

LXY11
LXY21, LXV11

LXY11/21-LXV11 DIA
CZLXAAO

AH-SB24A-MC
FICHE 1 OF 1

AUG 1981
COPYRIGHT © 1981
MADE IN USA



.NLIST SEQ,BIN,LOC
.REM_

IDENTIFICATION

PRODUCT CODE: AC-S822A-MC
PRODUCT NAME: CZLXAA0 LXY11/21-LXV11 DIA
DATE : MARCH 11, 1981
MAINTAINER: CSS PRODUCT GROUP DIAGNOSTIC
AUTHOR: JIM DUPRE

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 (C) 1981 BY DIGITAL EQUIPMENT CORPORATION

CONTENTS

- 1.0 ABSTRACT
- 2.0 REQUIREMENTS
- 2.1 EQUIPMENT
- 2.2 STORAGE
- 2.3 PRELIMINARY PROGRAMS
- ~~3.0 LOADING PROCEDURE~~
- 3.1 METHOD
- 4.0 STARTING PROCEDURE
- 4.1 CONTROL SWITCH SETTINGS
- 4.2 STARTING ADDRESS OR ADDRESSES
- 4.3 PROGRAM AND/OR OPERATOR OPERATION
- 5.0 OPERATING PROCEDURE
- 5.1 OPERATIONAL SWITCH SETTINGS
- 5.2 ABSENCE OF HARDWARE SWITCH REGISTER
- 5.3 IOT CHANGES
- 6.0 ERRORS
- 6.1 COMPUTER DETECTED ERRORS
- 6.2 VISUALLY DETECTED ERRORS

7.0 TEST DESCRIPTIONS

7.1 TEST 1 CONTROL AND OPERATOR INTERACTION

- 7.1.1 TEST 1 SECTION 1 PRINTER READY, TESTS POWER UP
- 7.1.2 TEST 1 SECTION 2 MANUAL PRINT SPEED TEST

7.2 PRINTING TESTS

- 7.2.1 TEST 2 DATA TRANSFER PATHS TEST
- 7.2.2 TEST 3 CHARACTER GENERATOR AND COMPARATOR TESTS
- 7.2.3 TEST 4 UNDERLINE TEST
- 7.2.4 TEST 5 LINE EDITING TEST
- 7.2.5 TEST 6 PRINT CONTROL TEST
- 7.2.6 TEST 7 MULTIPLE LINE ADVANCE TEST
- 7.2.7 TEST 8 HIGH SPEED PRINT TEST
- 7.2.8 TEST 9 SINGLE CHARACTER, ALL COLUMNS TEST
- 7.2.9 TEST 10 PROM PATTERN TEST
- 7.2.10 TEST 11 RIGHT & LEFT HAND WEDGES
- 7.2.11 TEST 12 HAMMER TEST
- 7.2.12 TEST 13 ELONGATED CHARACTERS TEST
- 7.2.13 TEST 14 UNDERLINED ELONGATED CHARACTERS TEST
- 7.2.14 TEST 15 ALTERNATE CHARACTER SET TEST
- 7.2.15 EVFU CHANNEL SLEWING TEST
- 7.2.16 PLOT TEST

7.3 SCOPE DRIVE TEST

1.0 ABSTRACT

THE LINE PRINTER DIAGNOSTIC PROGRAM IS DIVIDED INTO THREE SECTIONS. INTERNALLY DETECTED ERROR CONDITIONS ARE DISPLAYED ON THE TELEPRINTER, WHILE BRIEF DESCRIPTIONS OF EACH ERROR ARE PRESENTED IN THE LISTING. PRINT PATTERNS USED IN THESE TESTS HAVE BEEN CHOSEN FOR EASE OF VISUAL VERIFICATION.

THE FIRST SECTION IS DESIGNED TO CHECK-OUT THE PROCESSOR INTERFACE CONTROL ELECTRONICS AND THE INTER-COMMUNICATIONS DATA PATHS. IT WILL ALSO PERFORM ALL TESTS THAT REQUIRE OPERATOR INTERVENTION. THE SECOND SECTION IS A PRINTING TEST DESIGNED TO TEST THE LINE PRINTER MECHANISM ITSELF. THE LAST SECTION IS A SCOPE DRIVER ROUTINE FOR USE IN TROUBLE SHOOTING THE PRINTER INTERFACE.

2.0 REQUIREMENTS

2.1 EQUIPMENT

THIS DIAGNOSTIC SHOULD RUN ON ALL PDP-11 FAMILY COMPUTERS HAVING LINE PRINTER CONTROLS, LINE PRINTERS, AND TELETYPES COMPATIBLE WITH THE FOLLOWING:

LXY11/21 OR LXV11	LINE PRINTER INTERFACE
LXY11	PRINTRONIX MODEL 300 LINE PRINTER
	OR
LXY21	PRINTRONIX MODEL 600 LINE PRINTER
	OR
LXV11	PRINTRONIX MODEL 300 LINE PRINTER
TELETYPE	MODEL 33 OR EQUIVALENT CONSOLE UNIT

2.2 STORAGE

MEMORY LOCATIONS 0 - TO - 17004 ARE USED BY THIS DIAGNOSTIC.

2.3 PRELIMINARY PROGRAMS

ALL APPLICABLE PDP-11 DIAGNOSTICS SHOULD RUN ON THE PROCESSOR AND TELETYPE.

3.0 LOADING PROCEDURE

3.1 METHOD

POWER DOWN THE LINE PRINTER

POWER UP THE PROCESSOR ONLY

LOAD THE BOOTSTRAP AND ABSOLUTE LOADERS

LOAD THE LXY11/21 OR LXV11 DIAGNOSTIC PROGRAM TAPE

4.0 STARTING PROCEDURE

4.1 CONTROL SWITCH SETTINGS

SET CONTROL SWITCHES AS DESIRED - (SEE SECTION 5.1 FOR DESCRIPTION OF SWITCH FUNCTIONS) MAKE SURE SWITCH 0 IS DOWN BEFORE STARTING THE TEST.

4.2 STARTING ADDRESS OR ADDRESSES

THE INITIAL STARTING ADDRESS TO RUN THE ENTIRE LXY11/21-LXV11 DIAGNOSTIC IS LOCATION 200(8). TO SKIP THE OPERATOR INTERVENTION TESTS AND START WITH THE PRINTING TESTS, START AT LOCATION 600(8). TO RUN THE SPECIAL SCOPE DRIVER ROUTINE USE START ADDRESS 700(8) OR 720(8). TO START AT ANY OTHER TEST USE THE START ADDRESS FROM THE FOLLOWING TABLE:

START ADDRESS	TEST
300	EVFU CHANNEL SLEW TEST
304	PLOT TEST
400	PRINT SPEED TEST USING MANUAL TIMING
404	PRINT SPEED TEST USING KW11-L
410	PRINT SPEED TEST USING KW11-P

600	TEST 2 INTERFACE & DATA PATHS TEST (ALSO GENERAL PRINT TEST STARTING ADDRESS)
610	TEST 3 CHAR COMPARATOR TEST
614	TEST 4 UNDERLINE TEST
620	TEST 5 LINE EDITING TEST
624	TEST 6 PRINT CONTROL TEST
630	TEST 7 MULTIPLE LINE ADVANCE TEST
634	TEST 8 HIGH SPEED PRINT TEST
640	TEST 9 SINGLE CHAR, ALL COLUMNS
644	TEST 10 DRUM PATTERN CHAR TEST
650	TEST 11 SPURIOUS HAMMER FIRING TESTS (LEFT & RIGHT WEDGES)
654	TEST 12 HAMMER ALIGNMENT
660	TEST 13 ELONGATED CHARACTERS TEST
664	TEST 14 UNDERLINED ELONGATED CHARACTER TEST
670	TEST 15 ALTERNATE CHARACTER SET TEST
700	SCOPE DRIVER ROUTINE
720	SCOPE DRIVER WITHOUT LINE FEEDS

THE PROGRAM WILL START THROUGH THE TEST SEQUENCE BEGINNING WITH THE SELECTED TEST UNLESS SWITCH 12 IS SET TO LOOP ON TEST (SEE SECTION 5.1)

4.3 PROGRAM AND/OR OPERATOR ACTION

DURING INITIAL START-UP OF THE LINE PRINTER DIAGNOSTIC TEST, THE HEADER MESSAGE 'LXY11/21-LXV11 LINE PRINTER TEST' WILL BE TYPED OUT ON THE TELEPRINTER FOLLOWED BY EXECUTION OF THE PRINTER READY PORTION OF TEST 1. PRINTING OF THE MESSAGE 'POWER-UP' ON THE TELEPRINTER FOLLOWING THE TEST HEADER PRINT-OUT INDICATES START OF THIS TEST SEQUENCE. THIS TEST IS CARRIED OUT BY AN INTERACTIVE EXCHANGE BETWEEN THE OPERATOR AND THE TEST PROGRAM. THE OPERATIONAL DESCRIPTION OF THIS TEST APPEARS AS PART OF THE TEST DESCRIPTION FOR TEST 1 (SEE SECTION 7.1.1). AFTER SUCCESSFUL COMPLETION OF THIS SECTION OF TEST 1, THE PRINT SPEED TEST WILL BE PERFORMED. (SEE SECTION 7.1.2) IF THE EVFU IS AVAILABLE AND SWITCH 14 IS SET, THE EVFU TESTS WILL ALSO BE PERFORMED. AFTER COMPLETION OF ALL OF TEST 1, PRESS CONTINUE TO ENTER THE PRINTING TESTS DIRECTLY. NO OTHER OPERATOR ACTION WILL BE REQUIRED.

5.0 OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

THE USE OF THIS PROGRAM ON PROCESSORS NOT HAVING A HARDWARE SWITCH REGISTER NECESSITATES OPERATOR INTERACTION; THE OPERATOR MUST SET UP LOCATION 174 WITH THE SOFTWARE DISPLAY VALUES AND LOCATION 176 WITH THE SOFTWARE SWITCH VALUES. (SEE SECTION 5.2)

SWITCH	FUNCTION IN 'UP' POSITION
15	LOOP ON ERROR (IN TEST 1 ONLY)
14	TEST EVFU FUNCTION
13	DOWN - SKIP ALTERNATE CHARACTER SET TEST UP - DO ALTERNATE CHARACTER SET TEST
12	LOOP ON TEST
11	SEND ONLY ONE CHARACTER TO LINE PRINTER IN SCOPE DRIVER - THEN HALT
10	TEST PLOT FUNCTION
0	USED FOR PRINT SPEED MANUAL TIMING IF NO CLOCK AVAILABLE

1. SWITCH - 0

TO START PRINTING IN THE MANUAL PRINT SPEED TEST, PLACE SWITCH 0 IN THE UP POSITION. AT THE END OF ONE MINUTE PUT SWITCH 0 DOWN. THE APPROXIMATE PRINT SPEED WILL BE PRINTED ON BOTH THE LINE PRINTER AND THE TELEPRINTER. SWITCH 0 IS NOT USED IN ANY OTHER TESTS. MAKE SURE SWITCH 0 IS DOWN AT THE START OF THE TEST IF USING MANUAL TIMING OR UP IF USING AN INTERNAL CLOCK OPTION (KW11-L OR KW11-P).

2. SWITCH - 10

SWITCH 10 SHOULD BE SET IN THE UP FOR TESTING THE PLOT FUNCTION.

3. SWITCH - 11

SWITCH 11 IN THE UP POSITION CAUSES THE CONTENTS OF THE SWITCH REGISTER TO BE SENT ONLY ONCE TO THE LINE PRINTER THEN HALT IN THE SCOPE DRIVER ROUTINE. TO SEND ANOTHER CHARACTER, RESET SWITCHES AND DEPRESS CONTINUE. WITH SWITCH 11 DOWN, THE SWITCH REGISTER IS SENT CONTINUOUSLY TO THE LINE PRINTER WITH A LINE FEED SENT AFTER EVERY 132 CHARACTERS. TO STOP SENDING CHARACTERS, PUT SWITCH 11 UP.

4. SWITCH - 12

SWITCH 12 IN THE UP POSITION IS USED TO AUTOMATICALLY LOOP ON THE CURRENT TEST IF IN TESTS 2 TO 15. PLACING SWITCH 12 IN THE UP POSITION WILL FORCE THE PROGRAM TO CONSTANTLY LOOP ON THE CURRENT TEST. REPLACING THE SWITCH TO THE DOWN POSITION WILL MAKE THE PROGRAM RESUME ITS NORMAL TEST SEQUENCE AT THE COMPLETION OF THE CURRENT TEST.

5. SWITCH - 13

SWITCH 13 SHOULD BE SET UP IF THE ALTERNATE CHARACTER SET IS AVAILABLE. IF THE ALTERNATE SET IS NOT USED SWITCH 13 SHOULD BE DOWN.

6. SWITCH - 14

SWITCH 14 SHOULD BE SET UP IF THE EVFU IS TO BE TESTED.

7. SWITCH - 15

WITH SWITCH 15 IN THE DOWN POSITION THE PROGRAM WILL HALT AFTER AN ERROR TYPE OUT IN TEST 1. WITH SWITCH 15 IN THE UP POSITION, THE PROGRAM WILL LOOP ON THE ERROR IN TEST 1.

REFER TO SECTION 6.1 TO CONTINUE AFTER AN ERROR DURING ANY OTHER TESTS.

5.2 ABSENCE OF HARDWARE SWITCH REGISTER

WHEN THE DIAGNOSTIC IS STARTED AT ADDRESS 200(8), IT WILL DETERMINE IF THE PROCESSOR HAS A HARDWARE (H/W) SWITCH REGISTER (SWR). IF THERE IS NO H/W SWR, THE DIAGNOSTIC WILL USE THE SOFTWARE (S/W) SWR LOCATED AT ADDRESS 176(8).

THE DIAGNOSTIC WILL PROMPT THE OPERATOR WITH THE MESSAGE:

SWR = XXXXXX NEW SWR =

THE FIRST TIME THE SWR VALUE IS NEEDED, ANY TIME THEREAFTER, THE OPERATOR MAY CHANGE THE VALUE OF THE SWR BY ENTERING A CONTROL-G (^G) AT THE CONSOLE.

IF THERE IS NO H/W SWITCH REGISTER AND THE DIAGNOSTIC IS TO BE STARTED AT AN ADDRESS OTHER THAN 200(8):

ENTER THE NUMBER 176(8) IN LOCATION 1004(8)

ENTER THE INITIAL VALUE OF THE SWR IN LOCATION 176(8).

AFTER EXECUTION BEGINS, THE OPERATOR MAY CHANGE, AT ANY TIME,

THE VALUE OF THE SWR BY ENTERING A CONTROL-G (^G) AT THE CONSOLE.

THE FOLLOWING COMMANDS ALLOW THE OPERATOR TO MODIFY THE S/W SWR:

CONTROL-G (^G): ALLOWS MODIFICATION OF THE S/W SWR. ENTERING A ^G WILL RESULT IN THE FOLLOWING MESSAGE OUTPUT AT THE CONSOLE

SWR = XXXXXX NEW SWR =

THE OPERATOR MAY THEN ENTER UP TO SIX (6) OCTAL DIGITS. THE DIGITS MAY BE ANY COMBINATION OF :0,1,2,3,4,5,6,7,OR NO ENTRY AT ALL. ALL SWR VALUES ENTERED WILL BE TRUNCATED TO THE LOWER SIXTEEN (16) BITS. ENTERING MORE THAN SIX DIGITS OR A CHARACTER OTHER THAN A DIGIT RESULTS IN A "?" OUTPUT ON THE CONSOLE AND A REPEAT OF THE PROMPTING MESSAGE.

CARRIAGE RETURN (CR): ENTERS THE NEW SWR VALUE. IF NO DIGITS HAVE BEEN ENTERED, THE SWR VALUE REMAINS UNCHANGED.

CONTROL-U (^U): ERASES THE SWR VALUE BEING ENTERED. A CARRIAGE RETURN AND LINE FEED WILL BE OUTPUT AT THE CONSOLE. THE CORRECT SWR VALUE MAY THEN BE ENTERED.

ENTERING ANY CHARACTER BEFORE A CONTROL-G (^G) HAS BEEN ENTERED WILL RESULT IN A "?" OUTPUT AT THE CONSOLE.

NOTE: IT IS POSSIBLE FOR THE DIAGNOSTIC TO OUTPUT MESSAGES AT THE CONSOLE BEFORE THE NEW SWR VALUE HAS BEEN ENTERED. SHOULD THIS HAPPEN, THE OPERATOR SHOULD ENTER A CONTROL-U (^U) AND THEN ENTER THE CORRECT SWR VALUE.

5.3 IOT CHANGES

THE LINE PRINTER STATUS IS LOCATION 177514 AS USED BY THE PROGRAM.
THE LINE PRINTER VECTOR ADDRESS IS LOCATION 200 AS USED BY THE PROGRAM.
THE LINE PRINTER PSW IS LOCATION 202 AS USED BY THE PROGRAM.
THE LINE PRINTER BUFFER IS LOCATION 177516 AS USED BY THE PROGRAM.

FOR OTHER THAN THESE, PLACE THE CORRECT STATUS LOCATION IN LOCATION 1000(8) AND THE CORRECT BUFFER LOCATION IN LOCATION 1002(8), THE CORRECT VECTOR ADDRESS IN LOCATION 1030(8) AND THE CORRECT PSW IN LOCATION 1032(8).

6.0 ERRORS

6.1 COMPUTER DETECTED ERRORS

THE FOLLOWING DISCUSSION DESCRIBES (IN GENERAL) THE METHOD USED FOR INTERNAL ERROR DETECTION AND ERROR DISPLAY BY THE LINE PRINTER DIAGNOSTIC PROGRAM. MONITORING OF THE CURRENT CONDITION OF THE READY LINE AFTER EACH OPERATION IS CARRIED ON CONTINUOUSLY DURING ALL TESTS WHERE APPROPRIATE AND IS DESCRIBED IN THE FOLLOWING PARAGRAPHS. HOWEVER, ADDITIONAL TESTING IS PERFORMED ESPECIALLY DURING EXECUTION OF THE FIRST TEST. FOR A COMPLETE DESCRIPTION OF THE TESTING PROCEDURES USED IN TEST 1 AND THE CORRESPONDING ERROR CONDITIONS, THE READER IS REFERRED TO THE DESCRIPTION OF THE TEST AND THE TEST LISTING.

ERROR PRINT-OUTS ARE LIMITED TO THE ERROR NUMBER (ERROR COUNT). ADDITIONAL INFORMATION MAY BE OBTAINED FROM THE TEST DESCRIPTION OR FROM THE LISTING. TO FIND THE ERROR IN THE LISTING, SEE THE SYMBOL TABLE AT THE END OF THE LISTING TO FIND THE LOCATION OF THE ERROR.

ERROR TAGS WILL BE LISTED AS 'ERRXX' WHERE XX = ERROR NUMBER.

IN GENERAL, THE TEST PROGRAM MONITORS PROPER OPERATION OF THE LINE PRINTER AFTER EACH PRINTER OPERATION HAS BEEN COMPLETED, THROUGH THE PRINTER 'READY' LINE AND THE SETTING OF THE CHARACTER FLAG OF THE PRINTER 'DEMAND' RETURN LINE. WITH REGARDS TO THE READY LINE, THE FOLLOWING ERROR CONDITIONS, IF DETECTED WITHIN THE LINE PRINTER ITSELF, WILL CAUSE THE READY LINE TO DROP:

1. PAPER OUT OR TORN
2. FORMS THICKNESS LEVER OPEN (UP).
3. RIBBON STALL CONDITION
4. POWER SUPPLY FAULT
5. HAMMER BANK FAULT
6. EVFU ERROR
7. SWITCHED OFF LINE

IT SHOULD BE NOTED THAT THE 'DEMAND' RETURN FROM THE PRINTER IS CONDITIONAL UPON THE PRINTER 'READY' AND THEREFORE THESE ITEMS SHOULD BE CHECKED FIRST IN CASE OF DIFFICULTY.

6.2 VISUALLY DETECTED ERRORS

SINCE THE COMPUTER CAN ONLY DETECT THE CURRENT CONDITION OF THE READY AND DEMAND RETURN LINES AND DOES NOT RECEIVE ANY ADDITIONAL DATA BACK FROM THE LINE PRINTER, IT IS NECESSARY TO EXAMINE THE PRINT PATTERNS PRODUCED BY THE VARIOUS TEST ROUTINES OR RESORT TO MANUAL SCOPING PROCEDURES, AS PROVIDED BY THE SCOPE DRIVER ROUTINE, TO DETECT AND DIAGNOSE ADDITIONAL DIFFICULTIES. DETAILED DESCRIPTIONS OF EACH TEST PATTERN APPEARS IN THE DESCRIPTION OF THE CORRESPONDING TEST ROUTINES.

7.0 TEST DESCRIPTIONS

7.1 TEST 1 - CONTROL TESTS AND OPERATOR INTERACTIVE TESTS

TEST 1 IS MADE UP OF TWO SECTIONS LINKED TOGETHER AND EXECUTED IN SEQUENCE AS A SINGLE TEST. THE FOLLOWING DESCRIPTIONS TREAT EACH SECTION SEPARATELY.

7.1.1 TEST 1 - SECTION 1 - COMMAND DECODE, CONTROL INTERFACE

THIS PORTION OF TEST 1 IS DESIGNED AS A COMMAND DECODE AND CONTROL INTERFACE TEST AND INCLUDES CHECKOUT OF THE PRINTER INTERRUPT FACILITY. UPON INITIAL ENTRY INTO THIS ROUTINE. MANUAL INTERVENTION IS REQUIRED TO TEST THE VARIOUS TESTABLE ERROR (NON-READY) CONDITIONS OF THE PRINTER. THE OPERATING SEQUENCE IS DESCRIBED IN DETAIL BELOW.

THE PRINTER READY LINE CONTINUOUSLY MONITORS THE FOLLOWING CONDITIONS WITHIN THE PRINTER AND ITS TRUE STATE AT THE CONTROL ELECTRONICS INTERFACE IS CONDITIONAL UPON NONE OF THEM EXISTING:

1. PAPER OUT OR TORN
2. FORMS THICKNESS LEVER OPEN (UP).
3. RIBBON STALL CONDITION
4. POWER SUPPLY FAULT
5. HAMMER BANK FAULT
6. EVFU ERROR
7. SWITCHED OFF LINE

THE MANUAL-INTERACTIVE TEST SEQUENCE WHICH FOLLOWS IS DESIGNED TO TEST THE PROPER OPERATION OF THE READY LINE AS IT APPEARS AT THE INTERFACE ELECTRONICS WITH RESPECT TO THOSE OF THE ABOVE ITEMS WHICH ARE TESTABLE (I.E. - A,B,F&G) INITIAL MANUAL TEST SEQUENCE:

1. AFTER "POWER ON - TURN ON LINE" HAS BEEN TYPED ON THE TELEPRINTER BRING POWER - UP ON THE LINE PRINTER AND TURN ON LINE, MAKING SURE THAT THE PAPER IS IN PLACE IN THE TRACTORS AND THAT THE FORMS THICKNESS LEVER IS CLOSED (DOWN) .

2. DEPRESS CONTINUE, 'READY SET OK - TRY TORN PAPER SWITCH' WILL BE TYPED OUT IF PRINTER IS ON LINE AND NO ERRORS EXIST.
3. PAPER - TEAR THE PAPER OFF BELOW THE PRINTER FORMS THICKNESS LEVER AND USE THE MANUAL TOP OF FORM SWITCH TO DRIVE ALL THE PAPER OUT OF THE PRINTER AND OBSERVE THAT THE PRINTER READY LIGHT GOES OUT AND THE CHECK LIGHT GOES ON ON THE PRINTER CONTROL PANEL. ATTEMPT TO PLACE THE PRINTER ON LINE. THE ON-LINE AND READY LIGHTS ON THE PRINTER CONTROL PANEL SHOULD REMAIN OFF.
4. DEPRESS CONTINUE, AN ERROR TYPE-OUT (ERROR COUNT 2) WILL OCCUR IF THE PRINTER READY LINE REMAINS HIGH AT THE INTERFACE ELECTRONICS.
5. READY - AFTER SUCCESSFUL COMPLETION OF STEPS 3 AND 4 THE MESSAGE 'ERROR SET OK - TURN ON LINE' WILL BE TYPED. RESTORE PAPER TO THE TRACTORS, CLOSE THE FORMS THICKNESS LEVER AND PLACE THE PRINTER IN THE READY-ON LINE STATE. OBSERVE THAT BOTH THE ON-LINE AND READY LIGHTS COME ON ON THE PRINTER CONTROL PANEL.
6. DEPRESS CONTINUE, AN ERROR TYPE-OUT (ERROR COUNT 4) WILL OCCUR IF THE PRINTER READY LINE DOES NOT GO HIGH AT THE INTERFACE ELECTRONICS.
7. FORMS THICKNESS LEVER - AFTER SUCCESSFUL COMPLETION OF STEPS 5 & 6 THE MESSAGE 'READY SET OK-TRY, FORMS THICKNESS LEVER SWITCH' WILL BE TYPED. OPEN THE PRINTER FORMS THICKNESS LEVER AND OBSERVE THAT THE ON-LINE AND READY LIGHTS GO OUT AND THE CHECK ERROR LIGHT GOES ON ON THE PRINTER CONTROL PANEL.
8. DEPRESS CONTINUE, AN ERROR TYPE-OUT (ERROR COUNT 5) WILL OCCUR IF THE PRINTER READY LINE APPEARS TO REMAIN HIGH AT THE INTERFACE ELECTRONICS.
9. READY - AFTER SUCCESSFUL COMPLETION OF STEPS 7 & -8 THE MESSAGE 'ERROR SET OK - TURN ON LINE' WILL BE TYPED.
10. DEPRESS CONTINUE TO COMPLETE THE COMMAND AND REGISTER TESTING ALONG WITH THE INTERRUPT TESTING. IF ANY ERROR CONDITIONS EXIST, ERROR TYPE-OUTS GIVING THE ERROR COUNT WILL BE PRINTED. CHECK THE LISTING FOR DESCRIPTIONS OF THESE ERRORS.
11. SECTION 2 OF TEST 1 WILL BE ENTERED DIRECTLY UPON COMPLETION OF SECTION 1.

