

VT06

DISPLAY TERMINAL TESTS
MD-11-D6D-B

EP-D6D-DL
COPYRIGHT © 1971
FICHE 1 OF 1

FEB 1978
digital
MADE IN USA



IDENTIFICATION

PRODUCT CODE: MAINSEC-11-D6DB-D
PRODUCT NAME: VT26 DISPLAY TERMINAL TESTS
DATE: JULY, 1971
MAINTAINER: DIAGNOSTIC GROUP
AUTHOR: J. FRIEDRICH

COPYRIGHT © 1971
DIGITAL EQUIPMENT CORPORATION

NOTE: THIS PROGRAM IS A MODIFIED VERSION OF THE TELETYPE DIAGNOSTIC
MAINDEC-11-02AA

1. ABSTRACT

THE VT06 DISPLAY TERMINAL TESTS CONSISTS OF A PACKAGE OF TEST
PROGRAMS DESIGNED TO TEST THE VT06 INPUT-OUTPUT LOGIC, THE VT06
DISPLAY, AND THE KEYBOARD. ALL TESTS ARE INCLUDED IN ONE OBJECT
TAPE.

THE AVAILABLE TEST PROGRAMS ARE LISTED HERE IN NUMERICAL ORDER:

PRG0-COMBINED INPUT-OUTPUT LOGIC TESTS

PRG1-DISPLAY TEST

PRG2-KEYBOARD TEST

PRG3-PRINTER EXERCISER

PRG4-CLOCK ADJUSTMENT ROUTINE

PRG5-CLOCK ADJUSTMENT ROUTINE

PRG6-MAINTENANCE MODE SINGLE CHARACTER DATA TEST,

PRG7-MAINTENANCE MODE SPECIAL BINARY COUNT PATTERN TEST,

PRG8-ROLL-UP DISPLAY TEST

2. REQUIREMENTS

2.1 EQUIPMENT

A. PDP-11,20 SYSTEM, (4 K CORE),

B. VT06 DISPLAY TERMINAL

C. HIGH SPEED READER

THE VT06 MUST HAVE STANDARD TELETYPE ADDRESSES, REFER TO
SECTION 7.3 IF THE VT06 DOES NOT HAVE STANDARD PERIPHERAL
ADDRESSES,

2.2 STORAGE

THIS PROGRAM USES LOCATION 00200 THROUGH 11672.

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

4.1 VT06 IDENTIFICATION

THIS TEST DIAGNOSTIC ASSUMES THE VT06 IS A KSR35 WITH CURSOR CHARACTERS AND LOCATION 000224 IS SET TO 000001.

4.2 PRGM USE PROCEDURE

- A, SET VT06 TO ON-LINE.
- B, LOAD ADDRESS 000200
- C, SET SR TO 000000, PRESS START
- D, THE PROGRAM STOPS AT COMMON HALT.
- E, SET ANY DESIRED SR OPTIONS, NORMAL RUN IS WITH SR = 000000.

THIS PROGRAM'S SR OPTIONS ARE:

SR15 HALT AT END OF ROUTINE
 SR14 ENTER SCOPE MODE AFTER ERROR
 SR11 INHIBIT ITERATION
 SR10 LOOP PROGRAM
 SR9 SELECT ROUTINE
 SR6 THROU2H SR0 = NUMBER OF ROUTINE TO BE SELECTED.

SECTION 7,2 GIVES A COMPLETE EXPLANATION OF SR OPTIONS;

- F, PRESS CONTINUE, THE PROGRAM IS EXECUTED AND STOPS AT PROGRAM END HALT WHEN COMPLETED, PROVIDED NO ERRORS OCCUR.
- G, REFER TO SECTION 6, IF ERRORS OCCUR.

A DESCRIPTION OF THIS PROGRAM IS GIVEN IN SECTION 8,1

EXECUTION TIME:

ONE NORMAL ERROR FREE PASS TAKES APPROXIMATELY 4 MINUTES.

4.3 PRG1 USE PROCEDURE

- A. SET VT06 TO ON-LINE
- B. LOAD ADDRESS 000200.
- C. SET SR TO 000001; PRESS START
- D. PROGRAM STOPS AT COMMON HALT.
- E. SET ANY DESIRED SR OPTIONS, NORMAL RUN IS WITH SR = 000000.

THIS PROGRAM'S SR OPTIONS ARE:

- SR15 HALT AT END OF ROUTINE
- SR10 LOOP PROGRAM
- SR9 SELECT ROUTINE
- SR6 THROUGH SR8 = NUMBER OF ROUTINE TO BE SELECTED.

SECTION 7.2 GIVES A COMPLETE EXPLANATION OF SR OPTIONS.

- F. PRESS CONTINUE, THE VT06 WILL BE EXERCISED AND THE PROGRAM WILL STOP AT PROGRAM END HALT WHEN COMPLETED.
- G. ERROR DETECTION IS BY VISUAL INSPECTION OF DISPLAY.

A DESCRIPTION OF THIS PROGRAM IS GIVEN IN SECTION 0.3

EXECUTION TIME:

ONE NORMAL PASS TAKES APPROXIMATELY 12 MINUTES.

4.2

PRG2 USE PROCEDURE

- A. SET VTB6 ON-LINE.
- B. LOAD ADDRESS 000200.
- C. SET SR TO 000002; PRESS START
- D. THE PROGRAM TYPES "KEYBOARD TEST" AND STOPS AT COMMON HALT.
- E. SET ANY DESIRED SR OPTIONS, NORMAL RUN IS WITH SR = 000000.
THIS PROGRAM'S SR OPTIONS ARE:

- SR15 HALT AT END OF ROUTINE
- SR16 LOOP PROGRAM
- SR9 SELECT ROUTINE
- SR6 THROUGH SR8 = NUMBER OF ROUTINE TO BE SELECTED.

SECTION 7.2 GIVES A COMPLETE EXPLANATION OF SR OPTIONS.

- F. PRESS CONTINUE, FOLLOW TYPED INSTRUCTIONS, WHEN DONE PROGRAM STOPS AT PROGRAM END HALT.

A DESCRIPTION OF THIS PROGRAM IS GIVEN IN SECTION 8.5

EXECUTION TIME:

PROGRAM IS USER DEPENDENT.

4.3 PRG3 USE PROCEDURE

.....

- A. SET VT06 TO ON=LINE
- B. LOAD ADDRESS 000200
- C. SET SR TO 000003; PRESS START
- D. THE PROGRAM TYPES "TYPE IN DATA"
- E. KEY IN ANY FIVE CHARACTERS TO BE TYPED.
- F. KEY IN EITHER A RUBOUT FOR FULL SPEED TYPING, OR ANY OTHER CHARACTER FOR RANDOM STALLS BETWEEN CHARACTERS.
- G. 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.

A DESCRIPTION OF THIS PROGRAM IS GIVEN IN SECTION 0.0

4.4 PRG4 USE PROCEDURE

PRG4 IS USED AS AN AID IN ADJUSTING THE TRANSMITTER CLOCK,
AND IN OBSERVING THE DATA BITS AS THEY ARE SHIFTED OUT OF THE
TRANSMITTER BUFFER, A SCOPE IS REQUIRED.

TO ADJUST THE PUNCH CLOCK PROCEED AS FOLLOWS:

- A. LOAD ADDRESS 000200
- B. SET SR TO 00004, 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,

A DESCRIPTION OF THIS PROGRAM IS GIVEN IN SECTION 8.10

4.5 PRG9 USE PROCEDURE

PRG9 IS USED AS AN AID IN ADJUSTING THE RECEIVER CLOCK, AND IN OBSERVING THE DATA BITS AS THEY ARE SHIFTED INTO THE RECEIVER BUFFER, A SCOPE IS REQUIRED.

THE PROGRAM MAKES USE OF THE TRANSMIT MAINTENANCE BIT FEATURE IN ORDER TO CAUSE THE DATA OUTPUTTED TO THE TRANSMITTER BUFFER TO BE SHIFTED INTO THE RECEIVER BUFFER.

TO ADJUST THE RECEIVER CLOCK PROCEED AS FOLLOWS:

- A. LOAD ADDRESS 000200
- B. SET SR TO 000005, 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 TRANSMIT COMMANDS IN RIGHT HALF OF SR; THE SELECTED NUMBER SHOULD BE LONG ENOUGH FOR THE ENTIRE TRANSMIT/RECEIVE OPERATION TO COMPLETE, A SUGGESTED STARTING NUMBER IS 177,
- F. PRESS CONTINUE, THE PROGRAM RUNS CONTINUOUSLY, FIRST IT LOADS THE TRANSMITTER 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 TRANSMITTER BUFFER, THE RECEIVER CLOCK STARTS, AND THE DATA BITS ARE SHIFTED INTO THE RECEIVER BUFFER; AT THE END OF THE DELAY THE PROGRAM MOVES THE RECEIVER BUFFER CONTENTS TO REG B, AND ISSUES 5 RESET INSTRUCTIONS IN ORDER TO MAKE THE RECEIVER BUFFER CONTENTS VISIBLE IN THE RIGHT HALF OF THE DATA LIGHTS.
- G. SET UP A SCOPE AND DISPLAY THE RECEIVER CLOCK PULSES, ADJUST THE RECEIVER CLOCK ACCORDING TO SPECIFICATIONS.

A DESCRIPTION OF THIS PROGRAM IS GIVEN IN SECTION 0.11

4.6 PRG6 USE PROCEDURE

-
- A. LOAD ADDRESS 000200.
 - B. SET SR TO 000006; PRESS START
 - C. THE PROGRAM STOPS AT COMMON MALT.
 - D. SET CODE FOR CHARACTER TO BE TESTED IN THE LEFT HALF OF THE SR.
 - E. PRESS CONTINUE, THE PROGRAM RUNS CONTINUOUSLY, OUTPUTTING THE CHARACTER TO THE OUTPUT BUFFER AND CHECKING THAT THE RECEIVE BUFFER CONTAINS THE SAME CHARACTER WHEN THE RECEIVE DONE BIT BECOMES SET.
 - F. REFER TO SECTION 6, ERRORS, IF ERRORS OCCUR.

A DESCRIPTION OF THIS PROGRAM IS GIVEN IN SECTION 8.12

EXECUTION TIME:

CONTINUOUS RUNNING PROGRAM,

4.7 PRG7 USE PROCEDURE

-
- A. LOAD ADDRESS 000200.
 - B. SET SR TO 000007; PRESS START
 - C. THE PROGRAM RUNS CONTINUOUSLY, THE SPECIAL BINARY COUNT PATTERN IS OUTPUTTED TO THE OUTPUT BUFFER, EACH TIME THE RECEIVE DONE BIT BECOMES SET THE CHARACTER IN THE RECEIVE BUFFER IS CHECKED TO SEE THAT IT MATCHES THE PREVIOUSLY OUTPUTTED CHARACTER, THE PROGRAM STALLS RANDOMLY BETWEEN CHARACTERS, TO RUN AT FULL SPEED, SET SR0 TO A 1.
 - D. REFER TO SECTION 6, ERRORS, IF ERRORS OCCUR.

A DESCRIPTION OF THIS PROGRAM IS GIVEN IN SECTION 8.13

EXECUTION TIME:

CONTINUOUS RUNNING PROGRAM,

4.10 PRG10 USER PROCEDURE

-
- A. LOAD ADDRESS 000200.
 - B. SET SR TO 000007; PRESS START
 - C. PROGRAM RUNS CONTINUOUSLY, THE SCREEN IS FILLED WITH ALTERNATE LINES OF A CHARACTER AND ITS COMPLEMENT AND A LINE OF THE COMPLEMENT OF THE CHARACTER FOLLOWED BY THE CHARACTER. THIS TEST VERIFIES THE ROLL-UP CAPABILITY OF THE VT06.
 - D. REFER TO SECTION 6, ERRORS, IF ERRORS OCCUR.

A DESCRIPTION OF THIS PROGRAM IS GIVEN IN SECTION 8.14

EXECUTION TIME:

CONTINUOUS RUNNING PROGRAM,

5. PROGRAM AND/OR OPERATOR ACTION
.....5.1 NORMAL HALTS
.....

- LOC 001374 COMMON HALT, THIS HALT OCCURS WHENEVER THE PROGRAM IS AWAITING USER INTERVENTION, THE DATA LIGHTS CONTAIN THE ADDRESS OF INSTRUCTION THAT GENERATED THE CALL TO THE COMMON HALT.
- LOC 001504 END OF ROUTINE HALT, THIS HALT OCCURS AT THE END OF A TEST ROUTINE IF SR19 IS SET TO A 1. TO PROCEED, PRESS CONTINUE, PROGRAMS PRG0, PRG1, AND PRG2 USE THE ROUTINE END OPTION.
- LOC 002012 PROGRAM END HALT, THIS HALT NORMALLY OCCURS AT THE END OF PROGRAMS PRG0, PRG1 AND UNLESS THE LOOP PROGRAM OPTION IS SET, (SR10)

6. ERRORS
.....6.1 ERROR HALTS
.....

- LOC 001406 UNCONDITIONAL ERROR HALT, DATA LIGHTS CONTAIN ADDRESS OF INSTRUCTION THAT GENERATED THE ERROR CALL, REFER TO PROGRAM LISTING.
- LOC 001466 CONDITIONAL ERROR HALT, THIS CALL WILL ALWAYS OCCUR, UNLESS SR14 IS SET TO A 1 (SCOPE MODE) AND THE ERROR HAS OCCURRED AT LEAST ONCE, DATA LIGHTS CONTAIN ADDRESS OF INSTRUCTION THAT GENERATED ERROR CALL, REFER TO PROGRAM LISTING.
- LOC 001426 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 THE DATA LIGHTS CONTAIN THE EXPECTED 8 BIT DATA, THE RIGHT HALF CONTAINS THE RECEIVED 8 BIT DATA.

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 000176.

