```
;       RCSL:   44-RT 1032

;       VERSION:  13.04.75   JEP


;               PROGRAM
;               *******
;
;       RC 3600 - MAGNETIC TAPE RELIABILITY
;
;               TAPES
;
;       BINARY: RCSL  44-RT 1033        DEVICECODE  30
;       BINARY: RCSL  44-RT 1034        DEVICECODE  44
;
;1.ABSTRACT
;       THE TAPE CONTROL RELIABILITY IS A MAINTENANCE
;       PROGRAM INTENDED FOR RIGOROUS TESTING
;       OF A SYSTEM THAT HAS SUCCESSFULLY RUN
;       THE DIAGNOSTIC AND TIMING TEST.

;2.MACHINE REQUIREMENTS
;       NOVA FAMILY PROCESSOR
;       8K READ/WRITE CORE MEMORY       WARNING ! ! !
;       DISPLAY             DIS  X
;       NUMERIC KEYBOARD    NUK  X  OR TELETYPE
;       FUNCTION BUTTONS    FUB  X
;       MAGNETIC TAPE UNIT
;       LINEPRINTER, OPTIONALLY

;3.SWITCH SETTINGS
;       3.1 STARTING ADDRESSES
;               400 RELIABILITY TEST
;               401 INTERCHANGE TEST (READ&WRITE)
;               402 INTERCHANGE TEST (READ ONLY)
;               405 TEST LOOP BUILDER

;       3.2 CONTROL
;
;               SW2=USE TTY INSTEAD OF DISPLAY AND NUK
;               SW3=DON'T PRINT PARITY ERRORS
;               SW4=OLD CONTROLLER TYPE, SEE CHECK ROUTINE
;               SW5=WAIT 1SEC INSTEAD OF 3SEC AFTER DIS MESS
;               SW6=NO WAITING DISPLAY
;               SW7=HALT AFTER DISPLAY MESSAGE
;               SW8=CLEAR DISPLAY AFTER MESSAGE
;
;       3.3 START
;               OF PROGRAM THE BEST WAY TO INSURE
;               CORRECT SWITCH SETTINGS:
;               RESET
;               SET SWITCHES TO START ADDR. (3.1)
;               EXAMINE
;               SET SWITCHES TO CONTROL. (3.2)
;               CONTINUE

;       3.4 CPUNO
;               IF ASKED, INSERT 2 TO 6 DEPENDING HOW
;               FAST IS YOUR CPU. THIS WAY:
;               RESET
;               DEPOSIT # INTO AC2
;               SET SWITCHES TO CONTROL. (3.2)
;               CONTINUE.
```

;4.OPERATING PROCEDURE

LOAD PROGRAM USING THE BINARY LOADER

THE PROGRAM CONTAINS A STANDARD PRINTER TABLE. IF THE PRINTER HAS ANOTHER DRUM, CHANGE TABLE BY READING ANOTHER BINARY TAPE UPON THE PROGRAM. FOR DETAILS SEE RCSL 44-RT 528-536.

THIS PROGRAM HAS SOME ERROR HISTORY INFORMATION COUNTERS. AFTER A TEST CYCLE OR IF TERMINATED THE HISTORY IS WRITTEN, ALL NUMBERS IN DECIMAL. IF THE COUNTERS ARE CLEARED, THE MESSAGE "INFO BITS CLEAR" IS WRITTEN. EACH UNIT HAS ITS OWN SET OF COUNTERS:

PAR WR	PARITY ERRORS DURING WRITE/READ
PAR RD	COUNTED ONLY IN FIRST ATTEMT AND RESTORED IF PERMANENT COUNT
PERM WR	PERMANENT PARITYERROR IF NO MORE
PERM RD	RETRY READING OR AFTER ERASE IN WRITING. PARITY ERROR NOT COUNT.
DATA ER	DATA ERROR ONE COUNT/BLOCK OR EOF INCL. CHECKCHARS & BITS IN UNPACK
WD RD	WORDS READ SUM OF ALL BLOCKS.
WD WR	WORDS WRITTEN IN ALL BLOCKS. WORDS PRINTED IN DECIMAL.
RECORD#	THE RECORD# BEFORE HISTORY PRINT

4.0 PRESS INT (ESC) TO TERMINATE THE TEST, WAIT FOR DIS (TTO) WRITING TERMINATED, THEN PRESS KEY AT NUK (TTI):

(LF) NL=CLEAR ERRORS
SP=PRINT ERRORS
??=PRINT AND THEN CLEAR ERRORS

THEN THE TEST STOPS IN HALT:

1. RESTART TEST OR
2. PRESS CONTINUE

4.1 RELIABILITY TEST

SET ALL UNITS TO BE TESTED ONLINE.
START. ERRORS WILL BE REPORTED.

