPCRM
   (4)  027454  177777                                                    .WORD   -1
   (4)  027456  000001                                                    .WORD   T$LOLIM
   (4)  027460  004000                                                    .WORD   T$HILIM
 5469   027462                  GPRMD   NUMBM,32,D,-1,1,77777,YES
   (4)  027462  015052                                                    .WORD   T$CODE
   (4)  027464  030365                                                    .WORD   NUMBM
   (4)  027466  177777                                                    .WORD   -1
   (4)  027470  000001                                                    .WORD   T$LOLIM
   (4)  027472  077777                                                    .WORD   T$HILIM
 5470   027474                  GPRMD   PATTM,34,D,17,0,10,YES
   (4)  027474  016052                                                    .WORD   T$CODE
   (4)  027476  030405                                                    .WORD   PATTM
   (4)  027500  000017                                                    .WORD   17
```

```
                                                                        .WORD   T$LOLIM
                                                                        .WORD   T$HILIM
                        GPRMD   CMD4M,36,D,37,1,33,YES
                                                                        .WORD   T$CODE
                                                                        .WORD   CMD4M
                                                                        .WORD   37
                                                                        .WORD   T$LOLIM
                                                                        .WORD   T$HILIM
                        GPRMD   BPCRM,40,D,-1,1,DATCNT,YES
                                                                        .WORD   T$CODE
                                                                        .WORD   BPCRM
                                                                        .WORD   -1
                                                                        .WORD   T$LOLIM
                                                                        .WORD   T$HILIM
                        GPRMD   NUMBM,42,D,-1,1,77777,YES
                                                                        .WORD   T$CODE
                                                                        .WORD   NUMBM
                                                                        .WORD   -1
                                                                        .WORD   T$LOLIM
                                                                        .WORD   T$HILIM
                        GPRMD   PATTM,44,D,17,0,10,YES
                                                                        .WORD   T$CODE
                                                                        .WORD   PATTM
                                                                        .WORD   17
                                                                        .WORD   T$LOLIM
                                                                        .WORD   T$HILIM
                        GPRMD   CMD5M,46,D,37,1,33,YES
                                                                        .WORD   T$CODE
                                                                        .WORD   CMD5M
                                                                        .WORD   37
                                                                        .WORD   T$LOLIM
                                                                        .WORD   T$HILIM
                        GPRMD   BPCRM,50,D,-1,1,DATCNT,YES
                                                                        .WORD   T$CODE
                                                                        .WORD   BPCRM
                                                                        .WORD   -1
                                                                        .WORD   T$LOLIM
                                                                        .WORD   T$HILIM
                        GPRMD   NUMBM,52,D,-1,1,77777,YES
                                                                        .WORD   T$CODE
                                                                        .WORD   NUMBM
                                                                        .WORD   -1
                                                                        .WORD   T$LOLIM
                                                                        .WORD   T$HILIM
                        GPRMD   PATTM,54,D,17,0,10,YES
                                                                        .WORD   T$CODE
                                                                        .WORD   PATTM
                                                                        .WORD   17
                                                                        .WORD   T$LOLIM
                                                                        .WORD   T$HILIM
                        XFER    ENDSP2
                                                                        .WORD   T$CODE
ENDSP1:                 XFER    ENDSP
                                                                        .WORD   T$CODE
ENDSP2:                 GPRMD   CMD6M,56,D,37,1,33,YES
                                                                        .WORD   T$CODE
```

```
  (4)  027634  030442                                         .WORD   CMD6M
  (4)  027636  000037                                         .WORD   37
  (4)  027640  000001                                         .WORD   T$LOLIM
  (4)  027642  000033                                         .WORD   T$HILIM
 5482  027644                 GPRMD   BPCRM,60,D,-1,1,DATCNT,YES
  (4)  027644  030052                                         .WORD   T$CODE
  (4)  027646  030353                                         .WORD   BPCRM
  (4)  027650  177777                                         .WORD   -1
  (4)  027652  000001                                         .WORD   T$LOLIM
  (4)  027654  004000                                         .WORD   T$HILIM
 5483  027656                 GPRMD   NUMBM,62,D,-1,1,77777,YES
  (4)  027656  031052                                         .WORD   T$CODE
  (4)  027660  030365                                         .WORD   NUMBM
  (4)  027662  177777                                         .WORD   -1
  (4)  027664  000001                                         .WORD   T$LOLIM
  (4)  027666  077777                                         .WORD   T$HILIM
 5484  027670                 GPRMD   PATTM,64,D,17,0,10,YES
  (4)  027670  032052                                         .WORD   T$CODE
  (4)  027672  030405                                         .WORD   PATTM
  (4)  027674  000017                                         .WORD   17
  (4)  027676  000000                                         .WORD   T$LOLIM
  (4)  027700  000010                                         .WORD   T$HILIM
 5485  027702                 GPRMD   CMD7M,66,D,37,1,33,YES
  (4)  027702  033052                                         .WORD   T$CODE
  (4)  027704  030450                                         .WORD   CMD7M
  (4)  027706  000037                                         .WORD   37
  (4)  027710  000001                                         .WORD   T$LOLIM
  (4)  027712  000033                                         .WORD   T$HILIM
 5486  027714                 GPRMD   BPCRM,70,D,-1,1,DATCNT,YES
  (4)  027714  034052                                         .WORD   T$CODE
  (4)  027716  030353                                         .WORD   BPCRM
  (4)  027720  177777                                         .WORD   -1
  (4)  027722  000001                                         .WORD   T$LOLIM
  (4)  027724  004000                                         .WORD   T$HILIM
 5487  027726                 GPRMD   NUMBM,72,D,-1,1,77777,YES
  (4)  027726  035052                                         .WORD   T$CODE
  (4)  027730  030365                                         .WORD   NUMBM
  (4)  027732  177777                                         .WORD   -1
  (4)  027734  000001                                         .WORD   T$LOLIM
  (4)  027736  077777                                         .WORD   T$HILIM
 5488  027740                 GPRMD   PATTM,74,D,17,0,10,YES
  (4)  027740  036052                                         .WORD   T$CODE
  (4)  027742  030405                                         .WORD   PATTM
  (4)  027744  000017                                         .WORD   17
  (4)  027746  000000                                         .WORD   T$LOLIM
  (4)  027750  000010                                         .WORD   T$HILIM
 5489  027752                 GPRMD   CMD8M,76,D,37,1,33,YES
  (4)  027752  037052                                         .WORD   T$CODE
  (4)  027754  030456                                         .WORD   CMD8M
  (4)  027756  000037                                         .WORD   37
  (4)  027760  000001                                         .WORD   T$LOLIM
  (4)  027762  000033                                         .WORD   T$HILIM
 5490  027764                 GPRMD   BPCRM,100,D,-1,1,DATCNT,YES
  (4)  027764  040052                                         .WORD   T$CODE
  (4)  027766  030353                                         .WORD   BPCRM
  (4)  027770  177777                                         .WORD   -1
```

```
   (4)  027772  000001                                                  .WORD   T$LOLIM
   (4)  027774  004000                                                  .WORD   T$HILIM
5491    027776                      GPRMD   NUMBM,102,D,-1,1,77777,YES
   (4)  027776  041052                                                  .WORD   T$CODE
   (4)  030000  030365                                                  .WORD   NUMBM
   (4)  030002  177777                                                  .WORD   -1
   (4)  030004  000001                                                  .WORD   T$LOLIM
   (4)  030006  077777                                                  .WORD   T$HILIM
5492    030010                      GPRMD   PATTM,104,D,17,0,10,YES
   (4)  030010  042052                                                  .WORD   T$CODE
   (4)  030012  030405                                                  .WORD   PATTM
   (4)  030014  000017                                                  .WORD   17
   (4)  030016  000000                                                  .WORD   T$LOLIM
   (4)  030020  000010                                                  .WORD   T$HILIM
5493    030022              ENDSP:
5494    030022                      XFER    JMPMSG
   (5)  030022  176004                                                  .WORD   T$CODE
```

```
5496
5503                                                    .NLIST  BEX
5504  030024  046103  040505  020122          CLRM:   .ASCIZ  /CLEAR COUNTERS/
5505  030043     122  051505  052105          RRVM:   .ASCIZ  /RESET RANDOM VARIABLES/
5506  030072  040510  052114  040440          HAEM:   .ASCIZ  /HALT AFTER EACH CMD/
5507  030116  051120  047111  020124          RCVERM: .ASCIZ  /PRINT SOFT ERRORS/
5508  030140  047111  044510  044502          IRECM:  .ASCIZ  /INHIBIT RECOVERY/
5509  030161     102  042101  052040          BADTM:  .ASCIZ  /BAD TAPE SPOT DETECT/
5510  030206  044504  040523  046102          DINTM:  .ASCIZ  /DISABLE INTERRUPTS/
5511  030231     111  044 16  041111          IREM:   .ASCIZ  /INHIBIT RFC ERROR REPORT/
5512  030262  033515  032464  020064          RAMM:   .ASCIZ  /M7454 RAM DUMP/
5513  030301     103  040510  043516          CHGM:   .ASCIZ  /CHANGE CMD SEQ/
5514  030320  044103  051 01  041501          CHARM:  .ASCIZ  /CHARACTERISTICS CODE/
5515  030345     103  042 15  031057          CMD2M:  .ASCIZ  "CMD/2"
5516  030353     102  043122  041440          BPCRM:  .ASCIZ  /BRF COUNT/
5517  030365     043  047440  020106          NUMBM:  .ASCIZ  /# OF OPERATIONS/
5518  030405     120  052101  042524          PATTM:  .ASCIZ  /PATTERN/
5519                                                    .LIST   BEX
5520            030416                                  .EVEN
5521  030416                                    JMPMSG:
5522  030416                                            EXIT SFT
 (7)  030416  023004                                                .WORD   T$CODE
5523
5524                                                    .NLIST  BEX
5525  030420  046503  027504  000063          CMD3M:  .ASCIZ  "CMD/3"
5526  030426  046503  027504  000064          CMD4M:  .ASCIZ  "CMD/4"
5527  030434  046503  027504  000065          CMD5M:  .ASCIZ  "CMD/5"
5528  030442  046503  027504  000066          CMD6M:  .ASCIZ  "CMD/6"
5529  030450  046503  027504  000067          CMD7M:  .ASCIZ  "CMD/7"
5530  030456  046503  027504  000070          CMD8M:  .ASCIZ  "CMD/8"
5531                                                    .LIST   BEX
5532  030464                                            ENDSFT
 (2)                                                                .EVEN
 (3)  030464                                L10042:
5533
5534                                        ;*****************************************************************************
5535                                        ;*****************************************************************************
5536                                        ;       PATCH AREA
5537
5538  030464  000100                        PATCH:: .BLKW   64.
5539                                        ;*****************************************************************************
5540                                        ;*****************************************************************************
5541
5542  030664                                            LASTAD
 (2)                                                                .EVEN
 (2)  030664  030700                                                .WORD T$FREE
 (2)  030666  000004                                                .WORD T$SIZE
 (3)  030670                                L$LAST::
5543  030670                                            ENDMOD
```

```
 5545                                   .SBTTL  HARD CODED P-TBL
 5546
 5547                                   ;++
 5548                                   ;DIAG IS PRE-PARAMETERIZED PER TBL
 5549                                   ;--
 5550
 5551  030670                                   BGNSETUP 1
 5552  030670                                     BGNPTAB
  (4)  030670  000000                                                .WORD   0
  (3)  030672  000002                                                .WORD   L10045-.
  (3)  030674                           L10043:
 5553  030674  172522                              172522
 5554  030676  000224                              224
 5555  030700                                     ENDPTAB
  (3)  030700                           L10045:
 5556  030700                                   ENDSETUP
 5557
 5558          000001                   .END
```

```
ACK.C = 100000 G    1958# 2043 2048 2051 2054 2057 2060 2063 2066 2069 2072 2075 2078
                    2081 2084 2087 2090 2093 2096 2099 2102 2105 2108 2111 2114 2116
ADR   = 000020 G    1920# 4318
ALLEOT  003532 G    2318# 2726 2743 2759 2771 3810 4941* 4954* 5176* 5184* 5216* 5224* 5309*
                    5314*
ASSEMB= 000010      1728
ATTNM   004421 G    2481# 3271
AUDRPM  004731 G    2491# 4311
AUTODM  023006      4398 4422#
BADTM   030161      5456 5509#
BADTSW  002211 G    1850# 3349 4134
BFSEQ   024154      4586 4603 4611 4618 4625 4632 4639 4646 4653 4660 4667 4714#
BFSEQ0  024200      4585 4725#
BFSEQ1  024252      4602 4747#
BFSEQ2  024264      4610 4753#
BFSEQ3  024356      4617 4783#
BFSEQ4  024430      4624 4805#
BFSEQ5  024472      4631 4823#
BFSEQ6  024544      4638 4845#
BFSEQ7  024576      4645 4859#
BFSEQ8  024630      4652 4873#
BFSEQ9  024662      4659 4887#
BFSE10  024704      4666 4897#
BGNFLG= 003466      2292# 3109
BINC    015150      3673 3711#
BIT0  = 000001 G    1920# 3161 4231
BIT00 = 000001 G    1920# 2608 3909
BIT01 = 000002 G    1920#
BIT02 = 000004 G    1920#
BIT03 = 000010 G    1920#
BIT04 = 000020 G    1920#
BIT05 = 000040 G    1920#
BIT06 = 000100 G    1920#
BIT07 = 000200 G    1920#
BIT08 = 000400 G    1920#
BIT09 = 001000 G    1920#
BIT1  = 000002 G    1920# 4231
BIT10 = 002000 G    1920#
BIT11 = 004000 G    1920#
BIT12 = 010000 G    1920#
BIT13 = 020000 G    1920#
BIT14 = 040000 G    1920#
BIT15 = 100000 G    1920#
BIT2  = 000004 G    1920#
BIT3  = 000010 G    1920#
BIT4  = 000020 G    1920#
BIT5  = 000040 G    1920#
BIT6  = 000100 G    1920#
BIT7  = 000200 G    1920#
BIT8  = 000400 G    1920#
BIT9  = 001000 G    1920#
BOE   = 000400 G    1920#
BORERS  014374 G    3377 3553 3620# 5117
BPCRM   030353      5464 5468 5472 5476 5482 5486 5490 5516#
BRCPK   002330 G    2148#
BRFCNT  003426 G    2270# 2674* 2675* 2676 2677* 2679 2830* 2896 3674 3675 3889 3894
```

```
BRF.C = 004000 G  1963# 2051  2054  2057  2060  2063  2066  2069  2072  2075  2078  2081  2096
                  2099  2884  3166  3217  3622  3793  5358
BTADDR  002550 G  2221# 2736  3575  4149  5091
BTMSG1  014170    3572  3608#
BTMSG2  014255    3376  3609#
BTMSG3  014325    3592  3610#
BTPT    003520 G  2309# 2736* 2737* 3373  3375  3575* 3576  3577* 3579  4149* 4150  4157  5091*
                  5092*
BTRPT   017762    4135  4148#
BTO     003000 G  2221  2238#
BT1     003052 G  2222  2239#
BT2     003124 G  2223  2240#
BT3     003176 G  2224  2241#
CHAR    002220 G  1858# 5245
CHARM   030320    5462  5514#
CHGFLG  002215 G  1854# 4357*
CHGM    030301    5460  5513#
CHKERR  012004 G  3094  3195#
CH.EAI= 000040 G  1858  1986# 1988  2178  2345
CH.ERI= 000020 G  1987#
CH.ESS= 000200 G  1985#
CKDATA  016200 G  3856  3869  3889#
CKDCNT  016610    3898* 3907  3909  3933  3934* 3936  3942#
CKDFF   016612    3899* 3917  3922* 3927  3935  3936  3943#
CKHAE   017222 G  2684  3629  3638  3656  3827  4050# 5291  5353  5363
CKHRTN  017310    4053  4064#
CLN   = 101012 G  2111# 2400  4887
CLRERR  011502 G  3090  3098  3109# 4359
CLRFLG  002204 G  1845# 4237  4238*
CLRM    030024    5450  5504#
CMDAC   007614 G  2668  2788# 3626  3635  3652  3801  4081  5106  5286  5361
CMDASC  003734 G  2410# 2817
CMDD    002222 G  1859# 5246
CMDLG   003436 G  2274# 2734  2827* 2839* 2846* 2848* 2872* 3349  3393  3507  3515  3625* 3651*
                  3670  3774* 3781* 5310  5357*
CMDPKM  004166 G  2473# 4232
CMDPKT  002314 G  2134# 2543  2551  2552  2679* 2828* 2829* 2831  2836* 2838  2844* 2845  2854*
                  2855* 2861* 2862* 2870* 2873  2875* 2880  2882* 2884* 2885  3035* 3045  3231*
                  3360  3361  3396* 3404* 3622* 3623  3624* 3633* 3634  3647* 3648  3649* 3650*
                  3793* 3795* 3797  3798* 3896  4231  4684* 4687* 4689* 4693* 4696* 4698* 5358*
                  5359  5360*
CMDSAV  003432 G  2272# 2885* 3231  3623* 3634* 3648* 3797* 5359*
CMDSEQ  003544 G  2344# 2635  2664  4073  4079  4714  4939  4950* 4951* 4956  5094  5101  5118
                  5124  5174  5217  5245* 5256  5270
CMDSE2  003554 G  2348#
CMDTBL  003646 G  2377# 2811  5338
CMDWRD  003430 G  2271# 2606  2670  2688  2693  2704  2723  2739  2756  2767  2788  2879  2880*
                  3010  3040  3070  3073  3163  3166  3167  3217  3288  3359  3620  3621* 3622
                  3631  3632* 3633  3645  3646* 3772  3773* 3779  3780* 3793  3807  3818  3832
                  4054  4682* 4691* 5288  5295  5300  5317  5355  5356* 5358
CMD.C0= 000001 G  1981# 2048  2051  2054  2057  2060  2069  2072  2075  2078  2081  2090  2093
                  2102  2105  2693  2739  2767  2842  2868  3818  5300
CMD.C1= 000002 G  1980# 2048  2087  2102  2108  2111  2116  2859
CMD.C2= 000004 G  1979# 2057  2060  2081  2102  2114  2116  2852  2859  2868
CMD.C3= 000010 G  1978# 2048  2063  2066  2084  2087  2090  2093  2096  2099  2102  2105  2108
                  2111  2834
```

```
CMD.C4= 000020 G  1976#
CMD2M   030345    5463    5515#
CMD3M   030420    5467    5525#
CMD4M   030426    5471    5526#
CMD5M   030434    5475    5527#
CMD6M   030442    5481    5528#
CMD7M   030450    5485    5529#
CMD8M   030456    5489    5530#
CMPDAT  003410 G  2263#   3769*   3777*   3863*   3866*   3867    3870*   3873*
CNTBGN= 002560    2228#   2251    3673
CNTEND= 003330    2248#   2251
CNTLEN= 000550 G  2251#   4240
CODELM  004056 G  2470#   4594
COUNTE= 050416    4321#   4327    4339
CP.ADH= 000004 G  2023#
CP.ADL= 000002 G  2022#   2844*   2854*   2862*   2870*   3624*   3649*   3798*   5360*
CP.CMD= 000000 G  2021#
CP.CNT= 000006 G  2024#   2679*   2829*   2855*   2861*   3361    3650*   3896    4684*   4687*   4689*   4693*   4696*
                  4698*
CRLF    005277 G  2499#   4170
CRLFSP  005302 G  2500#   4156    4164
CTCC    003460 G  2284#   2551    3143*   3161    3200    3212
CVC.C = 040000 G  1959#   2043    2114
C$AU  = 000052    1728#   4551
C$AUTO= 000061    1728#   4420
C$BRK = 000022    1728#   3069    3125    4313    4336
C$BSEG= 000004    1728#
C$BSUB= 000002    1728#   4583    4600    4608    4615    4622    4629    4636    4643    4650    4657    4664    4680
C$CEFG= 000045    1728#
C$CLCK= 000062    1728#
C$CLEA= 000012    1728#   4474
C$CLOS= 000035    1728#
C$CLP1= 000006    1728#
C$CVEC= 000036    1728#   4395    4456
C$DCLN= 000044    1728#   3957    4314    4353
C$DODU= 000051    1728#   4016    4401
C$DRPT= 000024    1728#   2752    4010
C$DU  = 000053    1728#   4508
C$EDIT= 000003    1728#   1775
C$ERDF= 000055    1728#   3005    3030    3049    3087    3271    3322    3461    3476    3508    3518    4002
C$ERHR= 000056    1728#   3223    3255    3303    3526    3891    3895    3921
C$ERRO= 000060    1728#
C$ERSF= 000054    1728#   4232
C$ERSO= 000057    1728#   3351    3400    3430
C$ESCA= 000010    1728#
C$ESEG= 000005    1728#
C$ESUB= 000003    1728#   4598    4606    4613    4620    4627    4634    4641    4648    4655    4662    4672    4701
C$ETST= 000001    1728#   4908    5051    5150    5193    5230    5366
C$EXIT= 000032    1728#   4362    4460    4706    4961    5147    5189    5226    5326
C$GETB= 000026    1728#
C$GETW= 000027    1728#
C$GMAN= 000043    1728#   4059
C$GPHR= 000042    1728#   4292    4529
C$GPLO= 000030    1728#
C$GPRI= 000040    1728#
C$INIT= 000011    1728#   4378
```

```
C$INLP= 000020           1728#
C$MANI= 000050           1728#  4052
C$MEM = 000031           1728#  4349
C$MSG = 000023           1728#  2538  2572
C$OPEN= 000034           1728#
C$PNTB= 000014           1728#  2524  2525  2534  2541  2542  2548  3376  3572  3592  3702  3892  3896
                         3936
C$PNTF= 000017           1728#  4311  4329  4332  4352  4398  4408  4413  4493  4594  5068
C$PNTS= 000016           1728#  4128  4129  4130  4131  4132  4133  4137  4148  4151  4156  4160  4164
                         4170
C$PNTX= 000015           1728#  2550  2551  2552  2553  3485  3496  3747  3753  3928  4037  4038
C$QIO = 000377           1728#
C$RDBU= 000007           1728#
C$REFG= 000047           1728#  4250  4265  4273  4276
C$RESE= 000033           1728#
C$REVI= 000003           1728#  1775
C$RFLA= 000021           1728#  4270
C$RPT = 000025           1728#  4201
C$SEFG= 000046           1728#
C$SPRI= 000041           1728#  4317
C$SVEC= 000037           1728#  4297  4393  4533
C$TPRI= 000013           1728#
DATARD  003420 G         2267#  2607  2622  2844  3798  3845  3846  3902  4350*  5360
DATAWT  003416 G         2266#  2287  2870  2899  3038*  3085*  3181*  3437*  3649  3901  4349*  4350  4351
                         4674   4675
DATCNT= 004000 G         1864   1868  1872  1880  1884  2033#  2353  2357  2361  4350  4351  4754  4758
                         4762   4766  4770  4774  4778  4788  4850  4854  4860  4864  4868  4874  4878
                         4882   4919  4927  4995  5001  5041  5163  5211  5219  5464  5468  5472  5476
                         5482   5486  5490
DATRAT  003412 G         2264#  3859*  3861  3862*  3867
DEVTBL  002536 G         2212#  2524  2541  2687  2722  2755  3013  3063  3195  3345  3369  3406  3548
                         3556   3564  3644  3654  3776  3806  3828  3953  3956  3959  3976  3977  4014
                         4017   4127  4128  4256*  4262*  4289  4320  4329  4332  4391  4399  4408  4413
                         4454   4492*  4528*  4589  4594  5089  5109  5114  5130  5294
DFPTBL  002176 G         1818#
DFTSCH= 000040 G         1988#  2637
DIA   = 100006 G         2116#  2402
DIABLK= 003416 G         2287#  2862
DIACNT= 000020 G         2032#  2861
DIAGMC= 000000           1728
DINT    002212 G         1851#  3026  3076
DINTM   030206           5457   5510#
DLY   = 000020 G         2120#  2404
DLY.C = 000020 G         1977#  2120  5275
DOAGIN  003340 G         2256#  5072*  5131*
DRI   = 100013 G         2048#  2377  4729
DROPDM  004702 G         2490#  4493
DROPED  003530 G         2316#  3701  3850  3853  3951*  3972*  4007*  4025*
DROPN   017136           4014*  4029#  4399*  4493
DROPU   016716 G         3006   3031  3050  3088  3324  3380  3463  3478  3510  3520  3989#
DROPUA  017046           4011#  4409  4414
DRORTN  017124           4025#
DTAERM  005474 G         2518#  3891  3895  3921
DTAER2  004763 G         2492#  3928
DTAER3  005032 G         2493#  3936
DTAER4  005074 G         2494#  3892
```

```
DTAER5  005115 G        2495#   3896
EF.CON= 000036 G        1920#
EF.NEW= 000035 G        1920#   4273
EF.PWR= 000034 G        1920#   4265
EF.RES= 000037 G        1920#   4276
EF.STA= 000040 G        1920#   4250
EINC    015156          3694    3715#
END   = 177777 G        2122#   2216    2271    2272    2273    2369    2370    2371    2372    2373    2405    2665    2687
                        2722    2755    3013    3806    3956    4078    4127    4289    4320    4391    4454    4589    4715
                        4718    4745    4751    4781    4803    4821    4843    4857    4871    4885    4895    4905    4937
                        4952    4958    5078    5089    5099    5123    5172    5186    5214    5222    5260    5294
ENDERF= 003500          2303#   3089    3097
ENDFLG= 003536          2324#   4358
ENDSP   030022          5480    5493#
ENDSP1  027630          5461    5480#
ENDSP2  027632          5479    5481#
EOTFLG  003510 G        2308#   2689    2693    2724    2725    2734    2739    2757    2758    2767    3150*   3173    3808
                        3809    3818    3854    5296    5300
ERCVER  002207 G        1848#   3350    3399    3571    3591    3701
ERLOG   003474 G        2298#   3668    3669*
ERRREC  003477 G        2301#   3036    3201    3297    3350    3355*   3410*   3435*
ERS   = 100411 G        2105#   2398    3632    4791
ERSFLG  003534 G        2320#   3584*   3630    3639*   5116*
EVL   = 000004 G        1920#
EXALL   006760 G        2664#   4587    4604    4612    4619    4626    4633    4640    4647    4654    4661    4668    4670
                        4938    4953    4959    5079    5173    5187    5215    5223
EXARTN  007252          2701    2710#
EXCRTN  011164          3007    3032    3052#
EXCUTE  010576 G        2727    2730    2741    2744    3000#   3412    3439    3627    3636    3653    3849    4082    5107
EXPBOT  003522 G        2310#   3283    4581*   4918*   5074*   5140*   5162*   5206*   5242*
EXSUB   007254 G        2683    2721#   5287    5352    5362
E$END = 002100          1728#
E$LOAD= 000035          1728#   1775
FATSM   004457 G        2483#   3476
FIRSTU  016614 G        2686    2721    2754    3012    3805    3951#   4126    4287    4319    4390    4453    4588    5081
                        5293
FMT.C0= 000040 G        1972#   2043
FMT.C1= 000100 G        1969#   2043
FTLCNT  003320 G        2247#   3990*   4137    4397*   4407*   4412*
FUNRM   004437 G        2482#   3322
F$AU  = 000015          1728#   4519    4551
F$AUTO= 000020          1728#   4388    4420
F$BGN = 000040          1728#   1758    1888    1913    2518    2535    2540    2554    2582    2586    2590    2594    4087
                        4111    4118    4141    4207    4220    4362    4388    4444    4460    4484    4494    4519    4536
                        4553    4576    4578    4583    4598    4600    4606    4608    4613    4615    4620    4622    4627
                        4629    4634    4636    4641    4643    4648    4650    4655    4657    4662    4664    4672    4680
                        4701    4706    4908    4915    4961    5051    5066    5147    5150    5159    5189    5193    5203
                        5226    5230    5239    5326    5366    5367    5392    5403    5413    5441    5522    5543    5551
                        5552    5555    5556
F$CLEA= 000007          1728#   4444    4474
F$DU  = 000016          1728#   4484    4508
F$END = 000041          1728#   1758    1888    1913    2535    2538    2554    2572    2584    2588    2592    2596    4087
                        4111    4141    4201    4362    4378    4420    4460    4474    4494    4508    4536    4551    4553
                        4576    4578    4583    4598    4600    4606    4608    4613    4615    4620    4622    4627    4629
                        4634    4636    4641    4643    4648    4650    4655    4657    4662    4664    4672    4680    4701
                        4706    4908    4915    4961    5051    5066    5147    5150    5159    5189    5193    5203    5226
```

```
                    5230  5239  5326  5366  5367  5392  5413  5427  5522  5532  5543  5551  5552
                    5555  5556
F$HARD= 000004      1728# 5403  5413  5427  5455  5461  5479  5480  5494  5522
F$HW  = 000013      1728# 1818  1829
F$INIT= 000006      1728# 4220  4378
F$JMP = 000050      1728# 2535  2554  4141  4362  4460  4494  4536  4706  4961  5147  5189  5226
                    5326  5413  5522
F$MOD = 000000      1728# 1758  1888  1913  4087  4111  4553  4576  5367  5392  5543
F$MSG = 000011      1728# 2518  2538  2540  2572
F$PROT= 000021      1728# 4207  4211
F$PWR = 000017      1728#
F$RPT = 000012      1728# 4118  4201
F$SEG = 000003      1728#
F$SOFT= 000005      1728# 5441  5455  5461  5479  5480  5494  5522  5532
F$SRV = 000010      1728# 2582  2584  2586  2588  2590  2592  2594  2596
F$SUB = 000002      1728# 4583  4598  4600  4606  4608  4613  4615  4620  4622  4627  4629  4634
                    4636  4641  4643  4648  4650  4655  4657  4662  4664  4672  4680  4701
F$SW  = 000014      1728# 1838  1887
F$TEST= 000001      1728# 4578  4908  4915  5051  5066  5150  5159  5193  5203  5230  5239  5366
GCMDA   007670 G    2789  2794  2810#  4055
GENPAT  010300 G    2871  2895#
GES   = 100017 G    2102# 2142  2397  4737  5288
GETSTM  005243 G    2498# 4037
GIT     010572      2982  2987#
GOWAIT  011166 G    2760  2763  2769  2772  3063#  3413  3440  3628  3637  3655  3851  4083  5108
GSCPK   002324 G    2142# 3998  4323  4403
G$CNTO= 000200      1728#
G$DELM= 000372      1728# 3042  3071  3074  4233  4312  4324  4335  5279
G$DISP= 000003      1728#
G$EXCP= 000400      1728#
G$HILI= 000002      1728#
G$LOLI= 000001      1728#
G$NO  = 000000      1728#
G$OFFS= 000400      1728# 4059  5410  5411  5450  5451  5452  5453  5454  5456  5457  5458  5459
                    5460  5462  5463  5464  5465  5466  5467  5468  5469  5470  5471  5472  5473
                    5474  5475  5476  5477  5478  5481  5482  5483  5484  5485  5486  5487  5488
                    5489  5490  5491  5492
G$OFSI= 000376      1728# 4059  5410  5411  5450  5451  5452  5453  5454  5456  5457  5458  5459
                    5460  5462  5463  5464  5465  5466  5467  5468  5469  5470  5471  5472  5473
                    5474  5475  5476  5477  5478  5481  5482  5483  5484  5485  5486  5487  5488
                    5489  5490  5491  5492
G$PRMA= 000001      1728# 5410
G$PRMD= 000002      1728# 5411  5462  5463  5464  5465  5466  5467  5468  5469  5470  5471  5472
                    5473  5474  5475  5476  5477  5478  5481  5482  5483  5484  5485  5486  5487
                    5488  5489  5490  5491  5492
G$PRML= 000000      1728# 4059  5450  5451  5452  5453  5454  5456  5457  5458  5459  5460
G$RADA= 000140      1728#
G$RADB= 000000      1728#
G$RADD= 000040      1728# 5463  5464  5465  5466  5467  5468  5469  5470  5471  5472  5473  5474
                    5475  5476  5477  5478  5481  5482  5483  5484  5485  5486  5487  5488  5489
                    5490  5491  5492
G$RADL= 000120      1728# 4059  5450  5451  5452  5453  5454  5456  5457  5458  5459  5460
G$RADO= 000020      1728# 5410  5411  5462
G$XFER= 000004      1728# 5413  5455  5461  5479  5480  5494  5522
G$YES = 000010      1728# 4059  5410  5411  5450  5451  5452  5453  5454  5456  5457  5458  5459
                    5460  5462  5463  5464  5465  5466  5467  5468  5469  5470  5471  5472  5473
```