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 ADDRESS, 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 PRECEDES THE ONE REFERENCED BY THE DISPLAYED PC VALUE IS THE INSTRUCTION THAT WAS BEING EXECUTED WHEN THE FAILURE OCCURRED,

A NON-RECOVERABLE ERROR HALT FAILURE IS AN ABNORMAL CONDITION INDICATING A HARDWARE FAILURE, OR MOST UNLIKELY, A PROGRAM FAILURE; THIS PROGRAM ASSUMES THAT THE PROCESSOR IS IN OPERATING CONDITION IN ORDER TO TEST THE VTO6, ANY FURTHER STEPS TO DIAGNOSE A NON-RECOVERABLE ERROR ARE NOT WITHIN THE SCOPE OF THIS PROGRAM.

7. MISCELLANEOUS

7.1 SR OPTIONS

THE STANDARD SR OPTIONS ARE DESCRIBED HERE.

SR13 - HALT AT END OF ROUTINE, FOR THESE PROGRAMS CONSISTING OF A SET OF SEPARATE TEST ROUTINES, SR13 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 PRGB. THE OPTION CAUSES THE PROGRAM TO BYPASS ERROR HALTS, AND TO STAY IN THE FAILING ROUTINE. THIS OPTION WILL NOT BECOME ACTIVE UNTIL AN ERROR OCCURS. SR14 MUST BE ON BEFORE THE ERROR OCCURS, OR AT LEAST IT MUST BE SET BEFORE PRESSING CONTINUE AFTER AN ERROR HALT.

(7,2 CONT'D)

- SR13** - 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 SR13 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.
- SR10** - LOOP PROGRAM, THIS OPTION IS USED BY PROGRAMS PRG0, PRG1, AND PRG4. SETTING SR10 TO A 1 CAUSES THE PROGRAM TO REPEAT ITSELF UPON COMPLETION, INSTEAD OF STOPPING AT PROGRAM END HALT.
- SR9** - SELECT ROUTINE, THIS OPTION IS USED BY PROGRAMS PRG0, PRG1 AND PRG4. THE USER MAY ELECT TO RUN ONLY ONE SPECIFIC ROUTINE BY SETTING SR9 TO A 1, AND SR6 THROUGH SR8 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 PROGRAM PRG10. THIS PROGRAM OPERATES NORMALLY IN STALL MODE (TESTS OR EXERCISES ARE NOT FULL SPEED, BUT RANDOM DURATION DELAYS ARE INTRODUCED). SETTING SR8 TO A 1 CAUSE THE PROGRAM TO PERFORM THEIR TESTS AT FULL SPEED.

7,3

TESTING VT06 AT NON-STANDARD ADDRESSES AND/OR VECTORS

THIS PROGRAM CAN TEST A KL11 ASSIGNED TO NON-STANDARD ADDRESSES AND VECTORS PROVIDED THESE 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
001206	177560	RECEIVER CSR ADDRESS
001210	177562	RECEIVER BUFFER ADDRESS
001212	177564	TRANSMITTER CSR ADDRESS
001214	177566	TRANSMITTER BUFFER ADDRESS
001216	000060	RECEIVER VECTOR ADDRESS
001220	000200	RECEIVER PRIORITY LEVEL
001222	000064	TRANSMITTER VECTOR ADDRESS
001224	000200	TRANSMITTER PRIORITY LEVEL

- B. PROCEED TO USE PROGRAM, OR
- C. USING STANDARD DUMP ROUTINES, DUMP OUT THE ENTIRE PROGRAM IN ABSOLUTE FORMAT, TO HAVE AN UPDATED OBJECT TAPE THAT REFLECTS YOUR SYSTEM, OR
- D. DUMP OUT ONLY LOCATIONS 000204 THROUGH 000222, 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 000204 THROUGH 000222.

8. DESCRIPTION

NOTE1 THIS TEXT WAS ORIGINALLY WRITTEN TO DESCRIBE A TELETYPE;
THEREFORE, INTERPRET READER/KYBD AS RECEIVER AND PUNCH
AS TRANSMITTER,

8.1 PRGB PROGRAM DESCRIPTION

PRGB TESTS THE INPUT AND OUTPUT LOGIC IN ONE PROGRAM, THE PROGRAM
CONSISTS OF 35 TEST ROUTINES NUMBERED FROM 00 TO 42(8),

RTN0 TESTS ABILITY TO REFERENCE THE READER/KYBD STATUS WORD (TKS)
WITHOUT TRAPPING,
RTN1 TESTS ABILITY TO REFERENCE THE READER/KYBD BUFFER (TKB)
WITHOUT TRAPPING,
RTN2 TESTS ABILITY TO REFERENCE THE PRINTER/PUNCH STATUS WORD (TPS)
WITHOUT TRAPPING,
RTN3 TESTS ABILITY TO REFERENCE THE PRINTER/PUNCH BUFFER (TPB)
WITHOUT TRAPPING,
RTN4 TESTS ABILITY TO SET AND CLEAR THE READER/KYBD ID BIT,
RTN5 CHECKS THAT READER /KYBD ID BIT CAN BE CLEARED WITH RESET INSTRUCTION,
RTN6 CHECKS THAT 30 MSECS AFTER READER ENABLE THE BUSY BIT WAS SET,
RTN7 CHECKS THAT READER/KYBD BUSY BIT CAN BE READ RELIABLY,
RTN10 CHECKS THAT 0 MSECS AFTER THE BUSY BIT HAS SET THE READ
BUFFER IS RESET TO 200,
RTN11 CHECKS THAT READER DONE BIT SETS NO LATER THAN 200 MSECS
AFTER READER ENABLE,
RTN12 TESTS THAT READER/KYBD DONE BIT CAN BE READ RELIABLY,
RTN13 CHECKS THAT RESET INSTRUCTION CLEARS THE READER DONE BIT,
RTN14 CHECKS THAT REFERENCING READER BUFFER CLEARS DONE BIT,
RTN15 TESTS THAT READER DONE BIT IS CLEARED BY START BIT, (WHEN
BUSY BECOMES SET,)
RTN16 CHECKS THAT WHEN READER DONE BIT BECOMES SET, BUSY BIT IS
STILL SET,
RTN17 TESTS THAT READ BUFFER CAN BE READ RELIABLY,
RTN20 CHECKS THAT READER DONE BIT IS ABLE TO CAUSE AN INTERRUPT, IF THE
INTERRUPT IS SERVICED, IT WILL HAVE OCCURRED AT CORRECT VECTOR,
RTN21 TESTS THAT READER DONE DOES NOT CAUSE AN INTERRUPT WHEN THE PROCESSOR
IS AT THE SAME PRIORITY AS THE READER'S INTERRUPT REQUEST LEVEL,
RTN22 TESTS THAT READER DONE CAUSES INTERRUPT WHEN THE PROCESSOR IS AT A
PRIORITY ONE LEVEL LOWER THAN THE READER'S INTERRUPT REQUEST LEVEL,
RTN23 CHECKS THAT READER DONE DOES NOT REINTERRUPT AFTER RTI
INSTRUCTION WHEN DONE BIT IS LEFT SET,

(8,1 CONT'D)

RTN24 TESTS ABILITY TO SET AND CLEAR THE PUNCH ID BIT,
RTN25 CHECKS THAT PUNCH ID BIT CAN BE CLEARED WITH RESET INSTRUCTION,
RTN26 TESTS ABILITY TO SET AND CLEAR PUNCH MAINTENANCE BIT,
RTN27 CHECKS THAT RESET INSTRUCTION CLEARS THE MAINTENANCE BIT,
RTN30 TESTS THAT RESET SETS THE PUNCH READY BIT, AND THAT THE
READY BIT CAN BE READ RELIABLY,
RTN31 TESTS THAT PUNCH READY IS CLEARED BY LOADING THE PUNCH BUFFER,
RTN32 TESTS THAT BYTE LOADING PUNCH BUFFER+1 DOES NOT CLEAR THE
PUNCH READY BIT,
RTN33 CHECKS THAT THE PUNCH BECOMES READY NO LATER THAN 200 MSECS
AFTER BUFFER LOAD,
RTN34 CHECKS THAT PUNCH READY BIT CAN CAUSE INTERRUPT, IF THE INTERRUPT
IS SERVICED, IT WILL HAVE OCCURRED AT THE CORRECT VECTOR
RTN35 TESTS THAT PUNCH READY DOES NOT CAUSE AN INTERRUPT WHEN THE
PROCESSOR IS AT A PRIORITY AS THE READER'S INTERRUPT REQUEST
LEVEL,
RTN36 TESTS THAT PUNCH READY CAUSES AN INTERRUPT WHEN THE PROCESSOR
IS AT PRIORITY ONE LEVEL LOWER THAN THE PUNCH INTERRUPT
REQUEST LEVEL,
RTN40 CHECKS THAT PUNCH READY CAUSES AN INTERRUPT IMMEDIATELY UPON
LOWERING PROCESSOR PRIORITY TO 0,
RTN41 CHECKS FOR CORRECT OPERATION OF WAIT INSTRUCTION, (REFER TO
PROGRAM LISTING),
RTN42 TESTS THAT LOADING PUNCH BUFFER WITH MAINTENANCE BIT SET
CAUSES READER DONE BIT TO SET NO LATER THAN 200 MSECS.

(8,2 CONT'D)

RTN5 TYPES LINE OF CHARACTERS ABC
 RTN6 TYPES LINE OF CHARACTERS DEF
 RTN7 TYPES LINE OF CHARACTERS GHI
 RTN8 TYPES LINE OF CHARACTERS JKL
 RTN9 TYPES LINE OF CHARACTERS MNO
 RTN10 TYPES LINE OF CHARACTERS PQR
 RTN11 TYPES LINE OF CHARACTERS STU
 RTN12 TYPES LINE OF CHARACTERS VWX
 RTN13 TYPES LINE OF CHARACTERS YZ0
 RTN14 TYPES LINE OF CHARACTERS 123
 RTN15 TYPES LINE OF CHARACTERS 456
 RTN16 TYPES LINE OF CHARACTERS 789
 RTN17 TYPES LINE OF CHARACTERS !"#\$
 RTN18 TYPES LINE OF CHARACTERS %&'
 RTN19 TYPES LINE OF CHARACTERS ()
 RTN20 TYPES LINE OF CHARACTERS *+,
 RTN21 TYPES LINE OF CHARACTERS -./
 RTN22 TYPES LINE OF CHARACTERS :;<
 RTN23 TYPES LINE OF CHARACTERS =>?
 RTN24 TYPES LINE OF CHARACTERS @[\n
 RTN25 TYPES LINE OF CHARACTERS]^ AND LEFT ARROW
 RTN26 TYPES 2 LINES OF ALL CHARACTERS, FIRST LINE IS TYPED AT
 FULL SPEED, SECOND LINE IS TYPED WITH RANDOM STALLS,
 RTN27 TYPES 12 LINES OF ASR33 (BB1224-10) WORST CASE PATTERN,
 EVERY OTHER LINE IS TYPED WITH RANDOM STALLS, THE ASR33
 WORST CASE PATTERN IS /-W/W-
 RTN28 TYPES 12 LINES OF ASR35 (BB1224-11) WORST CASE PATTERN,
 EVERY OTHER LINE IS TYPED WITH RANDOM STALLS, THE ASR35
 WORST CASE PATTERN IS 'L9C7E

8.3

PRG2 PROGRAM DESCRIPTION

PRG2 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 VTOO CONTROL. THE FOLLOWING SECTIONS (8.3.1, 8.3.2)
DESCRIBE HOW THIS ROUTINE SHOULD BE USED TO TEST THE
SPECIAL CHARACTERS.

8.3.1 CURSOR TEST

- A. USING "SPACES" AND "LINE FEEDS" PUT THE LETTER "A" AT A KNOWN POSITION (5 "SPACES" AND 5 "LINE FEEDS")
- B. PLACE A "B" AT ANOTHER KNOWN POSITION (10 MORE "SPACES" AND 5 MORE "LINE FEEDS")
- C. HOME UP
- D. USING "CURSOR RIGHT" AND "CURSOR DOWN" POSITION THE CURSOR OVER THE "B". IT SHOULD REQUIRE EXACTLY 15 "CURSOR RIGHTS" AND 10 "CURSOR DOWNS".
- E. USING "CURSOR LEFT" AND "CURSOR UP" POSITION THE CURSOR OVER THE "A". IT SHOULD REQUIRE EXACTLY 5 "CURSOR-UPS" AND 10 "CURSOR LEFTS".

8.3.2 ERASE TEST

- A. FILL THE SCREEN WITH ANY CHARACTER AND RETURN CURSOR TO LEFT SIDE OF SCREEN.
- B. TYPE EOL (ERASE LINE) AND "CURSOR UP".
- C. REPEAT B 25 TIMES AND SCREEN SHOULD BE CLEAR.
- D. FILL SCREEN WITH ANY CHARACTER.
- E. "HOME UP" AND TYPE "EOF" (ERASE FIELD) AND SCREEN SHOULD BE CLEAR.

DESCRIPTION OF CURSOR CONTROL CODES AND SPECIAL FUNCTION CODES

TABLE B-3

OCTAL CODE	CORRESPONDING KEY	DESCRIPTION
07	BELL	CAUSES A "BEEP" TO SOUND IN THE SPEAKER INSIDE THE DISPLAY;
12	LINE FEED	CAUSES THE CURSOR TO MOVE DOWN ONE LINE POSITION UNTIL THE CURSOR REACHES THE BOTTOM LINE. WHEN THE CURSOR IS ON THE BOTTOM LINE, THE CODE WILL CAUSE THE TEXT TO ROLL UP ONE LINE (TOP LINE IS LOST);
15	RETURN	CAUSES THE CURSOR TO RETURN TO THE FIRST CHARACTER POSITION ON THE SAME LINE. CARRIAGE RETURN/ LINE FEED WILL NOT AUTOMATICALLY OCCUR AFTER THE SEVENTY-SECOND CHARACTER IS WRITTEN.
10	BACK SPACE	CAUSES THE CURSOR TO MOVE BACKWARD ONE CHARACTER SPACE. IF THE CURSOR IS AT CHARACTER POSITION ONE, THIS CODE HAS NO EFFECT.
37	ERASE EOF	CAUSES THE SCREEN TO BE ERASED FROM THE CURSOR POSITION TO THE END OF THE SCREEN.
36	ERASE EOL	CAUSES THE PORTION OF THE LINE FROM THE CURSOR POSITION TO THE END OF THAT LINE TO BE ERASED.
35	HOME UP	PLACES THE CURSOR IN THE FIRST CHARACTER POSITION OF THE FIRST LINE ON THE SCREEN.
34	HOME DOWN	PLACES THE CURSOR IN THE FIRST CHARACTER POSITION OF THE LAST LINE ON THE SCREEN.
32	CURSOR UP	MOVES THE CURSOR VERTICALLY UPWARD TO THE NEXT HIGHER LINE. IF THE CURSOR IS ON THE TOP LINE, THIS CODE HAS NO EFFECT.
13	CURSOR DOWN	MOVES THE CURSOR VERTICALLY DOWNWARD ONE LINE. IF THE CURSOR IS ON THE BOTTOM LINE, THIS HAS NO EFFECT.
OCTAL CODE	CORRESPONDING KEY	DESCRIPTION

31 CURSOR LEFT MOVES THE CURSOR HORIZONTALLY LEFT ONE POSITION. IF THE CURSOR IS AT CHARACTER POSITION ONE, THIS CODE HAS NO EFFECT.