4.2 INTERCHANGE TEST (READ&WRITE)

MOUNT TAPES ON ALL UNITS TO BE TESTED AND SET THEM ONLINE. START.
WHEN ASKED BY PROGRAM, REMOVE TAPES AND MOUNT THEM ON THE UNIT TO THE RIGHT. PRESS A NUK OR TTY KEY. ERRORS ARE REPORTED.

4.3 INTERCHANGE TEST (READ ONLY)

MOUNT TAPES WRITTEN USING THE MODE DESCRIBED ABOVE IN PAR. 4.2 ONTO ALL UNITS TO BE TESTED AND SET THEM ONLINE.
START. WHEN ASKED BY PROGRAM, REMOVE TAPES AND MOUNT THEM ON THE UNIT TO THE RIGHT. PRESS A NUK OR TTY KEY.
ERRORS ARE REPORTED.

```
;       4.4 TEST LOOP BUILDER

;       NL OR LF WILL TERMINATE EVERY QUESTION.
;       CAN OR RUBOUT WILL RUBOUT THE LAST CHAR.

;       SET THE UNIT TO BE TESTED ONLINE.
;       START.
;       PROGRAM WILL RESPOND WITH "UNIT".  TYPE IN THE
;       UNIT # OR NL, SEE BELOW.
;       PROGRAM WILL RESPOND WITH "WC", TYPE
;       IN WORD COUNT. (IN OCTAL, MAX 1777).
;       PROGRAM WILL RESPOND WITH "DATA",
;       TYPE IN THE DATA CODE -

;       DATA CODES:

;       USE
;       60      RAND = RANDOM DATA
;       61      SKEW = SKEW PATTERN
;       7011    ALL1 = ALL ONES
;       7000    ALL0 = ALL ZEROS
;       7001    ALT0 = ALTERNATE ONES AND ZEROS,
;                      STARTING WITH ZERO
;       7010    ALT1 = ALTERNATE ONES AND ZEROS,
;                      STARTING WITH ONES
;       80N-N   WDN-N= FILL WRITE BUFFER WITH DATA N-N
;                      WHERE N-N IS OCTAL NO. MAX 177777.

;       PROGRAM WILL THEN ASK FOR PARITY TYPE (PAR).
;       ENTER "ODD AND PE" OR "EVEN".

;       USE
;       90      EVEN
;       91      ODD AND PE

;       PROGRAM WILL RESPOND WITH "COMMAND
;       STRING",  KEY IN THE COMMAND STRING.

;       COMMANDS -

;       USE
;       +       /  = END COMMAND
;       -       ,  = PARAMETER SIGN

;       10      RD = READ                   +OCTAL #BLOCKS
;       11      RE = READ EOF
;       12      RW = REWIND
;       13      RU = REWIND OFFLINE
;       20      SF = SPACE FORWARD          +OCTAL #BLOCKS  *)
;       21      SB = SPACE BACK             +OCTAL #BLOCKS  *)
;       30      WT = WRITE                  +OCTAL #BLOCKS
;       31      WE = WRITE EOF
;       40      ER = ERASE

;       50      LOOP * = LOOP BACK TO *
;       51      LOOP = GO BACK TO BEGINNING OF LINE

;       SP      *  = NO OPERATION

;               *) STOP ON EOF AND BOT.
```

;
; SAMPLE COMMAND STRING
;
; 12+30-10-+12+ +10-10-+21-10-+50+ MEANS
; RW/WT,10,/RW/*/RD,10,/SB,10,/LOOP *
;
; THIS ROUTINE WILL REWIND, WRITE 8 RECORDS
; AND THEN READ 8, AND BACKSPACE 8,
; CONTINUALLY. NO LETTERS AT NUK, THEREFORE
; YOU HAVE TO KEY IN NUMBERS ONLY.
;
; PRESS INT TO TERMINATE THE TEST.
; SEE POINT 4.0
;
; TYPING "LINEFEED(NL OR LF)" IN RESPONSE TO "UNIT"
; WILL CAUSE THE PREVIOUS COMMAND STRING TO
; BE EXECUTED AGAIN.

;5. DESCRIPTION

; 5.1 RELIABILITY TEST
; THE RELIABILITY TEST ASSUMES THAT ALL
; UNIT BEING TESTED ARE FUNCTIONAL TO
; A CERTAIN EXTENT. IT SEARCHES FOR ALL
; ONLINE UNITS, REWINDS THEM AND WRITES
; A FILE MARK AT THE BOT. AMONG THESE UNITS
; IT PICKS ONE AT RANDOM AND PERFORMS THE
; FOLLOWING:
; A.PICK RANDOM NUMBER OF RECORDS (1-7)
; B.PICK RANDOM STARTING ADDRESS
;   COMMON FOR ALL RECORDS IN THAT FILE.
; C.GENERATE RANDOM WORD LENGTH BUFFER
;   MODULO 512 FOR EACH RECORD.
; D.WRITE THE FILE FOLLOWED BY FILE MARK.
; E.BACKSPACE OVER THE FILE.
; F.READ AND CHECK THE FILE.
; G.PICK ANOTHER UNIT AND START AGAIN.

