```
                        .REPT  0
```

000583

```
                        IDENTIFICATION
                        --------------

PRODUCT CODE:   MAINDEC-11-DZKLA-E-D

PRODUCT NAME:   KL11/DL11-A TELETYPE TESTS

DATE:           21-DECEMBER-1975

MAINTAINER:     DIAGNOSTIC GROUP

AUTHOR:         R. KOLLER
REVISED BY:     AL LOSCHAK              ;SUPPORT SOFTWARE SWITCH REGISTER

THIS MAINDEC OBSOLETES MAINDEC-11-DZKLAD
```

```
                                1.      ABSTRACT
                                        --------

                                        THIS MAINDEC CONSISTS OF A PACKAGE OF TEST PROGRAMS DESIGNED
                                        TO TEST AN ASR33, KSR33, ASR35, OR KSR35 TELETYPE WHEN ATTACHED
                                        TO A PDP11 SYSTEM THROUGH A KL11 OR DL11A TELETYPE CONTROL. ALL
                                        TESTS ARE INCLUDED IN A SINGLE OBJECT TAPE.  NOTE: THE
                                        FOLLOWING PROGRAMING FORMAT IS ILLEGAL AND IS NOT USED IN
                                        THIS PROGRAM - MESSAGE, FILLER, FILLER, RESET, AND ANOTHER
                                        MESSAGE IMMEDIATELY.

                                        THE AVAILABLE TEST PROGRAMS ARE LISTED HERE IN NUMERICAL ORDER:

                                        PRG0-COMBINED INPUT-OUTPUT LOGIC TESTS
                                        PRG1-READER TEST
                                        PRG2-PRINTER TEST
                                        PRG3-PUNCH TEST
                                        PRG4-KEYBOARD TEST
                                        PRG5-COMBINED READER-PUNCH-PRINTER TEST
                                        PRG6-READER EXERCISER-SPECIAL BINARY COUNT PATTERN
                                        PRG7-PRINTER EXERCISER
                                        PRG10-SPECIAL BINARY COUNT PATTERN TAPE GENERATOR
                                        PRG11-PUNCH CLOCK ADJUSTMENT ROUTINE
                                        PRG12-READER CLOCK ADJUSTMENT ROUTINE
                                        PRG13-MAINTENANCE MODE SINGLE CHARACTER DATA TEST.
                                        PRG14-MAINTENANCE MODE SPECIAL BINARY COUNT PATTERN TEST.

                                        PROGRAMS PRG0 THROUGH PRG5 ARE THE ACTUAL TELETYPE TESTS.
                                        PROGRAMS PRG6 THROUGH PRG14 ARE UTILITY AND MAINTENANCE ROUTINES.

                                2.      REQUIREMENTS
                                        ------------

                                2.1     EQUIPMENT
                                        ---------

                                        A.  PDP-11 SYSTEM. (4 K CORE).
                                        B.  ASR33, KSR33, ASR35, OR KSR35 TELETYPE WITH KL11 OR DL11A TTY CONTROL.

                                        THE TELETYPE MUST HAVE STANDARD PERIPHERAL ADDRESSES.
                                        REFER TO SECTION 7.3 IF THE TELETYPE DOES NOT HAVE STANDARD
                                        PERIPHERAL ADDRESSES.

                                2.2     STORAGE
                                        -------

                                        THIS PROGRAM USES LOCATION 00000 THROUGH 16000.

                                3.      LOADING PROCEDURE
                                        -----------------

                                        THIS PROGRAM'S OBJECT TAPE IS PUNCHED IN ABSOLUTE FORMAT.
                                        THE ABS LOADER IS USED TO LOAD THE PROGRAM.
```

```
                                4.      USE PROCEDURE
                                        ============

                                        THIS PROGRAM HAS BEEN MODIFIED TO RUN WITH OR WITHOUT
                                        A CONSOLE PROCESSOR.
                                        IF A CONSOLE MACHINE IS USED; THEN THE PROGRAM
                                        LOOKS AT THE HARDWARE SWITCH REGISTER.
                                        IF A CONSOLE-LESS MACHINE IS USED;THEN THE PROGRAM
                                        AUTOMATICALLY LOOKS AT THE CONTENTS OF LOCATION
                                        SOFTSR (176) AS A SWITCH REGISTER.

                                        IT'S THE RESPONSIBILITY OF THE OPERATOR TO SET UP
                                        THIS LOCATION PRIOR TO STARTING THE PROGRAM.

                                        IF IT HALTS THEN THE OPERATOR HAS TO SET UP THE
                                        SWITCH REGISTER AND RESTART AT THE NEXT LOCATION
                                        OF THE NORMAL HALT.
                                        IF A SWITCH REGISTER IS AVAILABLE ALL THAT THE
                                        OPERATOR HAS TO DO IS SET THE S.R. AND HIT CONTINUE.

                                4.1     TELETYPE IDENTIFICATION
                                        =======================

                                        BEFORE RUNNING ANY OF THE TEST PROGRAMS, DEPOSIT IN LOCATION
                                        001230 ONE OF THE FOLLOWING NUMBERS:

                                        000000 IF ASR33 IS ATTACHED, OR 000010 IF KSR33 IS ATTACHED,OR
                                        000001 IF ASR35 IS ATTACHED, OR 000011 IF KSR35 IS ATTACHED.

                                        TELETYPE IDENTIFICATION NEED BE DONE ONLY ONCE PER PROGRAM
                                        LOAD, UNLESS MORE THAN ONE TYPE OF TELETYPE IS ATTACHED TO SYSTEM.

                                4.2     PRG0 USE PROCEDURE (DESCRIPTION IN SECTION 8.1)
                                        ===============================================

                                        A.  IDENTIFY TELETYPE AS PER SECTION 4.1.
                                        B.  SET TELETYPE TO ON-LINE.
                                        C.  LOAD SPECIAL BINARY COUNT PATTERN TEST TAPE IN TAPE
                                            READER AND MAKE READER READY (DATA OVER PINS)
                                        D.  LOAD ADDRESS 000200
                                        E.  SET SR TO 000000.  PRESS START
                                        F.  THE PROGRAM STOPS AT COMMON HALT.
                                        G.  SET ANY DESIRED SR OPTIONS. NORMAL RUN IS WITH SR = 000000.

                                            THIS PROGRAM'S SR OPTIONS ARE: (EXPLAINED IN SECTION 7.2)

                                            SR15        HALT AT END OF ROUTINE
                                            SR14        SCOPE MODE
                                            SR11        INHIBIT ITERATION
                                            SR10        LOOP PROGRAM
                                            SR9         SELECT ROUTINE
                                            SR6 THROUGH SR0 = NUMBER OF ROUTINE TO BE SELECTED.

                                        H.  PRESS CONTINUE.  THE PROGRAM IS EXECUTED AND STOPS AT
```

```
                                        PROGRAM END HALT WHEN COMPLETED, PROVIDED NO ERRORS OCCUR.
                                    I.  REFER TO SECTION 6, IF ERRORS OCCUR.

                                    EXECUTION TIME: NORMAL ERROR FREE PASS ABOUT 4 MINUTES.
```

```
153
154
155                     4.3     PRG1 USE PROCEDURE (DESCRIPTION IN SECTION 8.2)
156                             ----------------------------------------------
157
158                             A.  IDENTIFY TELETYPE AS PER SECTION 4.1
159                             B.  SET TELETYPE TO ON-LINE
160                             C.  LOAD SPECIAL BINARY COUNT PATTERN TEST TAPE IN TAPE
161                                 READER AND MAKE READER READY.  DATA HOLES MUST BE OVER
162                                 THE READ PINS.
163                             D.  LOAD ADDRESS 000200.
164                             E.  SET SR TO 000001.  PRESS START
165                             F.  PROGRAM STOPS AT COMMON HALT.
166                             G.  SET ANY DESIRED SR OPTIONS. NORMAL RUN IS WITH SR = 000000.
167
168                                 THIS PROGRAM'S SR OPTIONS ARE: (EXPLAINED IN SECTION 7.2)
169
170                                 SR15        HALT AT END OF ROUTINE
171                                 SR11        INHIBIT ITERATION
172                                 SR10        LOOP PROGRAM
173                                 SR9         SELECT ROUTINE
174                                 SR6 THROUGH SR0 = NUMBER OF ROUTINE TO BE SELECTED.
175
176                             H.  PRESS CONTINUE.  READER IS TESTED.  PROGRAM HALTS AT
177                                 PROGRAM END HALT IF NO ERRORS OCCUR.
178                             I.  REFER TO SECTION 6. IF ERRORS OCCUR.
179
180                             EXECUTION TIME: NORMAL ERROR FREE PASS ABOUT 11 MINUTES.
181
182                     4.4     PRG2 USE PROCEDURE (DESCRIPTION IN SECTION 8.3)
183                             ----------------------------------------------
184
185                             A.  IDENTIFY TELETYPE AS PER SECTION 4.1
186                             B.  SET TELETYPE TO ON-LINE
187                             C.  LOAD ADDRESS 000200.
188                             D.  SET SR TO 000002.  PRESS START
189                             E.  PROGRAM STOPS AT COMMON HALT.
190                             F.  SET ANY DESIRED SR OPTIONS. NORMAL RUN IS WITH SR = 000000.
191
192                                 THIS PROGRAM'S SR OPTIONS ARE: (EXPLAINED IN SECTION 7.2)
193
194                                 SR15        HALT AT END OF ROUTINE
195                                 SR10        LOOP PROGRAM
196                                 SR9         SELECT ROUTINE
197                                 SR6 THROUGH SR0 = NUMBER OF ROUTINE TO BE SELECTED.
198
199                             H.  PRESS CONTINUE.  THE TELEPRINTER WILL BE EXERCISED AND
200                                 THE PROGRAM WILL STOP AT PROGRAM END HALT WHEN COMPLETED.
201                             I.  ERROR DETECTION IS BY VISUAL INSPECTION OF RESULTING PRINTOUT.
202
203                             EXECUTION TIME: NORMAL ERROR FREE PASS ABOUT 12 MINUTES.
204
```

```
                                4.5     PRG3 USE PROCEDURE (DESCRIPTION IN SECTION 8.4)
                                        ==============================================

                                        A.  IDENTIFY TELETYPE AS PER SECTION 4.1
                                        B.  WITH TELETYPE OFF-LINE, PUNCH FIVE INCHES OF BLANK
                                            LEADER.  SET TELETYPE TO ON-LINE.
                                        C.  LOAD BLANK LEADER IN PAPER TAPE READER AND MAKE
                                            READER READY.  PUNCH TO READER SLACK SHOULD NOT BE
                                            EXCESSIVE.
                                        D.  LOAD ADDRESS 000200.
                                        E.  SET SR TO 000003.  PRESS START
                                        F.  PROGRAM STOPS AT COMMON HALT.
                                        G.  SET ANY DESIRED SR OPTIONS. NORMAL RUN IS WITH SR = 000000.

                                            THIS PROGRAM'S SR OPTIONS ARE: (EXPLAINED IN SECTION 7.2)

                                            SR15        HALT AT END OF ROUTINE
                                            SR11        INHIBIT ITERATION
                                            SR10        LOOP PROGRAM
                                            SR9         SELECT ROUTINE
                                            SR6 THROUGH SR0 = NUMBER OF ROUTINE TO BE SELECTED.

                                        H.  PRESS CONTINUE.  THE PROGRAM WILL EXERCISE THE PUNCH
                                            AND VERIFY THE DATA PUNCHED.
                                        I.  IF ANY ERRORS OCCUR REFER TO SECTION 6. ERRORS.

                                        EXECUTION TIME: NORMAL ERROR FREE PASS ABOUT 40 MINUTES.

                                4.6     PRG4 USE PROCEDURE (DESCRIPTION IN SECTION 8.5)
                                        ==============================================

                                        A.  IDENTIFY TELETYPE AS PER SECTION 4.1.
                                        B.  SET TELETYPE ON-LINE.
                                        C.  LOAD ADDRESS 000200.
                                        D.  SET SR TO 000004.  PRESS START
                                        E.  THE PROGRAM TYPES "KEYBOARD TEST" AND STOPS AT COMMON HALT.
                                        F.  SET  ANY DESIRED SR OPTIONS. NORMAL RUN IS WITH SR = 000000.
                                            THIS PROGRAM'S SR OPTIONS ARE: (EXPLAINED IN SECTION 7.2)

                                            SR15        HALT AT END OF ROUTINE
                                            SR10        LOOP PROGRAM
                                            SR9         SELECT ROUTINE
                                            SR6 THROUGH SR0 = NUMBER OF ROUTINE TO BE SELECTED.

                                        G.  PRESS CONTINUE.  FOLLOW TYPED INSTRUCTIONS.  WHEN DONE
                                            PROGRAM STOPS AT PROGRAM END HALT.

                                        EXECUTION TIME: PROGRAM IS USER DEPENDENT.
```

```
                4.7     PRG5 USE PROCEDURE (DESCRIPTION IN SECTION 8.6)
                        ----------------------------------------------

                        A.  IDENTIFY TELETYPE AS PER SECTION 4.1.
                        B.  WITH TELETYPE OFF-LINE PUNCH FIVE INCHES OF BLANK LEADER.
                            SET TELETYPE TO ON-LINE.
                        C.  LOAD BLANK LEADER IN PAPER TAPE READER AND MAKE READER READY.
                            PUNCH TO READER SLACK SHOULD NOT BE EXCESSIVE.
                        D.  LOAD ADDRESS 000200.
                        E.  SET SR TO 000005.  PRESS START.
                        F.  PROGRAM STOPS AT COMMON HALT.
                        G.  SET ANY DESIRED SR OPTIONS. NORMAL RUN IS WITH SR = 000000.

                            THIS PROGRAM'S SR OPTIONS ARE: (EXPLAINED IN SECTION 7.2)
                            SR8         DISABLE STALL MODE AND RUN FULL SPEED
                        H.  PRESS CONTINUE. THE PRINTER, PUNCH, AND READER ARE EXERCISED IN
                            COMBINATION USING DATA CODES FROM 241 TO 337. THE PATTERN IS
                            ROTATED SO THAT EACH CHARACTER IS TYPED IN EVERY PRINT POSITION.
                        I.  REFER TO SECTION 6. ERRORS IF ERRORS OCCUR.

                        EXECUTION TIME: ONE 63 LINE PASS TAKES ABOUT 18 MINUTES.

                4.8     PRG6 USE PROCEDURE (DESCRIPTION IN SECTION 8.7)
                        ----------------------------------------------

                        A.  IDENTIFY TELETYPE AS PER SECTION 4.1
                        B.  SET TELETYPE TO ON-LINE
                        C.  LOAD SPECIAL BINARY COUNT PATTERN TEST TAPE IN TAPE READER
                            AND MAKE READER READY.  DATA HOLES MUST BE OVER READ PINS.
                        D.  LOAD ADDRESS 000200.
                        E.  SET SR TO 000006.  PRESS START
                        F.  THE PROGRAM EXERCISES THE READER CONTINUOUSLY.
                        G.  SR8  DISABLE STALL MODE AND RUN FULL SPEED OPTION MAY BE
                            USED AT ANY TIME. SECTION 7.2 DESCRIBES SR OPTIONS.
                        H.  REFER TO SECTION 6. ERRORS, IF ERRORS OCCUR.

                        EXECUTION TIME: CONTINUOUS RUNNING.

                4.9     PRG7 USE PROCEDURE (DESCRIPTION IN SECTION 8.8)
                        ----------------------------------------------

                        A.  IDENTIFY TELETYPE AS PER SECTION 4.1
                        B.  SET TELETYPE TO ON-LINE
                        C.  LOAD ADDRESS 000200
                        D.  SET SR TO 000007.  PRESS START
                        E.  THE PROGRAM TYPES "TYPE IN DATA"
                        F.  KEY IN ANY FIVE CHARACTERS TO BE TYPED.
                        G.  KEY IN EITHER A RUBOUT FOR FULL SPEED TYPING, OR ANY
                            OTHER CHARACTER FOR RANDOM STALLS BETWEEN CHARACTERS.
                        H.  THE PROGRAM TYPES CONTINUOUSLY LINES CONTAINING THE
                            FIVE CHARACTERS SPECIFIED, UNTIL SR15 IS SET TO A 1.  AT
                            THAT POINT THE PROGRAM GOES TO STEP E.
```

```
                                4.10    PRG10 USE PROCEDURE (DESCRIPTION IN SECTION 8.9)
                                        ================================================

                                        A.  IDENTIFY TELETYPE AS PER SECTION 4.1
                                        B.  SET TELETYPE TO ON-LINE
                                        C.  TURN ON PUNCH
                                        D.  LOAD ADDRESS 000200.
                                        E.  SET SR TO 000010.  PRESS START
                                        F.  THE PROGRAM CONTINUOSLY PUNCHES THE SPECIAL BINARY COUNT
                                            COUNT PATTERN UNTIL STOPPED BY USER.

                                        EXECUTION TIME: CONTINUOUS RUNNING.

                                4.11    PRG11 USE PROCEDURE (DESCRIPTION IN SECTION 8.10)
                                        ================================================

                                        PRG11 IS USED AS AN AID IN ADJUSTING THE TELETYPE PUNCH CLOCK,
                                        AND IN OBSERVING THE DATA BITS AS THEY ARE SHIFTED OUT OF THE
                                        PUNCH BUFFER. A SCOPE IS REQUIRED.

                                        TO ADJUST THE PUNCH CLOCK PROCEED AS FOLLOWS:

                                        A.  LOAD ADDRESS 000200
                                        B.  SET SR TO 000011. PRESS START.
                                        C.  PROGRAM STOPS AT COMMON HALT.
                                        D.  SET ANY DESIRED ASCII CODE IN LEFT HALF OF SR.
                                        E.  SET NUMBER OF MILLISECONDS TO DELAY BETWEEN PUNCH COMMANDS
                                            IN RIGHT HALF OF SR. THE NUMBER OF MILLISECONDS SELECTED
                                            SHOULD BE LONG ENOUGH FOR THE ENTIRE PUNCH OPERATION TO
                                            COMPLETE. A SUGGESTED STARTING NUMBER IS 177.
                                        F.  PRESS CONTINUE. THE PROGRAM RUNS CONTINUOUSLY. FIRST IT
                                            LOADS THE PUNCH BUFFER WITH THE CHARACTER IN SR LEFT, AND
                                            THEN DELAYS FOR THE NUMBER OF MILLISECONDS SPECIFIED IN SR
                                            RIGHT BEFORE RELOADING THE PUNCH BUFFER AGAIN.
                                        G.  SET UP A SCOPE AND DISPLAY THE PUNCH CLOCK PULSES. ADJUST
                                            THE PUNCH CLOCK ACCORDING TO SPECIFICATIONS.
```

4.12 PRG12 USE PROCEDURE (DESCRIPTION IN SECTION 8.11)

PRG12 IS USED AS AN AID IN ADJUSTING THE READER/KYBD CLOCK, AND IN OBSERVING THE DATA BITS AS THEY ARE SHIFTED INTO THE READER/KYBD BUFFER. A SCOPE IS REQUIRED.

THE PROGRAM MAKES USE OF THE PUNCH MAINTENANCE BIT FEATURE IN ORDER TO CAUSE THE DATA OUTPUTTED TO THE PUNCH BUFFER TO BE SHIFTED INTO THE READER/KYBD BUFFER.

TO ADJUST THE READER CLOCK PROCEED AS FOLLOWS:

A. LOAD ADDRESS 000200
B. SET SR TO 000012. PRESS START.
C. PROGRAM STOPS AT COMMON HALT.
D. SET ANY DESIRED ASCII CODE IN LEFT HALF OF SR.
E. SET NUMBER OF MILLISECONDS TO DELAY BETWEEN PUNCH COMMANDS IN RIGHT HALF OF SR. THE SELECTED NUMBER SHOULD BE LONG ENOUGH FOR THE ENTIRE PUNCH/READ OPERATION TO COMPLETE. A SUGGESTED STARTING NUMBER IS 177.
F. PRESS CONTINUE. THE PROGRAM RUNS CONTINUOUSLY. FIRST IT LOADS THE PUNCH BUFFER WITH THE CHARACTER IN SR LEFT, AND THEN DELAYS THE NUMBER OF MILLISECONDS SPECIFIED IN SR RIGHT. AS THE DATA BITS ARE SHIFTED OUT OF THE PUNCH BUFFER, THE READER CLOCK STARTS, AND THE DATA BITS ARE SHIFTED INTO THE READER BUFFER. AT THE END OF THE DELAY THE PROGRAM MOVES THE READ BUFFER CONTENTS TO REG 0, AND ISSUES 5 RESET INSTRUCTIONS IN ORDER TO MAKE THE READER BUFFER CONTENTS VISIBLE IN THE RIGHT HALF OF THE DATA LIGHTS.
G. SET UP A SCOPE AND DISPLAY THE READ CLOCK PULSES. ADJUST THE READER CLOCK ACCORDING TO SPECIFICATIONS.

4.13 PRG13 USE PROCEDURE (DESCRIPTION IN SECTION 8.12)

A. SET TELETYPE TO ON-LINE IF CHARACTER TO BE TESTED
   IS TO BE PUNCHED FOR VERIFICATION PURPOSES.
B. TURN ON PUNCH IF CHARACTER IS TO BE PUNCHED.
C. LOAD ADRESS 000200.
D. SET SR TO 000013. PRESS START
E. THE PROGRAM STOPS AT COMMON HALT.
F. SET CODE FOR CHARACTER TO BE TESTED IN THE LEFT HALF OF THE
   SR.
G. PRESS CONTINUE. THE PROGRAM RUNS CONTINUOUSLY, OUTPUTING THE
   CHARACTER TO THE OUTPUT BUFFER AND CHECKING THAT THE READ
   BUFFER CONTAINS THE SAME CHARACTER WHEN THE READER DONE BIT
   BECOMES SET.
H. REFER TO SECTION 6. ERRORS, IF ERRORS OCCUR.

EXECUTION TIME: CONTINUOUS RUNNING.

4.14 PRG14 USE PROCEDURE (DESCRIPTION IN SECTION 8.13)

A. SET TELETYPE TO ON-LINE AND TURN ON PUNCH IF DATA IS
   TO BE PUNCHED FOR VERIFICATION PURPOSES.
B. LOAD ADDRESS 000200.
C. SET SR TO 000014. PRESS START
D. THE PROGRAM RUNS CONTINUOUSLY. THE SPECIAL BINARY COUNT
   PATERN IS OUTPUTTED TO THE OUTPUT BUFFER. EACH TIME
   THE READER DONE BIT BECOMES SET THE CHARACTER IN THE
   READ BUFFER IS CHECKED TO SEE THAT IT MATCHES THE
   PREVIOUSLY OUTPUTTED CHARACTER. THE PROGRAM STALLS
   RANDOMLY BETWEEN CHARACTERS. TO RUN AT FULL SPEED, SET
   SR8 TO A 1.
E. REFER TO SECTION 6. ERRORS, IF ERRORS OCCUR.

EXECUTION TIME: CONTINUOUS RUNNING.

```
                        5.      PROGRAM AND/OR OPERATOR ACTION
                                ******************************

                        5.1     NORMAL HALTS
                                ************

                                LOC 001430      COMMON HALT.  THIS HALT OCCURS WHENEVER
                                                THE PROGRAM IS AWAITING  USER INTERVENTION.
                                                REGISTER ZERO CONTAIN THE ADDRESS OF
                                                INSTRUCTION THAT GENERATED THE CALL TO THE
                                                COMMON HALT.
                                                THIS ADDRESS IS DISPLAYED IN THE DATA LIGHTS,
                                                IF ANY.

                                LOC 001524      END OF ROUTINE HALT.  THIS HALT OCCURS AT THE
                                                END OF A TEST ROUTINE IF SR15 IS SET TO A 1.
                                                TO PROCEED, PRESS CONTINUE.  PROGRAMS PRG0,
                                                PRG1, PRG2, PRG3 AND PRG4 USE THE ROUTINE END
                                                OPTION.

                                LOC 2106        PROGRAM END HALT.THIS HALT NORMALLY OCCURS AT
                                                THE END OF PROGRAMS PRG0, PRG1, PRG2, PRG3, AND
                                                PRG4 UNLESS THE LOOP PROGRAM OPTION IS SET. (SR10)

                        6.      ERRORS
                                ******

                        6.1     ERROR HALTS
                                ***********

                                LOC 001442      UNCONDITIONAL ERROR HALT. REGISTER ZERO CONTAINS
                                                ADDRESS OF INSTRUCTION THAT GENERATED THE
                                                ERROR CALL.  REFER TO PROGRAM LISTING.
                                                THIS ADDRESS IS DISPLAYED IN THE DATA LIGHTS,
                                                IF ANY.

                                LOC 001504      CONDITIONAL ERROR HALT.  THIS CALL WILL ALWAYS
                                                OCCUR, UNLESS SR14 IS SET TO A 1. (SCOPE MODE)
                                                REGISTER ZERO CONTAIN ADDRESS OF INSTRUCTION THAT
                                                GENERATED ERROR CALL. REFER TO PROGRAM LISTING.
                                                THIS ADDRESS IS DISPLAYED IN THE DATA LIGHTS,
                                                IF ANY.

                                LOC 001462      DATA ERROR HALT.  OCCURS WHEN A PROGRAM OR
                                                ROUTINE CHECKING DATA FINDS THAT THE EXPECTED
                                                AND THE RECEIVED DATA DO NOT AGREE.  THE LEFT
                                                HALF OF REGISTER ZERO CONTAIN THE EXPECTED
                                                8 BIT DATA.  THE RIGHT HALF CONTAINS THE
                                                RECEIVED 8 BIT DATA.
                                                THIS DATA IS DISPLAYED IN THE DATA LIGHTS,
                                                IF ANY.
```

6.2 NON RECOVERABLE ERROR HALTS

A NON-RECOVERABLE ERROR HALT WILL OCCUR AT THE ADDRESSES LISTED BELOW IF THROUGH HARDWARE OR SOFTWARE FAILURE, PROGRAM CONTROL IS TRANSFERRED TO AN UNEXPECTED AREA BETWEEN 000000 AND 000776.

```
000002  RESERVED AREA
000006  ERROR TRAP
000012  RESERVED INSTRUCTION TRAP
000016  DEBUG TRAP
000022  IOT TRAP
000026  POWER FAIL TRAP
000040 THROUGH 000176   SYSTEM SOFTWARE AND INTERRUPT VECTOR AREA,
                        EXCEPT FOR KL11 INTERRUPT VECTORS.
```

TO FIND OUT WHERE THE PROGRAM WAS AT THE TIME THE FAILURE OCCURRED, PERFORM THE FOLLOWING STEPS:

A. EXAMINE THE CONTENTS OF REGISTER 6 (ADDRESS 177706).
B. TRANSFER THE CONTENTS OF REGISTER 6 TO THE SR, LOAD ADDRRESS, AND EXAMINE.
C. THE DATA SHOWN IN THE DATA LIGHTS IS THE VALUE OF THE PC WHEN THE FAILURE OCCURRED.
D. LOCATE IN THE PROGRAM LISTING THE DISPLAYED PC VALUE.
E. THE INSTRUCTION THAT IMMEDIATELY PREECEDES THE ONE REFERENCED BY THE DISPLAYED PC VALUE IS THE INSTRUCTION THAT WAS/WAS BEING EXECUTED WHEN THE FAILURE OCCURRED.

7. MISCELLANEOUS

7.1 TEST TAPES

MAINDEC-00-D2G4-PT SPECIAL BINARY COUNT PATTERN TEST TAPE IS RELEASED WITH THIS PROGRAM.

THE SPECIAL BINARY COUNT PATTERN TAPE IS PUNCHED WITH A PATTERN CONSISTING OF THE NUMBERS 000 THROUGH 377. EACH NUMBER IS IMMEDIATELY FOLLOWED BY ITS ONES COMPLEMENT NUMBER. FOR EXAMPLE:

001,376,002,375,003,374,004,373,ETC.

THE EASIEST WAY TO MAKE A SPECIAL BINARY COUNT PATTERN TEST LOOP IS TO OVERLAP THE TAPE AT THE POINT WHERE THE CHARACTERS 377,000,000,377 APPEAR. THAT SEQUENCE OF CHARACTERS APPEARS EVERY 512 CHARACTERS. THEREFORE, A MINIMUM SIZE TEST LOOP WOULD CONSIST OF 512 CHARACTERS.

7.2 SR OPTIONS

SR15 - HALT AT END OF ROUTINE. FOR THOSE PROGRAMS CONSISTING OF A SET OF SEPARATE TEST ROUTINES, SR15 SET TO A 1 CAUSES THE PROGRAM TO HALT UPON COMPLETION OF THE ROUTINE CURRENTLY BEING EXECUTED. THREE POSSIBLE USES OF THIS OPTION ARE:

A. TO STEP THROUGH A PROGRAM ONE ROUTINE AT A TIME.
B. WHEN AN UNPREDICTED FAILURE HAS OCCURRED (BLOW UP, HANG UP), TO ADVANCE THROUGH THE PROGRAM ONE ROUTINE AT A TIME UNTIL THE FAILURE OCCURS. THE ROUTINE FOLLOWING THE LAST IDENTIFIED ROUTINE WOULD BE THE FAILING ROUTINE.
C. WHEN A PROGRAM IS IN EXECUTION, TO DETERMINE HOW FAR THE PROGRAM HAS PROGRESSED.

SR14 - SCOPE. THIS OPTION IS USED ONLY BY PRG0. THE OPTION CAUSES THE PROGRAM TO BYPASS ERROR HALTS, AND TO STAY IN THE FAILING ROUTINE.

SR11 - INHIBIT ITERATION COUNT. THIS OPTION IS USED BY PRG0, PRG1, AND PRG3. THESE PROGRAMS CONSIST OF A SET OF ROUTINES EACH OF WHICH SPECIFIES THE NUMBER OF TIMES A TEST IS TO BE PERFORMED BY MEANS OF AN ITERATION COUNT. SETTING SR11 TO A 1 CAUSES THE PROGRAM TO DISREGARD THE ITERATION COUNT AND PERFORM THE TEST ONLY ONCE FOR EACH ROUTINE. TWO POSSIBLE USES OF THIS OPTION ARE:

A. QUICK PASS. THE USER MAY ELECT TO RUN THROUGH A PROGRAM QUICKLY TO FIND OUT IF ANY FAILURES SHOW IMMEDIATELY. A SUCCESSFUL QUICK PASS HOWEVER, DOES NOT GUARANTEE THAT THE SAME PROGRAM WILL RUN ERROR-FREE WHEN PERFORMING A NORMAL ITERATION PASS.

B. SKIP OVER FAILING ROUTINE. WHEN A ROUTINE HAS DETECTED A SOLID FAILURE, THE ERROR WILL BE REPORTED MANY TIMES. TO GO ON TO THE NEXT ROUTINE, THE USER CAN INHIBIT ITERATION. IT WILL BE NECESSARY TO CAUSE THE PROGRAM TO STOP AT THE END OF THE ROUTINE BY SETTING SR15 TO A 1. OTHERWISE THE PROGRAM WOULD QUICKLY RUN THROUGH THE NEXT ROUTINE(S) ALSO.

(7.2 CONT'D)

SR10 - LOOP PROGRAM. THIS OPTION IS USED BY PROGRAMS PRG0,PRG1,PRG2,PRG3, AND PRG4. SETTING SR10 TO A 1 CAUSES THE PROGRAM TO REPEAT ITSELF, INSTEAD OF STOPPING AT PROGRAM END HALT.

SR9 - SELECT ROUTINE. THIS OPTION IS USED BY PROGRAMS PRG0, PRG1,PRG2,PRG3, AND PRG4. THE USER MAY ELECT TO RUN ONLY ONE SPECIFIC ROUTINE BY SETTING SR9 TO A 1, AND SR6 THROUGH SR0 TO THE NUMBER OF THE DESIRED ROUTINE. REFER TO THE INDIVIDUAL PROGRAM DESCRIPTION IN SECTION 8 TO OBTAIN THE ROUTINE NUMBER. THE ROUTINE NUMBER SELECTED MUST BE A VALID NUMBER, OR AN ERROR HALT WILL OCCUR. THE SELECT ROUTINE OPTION WILL BE HONORED BY THE PROGRAM UPON COMPLETION OF THE CURRENT ROUTINE, OR UPON STARTING THE PROGRAM.