38 CURSOR RIGHT MOVES THE CURSOR HORIZONTALLY RIGHT ONE POSITION. IF THE CURSOR IS AT CHARACTER POSITION 72, THIS CODE HAS NO EFFECT.

48 SPACE THE CHARACTER AT THE CURSOR POSITION WHEN THE SPACE COMMAND IS ISSUED WILL BE ERASED THE CURSOR MOVES HORIZONTALLY ONE CHARACTER POSITION TO THE RIGHT IF THE CURSOR IS IN CHARACTER POSITION 72, THAT CHARACTER AT THE 72ND POSITION IS ERASED, BUT THE CURSOR WILL REMAIN AT THE 72ND POSITION.

THE CURSOR SWITCH AT THE REAR OF THE UNIT WILL CAUSE THE VT86 TO IGNORE (NO RESPONSE ON THE SCREEN) THE FOLLOWING CODES

18 BACK SPACE
37 ERASE EOP
36 ERASE EOL
35 HOME UP
34 HOME DOWN
32 CURSOR UP
13 CURSOR DOWN
31 CURSOR LEFT
38 CURSOR RIGHT WHEN IN THE "OFF" POSITION.

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.

8.8

PRG3 PROGRAM DESCRIPTION

PRG3 IS A PRINTER EXERCISER DESIGNED AS AN AID IN MAKING VT86 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.

THIS TEST CAN ALSO BE USED FOR ALIGNMENT BY FILLING THE SCREEN WITH E'S.

8.4 PRG4 PROGRAM DESCRIPTION

PRG11 IS USED AS AN AID IN ADJUSTING THE TRANSMITTER CLOCK WITH THE AID OF A SCOPE, THE PROGRAM PERFORMS THE FOLLOWING SEQUENCE:

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

8.5 PRG5 PROGRAM DESCRIPTION

PRG9 IS USED AS AN AID IN ADJUSTING THE RECEIVER 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 B,
- E. ISSUE 9 RESET INSTRUCTIONS TO "FIX" READ BUFFER CONTENTS IN RIGHT HALF OF DATA LIGHTS,
- F. GO TO STEP A,

8.6 PRG6 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,

8.7 PRG7 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,

8.10 PROGRAM 10 DESCRIPTION

.....

THE PURPOSE OF THIS TEST IS TO VERIFY THAT THE VT80 MEMORY HAS ROLL-UP CAPABILITIES; THE TEST FUNCTIONS AS FOLLOWS:

- A. A LINE 1, AND ITS COMPLEMENT , IS DISPLAYED
- B. THIS LINE IS FOLLOWED BY A LINE OF ITS EXACT COMPLEMENT

THIS PROCEDURE RUNS CONTINUOUSLY. IF SWITCH 15 IS HELD UP MOMENTARILY THE ASCII CODE FOR THE CHARACTER IS INCREMENTED BY ONE. BY UTILIZING SWITCH 15 IN THIS MANNER, PROGRAM 10 CAN TEST THE ROLL-UP CAPABILITY OF ALL CHARACTERS.

JKL11 TESTS FOR VT86 DISPLAY TERMINAL
 |COPYRIGHT 1971, DIGITAL EQUIPMENT CORP., MAYNARD, MASS.

|
 |PRG0= COMBINED INPUT-OUTPUT LOGIC TESTS,
 |PRG1= DISPLAY TEST,
 |PRG2= KEYBOARD TEST,
 |PRG3= PRINTER EXERCISER,
 |PRG4= PUNCH CLOCK ADJUSTMENT ROUTINE,
 |PRG5= READER CLOCK ADJUSTMENT ROUTINE,
 |PRG6= MAINTENANCE MODE SINGLE CHARACTER DATA TEST,
 |PRG7= MAINTENANCE MODE SPECIAL BINARY COUNT PATTERN DATA TEST,
 |PRG10= ROLE-UP 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 8 = NUMBER OF ROUTINE TO BE SELECTED,

	000000			
000000	000002		,+2	UNASSIGNED TRAP
000002	000000		HALT	
000004	000006	MACHER1	,+2	ISP 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	002146		EMTINT	EMT TRAP
000032	000340		PRTY7	
000034	002212		TRPINT	TRAP TRAP, SIMILAR TO EMT
000036	000340		PRTY7	
	000200		,REPT 200	
			,+2	
			HALT	TRAPPED TO PREVIOUS ADDRESS.
			,ENDR	
000040	000042		,+2	
000042	000000		HALT	TRAPPED TO PREVIOUS ADDRESS.
000044	000046		,+2	
000046	000000		HALT	TRAPPED TO PREVIOUS ADDRESS.
000050	000052		,+2	
000052	000000		HALT	TRAPPED TO PREVIOUS ADDRESS.
000054	000056		,+2	
000056	000000		HALT	TRAPPED TO PREVIOUS ADDRESS.
000060	000062		,+2	
000062	000000		HALT	TRAPPED TO PREVIOUS ADDRESS.
000064	000066		,+2	

000244	000246	,+2	
000246	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000250	000252	,+2	
000252	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000254	000256	,+2	
000256	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000260	000262	,+2	
000262	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000264	000266	,+2	
000266	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000270	000272	,+2	
000272	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000274	000276	,+2	
000276	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000300	000302	,+2	
000302	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000304	000306	,+2	
000306	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000310	000312	,+2	
000312	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000314	000316	,+2	
000316	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000320	000322	,+2	
000322	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000324	000326	,+2	
000326	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000330	000332	,+2	
000332	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000334	000336	,+2	
000336	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000340	000342	,+2	
000342	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000344	000346	,+2	
000346	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000350	000352	,+2	
000352	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000354	000356	,+2	
000356	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000360	000362	,+2	
000362	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000364	000366	,+2	
000366	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000370	000372	,+2	
000372	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000374	000376	,+2	
000376	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000400	000402	,+2	
000402	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000404	000406	,+2	
000406	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000410	000412	,+2	
000412	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000414	000416	,+2	
000416	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000420	000422	,+2	

000422	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000424	000420	,+2	
000426	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000430	000432	,+2	
000432	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000434	000436	,+2	
000436	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000440	000442	,+2	
000442	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000444	000446	,+2	
000446	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000450	000452	,+2	
000452	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000454	000456	,+2	
000456	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000460	000462	,+2	
000462	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000464	000466	,+2	
000466	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000470	000472	,+2	
000472	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000474	000476	,+2	
000476	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000500	000502	,+2	
000502	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000504	000506	,+2	
000506	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000510	000512	,+2	
000512	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000514	000516	,+2	
000516	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000520	000522	,+2	
000522	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000524	000526	,+2	
000526	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000530	000532	,+2	
000532	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000534	000536	,+2	
000536	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000540	000542	,+2	
000542	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000544	000546	,+2	
000546	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000550	000552	,+2	
000552	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000554	000556	,+2	
000556	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000560	000562	,+2	
000562	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000564	000566	,+2	
000566	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000570	000572	,+2	
000572	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
000574	000576	,+2	
000576	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.

000756	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000760	000762	,+2	
000762	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000764	000766	,+2	
000766	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000770	000772	,+2	
000772	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
000774	000776	,+2	
000776	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
001000	001002	,+2	
001002	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
001004	001006	,+2	
001006	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
001010	001012	,+2	
001012	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
001014	001016	,+2	
001016	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
001020	001022	,+2	
001022	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
001024	001026	,+2	
001026	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
001030	001032	,+2	
001032	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.
001034	001036	,+2	
001036	000000	HALT	ITRAPPED TO PREVIOUS ADDRESS.

IEQUATE STATEMENTS

177570	SR=177570
177776	CC=177776
177776	PSW=177776
000240	NOP=240
000000	OPEN=0
100000	HLTSH=BIT15
040000	SCOPSH=BIT14
020000	NPRYSH=BIT13
010000	NTRCSH=BIT12
004000	NITRSH=BIT11
002000	LPRGSH=BIT10
001000	SRTSH=BIT9
000400	BYPMAN=BIT8
100000	MANUAL=BIT15
100000	BIT15=100000
040000	BIT14=40000
020000	BIT13=20000
010000	BIT12=10000
004000	BIT11=4000
002000	BIT10=2000
001000	BIT9=1000
000400	BIT8=400
000200	BIT7=200
000100	BIT6=100
000040	BIT5=40
000020	BIT4=20
000010	BIT3=10
000004	BIT2=4
000002	BIT1=2
000000	BIT0=0
005726	POPSP=5726
022626	POPSP2=022626
000340	PRTY7=340
000300	PRTY6=300
000240	PRTY5=240
000200	PRTY4=200
000140	PRTY3=140
000100	PRTY2=100
000040	PRTY1=40
000000	PRTY0=0
104000	TYPE=EMT+0
104001	TYPE=EMT+1
104002	STALL=EMT+2
104003	ERROR=EMT+3
104004	DATCHK=EMT+4
104005	CHALT=EMT+5
104006	STRDRV=EMT+6
104007	STPCHV=EMT+7
104010	EHALT=EMT+10
104011	SRESET=EMT+11
104012	CHAIN=EMT+12
104013	CK33=EMT+13

IHALT SWITCH DEFINITION
 ISCOPE SWITCH DEFINITION
 IINHBIT PRINT SWITCH DEFINITION
 IINHBIT TRACE SWITCH DEFINITION
 IINHBIT ITERATION SWITCH DEFINITION
 ILOOP PROGRAM SWITCH DEFINITION
 ISELECT ROUTINE SWITCH DEFINITION
 IBYPASS MANUAL INTERVENTION DEFINITION.

IPOP THE STACK, SAME AS TST (6)+
 IPOP STACK TWICE, SAME AS CMP (6)+.(6)+
 IPRIORITY LEVEL DEFINITIONS

104014	CK35=EMT+14
104016	TYPLN3=EMT+16
104017	DATHLT=EMT+17
104020	SAVREG=EMT+20
104021	RSTREG=EMT+21
104022	CHKASR=EMT+22
104400	DELAY=TRAP+0
000007	BELL=007
011671	BLOCKA=DEND
011673	BLOCK1=BLOCKA+2
012003	BLOCK8=BLOCKA+112
012014	BLK8B=BLOCKA+123
012005	BLOCK2=BLOCKA+114
012016	BLK2=BLOCKA+125
012115	BLOCKC=BLOCKA+224
012126	BLKCC=BLOCKA+235

000200	000200	
000200	000167	001304
	001204	
001204	000000	
001206	177560	
001210	177562	
001212	177564	
001214	177566	
001216	000060	
001220	000200	
001222	000064	
001224	000200	
001226	000001	
001230	000000	
001232	000000	
001234	000000	
001236	000000	
001240	000000	
001242	000000	
001244	000000	
001246	000000	
001250	004526	
001252	006210	
001254	007604	
001256	010104	
001260	010264	
001262	010274	
001264	010364	
001266	010436	
001270	010540	

```

,0200
JMP START
,0,+1000
SPBOTI 0
TKSI 177560
TKBI 177562
TPSI 177564
TPBI 177566
TKVTRI 60
TKLVL1 PRTY4
TPVTRI 64
TPLVL1 PRTY4
TTYTYP1 01
PRGNUM1 OPEN
KSTART1 OPEN
CURTST1 OPEN
RTNNO1 OPEN
NXTST1 OPEN
ICTRI OPEN
SCOPTRI OPEN
PRGID1 OPEN
PRGTAB1 PRG0
PRG1
PRG2
PRG3
PRG4
PRG5
PRG6
PRG7
PRG10
    
```

```

GO TO START OF PROGRAM;
GET CODE OUT OF VECTOR AREA
BOTTOM OF STACK
ILSR CSR
ILSR BUFFER
ILSP CSR
ILSP BUFFER
ILSR INTERRUPT VECTOR
ILSR PRIORITY LEVEL
ILSP INTERRUPT VECTOR
ILSP PRIORITY LEVEL
ITTY 0 KSR35
ICONTAINS CURRENT PROGRAMS
ICURRENT PROGRAM START ADDRESS;
ICONTAINS ADDR OF CURRENT TEST;
ICONTAINS CURRENT TEST 0.
ICONTAINS ADDR OF NEXT TEST.
ICONTAINS CURRENT ITERATION COUNT
ICONTAINS CURRENT SCOPE POINTER.
ICONTAINS PROGRAM INDICATORS
IPRG0 START ADDRESS
IPRG1 START ADDRESS
IPRG2 START ADDRESS
IPRG3 START ADDRESS
IPRG4 START ADDRESS
IPRG5 START ADDRESS
IPRG6 START ADDRESS
IPRG7 START ADDRESS
IPRG10 START ADDRESS
    
```

001272 002744
 001274 003000
 001276 003104
 001300 001432
 001302 001412
 001304 001300
 001306 002474
 001310 002524
 001312 001400
 001314 002554
 001316 001672
 001320 002256
 001322 002272
 001324 002270
 001326 004244
 001330 001422
 001332 002334
 001334 002374
 001336 002310
 001340 003120
 001342 000000
 001344 000000
 001346 000000
 001350 000000
 001352 000000
 001354 000000
 001356 000000
 001360 000000
 001362 000000
 001364 000000

EMTTAB: TYP
 TYP
 TYP
 STAL
 ERR
 DTCHK
 CHLT
 STLSRV
 STLSPV
 ENLT
 SRSETT
 CHAINN
 CHK33
 CHK35
 CHK330
 TYPL3
 DTHLT
 SAVRG
 RSTRG
 CKASR
 TRPTAB: DLY
 RCNT: OPEN
 CRBUF: OPEN
 CHR1: OPEN
 CHR2: OPEN
 CHR3: OPEN
 ERCTR: OPEN
 CTRAI: OPEN
 CTRBI: OPEN
 CTRCI: OPEN
 CTRDI: OPEN

I POINTER TO TIMEOUT ROUTINE
 I POINTER TO CHAINED MESSAGES ROUTINE
 I POINTER TO RANDOM STALL ROUTINE
 I POINTER TO ERROR ROUTINE

I COMMON HALT

I POINTER TO ERROR HALT ROUTINE;

I CHARACTER COUNT
 I HOLDS ONE CHARACTER FROM READER.

