LA36

TERMINAL (DH11 & DJ11)

MD-11-DZLAD-C

EP-DZLAD-C-DL-A

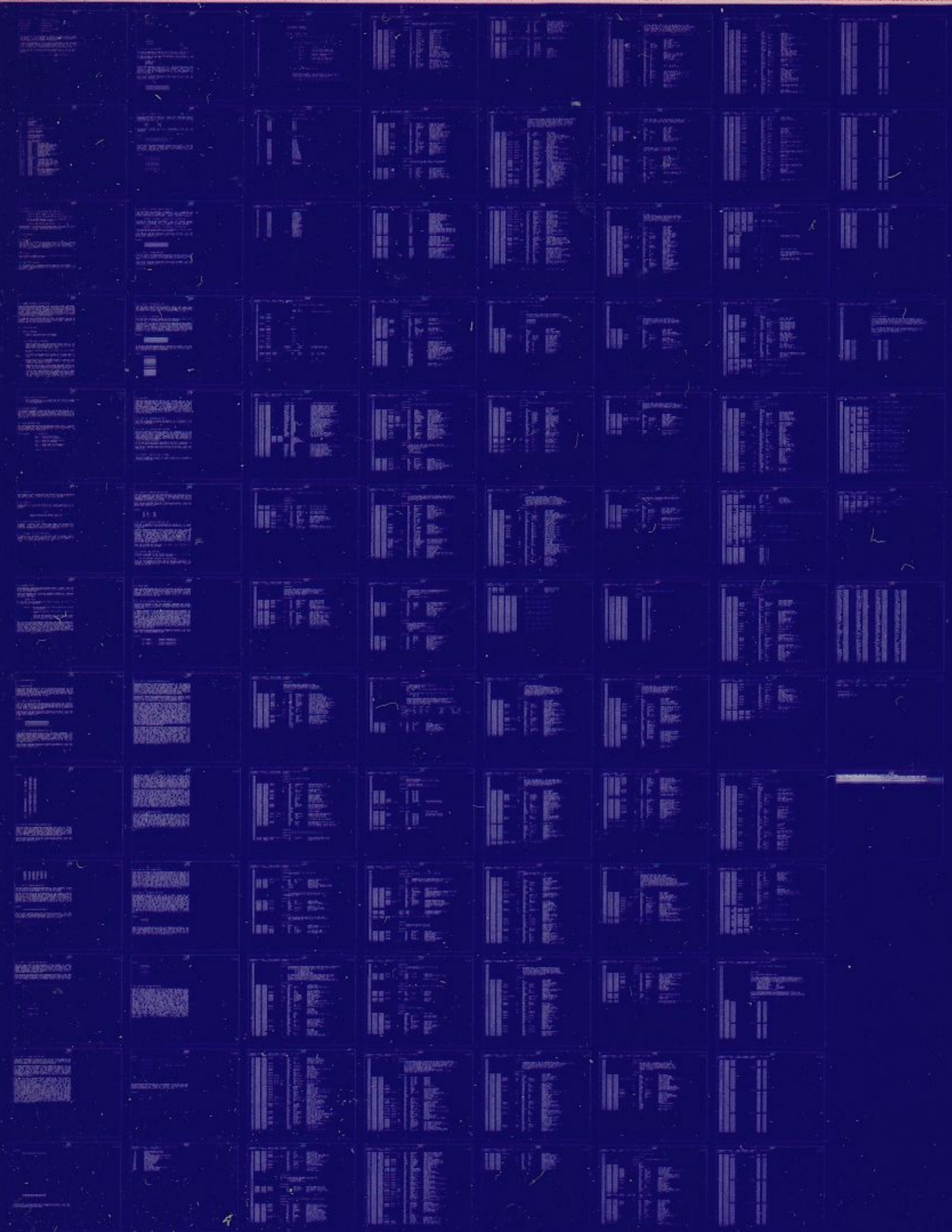
OCT 1976

COPYRIGHT ©1976

digital

FICHE 1 OF 1

Made In U.S.A.



IDENTIFICATION

SEC 000:

PROGRAM CODE: MAINDEC-11-DZLAC-C-D

PROGRAM NAME: LA36 TERMINAL DH11 & DZ11 INTERFACE

DATE CREATED: SEPTEMBER 1975

MAINTAINER: DIAGNOSTIC GROUP

AUTHOR: ROBERT BAKER

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

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED TO THE PURCHASER UNDER A LICENSE FOR USE ON A SINGLE COMPUTER SYSTEM AND CAN BE COPIED (WITH INCLUSION OF DIGITAL'S COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT AS MAY OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT © 1975, 1976, BY DIGITAL EQUIPMENT CORPORATION

TABLE OF CONTENTS

1.0 ABSTRACT

2.0 REQUIREMENTS
2.1 HARDWARE
2.2 SOFTWARE
2.3 PRELIMINARY PROGRAMS

3.0 LOADING PROCEDURE & INITIALIZATION

4.0 STARTING PROCEDURE
4.1 LOADING PROCESS

5.0 OPERATING PROCEDURE
5.1 DATA REGISTER CONTROL
5.2 KEYBOARD CONTROL

6.0 TEST DESCRIPTIONS
6.1 PRELIMINARY TESTS
6.1.1 - DATA PATH TEST
6.1.2 - PRINTABLE CHARACTER TEST
6.1.3 - NON-PRINTABLE CHARACTER TEST
6.1.4 - CARRIAGE RETURN TEST
6.1.5 - MULTIPLE LINE FEED TEST
6.1.6 - SINGLE LINE FEED TEST
6.1.7 - BACKSPACE TEST
6.1.8 - OVERPRINT TEST
6.1.9 - PRINTING FREQUENCY SWEEP TEST
6.1.10 - PRINTER BELL TEST
6.1.11 - TAPE TEST
6.2 ECHO TESTS
6.2.1 - CHARACTER ECHO TEST
6.2.2 - LINE ECHO TEST, FAST RATE
6.2.3 - LINE ECHO TEST, SLOW RATE
6.2.4 - CHARACTER/LINE ECHO TEST
6.2.5 - SELECTED PATTERN ECHO TEST
6.2.6 - BELL ECHO TEST
6.3 OPTIONS
6.3.1 - SECONDARY CHARACTER SET OPTION
6.3.2 - SELECTIVE ADDRESSING OPTION
6.3.3 - [illegible] OPTION
6.3.4 - [illegible] OPTION
6.3.5 - [illegible] OPTION
6.3.6 - [illegible] OPTION

1.0 ABSTRACT

THIS DIAGNOSTIC IS DIVIDED INTO FOUR BASIC SECTIONS:

1. A CHECK OF THE CONSOLE TERMINAL INTERFACE LOGIC
2. A CHECK OF THE PRINTING CHARACTERISTICS AND CONTROL LOGIC
3. AN ECHO PORTION DESIGNED TO CHECK THE KEYBOARD AND TO AID IN THE DIAGNOSIS OF TERMINAL PROBLEMS.
4. A CHECK OF THE VARIOUS LA36 OPTIONS.

PATTERNS USED BY THE PRINTING TESTS WERE CHOSEN FOR EASE OF VISUAL VERIFICATION. THE ECHO TESTS WERE DESIGNED FOR MAXIMUM FLEXIBILITY, WITH THE USER ALLOWING ANY DESIRED PATTERN TO BE USED.

2.0 REQUIREMENTS

2.1 EQUIPMENT

THE DIAGNOSTIC IS WRITTEN TO RUN ON ALL MODELS OF THE PDP-11 COMPUTER WITH EITHER DH11 OR DZ11 MULTIPLEXER. UP TO 16 MULTIPLEXERS + 256 TERMINALS ARE OPTIONAL. THE DIAGNOSTIC IS SET TO TEST THE TERMINALS AT 300 BAUD. TO CHANGE ANY PARAMETERS FOR THE CHAR INTERFACE REFER TO THE CHAR PARAMETER TABLE IN THE LISTING.

2.2 STORAGE

THE DIAGNOSTIC PROGRAM USES ALL OF 4K OF MEMORY WITH THE EXCEPTION OF THE AREA USED BY THE ABSOLUTE LOADER.

2.3 PRELIMINARY PROGRAMS

ALL APPLICABLE PDP-11 DIAGNOSTICS SHOULD BE RUN ON THE PROCESSOR. IF ERRORS ARE ENCOUNTERED DURING THE INTERFACE CHECK, REFER TO THE PROCESSOR INTERFACE DIAGNOSTIC FOR FURTHER HELP IN LOCATING THE PROBLEM IF NEEDED.

3.0 LOADING PROCEDURE & INITIALIZATION

LOAD THE CASS DIAGNOSTIC PROGRAM TAPE FOLLOWING NORMAL PROCEDURES. BEFORE STARTING THE PROGRAM, REFER TO THE EXISTING LINE TABLE (ELTAB) AND CLEAR THE PROPER BITS IN THE TABLE TO INDICATE WHICH TERMINALS ARE TO BE TESTED. A DETAILED DESCRIPTION IS CONTAINED IN THE PROGRAM LISTING. ALSO, REFER TO THE DESCRIPTION OF THE ROUTINE "DLY". TIME DELAYS USED BY THE PROGRAM ARE A FUNCTION OF THE CPU MODEL AND MEMORY TYPE AND SHOULD BE SET UP BEFORE RUNNING THE DIAGNOSTIC. THE ROUTINE IS PRESET FOR A PDP-11 40 WITH CORE MEMORY.

IF A HARDWARE SWITCH REGISTER DOES NOT EXIST, THE PROGRAM WILL USE THE CONTENTS OF LOCATION 176 AS THE VALUE OF THE SWITCHES. THEREFORE, BE SURE TO LOAD LOCATION 176 WITH THE SWITCH VALUE BEFORE STARTING THE PROGRAM WHEN NOT USING HARDWARE SWITCHES.

4.0 STARTING PROCEDURE

4.1 STARTING ADDRESSES

200.01 = EXECUTE WITH DH11 MULTIPLEXER
204.01 = EXECUTE WITH DZ11 MULTIPLEXER

4.1.1 EXECUTE WITH DH11 MULTIPLEXER

A. REFER TO SECTION 3.0 AND MAKE SURE THE PROPER BITS IN THE ELTAB TABLE INDICATING WHAT TERMINALS ARE TO BE TESTED HAVE BEEN CLEARED AND THE CORRECT DELAY COUNT FOR THE CPU AND MEMORY TYPE IN USE HAS BEEN SET IN TIMER.

B. SET SWITCH REGISTER = 200.01 AND PRESS THE LOAD ADDRESS SWITCH

C. SET THE SWITCH REGISTER BITS 7-0 EQUAL TO THE PAPER WIDTH IN TERMS OF THE NUMBER OF COLUMNS (OCTAL). REFER TO SECTION 5.1.4

D. SET SWITCH 8 UP IF IT IS DESIRED TO SELECT A SPECIFIC TEST RATHER THAN BEGIN THE NORMAL PRINTING TEST SEQUENCE. OTHERWISE, LEAVE SWITCH 8 DOWN.

E. PRESS THE START SWITCH. IF BIT 9 WERE ZERO WHEN STARTING THE NUMBER OF DH11'S UNDER TEST WILL BE PRINTED ON ALL EXISTING TERMINALS AND THE PRINTER TESTS ARE EXECUTED SEQUENTIALLY.

F. IF BIT 9 WERE 1 WHEN STARTING, THE NUMBER OF DH11'S UNDER TEST WILL BE INDICATED AND THE MESSAGE "SELECT TEST NUMBER" WILL BE PRINTED ON ALL EXISTING TERMINALS. THE PROGRAM WILL THEN BE WAITING FOR A TEST SELECTION VIA ANY TERMINAL KEYBOARD IF SWITCH 10 IS DOWN. REFER TO SECTION 5.2

7.1.2 EXECUTE WITH DZ11 MULTIPLEXER

A. SAME INSTRUCTIONS AS 7.1.1 EXCEPT THAT THE STARTING ADDRESS IN S IS 204 AND THE FIRST MESSAGE PRINTED WILL BE THE NUMBER OF DZ11'S UNDER TEST.

5.0 OPERATING PROCEDURE

THE PROGRAM IS GENERALLY CONTROLLED FROM A MULTIPLEXER TERMINAL, BUT A FEW SWITCH REGISTER CONTROLS ARE AVAILABLE. THE PRINTER TEST WILL OUTPUT TO ALL TERMINALS OR TO THE ONE UNDER TEST AS A FUNCTION OF SR BIT 13. ECHO TESTS WILL REFERENCE ONLY THE TERMINAL SELECTING THE TEST OR ALL TERMINALS DEPENDING ON THE SPECIFIC TEST AND THE SETTING OF SWITCH 13.

5.1 SWITCH REGISTER CONTROL

THE VARIOUS SWITCHES AND THEIR FUNCTIONS ARE LISTED BELOW. SWITCHES MAY BE CHANGED AND SET AS DESIRED EXCEPT AS NOTED IN THE SPECIFIC SWITCH DESCRIPTIONS. REFER TO THE DETAILED SWITCH DESCRIPTIONS FOR FURTHER, MORE COMPLETE INFORMATION.

SWITCH NUMBER	DESCRIPTION
15	UP = HALT AT END OF TEST DOWN = CONTINUE TEST SEQUENCE
13	UP = DRIVE ONLY SELECTED TERMINAL DOWN = DRIVE ALL TERMINALS
5	UP = SELECT TEST (AT START-UP ONLY) DOWN = START NORMAL TEST SEQUENCE
7-0	NUMBER OF COLUMNS AT START-UP

5.1.1 SWITCH 15

WITH SWITCH 15 IN THE UP POSITION, THE PROGRAM WILL HALT AT THE END OF THE CURRENT TEST. REPLACING SWITCH 15 TO THE DOWN POSITION AND PRESSING CONTINUE WILL CONTINUE THE NORMAL TEST OPERATION.

5.1.2 SWITCH 13

PLACING SWITCH 13 IN THE DOWN POSITION WILL CAUSE THE DRIVING OF ALL TERMINALS. IF SWITCH 13 IS UP, ONLY THE TERMINAL UNDER TEST IS DRIVEN.

NOTE

SWITCH 13 CAN ONLY BE CHANGED WHEN THE PROGRAM IS WAITING FOR A TEST SELECTION.

5.1.3 SWITCH 9 (AT START-UP ONLY)

TO SELECT A SPECIFIC TEST RATHER THAN START THE PRINTING TEST SEQUENCE, PLACE SWITCH 9 UP BEFORE STARTING THE DIAGNOSTIC. OTHERWISE, LEAVE SWITCH 9 DOWN TO RUN THE NORMAL TEST SEQUENCE. THIS SWITCH IS ONLY EFFECTIVE AT START-UP OF THE PROGRAM.

5.1.4 SWITCHES 7 TO 0 (AT START-UP ONLY)

AT START-UP ONLY, SWITCHES 7 TO 0 ARE USED TO SET THE DESIRED MAXIMUM NUMBER OF RETURNS THE DIAGNOSTIC IS TO TEST. IF THE NUMBER SET IS BETWEEN ZERO AND 255 (OR LESS THAN 30(10)), THE PROGRAM WILL DEFAULT TO 10. THE VALUE SET MUST BE IN TOTAL FORM.

5.2 KEYBOARD CONTROL

THE PROGRAM WILL ALWAYS BE UNDER KEYBOARD CONTROL. CONTROL FROM THE SWITCH REGISTER DURING PROGRAM EXECUTION IS ONLY POSSIBLE WITH SWITCHS 13 AND 15 AS STATED ABOVE.

TYPING THE "RUBOUT" (DEL) KEY ON ANY TERMINAL KEYBOARD WILL TERMINATE THE TEST IMMEDIATELY. AFTER TERMINATION OF THE TEST THE FOLLOWING MESSAGE WILL BE TYPED:

SELECT TEST #

AT THIS TIME, TYPE THE DESIRED TEST NUMBER FOLLOWED BY ANY ONE OF THE FOLLOWING CONTROL CHARACTERS:

. (PERIOD) = RUN THE SELECTED TEST ONCE AND RETURN FOR ANOTHER TEST SELECTION.

L = LOOP ON THE SELECTED TEST UNTIL A "RUBOUT" IS TYPED.

S = START THE TEST SEQUENCE WITH THE SELECTED TEST. CONTINUE TO LOOP ON THE PRINTING TEST SEQUENCE UNTIL A "RUBOUT" IS TYPED.

THE L OR S MAY BE EITHER UPPER OF LOWER CASE, BUT THE TEST NUMBER MUST ALWAYS BE A 2 DIGIT OCTAL NUMBER. FOR ALL ECHO TESTS, THE "L" AND "S" WILL ONLY RUN THE TEST ONCE (THE SAME AS IF TYPING A PERIOD). FOR ALL OPERATION TESTS, THE "S" WILL ONLY RUN THE TEST ONCE (THE SAME AS IF TYPING A PERIOD), HOWEVER, TYPING AN "L" WILL CAUSE THE PROGRAM TO LOOP ON THE SELECTED TEST. IF AN ERROR IS DETECTED IN THE TEST SELECTION (ILLEGAL TEST NUMBER OR CONTROL CHARACTER) A QUESTION MARK IS PRINTED AND THE MESSAGE WILL BE REPEATED.

6.0 TEST DESCRIPTIONS

6.1 PRINTING TESTS

THESE TESTS ARE DESIGNED AS A TEST OF THE PRINTING MECHANISM AND THE ASSOCIATED CONTROL LOGIC. AT THE BEGINNING OF EACH TEST, THE TEST NUMBER WILL BE PRINTED INDICATING WHICH TEST IS BEING EXECUTED AND, IF THE TEST IS A FUNCTION OF THE NUMBER OF COLUMNS, THE NUMBER OF COLUMNS BEING TESTED WILL BE INDICATED. A DETAILED DESCRIPTION AND SAMPLE PATTERNS FOR EACH PRINTING TEST FOLLOWS:

6.1.1 TEST 0 - DATA PATH TEST

THIS TEST IS USED TO TEST THE DATA LINES TO AND THROUGH THE INTERFACE AND TO THE TERMINAL. AN ALTERNATING BIT PATTERN IS SENT WHICH WILL PRINT ALTERNATING *'S AND U'S IN A CHECKERBOARD PATTERN TO THE MAXIMUM COLUMN WIDTH. THE STARTING CHARACTER FOR EACH LINE IS ALTERNATED AND A TOTAL OF FOUR LINES ARE PRINTED.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:

```
*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U
U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*
*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U
U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*
```

6.1.2 TEST 1 - PRINTABLE CHARACTER TEST

THIS TEST PRODUCES A CHECK OF ALL 94(10) PRINTABLE CHARACTERS. THE CHARACTERS ARE PRINTED IN GROUPS OF THREE WITH THREE GROUPS PER LINE, SEPARATED BY THREE SPACES BETWEEN GROUPS. THE FIRST COLUMN WILL CONTAIN ALL ASCII CODES FROM 040 TO 077. COLUMN TWO WILL CONTAIN ALL ASCII CODES FROM 100 TO 137 - PRIMARILY THE CAPITAL LETTER SET. THE LAST COLUMN WILL CONTAIN ALL ASCII CODES FROM 140 TO 176 - PRIMARILY THE SMALL LETTER SET.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:

```
         @@@   ```
   !!!   AAA   AAA
   """   BBB   BBB
   ###   CCC   CCC
   $$$   DDD   DDD
   %%%   EEE   EEE
   &&&   FFF   FFF
   '''   GGG   GGG
   (((   HHH   HHH
   )))   III   III
   ***   JJJ   JJJ
   +++   KKK   KKK
   ,,,   LLL   LLL
   ---   MMM   MMM
   ...   NNN   NNN
   ///   OOO   OOO
   000   PPP   PPP
   111   QQQ   QQQ
   222   RRR   RRR
   333   SSS   SSS
   444   TTT   TTT
   555   UUU   UUU
   666   VVV   VVV
   777   WWW   WWW
   888   XXX   XXX
   999   YYY   YYY
   :::   ZZZ   ZZZ
   ;;;   [[[
   <<<   \\\
   ===   ]]]
   >>>
   ???
```

6.1.3 TEST 2 - NON-PRINTABLE CHARACTER TEST

THIS TEST CHECKS ALL NON-PRINTABLE CHARACTERS THAT HAVE NO CONTROL FUNCTION IN THE LA36 TERMINAL OR THE LA36 OPTIONS (SUCH AS CR, LF, BS, & BEL). FIRST THE ASCII CODE WILL BE PRINTED FOLLOWED BY THE MNEMONIC AFTER A FEW SEPARATING SPACES. FOLLOWING THE MNEMONIC, THE ACTUAL CONTROL CHARACTER WILL BE SENT THREE TIMES AND NOTHING SHOULD HAPPEN AT THE PRINTER. THIS PATTERN IS REPEATED, THREE TIMES ON A LINE, UNTIL ALL OF THE NON-PRINTING CHARACTERS HAVE BEEN TESTED.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:

000	NUL	001	SOH	002	STX
005	ACK	020	DLE	021	DC1
022	DC2	023	DC3	024	DC4
025	NAK	026	SYN	027	ETB
030	CAN	031	EM	032	SUB
034	FS	035	GS	036	RS
037	US	177	DEL		

6.1.4 TEST 3 - CARRIAGE RETURN TEST

THIS TEST CHECKS THE CARRIAGE RETURN FROM ALL EVEN NUMBERED COLUMNS AND THE SPACING OF THE SOLENOID HEAD FROM THE LEFT MARGIN. IT IS ALSO A GOOD CHECK FOR PROPER OPERATION OF THE POSITION DECODER.

THE TEST PRINTS A FULL LINE OF ALTERNATING 0'S AND SPACES, STARTING WITH A 0. AT THE END OF THE LINE THE PRINT HEAD IS RETURNED TO THE LEFT MARGIN WITH A CARRIAGE RETURN. THE SPACES ARE THEN FILLED IN BY SPACING THE PRINT HEAD OUT FROM THE LEFT MARGIN TO THE FIRST SPACE, PRINTING AN "X", AND EXECUTING A CARRIAGE RETURN. THIS PATTERN IS REPEATED UNTIL THE LINE IS COMPLETED. CHECK TO SEE THAT ALL X'S ARE IN THE MIDDLE OF THE SPACE BETWEEN THE TWO ZEROES ON EITHER SIDE OF IT.

EXAMPLE:

```
0X0X0X0X0X0X0X0X0X0X0X0X0X0X0X
```

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THIS TEST WILL PRINT A LINE OF 0'S AND SPACES, THEN PRINT A DIAGINAL LINE OF X'S. TO CORRECTLY CHECK THE ENCODER, THE AUTO LINE FEED OPTION SHOULD BE DISABLED.

EXAMPLE:

```
0 0 0 0 0 0 0 0 0
 X
   X
     X
       X
         X
           X
             X
               X
```

6.1.5 TEST 4 - MULTIPLE LINE FEED TEST

THIS TEST CHECKS THE LINE FEED CAPABILITY OF THE PRINTER BY SENDING VARIOUS GROUPS OF LINE FEEDS INTERSPACED WITH REFERENCE LINES. THE NUMBER PRINTED AS THE REFERENCE LINE INDICATES THE NUMBER OF LINE FEEDS THAT FOLLOW. THE FIRST AND LAST LINES ALSO CONTAIN A STRING OF DASHES AS REFERENCE POINTS FOR MEASURING. THE TOTAL DISTANCE IS 63(10) LINES BETWEEN THE TWO DASHED LINES.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY CARRIAGE RETURN, THE NUMBER PRINTED WILL INDICATE ONE LESS THAN THE NUMBER OF LINE FEEDS (THE NUMBER OF BLANK LINES) THAT FOLLOW. THE TOTAL DISTANCE BETWEEN THE TWO DASHED LINES WILL THEN BE 69 LINES.

EXAMPLE:

```
01----------------------------------------------
02

04


05


16
    15 BLANK LINES
32
    31 BLANK LINES
33----------------------------------------------
```

5.1.5 TEST 5 - SINGLE LINE FEED TEST

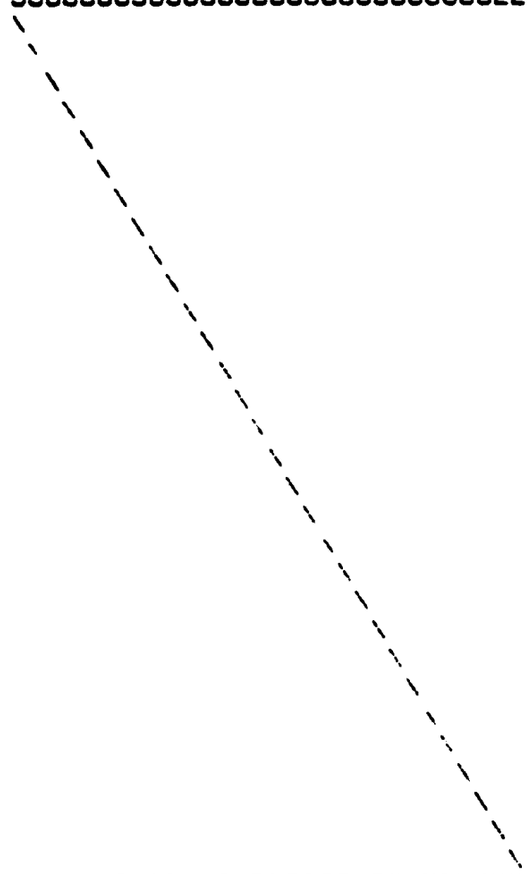
THIS TEST IS DESIGNED TO CHECK THE TIMING OF SINGLE LINE FEEDS AND THE CAPABILITY OF DOING LINE FEEDS IN ALL COLUMNS. TWO REFERENCE LINES ARE USED BY THIS TEST (AND TEST 6) WHICH ALSO CAN BE USED TO EASILY CHECK THE NUMBER OF COLUMNS THE PRINTER IS PRINTING.

THE FIRST REFERENCE LINE CONTAINS 130(10) ZEROES FOLLOWED BY TWO 2'S IF TESTING 132(10) COLUMNS. IF LESS THAN 132 COLUMNS, THE LINE WILL CONTAIN 0'S FOR TWO LESS THAN THE MAXIMUM NUMBER OF COLUMNS FOLLOWED BY THE TWO 2'S. THIS REFERENCE LINE IS A QUICK CHECK FOR 132(10) COLUMNS IF TESTING THE FULL 132(10) COLUMNS. THE SECOND REFERENCE LINE PRINTS A STRING OF NUMBERS (1 TO 9 & 0) REPEATED TO THE MAXIMUM COLUMN. THIS LINE, AGAIN, CAN BE USED AS A QUICK CHECK OF THE NUMBER OF COLUMNS.

THE LINE FEED TEST IS ACCOMPLISHED BY: PRINTING THE FIRST REFERENCE LINE OF 0'S AND TWO 2'S; THEN EITHER SENDING 60(10) 3'S, IF TESTING 132(10) COLUMNS, OR WAITING 1.8 SECONDS FOR AN LCV, IF TESTING LESS THAN 132(10) COLUMNS. IF TESTING 132(10) COLUMNS, NOTHING SHOULD HAPPEN, EXCEPT FOR AN LCV, AT THE END OF THE LINE. THE 3'S SHOULD BE LOST AND NEVER PRINTED. AFTER THE LCV, WITH THE PRINT HEAD AT THE EXTREME RIGHT, A CARRIAGE RETURN - LINE FEED WILL BE SENT FOLLOWED BY REPEATED BACKSLASHES "\" AND LINEFEEDS TO PRINT A DIAGONAL LINE DOWN THE PAPER. WHEN A BACKSLASH IS PRINTED IN THE MAXIMUM COLUMN, A CARRIAGE RETURN WILL BE SENT IMMEDIATELY AFTER THE LINE FEED AND THE SECOND REFERENCE LINE OF SEQUENTIAL NUMBERS WILL BE PRINTED. AFTER COMPLETING THE LINE, A CARRIAGE RETURN - LINE FEED WILL BE SENT AND THE PROGRAM WILL WAIT ONE SECOND FOR THE CARRIAGE RETURN FUNCTION TO COMPLETE. AFTER THE DELAY, THE REFERENCE LINE WILL BE REPEATED, THE LAST LINE BEING GUARANTEED TO BE CORRECT. ANY TIMING PROBLEMS DURING THE LINE FEEDS WILL SHOW AS MISS PRINTS OR MISSING CHARACTERS DURING THE FIRST 15(10) CHARACTERS OF THE MIDDLE REFERENCE LINE. ALSO, ANY PAPER FEED PROBLEMS WILL CAUSE MISS-ALIGNMENT OF THE SLASHES FORMING THE DIAGONAL LINE.

EXAMPLE:

```
0000000000000000000000000000022
```



```
12345678901234567890123456789 0
12345678901234567890123456789 0
```

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE EVERY PLACE A CARRIAGE RETURN IS EXECUTED.

EXAMPLE:

```
0000000022

\
 \
  \
   \
    \
     \
      \
       \
        \

1234567890

1234567890
```

6.1.7 TEST 6 - BACKSPACE TEST

THIS TEST IS DESIGNED TO TEST THE PRINT TIMING AS IN TEST 5 AS WELL AS THE BACKWARD AND FORWARD MOVEMENT OF THE PRINT SOLENOID HEAD.

THE TEST CONSISTS OF THE SAME FIRST REFERENCE LINE AS IN TEST 5, THEN A CARRIAGE RETURN-LINE FEED. A FULL LINE IS THEN PRINTED USING THE FOLLOWING PATTERN:

```
FORWARD SLASH    "/"
BACKSPACE
BACK SLASH       "\"
```

THIS PATTERN PRODUCES A LINE OF ALL X'S. THE TWO SLASHES SHOULD CROSS EXACTLY AT THE MIDDLE, PRODUCING THE X CHARACTER. WHEN THE LINE IS COMPLETED A CARRIAGE RETURN-LINE FEED IS SENT AND THE LAST TWO REFERENCE LINES ARE PRINTED AS IN TEST 5. ANY TIMING PROBLEMS WILL SHOW IN THE FIRST (0-10) CHARACTERS OF THE MIDDLE REFERENCE LINE, AGAIN AS IN TEST 5.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:

```
000000000000000000000000000022
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
123456789012345678901234567890
123456789012345678901234567890
```

5.1.8 TEST 7 - OVERPRINT TEST

THIS TEST IS DESIGNED TO CHECK THE SPACING AND REPEATABLE PRINTING CHARACTERISTICS OF THE PRINTER. THREE ROWS OF CHARACTERS ARE EACH OVERPRINTED SEVERAL TIMES. THE ROWS CONSIST OF THE FOLLOWING CHARACTERS ALTERNATED ACROSS THE LINE:

ROW 1 M-SP
ROW 2 SP-S
ROW 3 S-SP

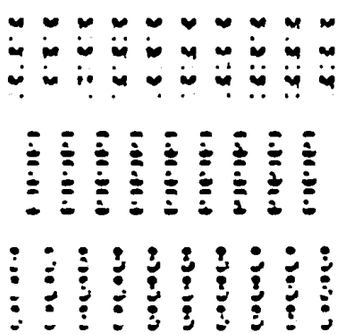
THE RESULTING PATTERN WILL BE A CHECKERBOARD PATTERN AND THE OVERPRINTED CHARACTERS SHOULD BE ALIGNED PROPERLY WITH THE INITIAL CHARACTER.