SR8 - DISABLE STALL MODE AND RUN FULL SPEED. USED BY PROGRAMS PRG5,PRG6, AND PRG14. THESE PROGRAMS OPERATE NORMALLY IN STALL MODE (TESTS OR EXERCISES ARE NOT FULL SPEED, BUT RANDOM DURATION DELAYS ARE INTRODUCED). SETTING SR8 TO A 1 CAUSES THE PROGRAMS TO PERFORM THEIR TESTS AT FULL SPEED.

7.3 TESTING KL11 AT NON-STANDARD ADDRESSES AND/OR VECTORS

THIS PROGRAM CAN TEST A KL11 ASSIGNED TO NON-STANDARD ADDRESSES AND VECTORS IF THOSE ADDRESSES ARE PROVIDED TO THE PROGRAM AS FOLLOWS:

A. IMMEDIATELY AFTER LOADING THE PROGRAM CHANGE THE FOLLOWING LOCATIONS. REFER TO PROGRAM LISTING.

LOCATION	FROM STANDARD	TO NON-STANDARD
001210	177560	RDR/KYBD CSR ADDRESS
001212	177562	RDR/KYBD BUFFER ADDRESS
001214	177564	PCH/PRINTER CSR ADDRESS
001216	177566	PCH/PRINTER BUFFER ADDRESS
001220	000060	RDR/KYBD VECTOR ADDRESS
001222	000200	RDR/KYBD PRIORITY LEVEL
001224	000064	PCH/PRINTER VECTOR ADDRESS
001226	000200	PCH/PRINTER PRIORITY LEVEL

B. PROCEED TO USE PROGRAM, OR

C. DUMP OUT ENTIRE PROGRAM IN ABSOLUTE FORMAT, TO HAVE AN UPDATED OBJECT TAPE THAT REFLECTS YOUR SYSTEM, OR

D. DUMP OUT ONLY LOCATIONS 001204 THROUGH 001222, AND SPLICE THE TAPE TO THE END OF THE STANDARD OBJECT TAPE. THIS PROCEDURE WOULD REQUIRE THAT THE SHORT LENGTH OF TAPE BE LOADED IMMEDIATELY AFTER THE MAIN PROGRAM, IN ORDER TO OVERLAY LOCATIONS 001204 THROUGH 001222.

```
619
620                             8.      DESCRIPTION
621                                     ***********
622
623                             8.1     PRG0 PROGRAM DESCRIPTION
624                                     ************************
625
626                                     PRG0 TESTS THE INPUT AND OUTPUT LOGIC IN ONE PROGRAM. THE PROGRAM
627                                     CONSISTS OF 36 TEST ROUTINES NUMBERED FROM 00 TO 43(8).
628
629                                     RTN0    TESTS ABILITY TO REFERENCE THE READER/KYBD STATUS WORD (TKS)
630                                             WITHOUT TRAPPING.
631                                     RTN1    TESTS ABILITY TO REFERENCE THE READER/KYBD BUFFER (TKB)
632                                             WITHOUT TRAPPING.
633                                     RTN2    TESTS ABILITY TO REFERENCE THE PRINTER/PUNCH STATUS WORD (TPS)
634                                             WITHOUT TRAPPING.
635                                     RTN3    TESTS ABILITY TO REFERENCE THE PRINTER/PUNCH BUFFER (TPB)
636                                             WITHOUT TRAPPING.
637                                     RTN4    TESTS ABILITY TO SET AND CLEAR THE READER/KYBD IE BIT.
638                                     RTN5    CHECKS THAT READER /KYBD ID BIT CAN BE CLEARED WITH RESET INSTRUCTION.
639                                     RTN6    CHECKS THAT READER DONE BIT SETS SOMETIME AFTER READER ENABLE.
640                                     RTN7    TESTS THAT DONE BIT CAN BE READ RELIABLY.
641                                     RTN10   TESTS THAT RESET CLEARS DONE BIT.
642                                     RTN11   CHECKS THAT REFERENCING READER BUFFER CLEARS DONE BIT.
643                                     RTN12   TESTS THAT BUSY BIT BIT SETS SOMETIME BEFORE DONE SETS.
644                                     RTN13   TESTS THAT READER DONE BIT IS CLEARED BY START BIT. (WHEN
645                                             BUSY BECOMES SET.)
646                                     RTN14   TESTS THAT READ BUFFER CAN BE READ RELIABLY.
647                                     RTN15   CHECKS THAT READER DONE BIT IS ABLE TO CAUSE AN INTERRUPT. IF THE
648                                             INTERRUPT IS SERVICED, IT WILL HAVE OCCURRED AT CORRECT VECTOR.
649                                     RTN16   TESTS THAT READER DONE DOES NOT CAUSE AN INTERRUPT WHEN THE PROCESSOR
650                                             IS AT THE SAME PRIORITY AS THE READER'S INTERRUPT REQUEST LEVEL.
651                                     RTN17   TESTS THAT READER DONE CAUSES  INTERRUPT WHEN THE PROCESSOR IS AT A
652                                              PRIORITY ONE LEVEL LOWER THAN THE READER'S INTERRUPT REQUEST LEVEL.
653                                     RTN20   CHECKS THAT READER DONE DOES NOT REINTERRUPT AFTER RTI
654                                             INSTRUCTION WHEN DONE BIT IS LEFT SET.
655                                     RTN21   TESTS ABILITY TO SET AND CLEAR THE PUNCH ID BIT.
656                                     RTN22   CHECKS THAT PUNCH ID BIT CAN BE CLEARED WITH RESET INSTRUCTION.
657                                     RTN23   TESTS ABILITY TO SET AND CLEAR PUNCH MAINTENANCE BIT.
658                                     RTN24   CHECKS THAT RESET INSTRUCTION CLEARS THE MAINTENANCE BIT.
659                                     RTN25   TESTS THAT RESET SETS THE PUNCH READY BIT, AND THAT THE
660                                             READY BIT CAN BE READ RELIABLY.
661                                     RTN26   TESTS THAT PUNCH READY IS CLEARED BY LOADING THE PUNCH BUFFER.
662                                     RTN27   TESTS THAT BYTE LOADING PUNCH BUFFER+1 DOES NOT CLEAR THE
663                                             PUNCH READY BIT.
664                                     RTN30   CHECKS THAT THE PUNCH BECOMES READY SOMETIME AFTER BUFFER LOAD.
665                                     RTN31   CHECKS THAT PUNCH READY BIT CAN CAUSE INTERRUPT. IF THE INTERRUPT
666                                             IS SERVICED, IT WILL HAVE OCCURRED AT THE CORRECT VECTOR
667                                     RTN32   TESTS THAT PUNCH READY DOES NOT CAUSE AN INTERRUPT WHEN THE
668                                             PROCESSOR IS AT A PRIORITY AS THE READER'S INTERRUPT REQUEST
669                                             LEVEL.
670                                     RTN33   TESTS THAT PUNCH READY CAUSES AN INTERRUPT WHEN THE PROCESSOR
```

8.1(CONT'D)

IS AT PRIORITY ONE LEVEL LOWER THAN THE PUNCH INTERRUPT REQUEST LEVEL.

RTN34 TESTS THAT READY DOES NOT REINTERRUPT AFTER RTI WITH READY SET.

RTN35 CHECKS THAT PUNCH READY CAUSES AN INTERRUPT IMMEDIATELY UPON LOWERING PROCESSOR PRIORITY TO 0.

RTN36 CHECKS FOR CORRECT OPERATION OF WAIT INSTRUCTION. (REFER TO PROGRAM LISTING).

RTN37 TESTS THAT LOADING PUNCH BUFFER WITH MAINTENANCE BIT SET CAUSES READER DONE BIT TO SET SOMETIME AFTER.

RTN40 TESTS THAT CLEARING READY/INTERRUPT ENABLE CLEARS PUNCH INTERRUPT REQUEST.

RTN41 TESTS THAT CLEARING READER DONE/INTERRUPT ENABLE CLEARS READER INTERRUPT REQUEST.

RTN42 TESTS THAT THE DL11A,B KEYBOARD JUMPERS ARE CUT PROPERLY

RTN43 TESTS THAT THE DL11A,B PRINTER JUMPERS ARE CUT PROPERLY

8.2 PRG1 PROGRAM DESCRIPTION

PRG1 CHECKS FOR CORRECT AND RELIABLE OPERATION OF THE PAPER TAPE READER. THE PROGRAM CONTAINS 3 ROUTINES NUMBERED FROM 00 TO 02.

RTN0 READS AND CHECKS 2000 CHARACTERS OF SPECIAL BINARY COUNT PATTERN AT FULL SPEED.

RTN1 READS AND CHECKS 1000 CHARACTERS OF SPECIAL BINARY COUNT PATTERN WITH RANDOM DURATION STALLS BETWEEN CHARACTERS.

RTN2 READS AND CHECKS 200 GROUPS OF CHARACTERS OF SPECIAL BINARY COUNT PATTERN. LENGTH OF EACH GROUP IS RANDOM, BUT DOES NOT EXCEED 15 CHARACTERS.

```
8.3     PRG2 PROGRAM DESCRIPTION
        ----------------------

        PRG2 EXERCISES THE TELEPRINTER FUNCTIONS. VERIFICATION OF TELEPRINTER
        OPERATION IS PERFORMED VISUALLY BY USER. THE PROGRAM CONTAINS
        29 ROUTINES NUMBERED FROM 00 TO 34(8).

        RTN0    CARRIAGE RETURN TEST. CHECKS THAT CARRIAGE CAN CORRECTLY
                RETURN TO PRINT POSITION 0 FROM EVERY OTHER PRINT POSITION.


        RTN1    RIGHT MARGIN TEST. THIS ROUTINE VERIFIES THAT THE RIGHT
                MARGIN IS CORRECTLY SET FOR 72 PRINT POSITIONS. THE TEST
                TYPES 73 CHARACTERS. IF THE RIGHT MARGIN IS CORRECTLY SET,
                CHARACTER 73 SHOULD OVERPRINT CHARACTER 72. THE TYPED LINE
                SHOULD LOOK AS FOLLOWS:

----I----I----I----I----I----I----I----I----I----I----I----I----I----I-I

        RTN2    SPACE TEST. CHECKS ABILITY OF TELEPRINTER TO SPACE CORRECTLY.
                THE TEST FIRST PRINTS REVERSE SLASHES (\) IN ALTERNATE PRINT
                POSITIONS, AND THEN SPACES FROM PRINT POSITION 0 TO EACH
                PRINT POSITION AND PRINTS A SLASH (/). THE TYPEOUT SHOULD
                LOOK AS FOLLOWS:

\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/




        RTN3    LINE FEED TEST. TESTS FOR ABILITY TO CORRECTLY PERFORM A
                LINE FEED. A RANDOM STALL OCCURS BETWEEN EACH LINE FEED. A
                CORRECTLY PERFORMED TEST WILL APPEAR AS DIAGONAL LINE BETWEEN
                PRINT POSITION 0 AND PRINT POSITION 72.

        RTN4    TAB TEST. ASR/KSR35 ONLY. CHECKS ABILITY OF TELEPRINTER TO TAB TO
                TO A TAB POSITION FROM 1 TO 7 POSITIONS AWAY FROM THE TAB.
                THE FIRST LINE TYPED MARKS THE TAB POSITIONS. SUBSEQUENT
                LINES ARE THE ACTUAL TAB TEST. THE TYPEOUT LOOKS AS FOLLOWS:

        /       /       /       /       /       /       /       /       /
        /       /       /       /       /       /       /       /       /
        /       /       /       /       /       /       /       /       /
        /       /       /       /       /       /       /       /       /
        /       /       /       /       /       /       /       /       /
        /       /       /       /       /       /       /       /       /
```

(8,3 CONT'D)

```
                RTN5    TYPES LINE OF CHARACTERS ABC
                RTN6    TYPES LINE OF CHARACTERS DEF
                RTN7    TYPES LINE OF CHARACTERS GHI
                RTN10   TYPES LINE OF CHARACTERS JKL
                RTN11   TYPES LINE OF CHARACTERS MNO
                RTN12   TYPES LINE OF CHARACTERS PQR
                RTN13   TYPES LINE OF CHARACTERS STU
                RTN14   TYPES LINE OF CHARACTERS VWX
                RTN15   TYPES LINE OF CHARACTERS YZ0
                RTN16   TYPES LINE OF CHARACTERS 123
                RTN17   TYPES LINE OF CHARACTERS 456
                RTN20   TYPES LINE OF CHARACTERS 789
                RTN21   TYPES LINE OF CHARACTERS !"#
                RTN22   TYPES LINE OF CHARACTERS $%&
                RTN23   TYPES LINE OF CHARACTERS '()
                RTN24   TYPES LINE OF CHARACTERS *+,
                RTN25   TYPES LINE OF CHARACTERS -./
                RTN26   TYPES LINE OF CHARACTERS :;<
                RTN27   TYPES LINE OF CHARACTERS =>?
                RTN30   TYPES LINE OF CHARACTERS @[\
                RTN31   TYPES LINE OF CHARACTERS ]^ AND LEFT ARROW
                RTN32   TYPES 2 LINES OF ALL CHARACTERS, FIRST LINE IS TYPED AT
                        FULL SPEED, SECOND LINE IS TYPED WITH RANDOM STALLS.
                RTN33   TYPES 12 LINES OF ASR33 WORST CASE PATTERN, EVERY OTHER
                        LINE IS TYPED WITH RANDOM STALLS. THE ASR33 WORST CASE
                        PATTERN IS '_W/W_
                RTN34   TYPES 12 LINES OF ASR35 WORST CASE PATTERN, EVERY OTHER
                        LINE IS TYPED WITH RANDOM STALLS. THE ASR35 WORST CASE
                        PATTERN IS '[?C?[
```

8.4 PRG3 PROGRAM DESCRIPTION

PRG3 CHECKS FOR CORRECT PUNCH OPERATION. THE PROGRAM CONTAINS 3 ROUTINES NUMBERED FROM 00 TO 02.

RTN0 PUNCHES 512 CHARACTERS OF SPECIAL BINARY COUNT PATTERN AT FULL SPEED, AND VERIFIES THE PUNCHED DATA WITH THE PAPER TAPE READER. TEST IS DONE 5 TIMES.

RTN1 PUNCHES 512 CHARACTERS OF SPECIAL BINARY COUNT PATTERN WITH RANDOM DURATION STALLS BETWEEN EACH CHARACTER PUNCHED. THE DATA PUNCHED IS VERIFIED WITH THE PAPER TAPE READER. THE TEST IS DONE 5 TIMES.

RTN2 PUNCHES 512 CHARACTERS OF SPECIAL BINARY COUNT PATTERN IN GROUPS OF RANDOM LENGTH (15 CHARACTERS MAXIMUM). AFTER EACH GROUP IS PUNCHED A RANDOM DURATION STALL OCCURS. THE DATA PUNCHED IS VERIFIED WITH THE PAPER TAPE READER. THE TEST IS DONE 5 TIMES.

8.5 PRG4 PROGRAM DESCRIPTION

PRG4 IS USED TO TEST THE TELETYPE KEYBOARD. THE PROGRAM CONTAINS 3 ROUTINES NUMBERED FROM 00 TO 02.

RTN0 TESTS THAT TELETYPE CONTROL RESPONDS WHEN USER DEPRESSES A KEYBOARD KEY.

RTN1 ECHO TEST. THE TEST ECHOES ONTO THE TELEPRINTER THE CHARACTER RECEIVED FROM THE KEYBOARD. WHEN THE TEST SENSES A RUBOUT CHARACTER THE TEST IS ENDED. THE TEST ENABLES THE USER TO DETERMINE IF ALL PRINTABLE CODES CAN BE SUCCESSFULLY SENT TO THE TELETYPE CONTROL.

RTN2 OCTAL EQUIVALENT TEST. THE OCTAL EQUIVALENT OF ANY CHARACTER RECEIVED BY THE CONTROL IS TYPED. SENSING A RUBOUT ENDS THE TEST. THIS TEST ENABLES THE USER TO DETERMINE THAT ALL CODES INCLUDING NON-PRINTABLE CONTROL CODES ARE BEING CORRECTLY SENT TO THE TELETYPE CONTROL.

8.6 PRG5 PROGRAM DESCRIPTION

PRG5 EXERCISES THE PAPER TAPE READER, PAPER TAPE PUNCH, AND THE TELEPRINTER IN COMBINATION. ITS PURPOSE IS TO SHOW UP ANY FAILURES THAT MAY OCCUR AS A RESULT OF INTERCACTION BETWEEN THE TELETYPE DEVICES AND THE PROCESSOR.

PRG5 IS A CONTINUOUS RUNNING PROGRAM. DATA CODES 241 THROUGH 337 ARE USED TO EXERCISE THE DEVICES. THE DATA IS OUTPUTTED TO THE PRINTER/PUNCH IN GROUPS OF 72 CHARACTERS PRECEDED BY CARRIAGE RETURN AND LINE FEED. ONE PROGRAM PASS CONSISTS OF 63 LINES FOLLOWED BY A BLANK LINE. THE DATA IS OUTPUTTED SO THAT EVERY CHARACTER WILL HAVE PRINTED IN EVERY PRINT POSITION BY THE TIME ONE PROGRAM PASS HAS BEEN COMPLETED.

THE PROGRAM WORKS IN THE INTERRUPT MODE, AND IS ARRANGED SO THAT THE PAPER TAPE READER WILL AT NO TIME BE MORE THAN 40 CHARACTERS BEHIND THE PUNCH (EXCLUDING INITIAL PUNCH TO READER SLACK.)

NORMAL OPERATION IS WITH RANDOM STALLS. SR8 MAY BE SET TO A 1 TO RUN THE TEST AT FULL SPEED.

8.7 PRG6 PROGRAM DESCRIPTION

PRG6 IS A CONTINUOUS RUNNING PROGRAM THAT EXERCISES THE PAPER TAPE READER USING A SPECIAL BINARY COUNT PATTERN TEST TAPE. THE PROGRAM STALLS RANDOMLY AFTER READING RANDOM LENGTH CHARACTER GROUPS (15 CHARACTERS MAXIMUM) AT FULL SPEED. SR8 MAY BE SET TO A 1 TO EXERCISE THE READER AT FULL SPEED.

8.8 PRG7 PROGRAM DESCRIPTION

PRG7 IS A PRINTER EXERCISER DESIGNED AS AN AID IN MAKING TELEPRINTER ADJUSTMENTS. THE PROGRAM PERMITS THE USER TO TYPE IN FIVE TEST CHARACTERS AND ONE FINAL CHARACTER THAT SIGNIFIES WHETHER FULL SPEED OR STALL OPERATION IS DESIRED. THE PROGRAM THEN TYPES LINES CONTAINING THE FIVE SELECTED CHARACTERS. WHEN THE USER WISHES TO CHANGE THE TEST CHARACTERS SR15 IS SET TO A 1. THE PROGRAM TERMINATES TYPING THE LINE BEFORE ACCEPTING NEW DATA.