7.1.2 TEST 1 - SECTION 2 - PRINT SPEED TIMING TEST.

THIS SECTION OF TEST 1 IS DESIGNED TO TIME THE PRINTER FOR ONE FULL MINUTE. DURING THIS TIME THE PRINTER WILL PRINT THE DIAGNOL OF THE PROM PATTERN SO THAT ONLY TWO HAMMERS (MAXIMUM) WILL FIRE AT ANY GIVEN INSTANT AND MAXIMUM PRINT TIME IS USED FOR EACH LINE.

IF A KW11-L OR KW11-P ARE AVAILABLE THEY WILL BE USED TO TIME THE PRINTER. IF BOTH ARE AVAILABLE, THE KW11-L WILL BE USED. IF NEITHER ARE AVAILABLE, MANUAL TIMING WILL BE USED. WHEN MANUAL TIMING IS USED INSTRUCTIONS WILL BE TYPED ON THE TELEPRINTER. TO START THE TIMING PLACE SWITCH 0 IN THE UP POSITION. AT THE END OF ONE FULL MINUTE PLACE SWITCH 0 IN THE DOWN POSITION TO STOP THE TIMING. IF USING AN INTERNAL CLOCK FOR TIMING, PLACE SWITCH 0 IN THE UP POSITION BEFORE STARTING THE TEST. WHICH EVER METHOD OF TIMING IS USED, AT THE END OF ONE FULL MINUTE THE APPROXIMATE PRINT SPEED WILL BE TYPED ON BOTH THE TELEPRINTER AND LINE PRINTER.

IF BOTH A KW11-L OR KW11-P ARE AVAILABLE OR IT IS DESIRED TO MANUALLY TIME THE PRINTER IF EITHER IS AVAILABLE USE THE FOLLOWING START ADDRESSES TO RUN THE DESIRED PRINT SPEED TIMING TEST:

400 FOR MANUAL TIMING
404 FOR KW11-L
410 FOR KW11-P

NOTE: IF THE LINE FREQUENCY IS 50 HZ. CHANGE THE CONTENTS OF 'MINCNT TO 5670(8) REFER TO THE END OF THE PRINTING ROUTINE. (SEARCH FOR 'MINCNT' IN THE CROSS REFERENCE LISTING)

7.2 LINE PRINTER PRINTING TESTS

TESTS 2 TO 15 PRODUCE VARIOUS PRINT PATTERNS DESIGNED FOR EASE OF VISUAL VERIFICATION. THESE TESTS CHECK ALL OF THE VARIOUS PRINTING ASPECTS OF THE PRINTER. DETAILED DESCRIPTIONS OF EACH INDIVIDUAL TEST FOLLOWS.

7.2.1 TEST 2 - DATA TRANSFER PATHS TEST

THIS TEST IS DESIGNED TO TEST THE DATA TRANSFER PATHS (WITH ALTERNATING ONES AND ZEROS), FROM THE PROCESSOR INTERFACE, THRU THE LINE PRINTER INPUT REGISTER, AND INTO THE PRINTER'S BUFFER. AN ALTERNATING STRING OF '*' AND 'U' CHARACTERS ARE TRANSMITTED TO THE PRINTER ON A FULL 132 COLUMN BASIS. SINCE THESE CHARACTERS ARE COMPLIMENTARY BITWISE, THEY PROVIDE BOTH A ONES AND ZEROES CHECK OF ALL TRANSMISSION LINES. END OF LINE IS SENSED WITHIN THE PROCESSOR AND A LINE FEED CHARACTER IS TRANSMITTED TO PRINT EACH LINE. PRINTING OF THE TEST LINE IS REPEATED 32 TIMES, ALTERNATING THE COLUMN POSITIONS OF THE '*' AND 'U' CHARACTERS TO PRODUCE A CHECKER-BOARD PATTERN.

7.2.2 TEST 3 - CHARACTER GENERATOR AND COMPARATOR TEST

TEST 3 IS DESIGNED PRIMARILY TO TEST THE LINE PRINTER CHARACTER GENERATOR AND COMPARATOR LOGIC AND ITS ABILITY TO DETECT AND ACT UPON BOTH PRINTABLE AND ILLEGAL (NON-PRINTING) CHARACTERS. A SERIES OF ALL 96 PRINTABLE CHARACTERS ARE TRANSMITTED IN SEQUENCE TO THE LINE PRINTER AND PRINTED ON A SINGLE LINE BEGINNING WITH THE SPACE CHARACTER. THIS IS FOLLOWED BY AN ALTERNATE LINE OF ALL 32 ILLEGAL CHARACTERS, EACH OF WHICH SHOULD BE CONVERTED TO A SPACE CHARACTER PRODUCING NO VISIBLE PRINTING. THIS SEQUENCE OF ALTERNATING ALL PRINTABLE CHARACTERS FOLLOWED BY ALL ILLEGAL CHARACTERS IS REPEATED 10 TIMES ALONG WITH AN EXTRA LINE OF ILLEGAL CHARACTERS INSERTED AT THE BEGINNING OF THE TEST TO PRODUCE 21 LINES OF PRINT (11 OF WHICH WILL BE BLANK).

7.2.3 TEST 4 - UNDERLINE TEST

THIS TEST CHECKS THAT ALL 132 PRINT SPACES CAN BE UNDERLINED.

7.2.4 TEST 5 - LINE EDITING TEST

THIS TEST CHECKS THE INTERNAL EDITING FOR CORRECT OPERATION. FULL LINES OF E'S ARE PRINTED BY SENDING A PAIR OF E'S AT A TIME THEN OVERPRINTING THOSE E'S PRINTED WITH SPACES AND ADDING ANOTHER PAIR OF E'S TO THE LINE UNTIL THE LINE IS COMPLETED. THEN A FULL LINE OF M'S ARE PRINTED FOR COMPARISON. A TOTAL OF 16 LINES ARE PRINTED DURING THIS TEST. THERE IS NO SHUTTLE IN THE LXY11/21-LXV11 LINE PRINTER. EACH COLUMN HAS A HAMMER. THE PRINTER LOGIC SELECTS WHICH HAMMER IS TO FIRE.

7.2.5 TEST 6 - PRINT CONTROL TEST

THIS TEST CHECKS THE PRINT CONTROL LOGIC BY SENDING MORE THAN 132 CHARACTERS BEFORE SENDING A PRINT COMMAND. THE PRINTER SHOULD SAVE THE FIRST 132 CHARACTERS RECEIVED AND PRINT THEM CORRECTLY WHEN THE PRINT COMMAND IS RECEIVED. ALL CHARACTERS AFTER THE FIRST 132 SHOULD BE LOST. THE PROGRAM SENDS A FULL LINE OF 132 ZEROS THEN THE FULL CHARACTER SET BEFORE SENDING A LINE FEED TO PRINT THE LINE. THE PRINTED LINE SHOULD CONTAIN ONLY ZEROS. THIS IS REPEATED USING ONES, TWOS, THREES, FOURS, AND FIVES. THEN A LINE OF SPACES ARE SENT AND THE FULL CHARACTER SET BEFORE THE LINE FEED. A BLANK LINE SHOULD BE PRINTED. AFTER THE BLANK LINE, THE NUMBERS 6 TO 9 ARE SENT AS BEFORE. A TOTAL OF 11 LINES WILL BE PRINTED WITH THE MIDDLE LINE BLANK.

7.2.6 TEST 7 - MULTIPLE LINE ADVANCE TEST

THIS TEST CHECKS THE MULTIPLE LINE ADVANCE OF THE LINE PRINTER. A LINE OF NUMBERS IS PRINTED THEN THE PAPER IS ADVANCED THAT NUMBER OF LINES. THUS THE NUMBER PRINTED WILL INDICATE THE NUMBER OF BLANK LINES FOLLOWING THAT LINE. THE NUMBER IS VARIED BETWEEN 2 AND 9, AND A LINE OF ALL ZEROS WILL END THE TEST.

7.2.7 TEST 8 - HIGH SPEED PRINT TEST

THIS TEST PRINTS ALL CHARACTERS BY PRINTING 16 CHARACTER GROUPS ON EACH LINE THEN SEQUENCING TO THE NEXT LINE WITH THE NEXT GROUP. THIS WILL TEST THE HAMMER SUPPLY FOR MAXIMUM CURRENT SURGE AND WILL TEST FOR WORST CASE NOISE SINCE ALL HAMMERS WILL FIRE AT ONCE ON EACH LINE.

7.2.8 TEST 9 - SINGLE CHAR, ALL COLUMNS TEST

THIS TEST IS DESIGNED AS AN ENDURANCE TEST OF THE LINE PRINTER AS WELL AS A CHARACTER CHECK OF THE PROM. 132 COLUMNS OF EACH OF THE 96 CHARACTERS ARE TRANSMITTED TO THE LINE PRINTER AND PRINTED IN ROTATION. A SAMPLE OF THE PRINT OUT FOLLOWS:

```
?????------?????  
@@@@@------@@@@@  
AAAAA------AAAAA  
BBBBB------BBBBB  
-----  
-----  
ZZZZZ------ZZZZZ
```

7.2.9 TEST 10 - PROM PATTERN TEST

THIS TEST IS DESIGNED TO PRODUCE AN IMAGE OF THE ENTIRE PROM PATTERN. THIS IS A WORST CASE NOISE AND ENDURANCE TEST, AND A CHECK OF THE PROM PATTERN.

7.2.10 TEST 11 - SPURIOUS HAMMER FIRING TEST

THIS TEST IS DESIGNED TO DETECT SPURIOUS HAMMER FIRINGS AND DEFECTIVE HAMMER DRIVERS DURING OPERATION OF THE LINE PRINTER. THE PATTERNS WHICH ARE PRODUCED ARE RIGHT AND LEFT HAND WEDGES, EACH COMPOSED OF 132 LINES OF PRINT AS FOLLOWS:

LEFT HAND WEDGE - WILL END EACH LINE WITH A "?" CHARACTER.

RIGHT HAND WEDGE - WILL START EACH LINE WITH A "?" CHARACTER.

ANY PRINT OUTSIDE OF THE WEDGE WILL BE CAUSED BY A HAMMER MISFIRE OR HAMMER BOUNCE.

7.2.11 TEST 12 - HAMMER ALIGNMENT TEST

THIS ROUTINE IS DESIGNED TO BE USED AS A DRIVER FOR MANUAL HAMMER ALIGNMENT AND INTENSITY ADJUSTMENTS ON THE LINE PRINTER. THIS TEST PRINTS A FULL 132 COLUMN LINE OF 'E' CHARACTERS FOR 63 LINES.

7.2.12 TEST 13 - ELONGATED CHARACTERS TEST

THIS TEST INSURE THAT ALL 96 CHARACTERS CAN BE PRINTED IN THE ELONGATED OR EXPANDED MODE.

7.2.13 TEST 14 - UNDERLINED ELONGATED CHARACTERS TEST

THIS TEST CHECKS THAT ELONGATED CHARACTERS CAN BE UNDERLINED.

7.2.14 TEST 15 - ALTERNATE CHARACTER SET TEST

WHEN ENABLED THIS TEST WILL PRINT THE OPTIONAL ALTERNATE CHARACTER SET IN THE SAME MANNER AS TEST 8.

7.2.15 EVFU TEST - EVFU CHANNEL SLEW COMMAND TEST

THIS TEST IS DESIGNED TO TEST THE EVFU CHANNEL SLEW COMMANDS. THE EVFU MEMORY IS FIRST LOADED WITH CHANNEL 1 IN THE FIRST LOCATION THEN THE SEQUENCE 21-35 OCTAL THROUGHOUT THE REMAINING MEMORY. CHANNEL 1 IS USED TO RESET THE POINTER TO THE TOP OF MEMORY THEN EACH CHANNEL, STARTING WITH CHANNEL 2 IS CALLED 10 TIMES. THE SPACING ON THE PAPER SHOULD BE AS SPECIFIED IN THE PRINTED LINE.

7.2.16 PLOT TEST

THIS TEST WILL CHECK THE PLOT CAPABILITIES OF THE LXY11/21-LXV11. IN THIS TEST TWO PLOTS 132 X 132 WILL BE PLOTTED THE FIRST WILL CONSIST OF A FLOATING SPACE STARTING AT THE LEFT AND MOVING ONE SPACE PER LINE. THE SECOND PLOT WILL BE A FLOATING DOT STARTING AT THE LEFT AND MOVING ONE SPACE RIGHT PER LINE. AT THE END OF THIS TEST ALL PRINTING TESTS WILL BE RESTARTED AT TEST 2.

7.3 SCOPE DRIVE ROUTINE

THE PRUPOSE OF THIS TEST SEQUENCE IS TO PROVIDE THE OPERATOR WITH A SHORT BUT COMPREHENSIVE SCOPE DRIVER ROUTINE FOR USE IN TROUBLE SHOOTING THE PRINTER INTERFACE CONTROL MODULE WITH THE SCOPE. DEPENDING ON THE SETTING OF SWITCH 11 THIS TEST WILL EITHER CONTINUALLY SEND WHATEVER CHARACTER IS SET IN THE SWITCH REGISTER TO THE LINE PRINTER, OR ONLY SEND IT ONCE AND HALT. (SEE DESCRIPTION OF SWITCH 11 OPERATION IN SECTION 5.1)

TO INSERT A LINE FEED CHARACTER AFTER EVERY 132 CHARACTERS, WHEN SENDING CHARACTERS CONTINUOUSLY, START AT LOCATION 700(8).

TO LEAVE OUT THE LINE FEED, START AT LOCATION 710(8). THIS ROUTINE SHOULD BE USEFUL WHEN TROUBLE SHOOTING THE EVFU.

WHEN SWITCH 11 IS UP, TO SEND ONLY ONE CHARACTER THEN HALT, DEPRESS CONTINUE TO SEND THE NEXT CHARACTER AFTER SETTING THE SWITCH REGISTER AS DESIRED. TO RESUME SENDING CONTINUOUS CHARACTERS, PLACE SWITCH 11 DOWN, SET THE SWITCHES, AND DEPRESS CONTINUE. TO STOP SENDING CONTINUOUSLY PLACE SWITCH 11 UP.

855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910

.TITLE YM-Z049A-AA (DECSPEC-11-BDFAD-A-D)
:COPYRIGHT (C) 1977 DIGITAL EQUIPMENT CORP., MAYNARD, MASS.
:***** LXY11/21-LXV11 LINE PRINTER/PLOTTER TEST *****

:LIST OF SWITCH SETTINGS USED IN THIS TEST

:SWITCH NO.	DESCRIPTION
:	15 LOOP ON ERROR IN TEST 1 ONLY !!!
:	14 TEST EVFU
:	13 TEST ALTERNATE CHARACTER SET
:	12 LOOP ON TEST
:	11 SEND ONLY ONE CHAR TO LINE PRINTER IN SCOPE TEST - THEN HALT
:	10 TEST PLOT FEATURE
:	0 USED TO TEST PRINT SPEED IN TEST 1 IF NO CLOCK IS AVAILABLE

000000	R0=%0
000001	R1=%1
000002	R2=%2
000003	R3=%3
000004	R4=%4
000005	R5=%5
000006	R6=%6
000007	R7=%7
000006	SP=R6
000007	PC=R7
100000	BIT15 =100000
040000	BIT14 =40000
020000	BIT13 =20000
010000	BIT12 =10000
004000	BIT11 =4000
002000	BIT10 =2000
001000	BIT9 =1000
000400	BIT8 =400
000200	BIT7 =200
000100	BIT6 =100
000040	BIT5 =40
000020	BIT4 =20
000010	BIT3 =10
000004	BIT2 =4
000002	BIT1 =2
000001	BIT0 =1

SL=36 ;START VFU LOAD
EL=37 ;END VFU LOAD
.ENABLE ABS
.ENABLE AMA
.=0
.REPT 100

```
911          .+2
912          HALT
913          .ENDR
914
915
916          000030          . =30
917
918          000030 010744    TYP
919          000032 000340    340
920
921
922          000042          . =42
923
924          000042 000000    0
925
926          000046          . =46
927          000046 010566    LOGICAL
928          000052          . =52
929          000052 040000    BIT14
930
931
932          000060          . =60
933          000060 011172    IKINT
934          000062 000300    300
935
936
937          000100          . =100
938
939          000100 003100    LKSRV
940          000102 000340    340
941
942          000104          CONVRT
943          000106 000340    340
944
945          000174          . =174
946          000174 000000    DISPREG: 0
947          000176 000000    SWREG: 0
948
949          000200          . =200
950
951          000200 012706 001000    MOV #1000,%6
952          000204 000137 001100    JMP SETUP
953
954
955          000300          . =300
956
957          000300 000137 014660    JMP DAV2
958          000304 000137 016222    JMP PLPRT
959
960
961          000400          . =400
962
963
964          000400 000137 002466    JMP SWTIME
965          000404 000137 002626    JMP KW11L
966          000410 000137 002546    JMP KW11P
```

:KEYBOARD INTERRUPT ROUTINE

:LINE CLOCK SERVICE ROUTINE

:1 MINUTE PRINT SPEED CHECK
:START FOR USING SWITCH REG FOR TIMING
:START FOR KW11-L LINE CLOCK
:START FOR KW11-P LINE CLOCK

```
967
968
969
970          000600          .=600
971
972 000600 012706 001000      MOV    #1000,%6      ;START OF PRINTING TESTS SEQUENCE
973 000604 000137 003310      JMP    TEST2         ;TEST 2
974 000610 000137 003552      JMP    TEST3         ;TEST 3
975 000614 000137 004114      JMP    ULPRT         ;TEST 4
976 000620 000137 004420      JMP    OVRPRT        ;TEST 5
977 000624 000137 004716      JMP    PRTCTL        ;TEST 6
978 000630 000137 005176      JMP    MLF           ;TEST 7
979 000634 000137 005412      JMP    HSPRT         ;TEST 8
980 000640 000137 005712      JMP    SNGCHR        ;TEST 9
981 000644 000137 006066      JMP    ROTATE        ;TEST 10
982 000650 000137 006332      JMP    LFTTR         ;TEST 11
983 000654 000137 007046      JMP    HAMALN        ;TEST 12
984 000660 000137 007216      JMP    EXPRT         ;TEST 13
985 000664 000137 007550      JMP    UEXPRT        ;TEST 14
986 000670 000137 010166      JMP    ACPRT         ;TEST 15
987
988          000700          .=700
989
990 000700 012737 016174 016220  MOV    #LSCA,LOSCOP ;SEND LF AFTER 132 CHARS
991 000706 000137 016052      JMP    SCOPE
992
993          000720          .=720
994
995 000720 012737 016052 016220  MOV    #SCOPE,LOSCOP ;NO LF'S SENT IN SCOPE ROUTINE
996 000726 000137 016052      JMP    SCOPE         ;DO SCOPE ROUTINE
997
998
999          001000          .=1000
1000
1001          ;LINE PRINTER HARDWARE REGISTERS
1002
1003 001000 177514      LPS:   177514      ;STATUS REGISTER
1004                                     ;BIT 15=ERROR
1005                                     ;BIT 7=READY
1006                                     ;BIT 6=INTERRUPT ENABLE
1007
1008 001002 177516      LPB:   177516      ;DATA BUFFER REGISTER
1009                                     ;BITS 0-6=7 BIT ASCII CHARACTER BUFFER
1010                                     ;BITS 7-15=NOT USED
1011
1012
1013 001004 177570      SWR:   177570
1014 001006 177570      DISPLAY:177570
1015 001010 177776      PSW:   177776
1016 001012 177566      TPB:   177566
1017 001014 177562      TKB:   177562
1018 001016 177564      TPS:   177564
1019 001020 177560      TKS:   177560
1020 001022 172542      CSBR:  172542
1021 001024 172540      PLKS:  172540
1022 001026 177546      LKS:   177546
```

1023 001030 000200
1024 001032 000202
1025 000240
1026 000000
1027 000002
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078

PTRVEC: .WORD 200
PTRPSW: .WORD 202
NOP =240
N =0
M =2

;MACRO FOR SETTING UP ERROR COUNT

.LIST ME

ERR'X': .MACR \$ERROR X
MOV #X, ERCOUNT ;SET UP ERROR COUNT X
N=N+1
.ENDM \$ERROR

;MACRO FOR PRINTING TEST NUMBER AT START OF TEST

.LIST ME

.MACR \$PRINT Y
MOV TNO'Y',MES15 ;SET TEST NUMBER FOR MESSAGE
JSR %4,PRNNT ;PRINT TEST NUMBER
M=M+1
.ENDM \$PRINT

;MACRO FOR WAITING FOR PRINTER TO PRINT OR SLEW

.LIST ME

.MACR \$WAIT
TSTB @LPS ;TEST READY
BPL -4 ;WAIT FOR READY
.ENDM \$WAIT

;MACRO FOR ENABLING KEYBOARD INTERRUPT IF THERE IS NO
;H/W SWITCH REGISTER AND THERE IS A S/W SWITCH REGISTER

.LIST ME

.MACR \$ENABLE
CMP #176,SWR ;S/W SWR ?
BNE +10 ;NO- CONTINUE
BIS #100,@TKS ;ENABLE KEYBOARD INTERRUPT
.ENDM \$ENABLE

;MACRO USED TO LOAD THE PSW WITH THE
;CORRECT PROCESSOR PRIORITY LEVEL

1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
(1)
(1)
(1)

001034 000000
001036 000000
C01040 000000
001042 000000
001044 000000
001046 000000
001050 000000
001052 000000
001054 000000
001056 000000
001060 000000
001062 000000
001064 000000
001066 000000
001070 000000
001072 000000
001074 000000
001076 000000

004437 010726
001104 000005
001106 013746 000004
001112 013746 000006
001116 012737 001132 000004
001124 005777 177654
001130 000407
001132 012737 000176 001004
001140 012737 000174 001006
001146 022626
001150 012637 000006
001154 012637 000004
001160
001160 022737 000176 001004
001166 001003
001170 052777 000100 177622

.LIST ME

.MACR \$SETPSW
MOV PC,-(SP) ;MOVE PRESENT LOCATION TO STACK
ADD #6,(SP) ;SET UP FOR NEXT INSTRUCTION
RTI ;LOAD PSW
.ENDM \$SETPSW

;MEMORY LOCATIONS USED AS PROGRAM FLAGS AND COUNTERS

SEGCNT: 0
CHRCNT: 0
CHRCNT: 0
CHRCNT: 0
LINCNT: 0
CYCCNT: 0
WORK: 0
SAVE: 0
ERCOUNT: 0
STRCHR: 0
STRCNT: 0
LEGCHR: 0
NUMCHR: 0
OFFSET: 0
DIGITS: 0
SIGNAL: 0
SET: 0
CHAR: 0
OCT: 0

;ROUTINE TO TEST THE MECH. OPERATION OF THE LXY11/21-LXV11

SETUP: JSR %4,TYPINT
RESET ;CLEAR WORLD
MOV 4,-(SP) ;SAVE CURRENT VECTORS
MOV 6,-(SP) ;
MOV #1\$,4 ;SET UP TIMEOUT VECTOR
TST @SWR ;TRY TO ACCESS HARDWARE SWR
BR 2\$;IF THERE, GO TO 2\$