EXAMPLE:

M M M M M M M M M M M M M M
S S S S S S S S S S S S S S
S S S S S S S S S S S S S S

WITH THE 6.10 LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EACH RECEIVED CARRIAGE RETURN, THE LINES WILL NOT BE OVERPRINTED. THERE WILL BE THREE LINES OF EACH CHARACTER WITH A BLANK LINE IN BETWEEN EACH GROUP OF CHARACTERS. THE CHARACTERS IN EACH GROUP SHOULD BE IN THE SAME COLUMNS.

EXAMPLE:



5.1.9 TEST 10 - PRINTING FREQUENCY SWEEP TEST

THIS TEST PRINTS THE CHARACTER "H" REPEATEDLY, 30(10) CHARACTERS PER LINE FOR FOUR LINES. DURING THE FIRST TWO LINES, THE TIME INTERVAL BETWEEN CHARACTERS IS INCREASED FROM 30(10) MILLISECONDS TO 1.8 SECONDS USING THE FOLLOWING FORMULA TO CREATE A LOGRITHMIC INCREASE:

NEW DELAY = OLD DELAY + OLD DELAY /6 + OLD DELAY /29

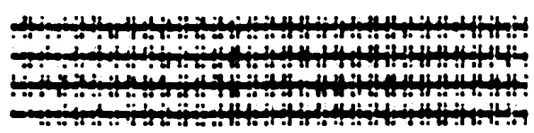
THE LAST TWO LINES DO JUST THE REVERSE. THE TIME INTERVAL BETWEEN CHARACTERS IS DECREASED FROM 1.8 SECONDS TO 30(10) MILLISECONDS USING THE FOLLOWING FORMULA TO AGAIN CREATE A LOGRITHMIC DECREASE:

NEW DELAY = OLD DELAY - OLD DELAY /6 - OLD DELAY /29

LOOK FOR POSSIBLE MISS-ALIGNMENT OF THE CHARACTERS OR SPACES BETWEEN CHARACTERS AS AN INDICATION OF TIMING PROBLEMS.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:



5.1.10 TEST 11 - RIBBON FEED TEST

THIS TEST CHECKS THE RIBBON FEED MECHANISM BY PRINTING A SINGLE COLUMN OF [illegible] FROM THE LEFT HAND MARGIN OF THE PAGE. VISUAL CHECK FOR PROPER OPERATION OF THE RIBBON FEED MECHANISM DURING THIS TEST.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:

6.1.11 TEST 11 - PRINTER BELL TEST

THIS TEST CHECKS THE PRINTER BELL BUFFER TO INSURE THAT EIGHT BELLS ARE DISTINCTLY HEARD, EVEN WHEN SENT AT THE MAXIMUM TRANSFER RATE. THE PROGRAM SENDS 8 BELL CODES AT THE MAXIMUM RATE TO THE PRINTER THEN WAITS 2.5 SECONDS TO ALLOW THE OPERATOR TO HEAR THE BELLS.

6.1.12 TEST 12 - LIFE TEST

THIS TEST RUNS CONTINUOUSLY AND IS RUN AS AN INDIVIDUAL, SPECIAL TEST. IT IS NOT PART OF THE STANDARD PRINTING TEST SEQUENCE.

THIS TEST PRINTS 2 LINES OF EACH PRINTABLE CHARACTER AND THEN REPEATS CONTINUOUSLY. THE SECOND LINE OF EACH CHARACTER IS OVERPRINTED 4 TIMES TO CONSERVE PAPER. AT THE END OF EACH COMPLETE PASS THROUGH THE CHARACTER SET, A MESSAGE IS PRINTED INDICATING THE NUMBER OF PASSES EXECUTED. IF ANY CHARACTER (EXCEPT "RUBOUT") IS TYPED ON THE KEYBOARD DURING THIS TEST, THE PATTERN WILL CHANGE AND RESTART WITH THE NEXT CHARACTER. THIS WILL ONLY HAPPEN IF KEYBOARD CONTROL IS IN USE.

EXAMPLE:

```
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
```

IF THE AUTO LINE FEED OPTION IS SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THE TEST WILL PRINT 80 LINES OF EACH CHARACTER WITH A BLANK LINE BETWEEN THE FIRST AND SECOND LINES AS WELL AS BETWEEN EACH GROUP OF CHARACTERS.

EXAMPLE:

```
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno

!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno

!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno

!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
```

6.2 ECHO TESTS

THESE TESTS ARE DESIGNED AS A TEST OF THE KEYBOARD AND AN AID IN LOCATING TROUBLES WITHIN THE TERMINAL. AT THE BEGINNING OF EACH TEST, THE TEST NUMBER WILL BE PRINTED INDICATING WHICH TEST IS BEING EXECUTED. TYPING A "RUBOUT" OR "DELETE" AT ANY TIME, WHETHER IN KEYBOARD CONTROL OR NOT, WILL EXIT THE CURRENT ECHO TEST AND PRINT A TEST TERMINATION MESSAGE. IF IN KEYBOARD CONTROL, THE SELECT TEST MESSAGE WILL BE PRINTED AND THE PROGRAM WILL AWAIT A TEST SELECTION AS USUAL. IN SWITCH REGISTER CONTROL, THE PROGRAM WILL HALT AT SELHLT WAITING FOR CONTROL VIA THE SWITCH REGISTER. A DETAILED DESCRIPTION OF EACH TEST FOLLOWS:

6.2.1 TEST 20 - CHARACTER ECHO TEST

THIS TEST IS DESIGNED TO OPERATE THE TERMINAL IN A SIMPLEX MODE. ANY CHARACTER TYPED ON THE KEYBOARD (EXCEPT A "RUBOUT") WILL BE ECHOED TO THE PRINTER.

6.2.2 TEST 21 - LINE ECHO TEST, FAST RATE

THIS TEST CONTINUALLY SENDS FULL LINES OF ANY CHARACTER UP TO THE MAXIMUM COLUMN WIDTH. THE TEST PRINTS A "0" CHARACTER WHEN STARTED UNTIL A KEY IS TYPED ON THE KEYBOARD. THE PROGRAM WILL THEN SEND THE TYPED CHARACTER UNTIL ANOTHER CHARACTER IS TYPED OR THE TEST IS TERMINATED BY TYPING A "RUBOUT". THE CHARACTERS ARE TRANSMITTED AT THE MAXIMUM RATE WITH A CARRIAGE RETURN-LINE FEED INSERTED AFTER EVERY 132(10) PRINTABLE CHARACTERS.

IF THE LAGS IS IN HALF DUPLEX WHEN RUNNING THIS TEST, CHARACTERS MAY BE LOST OR GARBLED WHENEVER A CHARACTER IS TYPED ON THE KEYBOARD.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

6.2.3 TEST 22 - LINE ECHO TEST, SLOW RATE

THIS TEST IS IDENTICAL TO TEST 21, EXCEPT A DELAY OF 1.5 SECONDS IS INSERTED BETWEEN EACH CHARACTER TO ALLOW THE PRINTER TO PRINT AN ENTIRE PRINTED CHARACTERS.

6.2.4 TEST 23 - CHARACTER CODE ECHO TEST

THIS TEST WILL PRINT THE OCTAL CODE RECEIVED BY THE PROCESSOR FOLLOWED BY THE CHARACTER OR THE MNEMONIC OF THE CHARACTER EVERY TIME A KEY IS PRESSED ON THE KEYBOARD. THE PARITY OF THE RECEIVED CODE WILL BE INDICATED AS EITHER ODD OR EVEN. ALLOW SUFFICIENT TIME BETWEEN CHARACTERS FOR THE LINE TO BE PRINTED.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:

```
201   A     ODD
203   C     ODD
215   CR    EVEN
240   SP    EVEN
```

6.2.5 TEST 24 - SELECTED PATTERN ECHO TEST

THIS TEST IS DESIGNED TO GIVE MAINTENANCE THE FLEXIBILITY TO CHOOSE THEIR OWN PATTERNS FOR ISOLATING ANY SPECIFIC PROBLEMS WHICH MAY ARISE IN THE FIELD.

TYPE ANY CHARACTERS (EXCEPT CONTROL-C AND RUBOUT) AND EACH CHARACTER WILL BE ECHOED AS TYPED. A MAXIMUM OF 256(10) CHARACTERS MAY BE INPUTTED. NO CARRIAGE RETURNS OR LINE FEEDS ARE INSERTED BY THE PROGRAM. ALL CHARACTERS MUST BE INPUTTED BY THE OPERATOR. TO TERMINATE THE INPUT STRING TYPE A CONTROL-C. THE PROGRAM WILL THEN CONTINUALLY ECHO THE INPUTTED PATTERN. TO STOP THE PRINTING, TYPE CONTROL-C. THE PROGRAM WILL STOP PRINTING THE PATTERN AND WILL WAIT FOR EITHER ANOTHER PATTERN INPUT TERMINATED BY A CONTROL-C, OR THE SAME PATTERN MAY BE USED AGAIN BY TYPING CONTROL-C. TO EXIT THE TEST AT ANY TIME, TYPE A "RUBOUT".

WHEN ANY OPTIONS ARE AVAILABLE, BE CAREFUL WHAT CHARACTERS OR CHARACTER SEQUENCES ARE SELECTED.

6.2.6 TEST 25 - BELL ECHO TEST

THIS TEST IS DESIGNED TO TEST THE BELL ON COLUMN 64 IF TYPING HAS OCCURRED ON THE LINE. THE TEST PRINTS A MESSAGE:

TYPE ANY PRINTABLE CHARACTER AND LISTEN FOR BELL .....

AFTER THE TEST MESSAGE IS PRINTED, TYPE ANY PRINTABLE CHARACTER ON THE KEYBOARD. THE CHARACTER WILL BE ECHOED AND THE BELL SHOULD RING. THE MESSAGE WILL THEN BE TYPED AGAIN. TYPE THE "RUBOUT" KEY TO TERMINATE THE TEST AT ANY TIME.

6.3 OPTION TESTS

THESE TESTS ARE DESIGNED AS A TEST OF THE VARIOUS OPTIONS IN WHATEVER COMBINATIONS THEY ARE AVAILABLE IN THE LA36. AT THE BEGINNING OF EACH TEST, THE TEST NUMBER WILL BE PRINTED INDICATING WHICH TEST IS BEING EXECUTED. TYPING A "RUBOUT" OR "DELETE" AT ANY TIME, WHETHER IN KEYBOARD CONTROL OR NOT, WILL EXIT THE CURRENT OPTION TEST. A DETAILED DESCRIPTION OF EACH TEST FOLLOWS:

6.3.1 TEST 30 - SECONDARY CHARACTER SET OPTION

THIS TEST IS DESIGNED TO TEST THE SECONDARY CHARACTER SET OPTION, TESTING THE ABILITY TO SELECT EITHER CHARACTER SET UNDER SOFTWARE CONTROL FROM THE CPU AND PRINTING THE CORRECT CHARACTERS WITHIN EACH CHARACTER SET.

A NUMBER IS PRINTED AT THE LEFT MARGIN INDICATING WHICH CHARACTER SET IS BEING PRINTED. #1 INDICATES THE PRIMARY SET AND #2 INDICATES THE SECONDARY SET (APL). AFTER THE NUMBER, THE APPROPRIATE SHIFT IN (SI) OR SHIFT OUT (SO) WILL BE SENT FOLLOWED BY THE ENTIRE PRINTABLE CHARACTER SET. IF LESS THAN 98 COLUMNS ARE BEING TESTED, A CARRIAGE RETURN - LINE FEED WILL BE INSERTED IN THE APPROPRIATE PLACES. THIS WILL BE REPEATED, ALTERNATING BETWEEN PRIMARY AND SECONDARY SETS UNTIL 16 LINES HAVE BEEN PRINTED (IF USING 98 OR MORE COLUMNS). THERE WILL BE A BLANK LINE BETWEEN EACH PAIR OF LINES TO SEPARATE EACH GROUPING. CHANGE LOCATION "T30SC" AT THE END OF TEST30 TO 377 (8) IF USING 8 BIT SELECTION CODE RATHER THAN THE SI AND SO TO SELECT CHARACTER SETS. THE TEST WILL THEN SET OR CLEAR BIT 8 INSTEAD OF SENDING THE SI OR SO TO SELECT CHARACTER SETS.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EACH RECEIVED CARRIAGE RETURN, THERE WILL BE EXTRA BLANK LINES EVERY PLACE A CARRIAGE RETURN IS SENT.

EXAMPLE:

```
#1= !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
#2= !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~

#1= !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
#2= !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
```

5.3.3 TEST 31 - SELECTIVE ADDRESSING OPTION

THIS TEST IS DESIGNED TO TEST THE VARIOUS FUNCTIONS OF THE SELECTIVE ADDRESSING OPTION. THE TEST FIRST SENDS AN "EOT" (004) TO DISABLE ALL TERMINALS AND TRIES TO PRINT AN ERROR MESSAGE. THE ERROR MESSAGE SHOULD NOT BE PRINTED ON ANY TERMINAL WITH THE SELECTIVE ADDRESSING OPTION. THEN A "BEL" (007) AND "STX" (002) ARE SENT TO SELECT ALL TERMINALS. AT THIS POINT THE TEST NUMBER IS PRINTED ON ALL TERMINALS. THUS, IF AN ERROR MESSAGE IS PRINTED BEFORE THE TEST NUMBER, THE EOT DID NOT DE-SELECT THE TERMINAL WHERE THE MESSAGE WAS PRINTED.

THE TEST NEXT SENDS AN EOT DIRECTLY FOLLOWED BY A STX, WITH NO SELECT CHARACTER. AGAIN, THE ERROR MESSAGE IS SENT TO ALL TERMINALS, WHICH SHOULD NOW BE ALL DE-SELECTED. THE ERROR MESSAGE SHOULD NOT BE PRINTED ON ANY TERMINAL WITH THE SELECTIVE ADDRESSING OPTION

THE NEXT SERIES OF CHECKS ARE MADE ON THE GROUP SELECT CHARACTER. A TABLE LOCATED AT THE END OF TEST 31 IN THE LISTING IS USED TO TEST VARIOUS GROUP SELECT CHARACTERS. THE FIRST ZERO ENCOUNTERED IN THE TABLE WILL INDICATE THE END OF THE TABLE AND THE TEST WILL GO TO THE NEXT SERIES OF CHECKS ON THE OPTION. THE TABLE IS PRESET WITH A SINGLE GROUP SELECT CHARACTER, THE LETTER "G", BUT ALLOWS ROOM TO TEST UP TO 8 DIFFERENT SELECT CODES. THIS TABLE SHOULD BE CHANGED TO CONTAIN THE VARIOUS GROUP SELECT CHARACTERS DESIRED TO TEST WITH ONE ASCII CODE PER LOCATION. THE TEST WILL THEN USE THE VARIOUS GROUP SELECT CHARACTERS TO SELECT TERMINALS AND PRINT A MESSAGE ON EACH SELECTED TERMINAL INDICATING THE GROUP SELECT CHARACTER USED. CHECK THAT THE CORRECT GROUP SELECT CHARACTER HAS ENABLED EACH TERMINAL. ALSO, IT MAY BE HELPFUL TO PLACE ;UNUSED SELECT CHARACTERS IN THE TABLE TO CHECK THAT THEY DO NOT SELECT TERMINALS. IF AN ERROR MESSAGE HAS PRINTED BETWEEN THE TEST NUMBER AND THE GROUP SELECT MESSAGE, THE TERMINAL WHERE THE MESSAGE WAS PRINTED WAS SELECTED BY AN EOT AND STX WITH NO SELECT CHARACTER BETWEEN THEM.

THE LAST SERIES OF CHECKS ARE MADE ON THE UNIQUE SELECT CHARACTER. A TABLE LOCATED AT THE END OF TEST 31 IN THE LISTING IS USED TO TEST VARIOUS UNIQUE SELECT CHARACTERS. THE FIRST ZERO ENCOUNTERED IN THE TABLE WILL INDICATE THE END OF THE TABLE. THE PROGRAM WILL SELECT ALL TERMINALS USING THE BEL CODE BEFORE EXITING THE TEST. THE TABLE IS PRESET WITH A SINGLE UNIQUE SELECT CHARACTER, THE LETTER "U", BUT ALLOWS ROOM TO TEST UP TO 16 DIFFERENT UNIQUE SELECT CODES. THIS TABLE SHOULD BE CHANGED TO CONTAIN THE VARIOUS UNIQUE SELECT CHARACTERS DESIRED TO TEST, WITH ONE ASCII CODE PER LOCATION. MAKE SURE THAT EACH CHARACTER IN THE TABLE IS A VALID UNIQUE SELECT CODE OR THE DIAGNOSTIC WILL HANG DURING THIS PORTION OF THE TEST. USING EACH UNIQUE SELECT CHARACTER IN TURN, THE TEST WILL PERFORM THE REMAINING CHECKS OF THE SELECTIVE ADDRESSING OPTION.

THE TEST WILL SEND AN EOT FOLLOWED BY THE CURRENT UNIQUE SELECT CHARACTER. BEFORE THE STX IS SENT, THE TEST WILL TRY TO PRINT THE ERROR MESSAGE ON ALL TERMINALS. THEN THE STX WILL BE SENT AND A MESSAGE WILL BE PRINTED TO INDICATE THE UNIQUE SELECT CHARACTER USED. CHECK THAT THE CORRECT UNIQUE SELECT CHARACTER HAS ENABLED EACH TERMINAL. IF AN ERROR MESSAGE IS PRINTED BEFORE THE UNIQUE SELECT MESSAGE, THE TERMINAL WHERE THE MESSAGE WAS PRINTED WAS ENABLED BEFORE THE STX WAS RECEIVED. A MESSAGE WILL THEN BE PRINTED TELLING THE OPERATOR TO TYPE ANY PRINTABLE CHARACTER TO CHECK THAT THE KEYBOARD IS ENABLED. WHATEVER CHARACTER IS TYPED WILL BE ECHOED TO THE TERMINAL.

THE FINAL SECTION OF THE TEST WILL USE A DUMMY SELECT CHARACTER. THE ASCII CODE FOR THIS SELECT CHARACTER IS LOCATED BETWEEN THE TWO SELECT CHARACTER TABLE AT THE END OF THE TEST. THIS LOCATION SHOULD CONTAIN THE ASCII CODE OF ANY UNUSED SELECT CHARACTER. THE TEST WILL SEND AN EOT FOLLOWED BY THE DUMMY SELECT CHARACTER AND AN STX. THE ERROR MESSAGE WILL BE LOADED TO ALL TERMINALS AND SHOULD NOT BE PRINTED ON ANY TERMINALS SINCE ALL SHOULD BE DE-SELECTED. NEXT AN ETX (003) FOLLOWED BY THE CURRENT UNIQYUE SELECT CHARACTER AND AN STX WILL BE SENT AND A PRINTED MESSAGE WILL INDICATE THE SELECT CHARACTER USED. ANOTHER EXT WILL BE SENT, FOLLOWED BY THE DUMMY SELECT CHARACTER AND AN STX THIS TIME. A MESSAGE WILL AGAIN BE PRINTED INDICATING THE CURRENT UNIQUE SELECT CHARACTER. ALL SELECTED TERMINALS SHOULD REMAIN SELECTED AND NO OTHER TERMINALS SHOULD GET SELECTED.

6.3.3 TEST 32 - ANSWER BACK OPTION

THIS TEST IS DESIGNED TO TEST THAT THE ANSWER BACK OPTION SENDS THE CORRECT MESSAGE UPON RECEIPT OF AN ENQ (005) OR UPON TYPING CONTROL-E OR THE HERE IS KEY ON THE KEYBOARD. THE TEST WILL SEND AN ENQ (005), READ THE MESSAGE, AND THEN PRINT OUT THE MESSAGE ON THE LA36. THE TEST WILL THEN ASK THE OPERATOR TO DEPRESS THE HERE IS KEY, READ THE MESSAGE, AND THEN PRINT OUT THE MESSAGE. FINALLY, THE TEST WILL TELL THE OPERATOR TO DEPRESS THE CONTROL-E KEY, READ THE MESSAGE, AND PRINT OUT THE MESSAGE. IF THE SELECTIVE ADDRESSING OPTION IS AVAILABLE, THE AUTO ANSWER BACK OPTION WILL NOT RESPOND TO ANOTHER ENQ AFTER THE FIRST ONE RECEIVED. THUS, YOU MAY HAVE TO DEPRESS THE REBOOT KEY TO END THE TEST.

6.3.4 TEST 33 - TOP OF FORM OPTION

THIS TEST IS DESIGNED TO TEST THE FORM FEED CAPIBILITY OF THE TOP OF FORM OPTION. A SET OF INSTRUCTIONS IS PRINTED FOR THE OPERATOR TO REMIND HIM TO DEPRESS THE TOP OF FORM RESET SWITCH AFTER MAKING EACH SWITCH SETTING. UPON COMPLETION OF EACH SETTING, AFTER DEPRESSING THE RESET SWITCH, TYPE ANY CHARACTER (EXCEPT RUBOUT) ON THE KEYBOARD TO TEST THAT SWITCH SETTING. THE REFERENCE LINES PRINTED WILL INDICATE THE LENGTH FORM FEED JUST EXECUTED AND THE NEXT SWITCH SETTING TO MAKE. THE 3 INCH FORM FEED IS TESTED TWICE BEFORE TESTING THE REMAINING POSITIONS. THE FIRST TIME, 16 OR 17 LINE FEEDS ARE EXECUTED BEFORE DOING THE FORM FEED, DEPENDING ON HOW THE AUTO LINE FEED OPTION IS SET UP. THE DIAGNOSTIC WILL THEN TEST EACH POSITION IN SEQUENCE FROM 3 TO 14 INCHES. THE SINGLE STEP POSITION IS NOT CHECKED.

6.3.5 TEST 34 - HORIZONTAL TAB OPTION

THIS TEST CHECKS THE ABILITY TO SET A TAB IN EVERY COLUMN AND AT PREDETERMINED INTERVALS, AS WELL AS THE ABILITY TO CLEAR ALL TABS. THE PROGRAM SETS A TAB IN THE PREDETERMINED COLUMN, DOES A BACKSPACE, AND PRINTS AN "O". AFTER THE LINE IS PRINTED AND THE TABS ARE SET, A CARRIAGE RETURN IS SENT AND THEN THE PRINT HEAD IS POSITIONED USING TABS AND X'S ARE PRINTED OVER THE O'S. SINCE THE FIRST LINE OF THE TEST SETS A TAB IN EVERY COLUMN, THE PRINT HEAD IS TABED ACROSS THE PAGE TWICE TO TEST ALL TABS. THE FIRST PASS CHECKS THE EVEN NUMBERED COLUMNS WHILE THE SECOND PASS CHECKS THE ODD NUMBERED COLUMNS. THE TEST SETS TABS IN EVERY COLUMN, EVERY OTHER COLUMN, AND EVERY 4, 8, 16, 32, 64, 128, & 132 COLUMNS. ALL HORIZONTAL TABS WILL BE CLEARED AT THE END OF THE TEST IF THE TEST IS RUN TO COMPLETION. IF A RUBOUT IS USED TO EXIT THE TEST BEFORE COMPLETION, THE TABS WILL STILL BE SET.

EXAMPLE:

```
OOOOOOOOOO
 O O O O O
   O   O
       O
```

WHEN THE AUTO LINE FEED OPTION IS SET UP TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BEFORE EACH REFERENCE LINE OF O'S AND THE X'S WIL BE PRINTED ON THE NEXT LINE UNDER THE O'S. THE FIRST LINE OF O'S WILL HAVE 2 LINES OF X'S UNDER IT, THE FIRST HAVING X'S IN ALL EVEN NUMBERED COLUMNS AND THE SECOND HAVING X'S IN ALL ODD NUMBERED COLUMNS.

EXAMPLE:

```
OOOOOOOOOOOO
 X X X X X X
X X X X X X

 O O O O O O
 X X X X X X

   O   O   O
   X   X   X

       O
       X
```

5.3.6 TEST 35 - VERTICAL TAB OPTION

THIS TEST CHECKS THE VERTICAL TAB OPTION BY TESTING THE ABILITY TO SET TABS IN VARIOUS POSITIONS OF A 14 INCH FORM. AN INSTRUCTION IS PRINTED TELLING THE OPERATOR TO SET A 14 INCH FORM LENGTH AND DEPRESS THE TOP OF FORM RESET SWITCH. WHEN READY, TYPE ANY CHARACTER (EXCEPT RUBOUT) ON THE KEYBOARD TO CONTINUE. THE TEST WILL SEND LINE FEEDS, SET TABS, AND PRINT REFERENCE LINES WHEREVER A TAB IS SET. AT THE END OF THE FORM, A MESSAGE WILL INDICATE TO EITHER REMOVE THE REFERENCE PAGE (WITHOUT TOUCHING THE KEYBOARD) OR RESET THE FIRST REFERENCE LINE. TO RESET THE REFERENCE PAGE IN THE PRINTER, OPEN THE PAPER TRACTORS AND PLACE THE FIRST REFERENCE LINE INFRONT OF THE PRINT HEAD. WHEN READY TO CONTINUE, TYPE ANY CHARACTER (EXCEPT RUBOUT) ON THE KEYBOARD. THE TEST WILL THEN REPRINT THE REFERENCE LINES, USING THE TABS INSTEAD OF LINE FEEDS TO ADVANCE THE PAPER. IF THE FIRST REFERENCE PAGE WAS REMOVED, HOLD IT AGAINST THE SECOND REFERENCE PAGE TO CHECK FOR PROPER PAPER ADVANCING USING TABS. IF THE REFERENCE PAGE WAS RESET IN THE PRINTER, THE SECOND SET OF REFERENCE LINES SHOULD HAVE PRINTED DIRECTLY OVER THE FIRST SET EXCEPT ON THE FIRST LINE WHERE THEY SHOULD BE SIDE-BY-SIDE. ALLOW FOR A SLIGHT VARIANCE IN PAPER POSITION WHEN CHECKING THAT THE REFERENCE LINES ARE CORRECT, LOOK FOR FULL LINE DIFFERENCES. THE TEST PRODUCES 0,1,2,3,4,5,6,7,8,9, & 10 BLANK LINES BETWEEN THE REFERNECES LINES, IN THAT ORDER.

EXAMPLE:

```
--------\\\\\\\\    --------\\\\\\\\
--------\\\\\\\\        |
                        |
--------\\\\\\\\        |
                        '----- FIRST REF.  LINE WITH TABS

--------\\\\\\\\


--------\\\\\\\\



--------\\\\\\\\
```

THE REFERENCE PATTERN PRODUCED WILL BE DIFFERENT IF THE AUTO LINE FEED OPTION IS SET UP TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN. THERE WILL, IN FACT, BE ONE EXTRA LINE BETWEEN EACH REFERENCE LINE DUE TO THE AUTO LINE FEED.

EXAMPLE:

```
--------\\\\\\\\    --------\\\\\\\\
--------\\\\\\\\

--------\\\\\\\\


--------\\\\\\\\


--------\\\\\\\\
```

N02.