001366	011600			COMMON HALT ROUTINE	
001370	102700	000002		CHLT1 MOV 0X0,X0	IDEVELOP ADDRESS OF CALLER;
001374	000000			SUB #2,X0	
001376	000002			HALT	IHALT, ADDRESS OF CALL INSTRUCTION
				RTI	IN DATA LIGHTS,
001400	011600			UNCONDITIONAL ERROR HALT ROUTINE,	
001402	102700	000002		EHLT1 MOV 0X0,X0	IDEVELOP ADDRESS OF CALLER;
001406	000000			SUB #2,X0	
001410	000002			HALT	IHALT, ADDR OF ERROR CALL
				RTI	IN DATA LIGHTS,
001412	126767	177726	177725	DATA CHECK ROUTINE,	
001420	001403			DTCHK1 CMPB CRBUF,CRBUF+1	ICOMPARE EXPECTED AND RECEIVED
001422	016700	177716		BEG DTCHKA	ICHARS; BRANCH IF SAME,
001426	000000			DTHLT1 MOV CRBUF,X0	IMOVE S/B AND HAS CHARS TO RB,
				HALT	IDATA ERROR HALT, GOOD CHAR IN
					IDATA LIGHTS LEFT, BAD CHAR IN DATA
					ILIGHTS RIGHT, EXIT,
001430	000002			DTCHKA1 RTI	
001432	032767	040000	176130	CONDITIONAL ERROR HALT,	
001440	001404			ERR1 BIT #SCOPSW,SR	ICHECK SCOPE SWITCH,
001442	005767	177600		BEG ERRA	IBRANCH IF NO SCOPE DESIRED,
001446	100001			TST PRGID	ISCOPING WANTED, FIRST ERROR;
001450	000002			BPL ERRA	INO SCOPE IF FIRST ERROR,
001452	052767	100000	177566	ERRA1 BIT #BIT15,PRGID	ISCOPE EXIT,
001460	011600			MOV 0X0,X0	ISSET ERROR INDICATOR,
001462	102700	000002		SUB #2,X0	IDEVELOP CALLER'S ADDRESS,
001466	000000			HALT	
001470	000002			RTI	IERROR HALT,
					EXIT,
001472	005767	176072		ROUTINE END HALT SUBROUTINE,	
001476	100003			SHLT1 TST SR	ICHECK HALT SWITCH,
001500	116700	177532		BPL SHLTA	IBRANCH IF NO HALT DESIRED,
001504	000000			MOV# RYNN0,X0	ICURRENT TEST # TO RB,
001506	000207			HALT	ROUTINE END HALT,
				SHLTA1 RTS X7	EXIT,

001510	012700	001204		STARTI	MOV	#SPBOT,X6	ISET BOTTOM OF SP STACK.
001514	005067	176250			CLR	PSW	
001520	012767	000000	176250		MOV	#0,MACHER	
001526	005067	177504			CLR	RTNNO	
001532	016700	176032			MOV	SR,X0	I(SR) TO R0
001536	042700	177760			BIC	#177760,X0	ILIMIT (SR) TO BITS 3-0
001542	020027	000014			CHP	X0,#14	ICOMPARE (SR) TO PROGRAM LIMIT
001546	101402				BLOS	CRTA	IINVALID PROGRAM NUMBER?
001550	104010			INCPRGI	EHALT		INO, INCORRECT PRG NUMBER
001552	000750				BR	START	ISTART OVER.
001554	005067	177460		CRTAI	CLR	PRGID	
001560	010067	177444			MOV	X0,PRGNUM	ISAVE PROGRAM NUMBER AT PRGNUM
001564	006100				ROL	X0	IROX2
001566	000170	001250			JMP	OPRGTAB(0)	IGO TO SELECTED PROGRAM.
001572	104009			SRSETI	CHALT		ISET SR OPTIONS DESIRED
001574	016767	177432	177436	GETRDYI	MOV	KSTART,NXTST	IADDR OF 1ST ROUTINE TO NXTST
001002	000167	000314			JMP	CLEAN	IGO CLEAN UP.
001006	004767	000204		GTRDYAI	JSR	X7,FORWD	IROLL FORWARD TO "NEXT" ROUTINE.
001012	032767	001000	175750	GTRDYBI	BIT	#SR7SW,SR	I CHECK FOR SELECT ROUTINE SWITCH
001020	001003				BNE	GTRDYC	I BRANCH IF SELECT ROUTINE SWITCH IS SET.
001022	004767	000240			JSR	X7,GOYST	IGO RUN CURRENT ROUTINE.
001026	000455				BR	CHNB	INO GO, MANUAL RTN BYPASSED.
001030	016700	175734		GTRDYCI	MOV	SR,X0	I(SR) TO R0
001034	042700	177600			BIC	#177600,X0	IMASK UNDESIRED BITS
001040	126700	177372			CHP	RTNNO,X0	ICOMPARE RTNNO TO (R0)
001044	001004				BNE	GTRDYD	I BRANCH IF ROUTINE NOT FOUND YET.
001046	004767	000222			JSR	X7,GOYST	IGO RUN ROUTINE.
001052	104010				EHALT		INO GO, MANUAL RTN SELECTED BYPASSED.
001054	000747				BR	GETRDY	
001056	022767	177777	177354	GTRDYDI	CHP	#-1,NXTST	INO, CHECK FOR LAST ROUTINE.
001064	001350				BNE	GTRDYA	ILAST ROUTINE?
001066	104010			INCRTNI	EHALT		IYES, INCORRECT ROUTINE SELECTED.
001070	000741				BR	GETRDY	ISTART OVER.
001072	005767	177350		CHAINNI	TST	PRGID	I TEST ERROR BIT IN PRGID.
001076	100013				BPL	CHNA	I BRANCH IF ERROR BIT NOT SET.
001700	032767	040000	175662		BIT	#SCOPSW,SR	I ERROR BIT SET, CHECK FOR SCOPE OPTION.
001706	001407				BEO	CHNA	ISCOPE SWITCH SET IN SR?
001710	022767	177777	177326		CHP	#-1,SCOPTR	IYES, CHECK SCOPE ENTRY POINTER
001716	001403				BEO	CHNA	I BRANCH IF SCOPE ENTRY IS -1.
001720	017716	177320			MOV	0SCOPTR,0X6	ISET UP TO GO SCOPING
001724	000002				RTI		IGO TO SCOPE ENTRY.
001726	042767	100000	177312	CHNAI	BIC	#BIT15,PRGID	ICLEAR ERROR BIT IN PRGID.
001734	032767	004000	175620		BIT	#NITRSH,SR	I TEST INHIBIT ITERATION SWITCH
001742	001004				BNE	CHNAA	IINHIBIT ITERATION?
001744	005367	177272			DEC	ICTR	INO
001750	001401				BEO	CHNAA	ICOUNT 0?
001752	000002				RTI		INO, RETURN TO TEST ROUTINE
001754	022620			CHNAAI	POPSP2		IPOP STACK TWICE
001756	004767	177510			JSR	X7,SHALT	IGO HALT IF HALT SWITCH IS SET

001762	032767	001000	175600	CHNDI	BIT	#SRTSW,SR	ICHECK SELECT ROUTINE SWITCH
001770	001301				BNE	GETRDY	ISELECT ROUTINE SWITCH SET?
001772	022767	177777	177240		CHP	#=1,NXTST	INO,
002000	001300				BNE	GTRDYA=4	ILAST TEST?
002002	032767	002000	175560		BIT	#LPRGSW,SR	IYES, TEST LOOP PROGRAM SWITCH,
002010	001271				BNE	GETRDY	ILOOP PROGRAM?
002012	000000			PRGENDI	HALT		INO, PROGRAM END.
002014	000762				BR	CHNB	
002016	016705	177210		FORWDI	MOV	NXTST,X5	IADDR OF NEXT ROUTINE TO R5,
002022	012567	177210			MOV	(5)+,R5NNO	IGET NEXT ROUTINE NUMBER,
002026	012567	177200			MOV	(5)+,NXTST	IGET ADDR OF NEXT "NEXT" ROUTINE,
002032	105767	177210			TSTB	PRGID	ICHECK IF PROGRAM SCOPE AND I COUNT
002036	100407				BMI	FORWDB	IPARAMETERS, BRANCH IF NOT,
002040	012567	177170			MOV	(5)+,ICTR	IGET ITERATION COUNT,
002044	012567	177174			MOV	(5)+,SCOPTR	IGET SCOPE LOOP ENTRY POINTER,
002050	010567	177160		FORWDAI	MOV	X5,CURTST	IADDR OF NOW CURRENT TEST TO CURTST,
002054	000207				RTS	X7	IEXIT FORWD SUBROUTINE,
002056	012767	177777	177160	FORWDBI	MOV	#=1,SCOPTR	IFORCE "NO SCOPE"
002064	012767	000001	177150		MOV	#1,ICTR	IFORCE I COUNT OF 1
002072	000766				BR	FORWDA	
002074	005767	177130		GOTSTI	TST	R5NNO	ICHECK FOR MANUAL RTN,
002100	100005				BPL	GOTSTA	IBRANCH IF NOT MANUAL RTN,
002102	032767	000400	175460		BIT	#BYPAN,SR	IMANUAL RTN, BYPASS IT?
002110	001401				BEO	GOTSTA	INO, RUN IT,
002112	000207				RTS	X7	IBYPASS MANUAL ROUTINE.
002114	005726			GOTSTAI	POPSP		
002116	000177	177112			JMP	#CURTST	IGO RUN TEST,
002122	012767	000000	175654	CLEANI	MOV	#6,MACHER	IRESET MACHER TRAP,
002130	005067	175642			CLR	PSW	
002134	012706	001204			MOV	#SPBOT,X6	ISET UP BOTTOM OF STACK,
002140	104011				SRESET		
002142	000167	177440			JMP	GTRDYA	
002146	011646			EMTINTI	MOV	0(X6),-(6)	IGET SAVED PC,
002150	102716	000002			SUB	#2,0X6	IDECREMENT PC BY 2,
002154	017616	000000			MOV	0(6),0X6	
002160	121627	000022			CHPB	0X6,#22	ICHECK THAT CALL IS
002164	101402				BLOS	EMTA	IWITHIN LIMITS,
002166	000000				HALT		ICALL NOT WITHIN LIMITS.
002170	000776				BR	,=2	
002172	006116			EMTAI	ROL	0X6	IEMT ARG X 2,
002174	042716	177001			BIC	#177001,0X6	IREMOVE 7 MSB,
002200	062716	001272			ADD	#EMTTAB,0X6	IFORM EMT RTN ADDR,
002204	017616	000000			MOV	0(6),0X6	
002210	000136				JMP	0(6)+	IGO TO EMT ROUTINE,
002212	011646			TRPINTI	MOV	0X6,-(6)	IGET SAVED PC,
002214	102716	000002			SUB	#2,0X6	IDECREMENT PC BY 2,
002220	017616	000000			MOV	0(6),0X6	
002224	121627	000000			CHPB	0X6,#8	ICHECK THAT EMT
002230	101402				BLOS	TRPA	IIS WITHIN LIMITS,
002232	000000				HALT		ITRAP CALL NOT IN LIMIT.
002234	000776				BR	,=2	