1\$: MOV #SWREG,SWR ;POINT TO SOFTWARE SWR
MOV #DISPREG,DISPLAY ;POINT TO SOFTWARE DISPLAY
CMP (SP)+,(SP)+ ;RESTORE STACK
2\$: MOV (SP)+,6 ;RESTORE TIMEOUT VECTORS
MOV (SP)+,4 ;
\$ENABLE
CMP #176,SWR ;S/W SWR ?
BNE .+10 ;NO- CONTINUE
BIS #100,@TKS ;ENABLE KEYBOARD INTERRUPT

```

1132
1133 001176 104000          EMT      +0
1134 001200 012147          MES1
                                ;TYPE DIAGNOSTIC TITLE
1135 001202 104000          EMT      +0
1136 001204 012212          MES2
                                ;TYPE RESTART ADDRESS INFO
1137 001206 104000          EMT      +0
                                ;TYPE MESSAGE
1138 001210 012237          MES3
                                ;POWER UP
1139 001212 000C00          HALT
                                ;DEPRESS CONTINUE WHEN READY TO START TEST
1140
1141 001214 005777 177560    STP1:   TST      @LPS
                                ;TEST FOR ERROR
1142 001220 100006          BPL      STP2
                                ;NO ERROR TEST FOR READY
1143 001222
(1) 001222 012737 000000 001052  ERR0:   MOV      #0,      ERCOUNT
                                ;SET UP ERROR COUNT 0
(1)
                                N=N+1
1144 001230 004537 011142    JSR      %5,STAER
                                ;REPORT ERROR BIT SET
1145 001234 000767          BR       STP1
                                ;GO TEST FOR ERROR
1146 001236 105777 177536    STP2:   TSTB   @LPS
                                ;TEST FOR READY
1147 001242 100406          BMI      STP3
                                ; READY SET OK
1148 001244
(1) 001244 012737 000001 001052  ERR1:   MOV      #1,      ERCOUNT
                                ;SET UP ERROR COUNT 1
(1)
                                N=N+1
1149 001252 004537 011142    JSR      %5,STAER
                                ;REPORT READY NOT SET
1150 001256 000767          BR       STP2
                                ;GO TEST FOR READY
1151 001260 104000          EMT      +0
                                ;TYPE MESSAGE
1152 001262 012270          MES4
                                ;PRINTER OK 'READY SET' TRY TORN PAPER SWITCH
1153 001264 000000          HALT
                                ;DEPRESS CONTINUE WHEN READY
1154 001266
1155 001266 012777 000014 177506    STP4:   MOV      #14,@LPB
                                ;SEND A 'FF' TO THE PRINTER
1156 001274 012777 000015 177500    MOV      #15,@LPB
                                ;ATTEMPT 'FF' BY SENDING A 'CR'
1157 001302 005777 177472    TST      @LPS
                                ;TEST FOR ERROR
1158 001306 100406          BMI      STP5
                                ;BRANCH IF ERROR SET
1159 001310
(1) 001310 012737 000002 001052  ERR2:   MOV      #2,      ERCOUNT
                                ;SET UP ERROR COUNT 2
(1)
                                N=N+1
1160 001316 004537 011142    JSR      %5,STAER
                                ;REPORT ERROR NOT SET
1161 001322 000761          BR       STP4
                                ;LOOP ON ERROR
1162 001324 104000          EMT      +0
                                ;TYPE MESSAGE
1163 001326 012415          MES6
                                ;ERROR SET OK - TURN ON LINE
1164 001330 000000          HALT
                                ;WAIT FOR OPERATOR
1165
1166 001332 005777 177442    STP5A:  TST      @LPS
                                ;TEST FOR ERROR
1167 001336 100006          BPL      STP5B
                                ;NO ERROR CONTINUE
1168 001340
(1) 001340 012737 000003 001052  ERR3:   MOV      #3,      ERCOUNT
                                ;SET UP ERROR COUNT 3
(1)
                                N=N+1
1169 001346 004537 011142    JSR      %5,STAER
                                ;REPORT ERROR SET
1170 001352 000767          BR       STP5A
                                ;LOOP ON ERROR
1171 001354 105777 177420    STP5B:  TSTB   @LPS
                                ;TEST READY
1172 001360 100406          BMI      STP5C
                                ;READY SET OK
1173 001362
(1) 001362 012737 000004 001052  ERR4:   MOV      #4,      ERCOUNT
                                ;SET UP ERROR COUNT 4
(1)
                                N=N+1
1174 001370 004537 011142    JSR      %5,STAER
                                ;REPORT ERROR NOT SET
1175 001374 000767          BR       STP5B
                                ;LOOP ON ERROR
1176 001376 104000          EMT      +0
                                ;TYPE MESSAGE
1177 001400 012334          MESS
                                ;READY SET OK - TRY FORMS THICKNESS LEVER SWITCH

```



```
1178 001402 000000          HALT          ;DEPRESS CONTINUE WHEN READY
1179
1180 001404 005777 177370    STP6:  TST      @LPS          ;TEST FOR ERROR
1181 001410 100406          BMI      STP7          ;BRANCH IF ERROR SET
1182 001412          $ERROR  \N
(1) 001412 012737 000005 001052 ERR5:  MOV      #5,      ERCOUNT      ;SET UP ERROR COUNT 5
(1)          000006
1183 001420 004537 011142    JSR      %5,STAER      ;REPORT ERROR NOT SET
1184 001424 000767          BR      STP6          ;LOOP ON ERROR
1185 001426 104000          STP7:  EMT      +0      ;TYPE MESSAGE
1186 001430 012415          MES6
1187 001432 000000          HALT          ;ERROR SET OK - TURN ON LINE
;DEPRESS CONTINUE WHEN READY
```

```

1189      ;TEST 1
1190      ;PERFORMS PRELIMINARY COMMAND AND REGISTER TESTING.
1191
1192      ;IS THE PRINTER FREE OF ERRORS
1193
1194      001434 000005      TEST1: RESET      ;CLEAR THE WORLD
1195      001436 005777 177336      TST      @LPS      ;IS ERROR FLAG CLEAR
1196      001442 100006      BPL      TEST1A    ;ERROR IS CLEAR OK
1197      001444      $ERROR  \N
(1) 001444 012737 000006 001052 ERR6: MOV      #6,      ERCOUNT      ;SET UP ERROR COUNT 6
(1)      000007
1198      001452 004537 011142      JSR      %5,STAER  ;REPORT ERROR SET
1199      001456 000766      BR      TEST1      ;LOOP ON ERROR
1200
1201      ;IS READY SET (NO ERRORS EXIST)
1202
1203      001460 000005      TEST1A: RESET     ;CLEAR THE WORLD
1204      001462 105777 177312      TSTB     @LPS     ;IS READY SET
1205      001466 100406      BMI     TEST1B    ;READY SET! PRINTER OK
1206      001470      $ERROR  \N
(1) 001470 012737 000007 001052 ERR7: MOV      #7,      ERCOUNT      ;SET UP ERROR COUNT 7
(1)      000010
1207      001476 004537 011142      JSR      %5,STAER  ;REPORT READY NOT SET
1208      001502 000766      BR      TEST1A    ;LOOP ON ERROR
1209
1210      ;DOES LOADING THE BUFFER RESET READY
1211
1212      001504 005037 001046      TEST1B: CLR      WORK      ;CLEAR COUNTER
1213      001510 012777 000015 177264      MOV      #15,@LPB   ;LOAD CARRIAGE RETURN INTO BUFFER
1214      001516 105777 177256      TSTB     @LPS     ;IS READY CLEAR
1215      001522 100006      BPL     LP1       ;READY IO CLEAR OK!
1216      001524      $ERROR  \N
(1) 001524 012737 000010 001052 ERR10: MOV     #10,     ERCOUNT     ;SET UP ERROR COUNT 10
(1)      000011
1217      001532 004537 011142      JSR      %5,STAER  ;REPORT READY STILL SET
1218      001536 000762      BR      TEST1B    ;LOOP ON ERROR
1219      001540 005777 177234      LP1:  TST      @LPS     ;IS THERE AN ERROR
1220      001544 100006      BPL     LP2       ;NO ERROR CONTINUE
1221      001546      $ERROR  \N
(1) 001546 012737 000011 001052 ERR11: MOV     #11,     ERCOUNT     ;SET UP ERROR COUNT 11
(1)      000012
1222      001554 004537 011142      JSR      %5,STAER  ;REPORT ERROR OCCURRED
1223      001560 000751      BR      TEST1B    ;LOOP ON ERROR
1224      001562 105777 177212      LP2:  TSTB     @LPS     ;IS THE PRINTER STILL BUSY
1225      001566 100411      BMI     TEST1C    ;NO! GO TO NEXT TEST
1226      001570 005237 001046      INC      WORK      ;YES! GO CHECK FLAGS
1227      001574 001361      BNE     LP1       ;PRINTER STILL BUSY WAIT
1228      001576      $ERROR  \N
(1) 001576 012737 000012 001052 ERR12: MOV     #12,     ERCOUNT     ;SET UP ERROR COUNT 12
(1)      000013
1229      001604 004537 011142      JSR      %5,STAER  ;ERROR REPORT TIME OUT
1230      001610 000735      BR      TEST1B    ;LOOP ON ERROR
1231
1232      ;CHECK INTERRUPT LEVEL OF PRINTER
1233      ;THE PRINTER SHOULD BE AT LEVEL 4
1234

```

```

1235 ;TEST THAT THE PRINTER WILL NOT INTERRUPT AT LEVEL 7
1236
1237 001612 012777 002076 177210 TEST1C: MOV #INT1C,@PTRVEC ;SET UP INT VECTOR
1238 001620 012777 000340 177204 MOV #340,@PTRPSW ;SET PRIORITY
1239 001626 005777 177146 TST @LPS ;TEST FOR ERROR
1240 001632 100006 BPL LP3 ;NO ERROR CONTINUE
1241 001634 $ERROR \N
(1) 001634 012737 000013 001052 ERR13: MOV #13, ERCOUNT ;SET UP ERROR COUNT 13
(1) 000014 N=N+1
1242 001642 004537 011142 JSR %5,STAER ;REPORT ERROR SET
1243 001646 000761 BR TEST1C ;LOOP ON ERROR
1244 001650 105777 177124 LP3: TSTB @LPS ;TST FOR READY
1245 001654 100406 BMI LP3X ;READY SET OK
1246 001656 $ERROR \N
(1) 001656 012737 000014 001052 ERR14: MOV #14, ERCOUNT ;SET UP ERROR COUNT 14
(1) 000015 N=N+1
1247 001664 004537 011142 JSR %5,STAER ;REPORT READY NOT SET
1248 001670 000750 BR TEST1C ;LOOP ON ERROR
1249 001672 $ERROR \N
(1) 001672 012737 000015 001052 LP3X: ERR15: MOV #15, ERCOUNT ;SET UP ERROR COUNT 15
(1) 000016 N=N+1
1250 001700 012746 000340 MOV #340,-(SP) ;LOCKUP PROCESSOR, NEW PRIORITY
1251 001704 $SETPSW
(1) 001704 010746 MOV PC,-(SP) ;MOVE PRESENT LOCATION TO STACK
(1) 001706 062716 000006 ADD #6,(SP) ;SET UP FOR NEXT INSTRUCTION
(1) 001712 000002 RTI ;LOAD PSW
1252 001714 052777 000100 177056 BIS #100,@LPS ;SET PRINTER INTO ENABLE
1253 001722 000240 NOP ;WAIT
1254 001724 042777 000100 177046 BIC #100,@LPS ;CLEAR PRINTER INT. ENABLE
1255
1256 ;TEST THAT THE PRINTER WILL NOT INTERRUPT AT LEVEL 6
1257
1258 001732 $ERROR \N
(1) 001732 012737 000016 001052 ERR16: MOV #16, ERCOUNT ;SET UP ERROR COUNT 16
(1) 000017 N=N+1
1259 001740 012746 000300 MOV #300,-(SP) ;SET PROCESSOR PRIORITY LEVEL 6
1260 001744 $SETPSW
(1) 001744 010746 MOV PC,-(SP) ;MOVE PRESENT LOCATION TO STACK
(1) 001746 062716 000006 ADD #6,(SP) ;SET UP FOR NEXT INSTRUCTION
(1) 001752 000002 RTI ;LOAD PSW
1261 001754 052777 000100 177016 BIS #100,@LPS ;SET PRINTER INT ENABLE
1262 001762 000240 NOP ;WAIT
1263 001764 042777 000100 177006 BIC #100,@LPS ;CLEAR PRINTER INT. ENABLE
1264
1265 ;TEST THAT THE PRINTER WILL NOT INT. AT
1266 ;PROCESSOR LEVEL 5
1267
1268 001772 $ERROR \N
(1) 001772 012737 000017 001052 ERR17: MOV #17, ERCOUNT ;SET UP ERROR COUNT 17
(1) 000020 N=N+1
1269 002000 012746 000240 MOV #240,-(SP) ;SET UP PROCESSOR TO LEVEL 5
1270 002004 $SETPSW
(1) 002004 010746 MOV PC,-(SP) ;MOVE PRESENT LOCATION TO STACK
(1) 002006 062716 000006 ADD #6,(SP) ;SET UP FOR NEXT INSTRUCTION
(1) 002012 000002 RTI ;LOAD PSW
1271 002014 052777 000100 176756 BIS #100,@LPS ;SET PRINTER INT ENABLE

```

```

1272 002022 000240          NOP          :WAIT
1273 002024 042777 000100 176746      BIC          #100,@LPS      :CLEAR INT ENABLE PRINTER OK
1274
1275          :TEST THAT THE PRINTER WILL NOT INTERRUPT
1276          :WHEN THE PROCESSOR IS AT LEVEL 4
1277
1278 002032          $ERROR \N
(1) 002032 012737 000020 001052 ERR20: MOV          #20,   ERCOUNT      ;SET UP ERROR COUNT 20
(1)          000021          N=N+1
1279 002040 012746 000200          MOV          #200,-(SP)      ;SET PROCESSOR TO LEVEL 4
1280 002044          $SETPSW
(1) 002044 010746          MOV          PC,-(SP)      ;MOVE PRESENT LOCATION TO STACK
(1) 002046 062716 000006          ADD          #6,(SP)      ;SET UP FOR NEXT INSTRUCTION
(1) 002052 000002          RTI          ;LOAD PSW
1281 002054 052777 000100 176716      BIS          #100,@LPS      ;SET PRINTER INT. ENABLE
1282 002062 000240          NOP          :WAIT
1283 002064 042777 000100 176706      BIC          #100,@LPS      ;CLEAR PRINTER INT ENABLE
1284 002072 000137 002110          JMP          TEST1D        ;PRINTER OK CONTINUE
1285
1286          :INTERRUPT HANDLE FOR TEST1C
1287          :RESTORE STACK AND REPORT ERROR
1288
1289 002076 022626          INT1C: CMP          (6)+,(6)+      ;RESTORE STACK
1290 002100 004537 011142          JSR          %5,STAER      ;REPORT ERROR
1291 002104 000137 001612          JMP          TEST1C        ;RE-ENTER TEST1C
1292
1293          :TEST THE ABILITY OF THE PRINTER TO INTERRUPT
1294          :AT PRIORITY LEVEL 4
1295
1296 002110 012777 002230 176712 TEST1D: MOV          #INT1D,@PTRVEC  ;SET UP INTERRUPT VECTOR
1297 002116 012777 000340 176706      MOV          #340,@PTRPSW  ;LOCK UP PRIORITIES
1298 002124 005777 176650          TST          @LPS         ;IS THERE A PRINTER ERROR
1299 002130 100006          BPL          LP4          ;NO! CONTINUE
1300 002132          $ERROR \N
(1) 002132 012737 000021 001052 ERR21: MOV          #21,   ERCOUNT      ;SET UP ERROR COUNT 21
(1)          000022          N=N+1
1301 002140 004537 011142          JSR          %5,STAER      ;REPORT PRINTER ERROR
1302 002144 000761          BR          TEST1D        ;LOOP ON ERROR
1303 002146 105777 176626          LP4:  TSTB          @LPS         ;IS READY SET
1304 002152 100406          BMI          LP5          ;YES - PRINTER READY
1305 002154          $ERROR \N
(1) 002154 012737 000022 001052 ERR22: MOV          #22,   ERCOUNT      ;SET UP ERROR COUNT 22
(1)          000023          N=N+1
1306 002162 004537 011142          JSR          %5,STAER      ;REPORT READY NOT SET
1307 002166 000750          BR          TEST1D        ;LOOP ON ERROR
1308 002170 012746 000140          LP5:  MOV          #140,-(SP)  ;SET PRIORITY TO LEVEL 3
1309 002174          $SETPSW
(1) 002174 010746          MOV          PC,-(SP)      ;MOVE PRESENT LOCATION TO STACK
(1) 002176 062716 000006          ADD          #6,(SP)      ;SET UP FOR NEXT INSTRUCTION
(1) 002202 000002          RTI          ;LOAD PSW
1310 002204 052777 000100 176566      BIS          #100,@LPS      ;SET PRINTER INTERRUPT ENABLE
1311 002212 000240          NOP          :WAIT
1312 002214          $ERROR \N
(1) 002214 012737 000023 001052 ERR23: MOV          #23,   ERCOUNT      ;SET UP ERROR COUNT 23
(1)          000024          N=N+1
1313 002222 004537 011142          JSR          %5,STAER      ;REPORT ERROR

```

```

1314 002226 000730          BR      TEST1D          ;LOOP ON ERROR
1315
1316          ;INTERRUPT HANDLER FOR TEST1D
1317
1318 002230 022626          INT1D:  CMP      (6)+,(6)+          ;RESET STACK
1319 002232 042777 000100 176540  BIC      #100,@LPS          ;CLEAR INT. ENABLE FOR PRINTER
1320 002240 012746 000000          MOV      #0,-(SP)          ;CLEAR PROCESSOR STATUS
1321 002244          $SETPSW
(1) 002244 010746          MOV      PC,-(SP)          ;MOVE PRESENT LOCATION TO STACK
(1) 002246 062716 000006          ADD      #6,(SP)          ;SET UP FOR NEXT INSTRUCTION
(1) 002252 000002          RTI
1322 002254 012777 012706 176546  MOV      #12706,@PTRVEC    ;RESET INSTRUCTION AT 200
1323 002262 012777 001000 176542  MOV      #1000,@PTRPSW    ;RESET INSTRUCTION AT 202
1324
1325
1326          ; GET INITIAL SWR VALUE
1327          ; IF THERE IS NO H/W SWR
1328
1329
1330 002270 022737 000176 001004  CMP      #176,SWR          ;S/W SWR ?
1331 002276 001044          BNE      SKIP              ;NO- CONTINUE
1332 002300 005037 001070          CLR      SIGNAL            ;INITIALIZE INTERRUPT ROUTINE
1333 002304 005037 001066          CLR      DIGITS            ;
1334 002310 005037 001072          CLR      SET                ;
1335 002314 005037 001074          CLR      CHAR                ;
1336 002320 013746 000034          MOV      34,-(SP)          ;SAVE VECTOR
1337 002324 013746 000036          MOV      36,-(SP)          ;SAVE VECTOR
1338 002330 012737 011172 000034  MOV      #TKINT,34          ;SET UP NEW VECTOR
1339 002336 012737 000300 000036  MOV      #300,36            ;SET UP NEW VECTOR
1340 002344 005237 001072          INC      SET                ;SET HEADER FLAG
1341 002350 104400          TRAP    +0                  ;ENTER INTERRUPT ROUTINE
1342 002352 005037 001072          CLR      SET                ;CLEAR HEADER FLAG
1343 002356 012637 000036          MOV      (SP)+,36          ;RESTORE VECTOR
1344 002362 012637 000034          MOV      (SP)+,34          ;RESTORE VECTOR
1345 002366 012777 000100 176424  MOV      #100,@TKS          ;ENABLE KEYBOARD INTERRUPT
1346 002374 000001          WT:   WAIT
1347 002376 000240          NOP
1348 002400 022737 000001 001070  CMP      #1,SIGNAL          ;SWR VALUE ENTERED ?
1349 002406 001772          BEQ      WT                ;NO- WAIT
1350 002410 000240          SKIP:  NOP
1351
1352
1353          ;1 MINUTE PRINT SPEED CHECK
1354          ;IF A KW11-L OR KW11-P ARE NOT AVAILABLE, THE SR BIT0 IS USED
1355          ;FOR MANUAL TIMING OF THE PRINTER.
1356
1357
1358 002412 012737 000002 000006  CLCKAV: MOV      #RTI,@#6          ;SET TRAP TO RETURN
1359 002420 012737 000006 000004  MOV      #6,@#4
1360 002426 000261          SEC
1361 002430 105777 176372          TSTB    @LKS                ;KW11-L AVAILABLE?
1362 002434 103404          BCS     1$                  ;NO, BRANCH
1363 002436 005037 000004          CLR      @#4                ;RESET TRAP VECTOR TO HALT
1364 002442 000137 002626          JMP     KW11L                ;USE KW11L FOR TIMING
1365 002446 000261          1$:   SEC
1366 002450 105777 176350          TSTB    @PLKS                ;KW11-P AVAILABLE?

```

```

1367 002454 103404          BCS      SWTIME      ;NO, USE SWITCH REG FOR TIMING
1368 002456 005037 000004    CLR      @#4        ;RESET TRAP VECTOR TO HALT
1369 002462 000137 002546    JMP      KW11P      ;USE KW11-P FOR TIMING
1370 002466          SWTIME: $ENABLE
(1) 002466 022737 000176 001004  CMP      #176,SWR      ;S/W SWR ?
(1) 002474 001003          BNE      .+10         ;NO- CONTINUE
(1) 002476 052777 000100 176314  BIS      #100,@TKS     ;ENABLE KEYBOARD INTERRUPT
1371 002504 005037 001042    CLR      LINCNT      ;CLEAR LINE COUNT
1372 002510 004437 010726    JSR      %4,TYPINT
1373 002514 005037 000004    CLR      @#4        ;RESET TRAP VECTOR TO HALT
1374 002520 104000          EMT      +0         ;TYPE MESSAGE
1375 002522 011731          MESC
1376 002524 012737 000002 003076  MOV      #2,DIA       ;PRINT SPEED CHECK USING MANUAL TIMING
1377 002532 032777 000001 176244 1$: BIT      #BIT0,@SWR  ;SET DUMMY ADDRESS
1378 002540 001774          BEQ      1$         ;START?
1379 002542 000137 002702    JMP      STAR0      ;WAIT FOR START
                                ;START PRINTING
1380
1381
1382          ;START FOR KW11-P.....
1383
1384 002546          KW11P: $ENABLE
(1) 002546 022737 000176 001004  CMP      #176,SWR      ;S/W SWR ?
(1) 002554 001003          BNE      .+10         ;NO- CONTINUE
(1) 002556 052777 000100 176234  BIS      #100,@TKS     ;ENABLE KEYBOARD INTERRUPT
1385 002564 005037 001042    CLR      LINCNT      ;CLEAR LINE COUNT
1386 002570 004437 010726    JSR      %4,TYPINT
1387 002574 012706 001000    MOV      #1000,%6    ;RESET STACK
1388 002600 013777 003072 176214  MOV      MINCNT,@CSBR ;SET CLOCK COUNT
1389 002606 013737 001024 003076  MOV      PLKS,DIA    ;STORE PLKS ADDRESS
1390 002614 012777 000105 176202  MOV      #105,@PLKS  ;START CLOCK
1391 002622 000137 002702    JMP      STAR0      ;START PRINTING
1392
1393          ;START FOR KW11-L.....
1394
1395 002626          KW11L: $ENABLE
(1) 002626 022737 000176 001004  CMP      #176,SWR      ;S/W SWR ?
(1) 002634 001003          BNE      .+10         ;NO- CONTINUE
(1) 002636 052777 000100 176154  BIS      #100,@TKS     ;ENABLE KEYBOARD INTERRUPT
1396 002644 005037 001042    CLR      LINCNT      ;CLEAR LINE COUNT
1397 002650 004437 010726    JSR      %4,TYPINT
1398 002654 012706 001000    MOV      #1000,%6    ;RESET STACK
1399 002660 013737 003072 003074  MOV      MINCNT,CNTR ;SET CLOCK COUNT
1400 002666 013737 001026 003076  MOV      LKS,DIA    ;STORE LKS ADDRESS
1401 002674 012777 000100 176124  MOV      #100,@LKS   ;ENABLE CLOCK INTERRUPT
1402
1403          ;PRINTING ROUTINE.....
1404
1405 002702          STAR0:
1406 002702 012737 000200 001060  STARCA: MOV      #200,LEGCHR ;LEGAL CHECK
1407 002710 012737 000140 001062  MOV      #140,NUMCHR  ;#CHARS
1408 002716 013737 001060 001054  STAROC: MOV     LEGCHR,STRCHR ;SET FIRST CHAR IF LP14
1409 002724 012737 000204 001036  STAROB: MOV     #132,,CHRCNT ;SET CHAR COUNT
1410 002732 012737 003270 001054  MOV      #PATTB,STRCHR ;INITIALIZE TABLE POINTER
1411 002740 012737 000021 001044  STARA:  MOV      #17,,CYCCNT ;SET GROUP COUNT
1412 002746 017737 176102 001040  MOV      @STRCHR,CHRGEN ;GET CHAR FROM TABLE
1413 002754 063737 001042 001040  ADD      LINCNT,CHRGEN ;ADD LINE COUNT

```

```

1414 002762 023737 001060 001040 1$:    CMP    LEGCHR,CHRGEN    ;LEGAL CHAR?
1415 002770 003004          BGT    STAR1          ;YES,  BRANCH
1416 002772 163737 001062 001040          SUB    NUMCHR,CHRGEN  ;NO,  MAKE LEGAL
1417 003000 000770          BR     1$             ;RECHECK CHAR
1418 003002 013777 001040 175772 STAR1:  MOV    CHRGEN,@LPB    ;LOAD BUFFER
1419 003010 005337 001036          DEC    CHRCNT        ;DECREMENT CHAR COUNT
1420 003014 001410          BEQ    STARED       ;BRANCH IF DONE LINE
1421 003016 005337 001044          DEC    CYCCNT       ;DECREMENT CYCCLE COUNT
1422 003022 001367          BNE    STAR1        ;CONTINUE IF NOT DONE GROUP
1423 003024 062737 000002 001054          ADD    #2,STRCHR    ;ADD 2 TO TABLE POINTER
1424 003032 000137 002740          JMP    STARA        ;CONTINUE
1425 003036 005237 001042          STARED: INC    LINCNT ;INCREMENT LINE COUNT
1426 003042 012777 000012 175732          MOV    #12,@LPB    ;SEND LF
1427 003050          $WAIT
   (1) 003050 105777 175724          TSTB  @LPS          ;TEST READY
   (1) 003054 100375          BPL  -.4            ;WAIT FOR READY
1428 003056 032777 000001 175720          BIT   #BIT0,@SWR   ;STOP PRINT?
1429 003064 001411          BEQ   CONVRT        ;YES,  BRANCH
1430 003066 000137 002724          JMP   STAROB        ;CONTINUE

```

1431
1432
1433

```

1434 003072 007020
1435 003074 000000
1436 003076 000002

```

```

MINCNT: 7020
CNTR: 0
DIA: 2

```

1437
1438

;NOTE -- PLACE 5670 (8) IN MINCNT FOR 50 HZ. LINE FREQUENCY !!!