; 5.2 INTERCHANGE TEST (RD&WT)
; THIS IS A WORST CASE SKEW TEST THAT
; IS PERFORMED TO TEST THE INTERCHANG-
; ABILITY OF TAPES WRITTEN ON ONE UNIT
; TO READING ON ANOTHER. IT PROCEEDS
; AS FOLLOWS:

; A.FIND AVAILABLE ONLINE UNITS.
; B.WRITE THE SKEW PATTERN 100 TIMES
;   ON EACH, BLOCKLENGTH 400.
; C.WRITE 100 RANDOM DATA RECORDS BLOCKLENGTH 400,
;   FOLLOWED BY 2 EOF, ON EACH UNIT.
; D.REWIND ALL UNITS.
; E.READ AND CHECK THE SKEW PATTERN ON
;   ALL UNITS.
; F.READ AND CHECK THE RANDOM DATA AND EOF'S
;   ON ALL UNITS.
; G.TYPE MESSAGE ON DIS OR TTY TO HAVE OPERATOR TO
;   MOVE TAPES TO ANOTHER UNIT FOR THE
;   READ TEST AGAIN. GO BACK TO STEP D.

5.3 INTERCHANGE TEST (READ ONLY)
THIS IS THE SAME AS THE PREVIOUSLY
DISCUSSED TEST IN PAR. 5.2, EXCEPT
THAT STEPS B THRU C ARE BYPASSED

5.4 TEST LOOP BUILDER
THIS PROGRAM IS INTENDED FOR USE BY
THE TECHNICIAN AS A DEBUGGING AID.
IF A CERTAIN SEQUENCE OF EVENTS IS
NEEDED TO SHOW UP A PARTICULAR PROBLEM
THAT SEQUENCE CAN BE TYPED IN AS A COMMAND
STRING WHICH WILL THEN BE PERFORMED.
THE SAME BASIC TAPE SUBROUTINES USED
IN THE RELIABILITY TEST ARE USED HERE,
LINKED TOGETHER BY THE COMMAND STRING
INTERPRETER.

5.5 GENERAL
THE HEART OF THE PROGRAM IS COMPOSED
OF SEVERAL TAPE HANDLING SUBROUTINES.
ERROR CHECKING IN THESE SUBROUTINES IS DESCRIBED
BELOW. ALL THE SUBROUTINES EXCEPT CHECKROUTINES
CONSULTATE BEFORE BEGINNING THE TOK
ROUTINE: CHECK OFFLINE AND REWINDING.

REWIND SUBROUTINE: REW

1. REWIND.
2. EXIT.

REWIND/OFFLINE SUBROUTINE: UNLD

1. REWIND.
2. GO OFFLINE.
3. EXIT.

ERASE SUBROUTINE: ERAS

1. IF WRITELOCK STATUS TERMINATE.
2. ERASE AND WAIT IN SKPBZ LOOP.
3. IF EOT SET SOFTWARE FLAG.
4. CHECK STATUS. IF DATA LATE, REWIND, ILLEGAL,
   BLOCKLENGTH OR PARITY PRINT AND EXIT.
5. EXIT.