002236	006110		TRPAI	ROL	0X6		I TRAP ARG X 2,
002240	042710	177001		BIC	#177001,0X6		I REMOVE 7 MSB,
002244	042710	001340		ADD	STRPTAB,0X6		I FORM TRAP RTN ADDR,
002250	017610	000000		MOV	0(0),0X6		
002254	000130			JMP	0(0)+		I GO TO TRAP ROUTINE,
002256	005767	176744	CHK331	TST	TTYTYP		I CHECK FOR 33,
002262	001002			BNE	,+0		I BRANCH IF NOT 33,
002264	042710	000002		ADD	#2,0X6		I +2 TO EXIT POINTER
002270	000002		CHK3301	RTI			I EXIT
002272	022767	000001	176720	CHK331	CMP	#1,TTYTYP	I CHECK FOR 35,
002300	001002			BNE	,+0		I BRANCH IF NOT 35,
002302	042710	000002		ADD	#2,0X6		I +2 TO EXIT POINTER
002306	000002			RTI			I EXIT
002310	032767	000010	176710	CKASRI	BIT	#BIT3,TTYTYP	I CHECK FOR ASR TTY,
002316	001001			BNE	,+4		I BRANCH IF NOT ASR,
002320	000002			RTI			I ASR, EXIT,
002322	022620			POPSP2			I POP STACK TWICE,
002324	012767	000001	176710	MOV	#1,ICTR		I FORCE I COUNT TO A 1,
002332	104012			CHAIN			I CHAIN TO BYPASS ROUTINE,
				ISAVE REGS 0 TO 4	SUBROUTINE,		
002334	012667	000030	SAVRG1	MOV	(0)+,SVRPC		I SAVE PC AND PSW,
002340	012667	000026		MOV	(0)+,SVRPSW		
002344	010446			MOV	X4,=(0)		I SAVE REGS 0 - 4
002346	010346			MOV	X3,=(0)		I IN STACK,
002350	010246			MOV	X2,=(0)		
002352	010146			MOV	X1,=(0)		
002354	010046			MOV	X0,=(0)		
002356	016746	000010		MOV	SVRPSW,=(0)		I RESTORE PC AND PSW,
002362	016746	000002		MOV	SVRPC,=(0)		
002366	000002			RTI			I EXIT,
002370	000000		SVRPC1	OPEN			
002372	000000		SVRPSW1	OPEN			
				I RESTORE REGS 0 TO 4	SUBROUTINE,		
002374	012667	000030	RSTRG1	MOV	(0)+,RSTPC		I SAVE PC AND PSW,
002400	012667	000026		MOV	(0)+,RSTPSW		
002404	012600			MOV	(0)+,X0		I RESTORE REGS 0 - 4
002406	012601			MOV	(0)+,X1		I FROM STACK,
002410	012602			MOV	(0)+,X2		
002412	012603			MOV	(0)+,X3		
002414	012604			MOV	(0)+,X4		
002416	016746	000010		MOV	RSTPSW,=(0)		I RESTORE PC AND PSW,
002422	016746	000002		MOV	RSTPC,=(0)		
002426	000002			RTI			I EXIT
002430	000000		RSTPC1	OPEN			
002432	000000		RSTPSW1	OPEN			

```

002434 012767 000310 000300  IROUTINE TO FETCH A CHARACTER
002442 005277 176540  AREADI  MOV     #200,,BRCTR  ISET UP DELAY COUNT.
002446 105777 176534  ARDAI   INC     #TKS           IENABLE READER.
002452 100407  BHI     ARDB           ICHECK DONE BIT.
002454 104400  DELAY  IBRANCH IF DONE.
002456 000001  1         IDELAY 1 MILLISECOND.
002460 005367 000256  DEC     BRCTR         ITIME UP?
002464 001370  BNE     ARDA         IBRANCH IF TIME NOT UP YET.
002466 104010  EHALT  IERROR, NO RESPONSE FROM READER.
002470 000761  BR      AREAD        ITRY AGAIN.
002472 000207  ARDBI  RTS     X7     IEXIT
IROUTINE TO SET LSR INTERRUPT VECTOR AND PRIORITY
002474 017667 000000 000012  STLSRVI MOV     0(0),STPRA+2  IMOVE VECTOR ADDR TO STPRA+2
002502 062710 000002  ADD     #2,#X0         ISET UP EXIT
002506 016701 176504  MOV     TKVTR,X1
002512 012721 000000  STPRAI MOV     #OPEN,(1)+  ISET VECTOR ADDRESS
002516 016721 176476  MOV     TKLVL,(1)+     ISET PRIORITY
002522 000002  RTI     IEXIT
IROUTINE TO SET LSP INTERRUPT VECTOR AND PRIORITY.
002524 017667 000000 000012  STLSPVI MOV     0(0),STPPA+2  IMOVE VECTOR ADDR TO STPPA+2
002532 062710 000002  ADD     #2,#X0         ISET UP EXIT
002536 016701 176460  MOV     TPVTR,X1
002542 012721 000000  STPPAI MOV     #OPEN,(1)+  ISET VECTOR ADDRESS.
002546 016721 176452  MOV     TPLVL,(1)+     ISET PRIORITY
002552 000002  RTI     IEXIT.
IROUTINE TO ISSUE RESET.
002554 012700 052525  SRSETTI MOV     #52525,X0  IDATA TO RB.
002560 005100  COM     X0             ICOMPLEMENT (RB),
002562 010067 177770  MOV     X0,SRSETT+2  I(RB) TO SRSETT+2.
002566 000005  RESET  IISSUE RESET, (RB) IS
002570 000002  RTI     IDISPLAYED, EXIT.
IRANDOM NUMBER GENERATOR, ROUTINE EXITS WITH NUMBER IN REGISTER 0.
002572 016700 000042  RNGENI MOV     RP1,X0
002576 006100  ROL     X0
002600 006100  ROL     X0
002602 066700 000034  ADD     RP2,X0
002606 010067 000026  MOV     X0,RP1
002612 006100  ROL     X0
002614 006100  ROL     X0
002616 066700 000020  ADD     RP2,#0
002622 006100  ROL     X0
002624 006100  ROL     X0
002626 010067 000010  MOV     X0,RP2
002632 016700 000002  MOV     RP1,X0
002636 000207  RTS     X7             IEXIT, NUMBER IN RB
002640 001233  RP1I   1233
002642 007622  RP2I   7622

```

```

002044 104000 BREADI STDRV ;SET READER VECTOR
002046 002712 BREADB ;TO BREADB
002050 092777 000101 176330 B10 #101,0TKS ;ENABLE LSR AND LSRI,
002056 012767 177777 000050 MOV #177777,BRCTR ;DELAY APPROX, 400 MSECS,
002064 009367 000052 DEC BRCTR
002070 001379 BNE ,-4
002072 009077 176310 CLR 0TKS ;CLEAR LSRI ENABLE,
002076 104010 EHALT ;NO RESPONSE HALT,
002700 000761 BR BREAD ;TRY AGAIN,
002702 117767 176302 176434 BREADAI MOVB 0TKB,CBUF ;CHAR READ TO CBUF,
002710 000207 RTS X7 ;EXIT SUBROUTINE,
002712 009077 176270 BREADBI CLR 0TKS ;CLEAR LSR INTERRUPT ENABLE,
002716 109777 176264 TSTB 0TKS ;TEST FOR DONE,
002722 100003 BPL BREADC ;BRANCH IF DONE NOT SET,
002724 012716 002702 MOV #BREADA,0X6 ;MODIFY INTERRUPT EXIT TO BREADA,
002730 000002 RTI ;OK, EXIT INTERRUPT,
002732 000000 BREADCI HALT ;HALT, DONE BIT NOT SET AFTER INTERRUPT,
002734 012716 002644 MOV #BREAD,0X6 ;SET UP TO RETRY,
002740 000002 RTI ;EXIT INTERRUPT,
002742 000000 BRCTR: OPEN ;SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE PRINTER,
;SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE PRINTER,
;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS,
;SET UP EXIT,
;ADDRESS OF MESSAGE TO RB,
;GET CHARACTER
;CHECK FOR"0"CHARACTER
;BRANCH IF NOT"0",
;TERMINATOR CHAR, DONE, EXIT.
;CHECK FOR"X",
;BRANCH IF"X"
;NOT"X",CHECK FOR"0",
;BRANCH IF "0"
;TYPE CHAR IN TYPDAT
002744 011000 TYP: MOV 0X6,XB
002746 062716 000002 ADD #2,0X6
002752 011000 MOV 0X0,XB
002754 112067 000104 TYPAI MOVB (0)+,TYPDAT
002760 122767 000100 000076 CMPB #100,TYPDAT
002766 001001 BNE TYPC
002770 000002 RTI
002772 122767 000045 000064 TYPCI CMPB #49,TYPDAT
003000 001416 BEQ TYPF
003002 122767 000043 000054 CMPB #43,TYPDAT
003010 001417 BEQ TYPG
003012 004767 000002 JSR X7,TYPD
003016 000756 BR TYPA
003020 116777 000040 176166 TYPDI MOVB TYPDAT,0TPB
003026 109777 176160 TSTB 0TPB
003032 100379 BPL ,-4
003034 000207 RTS X7
003036 112767 000015 000020 TYPFI MOVB #15,TYPDAT
003044 004767 177750 JSR X7,TYPD
003050 112767 000012 000000 TYPGI MOVB #12,TYPDAT
003056 004767 177730 JSR X7,TYPD
003062 000734 BR TYPA
003064 000000 TYPDAT: OPEN ;SUBROUTINE TO OUTPUT A SERIES OF ASCII MESSAGES ON TELETYPE PRINTER
;SUBROUTINE TO OUTPUT A SERIES OF ASCII MESSAGES ON TELETYPE PRINTER
;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS
;UPDATE TO NEXT MESSAGE ADDRESS
;ADDRESS OF MESSAGE TO TYPB
;CHECK FOR TERMINATOR
;BRANCH IF NOT TERMINATOR,
;TERMINATOR, EXIT
;CALL ON TYP SUB TO TYPE MESSAGE
;ADDRESS OF MESSAGE GOES HERE
;GO PROCESS NEXT MESSAGE
003066 011000 TYP: MOV 0X0,XB
003070 002716 000002 ADD #2,0X6
003074 011067 000014 MOV 0X0,TYPB
003100 022767 177777 000006 CMP #-1,TYPB
003106 001001 BNE TYPB
003110 000002 RTI
003112 104000 TYPB: TYPE
003114 000000 TYPB: OPEN
003116 000763 BR TYPB

```

```

)SUBROUTINE TO DELAY A SPECIFIED NUMBER OF MILLISECONDS
DLYI  MOV  0X6,DLCNT  IGET DELAY COUNT ADDRESS.
      ADD  #2,0X6    ISET UP EXIT ADDRESS
      MOV  0DLCNT,-(6) IDELAY COUNT TO STACK
      CLR  PSW       ISET PRIORITY 0
DLYAI MOV  #226,-(6)  I1 MSEC COUNT TO STACK
DLYBI DEC  0X6       IDECREMENT 1 MSEC COUNT
      BNE  DLYB      IBRANCH IF NOT 0.
      POPSP          IZERO; UNCOVER MSECS. COUNT.
      DEC  0X6       IDECREMENT IT
      BNE  DLYA      IBR IF NOT DONE DELAYING
      POPSP          IDONE
      RTI           IEXIT
DLCNTI OPEN          ICONTAINS MILLISECONDS COUNT ADDRESS.
)SUBROUTINE TO STALL A RANDOM NUMBER OF MILLISECONDS; MAXIMUM STALL
)DETERMINED BY CONTENTS OF LOC STLMASK;
STALI  BIT  #BIT14,PRGID ITEST FOR STALLS ALLOWED.
      BNE  STALAA     IALLOWED.
      RTI           INOT ALLOWED.
STALAAI JSR  X7,RNGEN  IGO GET RANDOM NUMBER.
      BIC  STLMASK,X8  I# IN R0, APPLY STALL MASK.
      BEQ  STALB      IBRANCH IF RESULT IS 0.
      MOV  X8,STALA
      DELAY
STALAI  OPEN          IDELAY
STALBI  RTI           IDELAY COUNT
STLMSKI OPEN          IDONE, EXIT,
)SUBROUTINE TO GENERATE RANDOM CHARACTER COUNT
GRCNTI JSR  X7,RNGEN  IGET RANDOM NUMBER
      BIC  RCHMSK,X8  IAPPLY MASK
      BEQ  GRCNT      ITRY AGAIN IF RESULT 0
      MOV  X8,RNCNT   ICOUNT TO RNCNT
      RTS  X7         IEXIT
RCHMSKI OPEN          IRANDOM CHARACTER MASK.
RNCNTI  OPEN          IRANDOM CHARACTER COUNT.
)SUBROUTINE TO COMPARE DATA READ FROM READER AGAINST EXPECTED DATA AND REPORT ERRORS;
BCHECKI JSR  X7,GTBIN  IGET BIN CHARACTER(IN R0)
      MOVB X8,CRBUF+1  I#0 CHAR TO CRBUF+1
      CMPB CRBUF,CRBUF+1 ICOMPARE S/B AND WAS CHARS;
      BNE  ,+6        IBRANCH IF NOT SAME;
      RTS  X7         ISAME, EXIT,
      DATHLT          I#0 HALT AND DISPLAY DATA.
      DEC  ERCTR      I3 ERRORS?
      BNE  ,+6        IBRANCH IF NOT 3 YET.
      JSR  X7,BSYNC   I3 ERRORS; RESYNC READER.
      RTS  X7         IEXIT,

```

```

)SUBROUTINE TO SYNC THE LSR TO A SPECIAL BINARY COUNT PATTERN TEST TAPE.
003310 004767 000162 )BSYNCI JSR X7,INBIN )INITIALIZE BINARY PATTERN
003314 004767 177324 JSR X7,BREAD )READ CHAR AND STORE AT CHR1
003320 116767 176020 176020 MOVB CRBUF,CHR1
003326 004767 177312 JSR X7,BREAD )READ CHAR AND STORE AT CHR2
003332 116767 176000 176010 MOVB CRBUF,CHR2
003340 004767 177300 JSR X7,BREAD )READ CHAR AND STORE AT CHR3.
003344 116767 175774 176000 MOVB CRBUF,CHR3
003352 004767 000012 JSR X7,SYNCA )GO SYNC
003356 000754 BR BSYNC )NO SYNC, TRY AGAIN.
003360 012767 000003 175766 MOV #3,ERCTR
003366 000207 RTS X7
003370 012767 001000 000074 SYNCAI MOV #512,,SYCTRA )SUCCESS,EXIT,
003376 012767 000012 000070 MOV #10,,SYCTRB )512 TO SYCTRA
003404 004767 000124 )SYNCBI JSR X7,GTBIN )10 TO SYCTRB
003410 120067 175732 CMPB X0,CHR1 )GET BIN CHARACTER(CHAR IN RB)
003414 001373 BNE SYNCC )COMPARE TO CHR1
003416 004767 000112 JSR X7,GTBIN )BRANCH IF NOT EQUAL
003422 120067 175722 CMPB X0,CHR2 )SAME, GET ANOTHER BIN CHAR,
003426 001409 BEQ SYNCD )COMPARE TO CHR2
003430 005367 000030 DEC SYCTRA )BRANCH IF EQUAL
003434 001363 BNE SYNCC )DECREMENT SYCTRA
003436 104010 )SYNCBI EHALT )BRANCH IF NOT DONE 512 TIMES,
003440 000207 RTS X7 )DONE 512, SYNC ERROR,
003442 004767 000066 )SYNCBI JSR X7,GTBIN )ERROR EXIT
003446 120067 175700 CMPB X0,CHR3 )GET BIN CHARACTER
003452 001404 BEQ SYNCE )COMPARE TO CHR3
003454 005367 000014 DEC SYCTRB )BRANCH IF SAME
003460 001351 BNE SYNCC )DECREMENT SYCTRB,
003462 000765 BR SYNCC )BRANCH IF NOT DONE 10 TIMES
003464 062716 000002 )SYNCBI ADD #2,0X0 )SYNC ERROR, BRANCH
003470 000207 RTS X7 )SET UP SUCCESS EXIT
003472 000000 )SYCTRAI OPEN )EXIT.
003474 000000 )SYCTRBI OPEN

)SUBROUTINE TO INITIALIZE BINARY COUNT PATTERNS
003476 012767 177777 000014 INBINI MOV #-1,RIND )SET ALL VARIABLES
003504 004567 000300 JSR X5,BMOVE )TO MINUS 1.
003510 003520 RIND
003512 003521 RIND+1
003514 000013 L1
003516 000207 RTS X7 )EXIT
003520 000000 RINDI OPEN
003522 000000 PT0I OPEN
003524 000000 PT1I OPEN
003526 000000 PINDI OPEN
003530 000000 PT0PI OPEN
003532 000000 PT1PI OPEN