8.9 PRG10 PROGRAM DESCRIPTION

PRG10 IS A TAPE GENERATOR THAT PUNCHES OUT THE SPECIAL BINARY COUNT PATTERN CONTINUOUSLY.

```
                                8.10    PRG11 PROGRAM DESCRIPTION
                                        ------------------------

                                        PRG11 IS USED AS AN AID IN ADJUSTING THE TELETYPE PUNCH CLOCK WITH
                                        THE AID OF A SCOPE. THE PROGRAM PERFORMS THE FOLLOWING SEQUENCE:

                                        A.  LOAD PUNCH BUFFER WITH ASCII CODE IN SR LEFT.
                                        B.  DELAY NUMBER OF MILLISECONDS SET IN SR RIGHT.
                                        C.  GO TO STEP A.

                                8.11    PRG12 PROGRAM DESCRIPTION
                                        ------------------------

                                        PRG12 IS USED AS AN AID IN ADJUSTING THE TELETYPE READER CLOCK.
                                        A SCOPE IS REQUIRED. THE PROGRAM PERFORMS THE FOLLOWING SEQUENCE:

                                        A.  SET PUNCH MAINTENANCE BIT.
                                        B.  LOAD PUNCH BUFFER WITH CODE IN SR LEFT.
                                        C.  DELAY NUMBER OF MILLISECONDS SET IN SR RIGHT.
                                        D.  MOVE CONTENTS OF READ BUFFER TO REGISTER 0.
                                        E.  ISSUE 5 RESET INSTRUCTIONS TO "FIX" READ BUFFER CONTENTS IN
                                            RIGHT HALF OF DATA LIGHTS.
                                        F.  GO TO STEP A.

                                8.12    PRG13 PROGRAM DESCRIPTION
                                        ------------------------

                                        USING THE PUNCH MAINTENANCE BIT FEATURE, PRG13 TAKES THE ASCII CODE
                                        SET IN SR LEFT AND USES IT TO CHECK THE ABILITY OF THE CONTROL TO
                                        OUTPUT AND RECEIVE DATA. THE PROGRAM PERFORMS THE FOLLOWING SEQUENCE:

                                        A.  SET PUNCH MAINTENANCE BIT.
                                        B.  LOAD PUNCH BUFFER WITH CODE IN SR LEFT.
                                        C.  WHEN READER DONE BIT SETS, COMPARE CODE IN SR LEFT WITH DATA
                                            IN READER BUFFER. HALT IF NOT SAME.
                                        D.  WAIT FOR PUNCH DONE BIT TO SET AND GO TO STEP B.

                                8.14    PRG14 PROGRAM DESCRIPTION
                                        ------------------------

                                        USING THE PUNCH MAINTENANCE BIT FEATURE PRG14 USES THE SPECIAL
                                        BINARY COUNT PATTERN TO CHECK ABILITY OF THE CONTROL TO OUTPUT
                                        AND RECEIVE DATA. THE PROGRAM PERFORMS THE FOLLOWING STEPS:

                                        A.  INITIALIZE BINARY COUNT PATTERN.
                                        B.  SET PUNCH MAINTENANCE BIT.
                                        C.  LOAD PUNCH BUFFER WITH BINARY COUNT CHARACTER.
                                        D.  WHEN READER DONE BIT SETS, COMPARE BINARY CHARACTER WITH DATA
                                            IN READ BUFFER. HALT IF NOT SAME.
                                        E.  WAIT FOR PUNCH DONE BIT TO SET AND GO TO STEP C.

                                        .ENDR
```

```
                                        .ABS
                                        .LIST   ME
                                        .NLIST  MD,MC
                                ;PRG0= COMBINED INPUT-OUTPUT LOGIC TESTS.
                                ;PRG1= READER TEST.
                                ;PRG2= PRINTER TEST.
                                ;PRG3= PUNCH TEST.
                                ;PRG4= KEYBOARD TEST.
                                ;PRG5= COMBINED TEST.
                                ;PRG6= READER EXERCISER, SPECIAL BINARY COUNT PATTERN.
                                ;PRG7= PRINTER EXERCISER.
                                ;PRG10= SPECIAL BINARY COUNT PATTERN TAPE GENERATOR.
                                ;PRG11= PUNCH CLOCK ADJUSTMENT ROUTINE.
                                ;PRG12= READER CLOCK ADJUSTMENT ROUTINE.
                                ;PRG13= MAINTENANCE MODE SINGLE CHARACTER DATA TEST.
                                ;PRG14= MAINTENANCE MODE SPECIAL BINARY COUNT PATTERN DATA TEST.
                                ;
                                ;STANDARD SR SWITCH OPTIONS (SWITCH SET TO A 1 )
                                ;
                                ;SR 15 = HALT AT END OF ROUTINE.
                                ;SR 14 = SCOPE.
                                ;SR 11 = INHIBIT ITERATION.
                                ;SR 10 =  LOOP PROGRAM
                                ;SR  9 = SELECT ROUTINE.
                                ;SR 8 = DISABLE STALL MODE AND RUN FULL SPEED.
                                ;SR 6 THROUGH SR 0 = NUMBER OF ROUTINE TO BE SELECTED.
        000000                          .=0
000000  000002                          .+2                     ;UNASSIGNED TRAP
000002  000000                          HALT
000004  000006                  MACHER: .+2                     ;SP OVERFLOW, BUS ERROR TRAP
000006  000000                          HALT
000010  000012                          .+2                     ;RESERVED INSTRUCTION TRAP
000012  000000                          HALT
000014  000016                          .+2                     ;TRACE TRAP
000016  000000                          HALT
000020  000022                          .+2                     ;TRAP TO CALL IOX
000022  000000                          HALT
000024  000026                          .+2                     ;POWER FAIL TRAP
000026  000000                          HALT
000030  002170                          EMTINT                  ;EMT TRAP
000032  000340                          PRTY7
000034  003344                          DLYX                    ;TRAP TRAP, SIMILAR TO EMT
000036  000340                          PRTY7
                                ;LOC 40 THROUGH 776 ARE FILLED WITH .+2 AND HALT.

        000046                          .=46
000046  002062                          LOGIC                   ;SET TO ADDRESS OF ACT 11 HOOKS

                                ;EQUATE STATEMENTS
        177776                          CC=177776
        177776                          PSW=177776
        001200                          SPBOT=1200
        000240                          NOP=240
```

```
  999                 000000                          OPEN=0
 1000                 100000                          MANUAL=BIT15
 1001                 000000                          R0=%0
 1002                 000006                          R6=%6
 1003                 000007                          PC=%7
 1004                 100000                          BIT15=100000
 1005                 040000                          BIT14=40000
 1006                 020000                          BIT13=20000
 1007                 010000                          BIT12=10000
 1008                 004000                          BIT11=4000
 1009                 002000                          BIT10=2000
 1010                 001000                          BIT9=1000
 1011                 000400                          BIT8=400
 1012                 000200                          BIT7=200
 1013                 000100                          BIT6=100
 1014                 000040                          BIT5=40
 1015                 000020                          BIT4=20
 1016                 000010                          BIT3=10
 1017                 000004                          BIT2=4
 1018                 000002                          BIT1=2
 1019                 000001                          BIT0=1
 1020                 005726                          POPSP=5726              ;POP THE STACK, SAME AS TST (6)+
 1021                 022626                          POPSP2=022626           ;POP STACK TWICE, SAME AS CMP (6)+,(6)+
 1022                 000340                          PRTY7=340               ;PRIORITY LEVEL DEFINITIONS
 1023                 000300                          PRTY6=300
 1024                 000240                          PRTY5=240
 1025                 000200                          PRTY4=200
 1026                 000140                          PRTY3=140
 1027                 000100                          PRTY2=100
 1028                 000040                          PRTY1=40
 1029                 000000                          PRTY0=0
 1030                 000000                          EMTX=0
 1031                 104400                          DELAYX=TRAP+0
 1032                 000007                          BELL=007
 1033                 015142                          BLOCKA=DEND
 1034                 015144                          BLOCK1=BLOCKA+2
 1035                 015254                          BLOCKB=BLOCKA+112
 1036                 015265                          BLKBB=BLOCKA+123
 1037                 015256                          BLOCK2=BLOCKA+114
 1038                 015267                          BLK2=BLOCKA+125
 1039                 015366                          BLOCKC=BLOCKA+224
 1040                 015377                          BLKCC=BLOCKA+235
 1041                 177777                          P0TLST=-1
 1042                 177777                          P1TLST=-1
 1043                 177777                          P2TLST=-1
 1044                 177777                          P3TLST=-1
 1045                 177777                          P4TLST=-1
 1046                 000174                          .=174
 1047  000174  177570                        SRPTR:  177570
 1048  000176  000000                        SOFTSR: 000000
 1049                 000200                          .=200
 1050  000200  012706  001200                        MOV     #SPBOT,%6               ;SET BOTTOM OF THE STACK
 1051  000204  016746  177576                        MOV     6,-(SP)         ;SAVE CURRENT VECTOR
 1052  000210  016746  177570                        MOV     4,-(SP)
```

```
 1053  000214  012767  000230  177562          MOV     #18,4                   ;SET UP TIME OUT VECTOR
 1054  000222  005777  177746                  TST     @SRPTR                  ;TRY TO REFERENCE THE
 1055                                                                          ;HARDWARE SWITCH REGISTER
 1056  000226  000404                          BR      2$                      ;BRANCH IF NO TIME OUT TRAP OCCURRS
 1057  000230  012767  000176  177736  1$:     MOV     #SOFTSR,SRPTR           ;CHANGE THE SWITCH REGISTER POINTER
 1058                                                                          ;TO POINT TO A SOFTWARE  SWITCH REGISTER
 1059  000236  022626                          CMP     (6)+,(6)+               ;RESTORE THE STACK
 1060  000240  012667  177540          2$:     MOV     (6)+,4                  ;RESTORE TIME OUT VECTOR
 1061  000244  012667  177536                  MOV     (6)+,6
 1062  000250  000167  001254                  JMP     START           ;GO TO START OF PROGRAM.
 1063          001210                          .=1210
 1064  001210  177560                  TKS:    177560                  ;LSR CSR
 1065  001212  177562                  TKB:    177562                  ;LSR BUFFER
 1066  001214  177564                  TPS:    177564                  ;LSP CSR
 1067  001216  177566                  TPB:    177566                  ;LSP BUFFER
 1068  001220  000060                  TKVTR:  60                      ;LSR INTERRUPT VECTOR
 1069  001222  000200                  TKLVL:  PRTY4                   ;LSR PRIORITY LEVEL
 1070  001224  000064                  TPVTR:  64                      ;LSP INTERRUPT VECTOR
 1071  001226  000200                  TPLVL:  PRTY4                   ;LSP PRIORITY LEVEL
 1072  001230  000000                  TTYTYP: OPEN
 1073  001232  000000                  PRGNUM: OPEN                    ;CONTAINS CURRENT PROGRAM#
 1074  001234  000000                  KSTART: OPEN                    ;CURRENT PROGRAM START ADDRESS.
 1075  001236  000000                  CURTST: OPEN                    ;CONTAINS ADDR OF CURRENT TEST.
 1076  001240  000000                  RTNNO:  OPEN                    ;CONTAINS CURRENT TEST #.
 1077  001242  000000                  NXTST:  OPEN                    ;CONTAINS ADDR OF NEXT TEST.
 1078  001244  000000                  ICTR:   OPEN                    ;CONTAINS CURRENT ITERATION COUNT
 1079  001246  000000                  SCOPTR: OPEN                    ;CONTAINS CURRENT SCOPE POINTER.
 1080  001250  000000                  PRGID:  OPEN                    ;CONTAINS PROGRAM INDICATORS
 1081  001252  005144                  PRGTAB: PRG0                    ;PRG0 START ADDRESS
 1082  001254  007652                          PRG1                    ;PRG1 START ADDRESS
 1083  001256  010026                          PRG2                    ;PRG2 START ADDRESS
 1084  001260  011462                          PRG3                    ;PRG3 START ADDRESS
 1085  001262  012110                          PRG4                    ;PRG4 START ADDRESS
 1086  001264  012410                          PRG5                    ;PRG5 START ADDRESS
 1087  001266  013370                          PRG6                    ;PRG6 START ADDRESS
 1088  001270  013454                          PRG7                    ;PRG7 START ADDRESS
 1089  001272  013626                          PRG10                   ;PRG10 START ADDRESS
 1090  001274  013662                          PRG11                   ;PRG11 START ADDRESS
 1091  001276  013672                          PRG12                   ;PRG12 START ADDRESS
 1092  001300  013762                          PRG13                   ;PRG13 START ADDRESS
 1093  001302  014034                          PRG14                   ;PRG14 START ADDRESS
 1094  001304                          EMTTAB:
 1095  001304  002666                          TYP                     ;POINTER FOR EMT CALL TYPE
 1096  001306  003014                          TYPS                    ;POINTER FOR EMT CALL TYPES
 1097  001310  003146                          STAL                    ;POINTER FOR EMT CALL STALL
 1098  001312  001466                          ERR                     ;POINTER FOR EMT CALL ERROR
 1099  001314  001446                          DTCHK                   ;POINTER FOR EMT CALL DATCHK
 1100  001316  001422                          CHLT                    ;POINTER FOR EMT CALL CHALT
 1101  001320  002414                          STLSRV                  ;POINTER FOR EMT CALL STRDRV
 1102  001322  002444                          STLSPV                  ;POINTER FOR EMT CALL STPCHV
 1103  001324  001434                          EHLT                    ;POINTER FOR EMT CALL EHALT
 1104  001326  002474                          SRSETT                  ;POINTER FOR EMT CALL SRESET
 1105  001330  001754                          CHAINN                  ;POINTER FOR EMT CALL SCOPE
 1106  001332  002210                          CHK33                   ;POINTER FOR EMT CALL CK33
```



```
 1107  001334  002226                          CHK35                   ;POINTER FOR EMT CALL CK35
 1108  001336  002224                          CHK33B                  ;POINTER FOR EMT CALL CK37
 1109  001340  004574                          TYPL3                   ;POINTER FOR EMT CALL TYPLN3
 1110  001342  001456                          DTHLT                   ;POINTER FOR EMT CALL DATHLT
 1111  001344  002264                          SAVRG                   ;POINTER FOR EMT CALL SAVREG
 1112  001346  002324                          RSTRG                   ;POINTER FOR EMT CALL RSTREG
 1113  001350  002240                          CKASR                   ;POINTER FOR EMT CALL CHKASR
 1114  001352  002512                          RSETT2                  ;POINTER FOR EMT CALL RESET2
 1115  001354  003046                          DLY                     ;POINTER FOR EMT CALL DELAY
 1116  001356  003206                          RDLY                    ;POINTER FOR EMT CALL RDELAY
 1117  001360  003306                          RSTAL                   ;POINTER FOR EMT CALL RSTALL
 1118  001362  000000                  RCNT:   OPEN                    ;CHARACTER COUNT
 1119  001364  000000                  CRBUF:  OPEN                    ;HOLDS ONE CHARACTER FROM READER.
 1120  001366  000000                  CHR1:   OPEN
 1121  001370  000000                  CHR2:   OPEN
 1122  001372  000000                  CHR3:   OPEN
 1123  001374  000000                  CHR1A:  OPEN
 1124  001376  000000                  CHR2A:  OPEN
 1125  001400  000000                  CHR3A:  OPEN
 1126  001402  000000                  ERCTR:  OPEN
 1127  001404  000000                  CTRA:   OPEN
 1128  001406  000000                  CTRB:   OPEN
 1129  001410  000000                  CTRC:   OPEN
 1130  001412  000000                  CTRD:   OPEN
 1131  001414  000000                  BRCTR:  OPEN
 1132  001416  000000                  DVDND:  OPEN
 1133  001420  000000                  DVQUOT: OPEN
 1134                                  ;COMMON HALT ROUTINE
 1135  001422  011600                  CHLT:   MOV     @X6,X0          ;DEVELOP ADDRESS OF CALLER.
 1136  001424  162700  000002                  SUB     #2,X0
 1137  001430  000000                          HALT                    ;HALT. ADDRESS OF CALL INSTRUCTION
 1138  001432  000002                          RTI                     ;IN DATA LIGHTS.
 1139                                  ;UNCONDITIONAL ERROR HALT ROUTINE.
 1140  001434  011600                  EHLT:   MOV     @X6,X0          ;DEVELOP ADDRESS OF CALLER.
 1141  001436  162700  000002                  SUB     #2,X0
 1142  001442  000000                          HALT                    ;HALT. ADDR OF ERROR CALL
 1143  001444  000002                          RTI                     ;IN DATA LIGHTS.
 1144                                  ;DATA CHECK ROUTINE.
 1145  001446  126767  177712  177711  DTCHK:  CMPB    CRBUF,CRBUF+1   ;COMPARE EXPECTED AND RECEIVED
 1146  001454  001403                          BEQ     DTCHKA          ;CHARS. BRANCH IF SAME.
 1147  001456  016700  177702          DTHLT:  MOV     CRBUF,X0        ;MOVE S/B AND WAS CHARS TO R0.
 1148  001462  000000                          HALT                    ;DATA ERROR HALT. GOOD CHAR IN
 1149                                                                  ;DATA LIGHTS LEFT, BAD CHAR IN DATA
 1150  001464  000002                  DTCHKA: RTI                     ;LIGHTS RIGHT. EXIT.
 1151                                  ;CONDITIONAL ERROR HALT.
 1152  001466  032777  040000  176500  ERR:    BIT     #BIT14,@SRPTR   ;SCOPE SWITCH SET?
 1153  001474  001004                          BNE     ERRA            ;BR IF YES.
 1154  001476  011600                          MOV     @X6,X0          ;DEVELOP CALLER'S ADDRESS.
 1155  001500  162700  000002                  SUB     #2,X0
 1156  001504  000000                          HALT                    ;ERROR HALT.
 1157  001506  000002                  ERRA:   RTI                     ;EXIT.
 1158                                  ;ROUTINE END HALT SUBROUTINE.
 1159  001510  032777  100000  176456  SHALT:  BIT     #BIT15,@SRPTR   ;HALT AT END OF TEST?
 1160  001516  001403                          BEQ     SHLTA           ;BR IF NOT.
```

```
 1161  001520  116700  177514                  MOVB    RTNNO,%0        ;CURRENT TEST # TO R0.
 1162  001524  000000                          HALT                    ;ROUTINE END HALT.
 1163  001526  000207                  SHLTA:  RTS     %7              ;EXIT.
```



```
 1164  001530  012706  001200          START:  MOV     #SPBOT,%6       ;SET BOTTOM OF SP STACK.
 1165  001534  005067  176236                  CLR     PSW
 1166  001540  012767  000006  176236          MOV     #6,MACHER
 1167  001546  005067  177466                  CLR     RTNNO
 1168  001552  005737  000042                  TST     @#42            ;CHAIN OR AUTO-ACCEPTANCE?
 1169  001556  001404                          BEQ     1$              ;BR IF NOT.
 1170  001560  004767  001616                  JSR     PC,TIMCAL       ;CALIBRATE DELAY ROUTINE.
 1171  001564  000167  012244                  JMP     PRG14           ;GO RUN PRG14.
 1172  001570  017700  176400          1$:     MOV     @SRPTR,%0               ;(@SRPTR) TO R0
 1173  001574  042700  177760                  BIC     #177760,%0      ;LIMIT (SR) TO BITS 3-0
 1174  001600  020027  000014                  CMP     %0,#14          ;COMPARE (SR) TO PROGRAM LIMIT
 1175  001604  101402                          BLOS    CRTA            ;VALID PROGRAM NUMBER?
 1176  001606  104010          INCPRG: EHALT                           ;NO, INCORRECT PRG NUMBER
 1177  001610  000747                          BR      START           ;START OVER.
 1178  001612  005067  177432          CRTA:   CLR     PRGID
 1179  001616  010067  177410                  MOV     %0,PRGNUM       ;SAVE PROGRAM NUMBER AT PRGNUM
 1180  001622  001404                          BEQ     CRTB            ;BR IF 0.
 1181  001624  004767  001552                  JSR     PC,TIMCAL       ;CALIBRATE DELAY ROUTINE.
 1182  001630  016700  177376                  MOV     PRGNUM,%0       ;PRGNUM BACK TO R0.
 1183  001634  000241          CRTB:   CLC
 1184  001636  006100                          ROL     %0              ;R0X2
 1185  001640  000170  001252                  JMP     @PRGTAB(0)      ;GO TO SELECTED PROGRAM.
 1186  001644  104005          SRSET:  CHALT                           ;SET SR OPTIONS DESIRED
 1187  001646  016767  177362  177366  GETRDY: MOV     KSTART,NXTST    ;ADDR OF 1ST ROUTINE TO NXTST
 1188  001654  012767  000006  176122  CLEAN:  MOV     #6,MACHER       ;RESET MACHER TRAP.
 1189  001662  012706  001200                  MOV     #SPBOT,%6       ;SET UP BOTTOM OF STACK.
 1190  001666  104400                          DELAYX
 1191  001670  104011                          SRESET
 1192  001672  005067  176100                  CLR     PSW
 1193  001676  004767  000210          GTRDYA: JSR     %7,FORWD        ;ROLL FORWARD TO "NEXT" ROUTINE.
 1194  001702  032777  001000  176264  GTRDYB: BIT     #BIT9,@SRPTR    ;SELECT ROUTINE SWITCH SET?
 1195  001710  001002                          BNE     GTRDYC          ;BRANCH IF YES.
 1196  001712  000177  177320                  JMP     @CURTST         ;RUN CURRENT ROUTINE.
 1197  001716  017700  176252          GTRDYC: MOV     @SRPTR,%0       ;(@SRPTR) TO R0
 1198  001722  042700  177600                  BIC     #177600,%0      ;MASK UNDESIRED BITS
 1199  001726  126700  177306                  CMPB    RTNNO,%0        ;COMPARE RTNNO TO (R0)
 1200  001732  001002                          BNE     GTRDYD          ;BRANCH IF ROUTINE NOT FOUND YET.
 1201  001734  000177  177276                  JMP     @CURTST         ;GO RUN ROUTINE.
 1202  001740  022767  177777  177274  GTRDYD: CMP     #-1,NXTST       ;NO, CHECK FOR LAST ROUTINE.
 1203  001746  001353                          BNE     GTRDYA          ;LAST ROUTINE?
 1204  001750  104010          INCRTN: EHALT                           ;YES, INCORRECT ROUTINE SELECTED.
 1205  001752  000735                          BR      GETRDY          ;START OVER.
 1206  001754  012706  001200          CHAINN: MOV     #SPBOT,R6       ;RESET STACK.
 1207  001760  032777  040000  176206          BIT     #BIT14,@SRPTR   ;SCOPE?
 1208  001766  001406                          BEQ     CHNA            ;BR IF NOT.
 1209  001770  022767  177777  177250          CMP     #-1,SCOPTR      ;YES. SCOPE POINTER = -1?
 1210  001776  001402                          BEQ     CHNA            ;BRANCH IF SCOPE ENTRY IS -1.
 1211  002000  000177  177242                  JMP     @SCOPTR         ;RETURN TO ROUTINE.
 1212  002004  032777  004000  176162  CHNA:   BIT     #BIT11,@SRPTR   ;INHIBIT ITERATION?
 1213  002012  001005                          BNE     CHNAA           ;BR IF YES.
 1214  002014  005367  177224                  DEC     ICTR            ;NO, ICTR 0?
 1215  002020  001402                          BEQ     CHNAA           ;BR IF YES.
 1216  002022  000177  177220                  JMP     @SCOPTR         ;NO, RETURN TO TEST ROUTINE
 1217  002026  004767  177456          CHNAA:  JSR     %7,$HALT        ;GO HALT IF HALT SWITCH IS SET
```

```
 1218  002032  032777  001000  176134  CHN8:   BIT     #BIT9,@SRPTR    ;SELECT ROUTINE?
 1219  002040  001382                          BNE     GETRDY          ;BR IF YES.
 1220  002042  022767  177777  177172          CMP     #-1,NXTST       ;NO. LAST TEST?
 1221  002050  001381                          BNE     CLEAN           ;BR IF NOT.
 1222  002052  013700  000042          CHNC:   MOV     @42,R0          ;GET CONTENTS OF 42.
 1223  002056  001407                          BEQ     HERE            ;BR IF 0.
 1224  002060  000005                          RESET
 1225  002062  004710                  LOGIC:  JSR     PC,(0)          ;RETURN TO MONITOR.
 1226  002064  000240  000240  000240          .WORD   NOP,NOP,NOP
 1227  002072  000167  011736                  JMP     PRG14           ;RETURN TO PRG14.
 1228  002076  032777  002000  176078  HERE:   BIT     #BIT10,@SRPTR   ;LOOP PROGRAM?
 1229  002104  001260                          BNE     GETRDY          ;BR IF YES.
 1230  002106  000000                          HALT                    ;PROGRAM END HALT.
 1231  002110  000656                          BR      GETRDY          ;RESTART.
 1232  002112  016705  177124          FORWD:  MOV     NXTST,R5        ;ADDR OF NEXT ROUTINE TO R5.
 1233  002116  012567  177116                  MOV     (5)+,RTNNO      ;GET NEXT ROUTINE NUMBER.
 1234  002122  012567  177114                  MOV     (5)+,NXTST      ;GET ADDR OF NEXT "NEXT" ROUTINE.
 1235  002126  105767  177116                  TSTB    PRGID           ;CHECK IF PROGRAM SCOPE AND I COUNT
 1236  002132  100407                          BMI     FORWDB          ;PARAMETERS. BRANCH IF NOT.
 1237  002134  012567  177104                  MOV     (5)+,ICTR       ;GET ITERATION COUNT.
 1238  002140  012567  177102                  MOV     (5)+,SCOPTR     ;GET SCOPE LOOP ENTRY POINTER.
 1239  002144  010567  177066          FORWDA: MOV     R5,CURTST       ;ADDR OF NOW CURRENT TEST TO CURTST.
 1240  002150  000207                          RTS     R7              ;EXIT FORWD SUBROUTINE.
 1241  002152  012767  177777  177066  FORWDB: MOV     #-1,SCOPTR      ;FORCE "NO SCOPE"
 1242  002160  012767  000001  177056          MOV     #1,ICTR         ;FORCE I COUNT OF 1
 1243  002166  000766                          BR      FORWDA
 1244                                  ;EMT INTERPRTER ROUTINE.
 1245  002170  010046                  EMTINT: MOV     R0,-(6)         ;PUSH R0.
 1246  002172  016600  000002                  MOV     2(6),R0         ;GET EMT PC.
 1247  002176  014000                          MOV     -(0),R0         ;GET EMT CALL.
 1248  002200  006300                          ASL     R0              ;TIMES 2.
 1249  002202  016000  171304                  MOV     EMTTAB-10000(0),R0 ;DEVELOP EMT ADDR.
 1250  002206  000200                          RTS     R0              ;GO TO EMT RTN. RESTORE R0.
 1251                                  ;ROUTINES TO DETERMINE TTY TYPES.
 1252  002210  032767  000001  177012  CHK33:  BIT     #1,TTYTYP       ;CHECK FOR 33
 1253  002216  001002                          BNE     CHK33B          ;BR IF NOT 33.
 1254  002220  062716  000002          CHK33A: ADD     #2,(6)          ;SET UP 33 EXIT.
 1255  002224  000002                  CHK33B: RTI                     ;EXIT.
 1256  002226  032767  000001  176774  CHK35:  BIT     #1,TTYTYP       ;CHECK FOR 35.
 1257  002234  001371                          BNE     CHK33A          ;BR IF 35.
 1258  002236  000002                          RTI                     ;NOT 35.
 1259  002240  032767  000010  176762  CKASR:  BIT     #BIT3,TTYTYP    ;CHECK FOR ASR TTY.
 1260  002246  001001                          BNE     .+4             ;BRANCH IF NOT ASR.
 1261  002250  000002                          RTI                     ;ASR. EXIT.
 1262  002252  022626                          POPSP2                  ;POP STACK TWICE.
 1263  002254  012767  000001  176762          MOV     #1,ICTR         ;FORCE I COUNT TO A 1.
 1264  002262  104012                          SCOPE                   ;SCOPE TO BYPASS ROUTINE.
 1265                                  ;SAVE REGS 0 TO 4 SUBROUTINE.
 1266  002264  012667  000030          SAVRG:  MOV     (6)+,SVRPC      ;SAVE PC AND PSW.
 1267  002270  012667  000026                  MOV     (6)+,SVRPSW
 1268  002274  010446                          MOV     R4,-(6)         ;SAVE REGS 0 - 4
 1269  002276  010346                          MOV     R3,-(6)         ;IN STACK.
 1270  002300  010246                          MOV     R2,-(6)
 1271  002302  010146                          MOV     R1,-(6)
```



```
 1272  002304  010046                          MOV     R0,-(6)
 1273  002306  016746  000010                  MOV     SVRPSW,-(6)     ;RESTORE PC AND PSW.
 1274  002312  016746  000002                  MOV     SVRPC,-(6)
 1275  002316  000002                          RTI                     ;EXIT.
 1276  002320  000000                  SVRPC:  OPEN
 1277  002322  000000                  SVRPSW: OPEN
 1278                                  ;RESTORE REGS 0 TO 4 SUBROUTINE.
 1279  002324  012667  000030          RSTRG:  MOV     (6)+,RSTPC      ;SAVE PC AND PSW.
 1280  002330  012667  000026                  MOV     (6)+,RSTPSW
 1281  002334  012600                          MOV     (6)+,R0         ;RESTORE REGS 0 - 4
 1282  002336  012601                          MOV     (6)+,R1         ;FROM STACK.
 1283  002340  012602                          MOV     (6)+,R2
 1284  002342  012603                          MOV     (6)+,R3
 1285  002344  012604                          MOV     (6)+,R4
 1286  002346  016746  000010                  MOV     RSTPSW,-(6)     ;RESTORE PC AND PSW.
 1287  002352  016746  000002                  MOV     RSTPC,-(6)
 1288  002356  000002                          RTI                     ;EXIT
 1289  002360  000000                  RSTPC:  OPEN
 1290  002362  000000                  RSTPSW: OPEN
 1291                                  ;ROUTINE TO FETCH A CHARACTER
 1292  002364  012767  000310  177022  AREAD:  MOV     #200.,BRCTR     ;SET UP DELAY COUNT.
 1293  002372  005277  176612                  INC     @TKS            ;ENABLE READER.
 1294  002376  104400                          DELAYX                  ;WAIT.
 1295  002400  105777  176604                  TSTB    @TKS            ;DONE SET?
 1296  002404  100402                          BMI     ARDB            ;BR IF YES.
 1297  002406  104010                          EHALT                   ;ERROR. NO RESPONSE FROM READER.
 1298  002410  000765                          BR      AREAD           ;TRY AGAIN.
 1299  002412  000207                  ARDB:   RTS     R7              ;EXIT
 1300                                  ;ROUTINE TO SET LSR INTERRUPT VECTOR AND PRIORITY
 1301  002414  017667  000000  000012  STLSRV: MOV     @(6),STPRA+2    ;MOVE VECTOR ADDR TO STPRA+2
 1302  002422  062716  000002                  ADD     #2,@#6          ;SET UP EXIT
 1303  002426  016701  176566                  MOV     TKVTR,R1
 1304  002432  012721  000000          STPRA:  MOV     #OPEN,(1)+      ;SET VECTOR ADDRESS
 1305  002436  016721  176560                  MOV     TKLVL,(1)+      ;SET PRIORITY
 1306  002442  000002                          RTI                     ;EXIT
 1307                                  ;ROUTINE TO SET LSP INTERRUPT VECTOR AND PRIORITY.
 1308  002444  017667  000000  000012  STLSPV: MOV     @(6),STPPA+2    ;MOVE VECTOR ADDR TO STPPA+2
 1309  002452  062716  000002                  ADD     #2,@#6          ;SET UP EXIT
 1310  002456  016701  176542                  MOV     TPVTR,R1
 1311  002462  012721  000000          STPPA:  MOV     #OPEN,(1)+      ;SET VECTOR ADDRESS.
 1312  002466  016721  176534                  MOV     TPLVL,(1)+      ;SET PRIORITY
 1313  002472  000002                          RTI                     ;EXIT.
 1314                                  ;ROUTINE TO ISSUE RESET.
 1315  002474  012700  052525          SRSETT: MOV     #52525,R0               ;DATA TO R0.
 1316  002500  005100                          COM     R0              ;COMPLEMENT (R0).
 1317  002502  010067  177770                  MOV     R0,@SRSETT+2    ;(R0) TO SRSETT+2.
 1318  002506  000005                          RESET                   ;ISSUE RESET. (R0) IS
 1319  002510  000002                          RTI                     ;DISPLAYED. EXIT.
 1320                                  ;DOUBLE RESET SUBROUTINE.
 1321  002512  104011                  RSETT2: SRESET
 1322  002514  104011                          SRESET
 1323  002516  000002                          RTI                     ;EXIT.
 1324                                  ;RANDOM NUMBER GENERATOR. ROUTINE EXITS WITH NUMBER IN REGISTER 0.
 1325  002520  016700  000042          RNGEN:  MOV     RP1,R0
```

```
1326  002524  006100                          ROL     %0
1327  002526  006100                          ROL     %0
1328  002530  066700  000034                  ADD     RP2,%0
1329  002534  010067  000026                  MOV     %0,RP1
1330  002540  006100                          ROL     %0
1331  002542  006100                          ROL     %0
1332  002544  066700  000020                  ADD     RP2,%0
1333  002550  006100                          ROL     %0
1334  002552  006100                          ROL     %0
1335  002554  010067  000010                  MOV     %0,RP2
1336  002560  016700  000002                  MOV     RP1,%0
1337  002564  000207                          RTS     %7              ;EXIT, NUMBER IN R0
1338  002566  001233          RP1:    1233
1339  002570  007622          RP2:    7622
1340  002572  104006          BREAD:  $TRDRV                  ;SET READER VECTOR
1341  002574  002632                  BREADB                  ;TO BREADB
1342  002576  052777  000101  176404          BIS     #101,@TKS       ;ENABLE LSR AND LSRI.
1343  002604  104024                          DELAY                   ;AWAIT INTERRUPT.
1344  002606  000310                          200.
1345  002610  005077  176374                  CLR     @TKS            ;CLEAR LSRI ENABLE.
1346  002614  104010                          EMALT                   ;NO RESPONSE HALT.
1347  002616  000765                          BR      BREAD           ;TRY AGAIN.
1348  002620  022626          BREADA: POPSP2
1349  002622  117767  176364  176534          MOVB    @TKB,CRBUF      ;CHAR READ TO CRBUF.
1350  002630  000207                          RTS     %7              ;EXIT SUBROUTINE.
1351  002632  005077  176352          BREADB: CLR     @TKS            ;CLEAR LSR INTERRUPT ENABLE.
1352  002636  105777  176346                  TSTB    @TKS            ;TEST FOR DONE.
1353  002642  100003                          BPL     BREADC          ;BRANCH IF DONE NOT SET.
1354  002644  012716  002620                  MOV     #BREADA,@%6     ;MODIFY INTERRUPT EXIT TO BREADA.
1355  002650  000002                          RTI                     ;OK. EXIT INTERRUPT.
1356  002652  000000          BREADC: HALT                    ;HALT, DONE BIT NOT SET AFTER INTERRUPT.
1357  002654  012716  002662                  MOV     #BREADD,(6)     ;POINT TO BREADD.
1358  002660  000002                          RTI                     ;EXIT INTERRUPT.
1359  002662  022626          BREADD: POPSP2
1360  002664  000742                          BR      BREAD           ;TRY AGAIN.
1361                                  ;SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE PRINTER.
1362  002666  011600          TYP:    MOV     @%6,%0          ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS.
1363  002670  062716  000002                  ADD     #2,@%6          ;SET UP EXIT.
1364  002674  011000                          MOV     @%0,%0          ;ADDRESS OF MESSAGE TO R0.