```
                        5474    5475    5476    5477    5478    5481    5482    5483    5484    5485    5486    5487    5488
                        5489    5490    5491    5492
HAE       002206 G      1847#   4050
HAEM      030072        5452    5506#
HALTM     004126 G      2472#   4056*   4057*   4058*   4059
HELP   =  000000        1716#   1723    1750    1767    1777    1793    1820    1839    1922    2458    2519    4120    4173
                        4222    4363    4446    4461    4486    4495    4521    4537    5405    5415    5443    5497
HERE      003341 G      2257#   4254*   5067    5069*
HOE    =  100000 G      1920#
HRDCNT    003310 G      2246#   3222*   3254*   3302*   3920*   4137
IBE    =  010000 G      1920#
IDU    =  000040 G      1920#
IER    =  020000 G      1920#
IE.C   =  000200 G      1968#   2043    3035
INIT10    021222        4231#   4234
INIT15    021442        4251    4265#
INIT16    021462        4266    4270#
INTFLG    003500 G      2307#   2583*   2587*   2591*   2595*   3028    3034*   3077    4534*
INTPRI=   000340 G      1775    2028#   4297    4533
IRE       003527 G      2315#   3221    3296    5243*
IREC      002210 G      1849#   3354    3405    3434    3700    4018    4318
IRECM     030140        5454    5508#
IREM      030231        5458    5511#
ISR    =  000100 G      1920#
IXE    =  004000 G      1920#
I$AU   =  000041        1728#   4519#   4551#
I$AUTO=   000041        1728#   4388#   4420#
I$CLN  =  000041        1728#   4444#   4460    4474#
I$DU   =  000041        1728#   4484#   4508#
I$HRD  =  000041        5403#   5427#
I$INIT=   000041        1728#   4220#   4362    4378#
I$MOD  =  000041        1728#   1758#   1888#   1913#   4087#   4111#   4553#   4576#   5367#   5392#   5543#
I$MSG  =  000041        1728#   2518#   2538#   2540#   2572#
I$PROT=   000040        1728#   4207#
I$PTAB=   000041        1728#   5552#   5555#
I$PWR  =  000041        1728#
I$RPT  =  000041        1728#   4118#   4201#
I$SEG  =  000041        1728#   4578    4583    4600    4608    4615    4622    4629    4636    4643    4650    4657    4664
                        4680    4915    5066    5159    5203    5239
I$SETU=   000041        1728#   5551#   5552    5556#
I$SFT  =  000041        5441#   5532#
I$SRV  =  000041        1728#   2582#   2584#   2586#   2588#   2590#   2592#   2594#   2596#
I$SUB  =  000041        1728#   4578    4583#   4598#   4600#   4606#   4608#   4613#   4615#   4620#   4622#   4627#   4629#
                        4634#   4636#   4641#   4643#   4648#   4650#   4655#   4657#   4662#   4664#   4672#   4680#   4701#
                        4915    5066    5159    5203    5239
I$TST  =  000041        1728#   4578#   4583    4600    4608    4615    4622    4629    4636    4643    4650    4657    4664
                        4680    4706    4908#   4915#   4961    5051#   5066#   5147    5150#   5159#   5189    5193#   5203#
                        5226    5230#   5239#   5326    5366#
JLOC      003454 G      2281#   5264*   5271*
JLOOP     003452 G      2280#   5254*   5265    5269*
JMP    =  000040 G      2118#   2403
JMPMSG    030416        5494    5521#
JMP.C  =  000040 G      1973#   2118    5261
J$JMP  =  000167        1728#   2535    2554    4141    4494    4536
LENMSK    003440 G      2275#   2675    4920*   5164*
LOE    =  040000 G      1920#
```

```
LOG     014664 G          3096    3668#   4008
LOT   = 000010 G          1920#
L$ACP   002110 G          1775#
L$APT   002036 G          1775#
L$AU    023246 G          1775    4519#
L$AUT   002070 G          1775#
L$AUTO  022516 G          1775    4388#
L$CCP   002106 G          1775#
L$CLEA  023140 G          1775    4444#
L$CO    002032 G          1775#
L$DEPO  002011 G          1775#
L$DESC  002140 G          1775    1806#
L$DESP  002076 G          1775#
L$DEVP  002060 G          1775#
L$DISP  002124 G          1775    1791#
L$DLY   002116 G          1775#   3042    3071    3074    4233    4312    4324    4335    5279
L$DTP   002040 G          1775#
L$DTYP  002034 G          1775#
L$DU    023202 G          1775    4484#
L$DUT   002072 G          1775#
L$DVTY  002166 G          1775    1807#
L$EF    002052 G          1775#
L$ENVI  002044 G          1775#
L$ETP   002102 G          1775#
L$EXP1  002046 G          1775#
L$EXP4  002064 G          1775#
L$EXP5  002066 G          1775#
L$HARD  027202 G          1775    5403#
L$HIME  002120 G          1775#
L$HPCP  002016 G          1775#
L$HPTP  002022 G          1775#
L$HW    002176 G          1775    1818#
L$ICP   002104 G          1775#
L$INIT  021222 G          1775    4220#
L$LADP  002026 G          1775#
L$LAST  030670 G          1775    5542#   5556
L$LOAD  002100 G          1775#
L$LUN   002074 G          1775#   3959*   3977*
L$MREV  002050 G          1775#
L$NAME  002000 G          1775#
L$PRIO  002042 G          1775#
L$PROT  021214 G          1775    4207#
L$PRT   002112 G          1775#
L$REPP  002062 G          1775#
L$REV   002010 G          1775#
L$RPT   017366 G          1775    4118#
L$SOFT  027254 G          1775    5441#
L$SPC   002056 G          1775#
L$SPCP  002020 G          1775#
L$SPTP  002024 G          1775#
L$STA   002030 G          1775#
L$SW    002204 G          1775    1838#
L$TEST  002114 G          1775#   3040    3287    3855
L$TIML  002014 G          1775#
L$UNIT  002012 G          1775#   4259
L10000  002202            1818    1829#
```

```
L10001  002312            1838    1887#
L10002  005640            2535    2538#
L10003  006564            2554    2572#
L10004  006572            2584#
L10005  006600            2588#
L10006  006606            2592#
L10007  006614            2596#
L10010  021212            4141    4201#
L10012  022514            4362    4378#
L10013  023004            4420#
L10014  023200            4460    4474#
L10015  023244            4494    4508#
L10016  023340            4536    4551#
L10017  024726            4706    4908#
L10020  023474            4598#
L10021  023520            4606#
L10022  023540            4613#
L10023  023560            4620#
L10024  023600            4627#
L10025  023620            4634#
L10026  023640            4641#
L10027  023660            4648#
L10030  023700            4655#
L10031  023720            4662#
L10032  023756            4672#
L10033  024142            4701#
L10034  025402            4961    5051#
L10035  026040            5147    5150#
L10036  026204            5189    5193#
L10037  026336            5226    5230#
L10040  027176            5326    5366#
L10041  027252            5403    5413    5427#
L10042  030464            5441    5522    5532#
L10043  030674            5552#
L10045  030700            5552    5555#
MBR   = 100012 G          2087#   2148    2392
MEMOM   022412            4352    4374#
MISCFG  003541 G          2331#   3385*   4051    4061*
MOD.C0= 000400 G          1967#   2043    2054    2066    2069    2072    2099    2105    2108    2606    2688    2723    2756
                          2845    3167    3168    3172    3807    5295
MOD.C1= 001000 G          1966#   2043    2069    2072    2075    2078    2081    2093    2096    2099    2111    3404
MOD.C2= 002000 G          1965#   2043    2084
MOD.C3= 004000 G          1962#   2043
MOVMSG  011552 G          3004    3048    3086    3092    3141#   4000    4341
MSGCNT= 000016 G          2031#   2177    3022    3146    3994
MSGPKA  002506 G          2189#   3018    3020    3144    3992    4590
MSGPKT  002340 G          2158#   2551    2553    2607    3147*   3150    3161    3206    3218    3283    3292    3675    3676
                          3889    3991    4038
MSGPK0  002356 G          2168#   2175    2189
MSGPK1  002374 G          2169#   2190
MSGPK2  002412 G          2170#   2191
MSGPK3  002430 G          2171#   2192
MS.RFC= 000004 G          1993#   2551    2607    3218    3675    3676    3889
MS.XS0= 000006 G          1994#   2553    3150    3283    3292    4038
MS.XS1= 000010 G          1995#   2553    4038
MS.XS2= 000012 G          1996#   2553    3161    3206    4038
```

```
MS.XS3= 000014 G   1997#  2553   3991   4038
NCMD.C= 177740 G   2043#  2832
NCNT    003422 G   2268#  2667   2703*  2865*  3041   3799*  3800   3829*  3831*  3858   5104   5110*  5112*
                   5133*  5138*  5285   5308   5316*
NCNT1   003424 G   2269#  2667   2864*  3800   3829   5104   5110   5285   5308*
NEXTSP  027322     5455   5457#
NEXTU   016662 G   2698   2748   2776   3015   3825   3972#  4138   4307   4345   4417   4457   4596   5141
                   5305
NINUSE= 177774 G   2042#  2212   2213   2214   2215   3063   3195   3345   3369   3406   3548   3556   3564
                   3644   3654   3776   3828   3953   3976   4017   4256   4492   5109   5114   5130
NOINTM  004505 G   2484#  3087
NRDYM   023102     4332   4413   4424#
NSSRM   004355 G   2479#  3005
NUMBM   030365     5465   5469   5473   5477   5483   5487   5491   5517#
NURTY1  005157 G   2496#  3702
OFLINM  005213 G   2497#  4329   4408
ONEFIL= 000001     1712#  1754   1889   1890   1897#  1904   4095#  4102   4555   4564#  5368   5369   5377#
                   5383
OPFLAG  003542 G   2337#  4270*  4318
OPP.C = 020000 G   1960#  2043   2072   2078   3396
O$APTS= 000000     1728#  1775
O$AU  = 000001     1728#  1765#  1775
O$BGNR= 000001     1728#  1765#  1775
O$BGNS= 000001     1728#  1765#  1775
O$DU  = 000001     1728#  1765#  1775
O$ERRT= 000000     1728#  1775
O$GNSW= 000001     1728#  1765#  1775
O$POIN= 000001     1728#  1765#  1775
O$SETU= 000001     1728#  1765#  1775   5542
PASCNT  003260 G   2243#  2524   2541   4011   4012*  4128   4300*  4302*  4593
PATCH   030464 G   5538#
PATERN  003456 G   2282#  2866*  2895
PATR0   010364 G   2906   2919#
PATR1   010422 G   2907   2932#
PATR2   010442 G   2908   2942#
PATR3   010452 G   2909   2948#
PATR4   010476 G   2910   2959#
PATR5   010510 G   2911   2966#
PATR6   010522 G   2912   2972#
PATR7   010542 G   2913   2981#  2986
PATR8   010574 G   2914   2991#
PATTBL  010342     2900   2906#
PATTM   030405     5466   5470   5474   5478   5484   5488   5492   5518#
PCMDWD  003434 G   2273#  2704*  2793   2879*  3168   3172   3288   3386*  3620*  3631*  3645*  3772*  3779*
                   3832*  5317*  5355*
PIRE    002213 G   1852#  5243
PNT   = 001000 G   1920#
PR:   = 002000 G   1920#
PRI00 = 000000 G   1920#  4317
PRI01 = 000040 G   1920#
PRI02 = 000100 G   1920#
PRI03 = 000140 G   1920#
PRI04 = 000200 G   1920#
PRI05 = 000240 G   1920#
PRI06 = 000300 G   1920#
PRI07 = 000340 G   1920#  2028   4393
```

```
PRXST   017140 G        4004    4037#   5289
PTCMDS  027050          5247    5248    5249    5250    5251    5252    5253    5337#
PWRFLG  003537 G        2329#   4268*   4272    4348    4360*
RAMDAT  003346 G        2261#   3731    3748
RAMDUM  013360 G        3224    3256    3272    3304    3323    3352    3379    3401    3431    3462    3477    3484#   3509
                        3519
RAMER   015164 G        3488    3490    3492    3495    3726#
RAMFHR  005310          2501#   3485
RAMHLD  003342          2259#   3487*   3489*   3491*   3494*   3732    3744
RAMIOP  005401          2502#   3747
RAMLIN  005464          2504#   3496
RAMM    030262          5459    5512#
RAMPD   005452          2503#   3753
RAMRSH  003344          2260#
RAMSIZ  003406 G        2262#   3486*   3493*   3733    3743    3749
RAMWRT  002214 G        1853#   3484
RANB    003442 G        2276#   2672*   2673    2983*   2984    4247*   4933    4996*   4997    5043*   5044
RANBC = 153624 G        2040#   4247
RANCMD  025210          4935    4967#
RANDOM  003523 G        2311#   2669    3219    3294    4580*   4917*   5073*   5161*   5205*   5241*
RANP  = 000007 G        1862    1866    1870    1874    1878    1882    1886    2037#   2351    2355    2359    2363    2367
                        4930    4999    5003    5046    5213    5221
RANRD   025250          4949    4975    4976    4977    4978    4979    4980    4981    4982    4994#
RANS    003444 G        2277#   2672    2673*   2674    2983    2984*   2985    4248*   4933*   4934    4996    5043
RANSC = 032561 G        2041#   4248
RANW    025350          5013    5027    5041#
RANWR   025324          4968    4969    4970    4971    4972    4973    4974    5012#   5170
RANWV   025336          4967    5026#
RCVERM  030116          5453    5507#
RDF   = 104001 G        2051#   2360    2378    3780    4691    4873    5000    5210
RDR   = 104401 G        2054#   2356    2379    3773    4682    4859    4994    5218    5356
RECCNT  003330 G        2249#   2524    2541    3037*   3038    3085    3160*   3164*   3169*   3174*   3181    3378*   3381*
                        3436*   3437    3581    4128    4306*
RECLOG  003473 G        2297#   3159    3162*   4006
RECRED  006562          2526*   2530    2531*   2534    2547*   2548    2570#
RECTAP  006616 G        2529    2546    2606#
RECUD   011636 G        3093    3159#
REPEAT= 050232          3551#   3562
RERM    004634 G        2488#   3351    3400    3430
RETRY = 050230          3549#   3569    3589
RETRYC  003466 G        2293#   3366*   3395    3399    3403*   3429    3433*   3516    3524    3563*   3572    3574    3588
                        3702
REWRT   014550          3555    3644#
RFBC    002660 G        2231#   4131
RFCERM  004340 G        2478#   3223
RFREC   002760 G        2236#   4132
RFUNR   002770 G        2237#   4133
RLEXM   004374 G        2480#   3518    3526
RNF   = 125401 G        2072#   2387    4867
RNOPSC= 177740 G        2036#   4928    4950    4997    5044
RNR   = 105401 G        2069#   2386    4863
RNYM    004570 G        2487#   4002
RPF   = 105001 G        2075#   2388    4877
RPR   = 125001 G        2078#   2389    4881
RPTCNT  003470 G        2294#   3367*   3557*   3560*   3561    3572    3586*
RPTFLG  003525 G        2313#   2750    2751*   4955*
```

```
RPT1A   020230            4128    4185#
RPT1B   020305            4129    4186#
RPT1C   020356            4130    41?~#
RPT1D   020427            4131    4188#
RPT1E   020655            4148    4192#
RPT1F   020533            4132    4190#
RPT1G   020604            4133    4191#
RPT1I   021001            4137    4195#
RPT1J   020705            4151    4193#
RPT1K   020772            4160    4194#
RRANV   002205 G          1846#   4246
RRBC    002620 G          2230#   4130
RRECL = 000020 G          2038#   3394    3524
RRREC   002740 G          2234#   4132
RRUNR   002750 G          2235#   4133
RRVM    030043            5451    5505#
RTLE    013512 G          3392    3428    3507#
RTLRTN  013646            3512    3530#
RWCPK   002334 G          2153#   3382    3459
RWD   = 102010 G          2084#   2153    2348    2364    2391    2651    3070    3163    4747    4823    4845    4891
RWERR   003475 G          2299#   3230    3362*   3514*   3585*   3693
R5SAVE  003462 G          2285#   3011*   3017    3989*   4026    4125*   4140    5127*   5129
SCCNT   003270 G          2244#   3202*   3300*   4137
SCERM   004314 G          2477#   3255
SCH   = 140004 G          2114#   2344    2401    2636    3010    4725    4733    4741
SCHBK   002446 G          2175#   2854    2856*   3018*
SCHCNT= 000010 G          2029#   2855
SEQEND  003634 G          2369#   4932    5169
SETCH   006714 G          2635#   4921    5075    5165    5207    5244
SETRW   006740 G          2651#   4923    4957    5076    5166    5185    5208
SETUP   007722 G          2666    2827#   4080    5102    5125    5284
SFF   = 105010 G          2096#   2395    3073    4809    4817
SFPTBL  002204 G          1838#
SFR   = 105410 G          2099#   2396    3073    4805    4813
SRF   = 104010 G          2063#   2384    4827    4835
SRR   = 104410 G          2066#   2385    3288    3621    4831    4839
STAERM  005642 G          2540#   3005    3030    3049    3087    3223    3255    3271    3303    3322    3351    3400    3430
                          3461    3476    3508    3518    3526    4002
STAER1  006154            2524    2541    2557#   2790*   2791*   2792*
STAER2  006332            2550    2561#
STAER3  006411            2551    2562#
STAER4  006447            2552    2563#
STAER5  006467            2553    2566#   4038
STAER6  006276            2534    2548    2560#
STAER7  006246            2525    2542    2559#   2796*   2797*   2798*
STAFLG  003536 G          2328#   4022*   4252*   4267*   4275    4605*   4922    4924*   5077*   5167*   5209*   5255*
STREAM  003533 G          2319#   3802    3823    5093*   5145*
STTIM   003414 G          2265#   3770*   3778*   3868    3872*
SVCGBL= 000000            1728#   1742#   1775    1791    1806    1807    1818    1838    2518    2540    2582    2586    2590
                          2594    4118    4207    4220    4388    4444    4484    4519    5403    5441    5542#
SVCINS= 000001            1728#   1739#   1775    1791    1806    1807    1818    1838    2524    2525    2534    2535    2538
                          2541    2542    2548    2550    2551    2552    2553    2554    2572    2584    2588    2592    2596
                          2752    3005    3030    3042    3049    3069    3071    3074    3087    3125    3223    3255    3271
                          3303    3322    3351    3376    3400    3430    3461    3476    3485    3496    3508    3518    3526
                          3572    3592    3702    3747    3753    3891    3892    3895    3896    3921    3928    3936    3957
                          4002    4010    4016    4037    4038    4052    4053    4059    4128    4129    4130    4131    4132
```

```
                    4133  4137  4141  4148  4151  4156  4160  4164  4170  4201  4232  4233  4250
                    4251  4265  4266  4270  4273  4276  4292  4297  4311  4312  4313  4314  4317
                    4324  4329  4332  4335  4336  4349  4352  4353  4362  4378  4393  4395  4398
                    4401  4408  4413  4420  4456  4460  4474  4493  4494  4508  4529  4533  4536
                    4551  4583  4594  4598  4600  4606  4608  4613  4615  4620  4622  4627  4629
                    4634  4636  4641  4643  4648  4650  4655  4657  4662  4664  4672  4680  4701
                    4706  4908  4961  5051  5063  5147  5150  5189  5193  5226  5230  5279  5326
                    5366  5403  5410  5411  5413  5427  5441  5450  5451  5452  5453  5454  5455
                    5456  5457  5458  5459  5460  5461  5462  5463  5464  5465  5466  5467  5468
                    5469  5470  5471  5472  5473  5474  5475  5476  5477  5478  5479  5480  5481
                    5482  5483  5484  5485  5486  5487  5488  5489  5490  5491  5492  5494  5522
                    5532  5542  5552
SVCSUB= 000000       1728#  1741#  4583  4600  4608  4615  4622  4629  4636  4643  4650  4657  4664
                    4680
SVCTAG= 000000       1728#  1743#  1829  1887  2538  2572  2584  2588  2592  2596  4059  4201  4378
                    4420  4474  4508  4551  4598  4606  4613  4620  4627  4634  4641  4648  4655
                    4662  4672  4701  4908  5051  5150  5193  5230  5366  5427  5532  5552  5555
SVCTST= 000000       1728#  1740#  4578  4915  5066  5159  5203  5239
SWBFLG  003526 G     2314#  2613  2881  3794  3908  4669*  4671*  4686*  4695*
SWB.C = 010000 G     1961#  2043  2882  3795
S$LSYM= 010000       1728#  1829#  1887#  2538#  2572#  2584#  2588#  2592#  2596#  4059#  4201#  4378#  4420#
                    4474#  4508#  4551#  4598#  4606#  4613#  4620#  4627#  4634#  4641#  4648#  4655#  4662#
                    4672#  4701#  4908#  5051#  5150#  5193#  5230#  5366#  5427#  5532#
TAPCAP  021112       4197#  5068
TCCRA   012232       3215  3238#
TCC0    012252 G     3238  3254#
TCC1    012274 G     3239  3271#  4342
TCC2    012312 G     3240  3283#
TCC3    012460 G     3241  3322#
TCC4    012502 G     3242  3345#
TCC5    013200 G     3243  3428#
TCC6    013302 G     3244  3459#
TCC7    013336 G     3245  3476#
TC2RTN  012456       3285  3289  3308#
TIME1   003446 G     2278#  2527*  2532  3000*  3002*  3003  3067*  3082*  3083  3084  3123*  3126*  3127
                    3923*  3924*  3928  4059  4322*  4340
TIMF2   003450 G     2279#  2528*  2533  3925*  3926*  3928  4334*  5278*  5280*
TOERM   004273 G     2476#  3049
TOOMM   004544 G     2486#  3030
TRAPD4  003540 G     2330#  4392*  4396  4431*
TRAP4   023132 G     4393  4431#
TSAM    004522 G     2485#  3303
TSBA  = 002456 G     2185#  3740
TSC.FC= 177717 G     1953#
TSC.TC= 177761 G     1954#  3142
TSDB    002456 G     2182#  2185  2551  3045*  3382*  3459*  3734  3998*  4295*  4323*  4403*  4531*
TSSR    002466 G     2183#  3009  3029  3047  3080  3127  3141  4294*  4325  4326  4394  4398  4405
                    4406  4530*
TSSREG  003464 G     2286#  2551  3029*  3141*  3142  3199  3205
TSVCT   002476 G     2184#  4296*  4297  4456  4532*  4533
TS.A16= 000400 G     1950#
TS.A17= 001000 G     1949#
TS.NBA= 002000 G     1948#
TS.NXM= 004000 G     1947#  3205
TS.OFL= 000100 G     1952#  4326  4406
TS.RMR= 010000 G     1946#
```

```
TS.SC = 100000 G        1943#   3199
TS.SPE= 020000 G        1945#
TS.SSR= 000200 G        1951#   3009    3047    3079    3127    4325    4405
TS.UPE= 040000 G        1944#   3205
TS4ADR  027226          5410    5422#
TS4CL   002526 G        2203#   4592*   4594
TS4INT  002516 G        2196#   4297    4533
TS4IN0  006566 G        2196    2582#
TS4IN1  006574 G        2197    2586#
TS4IN2  006602 G        2198    2590#
TS4IN3  006610 G        2199    2594#
TS4VCT  027243          5411    5423#
T$ARGC= 000001          1775#   2524#   2525#   2534#   2541#   2542#   2548#   2550#   2551#   2552#   2553#   3376#   3485#
                        3496#   3572#   3592#   3702#   3747#   3753#   3892#   3896#   3928#   3936#   4037#   4038#   4128#
                        4129#   4130#   4131#   4132#   4133#   4137#   4148#   4151#   4156#   4160#   4164#   4170#   4311#
                        4329#   4332#   4352#   4398#   4408#   4413#   4493#   4594#   5068#
T$CODE= 023004          4059#   5410#   5411#   5413#   5450#   5451#   5452#   5453#   5454#   5455#   5456#   5457#   5458#
                        5459#   5460#   5461#   5462#   5463#   5464#   5465#   5466#   5467#   5468#   5469#   5470#   5471#
                        5472#   5473#   5474#   5475#   5476#   5477#   5478#   5479#   5480#   5481#   5482#   5483#   5484#
                        5485#   5486#   5487#   5488#   5489#   5490#   5491#   5492#   5494#   5522#
T$ERRN= 000001          1728#   3005#   3030#   3049#   3087#   3223#   3255#   3271#   3303#   3322#   3351#   3400#   3430#
                        3461#   3476#   3508#   3518#   3526#   3891#   3895#   3921#   4002#   4232#
T$EXCP= 000000          5410#   5411#   5462#   5463#   5464#   5465#   5466#   5467#   5468#   5469#   5470#   5471#   5472#
                        5473#   5474#   5475#   5476#   5477#   5478#   5481#   5482#   5483#   5484#   5485#   5486#   5487#
                        5488#   5489#   5490#   5491#   5492#
T$FLAG= 000041          2535#   2554#   4141#   4362#   4460#   4494#   4536#   4706#   4961#   5147#   5189#   5226#   5326#
                        5413#   5522#
T$FREE= 030700          5542    5556#
T$GMAN= 000000          1728#
T$HILI= 000010          5410#   5411#   5462#   5463#   5464#   5465#   5466#   5467#   5468#   5469#   5470#   5471#   5472#
                        5473#   5474#   5475#   5476#   5477#   5478#   5481#   5482#   5483#   5484#   5485#   5486#   5487#
                        5488#   5489#   5490#   5491#   5492#
T$LAST= 000001          1728#   5542#   5551
T$LOLI= 000000          5410#   5411#   5462#   5463#   5464#   5465#   5466#   5467#   5468#   5469#   5470#   5471#   5472#
                        5473#   5474#   5475#   5476#   5477#   5478#   5481#   5482#   5483#   5484#   5485#   5486#   5487#
                        5488#   5489#   5490#   5491#   5492#
T$LSYM= 010000          1728#   1829    1887    2538    2572    2584    2588    2592    2596    4201    4378    4420    4474
                        4508    4551    4598    4606    4613    4620    4627    4634    4641    4648    4655    4662    4672
                        4701    4908    5051    5150    5193    5230    5366    5427    5532
T$LTNO= 000006          5542#
T$NEST= 177777          1728#   1758#   1818#   1829#   1838#   1887#   1888#   1913#   2518#   2538#   2540#   2572#   2582#
                        2584#   2586#   2588#   2590#   2592#   2594#   2596#   4087#   4111#   4118#   4201#   4207#   4211#
                        4220#   4378#   4388#   4420#   4444#   4474#   4484#   4508#   4519#   4551#   4553#   4576#   4578#
                        4583#   4598#   4600#   4606#   4608#   4613#   4615#   4620#   4622#   4627#   4629#   4634#   4636#
                        4641#   4643#   4648#   4650#   4655#   4657#   4662#   4664#   4672#   4680#   4701#   4908#   4915#
                        5051#   5066#   5150#   5159#   5193#   5203#   5230#   5239#   5366#   5367#   5392#   5403#   5413
                        5427#   5441#   5455    5461    5479    5480    5494    5522    5532#   5543#
T$NS0 = 000000          1758#   1888    1913#   4087    4111#   4553    4576#   5367    5392#   5543
T$NS1 = 000005          1818#   1829    1838#   1887    2518#   2538    2540#   2572    2582#   2584    2586#   2588    2590#
                        2592    2594#   2596    4118#   4201    4207#   4211    4220#   4378    4388#   4420    4444#   4474
                        4484#   4508    4519#   4551    4578#   4908    4915#   5051    5066#   5150    5159#   5193    5203#
                        5230    5239#   5366    5403#   5413    5427    5441#   5455    5461    5479    5480    5494    5522
                        5532
T$NS2 = 000002          4583#   4598    4600#   4606    4608#   4613    4615#   4620    4622#   4627    4629#   4634    4636#
                        4641    4643#   4648    4650#   4655    4657#   4662    4664#   4672    4680#   4701
T$PCNT= 000000          5551#   5552#
```

```
T$PTAB= 010044          5552#
T$PTHV= 000001          1775    5556#
T$PTNU= 000001          1728#   5552#   5556
T$SAVL= 177777          1728#
T$SEGL= 177777          1728#
T$SIZE= 000004          5542    5556#
T$SUBN= 000000          1728#   4578#   4583#   4600#   4608#   4615#   4622#   4629#   4636#   4643#   4650#   4657#   4664#
                        4680#   4915#   5066#   5159#   5203#   5239#
T$TAC' = 177777         1728#
T$TAGN= 010046          1728#   1818#   1838#   2518#   2540#   2582#   2586#   2590#   2594#   4118#   4207#   4220#   4388#
                        4444#   4484#   4519#   4578#   4583#   4600#   4608#   4615#   4622#   4629#   4636#   4643#   4650#
                        4657#   4664#   4680#   4915#   5066#   5159#   5203#   5239#   5403#   5441#   5551#   5552#
T$TEMP= 000000          1791#   1829#   1887#   1888#   2535#   2538#   2554#   2572#   2584#   2588#   2592#   2596#   4059#
                        4087#   4141#   4201#   4211#   4362#   4378#   4420#   4460#   4474#   4494#   4508#   4536#   4551#
                        4553#   4598#   4606#   4613#   4620#   4627#   4634#   4641#   4648#   4655#   4662#   4672#   4701#
                        4706#   4908#   4961#   5051#   5147#   5150#   5189#   5193#   5226#   5230#   5326#   5366#   5367#
                        5410#   5411#   5413#   5427#   5450#   5451#   5452#   5453#   5454#   5456#   5457#   5458#   5459#
                        5460#   5462#   5463#   5464#   5465#   5466#   5467#   5468#   5469#   5470#   5471#   5472#   5473#
                        5474#   5475#   5476#   5477#   5478#   5481#   5482#   5483#   5484#   5485#   5486#   5487#   5488#
                        5489#   5490#   5491#   5492#   5522#   5532#   5543#
T$TEST= 000006          1728#   4578#   4583    4600    4608    4615    4622    4629    4636    4643    4650    4657    4664
                        4680    4915#   5066#   5159#   5203#   5239#   5542
T$TSTM= 177777          1728#   2524    2525    2534    2538    2541    2542    2548    2550    2551    2552    2553    2572
                        2752    3005    3030    3049    3069    3087    3125    3223    3255    3271    3303    3322    3351
                        3376    3400    3430    3461    3476    3485    3496    3508    3518    3526    3572    3592    3702
                        3747    3753    3891    3892    3895    3896    3921    3928    3936    3957    4002    4010    4016
                        4037    4038    4052    4059    4128    4129    4130    4131    4132    4133    4137    4148    4151
                        4156    4160    4164    4170    4201    4232    4250    4265    4270    4273    4276    4292    4297
                        4311    4313    4314    4317    4329    4332    4336    4349    4352    4353    4362    4378    4393
                        4395    4398    4401    4408    4413    4420    4456    4460    4474    4493    4508    4529    4533
                        4551    4583    4594    4598    4600    4606    4608    4613    4615    4620    4622    4627    4629
                        4634    4636    4641    4643    4648    4650    4655    4657    4662    4664    4672    4680    4701
                        4706    4908    4961    5051    5068    5147    5150    5189    5193    5226    5230    5326    5366
T$TSTS= 000001          1728#   4578#   4915#   5066#   5159#   5203#   5239#
T$$AU = 010016          4519#   4536    4551
T$$AUT= 010013          4388#   4420
T$$CLE= 010014          4444#   4460    4474
T$$DAT= 010045          5552#   5555
T$$DU = 010015          4484#   4494    4508
T$$HAR= 010041          5403#   5413    5427
T$$HW = 010000          1818#   1829
T$$INI= 010012          4220#   4362    4378
T$$MSG= 010003          2518#   2535    2538    2540#   2554    2572
T$$PC = 000001          5551#   5556
T$$PRO= 010011          4207#
T$$PTA= 010044          5551#   5552#
T$$RPT= 010010          4118#   4141    4201
T$$SOF= 010042          5441#   5522    5532
T$$SRV= 010007          2582#   2584    2586#   2588    2590#   2592    2594#   2596
T$$SUB= 010033          4583#   4598    4600#   4606    4608#   4613    4615#   4620    4622#   4627    4629#   4634    4636#
                        4641    4643#   4648    4650#   4655    4657#   4662    4664#   4672    4680#   4701
T$$SW = 010001          1838#   1887
T$$TES= 010040          4578#   4706    4908    4915#   4961    5051    5066#   5147    5150    5159#   5189    5193    5203#
                        5226    5230    5239#   5326    5366
T1      023342 G        1791    4578#
T1SWB   003531 G        2317#   3903    4678*   4703*
```