```



```

;SPECIAL BINARY COUNT PATTERN SUBROUTINE, EXITS WITH BIN CHAR IN R0
003534 016767 177762 177762 GTBINI MOV PT0,PT1 ;PREVIOUS BIN CHAR TO PT1
003542 005167 177756 COM PT1
003546 005167 177746 COM RIND
003552 001002 BNE ,+6
003554 005267 177744 INC PT1
003560 042767 177400 177736 BIC #177400,PT1 ;MASK TO 8 BITS
003566 016767 177732 177726 MOV PT1,PT0 ;SAVE BIN CHAR IN PT0
003574 016700 177724 MOV PT1,X0 ;BIN CHAR TO R0,
003600 000207 RTS X7 ;EXIT,
003602 016767 177722 177722 GTBINPI MOV PT0P,PT1P ;PREVIOUS BIN CHAR TO PT1P
003610 005167 177716 COM PT1P
003614 005167 177706 COM PIND
003620 001002 BNE ,+6
003622 005267 177704 INC PT1P
003626 042767 177400 177676 BIC #177400,PT1P ;MASK TO 8 BITS,
003634 016767 177672 177666 MOV PT1P,PT0P ;SAVE BIN CHAR IN PT0P,
003642 016701 177664 MOV PT1P,X1 ;BIN CHAR TO R1,
003646 000207 RTS X7 ;EXIT,

;OCTAL TO ASCII CONVERT ROUTINES
003650 012500 ACNV6I MOV (5)+,X0 ;CONVERT TO 6 ASCII, GET OCTAL ADDRESS
003652 012567 000012 MOV (5)+,ACNV6 ;GET ASCII ADDRESS
003656 004767 000052 JSR X7,ACNV ;CONVERT TO ASCII
003662 004567 000122 JSR X9,BMOVE ;MOVE 6 CHARS TO ASCII ADDRESS
003666 003724 A1ST
003670 000000 ACNVBI OPEN
003672 000006 6
003674 000205 RTS X5 ;EXIT
003676 012500 ACNV4I MOV (5)+,X0 ;CONVERT TO 4 ASCII, GET OCTAL ADDRESS
003700 012567 000012 MOV (5)+,ACNV4 ;GET ASCII ADDRESS
003704 004767 000024 JSR X7,ACNV ;CONVERT TO ASCII
003710 004567 000074 JSR X9,BMOVE ;MOVE 4 CHARS TO ASCII ADDRESS,
003714 003726 A1ST+2
003716 000000 ACNVCI OPEN
003720 000004 4
003722 000205 RTS X5 ;EXIT
003724 000000 A1STI OPEN
003726 000000 OPEN
003730 000000 OPEN
003732 000000 ACNVXI OPEN
003734 012701 003732 ACNVI MOV #A1ST+6,X1 ;ADDR TO STORE ASCII TO R1
003740 012702 000006 MOV #6,X2 ;6 TO R2
003744 011067 177762 MOV #X0,ACNVX ;OCTAL WORD TO ACNVX
003750 016703 177756 ACNVMI MOV ACNVX,X3
003754 042703 177770 BIC #177770,X3 ;ISOLATE LEAST SIGNIFICANT OCTAL #
003760 062703 000060 ADD #60,X3 ;ADD 60 TO CONVERT TO ASCII
003764 110341 MOVB X3,-(1) ;STORE ASCII BYTE
003766 006067 177740 ROR ACNVX ;MOVE NEXT OCTAL DIGIT TO LEAST
003772 006067 177734 ROR ACNVX ;SIGNIFICANT POSITION
003776 006067 177730 ROR ACNVX
004002 005302 DEC X2 ;DONE 6 TIMES?
004004 001361 BNE ACNVM ;NO, REPEAT,
004006 000207 RTS X7 ;YES, EXIT,

```

```

004010 104020
004012 012501
004014 012502
004016 012503
004020 112122
004022 005303
004024 001375
004026 104021
004030 000205

004032 105777 175154
004036 100001
004040 000207
004042 104010
004044 000772

004046 004767 177760
004052 010077 175136
004056 105777 175130
004062 100375
004064 005000
004066 000207

004070 012700 011664
004074 013501
004076 012702 004176
004102 012767 000005 000060
004110 012267 000060
004114 004767 000010
004120 005367 000044
004124 001371
004126 000205
004130 005067 000036
004134 106701 000034
004140 103403
004142 005267 000024
004146 000772
004150 006701 000020
004154 002767 000060 000010
004162 116720 000004
004166 000207
004170 000000
004172 000000
004174 000000
004176 023420
004200 001750
004202 000144
004204 000012
004206 000001

;SUBROUTINE TO MOVE A VARIABLE NUMBER OF BYTES,
BMOVEI SAVREG ;SAVE REGS,
MOV (5)+,X1 ;GET FROM ADDRESS
MOV (5)+,X2 ;GET TO ADDRESS
MOV (5)+,X3 ;GET COUNT
BMOVAI MOVB (1)+,(2)+ ;MOVE BYTE
DEC X3 ;DECREMENT COUNT
BNE BMOVA ;BRANCH IF NOT DONE.
RSTREG ;RESTORE REGS,
RTS X5 ;DONE EXIT

;SUBROUTINE TO CHECK FOR PUNCH READY,
CPRDYI TSTB @TPB ;TEST FOR READY BIT,
BPL CPRDYA ;BRANCH IF READY NOT SET,
RTS X7 ;OK, EXIT,
CPRDYAI EMALT ;NOT READY, HALT,
BR CPRDY

;SUBROUTINE TO PUNCH ON LSP CHARACTER IN REG B,
LSPCHI JSR X7,CPRDY ;GO CHECK FOR PUNCH READY,
MOV X0,@TPB ;LOAD PUNCH BUFFER,
TSTB @TPB ;WAIT FOR DONE,
BPL , -4
CLR X0
RTS X7

;BINARY TO DECIMAL ASCII CONVERT SUBROUTINE,
BDCNVI MOV @DECVAL,X0 ;SET UP ADDR TO STORE DECIMAL ASCII IN R2
MOV @5,X1 ;BINARY VALUE TO R1,
MOV @ADTENP,X2 ;ADDR OF TEN POWER STRING TO R2,
MOV @5,CNVCTR ;SET UP FOR 5 POWER CONVERSIONS,
BDCNVAI MOV (2)+,TENPWR ;MOVE POWER OF TEN VALUE TO TENPWR,
JSR X7,SUBTEN ;PERFORM CONVERSION
DEC CNVCTR ;DONE 5 CONVERSIONS?
BNE BDCNVA ;BRANCH IF NOT YET 5,
RTS X5 ;YES, EXIT,
SUBTENI CLR DIGIT ;CLEAR DIGIT
SUBTNAI SUB TENPWR,X1 ;SUBTRACT TEN POWER FROM BINARY VALUE,
BCS SUBTND ;BRANCH IF UNSUCCESSFUL SUBTRACTION,
INC DIGIT
BR SUBTNA
SUBTNDI ADD TENPWR,X1 ;RESTORE SUBTRACTED VALUE,
ADD @0,DIGIT ;CONVERT (DIGIT) TO ASCII
MOVB DIGIT,(0)+ ;MOVE ASCII CHAR TO DECVAL FIELD,
RTS X7 ;EXIT,

CNVCTRI OPEN
DIGITI OPEN
TENPWR; OPEN
ADTENP; 10000,
1000,
100,
10,
1

```

```

004210 012767 000112 000024  ISUBROUTINE TO TYPE A LINE OF CHARACTERS
004216 012704 011671          TYPLNI  MOV      #74,,TCTR      I72 TO CHAR COUNT +CR,LF
004222 104002          TYPLAI  MOV      #BLOCKA,X4    ISET LINE ADDRESS IN R4.
004224 112400          TYPLBI  STALL                    ISTALL IF ALLOWED.
004226 004767 177614          MOVB    (4)+,X0      IGET CHARACTER
004232 005367 000004          JSR     X7,LSPCH     IGO OUTPUT CHARACTER.
004236 001371          DEC     TCTR         IDONE?
004240 000207          BNE    TYPLB        IBRANCH IF NOT DONE.
004242 000000          RTS     X7         IDONE, EXIT

TCTRI  OPEN
ISUBROUTINE TO TYPE LINE OF 3 CHARACTERS
004244 011667 000016 000010  TYPL3I  MOV      #X0,TPL3A    IDEVELOP AND SET ADDRESS OF
004250 017767 000012 000010  MOVB    #TPL3A,TPL3A   IATA IN TPL3A,
004256 062716 000002          ADD     #2,#X0        ISET UP EXIT,
004262 004567 000034          JSR     X5,FBF3       IFILL BUFFER WITH 3 CHARACTERS
004266 000000          TPL3AI  OPEN
004270 042767 040000 174750  BIC     #BIT14,PRGID   IDISABLE STALLS,
004276 004767 177706          JSR     X7,TYPLN     IGO TYPE LINE OF CHARACTERS,
004302 000002          RTI
004304 112767 000019 005357  STBFI  MOVB    #15,BLOCKA  IEXIT,
004312 112767 000012 005352  MOVB    #12,BLOCKA+1    ISUB TO SET UP BUFFER AREA,
004320 000207          RTS     X7         IEXIT

ISUBROUTINE TO FILL CHARACTER BUFFER WITH 3 CHARACTERS,
004322 012567 000004  FBF3I  MOV      (3)+,FBF3A    IMOVE 3 CHARS TO BUFFER,
004326 004567 177456          JSR     X5,BMOVE
004332 000000          FBF3AI  OPEN
004334 011673          BLOCK1
004336 000003          3
004340 004567 177444  FBF3BI  JSR     X5,BMOVE     IFILL 72 CHARACTERS BUFFER
004344 011673          BLOCK1              WITH 3 CHARACTERS
004346 011676          BLOCK1+3
004350 000109          69,
004352 004567 177432  JSR     X5,BMOVE
004356 011673          BLOCK1
004360 012005          BLOCK2
004362 000110          72,
004364 000205          RTS     X5         IEXIT

```

			ISUBROUTINE TO FILL BUFFER WITH ALL CHARACTERS	
004366	004567	177416	FBALL1 JSR X5,BMOVE	IFILL 92 CHAR BUFFER WITH
004372	010714		A	ALL CHARACTERS,
004374	011673		BLOCK1	
004376	000077		63,	
004400	004567	177404	JSR X5,BMOVE	
004404	010714		A	
004406	011772		BLOCK1+63,	
004410	000011		9,	
004412	004567	177372	JSR X5,BMOVE	
004416	011673		BLOCK1	
004420	012009		BLOCK2	
004422	000110		72,	
004424	000207		RTS X7	IFILL 92 CHAR BUFFER WITH
				ALL CHARACTERS,
			ISUB TO FILL BUFFER WITH 33 WORST CASE PATTERN,	
004426	004567	177356	FW3361 JSR X5,BMOVE	16 CHARACTER PATTERN TO BUFFER
004432	010700		A33WP6	
004434	011673		BLOCK1	
004436	000006		6	
004440	004567	177344	JSR X5,BMOVE	IFILL BUFFER WITH PATTERN,
004444	011673		BLOCK1	
004446	011701		BLOCK1+6	
004450	000102		60,	
004452	004567	177332	JSR X5,BMOVE	
004456	011673		BLOCK1	
004460	012009		BLOCK2	
004462	000110		72,	
004464	000207		RTS X7	IFILL 92 CHAR BUFFER WITH
				ALL CHARACTERS,
			ISUB TO FILL BUFFER WITH 35 WORST CASE PATTERN,	
004466	004567	177316	FW3561 JSR X5,BMOVE	16 CHARACTER PATTERN TO BUFFER
004472	010700		A35WP6	
004474	011673		BLOCK1	
004476	000006		6	
004500	004567	177304	JSR X5,BMOVE	IFILL BUFFER WITH PATTERN,
004504	011673		BLOCK1	
004506	011701		BLOCK1+6	
004510	000102		60,	
004512	004567	177272	JSR X5,BMOVE	
004516	011673		BLOCK1	
004520	012009		BLOCK2	
004522	000110		72,	
004524	000207		RTS X7	IFILL 92 CHAR BUFFER WITH
				ALL CHARACTERS,

```

IPRGR - INPUT-OUTPUT LOGIC TESTS
004326 012767 004540 174470 PRG01  MOV  @AT0,KSTART  IADDRESS OF 1ST ROUTINE TO KSTART,
004334 000167 175032          JMP  SRSET      IGO GET STARTED,
                                ITEST ABILITY TO REFERENCE THE KEYBOARD/READER STATUS WORD (TKS)
004340 000000          AT01  0          ITEST #,
004342 004572          AT1  1000,      INEXT TEST,
004344 001750          AT0A          I! COUNT,
004346 004556          MOV  @AT0E,MACHER ISCOPE ENTRY,
004350 012767 004566 173226 AT0A1  TST  @TKS     ISET UP MACHINE ERROR TRAP,
004356 005777 174424 AT0B1  CHAIN          IREFERENCE CODER STATUS WORD,
004362 104012          BR   AT0A        ICHAIN
004364 000774          AT0E1 ERROR        IREPEAT TEST,
004366 104003          BR   AT0B        IERROR, TRAPPED WHEN REFERENCING READER,
004370 000774          ITEST ABILITY TO REFERENCE THE KEYBOARD/READER BUFFER (TKB),
                                ITEST #,
004372 000001          AT11  1          INEXT TEST,
004374 004624          AT2  1000,      I! COUNT,
004376 001750          AT1A          ISCOPE ENTRY,
004000 004610          MOV  @AT1E,MACHER ISET UP MACHINE ERROR TRAP
004002 012767 004620 173174 AT1A1  TST  @TKB     IREFERENCE READER BUFFER,
004010 005777 174374 AT1B1  CHAIN          ICHAIN
004014 104012          BR   AT1A        IREPEAT TEST,
004016 000774          AT1E1 ERROR        IERROR, TRAPPED WHEN REFERENCING
004020 104003          BR   AT1B        IREADER BUFFER, (TKB),
004022 000774          ITEST ABILITY TO REFERENCE PUNCH/PRINTER STATUS WORD (TPS),
                                ITEST #,
004024 000002          AT21  2          INEXT TEST
004026 004650          AT3  1000,      I! COUNT,
004030 001750          AT2A          ISCOPE ENTRY,
004032 004642          MOV  @AT2E,MACHER ISETUP MACHINE ERROR TRAP,
004034 012767 004652 173142 AT2A1  TST  @TPS     IREFERENCE PUNCH/PRINTER STATUS WORD,
004042 005777 174344 AT2B1  CHAIN          ICHAIN
004046 104012          BR   AT2A        IREPEAT TEST,
004050 000774          AT2E1 ERROR        IERROR, TRAPPED WHEN REFERENCING
004052 104003          BR   AT2B        IPUNCH/PRINTER STATUS WORD (TPS),
004054 000774          ITEST ABILITY TO REFERENCE PUNCH/PRINTER BUFFER (TPB),
                                ITEST #,
004056 000003          AT31  3          INEXT TEST
004060 004710          AT4  1000,      I! COUNT,
004062 001750          AT3A          ISCOPE ENTRY,
004064 004674          MOV  @AT3E,MACHER ISETUP MACHINE ERROR TRAP,
004066 012767 004704 173110 AT3A1  TST  @TPB     IREFERENCE PUNCH/PRINTER BUFFER,
004074 005777 174314 AT3B1  CHAIN          ICHAIN
004700 104012          BR   AT3A        IREPEAT TEST,
004702 000774          AT3E1 ERROR        IERROR, TRAPPED WHEN REFERENCING
004704 104003          BR   AT3B        IPUNCH/PRINTER BUFFER, (TPB),
004706 000774