TABLE OF CONTENTS

17	SWITCH REGISTER OPTIONS
36	SPECIAL OPERTIONAL NOTES
45	EQUATES
104	TRAP CATCHER & STARTING ADDRESSES
147	SYMBOL DEFINITIONS
191	PROGRAM INITIALIZATION & CONTROL
627	TEST ADDRESS TABLE
663	EMT TRAP DECODER
709	COMMON ROUTINES
1277	PRINTER TESTS
1855	ECHO TESTS
2172	OPTION TESTS
2711	DH11 VARIABLE PARAMETER TABLE
3008	EXISTING LINE TABLE
3048	DIAGNOSTIC MESSAGES

```
                        .TITLE  MAINDEC-11-DZLAC-C


;LA36 TERMINAL DIAGNOSTIC
;DH11 AND DZ11 INTERFACES


;AUTHOR:   ROBERT W. BAKER
;COPYRIGHT 1974,1975, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754


                        .SBTTL  SWITCH REGISTER OPTIONS

;       SWITCH  POSITION        FUNCTION
;       15      UP              HALT AT END OF CURRENT TEST
;               DOWN            CONTINUE NORMAL TEST SEQUENCE
;       13      UP              DRIVE ONLY SELECTED TERMINAL
;               DOWN            DRIVE ALL TERMINALS
;       5       UP              SELECT TEST AT START-UP
;               DOWN            START NORMAL TEST SEQUENCE
;       00 - 07                 # OF COLUMNS AT START-UP


                        .SBTTL  SPECIAL OPERATIONAL NOTES

;1.--   BEFORE START-UP REFER TO THE DESCRIPTION OF THE SWITCH REGISTER
;       SWITCHES AS A FUNCTION OF THE EXPT MODE AND VERIFY THAT THE
;       SHOULD BE SET UP BEFORE RUNNING THE DIAGNOSTIC.

;2.--   THE DIAGNOSTIC WILL NOT RUN UNLESS THE ELTAB TABLE IS MANUALLY INITIALIZED.
```

```
[illegible]                                     .SBTTL  EQUATES

                                        ;       REGISTER EQUATES

                                        ;       SYSTEM EQUATES

                                                [illegible]
```

```
                                        ;       PROGRAM TRAP EQUATES

                                                TYPE=EMT+0
                                                CHAIN=EMT+1
                                                [illegible]=EMT+2
                                                DELAY=EMT+3
                                                TTYCTL=EMT+4
                                                TTYSTL=EMT+5
                                                CRLF=EMT+6
                                                SCRLF=EMT+7
                                                LF=EMT+10
                                                PRINTC=EMT+11
                                                PRTHDR=EMT+12
                                                READ=EMT+13
                                                [illegible]=EMT+14
                                                [illegible]=EMT+15
                                                [illegible]=EMT+16
                                                [illegible]=EMT+17
                                                [illegible]=EMT+20
                                                [illegible]=EMT+21
                                                [illegible]=EMT+22
```

```
                                        .SETL  TRAP CATCHER'S STARTING ADDRESSES


                                        .ENAB  [illegible]
                                        .ENAB  [illegible]          ;ENABLE ABSOLUTE ADDRESSING

        000000                          .=0
000000  000000                          [illegible]
000000  000000                          [illegible]
        000004                          .=4
000004                          MACHER:
        000030                          .=30
000030  000000                          [illegible]
000000  000000                          [illegible]
        000034                          .=34
000034  010000                          [illegible]
        000038                          .=38
000038  010000                          010000
        000074                          .=74
000074  000000                  [illegible]     [illegible]
                                ;STARTING ADDRESSES
000000  000000  000000                  [illegible]     [illegible]

        000100                          .=100
000100  000000                  [illegible]     [illegible]
```

```
                                        .SBTTL  SYMBOL DEFINITIONS

000554                  TIMER:  554                     ;1 MSEC COUNTER FOR ROUTINE "DELA "

000000                  CNTLSW: OPEN                    ;CONSOLE TERMINAL CONTROL SWITCH
000000                  RTNNO:  OPEN                    ;CONTAINS CURRENT TEST NUMBER
000000                  NXTST:  OPEN                    ;CONTAINS ADDRESS OF NEXT TEST
000000                  WIDTH:  OPEN                    ;CURRENT PAPER WIDTH, BINARY
000000                  REPT:   OPEN                    ;TEMP STORAGE FOR TESTS E0210E022
000000                  SPCNT:  OPEN                    ;COUNTER FOR TEST ROUTINE "P*3"
000000                  CURTST: OPEN                    ;ADDRESS OF CURRENT TEST
000000                  POSI:   OPEN                    ;POSITION COUNTER FOR TESTS E0210E022
000000                  TEMPCH: OPEN                    ;TEMP STOR FOR ECHO TESTS
000000                  PARITY: OPEN                    ;PARITY FLAG
000000                  PCHAR:  OPEN                    ;CHAR CODE WITH PARITY BIT
000000                  LPCNT:  OPEN                    ;COUNTER FOR TEST ROUTINE "PT*"
160020                  CHADR:  160020                  ;ADDRESS OF FIRST DHII, RECALCULATED
160020                  SCR:    160020                  ;SYSTEM CONTROL REGISTER
160022                  NRCRA:  160022                  ;NEXT RECEIVED CHARACTER REG.
160024                  LPR:    160024                  ;LINE PARAMETER REG
160026                  CARA:   160026                  ;CURRENT ADDRESS REGISTER
160030                  BYCR:   160030                  ;BYTE COUNT REG
160032                  BAR:    160032                  ;BUFFER ACTIVE REG
160036                  SSR:    160036                  ;SILO STATUS REG
000000                  CNTCH:  OPEN                    ;NO. OF THE DHII UNDER TEST
000000                  LINENO: OPEN                    ;NO. OF THE TERMINAL UNDER TEST
000000                  MASK:   OPEN                    ;MASK OF LINE NO. UNDER TEST
000001  000002  000004  BITTAB: 1,2,4                   ;TABLE OF LINE NO. MASKS
000010  000020  000040          10,20,40
000100  000200  000400          100,200,400
001000  002000  004000          1000,2000,4000
010000  020000  040000          10000,20000,40000
100000                          100000
000000                  ACTIV:  OPEN                    ;TEST ACTIVE INDICATOR
000000                  LOSW:   OPEN                    ;DHII UNDER TEST--DJ11 UNDER TEST
000000                  DJCNT:  OPEN                    ;COUNT OF DJ11'S ON BUS
000000                  DHCNT:  OPEN                    ;COUNT OF DH11'S ON BUS
000000                  SCR1:   OPEN                    ;TEMP STORAGE USED BY PRINTS
000000                  SCR2:   OPEN                    ;TEMP STORAGE USED BY PRINTS
160010                  CSR:    160010                  ;DH11 CONTROL STATUS REG
160012                  TCR:    160012                  ;DH11 TRANSMITTER CONTROL REG
160014                  TBUF:   160014                  ;DH11 TRANSMITTER BUFFER REG
160000                  DR:     160000                  ;DR REG ADDRESS
```

```
                                                .SBTTL  PROGRAM INITIALIZATION & CONTROL

                                        ;********
                                        ;
                                        ;PROGRAM START
                                        ;
                                        ;********

001034  012737  177777  001224  START1: MOV     #177777,ICSW    ;SET UP FOR DJ11 TEST
001042  000403          001224          BR      START1
001044  005037  001224          START:  CLR     ICSW            ;SET UP FOR DH11 TESTS

001050  012706  001100          STARTX: MOV     #SPBIT,SP       ;SET STACK POINTER
001054  013746  000114          MOV     @#114,-(SP)     ;SAVE CURRENT VECTORS
001060  013746  000004          MOV     @#4,-(SP)
001064  012737  001310  000004          MOV     #11$,@#4        ;SET TIMEOUT VECTOR
001072  005737  177734          TST     @#SR            ;TRY REFFERENCING SW REG
001076  000404          BR      11$             ;BRANCH IF DID NOT TIME OUT
001100  012737  000176  001244  10$:    MOV     #SWREG,SR       ;POINT TO SOFTWARE SW REG
001106  062706          CMP     (SP)+,SP        ;RESET STACK
001110  012637  000004          11$:    MOV     (SP)+,@#4       ;RESET VECTOR
001114  012637  000006          MOV     (SP)+,@#6
001120  104001          TRAP    1               ;CLEAN UP
001122  005037  001112          CLR     ERR     ;INITIALIZE TERMINAL ERROR SWITCH
001126  012700  001142          MOV     #PFAIL,R0       ;SET UP POWER FAIL ROUTINE
001132  010000          MOV     R0,@#PFAIL
```

```
                                        ;**********
                                        ;
                                        ;THIS NEXT PART CHECKS THE PRESENCE OF DJ11 OR DH11
                                        ;STARTING AT 776010. A MESSAGE WILL BE PRINTED ON
                                        ;THE CONSOLE TERMINAL INDICATING THE NUMBER
                                        ;PRESENT. THE PRINTER DIAGNOSTIC WILL ADDRESS EACH OF
                                        ;THE TERMINALS IN THE SYSTEM
                                        ;
                                        ;**********

001346  012737  001400  000034          MOV     #END3,MACHER    ;INIT TIME OUT TRAP
001354  005037  001226                  CLR     DJCNT           ;CLEAR DJ11 COUNTER
001360  012700  160010                  MOV     #160010,R0      ;ADDR OF FIRST DJ11
001364  005710                  1$:     TST     (R0)            ;REF DJ11
001366  062700  000010                  ADD     #10,R0          ;SET R0 TO NEXT DJ11
001372  005237  001226                  INC     DJCNT           ;INCREMENT COUNT OF DJ11'S
001376  000772                          BR      1$              ;TEST PRESENCE OF NEXT DJ11

001400  022626                  END3:   POPSPE                  ; POP 2 FROM STACK
001402  030027  000010                  BIT     R0,#10          ;CHECK IF R0 IS MULTIPLE OF 20
001406  001402                          BEQ     1$              ;SKIP IF YES
001410  062700  000010                  ADD     #10,R0          ;MAKE R0 FIRST DH11 ADDR
001414  010037  001134          1$:     MOV     R0,DHADR        ;SAVE AS FIRST DH11 ADDRES
001420  012737  001446  000034          MOV     #END4,MACHER    ;SET TIME OUT TRAP
001426  005037  001230                  CLR     DHCNT           ;CLEAR COUNT OF DH11'S
001432  005710                  2$:     TST     (R0)            ;TEST IF DH11 IS PRESENT
001434  062700  000020                  ADD     #20,R0          ;YES, SET R0 TO NEXT DH11
001440  005237  001230                  INC     DHCNT           ;INCREMENT COUNT OF DH11'S
001444  000772                          BR      2$              ;CHECK IF NEXT ONE IS PRESENT

001446  022626                  END4:   POPSPE                  ;POP 2 FROM STACK
```

```
                                        ;**********
                                        ;
                                        ;READ THE PAPER WIDTH, NUMBER OF COLUMNS,
                                        ;FROM SWITCH REGISTER POSITIONS 0-7. SAVE AND
                                        ;CONVERT TO 3 ASCII CHARACTERS. A WIDTH GT132
                                        ;OR LT30 COLUMNS (DECIMAL) WILL BE ABORTED TO 132.
                                        ;
                                        ;**********

001450  012701  177570                  MOV     $SR,R1          ;PUT (SR) INTO R1
001454  042701  177400                  BIC     #177400,R1      ;SAVE ONLY BITS 0-7
001460  020127  000204                  CMP     R1,#204         ;TEST NO. COLUMN GT132
001464  101003                          BHI     2$              ;COLUMNS GT132, DEFAULT TO 132
001466  020127  000035          1$:     CMP     R1,#35          ;CHECK IF NO. COLUMNS LT 30
001472  101002                          BHI     3$              ;NOT LT 30 NOR GT 132
001474  012701  000204          2$:     MOV     #204,R1         ;COLUMNS LT 30 OR GT 132, DEFAULT
001500  010137  001112          3$:     MOV     R1,WIDTH        ;SAVE NO. COLUMNS IN WIDTH
001504  012700  017045                  MOV     #HORO,R0        ;ADDR TO STORE ASCII COLUMN VALUE
001510  012702  000003                  MOV     #3,R2           ;DO A 3 CHAR. CONVERSION
001514  104015                          BTOASC                  ;CONVERT NO. COLUMNS TO ASCII
001516  005037  001106                  CLR     RTNNO           ;SET ROUTINE NO = 0
001522  000401                  4$:     BR      5$              ;PRINT TITLE FIRST TIME
001524  000405                          BR      6$              ;SKIP IT AFTER FIRST TIME
001526  104002                  5$:     TYPEM                   ;PRINT DIAGNOSTIC HEADER
001530  016620                          STARTM
001532  012737  000240  001522          MOV     #NOP,4$         ;TAKE OUT BRANCH INSTR
001540  005737  001224          6$:     TST     IOSW            ;CHECK IF DJ11 OR DH11 UNDER TEST
001544  001440                          BEQ     10$             ;BRANCH IF DH11
001546  013701  001226                  MOV     DJCNT,R1        ;GET NUMBER OF DJ11'S
001552  012700  017034                  MOV     #DJ11S,R0       ;ADDR TO STOR ASCII CHAR
001556  012702  000003                  MOV     #3,R2           ;NO. OF ASCII CHARS (3)
001562  104015                          BTOASC                  ;CONVERT TO ASCII
001564  104006                          CRLF
001566  104002                          TYPEM
001570  017034                          DJ11S                   ;OUTPUT MESSAGE
```

```
                                        ;********
                                        ;
                                        ; START ALL DJ11 RECEIVER SCANNERS
                                        ;
                                        ;********

001572  012701  160010                  MOV     #160010,R1      ;ADDR OF FIRST DJ11 SCP
001576  013703  001226                  MOV     DJCNT,R3        ;NO. OF DJ11 TO R3
001602  001002                          BNE     8$              ;BRANCH IF COUNT IS NOT ZERO
001604  000000                  7$:     HALT
001606  000776                          BR      7$              ;DO NOT CONTINUE
001610  052711  000010          8$:     BIS     #10,(R1)        ;CLEAR MOS
001614  012700  000005          9$:     MOV     #5,R0           ;DELAY TIME TO R0
001620  104003                          DELAY
001622  031127  000020                  BIT     (R1),#20        ;TEST IF CLEAR STILL BUSY
001626  001372                          BNE     9$              ;BRANCH IF BUSY
001630  052711  000001                  BIS     #1,(R1)         ;START SCANNER
001634  062701  000010                  ADD     #10,R1          ;ADDR OF NEXT SCR
001640  005303                          DEC     R3              ;DEC COUNT OF DJ11
001642  001362                          BNE     8$              ;NOT ZERO, START NEXT ONE
001644  000412                          BR      11$
001646  013701  001230          10$:    MOV     DHCNT,R1        ;NO. OF DH11'S TO R1
001652  012700  016755                  MOV     #DH11S,R0       ;ADDR TO STORE ASCII CHAR
001656  012702  000002                  MOV     #2,R2           ;NO. OF CHAR TO CONVERT
001662  104015                          BTOASC                  ;CONVERT TO ASCII
001664  104006                          CRLF
001666  104002                          TYPEM                   ;OUTPUT MESSAGE
001670  016755                          DH11S
001672  032777  000400  177344  11$:    BIT     #BIT8,@SR       ;CKECK IF WANT TEST SELECTION?
001700  001004                          BNE     12$             ;BRANCH IF BIT 8 IS SET (YES)
001702  012737  006410  001110          MOV     #PTO,NXTST      ;STARTUP PRINTING TESTS
001710  000430                          BR      CHAINY
001712  104002                  12$:    TYPEM                   ;TYPE SELECT TEST NO. MESSAGE
001714  017257                          MESG3
001716  005037  001222                  CLR     ACTIV           ;SET TEST ACTIVE STATE NOT ACTIVE
001722  005737  001224                  TST     IOSW            ;DH11 OR DJ11
001726  001452                          BEQ     TTY1A           ;WAIT FOR TERMINAL CONTROL FROM DH11
001730  000435                          BR      TTYJA           ;WAIT FOR TERMINAL CONTROL FROM DJ11


                                        ;**********
                                        ;
                                        ;CLEAN--INITIALIZES POINTERS BEFORE ENTERING A TEST
                                        ;
                                        ;**********

001732  012737  000006  000004  $CLEAN: MOV     #6,MACHER       ;SET UP MACHINE ERROR VECTOR
001740  005066  000002                  CLR     2(SP)           ;CLEAR PROCESSOR STATUS WORD
001744  000002                          RTI                     ;RETURN
```

```
                                        ;**********
                                        ;
                                        ;CHAINN--THIS PORTION IS THE COMMON RETURN
                                        ;       FOR ALL TESTS.
                                        ;
                                        ;**********

001746  032737  000001  001104  CHAINN: BIT     #1,CNTLSW       ;CHECK IF TERMINAL CONTROL
001754  001405                          BEQ     2$              ;BRANCH IF NOT
001756  005737  001224                  TST     IOSW            ;DH11 OR DJ11?
001762  001401                          BEQ     1$              ;BRANCH IF DH11
001764  104005                          TTYJTL                  ;WAIT FOR DJ11 TERMINAL CONTROL
001766  104004                  1$:     TTYCTL                  ;WAIT FOR DH11 TERMINAL CONTROL
001770  022626                  2$:     POPSP2                  ;CORRECT STACK

                                        ;**********
                                        ;
                                        ;       IF THE SR BIT 15 IS SET,THE CPU WILL HALT HERE WITH
                                        ;       THE TEST NUMBER IN R0. PRESS CONTINUE TO
                                        ;       RUN NEXT TEST
                                        ;
                                        ;**********

001772  005777  177246          CHAINY: TST     @SR             ;CHECK SW REG.
001776  100003                          BPL     1$              ;BRANCH IF NO HALT
002000  113700  001106                  MOVB    RTNNO,R0        ;CURRENT TEST NUMBER TO R0
002004  000000                          HALT
002006  104017                  1$:     CLEAN                   ;CLEAN UP
002010  012706  001100                  MOV     #SPBOT,SP       ;SET UP STACK POINTER
002014  104016                          FORWD                   ;SET UP VALUES FOR NEXT TEST
002016  000177  177076                  JMP     @CURTST         ;GO TO TEST


                                        ;**********
                                        ;
                                        ;TTYJ-- THIS ROUTINE IS USED WHEN THE DJ11S' ARE UNDER
                                        ;       TEST, OTHERWISE THE COMMENTS  AND INSTRUCTIONS ARE
                                        ;       THE SAME AS FOR TTY1.
                                        ;
                                        ;**********

002022  022626                  TTYJ:   POPSP2                  ;CORRECT STACK
002024  005737  001222          TTYJA:  TST     ACTIV           ;TEST IF ENTRY IS FROM A TEST
002030  001402                          BEQ     1$              ;BRANCH IF NOT
002032  000137  002410                  JMP     TTY1G
002036  004737  003016          1$:     JSR     PC,SCANDJ       ;LOOK FOR INPUT
002042  004737  003124                  JSR     PC,SETDJ        ;SET TERMINAL AS CONSOLE
002046  000137  002134                  JMP     TTY1B           ;GO TO CONTROL
```

```
                                        ;**********
                                        ;
                                        ;TTY1-- THIS ROUTINE IS USED WHEN THE DH11'S ARE UNDER TEST. IT PROCESSES
                                        ;       THE RESPONSE TO THE MESSAGE "SELECT TEST NO.". THE RESPONSE
                                        ;       MUST BE THE TWO DIGIT OCTAL TEST NO. FOLLOWED BY:
                                        ;               "L" TO LOOP ON TEST
                                        ;               "S" TO SEQUENCE ON TEST
                                        ;               "." TO EXECUTE TEST ONCE
                                        ;       ALL SPACES ARE IGNORED AND AN ILLEGAL INPUT WILL BE FLAGED
                                        ;       BY A "?" AND THE RETYPING OF THE ABOVE MESSAGE. THE FIRST
                                        ;       TERMINAL TO RESPOND IS THE TERMINAL UNDER TEST. ALL
                                        ;       PRINTER TESTS WILL OUTPUT TO ALL TERMINALS IF SR BIT 13 IS = 1
                                        ;       WHILE THE ECHO TESTS WILL RESPOND TO THE PRINTER UNDER TEST ONLY.
                                        ;
                                        ;**********
002052  022626                          TTY1:   POPSP2                  ;CORRECT STACK
002054  005737  001230                  TTY1A:  TST     DHCNT           ;BE SURE THAT THERE ARE DH11'S
002060  001002                                  BNE     2$              ;BRANCH IF YES
002062  000000                          1$:     HALT
002064  000776                                  BR      1$              ;DO NOT CONTINUE
002066  005737  001222                  2$:     TST     ACTIV           ;TEST IF ENTRY IS FROM A TEST
002072  001146                                  BNE     TTY1G           ;BRANCH IF IT IS
002074  004737  002604                          JSR     PC,SCANDH       ;LOOK FOR INPUT
002100  004737  002660                          JSR     PC,SETEPM       ;SET TERMINAL DATA
002104  042700  177600                  TTY1B:  BIC     #177600,R0      ;SAVE ONLY CHAR
002110  010037  001124                          MOV     R0,TEMPCH       ;GET CHAR
002114  020027  000040                  1$:     CMP     R0,#40          ;CHECK IF CHAR IS A SPACE
002120  001002                                  BNE     2$              ;BRANCH IF NOT
002122  104013                                  READ                    ;SPACE, LOOP WAITING FOR NEXT CHAR
002124  000773                                  BR      1$              ;GOT ONE
002126  012700  000036                  3$:     MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
002132  104003                                  DELAY
002134  013700  001124                          MOV     TEMPCH,R0       ;GET CHAR
002140  104021                                  ECHO                    ;ECHO CHAR
002142  104020                                  TESTC                   ;GO TEST CHAR
002144  000421                                  BR      11$             ;ERROR IN CHAR
002146  010005                                  MOV     R0,R5           ;OK, SAVE DIGIT IN R5, POS 5-3
002150  006305                                  ASL     R5
002152  006305                                  ASL     R5
002154  006305                                  ASL     R5
002156  104013                                  READ                    ;GO WAIT FOR NEXT CHAR
002160  020027  000040                  3$:     CMP     R0,#40          ;CHECK IF A SPACE
002164  001002                                  BNE     4$              ;BRANCH IF NOT A SPACE
002166  104013                                  READ                    ;WAIT FOR CHAR
002170  000773                                  BR      3$              ;GOT ONE, ECHO IT
002172  012700  000036                  4$:     MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
002176  104003                                  DELAY
002200  013700  001124                          MOV     TEMPCH,R0       ;GET CHAR
002204  104021                                  ECHO                    ;ECHO CHAR
002206  104020                                  TESTC                   ;GO CHECK CHAR
002210  000555                          11$:    BR      NG              ;ERROR IN CHAR
002212  060005                                  ADD     R0,R5           ;OK,R5 IS NOW OCTAL TEST NO.
002214  104013                                  READ                    ;GO WAIT FOR TERMINATOR
```

```
002216  020027  000040          5$:   CMP     R0,#40          ;CHECK IF A SPACE
002222  001002                        BNE     5$              ;BRANCH IF NOT A SPACE
002224  104013                        READ                    ;SPACE,WAIT SOME MORE
002226  000773                        BR      5$              ;GOT ONE  ECHO IT
002230  012700  000036          6$:   MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
002234  104003                        DELAY
002236  013700  001124                MOV     TEMPCH,R0       ;GET CHAR
002242  104021                        ECHO                    ;ECHO CHAR
002244  042700  000040                BIC     #BITS,R0        ;ALLOW LOWER CASE OR UPPER CASE
002250  020027  000114                CMP     R0,#114         ;IS IT AN "L"
002254  001413                        BEQ     7$              ;BRANCH IF YES
002256  020027  000123                CMP     R0,#123         ;NO, IS IT AN "S"
002262  001414                        BEQ     8$              ;BRANCH IF YES
002264  023727  001124  000056        CMP     TEMPCH,#56      ;NO, IS IT A "."
002272  001124                        BNE     NG              ;NO, ERROROR
002274  012737  000001  001104        MOV     #1,CNTLSW       ;SET BIT 0 ONLY IN CNTLSW
002302  000407                        BR      9$
002304  012737  004001  001104  7$:   MOV     #4001,CNTLSW    ;SET BITS 11 AND 0
002312  000403                        BR      9$
002314  012737  000401  001104  8$:   MOV     #401,CNTLSW     ;SET BITS 8 AND 0
002322  104017                  9$:   CLEAN                   ;CLEAN UP
002324  012706  001100                MOV     #SPBOT,SP       ;RESET SP
002330  010500                        MOV     R5,R0           ;TEST NO TO R0
002332  020027  000040                CMP     R0,#40          ;CHECK IF TEST NO. IS EQ OR GT 40
002336  103102                        BHIS    NG              ;ERROR IF YES
002340  020027  000020                CMP     R0,#20          ;CHECK IF THIS IS AN ECHO TEST
002344  103406                        BLO     10$             ;BRANCH IF NOT
002346  020027  000030                CMP     R0,#30          ;OPTION TEST?
002352  103003                        BHIS    10$             ;ALLOW LOOP ON OPTION TEST
002354  012737  000001  001104        MOV     #1,CNTLSW       ;YES, FORCE TO ONE TIME ONLY
002362  006300                  10$:  ASL     R0              ;TEST NO. * 2
002364  016037  003156  001110        MOV     PRGTAB(R0) NXTST;ADDR OF TEST TO NXTST
002372  001464                        BEQ     NG              ;BRANCH IF ILLEGAL TEST
002374  104016                        FORWD                   ;SET UP TEST PARAMETERS
002376  012737  000001  001222        MOV     #1,ACTIV        ;SET TEST ACTIVE IND
002404  000177  176510                JMP     @CURTST         ;GO TO TEST
002410  017700  176524          TTY1G: MOV    @NRCRA,R0       ;TEST ACTIVE,CHECK INPUT FROM DH11
002414  100040                        BPL     TTY1L           ;BRANCH IF NO DATA
002416  010004                        MOV     R0,R4           ;DATA, SAVE IT
002420  000300                        SWAB    R0              ;RIGHT JUSTIFY LINE NO.
002422  042700  177760                BIC     #177760,R0      ;CLEAR ALL BUT LINE NO.
002426  020037  001156                CMP     R0,LINENO       ;CHECK IF LINE NO. IS SAME AS TEST LINE
002432  001366                        BNE     TTY1G           ;NOT SAME, SEE IF ANY MORE IN SILO
002434  010400                        MOV     R4,R0           ;LINES ARE THE SAME, GET CHAR
002436  042700  177600                BIC     #177600,R0      ;SAVE 7 BITS OF CHAR
002442  020027  000177                CMP     R0,#177         ;CHECK IF A RUBOUT
002446  001360                        BNE     TTY1G           ;NOT A RUBOUT, SEE IF ANY MORE
002450  012706  001100          TTY1H: MOV    #SPBOT,SP       ;RESET STACK
002454  012737  000001  001104        MOV     #1,CNTLSW       ;CLEAR BITS 11 AND 8
002462  012700  000036                MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
002466  104003                        DELAY
002470  104002                        TYPEM                   ;OUTPUT MESSAGE
002472  017257                        MESG3
002474  005037  001222                CLR     ACTIV           ;CLEAR TEST ACTIVE STATE
```

```
002500  005737  001224                  TST     IOSW            ;DJ11 OR DH11 ?
002504  001402                          BEQ     1$              ;BRANCH IF DH11
002506  000137  002024                  JMP     TTYJA           ;WAIT FOR NEXT TEST FROM DJ11
002512  000137  002054          1$:     JMP     TTY1A           ;WAIT FOR NEXT TEST FROM DH11
002516  032737  004000  001104  TTY1L:  BIT     #BIT11,CNTLSW   ;CHECK IF LOOP ON TEST
002524  001401                          BEQ     1$              ;BRANCH IF NO LOOP
002526  000002                          RTI                     ;GO LOOP ON TEST
002530  032737  000400  001104  1$:     BIT     #BIT8,CNTLSW    ;CHECK IF LOOP ON SEQUENCE
002536  001744                          BEQ     TTY1H           ;BRANCH IF NO
002540  000137  001772                  JMP     CHAINY          ;GO LOOP ON SEQUENCE
002544  012700  000036          NG:     MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
002550  104003                          DELAY
002552  112700  000077                  MOVB    #77,R0          ;"?" TO TEMPCH
002556  104021                          ECHO                    ;PRINT A ?
002560  000733                          BR      TTY1H           ;TRY AGAIN FROM DH11


                                ;**********
                                ;
                                ;FORWARD--THIS ROUTINE TRANSFERS THE 2 ARGUMENTS
                                ;        FROM THE TEST ROUTINE. THEY ARE:
                                ;        1- ROUTINE NUMBER
                                ;        2- ADDRESS OF NEXT TEST
                                ;
                                ;**********

002562  013705  001110          $FORWD: MOV     NXTST,R5        ;ADDR OF NEXT TEST TO R5
002566  012537  001106                  MOV     (R5)+,RTNNO     ;GET NUMBER OF NEXT TEST
002572  012537  001110                  MOV     (R5)+,NXTST     ;GET ADDR OF FOLLOWING TEST
002576  010537  001120                  MOV     R5,CURTST       ;ENTRY POINT TO TEST IN CURTST
002602  000002                          RTI                     ;EXIT


                                ;**********
                                ;
                                ;SCANDH - ROUTINE TO SCAN DH CHANNELS LOOKING FOR INPUT
                                ;
                                ;**********

002604  013701  001230          SCANDH: MOV     DHCNT,R1        ;COUNT OF DH11'S TO R1
002610  005037  001154                  CLR     CNTDH           ;CLEAR DH11 POSITION COUNTER
002614  013700  001134                  MOV     DHADR,R0        ;ADDR OF FIRST DH11 TO R0
002620  005720                          TST     (R0)+           ;ADDR OF NRCRA
002622  010037  001140                  MOV     R0,NRCRA        ;SET UP NRCRA ADDRESS
002626  017700  176306          1$:     MOV     @NRCRA,R0       ;GET NEXT CHAR FROM SILO
002632  100410                          BMI     2$              ;BRANCH IF DATA IS PRESENT
002634  005301                          DEC     R1              ;DECREMENT COUNT OF DH11'S
002636  001762                          BEQ     SCANDH          ;START OVER IF ALL DONE
002640  062737  000020  001140          ADD     #20,NRCRA       ;SET UP ADDR FOR NEXT DH11
002646  005237  001154                  INC     CNTDH           ;INC DH11 POSITION COUNTER
002652  000765                          BR      1$              ;GO CHECK NEXT DH11 ON BUS
002654  010004                  2$:     MOV     R0,R4           ;SAVE LINE NO. AND CHAR
002656  000207                          RTS     PC              ;RETURN
```

```
                                        ;**********
                                        ;
                                        ;SETERM - THIS ROUTINE IS USED TO SET UP THE REGISTER ADDRESSES OF
                                        ;         THE TERMINAL CURRENTLY UNDER TEST.
                                        ;
                                        ;**********

[illegible]  [illegible]  001140        SETERM:  [illegible]     MACRA,R1        ;GET ADDR OF CURRENT MACRA
[illegible]  [illegible]                         [illegible]     (R1),R1         ;CHL ADDR OF SCR
[illegible]  [illegible]  001136                 MOV     R1,SCR          ;STORE SCR ADDR
[illegible]  [illegible]                         [illegible]     (R1)+,(R1)+     ;ADD 4
[illegible]  [illegible]  001142                 MOV     R1,LPR          ;STORE LPR ADDR
[illegible]  [illegible]                         [illegible]     (R1)+           ;ADD 2
[illegible]  [illegible]  001144                 MOV     R1,CARA         ;STORE CARA ADDR
[illegible]  [illegible]                         [illegible]     (R1)+           ;ADD 2
[illegible]  [illegible]  001146                 MOV     R1,BYCR         ;STORE BYCR ADDR
[illegible]  [illegible]                         [illegible]     (R1)+           ;ADD 2
[illegible]  [illegible]  001150                 MOV     R1,BAR          ;STORE BAR ADDR
[illegible]  [illegible]                         [illegible]     (R1)+,(R1)+     ;ADD 4
[illegible]  [illegible]  001152                 MOV     R1,SSR          ;STORE SSR ADDR
[illegible]  [illegible]                         [illegible]     R0              ;RIGHT JUSTIFY LINE NO.
[illegible]  [illegible]  [illegible]            [illegible]     #177760,R0      ;SAVE ONLY LINE NO.
[illegible]  [illegible]  [illegible]            [illegible]     R0,@SCR         ;SET UP SCR
[illegible]  [illegible]  [illegible]            [illegible]     #CH11DD,R1      ;GET ADDR OF VP TABLE
[illegible]  [illegible]  001154        1$:      [illegible]     CNTCH           ;CHL WHICH DH11 IS UNDER TEST
[illegible]  [illegible]                         [illegible]     [illegible]     ;BRANCH IF THIS IS THE ONE
[illegible]  [illegible]  000000                 [illegible]     #DD,R1          ;REPOSITION R1 IN VP TABLE
[illegible]  [illegible]  001156                 [illegible]     CNTCH           ;DEC DH11 POSITION COUNTER
[illegible]  [illegible]                         [illegible]     [illegible]
[illegible]  [illegible]  001158        2$:      [illegible]     R0,LINENO       ;SAVE LINE NO.
[illegible]  [illegible]                         [illegible]     R0              ;CREATE POINTER TO TABLE
[illegible]  [illegible]                         [illegible]     R0,R1           ;ADD LINE NO. TO VP TAB PTR
[illegible]  [illegible]                         [illegible]     [illegible]     ;PUT VP INTO LPR
[illegible]  [illegible]  [illegible]            [illegible]     [illegible]     ;CHL ENTRY FOR LINE NO. TABLE
[illegible]  [illegible]  [illegible]            [illegible]     [illegible]     ;IF LINE NO. HAS NOT BEEN
[illegible]  [illegible]                         [illegible]     [illegible]     ;DETERMINE LINE NO ADD TABLE
[illegible]  [illegible]                         [illegible]     [illegible]     ;ADD IN
```

```
                                        ;**********
                                        ;
                                        ;SCANDJ - ROUTINE TO SCAN CHANNELS LOOKING FOR INPUT
                                        ;
                                        ;********
003034  013700  000000  SCANDJ: MOV     DJCNT,R1        ;COUNT OF DJ11 TO R1
003038  005037  000054          CLR     CNTPOS          ;CLEAR POS. COUNTER
003036  012700  160012          MOV     #160012,R0      ;ADDR OF FIRST RECV BUFFER TO R0
003042  012703  000000          MOV     #ELTAB,R3       ;ADDR OF ELTAB TO R3
003046  005000  1:              CLR     R0              ;GET RECV BUFFER
003050  100413                  BMI     3$              ;BRANCH IF DATA
003052  005301  2:              DEC     R1              ;NO DATA, DEC COUNT OF DJ11
003054  001764                  BEQ     SCANDJ          ;START OVER IF ALL CHECKED
003056  062700  000010          ADD     #10,R0          ;ADDR OF NEXT RBUF
003062  005003                  CLR     (R3)+           ;INCREMENT ELTAB POINTER
003064  005237  000054          INC     CNTPOS          ;INCREMENT POS COUNTER
003070  000765                  BR      1$              ;DO NEXT DJ11
003072  010400  3:              MOV     (R0),R4         ;GET CONTENTS OF RBUF
003074  000000                  HALT
003076  042704  177760          BIC     #177760,R4      ;SAVE ONLY THE LINE NO.
003102  010437  000056          MOV     R4,LINENO       ;SAVE LINE NO.
003106  006304                  ASL     R4              ;LINE NO.*2
003110  062704  000000          ADD     #BITTAB,R4      ;ABS. ADDR OF LINE MASK
003114  010400                  MOV     R4,R0           ;SAVE LINE MASK
003116  005000                  CLR     R0              ;CHECK IF LINE EXISTS
003120  100004                  BPL     4$              ;BRANCH IF NOT
003122  005014                  CLR     (R4)            ;RESTORE BIT IN TABLE
003124  000000                  HALT                    ;BRANCH IF NULL
003126  000000  4$:             RTS     PC              ;RETURN

                                        ;**********
                                        ;
                                        ;SETCSR - THIS ROUTINE IS USED TO SET UP THE PARAMETERS
                                        ;         FOR THE DJ11 TERMINAL CURRENTLY UNDER TEST.
                                        ;
                                        ;**********
000140  010046  SETCSR: MOV     R0,-(SP)        ;SAVE ADDR FOR READ & TT 1-
000142  016700  000136          MOV     CSR,R0          ;GET ADDR OF CSR
000146  010037  000136          MOV     R0,CSR          ;ADDR OF CURRENT CSR
000150  062700  000004          ADD     #4,R0           ;CAL ADDR OF TCR
000154  010037  000140          MOV     R0,TCR          ;SAVE ADDR OF CURRENT TCR
000160  062700  000004          ADD     #4,R0           ;CAL ADDR OF TBUF
000164  010037  000142          MOV     R0,TBUF         ;SAVE ADDR OF CURRENT TBUF
000170  012600                  MOV     (SP)+,R0        ;RESTORE R0 FROM STACK
000172  000207                  RTS     PC              ;RETURN
```

```
                                        .SBTL  TEST ADDRESS TABLE

                                PRGTAB:
                                                        ;DATA PATH TEST
                                                        ;PRINTER CHARACTER TEST
                                                        ;NON-PRINTING CHARACTER TEST
                                                        ;CARRIAGE RETURN TEST
                                                        ;MULTIPLE LINE FEED TEST
                                                        ;SINGLE LINE FEED TEST
                                                        ;BACKSPACE TEST
                                                        ;O.ERPRINT TEST
                                                        ;PRINTING FREQUENCY SWEEP TEST
                                                        ;RIBBON FEED TEST
                                                        ;PRINTER BELL TEST
                                                        ;SPARE FOR ADDITIONAL PRINTER TEST
                                                        ;SPARE FOR ADDITIONAL PRINTER TEST
                                                        ;SPARE FOR ADDITIONAL PRINTER TEST
                                                        ;SPARE FOR ADDITIONAL PRINTER TEST
                                                        ;LIFE TEST

                                                        ;CHARACTER ECHO TEST
                                                        ;LINE ECHO TEST, FAST RATE
                                                        ;LINE ECHO TEST, SLOW RATE
                                                        ;CHARACTER/CODE ECHO TEST
                                                        ;SELECTIVE PATTERN ECHO TEST
                                                        ;BELL ECHO TEST
                                                        ;SPARE FOR ADDITIONAL ECHO TESTS
                                                        ;SPARE FOR ADDITIONAL ECHO TESTS

                                                        ;SECONDARY CHARACTER SET OPTION
                                                        ;SELECTIVE ADDRESSING OPTION
                                                        ;AUTO ANSWER BACK OPTION
                                                        ;FORM FEED OPTION
                                                        ;HORIZONTAL TAB OPTION
                                                        ;VERTICAL TAB OPTION
                                                        ;SPARE FOR ADDITIONAL OPTION TEST
                                                        ;SPARE FOR ADDITIONAL OPTION TEST
```

```
                                        .SBTTL  EMT TRAP DECODER

                                ;**********
                                ;
                                ;EMTINT--SERVICE ROUTINES FOR TRAPS THROUGH LOC. 30 AND 34.
                                ;
                                ;**********

003306  011646                  EMTINT: MOV     @SP,-(SP)       ;GET SAVED PC
003310  162716  000002                  SUB     #2,@SP          ;DECREMENT PC BY TWO
003314  011616  000000                  MOV     @(SP),@SP
003320  162627  000022                  CMPB    @SP,#22         ;CHECK THAT CALL IS WITHIN LIMITS
003324  101402                          BLOS    3$              ;BRANCH IF OK
003326  000000                  1$:     HALT
003330  000776                          BR      1$
003332  006116                  3$:     ROR     @SP             ;EMT ARGUMENT *2
003334  042716  177001                  BIC     #177001,@SP     ;REMOVE 7 MSB
003340  062716  003346                  ADD     #EMTTA2,@SP     ;FORM EMT RTN ADDRESS
003344  011616  000000                  MOV     @(SP),@SP
003350  005046                          CLR     -(SP)           ;CLEAR PSW
003352  012746  003330                  MOV     #3$,-(SP)
003356  000002                          RTI
003360  000138                  3$:     JMP     @(SP)+          ;GO TO EMT ROUTINE

003362  000377                  EMTTA3: TYP                     ;MESSAGE OUTPUT ROUTINE
003364  000400                          XTRAN                   ;COMMON TEST EXIT
003366  000402                          TYPEY                   ;MESSAGE OUTPUT ROUTINE, MULTI DEVICES
003370  000404                          DELAY                   ;DELAY ROUTINE
003372  000406                          TTI                     ;DH11 CONSOLE TERMINAL CONTROL
003374  000410                          TTO                     ;DZ11 CONSOLE TERMINAL CONTROL
003376  000412                          CRLF                    ;CARRIAGE RETURN, ALL TERMINALS
003400  000414                          SCRLF                   ;CARRIAGE RETURN, CONSOLE TERMINAL
003402  000416                          LF                      ;LINE FEED ONLY, ALL TERM'S
003404  000420                          PRTC                    ;PRINT CHAR, ALL TERM'S
003406  000422                          PRHDR                   ;PRINT TEST HEADER
003410  000424                          READC                   ;READ CHAR
003412  000426                          CR                      ;CARRIAGE RETURN ONLY, ALL TERM'S
003414  000430                          BTOASC                  ;BINARY TO ASCII CONVERSION
003416  000432                          PCRLF
003420  000434                          COLFRM
003422  000436                          CHECK                   ;CHECK CHAR
003424  000440                          PRTCC                   ;PRINT CHAR, CONSOLE ONLY
003426  000442                          CHECC                   ;CHECK IF READY
```

```
                                                .SBTTL  COMMON ROUTINES

                                        ;**********
                                        ;
                                        ;ECHO-- THIS ROUTINE ECHOS CHARACTERS ON THE TERMINAL UNDER TEST.
                                        ;
                                        ;**********

003400  005737  001234          $ECHO:  TST     ICSW            ;DJ11 OR DH11
003404  001006                          BNE     ECHODJ          ;BRANCH IF DJ11
003406  037737  175536  001160          BIT     @BAR,MASK       ;CHECK IF OK TO SEND TO DH11
003414  001371                          BNE     $ECHO           ;NO,WAIT UNTIL OK
003416  010037  001124                  MOV     R0,TEMPCH       ;CHAR INTO TEMPCH
003422  013737  001156  175506          MOV     LINENO,@SCR     ;SET LINE NUMBER
003430  012737  001124  175506          MOV     #TEMPCH,@CARA   ;OK,PUT ADDR OF CHAR INTO CARA
003436  012737  177777  175502          MOV     #-1,@BYCR       ;SET CHAR COUNT TO 1
003444  012737  000020  175500          MOV     #20.,@SSR       ;SET SILO OVERFLOW TO 1S
003452  053737  001160  175470          BIS     MASK,@BAR       ;SET TRANSMIT BIT
003460  000002                          RTI                     ;RETURN
003462  013737  001160  175550  ECHODJ: MOV     MASK,@TCR       ;SET LINE NO. IN TCR
003470  052737  000400  175540  1$:     BIS     #BITB,@CSR      ;START TRANSMITTER SCANNER
003476  005737  175534          2$:     TST     @CSR            ;CHECK IF OK
003502  100375                          BPL     2$              ;BRANCH IF NOT READY
003504  017704  175532                  MOV     @TBUF,R4        ;GET CONTENTS OF TBUF
003510  000304                          SWAB    R4              ;RIGHT JUSTIFY LINE NO.
003512  042704  177760                  BIC     #177760,R4      ;SAVE ONLY THE LINE NO.
003516  020437  001156                  CMP     R4,LINENO       ;BE SURE IT'S THE LINE UNDER TEST
003522  001362                          BNE     1$              ;BRANCH IF NOT
003524  110027  175516                  MOVB    R0,@TBUF        ;LOAD CHAR TO PRINT
003530  110037  001124                  MOVB    R0,TEMPCH       ;SAVE CHAR IN TEMPCH
003534  000002                          RTI                     ;RETURN

                                        ;**********
                                        ;
                                        ;INRDY-- CHECKS IF ANY INPUT FROM TERMINAL UNDER TEST.
                                        ;       RETURN VIA PC IF NO CHAR OR VIA PC+2 IF VALID CHAR.  CHAR WILL
                                        ;       BE IN BITS 6-0 OF TEMPCH.
                                        ;       CALLING SEQUENCE:
                                        ;         INRDY
                                        ;         NO CHAR RETURN
                                        ;         VALID CHAR RETURN
                                        ;
                                        ;**********

003536  017700  175576          $INRDY: MOV     @NRCRA,R0       ;GET CHAR
003542  100015                          BPL     1$              ;BRANCH IF NO CHAR
003544  010037  001124                  MOV     R0,TEMPCH       ;SAVE CHAR
003550  000300                          SWAB    R0              ;RIGHT JUSTIFY LINE NO.
003552  042700  177760                  BIC     #177760,R0      ;SAVE ONLY LINE NO.
003556  020037  001156                  CMP     R0,LINENO       ;CHECK IF SAME AS LINE UNDER TEST
003562  001365                          BNE     $INRDY          ;BRANCH IF NOT LINE UNDER TEST
003564  042737  177600  001124          BIC     #177600,TEMPCH  ;SAVE ONLY THE CHAR
003572  062716  000002                  ADD     #2,@SP          ;SET UP RETURN ADDR.
003576  000002                  1$:     RTI                     ;RETURN
```

```
                                        ;*********
                                        ;
                                        ;READ-- THIS ROUTINE READS INPUT FROM THE TERMINAL UNDER TEST ONLY
                                        ;       IF SW13 IS SET. OTHERWISE ALL TERMINALS ARE SCANNED FOR INPUT
                                        ;       AND TERMINAL WHERE CHARACTER IS RECEIVED IS SET AS THE
                                        ;       TERMINAL UNDER TEST.
                                        ;
                                        ;*********

003600  032777  020000  175436  $READ:  BIT     #BIT13,$SR      ;SW13 SET?
003606  001413                          BEQ     1$              ;BRANCH IF NOT
003610  [illegible]  175324             MOV     @MRCPA,R0       ;GET CHAR
003614  100371                          BPL     $READ           ;NO CHAR, WAIT FOR ONE
003616  010004                          MOV     R0,R4           ;VALID DATA
003620  000304                          SWAB    R4              ;RIGHT JUSTIFY LINE NO.
003622  042704  177760                  BIC     #177760,R4      ;SAVE ONLY LINE NO.
003626  020437  001138                  CMP     R4,LINENO       ;CHECK IF SAME AS LINE UNDER TEST
003632  001362                          BNE     $READ           ;BRANCH IF NOT
003634  000424                          BR      4$              ;CONTINUE
003636  010146                  1$:     MOV     R1,-(SP)        ;SAVE R1
003640  010246                          MOV     R2,-(SP)
003642  010346                          MOV     R3,-(SP)
003644  010446                          MOV     R4,-(SP)
003646  005737  001334                  TST     IOSW            ;DH OR DJ?
003652  001005                          BNE     2$              ;BRANCH IF DJ
003654  004737  002604                  JSR     PC,SCANDH       ;LOOK FOR INPUT
003660  004737  002660                  JSR     PC,SETERM       ;SETUP TERMINAL UNDER TEST
003664  000404                          BR      3$              ;CONTINUE
003666  004737  003016          2$:     JSR     PC,SCANDJ       ;LOOK FOR DJ INPUT
003672  004737  003124                  JSR     PC,SETDJ        ;SETUP DJ UNDER TEST
003676  012604                  3$:     MOV     (SP)+,R4        ;RESTORE REGS
003700  012603                          MOV     (SP)+,R3
003702  012602                          MOV     (SP)+,R2
003704  012601                          MOV     (SP)+,R1
003706  010037  001136          4$:     MOV     R0,PCHAR        ;SAVE CHAR WITH PARITY BIT
003712  113737  001136  001137          MOVB    PCHAR,PARITY+1  ;GET CODE FOR PARITY CHECK
003720  042737  177400  001136          BIC     #177400,PCHAR   ;CLEAR UNWANTED BITS
003726  042700  177600                  BIC     #177600,R0      ;SAVE ONLY THE CHAR
003732  010037  001124                  MOV     R0,TEMPCH
003736  012700  000011                  MOV     #11,R0          ;SET SHIFT COUNT
003742  042737  000377  001135          BIC     #377,PARITY     ;CLEAR PARITY FLAG
003750  005300                  5$:     DEC     R0              ;DECREMENT SHIFT COUNT
003752  001406                          BEQ     6$              ;EXIT IF DONE
003754  106337  001137                  RORB    PARITY+1        ;ROTATE CODE
003760  103373                          BCC     5$              ;CONTINUE IF CARRY BIT WAS A ZERO
003762  105137  001135                  COMB    PARITY          ;COMPLIMENT PARITY FLAG IF BIT WAS ONE
003766  000770                          BR      5$              ;CONTINUE
003770  012700  001124          6$:     MOV     TEMPCH,R0       ;RESTORE R0
003774  000002                          RTI                     ;RETURN
```