```
T1.1     023352           4583#
T1.10    023702           4657#
T1.11    023722           4664#
T1.12    024006           4580#
T1.2     023476           4600#
T1.3     023522           4608#
T1.4     023542           4615#
T1.5     023562           4622#
T1.6     023602           4629#
T1.7     023622           4636#
T1.8     023642           4643#
T1.9     023662           4650#
T2       024730 G         1791    4915#
T3       025404 G         1791    5066#
T4       026042 G         1791    5159#
T5       026206 G         1791    5203#
TSWEOT   027072           4945    5180    5311    5350#
T6       026340 G         1791    5239#
UAM   =  000200 G         1920#
UNL   =  100412 G         2108#   2399
UNREC    003476 G         2300#   3298*   3389*   3415*   3442*   3517*   3525*   3695
URERM    004656 G         2489#   3461    3508
VFEXC    015504 G         3775    3782    3793#   4683    4685    4688    4690    4692    4694    4697    4699
VFISU    015770 G         3811    3814    3820    3844#
VFYCNT   003300 G         2245#   3919*   4137
VFYDAT   015370 G         2706    3769#   5128    5319
VFYFLG   003524 G         2312#   2671    2874*   2877*   3219    3294    3713    5126*   5139*
VFY.C =  000100 G         1970#   2060    2873    2875
WLKZRO   010456           2949#   2954    2960
WRBC     002560 G         2229#   4129    4241*
WRECL =  000020 G         2039#   3516
WRITEM   017312 G         4072#   5115    5134
WRR   =  105005 G         2081#   2390    4853
WRREC    002720 G         2232#   3694    4132
WRT   =  104005 G         2057#   2352    2380    2670    3040    4787    4849    5012    5095    5119
WRTY     013650 G         3368    3548#
WRTYCT   003250 G         2242#   3365*   4148
WRTYER   003472 G         2296#   3356*   3373    3554*   3561    3568    3590
WRTYFG   003471 G         2295#   3095    3357    3363*   3384*
WRUNR    002730 G         2233#   4133
WSSR     011516 G         3014    3123#   3460    3737    3739    3999    4404    4455
WTM   =  100011 G         2090#   2393    2838    3288    4074    4783    4795
WTR   =  101011 G         2093#   2394    4799
WTV   =  104105 G         2060#   2381    4753    4757    4761    4765    4769    4773    4777    4897    4901    4926    5026
WTVERM   004250 G         2475#   3891    3895    3921
WTYBRF   014166           3361*   3605#   3650
WTYCMD   014162           3360*   3603#   3647
WTYWRD   014164           3359*   3386    3604#   3646
X$ALWA=  000000           1728#   5413    5479    5480    5494    5522
X$FALS=  000040           1728#   5461
X$OFFS=  000400           1728#   5413    5455    5461    5479    5480    5494    5522
X$TRUE=  000020           1728#   5455
XO.BOT=  000002 G         2006#   2689    2724    2734    2757    3173    3283    3292    3808    3854    5296
XO.EOT=  000001 G         2007#   2693    2725    2739    2758    2767    3809    3818    5300
XO.LET=  020000 G         2003#   3292
XO.ONL=  000100 G         2005#
```

```
X0.RLL= 010000 G        2004#   3292
X0.RLS= 040000 G        2002#   3292
X0.TMK= 100000 G        2001#   3292
X2.OPM= 100000 G        2011#   3161    3206
X3.DCK= 000010 G        2015#
X3.RNY= 157400 G        2016#   4001
ZROPAT  010426          2933#   2936    2943    2967
$BGNLE= 177777          1730#
$BRJMP= 177777          5258#   5324#
$ERFLG= 000400          1730#   2526#   2527#   2528#   2530#   2531#   2532#   2533#   2543#   2544#   2547#   2583#   2587#
                        2591#   2595#   2607#   2609#   2610#   2611#   2612#   2614#   2616#   2617#   2619#   2622#   2635#
                        2651#   2652#   2653#   2664#   2672#   2673#   2674#   2675#   2677#   2679#   2685#   2690#   2695#
                        2703#   2704#   2709#   2736#   2737#   2751#   2788#   2793#   2796#   2797#   2798#   2810#   2812#
                        2814#   2815#   2827#   2839#   2846#   2848#   2854#   2856#   2864#   2865#   2872#   2874#   2877#
                        2879#   2880#   2882#   2885#   2895#   2896#   2897#   2898#   2899#   2919#   2920#   2922#   2923#
                        2925#   2933#   2949#   2973#   2981#   3000#   3002#   3011#   3017#   3018#   3020#   3021#   3023#
                        3024#   3029#   3034#   3037#   3038#   3067#   3077#   3079#   3080#   3082#   3085#   3089#   3097#
                        3109#   3111#   3123#   3126#   3141#   3142#   3143#   3144#   3145#   3147#   3148#   3150#   3160#
                        3162#   3164#   3169#   3174#   3181#   3207#   3209#   3212#   3214#   3222#   3231#   3254#   3298#
                        3300#   3302#   3355#   3356#   3359#   3360#   3361#   3362#   3363#   3365#   3366#   3367#   3378#
                        3381#   3382#   3384#   3385#   3386#   3389#   3394#   3396#   3403#   3404#   3410#   3415#   3433#
                        3435#   3436#   3437#   3442#   3459#   3514#   3517#   3525#   3554#   3557#   3560#   3563#   3575#
                        3576#   3577#   3579#   3580#   3581#   3584#   3585#   3586#   3620#   3621#   3622#   3623#   3624#
                        3625#   3631#   3632#   3633#   3634#   3639#   3645#   3646#   3647#   3648#   3649#   3650#   3651#
                        3669#   3670#   3672#   3673#   3674#   3676#   3678#   3680#   3681#   3683#   3685#   3686#   3688#
                        3690#   3691#   3694#   3696#   3697#   3699#   3769#   3770#   3772#   3773#   3774#   3777#   3778#
                        3779#   3780#   3781#   3793#   3795#   3797#   3799#   3829#   3831#   3832#   3845#   3847#   3859#
                        3862#   3863#   3870#   3872#   3873#   3889#   3898#   3901#   3902#   3922#   3934#   3951#   3952#
                        3954#   3959#   3972#   3975#   3977#   3989#   3990#   3991#   3992#   3993#   3995#   3996#   3998#
                        4007#   4012#   4014#   4015#   4022#   4025#   4026#   4054#   4056#   4057#   4058#   4061#   4073#
                        4074#   4075#   4076#   4078#   4079#   4125#   4140#   4149#   4150#   4154#   4157#   4158#   4161#
                        4162#   4165#   4166#   4239#   4241#   4242#   4247#   4248#   4252#   4253#   4254#   4256#   4257#
                        4259#   4261#   4262#   4267#   4268#   4271#   4278#   4280#   4283#   4288#   4290#   4291#   4294#
                        4295#   4296#   4300#   4302#   4306#   4322#   4323#   4334#   4350#   4357#   4358#   4360#   4392#
                        4394#   4397#   4399#   4400#   4403#   4407#   4412#   4431#   4491#   4492#   4527#   4528#   4530#
                        4531#   4532#   4534#   4580#   4581#   4585#   4590#   4591#   4592#   4602#   4605#   4610#   4617#
                        4624#   4631#   4638#   4645#   4652#   4659#   4666#   4669#   4671#   4674#   4678#   4682#   4684#
                        4686#   4687#   4689#   4691#   4693#   4695#   4696#   4698#   4703#   4714#   4716#   4718#   4917#
                        4918#   4919#   4920#   4924#   4926#   4927#   4928#   4929#   4930#   4933#   4934#   4937#   4939#
                        4941#   4950#   4951#   4952#   4954#   4955#   4956#   4958#   4994#   4995#   4996#   4997#   4998#
                        4999#   5000#   5001#   5002#   5003#   5012#   5026#   5041#   5043#   5044#   5045#   5046#   5069#
                        5072#   5073#   5074#   5077#   5078#   5091#   5092#   5093#   5094#   5095#   5096#   5097#   5098#
                        5099#   5101#   5110#   5112#   5116#   5118#   5119#   5120#   5121#   5122#   5123#   5124#   5126#
                        5127#   5129#   5133#   5138#   5139#   5140#   5145#   5161#   5162#   5163#   5164#   5167#   5172#
                        5174#   5176#   5184#   5186#   5205#   5206#   5209#   5210#   5211#   5212#   5213#   5214#   5216#
                        5217#   5218#   5219#   5220#   5221#   5222#   5224#   5241#   5242#   5243#   5245#   5246#   5254#
                        5255#   5256#   5263#   5267#   5269#   5270#   5273#   5277#   5278#   5282#   5292#   5297#   5302#
                        5308#   5309#   5314#   5316#   5317#   5337#   5338#   5339#   5340#   5341#   5355#   5356#   5357#
                        5358#   5359#   5360#
$FSAND= 000310          1730#   2545    2606    2608    2613    2665    2667    2669    2670    2671    2676    2687    2688
                        2689    2693    2700    2722    2723    2724    2725    2726    2734    2739    2743    2750    2755
                        2756    2757    2758    2759    2767    2771    2811    2838    2845    2873    2881    2924    3003
                        3010    3013    3022    3028    3036    3040    3041    3047    3063    3070    3073    3076    3084
                        3095    3146    3159    3161    3163    3166    3167    3168    3172    3173    3195    3199    3200
                        3201    3205    3206    3217    3218    3219    3221    3230    3283    3287    3288    3292    3294
                        3296    3297    3345    3349    3350    3354    3357    3369    3375    3393    3395    3399    3405
```

$FSBAD= 000401

3406 3429 3434 3484 3507 3515 3516 3524 3548 3556 3564 3568 3571
3574 3578 3590 3591 3630 3644 3654 3668 3671 3675 3679 3684 3689
3693 3695 3700 3701 3771 3776 3794 3800 3806 3807 3808 3809 3810
3818 3828 3846 3850 3853 3854 3855 3858 3861 3867 3868 3890 3894
3903 3907 3908 3909 3927 3935 3953 3956 3994 4001 4006 4011 4017
4018 4050 4051 4127 4134 4152 4153 4163 4231 4237 4240 4246 4272
4275 4289 4298 4299 4310 4318 4320 4325 4326 4340 4348 4351 4391
4396 4405 4406 4454 4589 4593 4675 4715 4922 4932 5067 5089 5104
5109 5114 5130 5169 5260 5285 5288 5294 5295 5296 5300 5307 5310
1730# 2526 2527 2528 2530 2531 2532 2533 2543 2544 2545 2547 2583
2587 2591 2595 2606 2607 2608 2609 2610 2611 2612 2613 2614 2616
2617 2619 2622 2635 2651 2652 2653 2664 2665 2667 2669 2670 2671
2672 2673 2674 2675 2676 2677 2679 2685 2687 2688 2689 2690 2693
2695 2700 2703 2704 2709 2722 2723 2724 2725 2726 2734 2736 2737
2739 2743 2750 2751 2755 2756 2757 2758 2759 2767 2771 2788 2793
2796 2797 2798 2810 2811 2812 2814 2815 2827 2838 2839 2845 2846
2848 2854 2856 2864 2865 2872 2873 2874 2877 2879 2880 2881 2882
2885 2895 2896 2897 2898 2899 2919 2920 2922 2923 2924 2925 2933
2949 2973 2981 3000 3002 3003 3010 3011 3013 3017 3018 3020 3021
3022 3023 3024 3028 3029 3034 3036 3037 3038 3040 3041 3047 3063
3067 3070 3073 3076 3077 3079 3080 3082 3084 3085 3089 3095 3097
3109 3111 3123 3126 3141 3142 3143 3144 3145 3146 3147 3148 3150
3159 3160 3161 3162 3163 3164 3166 3167 3168 3169 3172 3173 3174
3181 3195 3199 3200 3201 3205 3206 3207 3209 3212 3214 3217 3218
3219 3221 3222 3230 3231 3254 3283 3287 3288 3292 3294 3296 3297
3298 3300 3302 3345 3349 3350 3354 3355 3356 3357 3359 3360 3361
3362 3363 3365 3366 3367 3369 3375 3378 3381 3382 3384 3385 3386
3389 3393 3394 3395 3396 3399 3403 3404 3405 3406 3410 3415 3429
3433 3434 3435 3436 3437 3442 3459 3484 3507 3514 3515 3516 3517
3524 3525 3548 3554 3556 3557 3560 3563 3564 3568 3571 3574 3575
3576 3577 3578 3579 3580 3581 3584 3585 3586 3590 3591 3620 3621
3622 3623 3624 3625 3630 3631 3632 3633 3634 3639 3644 3645 3646
3647 3648 3649 3650 3651 3654 3668 3669 3670 3671 3672 3673 3674
3675 3676 3678 3679 3680 3681 3683 3684 3685 3686 3688 3689 3690
3691 3693 3694 3695 3696 3697 3699 3700 3701 3769 3770 3771 3772
3773 3774 3776 3777 3778 3779 3780 3781 3793 3794 3795 3797 3799
3800 3806 3807 3808 3809 3810 3818 3828 3829 3831 3832 3845 3846
3847 3850 3853 3854 3855 3858 3859 3861 3862 3863 3867 3868 3870
3872 3873 3889 3890 3894 3898 3901 3902 3903 3907 3908 3909 3922
3927 3934 3935 3951 3952 3953 3954 3956 3959 3972 3975 3977 3989
3990 3991 3992 3993 3994 3995 3996 3998 4001 4006 4007 4011 4012
4014 4015 4017 4018 4022 4025 4026 4050 4051 4054 4056 4057 4058
4061 4073 4074 4075 4076 4078 4079 4125 4127 4134 4140 4149 4150
4152 4153 4154 4157 4158 4161 4162 4163 4165 4166 4231 4237 4239
4240 4241 4242 4246 4247 4248 4252 4253 4254 4256 4257 4259 4261
4262 4267 4268 4271 4272 4275 4278 4280 4283 4288 4289 4290 4291
4294 4295 4296 4298 4299 4300 4302 4306 4310 4318 4320 4322 4323
4325 4326 4334 4340 4348 4350 4351 4357 4358 4360 4391 4392 4394
4396 4397 4399 4400 440[illegible] 4405 4406 4407 4412 4431 4454 4491 4492
4527 4528 4530 4531 4532 4534 4580 4581 4585 4589 4590 4591 4592
4593 4602 4605 4610 4617 4624 4631 4638 4645 4652 4659 4666 4669
4671 4674 4675 4678 4682 4684 4686 4687 4689 4691 4693 4695 4696
4698 4703 4714 4715 4716 4718 4917 4918 4919 4920 4922 4924 4926
4927 4928 4929 4930 4932 4933 4934 4937 4939 4941 4950 4951 4952
4954 4955 4956 4958 4994 4995 4996 4997 4998 4999 5000 5001 5002
5003 5012 5026 5041 5043 5044 5045 5046 5067 5069 5072 5073 5074

```
                    5077  5078  5089  5091  5092  5093  5094  5095  5096  5097  5098  5099  5101
                    5104  5109  5110  5112  5114  5116  5118  5119  5120  5121  5122  5123  5124
                    5126  5127  5129  5130  5133  5138  5139  5140  5145  5161  5162  5163  5164
                    5167  5169  5172  5174  5176  5184  5186  5205  5206  5209  5210  5211  5212
                    5213  5214  5216  5217  5218  5219  5220  5221  5222  5224  5241  5242  5243
                    5245  5246  5254  5255  5256  5260  5263  5267  5269  5270  5273  5277  5278
                    5282  5285  5288  5292  5294  5295  5296  5297  5300  5302  5307  5308  5309
                    5310  5314  5316  5317  5337  5338  5339  5340  5341  5355  5356  5357  5358
                    5359  5360
$FSBLA= 000170       1730#  3562  3589  4339
$FSCAS= 000150       1730#
$FSDEC= 000220       1730#
$FSDO = 000340       1730#  2665  2667  2687  2722  2755  2811  3013  3022  3146  3679  3684  3689
                     3800  3806  3846  3953  3994  4127  4240  4289  4320  4391  4454  4589  4675
                     4715  4932  5089  5104  5169  5260  5285  5294
$FSFAL= 000405       1730#
$FSGOO= 000400       1730#  2526  2527  2528  2530  2531  2532  2533  2543  2544  2545  2547  2583
                     2587  2591  2595  2606  2607  2608  2609  2610  2611  2612  2613  2614  2616
                     2617  2619  2622  2635  2651  2652  2653  2664  2665  2667  2669  2670  2671
                     2672  2673  2674  2675  2676  2677  2679  2685  2687  2688  2689  2690  2693
                     2695  2700  2703  2704  2709  2722  2723  2724  2725  2726  2734  2736  2737
                     2739  2743  2750  2751  2755  2756  2757  2758  2759  2767  2771  2788  2793
                     2796  2797  2798  2810  2811  2812  2814  2815  2827  2838  2839  2845  2846
                     2848  2854  2856  2864  2865  2872  2873  2874  2877  2879  2880  2881  2882
                     2885  2895  2896  2897  2898  2899  2919  2920  2922  2923  2924  2925  2933
                     2949  2973  2981  3000  3002  3003  3010  3011  3013  3017  3018  3020  3021
                     3022  3023  3024  3028  3029  3034  3036  3037  3038  3040  3041  3047  3063
                     3067  3070  3073  3076  3077  3079  3080  3082  3084  3085  3089  3095  3097
                     3109  3111  3123  3126  3141  3142  3143  3144  3145  3146  3147  3148  3150
                     3159  3160  3161  3162  3163  3164  3166  3167  3168  3169  3172  3173  3174
                     3181  3195  3199  3200  3201  3205  3206  3207  3209  3212  3214  3217  3218
                     3219  3221  3222  3230  3231  3254  3283  3287  3288  3292  3294  3296  3297
                     3298  3300  3302  3345  3349  3350  3354  3355  3356  3357  3359  3360  3361
                     3362  3363  3365  3366  3367  3369  3375  3378  3381  3382  3384  3385  3386
                     3389  3393  3394  3395  3396  3399  3403  3404  3405  3406  3410  3415  3429
                     3433  3434  3435  3436  3437  3442  3459  3484  3507  3514  3515  3516  3517
                     3524  3525  3548  3554  3556  3557  3560  3563  3564  3568  3571  3574  3575
                     3576  3577  3578  3579  3580  3581  3584  3585  3586  3590  3591  3620  3621
                     3622  3623  3624  3625  3630  3631  3632  3633  3634  3639  3644  3645  3646
                     3647  3648  3649  3650  3651  3654  3668  3669  3670  3671  3672  3673  3674
                     3675  3676  3678  3679  3680  3681  3683  3684  3685  3686  3688  3689  3690
                     3691  3693  3694  3695  3696  3697  3699  3700  3701  3769  3770  3771  3772
                     3773  3774  3776  3777  3778  3779  3780  3781  3793  3794  3795  3797  3799
                     3800  3806  3807  3808  3809  3810  3818  3828  3829  3831  3832  3845  3846
                     3847  3850  3853  3854  3855  3858  3859  3861  3862  3863  3867  3868  3870
                     3872  3873  3889  3890  3894  3898  3901  3902  3903  3907  3908  3909  3922
                     3927  3934  3935  3951  3952  3953  3954  3956  3959  3972  3975  3977  3989
                     3990  3991  3992  3993  3994  3995  3996  3998  4001  4006  4007  4011  4012
                     4014  4015  4017  4018  4022  4025  4026  4050  4051  4054  4056  4057  4058
                     4061  4073  4074  4075  4076  4078  4079  4125  4127  4134  4140  4149  4150
                     4152  4153  4154  4157  4158  4161  4162  4163  4165  4166  4231  4237  4239
                     4240  4241  4242  4246  4247  4248  4252  4253  4254  4256  4257  4259  4261
                     4262  4267  4268  4271  4272  4274  4275  4277  4278  4280  4283  4288  4289
                     4290  4291  4293  4294  4295  4296  4298  4299  4300  4302  4306  4310  4318
                     4320  4322  4323  4325  4326  4334  4340  4348  4350  4351  4357  4358  4360
                     4391  4392  4394  4396  4397  4399  4400  4403  4405  4406  4407  4412  4431
```

```
                    4454    4491    4492    4527    4528    4530    4531    4532    4534    4580    4581    4585    4589
                    4590    4591    4592    4593    4602    4605    4610    4617    4624    4631    4638    4645    4652
                    4659    4666    4669    4671    4674    4675    4678    4682    4684    4686    4687    4689    4691
                    4693    4695    4696    4698    4703    4714    4715    4716    4718    4917    4918    4919    4920
                    4922    4924    4926    4927    4928    4929    4930    4932    4933    4934    4937    4939    4941
                    4950    4951    4952    4954    4955    4956    4958    4994    4995    4996    4997    4998    4999
                    5000    5001    5002    5003    5012    5026    5041    5043    5044    5045    5046    5067    5069
                    5072    5073    5074    5077    5078    5089    5091    5092    5093    5094    5095    5096    5097
                    5098    5099    5101    5104    5109    5110    5112    5114    5116    5118    5119    5120    5121
                    5122    5123    5124    5126    5127    5129    5130    5133    5138    5139    5140    5145    5161
                    5162    5163    5164    5167    5169    5172    5174    5176    5184    5186    5205    5206    5209
                    5210    5211    5212    5213    5214    5216    5217    5218    5219    5220    5221    5222    5224
                    5241    5242    5243    5245    5246    5254    5255    5256    5260    5263    5267    5269    5270
                    5273    5277    5278    5282    5285    5288    5292    5294    5295    5296    5297    5300    5302
                    5307    5308    5309    5310    5314    5316    5317    5337    5338    5339    5340    5341    5355
                    5356    5357    5358    5359    5360
$FSIF = 000110       1730#  2545    2549    2606    2608    2613    2615    2618    2620    2621    2623    2669    2670
                    2671    2676    2678    2680    2681    2682    2688    2689    2691    2692    2693    2696    2697
                    2700    2702    2723    2724    2725    2726    2728    2729    2731    2732    2733    2734    2738
                    2739    2742    2743    2745    2746    2747    2750    2753    2756    2757    2758    2759    2761
                    2762    2764    2765    2766    2767    2770    2771    2773    2774    2775    2838    2840    2845
                    2847    2849    2873    2876    2878    2881    2883    2924    2926    3003    3008    3010    3019
                    3028    3033    3036    3039    3040    3041    3043    3044    3047    3051    3063    3065    3066
                    3070    3072    3073    3075    3076    3078    3081    3084    3091    3095    3099    3100    3159
                    3161    3163    3165    3166    3167    3168    3170    3171    3172    3173    3175    3176    3177
                    3178    3179    3180    3182    3195    3197    3198    3199    3200    3201    3203    3204    3205
                    3206    3208    3210    3211    3213    3216    3217    3218    3219    3221    3225    3226    3227
                    3228    3229    3230    3232    3283    3286    3287    3288    3290    3291    3292    3294    3296
                    3297    3299    3301    3305    3306    3307    3345    3347    3348    3349    3350    3353    3354
                    3357    3369    3371    3372    3375    3383    3387    3388    3390    3391    3393    3395    3397
                    3398    3399    3402    3405    3406    3408    3409    3414    3416    3417    3429    3432    3434
                    3441    3443    3484    3497    3507    3513    3515    3516    3522    3523    3524    3528    3529
                    3548    3556    3558    3559    3564    3566    3567    3568    3570    3571    3573    3574    3578
                    3582    3583    3587    3590    3591    3593    3594    3595    3630    3640    3644    3654    3657
                    3658    3668    3671    3675    3677    3693    3695    3698    3700    3701    3703    3704    3705
                    3706    3707    3708    3771    3776    3783    3784    3794    3796    3807    3808    3809    3810
                    3812    3813    3815    3816    3817    3818    3821    3822    3823    3830    3850    3852    3853
                    3854    3855    3857    3858    3860    3861    3864    3865    3867    3868    3871    3874    3875
                    3876    3877    3878    3890    3893    3894    3897    3903    3905    3907    3908    3909    3912
                    3913    3914    3927    3929    3935    3937    3938    3939    3956    3958    4001    4003    4005
                    4006    4009    4011    4013    4017    4018    4023    4024    4050    4051    4060    4062    4063
                    4134    4136    4152    4153    4155    4163    4167    4169    4231    4235    4237    4244    4246
                    4249    4272    4274    4275    4277    4279    4281    4282    4284    4285    4286    4293    4298
                    4299    4301    4303    4304    4305    4310    4315    4318    4325    4326    4328    4330    4331
                    4333    4340    4343    4347    4348    4351    4354    4355    4396    4402    4405    4406    4410
                    4411    4415    4416    4593    4595    4922    4925    5067    5070    5109    5111    5114    5130
                    5136    5137    5288    5290    5295    5296    5298    5299    5300    5303    5304    5307    5310
                    5312    5313    5315
$FSINC= 000210       1730#  4322    4334    4337    4338
$FSLOO= 000200       1730#
$FSNAM= 000160       1730#  3549    3551    3562    3589    4321    4339
$FSNO = 000403       1730#  2526    2527    2528    2530    2531    2532    2533    2543    2544    2545    2547    2583
                    2587    2591    2595    2606    2607    2608    2609    2610    2611    2612    2614    2616    2617
                    2619    2622    2635    2651    2652    2653    2664    2665    2667    2670    2672    2673    2674
                    2675    2676    2677    2679    2685    2687    2688    2689    2690    2693    2695    2700    2703
                    2704    2709    2722    2723    2724    2725    2734    2736    2737    2739    2751    2755    2756
```

	2757	2758	2767	2768	2793	2796	2797	2798	2810	2811	2812	2814	2815
	2827	2838	2839	2845	2846	2848	2854	2856	2864	2865	2872	2873	2874
	2877	2879	2880	2882	2885	2895	2896	2897	2898	2899	2919	2920	2922
	2923	2924	2925	2933	2949	2973	2981	3000	3002	3003	3010	3011	3013
	3017	3018	3020	3021	3022	3023	3024	3029	3034	3037	3038	3040	3041
	3047	3063	3067	3070	3073	3077	3079	3080	3082	3084	3085	3089	3097
	3109	3111	3123	3126	3141	3142	3143	3144	3145	3146	3147	3148	3150
	3160	3161	3162	3163	3164	3166	3167	3168	3169	3172	3173	3174	3181
	3195	3199	3200	3205	3206	3207	3209	3212	3214	3217	3218	3222	3231
	3254	3283	3287	3288	3292	3298	3300	3302	3345	3349	3355	3356	3359
	3360	3361	3362	3363	3365	3366	3367	3369	3375	3378	3381	3382	3384
	3385	3386	3389	3393	3394	3395	3396	3399	3403	3404	3406	3410	3415
	3429	3433	3435	3436	3437	3442	3459	3507	3514	3515	3516	3517	3524
	3525	3548	3554	3556	3557	3560	3563	3564	3569	3574	3575	3576	3577
	3578	3579	3580	3581	3584	3585	3586	3620	3621	3622	3623	3624	3625
	3631	3632	3633	3634	3639	3644	3645	3646	3647	3648	3649	3650	3651
	3654	3669	3670	3671	3672	3673	3674	3675	3676	3678	3679	3680	3681
	3683	3684	3685	3686	3688	3689	3690	3691	3694	3696	3697	3699	3769
	3770	3772	3773	3774	3776	3777	3778	3779	3780	3781	3793	3795	3797
	3799	3800	3806	3807	3808	3809	3818	3828	3829	3831	3832	3845	3846
	3847	3854	3855	3858	3859	3861	3862	3863	3867	3868	3870	3872	3873
	3889	3890	3894	3898	3901	3902	3907	3909	3922	3927	3934	3935	3951
	3952	3953	3954	3956	3959	3972	3975	3977	3989	3990	3991	3992	3993
	3994	3995	3996	3998	4001	4007	4011	4012	4014	4015	4017	4022	4025
	4026	4054	4056	4057	4058	4061	4073	4074	4075	4076	4078	4079	4125
	4127	4140	4149	4150	4152	4153	4154	4157	4158	4161	4162	4163	4165
	4166	4231	4239	4240	4241	4242	4247	4248	4252	4253	4254	4256	4257
	4259	4261	4262	4267	4268	4271	4278	4280	4283	4288	4289	4290	4291
	4294	4295	4296	4298	4299	4300	4302	4306	4310	4318	4320	4323	4325
	4326	4327	4340	4350	4351	4357	4358	4360	4391	4392	4394	4397	4399
	4400	4403	4405	4406	4407	4412	4431	4454	4491	4492	4527	4528	4530
	4531	4532	4534	4580	4581	4585	4589	4590	4591	4592	4593	4602	4605
	4610	4617	4624	4631	4638	4645	4652	4659	4666	4669	4671	4674	4675
	4678	4682	4684	4686	4687	4689	4691	4693	4695	4696	4698	4703	4714
	4715	4716	4718	4917	4918	4919	4920	4924	4926	4927	4928	4929	4930
	4932	4933	4934	4937	4939	4941	4950	4951	4952	4954	4955	4956	4958
	4994	4995	4996	4997	4998	4999	5000	5001	5002	5003	5012	5026	5041
	5043	5044	5045	5046	5069	5072	5073	5074	5077	5078	5089	5091	5092
	5093	5094	5095	5096	5097	5098	5099	5101	5104	5109	5110	5112	5114
	5116	5118	5119	5120	5121	5122	5123	5124	5126	5127	5129	5130	5133
	5138	5139	5140	5145	5161	5162	5163	5164	5167	5169	5172	5174	5176
	5184	5186	5205	5206	5209	5210	5211	5212	5213	5214	5216	5217	5218
	5219	5220	5221	5222	5224	5241	5242	5243	5245	5246	5254	5255	5256
	5260	5263	5267	5269	5270	5273	5277	5278	5282	5285	5288	5292	5294
	5295	5296	5297	5300	5302	5307	5308	5309	5310	5314	5316	5317	5337
	5338	5339	5340	5341	5355	5356	5357	5358	5359	5360			
$FSOR = 000320	1730#	2545	2606	2608	2613	2665	2667	2669	2670	2671	2676	2687	2688
	2689	2693	2700	2722	2723	2724	2725	2726	2734	2739	2743	2750	2755
	2756	2757	2758	2759	2767	2771	2811	2838	2845	2873	2881	2924	3003
	3010	3013	3022	3028	3036	3040	3041	3047	3063	3070	3073	3076	3084
	3095	3146	3159	3161	3163	3166	3167	3168	3172	3173	3195	3199	3200
	3201	3205	3205	3217	3218	3219	3221	3230	3283	3287	3288	3292	3294
	3296	3297	3345	3349	3350	3354	3357	3369	3375	3393	3395	3399	3405
	3406	3429	3434	3484	3507	3515	3516	3524	3548	3556	3564	3568	3571
	3574	3578	3590	3591	3630	3644	3654	3668	3671	3675	3679	3684	3689
	3693	3695	3700	3701	3771	3776	3794	3800	3806	3807	3808	3809	3810