1439
1440

;LINE CLOCK SERVICE ROUTINE FOR KW11-L

1441
1442

```

1443 003100 005337 003074
1444 003104 001401
1445 003106 000002

```

```

LKSRV: DEC    CNTR    ;DECREMENT COUNTER
        BEQ    CONVRT ;EXIT IF 1 MINUTE
        RTI   ;RETURN

```

1446
1447

;ROUTINE TO PRINT NUMBER OF LINES PRINTED IN 1 MINUTE

1448
1449

```

1450 003110 042777 000100 177760 CONVRT: BIC    #100,@DIA ;DISABLE CLOCK INTERRUPT IF CLOCK AVAILABLE
1451 003116 005037 011056          CLR    TYPDAT      ;CLEAR DIGIT COUNT
1452 003122 012703 012775          MOV    #MES12,%3  ;SET MESSAGE POINTER
1453 003126 022737 000144 001042 1$:    CMP    #100.,LINCNT ;GREATER THAN 100?
1454 003134 003006          BGT    2$         ;NO,  PRINT HUNDRED'S DIGIT
1455 003136 162737 000144 001042          SUB    #100.,LINCNT ;YES,  SUBTRACT 100
1456 003144 005237 011056          INC    TYPDAT     ;INCREMENT HUNDRED'S DIGIT
1457 003150 000766          BR     1$         ;CONTINUE CONVERSION
1458 003152 062737 000060 011056 2$:    ADD    #60,TYPDAT  ;MAKE ASCII
1459 003160 113723 011056          MOVB  TYPDAT,(%3)+ ;STORE DIGIT
1460 003164 005037 011056          CLR    TYPDAT     ;CLEAR DIGIT COUNTER
1461 003170 022737 000012 001042 3$:    CMP    #10.,LINCNT ;GREATER THEN 10?
1462 003176 003006          BGT    4$         ;NO,  PRINT DIGIT
1463 003200 162737 000012 001042          SUB    #10.,LINCNT ;YES,  SUBTRACT 10
1464 003206 005237 011056          INC    TYPDAT     ;INCREMENT TEN'S DIGIT
1465 003212 000766          BR     3$         ;CONTINUE CONVERSION
1466 003214 062737 000060 011056 4$:    ADD    #60,TYPDAT  ;MAKE ASCII
1467 003222 113723 011056          MOVB  TYPDAT,(%3)+ ;STORE DIGIT

```

YM-Z049A-AA
CZLXAA.SRC

(DECSPEC-11-BDFAD-A-D)
12-MAR-81 10:43

MACY11 30A(1052) 12-MAR-81 11:24 PAGE 18-6

F 3

SEQ 0031

1468	003226	013737	001042	011056	MOV	LINCNT,TYPDAT	:GET ONE'S DIGIT:
1469	003234	062737	000060	011056	ADD	#60,TYPDAT	:MAKE ASCII
1470	003242	113723	011056		MOVB	TYPDAT,(%3)+	:STORE DIGIT
1471	003246	104000			EMT	+0	:TYPE MESSAGE
1472	003250	012736			MES11		:TYPE PRINT SPEED
1473	003252	012737	012734	010724	MOV	#MES11A,PRTMSG	:SET PRINTER MESSAGE ADDRESS
1474	003260	004437	010706		JSR	%4,RINT	:PRINT PRINTER SPEED ON LINE PRINTER
1475	003264	000137	003310		JMP	TEST2	:NEXT TEST
1476							
1477							
1478							
1479	003270	000040					
1480	003272	000117					
1481	003274	000076					
1482	003276	000055					
1483	003300	000134					
1484	003302	000113					
1485	003304	000072					
1486	003306	000051					
1487							
1488							

PATTB: 40
117
76
55
134
113
72
51


```

1490      :TEST 2
1491      :TESTS INTERFACE AND PRINTER DATA PATHS
1492      :WITH ALTERNATING ONES AND ZEROS
1493
1494      003310      TEST2: $ENABLE
(1) 003310 022737 000176 001004      CMP      #176,SWR      ;S/W SWR ?
(1) 003316 001003      BNE      .+10      ;NO- CONTINUE
(1) 003320 052777 000100 175472      BIS      #100,@TKS      ;ENABLE KEYBOARD INTERRUPT
1495 003326 004437 010726      JSR      %4,TYPINT
1496 003332 004537 010602      JSR      %5,PRINT      ;INITIALIZE PRINTER
1497 003336 000406      BR      TST2AX      ;BRANCH IF OK
1498 003340
(1) 003340 012737 000024 001052      ERR24: MOV      $ERROR \N      ;SET UP ERROR COUNT 24:
(1) 003340 000025      N=N+1
1499 003346 004537 011142      JSR      %5,STAER      ;REPORT PRINTER NOT READY
1500 003352 000000      HALT      ;HALT ON ERROR
1501 003354
(1) 003354 013737 014062 013250      TST2AX: $PRINT \M      ;SET TEST NUMBER FOR MESSAGE
(1) 003362 004437 010656      MOV      TNO2,MES15      ;PRINT TEST NUMBER
(1) 003362 000003      JSR      %4,PRNNT
1502 003366 012737 177740 001044      MOV      #-32.,CYCCNT      ;SET UP LINE COUNT FOR 32 LINES
1503 003374 012737 177574 001036      MOV      #-132.,CHRCNT      ;SET CHAR COUNT TO 132
1504 003402 013737 003456 001054      MOV      SCHRSW,STRCHR      ;SET CHAR. SWITCH TO U
1505 003410 005777 175364      T3A: TST      @LPS      ;TEST FOR ERROR
1506 003414 100006      BPL      -LP2B      ;NO ERROR CONTINUE
1507 003416
(1) 003416 012737 000025 001052      ERR25: MOV      $ERROR \N      ;SET UP ERROR COUNT 25
(1) 003416 000026      N=N+1
1508 003424 004537 011142      JSR      %5,STAER      ;REPORT ERROR SET
1509 003430 000000      HALT      ;HALT ON ERROR
1510 003432 000177 175416      LP2B: JMP      @STRCHR      ;LOAD CHAR
1511 003436 013737 003460 001054      T2A: MOV      RCHRSW,STRCHR      ;RESET CHAR. SWITCH
1512 003444 012737 000125 001050      MOV      #125,SAVE      ;STORE CHAR
1513 003452 000137 003476      JMP      T5A      ;LOAD CHAR
1514
1515 003456 003436      SCHRSW: T2A
1516 003460 003462      RCHRSW: T1A
1517
1518 003462 013737 003456 001054      T1A: MOV      SCHRSW,STRCHR      ;SET CHAR. SWITCH TO U
1519 003470 012737 000052 001050      MOV      #52,SAVE      ;STORE CHAR
1520 003476 013777 001050 175276      T5A: MOV      SAVE,@LPB      ;LOAD BUFFER
1521 003504 005237 001036      INC      CHRCNT      ;INC CHARACTER COUNT
1522 003510 001337      BNE      T3A      ;CONTINUE
1523 003512 012777 000012 175262      MOV      #12,@LPB      ;SEND LF
1524 003520
(1) 003520 105777 175254      $WAIT      ;TEST READY
(1) 003524 100375      TSTB      @LPS      ;WAIT FOR READY
1525 003526 012737 177574 001036      BPL      .-4      ;RESET CHAR COUNT
1526 003534 005237 001044      MOV      #-132.,CHRCNT      ;INC CYCLE COUNT
1527 003540 001356      INC      CYCCNT
1528 003542 032777 010000 175234      BNE      T5A      ;CONTINUE IF NOT DONE
1529 003550 001257      BIT      #BIT12,@SWR      ;LOOP ON TEST?
      BNE      TEST2      ;LOOP

```

```

1531      ;TEST 3
1532      ;TEST CHARACTER COMPARATOR WITH ALTERNATE LINES OF
1533      ;ALL CHARACTERS AND ILLEGAL CHARACTERS
1534
1535      TEST3: $ENABLE
(1) 003552 022737 000176 001004      CMP      #176,SWR      ;S/W SWR ?
(1) 003560 001003      BNE      .+10      ;NO- CONTINUE
(1) 003562 052777 000100 175230      BIS      #100,@TKS      ;ENABLE KEYBOARD INTERRUPT
1536 003570 004437 010726      JSR      %4,TYPINT
1537 003574      $PRINT      \M
(1) 003574 013737 014064 013250      MOV      TNO3,MES15      ;SET TEST NUMBER FOR MESSAGE
(1) 003602 004437 010656      JSR      %4,PRNNT      ;PRINT TEST NUMBER
(1) 000004      M=M+1
1538 003606 012737 177765 001044      MOV      #-13,CYCCNT      ;SET 21 LINES
1539 003614 000137 003746      JMP      LP2H      ;SEND ILLEGAL CHARS FIRST TO GIVE BLANK LINE
1540 003620 012737 177574 001036      T2B0: MOV      #-132.,CHRCNT      ;SET CHAR COUNT FOR 132
1541 003626 012737 000040 001040      T2B0A: MOV      #40,CHRGEN      ;SET FIRST CHAR.
1542 003634 005777 175140      T2B1: TST      @LPS      ;DOES THE PRINTER HAVE AN ERROR
1543 003640 100006      BPL      LP2E      ;BRANCH IF NO ERROR
1544 003642      $ERROR      \N
(1) 003642 012737 000026 001052      ERR26: MOV      #26, ERRCOUNT      ;SET UP ERROR COUNT 26
(1) 000027      N=N+1
1545 003650 004537 011142      JSR      %5,STAER      ;REPORT ERROR
1546 003654 000000      HALT      ;HALT ON ERROR
1547 003656 013777 001040 175116      LP2E: MOV      CHRGEN,@LPB      ;PRINT CHARACTER
1548 003664 005237 001036      INC      CHRCNT      ;INC. CHAR. COUNT
1549 003670 001420      BEQ      T2B2      ;BRANCH IF LINE IS FINISHED
1550 003672 005237 001040      INC      CHRGEN      ;NEXT CHAR
1551 003676 032777 020000 175100      BIT      #BIT13,@SWR      ;CHECK CHAR SET
1552 003704 001405      BEQ      T2B2B      ;BRANCH IF 64 CHARS
1553 003706 022737 000200 001040      CMP      #200,CHRGEN      ;LEGAL CHAR?
1554 003714 001744      BEQ      T2B0A      ;MAKE SPACE IF ILLEGAL
1555 003716 000746      BR      T2B1      ;CONTINUE IF LEGAL CHAR
1556 003720 022737 000140 001040      T2B2B: CMP      #140,CHRGEN      ;LEGAL CHAR?
1557 003726 001737      BEQ      T2B0A      ;MAKE SPACE IF ILLEGAL
1558 003730 000741      BR      T2B1      ;CONTINUE IF LEGAL CHAR
1559 003732 012777 000012 175042      T2B2: MOV      #12,@LPB      ;ISSUE LINE FEED
1560 003740      $WAIT
(1) 003740 105777 175034      TSTB      @LPS      ;TEST READY
(1) 003744 100375      BPL      .-4      ;WAIT FOR READY
1561 003746 005037 001040      LP2H: CLR      CHRGEN      ;FIRST ILLEGAL CHAR
1562 003752 005777 175022      T2B3: TST      @LPS      ;TEST FOR ERROR
1563 003756 100006      BPL      LDCH      ;BRANCH IF NO ERROR
1564 003760      $ERROR      \N
(1) 003760 012737 000027 001052      ERR27: MOV      #27, ERRCOUNT      ;SET UP ERROR COUNT 27
(1) 000030      N=N+1
1565 003766 004537 011142      JSR      %5,STAER      ;REPORT ERROR SET
1566 003772 000000      HALT      ;HALT ON ERROR
1567 003774 013777 001040 175000      LDCH: MOV      CHRGEN,@LPB      ;TRANSMIT CHARACTER
1568 004002 005237 001040      T2B4: INC      CHRGEN      ;NEXT CHAR
1569 004006 022737 000005 001040      CMP      #5,CHRGEN      ;TEST FOR PLOT MODE
1570 004014 001772      BEQ      T2B4      ;SKIP IF TRUE
1571 004016 022737 000006 001040      CMP      #6,CHRGEN      ;TEST FOR 8 LPI
1572 004024 001766      BEQ      T2B4      ;SKIP IF TRUE
1573 004026 022737 000010 001040      CMP      #10,CHRGEN      ;TEST FOR ELONGATED CHAR
1574 004034 001762      BEQ      T2B4      ;SKIP IF TRUE

```

YM-2049A-AA
CZLXAA.SRC

(DECSPEC-11-BDFAD-A-D)
12-MAR-81 10:43

MACY11 30A(1052) 12-MAR-81 11:24 PAGE 20-1

SEQ 0034

```
1575 004036 022737 000012 001040      CMP      #12,CHRGEN      ;TEST FOR LINE FEED
1576 004044 001756                      BEQ      T2B4      ;SKIP IF LF
1577 004046 022737 000014 001040      CMP      #14,CHRGEN ;TEST FOR LEGAL CHARACTER
1578 004054 002747                      BLT      LDCH      ;CONTINUE IF STILL ILLEGAL CHAR.
1579 004056 012777 000012 174716 T2B5:  MOV      #12,@LPB  ;ISSUE LINE FEED
1580 004064                      $WAIT
  (1) 004064 105777 174710      TSTB     @LPS      ;TEST READY
  (1) 004070 100375                      BPL      -4        ;WAIT FOR READY
1581 004072 005237 001044      INC      CYCNT     ;INCREMENT LINE COUNT
1582 004076 001250                      BNE      T230     ;CONTINUE IF NOT DONE
1583 004100 032777 010000 174676      BIT      #BIT12,@SWR ;CHECK TO LOOP ON TEST
1584 004106 001402                      BEQ      1$       ;SKIP
1585 004110 000137 003552                      JMP      TEST3    ;LOOP
1586 004114
```

1\$:

```

1588          :TEST 4
1589          :UNDERLINE TEST
1590
1591 004114    ULPRT: $ENABLE
(1) 004114 022737 000176 001004    CMP      #176,SWR      ;S/W SWR ?
(1) 004122 001003          BNE      .+10        ;NO- CONTINUE
(1) 004124 052777 000100 174666    BIS      #100,@TKS   ;ENABLE KEYBOARD INTERRUPT
1592 004132 004437 010726          JSR      %4,TYPINT
1593 004136          $PRINT  \M
(1) 004136 013737 014066 013250    MOV      TNJ4,MES15  ;SET TEST NUMBER FOR MESSAGE
(1) 004144 004437 010656          JSR      %4,PRNNT   ;PRINT TEST NUMBER
(1)          000005          M=M+1
1594 004150 012737 000200 001060    UL00A:  MOV      #200,LEGCHR ;LEGAL CHECK
1595 004156 012737 000140 001062    MOV      #140,NUMCHR ;#CHARS
1596 004164 012737 000040 001054    UL00:   MOV      #40,STRCHR  ;SET UP FIRST LINE
1597 004172 012737 000100 001042    MOV      #64,LINCNT  ;SET LINE COUNT FOR 1 PAGE
1598 004200 012737 177574 001036    UL0:    MOV      #-132,CHRCNT ;SET CHAR COUNT
1599 004206 013737 001054 001040    MOV      STRCHR,CHRGEN ;STORE START CHAR
1600 004214 005777 174560          UL1:    TST      @LPS      ;TEST FOR ERROR
1601 004220 100006          BPL      UL2        ;BRANCH IF NO ERROR
1602 004222          $ERROR  \N
(1) 004222 012737 000030 001052    ERR30:  MOV      #30,  ERCOUNT ;SET UP ERROR COUNT 30
(1)          000031          N=N+1
1603 004230 004537 011142          JSR      %5,STAER   ;REPORT ERROR SET
1604 004234 000000          HALT                    ;HALT ON ERROR
1605 004236 013777 001040 174536    UL2:    MOV      CHRGEN,@LPB ;LOAD BUFFER
1606 004244 005237 001036          INC      CHRCNT     ;INCREMENT CHAR COUNT
1607 004250 001412          BEQ      UL4        ;BRANCH IF DONE LINE
1608 004252 005237 001040          INC      CHRGEN    ;NEXT CHAR
1609 004256 023737 001060 001040    CMP      LEGCHR,CHRGEN ;LEGAL CHAR?
1610 004264 001353          BNE      UL1        ;BRANCH AND CONTINUE IF LEGAL CHAR
1611 004266 163737 001062 001040    SUB      NUMCHR,CHRGEN ;MAKE LEGAL
1612 004274 000747          BR      UL1         ;CONTINUE
1613 004276 012777 000015 174476    UL4:    MOV      #15,@LPB  ;SEND CR
1614 004304          $WAIT
(1) 004304 105777 174470          TSTB    @LPS        ;TEST READY
(1) 004310 100375          BPL      .-4        ;WAIT FOR READY
1615 004312 012737 177574 001036    MOV      #-132,CHRCNT ;RESET COUNT
1616 004320 012777 000137 174454    UL7:    MOV      #137,@LPB ;SEND UNDERLINE CODE
1617 004326 005777 174446          UL8:    TST      @LPS    ;READY?
1618 004332 100006          BPL      UL9        ;BR=NO ERROR
1619 004334          $ERROR  \N
(1) 004334 012737 000031 001052    ERR31:  MOV      #31,  ERCOUNT ;SET UP ERROR COUNT 31
(1)          000032          N=N+1
1620 004342 004537 011142          JSR      R5,STAER   ;REPORT ERROR SET
1621 004346 000000          HALT                    ;
1622 004350 005237 001036          UL9:    INC      CHRCNT  ;COUNT A CHARACTER
1623 004354 001361          BNE      UL7        ;BR=NOT DONE
1624 004356 012777 000012 174416    MOV      #12,@LPB  ;SEND LF
1625 004364          $WAIT
(1) 004364 105777 174410          TSTB    @LPS        ;TEST READY
(1) 004370 100375          BPL      .-4        ;WAIT FOR READY
1626 004372 005337 001042          DEC      LINCNT    ;DECREMENT LINE COUNT
1627 004376 002404          BLT      UL6        ;EXIT TEST IF DONE
1628 004400 012737 000040 001054    MOV      #40,STRCHR ;START CHAR A LEGAL CHAR?
1629 004406 000674          BR      UL0        ;CONTINUE IF LEGAL START CHAR

```

1630	004410	032777	010000	174366	UL6:	BIT	#BIT12,@SWR	:LOOP ON TEST?
1631	004416	001236				BNE	ULPRT	:LOOP
1632								
1633								
1634								
1635								

```

1637
1638
1639
1640
1641
1642
1643
1644 004420
(1) 004420 022737 000176 001004
(1) 004426 001003
(1) 004430 052777 000100 174362
1645 004436 004437 010726
1646 004442
(1) 004442 013737 014070 013250
(1) 004450 004437 010656
(1) 000006
1647 004454 012737 177760 001042
1648 004462 012737 177574 001036
1649 004470 012737 177776 001044
1650 004476 013737 001036 001056
1651 004504 062737 000205 001056
1652 004512 012737 000040 001040
1653 004520 000406
1654 004522 012737 000105 001040
1655 004530 013777 001040 174244
1656 004536 005777 174236
1657 004542 100006
1658 004544
(1) 004544 012737 000032 001052
(1) 000033
1659 004552 004537 011142
1660 004556 000000
1661 004560 005337 001056
1662 004564 003361
1663 004566 001755
1664 004570 005237 001036
1665 004574 001437
1666 004576 005237 001044
1667 004602 001352
1668 004604 012777 000015 174170
1669 004612
(1) 004612 105777 174162
(1) 004616 100375
1670 004620 005737 001036
1671 004624 001321
1672 004626 005237 001042
1673 004632 001425
1674 004634 032737 000001 001042
1675 004642 001707
1676 004644 012737 000115 001040
1677 004652 012737 177573 001036
1678 004660 005037 001056
1679 004664 005037 001044
1680 004670 000137 004536
1681 004674 012777 000012 174100
1682 004702 000137 004612

;TEST 5
;LINE EDITING TEST
;SENDS PAIRS OF E'S, THEN OVER PRINTS THEM WITH SPACES AND ADDS ANOTHER
;PAIR OF E'S TO THE LINE --- THIS IS REPEATED UNTIL A FULL LINE OF E'S
;HAVE BEEN PRINTED, THEN A FULL LINE OF M'S ARE PRINTED.

OVRPRT: $ENABLE
CMP #176,SWR ;S/W SWR ?
BNE .+10 ;NO- CONTINUE
BIS #100,@TKS ;ENABLE KEYBOARD INTERRUPT
JSR %4,TYPINT
$PRINT \M
MOV TN05,MES15 ;SET TEST NUMBER FOR MESSAGE
JSR %4,PRNNT ;PRINT TEST NUMBER
M=M+1
MOV #-16.,LINCNT ;SET LINE COUNT FOR 16 LINES
OVR: MOV #-132.,CHRCNT ;SET CHAR COUNT
OVR0: MOV #-2,CYCCNT ;SET CYCLE COUNT FOR A PAIR OF E'S
MOV CHRCNT,STRCNT ;NO. CHARS LEFT TO PRINT
ADD #133.,STRCNT ;NO. SPACES +1
MOV #40,CHRGEN ;SEND SPACE
BR OVR2 ;BRANCH
OVR4: MOV #105,CHRGEN ;SEND E
OVR1: MOV CHRGEN,@LPB ;LOAD BUFFER
OVR2: TST @LPS ;TEST FOR ERROR
BPL OVR3 ;BRANCH IF NO ERROR
$ERROR \N
ERR32: MOV #32, ERCOUNT ;SET UP ERROR COUNT 32
N=N+1
JSR %5,STAER ;REPORT ERROR SET
HALT
OVR3: DEC STRCNT ;DECREMENT SPACE COUNTER
BGT OVR1 ;BRANCH IF NOT DONE SPACES
BEQ OVR4 ;BRANCH IF NOT FIRST E
INC CHRCNT ;INCREMENT CHAR COUNT
BEQ OVR8 ;BRANCH IF DONE LINE
OVR5: INC CYCCNT ;INCREMENT CYCLE COUNT
BNE OVR1 ;CONTINUE SENDING E'S IF NOT DONE
MOV #15,@LPB ;SEND CR
OVR6: $WAIT
TSTB @LPS ;TEST READY
BPL .-4 ;WAIT FOR READY
TST CHRCNT ;LINE DONE?
BNE OVR0 ;NO, CONTINUE OVER PRINT
INC LINCNT ;YES, INCREMENT LINE COUNT
BEQ OVREXT ;EXIT IF DONE TEST
BIT #1,LINCNT ;WHICH LINE NEXT?
BEQ OVR ;BRANCH TO SEND E'S
MOV #115,CHRGEN ;SET UP TO SEND M'S
MOV #-133.,CHRCNT ;SET CHAR COUNT
CLR STRCNT ;CLEAR SPACE COUNT
CLR CYCCNT ;CLEAR CYCLE COUNT
JMP OVR2 ;PRINT LINE OF M'S
OVR8: MOV #12,@LPB ;SEND LF
JMP OVR6 ;CONTINUE

```

YM-Z049A-AA (DECSPEC-11-BDFAD-A-D) MACY11 30A(1052) 12-MAR-81^{M 3} 11:24 PAGE 22-1
CZLXAA.SRC 12-MAR-81 10:43

SEQ 0038

1683 004706 032777 010000 174070 OVREXT: BIT #BIT12,@SWR ;LOOP ON TEST?
1684 004714 001241 BNE OVRPRT ;LOOP

```

1686 :TEST 6
1687 :PRINT CONTROL TEST
1688 :SENDS FULL LINE OF SAME CHARACTER THEN FULL CHAR SET
1689 :SHOULD ONLY PRINT THE FIRST 132 CHARACTERS RECEIVED
1690
1691 004716 022737 000176 001004 PRTCTL: $ENABLE
(1) 004716 001003 CMP #176,SWR ;S/W SWR ?
(1) 004724 052777 000100 174064 BNE .+10 ;NO- CONTINUE
1692 004734 004437 010726 BIS #100,@TKS ;ENABLE KEYBOARD INTERRUPT
1693 004740 $PRINT \M
(1) 004740 013737 014072 013250 MOV TNO6,MES15 ;SET TEST NUMBER FOR MESSAGE
(1) 004746 004437 010656 JSR %4,PRNNT ;PRINT TEST NUMBER
(1) 000007 M=M+1
1694 004752 012737 000060 001054 MOV #60,STRCHR ;FIRST START CHAR
1695 004760 012737 177641 001034 PRT0: MOV #-95,SEGCNT ;SET OVERFLOW COUNT
1696 004766 012737 177574 001036 PRT2: MOV #-132,CHRCNT ;SET CHAR COUNT
1697 004774 013737 001054 001040 MOV STRCHR,CHRCNT ;GET START CHAR
1698 005002 005777 173772 PRT3: TST @LPS ;TEST FOR ERROR
1699 005006 100006 BPL PRT4 ;BRANCH IF NO ERROR
1700 005010 $ERROR \N
(1) 005010 012737 000033 001052 ERR33: MOV #33, ERCOUNT ;SET UP ERROR COUNT 33
(1) 000034 N=N+1
1701 005016 004537 011142 JSR %5,STAER ;REPORT ERROR SET
1702 005022 000000 HALT ;HALT ON ERROR
1703 005024 013777 001040 173750 PRT4: MOV CHRCNT,@LPB ;LOAD BUFFER
1704 005032 005237 001036 INC CHRCNT ;INCREMENT CHAR COUNT
1705 005036 002761 BLT PRT3 ;BRANCH IF NOT 132 CHARS
1706 005040 001433 BEQ PRTA ;START OVERFLOW
1707 005042 005237 001040 INC CHRCNT ;NEXT CHAR
1708 005046 005237 001034 INC SEGCNT ;INCREMENT OVERFLOW COUNT
1709 005052 001353 BNE PRT3 ;CONTINUE IF NOT DONE
1710 005054 012777 000012 173720 MOV #12,@LPB ;SEND LF
1711 005062 $WAIT
(1) 005062 105777 173712 TSTB @LPC ;TEST READY
(1) 005066 100375 BPL .-4 ;WAIT FOR READY
1712 005070 022737 000040 001054 CMP #40,STRCHR ;LAST START CHAR SPACE?
1713 005076 001421 BEQ PRT6 ;YES, BRANCH
1714 005100 022737 000065 001054 CMP #65,STRCHR ;LAST START CHAR 5?
1715 005106 001422 BEQ PRT7 ;YES, BRANCH
1716 005110 022737 000071 001054 CMP #71,STRCHR ;DONE?
1717 005116 001423 BEQ PRT8 ;YES
1718 005120 005237 001054 INC STRCHR ;NO, GET NEXT START CHAR
1719 005124 000137 004760 JMP PRT0 ;CONTINUE
1720 005130 012737 000041 001040 PRTA: MOV #41,CHRCNT ;GET FIRST CHAR IN SET
1721 005136 000137 005002 JMP PRT3 ;START OVERFLOW
1722 005142 012737 000066 001054 PRT6: MOV #66,STRCHR ;SET START CHAR TO 6
1723 005150 000137 004760 JMP PRT0 ;CONTINUE
1724 005154 012737 000040 001054 PRT7: MOV #40,STRCHR ;SET START CHAR TO SPACE
1725 005162 000137 004760 JMP PRT0 ;CONTINUE
1726 005166 032777 010000 173610 PRT8: BIT #BIT12,@SWR ;CHECK LOOP ON TEST
1727 005174 001250 BNE PRTCTL ;LOOP