H04


```
                                        ;XXXXXXXXXX
                                        ;
                                        ;TYPE-- A COMMON ROUTINE USED TO TYPE MESSAGES ON THE TERMINAL
                                        ;       UNDER TEST ONLY.  THE NULL CHARACTER TERMINATES
                                        ;       THE MESSAGE. CALLED THROUGH AN EMT TRAP.
                                        ;       CALLING SEQUENCE
                                        ;       TYPE
                                        ;       MESG      ;ADDRESS OF MESSAGE
                                        ;
                                        ;XXXXXXXXXX

003776  011601                          TYP:    MOV     @SP,R1          ;GET POINTER TO ADDR. OF MESG.
004000  062716  000002                          ADD     #2,@SP          ;SET UP RETURN ADDRESS
004004  011101                                  MOV     @R1,R1          ;ADDR. OF MESG TO R1
004006  112100                          1$:     MOVB    (R1)+,R0        ;GET CHAR
004010  100402                                  BMI     2$              ;BRANCH IF WANT AUTO CR-LF
004012  001003                                  BNE     3$              ;PRINT CHAR IF NOT NULL
004014  000002                                  RTI                     ;EXIT ON NULL CHAR
004016  104003                          2$:     SCRLF                   ;SEND A CRLF
004020  [illegible]                             BR      1$              ;GET NEXT CHAR
004022  [illegible]                     3$:     ECHO                    ;OUTPUT CHAR
004024  [illegible]                             BR      1$              ;GO GET NEXT CHAR

                                        ;XXXXXXXXXX
                                        ;
                                        ;TYPM---MULTI TYPE-A COMMON ROUTINE TO OUTPUT
                                        ;       A MESSAGE ON ALL DH11S.
                                        ;       THIS ROUTINE IS USED BY
                                        ;       THE PRINTER TESTS TO TYPE HEADINGS.
                                        ;       CALLING SEQUENCE:
                                        ;         TYPEM
                                        ;         MESGAD        ;ADDRESS OF MESSAGE
                                        ;
                                        ;XXXXXXXXXX

004026  011601                          TYPM:   MOV     @SP,R1          ;GET POINTER TO ADDR OF MESG
004030  062716  000002                          ADD     #2,@SP          ;SET UP RETURN ADDRESS
004034  011101                                  MOV     @R1,R1          ;ADDR OF MESG TO R1
004036  112100                          1$:     MOVB    (R1)+,R0        ;GET CHAR
004040  100402                                  BMI     2$              ;BRANCH IF WANT AUTO CR-LF
004042  001003                                  BNE     3$              ;CONTINUE IF NOT NULL
004044  000002                                  RTI                     ;RETURN TO CALLER
004046  104[illegible]                  2$:     CRLF                    ;SEND CR-LF
004050  [illegible]                             BR      1$              ;GET NEXT CHAR
004052  [illegible]                     3$:     PRINTC                  ;PRINT CHAR
004054  [illegible]                             BR      1$              ;GO GET NEXT CHAR.
```

```
;**********
;
;DELAY--A COMMON ROUTINE TO DELAY PROCESSING
;       A GIVEN NUMBER OF MSEC.
;       CALLING SEQUENCE:
;       MOV   #5,R0    ;R0 CONTAINS THE NUMBER OF MSEC DELAY DESIRED
;       DELAY
;
;       THE DELAY IS EFFECTED BY THE EXECUTION OF THE LOOP:
;          1$:  DEC  R1
;               BNE  1$
;
;       SINCE THE EXECUTION TIMES OF THE PDP11 LINE DOES VARY FROM
;       MACHINE TO MACHINE, THE VALUE AT SYMBOLIC LOCATION
;       "TIMER" MUST BE CHANGED TO THE APPROPRIATE VALUE AS SHOWN BELOW
;       BEFORE STARTING THE DIAGNOSTIC. "TIMER" IS INITIALIZED
;        FOR AN PDP-11/40 (TIMER = 554).
;
;MACHINE        05310   35840   15320                    11/45 $ 11/70
;                                                  BIPOLAR       MOS       CORE
;
;LOOP: DEC R1    3.4     .99     2.3                  .30          .51       .95
;      BNE LOOP  2.5     1.76    40.6                 .60          .98      1.13
;  TIME=         5.90US  2.75    41.9                .90USEC      1.48USEC        2.08SEC
;SET TIMER       351     554     314                 2127         1237      755
;
;XXXXXXXXXX
```

```
004056  010146          DLY:    MOV     R1,-(SP)        ;SAVE R1
004060  013701  001102          MOV     TIMER,R1        ;MOV 1MSEC LOOP CNT TO R1
004064  005301                  DEC     R1              ;DECREMENT COUNT
004066  001376                  BNE     1$              ;BRANCH IF NOT ZERO
004070  005300                  DEC     R0              ;DEC NO. OF MSEC DELAY
004072  001373                  BNE     1$              ;DELAY AGAIN IF NOT ZERO
004074  012601                  MOV     (SP)+,R1        ;ALL DONE RESTORE R1
004076  000002                  RTI
```

```
                                        ;**********
                                        ;
                                        ;PFAIL--POWER FAIL ROUTINE
                                        ;       SAVE ALL REGISTERS AND SET RESTART ADDRESS
                                        ;       INTO LOCATION 24
                                        ;
                                        ;RESTART--POWER FAIL RECOVERY
                                        ;        RESTORE ALL REGISTERS AND GO TO START
                                        ;
                                        ;**********

004100  010046                          PFAIL:  MOV     R0,-(SP)
004102  010146                                  MOV     R1,-(SP)
004104  010246                                  MOV     R2,-(SP)
004106  010346                                  MOV     R3,-(SP)
004110  010446                                  MOV     R4,-(SP)
004112  010546                                  MOV     R5,-(SP)
004114  013746  000024                          MOV     24,-(SP)
004120  010637  004134                          MOV     SP,SAVR6        ;SAVE STACK POSITION
004124  012737  004136  000024                  MOV     #RESTRT,24      ;STORE RESTART ADDRESS
004132  000000                                  HALT

004134  000000                          SAVR6:  .WORD   0

004136  104002                          RESTRT: TYPEM                   ;TYPE POWER MESG
004140  004172                                  1$
004142  013706  004134                          MOV     SAVR6,SP        ;RESTORE STACK POINTER
004146  012637  000024                          MOV     (SP)+,24        ;RESTORE PFAIL ADDRESS
004152  012605                                  MOV     (SP)+,R5
004154  012604                                  MOV     (SP)+,R4
004156  012603                                  MOV     (SP)+,R3
004160  012602                                  MOV     (SP)+,R2
004162  012601                                  MOV     (SP)+,R1
004164  012600                                  MOV     (SP)+,R0
004166  000137  001262                          JMP     STARTX

004172  050200  053517  051105  1$:             .ASCIZ  <ACRLF> POWER <ACRLF>
004203  000200                                  .EVEN
```

```
                                        ;**********
                                        ;
                                        ;BINARY TO ASCII CONVERSION (1 TO 5 ASCII CHARACTERS)
                                        ;
                                        ;CALLING SEQUENCE
                                        ;
                                        ;       MOV     ADDRESS OF LOC TO STORE FIRST ASCII CHAR. INTO R0
                                        ;       MOV     BINARY NUMBER TO BE CONVERTED INTO R1
                                        ;       MOV     NUMBER TO BE CONVERTED AS A POWER OF TEN INTO R2
                                        ;       BTOASC
                                        ;
                                        ;**********

004202  010237  004266          $BTASC: MOV     R2,CNVCTR       ;SAVE TEN POWER
004206  006302                          ASL     R2              ;R2*2
004210  062702  004274                  ADD     #ADTENP,R2      ;CALCULATE ADDRESS OF STARTING TEN POWER
004214  014237  004272          1$:     MOV     -(R2),TENPWR    ;POWER OF TEN VALUE TO TENPWR
004220  005037  004270                  CLR     DIGIT           ;CLEAR CURRENT DIGIT
004224  163701  004272          2$:     SUB     TENPWR,R1       ;SUBTRACT TEN POWER FROM BINARY VALUE
004230  103403                          BCS     3$              ;BRANCH IF END
004232  005237  004270                  INC     DIGIT
004236  000772                          BR      2$
004240  063701  004272          3$:     ADD     TENPWR,R1       ;RESTORE SUBTRACTED VALUE
004244  062737  000060  004270          ADD     #60,DIGIT       ;CONVERT (DIGIT) TO ASCII
004252  113720  004270                  MOVB    DIGIT,(R0)+     ;PUT ASCII CHAR INTO USER BUFFER
004256  005337  004266                  DEC     CNVCTR          ;FINISH ALL CHARS CALLED FOR
004262  001354                          BNE     1$              ;BRANCH IF NOT FINISHED
004264  000002                          RTI                     ;RETURN


004266  000000                  CNVCTR: .WORD   0               ;CONVERSION CHARACTER COUNT
004270  000000                  DIGIT:  .WORD   0               ;CONVERTED CHARACTER
004272  000000                  TENPWR: .WORD   0               ;CURRENT TEN POWER

004274  000001  000012  000144  ADTENP: .WORD   1.,10.,100.,1000.,10000.
004302  001750  023420


                                        ;********
                                        ;
                                        ;TESTC-- CHECKS FOR INPUTTED OCTAL DIGIT
                                        ;        BETWEEN A 0 AND A 7 INCLUSIVE
                                        ;
                                        ;********

004306  013700  001124          $TESTC: MOV     TEMPCH,R0       ;GET CHAR
004312  020027  000060                  CMP     R0,#60          ;CHECK IF NUMERIC AND EQ .GT 0
004316  103407                          BLO     1$              ;BRANCH ERROR
004320  020027  000067                  CMP     R0,#67          ;CHECK IF EQ OR LT 7
004324  101004                          BHI     1$              ;BRANCH ERROR
004326  062716  000002                  ADD     #2,@SP          ;SET UP RETURN ADDRESS
004332  042700  177770                  BIC     #177770,R0      ;SAVE ONLY THE DIGIT
004336  000002                  1$:     RTI                     ;NORMAL RETURN
```

```
                                        ;**********
                                        ;
                                        ;SCRLF-- A COMMON ROUTINE TO OUTPUT A CR AND LF TO
                                        ;       THE TEST TERMINAL ONLY.
                                        ;
                                        ;**********

004340  112700  000015          $SCRLF: MOVB    #15,R0          ;SEND A CR
004344  104021                          ECHO                    ;WAIT UNTIL PRINTER IS READY
004346  112700  000012                  MOVB    #12,R0          ;SEND A LF
004352  104021                          ECHO
004354  000002                          RTI                     ;RETURN TO CALLER


                                        ;XXXXXXXXXX
                                        ;
                                        ;CRLF-- ROUTINES TO SEND A CR AND/OR LF TO ALL TERMINALS.
                                        ;
                                        ;XXXXXXXXXX

004356  104014                  $CRLF:  CR                      ;SEND CR
004360  012700  000012          $LF:    MOV     #12,R0          ;LF TO R0
004364  104011                          PRINTC                  ;SEND IT
004366  000002                          RTI                     ;RETURN

004370  012700  000015          $CR:    MOV     #15,R0          ;CR TO R0
004374  104011                          PRINTC                  ;SEND IT
004376  000002                          RTI                     ;RETURN


                                        ;********
                                        ;
                                        ;ROUTINE TO PRINT TEST HEADER
                                        ;
                                        ;********

004400  104002                  $PRHDR: TYPEM                   ;PRINT MESG
004402  017033                          HDRMSG
004404  013700  001106                  MOV     RTNNO,R0        ;GET TEST NUMBER
004410  006200                          ASR     R0              ;GET FIRST DIGIT
004412  006200                          ASR     R0
004414  006200                          ASR     R0
004416  042700  177770                  BIC     #177770,R0      ;MASK FIRST DIGIT
004422  062700  000060                  ADD     #60,R0          ;MAKE ASCII
004426  104011                          PRINTC                  ;PRINT IT
004430  013700  001106                  MOV     RTNNO,R0        ;GET TEST NUMBER AGAIN
004434  042700  177770                  BIC     #177770,R0      ;MASK LAST DIGIT
004440  062700  000060                  ADD     #60,R0          ;MAKE ASCII
004444  104011                          PRINTC                  ;PRINT IT
004446  104006                          CRLF                    ;CR-LF
004450  104010                          LF                      ;BLANK LINE
004452  000002                          RTI                     ;RETURN
```

```
                                        ;**********
                                        ;
                                        ;PRINTC--THIS ROUTINE IS USED TO DRIVE EACH OF THE EXISTING TERMINALS
                                        ;        ON EACH OF THE EXISTING DH11'S( AS DEFINED BY THE SET UP IN ELTAB).
                                        ;        IF IN THE MAINTENANCE MODE SR BIT 13 CONTROLS WHETHER OR NOT
                                        ;        ALL DH11'S ARE DRIVEN OR ONLY THE TERMINAL UNDER TEST.  SET
                                        ;        BIT 13 DOWN TO DRIVE ALL TERMINALS ON ALL DH11'S. SET BIT 13 UP TO
                                        ;        DRIVE ONLY THE TERMINAL UNDER TEST.
                                        ;        EACH TERMINAL IS DRIVEN ONE CHARACTER AT A TIME.
                                        ;        PRINTC WILL LOOP WAITING FOR THE FIRST TERMINAL TO BE READY
                                        ;        ENTER WITH CHAR TO PRINT IN R0.
                                        ;
                                        ;**********

004454  010046                          $PRTC:  MOV     R0,-(SP)        ;SAVE R0
004456  010146                                  MOV     R1,-(SP)        ;SAVE R1
004460  010246                                  MOV     R2,-(SP)        ;SAVE R2
004462  010346                                  MOV     R3,-(SP)        ;SAVE R3
004464  010446                                  MOV     R4,-(SP)        ;SAVE R4
004466  010546                                  MOV     R5,-(SP)        ;SAVE R5
004470  005737  001224                          TST     IOSW            ;DH11 OR DJ11?
004474  001402                                  BEQ     1$
004476  000137  005324                          JMP     PRINTJ          ;GO TO DJ11 ROUTINE
004502  005737  001230                  1$:     TST     DHCNT           ;ANY DH11'S PRESENT?
004506  001562                                  BEQ     12$             ;RETURN IF NONE
004510  032777  020000  174526                  BIT     #BIT13,@SR      ;CHECK IF SR BIT13 IS SET
004516  001445                                  BEQ     6$              ;DRIVE ALL TERMINALS IF NOT SET
004520  104021                                  ECHO                    ;OUTPUT CHAR
004522  104022                                  INRDY                   ;CHECK IF ANY INPUT
004524  000440                                  BR      18$             ;NO,RETURN
004526  023727  001124  000177                  CMP     TEMPCH,#177     ;INPUT,CHECK IF A RUBOUT
004534  001402                                  BEQ     2$
004536  000137  005246                          JMP     ENDITR          ;NO RUBOUT, RETURN
004542  023727  001106  000024          2$:     CMP     RTNNO,#24       ;CHECK IF TEST 24
004550  001004                                  BNE     3$              ;BRANCH IF NOT
004552  012766  012054  000014                  MOV     #TERM,14(SP)    ;SET RETURN ADR
004560  000535                                  BR      12$             ;RETURN TO EXIT TEST PROPERLY
004562  023727  001106  000021          3$:     CMP     RTNNO,#21       ;TEST 21?
004570  001004                                  BNE     4$              ;BRANCH IF NOT
004572  012766  011064  000014                  MOV     #E021B,14(SP)   ;SET RETURN TO EXIT TEST PROPERLY
004600  000525                                  BR      12$             ;RETURN
004602  023727  001106  000022          4$:     CMP     RTNNO,#22       ;TEST 22?
004610  001004                                  BNE     5$              ;CONTINUE IF NOT
004612  012766  011144  000014                  MOV     #E022B,14(SP)   ;SET RETURN ADR
004620  000515                                  BR      12$             ;RETURN TO EXIT TEST PROPERLY
004622  000137  002450                  5$:     JMP     TTY1H           ;GO WAIT
004626  000137  005262                  18$:    JMP     ENDIT
004632  013737  001134  001232          6$:     MOV     DHADR,SCR1      ;INIT ADDR OF FIRST DH11
004640  012705  016560                          MOV     #ELTAB,R5       ;INIT ADDR TO EXISTING TERM TAB
004644  012704  015560                          MOV     #DH1100,R4      ;INIT ADDR TO VP TAB
004650  013703  001230                          MOV     DHCNT,R3        ;INIT DH11 COUNT
004654  012702  000001                  7$:     MOV     #1,R2           ;INIT CURRENT LINE NO.
004660  005001                                  CLR     R1              ;SET UP CURRENT CHANNEL NUMBER
004662  013737  001232  001234                  MOV     SCR1,SCR2       ;SET SCR2 = ADDR OF CURRENT DH11
```

```
004670  062737  000012  001234          ADD     #12,SCR2        ;SET SCR2 = ADDR OF BAR
004676  031502                  8$:     BIT     @R5,R2          ;TEST IF TERMINAL EXISTS
004700  001147                          BNE     17$             ;BRANCH IF NO TERMINAL
004702  037702  174326          9$:     BIT     @SCR2,R2        ;TEST IF OK TO SEND
004706  001375                          BNE     9$              ;TEST AGAIN
004710  062737  000004  001234          ADD     #4,SCR2         ;ADDR OF SILO STATUS
004716  112777  000020  174310          MOVB    #20,@SCR2       ;SET SILO OVERFLOW TO 16
004724  162737  000016  001234          SUB     #16,SCR2        ;SET SCR2 AS ADDR OF SCR
004732  110177  174276                  MOVB    R1,@SCR2        ;PUT CHANNEL NO. INTO SCR
004736  062737  000002  001234          ADD     #2,SCR2         ;SET CHAR BUF ADR
004744  005777  174264                  TST     @SCR2           ;ANY INPUT?
004750  100064                          BPL     16$             ;CONTINUE IF NONE
004752  017737  174256  001124          MOV     @SCR2,TEMPCH    ;GET CHAR
004760  042737  177600  001124          BIC     #177600,TEMPCH  ;MASK CHAR
004766  023727  001124  000177          CMP     TEMPCH,#177     ;CHECK IF RUBOUT
004774  001032                          BNE     14$             ;BRANCH IF NOT RUBOUT
004776  023727  001106  000024          CMP     RTNNO,#24       ;TEST 24?
005004  001004                          BNE     10$             ;BRANCH IF NOT
005006  012766  012054  000014          MOV     #TERM,14(SP)    ;SET RETURN ADR
005014  000517                          BR      ENDITD          ;RETURN TO EXIT TEST PROPERLY
005016  023727  001106  000021  10$:    CMP     RTNNO,#21       ;TEST21?
005024  001004                          BNE     11$             ;BRANCH IF NOT
005026  012766  011064  000014          MOV     #E021B,14(SP)   ;SET RETURN ADR
005034  000507                          BR      ENDITD          ;RETURN TO EXIT TEST PROPERLY
005036  023727  001106  000022  11$:    CMP     RTNNO,#22       ;TEST 22?
005044  001004                          BNE     13$             ;BRANCH IF NOT
005046  012766  011144  000014          MOV     #E022B,14(SP)   ;SET RETURN ADR
005054  000477                  12$:    BR      ENDITD          ;RETURN TO EXIT TEST PROPERLY
005056  000137  002450          13$:    JMP     TTY1H           ;CONTROL
005062  023727  001124  000003  14$:    CMP     TEMPCH,#3       ;CHAR = CONTROL-C ?
005070  001004                          BNE     15$             ;CONTINUE IF NOT
005072  023727  001106  000024          CMP     RTNNO,#24       ;TEST 24?
005100  001465                          BEQ     ENDITD          ;EXIT IF TEST 24
005102  013737  001124  001114  15$:    MOV     TEMPCH,REPT     ;SAVE CHAR FOR TESTS 21 AND 22
005110  010046                          MOV     R0,-(SP)        ;SAVE R0
005112  012700  000036                  MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
005116  104003                          DELAY
005120  012600                          MOV     (SP)+,R0        ;RESTORE R0
005122  062737  000002  001234  16$:    ADD     #2,SCR2         ;SCR2 EQ ADDR OF LPR
005130  011477  174100                  MOV     (R4),@SCR2      ;STORE VP INTO LPR
005134  062737  000002  001234          ADD     #2,SCR2         ;ADD 2 TO ADDDR IN SCR2
005142  010146                          MOV     R1,-(SP)        ;SAVE R1
005144  006301                          ASL     R1              ;FIND TABLE POINTER
005146  006301                          ASL     R1              ;TO STORE CHAR
005150  006301                          ASL     R1              ;FOR THIS CHANNEL
005152  006301                          ASL     R1
005154  060301                          ADD     R3,R1
005156  062701  006007                  ADD     #CHARAC-1,R1
005162  110011                          MOVB    R0,(R1)         ;STORE CHAR
005164  010177  174044                  MOV     R1,@SCR2        ;ADDR OF CHAR INTO CARA
005170  012601                          MOV     (SP)+,R1        ;RESTORE R1
005172  062737  000002  001234          ADD     #2,SCR2         ;ADD 2 TO ADDR IN SCR2
005200  012777  177777  174026          MOV     #177777,@SCR2   ;SET CHAR COUNT EQ 1
005206  062737  000002  001234          ADD     #2,SCR2         ;ADD 2 TO ADDR IN SCR2
```

```
                                                  ;SET LINE BIT IN BAR
                                                  ;INC PTR TO PTSB FOR NEXT TERM
                                                  ;INCREMENT CHANNEL NO.
                                                  ;ROTATE LINE NO. MASK TO NEXT POS.
                                                  ;DO NEXT TERM. ON SAME CHAN
                                                  ;TEST COUNT OF CHANNELS
                                                  ;BRANCH IF ALL DONE
                                                  ;SET UP FOR NEXT CHAN
                                                  ;INC PTR TO EXISTING TERM. TBL.
                                                  ;DO NEXT CHAN

                                                  ;GET CHAR FOR TEST 21 & 22
                                                  ;DELAY FOR HALF DUPLEX

                                                  ;CHAR = CONTROL ?
                                                  ;
                                                  ;
                                                  ;
                                                  ;
                                                  ;
                                                  ;
```

```
                                        ;**********
                                        ;
                                        ;PRINTJ--THIS ROUTINE IS USED TO DRIVE EACH OF THE EXISTING TERMINALS
                                        ;        ON EACH OF THE EXISTING DJ11S' (AS DEFINED BY THE SET UP IN
                                        ;        "ELTAB"  . IF IN THE MAINTENANCE MODE, SR BIT 13 CONTROLS
                                        ;        WHETHER OR NOT ALL DJ11S' ARE DRIVEN OR ONLY THE TERMINAL
                                        ;        UNDER TEST. EACH TERMINAL IS DRIVEN ONE AT A TIME. PRINTJ
                                        ;        WILL LOOP WAITING FOR THE FIRST ONE TO BE READY TO SEND
                                        ;        ENTER WITH CHARACTER TO SEND IN R0.
                                        ;
                                        ;**********

                                PRINTJ: TST     DJCNT           ;ANY DJ11S'
                                        BEQ     ENDIT           ;NO, RETURN
                                        BIT     #BIT13,SSF      ;CHECK IF SR 13 IS SET
                                        BEQ     41              ;CLEAR, DRIVE ALL TERMINALS
                                        TCKO                    ;OUTPUT CHAR TO TERM UNDER TEST
                                        TKCV                    ;CHECK IF ANY INPUT
                                        BR      ENDIT           ;NO, RETURN
                                        CMP     TEMPCH,#177     ;YES CHECK IF A RUBOUT
                                        BNE     ENDITR          ;NO, RETURN
                                        CMP     RTNNO,#24       ;TEST 24?
                                        BNE     1$              ;BRANCH IF NOT
                                        MOV     #TERM,14(SP)    ;SET RETURN ADR
                                        BR      ENDITD          ;RETURN TO EXIT TEST PROPERLY
                                1$:     CMP     RTNNO,#21       ;TEST 21?
                                        BNE     2$              ;BRANCH IF NOT
                                        MOV     #E0215,14(SP)   ;SET RETURN ADR
                                        BR      ENDITD          ;RETURN TO EXIT TEST PROPERLY
                                2$:     CMP     RTNNO,#22       ;TEST 22?
                                        BNE     3$              ;BRANCH IF NOT
                                        MOV     #E0223,14(SP)   ;SET RETURN ADR
                                        BR      ENDITD          ;RETURN TO EXIT TEST PROPERLY
                                3$:     CMP     TTYIM           ;GO WAIT
                                4$:     MOV     #160010,SCR1    ;INIT ADDR OF FIRST DJ11
                                        MOV     #ELTAB,R5       ;INIT ADDR OF EXISTING TERM TAB
                                        MOV     DJCNT,R3        ;NUM OF DJ11S' TO R3
                                5$:     MOV     #1,R2           ;INIT CURRENT LINE NO.
                                        CLR     R0              ;INIT CURRENT CHANNEL NO.
                                6$:     MOV     SCR1,SCR2       ;SET SCR2=ADDR OF CURRENT DJ11
                                        ADD     #4,SCR2         ;SCR2 IS ADDR OF TCR
                                        BIT     (R5),R2         ;TEST IF TERMINAL EXISTS
                                        BEQ     15$             ;BRANCH IF NO TERMINAL
                                        MOVB    R0,@SCR2        ;YES,SET LINE NO. IN TCR
                                        ADD     #4,SCR2         ;SCR2 IS NOW ADDR OF CSR
                                        TSTB    @SCR2           ;CHECK FOR INPUTS
                                        BPL     13$             ;CONTINUE IF NO INPUT
                                        MOV     #0,SCR2         ;SET CHAR BUF REG ADR
                                        MOVB    @SCR2,TEMPCH    ;GET INPUT CHAR
                                        BIC     #177600,TEMPCH  ;MASK CHAR
                                        CMPB    TEMPCH,#11      ;CHECK CHAR
                                        BNE     13$             ;BRANCH IF NOT RUBOUT
                                        CMP     RTNNO,#24       ;TEST 24?
                                        BNE     13$             ;BRANCH IF NOT
```

```
[illegible]  012064 000014          MOV   #TERM,14(SP)      ;SET RETURN ADR
[illegible]                         BR    ENDITC            ;RETURN TO EXIT TEST PROPERL
[illegible]  001126 000021   7$:    CMP   RTNNO,#21         ;TEST 21?
[illegible]                         BNE   9$                ;BRANCH IF NOT
[illegible]  011264 000014          MOV   #EC218,14(SP)     ;SET RETURN ADR
[illegible]                         BR    ENDITC            ;RETURN TO EXIT TEST PROPERL
[illegible]  001126 000022   8$:    CMP   RTNNO,#22         ;TEST 22?
[illegible]                         BNE   10$               ;BRANCH IF NOT
[illegible]  011144 000014          MOV   #EC223,14(SP)     ;SET RETURN ADR
[illegible]                  9$:    BR    ENDITC            ;RETURN TO EXIT TEST PROPERL
[illegible]  002450          10$:   JMP   [illegible]       ;GO TO CONTROL
[illegible]  001124 000003   11$:   CMP   TEMPCH,#3         ;CHAR = CONTROL-C ?
[illegible]                         BNE   12$               ;BRANCH IF NOT
[illegible]  001126 000024          CMP   RTNNO,#24         ;TEST 24?
[illegible]                         BEQ   9$                ;YES, RETURN TO TEST
[illegible]  001124 001114   12$:   MOV   TEMPCH,[illegible] ;SAVE CHAR FOR TESTS 21 & 22
[illegible]                         MOV   R0,-(SP)          ;SAVE R0
[illegible]  000036                 MOV   #30.,R0           ;DELAY FOR HALF DUPLEX
[illegible]                         DELAY
[illegible]                         MOV   (SP)+,R0          ;RESTORE R0
[illegible]                         MOV   #[illegible],R2   ;RESET PROPER ADR FOR PRINT
[illegible]                  13$:   BIS   #[illegible],DSCR2 ;START TRANSMITTER SCANNER
[illegible]                  14$:   TSTB  DSCR2             ;CHECK STATE OF CSR
[illegible]                         BPL   14$               ;BRANCH IF XMITTER NOT READ
[illegible]          001234         MOV   DSCR2,[illegible] ;READY, SCR2 NOW=ADDR OF TBUF
[illegible]                         MOV   @DSCR2,R4         ;GET CONTENTS OF TBUF
[illegible]                         [illegible] R4
[illegible]  177760                 BIC   #177760,R4        ;SAVE ONLY LINE NO.
[illegible]                         CMP   R4,R1             ;BE SURE SCANNER IS ON CORRECT LINE
[illegible]                         BNE   13$               ;BRANCH IF NOT, TRY AGAIN
[illegible]  172334                 MOVB  [illegible],DSCR2 ;LOAD TBUF WITH CHAR
[illegible]                  15$:   [illegible]             ;INCREMENT LINE NO.
[illegible]                         [illegible]             ;SET LINE NO. TO NEXT POSITION
[illegible]                         [illegible]             ;IF CARRY [illegible]
[illegible]                         [illegible]             ;[illegible] COUNT OF [illegible]
[illegible]                         [illegible]             ;BRANCH IF NOT DONE
[illegible]                         [illegible]             ;[illegible]
[illegible]  [illegible] 001232 16$: [illegible]            ;[illegible]
[illegible]                         [illegible]             ;[illegible]
[illegible]                         [illegible]             ;GO [illegible] NEXT [illegible]
[illegible]  001010 000400   CHARAC: .BLKB 256.            ;CHARACTER STORAGE FOR OUTPUT
```

```
                                        .SBTTL  PRINTER TESTS


                                ;++++++++++++
                                ;
                                ;PTC -- DATA PATH TEST---FOUR LINES OF ALTERNATING
                                ;       "*" AND "U" ARE PRINTED, OUT TO THE GIVEN PAPER
                                ;       WIDTH THE PATTERN WILL APPEAR AS FOLLOWS.
                                ;
                                ;       *U*U*U*U*U*U
                                ;       U*U*U*U*U*U*
                                ;       *U*U*U*U*U*U
                                ;       U*U*U*U*U*U*
                                ;
                                ;++++++++++++

                                PTC:    .                       ;TEST NUMBER
                                        PTC:                    ;NEXT TEST
                                        PRTHDR                  ;PRINT TEST HEADER
                                        TNPRM                   ;PRINT COLUMN MESS
                                        HDRCNT
                                        MOV     #"U*,R3         ;SET FIRST CHAR PAIR
                                        MOV     #16.,R2         ;SET LINE COUNT
                                        MOV     R3,R0           ;SET CHAR PAIR
                                        MOV     WIDTH,R1        ;SET COLUMN COUNT
                                        PRINTC                  ;PRINT CHAR
                                        SWAB    R0              ;SET NEXT CHAR
                                        DEC     R1              ;DEC COLUMN COUNT
                                        BNE     3$              ;FINISH LINE
                                        MOV     R3,R0           ;SET NEXT START CHAR
                                        CRLF                    ;CR-LF
                                        DEC     R2              ;DEC LINE COUNT
                                        BGT     2$              ;FINISH TEST
                                        RTS     PC              ;RETURN TO TEST
                                        .END                    ;END OF PTC
```

```
                                        ;XXXXXXXXXX
                                        ;
                                        ;PT1 -- PRINTER CHARACTER TEST --- PRINTS ALL PRINTABLE CHARACTERS
                                        ;
                                        ;XXXXXXXXXX

006464  000001                          PT1:    1                       ;TEST NUMBER
006466  006606                                  PT2                     ;NEXT TEST
006470  104012                                  PRTHDR                  ;PRINT TEST HEADER
006472  012701  000040                  1$:     MOV     #40,R1          ;SPACE TO R1
006476  012702  000100                          MOV     #100,R2         ;@ TO R2
006502  012703  000140                          MOV     #140,R3         ;` TO R3
006506  010100                          3$:     MOV     R1,R0           ;FIRST CHAR TO R0
006510  004737  006556                          JSR     PC,SPSP         ;SEND TWO SPACES
006514  010200                                  MOV     R2,R0           ;SECOND CHAR TO R0
006516  004737  006556                          JSR     PC,SPSP         ;SEND TWO SPACES
006522  012704  000003                          MOV     #3,R4           ;CHAR COUNT TO R4
006526  010300                                  MOV     R3,R0           ;THIRD CHAR TO R0
006530  104011                          3$:     PRINTC                  ;PRINT CHAR
006532  005304                                  DEC     R4              ;THREE TIMES ?
006534  001375                                  BNE     3$              ;BRANCH IF NOT
006536  104006                                  CRLF                    ;CR-LF
006540  122122                                  CMPB    (R1)+,(R2)+     ;SET NEXT CHARS
006542  105723                                  TSTB    (R3)+
006544  020327  000200                          CMP     R3,#200         ;CHECK IF ALL DONE
006550  103766                                  BLO     3$              ;BRANCH IF NOT
006552  104001                                  CHAIN                   ;EXIT TO NEXT TEST
006554  000746                                  BR      1$              ;REPEAT TEST

006556  012704  000003                  SPSP:   MOV     #3,R4           ;SET PRINT COUNT
006562  104011                          1$:     PRINTC                  ;PRINT CHAR
006564  005304                                  DEC     R4              ;DEC PRINT COUNT
006566  001375                                  BNE     1$              ;FINISH CHAR

006570  012700  000040                  SP0:    MOV     #40,R0          ;SPACE TO R0
006574  104011                                  PRINTC                  ;PRINT FIRST SPACE

006576  012700  000040                  SP1:    MOV     #40,R0          ;SPACE TO R0
[illegible]                                     PRINTC                  ;PRINT SECOND SPACE
[illegible]                                     [illegible]             ;RETURN
```

```
                                        ;\\\\\\\\\\
                                        ;
                                        ;PT3 -- NON-PRINTING CHARACTER TEST.  THIS TEST
                                        ;       PRINTS THE OCTAL CODE FOLLOWED BY THE MNEMONIC
                                        ;       OF ALL NON-PRINTING CHARACTERS.  FOLLOWING EACH
                                        ;       MNEMONIC, THE PRINTER IS DRIVEN BY THE NON-PRINTING
                                        ;       CODE (000 THROUGH 037 PLUS 177).  ALL CONTROL CHARACTERS
                                        ;       (INCLUDING THOSE FOR OPTIONS) WILL BE SKIPPED.
                                        ;       REFER TO THE DOCUMENT FOR A LIST OF THOSE PRINTED.
                                        ;
                                        ;\\\\\\\\\\

006600  000002                  PT3:    2                       ;TEST NUMBER
006610  000236                          PT3                     ;NEXT TEST
006612  104012                          PRTHDR                  ;PRINT TEST HEADER
006614  005003                  1$:     CLR     R3              ;NON-PRINTABLE CHAR..NULL IS FIRST
006616  012701  006774                  MOV     #LINE2,R1       ;ADDR OF CHAR STRING TO R1
006622  012702  000003          3$:     MOV     #3,R2           ;NO. OF CHAR GROUPS PER LINE
006626  012704  000010          4$:     MOV     #10,R4          ;NO. OF CHARS PER GROUP
006632  113100                  4$:     MOVB    (R1)+,R0        ;CHAR INTO R0
006634  104011                          PRINTO                  ;PRINT CHAR
006636  005304                          DEC     R4              ;8 CHARS. PRINTED?
006640  001374                          BNE     4$              ;BRANCH IF NOT
006642  022703  000003                  CMP     #3,R3           ;CHAR = STX?
006646  001420                          BEQ     2$              ;YES, SET NEXT CHAR
006650  022703  000004                  CMP     #4,R3           ;CHAR = EOT?
006654  001414                          BEQ     5$              ;YES, SET NEXT CHAR
006656  022703  000033                  CMP     #33,R3          ;CHAR = ESC?
006662  001412                          BEQ     2$              ;YES, SET NEXT CHAR
006664  022703  000007                  CMP     #7,R3           ;CHAR = BELL?
006670  001002                          BNE     5$              ;BRANCH IF NOT A BELL
006672  012703  000040                  MOV     #40,R3          ;SET NEXT CHAR
006676  022703  000040          5$:     CMP     #40,R3          ;IS IT THE LAST?
006702  001003                          BNE     6$              ;BRANCH IF NO
006704  000421                          BR      11$             ;YES, OUTPUT LAST CHAR (177)
006706  005203                  6$:     INC     R3              ;SKIP CHAR
006710  005203                  7$:     INC     R3              ;SKIP CHAR
006712  010300                  8$:     MOV     R3,R0           ;NON-PRINTABLE CHAR TO R0
006714  012704  000003                  MOV     #3,R4           ;A COUNT OF 3 TO R4
006720  104011                  9$:     PRINTO                  ;DRIVE PRINTER WITH NON-PRINTABLE CHAR
006722  005304                          DEC     R4              ;DECREMENT COUNTER
006724  001375                          BNE     9$              ;BRANCH IF NOT ZERO (3 TIMES)
006726  005203                          INC     R3              ;INCREMENT CHAR.CODE
006730  005302                          DEC     R2              ;DEC. GROUPS PER LINE COUNTER (3)
006732  001404                          BEQ     10$             ;BRANCH IF ZERO
006734  004737  006570                  JSR     PC,SP3          ;SEND 3 SPACES
006740  104011                          PRINTO
006742  000731                          BR      3$              ;CONTINUE
006744  104006                  10$:    CRLF                    ;SEND A CR,LF
006746  000725                          BR      3$              ;GO DO NEXT LINE
006750  012704  000003          11$:    MOV     #3,R4           ;A 3 COUNT TO R4
006754  012700  000177                  MOV     #177,R0         ;A DEL TO R0
006760  104011                  12$:    PRINTO                  ;PRINT CHAR
006762  005304                          DEC     R4              ;DECREMENT COUNTER
```

```
 1408  006764  001375                          BNE     1$              ;BRANCH IF NOT ZERO
 1409  006766  104006                          TRAP                    ;SEND A TRAP
 1410  006770  104001                          CHAIN                   ;CHAIN TO NEXT TEST
 1411  006772  000710                          BR      1$              ;REPEAT TEST
 1412
 1413
 1414  006774  030060  020060  047040  LINE2:  .ASCII   /0/0 NULL001 SOH002 STX/
 1415  007002  046125  030060  020061
 1416  007010  051440  044117  030060
 1417  007016  020062  051440  054124
 1418  007024  030060  020066  040440          .ASCII   /006 ACK020 DLE021 001/
 1419  007032  045503  031060  020060
 1420  007040  042040  042514  031060
 1421  007046  020061  042040  030503
 1422  007054  031060  020062  042040          .ASCII   /022 DC2023 DC3024 004/
 1423  007062  031103  031060  020063
 1424  007070  042040  031503  031060
 1425  007076  020064  042040  032103
 1426  007104  031060  020065  047040          .ASCII   /025 NAK026 SYN027 ETB/
 1427  007112  045501  031060  020066
 1428  007120  051440  047131  031060
 1429  007126  020067  042440  041124
 1430  007134  031460  020060  041440          .ASCII   /030 CAN031 EM 032 SUB/
 1431  007142  047101  031460  020061
 1432  007150  042440  020115  031460
 1433  007156  020062  051440  041125
 1434  007164  031460  020064  043040          .ASCII   /034 FS 035 GS 036 RS/
 1435  007172  020123  031460  020065
 1436  007200  043440  020123  031460
 1437  007206  020066  051040  020123
 1438  007214  031460  020067  053440          .ASCII   /037 US 177 DEL/
 1439  007222  020123  030461  020060
 1440  007230  042040  046125    040
 1441  007236                                  .EVEN
```

```
                                        ;\\\\\\\\\\
                                        ;
                                        ;PT3 --  CARRIAGE RETURN TEST --
                                        ;        THE LINE CONSISTS OF A STRING OF O'S AND
                                        ;        X'S.  FIRST, THE O'S ARE PRINTED OUT TO THE LAST
                                        ;        COLUMN WITH A SPACE SEPARATING EACH.  THEN THE
                                        ;        CARRIAGE IS SPACED TO THE FIRST BLANK SPACE, AND
                                        ;        PRINTED AND THE RETURNED TO THE MARGIN.  THIS
                                        ;        PROCESS IS CONTINUE UNTIL ALL SPACES BETWEEN
                                        ;        THE ZEROES HAVE BEEN FILLED.
                                        ;
                                        ;//////////
007236  000003                  PT3:    3                       ;TEST NUMBER
007240  007356                          PT4                     ;NEXT TEST
007242  104012                          PRTHDR                  ;PRINT TEST HEADER
007244  005037  001116          1$:     CLR     SPCNT           ;CLEAR SPACE COUNTER
007250  013701  001112                  MOV     WIDTH,R1        ;POSITION COUNTER TO R1
007254  012700  000117          2$:     MOV     #117,R0         ;"O" TO R0
007260  104011                          PRINTC                  ;PRINT THE "O"
007262  005301                          DEC     R1              ;DECREMENT POSITION COUNTER
007264  001404                          BEQ     3$              ;BRANCH IF 0
007266  004737  006576                  JSR     PC,SPC          ;PRINT A SPACE
007272  005301                          DEC     R1              ;DECREMENT POSITION COUNTER
007274  001367                          BNE     2$              ;BRANCH IF NOT ZERO
007276  104014                  3$:     CR                      ;SEND CR
007300  012737  000001  001116          MOV     #1,SPCNT        ;SPACE COUNTER SET TO 1
007306  013701  001116          4$:     MOV     SPCNT,R1        ;NO. OF SPACES TO R1
007312  004737  006576          5$:     JSR     PC,SPC          ;PRINT SPACE
007316  005301                          DEC     R1              ;DECREMENT SPACE COUNTER
007320  001374                          BNE     5$              ;BRANCH IF NOT ZERO
007322  012700  000130                  MOV     #130,R0         ;"X" INTO R0
007326  104011                          PRINTC                  ;PRINT "X"
007330  104014                          CR                      ;SEND CR
007332  062737  000002  001116          ADD     #2,SPCNT        ;INCREMENT SPACE COUNT BY 2
007340  023737  001116  001112          CMP     SPCNT,WIDTH     ;COMPARE POSITION COUNTER WITH COLM. COUNT
007346  103757                          BLO     4$              ;BRANCH IF LOWER
007350  104010                          LF                      ;SEND LF
007352  104001                          CHAIN                   ;CHAIN TO NEXT TEST
007354  000733                          BR      1$              ;REPEAT TEST
```

```
                                        ;XXXXXXXXXX
                                        ;
                                        ;PT4 -- MULTIPLE LINE FEED TEST -- 63 LINE FEEDS ARE
                                        ;       SENT WITH A REFERENCE LINE AT THE START AND END.
                                        ;       A NUMBER IS PRINTED WHICH INDICATES THE NUMBER OF LINE
                                        ;       FEEDS THAT WILL BE ISSUED BEFORE THE NEXT
                                        ;       NUMBER OR REFERENCE LINE IS PRINTED.
                                        ;
                                        ;XXXXXXXXXX

007356  000004                          PT4:    4                       ;TEST NUMBER
007360  007534                                  PT5                     ;NEXT TEST
007362  104012                                  PRTHDR                  ;PRINT TEST HEADER
007364  012737  000001  001132          1$:     MOV     #1,LFCNT        ;LINE FEED COUNT TO 1
007372  013701  001112                          MOV     WIDTH,R1        ;COLUMN COUNT TO R1
007376  012702  007516                          MOV     #LINE3,R2       ;ADDR OF NUMBER FIELD TO R2
007402  004737  007466                          JSR     PC,REF          ;PRINT REFERENCE LINE
007406  013701  001132                  2$:     MOV     LFCNT,R1        ;LINE FEED COUNT TO R1
007412  104010                          3$:     LF                      ;SEND LF
007414  005301                                  DEC     R1              ;DECREMENT COUNTER
007416  001375                                  BNE     3$              ;BRANCH IF NOT YET 0
007420  006337  001132                          ASL     LFCNT           ;DOUBLE LINE FEED COUNT
007424  022737  000100  001132                  CMP     #BIT6,LFCNT     ;TEST IF COUNT IS 32
007432  001406                                  BEQ     4$              ;BRANCH IF =32. END
007434  112200                                  MOVB    (R2)+,R0        ;NUMBER TO R0
007436  104011                                  PRINTC                  ;PRINT IT
007440  112200                                  MOVB    (R2)+,R0        ;NUMBER TO R0
007442  104011                                  PRINTC                  ;PRINT IT
007444  104014                                  CR                      ;SEND CR
007446  000757                                  BR      2$              ;DRIVE THE LINEFEEDS
007450  013701  001112                  4$:     MOV     WIDTH,R1        ;COLUMN COUNT TO R1
007454  004737  007466                          JSR     PC,REF          ;SEND END REFERENCE LINE
007460  104010                                  LF                      ;SEND LF
007462  104001                                  CHAIN
007464  000737                                  BR      1$              ;REPEAT TEST

007466  112200                          REF:    MOVB    (R2)+,R0        ;NUMBER TO R0
007470  104011                                  PRINTC                  ;PRINT IT
007472  112200                                  MOVB    (R2)+,R0        ;NUMBER TO R0
007474  104011                                  PRINTC                  ;PRINT IT
007476  005741                                  TST     -(R1)           ;DECREASE COUNTER BY 2
007500  012700  000137                          MOV     #137,R0         ;DASH (-) TO R0
007504  104011                          1$:     PRINTC                  ;PRINT IT
007506  005301                                  DEC     R1              ;DECREMENT COLUMN COUNTER
007510  001375                                  BNE     1$              ;BRANCH IF NO ZERO
007512  104014                                  CR                      ;SEND CR
007514  000207                                  RTS     PC              ;RETURN

007516  030460  031060  032060          LINE3:  .ASCII  /0102040816320C/
007524  034060  033061  031063
007532  030060
                                                .EVEN
```

```
                                        ;XXXXXXXXXX
                                        ;PT5--  SINGLE LINE FEED TEST --  TESTS THE LINE FEED
                                        ;       CAPABILITY FROM ALL COLUMNS.
                                        ;XXXXXXXXXX

007534  000005                          PT5:    5                       ;TEST NUMBER
007536  007740                                  PT6                     ;NEXT TEST
007540  104012                                  PRTHDR                  ;PRINT TEST HEADER
007542  013701  001112                  1$:     MOV     WIDTH,R1        ;COLUMN COUNT TO R1
007546  005741                                  TST     -(R1)           ;DECREASE BY 2
007550  012700  000060                          MOV     #60,R0          ;"0" TO R0
007554  104011                          2$:     PRINTC                  ;SEND 0
007556  005301                                  DEC     R1              ;DECREMENT COLUMN COUNTER
007560  001375                                  BNE     2$              ;BRANCH IF NOT ZERO
007562  012700  000062                          MOV     #62,R0          ;SEND A 2
007566  104011                                  PRINTC
007570  104011                                  PRINTC                  ;SEND A SECOND TWO
007572  023727  001112  000200                  CMP     WIDTH,#132.     ;COMPARE COLUMN COUNT
007600  001404                                  BEQ     3$              ;BRANCH IF EQ 132
007602  012700  003410                          MOV     #3410,R0        ;DELAY 1.8 SEC
007606  104003                                  DELAY
007610  000407                                  BR      5$
007612  012700  000063                  3$:     MOV     #63,R0          ;3'S TO R0
007616  012701  000100                          MOV     #100,R1         ;64 TO COUNTER
007622  104011                          4$:     PRINTC                  ;SEND CHARACTER
007624  005301                                  DEC     R1              ;DECREMENT COUNT
007626  001375                                  BNE     4$              ;BRANCH IF NOT ZERO
007630  104006                          5$:     CRLF                    ;SEND A CR,LF
007632  013701  001112                          MOV     WIDTH,R1        ;NO. COLUMNS TO R1
007636  012700  000134                  6$:     MOV     #134,R0         ;BACKSLASH TO R0
007642  104011                                  PRINTC                  ;SEND IT
007644  104010                                  LF                      ;SEND LF
007646  005301                                  DEC     R1              ;DECREMENT COUNTER
007650  001372                                  BNE     6$              ;BRANCH IF NOT ZERO.
007652  104014                                  CR                      ;SEND CR
007654  004737  007702                          JSR     PC,PTSAL        ;SEND REF LINE #1
007660  104006                                  CRLF                    ;SEND A CR,LF
007662  012700  001750                          MOV     #1750,R0        ;DELAY 1 SEC
007666  104003                                  DELAY
007670  004737  007702                          JSR     PC,PTSAL        ;SEND A SECOND REF. LINE
007674  104006                                  CRLF                    ;SEND A CR,LF
007676  104001                                  CHAIN                   ;CHAIN TO NEXT TEST
007700  000720                                  BR      1$              ;REPEAT TEST
007702  013701  001112                  PTSAL:  MOV     WIDTH,R1        ;COLUMN COUNT TO R1
007706  012700  000061                          MOV     #61,R0          ;"1" TO R0
007712  104011                          1$:     PRINTC                  ;PRINT R0
007714  005301                                  DEC     R1              ;DECREMENT COUNTER
007716  001407                                  BEQ     2$              ;BRANCH IF=0
007720  005200                                  INC     R0              ;INCREMENT CHARACTER
007722  020027  000071                          CMP     R0,#71          ;COMP CHAR TO "9"
007726  101771                                  BLOS    1$              ;BRANCH IF LOWER OR SAME
007730  012700  000060                          MOV     #60,R0          ;RESET CHAR TO "0"
007734  000766                                  BR      1$              ;CONTINUE
007736  000207                          2$:     RTS     PC              ;FINISHED, RETURN TO CALLER
```

```
                                        ;XXXXXXXXXX
                                        ;
                                        ;PT6--  BACKSPACE TEST -- A REFERENCE LINE SUCH AS IN
                                        ;       TEST PT5 IS PRINTED.  THE SECOND LINE CONSISTS
                                        ;       OF PRINTING A BACKSLASH, BACKSPACE AND FORWARD
                                        ;       SLASH COMBINATION OUT TO THE GIVEN COLUMN WIDTH.
                                        ;       THIS LINE IS THEN FOLLOWED BY THE SAME TWO REFERENCE
                                        ;       LINES AS PRINTED IN TEST PT5.
                                        ;
                                        ;XXXXXXXXXX

007740  000006                          PT6:    6                       ;TEST NUMBER
007742  010122                                  PT7                     ;NEXT TEST
007744  104012                                  PRTHDR                  ;PRINT TEST HEADER
007746  013701  001112                  1$:     MOV     WIDTH,R1        ;COLUMN COUNT TO R1
007752  005741                                  TST     -(R1)           ;DECREMENT BY 2
007754  012700  000060                          MOV     #60,R0          ;"0" TO R0
007760  104011                          2$:     PRINTC                  ;SEND 0
007762  005301                                  DEC     R1              ;DECREMENT COLUMN COUNTER
007764  001375                                  BNE     2$              ;BRANCH IF NOT ZERO
007766  012700  000062                          MOV     #62,R0          ;"2" TO R0
007772  104011                                  PRINTC                  ;SEND A "2"
007774  104011                                  PRINTC                  ;SEND A SECOND "2"
007776  023727  001112  000204                  CMP     WIDTH,#132.     ;COMPARE COLUMN COUNT
010004  001404                                  BEQ     3$              ;BRANCH IF EQ 132
010006  012700  003410                          MOV     #3410,R0        ;DELAY 1.8 SEC
010012  104003                                  DELAY
010014  000407                                  BR      5$
010016  012700  000063                  3$:     MOV     #63,R0          ;3'S TO R0
010022  012701  000100                          MOV     #100,R1         ;64 TO COUNTER
010026  104011                          4$:     PRINTC                  ;SEND CHAR
010030  005301                                  DEC     R1              ;DECREMENT COUNTER
010032  001375                                  BNE     4$              ;CONTINUE IF NOT DONE
010034  104006                          5$:     CRLF                    ;SEND A CR,LF
010036  013701  001112                          MOV     WIDTH,R1        ;COLUMN COUNT TO R1
010042  012700  000134                  6$:     MOV     #134,R0         ;BACKSLASH TO R0
010046  104011                                  PRINTC                  ;SEND IT
010050  012700  000010                          MOV     #10,R0          ;BACKSPACE TO R0
010054  104011                                  PRINTC                  ;SEND IT
010056  012700  000057                          MOV     #57,R0          ;FORWORD SLASH TO R0
010062  104011                                  PRINTC                  ;SEND IT
010064  005301                                  DEC     R1              ;END OF PAPER
010066  001365                                  BNE     6$              ;BRANCH IF NO
010070  104010                                  LF                      ;SEND LF
010072  104014                                  CR                      ;SEND CR
010074  004737  007702                          JSR     PC,PT5AL        ;SEND REF LINE #1
010100  104006                                  CRLF                    ;SEND A CR,LF
010102  012700  001750                          MOV     #1750,R0        ;DELAY 1 SEC
010106  104003                                  DELAY
010110  004737  007702                          JSR     PC,PT5AL        ;SEND SECOND REF LINE
010114  104006                                  CRLF                    ;SEND A CR,LF
010116  104001                                  CHAIN                   ;CHAIN TO NEXT TEST
010120  000712                                  BR      1$              ;REPEAT TEST
```

```
                                        ;XXXXXXXXXX
                                        ;
                                        ;PT7--   OVERPRINT TEST-- A ROW OF ALTERNATING M'S AND
                                        ;       SPACES ARE PRINTED, OUT TO THE LAST COLUMN AND OVERPRINTED TWICE.
                                        ;       A SECOND LINE OF ALTERNATING SPACES AND "@'S" IS THEN
                                        ;       SENT 3 TIMES AS THE FIRST LINE.  THIS IS FOLLOWED
                                        ;       BY A THIRD AND FINAL LINE OF ALTERNATING '&'
                                        ;       AND SPACES.
                                        ;
                                        ;XXXXXXXXXX

010122  000007                          PT7:    7                       ;TEST NUMBER
010124  010334                                  PT10                    ;NEXT TEST
010126  104012                                  PRTHDR                  ;PRINT TEST HEADER
010130  012703  000002                  1$:     MOV     #2,R3           ;2 COUNT TO R3
010134  013701  001112                  2$:     MOV     WIDTH,R1        ;NO. OF COLUMNS TO R1
010140  012700  000115                  3$:     MOV     #115,R0         ;PRINT M
010144  104011                                  PRINTC
010146  005301                                  DEC     R1              ;END OF LINE
010150  001404                                  BEQ     4$              ;BRANCH IF YES
010152  004737  006576                          JSR     PC,SPC          ;SEND SPACE
010156  005301                                  DEC     R1              ;END OF LINE?
010160  001367                                  BNE     3$              ;BRANCH IF NO
010162  022703  000002                  4$:     CMP     #2,R3           ;TEST R3
010166  001003                                  BNE     6$              ;BRANCH IF NOT FIRST TIME
010170  104014                          5$:     CR                      ;SEND CR
010172  005303                                  DEC     R3              ;DECREASE LINE COUNTER
010174  000757                                  BR      2$              ;REPEAT LINE
010176  005703                          6$:     TST     R3              ;THIRD TIME?
010200  001373                                  BNE     5$              ;BRANCH IF NOT
010202  104006                                  CRLF                    ;NEXT LINE
010204  005723                                  TST     (R3)+           ;REPEAT COUNTER TO R3
010206  013701  001112                  7$:     MOV     WIDTH,R1        ;COLUMN COUNT TO R1
010212  004737  006576                  8$:     JSR     PC,SPC          ;SEND SPACE
010216  005301                                  DEC     R1              ;DECREASE COLUMN COUNT
010220  001405                                  BEQ     9$              ;BRANCH IF 0, END OF LINE
010222  012700  000100                          MOV     #100,R0         ;"@" TO R0
010226  104011                                  PRINTC                  ;SEND IT
010230  005301                                  DEC     R1              ;DECREASE COLUMN COUNT
010232  001367                                  BNE     8$              ;BRANCH IF NOT 0 (NOT END)
010234  022703  000002                  9$:     CMP     #2,R3           ;END OF LINE, FIRST TIME?
010240  001003                                  BNE     11$             ;BRANCH IF NOT
010242  104014                          10$:    CR                      ;SEND CR
010244  005303                                  DEC     R3              ;DECREASE LINE COUNTER
010246  000757                                  BR      7$              ;REPEAT LINE
010250  005703                          11$:    TST     R3              ;TEST IF THIRD REPEAT
010252  001373                                  BNE     10$             ;BRANCH IF NOT
010254  104006                                  CRLF                    ;DO NEXT LINE
010256  005723                                  TST     (R3)+           ;LINE REPEAT COUNTER TO R3
010260  013701  001112                  12$:    MOV     WIDTH,R1        ;COLUMN COUNT TO R1
010264  012700  000046                  13$:    MOV     #46,R0          ;"&" TO R0
010270  104011                                  PRINTC                  ;PRINT IT
010272  005301                                  DEC     R1              ;DECREASE COLUMN COUNT
010274  001404                                  BEQ     14$             ;BRANCH IF END
```

```
010276  004737  006576                  JSR     PC,SPC          ;SEND SPACE
010302  005301                          DEC     R1              ;DECREASE COLUMN COUNT
010304  001367                          BNE     13$             ;BRANCH IF NOT END
010306  022703  000002          14$:    CMP     #2,R3           ;TEST IF FIRST TIME
010312  001003                          BNE     16$             ;BRANCH IF =2, FIRST TIME
010314  104014                  15$:    CR                      ;CARRIAGE RETURN
010316  005303                          DEC     R3              ;DECREASE REPEAT COUNTER
010320  000757                          BR      12$             ;PRINT LINE AGAIN
010322  005703                  16$:    TST     R3              ;TEST IF END, R3=0
010324  001373                          BNE     15$             ;BRANCH IF NOT END
010326  104006                          CRLF                    ;SEND CR,LF
010330  104001                          CHAIN                   ;CHAIN TO NEXT TEST
010332  000676                          BR      1$              ;REPEAT TEST
```

```
;
;PT10-- PRINTING FREQUENCY TEST-- 120 H'S ARE PRINTED ON 4 LINES
;       30 PER LINE. THE TEST IS SUCH THAT BETWEEN THE FIRST AND SECOND
;       "H" A 30 MSEC DELAY IS INTRODUCED. THIS DELAY IS THEN INCREASED
;       BETWEEN CHARACTERS OUT TO 60 CHARACTERS IN AN EXPONENTIAL
;       MANNER. THE DELAY IS THEN DECREASED IN THE SAME MANNER OUT TO THE
;       120TH CHARACTER. THIS DELAY IS CALCULATED AS FOLLOWS:
;
;       NEW DELAY = OLD DELAY (+ OR -) (OLD DELAY / 16 + OLD DELAY / 128)
;
;

PT10:   .TEST                   ;TEST NUMBER
        .NEXT                   ;NEXT TEST
        .PRINT/CR               ;PRINT TEST HEADER
        MOV     #30.,R1         ;SET R1=30
        MOV     #120.,R2        ;SET CHAR COUNT = 120
        MOV     #30.,@#DELAY    ;SET UP DELAY VALUE
        MOV     #110.,R0        ;"H" TO R0
        .PRINTC                 ;SEND IT
        MOV     #30.,R0         ;DELAY
        .DELAY                  ;DELAY
        DEC     R1              ;DEC. COUNT OF CHARS PER LINE
        BEQ     .               ;BRANCH IF 0, END OF LINE
        DEC     R2              ;DECREMENT CHAR COUNTER
        BEQ     .               ;BRANCH IF END
        MOV     @#DELAY,R4      ;GET OLD DELAY
        ASR     R4              ;DIV BY 16 OF OLD DELAY
        ASR     R4
        ASR     R4
        ASR     R4
        ASR     R4              ;DIV BY 32
        ASR     R4              ;DIV BY 128 OF OLD DELAY
        ADD     R4,R5           ;ADD 16 + 128 TO R5
        BIT     #1,R2           ;TEST WHICH HALF OF THE 120 CHARS.
        BEQ     .               ;BRANCH IF 1ST OR 2ND 60
        ADD     R5,@#DELAY      ;1ST 60, INCREASE DELAY BY 34 MSEC.
        BR      .               ;GO PRINT AGAIN
        SUB     R5,@#DELAY      ;2ND HALF, DEC DELAY BY 34 MSEC.
        BR      .               ;GO PRINT AGAIN
        .SEND/CRLF              ;SEND CRLF
        MOV     #30.,R1         ;SET R1=30
        .SEND/CR,LF             ;SEND CR,LF
        .END/.                  ;END OF THIS TEST
        .END                    ;END
```

```
;\\\\\\\\\\
;
;P11:-- RIBBON FEED TEST -- THIS TEST PRINTS A SINGLE [illegible]
;       [illegible] LINES DOWN THE LEFT MARGIN OF THE PAGE. [illegible]
;       CHECK THE RIBBON FEED MECHANISM FOR PROPER OPERATION.
;
;//////////

P11:    [illegible]                     ;TEST NUMBER
        [illegible]                     ;NEXT TEST
        PRTHDR                          ;PRINT TEST HEADER
        [illegible]     #30,R1          ;SET R1=24
        [illegible]     #13A,R0         ;SET CHAR = A
        PRINTC                          ;PRINT X
        [illegible]                     ;SEND CR-LF
        [illegible]     R1              ;DEC COUNT
        [illegible]     [illegible]     ;BRANCH IF NOT DONE TEST
        [illegible]                     ;[illegible]
        [illegible]     [illegible]     ;REPEAT [illegible]

;\\\\\\\\\\
;
;P12:-- PRINTER BELL TEST -- THE LAST TEST IN THE
;       PRINTER [illegible] SEQUENCE. THIS TEST [illegible]
;       EIGHT BELL SIGNALS TO THE PRINTER
;
;//////////

P12:    [illegible]                     ;THIS TEST
        [illegible]                     ;NEXT TEST
        [illegible]                     ;PRINT TEST HEADER
        [illegible]     [illegible]     ;COUNTER [illegible]
        [illegible]     [illegible]     ;[illegible] RC
        [illegible]                     ;[illegible]
        [illegible]     [illegible]     ;[illegible]
        [illegible]     [illegible]     ;BRANCH IF NOT [illegible]
        [illegible]                     ;[illegible]
        [illegible]     [illegible]     ;DELAY 2 SEC BEFORE RESTARTING
        [illegible]

        [illegible]     [illegible]     ;[illegible]
        [illegible]     [illegible]     ;[illegible]
        [illegible]
[illegible]:    [illegible]     [illegible]
        [illegible]
        [illegible]
        [illegible]
        [illegible]
        [illegible]
[illegible]:    [illegible]     [illegible]     ;[illegible]
        [illegible]     [illegible]     ;[illegible]
```

```
;*****************
;
;PT17--TTYE TEST
;       THIS TEST PRINTS 2 FULL LINES OF EACH PRINTABLE
;       CHARACTER AND OVERPRINTS THE SECOND LINE 4 TIMES.
;       THIS TEST IS CONTINUOUS RUNNING ONCE INITIATED.
;       LOOPING AUTOMATICALLY ON ITSELF.
;       END OF PASS COUNT IS CLEARED WHENEVER THE TEST IS RESTARTED.
;
;*****************

PT17B:  17                      ;TEST NUMBER
        PT173                   ;NEXT TEST
        JMP     PT17C           ;CONTINUE
PT17:   17                      ;TEST NUMBER
        PT173                   ;NEXT TEST
        CLR     PASCNT          ;CLEAR PASS COUNT
        JSR     PC,HDR          ;PRINT TEST HEADER
PT17C:  MOV     #41,REPT        ;SET START CHAR
        MOV     #120,R1         ;SET COLUMN COUNT
        MOVB    REPT,R0         ;GET CHAR
        JSR     PC,TYO          ;SEND CHAR
        DEC     R1              ;DECREMENT COUNT
        BNE     .-6             ;BRANCH IF NOT DONE
        JSR     PC,CRLF         ;SEND CR-LF
        MOV     #5,R0           ;SET OVERPRINT COUNT
        MOV     #120,R1         ;SET COLUMN COUNT
        MOVB    REPT,R0         ;GET CHAR
        JSR     PC,TYO          ;SEND CHAR
        DEC     R1              ;DECREMENT COUNT
        BNE     .-6             ;BRANCH IF NOT DONE
        JSR     PC,CR           ;SEND CR
        DEC     R0              ;DONE OVERPRINTS?
        BGT     .-34            ;NO, CONTINUE
        JSR     PC,CRLF         ;SEND CR
        INCB    REPT            ;SET NEXT CHAR
        CMPB    #177,REPT       ;DONE CHAR SET?
        BNE     .-46            ;NO, CONTINUE
        INC     PASCNT          ;INCREMENT PASS COUNT
        MOV     #PASMES,R0      ;SET MESSAGE ADDRESS
        MOV     PASCNT,R1       ;# TO CONVERT
        MOV     #4,R2           ;# DIGITS
        JSR     PC,CVBDO        ;CONVERT PASS COUNT TO ASCII
        JSR     PC,TYPE         ;TYPE PASS COUNT
        JSR     PC,CRLF
        JMP     PT17            ;REPEAT TEST

PASCNT: 0
```

```
                                        .SBTTL  ECHO TESTS

                                ;*************
                                ;
                                ;E020-- CHARACTER ECHO TEST-- ALL PRINTABLE AND
                                ;       NON-PRINTING CHARACTERS TYPED ON THE KEYBOARD
                                ;       ARE USED TO DRIVE THE PRINTER, ONE CHARACTER AT
                                ;       A TIME.  A "RUBOUT" WILL CAUSE THE TEST TO BE
                                ;       TERMINATED.
                                ;
                                ;*************

                                E020:   20                              ;TEST NUMBER
                                        E021                            ;NEXT TEST
                                        PRTHDR                          ;PRINT TEST HEADER
                                1$:     READ                            ;GO WAIT FOR KEYBOARD INPUT
                 000035                 MOV     #30.,R0                 ;DELAY FOR HALF DUPLEX
                                        DELAY
                                        MOV     TEMPCH,R0               ;GET CHAR
                                        CMPB    #177,R0                 ;CHECK IF RUBOUT
                                        BEQ     2$                      ;BRANCH IF YES
                                        PRTCHO                          ;NO, CHECK PRINTER READ
                                        BR      1$
                                2$:     TYPEM                           ;PRINT TERMINATION MESSAGE
                                        E020END
                                        CHAIN                           ;CHAIN TO NEXT TEST
                                        BR      1$                      ;REPEAT TEST
```

```
                                        ;X*****XXXXX
                                        ;
                                        ;ECH1-- ONE ECHO TEST, FAST RATE-- THIS TEST WILL
                                        ;       CAUSE THE CONTINUAL PRINTING OF "C" AT THE MAXIMUM
                                        ;       RATE UNTIL EITHER ANOTHER CHARACTER IS SELECTED
                                        ;       BY PRESSING A KEY ON THE KEYBOARD OR TERMINATION BY THE
                                        ;       RUBOUT.
                                        ;
                                        ;XXXXXXXXXX

000001                                  ECH1:   21                      ;TEST NUMBER
000002                                          ECH2                    ;NEXT TEST
000000                                          PRTHDR                  ;PRINT TEST HEADER
000060  001114                          ECH1A:  MOV     #60,REPT        ;CHARACTER TO BE REPEATED  C
000100  001138                          1$:     MOV     #100.,POSI      ;SET CHAR COUNT
001114                                  1$:     MOV     REPT,R0         ;CHAR TO TEMPCH
104014                                          PRINTC                  ;PRINT CHAR
001138                                          DEC     POSI            ;DECREMENT POSITION COUNTER
000000                                          BNE     1$              ;CONTINUE
000015                                          MOV     #15,R0          ;SEND A CR AND LF
000012                                          JMP     ECH2            ;
000000                                          BR      ECH1A           ;

                                        ECH2:   TYPEM                   ;PRINT TERMINATION MESS
                                                ECHEND
                                                CHAIN                   ;CHAIN TO NEXT TEST
                                                BR      ECH1A           ;REPEAT TEST
```

```
                                        ;XXXXXXXXXX
                                        ;
                                        ;EC22-- LINE ECHO TEST, SLOW RATE-- SAME AS EC21 EXCEPT
                                        ;       THAT A DELAY IS INTRODUCED BETWEEN CHARACTERS
                                        ;       TO PRODUCE AN LOV ACTION
                                        ;
                                        ;XXXXXXXXXX

                                        EC22:   .B2
                                                .EC23
                                                PRTHDR                  ;PRINT TEST HEADER
                                        EC22A:  MOV     #60,REPT        ;LOAD 0 AS INITIAL CHARACTER
                                        1$:     MOV     WID+4,POS:      ;SET CHAR COUNT
                                        1$:     MOV     REPT,R0         ;READY CHAR TO TEMPCH
                                                PRINTC                  ;OUTPUT CHAR
                                                MOV     #3410,R0
                                                DELAY
                                                DEC     POS:            ;DECREMENT POSITION COUNTER
                                                BNE     1$              ;BRANCH IF NOT DONE LINE
                                        2$:     TRCF                    ;SEND A CR AND LF
                                                BR      3$

                                        EC22B:  TYPEM                   ;PRINT TERMINATION MESSAGE
                                                .EOMSG
                                                .EMTY
                                                BR      EC22A           ;CHAIN TO NEXT TEST
```

```
                                        ;********
                                        ;
                                        ;       THIS FOLLOWING TABLE IS USED BY TEST EC23
                                        ;
                                        ;********

011154  052516  020114          XCNIC:  .ASCII  /NUL/
011160  047523  020110                  .ASCII  /SOH/
011164  052123  020130                  .ASCII  /STX/
011170  052105  020130                  .ASCII  /ETX/
011174  047505  020124                  .ASCII  /EOT/
011200  047105  020121                  .ASCII  /ENQ/
011204  041501  020113                  .ASCII  /ACK/
011210  042502  020114                  .ASCII  /BEL/
011214  051502  020040                  .ASCII  /BS/
011220  052110  020040                  .ASCII  /HT/
011224  043114  020040                  .ASCII  /LF/
011230  052126  020040                  .ASCII  /VT/
011234  043106  020040                  .ASCII  /FF/
011240  051103  020040                  .ASCII  /CR/
011244  047523  020040                  .ASCII  /SO/
011250  044523  020040                  .ASCII  /SI/
011254  046104  020105                  .ASCII  /DLE/
011260  041504  020061                  .ASCII  /DC1/
011264  041504  020062                  .ASCII  /DC2/
011270  041504  020063                  .ASCII  /DC3/
011274  041504  020064                  .ASCII  /DC4/
011300  040516  020113                  .ASCII  /NAK/
011304  054523  020126                  .ASCII  /SYN/
011310  052105  020102                  .ASCII  /ETB/
011314  040503  020116                  .ASCII  /CAN/
011320  046505  020040                  .ASCII  /EM/
011324  052523  020102                  .ASCII  /SUB/
011330  051505  020103                  .ASCII  /ESC/
011334  051506  020040                  .ASCII  /FS/
011340  051507  020040                  .ASCII  /GS/
011344  051522  020040                  .ASCII  /RS/
011350  051525  020040                  .ASCII  /US/
011354  052123  020040                  .ASCII  /SP/

                                        .EVEN
```

```
                                        ;XXXXXXXXXX
                                        ;
                                        ;E023-- CHARACTER CODE TEST-- ALL CHARACTERS SELECTED
                                        ;       WILL BE ECHOED ALONG WITH ITS OCTAL CODE.
                                        ;       A MNEMONIC WILL BE PRINTED INSTEAD OF THE CHARACTER
                                        ;       IF IT IS A NON-PRINTING CHARACTER.
                                        ;       THE PARITY OF THE RECEIVED CODE WILL BE
                                        ;       INDICATED AS EITHER EVEN OR ODD.
                                        ;
                                        ;XXXXXXXXXX

011360  000023                          E023:   23                      ;TEST NUMBER
011362  011702                                  E024                    ;NEXT TEST
011364  104012                                  PRTHDR                  ;PRINT TEST HEADER
011366  104013                          1$:     READ                    ;GO WAIT FOR CHARACTER
011370  012700  000036                          MOV     #GD..R0         ;DELAY FOR HALF DUPLEX
011372  104003                                  DELAY
011376  023727  001124  000041                  CMP     TEMPCH,#41      ;TEST IF CHAR IS PRINTABLE
011404  103015                                  BHIS    3$              ;BRANCH IF IT IS
011406  004737  011542                          JSR     PC,STRLN        ;STORE CODE INTO MESSAGE
011412  013700  001124                          MOV     TEMPCH,R0       ;GET CODE AGAIN
011416  006300                                  ASL     R0              ;MULT BY 2
011420  006300                                  ASL     R0              ;MULT BY 4
011422  062700  011154                          ADD     #MONIC,R0       ;ADD ADDR OF MNEMONIC TABLE
011426  004737  011620                          JSR     PC,MOVNUM       ;MOV MNEMONIC TO MESSAGE
011432  104000                          3$:     TYPE                    ;TYPE CODE AND MNEMONIC
011434  017237                                  E023M                   ;ADDRESS OF MESSAGE
011436  000752                                  BR      1$              ;GO WAIT FOR NEXT CHARACTER
011440  023727  001124  000177          3$:     CMP     TEMPCH,#177     ;TEST IF CHAR IS A RUBOUT
011446  001421                                  BEQ     4$              ;BRANCH IF RUBOUT
011450  012701  011672                          MOV     #MG24,R1
011454  113721  001124                          MOVB    TEMPCH,(R1)+
011460  112721  000040                          MOVB    #40,(R1)+
011464  112721  000040                          MOVB    #40,(R1)+
011470  112721  000040                          MOVB    #40,(R1)+
011474  004737  011542                          JSR     PC,STRLN        ;STORE CODE INTO MESSAGE
011500  012700  011672                          MOV     #MG24,R0        ;ADDR OF CHAR INTO R0
011504  004737  011620                          JSR     PC,MOVNUM       ;MOVE CHAR INTO MESSAGE
011510  000750                                  BR      2$              ;TYPE MESSAGE
011512  004737  011542                  4$:     JSR     PC,STRLN        ;RUBOUT, CONVERT AND STOR CODE
011516  012700  011676                          MOV     #MG25,R0        ;ADDR. OF DEL INTO R0
011522  004737  011620                          JSR     PC,MOVNUM       ;MOVE DEL INTO MESSAGE
011526  104000                                  TYPE                    ;TYPE MESSAGE
011530  017237                                  E023M                   ;ADDR OF MESSAGE
011532  104002                                  TYPEM
011534  017210                                  ECOEND
011536  104001                                  CHAIN                   ;CHAIN TO NEXT TEST
011540  000712                                  BR      1$              ;REPEAT TEST
```

```
011542  012702  000003          STRLN:  MOV     #3,R2           ;COUNT OF 3 TO R2
011546  012701  017241                  MOV     #LINES,R1       ;ADDR OF MESG TO R1
011552  062701  000003                  ADD     #3,R1           ;POINT TO LAST SPACE IN MESG
011556  013700  001130          1$:     MOV     PCHAR,R0        ;MOVE OCTAL CODE TO R0
011562  042700  177770                  BIC     #177770,R0      ;SAVE LS OCTAL CHAR.
011565  062700  000060                  ADD     #60,R0          ;MAKE ASCII
011572  110041                          MOVB    R0,-(R1)        ;MOVE INTO MESG
011574  005302                          DEC     R2              ;DECREMENT CHAR COUNTER
011576  001405                          BEQ     2$              ;BRANCH IF 3 MOVED
011600  006237  001130                  ASR     PCHAR           ;NOT THREE, SHIFT NEXT OCTAL
011604  006237  001130                  ASR     PCHAR           ;CHARACTER TO THE RIGHT
011610  006237  001130                  ASR     PCHAR           ;
011614  000760                          BR      1$              ;CONVERT AND STORE NEXT CHAR.
011616  000207                  2$:     RTS     PC              ;RETURN TO CALLER


011620  012701  017245          MOVNUM: MOV     #LINESA,R1      ;ADDR OF LINES IN R1
011624  012702  000004                  MOV     #4,R2           ;COUNT OF 3 TO R2
011630  112021                  1$:     MOVB    (R0)+,(R1)+     ;MOV 3 CHARS TO MESG AREA
011632  005302                          DEC     R2              ;DECREMENT COUNTER
011634  001375                          BNE     1$              ;BRANCH IF NOT ALL DONE
011636  105737  001126                  TSTB    PARITY          ;TEST PARITY FLAG
011642  001003                          BNE     2$              ;BRANCH IF ODD PARITY
011644  012700  017302                  MOV     #EVEN,R0        ;SET MESSAGE FOR EVEN PARITY
011650  000402                          BR      3$              ;CONTINUE
011652  012700  017306          2$:     MOV     #ODD,R0         ;SET MESSAGE FOR ODD PARITY
011656  012702  000004          3$:     MOV     #4,R2           ;SET WORD COUNT
011662  112021                  4$:     MOVB    (R0)+,(R1)+     ;MOVE CHAR TO MESSAGE
011664  005302                          DEC     R2              ;DEC CHAR COUNT
011666  001375                          BNE     4$              ;CONTINUE
011670  000207                          RTS     PC              ;RETURN

011672  020040  020040          MG24:   .ASCII  /  /            ;SAVE CHARACTER CODE

                                        .EVEN

011676  042504  020114          MG25:   .ASCII  /DEL/           ;MNEMONIC FOR RUBOUT

                                        .EVEN
```

```
                                        ;XXXXXXXXXX
                                        ;
                                        ;E024-- SELECTED PATTERN ECHO TEST-- SELECT 1 TO 256
                                        ;       CHARACTERS.  EACH WILL BE ECHOED
                                        ;       AND STORED UNTIL THE CNTL/C IS SELECTED.
                                        ;       AT THAT TIME ALL CHARACTERS WILL BE PRINTED AS
                                        ;       A CONTINOUS STRING UNTIL EITHER THE RUBOUT IS
                                        ;       SELECTED TO TERMINATE OR THE CNTL/C IS SELECTED
                                        ;       AGAIN.  A TERMINATING CNTL/C FOLLOWED BY A CNTL/C WILL ALWAYS
                                        ;       CAUSE THE LAST INPUTTED STRING TO BE REPEATED. A TERMINATING
                                        ;       CNTL/C FOLLOWED BY SOME OTHER CHARACTER WILL START A NEW
                                        ;       STRING.
                                        ;
                                        ;XXXXXXXXXXX

011702  000024                  E024:   24                      ;TEST NUMBER
011704  012466                          E025                    ;NEXT TEST
011706  104012                          PRTHDR                  ;PRINT TEST HEADER
011710  005001                  E024B:  CLR     R1              ;CLEAR CHARACTER COUNT
011712  012702  012064                  MOV     #BUFR,R2        ;ADDRESS OF BUFFER TO R2
011716  104013                  1$:     READ                    ;WAIT FOR INPUT
011720  012700  000036                  MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
011724  104003                          DELAY
011726  022737  000177  001124          CMP     #177,TEMPCH     ;TEST IF RUBOUT
011734  001447                          BEQ     TERM            ;BRANCH IF RUBOUT
011736  022737  000003  001124          CMP     #3,TEMPCH       ;TEST IF .CNTL-C
011744  001416                          BEQ     OUTPUT          ;BRANCH IF CNTL-C
011746  020127  000400                  CMP     R1,#256.        ;YES, CHECK IF CHAR CNT IS EQ. GT 256
011752  103361                          BHIS    1$              ;BRANCH IF YES, IGNORE CHAR
011754  013700  001124                  MOV     TEMPCH,R0       ;GET CHAR
011760  110022                          MOVB    R0,(R2)+        ;STORE CHAR INTO BUFFER
011762  005201                          INC     R1              ;INCREMENT CHARACTER COUNT
011764  104021                          ECHO                    ;OUTPUT CHAR
011766  000753                          BR      1$              ;GO WAIT FOR NEXT CHAR

011770  005037  001124          E024R:  CLR     TEMPCH          ;CLEAR CONTROL-C FROM BUFFER
011774  104006                          CRLF                    ;CONTROL-C RETURN FROM PRINT ROUTINE
011776  104010                          LF
012000  000743                          BR      E024B
```

```
                                        ;********
                                        ;
                                        ;SECTION TO OUTPUT CONTINOUS STRING
                                        ;
                                        ;********

012002  020227  012064          OUTPUT: CMP     R2,#BUFR        ;CHECK IF POINTER IS AT START OF BUFFER
012006  001405                          BEQ     1$              ;YES, DON'T STORE ^C IN TABLE
012010  113722  001124                  MOVB    TEMPCH,(R2)+    ;STORE ^C IN TABLE
012014  005037  001124                  CLR     TEMPCH          ;CLEAR CONTROL-C FROM BUFFER
012020  104006                          CRLF                    ;SEND A CR LF
012022  012702  012064          1$:     MOV     #BUFR,R2        ;BUFFER ADDRESS TO R2
012026  005037  001124                  CLR     TEMPCH          ;CLEAR CONTROL-C
012032  121227  000003                  CMPB    (R2),#3         ;FIRST CHAR IN TABLE ^C ?
012036  001724                          BEQ     E0248           ;YES, GO LOOK FOR MORE INPUT
012040  112200                  2$:     MOVB    (R2)+,R0        ;GET CHAR
012042  020027  000003                  CMP     R0,#3           ;DONE STRING?
012046  001765                          BEQ     1$              ;YES, RESTART STRING
012050  104011                          PRINTC                  ;OUTPUT CHAR
012052  000772                          BR      2$

012054  104002                  TERM:   TYPEM                   ;OUTPUT TERMINATION MESSAGE
012056  017210                          ECOEND
012060  104001                          CHAIN                   ;CHAIN TO NEXT TEST
012062  000712                          BR      E0243           ;REPEAT TEST

012064  000003                  BUFR:   3                       ;INITIALIZE FIRST CHAR AS CNTL-C IN TABLE
012066  000400                          .BLKB   256.            ;256 CHARACTER BUFFER
```

```
                                        ;XXXXXXXXXXX
                                        ;
                                        ;E025-- BELL ECHO TEST-- A MESSAGE IS PRINTED AND
                                        ;       THE TEST WAITS FOR SOME PRINTABLE CHARACTER
                                        ;       TO BE SELECTED ON THE KEYBOARD (GT040).  THIS
                                        ;       TEST IS VALID ONLY IF THE PAPER WIDTH IS GT 64
                                        ;       COLUMNS.  IF LT64 COLUMNS AN ILLEGAL BELL TEST
                                        ;       MESSAGE IS PRINTED.
                                        ;
                                        ;XXXXXXXXXXX

012466  000025                  E025:   25                      ;TEST NUMBER
012470  010756                          E020                    ;NEXT TEST HEADER
012472  104012                          PRTHDR                  ;PRINT TEST HEADER
012474  023727  001112  000101  1$:     CMP     WIDTH,#101      ;TEST IF COLUMN COUNTIS EQ.GT 64
012502  103424                          BLO     4$              ;BRANCH IF NOT
012504  104002                          TYPEM                   ;TYPE TEST MESG
012506  017063                          E025MA                  ;ON ALL TERM'S
012510  000402                          BR      3$              ;WAIT FOR CHAR
012512  104000                  2$:     TYPE                    ;TYPE TEST MESG ON TERM
012514  017063                          E025MA                  ;CHARACTER WAS RECEIVED ON
012516  104013                  3$:     READ                    ;WAIT FOR OPERATOR RESPONSE
012520  012700  000036                  MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
012524  104003                          DELAY
012526  113700  001124                  MOVB    TEMPCH,R0       ;CHAR TO R0
012532  020027  000040                  CMP     R0,#40          ;TEST IF PRINTABLE
012536  103767                          BLO     3$              ;BRANCH IF NON-PRINTABLE
012540  022700  000177                  CMP     #177,R0         ;CHECK IF CHAR IS RUBOUT
012544  001405                          BEQ     5$              ;BRANCH IF YES
012546  104021                          ECHO                    ;PRINT CHAR
012550  104007                          SCRLF                   ;SEND A CRLF
012552  000757                          BR      2$              ;REPEAT
012554  104002                  4$:     TYPEM                   ;TYPE ERROR MESSAGE
012556  017163                          E025MB
012560  104002                  5$:     TYPEM                   ;PRINT TERMINATION
012562  017210                          ECOEND
012564  104001                          CHAIN                   ;EXIT TO NEXT TEST
012566  000742                          BR      1$              ;REPEAT TEST
```

```
                                        .SBTTL  OPTION TESTS

                                ;XXXXXXXXXXXXXXX
                                ;
                                ;TEST30 -       SECONDARY CHARACTER SET OPTION
                                ;
                                ;XXXXXXXXXXXXXXX

012570  000030                  TEST30: 30
012572  012570                          TEST30
012574  104012                          PRTHDR                  ;PRINT TEST HEADER
012576  012704  000010                  MOV     #8.,R4          ;SET PASS COUNT
012602  104002  012666          2$:     TYPEM,  10$             ;INDICATE PRIMARY SET AND SEND "SI"
012606  012702  000177                  MOV     #177,R2         ;SET END CHAR
012612  004737  012700                  JSR     PC,30$          ;PRINT CHAR SET
012616  104002  012673                  TYPEM,  20$             ;INDICATE SECONDARY CHAR SET
012622  013702  012750                  MOV     T30SC,R2                ;SET CHAR SET LIMIT
012626  020227  000377                  CMP     R2,#377         ;USING 8 BITS INSTEAD OF SI?
012632  001403                          BEQ     3$              ;BRANCH IF YES
012634  012700  000016                  MOV     #16,R0          ;SET SO CHAR
012640  104011                          PRINTC                  ;SEND IT
012642  004737  012700          3$:     JSR     PC,30$          ;PRINT CHAR SET
012646  104006                          CRLF                    ;BLANK LINE
012650  005304                          DEC     R4              ;DEC PASS COUNT
012652  001353                          BNE     2$              ;FINISH TEST
012654  012700  000017                  MOV     #17,R0          ;SET SI CHAR AGAIN
012660  104011                          PRINTC                  ;MAKE SURE ON PRIMARY CHAR SET
012662  104001                          CHAIN                   ;NEXT TEST SELECTION OR LOOP
012664  000741                          BR      TEST30          ;LOOP ON TEST

012666  021417  036461     000  10$:    .ASCIZ  <17>/#1=/
012673     017  031043  000075  20$:    .ASCIZ  <17>/#2=/
                                        .EVEN

012700  010201                  30$:    MOV     R2,R1           ;GET LIMIT CHAR
012702  042701  177537                  BIC     #177537,R1      ;GET START CHAR
012706  013703  001112                  MOV     WIDTH,R3        ;GET COLUMN COUNT
012712  162703  000003                  SUB     #3,R3           ;SUBTRACT 3
012716  010100                  31$:    MOV     R1,R0           ;GET CHAR
012720  104011                          PRINTC                  ;PRINT IT
012722  005201                          INC     R1              ;NEXT CHAR
012724  020102                          CMP     R1,R2           ;DONE CHAR SET?
012726  001406                          BEQ     32$             ;EXIT IF DONE
012730  005303                          DEC     R3              ;DEC COLUMN COUNT
012732  001371                          BNE     31$             ;FINISH LINE
012734  104006                          CRLF                    ;SEND CR-LF WHEN DONE LINE
012736  013703  001112                  MOV     WIDTH,R3        ;RESET COLUMN COUNT
012742  000765                          BR      31$             ;CONTINUE
012744  104006                  32$:    CRLF                    ;SEND CR-LF
012746  000207                          RTS     PC              ;RETURN

012750  000177                  T30SC:  .WORD   177             ;CHAR SET LIMIT
                                                                ;CHANGE TO 377 WHEN USING 8 BIT CHAR SELECTION
```