```

```

                                ITEST ABILITY TO SET AND CLEAR READER/KYBD ID BIT
004710 000004 AT4I 4 ITEST #
004712 004774 AT5 I NEXT TEST
004714 001750 1000, I I COUNT
004716 004726 AT4A I SCOPE ENTRY
004720 012767 000340 173050 MOV #PRV7,PSW I SET PRIORITY 7,
004726 052777 000100 174252 AT4AI BIS #BIT6,0TKS I SET ID BIT IN TKS,
004734 032777 000100 174244 BIT #BIT6,0TKS I CHECK ID BIT IN TKS
004742 001002 BNE AT4B I BRANCH IF ID BIT IS SET,
004744 104003 AT4E1I ERROR I ERROR 1 ID BIT NOT SET,
004746 000410 BR AT4C
004750 042777 000100 174230 AT4BI BIC #BIT6,0TKS I CLEAR ID BIT IN TKS
004756 032777 000100 174222 BIT #BIT6,0TKS I CHECK ID BIT IN TKS,
004764 001401 BEQ AT4C I BRANCH IF ID BIT IS CLEARED,
004766 104003 AT4E2I ERROR I ERROR, ID BIT FAILED TO CLEAR,
004770 104012 AT4CI CHAIN I CHAIN
004772 000755 BR AT4A I REPEAT TEST,

                                ITEST ABILITY TO CLEAR ID BIT WITH RESET INSTRUCTION.
004774 000005 AT5I 5 ITEST #
004776 005040 AT24 I NEXT TEST
005000 000144 100, I I COUNT
005002 005012 AT5A I SCOPE ENTRY,
005004 012767 000340 172764 MOV #PRV7,PSW I SET PRIORITY 7,
005012 052777 000100 174166 AT5AI BIS #BIT6,0TKS I SET ID BIT IN TKS
005020 104011 SRESET I RESET
005022 032777 000100 174156 BIT #BIT6,0TKS I TEST ID BIT,
005030 001401 BEQ AT5B I BRANCH IF ID BIT IS CLEAR,
005032 104003 AT5E1 ERROR I ERROR, RESET FAILED TO CLEAR ID BIT,
005034 104012 AT5BI CHAIN I CHAIN
005036 000765 BR AT5A I REPEAT TEST,

```

```

      JTEST ABILITY TO SET AND CLEAR PUNCH ID BIT
005040 000024 AT24I 24 ITEST#
005042 005124 AT25 INEXT TEST,
005044 001750 1000, I! COUNT
005046 005056 AT24A ISCOPE ENRY,
005050 012767 000340 172720 MOV #PRTY7,PSW ISET PRIORITY 7,
005056 052777 000100 174126 AT24AI BIS #BIT6,OTPS ISET PUNCH ID BIT,
005064 032777 000100 174120 BIT #BIT6,OTPS ICHECK PUNCH ID BIT,
005072 001002 ONE AT24B IBRANCH IF PUNCH ID BIT IS SET,
005074 104003 AT24E1I ERROR IERROR1, PUNCH ID BIT DID NOT SET,
005076 000410 BR AT24C
005100 042777 000100 174104 AT24BI BIC #BIT6,OTPS ICLEAR PUNCH ID BIT,
005106 032777 000100 174076 BIT #BIT6,OTPS ICHECK PUNCH ID BIT,
005114 001401 BEQ AT24C IBRANCH IF PUNCH ID BIT IS CLEAR
005116 104003 AT24E2I ERROR IERROR2, PUNCH ID BIT FAILED TO CLEAR,
005120 104012 AT24CI CHAIN ICHAIN
005122 000776 BR AT24C IREPEAT TEST

      JTEST ABILITY TO CLEAR PUNCH ID BIT WITH RESET INSTRUCTION
005124 000025 AT25I 25 ITEST#
005126 005170 AT26 INEXT TEST,
005130 000144 100, I! COUNT,
005132 005142 AT25A ISCOPE ENRY,
005134 012767 000340 172634 MOV #PRTY7,PSW ISET PRIORITY 7,

005142 052777 000100 174042 AT25AI BIS #BIT6,OTPS ISET PUNCH ID BIT,
005150 104011 SRESET IRESET
005152 032777 000100 174032 BIT #BIT6,OTPS ICHECK PUNCH ID BIT,
005160 001401 BEQ AT25B IBRANCH IF PUNCH ID BIT IS CLEAR,
005162 104003 AT25E1 ERROR IERROR, RESET FAILED TO CLEAR PUNCH ID BIT,
005164 104012 AT25BI CHAIN ICHAIN
005166 000765 BR AT25A IREPEAT TEST,

      JTEST ABILITY TO SET AND CLEAR THE PUNCH MAINTENANCE BIT
005170 000026 AT26I 26 ITEST#
005172 005246 AT27 INEXT TEST
005174 001750 1000, I! COUNT
005176 005200 AT26A ISCOPE ENRY
005200 052777 000204 174004 AT26AI BIS #BIT2,OTPS ISET MAINTANCE BIT,
005206 032777 000004 173776 BIT #BIT2,OTPS ICHECK MAINTENANCE BIT
005214 001002 ONE AT26B IBRANCH IF MAINTENANCE BIT SET,
005216 104003 AT26E1I ERROR IERROR1, MAINTENANCE BIT FAILED TO SET,
005220 000410 BR AT26C
005222 042777 000004 173762 AT26BI BIC #BIT2,OTPS ICLEAR MAINTENANCE BIT,
005230 032777 000004 173754 BIT #BIT2,OTPS ICHECK MAINTENANCE BIT
005236 001401 BEQ AT26C IBRANCH IF MAINTENANCE BIT IS CLEAR,
005240 104003 AT26E2I ERROR IERROR2, MAINTENANCE BIT FAILED TO CLEAR,
005242 104012 AT26CI CHAIN ICHAIN
005244 000755 BR AT26A IREPEAT TEST

```

```

005246 000027          JTEST THAT RESET INSTRUCTION CLEARS THE MAINTENANCE BIT;
005250 005304          AT27I 27          ITEST#
005252 000144          AT30          INEXT TEST
005254 005256          100,          I! COUNT
005256 052777 000004 173726 AT27A1 B18 0012,0TPS ISCOPE ENTRY
005264 104011          SRESET          ISET MAINTENANCE BIT,
005266 032777 000004 173716 BIT 0012,0TPS ISSUE RESET
005274 001401          BEQ AT27B ICHECK MAINTENANCE BIT
005276 104003          AT27E1 ERROR IBRANCH IF MAINTENANCE BIT CLEAR,
005300 104012          AT27B1 CHAIN IERROR; RESET FAILED TO CLEAR
005302 000765          BR AT27A ITHE MAINTENANCE BIT; CHAIN,
          IREPEAT TEST,
          JTEST THAT RESET SETS THE PUNCH READY BIT, AND THAT READY CAN BE READ RELIABLY,
005304 000030          AT30I 30          ITEST#
005306 005330          AT31          INEXT TEST
005310 001750          1000,          I! COUNT
005312 005314          AT30A1 TSTB 0TPS ISCOPE ENTRY
005314 105777 173672 BM1 AT30B ICHECK PUNCH READY,
005320 100401          AT30E1 ERROR IBRANCH IF PUNCH READY IS SET,
005322 104003          AT30B1 CHAIN IERROR; RESET FAILED TO SET READY, OR FAILED TO READ IT,
005324 104012          BR AT30A ICHAIN
005326 000772          IREPEAT TEST,
          JTEST THAT PUNCH READY RESETS BY LOADING PUNCH BUFFER,
005330 000031          AT31I 31          ITEST#
005332 005366          AT32          INEXT TEST
005334 000024          20,          I! COUNT
005336 005340          AT31A1 DELAY ISCOPE ENTRY
005340 104400          150,          IWAIT 150 MSEC
005342 000226          SRESET          IRESET
005344 104011          CLR 0TPB ILOAD PUNCH BUFFER
005346 005077 173642 TSTB 0TPS ICHECK PUNCH READY BIT,
005352 105777 173634 BPL AT31B IBRANCH IF PUNCH READY IS CLEAR,
005356 100001          AT31E1 ERROR IERROR; BUFFER LOAD FAILED TO CLEAR READY,
          AT31B1 CHAIN ICHAIN
          BR AT31A IREPEAT TEST,
          JTEST THAT BYTE LOAD OF PUNCH BUFFER +1 DOES NOT RESET READY,
005366 000032          AT32I 32          ITEST#
005370 005430          AT33          INEXT TEST
005372 000024          20,          I! COUNT
005374 005376          AT32A1 DELAY ISCOPE ENTRY
005376 104400          150,          IWAIT 150 MSEC
005400 000226          SRESET          IRESET
005402 104011          MOV TPB,X0
005404 016700 173604 INC X0
005410 005200          CLRB 0X0 IBYTE LOAD PUNCH BUFFER+1
005412 105010          TSTB 0TPS ICHECK PUNCH READY BIT
005414 105777 173572 BM1 AT32B IBRANCH IF PUNCH READY STILL SET,
005420 100401          AT32E1 ERROR IERROR; BYTE LOAD OF PUNCH BUFFER+1
005422 104003          AT32B1 CHAIN ICLEARED READY, CHAIN
005424 104012          BR AT32A IREPEAT TEST,
005426 000763

```



```

005430 000033
005432 005470
005434 000024
005436 005440
005440 104400
005442 000226
005444 005077 173544
005450 104400
005452 000310
005454 105777 173532
005460 100401
005462 104003
005464 104012
005466 000764

ITEST THAT PUNCH BECOMES READY BY 200 MSECS AFTER BUFFER LOAD,
AT33I 33
AT33A AT34
20,
AT33AI AT33A
DELAY
150,
CLR 0TPB
DELAY
200,
TSTB 0TPS
0MI AT33B
AT33EI ERROR
AT33BI CHAIN
BR AT33A

ITEST #
INEXT TEST
II COUNT
ISCOPE ENTRY,
IWAIT 150 MSECS,
ILOAD PUNCH BUFFER,
IWAIT 200 MSECS,
ICHECK PUNCH READY BIT,
IBRANCH IF PUNCH READY IS SET,
IERROR, READY NOT SET 200 MSECS AFTER BUFFER LOAD,
ICHAIN
IREPEAT TEST,

ITEST THAT PUNCH READY BIT CAN CAUSE AN INTERRUPT, IF THE INTERRUPT
IS SERVICED, IT WILL HAVE OCCURRED AT THE CORRECT VECTOR,
AT34I 34
AT35
1000,
AT34AI AT34A
STPCHV
AT34C
CLR 0TPS
CLR PSW
0IS 0BIT6,0TPS
NOP
AT34EI ERROR
AT34BI CHAIN
BR AT34A
AT34CI POPSP2
BR AT34B

ITEST THAT PUNCH READY DOES NOT CAUSE AN INTERRUPT WITH PROCESSOR
AT SAME PRIORITY LEVEL AS THE PUNCH INTERRUPT REQUEST LEVEL,
AT35I 35
AT36
1000,
AT35AI AT35A
STPCHV
AT35E
MOV TPLVL,PSW
CLR 0TPS
0IS 0BIT6,0TPS
NOP
AT35BI CLR 0TPS
CHAIN
BR AT35A
AT35EI POPSP2
ERROR
BR AT35B

ITEST #
INEXT TEST
II COUNT
ISCOPE ENTRY
ISET PUNCH INTERRUPT SERVICE
ITO AT34C
IDISABLE PUNCH INTERRUPTS
ISET PRIORITY 0,
IENABLE PUNCH INTERRUPTS,
IPUNCH READY FAILED TO CAUSE
INTERRUPT, CHAIN
IREPEAT TEST,
IHERE IF INTERRUPT OCCURS, POP THE
ISTOCK TWICE,
ISET PROCESSOR TO SAME PRIORITY AS PUNCH,
IDISABLE PUNCH INTERRUPTS,
IENABLE PUNCH INTERRUPTS,
IOK IF NO INTERRUPT OCCURS,
ICHAIN
IREPEAT TEST,
IERROR, PUNCH INTERRUPTED WITH PROCESSOR
ISET TO SAVE PRIORITY AS THE PUNCH,