```



```
1729 ;TEST 7
1730 ;MULTIPLE LINE ADVANCE TEST
1731 ;TESTS MULTIPLE LINE ADVANCES AND TIMINGS
1732 ;PRINTS THE NUMBER OF LINES SKIPPED ON THE LINE PRINTER
1733
1734 005176 MLF: $ENABLE
(1) 005176 022737 000176 001004 CMP #176,SWR ;S/W SWR ?
(1) 005204 001003 BNE .+10 ;NO- CONTINUE
(1) 005206 052777 000100 173604 BIS #100,@TKS ;ENABLE KEYBOARD INTERRUPT
1735 005214 004437 010726 JSR %4,TYPINT
1736 005220 $PRINT \M
(1) 005220 013737 014074 013250 MOV TNO7,MES15 ;SET TEST NUMBER FOR MESSAGE
(1) 005226 004437 010656 JSR %4,PRNNT ;PRINT TEST NUMBER
(1) 000010 M=M+1
1737 005232 012737 005364 001054 MOV #TABSTR,STRCHR ;FIRST CHAR
1738 005240 012737 177574 001036 MLFA: MOV #-132.,CHRCNT ;SET CHAR COUNT
1739 005246 117737 173602 001040 MOVB @STRCHR,CHRCNT ;GET CHAR
1740 005254 001452 BEQ MLF4 ;BRANCH IF DONE
1741 005256 005777 173516 MLF0: TST @LPS ;TEST FOR ERROR
1742 005262 100006 BPL MLF1 ;CONTINUE IF NO ERROR
1743 005264 $ERROR \N
(1) 005264 012737 000034 001052 ERR34: MOV #34, ERRCOUNT ;SET UP ERROR COUNT 34
(1) 000035 N=N+1
1744 005272 004537 011142 JSR %5,STAER ;REPORT ERROR
1745 005276 000000 HALT ;HALT ON ERROR
1746 005300 013777 001040 173474 MLF1: MOV CHRCNT,@LPB ;LOAD BUFFER
1747 005306 005237 001036 INC CHRCNT ;INCREMENT CHAR COUNT
1748 005312 001361 BNE MLF0 ;CONTINUE
1749 005314 117737 173534 001042 MOVB @STRCHR,LINCNT ;GET ASCII LINE COUNT
1750 005322 042737 177770 001042 BIC #177770,LINCNT ;MAKE OCTAL
1751 005330 005237 001042 INC LINCNT ;ADD 1
1752 005334 012777 000012 173440 MLF2: MOV #12,@LPB ;SEND LF
1753 005342 $WAIT
(1) 005342 105777 173432 TSTB @LPS ;TEST READY
(1) 005346 100375 BPL .-4 ;WAIT FOR READY
1754 005350 005337 001042 DEC LINCNT ;DECREMENT LINE COUNT
1755 005354 001367 BNE MLF2 ;CONTINUE
1756 005356 005237 001054 INC STRCHR ;NEXT CHAR
```

```
1760 005362 000726          BR      MLFA          ;CONTINUE
1761
1762 005364 033462 033062 033463 TABSTR: .ASCIZ /272637463540/
      005372 033064 032463 030064
      005400      000
1763
1764          005402          .EVEN
1765
1766 005402 032777 010000 173374 MLF4:  BIT      #BIT12,@SWR      ;CHECK LOOP ON TEST
1767 005410 001272          BNE     MLF          ;LOOP
```

```
1769 ;TEST 8
1770 ;HIGH SPEED PRINT TEST
1771
1772 005412 HSPRT: $ENABLE
(1) 005412 022737 000176 001004 CMP #176,SWR ;S/W SWR ?
(1) 005420 001003 BNE .+10 ;NO- CONTINUE
(1) 005422 052777 000100 173370 BIS #100,@TKS ;ENABLE KEYBOARD INTERRUPT
1773 005430 004437 010726 JSR %4,TYPINT
1774 005434 $PRINT \M
(1) 005434 013737 014076 013250 MOV TNO10,MES15 ;SET TEST NUMBER FOR MESSAGE
(1) 005442 004437 010656 JSR %4,PRNNT ;PRINT TEST NUMBER
(1) 000011 M=M+1
1775 005446 012737 000200 001060 HS00A: MOV #200,LEGCHR ;LEGAL CHECK
1776 005454 012737 000140 001062 MOV #140,NUMCHR ;#CHARS
1777 005462 012737 000040 001054 HS00: MOV #40,STRCHR ;SET UP FIRST LINE
1778 005470 012737 000177 001042 MOV #127.,LINCNT ;SET LINE COUNT FOR 2 PAGES
1779 005476 012737 177574 001036 HS0: MOV #-132.,CHRCNT ;SET CHAR COUNT
1780 005504 012737 177757 001044 MOV #-17.,CYCCNT ;SET GROUP COUNT
1781 005512 013737 001054 001040 MOV STRCHR,CHRGEN ;STORE START CHAR
1782 005520 005777 173254 HS1: TST @LPS ;TEST FOR ERROR
1783 005524 100006 BPL HS2 ;BRANCH IF NO ERROR
1784 005526 $ERROR \N
(1) 005526 012737 000035 001052 ERR35: MOV #35, ERCOUNT ;SET UP ERROR COUNT 35
(1) 000036 N=N+1
1785 005534 004537 011142 JSR %5,STAER ;REPORT ERROR SET
1786 005540 000000 HALT ;HALT ON ERROR
1787 005542 013777 001040 173232 HS2: MOV CHRGEN,@LPB ;LOAD BUFFER
1788 005550 005237 001036 INC CHRCNT ;INCREMENT CHAR COUNT
1789 005554 001424 BEQ HS4 ;BRANCH IF DONE LINE
1790 005556 005237 001040 INC CHRGEN ;NEXT CHAR
1791 005562 005237 001044 INC CYCCNT ;INCREMENT GROUP COUNT
1792 005566 001410 BEQ HS3 ;BRANCH IF DONE GROUP
1793 005570 023737 001060 001040 CMP LEGCHR,CHRGEN ;LEGAL CHAR?
1794 005576 001350 BNE HS1 ;BRANCH AND CONTINUE IF LEGAL CHAR
1795 005600 163737 001062 001040 SUB NUMCHR,CHRGEN ;MAKE LEGAL
1796 005606 000744 BR HS1 ;CONTINUE
1797 005610 013737 001054 001040 HS3: MOV STRCHR,CHRGEN ;GET FIRST CHAR IN GROUP
1798 005616 012737 177757 001044 MOV #-17.,CYCCNT ;RESET CYCLE COUNT
1799 005624 000735 BR HS1 ;CONTINUE
1800 005626 012777 000012 173146 HS4: MOV #12,@LPB ;SEND LF
1801 005634 $WAIT
(1) 005634 105777 173140 TSTB @LPS ;TEST READY
(1) 005640 100375 BPL .-4 ;WAIT FOR READY
1802 005642 005337 001042 DEC LINCNT ;DECREMENT LINE COUNT
1803 005646 002413 BLT HS6 ;EXIT TEST IF DONE
1804 005650 162737 000004 001054 SUB #4,STRCHR ;SKIP 4 LINES ON DRUM, FIND START CHAR
1805 005656 022737 000040 001054 CMP #40,STRCHR ;START CHAR A LEGAL CHAR?
1806 005664 003704 BLE HS0 ;CONTINUE IF LEGAL START CHAR
1807 005666 063737 001062 001054 ADD NUMCHR,STRCHR ;MAKE LEGAL AND CONTINUE
1808 005674 000700 BR HS0 ;CONTINUE
1809 005676 032777 010000 173100 HS6: BIT #BIT12,@SWR ;LOOP ON TEST?
1810 005704 001242 BNE HSPRT ;LOOP
1811
1812
1813 005706 000137 005712 JMP SNGCHR ;JUMP TO TEST 9 AFTER COMPLETION
1814
```

```
1815
1816
1817
1818
1819
1820 005712
(1) 005712 022737 000176 001004
(1) 005720 001003
(1) 005722 052777 000100 173070
1821 005730 004437 010726
1822 005734
(1) 005734 013737 014100 013250
(1) 005742 004437 010656
(1) 000012
1823 005746 012737 177640 001042
1824 005754 012737 000040 001040
1825 005762 012737 177574 001036
1826 005770 005777 173004
1827 005774 100006
1828 005776
(1) 005776 012737 000036 001052
(1) 000037
1829 006004 004537 011142
1830 006010 000000
1831 006012 013777 001040 172762
1832 006020 005237 001036
1833 006024 001361
1834 006026 012777 000012 172746
1835 006034
(1) 006034 105777 172740
(1) 006040 100375
1836 006042 005237 001040
1837 006046 005237 001042
1838 006052 002743
1839 006054 001764
1840 006056 032777 010000 172720
1841 006064 001312

:TEST 9
:WORST CASE NOISE TEST
:SINGLE CHAR. ACROSS ALL COLS.

SNGCHR: $ENABLE
CMP #176,SWR ;S/W SWR ?
BNE .+10 ;NO- CONTINUE
BIS #100,@TKS ;ENABLE KEYBOARD INTERRUPT
JSR %4,TYPINT
$PRINT \M
MOV TNO11,MES15 ;SET TEST NUMBER FOR MESSAGE
JSR %4,PRNNT ;PRINT TEST NUMBER
M=M+1
MOV #-96.,LINCNT ;96 CHAR.
MOV #40,CHRGEN ;SET UP SPACE
S2A: MOV #-132.,CHRCNT ;SET CHAR COUNT FOR 132
S1: TST @LPS ;TEST FOR ERRORS
BPL XS1X ;BRANCH IF NO ERRORS
$ERROR \N
ERR36: MOV #36, ERCOUNT ;SET UP ERROR COUNT 36
N=N+1
JSR %5,STAER ;REPORT ERROR
HALT ;HALT ON ERROR
XS1X: MOV CHRGEN,@LPB ;LOAD PRINTER BUFFER
INC CHRCNT ;INCREMENT CHAR COUNT
BNE S1 ;CONTINUE IF NOT DONE LINE
S4X2: MOV #12,@LPB ;ISSUE LINE FEED
$WAIT
TSTB @LPS ;TEST READY
BPL .-4 ;WAIT FOR READY
INC CHRGEN ;+1 CHAR.
INC LINCNT ;+1 LINE COUNT
BLT S2A ;CONTINUE IF NOT DONE
BEQ S4X2 ;SEND BLANK LINE AT END OF TEST
LPS7: BIT #BIT12,@SWR ;CHECK TO LOOP ON TEST
BNE SNGCHR ;LOOP ON TEST
```

```
1843 ;TEST 10
1844 ;PROM PATTERN CHARACTER TEST
1845
1846 ROTATE: $ENABLE
(1) 006066 022737 000176 001004 CMP #176,SWR ;S/W SWR ?
(1) 006074 001003 BNE .+10 ;NO- CONTINUE
(1) 006076 052777 000100 172714 BIS #100,@TKS ;ENABLE KEYBOARD INTERRUPT
1847 006104 004437 010726 JSR %4,TYPINT
1848 006110 $PRINT \M
(1) 006110 013737 014102 013250 MOV TNO12,MES15 ;SET TEST NUMBER FOR MESSAGE
(1) 006116 004437 010656 JSR %4,PRNNT ;PRINT TEST NUMBER
(1) 000013 M=M+1
1849
1850 ROT0: MOV #177,LINCNT ;LAST CHAR
1851 006130 012737 000200 001060 MOV #200,LEGCHR ;LEGAL CHK
1852 006136 012737 000140 001062 MOV #140,NUMCHR ;#CHARS
1853 006144 005037 001044 ROT1: CLR CYCCNT ;CLEAR CYCLE COUNT
1854 006150 005237 001044 ROT2: INC CYCCNT ;INC CYCLE COUNT
1855 006154 005037 001040 CLR CHRGEN ;CLEAR POINTER
1856 006160 005237 001040 ROT3: INC CHRGEN ;INC POINTER
1857 006164 013737 001040 001054 MOV CHRGEN,STRCHR ;STORE POINTER
1858 006172 063737 001042 001054 ADD LINCNT,STRCHR ;FIND CHAR
1859 006200 023737 001054 001060 CMP STRCHR,LEGCHR ;LEGAL?
1860 006206 002403 BLT ROT4 ;BRANCH IF LEGAL
1861 006210 163737 001062 001054 SUB NUMCHR,STRCHR ;MAKE LEGAL
1862 006216 005777 172556 ROT4: TST @LPS ;TEST FOR ERRORS
1863 006222 100006 BPL ROT5 ;BRANCH IF NO ERRORS
1864 006224 $ERROR \N
(1) 006224 012737 000037 001052 ERR37: MOV #37, ERCOUNT ;SET UP ERROR COUNT 37
(1) 000040 N=N+1
1865 006232 004537 011142 JSR %5,STAER ;REPORT ERROR
1866 006236 000000 HALT ;HALT ON ERROR
1867 006240 013777 001054 172534 ROT5: MOV STRCHR,@LPB ;LOAD BUFFER
1868 006246 023727 001040 000021 CMP CHRGEN,#17. ;DONE GROUP?
1869 006254 001341 BNE ROT3 ;NO GET NEXT CHAR
1870 006256 023727 001044 000010 CMP CYCCNT,#8. ;DONE LINE?
1871 006264 001331 BNE ROT2 ;NO, NEXT GROUP
1872 006266 012777 000012 172506 MOV #12,@LPB ;YES, SEND LF
1873 006274 $WAIT
(1) 006274 105777 172500 TSTB @LPS ;TEST READY
(1) 006300 100375 BPL .-4 ;WAIT FOR READY
1874 006302 005337 001042 DEC LINCNT ;DECREMENT LINE COUNT
1875 006306 023727 001042 000037 CMP LINCNT,#37 ;DONE?
1876 006314 003313 BGT ROT1 ;NO, NEXT LINE
1877 006316 032777 010000 172460 BIT #BIT12,@SWR ;LOOP ON TEST?
1878 006324 001260 BNE ROTATE ;LOOP
1879
1880 006326 000137 006332 JMP LFTTR ;JUMP TO TEST 11 AFTER COMPLETION
1881
1882
```

```

1884          :TEST 11 ----- SPURIOUS HAMMER FIRING TEST
1885          :LEFT AND RIGHT TRIANGLES
1886
1887          : STARTING WITH A LEFT TRIANGLE
1888
1889 006332    LFTTR: $ENABLE
(1) 006332    022737 000176 001004    CMP      #176,SWR          ;S/W SWR ?
(1) 006340    001003          BNE      .+10           ;NO- CONTINUE
(1) 006342    052777 000100 172450    BIS      #100,@TKS      ;ENABLE KEYBOARD INTERRUPT
1890 006350    004437 010726          JSR      %4,TYPINT
1891 006354          $PRINT  \M
(1) 006354    013737 014104 013250    MOV      TNO13,MES15    ;SET TEST NUMBER FOR MESSAGE
(1) 006362    004437 010656          JSR      %4,PRNNT      ;PRINT TEST NUMBER
(1)          000014          M=M+1
1892 006366    012737 000204 001042    LFT:    MOV      #132.,LINCNT ;SET LINE COUNT
1893 006374    013737 001042 001036    LFT0:   MOV      LINCNT,CHRCNT ;STORE CHAR COUNT
1894 006402    012737 177757 001044          MOV      #-17.,CYCCNT  ;SET GROUP COUNT
1895 006410    013737 001036 001040          MOV      CHRCNT,CHRCNT ;FIND FIRST CHAR ON LINE...
1896 006416    022737 000022 001040    LFT1:   CMP      #18.,CHRCNT  ;MORE THAN 17 CHARS?
1897 006424    003004          BGT      LFT2          ;BRANCH IF LESS THAN 17
1898 006426    162737 000021 001040          SUB      #17.,CHRCNT  ;SUBTRACT 17, IF > 17
1899 006434    000770          BR       LFT1          ;CONTINUE
1900 006436    005437 001040          LFT2:   NEG      CHRCNT     ;NEGATE CHRCNT
1901 006442    062737 000100 001040          ADD      #100,CHRCNT  ;START CHAR IN CHRCNT
1902 006450    013737 001040 001054          MOV      CHRCNT,STRCHR ;STORE STARTING CHAR
1903 006456    005777 172316          LFT3:   TST      @LPS      ;TEST FOR ERROR
1904 006462    100006          BPL      LFT4          ;CONTINUE IF NO ERROR
1905 006464          $ERROR  \N
(1) 006464    012737 000040 001052  ERR40: MOV      #40,   ERCOUNT ;SET UP ERROR COUNT 40
(1)          000041          N=N+1
1906 006472    004537 011142          JSR      %5,STAER     ;REPORT ERROR SET
1907 006476    000000          HALT                    ;HALT ON ERROR
1908 006500    013777 001040 172274    LFT4:   MOV      CHRCNT,@LPB  ;LOAD BUFFER
1909 006506    005337 001036          DEC      CHRCNT       ;DECREMENT CHAR COUNT
1910 006512    001415          BEQ      LFT6         ;BRANCH IF DONE LINE
1911 006514    005237 001044          INC      CYCCNT       ;INCREMENT GROUP COUNT
1912 006520    001403          BEQ      LFT5         ;BRANCH IF DONE GROUP
1913 006522    005237 001040          INC      CHRCNT       ;NEXT CHAR IN GROUP
1914 006526    000753          BR       LFT3         ;CONTINUE
1915 006530    013737 001054 001040    LFT5:   MOV      STRCHR,CHRCNT ;GET START CHAR AGAIN
1916 006536    012737 177757 001044          MOV      #-17.,CYCCNT ;RESET GROUP COUNT
1917 006544    000744          BR       LFT3         ;CONTINUE
1918 006546    012777 000012 172226    LFT6:   MOV      #12,@LPB    ;SEND LF
1919 006554          $WAIT
(1) 006554    105777 172220          TSTB    @LPS          ;TEST READY
(1) 006560    100375          BPL      .-4          ;WAIT FOR READY
1920 006562    005337 001042          DEC      LINCNT       ;DECREMENT LINE COUNT
1921 006566    003302          BGT      LFT0         ;BRANCH IF NOT DONE
1922 006570    001766          BEQ      LFT6         ;SEND BLANK LINE AT END OF TEST
1923 006572    032777 010000 172204          BIT      #BIT12,@SWR  ;LOOP ON TEST?
1924 006600    001254          BNE      LFTTR        ;LOOP
1925
1926          :TEST 11 ----- CONTINUED
1927          :RIGHT TRIANGLE
1928
1929 006602    012737 000001 001042    RTRR:   MOV      #1,LINCNT ;INITIALIZE LINE

```

```

1930 006610 012737 000077 001040 RT1:  MOV #77,CHRGEN ;FIRST CHAR IS A ?
1931 006616 013737 001042 001044      MOV LINCNT,CYCCNT ;SAVE NO. CHARS ON LINE
1932 006624 012737 177757 001056      MOV #-17.,STRCNT ;SET GROUP COUNT
1933 006632 012737 000204 001036      MOV #132.,CHRCNT ;NO. CHARS PER LINE
1934 006640 163737 001042 001036      SUB LINCNT,CHRCNT ;SUBTRACT NO. OF CHARS ON LINE
1935 006646 001425                BEQ RT3 ;BRANCH IF NO SPACES ON THIS LINE
1936 006650 005777 172124      RT2:  TST @LPS ;TEST FOR ERROR
1937 006654 100006                BPL RT2A ;CONTINUE IF NO ERROR
1938 006656                $ERROR \N
(1) 006656 012737 000041 001052 ERR41: MOV #41, ERRCOUNT ;SET UP ERROR COUNT 41
(1) 000042
1939 006664 004537 011142      JSR %5,STAER ;REPORT ERROR SET
1940 006670 000000                HALT ;HALT ON ERROR
1941 006672 012777 000040 172102 RT2A:  MOV #40,@LPB ;LOAD BUFFER
1942 006700 005237 001056      INC STRCNT ;INCREMENT GROUP COUNT
1943 006704 001003                BNE RT2AA ;BRANCH IF NOT DONE GROUP
1944 006706 012737 177757 001056      MOV #-17.,STRCNT ;RESET GROUP COUNT
1945 006714 005337 001036      RT2AA: DEC CHRCNT ;DECREMENT SPACE COUNT
1946 006720 001353                BNE RT2 ;BRANCH IF NOT DONE SPACES
1947 006722 005777 172052      RT3:  TST @LPS ;TEST FOR ERROR
1948 006726 100006                BPL RT3A ;CONTINUE IF NO ERROR
1949 006730                $ERROR \N
(1) 006730 012737 000042 001052 ERR42: MOV #42, ERRCOUNT ;SET UP ERROR COUNT 42
(1) 000043
1950 006736 004537 011142      JSR %5,STAER ;REPORT ERROR SET
1951 006742 000000                HALT ;HALT ON ERROR
1952 006744 013777 001040 172030 RT3A:  MOV CHRGEN,@LPB ;LOAD BUFFER
1953 006752 005237 001040      INC CHRGEN ;NEXT CHAR
1954 006756 005237 001056      INC STRCNT ;INCREMENT GROUP COUNT
1955 006762 001006                BNE RT3B ;BRANCH IF NOT DONE GROUP
1956 006764 012737 177757 001056      MOV #-17.,STRCNT ;RESET GROUP COUNT
1957 006772 162737 000021 001040      SUB #17.,CHRGEN ;GET FIRST GROUP CHAR
1958 007000 005337 001044      RT3B: DEC CYCCNT ;DECREMENT CHAR COUNT
1959 007004 001346                BNE RT3 ;CONTINUE
1960 007006 012777 000012 171766      MOV #12,@LPB ;SEND LF
1961 007014                $WAIT
(1) 007014 105777 171760      TSTB @LPS ;TEST READY
(1) 007020 100375                BPL .-4 ;WAIT FOR READY
1962 007022 005237 001042      INC LINCNT ;INCREMENT LINE COUNT
1963 007026 022737 000205 001042      CMP #133.,LINCNT ;DONE?
1964 007034 003265                BGT RT1 ;BRANCH IF NOT DONE
1965 007036 032777 010000 171740      BIT #BIT12,@SWR ;LOOP ON TEST?
1966 007044 001256                BNE RTTR ;LOOP