B07


```
                          ;\\\\\\\\\\\\\\\\\\
                          ;
                          ;TEST31 -      SELECTIVE ADDRESSING OPTION
                          ;
                          ;\\\\\\\\\\\\\\\\\\

                          TEST31: 3:
                                    TEST31
                                    CRLF                     ;SEND CRLF
                  013542            MOV     34$,R0           ;SET EOT CHAR
                                    PRINTC                   ;SEND IT
                  013372            TYPEM,  10$              ;TRY PRINTING ERROR MESG
                  000007            MOV     #7,R0            ;SET BEL CHAR
                                    PRINTC                   ;SEND IT
                  000002            MOV     #2,R0            ;SET STX CHAR
                                    PRINTC                   ;SEND IT
                                    PRTHDR                   ;PRINT TEST HEADER ON ALL TERMINALS
                  013542            MOV     34$,R0           ;SET EOT CODE
                                    PRINTC                   ;SEND IT
                  000002            MOV     #2,R0            ;SET STX CODE
                                    PRINTC                   ;SEND IT
                  013372            TYPEM,  10$              ;TRY PRINT ERROR MESG
                  013520            MOV     #30$,R3          ;SET TABLE POINTER
                          1$:       TST     (R3)             ;CHECK TABLE POINTER
                                    BEQ     2$               ;NEXT PORTION OF TEST IF DONE
                  013542            MOV     34$,R0           ;SEND EOT CHAR
                                    PRINTC
                                    MOV     (R3),R0          ;SEND GROUP SELECT CHAR
                                    PRINTC                   ;SEND IT
                  000002            MOV     #2,R0            ;SEND STX CHAR
                                    PRINTC
                  013436            TYPEM,  14$              ;TYPE MESG
                                    MOV     (R3)+,R0         ;TYPE SELECT CHAR FOR MESG
                                    PRINTC
                                    CRLF                     ;CR-LF
                                    BR      1$               ;CONTINUE - NEXT SELECT CHAR
                  013546  2$:       MOV     #40$,R3          ;SET TABLE ADR
                          3$:       TST     (R3)             ;CHECK SELECT CHAR
                                    BEQ     4$               ;CONTINUE TEST
                  013542            MOV     34$,R0           ;SEND EOT CHAR
                                    PRINTC                   ;ALL TERMS OFF
                                    MOV     (R3),R0          ;GET UNIQUE SELECT CHAR
                                    PRINTC                   ;SEND IT - THAT TERM ON
                  013372            TYPEM,  10$              ;TRY PRINTNG ERROR MESG
                  000002            MOV     #2,R0            ;SEND STX
                                    PRINTC
                  013444            TYPEM,  15$              ;TYPE SELECT CHAR MESG
                                    MOV     (R3),R0          ;PRINT SELECT CHAR FOR MESG
                                    PRINTC
                                    CRLF
                  013463            TYPEM,  20$              ;PRINT MESG
                                    READ                     ;READ CHAR FROM SELECTED TERM
                  000036            MOV     #30.,R0          ;DELAY FOR HALF DUPLEX
                                    DELAY
```

```
000150  013156  000737  000177  001124          CMP     #177,TEMPCH     ;CHECK CHAR
000152  013164  001013                          BNE     5$              ;EXIT IF RUBOUT
000154  013166  013700  013548                  MOV     34$,R0          ;ENABLE ALL TERMINALS
000156  013172  104011                          PRINTC
000160  013174  012700  000007                  MOV     #7,R0
000162  013200  104011                          PRINTC
000164  013202  012700  000002                  MOV     #2,R0
000166  013206  104011                          PRINTC
000170  013210  000137  002450                  JMP     TTY1H           ;GO TO KYBD CONTROL
000172  013214  013700  001124          5$:     MOV     TEMPCH,R0       ;GET CHAR
000174  013220  104021                          ECHO                    ;ECHO CHAR
000176  013222  104006                          CRLF                    ;SEND CR-LF
000200  013224  013700  013548                  MOV     34$,R0          ;SEND EOT CHAR
000202  013230  104011                          PRINTC
000204  013232  013700  013544                  MOV     35$,R0          ;SEND DUMMY SELECT CHAR
000206  013236  104011                          PRINTC
000210  013240  012700  000002                  MOV     #2,R0           ;SEND STX
000212  013244  104011                          PRINTC
000214  013246  104002  013372                  TYPEM.  10$             ;TRY PRINTING ERROR MESSAGE
000216  013252  012700  000003                  MOV     #3,R0           ;SEND ETX
000220  013256  104011                          PRINTC
000222  013260  013700                          MOV     (R3)+,R0        ;SEND UNIQUE SELECT CHAR
000224  013262  104011                          PRINTC
000226  013264  012700  000002                  MOV     #2,R0           ;SEND STX
000230  013270  104011                          PRINTC
000232  013272  104002  013444                  TYPEM.  15$             ;PRINT SELECT MESS ON SELECTED TERMINAL
000234  013276  013300                          MOV     (R3)+,R0        ;PRINT SELECT CHAR FOR MESS
000236  013300  104011                          PRINTC
000240  013302  104006                          CRLF                    ;SEND CR-LF
000242  013304  012700  000003                  MOV     #3,R0           ;SEND ETX
000244  013310  104011                          PRINTC
000246  013312  013700  013544                  MOV     #35$,R0         ;SEND DUMMY SELECT CHAR
000250  013316  104011                          PRINTC
000252  013320  012700  000002                  MOV     #2,R0           ;SEND STX
000254  013324  104011                          PRINTC
000256  013326  104002  013444                  TYPEM.  15$             ;PRINT MESS ON SELECTED TERM
000260  013332  013300                          MOV     (R3)+,R0        ;PRINT SELECT CHAR FOR MESS
000262  013334  104011                          PRINTC
000264  013336  104006                          CRLF                    ;SEND CR-LF
000266  013340  000000                          BR      3$              ;CONTINUE
000270  013342  013700  013548          4$:     MOV     34$,R0          ;ENABLE ALL LINES
000272  013346  104011                          PRINTC                  ;BEFORE EXITING TEST
000274  013350  012700  000007                  MOV     #7,R0
000276  013354  104011                          PRINTC
000300  013356  012700  000002                  MOV     #2,R0
000302  013362  104011                          PRINTC
000304  013364  104000                          CHAIN                   ;CHAIN TO NEXT TEST IF LOOP ON TEST
000306  013366  000137  012732                  JMP     TEST1           ;LOOP ON TEST
```

```
000000  013372  051105  047522  026122  10$:    .ASCIZ  ERROR, ALL TERMINALS SHOULD BE OFF  DCPLF
000000  013400  040440  046114  053040
000000  013406  051105  044516  040516
000000  013414  051514  051440  047516
000000  013422  046125  020104  042502
000000  013430  047440  043106  000200
000000  013436  051101  052517  020120  14$:    .ASCIZ  GROUP
000000  013444  042523  042514  052103  15$:    .ASCIZ  SELECT CHAR =
000000  013452  041440  040510  020122
000000  013460  020075     000
000000  013463     124  050131  020105  20$:    .ASCIZ  TYPE ANY PRINTABLE CHAR ...
000000  013470  047101  020131  051103
000000  013476  047111  040524  046105
000000  013504  020105  044103  051101
000000  013512  020040  020056  000040
000000                                          .EVEN
000000
000000  013520  000107                  30$:    .BYTE                   ;DCPLF SELECT CHAR TABLE
000000  013522  000000                          .BYTE                   ;FIRST ZERO = END OF TABLE
000000  013524  000000                          .BYTE
000000  013526  000000                          .BYTE
000000  013530  000000                          .BYTE
000000  013532  000000                          .BYTE
000000  013534  000000                          .BYTE
000000  013536  000000                          .BYTE
000000  013540  000000                          .BYTE
000000
000000  013542  000004                  34$:    .BYTE 004               ;DESELECT CHAR = "EOT"
000000
000000  013544  000045                  35$:    .BYTE 045               ;DUMMY SELECT CHARACTER.
000000                                                                  ;IF "%" IS USED AS A DCPLF OR DCPLP SELECT
000000                                                                  ;CHARACTER, REPLACE WITH ANY UNUSED SELECT
000000                                                                  ;CHARACTER CODE.
000000
000000  013546  000106                  40$:    .BYTE 106               ;DCPLP SELECT CHAR TABLE
000000  013550  000000                          .BYTE                   ;FIRST ZERO = END OF TABLE
000000  013552  000000                          .BYTE
000000  013554  000000                          .BYTE
000000  013556  000000                          .BYTE
000000  013560  000000                          .BYTE
000000  013562  000000                          .BYTE
000000  013564  000000                          .BYTE
```

```
                                        ;XXXXXXXXXXXXXXXX
                                        ;
                                        ;TEST32 -        AUTO ANSWER BACK OPTION
                                        ;
                                        ;XXXXXXXXXXXXXXXX

013570  000032                          TEST32: 32
013572  013570                                  TEST32
013574  104017                                  PRTHDR                          ;PRINT TEST HEADER
013576  012700  000005                          MOV     #5,R0                   ;SEND ENQ CHAR
013602  104021                                  ECHO
013604  004737  013634                          JSR     PC,10$                  ;READ AND PRINT MESG
013610  104002  013722                          TYPEM,  20$                     ;TYPE INSTRUCTIONS
013614  004737  013634                          JSR     PC,10$                  ;READ AND PRINT MESG
013620  104002  013747                          TYPEM,  30$                     ;TYPE INSTRUCTIONS
013624  004737  013634                          JSR     PC,10$                  ;READ AND PRINT MESG
013630  104001                          2$:     CHAIN                           ;CHAIN TO NEXT TEST
013632  000756                                  BR      TEST32                  ;LOOP ON TEST

013634  012702  013773                  10$:    MOV     #STORE,R2               ;SET TABLE ADR
013640  104013                                  READ                            ;READ FIRST CHAR
013642  023727  001124  000177          5$:     CMP     TEMPCH,#177             ;CHAR = RUBOUT?
013650  001002                                  BNE     3$                      ;CONTINUE IF NOT RUBOUT
013652  000137  002450                          JMP     TTYIN                   ;GO TO KYBD CONTROL
013656  113722  001124                  3$:     MOVB    TEMPCH,(R2)+            ;STORE CHAR
013662  012703  000050                          MOV     #50,R3                  ;SET DELAY COUNT
013666  013701  001132                  4$:     MOV     TIMER,R1
013672  104022                          8$:     INRDY                           ;ANY INPUT?
013674  001401                                  BEQ     6$                      ;NO, WAIT FOR CHAR
013676  000761                                  BR      5$                      ;YES, READ CHAR
013700  005301                          6$:     DEC     R1                      ;DELAY WHILE WAITING FOR CHAR
013702  001373                                  BNE     8$
013704  005303                                  DEC     R3
013706  001367                                  BNE     4$
013710  105012                          7$:     CLRB    (R2)                    ;SET NULL AS TERMINATOR IN TABLE
013712  104002  013772                          TYPE,   STORE-1                 ;TYPE MESG ON TERMINAL RECEIVED ON
013716  104007                                  CRLF                            ;SEND CR-LF
013720  000207                                  RTS     PC                      ;RETURN TO TEST

013722  042504  051120  051505          20$:    .ASCIZ  <DEPRESS HERE IS KEY  ACRLF>
013730  020123  044110  051105
013736  020105  051511  045440
013744  100131  000
013747  ...     042504  051120          30$:    .ASCIZ  <DEPRESS CONTROL-E  ACRLF>
013754  051505  020123  047503
013762  052116  047522  026514
013770  000200
013772  200                                     .ASCIZ  <CRLF>

013773  000025                          STORE:  .BLKB   21.                     ;20 CHAR + TERMINATOR BUFFER

                                                .EVEN
```

```
                                        ;XXXXXXXXXXXXXXX
                                        ;
                                        ;TEST33 -       FORM FEED OPTION
                                        ;
                                        ;XXXXXXXXXXXXXXX

014030  000033                          TEST33: 33
014032  014030                                  TEST33
014034  104012                                  PRTHDR          ;PRINT TEST HEADER
014036  012705  014472                          MOV     #503$,R5        ;SET TABLE POINTER
014042  012704  014455                          MOV     #603$,R4        ;SET TABLE POINTER
014046  104002  014301                          TYPEM.  40$             ;PRINT INSTR
014050  012701  000020                          MOV     #16.,R1         ;SET LF COUNT TO 16
014054  104013                                  READ                    ;WAIT FOR KYBD FLAG
014056  012700  000036                          MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
014062  104003                                  DELAY
014064  022737  000177  001134                  CMP     #177,TEMPCH     ;CHECK FOR RUBOUT
014072  001003                                  BNE     10$             ;EXIT IF RUBOUT
014074  000137  002450                          JMP     ERR14           ;GO TO KYBD CONTROL
014102  104014                          10$:    CR                      ;SEND CR
014104  104010                          20$:    LF                      ;SEND LF
014106  005301                                  DEC     R1              ;DEC COUNT
014110  001375                                  BNE     20$             ;CONTINUE
014112  012700  000014                          MOV     #14,R0          ;SET FF
014116  104011                                  PRINTC                  ;SEND IT
014120  113701  014456                          MOVB    603$,R1         ;SET FILL COUNT
014124  005000                          30$:    CLR     R0              ;SET NULL
014126  104011                                  PRINTC                  ;SEND FILL
014130  005301                                  DEC     R1              ;DEC FILL COUNT
014132  001374                                  BNE     30$             ;CONTINUE
014134  104002  014417                          TYPEM.  405$            ;TYPE MESSAGE
014140  010537  014210                  50$:    MOV     R5,2$           ;SET MESG
014144  104013                                  READ                    ;WAIT FOR KYBD FLAG
014146  012700  000036                          MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
014152  104003                                  DELAY
014154  022737  000177  001134                  CMP     #177,TEMPCH     ;CHECK IF RUBOUT
014162  001003                                  BNE     10$             ;EXIT IF RUBOUT
014164  000137  002450                          JMP     ERR14           ;GO TO KYBD CONTROL
014170  104014                          100$:   CR                      ;SEND CR
014172  012700  000014                          MOV     #14,R0          ;SEND FF
014176  104011                                  PRINTC                  ;SEND IT
014200  113401                                  MOVB    (R4),R1         ;GET FILL COUNT
014202  005000                          110$:   CLR     R0              ;SET NULL
014204  104011                                  PRINTC                  ;SEND FILLS
014206  005301                                  DEC     R1              ;CONTINUE
014210  001374                                  BNE     110$            ;FINISH NULLS
014212  104002  014554                  120$:   TYPEM.  100$            ;PRINT MESG
014216  104002                                  TYPEM
014220  000000                          130$:   .WORD   0
014222  104002  014268                          TYPEM.  200$
014226  022605                                  CMP     (R5),+,R5 +     ;INC TABLE POINTER
014230  010537  014226                  140$:   MOV     R5,4$           ;SET MESG
014234  104002                                  TYPEM                   ;PRINT MESG
014236  000000                          150$:   .WORD   0
```

```
014230  104206  014271                  TYPEM   20$
014234  104003  014264                  TYPEM   10$
014240  005765  000054                  TST     4(R5)           ;DONE TEST?
014244  001331                          BNE     5$              ;NO, CONTINUE
014246  104206                          CRLF                    ;YES, SEND CR-LF
014250  104001                          CHAIN                   ;SELECT TEST OR LOOP
014252  000662                          BR      TEST33          ;LOOP ON TEST

014254  026455  026455  000055  10$:    .ASCIZ  -----
014262  020040  043106  020057  20$:    .ASCIZ  " FF  ST
014270     000
014271     042  047040  054105  30$:    .ASCIZ  " NEXT
014276  020124     000
014301     104  050105  042522  40$:    .ASCIZ  DEPRESS FORMFEED RESET SWITCH <CR><LF> AFTER EACH SWITCH SETTING <CR><LF>
014306  051523  043040  051117
014314  043115  042505  020104
014322  042522  042523  020124
014330  053523  052111  044103
014336  040600  052106  051105
014344  042440  041501  020110
014352  053523  052111  044103
014360  051440  052105  044524
014366  043516     200
014371     124  050131  020105          .ASCIZ  TYPE SPACE WHEN READY <CR><LF>
014376  050123  041501  020105
014404  044127  047105  051040
014412  040505  054504     200
014416     055  026455  020055  45$:    .ASCIZ  ---- SET  3 INCH FORMFEED ----
014424  042523  020124  031440
014432  044440  041516  020110
014440  047506  046523  042506
014446  042105  026440  026455
014454  000055
014456     003     005     010  50$:    .BYTE   3.5.8..17..20..26..30..35..40..46..49.
014461     021     024     030
014464     040     043     000
014467     000     104
014470
014472  014472                          .EVEN

014472  020040  000063  50$:    .ASCIZ  <   >
014476  027063  000065                  .ASCIZ  <3.5>
014502  020040  000064                  .ASCIZ  <  4>
014506  027065  000065                  .ASCIZ  <5.5>
014512  020040  000066                  .ASCIZ  <  6>
014516  020040  000067                  .ASCIZ  <  7>
014522  020040  000070                  .ASCIZ  <  8>
014526  027070  000065                  .ASCIZ  <8.5>
014532  030440  000061                  .ASCIZ  <10>
014536  030440  000062                  .ASCIZ  <12>
014542  030440  000064                  .ASCIZ  <14>
014546  020040  000040                  .ASCIZ  <  >
014552  000000                          .ASCIZ  <0>             ;END OF TABLE
```

```
                                        ;\\\\\\\\\\\\\\\\
                                        ;
                                        ;TEST34 -       HORIZONTAL TAB OPTION
                                        ;
                                        ;\\\\\\\\\\\\\\\\

014554  000034                          TEST34: 34
014556  014554                                  TEST34
014560  104012                                  PRTHDR                  ;PRINT TEST HEADER
014562  005004                                  CLR     R4              ;SET TABLE POINTER
014564  012737  014760  014756                  MOV     #16$,12$+2      ;RESET JUMP INSTR OR FIRST TIME TAB
014572  013703  001112                  3$:     MOV     WIDTH,R3        ;SET COLUMN COUNT
014576  012700  000033                          MOV     #33,R0          ;CLEAR OLD TABS
014602  104011                                  PRINTC
014604  012700  000062                          MOV     #62,R0
014610  104011                                  PRINTC
014612  104014                                  CR                      ;DO A CARRIAGE RETURN
014614  016401  015046                  3$:     MOV     20$(R4),R1      ;GET SPACE COUNT FOR TAB
014620  000405                                  BR      5$              ;SUBTRACT 1 FOR TAB SET
014622  012700  000040                  4$:     MOV     #40,R0          ;GET SPACE
014626  104011                                  PRINTC                  ;SEND IT
014630  005303                                  DEC     R3              ;DEC COLUMN COUNT
014632  001420                                  BEQ     6$              ;CR IF DONE LINE
014634  005301                          5$:     DEC     R1              ;DEC SPACE COUNT
014636  001371                                  BNE     4$              ;CONTINUE IF NOT DONE
014640  012700  000033                          MOV     #33,R0          ;SEND IT
014644  104011                                  PRINTC
014646  012700  000061                          MOV     #61,R0          ;SEND 1 - SET TAB
014652  104011                                  PRINTC
014654  012700  000010                          MOV     #10,R0          ;SET BACKSPACE
014660  104011                                  PRINTC
014662  012700  000117                          MOV     #'O,R0          ;PRINT O
014666  104011                                  PRINTC
014670  005303                                  DEC     R3              ;DEC COLUMN COUNT
014672  001350                                  BNE     3$              ;CONTINUE IF NOT DONE LINE
014674  104014                          6$:     CR                      ;SEND CR
014676  013703  001112                          MOV     WIDTH,R3        ;RESET COLUMN COUNT
014702  016401  015046                  17$:    MOV     20$(R4),R1      ;RESET COLUMN COUNT FOR TAB
014706  020127  000001                          CMP     R1,#1           ;ADD 1 IF FIRST LINE
014712  001001                                  BNE     11$
014714  005201                                  INC     R1
014716  016402  015072                  11$:    MOV     30$(R4),R2      ;SET FILL CHAR COUNT
014722  160103                                  SUB     R1,R3           ;SUBTRACT TAB FROM COLUMN COUNT
014724  002413                                  BLT     12$             ;BRANCH IF TOO MANY COLUMNS
014726  012700  000011                          MOV     #11,R0          ;SET TAB
014732  104011                                  PRINTC                  ;SEND IT
014734  005000                          14$:    CLR     R0              ;SET NULL CHAR - FILLS
014736  104011                                  PRINTC                  ;SEND FILL CHARS
014740  005302                                  DEC     R2              ;DEC FILL COUNT
014742  001374                                  BNE     14$             ;CONTINUE FILLS
014744  012700  000130                          MOV     #'X,R0          ;SET X CHAR
014750  104011                                  PRINTC                  ;SEND IT
014752  000761                                  BR      11$             ;CONTINUE
014754  000137  014760                  13$:    JMP     2$#16$          ;SKIP FOLLOWING AFTER FIRST TIME TAB
```

```
014760  012737  015006  014756  16$:    MOV     #15$,12$+2
014766  104014                          CR                      ;SEND CR
014770  012700  000130                  MOV     #'X,R0          ;PRINT X
014774  104011                          PRINTC
014776  013703  001112                  MOV     WIDTH,R3        ;RESET COLUMNC COUNT
015002  005303                          DEC     R3              ;SUBTRACT ONE FOR FIRST X CHAR
015004  000736                          BR      17$             ;CONTINUE
015006  104006                  15$:    CRLF                    ;SEND CR-LF
015010  005724                          TST     (R4)+           ;INC TABLE POINTER
015012  016401  015046                  MOV     20$(R4),R1      ;GET COLUMN COUNT FOR TAB
015016  001403                          BEQ     13$             ;EXIT IF DONE TABLE (0)
015020  020137  001112                  CMP     R1,WIDTH        ;CHECK IF TOO LARGE
015024  101662                          BLOS    2$              ;CONTINUE TEST, OK
015026  012700  000033          13$:    MOV     #33,R0          ;CLEAR ALL TABS BEFORE EXITING
015032  104011                          PRINTC
015034  012700  000062                  MOV     #62,R0
015040  104011                          PRINTC
015042  104001                          CHAIN                   ;SELECT TEST OR LOOP ON TEST
015044  000643                          BR      TEST34          ;LOOP ON TEST

015046  000001  000002  000004  20$:    .WORD   1,2,4,6..16..32..64..128..132..0
015054  000010  000020  000040
015062  000100  000200  000204
015070  000000
015072  000001  000002  000003  30$:    .WORD   1,2,3,5,9..18..36..71..73..0
015100  000004  000011  000033
015106  000005  000012  000035
015112  000007  000016  000037
015116  000011  000000
```

```
                                        ;XXXXXXXXXXXXXXX
                                        ;
                                        ;TEST35 -       VERTICAL TAB OPTION
                                        ;
                                        ;XXXXXXXXXXXXXXX

015116  000035                          TEST35: 35
015120  015116                                  TEST35
015122  104012                                  FRTHDR                  ;PRINT TEST HEADER
015124  104003                          1$:     TYPEM                   ;TYPE INSTR
015126  015437                                  20$
015130  104013                                  READ                    ;WAIT FOR KYBD FLAG
015132  012700  000036                          MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
015136  104003                                  DELAY
015140  022737  000177  001124                  CMP     #177,TEMPCH     ;CHECK CHAR
015146  001505                                  BEQ     12$             ;EXIT TEST IF RUBOUT
015150  005004                                  CLR     R4              ;SET LINE COUNT
015152  012700  000033                          MOV     #33,R0          ;CLEAR VERTICAL TABS
015156  104011                                  PRINTC
015160  012700  000064                          MOV     #64,R0
015164  104011                                  PRINTC
015166  104002                          2$:     TYPEM                   ;TYPE REF LINE
015170  015415                                  15$
015172  005204                                  INC     R4              ;INC LINE COUNT
015174  020427  000013                          CMP     R4,#13          ;CHECK IT
015200  003013                                  BGT     35$             ;BRANCH IF DONE REF.
015202  010401                                  MOV     R4,R1           ;GET LF COUNT
015204  104010                          3$:     LF                      ;SEND LF
015206  005301                                  DEC     R1              ;DEC LF COUNT
015210  001375                                  BNE     3$              ;CONTINUE
015212  012700  000033                          MOV     #33,R0          ;SET TAB FOR THIS LINE
015216  104011                                  PRINTC
015220  012700  000063                          MOV     #63,R0
015224  104011                                  PRINTC
015226  000757                                  BR      2$              ;CONTINUE
015230  012700  000014                  35$:    MOV     #14,R0          ;SEND FF
015234  104011                                  PRINTC
015236  104002                                  TYPEM                   ;TYPE MESG
015240  015531                                  30$
015242  104013                                  READ                    ;WAIT FOR KYBD FLAG
015244  012700  000036                          MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
015250  104003                                  DELAY
015252  022737  000177  001124                  CMP     #177,TEMPCH     ;CHECK CHAR
015260  001440                                  BEQ     12$             ;EXIT TEST IF RUBOUT
015262  005004                                  CLR     R4              ;RESET LF COUNT
015264  104002                          4$:     TYPEM                   ;TYPE REF LINE
015266  015414                                  10$
015270  005204                                  INC     R4              ;INC LINE COUNT
015272  020427  000013                          CMP     R4,#13          ;CHECK IT
015276  003014                                  BGT     9$              ;BRANCH IF DONE
015300  012700  000013                          MOV     #13,R0          ;SEND TAB
015304  104011                                  PRINTC
015306  010401                                  MOV     R4,R1           ;SET FILL COUNT
015310  162701  000020                          SUB     #16.,R1         ;SUBTRACT 16
```