1365  002676  112067  000110          TYPA:   MOVB    (0)+,TYPDAT     ;GET CHARACTER
1366  002702  122767  000100  000102          CMPB    #100,TYPDAT     ;CHECK FOR"@"CHARACTER
1367  002710  001003                          BNE     TYPC            ;BRANCH IF NOT"@".
1368  002712  104024                          DELAY                   ;WAIT 100 MSECS.
1369  002714  000144                          100.
1370  002716  000002                          RTI                     ;TERMINATOR CHAR. DONE. EXIT.
1371  002720  122767  000045  000064  TYPC:   CMPB    #45,TYPDAT      ;CHECK FOR"%".
1372  002726  001416                          BEQ     TYPF            ;BRANCH IF"%"
1373  002730  122767  000043  000054          CMPB    #43,TYPDAT      ;NOT"%",CHECK FOR"#".
1374  002736  001417                          BEQ     TYPG            ;BRANCH IF "#"
1375  002740  004767  000002                  JSR     %7,TYPD         ;TYPE CHAR IN TYPDAT
1376  002744  000754                          BR      TYPA
1377  002746  116777  000040  176242  TYPD:   MOVB    TYPDAT,@TPB     ;OUTPUT CHARACTER TO PRINTER
1378  002754  105777  176234                  TSTB    @TPS            ;WAIT FOR DONE FLAG.
1379  002760  100375                          BPL     .-4
```



```
1380  002762  000207                          RTS     %7              ;EXIT
1381  002764  112767  000015  000020  TYPF:   MOVB    #15,TYPDAT      ;MOVE CARRIAGE RETURN CODE TO TYPDAT
1382  002772  004767  177750                  JSR     %7,TYPD         ;GO TYPE CHAR.
1383  002776  112767  000012  000006  TYPG:   MOVB    #12,TYPDAT      ;MOVE LF CODE TO TYPDAT.
1384  003004  004767  177736                  JSR     %7,TYPD         ;GO TYPE CHAR.
1385  003010  000732                          BR      TYPA
1386  003012  000000          TYPDAT: OPEN
1387                                  ;SUBROUTINE TO OUTPUT A SERIES OF ASCII MESSAGES ON TELETYPE PRINTER
1388  003014  011600          TYPS:   MOV     @%6,%0          ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS
1389  003016  062716  000002                  ADD     #2,@%6          ;UPDATE TO NEXT MESSAGE ADDRESS
1390  003022  011067  000014                  MOV     @%0,TYPSB       ;ADDRESS OF MESSAGE TO TYPSB
1391  003026  022767  177777  000006          CMP     #-1,TYPSB       ;CHECK FOR TERMINATOR
1392  003034  001001                          BNE     TYPSA           ;BRANCH IF NOT TERMINATOR.
1393  003036  000002                          RTI                     ;TERMINATOR, EXIT
1394  003040  104000          TYPSA:  TYPE                    ;CALL ON TYP SUB TO TYPE MESSAGE
1395  003042  000000          TYPSB:  OPEN                    ;ADDRESS OF MESSAGE GOES HERE
1396  003044  000763                          BR      TYPS            ;GO PROCESS NEXT MESSAGE
1397                                  ;SUBROUTINE TO DELAY A SPECIFIED NUMBER OF MILLISECONDS
1398          003050                  DLCNT=.+2
1399  003046  011627  000000          DLY:    MOV     (6),#0          ;GET DELAY COUNT ADDRESS.
1400  003052  062716  000002                  ADD     #2,@%6          ;SET UP EXIT ADDRESS
1401  003056  017767  177766  177764          MOV     @DLCNT,DLCNT    ;DELAY COUNT TO STACK
1402  003064  005067  174706                  CLR     PSW             ;SET PRIORITY 0
1403          003072                  MSEC=.+2
1404  003070  012767  000000  000046  DLYA:   MOV     #0,DLYT         ;1 MSEC COUNT TO DLYT
1405  003076  016767  000042  000040  DLYB:   MOV     DLYT,DLYT
1406  003104  016767  000034  000032          MOV     DLYT,DLYT
1407  003112  016767  000026  000024          MOV     DLYT,DLYT
1408  003120  016767  000020  000016          MOV     DLYT,DLYT
1409  003126  005367  000012                  DEC     DLYT            ;DECREMENT DLYT.
1410  003132  001361                          BNE     DLYB            ;BRANCH IF NOT 0.
1411  003134  005367  177710                  DEC     DLCNT           ;DECREMENT COUNT
1412  003140  001353                          BNE     DLYA            ;BR IF NOT DONE DELAYING
1413  003142  000002                          RTI                     ;EXIT.
1414  003144  000000          DLYT:   OPEN
1415                                  ;SUBROUTINE TO STALL A RANDOM NUMBER OF MILLISECONDS. MAXIMUM STALL
1416                                  ;DETERMINED BY CONTENTS OF LOC STLMSK.
1417  003146  032767  040000  176074  STAL:   BIT     #BIT14,PRGID    ;TEST FOR STALLS ALLOWED.
1418  003154  001001                          BNE     STALAA          ;ALLOWED.
1419  003156  000002                          RTI                     ;NOT ALLOWED.
1420  003160  004767  177334          STALAA: JSR     %7,RNGEN        ;GO GET RANDOM NUMBER.
1421  003164  046700  000014                  BIC     STLMSK,%0       ;# IN R0, APPLY STALL MASK.
1422  003170  001404                          BEQ     STALB           ;BRANCH IF RESULT IS 0.
1423  003172  010067  000002                  MOV     %0,STALA
1424  003176  104024                          DELAY                   ;DELAY
1425  003200  000000          STALA:  OPEN                    ;DELAY COUNT
1426  003202  000002          STALB:  RTI                     ;DONE, EXIT.
1427  003204  000000          STLMSK: OPEN                    ;STALL MASK.
1428                                  ;READER DELAY ROUTINE.
1429          003210                  RDLCNT=.+2
1430  003206  011627  000000          RDLY:   MOV     (6),#0
1431  003212  062716  000002                  ADD     #2,(6)
1432  003216  017767  177766  177764          MOV     @RDLCNT,RDLCNT  ;GET DELAY COUNT.
1433  003224  005067  174546                  CLR     PSW
```

```
1434  003230  016767  177636  000046  RDLYA:  MOV     MSEC,DLYR
1435  003236  016767  000042  000040  RDLYB:  MOV     DLYR,DLYR
1436  003244  016767  000034  000032          MOV     DLYR,DLYR
1437  003252  016767  000026  000024          MOV     DLYR,DLYR
1438  003260  016767  000020  000016          MOV     DLYR,DLYR
1439  003266  005367  000012                  DEC     DLYR            ;DECREMENT DLYR.
1440  003272  001361                          BNE     RDLYB           ;BR IF NOT 0.
1441  003274  005367  177710                  DEC     RDLCNT          ;DECREMENT COUNT.
1442  003300  001353                          BNE     RDLYA           ;BR IF NOT 0.
1443  003302  000002                          RTI                     ;EXIT.
1444  003304  000000                  DLYR:   OPEN
1445                                  ;READER STALL ROUTINE.
1446  003306  032767  040000  175734  RSTAL:  BIT     #BIT14,PRSID    ;STALL?
1447  003314  001001                          BNE     RSTLA           ;BR IF YES.
1448  003316  000002                          RTI                     ;NO. EXIT.
1449  003320  004767  177174          RSTLA:  JSR     PC,RNGEN        ;GET RANDOM NUMBER.
1450  003324  046700  177654                  BIC     STLMSK,%0
1451  003330  001404                          BEQ     RSTLB
1452  003332  010067  000002                  MOV     %0,RSTLAA
1453  003336  104025                          RDELAY                  ;DELAY.
1454  003340  000000                  RSTLAA: OPEN
1455  003342  000002                  RSTLB:  RTI                     ;DONE. EXIT.
1456                                  ;SUB TO DELAY X TIME.
1457          003350                          DLYR0=DLYX+4
1458          003362                          DLYR1=DLYXA+4
1459  003344  012727  000144  000000  DLYX:   MOV     #100.,#0        ;
1460  003352  005067  174420                  CLR     PSW
1461  003356  012727  001750  000000  DLYXA:  MOV     #1000.,#0       ;COUNT TO DLYR1.
1462  003364  005367  177772          DLYXB:  DEC     DLYR1           ;DECREMENT DLYR1.
1463  003370  001375                          BNE     DLYXB           ;BR IF NOT 0.
1464  003372  005367  177752                  DEC     DLYR0           ;DECREMENT DLYR0.
1465  003376  001367                          BNE     DLYXA           ;BR IF NOT 0.
1466  003400  000002                          RTI                     ;EXIT.
1467                                  ;DELAY ROUTINE CALIBRATE ROUTINE.
1468  003402  004767  000050          TIMCAL: JSR     PC,TSPCH        ;OUTPUT CHAR.
1469  003406  004767  000044                  JSR     PC,TSPCH        ;OUTPUT CHAR.
1470  003412  104400                          DELAYX
1471  003414  104011                          SRESET
1472  003416  005067  175776                  CLR     DVQUOT
1473  003422  016767  175766  175766          MOV     BRCTR,DVDND     ;SAVE BRCTR CONTENTS.
1474  003430  162767  000144  175760  TIMCLA: SUB     #100.,DVDND     ;DVDND=100
1475  003436  103403                          BLO     TIMCLB          ;BR IF UNSUCCESFUL.
1476  003440  005267  175754                  INC     DVQUOT          ;INCR QUOTIENT.
1477  003444  000771                          BR      TIMCLA
1478  003446  016767  175746  177416  TIMCLB: MOV     DVQUOT,MSEC     ;SAVE 1 MSEC CONSTANT.
1479  003454  000207                          RTS     PC              ;EXIT.
1480  003456  104007                  TSPCH:  STPCHV                  ;SET UP VECTOR.
1481  003460  003542                          TSPCHA
1482  003462  005067  175726                  CLR     BRCTR
1483  003466  005077  175524                  CLR     @TPB            ;LOAD BUFFER WITH 0.
1484  003472  052777  000100  175514          BIS     #BIT6,@TPS      ;ENABLE INTERRUPT.
1485  003500  016767  175710  175706  TSPCHC: MOV     BRCTR,BRCTR
1486  003506  016767  175702  175700          MOV     BRCTR,BRCTR
1487  003514  016767  175674  175672          MOV     BRCTR,BRCTR
```



```
1488  003522  016767  175666  175664          MOV     BRCTR,BRCTR
1489  003530  005267  175660                  INC     BRCTR
1490  003534  001361                          BNE     TSPCHC          ;BR IF RESULT NOT 0.
1491  003536  001434                          EHLT                    ;NO INTERRUPT FROM PUNCH/PRINTER.
1492  003540  000777                          BR      .
1493  003542  012716  003550          TSPCHA: MOV     #TSPCHB,(6)     ;MODIFY INTERRUPT EXIT TP STPCHB.
1494  003546  000002                          RTI                     ;EXIT INTERRUPT.
1495  003550  000207                  TSPCHB: RTS     PC              ;EXIT.
1496                                  ;SUBROUTINE TO GENERATE RANDOM CHARACTER COUNT
1497  003552  004767  176742          GRCNT:  JSR     %7,RNGEN        ;GET RANDOM NUMBER
1498  003556  046700  000010                  BIC     RCMSK,%0        ;APPLY MASK
1499  003562  001773                          BEQ     GRCNT           ;TRY AGAIN IF RESULT 0
1500  003564  010067  000004                  MOV     %0,RNCNT        ;COUNT TO RNCNT
1501  003570  000207                          RTS     %7              ;EXIT.
1502  003572  000000                  RCMSK:  OPEN                    ;RANDOM CHARACTER MASK.
1503  003574  000000                  RNCNT:  OPEN                    ;RANDOM CHARACTER COUNT.
1504                                  ;SUB TO COMPARE READER DATA AGAINST EXPECTED DATA AND REPORT ERRORS.
1505  003576  004767  000262          BCHECK: JSR     %7,GTBIN        ;GET BIN CHARACTER(IN R0)
1506  003602  110067  175557                  MOVB    %0,CRBUF+1      ;S/B CHAR TO CRBUF+1
1507  003606  126767  175552  175551          CMPB    CRBUF,CRBUF+1   ;COMPARE S/B AND WAS CHARS.
1508  003614  001001                          BNE     .+4             ;BRANCH IF NOT SAME.
1509  003616  000207                          RTS     %7              ;SAME, EXIT.
1510  003620  104017                          DATHLT                  ;GO HALT AND DISPLAY DATA.
1511  003622  005367  175554                  DEC ERCTR               ;3 ERRORS?
1512  003626  001002                          BNE     .+6             ;BRANCH IF NOT 3 YET.
1513  003630  004767  000002                  JSR     %7,BSYNC        ;3 ERRORS, RESYNC READER.
1514  003634  000207                          RTS     %7              ;EXIT.
1515                                  ;SUBROUTINE TO SYNC THE LSR TO A SPECIAL BINARY COUNT PATTERN TEST TAPE.
1516  003636  004767  000164          BSYNC:  JSR     %7,INBIN        ;INITIALIZE BINARY PATTERN
1517  003642  004767  176724                  JSR     %7,BREAD        ;READ CHAR AND STORE AT CHR1
1518  003646  116767  175512  175512          MOVB    CRBUF,CHR1
1519  003654  004767  176712                  JSR     %7,BREAD        ;READ CHAR AND STORE AT CHR2
1520  003660  116767  175500  175502          MOVB    CRBUF,CHR2
1521  003666  004767  176700                  JSR     %7,BREAD        ;READ CHAR AND STORE AT CHR3.
1522  003672  116767  175466  175472          MOVB    CRBUF,CHR3
1523  003700  004767  000012                  JSR     %7,SYNCA        ;GO SYNC
1524  003704  000754                          BR      BSYNC           ;NO SYNC. TRY AGAIN.
1525  003706  012767  000003  175466          MOV     #3,ERCTR
1526  003714  000207                          RTS     %7              ;SUCCESS,EXIT.
1527  003716  012767  001000  000100  SYNCA:  MOV     #512.,SYCTRA    ;512 TO SYCTRA.
1528  003724  004767  000134          SYNCB:  JSR     %7,GTBIN        ;BIN CHAR TO CHR1A.
1529  003730  010067  175440                  MOV     %0,CHR1A
1530  003734  004767  000124                  JSR     %7,GTBIN        ;BIN CHAR TO CHR2A.
1531  003740  010067  175432                  MOV     %0,CHR2A
1532  003744  004767  000114                  JSR     %7,GTBIN        ;BIN CHAR TO CHR3A.
1533  003750  010067  175424                  MOV     %0,CHR3A
1534  003754  026767  175406  175412          CMP     CHR1,CHR1A      ;MATCH?
1535  003762  001013                          BNE     SYNCC           ;BR IF NOT.
1536  003764  026767  175400  175404          CMP     CHR2,CHR2A      ;MATCH?
1537  003772  001007                          BNE     SYNCC           ;BR IF NOT.
1538  003774  026767  175372  175376          CMP     CHR3,CHR3A      ;MATCH?
1539  004002  001003                          BNE     SYNCC           ;BR IF NOT.
1540  004004  062716  000002                  ADD     #2,(6)          ;SET UP SYNCED EXIT.
1541  004010  000207                          RTS     %7              ;EXIT.
```

```
1542  004012  005367  000006          SYNCC:  DEC     $YCTRA          ;TRIED 512 TIMES?
1543  004016  001342                          BNE     $YNCB           ;BR IF NOT.
1544  004020  104010                          EHALT                   ;YES, SYNC ERROR.
1545  004022  000207                          RTS     %7              ;SYNC ERROR EXIT.
1546  004024  000000          $YCTRA: OPEN
1547                          ;SUBROUTINE TO INITIALIZE BINARY COUNT PATTERNS.
1548  004026  012767  177777  000014  INBIN:  MOV     #-1,RIND        ;SET ALL VARIABLES
1549  004034  004567  000300                  JSR     %5,BMOVE        ;TO MINUS 1.
1550  004040  004050                          RIND
1551  004042  004051                          RIND+1
1552  004044  000013                          11.
1553  004046  000207                          RTS     %7              ;EXIT
1554  004050  000000          RIND:   OPEN
1555  004052  000000          PT0:    OPEN
1556  004054  000000          PT1:    OPEN
1557  004056  000000          PIND:   OPEN
1558  004060  000000          PT0P:   OPEN
1559  004062  000000          PT1P:   OPEN
1560                          ;SPECIAL BINARY COUNT PATTERN SUBROUTINE. EXITS WITH BIN CHAR IN R0
1561  004064  016767  177762  177762  GTBIN:  MOV     PT0,PT1         ;PREVIOUS BIN CHR TO PT1
1562  004072  005167  177756                  COM     PT1
1563  004076  005167  177746                  COM     RIND
1564  004102  001002                          BNE     .+6
1565  004104  005267  177744                  INC     PT1
1566  004110  042767  177400  177736          BIC     #177400,PT1     ;MASK TO 8 BITS
1567  004116  016767  177732  177726          MOV     PT1,PT0         ;SAVE BIN CHAR IN PT0
1568  004124  016700  177724                  MOV     PT1,%0          ;BIN CHAR TO R0.
1569  004130  000207                          RTS     %7              ;EXIT.
1570  004132  016767  177722  177722  GTBINP: MOV     PT0P,PT1P       ;PREVIOUS BIN CHAR TO PT1P
1571  004140  005167  177716                  COM     PT1P
1572  004144  005167  177706                  COM     PIND
1573  004150  001002                          BNE     .+6
1574  004152  005267  177704                  INC     PT1P
1575  004156  042767  177400  177676          BIC     #177400,PT1P    ;MASK TO 8 BITS.
1576  004164  016767  177672  177666          MOV     PT1P,PT0P       ;SAVE BIN CHAR IN PT0P.
1577  004172  016701  177664                  MOV     PT1P,%1         ;BIN CHAR TO R1.
1578  004176  000207                          RTS     %7              ;EXIT.
1579                          ;OCTAL TO ASCII CONVERT ROUTINES
1580  004200  012500          ACNV6:  MOV     (5)+,%0         ;CONVERT TO 6 ASCII, GET OCTAL ADDRESS
1581  004202  012567  000012                  MOV     (5)+,ACNV8      ;GET ASCII ADDRESS
1582  004206  004767  000052                  JSR     %7,ACNV         ;CONVERT TO ASCII
1583  004212  004567  000122                  JSR     %5,BMOVE        ;MOVE 6 CHARS TO ASCII ADDRESS
1584  004216  004254                          A1ST
1585  004220  000000          ACNV8:  OPEN
1586  004222  000006                          6
1587  004224  000205                          RTS     %5              ;EXIT
1588  004226  012500          ACNV4:  MOV     (5)+,%0         ;CONVERT TO 4 ASCII, GET OCTAL ADDRESS
1589  004230  012567  000012                  MOV     (5)+,ACNVC      ;GET ASCII ADDRESS
1590  004234  004767  000024                  JSR     %7,ACNV         ;CONVERT TO ASCII
1591  004240  004567  000074                  JSR     %5,BMOVE        ;MOVE 4 CHARS TO ASCII ADDRESS.
1592  004244  004256                          A1ST+2
1593  004246  000000          ACNVC:  OPEN
1594  004250  000004                          4
1595  004252  000205                          RTS     %5              ;EXIT
```



```
1596  004254  000000          A1ST:   OPEN
1597  004256  000000                  OPEN
1598  004260  000000                  OPEN
1599  004262  000000          ACNVX:  OPEN
1600  004264  012701  004262  ACNV:   MOV     #A1ST+6,%1      ;ADDR TO STORE ASCII TO R1
1601  004270  012702  000006          MOV     #6,%2           ;6 TO R2
1602  004274  011067  177762          MOV     @%0,ACNVX       ;OCTAL WORD TO ACNVX
1603  004300  016703  177756  ACNVM:  MOV     ACNVX,%3
1604  004304  042703  177770          BIC     #177770,%3      ;ISOLATE LEAST SIGNIFICANT OCTAL #
1605  004310  062703  000060          ADD     #60,%3          ;ADD 60 TO CONVERT TO ASCII
1606  004314  110341                  MOVB    %3,-(1)         ;STORE ASCII BYTE
1607  004316  006067  177740          ROR     ACNVX           ;MOVE NEXT OCTAL DIGIT TO LEAST
1608  004322  006067  177734          ROR     ACNVX           ;SIGNIFICANT POSITION
1609  004326  006067  177730          ROR     ACNVX
1610  004332  005302                  DEC     %2              ;DONE 6 TIMES?
1611  004334  001361                  BNE     ACNVM           ;NO, REPEAT.
1612  004336  000207                  RTS     %7              ;YES, EXIT.
1613                          ;SUBROUTINE TO MOVE A VARIABLE NUMBER OF BYTES.
1614  004340  104020          BMOVE:  SAVREG                  ;SAVE REGS.
1615  004342  012501                  MOV     (5)+,%1         ;GET"FROM"ADDRESS
1616  004344  012502                  MOV     (5)+,%2         ;GET"TO"ADDRESS
1617  004346  012503                  MOV     (5)+,%3         ;GET COUNT
1618  004350  112122          BMOVA:  MOVB    (1)+,(2)+       ;MOVE BYTE
1619  004352  005303                  DEC     %3              ;DECREMENT COUNT
1620  004354  001375                  BNE     BMOVA           ;BRANCH IF NOT DONE.
1621  004356  104021                  RSTREG                  ;RESTORE REGS.
1622  004360  000205                  RTS     %5              ;DONE EXIT
1623                          ;SUBROUTINE TO CHECK FOR PUNCH READY.
1624  004362  105777  174626  CPRDY:  TSTB    @TPS            ;TEST FOR READY BIT.
1625  004366  100001                  BPL     CPRDYA          ;BRANCH IF READY NOT SET.
1626  004370  000207                  RTS     %7              ;OK, EXIT.
1627  004372  104010          CPRDYA: EHALT                   ;NOT READY, HALT.
1628  004374  000772                  BR      CPRDY
1629                          ;SUBROUTINE TO PUNCH ON LSP CHARACTER IN REG 0.
1630  004376  004767  177760  LSPCH:  JSR     %7,CPRDY        ;GO CHECK FOR PUNCH READY.
1631  004402  010077  174610          MOV     %0,@TPB         ;LOAD PUNCH BUFFER.
1632  004406  105777  174602          TSTB    @TPS            ;WAIT FOR DONE.
1633  004412  100375                  BPL     .-4
1634  004414  005000                  CLR     %0
1635  004416  000207                  RTS     %7              ;DONE, EXIT.
1636                          ;BINARY TO DECIMAL ASCII CONVERT SUBROUTINE.
1637  004420  012700  015135  BDCNV:  MOV     #DECVAL,%0      ;SET UP ADDR TO STORE DECIMAL ASCII IN R0
1638  004424  013501                  MOV     @(5)+,%1        ;BINARY VALUE TO R1.
1639  004426  012702  004526          MOV     #ADTENP,%2      ;ADDR OF TEN POWER STRING TO R2.
1640  004432  012767  000005  000060  MOV     #5,CNVCTR       ;SET UP FOR 5 POWER CONVERSIONS.
1641  004440  012267  000060  BDCNVA: MOV     (2)+,TENPWR     ;MOVE POWER OF TEN VALUE TO TENPWR.
1642  004444  004767  000010          JSR     %7,SUBTEN       ;PERFORM CONVERSION
1643  004450  005367  000044          DEC     CNVCTR          ;DONE 5 CONVERSIONS?
1644  004454  001371                  BNE     BDCNVA          ;BRANCH IF NOT YET 5.
1645  004456  000205                  RTS     %5              ;YES, EXIT.
1646  004460  005067  000036  SUBTEN: CLR     DIGIT           ;CLEAR DIGIT
1647  004464  166701  000034  SUBTNA: SUB     TENPWR,%1       ;SUBTRACT TEN POWER FROM BINARY VALUE.
1648  004470  103403                  BCS     SUBTNB          ;BRANCH IF UNSUCCESSFUL SUBTRACTION.
1649  004472  005267  000024          INC     DIGIT
```

```
 1650  004476  000772                        BR      SUBTNA
 1651  004500  066701  000020        SUBTNB: ADD     TENPWR,X1       ;RESTORE SUBTRACTED VALUE.
 1652  004504  062767  000060  000010        ADD     #60,DIGIT       ;CONVERT (DIGIT) TO ASCII
 1653  004512  116720  000004                MOVB    DIGIT,(0)+      ;MOVE ASCII CHAR TO DECVAL FIELD.
 1654  004516  000207                        RTS     X7              ;EXIT.
 1655  004520  000000        CNVCTR: OPEN
 1656  004522  000000        DIGIT:  OPEN
 1657  004524  000000        TENPWR: OPEN
 1658  004526  023420        ADTENP: 10000.
 1659  004530  001750                10000.
 1660  004532  000144                100.
 1661  004534  000012                10.
 1662  004536  000001                1
 1663                          ;SUBROUTINE TO TYPE A LINE OF CHARACTERS
 1664  004540  012767  000114  000024 TYPLN:  MOV     #76.,TCTR       ;76 TO CHAR COUNT
 1665  004546  012704  015142        TYPLA:  MOV     #BLOCKA,X4      ;SET LINE ADDRESS IN R4.
 1666  004552  104002                TYPLB:  STALL                   ;STALL IF ALLOWED.
 1667  004554  112400                        MOVB    (4)+,X0         ;GET CHARACTER
 1668  004556  004767  177614                JSR     X7,LSPCH        ;GO OUTPUT CHARACTER.
 1669  004562  005367  000004                DEC     TCTR            ;DONE?
 1670  004566  001371                        BNE     TYPLB           ;BRANCH IF NOT DONE.
 1671  004570  000207                        RTS     X7              ;DONE. EXIT
 1672  004572  000000        TCTR:   OPEN
 1673                          ;SUBROUTINE TO TYPE LINE OF 3 CHARACTERS
 1674  004574  011667  000016        TYPL3:  MOV     @X6,TPL3A       ;DEVELOP AND SET ADDRESS OF
 1675  004600  017767  000012  000010        MOV     @TPL3A,TPL3A    ;DATA IN TPL3A.
 1676  004606  062716  000002                ADD     #2,@X6          ;SET UP EXIT.
 1677  004612  004567  000064                JSR     X5,FBF3         ;FILL BUFFER WITH 3 CHARACTERS
 1678  004616  000000        TPL3A:  OPEN
 1679  004620  042767  040000  174422        BIC     #BIT14,PRQID    ;DISABLE STALLS.
 1680  004626  004767  177706                JSR     X7,TYPLN        ;GO TYPE LINE OF CHARACTERS.
 1681  004632  000002                        RTI                     ;EXIT.
 1682  004634  112767  000015  010300 STBF:  MOVB    #15,BLOCKA      ;SUB TO SET UP BUFFER AREA.
 1683  004642  112767  000012  010273        MOVB    #12,BLOCKA+1
 1684  004650  112767  000015  010376 STBFA: MOVB    #15,BLOCKB
 1685  004656  112767  000012  010371        MOVB    #12,BLOCKB+1
 1686  004664  112767  000015  010474        MOVB    #15,BLOCKC
 1687  004672  112767  000012  010467        MOVB    #12,BLOCKC+1
 1688  004700  000207                        RTS     X7              ;EXIT
 1689                          ;SUBROUTINE TO FILL CHARACTER BUFFER WITH 3 CHARACTERS.
 1690  004702  012567  000004        FBF3:   MOV     (5)+,FBF3A
 1691  004706  004567  177426                JSR     X5,BMOVE        ;MOVE 3 CHARS TO BUFFER.
 1692  004712  000000        FBF3A:  OPEN
 1693  004714  015144                BLOCK1
 1694  004716  000003                3
 1695  004720  004567  177414        FBF3B:  JSR     X5,BMOVE        ;FILL 72 CHARACTERS BUFFER
 1696  004724  015144                BLOCK1                          ;WITH 3 CHARACTERS
 1697  004726  015147                BLOCK1+3
 1698  004730  000105                69.
 1699  004732  004567  177402                JSR     X5,BMOVE
 1700  004736  015144                BLOCK1
 1701  004740  015256                BLOCK2
 1702  004742  000110                72.
 1703  004744  000205                RTS     X5              ;EXIT
```



```
 1704                          ;SUBROUTINE TO FILL BUFFER WITH ALL CHARACTERS
 1705  004746  004567  177366        FBALL:  JSR     X5,BMOVE        ;FILL 72 CHAR BUFFER WITH
 1706  004752  014176                A                               ;ALL CHARACTERS.
 1707  004754  015144                BLOCK1
 1708  004756  000077                63.
 1709  004760  004567  177354                JSR     X5,BMOVE
 1710  004764  014176                A
 1711  004766  015243                BLOCK1+63.
 1712  004770  000011                9.
 1713  004772  004567  177342                JSR     X5,BMOVE
 1714  004776  015144                BLOCK1
 1715  005000  015256                BLOCK2
 1716  005002  000110                72.
 1717  005004  000207                RTS     X7              ;EXIT.
 1718                          ;SUB TO FILL BUFFER WITH 33 WORST CASE PATTERN.
 1719  005006  004567  177326        FW336:  JSR     X5,BMOVE        ;6 CHARACTER PATTERN TO BUFFER
 1720  005012  014162                A33WP6
 1721  005014  015144                BLOCK1
 1722  005016  000006                6
 1723  005020  004567  177314                JSR     X5,BMOVE        ;FILL BUFFER WITH PATTERN.
 1724  005024  015144                BLOCK1
 1725  005026  015152                BLOCK1+6
 1726  005030  000102                66.
 1727  005032  004567  177302                JSR     X5,BMOVE
 1728  005036  015144                BLOCK1
 1729  005040  015256                BLOCK2
 1730  005042  000110                72.
 1731  005044  000207                RTS     X7              ;EXIT
 1732                          ;SUB TO FILL BUFFER WITH 35 WORST CASE PATTERN.
 1733  005046  004567  177266        FW356:  JSR     X5,BMOVE        ;6 CHARACTER PATTERN TO BUFFER
 1734  005052  014170                A35WP6
 1735  005054  015144                BLOCK1
 1736  005056  000006                6
 1737  005060  004567  177254                JSR     X5,BMOVE        ;FILL BUFFER WITH PATTERN.
 1738  005064  015144                BLOCK1
 1739  005066  015152                BLOCK1+6
 1740  005070  000102                66.
 1741  005072  004567  177242                JSR     X5,BMOVE
 1742  005076  015144                BLOCK1
 1743  005100  015256                BLOCK2
 1744  005102  000110                72.
 1745  005104  000207                RTS     X7              ;EXIT.
 1746                          ;ROUTINE TO GET CHARACTER FROM KEYBOARD.
 1747  005106  005777  174100        GKBCR:  TST     @TKB            ;CLEAR DONE.
 1748  005112  105777  174072                TSTB    @TKS            ;WAIT FOR DONE FLAG.
 1749  005116  100375                        BPL     .-4
 1750  005120  117767  174066  174236        MOVB    @TKB,CRBUF      ;CHARACTER TO CRBUF.
 1751  005126  116723  174232                MOVB    CRBUF,(3)+      ;CHARACTER TO LINE BUFFER.
 1752  005132  116700  174226                MOVB    CRBUF,X0
 1753  005136  004767  177234                JSR     X7,LSPCH        ;ECHO CHARACTER.
 1754  005142  000207                        RTS     X7
```

```
                                        .SBTTL  PRG0 - INPUT-OUTPUT LOGIC TEST8
                                        Z=0
                                        X=-1
005144  012767  005156  174062  PRG0:   MOV     #P0T0,K8TART    ;ADDRESS OF 1ST ROUTINE TO K8TART.
005152  000167  174466                  JMP     8RSET           ;GO GET STARTED.
                                ;*****************************************************************************
005156  000000                  P0T0:   0                       ; PRG0 TEST ROUTINE 0                       *
005160  005206                          P0T1                    ;ADDRESS OF NEXT ROUTINE                    *
005162  001750                          1000.                   ;TEST ITERATION COUNT                       *
005164  005174                          P0AA                    ;SCOPE ENTRY POINT                          *
                                ;*****************************************************************************
                                ;TEST ABILITY TO REFERENCE THE KEYBOARD/READER STATUS WORD (TK8)
005166  012767  005202  172610          MOV     #P0AE,MACHER    ;SET UP MACHINE ERROR TRAP.
005174  005777  174010  P0AA:   TST     @TK8            ;REFERENCE CODER STATUS WORD.
005200  104012                          SCOPE                   ;SCOPE
005202  104003          P0AE:   ERROR                   ;ERROR. TRAPPED WHEN REFERENCING READER.
005204  104012                          SCOPE                   ;STATUS WORD (TK8).
                                ;*****************************************************************************
005206  000001          P0T1:   1                       ; PRG0 TEST ROUTINE 1                       *
005210  005236                          P0T2                    ;ADDRESS OF NEXT ROUTINE                    *
005212  001750                          1000.                   ;TEST ITERATION COUNT                       *
005214  005224                          P0BA                    ;SCOPE ENTRY POINT                          *
                                ;*****************************************************************************
                                ;TEST ABILITY TO REFERENCE THE KEYBOARD/READER BUFFER (TKB).
005216  012767  005232  172560          MOV     #P0BE,MACHER    ;SET UP MACHINE ERROR TRAP
005224  005777  173762  P0BA:   TST     @TKB            ;REFERENCE READER BUFFER.
005230  104012                          SCOPE                   ;SCOPE
005232  104003          P0BE:   ERROR                   ;ERROR. TRAPPED WHEN REFERENCING
005234  104012                          SCOPE                   ;READER BUFFER. (TKB).
                                ;*****************************************************************************
005236  000002          P0T2:   2                       ; PRG0 TEST ROUTINE 2                       *
005240  005266                          P0T3                    ;ADDRESS OF NEXT ROUTINE                    *
005242  001750                          1000.                   ;TEST ITERATION COUNT                       *
005244  005254                          P0CA                    ;SCOPE ENTRY POINT                          *
                                ;*****************************************************************************
                                ;TEST ABILITY TO REFERENCE PUNCH/PRINTER STATUS WORD (TP8).
005246  012767  005262  172530          MOV     #P0CE,MACHER    ;SETUP MACHINE ERROR TRAP.
005254  005777  173734  P0CA:   TST     @TP8            ;REFERENCE PUNCH/PRINTER STATUS WORD.
005260  104012                          SCOPE                   ;SCOPE
005262  104003          P0CE:   ERROR                   ;ERROR. TRAPPED WHEN REFERENCING
005264  104012                          SCOPE                   ;PUNCH/PRINTER STATUS WORD (TP8).
                                ;*****************************************************************************
005266  000003          P0T3:   3                       ; PRG0 TEST ROUTINE 3                       *
005270  005316                          P0T4                    ;ADDRESS OF NEXT ROUTINE                    *
005272  001750                          1000.                   ;TEST ITERATION COUNT                       *
005274  005304                          P0DA                    ;SCOPE ENTRY POINT                          *
                                ;*****************************************************************************
                                ;TEST ABILITY TO REFERENCE PUNCH/PRINTER BUFFER (TPB).
005276  012767  005312  172500          MOV     #P0DE,MACHER    ;SETUP MACHINE ERROR TRAP.
005304  005777  173706  P0DA:   TST     @TPB            ;REFERENCE PUNCH/PRINTER BUFFER.
005310  104012                          SCOPE                   ;SCOPE
005312  104003          P0DE:   ERROR                   ;ERROR. TRAPPED WHEN REFERENCING
005314  104012                          SCOPE                   ;PUNCH/PRINTER BUFFER. (TP8).
                                ;*****************************************************************************
```



```
005316  000004          P0T4:   4                       ; PRG0 TEST ROUTINE 4                       *
005320  005400                          P0T5                    ;ADDRESS OF NEXT ROUTINE                    *
005322  001750                          1000.                   ;TEST ITERATION COUNT                       *
005324  005334                          P0EA                    ;SCOPE ENTRY POINT                          *
                                ;*****************************************************************************
                                ;TEST ABILITY TO SET AND CLEAR READER/KYBD IE BIT.
005326  012767  000340  172442          MOV     #PRTY7,PSW      ;SET PRIORITY 7.
005334  052777  000100  173646  P0EA:   BIS     #BIT6,@TKS      ;SET ID BIT IN TK8.
005342  032777  000100  173640          BIT     #BIT6,@TKS      ;CHECK ID BIT IN TK8
005350  001002                          BNE     P0EB            ;BRANCH IF ID BIT IS SET.
005352  104003                          ERROR                   ;ERROR 1 ID BIT NOT SET.
005354  104012                          SCOPE
005356  042777  000100  173624  P0EB:   BIC     #BIT6,@TKS      ;CLEAR ID BIT IN TK8
005364  032777  000100  173616          BIT     #BIT6,@TKS      ;CHECK ID BIT IN TK8.
005372  001401                          BEQ     P0EC            ;BRANCH IF ID BIT IS CLEARED.
005374  104003                          ERROR                   ;ERROR. ID BIT FAILED TO CLEAR.
005376  104012          P0EC:   SCOPE                   ;SCOPE
                                ;*****************************************************************************
005400  000005          P0T5:   5                       ; PRG0 TEST ROUTINE 5                       *
005402  005442                          P0T6                    ;ADDRESS OF NEXT ROUTINE                    *
005404  000144                          100.                    ;TEST ITERATION COUNT                       *
005406  005416                          P0FA                    ;SCOPE ENTRY POINT                          *
                                ;*****************************************************************************
                                ;TEST ABILITY TO CLEAR ID BIT WITH RESET INSTRUCTION.
005410  012767  000340  172360          MOV     #PRTY7,PSW      ;SET PRIORITY 7.
005416  052777  000100  173564  P0FA:   BIS     #BIT6,@TKS      ;SET ID BIT IN TK8