```
                    3818    3828    3846    3850    3853    3854    3855    3858    3861    3867    3868    3890    3894
                    3903    3907    3908    3909    3927    3935    3953    3956    3994    4001    4006    4011    4017
                    4018    4050    4051    4127    4134    4152    4153    4163    4231    4237    4240    4246    4272
                    4275    4289    4298    4299    4310    4318    4320    4325    4326    4340    4348    4351    4391
                    4396    4405    4406    4454    4589    4593    4675    4715    4922    4932    5067    5089    5104
                    5109    5114    5130    5169    5260    5285    5288    5294    5295    5296    5300    5307    5310
$FSRTN= 000300       1730#
$FSSEL= 000140       1730#
$FSTHE= 000330       1730#   2545    2606    2608    2613    2669    2670    2671    2676    2688    2689    2693    2700
                    2723    2724    2725    2726    2734    2739    2743    2750    2756    2757    2758    2759    2767
                    2771    2838    2845    2873    2881    2924    3003    3010    3028    3036    3040    3041    3047
                    3063    3070    3073    3076    3084    3095    3159    3161    3163    3166    3167    3168    3172
                    3173    3195    3199    3200    3201    3205    3206    3217    3218    3219    3221    3230    3283
                    3287    3288    3292    3294    3296    3297    3345    3349    3350    3354    3357    3369    3375
                    3393    3395    3399    3405    3406    3429    3434    3484    3507    3515    3516    3524    3548
                    3556    3564    3568    3571    3574    3578    3590    3591    3630    3644    3654    3668    3671
                    3675    3693    3695    3700    3701    3771    3776    3794    3807    3808    3809    3810    3818
                    3828    3850    3853    3854    3855    3858    3861    3867    3868    3890    3894    3903    3907
                    3908    3909    3927    3935    3956    4001    4006    4011    4017    4018    4050    4051    4134
                    4152    4153    4163    4231    4237    4246    4272    4275    4298    4299    4310    4318    4325
                    4326    4340    4348    4351    4396    4405    4406    4593    4922    5067    5109    5114    5130
                    5288    5295    5296    5300    5307    5310
$FSTRU= 000404       1730#
$FSUNT= 000130       1730#   3001    3009    3068    3083    3110    3112    3124    3127    3364    3373    3550    3552
                    3561    3588    3906    3933    3974    3976    4159    4168    4255    4258    4260    4263    4931
                    4940    5168    5175
$FSWHI= 000120       1730#   2665    2667    2687    2693    2699    2705    2708    2722    2734    2739    2749    2755
                    2767    2777    2811    2813    3013    3016    3022    3025    3040    3073    3146    3149    3161
                    3205    3219    3283    3288    3294    3349    3350    3399    3679    3682    3684    3687    3689
                    3692    3701    3800    3806    3818    3826    3834    3846    3848    3953    3955    3994    3997
                    4127    4139    4240    4243    4289    4308    4318    4320    4346    4391    4418    4454    4458
                    4589    4597    4675    4677    4715    4717    4932    4936    5089    5104    5113    5143    5169
                    5171    5260    5285    5294    5300    5306    5318    5322
$FSYES= 000402       1730#   2526    2527    2528    2530    2531    2532    2533    2543    2544    2545    2547    2549
                    2583    2587    2591    2595    2606    2607    2608    2609    2610    2611    2612    2613    2614
                    2615    2616    2617    2618    2619    2620    2621    2622    2623    2635    2651    2652    2653
                    2664    2665    2667    2669    2670    2671    2672    2673    2674    2675    2676    2677    2678
                    2679    2680    2681    2682    2685    2687    2688    2689    2690    2691    2692    2693    2695
                    2696    2697    2700    2702    2703    2704    2709    2722    2723    2724    2725    2726    2728
                    2729    2731    2732    2733    2734    2736    2737    2738    2739    2742    2743    2745    2746
                    2747    2750    2751    2753    2755    2756    2757    2758    2759    2761    2762    2764    2765
                    2766    2767    2770    2771    2773    2774    2775    2788    2793    2796    2797    2798    2810
                    2811    2812    2814    2815    2827    2838    2839    2840    2845    2846    2847    2848    2849
                    2854    2856    2864    2865    2872    2873    2874    2876    2877    2878    2879    2880    2881
                    2882    2883    2885    2895    2896    2897    2898    2899    2919    2920    2922    2923    2924
                    2925    2926    2933    2949    2973    2981    3000    3002    3003    3008    3010    3011    3013
                    3017    3018    3019    3020    3021    3022    3023    3024    3028    3029    3033    3034    3036
                    3037    3038    3039    3040    3041    3043    3044    3047    3051    3063    3065    3066    3067
                    3070    3072    3073    3075    3076    3077    3078    3079    3080    3081    3082    3084    3085
                    3089    3091    3095    3097    3099    3100    3109    3111    3123    3126    3141    3142    3143
                    3144    3145    3146    3147    3148    3150    3159    3160    3161    3162    3163    3164    3165
                    3166    3167    3168    3169    3170    3171    3172    3173    3174    3175    3176    3177    3178
                    3179    3180    3181    3182    3195    3197    3198    3199    3200    3201    3203    3204    3205
                    3206    3207    3208    3209    3210    3211    3212    3213    3214    3216    3217    3218    3219
                    3221    3222    3225    3226    3227    3228    3229    3230    3231    3232    3254    3283    3286
                    3287    3288    3290    3291    3292    3294    3296    3297    3298    3299    3300    3301    3302
```

3305 3306 3307 3345 3347 3348 3349 3350 3353 3354 3355 3356 3357
3359 3360 3361 3362 3363 3365 3366 3367 3369 3371 3372 3375 3378
3381 3382 3383 3384 3385 3386 3387 3388 3389 3390 3391 3393 3394
3395 3396 3397 3398 3399 3402 3403 3404 3405 3406 3408 3409 3410
3414 3415 3416 3417 3429 3432 3433 3434 3435 3436 3437 3441 3442
3443 3459 3484 3497 3507 3513 3514 3515 3516 3517 3522 3523 3524
3525 3528 3529 3548 3554 3556 3557 3558 3559 3560 3563 3564 3566
3567 3568 3570 3571 3573 3574 3575 3576 3577 3578 3579 3580 3581
3582 3583 3584 3585 3586 3587 3590 3591 3593 3594 3595 3620 3621
3622 3623 3624 3625 3630 3631 3632 3633 3634 3639 3640 3644 3645
3646 3647 3648 3649 3650 3651 3654 3657 3658 3668 3669 3670 3671
3672 3673 3674 3675 3676 3677 3678 3679 3680 3681 3683 3684 3685
3686 3688 3689 3690 3691 3693 3694 3695 3696 3697 3698 3699 3700
3701 3703 3704 3705 3706 3707 3708 3769 3770 3771 3772 3773 3774
3776 3777 3778 3779 3780 3781 3783 3784 3793 3794 3795 3796 3797
3799 3800 3806 3807 3808 3809 3810 3812 3813 3815 3816 3817 3818
3821 3822 3828 3829 3830 3831 3832 3845 3846 3847 3850 3852 3853
3854 3855 3857 3858 3859 3860 3861 3862 3863 3864 3865 3867 3868
3870 3871 3872 3873 3874 3875 3876 3877 3878 3889 3890 3893 3894
3897 3898 3901 3902 3903 3905 3907 3908 3909 3912 3913 3914 3922
3927 3929 3934 3935 3937 3938 3939 3951 3952 3953 3954 3956 3958
3959 3972 3975 3977 3989 3990 3991 3992 3993 3994 3995 3996 3998
4001 4003 4005 4006 4007 4009 4011 4012 4013 4014 4015 4017 4018
4022 4023 4024 4025 4026 4050 4051 4054 4056 4057 4058 4060 4061
4062 4063 4073 4074 4075 4076 4078 4079 4125 4127 4134 4136 4140
4149 4150 4152 4153 4154 4155 4157 4158 4161 4162 4163 4165 4166
4167 4169 4231 4235 4237 4239 4240 4241 4242 4244 4246 4247 4248
4249 4252 4253 4254 4256 4257 4259 4261 4262 4267 4268 4271 4272
4274 4275 4277 4278 4279 4280 4281 4282 4283 4284 4285 4286 4288
4289 4290 4291 4293 4294 4295 4296 4298 4299 4300 4301 4302 4303
4304 4305 4306 4310 4315 4318 4320 4323 4325 4326 4328 4330 4331
4333 4340 4343 4347 4348 4350 4351 4354 4355 4357 4358 4360 4391
4392 4394 4396 4397 4399 4400 4402 4403 4405 4406 4407 4410 4411
4412 4415 4416 4431 4454 4491 4492 4527 4528 4530 4531 4532 4534
4580 4581 4585 4589 4590 4591 4592 4593 4595 4602 4605 4610 4617
4624 4631 4638 4645 4652 4659 4666 4669 4671 4674 4675 4678 4682
4684 4686 4687 4689 4691 4693 4695 4696 4698 4703 4714 4715 4716
4718 4917 4918 4919 4920 4922 4924 4925 4926 4927 4928 4929 4930
4932 4933 4934 4937 4939 4941 4950 4951 4952 4954 4955 4956 4958
4994 4995 4996 4997 4998 4999 5000 5001 5002 5003 5012 5026 5041
5043 5044 5045 5046 5067 5069 5070 5072 5073 5074 5077 5078 5089
5091 5092 5093 5094 5095 5096 5097 5098 5099 5101 5104 5109 5110
5111 5112 5114 5116 5118 5119 5120 5121 5122 5123 5124 5126 5127
5129 5130 5133 5136 5137 5138 5139 5140 5145 5161 5162 5163 5164
5167 5169 5172 5174 5176 5184 5186 5205 5206 5209 5210 5211 5212
5213 5214 5216 5217 5218 5219 5220 5221 5222 5224 5241 5242 5243
5245 5246 5254 5255 5256 5260 5263 5267 5269 5270 5273 5277 5278
5282 5285 5288 5290 5292 5294 5295 5296 5297 5298 5299 5300 5302
5303 5304 5307 5308 5309 5310 5312 5313 5314 5315 5316 5317 5337
5338 5339 5340 5341 5355 5356 5357 5358 5359 5360

$IFLEV= 177777

1730# 2545# 2549# 2606# 2608# 2613# 2615# 2620# 2623# 2669# 2670# 2671# 2676#
2678# 2680# 2681# 2682# 2688# 2689# 2691# 2693# 2696# 2697# 2700# 2702# 2723#
2724# 2725# 2726# 2728# 2731# 2732# 2734# 2738# 2739# 2743# 2745# 2746# 2747#
2750# 2753# 2756# 2757# 2758# 2759# 2761# 2764# 2765# 2767# 2771# 2773# 2774#
2775# 2838# 2840# 2845# 2849# 2873# 2878# 2881# 2883# 2924# 2926# 3003# 3008#
3010# 3019# 3028# 3033# 3036# 3039# 3040# 3041# 3043# 3044# 3047# 3051# 3063#

PARAMETER CODING MACY11 30(1046) 12-JUL-83 09:44 PAGE 80-23
CZTUVB.P11 12-JUL-83 09:26 CROSS REFERENCE TABLE -- USER SYMBOLS SEQ 0195

```
                    3066#  3070#  3072#  3073#  3075#  3076#  3081#  3084#  3095#  3099#  3100#  3159#  3161#
                    3163#  3166#  3167#  3168#  3170#  3172#  3173#  3175#  3176#  3177#  3178#  3179#  3180#
                    3182#  3195#  3198#  3199#  3200#  3201#  3203#  3204#  3205#  3206#  3210#  3213#  3217#
                    3218#  3219#  3221#  3225#  3226#  3227#  3228#  3229#  3230#  3232#  3283#  3286#  3287#
                    3288#  3290#  3291#  3292#  3294#  3296#  3297#  3301#  3305#  3306#  3307#  3345#  3348#
                    3349#  3350#  3353#  3354#  3357#  3369#  3372#  3375#  3383#  3387#  3390#  3393#  3395#
                    3397#  3398#  3399#  3402#  3405#  3406#  3409#  3416#  3417#  3429#  3432#  3434#  3443#
                    3484#  3497#  3507#  3513#  3515#  3516#  3522#  3524#  3528#  3529#  3548#  3556#  3559#
                    3564#  3567#  3568#  3571#  3573#  3574#  3578#  3582#  3583#  3587#  3590#  3591#  3593#
                    3594#  3595#  3630#  3640#  3644#  3654#  3657#  3658#  3668#  3671#  3675#  3677#  3693#
                    3695#  3700#  3701#  3703#  3704#  3705#  3706#  3707#  3708#  3771#  3776#  3783#  3784#
                    3794#  3796#  3807#  3808#  3809#  3810#  3812#  3815#  3816#  3818#  3821#  3822#  3828#
                    3830#  3850#  3852#  3853#  3854#  3855#  3858#  3861#  3864#  3865#  3867#  3868#  3874#
                    3875#  3876#  3877#  3878#  3890#  3894#  3903#  3905#  3907#  3908#  3909#  3912#  3913#
                    3914#  3927#  3929#  3935#  3937#  3938#  3939#  3956#  3958#  4001#  4005#  4006#  4009#
                    4011#  4013#  4017#  4018#  4023#  4024#  4050#  4051#  4062#  4063#  4134#  4136#  4152#
                    4153#  4155#  4163#  4167#  4169#  4231#  4235#  4237#  4244#  4246#  4249#  4272#  4274#
                    4275#  4277#  4281#  4284#  4285#  4286#  4293#  4298#  4299#  4303#  4304#  4305#  4310#
                    4315#  4318#  4325#  4326#  4330#  4333#  4340#  4343#  4347#  4348#  4351#  4354#  4355#
                    4396#  4405#  4406#  4410#  4415#  4416#  4593#  4595#  4922#  4925#  5067#  5070#  5109#
                    5111#  5114#  5130#  5136#  5137#  5288#  5290#  5295#  5296#  5298#  5300#  5303#  5304#
                    5307#  5310#  5312#  5315#
$ISK0 = 000001      2545#  2549   2606#  2623   2669#  2682   2688#  2697   2700#  2702   2723#  2747   2750#
                    2753   2756#  2775   2838#  2840   2845#  2849   2873#  2878   2881#  2883   2924#  2926
                    3003#  3008   3010#  3019   3028#  3033   3036#  3039   3040#  3044   3047#  3051   3063#
                    3066   3070#  3072   3073#  3075   3076#  3081   3084#  3100   3159#  3182   3195#  3198
                    3199#  3229   3230#  3232   3283#  3286   3287#  3291   3292#  3307   3345#  3348   3349#
                    3417   3429#  3432   3434#  3443   3484#  3497   3507#  3513   3515#  3529   3548#  3595
                    3630#  3640   3644#  3658   3668#  3708   3771#  3784   3794#  3796   3807#  3822   3828#
                    3830   3850#  3852   3853#  3878   3890#  3939   3956#  3958   4001#  4005   4006#  4009
                    4011#  4013   4017#  4024   4050#  4063   4134#  4136   4152#  4169   4231#  4235   4237#
                    4244   4246#  4249   4272#  4286   4293#  4305   4310#  4315   4318#  4347   4348#  4355
                    4396#  4416   4593#  4595   4922#  4925   5067#  5070   5109#  5111   5114#  5137   5288#
                    5290   5295#  5304   5307#  5315
$ISK1 = 000001      2608#  2620   2670#  2681   2689#  2691   2693#  2696   2724#  2732   2734#  2738   2739#
                    2746   2757#  2765   2767#  2774   3041#  3043   3095#  3099   3161#  3180   3200#  3204
                    3205#  3213   3217#  3228   3288#  3290   3294#  3306   3350#  3353   3354#  3390   3393#
                    3398   3399#  3402   3405#  3416   3516#  3522   3524#  3528   3556#  3559   3564#  3567
                    3568#  3587   3590#  3594   3654#  3657   3671#  3707   3776#  3783   3808#  3816   3818#
                    3821   3854#  3877   3894#  3938   4018#  4023   4051#  4062   4153#  4155   4163#  4167
                    4274#  4285   4298#  4304   4325#  4333   4340#  4343   4351#  4354   4405#  4415   5130#
                    5136   5296#  5298   5300#  5303   5310#  5312
$ISK2 = 000001      2613#  2615   2671#  2680   2725#  2731   2743#  2745   2758#  2764   2771#  2773   3163#
                    3179   3201#  3203   3206#  3210   3218#  3227   3296#  3305   3357#  3387   3395#  3397
                    3406#  3409   3571#  3573   3574#  3583   3591#  3593   3675#  3677   3693#  3706   3809#
                    3815   3855#  3876   3903#  3905   3907#  3914   3927#  3929   3935#  3937   4275#  4284
                    4299#  4303   4326#  4330   4406#  4410
$ISK3 = 000001      2676#  2678   2726#  2728   2759#  2761   3166#  3178   3219#  3226   3297#  3301   3369#
                    3372   3375#  3383   3578#  3582   3695#  3705   3810#  3812   3858#  3865   3867#  3875
                    3908#  3913   4277#  4281
$ISK4 = 000001      3167#  3177   3221#  3225   3700#  3704   3861#  3864   3868#  3874   3909#  3912
$ISK5 = 000001      3168#  3170   3172#  3176   3701#  3703
$ISK6 = 000001      3173#  3175
$LOCTA= 177777      1730#  2545   2549   2606   2608   2613   2615   2618   2620   2621   2623   2665   2667
                    2669   2670   2671   2676   2678   2680   2681   2682   2687   2688   2689   2691   2692
                    2693   2696   2697   2699   2700   2702   2705   2708   2722   2723   2724   2725   2726
```

```
                    2728  2729  2731  2732  2733  2734  2738  2739  2742  2743  2745  2746  2747
                    2749  2750  2753  2755  2756  2757  2758  2759  2761  2762  2764  2765  2766
                    2767  2770  2771  2773  2774  2775  2777  2811  2813  2838  2840  2845  2847
                    2849  2873  2876  2878  2881  2883  2924  2926  3001  3003  3008  3009  3010
                    3013  3016  3019  3022  3025  3028  3033  3036  3039  3040  3041  3043  3044
                    3047  3051  3063  3065  3066  3068  3070  3072  3073  3075  3076  3078  3081
                    3083  3084  3091  3095  3099  3100  3110  3112  3124  3127  3146  3149  3159
                    3161  3163  3165  3166  3167  3168  3170  3171  3172  3173  3175  3176  3177
                    3178  3179  3180  3182  3195  3197  3198  3199  3200  3201  3203  3204  3205
                    3206  3208  3210  3211  3213  3216  3217  3218  3219  3221  3225  3226  3227
                    3228  3229  3230  3232  3283  3286  3287  3288  3290  3291  3292  3294  3296
                    3297  3299  3301  3305  3306  3307  3345  3347  3348  3349  3350  3353  3354
                    3357  3364  3369  3371  3372  3373  3375  3383  3387  3388  3390  3391  3393
                    3395  3397  3398  3399  3402  3405  3406  3408  3409  3414  3416  3417  3429
                    3432  3434  3441  3443  3484  3497  3507  3513  3515  3516  3522  3523  3524
                    3528  3529  3548  3550  3552  3556  3558  3559  3561  3562  3564  3566  3567
                    3568  3569  3570  3571  3573  3574  3578  3582  3583  3587  3588  3589  3590
                    3591  3593  3594  3595  3630  3640  3644  3654  3657  3658  3668  3671  3675
                    3677  3679  3682  3684  3687  3689  3692  3693  3695  3698  3700  3701  3703
                    3704  3705  3706  3707  3708  3771  3776  3783  3784  3794  3796  3800  3806
                    3807  3808  3809  3810  3812  3813  3815  3816  3817  3818  3821  3822  3826
                    3828  3830  3834  3846  3848  3850  3852  3853  3854  3855  3857  3858  3860
                    3861  3864  3865  3867  3868  3871  3874  3875  3876  3877  3878  3890  3893
                    3894  3897  3903  3905  3906  3907  3908  3909  3912  3913  3914  3927  3929
                    3933  3935  3937  3938  3939  3953  3955  3956  3958  3974  3976  3994  3997
                    4001  4003  4005  4006  4009  4011  4013  4017  4018  4023  4024  4050  4051
                    4060  4062  4063  4127  4134  4136  4139  4152  4153  4155  4159  4163  4167
                    4168  4169  4231  4235  4237  4240  4243  4244  4246  4249  4255  4258  4260
                    4263  4272  4274  4275  4277  4279  4281  4282  4284  4285  4286  4289  4293
                    4298  4299  4301  4303  4304  4305  4308  4310  4315  4318  4320  4322  4325
                    4326  4327  4328  4330  4331  4333  4334  4337  4338  4339  4340  4343  4346
                    4347  4348  4351  4354  4355  4391  4396  4402  4405  4406  4410  4411  4415
                    4416  4418  4454  4458  4589  4593  4595  4597  4675  4677  4715  4717  4922
                    4925  4931  4932  4936  4940  5067  5070  5089  5104  5109  5111  5113  5114
                    5130  5136  5137  5143  5168  5169  5171  5175  5260  5285  5288  5290  5294
                    5295  5296  5298  5299  5300  5303  5304  5306  5307  5310  5312  5313  5315
                    5318  5322
$LSTCN= 177777      1730#  2545  2549  2606  2608  2613  2615  2618  2620  2621  2623  2665  2667
                    2669  2670  2671  2676  2678  2680  2681  2682  2687  2688  2689  2691  2692
                    2693  2696  2697  2699  2700  2702  2705  2708  2722  2723  2724  2725  2726
                    2728  2729  2731  2732  2733  2734  2738  2739  2742  2743  2745  2746  2747
                    2749  2750  2753  2755  2756  2757  2758  2759  2761  2762  2764  2765  2766
                    2767  2770  2771  2773  2774  2775  2777  2811  2813  2838  2840  2845  2847
                    2849  2873  2876  2878  2881  2883  2924  2926  3001  3003  3008  3009  3010
                    3013  3016  3019  3022  3025  3028  3033  3036  3039  3040  3041  3043  3044
                    3047  3051  3063  3065  3066  3068  3070  3072  3073  3075  3076  3078  3081
                    3083  3084  3091  3095  3099  3100  3110  3112  3124  3127  3146  3149  3159
                    3161  3163  3165  3166  3167  3168  3170  3171  3172  3173  3175  3176  3177
                    3178  3179  3180  3182  3195  3197  3198  3199  3200  3201  3203  3204  3205
                    3206  3208  3210  3211  3213  3216  3217  3218  3219  3221  3225  3226  3227
                    3228  3229  3230  3232  3283  3286  3287  3288  3290  3291  3292  3294  3296
                    3297  3299  3301  3305  3306  3307  3345  3347  3348  3349  3350  3353  3354
                    3357  3364  3369  3371  3372  3373  3375  3383  3387  3388  3390  3391  3393
                    3395  3397  3398  3399  3402  3405  3406  3408  3409  3414  3416  3417  3429
                    3432  3434  3441  3443  3484  3497  3507  3513  3515  3516  3522  3523  3524
                    3528  3529  3548  3549  3550  3551  3552  3556  3558  3559  3561  3562  3564
```

```
                    3566  3567  3568  3570  3571  3573  3574  3578  3582  3583  3587  3588  3589
                    3590  3591  3593  3594  3595  3630  3640  3644  3654  3657  3658  3668  3671
                    3675  3677  3679  3682  3684  3687  3689  3692  3693  3695  3698  3700  3701
                    3703  3704  3705  3706  3707  3708  3771  3776  3783  3784  3794  3796  3800
                    3806  3807  3808  3809  3810  3812  3813  3815  3816  3817  3818  3821  3822
                    3826  3828  3830  3834  3846  3848  3850  3852  3853  3854  3855  3857  3858
                    3860  3861  3864  3865  3867  3868  3871  3874  3875  3876  3877  3878  3890
                    3893  3894  3897  3903  3905  3906  3907  3908  3909  3912  3913  3914  3927
                    3929  3933  3935  3937  3938  3939  3953  3955  3956  3958  3974  3976  3994
                    3997  4001  4003  4005  4006  4009  4011  4013  4017  4018  4023  4024  4050
                    4051  4060  4062  4063  4127  4134  4136  4139  4152  4153  4155  4159  4163
                    4167  4168  4169  4231  4235  4237  4240  4243  4244  4246  4249  4255  4258
                    4260  4263  4272  4274  4275  4277  4279  4281  4282  4284  4285  4286  4289
                    4293  4298  4299  4301  4303  4304  4305  4308  4310  4315  4318  4320  4321
                    4322  4325  4326  4328  4330  4331  4333  4334  4337  4338  4339  4340  4343
                    4346  4347  4348  4351  4354  4355  4391  4396  4402  4405  4406  4410  4411
                    4415  4416  4418  4454  4458  4589  4593  4595  4597  4675  4677  4715  4717
                    4922  4925  4931  4932  4936  4940  5067  5070  5089  5104  5109  5111  5113
                    5114  5130  5136  5137  5143  5168  5169  5171  5175  5260  5285  5288  5290
                    5294  5295  5296  5298  5299  5300  5303  5304  5306  5307  5310  5312  5313
                    5315  5318  5322
$LSTIN= 000001      1730#  1737#  2526  2527  2528  2530  2531  2532  2533  2543  2544  2545  2547
                    2583  2587  2591  2595  2606  2607  2608  2609  2610  2611  2612  2613  2614
                    2616  2617  2618  2619  2621  2622  2635  2651  2652  2653  2664  2665  2667
                    2669  2670  2671  2672  2673  2674  2675  2676  2677  2679  2685  2687  2688
                    2689  2690  2692  2693  2695  2699  2700  2703  2704  2705  2708  2709  2722
                    2723  2724  2725  2726  2729  2733  2734  2736  2737  2739  2742  2743  2749
                    2750  2751  2755  2756  2757  2758  2759  2762  2766  2767  2770  2771  2777
                    2788  2793  2796  2797  2798  2810  2811  2812  2813  2814  2815  2827  2838
                    2839  2845  2846  2847  2848  2854  2856  2864  2865  2872  2873  2874  2876
                    2877  2879  2880  2881  2882  2885  2895  2896  2897  2898  2899  2919  2920
                    2922  2923  2924  2925  2933  2949  2973  2981  3000  3002  3003  3009  3010
                    3011  3013  3016  3017  3018  3020  3021  3022  3023  3024  3025  3028  3029
                    3034  3036  3037  3038  3040  3041  3047  3063  3065  3067  3070  3073  3076
                    3077  3078  3079  3080  3082  3083  3084  3085  3089  3091  3095  3097  3109
                    3111  3112  3123  3126  3127  3141  3142  3143  3144  3145  3146  3147  3148
                    3149  3150  3159  3160  3161  3162  3163  3164  3165  3166  3167  3168  3169
                    3171  3172  3173  3174  3181  3195  3197  3199  3200  3201  3205  3206  3207
                    3208  3209  3211  3212  3214  3216  3217  3218  3219  3221  3222  3230  3231
                    3254  3283  3287  3288  3292  3294  3296  3297  3298  3299  3300  3302  3345
                    3347  3349  3350  3354  3355  3356  3357  3359  3360  3361  3362  3363  3365
                    3366  3367  3369  3371  3373  3375  3378  3381  3382  3384  3385  3386  3388
                    3389  3391  3393  3394  3395  3396  3399  3403  3404  3405  3406  3408  3410
                    3411  3414  3415  3429  3433  3434  3435  3436  3437  3438  3441  3442  3459
                    3484  3507  3511  3514  3515  3516  3517  3521  3523  3524  3525  3527  3548
                    3554  3556  3557  3558  3560  3561  3563  3564  3566  3568  3569  3571  3574
                    3575  3576  3577  3578  3579  3580  3581  3584  3585  3586  3588  3590  3591
                    3620  3621  3622  3623  3624  3625  3630  3631  3632  3633  3634  3639  3644
                    3645  3646  3647  3648  3649  3650  3651  3654  3668  3669  3670  3671  3672
                    3673  3674  3675  3676  3678  3679  3680  3681  3682  3683  3684  3685  3686
                    3687  3688  3689  3690  3691  3692  3693  3694  3695  3696  3697  3698  3699
                    3700  3701  3769  3770  3771  3772  3773  3774  3776  3777  3778  3779  3780
                    3781  3793  3794  3795  3797  3799  3800  3806  3807  3808  3809  3810  3813
                    3817  3818  3826  3828  3829  3831  3832  3834  3845  3846  3847  3848  3850
                    3853  3854  3855  3857  3858  3859  3860  3861  3862  3863  3867  3868  3870
                    3871  3872  3873  3889  3890  3893  3894  3897  3898  3901  3902  3903  3907
```

3908 3909 3922 3927 3933 3934 3935 3951 3952 3953 3954 3955 3956
3959 3972 3975 3976 3977 3989 3990 3991 3992 3993 3994 3995 3996
3997 3998 4001 4003 4006 4007 4011 4012 4014 4015 4017 4018 4022
4025 4026 4050 4051 4054 4056 4057 4058 4060 4061 4073 4075
4076 4078 4079 4125 4127 4134 4139 4140 4149 4150 4152 4154
4157 4158 4161 4162 4163 4165 4166 4168 4231 4237 4239 4240 4241
4242 4243 4246 4247 4248 4252 4253 4254 4256 4257 4258 4259 4261
4262 4263 4267 4268 4271 4272 4274 4275 4277 4278 4279 4280 4282
4283 4288 4289 4290 4291 4293 4294 4295 4296 4298 4299 4300 4301
4302 4306 4308 4310 4318 4320 4322 4323 4325 4326 4327 4331 4334
4337 4338 4340 4346 4348 4350 4351 4357 4358 4360 4391 4392 4394
4396 4397 4399 4400 4402 4403 4405 4406 4407 4411 4412 4418 4431
4454 4458 4491 4492 4527 4528 4530 4531 4532 4534 4580 4581 4585
4589 4590 4591 4592 4593 4597 4602 4605 4610 4617 4624 4631 4638
4645 4652 4659 4666 4669 4671 4674 4675 4677 4678 4682 4684 4686
4687 4689 4691 4693 4695 4696 4698 4703 4714 4715 4716 4717 4718
4917 4918 4919 4920 4922 4924 4926 4927 4928 4929 4930 4932 4933
4934 4936 4937 4939 4940 4941 4950 4951 4952 4954 4955 4956 4958
4994 4995 4996 4997 4998 4999 5000 5001 5002 5003 5012 5026 5041
5043 5044 5045 5046 5067 5069 5072 5073 5074 5077 5078 5089 5091
5092 5093 5094 5095 5096 5097 5098 5099 5101 5104 5109 5110 5112
5113 5114 5116 5118 5119 5120 5121 5122 5123 5124 5126 5127 5129
5130 5133 5138 5139 5140 5143 5145 5161 5162 5163 5164 5167 5169
5171 5172 5174 5175 5176 5184 5186 5205 5206 5209 5210 5211 5212
5213 5214 5216 5217 5218 5219 5220 5221 522[illegible] 5224 5241 5242 5243
5245 5246 5254 5255 5256 5260 5263 5267 5269 5270 5273 5277 5278
5282 5285 5288 5292 5294 5295 5296 5297 5299 5300 5302 5306 5307
5308 5309 5310 5313 5314 5316 5317 5318 5322 5337 5338 5339 5340
5341 5355 5356 5357 5358 5359 5360

$LSTST= 177777

1730# 2545 2549 2606 2608 2613 2615 2618 2620 2621 2623 2665 2667
2669 2670 2671 2676 2678 2680 2681 2682 2687 2688 2689 2691 2692
2693 2696 2697 2699 2700 2702 2705 2708 2722 2723 2724 2725 2726
2728 2729 2731 2732 2733 2734 2738 2739 2742 2743 2745 2746 2747
2749 2750 2753 2755 2756 2757 2758 2759 2761 2762 2764 2765 2766
2767 2770 2771 2773 2774 2775 2777 2811 2813 2838 2840 2845 2847
2849 2873 2876 2878 2881 2883 2924 2926 3001 3003 3008 3009 3010
3013 3016 3019 3022 3025 3028 3033 3036 3039 3040 3041 3043 3044
3047 3051 3063 3065 3066 3068 3070 3072 3073 3075 3076 3078 3081
3083 3084 3091 3095 3099 3100 3110 3112 3124 3127 3146 3149 3159
3161 3163 3165 3166 3167 3168 3170 3171 3172 3173 3175 3176 3177
3178 3179 3180 3182 3195 3197 3198 3199 3200 3201 3203 3204 3205
3206 3208 3210 3211 3213 3216 3217 3218 3219 3221 3225 3226 3227
3228 3229 3230 3232 3283 3286 3287 3288 3290 3291 3292 3294 3296
3297 3299 3301 3305 3306 3307 3345 3347 3348 3349 3350 3353 3354
3357 3364 3369 3371 3372 3373 3375 3383 3387 3388 3390 3391 3393
3395 3397 3398 3399 3402 3405 3406 3408 3409 3414 3416 3417 3429
3432 3434 3441 3443 3484 3497 3507 3513 3515 3516 3522 3523 3524
3528 3529 3548 3549 3550 3551 3552 3556 3558 3559 3561 3562 3564
3566 3567 3568 3570 3571 3573 3574 3578 3582 3583 3587 3588 3589
3590 3591 3593 3594 3595 3630 3640 3644 365[illegible] 3657 3658 3668 3671
3675 3677 3679 3682 3684 3687 3689 3692 3693 3695 3698 3700 3701
3703 3704 3705 3706 3707 3708 3771 3776 3783 3784 3794 3796 3800
3806 3807 3808 3809 3810 3812 3813 3815 3816 3817 3818 3821 3822
3826 3828 3830 3834 3846 3848 3850 3852 3853 3854 3855 3857 3858
3860 3861 3864 3865 3867 3868 3871 3874 3875 3876 3877 3878 3890
3893 3894 3897 3903 3905 3906 3907 3908 3909 3912 3913 3914 3927