K07


```
015314  003763                          BLE     4$              ;SKIP NULLS IF COUNT < 0
015316  005000                  5$:     CLR     R0              ;SET NULL CHAR
015320  104011                          PRINTC                  ;SEND NULLS
015322  005301                          DEC     R1              ;DEC COUNT
015324  001374                          BNE     5$              ;FINISH NULLS
015326  000756                          BR      4$              ;CONTINUE
015330  012700  000033          8$:     MOV     #33,R0          ;CLEAR VERTICAL TABS
015334  104011                          PRINTC
015336  012700  000064                  MOV     #64,R0
015342  104011                          PRINTC
015344  012700  000014                  MOV     #14,R0          ;SEND FF
015350  104011                          PRINTC
015352  104006                          CRLF                    ;SEND CR-LF
015354  104010                          LF                      ;SEND BLANK LINE
015356  104001                          CHAIN                   ;SELECT TEST OR LOOP ON TEST
015360  000656                          BR      TEST35          ;LOOP ON TEST
015362  012700  000033          12$:    MOV     #33,R0          ;CLEAR ALL TABS BEFORE EXITING TEST
015366  104011                          PRINTC
015370  012700  000064                  MOV     #64,R0
015374  104011                          PRINTC
015376  012700  000014                  MOV     #14,R0          ;SEND FF
015402  104011                          PRINTC
015404  104006                          CRLF
015406  104010                          LF
015410  000137  002450                  JMP     TTY1H           ;GO KYBD CONTROL

015414     040                  10$:    .ASCII  <40>
015415     055  026455  026455  15$:    .ASCIZ  /---------<<<<<<<<<15>
015422  026455  036055  036074
015430  036074  036074  006474
015436     000
015437     123  052105  030440  20$:    .ASCII  /SET 14 INCH FORM FEED/<ACRLF>
015444  020064  047111  044103
015452  043040  051117  020115
015460  042506  042105     200
015465     104  050105  042522          .ASCIZ  /DEPRESS TOP OF FORM RESET SWITCH/<ACRLF><12><12>
015472  051523  052040  050117
015500  047440  020106  047505
015506  046522  051040  051505
015514  052105  051440  044527
015522  041524  100110  005012
015530     000
015531     001  000401  042522  30$:    .ASCIZ  <1><1><1>/REMOVE/<57>/RESET REF
015536  047515  042526  027440
015544  051040  051505  052105
015552  051040  043105     000
        015560                          .EVEN
```