READ SUBROUTINE: READ

1. CHECK STATUS AND STARTADDRESS.
2. IF DATA LATE,REWIND,ILLEGAL,BL.LENGTH OR
   PARITY STATUS, PRINT MESSAGE AND TERMINATE.
3. READ AND WAIT IN SKPBZ LOOP.
4. IF DATA LATE, REWIND, EOF, BL.LENGTH OR
   ILLEGAL, PRINT MESSAGE AND TERMINATE.
5. IF PARITY PRINT MESSAGE FIRST TIME ONLY
   AND RETRY UP TO 8 TIMES. DO NOT RETRY IN
   READ UNPACK MODE.
6. IF EOT SET SOFTWARE FLAG.
7. EXIT.

```
;       CHECK SUBROUTINE:  CHECK

;       1. CHECK ADDRESS COUNTER.
;       2. CHECK DATA IN INPUT BUFFER AGAINST DATA
;          THAT WAS WRITTEN.  PRINT MAXIMUM OF 3
;          DIFFERENCES IN EACH RECORD.
;          INPUTBUFFER=BUFF+10
;          SW4=1          RETRY READ UP TO 8 TIMES
;                         IN READ NORMAL MODE.
;                         ADDRESS IS ACTUAL.
;                         WORD# TOO, 1,2,3,...WC.
;          SW4=0          DO NOT RETRY READ, READ UNPACKED MODE,
;                         DO CONNECT CHARACTERS TO WORDS
;                         BEFORE CKECKING. THEREFORE ONLY
;                         EVERY SECOND ADDR AND WORD# ARE
;                         WRITTEN AS THERE CONTENTS ARE
;                         TWO WORDS PACKED.
;                         WORD#=CHAR# OF 2.CHAR
;                         ADDR.=ADDR. OF 2.CHAR
;                         THE WORDCOUNT USED IS (WRITEWC.2)+2
;                         FOR UNPACKED ROOM PLUS THE CRC,LPC.
;                         CHAR# COUNT 1,2,3,...WC.

;       3. CHECK THE FIRST WORD AFTER BLOCK IS UNTOUCHED,
;          (AFT=WORD), SHOULD BE ALL ZEROES.

;       4. ONLY IF SW4=0. NOW USE ACTUAL READ WC.

;       A. PE AND NRZI
;          COMPUTE PARITY BIT AND CHECK THE RECEIVED
;          PARITY BIT, ERRORS; THEN WRITE CHAR AND CHAR#.
;          CHECK ALSO THE SAME WAY THE E-BIT AND THAT NO
;          C-BIT OCCURS. ADDRESS IS ACTUAL.
;          CHAR# COUNT 1,2,3,...WC.

;       B. PE ONLY
;          CHECK FOR NO CHECKCHARS.

;       C. NRZI ONLY
;          COMPUTE LPC ALL 9 BIT, CHECK ALL 9 AND THE C AND
;          E-BIT. IF OK THEN THE C-BIT SHOULD BE 1 AND THE
;          E-BIT 0. IF NOT CHECK FCO 1130.

;       D. NRZI 9 TRACK ONLY
;          COMPUTE CRC THE SAME WAY AS THE LPC.

;       E. TAKE SOME ACTION FOR CRC OR LPC ZERO.

;       WRITE SUBROUTINE:  WRITE

;       1. CHECK STATUS AND STARTADDRESS.
;       2. IF DATA LATE, WRITE LOCK, ILLEGAL, REWIND,
;          BL.LENGTH OR PARITY, PRINT MESSAGE AND TERMINATE.
;       3. WRITE AND WAIT FOR INTERRUPT.
;       4. IF NO INTERRUPT PRINT MESSAGE AND TERMINATE.
;          IF INTERRUPT, CHECK DISABLE FLAG AND PRINT
;          MESSAGE IF IT FAILS.
;       5. CHECK STATUS. IF DATA LATE, REWIND, EOF,
;          BL.LENGTH OR ILLEGAL PRINT AND EXIT.
;       6. CHECK ADDRESS COUNTER.
;       7. IF PARITY ERROR PRINT MESSAGE ONCE AND RETRY
;          UP TO 8 TIMES.  IF NO-GO ERASE TAPE
;          AND TRY AGAIN.
;       8. IF EOT SET SOFTWARE FLAG.
;       9. EXIT.
```

SPACE SUBROUTINE: SPACE/SPEOF

1. SPACE # SPECIFIED OR TO EOF.
2. IF BOT TERMINATE.
3. IF SOFTWARE COUNTS DO NOT AGREE
   WITH #SPECIFIED, PRINT MESSAGE AND TERMINATE.
4. IF SOFTWARE COUNTS DO NOT AGREE
   WITH #REC IN FILE, SPACING A WHOLE
   FILE, PRINT MESSAGE AND TERMINATE.
5. IF DATA LATE, REWIND, ILLEGAL, BLOCKLENGTH OR
   PARITY PRINT MESSAGE.
6. IF EOT SET SOFTWARE FLAG.
7. EXIT.

READ EOF SUBROUTINE: RDEOF

1. CHECK STATUS AND STARTADDRESS.
2. IF DATA LATE, REWIND, ILLEGAL, BL.LENGTH OR
   PARITY STATUS, PRINT MESSAGE AND TERMINATE.
3. READ EOF AND WAIT IN SKPBZ LOOP.
4. CHECK STATUS. IF DATA LATE, REWIND, ILLEGAL OR
   BL.LENGTH PRINT MESSAGE AND TERMINATE.
5. IF PARITY STATUS PRINT MESSAGE FIRST TIME ONLY
   AND RETRY UP TO 8 TIMES. DO NOT RETRY IN READ
   UNPACK MODE.
6. IF EOT SET SOFTWARE FLAG.
7. EXIT.

CHECK EOF SUBROUTINE: CHEOF

1. CHECK EOF STATUS.
2. CHECK ADDRESSCOUNTER; PE = 0, NRZI NORMAL = 1,
   UNPAC = 2.
3. CHECK IN MEMORY 3 WORDS OF CORRECT DATA:
   FM OR NOTHING, LPC OR NOTHING, NOTHING.
   THE LPC ONLY IF READ UNPACK MODE.
4. EXIT.