```
                    3929  3933  3935  3937  3938  3939  3953  3955  3956  3958  3974  3976  3994
                    3997  4001  4003  4005  4006  4009  4011  4013  4017  4018  4023  4024  4050
                    4051  4060  4062  4063  4127  4134  4136  4139  4152  4153  4155  4159  4163
                    4167  4168  4169  4231  4235  4237  4240  4243  4244  4246  4249  4255  4258
                    4260  4263  4272  4274  4275  4277  4279  4281  4282  4284  4285  4286  4289
                    4293  4298  4299  4301  4303  4304  4305  4308  4310  4315  4318  4320  4321
                    4322  4325  4326  4328  4330  4331  4333  4334  4337  4338  4339  4340  4343
                    4346  4347  4348  4351  4354  4355  4391  4396  4402  4405  4406  4410  4411
                    4415  4416  4418  4454  4458  4589  4593  4595  4597  4675  4677  4715  4717
                    4922  4925  4931  4932  4936  4940  5067  5070  5089  5104  5109  5111  5113
                    5114  5130  5136  5137  5143  5168  5169  5171  5175  5260  5285  5288  5290
                    5294  5295  5296  5298  5299  5300  5303  5304  5306  5307  5310  5312  5313
                    5315  5318  5322
$LSTTA= 000001       1730#  1738#  2549  2615  2618  2620  2621  2623  2665  2667  2678  2680  2681
                    2682  2687  2691  2692  2693  2696  2697  2699  2702  2705  2708  2722  2728
                    2729  2731  2732  2733  2738  2739  2742  2745  2746  2747  2749  2753  2755
                    2761  2762  2764  2765  2766  2767  2770  2773  2774  2775  2777  2811  2813
                    2840  2847  2849  2876  2878  2883  2926  3001  3008  3013  3016  3019  3022
                    3025  3033  3039  3043  3044  3051  3065  3066  3068  3072  3073  3075  3078
                    3081  3083  3091  3099  3100  3110  3124  3127  3146  3149  3165  3170  3171
                    3175  3176  3177  3178  3179  3180  3182  3197  3198  3203  3204  3205  3208
                    3210  3211  3213  3216  3219  3225  3226  3227  3228  3229  3232  3286  3290
                    3291  3294  3299  3301  3305  3306  3307  3347  3348  3353  3364  3371  3372
                    3373  3383  3387  3388  3390  3391  3397  3398  3402  3408  3409  3414  3416
                    3417  3432  3441  3443  3497  3513  3522  3523  3528  3529  3550  3552  3558
                    3559  3561  3562  3566  3567  3570  3573  3582  3583  3587  3589  3593  3594
                    3595  3640  3657  3658  3677  3679  3682  3684  3687  3689  3692  3698  3703
                    3704  3705  3706  3707  3708  3783  3784  3796  3800  3806  3812  3813  3815
                    3816  3817  3818  3821  3822  3826  3830  3834  3846  3848  3852  3857  3860
                    3864  3865  3871  3874  3875  3876  3877  3878  3893  3897  3905  3906  3912
                    3913  3914  3929  3937  3938  3939  3953  3955  3958  3974  3994  3997  4003
                    4005  4009  4013  4023  4024  4060  4062  4063  4127  4136  4139  4155  4159
                    4167  4169  4235  4240  4243  4244  4249  4255  4260  4279  4281  4282  4284
                    4285  4286  4289  4301  4303  4304  4305  4308  4315  4320  4322  4328  4330
                    4331  4333  4334  4337  4338  4339  4343  4346  4347  4354  4355  4391  4402
                    4410  4411  4415  4416  4418  4454  4458  4589  4595  4597  4675  4677  4715
                    4717  4925  4931  4932  4936  5070  5089  5104  5111  5113  5136  5137  5143
                    5168  5169  5171  5260  5285  5290  5294  5298  5299  5300  5303  5304  5306
                    5312  5313  5315  5318  5322
$MCALL= ****** U     1730
$NESTL= 177777       1730#  2545#  2549#  2606#  2608#  2613#  2615#  2618  2620#  2621  2623#  2665#  2667#
                    2669#  2670#  2671#  2676#  2678#  2680#  2681#  2682#  2687#  2688#  2689#  2691#  2692
                    2693#  2696#  2697#  2699#  2700#  2702#  2705#  2708#  2722#  2723#  2724#  2725#  2726#
                    2728#  2729   2731#  2732#  2733   2734#  2738#  2739#  2742   2743#  2745#  2746#  2747#
                    2749#  2750#  2753#  2755#  2756#  2757#  2758#  2759#  2761#  2762   2764#  2765#  2766
                    2767#  2770   2771#  2773#  2774#  2775#  2777#  2811#  2813#  2838#  2840#  2845#  2847
                    2849#  2873#  2876   2878#  2881#  2883#  2924#  2926#  3001#  3003#  3008#  3009#  3010#
                    3013#  3016#  3019#  3022#  3025#  3028#  3033#  3036#  3039#  3040#  3041#  3043#  3044#
                    3047#  3051#  3063#  3065   3066#  3068#  3070#  3072#  3073#  3075#  3076#  3078   3081#
                    3083#  3084#  3091   3095#  3099#  3100#  3110#  3112#  3124#  3127#  3146#  3149#  3159#
                    3161#  3163#  3165   3166#  3167#  3168#  3170#  3171   3172#  3173#  3175#  3176#  3177#
                    3178#  3179#  3180#  3182#  3195#  3197   3198#  3199#  3200#  3201#  3203#  3204#  3205#
                    3206#  3208   3210#  3211   3213#  3216   3217#  3218#  3219#  3221#  3225#  3226#  3227#
                    3228#  3229#  3230#  3232#  3283#  3286#  3287#  3288#  3290#  3291#  3292#  3294#  3296#
                    3297#  3299   3301#  3305#  3306#  3307#  3345#  3347   3348#  3349#  3350#  3353#  3354#
                    3357#  3364#  3369#  3371   3372#  3373#  3375#  3383#  3387#  3388   3390#  3391   3393#
```

```
                    3395#  3397#  3398#  3399#  3402#  3405#  3406#  3408   3409#  3414   3416#  3417#  3429#
                    3432#  3434#  3441   3443#  3484#  3497#  3507#  3513#  3515#  3516#  3522#  3523   3524#
                    3528#  3529#  3548#  3549#  3550#  3551#  3552#  3556#  3558   3559#  3561#  3562#  3564#
                    3566   3567#  3568#  3570   3571#  3573#  3574#  3578#  3582#  3583#  3587#  3588#  3589#
                    3590#  3591#  3593#  3594#  3595#  3630#  3640#  3644#  3654#  3657#  3658#  3668#  3671#
                    3675#  3677#  3679#  3682#  3684#  3687#  3689#  3692#  3693#  3695#  3698   3700#  3701#
                    3703#  3704#  3705#  3706#  3707#  3708#  3771#  3776#  3783#  3784#  3794#  3796#  3800#
                    3806#  3807#  3808#  3809#  3810#  3812#  3813   3815#  3816#  3817   3818#  3821#  3822#
                    3826#  3828#  3830#  3834#  3846#  3848#  3850#  3852#  3853#  3854#  3855#  3857   3858#
                    3860   3861#  3864#  3865#  3867#  3868#  3871   3874#  3875#  3876#  3877#  3878#  3890#
                    3893   3894#  3897   3903#  3905#  3906#  3907#  3908#  3909#  3912#  3913#  3914#  3927#
                    3929#  3933#  3935#  3937#  3938#  3939#  3953#  3955#  3956#  3958#  3974#  3976#  3994#
                    3997#  4001#  4003   4005#  4006#  4009#  4011#  4013#  4017#  4018#  4023#  4024#  4050#
                    4051#  4060   4062#  4063#  4127#  4134#  4136#  4139#  4152#  4153#  4155#  4159#  4163#
                    4167#  4168#  4169#  4231#  4235#  4237#  4240#  4243#  4244#  4246#  4249#  4255#  4258#
                    4260#  4263#  4272#  4274#  4275#  4277#  4279   4281#  4282   4284#  4285#  4286#  4289#
                    4293#  4298#  4299#  4301   4303#  4304#  4305#  4308#  4310#  4315#  4318#  4320#  4321#
                    4322#  4325#  4326#  4328   4330#  4331   4333#  4334#  4337#  4338#  4339#  4340#  4343#
                    4346#  4347#  4348#  4351#  4354#  4355#  4391#  4396#  4402   4405#  4406#  4410#  4411
                    4415#  4416#  4418#  4454#  4458#  4589#  4593#  4595#  4597#  4675#  4677#  4715#  4717#
                    4922#  4925#  4931#  4932#  4936#  4940#  5067#  5070#  5089#  5104#  5109#  5111#  5113#
                    5114#  5130#  5136#  5137#  5143#  5168#  5169#  5171#  5175#  5260#  5285#  5288#  5290#
                    5294#  5295#  5296#  5298#  5299   5300#  5303#  5304#  5306#  5307#  5310#  5312#  5313
                    5315#  5318#  5322#
$NSK0 = 000120      2545#  2549   2606#  2621   2623   2665#  2708   2722#  2749   2750#  2753   2755#  2777
                    2811#  2813   2838#  2840   2845#  2847   2849   2873#  2876   2878   2881#  2883   2924#
                    2926   3001#  3009   3010#  3019   3022#  3025   3028#  3033   3036#  3039   3040#  3044
                    3047#  3051   3063#  3065   3066   3068#  3083   3084#  3091   3100   3110#  3112   3124#
                    3127   3146#  3149   3159#  3182   3195#  3197   3198   3199#  3216   3229   3230#  3232
                    3283#  3286   3287#  3291   3292#  3307   3345#  3347   3348   3349#  3391   3417   3429#
                    3432   3434#  3441   3443   3484#  3497   3507#  3513   3515#  3523   3529   3548#  3595
                    3630#  3640   3644#  3658   3668#  3708   3771#  3784   3794#  3796   3800#  3834   3846#
                    3848   3850#  3852   3853#  3878   3890#  3893   3939   3953#  3955   3956#  3958   3974#
                    3976   3994#  3997   4001#  4003   4005   4006#  4009   4011#  4013   4017#  4024   4050#
                    4063   4127#  4139   4152#  4169   4231#  4235   4237#  4244   4246#  4249   4255#  4258
                    4260#  4263   4272#  4286   4289#  4308   4310#  4315   4318#  4347   4348#  4355   4391#
                    4418   4454#  4458   4589#  4597   4675#  4677   4715#  4717   4922#  4925   4931#  4940
                    5067#  5070   5089#  5143   5168#  5175   5260#  5322
$NSK1 = 000120      2608#  2618   2620   2667#  2705   2723#  2733   2747   2756#  2766   2775   3003#  3008
                    3013#  3016   3041#  3043   3070#  3072   3073#  3075   3076#  3078   3081   3095#  3099
                    3161#  3180   3200#  3204   3205#  3211   3213   3217#  3228   3288#  3290   3294#  3306
                    3350#  3353   3354#  3388   3390   3393#  3398   3399#  3402   3405#  3414   3416   3516#
                    3522   3524#  3528   3549#  3589   3590#  3594   3654#  3657   3671#  3707   3776#  3783
                    3806#  3826   3828#  3830   3854#  3877   3894#  3897   3938   4018#  4023   4051#  4060
                    4062   4134#  4136   4153#  4155   4159#  4168   4240#  4243   4274#  4285   4293#  4305
                    4320#  4346   4351#  4354   4396#  4402   4416   4593#  4595   4932#  4936   5104#  5113
                    5114#  5137   5169#  5171   5285#  5318
$NSK2 = 000110      2613#  2615   2669#  2682   2687#  2699   2700#  2702   2724#  2732   2734#  2738   2739#
                    2742   2746   2757#  2765   2767#  2770   2774   3163#  3165   3179   3201#  3203   3206#
                    3208   3210   3218#  3227   3296#  3305   3357#  3387   3395#  3397   3406#  3408   3409
                    3550#  3588   3591#  3593   3675#  3677   3679#  3682   3684#  3687   3689#  3692   3693#
                    3706   3807#  3817   3822   3855#  3857   3876   3903#  3905   3906#  3933   3935#  3937
                    4163#  4167   4275#  4282   4284   4298#  4304   4321#  4339   4340#  4343   4405#  4411
                    4415   5109#  5111   5130#  5136   5288#  5290   5294#  5306   5307#  5313   5315
$NSK3 = 000110      2670#  2681   2688#  2692   2697   2725#  2729   2731   2743#  2745   2758#  2762   2764
                    2771#  2773   3166#  3178   3219#  3226   3297#  3299   3301   3364#  3373   3375#  3383
```

```
                    3551#  3562   3564#  3566   3567   3568#  3570   3587   3695#  3698   3705   3808#  3816
                    3818#  3821   3858#  3860   3865   3867#  3875   3907#  3914   3927#  3929   4277#  4279
                    4281   4299#  4301   4303   4322#  4338   4406#  4410   5295#  5299   5304   5310#  5312
$NSK4 = 000110      2671#  2680   2689#  2691   2693#  2696   2726#  2728   2759#  2761   3167#  3171   3177
                    3221#  3225   3369#  3371   3372   3552#  3561   3571#  3573   3574#  3583   3700#  3704
                    3809#  3813   3815   3861#  3864   3868#  3871   3874   3908#  3913   4325#  4331   4333
                    4334#  4337   5296#  5298   5300#  5303
$NSK5 = 000110      2676#  2678   3168#  3170   3172#  3176   3556#  3558   3559   3578#  3582   3701#  3703
                    3810#  3812   3909#  3912   4326#  4328   4330
$NSK6 = 000110      3173#  3175
$SAVLE= 177777      1730#  2699#  2705#  2708#  2749#  2777#  2813#  3016#  3025#  3149#  3682#  3687#  3692#
                    3826#  3834#  3848#  3955#  3997#  4139#  4243#  4308#  4322#  4334#  4346#  4418#  4458#
                    4597#  4677#  4717#  4936#  5113#  5143#  5171#  5306#  5318#  5322#
$SSK0 = 050474      2699#  2705#  2708#  2749#  2777#  2813#  3016#  3025#  3149#  3682#  3687#  3692#  3826#
                    3834#  3848#  3955#  3997#  4139#  4243#  4308#  4322#  4334#  4346#  4418#  4458#  4597#
                    4677#  4717#  4936#  5113#  5143#  5171#  5306#  5318#  5322#
$TAGLE= 177777      1730#  2545#  2549#  2606#  2608#  2613#  2615#  2618#  2620#  2621#  2623#  2665#  2667#
                    2669#  2670#  2671#  2676#  2678#  2680#  2681#  2682#  2687#  2688#  2689#  2691#  2692#
                    2693#  2696#  2697#  2699#  2700#  2702#  2705#  2708#  2722#  2723#  2724#  2725#  2726#
                    2728#  2729#  2731#  2732#  2733#  2734#  2738#  2739#  2742#  2743#  2745#  2746#  2747#
                    2749#  2750#  2753#  2755#  2756#  2757#  2758#  2759#  2761#  2762#  2764#  2765#  2766#
                    2767#  2770#  2771#  2773#  2774#  2775#  2777#  2811#  2813#  2838#  2840#  2845#  2847#
                    2849#  2873#  2876#  2878#  2881#  2883#  2924#  2926#  3001#  3003#  3008#  3009#  3010#
                    3013#  3016#  3019#  3022#  3025#  3028#  3033#  3036#  3039#  3040#  3041#  3043#  3044#
                    3047#  3051#  3063#  3065#  3066#  3068#  3070#  3072#  3073#  3075#  3076#  3078#  3081#
                    3083#  3084#  3091#  3095#  3099#  3100#  3110#  3112#  3124#  3127#  3146#  3149#  3159#
                    3161#  3163#  3165#  3166#  3167#  3168#  3170#  3171#  3172#  3173#  3175#  3176#  3177#
                    3178#  3179#  3180#  3182#  3195#  3197#  3198#  3199#  3200#  3201#  3203#  3204#  3205#
                    3206#  3208#  3210#  3211#  3213#  3216#  3217#  3218#  3219#  3221#  3225#  3226#  3227#
                    3228#  3229#  3230#  3232#  3283#  3286#  3287#  3288#  3290#  3291#  3292#  3294#  3296#
                    3297#  3299#  3301#  3305#  3306#  3307#  3345#  3347#  3348#  3349#  3350#  3353#  3354#
                    3357#  3364#  3369#  3371#  3372#  3373#  3375#  3383#  3387#  3388#  3390#  3391#  3393#
                    3395#  3397#  3398#  3399#  3402#  3405#  3406#  3408#  3409#  3414#  3416#  3417#  3429#
                    3432#  3434#  3441#  3443#  3484#  3497#  3507#  3513#  3515#  3516#  3522#  3523#  3524#
                    3528#  3529#  3548#  3550#  3552#  3556#  3558#  3559#  3561#  3564#  3566#  3567#  3568#
                    3570#  3571#  3573#  3574#  3578#  3582#  3583#  3587#  3588#  3590#  3591#  3593#  3594#
                    3595#  3630#  3640#  3644#  3654#  3657#  3658#  3668#  3671#  3675#  3677#  3679#  3682#
                    3684#  3687#  3689#  3692#  3693#  3695#  3698#  3700#  3701#  3703#  3704#  3705#  3706#
                    3707#  3708#  3771#  3776#  3783#  3784#  3794#  3796#  3800#  3806#  3807#  3808#  3809#
                    3810#  3812#  3813#  3815#  3816#  3817#  3819#  3821#  3822#  3826#  3828#  3830#  3834#
                    3846#  3848#  3850#  3852#  3853#  3854#  3855#  3857#  3858#  3860#  3861#  3864#  3865#
                    3867#  3868#  3871#  3874#  3875#  3876#  3877#  3878#  3890#  3893#  3894#  3897#  3903#
                    3905#  3906#  3907#  3908#  3909#  3912#  3913#  3914#  3927#  3929#  3933#  3935#  3937#
                    3938#  3939#  3953#  3955#  3956#  3958#  3974#  3976#  3994#  3997#  4001#  4003#  4005#
                    4006#  4009#  4011#  4013#  4017#  4018#  4023#  4024#  4050#  4051#  4060#  4062#  4063#
                    4127#  4134#  4136#  4139#  4152#  4153#  4155#  4159#  4163#  4167#  4168#  4169#  4231#
                    4235#  4237#  4240#  4243#  4244#  4246#  4249#  4255#  4258#  4260#  4263#  4272#  4274#
                    4275#  4277#  4279#  4281#  4282#  4284#  4285#  4286#  4289#  4293#  4298#  4299#  4301#
                    4303#  4304#  4305#  4308#  4310#  4315#  4318#  4320#  4322#  4325#  4326#  4328#  4330#
                    4331#  4333#  4334#  4337#  4338#  4340#  4343#  4346#  4347#  4348#  4351#  4354#  4355#
                    4391#  4396#  4402#  4405#  4406#  4410#  4411#  4415#  4416#  4418#  4454#  4458#  4589#
                    4593#  4595#  4597#  4675#  4677#  4715#  4717#  4922#  4925#  4931#  4932#  4936#  4940#
                    5067#  5070#  5089#  5104#  5109#  5111#  5113#  5114#  5130#  5136#  5137#  5143#  5168#
                    5169#  5171#  5175#  5260#  5285#  5288#  5290#  5294#  5295#  5296#  5298#  5299#  5300#
                    5303#  5304#  5306#  5307#  5310#  5312#  5313#  5315#  5318#  5322#
$TAGNU= 050512      1730#  2545#  2606#  2608#  2613#  2618#  2621#  2665#  2667#  2669#  2670#  2671#  2676#
```

$TEMP = 000402

2687# 2688# 2689# 2692# 2693# 2700# 2722# 2723# 2724# 2725# 2726# 2729# 2733#
2734# 2739# 2742# 2743# 2750# 2755# 2756# 2757# 2758# 2759# 2762# 2766# 2767#
2770# 2771# 2811# 2838# 2845# 2847# 2873# 2876# 2881# 2924# 3001# 3003# 3010#
3013# 3022# 3028# 3036# 3040# 3041# 3047# 3063# 3065# 3068# 3070# 3073# 3076#
3078# 3083# 3084# 3091# 3095# 3110# 3124# 3127# 3146# 3159# 3161# 3163# 3165#
3166# 3167# 3168# 3171# 3172# 3173# 3195# 3197# 3199# 3200# 3201# 3205# 3206#
3208# 3211# 3216# 3217# 3218# 3219# 3221# 3230# 3283# 3287# 3288# 3292# 3294#
3296# 3297# 3299# 3345# 3347# 3349# 3350# 3354# 3357# 3364# 3369# 3371# 3373#
3375# 3388# 3391# 3393# 3395# 3399# 3405# 3406# 3408# 3414# 3429# 3434# 3441#
3484# 3507# 3515# 3516# 3523# 3524# 3548# 3549# 3550# 3551# 3552# 3556# 3558#
3561# 3564# 3566# 3568# 3570# 3571# 3574# 3578# 3590# 3591# 3630# 3644# 3654#
3668# 3671# 3675# 3679# 3684# 3689# 3693# 3695# 3698# 3700# 3701# 3771# 3776#
3794# 3800# 3806# 3807# 3808# 3809# 3810# 3813# 3817# 3818# 3828# 3846# 3850#
3853# 3854# 3855# 3857# 3858# 3860# 3861# 3867# 3868# 3871# 3890# 3893# 3894#
3897# 3903# 3906# 3907# 3908# 3909# 3927# 3935# 3953# 3956# 3974# 3994# 4001#
4003# 4006# 4011# 4017# 4018# 4050# 4051# 4060# 4127# 4134# 4152# 4153# 4159#
4163# 4231# 4237# 4240# 4246# 4255# 4260# 4272# 4274# 4275# 4277# 4279# 4282#
4289# 4293# 4298# 4299# 4301# 4310# 4318# 4320# 4321# 4322# 4325# 4326# 4328#
4331# 4334# 4340# 4348# 4351# 4391# 4396# 4402# 4405# 4406# 4411# 4454# 4589#
4593# 4675# 4715# 4922# 4931# 4932# 5067# 5089# 5104# 5109# 5114# 5130# 5168#
5169# 5260# 5285# 5288# 5294# 5295# 5296# 5299# 5300# 5307# 5310# 5313#
2526# 2527# 2528# 2530# 2531# 2532# 2533# 2543# 2544# 2547# 2549# 2583# 2587#
2591# 2595# 2607# 2609# 2610# 2611# 2612# 2614# 2615# 2616# 2617# 2618# 2619#
2620# 2621# 2622# 2623# 2635# 2551# 2652# 2653# 2664# 2672# 2673# 2674# 2675#
2677# 2678# 2679# 2680# 2681# 2682# 2685# 2690# 2691# 2692# 2695# 2696# 2697#
2699# 2702# 2703# 2704# 2705# 2708# 2709# 2728# 2729# 2731# 2732# 2733# 2736#
2737# 2738# 2742# 2745# 2746# 2747# 2749# 2751# 2753# 2761# 2762# 2764# 2765#
2766# 2770# 2773# 2774# 2775# 2777# 2788# 2793# 2796# 2797# 2798# 2810# 2812#
2813# 2814# 2815# 2827# 2839# 2840# 2846# 2847# 2848# 2849# 2854# 2856# 2864#
2865# 2872# 2874# 2876# 2877# 2878# 2879# 2880# 2882# 2883# 2885# 2895# 2896#
2897# 2898# 2899# 2919# 2920# 2922# 2923# 2925# 2926# 2933# 2949# 2973# 2981#
3000# 3002# 3008# 3009# 3011# 3016# 3017# 3018# 3019# 3020# 3021# 3023# 3024#
3025# 3029# 3033# 3034# 3037# 3038# 3039# 3043# 3044# 3051# 3065# 3066# 3067#
3072# 3075# 3077# 3078# 3079# 3080# 3081# 3082# 3083# 3085# 3089# 3091# 3097#
3099# 3100# 3109# 3111# 3112# 3123# 3126# 3127# 3141# 3142# 3143# 3144# 3145#
3147# 3148# 3149# 3150# 3160# 3162# 3164# 3165# 3169# 3170# 3171# 3174# 3175#
3176# 3177# 3178# 3179# 3180# 3181# 3182# 3197# 3198# 3203# 3204# 3207# 3208#
3209# 3210# 3211# 3212# 3213# 3214# 3216# 3222# 3225# 3226# 3227# 3228# 3229#
3231# 3232# 3254# 3286# 3290# 3291# 3298# 3299# 3300# 3301# 3302# 3305# 3306#
3307# 3347# 3348# 3353# 3355# 3356# 3359# 3360# 3361# 3362# 3363# 3365# 3366#
3367# 3371# 3372# 3373# 3378# 3381# 3382# 3383# 3384# 3385# 3386# 3387# 3388#
3389# 3390# 3391# 3394# 3396# 3397# 3398# 3402# 3403# 3404# 3408# 3409# 3410#
3414# 3415# 3416# 3417# 3432# 3433# 3435# 3436# 3437# 3441# 3442# 3443# 3459#
3497# 3513# 3514# 3517# 3522# 3523# 3525# 3528# 3529# 3554# 3557# 3558# 3559#
3560# 3561# 3562# 3563# 3566# 3567# 3570# 3573# 3575# 3576# 3577# 3579# 3580#
3581# 3582# 3583# 3584# 3585# 3586# 3587# 3588# 3589# 3593# 3594# 3595# 3620#
3621# 3622# 3623# 3624# 3625# 3631# 3632# 3633# 3634# 3639# 3640# 3645# 3646#
3647# 3648# 3649# 3650# 3651# 3657# 3658# 3669# 3670# 3672# 3673# 3674# 3676#
3677# 3678# 3680# 3681# 3682# 3683# 3685# 3686# 3687# 3688# 3690# 3691# 3692#
3694# 3696# 3697# 3698# 3699# 3703# 3704# 3705# 3706# 3707# 3708# 3769# 3770#
3772# 3773# 3774# 3777# 3778# 3779# 3780# 3781# 3783# 3784# 3793# 3795# 3796#
3797# 3799# 3812# 3813# 3815# 3816# 3817# 3821# 3822# 3826# 3829# 3830# 3831#
3832# 3834# 3845# 3847# 3848# 3852# 3857# 3859# 3860# 3862# 3863# 3864# 3865#
3870# 3871# 3872# 3873# 3874# 3875# 3876# 3877# 3878# 3889# 3893# 3897# 3898#
3901# 3902# 3905# 3912# 3913# 3914# 3922# 3929# 3933# 3934# 3937# 3938# 3939#
3951# 3952# 3954# 3955# 3958# 3959# 3972# 3975# 3976# 3977# 3989# 3990# 3991#

```
                    3992#  3993#  3995#  3996#  3997#  3998#  4003#  4005#  4007#  4009#  4012#  4013#  4014#
                    4015#  4022#  4023#  4024#  4025#  4026#  4054#  4056#  4057#  4058#  4060#  4061#  4062#
                    4063#  4073#  4074#  4075#  4076#  4078#  4079#  4125#  4136#  4139#  4140#  4149#  4150#
                    4154#  4155#  4157#  4158#  4161#  4162#  4165#  4166#  4167#  4168#  4169#  4235#  4239#
                    4241#  4242#  4243#  4244#  4247#  4248#  4249#  4252#  4253#  4254#  4256#  4257#  4258#
                    4259#  4261#  4262#  4263#  4267#  4268#  4271#  4278#  4279#  4280#  4281#  4282#  4283#
                    4284#  4285#  4286#  4288#  4290#  4291#  4294#  4295#  4296#  4300#  4301#  4302#  4303#
                    4304#  4305#  4306#  4308#  4315#  4322#  4323#  4328#  4330#  4331#  4333#  4334#  4337#
                    4338#  4339#  4343#  4346#  4347#  4350#  4354#  4355#  4357#  4358#  4360#  4392#  4394#
                    4397#  4399#  4400#  4402#  4403#  4407#  4410#  4411#  4412#  4415#  4416#  4418#  4431#
                    4458#  4491#  4492#  4527#  4528#  4530#  4531#  4532#  4534#  4580#  4581#  4585#  4590#
                    4591#  4592#  4595#  4597#  4602#  4605#  4610#  4617#  4624#  4631#  4638#  4645#  4652#
                    4659#  4666#  4669#  4671#  4674#  4677#  4678#  4682#  4684#  4686#  4687#  4689#  4691#
                    4693#  4695#  4696#  4698#  4703#  4714#  4716#  4717#  4718#  4917#  4918#  4919#  4920#
                    4924#  4925#  4926#  4927#  4928#  4929#  4930#  4933#  4934#  4936#  4937#  4939#  4940#
                    4941#  4950#  4951#  4952#  4954#  4955#  4956#  4958#  4994#  4995#  4996#  4997#  4998#
                    4999#  5000#  5001#  5002#  5003#  5012#  5026#  5041#  5043#  5044#  5045#  5046#  5069#
                    5070#  5072#  5073#  5074#  5077#  5078#  5091#  5092#  5093#  5094#  5095#  5096#  5097#
                    5098#  5099#  5101#  5110#  5111#  5112#  5113#  5116#  5118#  5119#  5120#  5121#  5122#
                    5123#  5124#  5126#  5127#  5129#  5133#  5136#  5137#  5138#  5139#  5140#  5143#  5145#
                    5161#  5162#  5163#  5164#  5167#  5171#  5172#  5174#  5175#  5176#  5184#  5186#  5205#
                    5206#  5209#  5210#  5211#  5212#  5213#  5214#  5216#  5217#  5218#  5219#  5220#  5221#
                    5222#  5224#  5241#  5242#  5243#  5245#  5246#  5254#  5255#  5256#  5263#  5267#  5269#
                    5270#  5273#  5277#  5278#  5282#  5290#  5292#  5297#  5298#  5299#  5302#  5303#  5304#
                    5306#  5308#  5309#  5312#  5313#  5314#  5315#  5316#  5317#  5318#  5322#  5337#  5338#
                    5339#  5340#  5341#  5355#  5356#  5357#  5358#  5359#  5360#
$TSK0 = 050473      2545#  2549   2606#  2621#  2623   2665#  2708   2722#  2749   2750#  2753   2755#  2777
                    2811#  2813   2838#  2840   2845#  2847#  2849   2873#  2876#  2878   2881#  2883   2924#
                    2926   3001#  3009   3010#  3019   3022#  3025   3028#  3033   3036#  3039   3040#  3044
                    3047#  3051   3063#  3065#  3066   3068#  3083   3084#  3091#  3100   3110#  3112   3124#
                    3127   3146#  3149   3159#  3182   3195#  3197#  3198   3199#  3216#  3229   3230#  3232
                    3283#  3286   3287#  3291   3292#  3307   3345#  3347#  3348   3349#  3391#  3417   3429#
                    3432   3434#  3441#  3443   3484#  3497   3507#  3513   3515#  3523#  3529   3548#  3595
                    3630#  3640   3644#  3658   3668#  3708   3771#  3784   3794#  3796   3800#  3834   3846#
                    3848   3850#  3852   3853#  3878   3890#  3893#  3939   3953#  3955   3956#  3958   3974#
                    3976   3994#  3997   4001#  4003#  4005   4006#  4009   4011#  4013   4017#  4024   4050#
                    4063   4127#  4139   4152#  4169   4231#  4235   4237#  4244   4246#  4249   4255#  4258
                    4260#  4263   4272#  4286   4289#  4308   4310#  4315   4318#  4347   4348#  4355   4391#
                    4418   4454#  4458   4589#  4597   4675#  4677   4715#  4717   4922#  4925   4931#  4940
                    5067#  5070   5089#  5143   5168#  5175   5260#  5322
$TSK1 = 050474      2608#  2618#  2620   2665#  2708   2722#  2749   2755#  2777   2811#  2813   3003#  3008
                    3013#  3016   3022#  3025   3041#  3043   3070#  3072   3073#  3075   3076#  3078#  3081
                    3095#  3099   3146#  3149   3161#  3180   3200#  3204   3205#  3211#  3213   3217#  3228
                    3288#  3290   3294#  3306   3350#  3353   3354#  3388#  3390   3393#  3398   3399#  3402
                    3405#  3414#  3416   3516#  3522   3524#  3528   3550#  3588   3590#  3594   3654#  3657
                    3671#  3707   3776#  3783   3800#  3834   3846#  3848   3854#  3877   3894#  3897#  3938
                    3953#  3955   3994#  3997   4018#  4023   4051#  4060#  4062   4127#  4139   4153#  4155
                    4159#  4168   4240#  4243   4274#  4285   4289#  4308   4320#  4346   4351#  4354   4391#
                    4418   4454#  4458   4589#  4597   4675#  4677   4715#  4717   4932#  4936   5089#  5143
                    5169#  5171   5260#  5322
$TSK2 = 050475      2613#  2615   2667#  2705   2723#  2733#  2747   2756#  2766#  2775   3013#  3016   3163#
                    3165#  3179   3201#  3203   3206#  3208#  3210   3218#  3227   3296#  3305   3357#  3387
                    3395#  3397   3406#  3408#  3409   3552#  3561   3564#  3566#  3567   3568#  3570#  3587
                    3591#  3593   3675#  3677   3679#  3682   3684#  3687   3689#  3692   3693#  3706   3806#
                    3826   3828#  3830   3855#  3857#  3876   3903#  3905   3906#  3933   3935#  3937   4134#
                    4136   4163#  4167   4240#  4243   4275#  4282#  4284   4293#  4305   4320#  4346   4396#
```