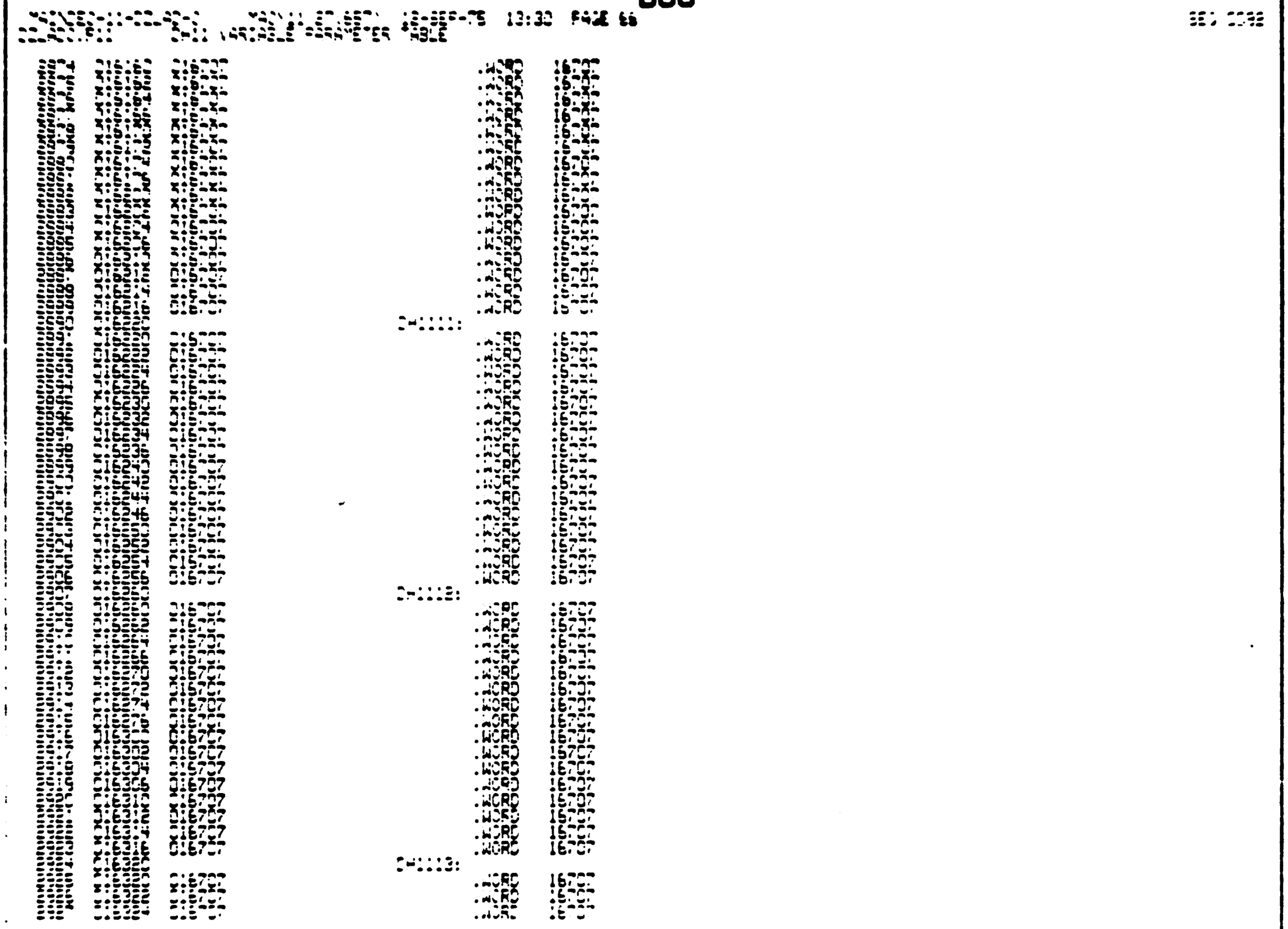
```
                                        .SBTTL  DH11 VARIABLE PARAMETER TABLE


                                ;**********
                                ;
                                ;DH11 PROGRAMMABLE PRAMETER TABLE
                                ;
                                ;THIS TABLE IS PROVIDED TO ALLOW THE DEGREE OF FREEDOM AVAILABLE
                                ;IN THE DH11 PROGRAMMABLE MULTIPLEXER. THERE ARE 256 ENTRIES, ONE
                                ;FOR EACH OF THE POSSIBLE 16 LINES ON EACH OF THE POSSIBLE 16
                                ;DH11'S.  EACH OF THE 256 LINES ARE INITIALIZED AS FOLLOWS:(=16707)
                                ;       CHARACTER LENGTH        = 8 BITS
                                ;       NO. OF STOP BITS        = 2
                                ;       PARITY GEN. AND DET.    NONE
                                ;       OPERATING MODE          FULL DUPLEX
                                ;       TRANSMITTER SPEED       = 300 BAUD
                                ;       RECEIVER SPEED  = 300 BAUD
                                ;
                                ;IF ANY LINE SHOULD DIFFER, THE APPROPRIATE ENTRY FOR THE LINE
                                ;SHOULD BE CHANGED BEFORE RUNNING THE DIAGNOSTIC.  REFER TO THE LINE
                                ;PARAMETER REGISTER IN THE DH11 PROGRAMMING MANUAL FOR INSTRUCTIONS.
                                ;
                                ;**********

015560                          DH1100:
015560  016707                          .WORD   16707
015562  016707                          .WORD   16707
015564  016707                          .WORD   16707
015566  016707                          .WORD   16707
015570  016707                          .WORD   16707
015572  016707                          .WORD   16707
015574  016707                          .WORD   16707
015576  016707                          .WORD   16707
015600  016707                          .WORD   16707
015602  016707                          .WORD   16707
015604  016707                          .WORD   16707
015606  016707                          .WORD   16707
015610  016707                          .WORD   16707
015612  016707                          .WORD   16707
015614  016707                          .WORD   16707
015616  016707                          .WORD   16707
015620                          DH1101:
015620  016707                          .WORD   16707
015622  016707                          .WORD   16707
015624  016707                          .WORD   16707
015626  016707                          .WORD   16707
015630  016707                          .WORD   16707
015632  016707                          .WORD   16707
015634  016707                          .WORD   16707
015636  016707                          .WORD   16707
015640  016707                          .WORD   16707
015642  016707                          .WORD   16707
015644  016707                          .WORD   16707
```