```

```

1968      ;TEST 12
1969      ;HAMMER ALIGMENT
1970
1971 007046      HAMALN: $ENABLE
(1) 007046 022737 000176 001004      CMP      #176,SWR      ;S/W SWR ?
(1) 007054 001003      BNE      .+10      ;NO- CONTINUE
(1) 007056 052777 000100 171734      BIS      #100,@TKS      ;ENABLE KEYBOARD INTERRUPT
1972 007064 004437 010726      JSR      %4,TYPINT
1973 007070      $PRINT      VM
(1) 007070 013737 014106 013250      MOV      TNJ14,MES15      ;SET TEST NUMBER FOR MESSAGE
(1) 007076 004437 010656      JSR      %4,PRNNT      ;PRINT TEST NUMBER
(1) 000015      M=M+1
1974 007102 012737 177701 001042      MOV      #-63.,LINCNT      ;SET UP FOR 63 LINES
1975 007110 012737 177574 001036      HAM1X: MOV      #-132.,CHRCNT      ;SET CHAR COUNT
1976 007116 005777 171656      HAM2:  TST      @LPS      ;CHECK FOR ERROR
1977 007122 100006      BPL      XHAM1      ;BRANCH IF NO ERROR
1978 007124      $ERROR      \N
(1) 007124 012737 000043 001052      ERR43: MOV      #43, ERCCOUNT      ;SET UP ERROR COUNT 43
(1) 000044      N=N+1
1979 007132 004537 011142      JSR      %5,STAER      ;REPORT ERROR OCCURRED
1980 007136 000000      HALT      ;HALT ON ERROR
1981 007140      XHAM1: $WAIT
(1) 007140 105777 171634      TSTB     @LPS      ;TEST READY
(1) 007144 100375      BPL      .-4      ;WAIT FOR READY
1982 007146 100375      BPL      .-4      ;WAIT FOR READY
1983 007150 012777 000105 171624      XHAM1X: MOV      #105,@LPB      ;TRANSMIT E TO PRINTER
1984 007156 005237 001036      INC      CHRCNT      ;+1 CHAR COUNT
1985 007162 001355      BNE      HAM2      ;TRANSMIT ANOTHER CHAR.
1986 007164 012777 000012 171610      MOV      #12,@LPB      ;TRANSMIT LINE FEED
1987 007172      $WAIT
(1) 007172 105777 171602      TSTB     @LPS      ;TEST READY
(1) 007176 100375      BPL      .-4      ;WAIT FOR READY
1988 007200 005237 001042      INC      LINCNT      ;+1 TO COUNT
1989 007204 001341      BNE      HAM1X      ;GO DO NEXT LINE
1990 007206 032777 010000 171570      BIT      #BIT12,@SWR      ;CHECK TO LOOP ON TEST
1991 007214 001314      BNE      HAMALN      ;LOOP ON TEST
1992

```



```

1994
1995
1996
1997
1998 007216
(1) 007216 022737 000176 001004
(1) 007224 001003
(1) 007226 052777 000100 171564
1999 007234 004437 010726
2000 007240
(1) 007240 013737 014110 013250
(1) 007246 004437 010656
(1) 000016
2001 007252 012737 000200 001060 EX00A: MOV #200,LEGCHR ;LEGAL CHECK
2002 007260 012737 000140 001062 MOV #140,NUMCHR ;#CHARS
2003 007266 012737 000040 001054 EX00: MOV #40,STRCHR ;SET UP FIRST LINE
2004 007274 012737 000060 001042 MOV #48,LINCNT ;SET LINE COUNT FOR 48 LINES
2005 007302 012737 177574 001036 EX0: MOV #-132,CHRCNT ;SET CHAR COUNT
2006 007310 012737 177757 001044 MOV #-17,CYCCNT ;SET GROUP COUNT
2007 007316 013737 001054 001040 MOV STRCHR,CHRGEN ;STORE START CHAR
2008 007324 005777 171450 EX1: TST @LPS ;TEST FOR ERROR
2009 007330 100006 BPL EX2 ;BRANCH IF NO ERROR
2010 007332
(1) 007332 012737 000044 001052 ERR44: MOV #44, ERCOUNT ;SET UP ERROR COUNT 44
(1) 000045
2011 007340 004537 011142 JSR %5,STAER ;REPORT ERROR SET
2012 007344 000000 HALT ;HALT ON ERROR
2013 007346 013777 001040 171426 EX2: MOV CHRGEN,@LPB ;LOAD BUFFER
2014 007354 005237 001036 INC CHRCNT ;INCREMENT CHAR COUNT
2015 007360 001424 BEQ EX4 ;BRANCH IF DONE LINE
2016 007362 005237 001040 INC CHRGEN ;NEXT CHAR
2017 007366 005237 001044 INC CYCCNT ;INCREMENT GROUP COUNT
2018 007372 001410 BEQ EX3 ;BRANCH IF DONE GROUP
2019 007374 023737 001060 001040 CMP LEGCHR,CHRGEN ;LEGAL CHAR?
2020 007402 001350 BNE EX1 ;BRANCH AND CONTINUE IF LEGAL CHAR
2021 007404 163737 001062 001040 SUB NUMCHR,CHRGEN ;MAKE LEGAL
2022 007412 000744 BR EX1 ;CONTINUE
2023 007414 013737 001054 001040 EX3: MOV STRCHR,CHRGEN ;GET FIRST CHAR IN GROUP
2024 007422 012737 177757 001044 MOV #-17,CYCCNT ;RESET CYCLE COUNT
2025 007430 000735 BR EX1 ;CONTINUE
2026 007432 012777 000010 171342 EX4: MOV #10,@LPB ;SEND ELONGATE CODE
2027 007440
(1) 007440 105777 171334 $WAIT
(1) 007444 100375 TSTB @LPS ;TEST READY
2028 007446 005777 171326 BPL -4 ;WAIT FOR READY
2029 007452 100006 TST @LPS ;ERROR?
2030 007454
(1) 007454 012737 000045 001052 ERR45: MOV #45, ERCOUNT ;SET UP ERROR COUNT 45
(1) 000046
2031 007462 004537 011142 JSR R5,STAER ;REPORT ERROR SET
2032 007466 000000 HALT
2033 007470 012777 000012 171304 EX7: MOV #12,@LPB ;SEND LF
2034 007476
(1) 007476 105777 171276 $WAIT
(1) 007502 100375 TSTB @LPS ;TEST READY
2035 007504 005337 001042 BPL -4 ;WAIT FOR READY
DEC LINCNT ;DECREMENT LINE COUNT

```

YM-Z049A-AA
CZLXAA.SRC

(DECSPEC-11-BDFAD-A-D)
12-MAR-81 10:43

MACY11 30A(1052) 12-MAR-81^{K 4} 11:24 PAGE 32-1

SEQ 0049

2036 007510 002413
2037 007512 162737 000004 001054
2038 007520 022737 000040 001054
2039 007526 003665
2040 007530 063737 001062 001054
2041 007536 000661
2042 007540 032777 010000 171236
2043 007546 001223
2044
2045
2046
2047

EX6:

BLT EX6
SUB #4,STRCHR
CMP #40,STRCHR
BLE EXO
ADD NUMCHR,STRCHR
BR EXO
BIT #BIT12,@SWR
BNE EXPRT

:EXIT TEST IF DONE
:SKIP 4 LINES ON DRUM, FIND START CHAR
:START CHAR A LEGAL CHAR?
:CONTINUE IF LEGAL START CHAR
:MAKE LEGAL AND CONTINUE
:CONTINUE
:LOOP ON TEST?
:LOOP

```

2049
2050      :TEST 14
2051      :UNDERLINED ELONGATED CHARACTERS TEST
2052
2053      UEXPRT: $ENABLE
(1) 007550 022737 000176 001004      CMP      #176,SWR      ;S/W SWR ?
(1) 007556 001003                    BNE      .+10        ;NO- CONTINUE
(1) 007560 052777 000100 171232      BIS      #100,@TKS   ;ENABLE KEYBOARD INTERRUPT
2054 007566 004437 010726      JSR      %4,TYPINT
2055 007572                    $PRINT   \M
(1) 007572 013737 014112 013250      MOV      TNO16,MES15 ;SET TEST NUMBER FOR MESSAGE
(1) 007600 004437 010656      JSR      %4,PRNNT    ;PRINT TEST NUMBER
(1) 000017
2056 007604 012737 000200 001060      UEX00A: MOV      #200,LEGCHR ;LEGAL CHECK
2057 007612 012737 000140 001062      MOV      #140,NUMCHR ;#CHARS
2058 007620 012737 000040 001054      UEX00:  MOV      #40,STRCHR ;SET UP FIRST LINE
2059 007626 012737 000060 001042      MOV      #48,LINCNT  ;SET LINE COUNT FOR 48 LINES
2060 007634 012737 177574 001036      UEX0:   MOV      #-132.,CHRCNT ;SET CHAR COUNT
2061 007642 012737 177757 001044      MOV      #-17.,CYCCNT ;SET GROUP COUNT
2062 007650 013737 001054 001040      MOV      STRCHR,CHRGEN ;STORE START CHAR
2063 007656 005777 171116      UEX1:   TST      @LPS    ;TEST FOR ERROR
2064 007662 100006                    BPL      UEX2        ;BRANCH IF NO ERROR
2065 007664                    $ERROR   \N
(1) 007664 012737 000046 001052      ERR46: MOV      #46,   ERCOUNT ;SET UP ERROR COUNT 46
(1) 000047
2066 007672 004537 011142                    JSR      %5,STAER    ;REPORT ERROR SET
2067 007676 000000                    HALT                      ;HALT ON ERROR
2068 007700 013777 001040 171074      UEX2:   MOV      CHRGEN,@LPB ;LOAD BUFFER
2069 007706 005237 001036                    INC      CHRCNT       ;INCREMENT CHAR COUNT
2070 007712 001424                    BEQ      UEX4         ;BRANCH IF DONE LINE
2071 007714 005237 001040                    INC      CHRGEN       ;NEXT CHAR
2072 007720 005237 001044                    INC      CYCCNT       ;INCREMENT GROUP COUNT
2073 007724 001410                    BEQ      UEX3         ;BRANCH IF DONE GROUP
2074 007726 023737 001060 001040      CMP      LEGCHR,CHRGEN ;LEGAL CHAR?
2075 007734 001350                    BNE      UEX1         ;BRANCH AND CONTINUE IF LEGAL CHAR
2076 007736 163737 001062 001040      SUB      NUMCHR,CHRGEN ;MAKE LEGAL
2077 007744 000744                    BR       UEX1         ;CONTINUE
2078 007746 013737 001054 001040      UEX3:   MOV      STRCHR,CHRGEN ;GET FIRST CHAR IN GROUP
2079 007754 012737 177757 001044      MOV      #-17.,CYCCNT ;RESET CYCLE COUNT
2080 007762 000735                    BR       UEX1         ;CONTINUE
2081 007764 012777 000010 171010      UEX4:   MOV      #10,@LPB   ;SEND ELONGATE CODE
2082 007772                    $WAIT
(1) 007772 105777 171002                    TSTB    @LPS         ;TEST READY
(1) 007776 100375                    BPL      .-4          ;WAIT FOR READY
2083 010000 005777 170774                    TST     @LPS         ;ERROR?
2084 010004 100006                    BPL      UEX7        ;BR=NO
2085 010006                    $ERROR   \N
(1) 010006 012737 000047 001052      ERR47: MOV      #47,   ERCOUNT ;SET UP ERROR COUNT 47
(1) 000050
2086 010014 004537 011142                    JSR      R5,STAER    ;REPORT ERROR SET
2087 010020 000000                    HALT
2088 010022 012777 000015 170752      UEX7:   MOV      #15,@LPB   ;SEND CR
2089 010030                    $WAIT
(1) 010030 105777 170744                    TSTB    @LPS         ;TEST READY
(1) 010034 100375                    BPL      .-4          ;WAIT FOR READY
2090 010036 012737 177574 001036      MOV      #-132.,CHRCNT ;RESET COUNT

```

```

2091 010044 012777 000137 170730 UEX7A: MOV #137,@LPB ;SEND UNDERLINE CODE
2092 010052 005777 170722 UEX8: TST @LPS ;READY?
2093 010056 100006 BPL UEX9 ;BR=NO ERROR
2094 010060 $ERROR \N
(1) 010060 012737 000050 001052 ERR50: MOV #50, ERCOUNT ;SET UP ERROR COUNT 50
(1) 000051 N=N+1
2095 010066 004537 011142 JSR R5,STAER ;REPORT ERROR SET
2096 010072 000000 HALT
2097 010074 005237 001036 UEX9: INC CHRCNT ;COUNT A CHARACTER
2098 010100 001361 BNE UEX7A ;BR=NOT DONE
2099 010102 012777 000012 170672 MOV #12,@LPB ;SEND LF
2100 010110 $WAIT
(1) 010110 105777 170664 TSTB @LPS ;TEST READY
(1) 010114 100375 BPL .-4 ;WAIT FOR READY
2101 010116 005337 001042 DEC LINCNT ;DECREMENT LINE COUNT
2102 010122 002413 BLT UEX6 ;EXIT TEST IF DONE
2103 010124 162737 000004 001054 SUB #4,STRCHR ;SKIP 4 LINES ON DRUM, FIND START CHAR
2104 010132 022737 000040 001054 CMP #40,STRCHR ;START CHAR A LEGAL CHAR?
2105 010140 003635 BLE UEX0 ;CONTINUE IF LEGAL START CHAR
2106 010142 063737 001062 001054 ADD NUMCHR,STRCHR ;MAKE LEGAL AND CONTINUE
2107 010150 000631 BR UEX0 ;CONTINUE
2108 010152 032777 010000 170624 UEX6: BIT #BIT12,@SWR ;LOOP ON TEST?
2109 010160 001402 BEQ UEX10 ;BR=NO
2110 010162 000137 007550 JMP UEXPRT ;LOOP
2111 010166 UEX10:
2112
2113
2114
2115

```

```

2117
2118
2119
2120
2121
2122 010166 032777 020000 170610 ACPRT: BIT #BIT13,@SWR ;CHECK FOR ALTERNATE CHARACTER SET TEST
2123 010174 001002 BNE ACPRTA ;BR IF PRESENT
2124 010176 000137 010526 JMP AC7 ;EXIT OTHERWISE
2125 010202 ACPRTA: $ENABLE
(1) 010202 022737 000176 001004 CMP #176,SWR ;S/W SWR ?
(1) 010210 001003 BNE .+10 ;NO- CONTINUE
(1) 010212 052777 000100 170600 BIS #100,@TKS ;ENABLE KEYBOARD INTERRUPT
2126 010220 004437 010726 JSR %4,TYPINT
2127 010224 $PRINT \M
(1) 010224 013737 014114 013250 MOV TNO17,MES15 ;SET TEST NUMBER FOR MESSAGE
(1) 010232 004437 010656 JSR %4,PRNNT ;PRINT TEST NUMBER
(1) 000020 M=M+1
2128 010236 012737 000200 001060 ACOOA: MOV #200,LEGCHR ;LEGAL CHECK
2129 010244 012737 000140 001062 MOV #140,NUMCHR ;#CHARS
2130 010252 012737 000040 001054 ACOO: MOV #40,STRCHR ;SET UP FIRST LINE
2131 010260 012737 000177 001042 MOV #127, LINCNT ;SET LINE COUNT FOR 2 PAGES
2132 010266 012737 177574 001036 ACO: MOV #-132, CHRCNT ;SET CHAR COUNT
2133 010274 012737 177757 001044 MOV #-17, CYCCNT ;SET GROUP COUNT
2134 010302 013737 001054 001040 MOV STRCHR,CHRGEN ;STORE START CHAR
2135 010310 005777 170464 ACOA: TST @LPS ;ERROR?
2136 010314 100006 BPL ACOB ;BR=NO
2137 010316 $ERROR \N
(1) 010316 012737 000051 001052 ERR51: MOV #51, ERCOUNT ;SET UP ERROR COUNT 51
(1) 000052 N=N+1
2138 010324 004537 011142 JSR R5,STAER ;REPORT ERROR SET
2139 010330 000000 HALT
2140 010332 012777 000016 170442 ACOB: MOV #16,@LPB ;SEND SHIFT OUT CODE
2141 010340 005777 170434 AC1: TST @LPS ;TEST FOR ERROR
2142 010344 100006 BPL AC2 ;BRANCH IF NO ERROR
2143 010346 $ERROR \N
(1) 010346 012737 000052 001052 ERR52: MOV #52, ERCOUNT ;SET UP ERROR COUNT 52
(1) 000053 N=N+1
2144 010354 004537 011142 JSR %5,STAER ;REPORT ERROR SET
2145 010360 000000 HALT ;HALT ON ERROR
2146 010362 013777 001040 170412 AC2: MOV CHRGEN,@LPB ;LOAD BUFFER
2147 010370 005237 001036 INC CHRCNT ;INCREMENT CHAR COUNT
2148 010374 001424 BEQ AC4 ;BRANCH IF DONE LINE
2149 010376 005237 001040 INC CHRGEN ;NEXT CHAR
2150 010402 005237 001044 INC CYCCNT ;INCREMENT GROUP COUNT
2151 010406 001410 BEQ AC3 ;BRANCH IF DONE GROUP
2152 010410 023737 001060 001040 CMP LEGCHR,CHRGEN ;LEGAL CHAR?
2153 010416 001350 BNE AC1 ;BRANCH AND CONTINUE IF LEGAL CHAR
2154 010420 163737 001062 001040 SUB NUMCHR,CHRGEN ;MAKE LEGAL
2155 010426 000744 BR AC1 ;CONTINUE
2156 010430 013737 001054 001040 AC3: MOV STRCHR,CHRGEN ;GET FIRST CHAR IN GROUP
2157 010436 012737 177757 001044 MOV #-17, CYCCNT ;RESET CYCLE COUNT
2158 010444 000735 BR AC1 ;CONTINUE
2159 010446 012777 000012 170326 AC4: MOV #12,@LPB ;SEND LF
2160 010454 $WAIT
(1) 010454 105777 170320 TSTB @LPS ;TEST READY
(1) 010460 100375 BPL .-4 ;WAIT FOR READY

```

2161	010462	005337	001042			DEC	LINCNT	;DECREMENT LINE COUNT
2162	010466	002413				BLT	AC6	;EXIT TEST IF DONE
2163	010470	162737	000004	001054		SUB	#4,STRCHR	;SKIP 4 LINES ON DRUM, FIND START CHAR
2164	010476	022737	000040	001054		CMP	#40,STRCHR	;START CHAR A LEGAL CHAR?
2165	010504	003701				BLE	ACOA	;CONTINUE IF LEGAL START CHAR
2166	010506	063737	001062	001054		ADD	NUMCHR,STRCHR	;MAKE LEGAL AND CONTINUE
2167	010514	000675				BR	ACOA	;CONTINUE
2168	010516	032777	010000	170260	AC6:	BIT	#BIT12,@SWR	;LOOP ON TEST?
2169	010524	001220				BNE	ACPRT	;LOOP
2170								
2171								
2172								
2173								
2174	010526	032777	040000	170250	AC7:	BIT	#BIT14,@SWR	;EVFU AVAILABLE?
2175	010534	001402				BEQ	HAMX	;NO, RECYCLE PRINTING TESTS
2176	010536	000137	014122			JMP	EVFU	;YES, DO EVFU PRINTING TESTS
2177	010542	032777	002000	170234	HAMX:	BIT	#BIT10,@SWR	;PLOT TESTS DESIRED?
2178	010550	001402				BEQ	HAMX1	;BR=NO
2179	010552	000137	016222			JMP	PLPRT	;GO DO PLOT TESTS
2180	010556				HAMX1:			
2181	010556	013700	000042			MOV	@#42,R0	
2182	010562	001405				BEQ	DOAGN	
2183	010564	000005				RESET		
2184	010566				LOGICAL:			
2185	010566	004710				JSR	PC,(R0)	
2186	010570	000240				NOP		
2187	010572	000240				NOP		
2188	010574	000240				NOP		
2189	010576				DOAGN:			
2190	010576	000137	003310			JMP	TEST2	;RESTART

2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243

010602 005777 170172
010606 100403
010610 105777 170164
010614 100403
010616 062705 000002
010622 000205
010624 012777 000014 170150
010632 105777 170142
010636 100003
010640 062705 000002
010644 000205
010646
(1) 010646 105777 170126
(1) 010652 100375
010654 000205

;MISC. ROUTINES

;ROUTINE TO INITIALIZE PRINTER
;ENTER FROM JSR %5, PRTINT

PRTINT: TST @LPS ;TEST FOR ERROR
BMI PRTINO ;BRANCH IF ERROR
TSTB @LPS ;TEST FOR READY
BMI RDYOK ;READY SET OK
PRTINO: ADD #2,%5 ;SET UP FOR ERROR REPORT
RTS %5 ;REPORT READY NOT SET
RDYOK: MOV #14,@LPB ;ISSUE FORM FEED
TSTB @LPS ;TEST FOR READY NOT SET
BPL NTRDY ;READY NOT SET OK
ADD #2,%5 ;SET UP FOR REPORT
RTS %5 ;EXIT AND REPORT
NTRDY: \$WAIT
TSTB @LPS ;TEST READY
BPL -4 ;WAIT FOR READY
RTS %5 ;READY SET EXIT

;ROUTINE TO OUTPUT ASCII MESSAGES ON THE LINE PRINTER

PRNNT: MOV #MES14,PRTMSG ;PRINT TEST NUMBER
TST @LPS ;TEST FOR ERROR
BPL RINT ;BRANCH IF OK
\$ERROR \N
ERR53: MOV #53, ERRCOUNT ;SET UP ERROR COUNT 53
N=N+1
JSR %5,STAER ;REPORT ERROR SET
HALT ;HALT ON ERROR
RINT: MOV LPS,TPS ;SET VECTORS -
MOV LPB,TPB ;TO PRINT ON LINE PRINTER
EMT +0 ;PRINT
PRTMSG: MES14 ;MESSAGE
TYPINT: MOV #177564,TPS ;RESET VECTORS
MOV #177566,TPB ;FOR TTY
RTS %4 ;RETURN

;SUBROUTINE TO OUTPUT ASCII MESSAGES ON TELETYPE PRINTER

TYP: MOV @%6,%0 ;GET ADDR. THAT CONTAINS MESS.
ADD #2,@%6 ;SET UP EXIT
MOV @%0,%0 ;ADDRESS OF MESSAGE IN RO
TYP A: MOV B (0)+,TYPDAT ;GET CHARACTER
BNE TYP C ;BRANCH IF NOT DONE
RTI ;EXIT
TYP C: CMPB #45,TYPDAT ;CHECK FOR '%'
BEQ TYP F ;BRANCH IF '%'
CMPB #43,TYPDAT ;CHECK FOR '#'

```

2244 011002 001417          BEQ      TYPG      ;BRANCH IF '#'
2245 011004 004737 011012    JSR      %7,TYPD   ;TYPE CHARACTER IN TYPDAT
2246 011010 000761          BR       TYPA      ;NEXT CHAR IN MESSAGE
2247 011012 113777 011056 167772 TYPD:  MOVB     TYPDAT,@TPB ;OUTPUT CHARACTER TO PRINTER
2248 011020 105777 167772    TYPD0: TSTB     @TPS
2249 011024 100375          BPL     -4
2250 011026 000207          RTS     %7        ;CHAR. TYPED EXIT
2251 011030 112737 000012 011056 TYPF:  MOVB     #12,TYPDAT ;OUTPUT LF
2252 011036 004737 011012    JSR     %7,TYPD   ;GO TYPE CHAR.
2253 011042 112737 000015 011056 TYPG:  MOVB     #15,TYPDAT ;OUTPUT CR
2254 011050 004737 011012    JSR     %7,TYPD   ;GO TYPE CHAR.
2255 011054 000737          BR      TYPA
2256 011056 000000          TYPDAT: 0
2257
2258
2259          ;ROUTINE TO CONVERT OCTAL TO ASCII
2260
2261          ;ENTER ROUTINE AS FOLLOWS
2262          ;      JSR      %5,CONV
2263          ;XXXXXXXX=ADDRESS OF NUMBER TO BE CONVERTED
2264          ;XXXXXXXX=ADDRESS OF ASCII MESSAGE
2265          ;XXXXXXXX=NUMBER OF OCTAL NO.'S TO BE CONVERTED
2266
2267 011060 013537 011140    CONV:  MOV      @(%5)+,ACNVX ;ADDRS OF NO. TO BE CONVERTED
2268 011064 012501          MOV      (%5)+,%1      ;ADDRESS OF MESSAGE
2269 011066 012502          MOV      (%5)+,%2      ;NUMBER OF ASCII CHARACTERS
2270 011070 060201          ADD      %2,%1        ;FIRST CHAR ADDRESS
2271 011072 013703 011140    ACVN:  MOV      ACNVX,%3   ;STORE NUMBER
2272 011076 042703 177770    BIC     #177770,%3     ;ISOLATE LEAST SIGNIFICANT BIT
2273 011102 062703 000060    ADD      #60,%3       ;SET UP ASCII CHARACTER
2274 011106 110341          MOVB    %3,-(1)       ;STORE CHARACTER
2275 011110 000241          CLC
2276 011112 006037 011140    ROR     ACNVX         ;GET NEXT SIGNIFICANT BIT ...
2277 011116 000241          CLC
2278 011120 006037 011140    ROR     ACNVX
2279 011124 000241          CLC
2280 011126 006037 011140    ROR     ACNVX
2281 011132 005302          DEC     %2            ;-1 FROM ASCII CHAR. CNT
2282 011134 001356          BNE    ACVN          ;CONVERT NEXT CHARACTER
2283 011136 000205          RTS     %5            ;EXIT! CONVERSION DONE
2284
2285 011140 000000          ACNVX:  0            ;WORK REGISTER
2286
2287          ;ROUTINE TO REPORT ERROR COUNT
2288
2289 011142 004537 011060    STAER: JSR      %5,CONV   ;CONVERT OCTAL TO ASCII
2290 011146 001052          ERCOUNT
2291 011150 011570          HED1
2292 011152 000003          3
2293 011154 104000          EMT     +0           ;TYPE ERROR MESSAGE
2294 011156 011570          HED1
2295 011160 005777 167620    TST     @SWR         ;TEST FOR HALT ON ERROR
2296 011164 100401          BMI     .+4          ;BRANCH IF NO HALT WANTED
2297 011166 000000          HALT
2298 011170 000205          RTS     %5            ;RETURN
2299