```
                    4402#  4416   4593#  4595   4932#  4936   5104#  5113   5114#  5137   5169#  5171   5285#
                    5318
$TSK3 = 050476      2667#  2705   2724#  2732   2734#  2738   2739#  2742#  2746   2757#  2765   2767#  2770#
                    2774   3166#  3178   3219#  3226   3297#  3299#  3301   3364#  3373   3375#  3383   3556#
                    3558#  3559   3571#  3573   3574#  3583   3679#  3682   3684#  3687   3689#  3692   3695#
                    3698#  3705   3806#  3826   3858#  3860#  3865   3867#  3875   3907#  3914   3927#  3929
                    4277#  4279#  4281   4298#  4304   4322#  4338   4340#  4343   4405#  4411#  4415   5104#
                    5113   5130#  5136   5285#  5318
$TSK4 = 050511      2669#  2682   2687#  2699   2700#  2702   2725#  2729#  2731   2743#  2745   2758#  2762#
                    2764   2771#  2773   3167#  3171#  3177   3221#  3225   3369#  3371#  3372   3578#  3582
                    3700#  3704   3807#  3817#  3822   3861#  3864   3868#  3871#  3874   3908#  3913   4299#
                    4301#  4303   4322#  4338   4406#  4410   5109#  5111   5288#  5290   5294#  5306   5307#
                    5313#  5315
$TSK5 = 050510      2670#  2681   2687#  2699   2726#  2728   2759#  2761   3168#  3170   3172#  3176   3701#
                    3703   3808#  3816   3818#  3821   3909#  3912   4325#  4331#  4333   4334#  4337   5294#
                    5306   5310#  5312
$TSK6 = 050504      2671#  2680   2688#  2692#  2697   3173#  3175   3809#  3813#  3815   4326#  4328#  4330
                    4334#  4337   5295#  5299#  5304
$TSK7 = 050506      2676#  2678   2689#  2691   2693#  2696   3810#  3812   5296#  5298   5300#  5303
$$ARGC= 000000      1730#
$$BYTE= 000403      1730#  2545#  2606#  2608#  2613#  2665#  2667#  2669#  2670#  2671#  2676#  2687#  2688#
                    2689#  2693#  2700#  2722#  2723#  2724#  2725#  2726#  2734#  2739#  2743#  2750#  2755#
                    2756#  2757#  2753#  2759#  2767#  2771#  2811#  2838#  2845#  2873#  2881#  2924#  3003#
                    3010#  3013#  3022#  3028#  3036#  3040#  3041#  3047#  3063#  3070#  3073#  3076#  3084#
                    3095#  3146#  3159#  3161#  3163#  3166#  3167#  3168#  3172#  3173#  3195#  3199#  3200#
                    3201#  3205#  3206#  3217#  3218#  3219#  3221#  3230#  3283#  3287#  3288#  3292#  3294#
                    3296#  3297#  3345#  3349#  3350#  3354#  3357#  3369#  3375#  3393#  3395#  3399#  3405#
                    3406#  3429#  3434#  3484#  3507#  3515#  3516#  3524#  3548#  3556#  3564#  3568#  3571#
                    3574#  3578#  3590#  3591#  3630#  3644#  3654#  3668#  3671#  3675#  3679#  3684#  3689#
                    3693#  3695#  3700#  3701#  3771#  3776#  3794#  3800#  3806#  3807#  3808#  3809#  3810#
                    3818#  3828#  3846#  3850#  3853#  3854#  3855#  3858#  3861#  3867#  3868#  3890#  3894#
                    3903#  3907#  3908#  3909#  3927#  3935#  3953#  3956#  3994#  4001#  4006#  4011#  4017#
                    4018#  4050#  4051#  4127#  4134#  4152#  4153#  4163#  4231#  4237#  4240#  4246#  4272#
                    4275#  4289#  4298#  4299#  4310#  4318#  4320#  4325#  4326#  4340#  4348#  4351#  4391#
                    4396#  4405#  4406#  4454#  4589#  4593#  4675#  4715#  4922#  4932#  5067#  5089#  5104#
                    5109#  5114#  5130#  5169#  5260#  5285#  5288#  5294#  5295#  5296#  5300#  5307#  5310#
$$CASE= 000000      1730#
$$DST = 000000      1730#
$$ELOC= 000402      1730#  2545#  2549#  2606#  2608#  2613#  2615#  2618   2620#  2621   2623#  2669#  2670#
                    2671#  2676#  2678#  2680#  2681#  2682#  2688#  2689#  2691#  2692   2693#  2696#  2697#
                    2700#  2702#  2723#  2724#  2725#  2726#  2728#  2729   2731#  2732#  2733   2734#  2738#
                    2739#  2742   2743#  2745#  2746#  2747#  2750#  2753#  2756#  2757#  2758#  2759#  2761#
                    2762   2764#  2765#  2766   2767#  2770   2771#  2773#  2774#  2775#  2838#  2840#  2845#
                    2847   2849#  2873#  2876   2878#  2881#  2883#  2924#  2926#  3003#  3008#  3010#  3019#
                    3028#  3033#  3036#  3039#  3040#  3041#  3043#  3044#  3047#  3051#  3063#  3065   3066#
                    3070#  3072#  3073#  3075#  3076#  3078   3081#  3084#  3091   3095#  3099#  3100#  3159#
                    3161#  3163#  3165   3166#  3167#  3168#  3170#  3171   3172#  3173#  3175#  3176#  3177#
                    3178#  3179#  3180#  3182#  3195#  3197   3198#  3199#  3200#  3201#  3203#  3204#  3205#
                    3206#  3208   3210#  3211   3213#  3216   3217#  3218#  3219#  3221#  3225#  3226#  3227#
                    3228#  3229#  3230#  3232#  3283#  3286#  3287#  3288#  3290#  3291#  3292#  3294#  3296#
                    3297#  3299   3301#  3305#  3306#  3307#  3345#  3347   3348#  3349#  3350#  3353#  3354#
                    3357#  3369#  3371   3372#  3375#  3383#  3387#  3388   3390#  3391   3393#  3395#  3397#
                    3398#  3399#  3402#  3405#  3406#  3408   3409#  3414   3416#  3417#  3429#  3432#  3434#
                    3441   3443#  3484#  3497#  3507#  3513#  3515#  3516#  3522#  3523   3524#  3528#  3529#
                    3548#  3556#  3558   3559#  3564#  3566   3567#  3568#  3569#  3570   3571#  3573#  3574#
                    3578#  3582#  3583#  3587#  3590#  3591#  3593#  3594#  3595#  3630#  3640#  3644#  3654#
```

```
                    3657#  3658#  3668#  3671#  3675#  3677#  3693#  3695#  3698   3700#  3701#  3703#  3704#
                    3705#  3706#  3707#  3708#  3771#  3776#  3783#  3784#  3794#  3796#  3807#  3808#  3809#
                    3810#  3812#  3813   3815#  3816#  3817   3818#  3821#  3822#  3828#  3830#  3850#  3852#
                    3853#  3854#  3855#  3857   3858#  3860   3861#  3864#  3865#  3867#  3868#  3371   3874#
                    3875#  3876#  3877#  3878#  3890#  3893   3894#  3897   3903#  3905#  3907#  3908#  3909#
                    3912#  3913#  3914#  3927#  3929#  3935#  3937#  3938#  3939#  3956#  3958#  4001#  4003
                    4005#  4006#  4009#  4011#  4013#  4017#  4018#  4023#  4024#  4050#  4051#  4060   4062#
                    4063#  4134#  4136#  4152#  4153#  4155#  4163#  4167#  4169#  4231#  4235#  4237#  4244#
                    4246#  4249#  4272#  4274#  4275#  4277#  4279   4281#  4282   4284#  4285#  4286#  4293#
                    4298#  4299#  4301   4303#  4304#  4305#  4310#  4315#  4318#  4325#  4326#  4327#  4328
                    4330#  4331   4333#  4340#  4343#  4347#  4348#  4351#  4354#  4355#  4396#  4402   4405#
                    4406#  4410#  4411   4415#  4416#  4593#  4595#  4922#  4925#  5067#  5070#  5109#  5111#
                    5114#  5130#  5136#  5137#  5288#  5290#  5295#  5296#  5298#  5299   5300#  5303#  5304#
                    5307#  5310#  5312#  5313   5315#
$$ERFL= 000000      1730#  3562#  3589#  4339#
$$FLAG= 000001      1730#  2545#  2549#  2606#  2608#  2613#  2615#  2620#  2623#  2665#  2667#  2669#  2670#
                    2671#  2676#  2678#  2680#  2681#  2682#  2687#  2688#  2689#  2691#  2693#  2696#  2697#
                    2700#  2702#  2722#  2723#  2724#  2725#  2726#  2728#  2731#  2732#  2734#  2738#  2739#
                    2743#  2745#  2746#  2747#  2750#  2753#  2755#  2756#  2757#  2758#  2759#  2761#  2764#
                    2765#  2767#  2771#  2773#  2774#  2775#  2811#  2838#  2840#  2845#  2849#  2873#  2878#
                    2881#  2883#  2924#  2926#  3003#  3008#  3010#  3013#  3019#  3022#  3028#  3033#  3036#
                    3039#  3040#  3041#  3043#  3044#  3047#  3051#  3063#  3066#  3070#  3072#  3073#  3075#
                    3076#  3081#  3084#  3095#  3099#  3100#  3146#  3159#  3161#  3163#  3166#  3167#  3168#
                    3170#  3172#  3173#  3175#  3176#  3177#  3178#  3179#  3180#  3182#  3195#  3198#  3199#
                    3200#  3201#  3203#  3204#  3205#  3206#  3210#  3213#  3217#  3218#  3219#  3221#  3225#
                    3226#  3227#  3228#  3229#  3230#  3232#  3283#  3286#  3287#  3288#  3290#  3291#  3292#
                    3294#  3296#  3297#  3301#  3305#  3306#  3307#  3345#  3348#  3349#  3350#  3353#  3354#
                    3357#  3369#  3372#  3375#  3383#  3387#  3390#  3393#  3395#  3397#  3398#  3399#  3402#
                    3405#  3406#  3409#  3416#  3417#  3429#  3432#  3434#  3443#  3484#  3497#  3507#  3513#
                    3515#  3516#  3522#  3524#  3528#  3529#  3548#  3556#  3559#  3564#  3567#  3568#  3571#
                    3573#  3574#  3578#  3582#  3583#  3587#  3590#  3591#  3593#  3594#  3595#  3630#  3640#
                    3644#  3654#  3657#  3658#  3668#  3671#  3675#  3677#  3679#  3684#  3689#  3693#  3695#
                    3700#  3701#  3703#  3704#  3705#  3706#  3707#  3708#  3771#  3776#  3783#  3784#  3794#
                    3796#  3800#  3806#  3807#  3808#  3809#  3810#  3812#  3815#  3816#  3818#  3821#  3822#
                    3828#  3830#  3846#  3850#  3852#  3853#  3854#  3855#  3858#  3861#  3864#  3865#  3867#
                    3868#  3874#  3875#  3876#  3877#  3878#  3890#  3894#  3903#  3905#  3907#  3908#  3909#
                    3912#  3913#  3914#  3927#  3929#  3935#  3937#  3938#  3939#  3953#  3956#  3958#  3994#
                    4001#  4005#  4006#  4009#  4011#  4013#  4017#  4018#  4023#  4024#  4050#  4051#  4062#
                    4063#  4127#  4134#  4136#  4152#  4153#  4155#  4163#  4167#  4169#  4231#  4235#  4237#
                    4240#  4244#  4246#  4249#  4272#  4274#  4275#  4277#  4281#  4284#  4285#  4286#  4289#
                    4293#  4298#  4299#  4303#  4304#  4305#  4310#  4315#  4318#  4320#  4325#  4326#  4330#
                    4333#  4340#  4343#  4347#  4348#  4351#  4354#  4355#  4391#  4396#  4405#  4406#  4410#
                    4415#  4416#  4454#  4589#  4593#  4595#  4675#  4715#  4922#  4925#  4932#  5067#  5070#
                    5089#  5104#  5109#  5111#  5114#  5130#  5136#  5137#  5169#  5260#  5285#  5288#  5290#
                    5294#  5295#  5296#  5298#  5300#  5303#  5304#  5307#  5310#  5312#  5315#
$$FROM= 000000      1730#
$$LOC  = 027006     1730#  2545#  2606#  2608#  2613#  2665#  2667#  2669#  2670#  2671#  2676#  2687#  2688#
                    2689#  2693#  2700#  2722#  2723#  2724#  2725#  2726#  2734#  2739#  2743#  2750#  2755#
                    2756#  2757#  2758#  2759#  2767#  2771#  2811#  2838#  2845#  2873#  2881#  2924#  3003#
                    3009#  3010#  3013#  3022#  3028#  3036#  3040#  3041#  3047#  3063#  3070#  3073#  3076#
                    3083#  3084#  3095#  3112#  3127#  3146#  3159#  3161#  3163#  3166#  3167#  3168#  3172#
                    3173#  3195#  3199#  3200#  3201#  3205#  3206#  3217#  3218#  3219#  3221#  3230#  3283#
                    3287#  3288#  3292#  3294#  3296#  3297#  3345#  3349#  3350#  3354#  3357#  3369#  3373#
                    3375#  3393#  3395#  3399#  3405#  3406#  3429#  3434#  3484#  3507#  3515#  3516#  3524#
                    3548#  3556#  3561#  3564#  3568#  3571#  3574#  3578#  3588#  3590#  3591#  3630#  3644#
                    3654#  3668#  3671#  3675#  3679#  3684#  3689#  3693#  3695#  3700#  3701#  3771#  3776#
```

```
                    3794#  3800#  3806#  3807#  3808#  3809#  3810#  3818#  3828#  3846#  3850#  3853#  3854#
                    3855#  3858#  3861#  3867#  3868#  3890#  3894#  3903#  3907#  3908#  3909#  3927#  3933#
                    3935#  3953#  3956#  3976#  3994#  4001#  4006#  4011#  4017#  4018#  4050#  4051#  4127#
                    4134#  4152#  4153#  4163#  4168#  4231#  4237#  4240#  4246#  4258#  4263#  4272#  4274#
                    4275#  4277#  4289#  4293#  4298#  4299#  4310#  4318#  4320#  4325#  4326#  4340#  4348#
                    4351#  4391#  4396#  4405#  4406#  4454#  4589#  4593#  4675#  4715#  4922#  4932#  4940#
                    5067#  5089#  5104#  5109#  5114#  5130#  5169#  5175#  5260#  5285#  5288#  5294#  5295#
                    5296#  5300#  5307#  5310#
$$LOCN= 000000      1730#
$$REG = 177777      1730#
$$RETU= 000000      1730#
$$RTN1= 000000      1730#
$$RTN2= 000000      1730#
$$SRC = 000000      1730#
$$TGSV= 000000      1730#
$$TGS1= 000000      1730#
$$TGS2= 000000      1730#
$$TO  = 000000      1730#
$$$TAG= 050000      1730#
.     = 030700      1756#  1807#  2133#  2141#  2147#  2152#  2158#  2168#  2169#  2170#  2171#  2182#  2183#
                    2184#  2228   2229#  2230#  2231#  2232#  2233#  2234#  2235#  2236#  2237#  2238#  2239#
                    2240#  2241#  2242#  2243#  2244#  2245#  2246#  2247#  2248   2249#  2261#  2292   2303
                    2307#  2308#  2323#  2324   2368#  2444#  2474#  2506#  2535   2545   2554   2606   2608
                    2613   2665   2667   2669   2670   2671   2676   2687   2688   2689   2693   2700   2722
                    2723   2724   2725   2726   2734   2739   2743   2750   2755   2756   2757   2758   2759
                    2767   2771   2811   2838   2845   2873   2881   2924   3003   3009   3010   3013   3022
                    3028   3036   3040   3041   3042   3047   3063   3070   3071   3073   3074   3076   3083
                    3084   3095   3112   3127   3146   3159   3161   3163   3166   3167   3168   3172   3173
                    3195   3199   3200   3201   3205   3206   3217   3218   3219   3221   3230   3283   3287
                    3288   3292   3294   3296   3297   3345   3349   3350   3354   3357   3369   3373   3375
                    3393   3395   3399   3405   3406   3429   3434   3484   3507   3515   3516   3524   3548
                    3556   3561   3564   3568   3571   3574   3578   3588   3590   3591   3630   3644   3654
                    3668   3671   3675   3679   3684   3689   3693   3695   3700   3701   3771   3776   3794
                    3800   3806   3807   3808   3809   3810   3818   3828   3846   3850   3853   3854   3855
                    3858   3861   3867   3868   3890   3894   3903   3907   3908   3909   3927   3933   3935
                    3953   3956   3976   3994   4001   4006   4011   4017   4018   4050   4051   4127   4134
                    4141   4152   4153   4163   4168   4231   4233   4237   4240   4246   4258   4263   4272
                    4274   4275   4277   4289   4293   4298   4299   4310   4312   4318   4320   4324   4325
                    4326   4335   4340   4348   4351   4362   4376#  4391   4396   4405   4406   4425#  4454
                    4460   4494   4536   4589   4593   4675   4706   4715   4922   4932   4940   4961   5067
                    5089   5104   5109   5114   5130   5147   5169   5175   5189   5226   5260   5279   5285
                    5288   5294   5295   5296   5300   5307   5310   5326   5413   5455   5461   5479   5480
                    5494   5520#  5522   5538#  5552   5556
```

```
        3227    3228    3229    3232    3286    3290    3291    3301    3305    3306    3307    3348    3353    3372    3383
        3387    3390    3397    3398    3402    3409    3416    3417    3432    3443    3497    3513    3522    3528    3529
        3559    3567    3573    3582    3583    3587    3593    3594    3595    3640    3657    3658    3677    3703    3704
        3705    3706    3707    3708    3783    3784    3796    3812    3815    3816    3821    3822    3830    3852    3864
        3865    3874    3875    3876    3877    3878    3905    3912    3913    3914    3929    3937    3938    3939    3958
        4005    4009    4013    4023    4024    4062    4063    4136    4155    4167    4169    4235    4244    4249    4281
        4284    4285    4286    4303    4304    4305    4315    4330    4333    4343    4347    4354    4355    4410    4415
        4416    4595    4925    5070    5111    5136    5137    5290    5298    5303    5304    5312    5315
ENDINC  1730#   4337    4338
ENDINI  4378
ENDLOO  1730#
ENDMUD  1888    4087    4553    5367    5543
ENDMSG  2538    2572
ENDPRO  4211
ENDPTA  5555
ENDRPT  4201
ENDRTN  1730#
ENDSEL  1730#
ENDSET  5556
ENDSFT  5532
ENDSRV  2584    2588    2592    2596
ENDSUB  4598    4606    4613    4620    4627    4634    4641    4648    4655    4662    4672    4701
ENDSW   1887
ENDTST  4908    5051    5150    5193    5230    5366
EQUALS  1920
ERRDF   3005    3030    3049    3087    3271    3322    3461    3476    3508    3518    4002
ERRHRD  3223    3255    3303    3526    3891    3895    3921
ERRSF   4232
ERRSOF  3351    3400    3430
EXIF    1730#
EXIFB   1730#
EXIT    2535    2554    4141    4362    4460    4494    4536    4706    4961    5147    5189    5226    5326    5413    5522
GMANIL  4059
GPHARD  4292    4529
GPRMA   5410
GPRMD   5411    5462    5463    5464    5465    5466    5467    5468    5469    5470    5471    5472    5473    5474    5475
        5476    5477    5478    5481    5482    5483    5484    5485    5486    5487    5488    5489    5490    5491    5492
GPRML   4059#   5450    5451    5452    5453    5454    5456    5457    5458    5459    5460
HEADER  1775
IF      1730#   2545    2606    2608    2670    2676    2688    2689    2693    2700    2723    2724    2725    2734    2739
        2756    2757    2758    2767    2838    2845    2873    2924    3003    3010    3040    3041    3047    3063    3070
        3073    3084    3161    3163    3166    3167    3168    3172    3173    3195    3199    3200    3205    3206    3217
        3218    3283    3287    3288    3292    3345    3349    3369    3375    3393    3395    3399    3406    3429    3507
        3515    3516    3524    3548    3556    3564    3574    3578    3644    3654    3671    3675    3776    3807    3808
        3809    3818    3828    3854    3855    3858    3861    3867    3868    3890    3894    3907    3909    3927    3935
        3956    4001    4011    4017    4152    4153    4163    4231    4298    4299    4310    4325    4326    4340    4351
        4405    4406    4593    5109    5114    5130    5288    5295    5296    5300    5307    5310
IFB     1730#   2613    2669    2671    2726    2743    2750    2759    2771    2881    3028    3036    3076    3095    3159
        3201    3219    3221    3230    3294    3296    3297    3350    3354    3357    3405    3434    3484    3568    3571
        3590    3591    3630    3668    3693    3695    3700    3701    3771    3794    3810    3850    3853    3903    3908
        4006    4018    4050    4051    4134    4237    4246    4272    4275    4318    4348    4396    4922    5067
IFCOND  1730#   4274    4277    4293
IF.ERR  1730#
IF.NO.  1730#
INCR    1730#   4322    4334
INCRU   1730#
```

	5481#	5482#	5483#	5484#	5485#	5486#	5487#	5488#	5489#	5490#	5491#	5492#			
MSENDE	1829#	1887#	1888#	2538#	2572#	2584#	2588#	2592#	2596#	4087#	4201#	4378#	4420#	4474#	4508#
	4551#	4553#	4598#	4606#	4613#	4620#	4627#	4634#	4641#	4648#	4655#	4662#	4672#	4701#	4908#
	5051#	5150#	5193#	5230#	5366#	5367#	5427#	5532#	5543#						
MSERRI	3005#	3030#	3049#	3087#	3223#	3255#	3271#	3303#	3322#	3351#	3400#	3430#	3461#	3476#	3508#
	3518#	3526#	3891#	3895#	3921#	4002#	4232#								
MSEXCP	5410#	5411#	5462#	5463#	5464#	5465#	5466#	5467#	5468#	5469#	5470#	5471#	5472#	5473#	5474#
	5475#	5476#	5477#	5478#	5481#	5482#	5483#	5484#	5485#	5486#	5487#	5488#	5489#	5490#	5491#
	5492#														
MSEXIT	2535#	2554#	4141#	4362#	4460#	4494#	4536#	4706#	4961#	5147#	5189#	5226#	5326#	5413#	5522#
MSEXSE	2535#	2554#	4141#	4362#	4460#	4494#	4536#	4706#	4961#	5147#	5189#	5226#	5326#	5413#	5522#
MSEXTJ	2535#	2554#	4141#	4362#	4460#	4494#	4536#	4706#	4961#	5147#	5189#	5226#	5326#	5413#	5522#
MSGEN	1775#	1791#	1806#	1807#	1818#	1829#	1838#	1887#	2518#	2538#	2540#	2572#	2582#	2584#	2586#
	2588#	2590#	2592#	2594#	2596#	4059#	4118#	4201#	4207#	4220#	4378#	4388#	4420#	4444#	4474#
	4484#	4508#	4519#	4551#	4578#	4583#	4598#	4600#	4606#	4608#	4613#	4615#	4620#	4622#	4627#
	4629#	4634#	4636#	4641#	4643#	4648#	4650#	4655#	4657#	4662#	4664#	4672#	4680#	4701#	4908#
	4915#	5051#	5066#	5150#	5159#	5193#	5203#	5230#	5239#	5366#	5403#	5427#	5441#	5532#	5542#
	5552#	5555#													
MSGENB	4059#														
MSGETS	1829#	1887#	1888#	2538#	2572#	2584#	2588#	2592#	2596#	4087#	4201#	4211#	4378#	4420#	4474#
	4508#	4551#	4553#	4598#	4606#	4613#	4620#	4627#	4634#	4641#	4648#	4655#	4662#	4672#	4701#
	4908#	5051#	5150#	5193#	5230#	5366#	5367#	5413#	5427#	5455#	5461#	5479#	5480#	5494#	5522#
	5532#	5543#													
MSGETT	2535#	2554#	4141#	4362#	4460#	4494#	4536#	4706#	4961#	5147#	5189#	5226#	5326#	5413#	5455#
	5461#	5479#	5480#	5494#	5522#										
MSGNGB	1758#	1775#	1791#	1806#	1807#	1818#	1838#	1913#	2518#	2540#	2582#	2586#	2590#	2594#	4111#
	4118#	4207#	4220#	4388#	4444#	4484#	4519#	4576#	5392#	5403#	5441#	5542#			
MSGNIN	1775#	1791#	1806#	1807#	1818#	1838#	2524#	2525#	2534#	2535#	2538#	2541#	2542#	2548#	2550#
	2551#	2552#	2553#	2554#	2572#	2584#	2588#	2592#	2596#	2752#	3005#	3030#	3042#	3049#	3069#
	3071#	3074#	3087#	3125#	3223#	3255#	3271#	3303#	3322#	3351#	3376#	3400#	3430#	3461#	3476#
	3485#	3496#	3508#	3518#	3526#	3572#	3592#	3702#	3747#	3753#	3891#	3892#	3895#	3896#	3921#
	3928#	3936#	3957#	4002#	4010#	4016#	4037#	4038#	4052#	4053#	4059#	4128#	4129#	4130#	4131#
	4132#	4133#	4137#	4141#	4148#	4151#	4156#	4160#	4164#	4170#	4201#	4232#	4233#	4250#	4251#
	4265#	4266#	4270#	4273#	4276#	4292#	4297#	4311#	4312#	4313#	4314#	4317#	4324#	4329#	4332#
	4335#	4336#	4349#	4352#	4353#	4362#	4378#	4393#	4395#	4398#	4401#	4408#	4413#	4420#	4456#
	4460#	4474#	4493#	4494#	4508#	4529#	4533#	4536#	4551#	4583#	4594#	4598#	4600#	4606#	4608#
	4613#	4615#	4620#	4622#	4627#	4629#	4634#	4636#	4641#	4643#	4648#	4650#	4655#	4657#	4662#
	4664#	4672#	4680#	4701#	4706#	4908#	4961#	5051#	5068#	5147#	5150#	5189#	5193#	5226#	5230#
	5279#	5326#	5366#	5403#	5410#	5411#	5413#	5427#	5441#	5450#	5451#	5452#	5453#	5454#	5455#
	5456#	545?#	5458#	5459#	5460#	5461#	5462#	5463#	5464#	5465#	5466#	5467#	5468#	5469#	5470#
	5471#	5472#	5473#	5474#	5475#	5476#	5477#	5478#	5479#	5480#	5481#	5482#	5483#	5484#	5485#
	5486#	5487#	5488#	5489#	5490#	5491#	5492#	5494#	5522#	5532#	5542#	5552#			
MSGNLS	4059#														
MSGNSU	4583#	4600#	4608#	4615#	4622#	4629#	4636#	4643#	4650#	4657#	4664#	4680#			
MSGNTA	1829#	1887#	2538#	2572#	2584#	2588#	2592#	2596#	4201#	4378#	4420#	4474#	4508#	4551#	4598#
	4606#	4613#	4620#	4627#	4634#	4641#	4648#	4655#	4662#	4672#	4701#	4908#	5051#	5150#	5193#
	5230#	5366#	5427#	5532#	5552#	5555#									
MSGNTE	4578#	4915#	5066#	5159#	5203#	5239#									
MSHAPT	1775#														
MSHNAP	1775#														
MSINCR	1758#	1818#	1838#	1913#	2518#	2524#	2525#	2534#	2538#	2540#	2541#	2542#	2548#	2550#	2551#
	2552#	2553#	2572#	2582#	2586#	2590#	2594#	2752#	3005#	3030#	3049#	3069#	3087#	3125#	3223#
	3255#	3271#	3303#	3322#	3351#	3376#	3400#	3430#	3461#	3476#	3485#	3496#	3508#	3518#	3526#
	3572#	3592#	3702#	3747#	3753#	3891#	3892#	3895#	3896#	3921#	3928#	3936#	3957#	4002#	4010#
	4016#	4037#	4038#	4052#	4059#	4111#	4118#	4128#	4129#	4130#	4131#	4132#	4133#	4137#	4148#
	4151#	4156#	4160#	4164#	4170#	4201#	4207#	4220#	4232#	4250#	4265#	4270#	4273#	4276#	4292#