005424  104011                          SRESET                  ;RESET
005426  032777  000100  173554          BIT     #BIT6,@TKS      ;TEST ID BIT.
005434  001401                          BEQ     .+4             ;BRANCH IF ID BIT IS CLEAR.
005436  104003                          ERROR                   ;ERROR. RESET FAILED TO CLEAR ID BIT.
005440  104012                          SCOPE                   ;SCOPE
                                ;*****************************************************************************
005442  000006          P0T6:   6                       ; PRG0 TEST ROUTINE 6                       *
005444  005476                          P0T7                    ;ADDRESS OF NEXT ROUTINE                    *
005446  000024                          20.                     ;TEST ITERATION COUNT                       *
005450  005454                          P0GA                    ;SCOPE ENTRY POINT                          *
                                ;*****************************************************************************
                                ;TEST THAT READER DONE BIT SETS SOMETIME AFTER RDR ENABLE.
005452  104022                          CHKA8R
005454  104023          P0GA:   RESET2
005456  005277  173526                  INC     @TK8            ;ENABLE READER.
005462  104400                          DELAYX                  ;WAIT.
005464  105777  173520                  TSTB    @TK8            ;CHECK FOR DONE
005470  100401                          BMI     .+4             ;BRANCH IF DONE BIT SET.
005472  104003                          ERROR                   ;DONE NOT SET SOMETIME AFTER RDR ENB.
005474  104012                          SCOPE                   ;SCOPE
                                ;*****************************************************************************
005476  000007          P0T7:   7                       ; PRG0 TEST ROUTINE 7                       *
005500  005530                          P0T10                   ;ADDRESS OF NEXT ROUTINE                    *
005502  001750                          1000.                   ;TEST ITERATION COUNT                       *
005504  005516                          P0HA                    ;SCOPE ENTRY POINT                          *
                                ;*****************************************************************************
                                ;TEST THAT DONE BIT READ8 RELIABLY.
005506  104022                          CHKA8R
```

```
 1863  005510  104023                        RESET2
 1864  005512  004767  174646                JSR     %7,AREAD        ;ENABLE READER, COME BACK WHEN DONE SET.
 1865  005516  105777  173466  P0MA:         TSTB    @TKS            ;TEST FOR DONE
 1866  005522  100401                        BMI     .+4             ;BRANCH IF DONE FOUND SET.
 1867  005524  104003                        ERROR                   ;ERROR, DONE BIT NOT FOUND SET.
 1868  005526  104012                        SCOPE                   ;SCOPE
 1869                                ;*****************************************************************************
 1870  005530  000010                P0T10:  10                      ; PRG0 TEST ROUTINE 10                      *
 1871  005532  005564                        P0T11                   ;ADDRESS OF NEXT ROUTINE                    *
 1872  005534  000024                        20.                     ;TEST ITERATION COUNT                       *
 1873  005536  005544                        P0IA                    ;SCOPE ENTRY POINT                          *
 1874                                ;*****************************************************************************
 1875                                ;TEST THAT RESET CLEARS DONE BIT
 1876  005540  104022                        CHKASR
 1877  005542  104023                        RESET2
 1878  005544  004767  174614        P0IA:   JSR     %7,AREAD        ;ENABLE READER, COME BACK WHEN DONE SET.
 1879  005550  104011                        SRESET                  ;ISSUE RESET.
 1880  005552  105777  173432                TSTB    @TKS            ;TEST FOR DONE BIT
 1881  005556  100001                        BPL     .+4             ;BRANCH IF DONE BIT RESET.
 1882  005560  104003                        ERROR                   ;ERROR, RESET FAILED TO CLEAR DONE.
 1883  005562  104012                        SCOPE                   ;SCOPE
 1884                                ;*****************************************************************************
 1885  005564  000011                P0T11:  11                      ; PRG0 TEST ROUTINE 11                      *
 1886  005566  005622                        P0T12                   ;ADDRESS OF NEXT ROUTINE                    *
 1887  005570  000024                        20.                     ;TEST ITERATION COUNT                       *
 1888  005572  005600                        P0JA                    ;SCOPE ENTRY POINT                          *
 1889                                ;*****************************************************************************
 1890                                ;TEST THAT REFERENCING READER DATA BUFFER CLEARS DONE
 1891  005574  104022                        CHKASR
 1892  005576  104023                        RESET2
 1893  005600  004767  174560        P0JA:   JSR     %7,AREAD        ;ENABLE READER, RETURN WHEN DONE SET.
 1894  005604  105777  173402                TSTB    @TKB            ;REFERENCE READ BUFFER.
 1895  005610  105777  173374                TSTB    @TKS            ;TEST FOR DONE BIT
 1896  005614  100001                        BPL     .+4             ;BRANCH IF DONE NOT SET.
 1897  005616  104003                        ERROR                   ;REFERENCE TO BUFFER DID NOT RESET DONE.
 1898  005620  104012                        SCOPE                   ;SCOPE
 1899                                ;*****************************************************************************
 1900  005622  000012                P0T12:  12                      ; PRG0 TEST ROUTINE 12                      *
 1901  005624  005664                        P0T13                   ;ADDRESS OF NEXT ROUTINE                    *
 1902  005626  000024                        20.                     ;TEST ITERATION COUNT                       *
 1903  005630  005634                        P0KA                    ;SCOPE ENTRY POINT                          *
 1904                                ;*****************************************************************************
 1905                                ;CHECK THAT BUSY SETS SOMETIME BEFORE DONE SETS.
 1906  005632  104022                        CHKASR
 1907  005634  104023                P0KA:   RESET2
 1908  005636  005277  173346                INC     @TKS            ;ENABLE READER.
```



```
 1909  005642  032777  004000  173340  P0KB:   BIT     #BIT11,@TKS     ;BUSY SET?
 1910  005650  001004                        BNE     P0KC            ;BR IF YES.
 1911  005652  105777  173332                TSTB    @TKS            ;NO,DONE SET?
 1912  005656  100371                        BPL     P0KB            ;BR IF NOT.
 1913  005660  104003                        ERROR                   ;BUSY BIT FAILED TO SET BEFORE DONE SET.
 1914  005662  104012                P0KC:   SCOPE
 1915                                ;*****************************************************************************
 1916  005664  000013                P0T13:  13                      ; PRG0 TEST ROUTINE 13                      *
 1917  005666  005732                        P0T14                   ;ADDRESS OF NEXT ROUTINE                    *
 1918  005670  000024                        20.                     ;TEST ITERATION COUNT                       *
 1919  005672  005676                        P0LA                    ;SCOPE ENTRY POINT                          *
 1920                                ;*****************************************************************************
 1921                                ;TEST THAT DONE IS RESET BY START BIT (WHEN BUSY BECOMES SET).
 1922  005674  104022                        CHKASR
 1923  005676  104023                P0LA:   RESET2
 1924  005700  004767  174460                JSR     %7,AREAD        ;ENABLE READER, RETURN WHEN DONE SET.
 1925  005704  005277  173300                INC     @TKS            ;ENABLE READER.
 1926  005710  032777  004000  173272        BIT     #BIT11,@TKS     ;WAIT FOR BUSY TO SET.
 1927  005716  001774                        BEQ     .-6
 1928  005720  105777  173264                TSTB    @TKS            ;TEST FOR DONE BIT.
 1929  005724  100001                        BPL     .+4             ;BRANCH IF DONE NOT SET.
 1930  005726  104003                        ERROR                   ;ERROR, START BIT FAILED TO RESET DONE.
 1931  005730  104012                        SCOPE                   ;SCOPE
 1932                                ;*****************************************************************************
 1933  005732  000014                P0T14:  14                      ; PRG0 TEST ROUTINE 14                      *
 1934  005734  006006                        P0T15                   ;ADDRESS OF NEXT ROUTINE                    *
 1935  005736  001750                        1000.                   ;TEST ITERATION COUNT                       *
 1936  005740  005760                        P0MA                    ;SCOPE ENTRY POINT                          *
 1937                                ;*****************************************************************************
 1938                                ;TEST THAT READ BUFFER CAN BE READ RELIABLY.
 1939  005742  104022                        CHKASR
 1940  005744  104023                        RESET2
 1941  005746  004767  174412                JSR     %7,AREAD        ;ENABLE READER, RETURN WHEN DONE SET.
 1942  005752  117767  173234  173405        MOVB    @TKB,CRBUF+1    ;BUFFER CONTENTS TO CRBUF+1
 1943  005760  117767  173226  173376  P0MA: MOVB    @TKB,CRBUF      ;BUFFER CONTENTS TO CRBUF
 1944  005766  126767  173372  173371        CMPB    CRBUF,CRBUF+1   ;COMPARE CONTENTS OF CRBUF AND CRBUF+1
 1945  005774  001403                        BEQ     P0MB            ;BRANCH IF SAME.
 1946  005776  016700  173362                MOV     CRBUF,%0        ;NOT SAME, ERROR, HALT WITH 1ST READ CHAR
 1947  006002  000000                        HALT                    ;IN DATA BYTES LEFT, SUBSEQUENT READ IN DATA BYTES RIGHT
 1948  006004  104012                P0MB:   SCOPE
 1949                                ;*****************************************************************************
 1950  006006  000015                P0T15:  15                      ; PRG0 TEST ROUTINE 15                      *
 1951  006010  006036                        P0T16                   ;ADDRESS OF NEXT ROUTINE                    *
 1952  006012  001750                        1000.                   ;TEST ITERATION COUNT                       *
 1953  006014  006032                        P0NA                    ;SCOPE ENTRY POINT                          *
 1954                                ;*****************************************************************************
 1955                                ;TEST THAT READER DONE BIT IS ABLE TO CAUSE INTERRUPT, IF THE INTERRUPT IS
 1956                                ;SERVICED, IT WILL HAVE OCURRED AT CORRECT VECTOR.
 1957  006016  104022                        CHKASR
 1958  006020  104006                        STRDRV                  ;SET UP READER VECTOR TO P0NC
 1959  006022  006054                        P0NC
 1960  006024  104023                        RESET2
 1961  006026  004767  174332                JSR     PC,AREAD        ;ENABLE READER, RETURN WHEN DONE SET.
 1962  006032  005077  173152        P0NA:   CLR     @TKS            ;DISABLE READER INTERRUPTS
```

```
 1963  006036  005067  171734                  CLR     PSW             ;ENABLE READER, RETURN WHEN DONE SET.
 1964  006042  052777  000100  173140          BIS     #BIT6,@TKS      ;ENABLE READER INTERRUPT,
 1965  006050  000240                          NOP
 1966  006052  104003                  AT20E:  ERROR                   ;ERROR, READER FAILED TO INTERRUPT.
 1967  006054  104012                  P0NC:   SCOPE                   ;HERE IF INTERRUPT OCCURS.
 1968                                  ;*****************************************************************************
 1969  006056  000016                  P0T16:  16                      ; PRG0 TEST ROUTINE 16                      *
 1970  006060  006136                          P0T17                   ;ADDRESS OF NEXT ROUTINE                    *
 1971  006062  001750                          1000.                   ;TEST ITERATION COUNT                       *
 1972  006064  006102                          P0DA                    ;SCOPE ENTRY POINT                          *
 1973                                  ;*****************************************************************************
 1974                                  ;TEST THAT DONE DOES NOT CAUSE INTERRUPT WITH PROCESSOR AT SAME
 1975                                  ;PRIORITY LEVEL AS THE READERS INTERRUPT REQUEST LEVEL.
 1976  006066  104022                          CHKASR
 1977  006070  104006                          STRDRV                  ;SET READER VECTOR TO P0DE.
 1978  006072  006132                          P0DE
 1979  006074  104023                          RESET2
 1980  006076  004767  174262                  JSR     %7,AREAD        ;ENABLE READER, RETURN WHEN DONE SET.
 1981  006102  005077  173102          P0DA:   CLR     @TKS            ;DISABLE READER INTERRUPTS.
 1982  006106  016767  173110  171662          MOV     TKLVL,PSW       ;SET PROCESSOR TO SAME PRIORITY AS READER'S.
 1983  006114  052777  000100  173066          BIS     #BIT6,@TKS      ;ENABLE READER INTERRUPTS.
 1984  006122  000240                          NOP                     ;NO OP.
 1985  006124  005077  173060                  CLR     @TKS            ;OK IF NO INTERRUPT OCCURS.
 1986  006130  104012                          SCOPE                   ;SCOPE
 1987  006132  104003                  P0DE:   ERROR                   ;ERROR, READER ERRONEOUSLY INTERRUPTED
```



```
 1988                                                                    ;WITH PROCESSOR AT SAME PRIORITY
 1989                                                                    ;LEVEL AS THE READER, OR THE READER
 1990  006134  104012                          SCOPE                   ;IS AT HIGHER PRIORITY THAN SPECIFIED AT TKLVL.
 1991                                  ;*****************************************************************************
 1992  006136  000017                  P0T17:  17                      ; PRG0 TEST ROUTINE 17                      *
 1993  006140  006222                          P0T20                   ;ADDRESS OF NEXT ROUTINE                    *
 1994  006142  001750                          1000.                   ;TEST ITERATION COUNT                       *
 1995  006144  006162                          P0PA                    ;SCOPE ENTRY POINT                          *
 1996                                  ;*****************************************************************************
 1997                                  ;TEST THAT DONE CAUSES INTERRUPT WITH PROCESSOR AT PRIORITY ONE LEVEL LOWER
 1998                                  ;THAN THE READER'S INTERRUPT PRIORITY LEVEL.
 1999  006146  104022                          CHKASR
 2000  006150  104006                          STRDRV                  ;SET READER INTERRUPT SERVICE TO
 2001  006152  006214                          P0PB                    ;P0PB.
 2002  006154  104023                          RESET2
 2003  006156  004767  174202                  JSR     %7,AREAD        ;ENABLE READER, RETURN WHEN DONE SET.
 2004  006162  005077  173022          P0PA:   CLR     @TKS            ;DISABLE READER INTERRUPTS
 2005  006166  016767  173030  171602          MOV     TKLVL,PSW       ;SET PROCESSOR PRIORITY ONE LEVEL LOWER
 2006  006174  162767  000040  171574          SUB     #40,PSW         ;THAN READER,(SPECIFIED AT TKLVL).
 2007  006202  052777  000100  173000          BIS     #BIT6,@TKS      ;ENABLE READER INTERRUPTS.
 2008  006210  000240                          NOP
 2009  006212  104003                          ERROR                   ;FAILED TO INTERRUPT WITH PC AT PRIORITY ONE LEVEL LOWER
 2010                                                                   ;HERE IF INTERRUPT OCCURS, OK, POP STACK TWICE
 2011  006214  005077  172770          P0PB:   CLR     @TKS            ;DISABLE READER INTERRUPTS
 2012  006220  104012                          SCOPE                   ;SCOPE
 2013                                  ;*****************************************************************************
 2014  006222  000020                  P0T20:  20                      ; PRG0 TEST ROUTINE 20                      *
 2015  006224  006326                          P0T21                   ;ADDRESS OF NEXT ROUTINE                    *
 2016  006226  001750                          1000.                   ;TEST ITERATION COUNT                       *
 2017  006230  006242                          P0QA                    ;SCOPE ENTRY POINT                          *
 2018                                  ;*****************************************************************************
 2019                                  ;TEST THAT DONE DOES NOT REINTERRUPT AFTER RTI WHEN DONE IS NOT CLEARED.
 2020  006232  104022                          CHKASR
 2021  006234  104023                          RESET2
 2022  006236  004767  174122                  JSR     %7,AREAD        ;ENABLE READER, RETURN WHEN DONE SET.
 2023  006242  104006                  P0QA:   STRDRV                  ;SET READER INTERRUPT SERVICE
 2024  006244  006276                          P0QC                    ;TO P0QC.
 2025  006246  005077  172736                  CLR     @TKS            ;DISABLE READER INTERRUPTS.
 2026  006252  052777  000100  172730          BIS     #BIT6,@TKS      ;ENABLE READER INTERRUPTS.
 2027  006260  005067  171512                  CLR     PSW             ;SET PRIORITY 0.
 2028  006264  000240                          NOP
 2029  006266  104003                          ERROR                   ;ERROR1 FAILED TO INTERRUPT
 2030  006270  005077  172714                  CLR     @TKS            ;DISABLE READER INTERRUPTS.
 2031  006274  104012                          SCOPE                   ;SCOPE
 2032  006276  012777  006316  172714  P0QC:   MOV     #P0QE,@TKVTR    ;CHANGE INTERRUPT VECTOR TO P0QE
 2033  006304  012716  006312                  MOV     #P0QD,@%6
 2034  006310  000002                          RTI                     ;RETURN FROM INTERRUPT
 2035  006312  000240                  P0QD:   NOP
 2036  006314  000401                          BR      .+4             ;OK IF NO REINTERRUPT OCCURS.
 2037  006316  104003                  P0QE:   ERROR                   ;DONE REINTERRUPTED AFTER
 2038                                                                   ;RTI WITH DONE BIT LEFT ON.
 2039  006320  005077  172664                  CLR     @TKS            ;DISABLE READER INTERRUPTS.
 2040  006324  104012                          SCOPE                   ;SCOPE.
 2041                                  ;*****************************************************************************
```

```
 2042  006326  000021                    P0T21:  21                      ; PRG0 TEST ROUTINE 21                  *
 2043  006330  006410                            P0T22                   ;ADDRESS OF NEXT ROUTINE                *
 2044  006332  001750                            1000.                   ;TEST ITERATION COUNT                   *
 2045  006334  006344                            P0RA                    ;SCOPE ENTRY POINT                      *
 2046                                    ;*************************************************************************
 2047                                    ;TEST ABILITY TO SET AND CLEAR PUNCH ID BIT
 2048  006336  012767  000340  171432            MOV     #PRTY7,PSW      ;SET PRIORITY 7,
 2049  006344  052777  000100  172642    P0RA:   BIS     #BIT6,@TPS      ;SET PUNCH ID BIT,
 2050  006352  032777  000100  172634            BIT     #BIT6,@TPS      ;CHECK PUNCH ID BIT,
 2051  006360  001002                            BNE     .+6             ;BRANCH IF PUNCH ID BIT IS SET,
 2052  006362  104003                            ERROR                   ;ERROR1, PUNCH ID BIT DID NOT SET,
 2053  006364  104012                            SCOPE
 2054  006366  042777  000100  172620            BIC     #BIT6,@TPS      ;CLEAR PUNCH ID BIT,
 2055  006374  032777  000100  172612            BIT     #BIT6,@TPS      ;CHECK PUNCH ID BIT,
 2056  006402  001401                            BEQ     .+4             ;BRANCH IF PUNCH ID BIT IS CLEAR
 2057  006404  104003                            ERROR                   ;ERROR2, PUNCH ID BIT FAILED TO CLEAR,
 2058  006406  104012                            SCOPE                   ;SCOPE
 2059                                    ;*************************************************************************
 2060  006410  000022                    P0T22:  22                      ; PRG0 TEST ROUTINE 22                  *
 2061  006412  006452                            P0T23                   ;ADDRESS OF NEXT ROUTINE                *
 2062  006414  000024                            20.                     ;TEST ITERATION COUNT                   *
 2063  006416  006426                            P0SA                    ;SCOPE ENTRY POINT                      *
 2064                                    ;*************************************************************************
 2065                                    ;TEST ABILITY TO CLEAR PUNCH ID BIT WITH RESET INSTRUCTION
 2066  006420  012767  000340  171350            MOV     #PRTY7,PSW      ;SET PRIORITY 7,
 2067  006426  052777  000100  172560    P0SA:   BIS     #BIT6,@TPS      ;SET PUNCH ID BIT,
 2068  006434  104011                            SRESET                  ;RESET
 2069  006436  032777  000100  172550            BIT     #BIT6,@TPS      ;CHECK PUNCH ID BIT,
 2070  006444  001401                            BEQ     .+4             ;BRANCH IF PUNCH ID BIT IS CLEAR,
 2071  006446  104003                            ERROR                   ;ERROR, RESET FAILED TO CLEAR PUNCH ID BIT,
 2072  006450  104012                            SCOPE                   ;SCOPE
 2073                                    ;*************************************************************************
 2074  006452  000023                    P0T23:  23                      ; PRG0 TEST ROUTINE 23                  *
 2075  006454  006526                            P0T24                   ;ADDRESS OF NEXT ROUTINE                *
 2076  006456  001750                            1000.                   ;TEST ITERATION COUNT                   *
 2077  006460  006462                            P0TA                    ;SCOPE ENTRY POINT                      *
 2078                                    ;*************************************************************************
 2079                                    ;TEST ABILITY TO SET AND CLEAR THE PUNCH MAINTENANCE BIT
 2080  006462  052777  000004  172524    P0TA:   BIS     #BIT2,@TPS      ;SET MAINTANCE BIT,
 2081  006470  032777  000004  172516            BIT     #BIT2,@TPS      ;CHECK MAINTENANCE BIT
 2082  006476  001002                            BNE     .+6             ;BRANCH IF MAINTENANCE BIT SET,
 2083  006500  104003                            ERROR                   ;ERROR1, MAINTENANCE BIT FAILED TO SET,
 2084  006502  104012                            SCOPE
 2085  006504  042777  000004  172502            BIC     #BIT2,@TPS      ;CLEAR MAINTENANCE BIT,
 2086  006512  032777  000004  172474            BIT     #BIT2,@TPS      ;CHECK MAINTENANCE BIT
 2087  006520  001401                            BEQ     .+4             ;BRANCH IF MAINTENANCE BIT IS CLEAR,
 2088  006522  104003                            ERROR                   ;ERROR2, MAINTENANCE BIT FAILED TO CLEAR,
 2089  006524  104012                            SCOPE                   ;SCOPE
 2090                                    ;*************************************************************************
 2091  006526  000024                    P0T24:  24                      ; PRG0 TEST ROUTINE 24                  *
 2092  006530  006562                            P0T25                   ;ADDRESS OF NEXT ROUTINE                *
 2093  006532  000024                            20.                     ;TEST ITERATION COUNT                   *
 2094  006534  006536                            P0UA                    ;SCOPE ENTRY POINT                      *
 2095                                    ;*************************************************************************
```



```
 2096                                    ;TEST THAT RESET INSTRUCTION CLEARS THE MAINTENANCE BIT,
 2097  006536  052777  000004  172450    P0UA:   BIS     #BIT2,@TPS      ;SET MAINTENANCE BIT,
 2098  006544  104011                            SRESET                  ;ISSUE RESET
 2099  006546  032777  000004  172440            BIT     #BIT2,@TPS      ;CHECK MAINTENANCE BIT
 2100  006554  001401                            BEQ     .+4             ;BRANCH IF MAINTENANCE BIT CLEAR,
 2101  006556  104003                            ERROR                   ;ERROR, RESET FAILED TO CLEAR
 2102  006560  104012                            SCOPE                   ;THE MAINTENANCE BIT, SCOPE,
 2103                                    ;*************************************************************************
 2104  006562  000025                    P0T25:  25                      ; PRG0 TEST ROUTINE 25                  *
 2105  006564  006604                            P0T26                   ;ADDRESS OF NEXT ROUTINE                *
 2106  006566  001750                            1000.                   ;TEST ITERATION COUNT                   *
 2107  006570  006572                            P0VA                    ;SCOPE ENTRY POINT                      *
 2108                                    ;*************************************************************************
 2109                                    ;TEST THAT RESET SETS THE PUNCH READY BIT, AND THAT READY CAN BE READ RELIABLY,
 2110  006572  105777  172416    P0VA:   TSTB    @TPS            ;CHECK PUNCH READY,
 2111  006576  100401                    BMI     .+4             ;BRANCH IF PUNCH READY IS SET,
 2112  006600  104003                    ERROR                   ;ERROR, RESET FAILED TO SET READY, OR FAILED TO READ  IT
 2113  006602  104012                    SCOPE                   ;SCOPE
 2114                                    ;*************************************************************************
 2115  006604  000026                    P0T26:  26                      ; PRG0 TEST ROUTINE 26                  *
 2116  006606  006634                            P0T27                   ;ADDRESS OF NEXT ROUTINE                *
 2117  006610  000024                            20.                     ;TEST ITERATION COUNT                   *
 2118  006612  006614                            P0WA                    ;SCOPE ENTRY POINT                      *
 2119                                    ;*************************************************************************
 2120                                    ;TEST THAT PUNCH READY RESETS BY LOADING PUNCH BUFFER,
 2121  006614  104023                    P0WA:   RESET2
 2122  006616  005077  172374                    CLR     @TPB            ;LOAD PUNCH BUFFER
 2123  006622  105777  172366                    TSTB    @TPS            ;CHECK PUNCH READY BIT,
 2124  006626  100001                            BPL     .+4             ;BR IF PUNCH READY IS CLEAR,
 2125  006630  104003                            ERROR                   ;ERROR, BUFFER LOAD FAILED TO CLEAR READY,
 2126  006632  104012                            SCOPE                   ;SCOPE
 2127                                    ;*************************************************************************
 2128  006634  000027                    P0T27:  27                      ; PRG0 TEST ROUTINE 27                  *
 2129  006636  006670                            P0T30                   ;ADDRESS OF NEXT ROUTINE                *
 2130  006640  000024                            20.                     ;TEST ITERATION COUNT                   *
 2131  006642  006644                            P0XA                    ;SCOPE ENTRY POINT                      *
 2132                                    ;*************************************************************************
 2133                                    ;TEST THAT BYTE LOAD OF PUNCH BUFFER +1 DOES NOT RESET READY,
 2134  006644  104023                    P0XA:   RESET2
 2135  006646  016700  172344                    MOV     TPB,%0
 2136  006652  005200                            INC     %0
 2137  006654  105010                            CLRB    @%0             ;BYTE LOAD PUNCH BUFFER+1
 2138  006656  105777  172332                    TSTB    @TPS            ;CHECK PUNCH READY BIT
 2139  006662  100401                            BMI     .+4             ;BRANCH IF PUNCH READY STILL SET,
 2140  006664  104003                            ERROR                   ;ERROR, BYTE LOAD OF PUNCH BUFFER+1
 2141  006666  104012                            SCOPE                   ;CLEARED READY, SCOPE
 2142                                    ;*************************************************************************
 2143  006670  000030                    P0T30:  30                      ; PRG0 TEST ROUTINE 30                  *
 2144  006672  006722                            P0T31                   ;ADDRESS OF NEXT ROUTINE                *
 2145  006674  000024                            20.                     ;TEST ITERATION COUNT                   *
 2146  006676  006700                            P0YA                    ;SCOPE ENTRY POINT                      *
 2147                                    ;*************************************************************************
 2148                                    ;TEST THAT PUNCH BECOMES READY SOMETIME AFTER BUFFER LOAD,
 2149  006700  104023                    P0YA:   RESET2
```

```
2150  006702  005077  172310                  CLR     @TPB            ;LOAD PUNCH BUFFER.
2151  006706  104400                          DELAYX                  ;WAIT.
2152  006710  105777  172300                  TSTB    @TPS            ;CHECK PUNCH READY BIT.
2153  006714  100401                          BMI     .+4             ;BRANCH IF PUNCH READY IS SET.
2154  006716  104003                          ERROR                   ;READY NOT SET SOMETIME AFTER BUFFER LOAD.
2155  006720  104012                          SCOPE                   ;SCOPE
2156                                  ;*************************************************************************
2157  006722  000031                  P0T31:  31                      ; PRG0 TEST ROUTINE 31                  *
2158  006724  006762                          P0T32                   ;ADDRESS OF NEXT ROUTINE                *
2159  006726  001750                          1000.                   ;TEST ITERATION COUNT                   *
2160  006730  006736                          P0ZA                    ;SCOPE ENTRY POINT                      *
2161                                  ;*************************************************************************
2162                                  ;TEST THAT PUNCH READY BIT CAN CAUSE AN INTERRUPT, IF THE INTERRUPT
2163                                  ;IS SERVICED, IT WILL HAVE OCCURRED AT THE CORRECT VECTOR.
2164  006732  104007                          STPCHV                  ;SET PUNCH INTERRUPT SERVICE
2165  006734  006760                          P0ZB                    ;TO P0ZB
2166  006736  005077  172252          P0ZA:   CLR     @TPS            ;DISABLE PUNCH INTERRUPTS
2167  006742  005067  171030                  CLR     PSW             ;SET PRIORITY 0.
2168  006746  052777  000100  172240          BIS     #BIT6,@TPS      ;ENABLE PUNCH INTERRUPTS.
2169  006754  000240                          NOP
2170  006756  104003                          ERROR                   ;PUNCH READY FAILED TO CAUSE
2171  006760  104012          P0ZB:   SCOPE                   ;INTERRUPT. SCOPE
2172                                  ;*************************************************************************
2173  006762  000032                  P0T32:  32                      ; PRG0 TEST ROUTINE 32                  *
2174  006764  007032                          P0T33                   ;ADDRESS OF NEXT ROUTINE                *
2175  006766  001750                          1000.                   ;TEST ITERATION COUNT                   *
2176  006770  006776                          P0AAA                   ;SCOPE ENTRY POINT                      *
2177                                  ;*************************************************************************
2178                                  ;TEST THAT PUNCH READY DOES NOT CAUSE AN INTERRUPT WITH PROCESSOR
2179                                  ;AT SAME PRIORITY LEVEL AS THE PUNCH INTERRUPT REQUEST LEVEL.
2180  006772  104007                          STPCHV                  ;SET PUNCH INTERRUPT SERVICE
2181  006774  007026                          P0AAE                   ;TO P0AAE.
2182  006776  016767  172224  170772  P0AAA:  MOV     TPLVL,PSW       ;SET PROCESSOR TO SAME PRIORITY AS PUNCH.
2183  007004  005077  172204                  CLR     @TPS            ;DISABLE PUNCH INTERRUPTS.
2184  007010  052777  000100  172176          BIS     #BIT6,@TPS      ;ENABLE PUNCH INTERRUPTS.
2185  007016  000240                          NOP
2186  007020  005077  172170          P0AAB:  CLR     @TPS            ;OK IF NO INTERRUPT OCCURS.
2187  007024  104012                          SCOPE                   ;SCOPE
2188  007026  104003                  P0AAE:  ERROR                   ;ERROR, PUNCH INTERRUPTED WITH PROCESSOR
2189  007030  000773                          BR      P0AAB           ;SET TO SAVE PRIORITY AS THE PUNCH.
2190                                  ;*************************************************************************
2191  007032  000033                  P0T33:  33                      ; PRG0 TEST ROUTINE 33                  *
2192  007034  007106                          P0T34                   ;ADDRESS OF NEXT ROUTINE                *
2193  007036  001750                          1000.                   ;TEST ITERATION COUNT                   *
2194  007040  007046                          P0BAA                   ;SCOPE ENTRY POINT                      *
2195                                  ;*************************************************************************
2196                                  ;TEST THAT THE PUNCH INTERRUPTS WITH PROCESSOR AT PRIORITY ONE LEVEL LOWER
2197                                  ;THAN THE PUNCH PRIORITY.
2198  007042  104007                          STPCHV                  ;SET PUNCH INTERRUPT SERVICE
2199  007044  007100                          P0BAC                   ;TO P0BAC.
2200  007046  005077  172142          P0BAA:  CLR     @TPS            ;DISABLE PUNCH INTERRUPTS
2201  007052  016767  172150  170716          MOV     TPLVL,PSW       ;SET PROCESSOR PRIORITY ONE LEVEL
2202  007060  162767  000040  170710          SUB     #40,PSW         ;LOWER THAN PUNCH PRIORITY
2203  007066  052777  000100  172120          BIS     #BIT6,@TPS      ;ENABLE PUNCH INTERRUPTS
```



```
2204  007074  000240                          NOP
2205  007076  104003                          ERROR                   ;ERROR, PUNCH FAILED TO INTERRUPT.
2206  007100  005077  172110          P0BAC:  CLR     @TPS            ;THE STOCK TWICE, DISABLE PUNCH INTERRUPT
2207  007104  104012                          SCOPE                   ;SCOPE