```



```

2300
2301      :KEYBOARD INTERRUPT ROUTINE
2302      :FOR ACCESS TO THE S/W SWITCH REGISTER
2303
2304
2305 011172 010046      TKINT: MOV    %0,-(SP)      ;SAVE REGISTERS
2306 011174 010146      MOV    %1,-(SP)      ;
2307 011176 010246      MOV    %2,-(SP)      ;
2308 011200 010346      MOV    %3,-(SP)      ;
2309 011202 010446      MOV    %4,-(SP)      ;
2310 011204 010546      MOV    %5,-(SP)      ;
2311 011206 005737 001072  TST    SET          ;INITIAL SWR ENTRY ?
2312 011212 001130      BNE    TYP5WR        ;YES-PRINT HEADER
2313 011214 005737 001070  TST    SIGNAL        ;PREVIOUS CONTROL-G INPUT ?
2314 011220 001477      BEQ    CNTRLG        ;YES-CONTINUE
2315 011222 017737 167566 001074  MOV    @TKB,CHAR     ;GET INPUT CHARACTER
2316 011230 042737 177600 001074  BIC    #177600,CHAR  ;STRIP OFF PARITY BIT
2317 011236 022737 000015 001074  CMP    #15,CHAR      ;CARRIAGE RETURN ?
2318 011244 001456      BEQ    DGTS          ;YES-CONTINUE
2319 011246 022737 000025 001074  CMP    #25,CHAR     ;CONTROL-U INPUT ?
2320 011254 001530      BEQ    TK4           ;YES-CONTINUE
2321 011256 023727 001074 000060  CMP    CHAR,#60     ;LEGAL CHECK: LESS THAN 60 ?
2322 011264 100001      BPL    TK1           ;NO-CONTINUE
2323 011266 000466      BR     WT3           ;YES-PRINT '?'
2324 011270 022737 000067 001074  TK1:  CMP    #67,CHAR  ;LEGAL CHECK: GREATER THAN 67 ?
2325 011276 100001      BPL    TK2           ;NO-CONTINUE
2326 011300 000461      BR     WT3           ;YES-PRINT '?'
2327 011302 005237 001066      TK2:  INC    DIGITS   ;NEXT DIGIT OF SWR INPUT
2328 011306 022737 000006 001066  CMP    #6,DIGITS   ;MORE THAN SIX DIGITS ?
2329 011314 100453      BMI    WT3           ;YES-PRINT '?'
2330 011316 105777 167474      WT2:  TSTB   @TPS     ;TTY PRINTER READY ?
2331 011322 100375      BPL    WT2           ;NO-WAIT
2332 011324 013777 001074 167460  MOV    CHAR,@TPB   ;PRINT CHARACTER
2333 011332 162737 000060 001074  SUB    #60,CHAR    ;CONVERT TO OCTAL
2334 011340 022737 000001 001066  CMP    #1,DIGITS   ;FIRST DIGIT ?
2335 011346 001411      BEQ    TK5           ;YES-CONTINUE
2336 011350 000241      CLC                    ;ROTATE LEFT THREE
2337 011352 006137 001076      ROL    OCT          ;TIMES
2338 011356 000241      CLC                    ;THIS WILL SHIFT
2339 011360 006137 001076      ROL    OCT          ;SWR VALUE ONE
2340 011364 000241      CLC                    ;PLACE LEFT
2341 011366 006137 001076      ROL    OCT          ;OCTAL.
2342 011372 063737 001074 001076  TK5:  ADD    CHAR,OCT   ;NEW VALUE OF SWR
2343 011400 000464      BR     TK6           ;RETURN FROM INTERRUPT
2344 011402 005737 001066      DGTS: TST    DIGITS   ;SWR VALUE CHANGED ?
2345 011406 001451      BEQ    TK3           ;NO-RETURN,NO CHANCE TO SWR
2346 011410 013777 001076 167366  MOV    OCT,@SWR    ;YES-ENTER NEW SWR VALUE
2347 011416 000445      BR     TK3           ;RETURN FROM INTERRUPT
2348 011420 017737 167370 001074  CNTRLG: MOV   @TKB,CHAR ;GET CHARACTER
2349 011426 042737 177600 001074  BIC    #177600,CHAR ;STRIP OFF PARITY BIT
2350 011434 022737 000007 001074  CMP    #7,CHAR     ;CONTROL-G INPUT ?
2351 011442 001414      BEQ    TYP5WR        ;YES-PRINT HEADER
2352 011444 105777 167346      WT3:  TSTB   @TPS     ;TTY PRINTER READY ?
2353 011450 100375      BPL    WT3           ;NO-WAIT
2354 011452 013777 001074 167332  MOV    CHAR,@TPB   ;PRINT CHARACTER
2355 011460 104000      EMT    +0          ;PRINT '?'

```

```

2356 011462 013754 MES22
2357 011464 005737 001070 TST SIGNAL ;BAD VALUE ?
2358 011470 001001 BNE TYP5WR ;YES-PRINT HEADER
2359 011472 000427 BR TK6 ;RETURN FROM INTERRUPT
2360 011474 012737 000001 001070 TYP5WR: MOV #1,SIGNAL ;SET FLAG: CONTROL-G ENTERED
2361 011502 104000 EMT +0 ;PRINT HEADER
2362 011504 013760 MES23
2363 011506 004537 011060 JSR %5,CONV ;CONVERT SWR VALUE TO ASCII
2364 011512 000176 176
2365 011514 014010 MES25
2366 011516 000006 6 ;PRINT SWR VALUE
2367 011520 104000 EMT +0 ;PRINT SWR VALUE
2368 011522 014010 MES25
2369 011524 104000 EMT +0 ;PRINT HEADER
2370 011526 013771 MES24
2371 011530 000404 BR TK7 ;RETURN FROM INTERRUPT
2372 011532 005037 001070 TK3: CLR SIGNAL ;CLEAR CONTROL-G FLAG
2373 011536 104000 TK4: EMT +0 ;PRINT LINE FEED AND CARRIAGE RETURN
2374 011540 013752 MES21
2375 011542 005037 0C1066 TK7: CLR DIGITS ;CLEAR DIGIT COUNT
2376 011546 005037 001076 CLR OCT ;CLEAR SWR INPUT
2377 011552 012605 TK6: MOV (SP)+,%5 ;RESTORE REGISTERS
2378 011554 012604 MOV (SP)+,%4
2379 011556 012603 MOV (SP)+,%3
2380 011560 012602 MOV (SP)+,%2
2381 011562 012601 MOV (SP)+,%1
2382 011564 012600 MOV (SP)+,%0
2383 011566 000002 RTI ; RETURN FROM INTERRUPT
2384
2385
2389
2390 011570 020040 020040 051105 HED1: .ASCIZ / ERROR COUNT%/
2391 011611 105 051122 051117 MESA: .ASCIZ /ERROR SET OK - CLEAR & TURN ON LINE%/
2392 011656 051105 047522 020122 MESB: .ASCIZ /ERROR SET OK - CLEAR AND TRY NEXT CHANNEL%/
2393 011731 045 051120 047111 MESC: .ASCIZ /%PRINT SPEED CHECK USING MANUAL TIMING%/
2394 012000 050045 052125 051440 .ASCIZ /%PUT SWITCH 0 UP TO START TIMING%/
2395 012041 045 052520 020124 .ASCIZ /%PUT SWITCH 0 DOWN AT END OF 1 MINUTE%/
2396 012110 051445 040524 052122 MESDD: .ASCIZ /%STARTING EVFU PRINTING TESTS%/
2397 012147 045 054114 030531 MES1: .ASCIZ \%LXY11/21-LXV11 LINE PRINTER TEST%\
2398 012212 042522 052123 051101 MES2: .ASCIZ /RESTART ADDRESS 600%/
2399 012237 120 053517 051105 MES3: .ASCIZ /POWER ON - TURN ON LINE%/
2400 012270 047117 046040 047111 MES4: .ASCIZ /ON LINE OK - TRY TORN PAPER SWITCH%/
2401 012334 042522 042101 020131 MES5: .ASCIZ /READY SET OK - TRY FORMS THICKNESS LEVER SWITCH%/
2402 012415 105 051122 051117 MES6: .ASCIZ /ERROR SET OK - TURN ON LINE%/
2403 .EVEN
2404 012452 042522 MES7A: .ASCIZ /RE/
2405 012454 042523 020124 047524 MES7: .ASCIZ /SET TOP OF FORM SWITCH TO /
2406 012510 020040 020040 044440 MES8: .ASCIZ / INCHES%/
2407 012526 .EVEN
2408 012526 026455 026455 026455 MES9: .ASCIZ /----- THIS LINE SHOULD BE /
2409 012623 040 020040 020040 MES10: .ASCIZ / INCHES FROM THE LAST LINE -----/
2410 012734 005012 MES11A: .ASCIZ <12><12>
2411 012736 050045 044522 052116 MES11: .ASCIZ /%PRINT SPEED IS APPROXIMATELY /
2412 012775 040 020040 020040 MES12: .ASCIZ / LINES PER MINUTE%/
2413 013024 026455 026455 026455 MES13: .ASCIZ /-----/
2414 013106 026455 026455 026455 .ASCIZ /-----/

```

```
2415 013170 026455 026455 026455 .ASCIZ /-----#/  
2416 .EVEN  
2417 013232 005012 042524 052123 MES14: .ASCII <12><12>/TEST NUMBER /  
2418 013250 020040 005012 000012 MES15: .ASCIZ / /<12><12><12>  
2419 .EVEN  
2420 013256 026455 026455 026455 MES16: .ASCII /-----START-LINE/  
2421 013364 026455 026455 026455 .ASCIZ /-----/  
2422 013464 .EVEN  
2423 013464 026455 026455 MES18: .ASCII /-----THERE SHOULD BE /  
2424 013556 020040 020040 020040 MES19: .ASCIZ / BLANK LINES BEFORE THIS LINE -----  
2425 .EVEN  
2426 013672 052040 051505 044524 MES20: .ASCII / TESTING CHANNEL SLEWING USING CHANNEL NO. /  
2427 013746 020040 000 MES20A: .ASCIZ / /  
2428 013752 .EVEN  
2429 013752 000045 MES21: .ASCIZ /%/  
2430 013754 037440 000045 MES22: .ASCIZ / ?%/  
2431 013760 051445 051127 036440 MES23: .ASCIZ /%SWR = /  
2432 013771 040 020040 042516 MES24: .ASCIZ / NEW SWR = /  
2433 014010 020040 020040 020040 MES25: .ASCIZ / /  
2434 014017 120 047514 020124 MES26: .ASCIZ /PLOT TEST/  
2435 014031 105 043126 020125 MES27: .ASCIZ %EVFU SLEW TEST%  
2436 .EVEN  
2437 014050 030504 TNDV1: .ASCII /D1/ ;TEST NUMBERS FOR EVFU TESTS  
2438 014052 031104 TNDV2: .ASCII /D2/  
2439 014054 031504 TNDV3: .ASCII /D3/  
2440 014056 020060 TNO0: .ASCII /0 /  
2441 014060 020061 TNO1: .ASCII /1 /  
2442 014062 020062 TNO2: .ASCII /2 /  
2443 014064 020063 TNO3: .ASCII /3 /  
2444 014066 020064 TNO4: .ASCII /4 /  
2445 014070 020065 TNO5: .ASCII /5 /  
2446 014072 020066 TNO6: .ASCII /6 /  
2447 014074 020067 TNO7: .ASCII /7 /  
2448 014076 020070 TNO10: .ASCII /8 /  
2449 014100 020071 TNO11: .ASCII /9 /  
2450 014102 030061 TNO12: .ASCII /10/  
2451 014104 030461 TNO13: .ASCII /11/  
2452 014106 031061 TNO14: .ASCII /12/  
2453 014110 031461 TNO15: .ASCII /13/  
2454 014112 032061 TNO16: .ASCII /14/  
2455 014114 032461 TNO17: .ASCII /15/  
2456 014116 033061 TNO20: .ASCII /16/  
2457 014120 033461 TNO21: .ASCII /17/  
2458 .EVEN
```

2460
2461
2462
2463
2464
2465
2466
2467
(1)
(1)
(1)
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
(1)
(1)
2480
2481
2482
2483
2484
2485
2486
2487
(1)
(1)
2488
2489
2490
2491
2492
2493
2494
(1)
(1)
2495
2496
2497
(1)
(1)
2498
2499
2500
2501
2502
2503
2504

014122 000137 014660
014126 022737 000176 001004
014134 001003
014136 052777 000100 164654
014144 004437 010726
014150 013737 016050 013560
014156 104000
014160 012110
014162 012737 000220 014654
014170 012737 000221 014656
014176 013737 014050 013250
014204 004437 010656
014210 012737 014606 001040
014216 005777 164556
014222 100010
014224
(1) 014224 012737 000054 001052
(1) 000055
014232 004537 011142
014236 000000
014240 000137 014210
014244 017777 164570 164530
014252 062737 000002 001040
014260 005777 164554
014264 001405
014266
(1) 014266 105777 164506
(1) 014272 100375
014274 000137 014216
014300 012737 000002 001044
014306 012737 013256 010724
014314 004437 010706
014320 005777 164454
014324 100006
014326
(1) 014326 012737 000055 001052
(1) 000056
014334 004537 011142
014340 000000
014342
(1) 014342 105777 164432
(1) 014346 100375
014350 012737 013464 010724
014356 013737 014654 001040
014364 012737 014056 001054
014372 012737 000020 001036
014400 005777 164374
014404 100006
014406

```
;EVFU PRINTING TESTS IF EVFU IS AVAILABLE -- SET SWITCH 14

;TESTS D1 AND D2
;CHECK EVFU LINE COUNT SLEWING

EVFU:  JMP      DAV2      ;SKIP OVER TESTS. MAY USE THESE IN THE FUTURE
       $ENABLE
       CMP      #176,SWR      ;S/W SWR ?
       BNE      .+10         ;NO- CONTINUE
       BIS      #100,@TKS     ;ENABLE KEYBOARD INTERRUPT
       JSR      %4,TYPINT     ;INITIALIZE
       MOV      SPSP,MES19+2
       EMT      +0           ;TYPE MESSAGE
       MESDD
       MOV      #220,DAVI1    ;STARTING EVFU TESTS
       MOV      #221,DAVI2    ;SET EVFU INSTRUCTIONS
       MOV      TNDV1,MES15   ;SET TEST NUMBER FOR MESSAGE
       JSR      %4,PRNNT      ;PRINT TEST NUMBER
       DAV0:  MOV      #DAVTAB,CHRGEN ;SET TABLE POINTER
       DAV0U: TST      @LPS     ;TEST FOR ERROR
       BPL      DAV1         ;BRANCH IF NO ERROR
       $ERROR  \N
       (1)  MOV      #54,   ERCOUNT ;SET UP ERROR COUNT 54
       (1)  N=N+1
       JSR      %5,STAER     ;REPORT ERROR SET
       HALT
       JMP      DAV0         ;HALT ON ERROR
       DAV1: MOV      @CHRGEN,@LPB ;RESTART TEST
       ADD      #2,CHRGEN    ;LOAD EVFU
       TST      @CHRGEN     ;INCREMENT TABLE POINTER
       BEQ      D5          ;TEST IF DONE LOAD
       $WAIT
       TSTB    @LPS         ;CONTINUE IF DONE
       BPL      .-4         ;WAIT
       JMP      DAV00       ;TEST READY
       D5:   MOV      #2,CYCCNT ;TEST READY
       D0:   MOV      #MES16,PRMSG ;WAIT FOR READY
       JSR      %4,RINT     ;SET CYCLE COUNT
       TST      @LPS         ;SET MESSAGE ADDRESS
       BPL      D1          ;PRINT MESSAGE
       $ERROR  \N          ;TEST FOR ERROR
       (1)  MOV      #55,   ERCOUNT ;CONTINUE IF NO ERROR
       (1)  N=N+1           ;SET UP ERROR COUNT 55
       JSR      %5,STAER     ;REPORT ERROR SET
       HALT
       D1:   $WAIT
       TSTB    @LPS         ;HALT ON ERROR
       BPL      .-4         ;TEST READY
       MOV      #MES18,PRMSG ;WAIT FOR READY
       MOV      DAVI1,CHRGEN ;SET MESSAGE ADDRESS
       MOV      #TNOO,STRCHR ;FIRST EVFU INSTRUCTION
       MOV      #16.,CHRCNT ;SET TABLE POINTER
       D2:   TST      @LPS     ;SET TABLE COUNT
       BPL      D3          ;TEST FOR ERROR
       $ERROR  \N          ;CONTINUE IF NO ERRORS
```

```
(1) 014406 012737 000056 001052 ERR56: MOV #56, ERCCOUNT ;SET UP ERROR COUNT 56
(1) 000057 N=N+1
2505 014414 004437 011142 JSR %4,STAER ;REPORT ERROR SET
2506 014420 000000 HALT ;HALT ON ERROR
2507 014422 013777 001040 164352 D3: MOV CHRGEN,@LPB ;SEND EVFU INSTR.
2508 014430 $WAIT ;WAIT
(1) 014430 105777 164344 TSTB @LPS ;TEST READY
(1) 014434 100375 BPL .-4 ;WAIT FOR READY
2509 014436 017737 164412 013556 MOV @STRCHR,MES19 ;SET PRINTER MESSAGE
2510 014444 004437 010706 JSR %4,RINT ;PRINT MESSAGE
2511 014450 022737 000020 001036 CMP #16.,CHRCNT ;CHECK IF JUST DONE 0 SLEW
2512 014456 001003 BNE D31 ;BR IF NOT
2513 014460 062737 000002 001054 D31: ADD #2,STRCHR ;CORRECT FOR AUTO LF
2514 014466 005337 001036 DEC CHRCNT ;DEC TABLE COUNT
2515 014472 001407 BEQ D4 ;EXIT TEST IF DONE
2516 014474 005237 001040 INC CHRGEN ;NEXT EVFU INSTR.
2517 014500 062737 000002 001054 ADD #2,STRCHR ;INC TABLE POINTER
2518 014506 000137 014400 JMP D2 ;CONTINUE
2519 014512 005337 001044 D4: DEC CYCCNT ;DEC CYCLE COUNT
2520 014516 001415 BEQ DEX0 ;EXIT IF DONE
2521 014520 062737 000140 014654 ADD #140,DAVI1 ;CHANGE EVFU INSTR.
2522 014526 062737 000140 014656 ADD #140,DAVI2 ;CHANGE EVFU INSTR.
2523 014534 013737 014052 013250 MOV TNDV2,MES15 ;SET TEST NUMBER FOR MESSAGE
2524 014542 004437 010656 JSR %4,PRNNT ;PRINT TEST NUMBER
2525 014546 000137 014306 JMP D0 ;RETEST LINE COUNT SLEWING
2526 014552 012737 000220 014654 DEX0: MOV #220,DAVI1 ;RESET EVFU INSTR.
2527 014560 012737 000221 014656 MOV #221,DAVI2 ;RESET EVFU INSTR.
2528 014566 032777 010000 164210 BIT #BIT12,@SWR ;LOOP ON TEST?
2529 014574 001002 BNE 1$ ;LOOP
2530 014576 000137 014660 JMP DAV2 ;NEXT TEST
2531 014602 000137 014122 1$: JMP EVFU ;LOOP
2532
2533
2534 014606 000036 DAVTAB: SL ;EVFU LOAD TABLE
2535 014610 000001 1
2536 014612 000002 2
2537 014614 000003 3
2538 014616 000004 4
2539 014620 000005 5
2540 014622 000006 6
2541 014624 000007 7
2542 014626 000010 10
2543 014630 000011 11
2544 014632 000012 12
2545 014634 000013 13
2546 014636 000014 14
2547 014640 000015 15
2548 014642 000016 16
2549 014644 000017 17
2550 014646 000020 20
2551 014650 000037 EL
2552 014652 000000 0
2553
2554
2555 014654 000220 DAVI1: 220
2556 014656 000221 DAVI2: 221
```

```
2557
2558           :TEST D3
2559           :CHECK EVFU CHANNEL SLEW COMMANDS
2560
2561 014660      DAV2: $ENABLE
(1) 014660 022737 000176 001004      CMP #176,SWR           ;S/W SWR ?
(1) 014666 001003                    BNE .+10              ;NO- CONTINUE
(1) 014670 052777 000100 164122      BIS #100,@TKS         ;ENABLE KEYBOARD INTERRUPT
2562 014676 004437 010726 164122      JSR %4,TYPINT        ;INITIALIZE
2563 014702 013737 016050 013560      MOV SPSP,MES19+2
2564 014710 012737 014031 010724      MOV #MES27,PRTMSG   ;POINT TO MESSAGE
2565 014716 004437 010706            JSR R4,RINT
2566 014722 012737 014056 015370      MOV #TNOO,MTABP     ;SET MESSAGE TABLE POINTER
2567 014730 012737 016012 015364      MOV #ITAB,ITABP     ;SET INSTRUCTION TABLE POINTER
2568 014736 017737 000422 001054      MOV @ITABP,STRCHR   ;SAT FIRST INSTRUCTION
2569 014744 012737 014062 015372      MOV #TNO2,HTABP     ;SET HEADER MESSAGE TABLE POINTER
2570 014752 012737 015374 015362      LOAD: MOV #DTAB,DTABP ;SET DATA TABLE POINTER
2571 014760 017737 000376 001040      MOV @DTABP,CHRGEN   ;SET FIRST DATA PAIR
2572 014766 005777 164006            TST @LPS             ;TEST FOR ERROR
2573 014772 100007                    BPL DL1              ;BRANCH IF NO ERROR
2574 014774
(1) 014774 012737 000057 001052      ERR57: MOV #57, ERCOUNT ;SET UP ERROR COUNT 57
(1) 000060
2575 015002 004537 011142            JSR %5,STAER        ;REPORT ERROR SET
2576 015006 000000                    HALT                 ;HALT ON ERROR
2577 015010 000760                    BR LOAD              ;RESTART LOAD
2578 015012 013777 001040 163762      DL1: MOV CHRGEN,@LPB  ;LOAD EVFU
2579 015020
(1) 015020 105777 163754            $WAIT                ;WAIT
(1) 015024 100375                    TSTB @LPS           ;TEST READY
2580 015026 005777 163746            BPL .-4              ;WAIT FOR READY
2581 015032 100010                    TST @LPS           ;TEST FOR ERROR
2582 015034
(1) 015034 012737 000060 001052      ERR60: MOV #60, ERCOUNT ;SET UP ERROR COUNT 60
(1) 000061
2583 015042 004537 011142            JSR %5,STAER        ;REPORT ERROR SET
2584 015046 000000                    HALT                 ;HALT ON ERROR
2585 015050 000137 014752            JMP LOAD              ;RESTART LOAD
2586 015054 062737 000002 015362      DL6: ADD #2,DTABP     ;INC DATA TABLE POINTER
2587 015062 017737 000274 001040      MOV @DTABP,CHRGEN   ;SET NEXT DATA PAIR
2588 015070 022737 077777 001040      CMP #77777,CHRGEN  ;DONE LOAD?
2589 015076 001345                    BNE DL1
2590
2591           ;START OF CHANNEL SLEW TESTS
2592
2593 015100 012777 000020 163674      DL8: MOV #20,@LPB    ;SEND CHAN. 1 TO INIT EVFU
2594 015106
(1) 015106 105777 163666            $WAIT
(1) 015112 100375                    TSTB @LPS           ;TEST READY
2595 015114
(1) 015114 017737 000252 013746      DL8A: MOV @HTABP,MES20A ;SET HEADER MSSG ADDRESS
2596 015114 012737 013672 010724      MOV #MES20,PRTMSG   ;SET HEADER MSG ADDRESS
2597 015122 004437 010706            JSR %4,RINT         ;PRINT HEADER MESSAGE
2598 015130 012737 000012 001056      DL9A: MOV #12,STRCNT  ;SET FOR 10 TRYS
2599 015134 013777 001054 163632      DL9: MOV STRCHR,@LPB ;SEND EVFU INSTRUCTION
2600 015142
2601 015150                    $WAIT                ;WAIT
```

```

(1) 015150 105777 163624      TSTB   @LPS           ;TEST READY
(1) 015154 100375              BPL     -4           ;WAIT FOR READY
2602 015156 005777 163616      TST    @LPS           ;TEST FOR ERROR
2603 015162 100010              BPL    DL10          ;BRANCH IF OK
2604 015164                      $ERROR  \N
(1) 015164 012737 000061 001052 ERR61: MOV    #61,   ERCOUNT      ;SET UP ERROR COUNT 61
(1) 015164 000062              N=N+1
2605 015172 004537 011142      JSR    %5,STAER      ;REPORT ERROR SET
2606 015176 000000              HALT                    ;HALT ON ERROR
2607 015200 000137 014752      JMP                    ;RELOAD EVFU
2608 015204 022737 000012 001056 DL10:  CMP    #12,STRCNT      ;FIRST TIME?
2609 015212 001404              BEQ    DL10A          ;BR=YES
2610 015214 013737 014106 013556   MOV    TNO14,MES19    ;SET MESSAGE
2611 015222 000403              BR     DL10B          ;BYPASS
2612 015224 017737 000140 013556 DL10A: MOV    @MTABP,MES19   ;SET MESSAGE
2613 015232 012737 013464 010724 DL10B: MOV    #MES18,PRTMSG ;SET MSG ADDRESS
2614 015240 004437 010706      JSR    %4,RINT        ;PRINT MESSAGE
2615 015244 005337 001056      DEC    STRCNT         ;DEC INSTR COUNT
2616 015250 001334              BNE    DL9            ;FINISH TESTING THIS CHANNEL
2617 015252 062737 000002 015370   ADD    #2,MTABP       ;INC MSG TABLE POINTER
2618 015260 062737 000002 015372   ADD    #2,HTABP       ;INC HEADER MSG TABLE POINTER
2619 015266 062737 000002 015364   ADD    #2,ITABP       ;INC INSTR TABLE POINTER
2620 015274 005777 000064      TST    @ITABP         ;DONE?
2621 015300 001404              BEQ    DL12           ;BR=YES
2622 015302 017737 000056 001054   MOV    @ITABP,STRCHR  ;GET INSTRUCTION
2623 015310 001273              BNE    DL8            ;CONTINUE IF NOT DONE TEST
2624 015312 012777 000036 163462 DL12:  MOV    #SL,@LPB       ;TURN OFF EVFU
2625 015320                      $WAIT
(1) 015320 105777 163454      TSTB   @LPS           ;TEST READY
(1) 015324 100375              BPL     -4           ;WAIT FOR READY
2626 015326 032777 010000 163450   BIT    #BIT12,@SWR    ;LOOP ON TEST?
2627 015334 001402              BEQ    DLEX           ;LOOP ON TEST
2628 015336 000137 014660      JMP    DAV2           ;DO PLOT TESTS?
2629 015342 032777 002000 163434 DLEX:  BIT    #BIT10,@SWR ;BR=NO
2630 015350 001402              BEQ    DLEX1          ;GO DO PLOT TESTS
2631 015352 000137 016222      JMP    PLPRT          ;RECYCLE PRINTING TESTS
2632 015356 000137 003310 DLEX1: JMP    TEST2
2633
2634 015362 000000      DTABP: 0              ;DATA TABLE POINTER
2635 015364 000000      ITABP: 0              ;INSTRUCTION TABLE POINTER
2636 015366 000000      ICTABP: 0             ;INSTR COUNT TABLE POINTER
2637 015370 000000      MTABP: 0              ;MESSAGE TABLE POINTER
2638 015372 000000      HTABP: 0              ;HEADER MESSAGE TABLE POINTER
2639
2640                      ;DATA TABLE FOR EVFU LOAD
2641
2642 015374 000036      DTAB:  SL              ;START LOAD
2643 015376 000020              20              ;WORD # 1
2644 015400 000021              21              ;WORD # 2
2645 015402 000022              22              ;WORD # 3
2646 015404 000023              23              ;WORD # 4
2647 015406 000024              24              ;WORD # 5
2648 015410 000025              25              ;WORD # 6
2649 015412 000026              26              ;WORD # 7
2650 015414 000027              27              ;WORD # 8
2651 015416 000030              30              ;WORD # 9