WRITE EOF SUBROUTINE: WEOF

1. IF WRITELOCK STATUS TERMINATE.
2. WRITE EOF AND WAIT IN SKPBZ LOOP.
3. IF EOT SET SOFTWARE FLAG.
4. CHECK STATUS. IF DATA LATE, REWIND, ILLEGAL,
   BL.LENGTH OR PARITY, PRINT AND EXIT.
5. IF NO EOF STATUS TERMINATE.
6. EXIT.

6.SKEW PATTERN GENERATOR

GENERATES FOLLOWING DATA:
100200-100100-100040-100020-100010-100004-
100002-100001-040200-040100-040040-040020-
040010-040004-040002-040001-020200-020100....
AT TAPE IS WRITTEN:

```
10000000  X)     AFTER 8 X) MARKED WILL FOLLOW
10000000  Y)     THE NEXT SET OF 8 X) MARKED.
10000000  X)     THE Y) MARKED WILL SHIFT CYCLICALLY
01000000  Y)     1 TO THE RIGHT EVERYTIME. THE
10000000  X)     X) MARKED WILL SHIFT CYCLICALLY
00100000  Y)     1 TO THE RIGHT EVERY 8 Y) SHIFTS.
10000000  X)
00010000  Y)
10000000  X)
00001000  Y)
10000000  X) ...
```

```
;7.STATUS BIT TABLE

;        0        0

;        1        OFF-LINE
;        2        REWINDING
;        3        BOT

;        4        PE
;        5        WRITE LOCK
;        6        ILLEGAL

;        7        EOF
;        8        BLOCK LENGTH ERROR
;        9        DATA LATE

;        10       PARITY
;        11       EOT
;        12       ODD CHARACTER

;        13       7 TRACK (ERR. TRACK)
;        14       0
;        15       0

;8.BIT ALLOCATION IN DATA WORD:

;        BIT #    9 TRACK 9 TRACK UNPACKED

;        0        B0       E = PARITY ERROR

;        1        B1       C = CHECK CHARACTER ( CRC-LPC )
;        2        B2
;        3        B3
;1.CHAR                     B2-B6 ALLWAYS ZERO
;        4        B4
;        5        B5
;        6        B6

;        7        B7       P = PARITY BIT
;-----------------------------------------------
;        8        B0
;        9        B1

;        10       B2
;        11       B3       DATA BIT B0-B7
;        12       B4
;2.CHAR
;        13       B5
;        14       B6
;        15       B7

;                          7 TRACK BIT B0 AND B1 ALLWAYS ZERO.
```

```
;9.FILE AND BLOCKNUMBERING.
;       BE CAREFUL, THE NUMBERING IS ONLY SOFTWARE, NO
;       CHECKING IS POSSIBLE. IT IS RELIABLE UNTILL THE
;       TAPE MISS A BLOCK OR EOF. THEREFORE BY SOME ERRORS
;       THE PROGRAM SIMPLY TERMINATE BECAUSE THE INFORMATION
;       WILL OTHERWISE BE CONFUSING.

;       O       RECORD# FILE# (ONLY RELTEST)
;       O  BOT
;                  0        0
;       Y
;       Y  EOF
;                  1        1
;       X
;       X  DATA
;                  2        1
;       X
;       X  DATA
;                  3        1
;       Y
;       Y  EOF
;                  4        2
;       X
;       X  DATA
;                  5        2
;       X
;       X  DATA
;                  6        2
;       X
;       X  DATA
;                  7        2
;       X
;       X  DATA
;                  8        2
;       Y
;       Y  EOF
;                  9        3
;       X
;       X  DATA
;                  10       3
;       Y
;       Y  EOF
;                  11       4

;       ADDITIONALLY IN RELTEST FOLLOWING INFORMATION:
;       REC# IN FILE:   THE BLOCKS IN EACH FILE ARE
;                       NUMBERED 1,2,3...7.
;       # OF REC IN FILE: THE AMOUNT OF DATABLOCKS IN THAT FILE.
;
;       THESE 2 COUNTERS ARE ONLY MOVED IN READ AND WRITE
;       ROUTINES, THEREFORE IN OTHER MESSAGES E. G. FROM SPACE
;       ROUTINE THE INFORMATION ARE THE STATE BEFORE SPACE.
;
;       BUFFADDR:       THE LAST USED BUFFERADDRESS.
;                       FIRST NUMBER FOR WRITE
;                       LAST NUMBER FOR READ.
;       WORDCOUNT:      THE LAST USED WORDCOUNT (BL.LENGTH).
;                       FIRST NUMBER IF READ NORMAL
;                       LAST NUMBER IF READ UNPACKED.
```