2208                                  ;*************************************************************************
2209  007106  000034                  P0T34:  34                      ; PRG0 TEST ROUTINE 34                  *
2210  007110  007202                          P0T35                   ;ADDRESS OF NEXT ROUTINE                *
2211  007112  001750                          1000.                   ;TEST ITERATION COUNT                   *
2212  007114  007116                          P0CAA                   ;SCOPE ENTRY POINT                      *
2213                                  ;*************************************************************************
2214                                  ;TEST THAT PUNCH READY DOES NOT REINTERRUPT AFTER RTI WHEN READY
2215                                  ;BIT HAS NOT BEEN RESET.
2216  007116  104007                  P0CAA:  STPCHV                  ;SET PUNCH INTERRUPT SERVICE TO
2217  007120  007152                          P0CAC                   ;TO P0CAC.
2218  007122  005077  172066                  CLR     @TPS            ;DISABLE PUNCH INTERRUPTS
2219  007126  005067  170644                  CLR     PSW             ;SET PROCESSOR PRIORITY TO 0
2220  007132  052777  000100  172054          BIS     #BIT6,@TPS      ;ENABLE PUNCH INTERRUPTS
2221  007140  000240                          NOP
2222  007142  104003                          ERROR                   ;ERROR 1. PUNCH FAILED TO INTERRUPT.
2223  007144  005077  172044                  CLR     @TPS            ;DISABLE PUNCH INTERRUPT.
2224  007150  104012                          SCOPE                   ;SCOPE
2225  007152  012777  007172  172044  P0CAC:  MOV     #P0CAE,@TPVTR   ;HERE IF INTERRUPT OCCURS, CHANGE
2226  007160  012716  007166                  MOV     #P0CAD,@X6      ;PUNCH VECTOR TO P0CAE AND EXIT
2227  007164  000002                          RTI                     ;INTERRUPT
2228  007166  000240                  P0CAD:  NOP                     ;OK IF NO REINTERRUPT OCCURS
2229  007170  000401                          BR      P0CAF
2230  007172  104003                  P0CAE:  ERROR                   ;ERR 2, PUNCH REINTERRUPTED AFTER
2231  007174  005077  172014          P0CAF:  CLR     @TPS            ;RTI WITH READY BIT LEFT ON
2232  007200  104012                          SCOPE                   ;SCOPE
2233                                  ;*************************************************************************
2234  007202  000035                  P0T35:  35                      ; PRG0 TEST ROUTINE 35                  *
2235  007204  007260                          P0T36                   ;ADDRESS OF NEXT ROUTINE                *
2236  007206  001750                          1000.                   ;TEST ITERATION COUNT                   *
2237  007210  007216                          P0DAA                   ;SCOPE ENTRY POINT                      *
2238                                  ;*************************************************************************
2239                                  ;TEST THAT THE PUNCH INTERRUPTS IMMEDIATELY UPON LOWERING
2240                                  ;PROCESSOR PRIORITY TO 0.
2241  007212  104007                          STPCHV                  ;SET PUNCH INTERRUPT
2242  007214  007252                          P0DAC                   ;TO P0DAC.
2243  007216  012767  000340  170552  P0DAA:  MOV     #PRTY7,PSW      ;SET PROCESSOR PRIORITY TO 7.
2244  007224  005077  171764                  CLR     @TPS            ;DISABLE PUNCH INTERRUPTS
2245  007230  052777  000100  171756          BIS     #BIT6,@TPS      ;ENABLE PUNCH INTERRUPTS
2246  007236  005067  170534                  CLR     PSW             ;LOWER PROCESSOR PRIORITY TO 0.
2247  007242  012767  000340  170526          MOV     #PRTY7,PSW      ;RAISE PRIORITY TO 7.
2248  007250  104003                          ERROR                   ;ERROR, PUNCH FAILED TO INTERRUPT
2249                                                                  ;IMMEDIATELY AFTER CP PRIORITY WAS SET TO 0.
2250  007252  005077  171736          P0DAC:  CLR     @TPS            ;DISABLE PUNCH INTERRUPTS
2251  007256  104012                          SCOPE                   ;SCOPE
2252                                  ;*************************************************************************
2253  007260  000036                  P0T36:  36                      ; PRG0 TEST ROUTINE 36                  *
2254  007262  007342                          P0T37                   ;ADDRESS OF NEXT ROUTINE                *
2255  007264  000024                          20.                     ;TEST ITERATION COUNT                   *
2256  007266  007276                          P0EAA                   ;SCOPE ENTRY POINT                      *
2257                                  ;*************************************************************************
```

```
                                        ;TEST FOR CORRECT OPERATION OF THE WAIT INSTRUCTION.  A WAIT INSTRUCTION
                                        ;IS PERFORMED WHILE WAITING FOR A PUNCH INTERRUPT.  WHEN THE INTERRUPT
                                        ;OCCURS, THE SERVICE ROUTINE CHANGES THE WAIT INSTRUCTION TO AN ERROR
                                        ;CALL AND THEN EXITS THE INTERRUPT WITH AN RTI.  EXITING THE INTERRUPT
                                        ;SHOULD RETURN CONTROL TO THE INSTRUCTION FOLLOWING THE WAIT INSTRUCTION.
                                        ;IF CONTROL IS INSTEAD RETURNED TO THE SAME LOCATION WHERE THE WAIT
                                        ;INSTRUCTION WAS LOCATED AN ERROR CALL WILL OCCUR, INDICATING A FAILURE
                                        ;OF THE WAIT INSTRUCTION.
007270  104023                                RESET2
007272  104007                                STPCHV                  ;SET PUNCH INTERRUPT SERVICE
007274  007332                                P0EAC                   ;TO P0EAC.
007276  012767  000001  000016  P0EAA:  MOV     #WAIT,P0EAB             ;MOVE WAIT INSTRUCTION TO P0EAB
007304  005077  171706                        CLR     @TPB            ;LOAD PUNCH BUFFER (ENABLES PUNCH)
007310  052777  000100  171676                BIS     #BIT6,@TPS      ;ENABLE PUNCH INTERRUPTS
007316  005067  170434                        CLR     PSW             ;SET PRIORITY 0.
007322  000000                        P0EAB:  OPEN                    ;THIS LOCATION CAN BE EITHER
                                                                        ;A WAIT INSTRUCTION OR AN ERROR CALL.
                                                                        ;IF AN ERROR CALL IS EXECUTED, IT
                                                                        ;INDICATES A FAILURE OF THE WAIT INSTRUCTION.
007324  005077  171664                        CLR     @TPS            ;DISABLE PUNCH INTERRUPTS
007330  104012                                SCOPE                   ;SCOPE
007332  012767  104003  177762  P0EAC:  MOV     #ERROR,P0EAB            ;MOVE ERROR CALL TO P0EAB.
007340  000002                                RTI                     ;EXIT INTERRUPT.
                                        ;*************************************************************************
007342  000037                        P0T37:  37                      ; PRG0 TEST ROUTINE 37                  *
007344  007402                                P0T40                   ;ADDRESS OF NEXT ROUTINE                *
007346  000024                                20.                     ;TEST ITERATION COUNT                   *
007350  007354                                P0FAA                   ;SCOPE ENTRY POINT                      *
                                        ;*************************************************************************
                                        ;TEST THAT LOADING THE PUNCH BUFFER WITH THE MAINTENANCE BIT SET
                                        ;CAUSES THE READER DONE BIT TO SET SOMETIME AFTER.
007352  104023                                RESET2
007354  052777  000004  171632  P0FAA:  BIS     #BIT2,@TPS              ;SET MAINTENANCE BIT
007362  005077  171630                        CLR     @TPB            ;LOAD PUNCH BUFFER
007366  104400                                DELAYX                  ;WAIT.
007370  105777  171614                        TSTB    @TKS            ;TEST READER DONE BIT
007374  100401                                BMI     .+4             ;BRANCH IF READER DONE BIT SET.
007376  104003                                ERROR                   ;ERROR. SOMETIME AFTER PUNCH
                                                                        ;BUFFER LOAD WITH MAINTENANCE BIT
                                                                        ;SET THE READER DONE BIT WAS NOT SET
007400  104012                                SCOPE                   ;SCOPE
                                        ;*************************************************************************
007402  000040                        P0T40:  40                      ; PRG0 TEST ROUTINE 40                  *
007404  007472                                P0T41                   ;ADDRESS OF NEXT ROUTINE                *
007406  000024                                20.                     ;TEST ITERATION COUNT                   *
007410  007416                                P0GAA                   ;SCOPE ENTRY POINT                      *
                                        ;*************************************************************************
                                        ;TEST THAT CLEARING PUNCH READY AND/OR IE BIT CLEARS PUNCH INTERRUPT REQUEST.
007412  104007                                STPCHV                  ;SET PUNCH VECTOR TO P0GAB.
007414  007466                                P0GAB
007416  104023                        P0GAA:  RESET2
007420  012767  000340  170350                MOV     #PRTY7,PSW      ;SET PRIORITY 7.
007426  052777  000100  171560                BIS     #BIT6,@TPS      ;ENABLE PUNCH INTERRUPTS.
007434  005077  171556                        CLR     @TPB            ;OUTPUT CHAR.
```



```
007440  105777  171550                        TSTB    @TPS            ;WAIT FOR PUNCH READY.
007444  100375                                BPL     .-4
007446  005077  171542                        CLR     @TPS            ;DISABLE PUNCH INTERRUPTS.
007452  005077  171540                        CLR     @TPB            ;LOAD BUFFER TO CLEAR PUNCH READY.
007456  005067  170314                        CLR     PSW             ;SET PRIORITY 0.
007462  000240                                NOP
007464  104012                                SCOPE                   ;OK IF NO INTERRUPT OCCURS.
007466  104003                        P0GAB:  ERROR                   ;READY CLEAR AND/OR IE BIT CLEAR DID NOT
007470  104012                                SCOPE                   ;PREVENT PUNCH/PRINTER INTERRUPT.
                                                                        ;(INTERRUPT REQUEST DID NOT CLEAR.)
                                        ;*************************************************************************
007472  000041                        P0T41:  41                      ; PRG0 TEST ROUTINE 41                  *
007474  007570                                P0T42                   ;ADDRESS OF NEXT ROUTINE                *
007476  000024                                20.                     ;TEST ITERATION COUNT                   *
007500  007506                                P0HAA                   ;SCOPE ENTRY POINT                      *
                                        ;*************************************************************************
                                        ;TEST THAT CLEARING READER DONE AND/OR IE BIT CLEARS READER INTERRUPT REQUEST.
007502  104006                                STRDRV                  ;SET READER VECTOR TO P0HAB.
007504  007564                                P0HAB
007506  104023                        P0HAA:  RESET2
007510  012767  000340  170260                MOV     #PRTY7,PSW      ;SET PRIORITY 7.
007516  052777  000004  171470                BIS     #BIT2,@TPS      ;SET MAINTENANCE MODE.
007524  005077  171466                        CLR     @TPB            ;OUTPUT CHAR.
007530  052777  000100  171452                BIS     #BIT6,@TKS      ;ENABLE READER INTERRUPTS.
007536  105777  171446                        TSTB    @TKS            ;WAIT FOR READER DONE.
007542  100375                                BPL     .-4
007544  005077  171440                        CLR     @TKS            ;DISABLE READER INTERRUPTS.
007550  005777  171436                        TST     @TKB            ;CLEAR READER DONE.
007554  005067  170216                        CLR     PSW             ;SET PRIORITY 0.
007560  000240                                NOP
007562  104012                                SCOPE                   ;OK IF NO INTERRUPT OCCURS.
007564  104003                        P0HAB:  ERROR                   ;DONE CLEARED AND/OR IE CLEARED DID NOT
007566  104012                                SCOPE                   ;PREVENT READER INTERRUPT.
                                        ;*************************************************************************
007570  000042                        P0T42:  42                      ; PRG0 TEST ROUTINE 42                  *
007572  007620                                P0T43                   ;ADDRESS OF NEXT ROUTINE                *
007574  001000                                1000                    ;TEST ITERATION COUNT                   *
007576  007600                                P0JAA                   ;SCOPE ENTRY POINT                      *
                                        ;*************************************************************************
                                        ;TEST THE DL11A,B KEYBOARD JUMPERS ARE CUT PROPERLY
007600  012777  173476  171402  P0JAA:  MOV     #173476,@TKS            ;ATTEMPT TO SET NON-SETABLE BITS
007606  005777  171376                        TST     @TKS            ;DID ANY SET?
007612  001401                                BEQ     .+4             ;BR IF NO
007614  104003                                ERROR
007616  104012                                SCOPE
                                        ;*************************************************************************
007620  000043                        P0T43:  43                      ; PRG0 TEST ROUTINE 43                  *
007622  177777                                P0TLST                  ;ADDRESS OF NEXT ROUTINE                *
007624  001000                                1000                    ;TEST ITERATION COUNT                   *
007626  007630                                P0KAA                   ;SCOPE ENTRY POINT                      *
                                        ;*************************************************************************
                                        ;TEST THE DL11A,B PRINTER JUMPERS ARE CUT PROPERLY
007630  012777  177473  171356  P0KAA:  MOV     #177473,@TPS            ;ATTEMPT TO SET NON-SETABLE BITS
007636  022777  000200  171350                CMP     #BIT7,@TPS      ;DID ANY SET?
```

```
 2366  007644  001401                        BEQ     .+4             ;BR IF NO
 2367  007646  104003                        ERROR
 2368  007650  104012                        SCOPE
```



```
 2369                                        .SBTTL  PRG1 READER TEST
 2370          000001                        Z=1
 2371          177777                        X=-1
 2372  007652  012767  007706  171354  PRG1:  MOV     #P1T0,KSTART    ;SET ADDRESS OF FIRST ROUTINE
 2373  007660  012767  177760  173704        MOV     #177760,RCMSK
 2374  007666  012767  177400  173310        MOV     #177400,STLMSK  ;SET STALL LIMIT.
 2375  007674  052767  040000  171346        BIS     #BIT14,PRGID    ;ALLOW STALLS
 2376  007702  000167  171736                JMP     SRSET           ;GO GET STARTED.
 2377                                ;*****************************************************************************
 2378  007706  000000                P1T0:  0                       ; PRG1 TEST ROUTINE 0                       *
 2379  007710  007734                        P1T1                    ;ADDRESS OF NEXT ROUTINE                    *
 2380  007712  003720                        2000.                   ;TEST ITERATION COUNT                       *
 2381  007714  007722                        P1AA                    ;SCOPE ENTRY POINT                          *
 2382                                ;*****************************************************************************
 2383                                ;READ AND CHECK 2000 CHARACTERS OF SPECIAL BINARY COUNT PATTERN, FULL SPEED.
 2384  007716  004767  173714                JSR     %7,BSYNC        ;SYNC READER; SET ERROR COUNTER.
 2385  007722  004767  172644        P1AA:  JSR     %7,BREAD        ;GO READ CHARACTER
 2386  007726  004767  173644                JSR     %7,BCHECK       ;GO CHECK CHARACTER READ.
 2387  007732  104012                        SCOPE                   ;SCOPE
 2388                                ;*****************************************************************************
 2389  007734  000001                P1T1:  1                       ; PRG1 TEST ROUTINE 1                       *
 2390  007736  007764                        P1T2                    ;ADDRESS OF NEXT ROUTINE                    *
 2391  007740  001750                        1000.                   ;TEST ITERATION COUNT                       *
 2392  007742  007750                        P1BA                    ;SCOPE ENTRY POINT                          *
 2393                                ;*****************************************************************************
 2394                                ;READ AND CHECK 1000 CHARACTERS OF SPECIAL BINARY COUNT PATTERN,
 2395                                ;RANDOM STALL BETWEEN CHARACTERS.
 2396  007744  004767  173666                JSR     %7,BSYNC        ;SYNC READER; SET ERROR COUNTER
 2397  007750  104002                P1BA:  STALL                   ;RANDOM STALL
 2398  007752  004767  172614                JSR     %7,BREAD        ;GO READ CHARACTER
 2399  007756  004767  173614                JSR     %7,BCHECK       ;GO CHECK CHARACTER READ
 2400  007762  104012                        SCOPE                   ;SCOPE
 2401                                ;*****************************************************************************
 2402  007764  000002                P1T2:  2                       ; PRG1 TEST ROUTINE 2                       *
 2403  007766  177777                        P1TLST                  ;ADDRESS OF NEXT ROUTINE                    *
 2404  007770  000310                        200.                    ;TEST ITERATION COUNT                       *
 2405  007772  010000                        P1CA                    ;SCOPE ENTRY POINT                          *
 2406                                ;*****************************************************************************
 2407                                ;READ AND CHECK 200 CHARACTER GROUPS OF SPECIAL BINARY COUNT PATTERN,
 2408                                ;RANDOM LENGTH
 2409                                ;GROUPS (BETWEEN 1 AND 15), RANDOM STALL BETWEEN GROUPS (0 TO 127 MSECS).
 2410  007774  004767  173636                JSR     %7,BSYNC        ;SYNC READER; SET ERROR COUNTER.
 2411  010000  004767  173546        P1CA:  JSR     %7,GRCNT        ;GENERATE RANDOM CHARACTER COUNT.
 2412  010004  104002                        STALL                   ;RANDOM STALL (0 TO 127 MSECS)
 2413  010006  004767  172560        P1CC:  JSR     %7,BREAD        ;GO READ CHARATER
 2414  010012  004767  173560                JSR     %7,BCHECK       ;GO CHECK CHARACTER READ
 2415  010016  005367  173552                DEC     RNCNT           ;DECREMENT RANDOM CHAR COUNT
 2416  010022  001371                        BNE     P1CC            ;GO READ AGAIN IF COUNT NOT 0.
 2417  010024  104012                        SCOPE                   ;SCOPE
```

```
                                        .SBTTL  PRG2-PRINTER TEST8
                    000002              Z=2
                    177777              %%=-1
010026  012767  010060  171200  PRG2:   MOV     #P2T0,K$TART    ;SET ADDRESS IF 1ST ROUTINE.
010034  052767  000200  171206          BIS     #BIT7,PRGID
010042  012767  177600  173134          MOV     #177600,STLMSK  ;SET STALL LIMIT
010050  004767  174560                  JSR     %7,8TBF         ;SET UP BUFFER AREA.
010054  000167  171564                  JMP     SRSET           ;GO GET STARTED.
                                ;*****************************************************************************
010060  000000                  P2T0:   0                       ; PRG2 TEST ROUTINE 0                   *
010062  010206                          P2T1                    ;ADDRESS OF NEXT ROUTINE                *
                                ;*****************************************************************************
                                ;CARRIAGE RETURN TEST.
010064  104000                          TYPE                    ;TYPE TITLE.
010066  014366                          CRTST
010070  012767  000122  171264          MOV     #82.,RCNT
010076  104015                          CK37
010100  162767  000011  171254          SUB     #9.,RCNT
010106  112700  000134                  MOVB    #'\,%0          ;PRINT
010112  004767  174260                  JSR     %7,L8PCH        ;"\"
010116  016767  171240  171260          MOV     RCNT,CTRA       ;RCNT TO CTRA
010124  005367  171254          CT0A:   DEC     CTRA            ;DECREMENT CTRA
010130  001001                          BNE     CT0B            ;BRANCH IF NOT 0
010132  104012                          SCOPE                   ;0, SCOPE
010134  016767  171244  171244  CT0B:   MOV     CTRA,CTRB       ;SPACE COUNT TO CTRB.
010142  112700  000040          CT0C:   MOVB    #40,%0
010146  004767  174224                  JSR     %7,L8PCH        ;SPACE.
010152  005367  171230                  DEC     CTRB            ;DECREMENT CTRB.
010156  001371                          BNE     CT0C            ;BRANCH IF NOT DONE SPACING.
010160  112700  000015                  MOVB    #15,%0
010164  004767  174206                  JSR     %7,L8PCH        ;CARRIAGE RETURN.
010170  004767  174202                  JSR     %7,L8PCH        ;DUMMY CYCLE.
010174  112700  000057                  MOVB    #'/,%0
010200  004767  174172                  JSR     %7,L8PCH        ;PRINT "/".
010204  000747                          BR      CT0A
                                ;*****************************************************************************
010206  000001                  P2T1:   1                       ; PRG2 TEST ROUTINE 1                   *
010210  010272                          P2T2                    ;ADDRESS OF NEXT ROUTINE                *
                                ;*****************************************************************************
                                ;RIGHT MARGIN TEST
010212  104000                          TYPE                    ;TYPE TITLE
010214  014417                          RMTST
010216  012767  000016  171160          MOV     #14.,CTRA       ;SET UP FOR 33/35
010224  012767  014347  000034          MOV     #RM33B,RMB
010232  104015                          CK37                    ;37
010234  000406                          BR      CT1A            ;NO
010236  012767  000017  171140          MOV     #15.,CTRA       ;YES,
010244  012767  014353  000014          MOV     #RM37A,RMB      ;SET UP FOR 37.
010252  104000          CT1A:   TYPE                    ;TYPE----I
010254  014341                          RM33A
010256  005367  171122                  DEC     CTRA            ;DONE N TIMES,
010262  001373                          BNE     CT1A            ;BRANCH IF NOT N TIMES
010264  104000                          TYPE                    ;TYPE-I-.
010266  000000                  RMB:    OPEN
```



```
010270  104012                          SCOPE                   ;SCOPE.
                                ;*****************************************************************************
010272  000002                  P2T2:   2                       ; PRG2 TEST ROUTINE 2                   *
010274  010426                          P2T3                    ;ADDRESS OF NEXT ROUTINE                *
                                ;*****************************************************************************
                                ;SPACE TEST
010276  104000                          TYPE                    ;TYPE TITLE.
010300  014445                          SPTST
010302  012767  000044  171074          MOV     #36.,CTRA       ;33/35 COUNT TO CTRA.
010310  104000          CT2A:   TYPE                    ;TYPE SPACE,\.
010312  014363                          SPTSTC
010314  005367  171064                  DEC     CTRA            ;DONE TIMES SET IN CTRA?
010320  001373                          BNE     CT2A            ;BRANCH IF NOT DONE
010322  012767  000044  171054          MOV     #36.,CTRA       ;SET UP CTRA COUNT FOR 33/35
010330  012767  000001  171050  CT2B:   MOV     #1,CTRB
010336  016767  171044  171044  CT2C:   MOV     CTRB,CTRC
010344  112700  000015                  MOVB    #15,%0          ;CARRIAGE RETURN.
010350  004767  174022                  JSR     %7,L8PCH
010354  004767  174016                  JSR     %7,L8PCH        ;DUMMY CYCLE.
010360  112700  000040          CT2D:   MOVB    #40,%0          ;SPACE NUMBER OF TIMES
010364  004767  174006                  JSR     %7,L8PCH        ;SET IN CTRC.
010370  005367  171014                  DEC     CTRC            ;DONE SPACING.
010374  001371                          BNE     CT2D            ;BRANCH IF NOT DONE SPACING.
010376  112700  000057                  MOVB    #'/,%0          ;DONE. TYPE A "/".
010402  004767  173770                  JSR     %7,L8PCH
010406  005367  170772                  DEC     CTRA            ;DONE 36 TIMES?
010412  001001                          BNE     CT2E            ;BRANCH IF NOT DONE.
010414  104012                          SCOPE                   ;DONE, SCOPE.
010416  062767  000002  170762  CT2E:   ADD     #2,CTRB         ;MODIFY CTRB FOR NEXT TRY.
010424  000744                          BR      CT2C            ;GO DO IT AGAIN.
                                ;*****************************************************************************
010426  000003                  P2T3:   3                       ; PRG2 TEST ROUTINE 3                   *
010430  010520                          P2T4                    ;ADDRESS OF NEXT ROUTINE                *
                                ;*****************************************************************************
                                ;LINE FEED TEST
010432  104000                          TYPE                    ;TYPE TITLE
010434  014464                          LFTST
010436  052767  040000  170604          BIS     #BIT14,PRGID    ;ALLOW STALLS,
010444  012767  000110  170732          MOV     #72.,CTRA       ;SET 33/35 LINE FEED COUNT.
010452  104015                          CK37                    ;37?
010454  000403                          BR      CT3A            ;NO.
010456  062767  000011  170720          ADD     #9.,CTRA        ;INCREMENT LINE FEED COUNT BY 9.
010464  112700  000134          CT3A:   MOVB    #'\,%0          ;TYPE "\"
010470  004767  173702                  JSR     %7,L8PCH
010474  112700  000012                  MOVB    #12,%0          ;LINE FEED.
010500  004767  173672                  JSR     %7,L8PCH
010504  005367  170674                  DEC     CTRA            ;DONE N TIMES?
010510  001001                          BNE     CT3B            ;BRANCH IF NOT DONE.
010512  104012                          SCOPE                   ;DONE, SCOPE
010514  104002          CT3B:   STALL                   ;STALL
010516  000762                          BR      CT3A            ;REPEAT
                                ;*****************************************************************************
010520  000004                  P2T4:   4                       ; PRG2 TEST ROUTINE 4                   *
010522  010732                          P2T5                    ;ADDRESS OF NEXT ROUTINE                *
```

```
 2526                                   ;*************************************************************************
 2527                                   ;TAB TEST
 2528  010524  012767  000011  000074          MOV     #9.,TBCNT       ;SET TAB COUNT.
 2529  010532  104014                          CK3S                    ;3S?
 2530  010534  104012                          SCOPE                   ;NO.
 2531  010536  004567  000040                  JSR     %5,TPBM         ;TYPE MARKERS
 2532  010542  000007                          7
 2533  010544  104000                          TYPE
 2534  010546  014315                          TBMRK+1
 2535  010550  012767  000007  170626  CT4A:   MOV     #7,CTRA         ;LINE COUNT TO CTRA
 2536  010556  005067  000046                  CLR     SPCNT           ;0 TO SPACE COUNT.
 2537  010562  004767  000044          CT4B:   JSR     %7,TABP         ;GO SPACE-TAB.
 2538  010566  005267  000036                  INC     SPCNT           ;INCREMENT SPACE COUNT.
 2539  010572  005367  170606                  DEC     CTRA            ;DONE 7 LINES?
 2540  010576  001371                          BNE     CT4B            ;BRANCH IF NOT DONE.
 2541  010600  104012                          SCOPE                   ;DONE, SCOPE.
 2542  010602  012567  170576          TPBM:   MOV     (5)+,CTRA       ;TYPE TEST TITLE.
 2543  010606  104000                          TYPE
 2544  010610  014300                          TBTST
 2545  010612  104000                  TPBMA:  TYPE                    ;TYPE MARKERS
 2546  010614  014326                          TBMRK1
 2547  010616  005367  170562                  DEC     CTRA
 2548  010622  001373                          BNE     TPBMA
 2549  010624  000205                          RTS     %5              ;EXIT.
 2550  010626  000000                  TBCNT:  OPEN                    ;TAB COUNT
 2551  010630  000000                  SPCNT:  OPEN                    ;SPACE COUNT
 2552  010632  104000                  TABP:   TYPE                    ;CRLF.
 2553  010634  014337                          CRLF
 2554  010636  016767  177764  170542          MOV     TBCNT,CTRB      ;TAB COUNT TO CTRB
 2555  010644  016767  177760  170536  TABPA:  MOV     SPCNT,CTRC      ;SPACE COUNT TO CTRC
 2556  010652  001407                          BEQ     TABPC           ;BRANCH IF SPACE COUNT IS 0.
 2557  010654  112700  000040          TABPB:  MOVB    #40,%0          ;SPACE
 2558  010660  004767  173512                  JSR     %7,LSPCH
 2559  010664  005367  170520                  DEC     CTRC            ;DECREMENT SPACE COUNT
 2560  010670  001371                          BNE     TABPB           ;BRANCH IF NOT YET 0.
 2561  010672  112700  000011          TABPC:  MOVB    #11,%0          ;TAB
 2562  010676  004767  173474                  JSR     %7,LSPCH
 2563  010702  004767  173470                  JSR     %7,LSPCH        ;DUMMY CYCLE
 2564  010706  004767  173464                  JSR     %7,LSPCH        ;DUMMY CYCLE.
 2565  010712  112700  000057                  MOVB    #'/,%0          ;TYPE "/"
 2566  010716  004767  173454                  JSR     %7,LSPCH
 2567  010722  005367  170460                  DEC     CTRB            ;DECREMENT TAB COUNT.
 2568  010726  001346                          BNE     TABPA           ;BRANCH IF NOT DONE TABBING.
 2569  010730  000207                          RTS     %7              ;DONE, EXIT.
 2570                                   ;*************************************************************************
 2571  010732  000005                  P2T5:   5                       ; PRG2 TEST ROUTINE 5                   *
 2572  010734  010750                          P2T6                    ;ADDRESS OF NEXT ROUTINE                *
 2573                                   ;*************************************************************************
 2574                                   ;TYPE LINE OF CHARACTERS ABC
 2575  010736  104000                          TYPE                    ;TYPE "CHARACTER TESTS"
 2576  010740  014507                          CHRTST
 2577  010742  104016                          TYPLN3                  ;TYPE LINE
 2578  010744  014176                          A
 2579  010746  104012                          SCOPE                   ;SCOPE
```



```
 2580                                   ;*************************************************************************
 2581  010750  000006                  P2T6:   6                       ; PRG2 TEST ROUTINE 6                   *
 2582  010752  010762                          P2T7                    ;ADDRESS OF NEXT ROUTINE                *
 2583                                   ;*************************************************************************
 2584                                   ;TYPE LINE OF CHARACTERS DEF
 2585  010754  104016                          TYPLN3                  ;TYPE LINE
 2586  010756  014201                          D
 2587  010760  104012                          SCOPE                   ;SCOPE
 2588                                   ;*************************************************************************
 2589  010762  000007                  P2T7:   7                       ; PRG2 TEST ROUTINE 7                   *
 2590  010764  010774                          P2T10                   ;ADDRESS OF NEXT ROUTINE                *
 2591                                   ;*************************************************************************
 2592                                   ;TYPE LINE OF CHARACTERS GHI
 2593  010766  104016                          TYPLN3                  ;TYPE LINE
 2594  010770  014204                          G
 2595  010772  104012                          SCOPE                   ;SCOPE
 2596                                   ;*************************************************************************
 2597  010774  000010                  P2T10:  10                      ; PRG2 TEST ROUTINE 10                  *
 2598  010776  011006                          P2T11                   ;ADDRESS OF NEXT ROUTINE                *
 2599                                   ;*************************************************************************
 2600                                   ;TYPE LINE OF CHARACTERS OF JKL
 2601  011000  104016                          TYPLN3                  ;TYPELINE
 2602  011002  014207                          J
 2603  011004  104012                          SCOPE                   ;SCOPE
 2604                                   ;*************************************************************************
 2605  011006  000011                  P2T11:  11                      ; PRG2 TEST ROUTINE 11                  *
 2606  011010  011020                          P2T12                   ;ADDRESS OF NEXT ROUTINE                *
 2607                                   ;*************************************************************************
 2608                                   ;TYPE LINE OF CHARACTERS MNO
 2609  011012  104016                          TYPLN3                  ;TYPE LINE
 2610  011014  014212                          M
 2611  011016  104012                          SCOPE                   ;SCOPE
 2612                                   ;*************************************************************************
 2613  011020  000012                  P2T12:  12                      ; PRG2 TEST ROUTINE 12                  *
 2614  011022  011032                          P2T13                   ;ADDRESS OF NEXT ROUTINE                *
 2615                                   ;*************************************************************************
 2616                                   ;TYPE LINE OF CHARACTERS PQR
 2617  011024  104016                          TYPLN3                  ;TYPE LINE