```
                4297#  4311#  4313#  4314#  4317#  4329#  4332#  4336#  4349#  4352#  4353#  4362#  4378#  4388#  4393#
                4395#  4398#  4401#  4408#  4413#  4420#  4444#  4456#  4460#  4474#  4484#  4493#  4508#  4519#  4529#
                4533#  4551#  4576#  4578#  4583#  4594#  4598#  4600#  4606#  4608#  4613#  4615#  4620#  4622#  4627#
                4629#  4634#  4636#  4641#  4643#  4648#  4650#  4655#  4657#  4662#  4664#  4672#  4680#  4701#  4706#
                4908#  4915#  4961#  5051#  5066#  5068#  5147#  5150#  5159#  5189#  5193#  5203#  5226#  5230#  5239#
                5326#  5366#  5392#  5403#  5441#  5551#  5552#
M$LDRO          4016#  4250#  4265#  4273#  4276#  4292#  4317#  4395#  4401#  4456#  4529#
M$MCHI          1728#
M$MCLO          1728#
M$POP           1829#  1887#  1888#  2538#  2572#  2584#  2588#  2592#  2596#  4087#  4201#  4211#  4378#  4420#  4474#
                4508#  4551#  4553#  4598#  4606#  4613#  4620#  4627#  4634#  4641#  4648#  4655#  4662#  4672#  4701#
                4908#  5051#  5150#  5193#  5230#  5366#  5367#  5427#  5532#  5543#
M$PRIN          2524#  2525#  2534#  2541#  2542#  2548#  2550#  2551#  2552#  2553#  3376#  3485#  3496#  3572#  3592#
                3702#  3747#  3753#  3892#  3896#  3928#  3936#  4037#  4038#  4128#  4129#  4130#  4131#  4132#  4133#
                4137#  4148#  4151#  4156#  4160#  4164#  4170#  4311#  4329#  4332#  4352#  4398#  4408#  4413#  4493#
                4594#  5068#
M$PUSH          1758#  1818#  1838#  1913#  2518#  2540#  2582#  2586#  2590#  2594#  4111#  4118#  4207#  4220#  4388#
                4444#  4484#  4519#  4576#  4578#  4583#  4600#  4608#  4615#  4622#  4629#  4636#  4643#  4650#  4657#
                4664#  4680#  4915#  5066#  5159#  5203#  5239#  5392#  5403#  5441#
M$PUT           2524#  2525#  2534#  2541#  2542#  2548#  2550#  2551#  2552#  2553#  3376#  3485#  3496#  3572#  3592#
                3702#  3747#  3753#  3892#  3896#  3928#  3936#  4037#  4038#  4128#  4129#  4130#  4131#  4132#  4133#
                4137#  4148#  4151#  4156#  4160#  4164#  4170#  4297#  4311#  4329#  4332#  4352#  4393#  4398#  4408#
                4413#  4493#  4533#  4594#  5068#
M$PUT1          2524#  2525#  2534#  2541#  2542#  2548#  2550#  2551#  2552#  2553#  3376#  3485#  3496#  3572#  3592#
                3702#  3747#  3753#  3892#  3896#  3928#  3936#  4037#  4038#  4128#  4129#  4130#  4131#  4132#  4133#
                4137#  4148#  4151#  4156#  4160#  4164#  4170#  4297#  4311#  4329#  4332#  4352#  4393#  4398#  4408#
                4413#  4493#  4533#  4594#  5068#
M$RADI          4059#  5410#  5411#  5450#  5451#  5452#  5453#  5454#  5456#  5457#  5458#  5459#  5460#  5462#  5463#
                5464#  5465#  5466#  5467#  5468#  5469#  5470#  5471#  5472#  5473#  5474#  5475#  5476#  5477#  5478#
                5481#  5482#  5483#  5484#  5485#  5486#  5487#  5488#  5489#  5490#  5491#  5492#
M$RNRO          4270#  4292#  4349#  4529#
M$SETS          1758#  1818#  1838#  1913#  2518#  2540#  2582#  2586#  2590#  2594#  4111#  4118#  4207#  4220#  4388#
                4444#  4484#  4519#  4576#  4578#  4583#  4600#  4608#  4615#  4622#  4629#  4636#  4643#  4650#  4657#
                4664#  4680#  4915#  5066#  5159#  5203#  5239#  5392#  5403#  5441#
M$SVC           2524#  2525#  2534#  2535#  2538#  2541#  2542#  2548#  2550#  2551#  2552#  2553#  2554#  2572#  2752#
                3005   3030   3049   3069#  3087   3125#  3223   3255   3271   3303   3322   3351   3376#  3400   3430
                3461   3476   3485#  3496#  3508   3518   3526   3572#  3592#  3702#  3747#  3753#  3891   3892#  3895
                3896#  3921   3928#  3936#  3957#  4002   4010#  4016#  4037#  4038#  4052#  4059#  4128#  4129#  4130#
                4131#  4132#  4133#  4137#  4141#  4148#  4151#  4156#  4160#  4164#  4170#  4201#  4232   4250#  4265#
                4270#  4273#  4276#  4292#  4297#  4311#  4313#  4314#  4317#  4329#  4332#  4336#  4349#  4352#  4353#
                4362#  4378#  4393#  4395#  4398#  4401#  4408#  4413#  4420#  4456#  4460#  4474#  4493#  4494#  4508#
                4529#  4533#  4536#  4551#  4583#  4594#  4598#  4600#  4606#  4608#  4613#  4615#  4620#  4622#  4627#
                4629#  4634#  4636#  4641#  4643#  4648#  4650#  4655#  4657#  4662#  4664#  4672#  4680#  4701#  4706#
                4908#  4961#  5051#  5068#  5147#  5150#  5189#  5193#  5226#  5230#  5326#  5366#  5413#  5522#
M$TLAB          2524#  2525#  2534#  2538#  2541#  2542#  2548#  2550#  2551#  2552#  2553#  2572#  2752#  3005#  3030#
                3049#  3069#  3087#  3125#  3223#  3255#  3271#  3303#  3322#  3351#  3376#  3400#  3430#  3461#  3476#
                3485#  3496#  3508#  3518#  3526#  3572#  3592#  3702#  3747#  3753#  3891#  3892#  3895#  3896#  3921#
                3928#  3936#  3957#  4002#  4010#  4016#  4037#  4038#  4052#  4059#  4128#  4129#  4130#  4131#  4132#
                4133#  4137#  4148#  4151#  4156#  4160#  4164#  4170#  4201#  4232#  4250#  4265#  4270#  4273#  4276#
                4292#  4297#  4311#  4313#  4314#  4317#  4329#  4332#  4336#  4349#  4352#  4353#  4362#  4378#  4393#
                4395#  4398#  4401#  4408#  4413#  4420#  4456#  4460#  4474#  4493#  4508#  4529#  4533#  4551#  4583#
                4594#  4598#  4600#  4606#  4608#  4613#  4615#  4620#  4622#  4627#  4629#  4634#  4636#  4641#  4643#
                4648#  4650#  4655#  4657#  4662#  4664#  4672#  4680#  4701#  4706#  4908#  4961#  5051#  5068#  5147#
                5150#  5189#  5193#  5226#  5230#  5326#  5366#
M$TSTL          2524#  2525#  2534#  2538#  2541#  2542#  2548#  2550#  2551#  2552#  2553#  2572#  2752#  3005#  3030#
                3049#  3069#  3087#  3125#  3223#  3255#  3271#  3303#  3322#  3351#  3376#  3400#  3430#  3461#  3476#
```

```
            3485#  3496#  3508#  3518#  3526#  3572#  3592#  3702#  3747#  3753#  3891#  3892#  3895#  3896#  3921#
            3928#  3936#  3957#  4002#  4010#  4016#  4037#  4038#  4052#  4059#  4128#  4129#  4130#  4131#  4132#
            4133#  4137#  4148#  4151#  4156#  4160#  4164#  4170#  4201#  4232#  4250#  4265#  4270#  4273#  4276#
            4292#  4297#  4311#  4313#  4314#  4317#  4329#  4332#  4336#  4349#  4352#  4353#  4362#  4378#  4393#
            4395#  4398#  4401#  4408#  4413#  4420#  4456#  4460#  4474#  4493#  4508#  4529#  4533#  4551#  4583#
            4594#  4598#  4600#  4606#  4608#  4613#  4615#  4620#  4622#  4627#  4629#  4634#  4636#  4641#  4643#
            4648#  4650#  4655#  4657#  4662#  4664#  4672#  4680#  4701#  4706#  4908#  4961#  5051#  5068#  5147#
            5150#  5189#  5193#  5226#  5230#  5326#  5366#
MSWORD      1775#  1791#  2535#  2554#  3005#  3030#  3049#  3087#  3223#  3255#  3271#  3303#  3322#  3351#  3400#
            3430#  3461#  3476#  3508#  3518#  3526#  3891#  3895#  3921#  4002#  4059#  4141#  4232#  4362#  4460#
            4494#  4536#  4706#  4961#  5147#  5189#  5226#  5326#  5410#  5411#  5413#  5450#  5451#  5452#  5453#
            5454#  5455#  5456#  5457#  5458#  5459#  5460#  5461#  5462#  5463#  5464#  5465#  5466#  5467#  5468#
            5469#  5470#  5471#  5472#  5473#  5474#  5475#  5476#  5477#  5478#  5479#  5480#  5481#  5482#  5483#
            5484#  5485#  5486#  5487#  5488#  5489#  5490#  5491#  5492#  5494#  5522#  5552#
MSXFER      5413#  5455#  5461#  5479#  5480#  5494#  5522#
POINTE      1765
POP         1730#  3411   3438   3511   3521   3527
PRINTB      2524   2525   2534   2541   2542   2548   3376   3572   3592   3702   3892   3896   3936
PRINTF      4311   4329   4332   4352   4398   4408   4413   4493   4594   5068
PRINTS      4128   4129   4130   4131   4132   4133   4137   4148   4151   4156   4160   4164   4170
PRINTX      2550   2551   2552   2553   3485   3496   3747   3753   3928   4037   4038
PUSH        1730#
READEF      4250   4265   4273   4276
REPEAT      1730#  3001   3068   3110   3124   3364   3550   3552   3906   3974   4159   4255   4260   4931   5168
RETURN      1730#
RFLAGS      4270
ROUTIN      1730#
SAVR14      1730#
SELECT      1730#
SETPRI      4317
SETVEC      4297   4393   4533
STRUCT      1729#  1730
SVC         1727#  1728
UNTIL       1730#  3009   3083   3112   3127   3588   3933   3976   4168   4258   4263   4940   5175
UNTILB      1730#  3373   3561
WHILE       1730#  2665   2667   2687   2722   2755   2811   3013   3022   3146   3679   3684   3689   3800   3806
            3846   3953   3994   4127   4240   4289   4320   4391   4454   4589   4675   4715   4932   5089   5104
            5169   5260   5285   5294
WHILEB      1730#
XFER        2535#  2554#  4141#  4362#  4460#  4494#  4536#  4706#  4961#  5147#  5189#  5226#  5326#  5413#  5479
            5480   5494   5522#
XFERF       5461
XFERT       5455
$ADDON      1730#  2545   2606   2608   2613   2618   2621   2665   2667   2669   2670   2671   2676   2687   2688
            2689   2692   2693   2699   2700   2705   2708   2722   2723   2724   2725   2726   2729   2733   2734
            2739   2742   2743   2749   2750   2755   2756   2757   2758   2759   2762   2766   2767   2770   2771
            2777   2811   2813   2838   2845   2847   2873   2876   2881   2924   3001   3003   3010   3013   3016
            3022   3025   3028   3036   3040   3041   3047   3063   3065   3068   3070   3073   3076   3078   3083
            3084   3091   3095   3110   3124   3127   3146   3149   3159   3161   3163   3165   3166   3167   3168
            3171   3172   3173   3195   3197   3199   3200   3201   3205   3206   3208   3211   3216   3217   3218
            3219   3221   3230   3283   3287   3288   3292   3294   3296   3297   3299   3345   3347   3349   3350
            3354   3357   3364   3369   3371   3373   3375   3388   3391   3393   3395   3399   3405   3406   3408
            3414   3429   3434   3441   3484   3507   3515   3516   3523   3524   3548   3549   3550   3551   3552
            3556   3558   3561   3564   3566   3568   3570   3571   3574   3578   3590   3591   3630   3644   3654
            3668   3671   3675   3679   3682   3684   3687   3689   3692   3693   3695   3698   3700   3701   3771
            3776   3794   3800   3806   3807   3808   3809   3810   3813   3817   3818   3826   3828   3834   3846
```

```
                3848    3850    3853    3854    3855    3857    3858    3860    3861    3867    3868    3871    3890    3893    3894
                3897    3903    3906    3907    3908    3909    3927    3935    3953    3955    3956    3974    3994    3997    4001
                4003    4006    4011    4017    4018    4050    4051    4060    4127    4134    4139    4152    4153    4159    4163
                4231    4237    4240    4243    4246    4255    4260    4272    4274    4275    4277    4279    4282    4289    4293
                4298    4299    4301    4308    4310    4318    4320    4321    4322    4325    4326    4328    4331    4334    4340
                4346    4348    4351    4391    4396    4402    4405    4406    4411    4418    4454    4458    4589    4593    4597
                4675    4677    4715    4717    4922    4931    4932    4936    5067    5089    5104    5109    5113    5114    5130
                5143    5168    5169    5171    5260    5285    5288    5294    5295    5296    5299    5300    5306    5307    5310
                5313    5318    5322
$AND            1730#   2734    3040    3161    3283    3288    3349    3350    3399    3701    4318
$BRANC          1730#   2545    2606    2608    2613    2618    2621    2665    2667    2669    2670    2671    2676    2687    2688
                2689    2692    2693    2699    2700    2705    2708    2722    2723    2724    2725    2726    2729    2733    2734
                2739    2742    2743    2749    2750    2755    2756    2757    2758    2759    2762    2766    2767    2770    2771
                2777    2811    2813    2838    2845    2847    2873    2876    2881    2924    3003    3009    3010    3013    3016
                3022    3025    3028    3036    3040    3041    3047    3063    3065    3070    3073    3076    3078    3083    3084
                3091    3095    3112    3127    3146    3149    3159    3161    3163    3165    3166    3167    3168    3171    3172
                3173    3195    3197    3199    3200    3201    3205    3206    3208    3211    3216    3217    3218    3219    3221
                3230    3283    3287    3288    3292    3294    3296    3297    3299    3345    3347    3349    3350    3354    3357
                3369    3371    3373    3375    3388    3391    3393    3395    3399    3405    3406    3408    3414    3429    3434
                3441    3484    3507    3515    3516    3523    3524    3548    3556    3558    3561    3564    3566    3568    3569
                3571    3574    3578    3588    3590    3591    3630    3644    3654    3668    3671    3675    3679    3682    3684
                3687    3689    3692    3693    3695    3698    3700    3701    3771    3776    3794    3800    3806    3807    3808
                3809    3810    3813    3817    3818    3826    3828    3834    3846    3848    3850    3853    3854    3855    3857
                3858    3860    3861    3867    3868    3871    3890    3893    3894    3897    3903    3907    3908    3909    3927
                3933    3935    3953    3955    3956    3976    3994    3997    4001    4003    4006    4011    4017    4018    4050
                4051    4060    4127    4134    4139    4152    4153    4163    4168    4231    4237    4240    4243    4246    4258
                4263    4272    4274    4275    4277    4279    4282    4289    4293    4298    4299    4301    4308    4310    4318
                4320    4322    4325    4326    4327    4331    4334    4337    4338    4340    4346    4348    4351    4391    4396
                4402    4405    4406    4411    4418    4454    4458    4589    4593    4597    4675    4677    4715    4717    4922
                4932    4936    4940    5067    5089    5104    5109    5113    5114    5130    5143    5169    5171    5175    5260
                5285    5288    5294    5295    5296    5299    5300    5306    5307    5310    5313    5318    5322
$BRCOD          1730#   2693    2739    2767    3073    3083    3127    3205    3219    3294    3373    3561    3818    4322    4334
                5300
$CALL           1730#
$CHECK          1730#   2545    2606    2608    2613    2665    2667    2669    2670    2671    2676    2687    2688    2689    2693
                2700    2722    2723    2724    2725    2726    2734    2739    2743    2750    2755    2756    2757    2758    2759
                2767    2771    2811    2838    2845    2873    2881    2924    3003    3010    3013    3022    3028    3036    3040
                3041    3047    3063    3070    3073    3076    3084    3095    3146    3159    3161    3163    3166    3167    3168
                3172    3173    3195    3199    3200    3201    3205    3206    3217    3218    3219    3221    3230    3283    3287
                3288    3292    3294    3296    3297    3345    3349    3350    3354    3357    3369    3375    3393    3395    3399
                3405    3406    3429    3434    3484    3507    3515    3516    3524    3548    3556    3564    3568    3571    3574
                3578    3590    3591    3630    3644    3654    3668    3671    3675    3679    3684    3689    3693    3695    3700
                3701    3771    3776    3794    3800    3806    3807    3808    3809    3810    3818    3828    3846    3850    3853
                3854    3855    3858    3861    3867    3868    3890    3894    3903    3907    3908    3909    3927    3935    3953
                3956    3994    4001    4006    4011    4017    4018    4050    4051    4127    4134    4152    4153    4163    4231
                4237    4240    4246    4272    4275    4289    4298    4299    4310    4318    4320    4325    4326    4340    4348
                4351    4391    4396    4405    4406    4454    4589    4593    4675    4715    4922    4932    5067    5089    5104
                5109    5114    5130    5169    5260    5285    5288    5294    5295    5296    5300    5307    5310
$CKOP1          1730#   2526    2527    2528    2530    2531    2532    2533    2547    2611    2619    2622    2635    2651    2652
                2653    2664    2674    2677    2679    2685    2690    2695    2704    2709    2736    2737    2751    2788    2793
                2796    2797    2798    2810    2814    2827    2839    2846    2848    2854    2856    2864    2865    2872    2874
                2877    2879    2880    2885    2919    2922    2925    3000    3011    3017    3018    3020    3021    3023    3029
                3034    3038    3067    3077    3079    3089    3097    3109    3111    3123    3141    3144    3145    3147    3150
                3164    3181    3207    3209    3212    3231    3359    3360    3361    3366    3367    3381    3382    3384    3386
                3437    3459    3554    3557    3575    3577    3579    3581    3585    3586    3620    3621    3623    3624    3625
                3631    3632    3633    3634    3639    3645    3646    3647    3648    3649    3650    3651    3670    3769    3770
```

	3772	3773	3774	3777	3778	3779	3780	3781	3797	3799	3832	3847	3859	3862	3863
	3870	3872	3873	3901	3902	3951	3952	3959	3972	3977	3989	3992	3993	3995	3998
	4014	4026	4054	4056	4057	4058	4061	4073	4074	4075	4076	4078	4079	4125	4140
	4149	4154	4158	4239	4241	4247	4248	4253	4254	4256	4271	4278	4288	4294	4296
	4302	4306	4322	4323	4334	4357	4358	4360	4392	4394	4399	4403	4492	4528	4530
	4532	4534	4580	4581	4585	4590	4602	4605	4610	4617	4624	4631	4638	4645	4652
	4659	4666	4669	4671	4682	4684	4686	4687	4689	4691	4693	4695	4696	4698	4703
	4714	4716	4718	4917	4918	4920	4924	4926	4927	4928	4929	4930	4937	4939	4950
	4951	4952	4954	4955	4956	4958	4994	4995	4998	4999	5000	5001	5002	5003	5012
	5026	5041	5045	5046	5069	5072	5073	5074	5077	5078	5091	5092	5093	5094	5095
	5096	5097	5098	5099	5101	5116	5118	5119	5120	5121	5122	5123	5124	5126	5127
	5129	5133	5138	5139	5140	5145	5161	5162	5164	5167	5172	5174	5184	5186	5205
	5206	5209	5210	5211	5212	5213	5214	5217	5218	5219	5220	5221	5222	5224	5241
	5242	5243	5245	5246	5254	5255	5256	5270	5278	5292	5297	5302	5314	5317	5338
	5339	5340	5341	5355	5356	5357	5359	5360							
$CKOP2	1730#	2543	2544	2583	2587	2591	2595	2607	2609	2610	2612	2614	2616	2617	2672
	2673	2675	2703	2812	2815	2882	2895	2896	2897	2898	2899	2920	2923	2933	2949
	2973	2981	3002	3024	3037	3080	3082	3085	3126	3142	3143	3148	3160	3162	3169
	3174	3214	3222	3254	3298	3300	3302	3355	3356	3362	3363	3365	3378	3385	3389
	3394	3396	3403	3404	3410	3415	3433	3435	3436	3442	3514	3517	3525	3560	3563
	3576	3580	3584	3622	3669	3672	3673	3674	3676	3678	3680	3681	3683	3685	3686
	3688	3690	3691	3694	3696	3697	3699	3793	3795	3829	3831	3845	3889	3898	3922
	3934	3954	3975	3990	3991	3996	4007	4012	4015	4022	4025	4150	4157	4161	4162
	4165	4166	4242	4252	4257	4259	4261	4262	4267	4268	4280	4283	4290	4291	4295
	4300	4322	4334	4350	4397	4400	4407	4412	4431	4491	4527	4531	4591	4592	4674
	4678	4919	4933	4934	4941	4996	4997	5043	5044	5110	5112	5163	5176	5216	5263
	5267	5269	5273	5277	5282	5308	5309	5316	5337	5358					
$CKR6	1730#														
$CMND	1730#	2545	2606	2608	2613	2665	2667	2669	2670	2671	2676	2687	2688	2689	2693
	2700	2722	2723	2724	2725	2726	2734	2739	2743	2750	2755	2756	2757	2758	2759
	2767	2771	2811	2838	2845	2873	2881	2924	3003	3010	3013	3022	3028	3036	3040
	3041	3047	3063	3070	3073	3076	3084	3095	3146	3159	3161	3163	3166	3167	3168
	3172	3173	3195	3199	3200	3201	3205	3206	3217	3218	3219	3221	3230	3283	3287
	3288	3292	3294	3296	3297	3345	3349	3350	3354	3357	3369	3375	3393	3395	3399
	3405	3406	3429	3434	3484	3507	3515	3516	3524	3548	3556	3564	3568	3571	3574
	3578	3590	3591	3630	3644	3654	3668	3671	3675	3679	3684	3689	3693	3695	3700
	3701	3771	3776	3794	3800	3806	3807	3808	3809	3810	3818	3828	3846	3850	3853
	3854	3855	3858	3861	3867	3868	3890	3894	3903	3907	3908	3909	3927	3935	3953
	3956	3994	4001	4006	4011	4017	4018	4050	4051	4127	4134	4152	4153	4163	4231
	4237	4240	4246	4272	4275	4289	4298	4299	4310	4318	4320	4325	4326	4340	4348
	4351	4391	4396	4405	4406	4454	4589	4593	4675	4715	4922	4932	5067	5089	5104
	5109	5114	5130	5169	5260	5285	5288	5294	5295	5296	5300	5307	5310		
$COMPA	1730#	2545	2606	2608	2613	2665	2667	2669	2670	2671	2676	2687	2688	2689	2693
	2700	2722	2723	2724	2725	2726	2734	2739	2743	2750	2755	2756	2757	2758	2759
	2767	2771	2811	2838	2845	2873	2881	2924	3003	3010	3013	3022	3028	3036	3040
	3041	3047	3063	3070	3073	3076	3084	3095	3146	3159	3161	3163	3166	3167	3168
	3172	3173	3195	3199	3200	3201	3205	3206	3217	3218	3219	3221	3230	3283	3287
	3288	3292	3294	3296	3297	3345	3349	3350	3354	3357	3369	3375	3393	3395	3399
	3405	3406	3429	3434	3484	3507	3515	3516	3524	3548	3556	3564	3568	3571	3574
	3578	3590	3591	3630	3644	3654	3668	3671	3675	3679	3684	3689	3693	3695	3700
	3701	3771	3776	3794	3800	3806	3807	3808	3809	3810	3818	3828	3846	3850	3853
	3854	3855	3858	3861	3867	3868	3890	3894	3903	3907	3908	3909	3927	3935	3953
	3956	3994	4001	4006	4011	4017	4018	4050	4051	4127	4134	4152	4153	4163	4231
	4237	4240	4246	4272	4274	4275	4277	4289	4293	4298	4299	4310	4318	4320	4322
	4325	4326	4334	4340	4348	4351	4391	4396	4405	4406	4454	4589	4593	4675	4715
	4922	4932	5067	5089	5104	5109	5114	5130	5169	5260	5285	5288	5294	5295	5296

```
        5300    5307    5310
$COUNT  1730#
$DO     1730#   2665    2667    2687    2722    2755    2811    3013    3022    3146    3679    3684    3689    3800    3806
        3846    3953    3994    4127    4240    4289    4320    4391    4454    4589    4675    4715    4932    5089    5104
        5169    5260    5285    5294
$ELSE   1730#
$ERRMS  1730#
$EXIFA  1730#
$EXIFO  1730#
$EXIF2  1730#
$EXIF3  1730#
$GENBR  1730#   2545    2606    2608    2613    2618    2621    2665    2667    2669    2670    2671    2676    2687    2688
        2689    2692    2693    2699    2700    2705    2708    2722    2723    2724    2725    2726    2729    2733    2734
        2739    2742    2743    2749    2750    2755    2756    2757    2758    2759    2762    2766    2767    2770    2771
        2777    2811    2813    2838    2845    2847    2873    2876    2881    2924    3003    3009    3010    3013    3016
        3022    3025    3028    3036    3040    3041    3047    3063    3065    3070    3073    3076    3078    3083    3084
        3091    3095    3112    3127    3146    3149    3159    3161    3163    3165    3166    3167    3168    3171    3172
        3173    3195    3197    3199    3200    3201    3205    3206    3208    3211    3216    3217    3218    3219    3221
        3230    3283    3287    3288    3292    3294    3296    3297    3299    3345    3347    3349    3350    3354    3357
        3369    3371    3373    3375    3388    3391    3393    3395    3399    3405    3406    3408    3414    3429    3434
        3441    3484    3507    3515    3516    3523    3524    3548    3556    3558    3561    3564    3566    3568    3569
        3571    3574    3578    3588    3590    3591    3630    3644    3654    3668    3671    3675    3679    3682    3684
        3687    3689    3692    3693    3695    3698    3700    3701    3771    3776    3794    3800    3806    3807    3808
        3809    3810    3813    3817    3818    3826    3828    3834    3846    3848    3850    3853    3854    3855    3857
        3858    3860    3861    3867    3868    3871    3890    3893    3894    3897    3903    3907    3908    3909    3927
        3933    3935    3953    3955    3956    3976    3994    3997    4001    4003    4006    4011    4017    4018    4050
        4051    4060    4127    4134    4139    4152    4153    4163    4168    4231    4237    4240    4243    4246    4258
        4263    4272    4274    4275    4277    4279    4282    4289    4293    4298    4299    4301    4308    4310    4318
        4320    4322    4325    4326    4327    4331    4334    4337    4338    4340    4346    4348    4351    4391    4396
        4402    4405    4406    4411    4418    4454    4458    4589    4593    4597    4675    4677    4715    4717    4922
        4932    4936    4940    5067    5089    5104    5109    5113    5114    5130    5143    5169    5171    5175    5260
        5285    5288    5294    5295    5296    5299    5300    5306    5307    5310    5313    5318    5322
$GENTA  1730#   2549    2615    2618    2620    2621    2623    2665    2667    2678    2680    2681    2682    2687    2691
        2692    2693    2696    2697    2699    2702    2705    2708    2722    2728    2729    2731    2732    2733    2738
        2739    2742    2745    2746    2747    2749    2753    2755    2761    2762    2764    2765    2766    2767    2770
        2773    2774    2775    2777    2811    2813    2840    2847    2849    2876    2878    2883    2926    3001    3008
        3013    3016    3019    3022    3025    3033    3039    3043    3044    3051    3065    3066    3068    3072    3073
        3075    3078    3081    3083    3091    3099    3100    3110    3124    3127    3146    3149    3165    3170    3171
        3175    3176    3177    3178    3179    3180    3182    3197    3198    3203    3204    3205    3208    3210    3211
        3213    3216    3219    3225    3226    3227    3228    3229    3232    3286    3290    3291    3294    3299    3301
        3305    3306    3307    3347    3348    3353    3364    3371    3372    3373    3383    3387    3388    3390    3391
        3397    3398    3402    3408    3409    3414    3416    3417    3432    3441    3443    3497    3513    3522    3523
        3528    3529    3550    3552    3558    3559    3561    3562    3566    3567    3570    3573    3582    3583    3587
        3589    3593    3594    3595    3640    3657    3658    3677    3679    3682    3684    3687    3689    3692    3698
        3703    3704    3705    3706    3707    3708    3783    3784    3796    3800    3806    3812    3813    3815    3816
        3817    3818    3821    3822    3826    3830    3834    3846    3848    3852    3857    3860    3864    3865    3871
        3874    3875    3876    3877    3878    3893    3897    3905    3906    3912    3913    3914    3929    3937    3938
        3939    3953    3955    3958    3974    3994    3997    4003    4005    4009    4013    4023    4024    4060    4062
        4063    4127    4136    4139    4155    4159    4167    4169    4235    4240    4243    4244    4249    4255    4260
        4279    4281    4282    4284    4285    4286    4289    4301    4303    4304    4305    4308    4315    4320    4322
        4328    4330    4331    4333    4334    4337    4338    4339    4343    4346    4347    4354    4355    4391    4402
        4410    4411    4415    4416    4418    4454    4458    4589    4595    4597    4675    4677    4715    4717    4925
        4931    4932    4936    5070    5089    5104    5111    5113    5136    5137    5143    5168    5169    5171    5260
        5285    5290    5294    5298    5299    5300    5303    5304    5306    5312    5313    5315    5318    5322
$IF     1730#   2545    2606    2608    2613    2669    2670    2671    2676    2688    2689    2693    2700    2723    2724
        2725    2726    2734    2739    2743    2750    2756    2757    2758    2759    2767    2771    2838    2845    2873
```

	2881	2924	3003	3010	3028	3036	3040	3041	3047	3063	3070	3073	3076	3084	3095
	3159	3161	3163	3166	3167	3168	3172	3173	3195	3199	3200	3201	3205	3206	3217
	3218	3219	3221	3230	3283	3287	3288	3292	3294	3296	3297	3345	3349	3350	3354
	3357	3369	3375	3393	3395	3399	3405	3406	3429	3434	3484	3507	3515	3516	3524
	3548	3556	3564	3568	3571	3574	3578	3590	3591	3630	3644	3654	3668	3671	3675
	3693	3695	3700	3701	3771	3776	3794	3807	3808	3809	3810	3818	3828	3850	3853
	3854	3855	3858	3861	3867	3868	3890	3894	3903	3907	3908	3909	3927	3935	3956
	4001	4006	4011	4017	4018	4050	4051	4134	4152	4153	4163	4231	4237	4246	4272
	4275	4298	4299	4310	4318	4325	4326	4340	4348	4351	4396	4405	4406	4593	4922
	5067	5109	5114	5130	5288	5295	5296	5300	5307	5310					
$IFCOD	1730#	2545	2606	2608	2613	2665	2667	2669	2670	2671	2676	2687	2688	2689	2693
	2700	2722	2723	2724	2725	2726	2734	2739	2743	2750	2755	2756	2757	2758	2759
	2767	2771	2811	2838	2845	2873	2881	2924	3003	3009	3010	3013	3022	3028	3036
	3040	3041	3047	3063	3070	3073	3076	3083	3084	3095	3112	3127	3146	3159	3161
	3163	3166	3167	3168	3172	3173	3195	3199	3200	3201	3205	3206	3217	3218	3219
	3221	3230	3283	3287	3288	3292	3294	3296	3297	3345	3349	3350	3354	3357	3369
	3373	3375	3393	3395	3399	3405	3406	3429	3434	3484	3507	3515	3516	3524	3548
	3556	3561	3564	3568	3571	3574	3578	3588	3590	3591	3630	3644	3654	3668	3671
	3675	3679	3684	3689	3693	3695	3700	3701	3771	3776	3794	3800	3806	3807	3808
	3809	3810	3818	3828	3846	3850	3853	3854	3855	3858	3861	3867	3868	3890	3894
	3903	3907	3908	3909	3927	3933	3935	3953	3956	3976	3994	4001	4006	4011	4017
	4018	4050	4051	4127	4134	4152	4153	4163	4168	4231	4237	4240	4246	4258	4263
	4272	4275	4289	4298	4299	4310	4318	4320	4325	4326	4340	4348	4351	4391	4396
	4405	4406	4454	4589	4593	4675	4715	4922	4932	4940	5067	5089	5104	5109	5114
	5130	5169	5175	5260	5285	5288	5294	5295	5296	5300	5307	5310			
$IFCON	1730#	4274	4277	4293											
$IFOPR	1730#	2545	2606	2608	2613	2665	2667	2669	2670	2671	2676	2687	2688	2689	2693
	2700	2722	2723	2724	2725	2726	2734	2739	2743	2750	2755	2756	2757	2758	2759
	2767	2771	2811	2838	2845	2873	2881	2924	3003	3009	3010	3013	3022	3028	3036
	3040	3041	3047	3063	3070	3073	3076	3083	3084	3095	3112	3127	3146	3159	3161
	3163	3166	3167	3168	3172	3173	3195	3199	3200	3201	3205	3206	3217	3218	3219
	3221	3230	3283	3287	3288	3292	3294	3296	3297	3345	3349	3350	3354	3357	3369
	3373	3375	3393	3395	3399	3405	3406	3429	3434	3484	3507	3515	3516	3524	3548
	3556	3561	3564	3568	3571	3574	3578	3588	3590	3591	3630	3644	3654	3668	3671
	3675	3679	3684	3689	3693	3695	3700	3701	3771	3776	3794	3800	3806	3807	3808
	3809	3810	3818	3828	3846	3850	3853	3854	3855	3858	3861	3867	3868	3890	3894
	3903	3907	3908	3909	3927	3933	3935	3953	3956	3976	3994	4001	4006	4011	4017
	4018	4050	4051	4127	4134	4152	4153	4163	4168	4231	4237	4240	4246	4258	4263
	4272	4274	4275	4277	4289	4293	4298	4299	4310	4318	4320	4325	4326	4340	4348
	4351	4391	4396	4405	4406	4454	4589	4593	4675	4715	4922	4932	4940	5067	5089
	5104	5109	5114	5130	5169	5175	5260	5285	5288	5294	5295	5296	5300	5307	5310
$LET	1730#	2526	2527	2528	2530	2531	2532	2533	2543	2544	2547	2583	2587	2591	2595
	2607	2609	2610	2611	2612	2614	2616	2617	2619	2622	2635	2651	2652	2653	2664
	2672	2673	2674	2675	2677	2679	2685	2690	2695	2703	2704	2709	2736	2737	2751
	2788	2793	2796	2797	2798	2810	2812	2814	2815	2827	2839	2846	2848	2854	2856
	2864	2865	2872	2874	2877	2879	2880	2882	2885	2895	2896	2897	2898	2899	2919
	2920	2922	2923	2925	2933	2949	2973	2981	3000	3002	3011	3017	3018	3020	3021
	3023	3024	3029	3034	3037	3038	3067	3077	3079	3080	3082	3085	3089	3097	3109
	3111	3123	3126	3141	3142	3143	3144	3145	3147	3148	3150	3160	3162	3164	3169
	3174	3181	3207	3209	3212	3214	3222	3231	3254	3298	3300	3302	3355	3356	3359
	3360	3361	3362	3363	3365	3366	3367	3378	3381	3382	3384	3385	3386	3389	3394
	3396	3403	3404	3410	3415	3433	3435	3436	3437	3442	3459	3514	3517	3525	3554
	3557	3560	3563	3575	3576	3577	3579	3580	3581	3584	3585	3586	3620	3621	3622
	3623	3624	3625	3631	3632	3633	3634	3639	3645	3646	3647	3648	3649	3650	3651
	3669	3670	3672	3673	3674	3676	3678	3680	3681	3683	3685	3686	3688	3690	3691
	3694	3696	3697	3699	3769	3770	3772	3773	3774	3777	3778	3779	3780	3781	3793