```

2652	015420	000031	31	:WORD #	10
2653	015422	000032	32	:WORD #	11
2654	015424	000033	33	:WORD #	12
2655	015426	000034	34	:WORD #	13
2656	015430	000035	35	:WORD #	14
2657	015432	000021	21	:WORD #	15
2658	015434	000022	22	:WORD #	16
2659	015436	000023	23	:WORD #	17
2660	015440	000024	24	:WORD #	18
2661	015442	000025	25	:WORD #	19
2662	015444	000026	26	:WORD #	20
2663	015446	000027	27	:WORD #	21
2664	015450	000030	30	:WORD #	22
2665	015452	000031	31	:WORD #	23
2666	015454	000032	32	:WORD #	24
2667	015456	000033	33	:WORD #	25
2668	015460	000034	34	:WORD #	26
2669	015462	000035	35	:WORD #	27
2670	015464	000021	21	:WORD #	28
2671	015466	000022	22	:WORD #	29
2672	015470	000023	23	:WORD #	30
2673	015472	000024	24	:WORD #	31
2674	015474	000025	25	:WORD #	32
2675	015476	000026	26	:WORD #	33
2676	015500	000027	27	:WORD #	34
2677	015502	000030	30	:WORD #	35
2678	015504	000031	31	:WORD #	36
2679	015506	000032	32	:WORD #	37
2680	015510	000033	33	:WORD #	38
2681	015512	000034	34	:WORD #	39
2682	015514	000035	35	:WORD #	40
2683	015516	000021	21	:WORD #	41
2684	015520	000022	22	:WORD #	42
2685	015522	000023	23	:WORD #	43
2686	015524	000024	24	:WORD #	44
2687	015526	000025	25	:WORD #	45
2688	015530	000026	26	:WORD #	46
2689	015532	000027	27	:WORD #	47
2690	015534	000030	30	:WORD #	48
2691	015536	000031	31	:WORD #	49
2692	015540	000032	32	:WORD #	50
2693	015542	000033	33	:WORD #	51
2694	015544	000034	34	:WORD #	52
2695	015546	000035	35	:WORD #	53
2696	015550	000021	21	:WORD #	54
2697	015552	000022	22	:WORD #	55
2698	015554	000023	23	:WORD #	56
2699	015556	000024	24	:WORD #	57
2700	015560	000025	25	:WORD #	58
2701	015562	000026	26	:WORD #	59
2702	015564	000027	27	:WORD #	60
2703	015566	000030	30	:WORD #	61
2704	015570	000031	31	:WORD #	62
2705	015572	000032	32	:WORD #	63
2706	015574	000033	33	:WORD #	64
2707	015576	000034	34	:WORD #	65

2708	015600	000035	35	:WORD #	66
2709	015602	000021	21	:WORD #	67
2710	015604	000022	22	:WORD #	68
2711	015606	000023	23	:WORD #	69
2712	015610	000024	24	:WORD #	70
2713	015612	000025	25	:WORD #	71
2714	015614	000026	26	:WORD #	72
2715	015616	000027	27	:WORD #	73
2716	015620	000030	30	:WORD #	74
2717	015622	000031	31	:WORD #	75
2718	015624	000032	32	:WORD #	76
2719	015626	000033	33	:WORD #	77
2720	015630	000034	34	:WORD #	78
2721	015632	000035	35	:WORD #	79
2722	015634	000021	21	:WORD #	80
2723	015636	000022	22	:WORD #	81
2724	015640	000023	23	:WORD #	82
2725	015642	000024	24	:WORD #	83
2726	015644	000025	25	:WORD #	84
2727	015646	000026	26	:WORD #	85
2728	015650	000027	27	:WORD #	86
2729	015652	000030	30	:WORD #	87
2730	015654	000031	31	:WORD #	88
2731	015656	000032	32	:WORD #	89
2732	015660	000033	33	:WORD #	90
2733	015662	000034	34	:WORD #	91
2734	015664	000035	35	:WORD #	92
2735	015666	000021	21	:WORD #	93
2736	015670	000022	22	:WORD #	94
2737	015672	000023	23	:WORD #	95
2738	015674	000024	24	:WORD #	96
2739	015676	000025	25	:WORD #	97
2740	015700	000026	26	:WORD #	98
2741	015702	000027	27	:WORD #	99
2742	015704	000030	30	:WORD #	100
2743	015706	000031	31	:WORD #	101
2744	015710	000032	32	:WORD #	102
2745	015712	000033	33	:WORD #	103
2746	015714	000034	34	:WORD #	104
2747	015716	000035	35	:WORD #	105
2748	015720	000021	21	:WORD #	106
2749	015722	000022	22	:WORD #	107
2750	015724	000023	23	:WORD #	108
2751	015726	000024	24	:WORD #	109
2752	015730	000025	25	:WORD #	110
2753	015732	000026	26	:WORD #	111
2754	015734	000027	27	:WORD #	112
2755	015736	000030	30	:WORD #	113
2756	015740	000031	31	:WORD #	114
2757	015742	000032	32	:WORD #	115
2758	015744	000033	33	:WORD #	116
2759	015746	000034	34	:WORD #	117
2760	015750	000035	35	:WORD #	118
2761	015752	000021	21	:WORD #	119
2762	015754	000022	22	:WORD #	120
2763	015756	000023	23	:WORD #	121

```

2764 015760 000024      24      :WORD # 122
2765 015762 000025      25      :WORD # 123
2766 015764 000026      26      :WORD # 124
2767 015766 000027      27      :WORD # 125
2768 015770 000030      30      :WORD # 126
2769 015772 000031      31      :WORD # 127
2770 015774 000032      32      :WORD # 128
2771 015776 000033      33      :WORD # 129
2772 016000 000034      34      :WORD # 130
2773 016002 000035      35      :WORD # 131
2774 016004 000035      35      :WORD # 132
2775 016006 000037      EL      :STOP LOAD!
2776 016010 077777      77777  :STOP!!
  
```

;INSTRUCTION TABLE FOR EVFU CHANNEL SLEW INSTRUCTIONS

```

2777
2778
2779
2780 016012 000021      21      :CHANNEL 2
2781 016014 000022      22      :CHANNEL 3
2782 016016 000023      23      :CHANNEL 4
2783 016020 000024      24      :CHANNEL 5
2784 016022 000025      25      :CHANNEL 6
2785 016024 000026      26      :CHANNEL 7
2786 016026 000027      27      :CHANNEL 8
2787 016030 000030      30      :CHANNEL 9
2788 016032 000031      31      :CHANNEL 10
2789 016034 000032      32      :CHANNEL 11
2790 016036 000033      33      :CHANNEL 12
2791 016040 000034      34      :CHANNEL 13
2792 016042 000035      35      :CHANNEL 14
2793 016044 000000      0       :END OF TABLE
2794
  
```

```

FS: .ASCII /3 /
SPSP: .ASCII / /
  
```

;SCOPE LOOP ROUTINE

;SET CHARACTER IN SWITCH REGISTER -0.

```

2800
2801
2802 016052 022737 000176 001004  SCOPE: $ENABLE
      (1) 016052 001003          CMP #176,SWR          ;S/W SWR ?
      (1) 016060 052777 000100 162730  BNE .+10             ;NO- CONTINUE
      (1) 016062 017737 162704 001050  BIS #100,@TKS       ;ENABLE KEYBOARD INTERRUPT
2803 016070 004437 010726          JSR %4,TYPINT
2804 016074 017737 162704 001050  MOV @SWR,SAVE      ;FETCH SWITCHES
2805 016102 012737 177574 001036  MOV #-132,CHRCNT  ;SET CHAR COUNT
2806 016110 042737 177400 001050  BIC #177400,SAVE  ;MASK CHARACTER
2807 016116          LDLPX: $WAIT
      (1) 016116 105777 162656          TSTB @LPS           ;TEST READY
      (1) 016122 100375          BPL .-4             ;WAIT FOR READY
2808 016124 005777 162650          TST @LPS           ;TEST FOR ERROR
  
```

```

2810 016130 100006          BPL      LPSCOPE          ;BRANCH IF NO ERROR
2811 016132          $ERROR      \N
(1) 016132 012737 000062 001052 ERR62: MOV      #62,   ERCOUNT          ;SET UP ERROR COUNT 62
(1)          000063          N=N+1
2812 016140 004537 011142          JSR      %5,STAER          ;REPORT ERROR SET
2813 016144 000000          HALT          ;HALT ON ERROR
2814 016146 013777 001050 162626 LPSCOPE:MOV     SAVE,@LPB          ;LOAD PRINTER BUFFER
2815 016154 032777 004000 162622          BIT      #BIT11,@SWR          ;SEND ONLY ONE CHAR?
2816 016162 001402          BEQ      LSCO              ;NO, BRANCH
2817 016164 000000          HALT          ;HALT - WAIT FOR OPERATOR
2818 016166 000731          BR          ;NEXT CHAR
2819 016170 000177 000024          LSCO:  JMP      @LOSCOP          ;SEND LF?
2820 016174 005237 001036          LSCA:  INC      CHRCNT          ;INCREMENT CHAR COUNT
2821 016200 001346          BNE      LDLPX             ;CONTINUE IF NOT DONE LINE
2822 016202 012777 000012 162572          MOV      #12,@LPB          ;SEND LF
2823 016210          $WAIT
(1) 016210 105777 162564          TSTB     @LPS              ;TEST READY
(1) 016214 100375          BPL      -4                ;WAIT FOR READY
2824 016216 000715          BR          SCOPE          ;CONTINUE
2825
2826
2827 016220 016174          LOSCOP: LSCA
2828
2829
2830
2831          ;PLOT TEST
2832          ;DRAWS 132 LINES WITH FLOATING SPACE THEN 132 LINES WITH FLOATING DOT
2833
2834 016222          PLPRT: $ENABLE
(1) 016222 022737 000176 001004          CMP      #176,SWR          ;S/W SWR ?
(1) 016230 001003          BNE      +10              ;NO- CONTINUE
(1) 016232 052777 000100 162560          BIS      #100,@TKS          ;ENABLE KEYBOARD INTERRUPT
2835 016240 012737 014017 010724          MOV      #MES26,PRTMSG          ;PRINT TEST HEADER
2836 016246 004437 010706          JSR      R4,RINT
2837 016252 012777 000012 162522          MOV      #12,@LPB          ;DO 1 BLANK LINE
2838 016260          $WAIT
(1) 016260 105777 162514          TSTB     @LPS              ;TEST READY
(1) 016264 100375          BPL      -4                ;WAIT FOR READY
2839 016266 012737 000240 016450 PL1:  MOV      #240,LGL2          ;SET SUBROUTINE FOR FLOATING SPACE
2840 016274 004737 016326          JSR      PC,LINGO          ;GO DO IT
2841 016300 012737 005101 016450 PL2:  MOV      #5101,LGL2          ;SET SUBROUTINE FOR FLOATING DOT
2842 016306 004737 016326          JSR      PC,LINGO          ;GO DO IT
2843 016312 032777 010000 162464          BIT      #BIT12,@SWR          ;LOOP ON TEST?
2844 016320 001362          BNE      PL1              ;BR=YES
2845 016322 000137 003310          JMP      TEST2            ;GO TO TEST 2
2846 016326 012737 000001 001042 LINGO:MOV     #1,LINCNT          ;SET LINCNT
2847 016334 012737 000001 016556          MOV      #1,DRKLIT
2848 016342 012737 000001 016560          MOV      #1,BYTPOS          ;SET BYTE POSITION
2849 016350 012737 000001 016562 LGL0:MOV     #1,DOTPOS          ;SET DOT POSITION
2850 016356 012701 000077          LGL1:  MOV      #77,R1
2851 016362 023737 001042 016560          CMP      LINCNT,BYTPOS
2852 016370 001027          BNE      LGL2
2853 016372 013703 016562          MOV      DOTPOS,R3
2854 016376 023703 016560          CMP      BYTPOS,R3
2855 016402 103422          BLO      LGL2
2856 016404 062703 000006          ADD      #6,R3

```

2857	016410	023703	016560		CMP	BYTPOS,R3	
2858	016414	103015			BCC	LGL2	
2859	016416	043701	016556		BIC	DRKLIT,R1	
2860	016422	005237	016560		INC	BYTPOS	
2861	016426	006337	016556		ASL	DRKLIT	
2862	016432	032737	000100	016556	BIT	#100,DRKLIT	
2863	016440	001403			BEQ	LGL2	
2864	016442	012737	000001	016556	MOV	#1,DRKLIT	
2865	016450	000240			NOP		;NOP IF DRAWING FLOAT SPACE 5101 IF NOT
2866	016452	042701	177700		BIC	#177700,R1	
2867	016456	052701	000100		BIS	#100,R1	
2868	016462	004737	016542		JSR	PC,PLOT	;GO PLOT
2869	016466	062737	000006	016562	ADD	#6,DOTPOS	
2870	016474	023727	016562	001431	CMP	DOTPOS,#793.	
2871	016502	103725			BLO	LGL1	
2872	016504	012701	000005		MOV	#5,R1	
2873	016510	004737	016542		JSR	PC,PLOT	
2874	016514	012701	000012		MOV	#12,R1	
2875	016520	004737	016542		JSR	PC,PLOT	
2876	016524	005237	001042		INC	LINCNT	
2877	016530	023727	001042	001431	CMP	LINCNT,#793.	
2878	016536	001304			BNE	LGL0	
2879	016540	000207			RTS	PC	
2880							
2881	016542	010177	162234		PLOT: MOV	R1,@LPB	;SEND CHARACTER
2882	016546	105777	162226		1\$: TSTB	@LPS	;WAIT FOR READY
2883	016552	100375			BPL	1\$	
2884	016554	000207			RTS	PC	
2885	016556	000000			DRKLIT: 0		
2886	016560	000000			BYTPOS: 0		
2887	016562	000000			DOTPOS: 0		
2888							
2889							
2890							
2891							
2892		000001			.END		

YM-Z049A-AA
CZLXAA.SRC

(DECSPEC-11-BDFAD-A-D)
12-MAR-81 10:43

MACY11 30A(1052) 12-MAR-81 11:24 PAGE 38-2
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0070

ERR23	002214	1312#			
ERR24	003340	1498#			
ERR25	003416	1507#			
ERR26	003642	1544#			
ERR27	003760	1564#			
ERR3	001340	1168#			
ERR30	004222	1602#			
ERR31	004334	1619#			
ERR32	004544	1658#			
ERR33	005010	1700#			
ERR34	005264	1743#			
ERR35	005526	1784#			
ERR36	005776	1828#			
ERR37	006224	1864#			
ERR4	001362	1173#			
ERR40	006464	1905#			
ERR41	006656	1938#			
ERR42	006730	1949#			
ERR43	007124	1978#			
ERR44	007332	2010#			
ERR45	007454	2030#			
ERR46	007664	2065#			
ERR47	010006	2085#			
ERR5	001412	1182#			
ERR50	010060	2094#			
ERR51	010316	2137#			
ERR52	010346	2143#			
ERR53	010672	2222#			
ERR54	014224	2479#			
ERR55	014326	2494#			
ERR56	014406	2504#			
ERR57	014774	2574#			
ERR6	001444	1197#			
ERR60	015034	2582#			
ERR61	015164	2604#			
ERR62	016132	2811#			
ERR7	001470	1206#			
EVFU	014122	2176	2466#	2531	
EXPRT	007216	984	1998#	2043	
EX0	007302	2005#	2039	2041	
EX00	007266	2003#			
EX00A	007252	2001#			
EX1	007324	2008#	2020	2022	2025
EX2	007346	2009	2013#		
EX3	007414	2018	2023#		
EX4	007432	2015	2026#		
EX6	007540	2036	2042#		
EX7	007470	2029	2033#		
FS	016046	2795#			
HAMALN	007046	983	1971#	1991	
HAMX	010542	2175	2177#		
HAMX1	010556	2178	2180#		
HAM1X	007110	1975#	1989		
HAM2	007116	1976#	1985		
HED1	011570	2291	2294	2390#	
HSPRT	005412	979	1772#	1810	

YM-Z049A-AA
CZLXAA.SRC

(DECSPEC-11-BDFAD-A-D)
12-MAR-81 10:43

MACY11 30A(1052) 12-MAR-81 11:24 PAGE 38-5
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0073

N = 000063	1026#	1143#	1148#	1159#	1168#	1173#	1182#	1197#	1206#	1216#	1221#	1228#	1241#
	1246#	1249#	1258#	1268#	1278#	1300#	1305#	1312#	1498#	1507#	1544#	1564#	1602#
	1619#	1658#	1700#	1743#	1784#	1828#	1864#	1905#	1938#	1949#	1978#	2010#	2030#
	2065#	2085#	2094#	2137#	2143#	2222#	2479#	2494#	2504#	2574#	2582#	2604#	2811#
NOP = 000240	1025#												
NTRDY 010646	2209	2212#											
NUMCHR 001062	1107#	1407*	1416	1595*	1611	1776*	1795	1807	1852*	1861	2002*	2021	2040
	2057*	2076	2106	2129*	2154	2166							
	1113#	2337*	2339*	2341*	2342*	2346	2376*						
OCT 001076	1108#												
OFFSET 001064	1648#	1675											
OVR 004462	1673	1683#											
OVREXT 004706	976	1644#	1684										
OVRPRT 004420	1649#	1671											
CVRO 004470	1655#	1662	1667										
OVR1 004530	1653	1656#	1680										
OVR2 004536	1657	1661#											
OVR3 004560	1654#	1663											
OVR4 004522	1666#												
OVR5 004576	1669#	1682											
OVR6 C04612	1665	1681#											
OVR8 004674	1410	1479#											
PATTB 003270	1021#	1366	1389	1390*									
PLKS 001024	2868	2873	2875	2881#									
PLOT 016542	958	2179	2631	2834#									
PLPRT 016222	2839#	2844											
PL1 016266	2841#												
PL2 016300	1501	1537	1593	1646	1693	1736	1774	1822	1848	1891	1973	2000	2055
PRNNT 010656	2127	2219#	2475	2524									
	1706	1720#											
PRTA 005130	977	1691#	1727										
PRTCTL 004716	1496	2201#											
PRTINT 010602	2202	2205#											
PRTINO 010616	1473*	2219*	2228#	2490*	2498*	2564*	2597*	2613*	2835*				
PRTMSG 010724	1695#	1719	1723	1725									
PRT0 004760	1696#												
PRT2 004766	1698#	1705	1709	1721									
PRT3 005002	1699	1703#											
PRT4 005024	1713	1722#											
PRT6 005142	1715	1724#											
PRT7 005154	1717	1726#											
PRT8 005166	1015#												
PSW 001010	1024#	1238*	1297*	1323*									
PTRPSW 001032	1023#	1237*	1296*	1322*									
PTRVEC 001030	1511	1516#											
RCHRSW 003460	2204	2207#											
RDYOK 010624	1474	2221	2225#	2491	2510	2565	2598	2614	2836				
RINT 010706	981	1846#	1878										
ROTATE 006066	1850#												
ROTO 006122	1853#	1876											
ROT1 006144	1854#	1871											
ROT2 006150	1856#	1869											
ROT3 006160	1860	1862#											
ROT4 006216	1863	1867#											
ROT5 006240	1929#	1966											
RTTR 006602	1930#	1964											
RT1 006610													

YM-Z049A-AA
CZLXAA.SRC

(DECSPEC-11-BDFAD-A-D)
12-MAR-81 10:43

MACY11 30A(1052) 12-MAR-81 11:24 PAGE 38-8
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0076

T2B0A	003626	1541#	1554	1557															
T2B1	003634	1542#	1555	1558															
T2B2	003732	1549	1559#																
T2B2B	003720	1552	1556#																
T2B3	003752	1562#																	
T2B4	004002	1568#	1570	1572	1574	1576													
T2B5	004056	1579#																	
T3A	003410	1505#	1522																
T5A	003476	1513	1520#	1527															
UEXPRT	007550	985	2053#	2110															
UEXO	007634	2060#	2105	2107															
UEX00	007620	2058#																	
UEX00A	007604	2056#																	
UEX1	007656	2063#	2075	2077	2080														
UEX10	010166	2109	2111#																
UEX2	007700	2064	2068#																
UEX3	007746	2073	2078#																
UEX4	007764	2070	2081#																
UEX6	010152	2102	2108#																
UEX7	C10022	2084	2088#																
UEX7A	010044	2091#	2098																
UEX8	010052	2092#																	
UEX9	010074	2093	2097#																
ULPRT	004114	975	1591#	1631															
ULO	004200	1598#	1629																
UL00	004164	1596#																	
UL00A	004150	1594#																	
UL1	004214	1600#	1610	1612															
UL2	004236	1601	1605#																
UL4	004276	1607	1613#																
UL6	004410	1627	1630#																
UL7	004320	1616#	1623																
UL8	004326	1617#																	
UL9	004350	1618	1622#																
WORK	001046	1101#	1212*	1226*															
WT	002374	1346#	1349																
T2	011316	2330#	2331																
T3	011444	2323	2326	2329	2352#	2353													
XHAM1	007140	1977	1981#																
XHAM1X	007150	1983#																	
XS1X	006012	1827	1831#																
.	= 016564	908#	913	916#	922#	926#	928#	932#	937#	945#	949#	955#	961#	970#					
		988#	993#	999#	1131	1370	1384	1395	1427	1494	1524	153	1560	1580					
		1591	1614	1625	1644	1669	1691	1711	1734	1753	1764#	1772	1801	1820					
		1835	1846	1873	1889	1919	1961	1971	1981	1982	1987	1998	2027	2034					
		2053	2082	2089	2100	2125	2160	2212	2249	2296	2407#	2422#	2428#	2467					
		2487	2497	2508	2561	2579	2594	2601	2625	2802	2807	2823	2834	2838					

YM-Z049A-AA
CZLXAA.SRC

(DECSPEC-11-BDFAD-A-D)
12-MAR-81 10:43

MACY11 30A(1052) 12-MAR-81 11:24 PAGE 39
CROSS REFERENCE TABLE -- MACRO NAMES

SEQ 0077

\$ENABL	1068#	1131	1370	1384	1395	1494	1535	1591	1644	1691	1734	1772	1820	1846	1889
	1971	1998	2053	2125	2467	2561	2802	2834							
\$ERROR	1033#	1143	1148	1159	1168	1173	1182	1197	1206	1216	1221	1228	1241	1246	1249
	1258	1268	1278	1300	1305	1312	1498	1507	1544	1564	1602	1619	1658	1700	1743
	1784	1828	1864	1905	1938	1949	1978	2010	2030	2065	2085	2094	2137	2143	2222
	2479	2494	2504	2574	2582	2604	2811								
\$PRINT	1043#	1501	1537	1593	1646	1693	1736	1774	1822	1848	1891	1973	2000	2055	2127
\$SETPS	1083#	1251	1260	1270	1280	1309	1321								
\$WAIT	1054#	1427	1524	1560	1580	1614	1625	1669	1711	1753	1801	1835	1873	1919	1961
	1981	1987	2027	2034	2082	2089	2100	2160	2212	2487	2497	2508	2579	2594	2601
	2625	2807	2823	2838											

. ABS. 016564 000

ERRORS DETECTED: 0

CZLXAA,CZLXAA,SEQ/CRF=CZLXAA.SRC
RUN-TIME: 11 24 4 SECONDS
RUN-TIME RATIO: 161/39=4.1
CORE USED: 9K (17 PAGES)