 2618  011026  014215                          P
 2619  011030  104012                          SCOPE                   ;SCOPE
 2620                                   ;*************************************************************************
 2621  011032  000013                  P2T13:  13                      ; PRG2 TEST ROUTINE 13                  *
 2622  011034  011044                          P2T14                   ;ADDRESS OF NEXT ROUTINE                *
 2623                                   ;*************************************************************************
 2624                                   ;TYPE LINE OF CHARACTERS STU
 2625  011036  104016                          TYPLN3
 2626  011040  014220                          S
 2627  011042  104012                          SCOPE
 2628                                   ;*************************************************************************
 2629  011044  000014                  P2T14:  14                      ; PRG2 TEST ROUTINE 14                  *
 2630  011046  011056                          P2T15                   ;ADDRESS OF NEXT ROUTINE                *
 2631                                   ;*************************************************************************
 2632                                   ;TYPE LINE OF CHARACTERS VWX
 2633  011050  104016                          TYPLN3                  ;TYPE LINE
```

```
2634  011052  014223                        V
2635  011054  104012                        SCOPE                   ;SCOPE
2636                                  ;*************************************************************************
2637  011056  000015                  P2T15:  15                      ; PRG2 TEST ROUTINE 15                  *
2638  011060  011070                          P2T16                   ;ADDRESS OF NEXT ROUTINE                *
2639                                  ;*************************************************************************
2640                                  ;TYPE LINE OF CHARACTERS YZ0
2641  011062  104016                          TYPLN3                  ;TYPE LINE
2642  011064  014226                          Y
2643  011066  104012                          SCOPE                   ;SCOPE
2644                                  ;*************************************************************************
2645  011070  000016                  P2T16:  16                      ; PRG2 TEST ROUTINE 16                  *
2646  011072  011102                          P2T17                   ;ADDRESS OF NEXT ROUTINE                *
2647                                  ;*************************************************************************
2648                                  ;TYPE LINE OF CHARACTERS 123
2649  011074  104016                          TYPLN3                  ;TYPE LINE
2650  011076  014231                          ONE
2651  011100  104012                          SCOPE                   ;SCOPE
2652                                  ;*************************************************************************
2653  011102  000017                  P2T17:  17                      ; PRG2 TEST ROUTINE 17                  *
2654  011104  011114                          P2T20                   ;ADDRESS OF NEXT ROUTINE                *
2655                                  ;*************************************************************************
2656                                  ;TYPE LINE OF CHARACTERS 456
2657  011106  104016                          TYPLN3                  ;TYPE LINE
2658  011110  014234                          FOUR
2659  011112  104012                          SCOPE                   ;SCOPE
2660                                  ;*************************************************************************
2661  011114  000020                  P2T20:  20                      ; PRG2 TEST ROUTINE 20                  *
2662  011116  011126                          P2T21                   ;ADDRESS OF NEXT ROUTINE                *
2663                                  ;*************************************************************************
2664                                  ;TYPE LINE OF CHARACTERS 789
2665  011120  104016                          TYPLN3                  ;TYPE LINE
2666  011122  014237                          SEVEN
2667  011124  104012                          SCOPE                   ;SCOPE
2668                                  ;*************************************************************************
2669  011126  000021                  P2T21:  21                      ; PRG2 TEST ROUTINE 21                  *
2670  011130  011140                          P2T22                   ;ADDRESS OF NEXT ROUTINE                *
2671                                  ;*************************************************************************
2672                                  ;TYPE LINE OF CHARACTERS1"#
2673  011132  104016                          TYPLN3                  ;TYPE LINE
2674  011134  014242                          C41
2675  011136  104012                          SCOPE                   ;SCOPE
2676                                  ;*************************************************************************
2677  011140  000022                  P2T22:  22                      ; PRG2 TEST ROUTINE 22                  *
2678  011142  011152                          P2T23                   ;ADDRESS OF NEXT ROUTINE                *
2679                                  ;*************************************************************************
2680                                  ;TYPE LINE OF CHARACTERS $%&
2681  011144  104016                          TYPLN3                  ;TYPE LINE
2682  011146  014245                          C44
2683  011150  104012                          SCOPE                   ;SCOPE
2684                                  ;*************************************************************************
2685  011152  000023                  P2T23:  23                      ; PRG2 TEST ROUTINE 23                  *
2686  011154  011164                          P2T24                   ;ADDRESS OF NEXT ROUTINE                *
2687                                  ;*************************************************************************
```



```
2688                                  ;TYPE LINE OF CHARACTERS '()
2689  011156  104016                          TYPLN3                  ;TYPE LINE
2690  011160  014250                          C47
2691  011162  104012                          SCOPE                   ;SCOPE,
2692                                  ;*************************************************************************
2693  011164  000024                  P2T24:  24                      ; PRG2 TEST ROUTINE 24                  *
2694  011166  011176                          P2T25                   ;ADDRESS OF NEXT ROUTINE                *
2695                                  ;*************************************************************************
2696                                  ;TYPE LINE OF CHARACTERS *+,
2697  011170  104016                          TYPLN3                  ;TYPE LINE
2698  011172  014253                          C52
2699  011174  104012                          SCOPE                   ;SCOPE
2700                                  ;*************************************************************************
2701  011176  000025                  P2T25:  25                      ; PRG2 TEST ROUTINE 25                  *
2702  011200  011210                          P2T26                   ;ADDRESS OF NEXT ROUTINE                *
2703                                  ;*************************************************************************
2704                                  ;TYPE LINE OF CHARACTERS -./
2705  011202  104016                          TYPLN3                  ;TYPE LINE
2706  011204  014256                          C55
2707  011206  104012                          SCOPE                   ;SCOPE
2708                                  ;*************************************************************************
2709  011210  000026                  P2T26:  26                      ; PRG2 TEST ROUTINE 26                  *
2710  011212  011222                          P2T27                   ;ADDRESS OF NEXT ROUTINE                *
2711                                  ;*************************************************************************
2712                                  ;TYPE LINE OF CHARACTERS 9;<
2713  011214  104016                          TYPLN3                  ;TYPE LINE
2714  011216  014261                          C72
2715  011220  104012                          SCOPE                   ;SCOPE
2716                                  ;*************************************************************************
2717  011222  000027                  P2T27:  27                      ; PRG2 TEST ROUTINE 27                  *
2718  011224  011234                          P2T30                   ;ADDRESS OF NEXT ROUTINE                *
2719                                  ;*************************************************************************
2720                                  ;TYPE LINE OF CHARACTERS =>?
2721  011226  104016                          TYPLN3                  ;TYPE LINE
2722  011230  014264                          C75
2723  011232  104012                          SCOPE                   ;SCOPE,
2724                                  ;*************************************************************************
2725  011234  000030                  P2T30:  30                      ; PRG2 TEST ROUTINE 30                  *
2726  011236  011246                          P2T31                   ;ADDRESS OF NEXT ROUTINE                *
2727                                  ;*************************************************************************
2728                                  ;TYPE LINE OF CHARACTERS @[\
2729  011240  104016                          TYPLN3                  ;TYPE LINE
2730  011242  014267                          C100
2731  011244  104012                          SCOPE                   ;SCOPE
2732                                  ;*************************************************************************
2733  011246  000031                  P2T31:  31                      ; PRG2 TEST ROUTINE 31                  *
2734  011250  011260                          P2T32                   ;ADDRESS OF NEXT ROUTINE                *
2735                                  ;*************************************************************************
2736                                  ;TYPE LINE OF CHARACTERS ]^AND LEFT ARROW
2737  011252  104016                          TYPLN3                  ;TYPE LINE
2738  011254  014272                          C135
2739  011256  104012                          SCOPE                   ;SCOPE
2740                                  ;*************************************************************************
2741  011260  000032                  P2T32:  32                      ; PRG2 TEST ROUTINE 32                  *
```

```
 2742  011262  011316                          P2T33                   ;ADDRESS OF NEXT ROUTINE                *
 2743                                          ;************************************************************************
 2744                                          ;TYPE 2 LINES OF ALL CHARACTERS, FIRST LINE FULL SPEED, SECOND LINE WITH STALLS.
 2745  011264  004767  173456                  JSR     %7,FBALL        ;FILL BUFFER WITH ALL CHARACTERS.
 2746  011270  042767  040000  167752          BIC     #BIT14,PRGID    ;CLEAR STALL BIT IN PRGID
 2747  011276  004767  173236                  JSR     %7,TYPLN        ;TYPE LINE.
 2748  011302  052767  040000  167740          BIS     #BIT14,PRGID    ;SET STALL BIT IN PRGID
 2749  011310  004767  173224                  JSR     %7,TYPLN        ;TYPE LINE.
 2750  011314  104012                          SCOPE                   ;SCOPE.
 2751                                          ;************************************************************************
 2752  011316  000033                  P2T33:  33                      ; PRG2 TEST ROUTINE 33                  *
 2753  011320  011400                          P2T34                   ;ADDRESS OF NEXT ROUTINE                *
 2754                                          ;************************************************************************
 2755                                          ;TYPE 12 LINES OF ASR33 WORST CASE PATTERN, ALTERNATE LINES WITH STALLS.
 2756  011322  104013                          CK33                    ;33?
 2757  011324  104012                          SCOPE                   ;NO, BYPASS TEST.
 2758  011326  104000                          TYPE                    ;TYPE "WORST CASE PATTERN TEST"
 2759  011330  014533                          WCPTST
 2760  011332  004767  173450                  JSR     %7,FW336        ;PATTERN TO BUFFER.
 2761  011336  012767  000006  170040          MOV     #6,CTRA         ;SET COUNT TO 6
 2762  011344  042767  040000  167676  CT33A:  BIC     #BIT14,PRGID    ;CLEAR STALL BIT IN PRGID.
 2763  011352  004767  173162                  JSR     %7,TYPLN        ;TYPE LINE
 2764  011356  052767  040000  167664          BIS     #BIT14,PRGID    ;SET STALL BIT IN PRGID.
 2765  011364  004767  173150                  JSR     %7,TYPLN        ;TYPE LINE.
 2766  011370  005367  170010                  DEC     CTRA            ;DONE 6 TIMES?
 2767  011374  001363                          BNE     CT33A           ;BRANCH IF NOT 6 TIMES YET.
 2768  011376  104012                          SCOPE                   ;DONE, SCOPE.
 2769                                          ;************************************************************************
 2770  011400  000034                  P2T34:  34                      ; PRG2 TEST ROUTINE 34                  *
 2771  011402  177777                          P2TLST                  ;ADDRESS OF NEXT ROUTINE                *
 2772                                          ;************************************************************************
 2773                                          ;TYPE 12 LINES OF ASR35 WORST CASE PATTERN, ALTERNATE LINES WITH STALLS.
 2774  011404  104014                          CK35                    ;35?
 2775  011406  104012                          SCOPE                   ;NO, BYPASS TEST.
 2776  011410  104000                          TYPE                    ;TYPE "WORST CASE PATTERN TEST"
 2777  011412  014533                          WCPTST
 2778  011414  004767  173426                  JSR     %7,FW356        ;PATTERN TO BUFFER.
 2779  011420  012767  000006  167756          MOV     #6,CTRA         ;SET COUNT TO 6,
 2780  011426  042767  040000  167614  CT34A:  BIC     #BIT14,PRGID    ;CLEAR STALL BIT  IN PRGID.
 2781  011434  004767  173100                  JSR     %7,TYPLN        ;TYPE LINE.
 2782  011440  052767  040000  167602          BIS     #BIT14,PRGID    ;SET STALL BIT IN PRGID.
 2783  011446  004767  173066                  JSR     %7,TYPLN        ;TYPE LINE
 2784  011452  005367  167726                  DEC     CTRA            ;DONE 6 TIMES?
 2785  011456  001363                          BNE     CT34A           ;BRANCH IF NOT 6 TIMES YET.
 2786  011460  104012                          SCOPE                   ;DONE, SCOPE.
```



```
 2787                                                  .SBTTL  PRG3=PUNCH TEST
 2788                                          ;PRG3 = PUNCH TEST
 2789          000003                          Z=3
 2790          177777                          X=-1
 2791  011462  012767  011510  167544  PRG3:   MOV     #P3T0,KSTART    ;ADDR OF 1ST ROUTINE TO KSTART.
 2792  011470  052767  040000  167552          BIS     #BIT14,PRGID    ;ALLOW STALLS.
 2793  011476  012767  177400  171500          MOV     #177400,STLMSK  ;SET STALL MASK
 2794  011504  000167  170134                  JMP     SRSET           ;GO GET STARTED
 2795                                          ;************************************************************************
 2796  011510  000000                  P3T0:   0                       ; PRG3 TEST ROUTINE 0                   *
 2797  011512  011566                          P3T1                    ;ADDRESS OF NEXT ROUTINE                *
 2798  011514  000005                          5                       ;TEST ITERATION COUNT                   *
 2799  011516  011520                          P3AA                    ;SCOPE ENTRY POINT                      *
 2800                                          ;************************************************************************
 2801                                          ;PUNCH SPECIAL BINARY COUNT PATTERN IN PUNCH MODE 0 (FULL SPEED)
 2802  011520  012767  001000  167634  P3AA:   MOV     #512.,RCNT      ;SET CHARACTER COUNT TO 512
 2803  011526  004767  000216                  JSR     %7,PFRNT        ;GO PUNCH FRONT END.
 2804  011532  004767  172270                  JSR     %7,INBIN        ;INITIALIZE SPECIAL BINARY COUNT
 2805  011536  004767  172322          P3AB:   JSR     %7,GTBIN        ;GET BINARY CHARACTER
 2806  011542  004767  172630                  JSR     %7,LSPCH        ;GO PUNCH THE CHARACTER
 2807  011546  005367  167610                  DEC     RCNT            ;DECREMENT CHAR COUNT.
 2808  011552  001371                          BNE     P3AB            ;BRANCH IF COUNT NOT YET 0 YET.
 2809  011554  004767  000206                  JSR     %7,PLTLR        ;PUNCH TRAILER.
 2810  011560  004767  000230                  JSR     %7,PCHECK       ;CHECK DATA PUNCHED.
 2811  011564  104012                          SCOPE                   ;SCOPE
 2812                                          ;************************************************************************
 2813  011566  000001                  P3T1:   1                       ; PRG3 TEST ROUTINE 1                   *
 2814  011570  011646                          P3T2                    ;ADDRESS OF NEXT ROUTINE                *
 2815  011572  000005                          5                       ;TEST ITERATION COUNT                   *
 2816  011574  011576                          P3BA                    ;SCOPE ENTRY POINT                      *
 2817                                          ;************************************************************************
 2818                                          ;PUNCH SPECIAL BINARY COUNT PATTERN IN PUNCH MODE 1 (RANDOM STALLS AFTER
 2819                                          ;PUNCHING EACH CHARACTER.)
 2820  011576  012767  001000  167556  P3BA:   MOV     #512.,RCNT      ;SET CHARACTER COUNT TO 512.
 2821  011604  004767  000140                  JSR     %7,PFRNT        ;GO PUNCH FRONT END.
 2822  011610  004767  172212                  JSR     %7,INBIN        ;INITIALIZE SPECIAL BINARY COUNT.
 2823  011614  004767  172244          P3BB:   JSR     %7,GTBIN        ;GET BINARY CHARACTER.
 2824  011620  004767  172552                  JSR     %7,LSPCH        ;GO PUNCH THE CHARACTER.
 2825  011624  104002                          STALL                   ;RANDOM STALL.
 2826  011626  005367  167530                  DEC     RCNT            ;DECREMENT CHAR COUNT.
 2827  011632  001370                          BNE     P3BB            ;BRANCH IF COUNT NOT YET 0.
 2828  011634  004767  000126                  JSR     %7,PLTLR        ;PUNCH TRAILER.
 2829  011640  004767  000150                  JSR     %7,PCHECK       ;CHECK DATA PUNCHED.
 2830  011644  104012                          SCOPE                   ;SCOPE
 2831                                          ;************************************************************************
 2832  011646  000002                  P3T2:   2                       ; PRG3 TEST ROUTINE 2                   *
 2833  011650  177777                          P3TLST                  ;ADDRESS OF NEXT ROUTINE                *
 2834  011652  000005                          5                       ;TEST ITERATION COUNT                   *
 2835  011654  011664                          P3CA                    ;SCOPE ENTRY POINT                      *
 2836                                          ;************************************************************************
 2837                                          ;PUNCH SPECIAL BINARY COUNT PATTERN IN PUNCH MODE 2,
 2838                                          ;(RANDOM STALL BEFORE PUNCHING RANDOM LENGTH GROUP OF CHARACTERS),
 2839                                          ;MAXIMUM GROUP LENGTH= 15)
 2840  011656  012767  177760  171706          MOV     #177760,RCMSK   ;SET CHAR GROUP MASK FOR 17(8) MAX),
```

```
 2841  011664  012767  001000  167470  P3CA:   MOV     #512.,RCNT      ;SET CHARACTER COUNT TO 512.
 2842  011672  004767  000052          P3CB:   JSR     %7,PFRNT        ;GO PUNCH FRONT END.
 2843  011676  004767  172124                  JSR     %7,INBIN        ;INITIALIZE BPECIAL BINARY COUNT.
 2844  011702  004767  171644                  JSR     %7,GRCNT        ;GENERATE RANDOM CHARACTER COUNT
 2845  011706  104002                          STALL                   ;RANDOM STALL.
 2846  011710  004767  172150          P3CC:   JSR     %7,GTBIN        ;GET BINARY CHARACTER.
 2847  011714  004767  172456                  JSR     %7,L8PCH        ;PUNCH THE CHARACTER.
 2848  011720  005367  167436                  DEC     RCNT            ;DECREMENT CHAR COUNT
 2849  011724  001404                          BEQ     P3CD            ;BRANCH IF COUNT IS 0.
 2850  011726  005367  171642                  DEC     RNCNT           ;NOT 0. DECREMENT RANDOM CHAR COUNT.
 2851  011732  001366                          BNE     P3CC            ;BRANCH IF COUNT NOT YET 0.
 2852  011734  000762                          BR      P3CB            ;BRANCH IF COUNT 0.
 2853  011736  004767  000024          P3CD:   JSR     %7,PLTLR        ;PUNCH TRAILER.
 2854  011742  004767  000046                  JSR     %7,PCHECK       ;CHECK DATA PUNCHED.
 2855  011746  104012                          SCOPE                   ;SCOPE.
 2856                                  ;ROUTINE TO PUNCH FRONT END.
 2857  011750  004767  000012          PFRNT:  JSR     %7,PLTLR        ;PUNCH LEADER
 2858  011754  012700  000377                  MOV     #377,%0
 2859  011760  004767  172412                  JSR     %7,L8PCH        ;PUNCH SYNC CHARACTER. (RUBOUT)
 2860  011764  000207                          RTS     %7              ;EXIT.
 2861  011766  012767  000106  167410  PLTLR:  MOV     #70.,CTRA       ;SET CTRA TO 70.
 2862  011774  012700  000177          PLTRA:  MOV     #177,%0
 2863  012000  004767  172372                  JSR     %7,L8PCH        ;PUNCH CODE 177 FOR LEADER/TRAILER
 2864  012004  005367  167374                  DEC     CTRA            ;PUNCHED 70?
 2865  012010  001371                          BNE     PLTRA           ;BRANCH IF NOT YET 70.
 2866  012012  000207                          RTS     %7              ;DONE EXIT.
 2867  012014  012767  000226  167362  PCHECK: MOV     #150.,CTRA      ;SET SYNC COUNT TO 150.
 2868  012022  004767  170544          PCHKA:  JSR     %7,BREAD        ;READ CHARACTER
 2869  012026  122767  000377  167330          CMPB    #377,CRBUF      ;IS IT SYNC CHARACTER? (377)
 2870  012034  001405                          BEQ     PCHKB           ;BRANCH IF SYNC CHAR FOUND.
 2871  012036  005367  167342                  DEC     CTRA            ;NOT FOUND. DECREMENT CTRA.
 2872  012042  001367                          BNE     PCHKA           ;BRANCH IF NOT 150 CHARS READ YET.
 2873  012044  104010                          EHALT                   ;150 CHARS READ AND NO SYNC. HALT.
 2874  012046  000762                          BR      PCHECK          ;TRY AGAIN.
 2875  012050  004767  171752          PCHKB:  JSR     %7,INBIN        ;INITIALIZE BINARY COUNT.
 2876  012054  012767  001000  167322          MOV     #512.,CTRA      ;SET CHARACTER COUNT TO 512.
 2877  012062  004767  170504          PCHKC:  JSR     %7,BREAD        ;READ CHARACTER.
 2878  012066  004767  171772                  JSR     %7,GTBIN        ;GET BINARY COUNT CHARACTER.
 2879  012072  110067  167267                  MOVB    %0,CRBUF+1
 2880  012076  104004                          DATCHK                  ;COMPARE CHARACTERS.
 2881  012100  005367  167300          PCHKD:  DEC     CTRA            ;512 CHARS READ?
 2882  012104  001366                          BNE     PCHKC           ;BRANCH IF NOT 512 CHARS YET.
 2883  012106  000207                          RTS     %7              ;EXIT.
```



```
 2884                                          .SBTTL  PRG4-KEYBOARD TEST
 2885          000004                          Z=4
 2886          177777                          X=-1
 2887  012110  012767  012134  167116  PRG4:   MOV     #P4T0,KSTART
 2888  012116  052767  000200  167124          BIS     #BIT7,PRGID
 2889  012124  104000                          TYPE
 2890  012126  014567                          KMSG1
 2891  012130  000167  167510                  JMP     SRSET
 2892                                  ;************************************************************************
 2893  012134  000000                  P4T0:   0                       ; PRG4 TEST ROUTINE 0                   *
 2894  012136  012240                          P4T1                    ;ADDRESS OF NEXT ROUTINE                *
 2895                                  ;************************************************************************
 2896                                  ;TEST THAT PRESSING KEY SETS DONE FLAG.
 2897  012140  012767  000005  167236          MOV     #5,CTRA
 2898  012146  104006                  ET0A:   STRDRV
 2899  012150  012204                          ET0B
 2900  012152  104000                          TYPE                                    ;TYPE "PRESS A KEY WITHIN 10 SECS."
 2901  012154  014605                          KMSG2
 2902  012156  052777  000100  167024          BIS     #BIT6,@TKS              ;ENABLE KYBD INTERRUPT.
 2903  012164  005067  165606                  CLR     PSW
 2904  012170  104024                          DELAY                           ;WAIT 10 SECONDS
 2905  012172  023420                          10000.
 2906  012174  104000                          TYPE                            ;TYPE "NO KEYBOARD REQUEST."
 2907  012176  015007                          KMSG6
 2908  012200  104010                          EHALT                           ;HALT.
 2909  012202  000411                          BR      ET0CA
 2910  012204  105777  167000          ET0B:   TSTB    @TKS                    ;TEST FOR DONE BIT ON
 2911  012210  100403                          BMI     ET0C                    ;BRANCH IF DONE BIT SET.
 2912  012212  104000                          TYPE                            ;DONE BIT NOT SET. TYPE:FALSE  KEY=
 2913  012214  015035                          KMSG7                           ;BOARD OR READER INTERRUPT.
 2914  012216  104010                          EHALT                           ;HALT
 2915  012220  012716  012226          ET0C:   MOV     #ET0CA,@X6
 2916  012224  000002                          RTI                             ;EXIT INTERRUPT.
 2917  012226  104011                  ET0CA:  SRESET
 2918  012230  005367  167150                  DEC     CTRA                    ;DONE 5 TIMES?
 2919  012234  001344                          BNE     ET0A                    ;BRANCH IF NOT DONE.
 2920  012236  104012                          SCOPE                           ;SCOPE
 2921                                  ;************************************************************************
 2922  012240  000001                  P4T1:   1                       ; PRG4 TEST ROUTINE 1                   *
 2923  012242  012320                          P4T2                    ;ADDRESS OF NEXT ROUTINE                *
 2924                                  ;************************************************************************
 2925                                  ;ECHO TEST. KEYED CHARACTER IS TYPED. RUBOUT ENDS ROUTINE.
 2926  012244  104000                          TYPE                            ;TYPE TITLE AND INSTRUCTIONS.
 2927  012246  014645                          KMSG3
 2928  012250  105777  166734          ET1A:   TSTB    @TKS                    ;WAIT FOR DONE FLAG
 2929  012254  100375                          BPL     .-4
 2930  012256  117767  166730  167100          MOVB    @TKB,CRBUF              ;MOVE KYBD CHAR TO CRBUF.
 2931  012264  116777  167074  166724          MOVB    CRBUF,@TPB              ;ECHO CHAR READ.
 2932  012272  105777  166716                  TSTB    @TPS                    ;WAIT FOR PRINTER DONE.
 2933  012276  100375                          BPL     .-4
 2934  012300  042767  000200  167056          BIC     #BIT7,CRBUF             ;CLEAR BIT 7 FROM CRBUF.
 2935  012306  122767  000177  167050          CMPB    #177,CRBUF              ;COMPARE CRBUF TO RUBOUT (177)
 2936  012314  001355                          BNE     ET1A                    ;BRANCH IF NOT RUBOUT (177)
 2937  012316  104012                          SCOPE                           ;SCOPE
```

```
                                ;*************************************************************************
012320  000002                P4T2:   2                       ; PRG4 TEST ROUTINE 2                   *
012322  177777                        P4TLST                  ;ADDRESS OF NEXT ROUTINE                *
                                ;*************************************************************************
                                ;OCTAL EQUIVALENT TEST. THE OCTAL EQUIVALENT OF ANY CHARACTER KEYED
                                ;IS PRINTED. RUBOUT ENDS ROUTINE.
012324  104001                        TYPES                           ;TYPE TITLE AND INSTRUCTIONS.
012326  014747                        KMSG4
012330  014660                        KMSG3A
012332  177777                        -1
012334  005067  167024                CLR     CRBUF
012340  105777  166644        ET2A:   TSTB    @TKS                    ;WAIT FOR DONE FLAG.
012344  100375                        BPL     .-4
012346  117767  166640  167010        MOVB    @TKB,CRBUF              ;CHARACTER TO CRBUF
012354  004567  171646                JSR     %5,ACNV4                ;CONVERT CHAR IN CRBUF TO
012360  001364                        CRBUF                           ;PRINTABLE OCTAL
012362  015001                        OCTEQV
012364  104000                        TYPE                            ;TYPE OCTAL EQUIVALENT
012366  014777                        KMSG5
012370  042767  000200  166766        BIC     #BIT7,CRBUF             ;CLEAR BIT 7 FROM CRBUF
012376  022767  000177  166760        CMP     #177,CRBUF              ;TEST FOR RUBOUT CHARACTER.
012404  001355                        BNE     ET2A                    ;BRANCH IF NOT RUBOUT (177).
012406  104012                        SCOPE                           ;SCOPE.
```



```
                                        .SBTTL  PRG5 COMBINED TEST
012410  104005                PRG5:   CHALT                   ;SETUP HALT.
012412  004767  17221o                JSR     %7,STBF         ;SETUP BUFFER
012416  052767  040000  166624        BIS     #BIT14,PRGID    ;ALLOW STALLS.
012424  012767  177600  170552        MOV     #177600,STLMSK  ;SETMAX STALL TO 255 MSECS.
012432  005067  000054                CLR     PCHCNT
012436  005067  000046                CLR     RBUSY
012442  104006                        STRDRV                  ;SET READER SERVICE TO RZERO.
012444  013064                        RZERO
012446  104007                        STPCHV                  ;SET PUNCH SERVICE TO PCHDAT.
012450  012734                        PCHDAT
012452  012767  000241  000612        MOV     #241,SEED0      ;INITIALIZE PUNCH
012460  004767  000574                JSR     %7,INIT0
012464  012767  000077  166714        MOV     #63.,CTRB
012472  112377  166520                MOVB    (3)+,@TPB       ;PUNCH FIRST CHAR.
012476  052777  000100  166510        BIS     #BIT6,@TPS      ;ENABLE PUNCH INTERRUPT.
012504  104400                        DELAYX                  ;STALL WHILE AWAITING INTERRUPTS.
012506  000776                        BR      .-2             ;BACK TO WAIT.
012510  000000                RBUSY:  OPEN
012512  000000                PCHCNT: OPEN
012514  105777  166474        TSTPCH: TSTB    @TPS            ;CHECK FOR DONE.
012520  100401                        BMI     .+4             ;BRANCH IF DONE SET.
012522  104010                        EHALT                   ;NOT DONE.  FALSE INTERRUPT.
012524  005267  177762                INC     PCHCNT          ;INCREMENT PUNCH COUNT.
012530  000207                        RTS     %7              ;EXIT.
012532  026727  177754  000024  PCONT: CMP     PCHCNT,#20.     ;CHECK PUNCH COUNT.
012540  103424                        BLO     PCONTC          ;BRANCH IF LESS THAN 20.
012542  105767  177742                TSTB    RBUSY           ;READER BUSY?
012546  100406                        BMI     PCONTA          ;BRANCH IF READER BUSY.
012550  052767  000200  177732        BIS     #BIT7,RBUSY     ;NOT BUSY.  SET IT BUSY.
012556  052777  000101  166424        BIS     #101,@TKS       ;ENABLE READER AND INTERRUPT.
012564  026727  177722  000050 PCONTA: CMP     PCHCNT,#40.     ;RECHECK PUNCH COUNT.
012572  101402                        BLOS    PCONTB          ;BRANCH IF EQUAL OR LESS THAN 40.
012574  005077  166414                CLR     @TPS            ;DISABLE PUNCH INTERRUPT.
012600  032777  000400  165366 PCONTB: BIT     #BIT8,@SRPTR    ;CHECK FOR FULL SPEED RUN.
012606  001001                        BNE     PCONTC          ;BRANCH IF FULL SPEED DESIRED.
012610  104002                        STALL                   ;GO STALL.
012612  112377  166400        PCONTC: MOVB    (3)+,@TPB       ;LOAD PUNCH BUFFER.
012616  000002                        RTI                     ;EXIT INTERRUPT.
012620  105777  166364        TSTRDR: TSTB    @TKS            ;CHECK FOR READER DONE.
012624  100401                        BMI     .+4             ;BRANCH IF DONE SET.
012626  104010                        EHALT                   ;NOT DONE.  FALSE INTERRUPT.
012630  117767  166356  166526        MOVB    @TKB,CRBUF      ;CHECK READ TO CRBUF.
012636  005367  177650                DEC     PCHCNT          ;DECREMENT PUNCH COUNT.
012642  000207                        RTS     %7              ;EXIT.
012644  005767  177642        RCONT:  TST     PCHCNT          ;TEST PUNCH COUNT.
012650  001006                        BNE     RCONTA          ;BRANCH IF COUNT NOT 0.
012652  042767  000200  177630        BIC     #BIT7,RBUSY     ;COUNT 0.  CLEAR RBUSY.
012660  005077  166324                CLR     @TKS            ;CLEAR READER INTERRUPT ENABLE.
012664  000002                        RTI                     ;EXIT INTERRUPT
012666  026727  177620  000024 RCONTA: CMP     PCHCNT,#20.     ;COUNT LARGER THAN 20?
012674  101014                        BHI     RCONTC          ;BRANCH IF COUNT LARGER THAN 20.
012676  032777  000100  166310        BIT     #BIT6,@TPS      ;NOT LARGER.  PUNCH INTERRUPT ENABLED?
012704  001003                        BNE     RCONTB          ;BRANCH IF ENABLED.
```

```
 3015  012706  052777  000100  166300          BIS     #BIT6,@TPS      ;ENABLE PUNCH INTERRUPTS.
 3016  012714  032777  000400  165252  RCONTB: BIT     #BIT8,@SRPTR    ;CHECK FOR FULL SPEED RUN.
 3017  012722  001001                          BNE     RCONTC          ;BRANCH IF FULL SPEED DESIRED.
 3018  012724  104026                          RSTALL                  ;GO STALL.
 3019  012726  005277  166256          RCONTC: INC     @TKS            ;ENABLE READER.
 3020  012732  000002                          RTI                     ;EXIT INTERRUPT.
 3021  012734  004767  177554          PCHDAT: JSR     %7,TSTPCH       ;CHECK PUNCH.
 3022  012740  005367  166440                  DEC     CTRA            ;74 CHARS OUTPUTTED?
 3023  012744  001272                          BNE     PCONT           ;BRANCH IF NOT.
 3024  012746  005367  166434                  DEC     CTRB            ;63 LINES OUTPUTTED?
 3025  012752  001405                          BEQ     PCHDTA          ;BRANCH IF YES.