```
        3795  3797  3799  3829  3831  3832  3845  3847  3859  3862  3863  3870  3872  3873  3889
        3898  3901  3902  3922  3934  3951  3952  3954  3959  3972  3975  3977  3989  3990  3991
        3992  3993  3995  3996  3998  4007  4012  4014  4015  4022  4025  4026  4054  4056  4057
        4058  4061  4073  4074  4075  4076  4078  4079  4125  4140  4149  4150  4154  4157  4158
        4161  4162  4165  4166  4239  4241  4242  4247  4248  4252  4253  4254  4256  4257  4259
        4261  4262  4267  4268  4271  4278  4280  4283  4288  4290  4291  4294  4295  4296  4300
        4302  4306  4323  4350  4357  4358  4360  4392  4394  4397  4399  4400  4403  4407  4412
        4431  4491  4492  4527  4528  4530  4531  4532  4534  4580  4581  4585  4590  4591  4592
        4602  4605  4610  4617  4624  4631  4638  4645  4652  4659  4666  4669  4671  4674  4678
        4682  4684  4686  4687  4689  4691  4693  4695  4696  4698  4703  4714  4716  4718  4917
        4918  4919  4920  4924  4926  4927  4928  4929  4930  4933  4934  4937  4939  4941  4950
        4951  4952  4954  4955  4956  4958  4994  4995  4996  4997  4998  4999  5000  5001  5002
        5003  5012  5026  5041  5043  5044  5045  5046  5069  5072  5073  5074  5077  5078  5091
        5092  5093  5094  5095  5096  5097  5098  5099  5101  5110  5112  5116  5118  5119  5120
        5121  5122  5123  5124  5126  5127  5129  5133  5138  5139  5140  5145  5161  5162  5163
        5164  5167  5172  5174  5176  5184  5186  5205  5206  5209  5210  5211  5212  5213  5214
        5216  5217  5218  5219  5220  5221  5222  5224  5241  5242  5243  5245  5246  5254  5255
        5256  5263  5267  5269  5270  5273  5277  5278  5282  5292  5297  5302  5308  5309  5314
        5316  5317  5337  5338  5339  5340  5341  5355  5356  5357  5358  5359  5360
$LPCNT  1730#  4322  4334
$OPABS  1730#
$OPADD  1730#  2583  2587  2591  2595  2607  2609  2672  2673  2703  2812  2896  2899  2923  3024
        3037  3148  3162  3169  3222  3254  3298  3300  3302  3355  3356  3362  3363  3365  3385
        3389  3403  3410  3415  3433  3435  3436  3442  3514  3517  3525  3560  3563  3576  3580
        3584  3669  3673  3674  3678  3681  3683  3686  3688  3691  3694  3696  3697  3699  3831
        3845  3922  3934  3954  3975  3990  3996  4007  4022  4025  4157  4161  4162  4242  4252
        4267  4268  4280  4283  4290  4300  4322  4334  4350  4397  4407  4412  4431  4591  4674
        4678  4933  4941  4996  5043  5112  5176  5216  5263  5267  5269  5273  5277  5282  5308
        5309  5316
$OPAND  1730#
$OPCD1  1730#  2611  3079  4278  4920  4928  4950  5164
$OPCD2  1730#  2543  2544  2583  2587  2591  2595  2607  2609  2610  2612  2614  2616  2617  2672
        2673  2675  2703  2812  2815  2882  2895  2896  2897  2898  2899  2920  2923  2933  2949
        2973  2981  3002  3024  3037  3080  3082  3085  3126  3142  3143  3148  3160  3162  3169
        3174  3214  3222  3254  3298  3300  3302  3355  3356  3362  3363  3365  3378  3385  3389
        3394  3396  3403  3404  3410  3415  3433  3435  3436  3442  3514  3517  3525  3560  3563
        3576  3580  3584  3622  3669  3672  3673  3674  3676  3678  3680  3681  3683  3685  3686
        3688  3690  3691  3694  3696  3697  3699  3793  3795  3829  3831  3845  3889  3898  3922
        3934  3954  3975  3990  3991  3996  4007  4012  4015  4022  4025  4150  4157  4161  4162
        4165  4166  4242  4252  4257  4259  4261  4262  4267  4268  4280  4283  4290  4291  4295
        4300  4322  4334  4350  4397  4400  4407  4412  4431  4491  4527  4531  4591  4592  4674
        4678  4919  4933  4934  4941  4996  4997  5043  5044  5110  5112  5163  5176  5216  5263
        5267  5269  5273  5277  5282  5308  5309  5316  5337  5358
$OPCOM  1730#  3079  4278  4920  4928  4950  5164
$OPDEF  1730#  2526  2527  2528  2530  2531  2532  2533  2543  2544  2545  2547  2583  2587  2591
        2595  2606  2607  2608  2609  2610  2611  2612  2613  2614  2616  2617  2618  2619  2621
        2622  2635  2651  2652  2653  2664  2665  2667  2669  2670  2671  2672  2673  2674  2675
        2676  2677  2679  2685  2687  2688  2689  2690  2692  2693  2695  2699  2700  2703  2704
        2705  2708  2709  2722  2723  2724  2725  2726  2729  2733  2734  2736  2737  2739  2742
        2743  2749  2750  2751  2755  2756  2757  2758  2759  2762  2766  2767  2770  2771  2777
        2788  2793  2796  2797  2798  2810  2811  2812  2813  2814  2815  2827  2838  2839  2845
        2846  2847  2848  2854  2856  2864  2865  2872  2873  2874  2876  2877  2879  2880  2881
        2882  2885  2895  2896  2897  2898  2899  2919  2920  2922  2923  2924  2925  2933  2949
        2973  2981  3000  3002  3003  3009  3010  3011  3013  3016  3017  3018  3020  3021  3022
        3023  3024  3025  3028  3029  3034  3036  3037  3038  3040  3041  3047  3063  3065  3067
        3070  3073  3076  3077  3078  3079  3080  3082  3083  3084  3085  3089  3091  3095  3097
```

```
                3109  3111  3112  3123  3126  3127  3141  3142  3143  3144  3145  3146  3147  3148  3149
                3150  3159  3160  3161  3162  3163  3164  3165  3166  3167  3168  3169  3171  3172  3173
                3174  3181  3195  3197  3199  3200  3201  3205  3206  3207  3208  3209  3211  3212  3214
                3216  3217  3218  3219  3221  3222  3230  3231  3254  3283  3287  3288  3292  3294  3296
                3297  3298  3299  3300  3302  3345  3347  3349  3350  3354  3355  3356  3357  3359  3360
                3361  3362  3363  3365  3366  3367  3369  3371  3373  3375  3378  3381  3382  3384  3385
                3386  3388  3389  3391  3393  3394  3395  3396  3399  3403  3404  3405  3406  3408  3410
                3411  3414  3415  3429  3433  3434  3435  3436  3437  3438  3441  3442  3459  3484  3507
                3511  3514  3515  3516  3517  3521  3523  3524  3525  3527  3548  3554  3556  3557  3558
                3560  3561  3563  3564  3566  3568  3569  3571  3574  3575  3576  3577  3578  3579  3580
                3581  3584  3585  3586  3588  3590  3591  3620  3621  3622  3623  3624  3625  3630  3631
                3632  3633  3634  3639  3644  3645  3646  3647  3648  3649  3650  3651  3654  3668  3669
                3670  3671  3672  3673  3674  3675  3676  3678  3679  3680  3681  3682  3683  3684  3685
                3686  3687  3688  3689  3690  3691  3692  3693  3694  3695  3696  3697  3698  3699  3700
                3701  3769  3770  3771  3772  3773  3774  3776  3777  3778  3779  3780  3781  3793  3794
                3795  3797  3799  3800  3806  3807  3808  3809  3810  3813  3817  3818  3826  3828  3829
                3831  3832  3834  3845  3846  3847  3848  3850  3853  3854  3855  3857  3858  3859  3860
                3861  3862  3863  3867  3868  3870  3871  3872  3873  3889  3890  3893  3894  3897  3898
                3901  3902  3903  3907  3908  3909  3922  3927  3933  3934  3935  3951  3952  3953  3954
                3955  3956  3959  3972  3975  3976  3977  3989  3990  3991  3992  3993  3994  3995  3996
                3997  3998  4001  4003  4006  4007  4011  4012  4014  4015  4017  4018  4022  4025  4026
                4050  4051  4054  4056  4057  4058  4060  4061  4073  4074  4075  4076  4078  4079  4125
                4127  4134  4139  4140  4149  4150  4152  4153  4154  4157  4158  4161  4162  4163  4165
                4166  4168  4231  4237  4239  4240  4241  4242  4243  4246  4247  4248  4252  4253  4254
                4256  4257  4258  4259  4261  4262  4263  4267  4268  4271  4272  4274  4275  4277  4278
                4279  4280  4282  4283  4288  4289  4290  4291  4293  4294  4295  4296  4298  4299  4300
                4301  4302  4306  4308  4310  4318  4320  4322  4323  4325  4326  4327  4331  4334  4337
                4338  4340  4346  4348  4350  4351  4357  4358  4360  4391  4392  4394  4396  4397  4399
                4400  4402  4403  4405  4406  4407  4411  4412  4418  4431  4454  4458  4491  4492  4527
                4528  4530  4531  4532  4534  4580  4581  4585  4589  4590  4591  4592  4593  4597  4602
                4605  4610  4617  4624  4631  4638  4645  4652  4659  4666  4669  4671  4674  4675  4677
                4678  4682  4684  4686  4687  4689  4691  4693  4695  4696  4698  4703  4714  4715  4716
                4717  4718  4917  4918  4919  4920  4922  4924  4926  4927  4928  4929  4930  4932  4933
                4934  4936  4937  4939  4940  4941  4950  4951  4952  4954  4955  4956  4958  4994  4995
                4996  4997  4998  4999  5000  5001  5002  5003  5012  5026  5041  5043  5044  5045  5046
                5067  5069  5072  5073  5074  5077  5078  5089  5091  5092  5093  5094  5095  5096  5097
                5098  5099  5101  5104  5109  5110  5112  5113  5114  5116  5118  5119  5120  5121  5122
                5123  5124  5126  5127  5129  5130  5133  5138  5139  5140  5143  5145  5161  5162  5163
                5164  5167  5169  5171  5172  5174  5175  5176  5184  5186  5205  5206  5209  5210  5211
                5212  5213  5214  5216  5217  5218  5219  5220  5221  5222  5224  5241  5242  5243  5245
                5246  5254  5255  5256  5260  5263  5267  5269  5270  5273  5277  5278  5282  5285  5288
                5292  5294  5295  5296  5297  5299  5300  5302  5306  5307  5308  5309  5310  5313  5314
                5316  5317  5318  5322  5337  5338  5339  5340  5341  5355  5356  5357  5358  5359  5360
$OPEQU          1730#
$OPNAN          1730#
$OPNEG          1730#
$OPNOR          1730#
$OPNOT          1730#  2543  2610  2616  2675  2897  3080  3142  3622  3793  3991  4592  4934  4997  5044
                5358
$OPOR           1730#  2617  2882  3396  3404  3795
$OPROT          1730#
$OPRO           1730#  2526  2527  2528  2530  2531  2532  2533  2547  2619  2622  2635  2651  2652  2653
                2664  2674  2677  2679  2685  2690  2695  2704  2709  2736  2737  2751  2788  2793  2796
                2797  2798  2810  2814  2827  2839  2846  2848  2854  2856  2864  2865  2872  2874  2877
                2879  2880  2885  2919  2922  2925  3000  3011  3017  3018  3020  3021  3023  3029  3034
                3038  3067  3077  3089  3097  3109  3111  3123  3141  3144  3145  3147  3150  3164  3181
```

```
                3207  3209  3212  3231  3359  3360  3361  3366  3367  3381  3382  3384  3386  3437  3459
                3554  3557  3575  3577  3579  3581  3585  3586  3620  3621  3623  3624  3625  3631  3632
                3633  3634  3639  3645  3646  3647  3648  3649  3650  3651  3670  3769  3770  3772  3773
                3774  3777  3778  3779  3780  3781  3797  3799  3832  3847  3859  3862  3863  3870  3872
                3873  3901  3902  3951  3952  3959  3972  3977  3989  3992  3993  3995  3998  4014  4026
                4054  4056  4057  4058  4061  4073  4074  4075  4076  4078  4079  4125  4140  4149  4154
                4158  4239  4241  4247  4248  4253  4254  4256  4271  4288  4294  4296  4302  4306  4322
                4323  4334  4357  4358  4360  4392  4394  4399  4403  4492  4528  4530  4532  4534  4580
                4581  4585  4590  4602  4605  4610  4617  4624  4631  4638  4645  4652  4659  4666  4669
                4671  4682  4684  4686  4687  4689  4691  4693  4695  4696  4698  4703  4714  4716  4718
                4917  4918  4924  4926  4927  4929  4930  4937  4939  4951  4952  4954  4955  4956  4958
                4994  4995  4998  4999  5000  5001  5002  5003  5012  5026  5041  5045  5046  5069  5072
                5073  5074  5077  5078  5091  5092  5093  5094  5095  5096  5097  5098  5099  5101  5116
                5118  5119  5120  5121  5122  5123  5124  5126  5127  5129  5133  5138  5139  5140  5145
                5161  5162  5167  5172  5174  5184  5186  5205  5206  5209  5210  5211  5212  5213  5214
                5217  5218  5219  5220  5221  5222  5224  5241  5242  5243  5245  5246  5254  5255  5256
                5270  5278  5292  5297  5302  5314  5317  5338  5339  5340  5341  5355  5356  5357  5359
                5360
$OPR1           1730#  2611  3079  4278  4920  4928  4950  5164
$OPR2           1730#  2543  2544  2583  2587  2591  2595  2607  2609  2610  2612  2614  2616  2617  2672
                2673  2675  2703  2812  2815  2882  2895  2896  2897  2898  2899  2920  2923  2933  2949
                2973  2981  3002  3024  3037  3080  3082  3085  3126  3142  3143  3148  3160  3162  3169
                3174  3214  3222  3254  3298  3300  3302  3355  3356  3362  3363  3365  3378  3385  3389
                3394  3396  3403  3404  3410  3415  3433  3435  3436  3442  3514  3517  3525  3560  3563
                3576  3580  3584  3622  3669  3672  3673  3674  3676  3678  3680  3681  3683  3685  3686
                3688  3690  3691  3694  3696  3697  3699  3793  3795  3829  3831  3845  3889  3898  3922
                3934  3954  3975  3990  3991  3996  4007  4012  4015  4022  4025  4150  4157  4161  4162
                4165  4166  4242  4252  4257  4259  4261  4262  4267  4268  4280  4283  4290  4291  4295
                4300  4322  4334  4350  4397  4400  4407  4412  4431  4491  4527  4531  4591  4592  4674
                4678  4919  4933  4934  4941  4996  4997  5043  5044  5110  5112  5163  5176  5216  5263
                5267  5269  5273  5277  5282  5308  5309  5316  5337  5358
$OPSHF          1730#  2815  2895  3143  3214  3394  4015  4150  4259  4262  4291  4400  4491  4527  5337
$OPSUB          1730#  2544  2612  2614  2898  2920  2933  2949  2973  2981  3002  3082  3085  3126  3160
                3174  3378  3672  3676  3680  3685  3690  3829  3889  3898  4012  4165  4166  4257  4261
                4295  4531  4919  5110  5163  5337
$OPSWB          1730#  2611
$OPXOR          1730#
$OR             1730#  2693  2739  2767  3073  3205  3219  3294  3818  5300
$PUT            1730#
$STRUC          1730#
$SUBON          1730#  2549  2615  2618  2620  2621  2623  2678  2680  2681  2682  2691  2692  2696  2697
                2699  2702  2705  2708  2728  2729  2731  2732  2733  2738  2742  2745  2746  2747  2749
                2753  2761  2762  2764  2765  2766  2770  2773  2774  2775  2777  2813  2840  2847  2849
                2876  2878  2883  2926  3008  3009  3016  3019  3025  3033  3039  3043  3044  3051  3065
                3066  3072  3075  3078  3081  3083  3091  3099  3100  3112  3127  3149  3165  3170  3171
                3175  3176  3177  3178  3179  3180  3182  3197  3198  3203  3204  3208  3210  3211  3213
                3216  3225  3226  3227  3228  3229  3232  3286  3290  3291  3299  3301  3305  3306  3307
                3347  3348  3353  3371  3372  3373  3383  3387  3388  3390  3391  3397  3398  3402  3408
                3409  3414  3416  3417  3432  3441  3443  3497  3513  3522  3523  3528  3529  3558  3559
                3561  3562  3566  3567  3570  3573  3582  3583  3587  3588  3589  3593  3594  3595  3640
                3657  3658  3677  3682  3687  3692  3698  3703  3704  3705  3706  3707  3708  3783  3784
                3796  3812  3813  3815  3816  3817  3821  3822  3826  3830  3834  3848  3852  3857  3860
                3864  3865  3871  3874  3875  3876  3877  3878  3893  3897  3905  3912  3913  3914  3929
                3933  3937  3938  3939  3955  3958  3976  3997  4003  4005  4009  4013  4023  4024  4060
                4062  4063  4136  4139  4155  4167  4168  4169  4235  4243  4244  4249  4258  4263  4279
                4281  4282  4284  4285  4286  4301  4303  4304  4305  4308  4315  4322  4328  4330  4331
```

```
                4333    4334    4337    4338    4339    4343    4346    4347    4354    4355    4402    4410    4411    4415    4416
                4418    4458    4595    4597    4677    4717    4925    4936    4940    5070    5111    5113    5136    5137    5143
                5171    5175    5290    5298    5299    5303    5304    5306    5312    5313    5315    5318    5322
$THEN           1730#   2545    2606    2608    2613    2669    2670    2671    2676    2688    2689    2693    2700    2723    2724
                2725    2726    2734    2739    2743    2750    2756    2757    2758    2759    2767    2771    2838    2845    2873
                2881    2924    3003    3010    3028    3036    3040    3041    3047    3063    3070    3073    3076    3084    3095
                3159    3161    3163    3166    3167    3168    3172    3173    3195    3199    3200    3201    3205    3206    3217
                3218    3219    3221    3230    3283    3287    3288    3292    3294    3296    3297    3345    3349    3350    3354
                3357    3369    3375    3393    3395    3399    3405    3406    3429    3434    3484    3507    3515    3516    3524
                3548    3556    3564    3568    3571    3574    3578    3590    3591    3630    3644    3654    3668    3671    3675
                3693    3695    3700    3701    3771    3776    3794    3807    3808    3809    3810    3818    3828    3850    3853
                3854    3855    3858    3861    3867    3868    3890    3894    3903    3907    3908    3909    3927    3935    3956
                4001    4006    4011    4017    4018    4050    4051    4134    4152    4153    4163    4231    4237    4246    4272
                4275    4298    4299    4310    4318    4325    4326    4340    4348    4351    4396    4405    4406    4593    4922
                5067    5109    5114    5130    5288    5295    5296    5300    5307    5310
$TILA           1730#
$TILO           1730#
$UNTL2          1730#   3083    3127    3373    3561
$UNTL3          1730#
$WHILE          1730#   2665    2667    2687    2722    2755    2811    3013    3022    3146    3679    3684    3689    3800    3806
                3846    3953    3994    4127    4240    4289    4320    4391    4454    4589    4675    4715    4932    5089    5104
                5169    5260    5285    5294
$$DEFA          1730#
$$ENDS          1730#
$$ERRO          1730#
$$GEN           1730#   2549    2615    2618    2620    2621    2623    2665    2667    2678    2680    2681    2682    2687    2691
                2692    2693    2696    2697    2699    2702    2705    2708    2722    2728    2729    2731    2732    2733    2738
                2739    2742    2745    2746    2747    2749    2753    2755    2761    2762    2764    2765    2766    2767    2770
                2773    2774    2775    2777    2811    2813    2840    2847    2849    2876    2878    2883    2926    3001    3008
                3013    3016    3019    3022    3025    3033    3039    3043    3044    3051    3065    3066    3068    3072    3073
                3075    3078    3081    3083    3091    3099    3100    3110    3124    3127    3146    3149    3165    3170    3171
                3175    3176    3177    3178    3179    3180    3182    3197    3198    3203    3204    3205    3208    3210    3211
                3213    3216    3219    3225    3226    3227    3228    3229    3232    3286    3290    3291    3294    3299    3301
                3305    3306    3307    3347    3348    3353    3364    3371    3372    3373    3383    3387    3388    3390    3391
                3397    3398    3402    3408    3409    3414    3416    3417    3432    3441    3443    3497    3513    3523    3523
                3528    3529    3550    3552    3558    3559    3561    3562    3566    3567    3570    3573    3582    3583    3587
                3589    3593    3594    3595    3640    3657    3658    3677    3679    3682    3684    3687    3689    3692    3698
                3703    3704    3705    3706    3707    3708    3783    3784    3796    3800    3806    3812    3813    3815    3816
                3817    3818    3821    3822    3826    3830    3834    3846    3848    3852    3857    3860    3864    3865    3871
                3874    3875    3876    3877    3878    3893    3897    3905    3906    3912    3913    3914    3929    3937    3938
                3939    3953    3955    3958    3974    3994    3997    4003    4005    4009    4013    4023    4024    4060    4062
                4063    4127    4136    4139    4155    4159    4167    4169    4235    4240    4243    4244    4249    4255    4260
                4279    4281    4282    4284    4285    4286    4289    4301    4303    4304    4305    4308    4315    4320    4322
                4328    4330    4331    4333    4334    4337    4338    4339    4343    4346    4347    4354    4355    4391    4402
                4410    4411    4415    4416    4418    4454    4458    4589    4595    4597    4675    4677    4715    4717    4925
                4931    4932    4936    5070    5089    5104    5111    5113    5136    5137    5143    5168    5169    5171    5260
                5285    5290    5294    5298    5299    5300    5303    5304    5306    5312    5313    5315    5318    5322
$$GETS          1730#   2549    2615    2618    2620    2621    2623    2678    2680    2681    2682    2691    2692    2696    2697
                2699    2702    2705    2708    2728    2729    2731    2732    2733    2738    2742    2745    2746    2747    2749
                2753    2761    2762    2764    2765    2766    2770    2773    2774    2775    2777    2813    2840    2847    2849
                2876    2878    2883    2926    3008    3009    3016    3019    3025    3033    3039    3043    3044    3051    3065
                3066    3072    3075    3078    3081    3083    3091    3099    3100    3112    3127    3149    3165    3170    3171
                3175    3176    3177    3178    3179    3180    3182    3197    3198    3203    3204    3208    3210    3211    3213
                3216    3225    3226    3227    3228    3229    3232    3286    3290    3291    3299    3301    3305    3306    3307
                3347    3348    3353    3371    3372    3373    3383    3387    3388    3390    3391    3397    3398    3402    3408
                3409    3414    3416    3417    3432    3441    3443    3497    3513    3522    3523    3528    3529    3558    3559
```

```
                3561  3562  3566  3567  3570  3573  3582  3583  3587  3588  3589  3593  3594  3595  3640
                3657  3658  3677  3682  3687  3692  3698  3703  3704  3705  3706  3707  3708  3783  3784
                3796  3812  3813  3815  3816  3817  3821  3822  3826  3830  3834  3848  3852  3857  3860
                3864  3865  3871  3874  3875  3876  3877  3878  3893  3897  3905  3912  3913  3914  3929
                3933  3937  3938  3939  3955  3958  3976  3997  4003  4005  4009  4013  4023  4024  4060
                4062  4063  4136  4139  4155  4167  4168  4169  4235  4243  4244  4249  4258  4263  4279
                4281  4282  4284  4285  4286  4301  4303  4304  4305  4308  4315  4322  4328  4330  4331
                4333  4334  4337  4338  4339  4343  4346  4347  4354  4355  4402  4410  4411  4415  4416
                4418  4458  4595  4597  4677  4717  4925  4936  4940  5070  5111  5113  5136  5137  5143
                5171  5175  5290  5298  5299  5303  5304  5306  5312  5313  5315  5318  5322
$$GETT          1730#  2618  2621  2692  2729  2733  2742  2762  2766  2770  2847  2876  3065  3078  3091
                3165  3171  3197  3208  3211  3216  3299  3347  3371  3388  3391  3408  3414  3441  3523
                3558  3566  3570  3698  3813  3817  3857  3860  3871  3893  3897  4003  4060  4279  4282
                4301  4328  4331  4402  4411  5299  5313
$$LPCN          1730#  4322  4334
$$POP           1730#  2549  2615  2618  2620  2621  2623  2678  2680  2681  2682  2691  2692  2696  2697
                2699  2702  2705  2708  2728  2729  2731  2732  2733  2738  2742  2745  2746  2747  2749
                2753  2761  2762  2764  2765  2766  2770  2773  2774  2775  2777  2813  2840  2847  2849
                2876  2878  2883  2926  3008  3009  3016  3019  3025  3033  3039  3043  3044  3051  3065
                3066  3072  3075  3078  3081  3083  3091  3099  3100  3112  3127  3149  3165  3170  3171
                3175  3176  3177  3178  3179  3180  3182  3197  3198  3203  3204  3208  3210  3211  3213
                3216  3225  3226  3227  3228  3229  3232  3286  3290  3291  3299  3301  3305  3306  3307
                3347  3348  3353  3371  3372  3373  3383  3387  3388  3390  3391  3397  3398  3402  3408
                3409  3414  3416  3417  3432  3441  3443  3497  3513  3522  3523  3528  3529  3558  3559
                3561  3562  3566  3567  3570  3573  3582  3583  3587  3588  3589  3593  3594  3595  3640
                3657  3658  3677  3682  3687  3692  3698  3703  3704  3705  3706  3707  3708  3783  3784
                3796  3812  3813  3815  3816  3817  3821  3822  3826  3830  3834  3848  3852  3857  3860
                3864  3865  3871  3874  3875  3876  3877  3878  3893  3897  3905  3912  3913  3914  3929
                3933  3937  3938  3939  3955  3958  3976  3997  4003  4005  4009  4013  4023  4024  4060
                4062  4063  4136  4139  4155  4167  4168  4169  4235  4243  4244  4249  4258  4263  4279
                4281  4282  4284  4285  4286  4301  4303  4304  4305  4308  4315  4322  4328  4330  4331
                4333  4334  4337  4338  4339  4343  4346  4347  4354  4355  4402  4410  4411  4415  4416
                4418  4458  4595  4597  4677  4717  4925  4936  4940  5070  5111  5113  5136  5137  5143
                5171  5175  5290  5298  5299  5303  5304  5306  5312  5313  5315  5318  5322
$$PUSH          1730#  2545  2606  2608  2613  2618  2621  2665  2667  2669  2670  2671  2676  2687  2688
                2689  2692  2693  2699  2700  2705  2708  2722  2723  2724  2725  2726  2729  2733  2734
                2739  2742  2743  2749  2750  2755  2756  2757  2758  2759  2762  2766  2767  2770  2771
                2777  2811  2813  2838  2845  2847  2873  2876  2881  2924  3001  3003  3010  3013  3016
                3022  3025  3028  3036  3040  3041  3047  3063  3065  3068  3070  3073  3076  3078  3084
                3091  3095  3110  3124  3146  3149  3159  3161  3163  3165  3166  3167  3168  3171  3172
                3173  3195  3197  3199  3200  3201  3205  3206  3208  3211  3216  3217  3218  3219  3221
                3230  3283  3287  3288  3292  3294  3296  3297  3299  3345  3347  3349  3350  3354  3357
                3364  3369  3371  3375  3388  3391  3393  3395  3399  3405  3406  3408  3414  3429  3434
                3441  3484  3507  3515  3516  3523  3524  3548  3549  3550  3551  3552  3556  3558  3564
                3566  3568  3570  3571  3574  3578  3590  3591  3630  3644  3654  3668  3671  3675  3679
                3682  3684  3687  3689  3692  3693  3695  3698  3700  3701  3771  3776  3794  3800  3806
                3807  3808  3809  3810  3813  3817  3818  3826  3828  3834  3846  3848  3850  3853  3854
                3855  3857  3858  3860  3861  3867  3868  3871  3890  3893  3894  3897  3903  3906  3907
                3908  3909  3927  3935  3953  3955  3956  3974  3994  3997  4001  4003  4006  4011  4017
                4018  4050  4051  4060  4127  4134  4139  4152  4153  4159  4163  4231  4237  4240  4243
                4246  4255  4260  4272  4274  4275  4277  4279  4282  4289  4293  4298  4299  4301  4308
                4310  4318  4320  4321  4322  4325  4326  4328  4331  4334  4340  4346  4348  4351  4391
                4396  4402  4405  4406  4411  4418  4454  4458  4589  4593  4597  4675  4677  4715  4717
                4922  4931  4932  4936  5067  5089  5104  5109  5113  5114  5130  5143  5168  5169  5171
                5260  5285  5288  5294  5295  5296  5299  5300  5306  5307  5310  5313  5318  5322
$$SELE          1730#
```

```
$$SETS  1730#  2545  2606  2608  2613  2618  2621  2665  2667  2669  2670  2671  2676  2687  2688
        2689   2692  2693  2699  2700  2705  2708  2722  2723  2724  2725  2726  2729  2733  2734
        2739   2742  2743  2749  2750  2755  2756  2757  2758  2759  2762  2766  2767  2770  2771
        2777   2811  2813  2838  2845  2847  2873  2876  2881  2924  3001  3003  3010  3013  3016
        3022   3025  3028  3036  3040  3041  3047  3063  3065  3068  3070  3073  3076  3078  3084
        3091   3095  3110  3124  3146  3149  3159  3161  3163  3165  3166  3167  3168  3171  3172
        3173   3195  3197  3199  3200  3201  3205  3206  3208  3211  3216  3217  3218  3219  3221
        3230   3283  3287  3288  3292  3294  3296  3297  3299  3345  3347  3349  3350  3354  3357
        3364   3369  3371  3375  3388  3391  3393  3395  3399  3405  3406  3408  3414  3429  3434
        3441   3484  3507  3515  3516  3523  3524  3548  3549  3550  3551  3552  3556  3558  3564
        3566   3568  3570  3571  3574  3578  3590  3591  3630  3644  3654  3668  3671  3675  3679
        3682   3684  3687  3689  3692  3693  3695  3698  3700  3701  3771  3776  3704  3800  3806
        3807   3808  3809  3810  3813  3817  3818  3826  3828  3834  3846  3848  3850  3853  3854
        3855   3857  3858  3860  3861  3867  3868  3871  3890  3893  3894  3897  3903  3906  3907
        3908   3909  3927  3935  3953  3955  3956  3974  3994  3997  4001  4003  4006  4011  4017
        4018   4050  4051  4060  4127  4134  4139  4152  4153  4159  4163  4231  4237  4240  4243
        4246   4255  4260  4272  4274  4275  4277  4279  4282  4289  4293  4298  4299  4301  4308
        4310   4318  4320  4321  4322  4325  4326  4328  4331  4334  4340  4346  4348  4351  4391
        4396   4402  4405  4406  4411  4418  4454  4458  4589  4593  4597  4675  4677  4715  4717
        4922   4931  4932  4936  5067  5089  5104  5109  5113  5114  5130  5143  5168  5169  5171
        5260   5285  5288  5294  5295  5296  5299  5300  5306  5307  5310  5313  5318  5322
$$SETT  1730#

. ABS.  030700     000

 ERRORS DETECTED:  0

 CZTUVB,CZTUVB/CRF=SVC.SML/ML,SPMAC.SML/ML,CZTUVB.P11
 RUN-TIME: 149 152 10 SECONDS
 RUN-TIME RATIO: 347/312=1.1
 CORE USED:  31K  (62 PAGES)
```