;10.MESSAGES FROM THIS TEST.

```
;       ABOUT THE NUMBERS:
;       0-5     DIGITS IS A DECIMAL NUMBER
;               RANGE -32768 TO -1 AND 0 TO 32767.
;       6       DIGITS IS A OCTAL NUMBER.
;       8 OR 16 DIGITS IS A BINARY NUMBER.
;       10      DIGITS IS A DECIMAL POSITIVE NUMBER.

;       ABOUT THE TEXTS:
;       ALL TEXTSTRINGS ARE LISTED BELOW, MAY OCCURE IN
;       OTHER COMBINATIONS. IF A TAPE COMMAND ROUTINE
;       CAUSES MORE THAN ONE ERRORMESSAGE THE HEAD
;       MESSAGE IS ONLY PRINTED ONCE, EACH ERRORMESSAGE
;       THEN SEPARATED BY A NL.
;       MOST MESSAGES EXCEPT THE FOLLOWING ARE BEGINNING
;       WITH THE NAME OF THE ROUTINE CAUSING THE PRINT
;       FOLLOWED BY THE RECORD# (HEAD MESSAGE).

;       DISPLAY                 TTY/LPT

;  INT DEVICE000000     INTERRUPT FROM DEVICE 000000

;  ERROR TERMINATED     ERROR TERMINATED

;  TERMINATED           TERMINATED

;  INFOBITS CLEAR       INFOBITS CLEAR

;  RTC IS UNSTABLE,
;   SET CPUNO > AC2     RTC IS UNSTABLE, SET CPUNO > AC2

;  NO UNITS ONLINE      NO ONLINE UNITS AVAILABLE

;  ROTATE TAPES,KEY     ROTATE TAPES FROM UNIT 0 TO 1,
;                       1 TO 2, ETC. PRESS KEY TO CONTINUE

;  UNIT     1           UNIT          1           2
;  PAR WR     3         PAR WR        3           0
;  PAR RD     0         PAR RD        0           1
;  PERM WR    0         PERM WR       0           0
;  PERM RD    0         PERM RD       0           0
;  DATA ER    0         DATA ER       0           1
;  WD WR::::::3529      WD WR   ::::::3529    ::::::7917
;  WD RD::::::3529      WD RD   ::::::3529    ::::::7889
;  RECORD#   209        RECORD#     209           317
```

HERE FOLLOWS THE HEADED TEXTS:

DISPLAY	TTY/LPT
RTEST-UNIT 0	
RECORD# 209	RTEST-UNIT 0 RECORD# 209
WRITE EOT	WRITE EOT
READ EOT	READ EOT
WR EOF EOT	WR EOF EOT
RD EOF EOT	RD EOF EOT
	RTEST-UNIT 0 RECORD# 209
	FILE #: 5 REC# IN FILE: 4
	# OF REC IN FILE: 5
	BUFFADDR: 011157 011167
	WORDCOUNT: 000400 001002
	WRITE EOT
ITEST-UNIT 0	
RECORD# 209	ITEST-UNIT 0 RECORD# 209
LAST INTERCHANGE	LAST INTERCHANGE
BTEST-UNIT 0	
RECORD# 209	BTEST-UNIT 0 RECORD# 209
WRITE EOT	WRITE EOT
READ EOT	READ EOT
WR EOF EOT	WR EOF EOT
RD EOF EOT	RD EOF EOT
ERASE EOT	ERASE EOT
TESTLOOP,UNIT? 2	TESTLOOP,UNIT? 2
WC,MAX1777: 1777	WC,MAX1777: 1777
DATA XXYYYYYY	DATA XXYYYYYY
PARITY(90,91) 91	PARITY(90,91) 91
COMMAND STRING	COMMAND STRING
ILL. COM. STRING	ILLEGAL COMMAND STRING
REW-UNIT 0	
RECORD# 0	REW-UNIT 0 RECORD# 0
OFLIN-UNIT 0	
RECORD# 0	OFLIN-UNIT 0 RECORD# 0
ERASE-UNIT 0	
RECORD# 209	ERASE-UNIT 0 RECORD# 209
WRITE LOCK	WRITE LOCK
ERASE-UNIT 0	
RECORD# 209	ERASE-UNIT 0 RECORD# 209
024000 STATUS A.	024000 = STATUS AFTER COMMAND
021740 UNWANTED	021740 = NOT ALLOWED STATUSBITS
SPACE-UNIT 0	
RECORD# 209	SPACE-UNIT 0 RECORD# 209
COM SPACE- 6	COMMAND IS SPACE - 6
004040 STATUS A.	004040 = STATUS AFTER COMMAND
021340 UNWANTED	021340 = NOT ALLOWED STATUSBITS
SPACE-UNIT 0	
RECORD# 209	SPACE-UNIT 0 RECORD# 209
SPACE EOT	SPACE EOT
SPEOF-UNIT 0	
RECORD# 209	SPACE-UNIT 0 RECORD# 209
SPACE STOP BOT	SPACE STOP BOT
NO EOF STATUS	NO EOF STATUS

```
 SPACE-UNIT      0
  RECORD#     209      SPACE-UNIT      0 RECORD#     209
 SPACE STOP EOF       SPACE STOP EOF

 COM SPACE-      6     COMMAND IS SPACE =      6
 # SPACED:       5     RECORDS DETECTED =      5
 # IN FILE:      7     RECORDS INCL EOF =      7

 WRITE-UNIT      0
  RECORD#     209      WRITE-UNIT      0 RECORD#     209
  WRITE LOCK            WRITE LOCK

 WRITE-UNIT      0
  RECORD#     209      WRITE-UNIT      0 RECORD#     209
 000200 STATUS C       000200  = STATUS AFTER C PULSE
 021340 UNWANTED       021340  = NOT ALLOWED STATUSBITS
 UNIT NOT STARTED      UNIT NOT STARTED

 WRITE-UNIT      0
  RECORD#     209      WRITE-UNIT      0 RECORD#     209
 000100 STATUS A.      000100  = STATUS AFTER COMMAND
 021700 UNWANTED       021700  = NOT ALLOWED STATUSBITS
 NO REC# COUNT         NO REC# COUNT

 WRITE-UNIT      0
  RECORD#     209      WRITE-UNIT      0 RECORD#     209
 012556 ADDR SENT      012556   ADDR SENT
 112556 ADDR RCVD      112556   ADDR RCVD REGISTER LOAD

 WRITE-UNIT      0
  RECORD#     209      WRITE-UNIT      0 RECORD#     209
 ADDRCOUNT 012704      ADDRCOUNT 012704  BAD: 012714
   BAD: 012714         BUFF-ADDR PLUS WC DO NOT EQ. DIB

 WRITE-UNIT      0
  RECORD#     209      WRITE-UNIT      0 RECORD#     209
  PARITY STATUS         PARITY STATUS

 WRITE-UNIT      0
  RECORD#     209      WRITE-UNIT      0 RECORD#     209
 MISSED INTERRUPT      MISSED INTERRUPT ON WRITE

 WRITE-UNIT      0
  RECORD#     209      WRITE-UNIT      0 RECORD#     209
 INT. BUT DISABLE      INTERRUPT AFTER DISABLE FLAG SET

 READ-UNIT       0
  RECORD#     209      READ-UNIT       0  RECORD#     209
 012556 ADDR SENT      012556   ADDR SENT
 112556 ADDR RCVD      112556   ADDR RCVD REGISTER LOAD

 READ-UNIT       0
  RECORD#     209      READ-UNIT       0  RECORD#     209
 000200 STATUS C       000200  = STATUS AFTER C PULSE
 021340 UNWANTED       021340  = NOT ALLOWED STATUSBITS

 READ-UNIT       0
  RECORD#     209      READ-UNIT       0  RECORD#     209
 000100 STATUS A.      000100  = STATUS AFTER COMMAND
 021700 UNWANTED       021700  = NOT ALLOWED STATUSBITS

 READ-UNIT       0
  RECORD#     209      READ-UNIT       0  RECORD#     209
  PARITY STATUS         PARITY STATUS
```

```
READ-UNIT      0
 RECORD#     209
READ UNPAC MODE
ADDRCOUNT 015066
 BAD: 015166
```

```
READ-UNIT      0  RECORD#     209
READ UNPAC MODE
ADDRCOUNT 015066  BAD: 015166
BUFF-ADDR PLUS WC DO NOT EQ. DIB
```

```
READ-UNIT      0
 RECORD#     209
READ UNPAC MODE
GOOD  BAD  ADDR  W#
127315  127314
014622  000002