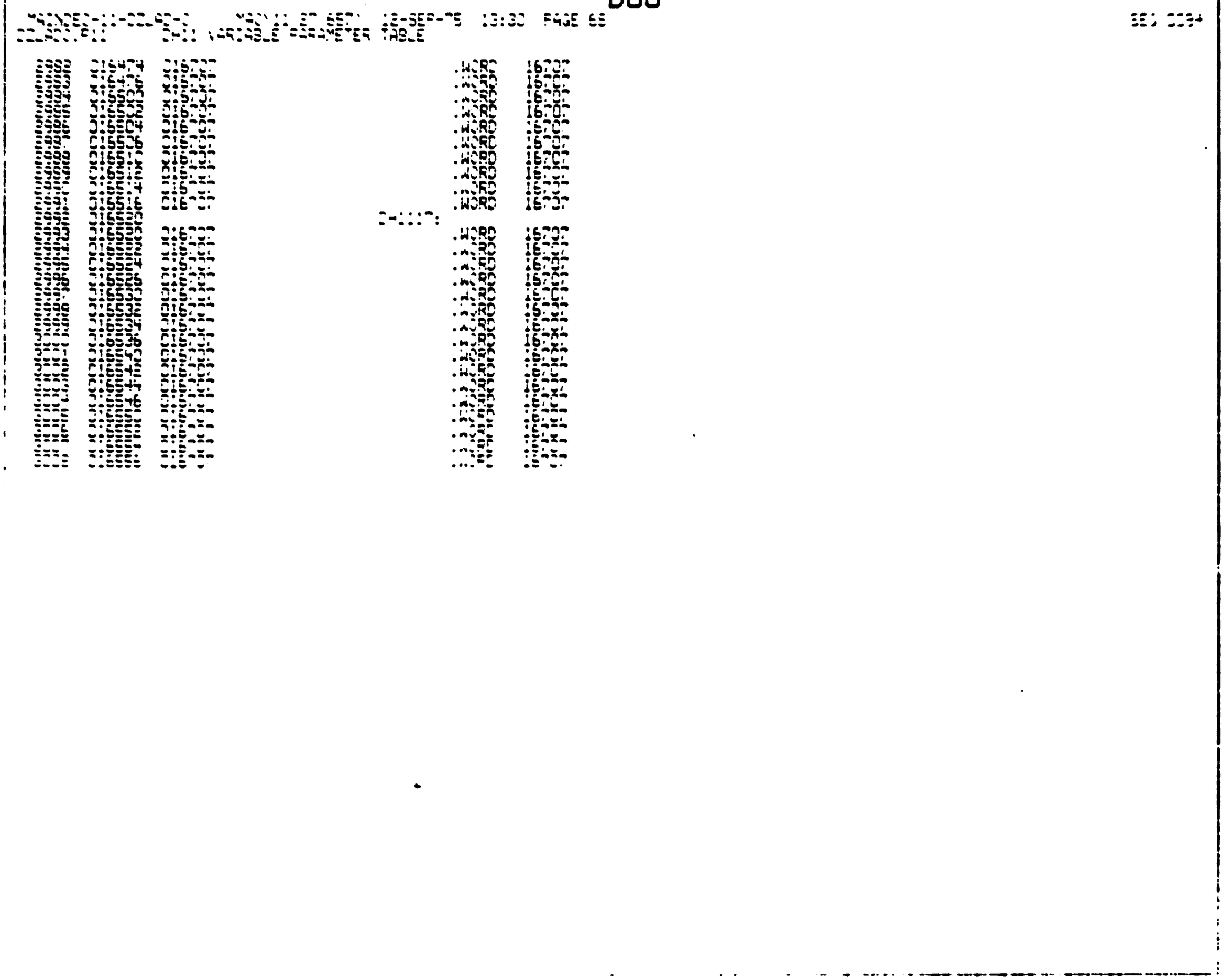
```
015646  016707                          .WORD   16707
015650  016707                          .WORD   16707
015652  016707                          .WORD   16707
015654  016707                          .WORD   16707
015656  016707                          .WORD   16707
015660                          DH11D2:
015660  016707                          .WORD   16707
015662  016707                          .WORD   16707
015664  016707                          .WORD   16707
015666  016707                          .WORD   16707
015670  016707                          .WORD   16707
015672  016707                          .WORD   16707
015674  016707                          .WORD   16707
015676  016707                          .WORD   16707
015700  016707                          .WORD   16707
015702  016707                          .WORD   16707
015704  016707                          .WORD   16707
015706  016707                          .WORD   16707
015710  016707                          .WORD   16707
015712  016707                          .WORD   16707
015714  016707                          .WORD   16707
015716  016707                          .WORD   16707
015720                          DH11D3:
015720  016707                          .WORD   16707
015722  016707                          .WORD   16707
015724  016707                          .WORD   16707
015726  016707                          .WORD   16707
015730  016707                          .WORD   16707
015732  016707                          .WORD   16707
015734  016707                          .WORD   16707
015736  016707                          .WORD   16707
015740  016707                          .WORD   16707
015742  016707                          .WORD   16707
015744  016707                          .WORD   16707
015746  016707                          .WORD   16707
015750  016707                          .WORD   16707
015752  016707                          .WORD   16707
015754  016707                          .WORD   16707
015756  016707                          .WORD   16707
015760                          DH11D4:
015760  016707                          .WORD   16707
015762  016707                          .WORD   16707
015764  016707                          .WORD   16707
015766  016707                          .WORD   16707
015770  016707                          .WORD   16707
015772  016707                          .WORD   16707
015774  016707                          .WORD   16707
015776  016707                          .WORD   16707
016000  016707                          .WORD   16707
016002  016707                          .WORD   16707
016004  016707                          .WORD   16707
016006  016707                          .WORD   16707
016010  016707                          .WORD   16707
016012  016707                          .WORD   16707
```

```
016014  016707                          .WORD   16707
016016  016707                          .WORD   16707
016020                          DH1105:
016020  016707                          .WORD   16707
016022  016707                          .WORD   16707
016024  016707                          .WORD   16707
016026  016707                          .WORD   16707
016030  016707                          .WORD   16707
016032  016707                          .WORD   16707
016034  016707                          .WORD   16707
016036  016707                          .WORD   16707
016040  016707                          .WORD   16707
016042  016707                          .WORD   16707
016044  016707                          .WORD   16707
016046  016707                          .WORD   16707
016050  016707                          .WORD   16707
016052  016707                          .WORD   16707
016054  016707                          .WORD   16707
016056  016707                          .WORD   16707
016060                          DH1106:
016060  016707                          .WORD   16707
016062  016707                          .WORD   16707
016064  016707                          .WORD   16707
016066  016707                          .WORD   16707
016070  016707                          .WORD   16707
016072  016707                          .WORD   16707
016074  016707                          .WORD   16707
016076  016707                          .WORD   16707
016100  016707                          .WORD   16707
016102  016707                          .WORD   16707
016104  016707                          .WORD   16707
016106  016707                          .WORD   16707
016110  016707                          .WORD   16707
016112  016707                          .WORD   16707
016114  016707                          .WORD   16707
016116  016707                          .WORD   16707
016120                          DH1107:
016120  016707                          .WORD   16707
016122  016707                          .WORD   16707
016124  016707                          .WORD   16707
016126  016707                          .WORD   16707
016130  016707                          .WORD   16707
016132  016707                          .WORD   16707
016134  016707                          .WORD   16707
016136  016707                          .WORD   16707
016140  016707                          .WORD   16707
016142  016707                          .WORD   16707
016144  016707                          .WORD   16707
016146  016707                          .WORD   16707
016150  016707                          .WORD   16707
016152  016707                          .WORD   16707
016154  016707                          .WORD   16707
016156  016707                          .WORD   16707
016160                          DH1110:
```



C08


```
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]             C-1114:
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]             C-1115:
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]             C-1116:
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
[illegible]  [illegible]  [illegible]          .WORD   16707
```



```
                    .SBTL  EXISTING LINE TABLE


                    ;**********
                    ;
                    ;EXISTING LINE TABLE
                    ;
                    ;THIS 16 WORD TABLE, ONE ENTRY FOR EACH OF THE POSSIBLE 16 CHANNELS
                    ;IN BOTH INDICATES WHICH OF THE LINES HAVE A TERMINAL CONNECTED
                    ;THIS TABLE SHOULD BE INITIALIZED BEFORE RUNNING THE DIAGNOSTIC. SET
                    ;IS LINE #0 WHILE SET 15 IS LINE #15. A 1 INDICATES THAT A TERMINAL IS
                    ;NOT CONNECTED WHILE A 0 INDICATES THAT THE TERMINAL DOES EXIST ON THE
                    ;LINE.
                    ;
                    ;EX. IF THERE ARE TERMINALS ON LINE #0,#5,#10 AND#13 ONLY ON THE
                    ;FIRST CHAN AND NO OTHER CHAN EXIST, SET THE FIRST EXISTING LINE
                    ;TABLE ENTRIES AND ALL OTHER WORDS TO 1.  THE DIAGNOSTIC IS
                    ;INITIALIZED SO THAT NO TERMINALS EXIST.
                    ;
                    ;**********

ELTAB:  .WORD   177777          ;FIRST CHAN
        .WORD   177777
        .WORD   177777
        .WORD   177777
        .WORD   177777
        .WORD   177777
        .WORD   177777
        .WORD   177777
        .WORD   177777
        .WORD   177777
        .WORD   177777
        .WORD   177777
        .WORD   177777
        .WORD   177777
        .WORD   177777
        .WORD   177777
```

```
                                                .SBTTL  DIAGNOSTIC MESSAGES

016620  000000  042516  047111  STARTM: .ASCIZ  <CRLF>  <LF>  "MAINDEC-11-02LAC-C"  <CRLF>
016626  042000  000000  000000  000461
016634  042000  046101  042101
016640  042455  000000
016645     114  031501  000066          .ASCIZ  "LAB TERMINAL DIAGNOSTIC"  <CRLF>
016652  042524  046620  047111
016660  046101  042040  042511
016666  047104  051511  044524
016674  100103
016676  044104  000461  000040          .ASCIZ  "DH11 & DZ11 INTERFACES"  <CRLF>  <LF>
016704  042040  000516  000521
016712  047111  042524  042122
016720  041001  051505  005200
016726     000
016730     000  042412  042116  ENDPAS: .ASCIZ  <CRLF>  <LF>  "END OF PASS"
016737  047440  000000  040000
016745  051523  000040
016752  000000  000000  005200  PASMES: .ASCIZ  "0000"  <CRLF>  <LF>
016755     000
016757     000  000000  044104  DH11:   .ASCIZ  "DC DH11'S UNDER TEST"  <CRLF>  <LF>
016763  000461  051447  052440
016771  042116  051105  052040
016775  051505  100124  000012
017000  000000  042040  000512  DZ11:   .ASCIZ  "DC DZ11'S UNDER TEST"  <CRLF>  <LF>
017006  023461  020123  047105
017014  042504  020120  042524
017020  052123  000000  000000
017026     000  000501  042524  FORMSG: .ASCIZ  <CRLF>  <LF>  <LF>  "TEST #"
017033  052103  021440
017041     000  000000  041440  FORD:   .ASCIZ  "000 COLUMNS"  <CRLF>  <LF>
017047  046101  046525  051516
017055  052000     000
017057     194  000101  000105  EOBSMA: .ASCIZ  "TYPE ANY PRINTABLE CHARACTER
017065  047101  000000  000000
017073  047101  047000  046000
017101  020106  044103  051101
017107  000000  044000  000102
017115  047101  020000  044514          .ASCIZ  "AND LISTEN FOR BELL....................."
017123  052103  047106  042040
017131  051117  041040  046105
017137  000000  020000  000000
017145  027056  027056  027056
017153  027056  027056  027056
017161  000000  000000  000000  EOBSMB: .ASCIZ  <CRLF>  "IF BELLS COLUMNS"  <CRLF>
017167  000000  000000  000000
017175  000000  000000  000000
017203  000000  000000  000000  EOBPC:  .ASCIZ  <CRLF>  "DH11 TEST TERMINATED"  <CRLF>
017211  000000  000000  000000
017217  000000  000000  000000
```

G08


```
[illegible]   ESTADM: .ASCII
[illegible]   LINES:  .ASCII              ;MESG FOR TEST E023
[illegible]   LINESA: .ASCII          <CRLF>
[illegible]   MESG3:  .ASCII  <CRLF> IT IS SELECT TEST #
[illegible]
[illegible]   .EVEN   .ASCII  .EVEN
              .EVEN
[illegible]   .END
```

[illegible] =	000200	[illegible]	001232	ADTENF	004274	BAR	001150
[illegible] =	[illegible]	[illegible] =	000001	BIT1 =	000002	BIT10 =	002000
[illegible] =	[illegible]	[illegible] =	[illegible]	BIT13 =	020000	BIT14 =	040000
[illegible] =	[illegible]	[illegible] =	000004	BIT3 =	000010	BIT4 =	000020
[illegible] =	[illegible]	[illegible] =	000100	BIT7 =	000200	BIT8 =	000400
[illegible] =	[illegible]	CLOPAS =	104015	BUFR	012054	BYCR	001146
CHARA	001144	CHAIN =	104001	CHAINN	001746	CHAINY	001772
CHARAC	[illegible]	CLEAN =	104017	CNTDH	001154	CNTLSW	001104
[illegible]	[illegible]	CR =	104014	CRLF =	104006	CSR	001236
[illegible]	[illegible]	DELAY =	104003	DHADR	001134	DHCNT	001230
[illegible]	[illegible]	DH1100	015560	DH1101	015620	DH1102	015660
[illegible]	[illegible]	DH1104	015760	DH1105	016020	DH1106	016060
[illegible]	[illegible]	DH1110	016160	DH1111	016220	DH1112	016260
[illegible]	[illegible]	DH1114	016360	DH1115	016420	DH1116	016460
[illegible]	[illegible]	DIGIT	004270	DISPRE	000174	DJCNT	001226
[illegible]	[illegible]	[illegible]	004056	ECHO =	104021	ECHODJ	003462
[illegible]	[illegible]	[illegible]AB	016560	EMTINT	003256	EMTTAB	003332
[illegible]	[illegible]	[illegible]TD	005254	ENDITR	005246	ENDPAS	016720
[illegible]	[illegible]	ENDT	001446	E020	010756	E021	011022
[illegible]	[illegible]	E021B	011054	E022	011074	E022A	011102
[illegible]	[illegible]	E023	011360	E023M	017237	E024	017202
[illegible]	[illegible]	E024R	[illegible]	E025	012466	E025MA	017063
[illegible]	[illegible]	EVEN	017302	FORWD =	104016	HDRMSG	017033
[illegible]	[illegible]	HERE	010600	INRDY =	104022	IOSW	001234
[illegible] =	[illegible]	LFCNT	001132	LINENO	001156	LINES	006774
[illegible]	[illegible]	LINES	017241	LINESA	017245	LOGICA	010570
[illegible]	[illegible]	MACHER	[illegible]	MASK	001160	MESSG	017255
[illegible]	[illegible]	MCGES	011676	MONIC	011154	MOVNUM	011620
[illegible]	[illegible]	MRORA	001140	NXTST	001110	OOO	017306
[illegible] =	[illegible]	OUTPUT	012002	PARITY	001126	PASCNT	010754
[illegible]	[illegible]	P0	=%000007	PCHAR	001130	PFAIL	004100
[illegible] =	[illegible]	POST	001122	PRGTAB	003156	PRINTC=	104011
[illegible]	[illegible]	PRTHDR=	104012	PT0	006410	PT1	006464
[illegible]	[illegible]	PT11	010472	PT12	010524	PT12A	010532
[illegible]	[illegible]	PT17B	010604	PT17D	010626	PT2	006526
[illegible]	[illegible]	PT4	007356	PT5	007534	PT5A	007702
[illegible]	[illegible]	PT7	010122	READ =	104013	REF	007466
[illegible]	[illegible]	RESTRT	004136	RINNO	001106	R0	=%000000
[illegible]	[illegible]	R2	=%000002	R3	=%000003	R4	=%000004
[illegible]	[illegible]	SAVR6	004134	SCANDH	002604	SCANDJ	003016
[illegible]	[illegible]	SCRLF =	104007	SCR1	001232	SCR2	001234
[illegible]	[illegible]	SETERM	002660	SP	=%000006	SPBOT	001100
[illegible]	[illegible]	SPCNT	001116	SPEP	006556	SP2	006[illegible]
[illegible]	[illegible]	SSR	001152	START	001256	STARTM	016620
[illegible]RTX	[illegible]	START1	001246	STORE	013773	STRLN	011542
[illegible]RSG	[illegible]	TBUF	001242	TCR	001240	TEMPCH	001124
[illegible]AR	004272	TERM	012054	TESTC =	104020	TEST30	012570
[illegible]	[illegible]	TEST32	013570	TEST33	014020	TEST34	014554
[illegible]	[illegible]	TIMER	001102	TTYCTL=	104004	TTYJ	002022
[illegible]	[illegible]	TTYCTL=	104005	TTY1	002052	TTY1A	002054
[illegible]	[illegible]	TTY1G	002410	TTY1H	002450	TTY1L	002516
[illegible]	[illegible]	TYPE =	104000	TYPEM =	104002	TYPM	004036
[illegible]	[illegible]	[illegible]TH	001112	$BTASK	004202	$CLEAN	[illegible]
[illegible]	[illegible]	$CRLF	004356	$ECHO	[illegible]	[illegible]	[illegible]

$ERR=	000536	$LF	000363	$ERHDR	004400	$PRTC	004454
$REG	000550	$SCRLF	004343	$TEST	004306	.	= 017212

ERRORS DETECTED: 0

[illegible]
[illegible] SECTORS
RUN TIME: [illegible]

Spooler runtime 11 Seconds, 348 KCS, 332 disk reads, 3 disk writes, 99 pages

[illegible] Date 14-Oct-76 16:23:47 Monitor IPC-0 [illegible] (103) [illegible]

```
0000000000
```