 3026  012754  005267  000312                  INC     SEED0           ;NO.  SETUP FOR NEXT LINE.
 3027  012760  004767  000274                  JSR     %7,INIT0        ;SETUP LINE.  74 TO CTRA
 3028  012764  000662                          BR      PCONT           ;CONTINUE.
 3029  012766  105067  002152          PCHDTA: CLRB    BLOCK1          ;FILL PUNCH BUFFER WITH ZEROES.
 3030  012772  004567  171342                  JSR     %5,BMOVE
 3031  012776  015144                          BLOCK1
 3032  013000  015145                          BLOCK1+1
 3033  013002  000107                          71.
 3034  013004  012703  015142                  MOV     #BLOCKA,%3      ;PUNCH BUFFER ADDRESS TO R3.
 3035  013010  012767  000024  166366          MOV     #20.,CTRA       ;SET CHAR COUNT TO 20.
 3036  013016  104007                          STPCHV                  ;SET PUNCH SERVICE TO PCHZER.
 3037  013020  013024                          PCHZER
 3038  013022  000643                          BR      PCONT           ;CONTINUE.
 3039  013024  004767  177464          PCHZER: JSR     %7,TSTPCH       ;CHECK PUNCH.
 3040  013030  005367  166350                  DEC     CTRA            ;ALL CHARS OUTPUTTED?
 3041  013034  001236                          BNE     PCONT           ;BRANCH IF NOT.
 3042  013036  012767  000241  000226          MOV     #241,SEED0      ;YES
 3043  013044  004767  000210                  JSR     %7,INIT0        ;SETUP LINE. 74 TO CTRA
 3044  013050  012767  000077  166330          MOV     #63.,CTRB       ;SET LINE COUNT TO 63
 3045  013056  104007                          STPCHV                  ;SET PUNCH SERVICE TO PCHDAT.
 3046  013060  012734                          PCHDAT
 3047  013062  000623                          BR      PCONT           ;CONTINUE.
 3048  013064  004767  177530          RZERO:  JSR     %7,TSTRDR       ;CHECK READER.
 3049  013070  105767  166270                  TSTB    CRBUF           ;TEST CHARACTER READ.
 3050  013074  001663                          BEQ     RCONT           ;BRANCH IF 0.
 3051  013076  004767  000002                  JSR     %7,RZERA        ;SET UP TO READ DATA.
 3052  013102  000415                          BR      RDATA
 3053  013104  012767  000241  000204  RZERA:  MOV     #241,SEED1      ;SET UP LINE.  74 TO CTRC
 3054  013112  004767  000166                  JSR     %7,INIT1
 3055  013116  012767  000077  166266          MOV     #63.,CTRD       ;SET LINE COUNT TO 63.
 3056  013124  104006                          STRDRV                  ;SET READER SERVICE TO RDAT.
 3057  013126  013132                          RDAT
 3058  013130  000207                          RTS     %7              ;EXIT
 3059  013132  004767  177462          RDAT:   JSR     %7,TSTRDR       ;CHECK READER.
 3060  013136  112467  166223          RDATA:  MOVB    (4)+,CRBUF+1    ;MOVE EXPECTED CHAR TO CRBUF+1
 3061  013142  104004                          DATCHK                  ;CHECK DATA.
 3062  013144  005367  166240                  DEC     CTRC            ;74 CHARACTERS CHECKED?
 3063  013150  001235                          BNE     RCONT           ;BRANCH IF NOT.
 3064  013152  005367  166234                  DEC     CTRD            ;63 LINES CHECKED?
 3065  013156  001405                          BEQ     RDATB           ;BRANCH IF YES.
 3066  013160  005267  000132                  INC     SEED1           ;NO.  SETUP NEXT LINE AND
 3067  013164  004767  000114                  JSR     %7,INIT1        ;74 TO CTRC.
 3068  013170  000625                          BR      RCONT           ;CONTINUE.
```



```
 3069  013172  105067  002060          RDATB:  CLRB    BLOCK2          ;FILL READ BUFFER WITH ZEROES.
 3070  013176  004567  171136                  JSR     %5,BMOVE
 3071  013202  015256                          BLOCK2
 3072  013204  015257                          BLOCK2+1
 3073  013206  000107                          71.
 3074  013210  012704  015254                  MOV     #BLOCKB,%4      ;READ BUFFER ADDRESS TO R4
 3075  013214  012767  000024  166166          MOV     #20.,CTRC       ;SET CHAR COUNT TO 20.
 3076  013222  104006                          STRDRV                  ;SET READER SERVICE TO R20ZER
 3077  013224  013230                          R20ZER
 3078  013226  000606                  RDATC:  BR      RCONT           ;CONTINUE.
 3079  013230  004767  177364          R20ZER: JSR     %7,TSTRDR       ;CHECK READER
 3080  013234  112467  166125                  MOVB    (4)+,CRBUF+1    ;MOVE EXPECTED CHAR TO CRBUF+1
 3081  013240  104004                          DATCHK                  ;CHECK DATA.
 3082  013242  005367  166142                  DEC     CTRC            ;ALL CHARS CHECKED?
 3083  013246  001367                          BNE     RDATC           ;BRANCH IF NOT.
 3084  013250  004767  177630                  JSR     %7,RZERA        ;SET UP TO READ DATA.
 3085  013254  000167  177364                  JMP     RCONT
 3086  013260  012703  015142          INIT0:  MOV     #BLOCKA,%3      ;PUNCH BUFFER ADDRESS TO R3
 3087  013264  004567  000040                  JSR     %5,DTFL         ;FILL PUNCH BUFFER WITH DATA
 3088  013270  015144                          BLOCK1                  ;STARTING WITH CHAR IN SEED0
 3089  013272  000000                  SEED0:  OPEN
 3090  013274  012767  000112  166102          MOV     #74.,CTRA
 3091  013302  000207                          RTS     %7              ;EXIT
 3092  013304  012704  015254          INIT1:  MOV     #BLOCKB,%4      ;READ BUFFER ADDRESS TO R4
 3093  013310  004567  000014                  JSR     %5,DTFL         ;FILL READ BUFFER WITH DATA
 3094  013314  015256                          BLOCK2                  ;STARTING WITH A CHAR IN SEED1.
 3095  013316  000000                  SEED1:  OPEN
 3096  013320  012767  000112  166062          MOV     #74.,CTRC
 3097  013326  000207                          RTS     %7              ;EXIT
 3098  013330  012502                  DTFL:   MOV     (5)+,%2         ;STARTING ADDRESS TO R2.
 3099  013332  012501                          MOV     (5)+,%1         ;SEED TO R1.
 3100  013334  012767  000110  166020          MOV     #72.,RCNT       ;CHAR COUNT TO RCNT.
 3101  013342  022701  000340          DTFLA:  CMP     #340,%1         ;(R1)EQUAL 340?
 3102  013346  001002                          BNE     DTFLB           ;BRANCH IF NOT.
 3103  013350  012701  000241                  MOV     #241,%1         ;EQUAL.  RESET TO 241.
 3104  013354  110122                  DTFLB:  MOVB    %1,(2)+         ;MOVE CHAR TO BUFFER
 3105  013356  005201                          INC     %1              ;INCREMENT (R1).
 3106  013360  005367  165776                  DEC     RCNT            ;BUFFER FULL?
 3107  013364  001366                          BNE     DTFLA           ;BRANCH IF NOT.
 3108  013366  000205                          RTS     %5              ;YES.  EXIT
```

```
                                        .SBTTL  PRG6, PRG7
                                    ;PRG6=READER EXERCISER, SPECIAL BINARY COUNT PATTERN
                                    ;SR15=HALT ON ERROR, SR14=0 STALL, SR14=1 FULL SPEED
013370  012767  177600  167606   PRG6:  MOV     #177600,STLMSK  ;SET STALL LIMIT
013376  012767  177760  170166          MOV     #177760,RCMSK   ;SET RANDOM CHARACTER LIMIT.
013404  052767  040000  165636          BIS     #BIT14,PRGID    ;ALLOW STALLS.
013412  004767  170228                  JSR     %7,SSYNC        ;SYNC READER.
013416  004767  170130          GTA:    JSR     %7,GRCNT        ;GENERATE RANDOM CHAR COUNT.
013422  032777  000400  164544          BIT     #BIT8,@SRPTR    ;CHECK FOR FULL SPEED RUN
013430  001001                  BNE     GTB             ;BRANCH IF FULL SPEED DESIRED.
013432  104002                  STALL                   ;STALL.
013434  004767  167132          GTB:    JSR     %7,BREAD        ;READ CHARACTER
013440  004767  170132          JSR     %7,BCHECK       ;GO CHECK IT
013444  005367  170124          DEC     RNCNT           ;DECREMENT CHAR COUNT
013450  001371                  BNE     GTB             ;BRANCH IF COUNT NOT 0.
013452  000761                  BR      GTA             ;COUNT 0, START OVER.
                                    ;PRG7=PRINTER EXERCISER, KEYBOARD CONTROLLED.
                                    ;TYPES LINES WITH ANY 5 CHARACTERS, STALLS OR FULL SPEED.
013454  004767  171154          PRG7:   JSR     %7,STBF         ;SET UP BUFFER.
013460  104000                  TYPE                    ;TYPE TITLE
013462  015063                  P7MG1
013464  052767  040000  165556 HTA:    BIS     #BIT14,PRGID    ;SET STALL BIT IN PRGID.
013472  012767  177600  167504          MOV     #177600,STLMSK  ;SET STALL MASK.
013500  012703  015144                  MOV     #BLOCK1,%3
013504  104000                  TYPE                    ;TYPE "TYPE IN DATA".
013506  015111                  P7MG2
013510  005777  165476                  TST     @TKB            ;CLEAR BUFFER.
013514  012767  000006  165662          MOV     #6,CTRA         ;CHAR COUNT TO CTRA.
013522  004767  171360          HTB:    JSR     %7,GKBCR        ;GET AND STORE KYBD CHARACTER.
013526  005367  165652                  DEC     CTRA            ;GOT 6 CHARACTERS?
013532  001373                  BNE     HTB             ;BRANCH IF NOT 6 CHARS YET.
013534  042767  000200  165622          BIC     #BIT7,CRBUF
013542  122767  000177  165614          CMPB    #177,CRBUF      ;CHECK 6TH CHAR FOR RUBOUT.
013550  001013                  BNE     HTC             ;BRANCH IF NOT A RUBOUT.
013552  042767  040000  165470          BIC     #BIT14,PRGID    ;RUBOUT, CLEAR STALL BIT IN PRGID.
013560  104015                  CK37                    ;37?
013562  000406                  BR      HTC             ;NO.
013564  004567  170550                  JSR     %5,BMOVE        ;YES, FILL 81 CHAR LINE.
013570  015144                  BLOCK1
013572  015151                  BLOCK1+5
013574  000114                  76.
013576  000405                  BR      HTD
013600  004567  170534          HTC:    JSR     %5,BMOVE        ;FILL 72 CHAR LINE.
013604  015144                  BLOCK1
013606  015151                  BLOCK1+5
013610  000103                  67.
013612  004767  170722          HTD:    JSR     %7,TYPLN        ;TYPE LINE.
013616  005777  164352                  TST     @SRPTR          ;CHANGE DATA? (SR15=1).
013622  100720                  BMI     HTA             ;YES, GO CHANGE DATA
013624  000772                  BR      HTD             ;NO CONTINUE WITH SAME DATA.
```



```
                                        .SBTTL  PRG10, PRG11, PRG12
                                    ;PRG10, PUNCH SPECIAL BINARY COUNT PATTERN TEST TAPE
013626  012746  000024          PRG10:  MOV     #20.,-(6)       ;PUNCH 20 BLANK CHAR, LEADER
013632  005000                  CLR     %0
013634  004767  170536          PRG10A: JSR     %7,LSPCH
013640  005316                  DEC     @%6
013642  001374                  BNE     PRG10A
013644  004767  170156                  JSR     %7,INBIN        ;INITIALIZE SPECIAL BINARY COUNT
013650  004767  170210          PRG10B: JSR     %7,GTBIN        ;GET BINARY CHARACTER.
013654  004767  170516                  JSR     %7,LSPCH        ;PUNCH CHARACTER
013660  000773                  BR      PRG10B          ;REPEAT.
                                    ;
                                    ;
                                    ;PRG11=PUNCH CLOCK ADJUSTMENT ROUTINE.
                                    ;OUTPUTS CHARACTER SET IN LEFT HALF OF SR, AND
                                    ;STALLS FOR NUMBER OF MILLISECONDS SET IN RIGHT HALF OF SR.
013662  104005          PRG11:  CHALT                   ;HALT TO SET SR.
013664  004767  000036          ITA:    JSR     %7,C1112        ;GO OUTPUT CHARACTER SET IN LEFT
013670  000775                  BR      ITA             ;HALF OF SR AND STALL PER SR RIGHT.
                                    ;
                                    ;
                                    ;PRG12=READER CLOCK ADJUSTMENT ROUTINE.
                                    ;PERFORMS SAME FUNCTION AS PRG11, AND IN ADDITION,
                                    ;USING THE PUNCH MAINTENANCE BIT, SHIFTS OUTPUT OF PUNCH
                                    ;SHIFT REGISTER ONTO THE READER BUFFER. THE CONTENTS OF THE
                                    ;READER BUFFER ARE THEN "FIXED" ON THE CONSOLE DATA LIGHTS
                                    ;BY ISSUING A RESET WITH CONTENTS OF READER BUFFER LOADED IN R0.
013672  104005          PRG12:  CHALT                   ;HALT TO SET SR.
013674  004767  000020          JTA:    JSR     %7,C1112M       ;GO OUTPUT CHARACTER FROM SR LEFT AND
013700  017700  165306                  MOV     @TKB,%0         ;STALL PER SR RIGHT, (TKB) TO R0.
013704  000005                  RESET                   ;"FIX" (TKB) IN DATA LIGHTS.
013706  000005                  RESET
013710  000005                  RESET
013712  000005                  RESET
013714  000005                  RESET
013716  000766                  BR      JTA             ;REPEAT.
                                    ;
013720  052777  000004  165266 C1112M: BIS     #4,@TPS         ;SET MAINTENANCE MODE (PUNCH).
013726  117767  164242  000022 C1112:  MOVB    @SRPTR,XTY      ;STALL COUNT TO XTY.
013734  005767  000016                  TST     XTY             ;DISREGARD 0 DELAY.
013740  001002                  BNE     C1112A
013742  005267  000010                  INC     XTY
013746  117777  164223  165242 C1112A: MOVB    @SRPTR+1,@TPB   ;LOAD PUNCH BUFFER.
013754  104024                  DELAY                   ;DELAY (APPROXIMATELY) THE NUMBER OF
013756  000000          XTY:    OPEN                    ;MSECS, SPECIFIED AT SR RIGHT
013760  000207                  RTS     %7              ;EXIT
```

```
  3205                                          .SBTTL  PRG13, PRG14
  3206                                  ;PRG13-MAINTENANCE MODE SINGLE CHARACTER DATA TEST.
  3207                                  ;WITH MAINTENANCE MODE SET, OUTPUTS ONTO PUNCH BUFFER AND BACK ONTO
  3208                                  ;READER BUFFER THE CHARACTER SET IN SR LEFT. THE CHARACTER IN THE
  3209                                  ;READER BUFFER IS COMPARED TO THE CHARACTER IN SR LEFT. IF THE 2 CHARACTERS
```



```
  3210                                  ;DISAGREE THE PROGRAM HALTS, THE DATA LIGHTS WILL THEN CONTAIN:
  3211                                  ;
  3212                                  ;LEFT HALF: THE EXPECTED CHARACTER (SR LEFT).
  3213                                  ;RIGHT HALF: THE CHARACTER IN THE READER BUFFER.
  3214  013762  104005                 PRG13:  CHALT                   ;HALT TO SET SR.
  3215  013764  052777  000004  165222 KTA:    BIS     #4,@TPS         ;SET MAINTENANCE MODE.
  3216  013772  105777  165216         KTB:    TSTB    @TPS            ;WAIT FOR READY.
  3217  013776  100375                         BPL     .-4
  3218  014000  117767  164171  165357         MOVB    @SRPTR+1,CRBUF+1        ;S/B CHAR TO CRBUF+1.
  3219  014006  116777  165353  165202         MOVB    CRBUF+1,@TPB    ;OUTPUT CHARACTER.
  3220  014014  105777  165170                 TSTB    @TKS            ;WAIT FOR READER DONE FLAG.
  3221  014020  100375                         BPL     .-4
  3222  014022  117767  165164  165334         MOVB    @TKB,CRBUF      ;CHAR READ TO CRBUF.
  3223  014030  104004                         DATCHK                  ;GO CHECK AGAINST S/B CHAR.
  3224  014032  000754                         BR      KTA             ;REPEAT.
  3225
  3226                                  ;PRG14-MAINTENANCE MODE SPECIAL BINARY COUNT PATTERN DATA TEST.
  3227                                  ;PERFORMS SAME OPERATION AS PRG13, EXCEPT THAT SPECIAL BINARY COUNT
  3228                                  ;PATTERN IS USED.
  3229  014034  012767  002000  165342 PRG14:  MOV     #1024.,CTRA     ;SET UP FOR 1024 CHECKS.
  3230  014042  004767  167760                 JSR     %7,INBIN        ;INITIALIZE BINARY COUNT
  3231  014046  012767  177600  167130         MOV     #177600,STLMSK  ;SET STALL LIMIT
  3232  014054  052767  040000  165166         BIS     #BIT14,PRGID    ;ALLOW STALLS
  3233  014062  052777  000004  165124 LTA:    BIS     #4,@TPS         ;SET MAINTENANCE MODE.
  3234  014070  032777  000400  164076         BIT     #BIT8,@SRPTR    ;CHECK STALL SWITCH
  3235  014076  001001                         BNE     LTB             ;BRANCH IF NO STALL WANTED
  3236  014100  104002                         STALL                   ;STALL
  3237  014102  105777  165106         LTB:    TSTB    @TPS            ;WAIT FOR READY.
  3238  014106  100375                         BPL     .-4
  3239  014110  004767  170016                 JSR     %7,GTBINP       ;GET BIN CHARACTER.
  3240  014114  110167  165245                 MOVB    %1,CRBUF+1      ;MOVE TO S/B CHAR.
  3241  014120  110177  165072                 MOVB    %1,@TPB         ;OUTPUT BIN CHARACTER.
  3242  014124  105777  165060                 TSTB    @TKS            ;WAIT FOR READER DONE.
  3243  014130  100375                         BPL     .-4
  3244  014132  117767  165054  165224         MOVB    @TKB,CRBUF      ;CHAR IN READ BUFFER TO CRBUF.
  3245  014140  104004                         DATCHK                  ;GO CHECK AGAINST S/B CHAR.
  3246  014142  005737  000042                 TST     @#42            ;CHAIN OR AUTO ACCEPT?
  3247  014146  001745                         BEQ     LTA             ;BR IF NOT.
  3248  014150  005367  165230                 DEC     CTRA            ;DONE REQUIRED TIMES?
  3249  014154  001342                         BNE     LTA             ;BR IF NOT.
  3250  014156  000167  165670                 JMP     CHNC            ;YES, GO EXIT.
```

```
 3251  014162      047      137      127  A33WP6: .BYTE   047,137,127,057,127,137
 3252  014165      057      127      137
 3253  014170      047      133      077  A35WP6: .BYTE   047,133,077,103,077,133
 3254  014173      103      077      133
 3255  014176      101      102      103  A:      .BYTE   101,102,103
 3256  014201      104      105      106  D:      .BYTE   104,105,106
 3257  014204      107      110      111  G:      .BYTE   107,110,111
 3258  014207      112      113      114  J:      .BYTE   112,113,114
 3259  014212      115      116      117  M:      .BYTE   115,116,117
 3260  014215      120      121      122  P:      .BYTE   120,121,122
 3261  014220      123      124      125  S:      .BYTE   123,124,125
 3262  014223      126      127      130  V:      .BYTE   126,127,130
 3263  014226      131      132      060  Y:      .BYTE   131,132,060
 3264  014231      061      062      063  ONE:    .BYTE   061,062,063
 3265  014234      064      065      066  FOUR:   .BYTE   064,065,066
 3266  014237      067      070      071  SEVEN:  .BYTE   067,070,071
 3267  014242      041      042      043  C41:    .BYTE   041,042,043
 3268  014245      044      045      046  C44:    .BYTE   044,045,046
 3269  014250      047      050      051  C47:    .BYTE   047,050,051
 3270  014253      052      053      054  C52:    .BYTE   052,053,054
 3271  014256      055      056      057  C55:    .BYTE   055,056,057
 3272  014261      072      073      074  C72:    .BYTE   072,073,074
 3273  014264      075      076      077  C75:    .BYTE   075,076,077
 3274  014267      100      133      134  C100:   .BYTE   100,133,134
 3275  014272      135      136      137  C135:   .BYTE   135,136,137
 3276  014275      377      000      377  C377:   .BYTE   377,000,377
 3277  014300  021445  040524  020102  TBTST:  .ASCII  '%#TAB TEST%#'
 3278  014306  042524  052123  021445
 3279  014314  020040  020040  020040  TBMRK:  .ASCII  '        /@'
 3280  014322  020040  040057
 3281  014326  020040  020040  020040  TBMRK1: .ASCII  '       /@'
 3282  014334  027440      100
 3283  014337      045      100          CRLF:   .ASCII  '%@'
 3284  014341      055  026455  044455  RM33A:  .ASCII  '----I@'
 3285  014346      100
 3286  014347      055  026511      100  RM33B:  .ASCII  '-I-@'
 3287  014353      055  026455  044455  RM37A:  .ASCII  '----I-I@'
 3288  014360  044455      100
 3289  014363      134  040040          SPTSTC: .ASCII  '\ @'
 3290  014366  021445  040503  051122  CRTST:  .ASCII  '%#CARRIAGE RETURN TEST%#@'
 3291  014374  040511  042507  051040
 3292  014402  052105  051125  020116
 3293  014410  042524  052123  021445
 3294  014416      100
 3295  014417      045  051043  043511  RMTST:  .ASCII  '%#RIGHT MARGIN TEST%#@'
 3296  014424  052110  046440  051101
 3297  014432  044507  020116  042524
 3298  014440  052123  021445      100
 3299  014445      045  051443  040520  SPTST:  .ASCII  '%#SPACE TEST%#@'
 3300  014452  042503  052040  051505
 3301  014460  022524  040043
 3302  014464  021445  044514  042516  LFTST:  .ASCII  '%#LINE FEED TEST%#@'
 3303  014472  043040  042505  020104
 3304  014500  042524  052123  021445
```



```
 3305  014506      100
 3306  014507      045  041443  040510  CHRTST: .ASCII  '%#CHARACTER TEST3%#@'
 3307  014514  040522  052103  051105
 3308  014522  052040  051505  051524
 3309  014530  021445      100
 3310  014533      045  053443  051117  WCPTST: .ASCII  '%#WORST CASE PATTERN TEST%#@'
 3311  014540  052123  041440  051501
 3312  014546  020105  040520  052124
 3313  014554  051105  020116  042524
 3314  014562  052123  021445      100
 3315  014567      045  043443  041131  KMSG1:  .ASCII  '%#KYBD TEST%#@'
 3316  014574  020104  042524  052123
 3317  014602  021445      100
 3318  014605      045  051120  051505  KMSG2:  .ASCII  '%PRESS A KEY WITHIN 10 SECONDS,@'
 3319  014612  020123  020101  042513
 3320  014620  020131  044527  044124
 3321  014626  047111  030440  020060
 3322  014634  042523  047503  042116
 3323  014642  027123      100
 3324  014645      045  042443  044103  KMSG3:  .ASCII  '%#ECHO TEST'
 3325  014652  020117  042524  052123
 3326  014660  041445  040510  040522  KMSG3A: .ASCII  '%CHARACTER KEYED WILL BE TYPED,'
 3327  014666  052103  051105  045440
 3328  014674  054505  042105  053440
 3329  014702  046111  020114  042502
 3330  014710  052040  050131  042105
 3331  014716      056
 3332  014717      045  052522  047502          .ASCII  '%RUBOUT ENDS ROUTINE,%#@'
 3333  014724  052125  042440  042116
 3334  014732  020123  047522  052125
 3335  014740  047111  027105  021445
 3336  014746      100
 3337  014747      045  047443  052103  KMSG4:  .ASCII  '%#OCTAL EQUIVALENT TEST@'
 3338  014754  046101  042440  052521
 3339  014762  053111  046101  047105
 3340  014770  020124  042524  052123
 3341  014776      100
 3342  014777      045      040          KMSG5:  .ASCII  '% '
 3343  015001      040  020040  022440  OCTEQV: .ASCII  '    %@'
 3344  015006      100
 3345  015007      045  047516  045440  KMSG6:  .ASCII  '%NO KEYBOARD REQUEST,@'
 3346  015014  054505  047502  051101
 3347  015022  020104  042522  052521
 3348  015030  051505  027124      100
 3349  015035      045  040506  051514  KMSG7:  .ASCII  '%FALSE KYBD INTERRUPT@'
 3350  015042  020105  054513  042102
 3351  015050  044440  052116  051105
 3352  015056  052522  052120      100
 3353  015063      045  050043  044522  P7MG1:  .ASCII  '%#PRINTER EXERCISER%#@'
 3354  015070  052116  051105  042440
 3355  015076  042530  041522  051511
 3356  015104  051105  021445      100
 3357  015111      045  052043  050131  P7MG2:  .ASCII  '%#TYPE IN DATA :@'
 3358  015116  020105  047111  042040
```

```
  3359  015124  052101  020101  040072
  3360  015132  020125     100          BKSU:   .ASCII  'U 0'
  3361  015135     040  020040  020040  DECVAL: .ASCII  '     '
  3362  015142  000001                  DEND:   .END
```



```
A       014176          ACNV    004264          ACNVB   004220          ACNVC   004246
ACNVM   004300          ACNVX   004262          ACNV4   004226          ACNV6   004200
ADTENP  004526          ARDB    002412          AREAD   002364          AT20E   006052
A1ST    004254          A33WP6  014162          A35WP6  014170          BCHECK  003576
BDCNV   004420          BDCNVA  004440          BELL    = 000007        BIT0    = 000001
BIT1    = 000002        BIT10   = 002000        BIT11   = 004000        BIT12   = 010000
BIT13   = 020000        BIT14   = 040000        BIT15   = 100000        BIT2    = 000004
BIT3    = 000010        BIT4    = 000020        BIT5    = 000040        BIT6    = 000100
BIT7    = 000200        BIT8    = 000400        BIT9    = 001000        BKSU    015132
BLKBB   = 015265        BLKCC   = 015377        BLK2    = 015267        BLOCKA= 015142
BLOCKB= 015254          BLOCKC= 015366          BLOCK1= 015144          BLOCK2= 015256
BMOVA   004350          BMOVE   004340          BRCTR   001414          BREAD   002572
BREADA  002620          BREADB  002632          BREADC  002652          BREADD  002662
BSYNC   003636          CC      = 177776        CHAINN  001754          CHALT   = 104005
CHKASR= 104022          CHK33   002210          CHK33A  002220          CHK33B  002224
CHK35   002226          CHLT    001422          CHNA    002004          CHNAA   002026
CHNB    002032          CHNC    002052          CHRTST  014507          CHR1    001366
CHR1A   001374          CHR2    001370          CHR2A   001376          CHR3    001372
CHR3A   001400          CKASR   002240          CK33    = 104013        CK35    = 104014
CK37    = 104015        CLEAN   001654          CNVCTR  004520          CPRDY   004362
CPRDYA  004372          CRBUF   001364          CRLF    014337          CRTA    001612
CRTB    001634          CRTST   014366          CTRA    001404          CTRB    001406
CTRC    001410          CTRD    001412          CT0A    010124          CT0B    010134
CT0C    010142          CT1A    010252          CT2A    010310          CT2B    010330
CT2C    010336          CT2D    010360          CT2E    010416          CT3A    010464
CT3B    010514          CT33A   011344          CT34A   011426          CT4A    010550
CT4B    010562          CURTST  001236          C100    014267          C1112   013726
C1112A  013746          C1112M  013720          C135    014272          C377    014275
C41     014242          C44     014245          C47     014250          C52     014253
C55     014256          C72     014261          C75     014264          D       014201
DATCHK= 104004          DATHLT= 104017          DECVAL  015135          DELAY   = 104024
DELAYX= 104400          DEND    015142          DIGIT   004522          DLCNT   = 003050
DLY     003046          DLYA    003070          DLYB    003076          DLYR    003304
DLYR0   = 003350        DLYR1   = 003362        DLYT    003144          DLYX    003344
DLYXA   003356          DLYXB   003364          DTCHK   001446          DTCHKA  001464
DTFL    013330          DTFLA   013342          DTFLB   013354          DTHLT   001456
DVDND   001416          DVQUOT  001420          EHALT   = 104010        EHLT    001434
EMTINT  002170          EMTTAB  001304          EMTX    = 000027        ERCTR   001402
ERR     001466          ERRA    001506          ERROR   = 104003        ET0A    012146
ET0B    012204          ET0C    012220          ET0CA   012226          ET1A    012250
ET2A    012340          FBALL   004746          FBF3    004702          FBF3A   004712
FBF3B   004720          FORWD   002112          FORWDA  002144          FORWDB  002152
FOUR    014234          FW336   005006          FW356   005046          G       014204
GETRDY  001646          GKBCR   005106          GRCNT   003552          GTA     013416
GTB     013434          GTBIN   004064          GTBINP  004132          GTRDYA  001676
GTRDYB  001702          GTRDYC  001716          GTRDYD  001740          HERE    002076
HTA     013464          HTB     013522          HTC     013600          HTD     013612
ICTR    001244          INBIN   004026          INCPRG  001606          INCRTN  001750
INIT0   013260          INIT1   013304          ITA     013664          J       014207
JTA     013674          KMSG1   014567          KMSG2   014605          KMSG3   014645
KMSG3A  014660          KMSG4   014747          KMSG5   014777          KMSG6   015007
KMSG7   015035          KSTART  001234          KTA     013764          KTB     013772
LFTST   014464          LOGIC   002062          L3PCH   004376          LTA     014062
LTB     014102          M       014212          MACHER  000004          MANUAL= 100000
```

```
M8EC  = 003072          NOP   = 000240          NXT8T   001242          OCTEQV  013001
ONE     014231          OPEN  = 000000          P       014215          PC     =%000007
PCHCNT  012512          PCHDAT  012734          PCHDTA  012766          PCHECK  012014
PCHKA   012022          PCHKB   012050          PCHKC   012062          PCHKD   012100
PCHZER  013024          PCONT   012532          PCONTA  012564          PCONTB  012600
PCONTC  012612          PFRNT   011750          PIND    004056          PLTLR   011766
PLTRA   011774          POP8P  = 005726         POP8P2= 022626          PRGID   001250
PRGNUM  001232          PRGTAB  001252          PRG0    005144          PRG1    007652
PRG10   013626          PRG10A  013634          PRG10B  013650          PRG11   013662
PRG12   013672          PRG13   013762          PRG14   014034          PRG2    010026
PRG3    011462          PRG4    012110          PRG5    012410          PRG6    013370
PRG7    013454          PRTY0  = 000000         PRTY1  = 000040         PRTY2  = 000100
PRTY3  = 000140         PRTY4  = 000200         PRTY5  = 000240         PRTY6  = 000300
PRTY7  = 000340         PSW    = 177776         PT0     004052          PT0P    004060
PT1     004054          PT1P    004062          P0AA    005174          P0AAA   006776
P0AAB   007020          P0AAE   007026          P0AE    005202          P0BA    005224
P0BAA   007046          P0BAC   007100          P0BE    005232          P0CA    005254
P0CAA   007116          P0CAC   007152          P0CAD   007166          P0CAE   007172
P0CAF   007174          P0CE    005262          P0DA    005304          P0DAA   007216
P0DAC   007252          P0DE    005312          P0EA    005334          P0EAA   007276
P0EAB   007322          P0EAC   007332          P0EB    005356          P0EC    005376
P0FA    005416          P0FAA   007354          P0GA    005454          P0GAA   007416
P0GAB   007466          P0HA    005516          P0HAA   007506          P0HAB   007564
P0IA    005544          P0JA    005600          P0JAA   007600          P0KA    005634
P0KAA   007630          P0KB    005642          P0KC    005662          P0LA    005676
P0MA    005760          P0MB    006004          P0NA    006032          P0NC    006054
P0OA    006102          P0OE    006132          P0PA    006162          P0PB    006214
P0QA    006242          P0QC    006276          P0QD    006312          P0QE    006316
P0RA    006344          P0SA    006426          P0TA    006462          P0TLST= 177777
P0T0    005156          P0T1    005206          P0T10   005530          P0T11   005564
P0T12   005622          P0T13   005664          P0T14   005732          P0T15   006006
P0T16   006056          P0T17   006136          P0T2    005236          P0T20   006222
P0T21   006326          P0T22   006410          P0T23   006452          P0T24   006526
P0T25   006562          P0T26   006604          P0T27   006634          P0T3    005266
P0T30   006670          P0T31   006722          P0T32   006762          P0T33   007032
P0T34   007106          P0T35   007202          P0T36   007260          P0T37   007342
P0T4    005316          P0T40   007402          P0T41   007472          P0T42   007570
P0T43   007620          P0T5    005400          P0T6    005442          P0T7    005476
P0UA    006536          P0VA    006572          P0WA    006614          P0XA    006644
P0YA    006700          P0ZA    006736          P0ZB    006760          P1AA    007722
P1BA    007750          P1CA    010000          P1CC    010006          P1TLST= 177777
P1T0    007706          P1T1    007734          P1T2    007764          P2TLST= 177777
P2T0    010060          P2T1    010206          P2T10   010774          P2T11   011006
P2T12   011020          P2T13   011032          P2T14   011044          P2T15   011056
P2T16   011070          P2T17   011102          P2T2    010272          P2T20   011114
P2T21   011126          P2T22   011140          P2T23   011152          P2T24   011164
P2T25   011176          P2T26   011210          P2T27   011222          P2T3    010426
P2T30   011234          P2T31   011246          P2T32   011260          P2T33   011316
P2T34   011400          P2T4    010520          P2T5    010732          P2T6    010750
P2T7    010762          P3AA    011520          P3AB    011536          P3BA    011576
P3BB    011614          P3CA    011664          P3CB    011702          P3CC    011710
P3CD    011736          P3TLST= 177777          P3T0    011510          P3T1    011566
P3T2    011646          P4TLST= 177777          P4T0    012134          P4T1    012240
P4T2    012320          P7MG1   015063          P7MG2   015111          RBUSY   012510
```



```
RCMSK   003572          RCNT    001362          RCONT   012644          RCONTA  012666
RCONTB  012714          RCONTC  012726          RDAT    013132          RDATA   013136
RDATB   013172          RDATC   013226          RDELAY= 104025          RDLCNT= 003210
RDLY    003206          RDLYA   003230          RDLYB   003236          RESET2= 104023
RIND    004050          RMB     010266          RMTST   014417          RM33A   014341
RM338   014347          RM37A   014353          RNCNT   003574          RNGEN   002520
RP1     002566          RP2     002570          RSETT2  002512          RSTAL   003306
RSTALL= 104026          RSTLA   003320          RSTLAA  003340          RSTLB   003342
RSTPC   002360          RSTPSW  002362          RSTREG= 104021          RSTRG   002324
RTNNO   001240          RZERA   013104          RZERO   013064          R0     =%000000
R1     =%000001         R2     =%000002         R20ZER  013230          R3     =%000003
R4     =%000004         R5     =%000005         R6     =%000006         S       014220
SAVREG= 104020          SAVRG   002264          SCOPE  = 104012         SCOPTR  001246
SEED0   013272          SEED1   013316          SEVEN   014237          SHALT   001510
SHLTA   001526          SOFTSR  000176          SP     =%000006         SPBOT  = 001200
SPCNT   010630          SPTST   014445          SPTSTC  014363          SRESET= 104011
SRPTR   000174          SRSET   001644          SRSETT  002474          STAL    003146
STALA   003200          STALAA  003160          STALB   003202          STALL  = 104002
START   001530          STBF    004634          STBFA   004650          STLMSK  003204
STLSPV  002444          STLSRV  002414          STPCHV= 104007          STPPA   002462
STPRA   002432          STRDRV= 104006          SUBTEN  004460          SUBTNA  004464
SUBTNB  004500          SVRPC   002320          SVRPSW  002322          SYCTRA  004024
SYNCA   003716          SYNCB   003724          SYNCC   004012          TABP    010632
TABPA   010644          TABPB   010654          TABPC   010672          TBCNT   010626
TBMRK   014314          TBMRK1  014326          TBTST   014300          TCTR    004572
TENPWR  004524          TIMCAL  003402          TIMCLA  003430          TIMCLB  003446
TKB     001212          TKLVL   001222          TKS     001210          TKVTR   001220
TPB     001216          TPBM    010602          TPBMA   010612          TPLVL   001226
TPL3A   004616          TPS     001214          TPVTR   001224          TSPCH   003456
TSPCHA  003542          TSPCHB  003550          TSPCHC  003500          TSTPCH  012514
TSTRDR  012620          TTYTYP  001230          TYP     002666          TYPA    002676
TYPC    002720          TYPD    002746          TYPDAT  003012          TYPE   = 104000
TYPES  = 104001         TYPF    002764          TYPG    002776          TYPLA   004546
TYPLB   004552          TYPLN   004540          TYPLN3= 104016          TYPL3   004574
TYPS    003014          TYPSA   003040          TYPSB   003042          V       014223
WCPTST  014533          X      = 000002         XTY     013756          Y       014226
Z      = 000004         .      = 015142
```

```
ERRORS DETECTED:  0

*DZKLAE,DZKLAE=DZKLAE./SOL
RUN-TIME:  16 27  0 SECONDS
CORE USED:  15K
```