145277  145276
014624  000004
026540  026541
014626  000006
```

```
READ-UNIT      0  RECORD#     209
READ UNPAC MODE
GOOD    BAD     ADDR    WORD#
127315  127314  014622  000002
145277  145276  014624  000004
026540  026541  014626  000006
```

```
READ-UNIT      0
 RECORD#     209
READBIT 11001000
11001100 & BYTE
EXP.BIT 00000001
 CHAR#  000001
  ADDR: 014622
ERROR IN PARITY
 C,E BIT BIT 2-6
```

```
READ-UNIT      0  RECORD#     209
READBIT 1100100011001100 & BYTE
EXP.BIT 00000001 CHAR#  000001
  ADDR: 014622
ERROR IN PARITY, C,E BIT BIT 2-6
```

```
READ-UNIT      0
 RECORD#     209
 CRC-WORD 000000
  BAD:  234123
 CHAR#     000123
  ADDR: 144625
 LPC-WORD 000000
  BAD:  066555
 CHAR#     000124
  ADDR: 144626
PE EXP. NO CHECK
```

```
READ-UNIT      0  RECORD#     209
 CRC-WORD 000000  BAD:  234123
 CHAR#     000123  ADDR: 144625
 LPC-WORD 000000  BAD:  066555
 CHAR#     000124  ADDR: 144626
PE EXP. NO CHECK CHARACTER HERE
```

```
READ-UNIT      0
 RECORD#     209
 LPC-WORD 040224
  BAD:  177777
 CHAR#     000144
  ADDR: 133556
LPC-ERROR, CRC=0
LPC-ERROR
```

```
READ-UNIT      0  RECORD#     209
 LPC-WORD 040224  BAD:  177777
 CHAR#     000144  ADDR: 133556
ERROR IN CHECKCHAR LPC
CRC EXPECTED ALL ZEROES
```

```
READ-UNIT      0
 RECORD#     209
 CRC-WORD 040255
  BAD:  071270
 CHAR#     000440
  ADDR: 017044
CRC-ERROR
```

```
READ-UNIT      0  RECORD#     209
 CRC-WORD 040255  BAD:  071270
 CHAR#     000440  ADDR: 017044
ERROR IN CHECKCHAR CRC
```

```
READ-UNIT      0
 RECORD#     209
 LPC-WORD 040634
  BAD:  000000
 CHAR#     000122
  ADDR: 015333
 NO LPC FOUND
```

```
READ-UNIT      0  RECORD#     209
 LPC-WORD 040634  BAD:  000000
 CHAR#     000122  ADDR: 015333
NO LPC FOUND
```

```
;  READ-UNIT      0
;   RECORD#     209        READ-UNIT      0  RECORD#     209
;   CRC-WORD 040333        CRC-WORD 040333  BAD:  000333
;    BAD:  000333
;   CHAR#     000245        CHAR#     000245  ADDR: 014553
;    ADDR: 014553
;   NO CRC FOUND           NO CRC FOUND

;  READ-UNIT      0
;   RECORD#     209        READ-UNIT      0  RECORD#     209
;   AFT-WORD 000000        AFT-WORD 000000  BAD:  015372
;    BAD:  015372
;   CHAR#     000246        CHAR#     000246  ADDR: 014554
;    ADDR: 014554
;   BLOCK TOO LONG         BLOCK TOO LONG

;  WFILE-UNIT     0
;   RECORD#     209        WFILE-UNIT      0 RECORD#     209
;   WRITE LOCK              WRITE LOCK

;  WFILE-UNIT     0
;   RECORD#     209        WFILE-UNIT      0 RECORD#     209
;  000040 STATUS A.        000040  = STATUS AFTER COMMAND
;  021340 UNWANTED         021340  = NOT ALLOWED STATUSBITS

;  WFILE-UNIT     0
;   RECORD#     209        WFILE-UNIT      0 RECORD#     209
;   NO EOF STATUS           NO EOF STATUS

;  EOF-UNIT       0
;   RECORD#     209        EOF-UNIT       0   RECORD#     209
;  000200 STATUS C         000200  = STATUS AFTER C PULSE
;  021340 UNWANTED         021340  = NOT ALLOWED STATUSBITS

;  EOF-UNIT       0
;   RECORD#     209        EOF-UNIT       0   RECORD#     209
;  014435 ADDR SENT        014435  ADDR SENT
;  114435 ADDR RCVD        114435  ADDR RCVD REGISTER LOAD

;  EOF-UNIT       0
;   RECORD#     209        EOF-UNIT       0   RECORD#     209
;  000600 STATUS A.        000600  = STATUS AFTER COMMAND
;  021300 UNWANTED         021300  = NOT ALLOWED STATUSBITS

;  EOF-UNIT       0
;   RECORD#     209        EOF-UNIT       0   RECORD#     209
;   PARITY STATUS           PARITY STATUS

;  EOF-UNIT       0
;   RECORD#     209        EOF-UNIT       0   RECORD#     209
;   NO EOF STATUS           NO EOF STATUS

;  EOF-UNIT       0
;   RECORD#     209        EOF-UNIT       0   RECORD#     209
;  ADDRCOUNT 017066        ADDRCOUNT 017066  BAD: 017066
;    BAD: 017066
;    1.WORD  000023          1.WORD  000023  BAD: 000123
;    BAD: 000123
;    2.WORD  040023          2.WORD  040023  BAD: 040123
;    BAD: 040123
;    3.WORD  000000          3.WORD  000000  BAD: 000000
;    BAD: 000000


;TAPE 1

.EOT
```