```

005612	000036			ATEST THAT THE PUNCH INTERRUPTS WITH PROCESSOR AT PRIORITY ONE LEVEL LOWER THAN THE PUNCH PRIORITY,				
005614	005674			AT36I	36			ATEST #
005616	001750				AT37			INEXT TEST
005620	005626				1000,			II COUNT
005622	104027				AT36A			ISCOPE ENTRY
005624	005662				STPCHV			ISET PUNCH INTERRUPT SERVICE
005626	005077	173360			AT36B			ITO AT36B;
005632	016767	173360	172136	AT36AI	CLR	OTPS		IDISABLE PUNCH INTERRUPTS
005640	162767	000040	172130		MOV	TPLVL,PSW		ISET PROCESSOR PRIORITY ONE LEVEL
005646	052777	000100	173336		SUB	#40,PSW		ILOWER THAN PUNCH PRIORITY
005654	000240				BIS	#BIT6,OTPS		IENABLE PUNCH INTERRUPTS
005656	104003				NOP			
005660	000401				ERROR			IERROR; PUNCH FAILED TO INTERRUPT,
005662	222626				BR	AT36C		
005664	005077	173322		AT36BI	POPSP2			HERE IF INTERRUPT OCCURS, PCP
005670	104012			AT36CI	CLR	OTPS		THE STOCK TWICE, DISABLE PUNCH INTERRUPT
005672	000755				CHAIN			ICHAIN
					BR	AT36A		IREPEAT TEST,
								REINTERRUPT AFTER RTI WHEN READY
005674	200037			ATEST THAT PUNCH READY DOES NOT				
005676	205772			IBIT HAS NOT BEEN RESET,				
005700	201750			AT37I	37			ATEST #
005702	005724				AT40			INEXT TEST
005704	104027				1000,			II COUNT
005706	005742				AT37A			ISCOPE ENTRY
005710	005077	173276		AT37AI	STPCHV			ISET PUNCH INTERRUPT SERVICE TO
005714	005067	172056			AT37C			IAT37C
005720	052777	000100	173264		CLR	OTPS		IDISABLE PUNCH INTERRUPTS
005726	000240				CLR	PSW		ISET PROCESSOR PRIORITY TO #
005730	104023				BIS	#BIT6,OTPS		IENABLE PUNCH INTERRUPTS
005732	005077	173254			NOP			
005736	104012			AT37E1I	ERROR			IERROR 1, PUNCH FAILED TO INTERRUPT,
005740	000761			AT37B1	CLR	OTPS		IDISABLE PUNCH INTERRUPT,
005742	012777	005762	173252		CHAIN			ICHAIN
005750	012716	205756			BR	AT37A		IREPEAT TEST,
005754	000022			AT37C1	MOV	#AT37E2,OTPVTR		HERE IF INTERRUPT OCCURS, CHANGE
005756	000240				MOV	#AT37D,0X6		PUNCH VECTOR TO AT37E2 AND EXIT
005760	000764				RTI			I INTERRUPT
005762	022626			AT37D1	NOP			IOK IF NO REINTERRUPT OCCURS
005764	104023				BR	AT37B		
005766	000761			AT37E2I	POPSP2			IERROR 2, PUNCH REINTERRUPTED AFTER
					ERROR			RTI WITH READY BIT LEFT ON
					BR	AT37B		

TEST THAT THE PUNCH INTERRUPTS IMMEDIATELY UPON LOWERING
PROCESSOR PRIORITY TO B.

```

005770 000040
005772 000054
005774 001750
005776 000004
006000 104007
006002 000042
006004 012767 000340 171764 AT40A1
006012 005077 173174
006016 052777 000100 173166
006024 005067 171746
006030 012767 000340 171740
006036 104003
006040 000401
006042 022626
006044 005077 173142
006050 104012
006052 000754
    
```

```

TEST #
NEXT TEST
COUNT
SCOPE ENTRY
SET PUNCH INTERRUPT
SERVICE TO AT40B
SET PROCESSOR PRIORITY TO 7.
DISABLE PUNCH INTERRUPTS
ENABLE PUNCH INTERRUPTS
LOWER PROCESSOR PRIORITY TO B.
RAISE PRIORITY TO 7.
ERROR, PUNCH FAILED TO INTERRUPT
IMMEDIATELY AFTER CP PRIORITY WAS SET TO B.
HERE IF INTERRUPT OCCURS
DISABLE PUNCH INTERRUPTS
CHAIN
REPEAT TEST
    
```

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.

```

006054 000041
006056 000142
006060 000062
006062 000074
006064 104400
006066 000226
006070 104007
006072 000132
006074 012767 000001 000010 AT41A1
006102 005077 173100
006106 052777 000100 173070
006114 005067 171650
006120 000000

006122 005077 173064
006126 104012
006130 000761
006132 012767 104003 177760 AT41C1
006140 000002
    
```

```

TEST#
NEXT TEST
COUNT
SCOPE ENTRY
WAIT 150 MSECS
SET PUNCH INTERRUPT SERVICE
TO AT41C
MOVE WAIT INSTRUCTION TO AT41B
LOAD PUNCH BUFFER (ENABLES PUNCH)
ENABLE PUNCH INTERRUPTS
SET PRIORITY B.
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.
DISABLE PUNCH INTERRUPTS
CHAIN
REPEAT TEST
MOVE ERROR CALL TO AT41B.
EXIT INTERRUPT.
    
```

12

ITEST THAT LOADING THE PUNCH BUFFER WITH THE MAINTENANCE BIT SET
 ICAUSES THE READER DONE BIT TO SET AFTER APPROX, 200 MSECS

000142	000042			AT42I	42			ITEST 0
000144	177777				-1			ILAST TEST
000146	000062				90,			II CCUNT
000150	006156				AT42A			ISCOPE ENTRY
000152	104400				DELAY			
000154	000226				150,			
000156	052777	000204	173026	AT42A1	BIS	@BIT2,@TPS		ISSET MAINTENANCE BIT
000164	005077	173024			CLR	@TPB		ILOAD PUNCH BUFFER
000170	104400				DELAY			IWAIT 200 MSECS
000172	000310				200,			
000174	105777	173006			TSTB	@TKS		ITEST READER DONE BIT
000200	100401				BMI	AT42B		IIF BRANCH IF READER DONE BIT SET;
000202	104003			AT42E1	ERROR			IIF ERROR; 200 MSECS AFTER PUNCH
								IIF BUFFER LOAD WITH MAINTENANCE BIT
								ISSET THE READER DONE BIT WAS NOT SET
000204	104012			AT42B1	CHAIN			ICHAIN
000206	000763				BR	AT42A		IIF REPEAT TEST

```

I PRG2-PRINTER TESTS
006210 012767 006242 173014 PRG11 MOV #CTB,KSTART ISET ADDRESS IF 1ST ROUTINE,
006216 052767 000200 173022 B15 #017,PRGID IBYPASS SCOPE AND ICNT,
006224 012767 177600 174770 MOV #177600,STLMSK ISET STALL LIMIT
006232 004767 176040 JSR X7,STBF ISET UP BUFFER AREA,
006236 000167 173330 JMP SRSET IGO GET STARTED,

ICARRIAGE RETURN TEST,
CT01 0 ITEST#
CT1 INEXT TEST ADDRESS,
TYPE ITYPE TITLE,
CRTST
MOV #72,,RCNT
MOV RCNT,CTRA IRCNT TO CTRA
CT0A1 DEC CTRA IDECREMENT CTRA
BNE CT0B IBRANCH IF NOT 0
CHAIN IB; CHAIN
CT0B1 MOV CTRA,CTRB ISPACE COUNT TO CTRB,
CT0C1 MOVB #105,X0 ICHAR#E
JSR X7,LSPCH ISPACE;
DEC CTRB IDECREMENT CTRB,
BNE CT0C IBRANCH IF NOT DONE SPACING.
MOVB #15,X0
JSR X7,LSPCH ICARRIAGE RETURN,
MOV #12,X0 ILINE FEED
JSR X7,LSPCH
BR CT0A I

IRIGHT MARGIN TEST
CT11 1 ITEST#
CT2 INEXT TEST,
TYPE ITYPE TITLE
RMTST
MOV #14,,CTRA ISET UP FOR 33/35
MOV #RM330,RMB
CT1A1 TYPE ITYPE=---;
RMT3A DEC CTRA IDONE N TIMES,
BNE CT1A IBRANCH IF NOT N TIMES
TYPE ITYPE=-;
RMB1 OPEN ICHAIN
CHAIN

```

PALX11

V003

26-JUL-71

16102 PAGE 26

```
ICURSOR RIGHT TEST
CT2I  2
      CT3
      TYPE
      SPTST
      MOV      #36,,CTRA
CT2AI TYPE
      SPTSTC
      DEC      CTRA
      BNE      CT2A
      MOV      #36,,CTRA
      MOV      #1,CTRB
      MOV      CTRB,CTRC
      MOV      #15,XB
      JSR      X7,LSPCH
      JSR      X7,LSPCH
CT2DI CLR      XB
      JSR      X7,LSPCH
      JSR      X7,LSPCH
      JSR      X7,LSPCH
      MOV      #38,XB
      JSR      X7,LSPCH
      DEC      CTCR
      BNE      CT2D
      MOV      #'/,XB
      JSR      X7,LSPCH
      DEC      CTRA
      BNE      CT2E
      CHAIN
CT2EI ADD      #2,CTRB
      BR       CT2C
ILINE FEED TEST
CT3I  3
      CT4
      TYPE
      LFTST
      BIS      #BIT14,PRGID
      MOV      #72,,CTRA
CT3AI MOV      #'\,XB
      JSR      X7,LSPCH
      MOV      #12,XB
      JSR      X7,LSPCH
      DEC      CTRA
      BNE      CT3B
      CHAIN
CT3BI STALL
      BR       CT3A

ITEST#
INEXT TEST
ITYPE TITLE,

I33/35 COUNT TO CTRA,
ITYPE SPACE,\,

IDONE TIMES SET IN CTRA?
IBRANCH IF NOT DONE
ISET UP CTRA COUNT FOR 33/35

ICARRIAGE RETURN,

IDUMMY CYCLE,
INULL CHAR FOR FILLER
ITRANSMIT NULL CHAR
ITRANSMIT NULL CHAR
ITRANSMIT NULL CHAR
ICURSOR RIGHT
ISET IN CTCR,
IDONE SPACING,
IBRANCH IF NOT DONE SPACING,
IDONE, TYPE A "/",

IDONE 36 TIMES?
IBRANCH IF NOT DONE,
IDONE, CHAIN,
IMODIFY CTRB FOR NEXT TRY,
IGO DO IT AGAIN,

ITEST #
INEXT TEST,
ITYPE TITLE

IALLOW STALLS,
ISET 33/35 LINE FEED COUNT,
ITYPE "\,

ILINE FEED,

IDONE N TIMES?
IBRANCH IF NOT DONE,
IDONE, CHAIN
ISTALL
IREPEAT
```

006642	000004					ISPACE TEST					
006644	007054					CT4I	4				ITEST0
006646	012767	000011	000074			CT5					INEXT TEST,
006654	104014					MOV	#9,,TBCNT				ISET TAB COUNT,
006656	104012					CK35					I357
006660	004567	000040				CHAIN					INC,
006664	000007					JSR	X5,TPBM				ITYPE MARKERS
006666	104000					7					
006670	011035					TYPE					
006672	012767	002007	172456		CT4AI	MOV	#7,CTRA				ILINE COUNT TO CTRA
006700	005067	000046				CLR	SPCNT				IB TO SPACE COUNT,
006704	004767	000044			CT4BI	JSR	X7,TABP				IGO SPACE-TAB,
006710	005267	000036				INC	SPCNT				IINCREMENT SPACE COUNT,
006714	005367	172436				DEC	CTRA				IDONE ? LINES?
006720	001371					BNE	CT4B				IBRANCH IF NOT DONE,
006722	104012					CHAIN					IDONE, CHAIN,
006724	012567	172426			TPBMI	MOV	(5)+,CTRA				ITYPE TEST TITLE,
006730	104000					TYPE					
006732	011016					TBTST					
006734	104000				TPBMAI	TYPE					ITYPE MARKERS
006736	011046					TBMRK1					
006740	005367	172412				DEC	CTRA				
006744	001373					BNE	TPBMA				
006746	000205					RTS	X5				IFEXIT,
006750	000000				TBCNTI	OPEN					ITAB COUNT
006752	000000				SPCNTI	OPEN					ISPACE COUNT
006754	104000				TABPI	TYPE					ICRLF,
006756	011057					CRLF					
006760	010767	177764	172372			MOV	TBCNT,CTRB				ITAB COUNT TO CTRB
006766	010767	177760	172366		TABPAI	MOV	SPCNT,CTRC				ISPACE COUNT TO CTRC
006774	001407					BEO	TABPC				IBRANCH IF SPACE COUNT IS 0,
006776	112700	000040			TABPBI	MOVB	#40,XB				ISPACE
007002	004767	175040				JSR	X7,LSPCH				
007006	005367	172350				DEC	CTRC				IDECREMENT SPACE COUNT
007012	001371					BNE	TABPB				IBRANCH IF NOT YET 0,
007014	112700	000011			TABPCI	MOVB	#11,XB				ITAB
007020	004767	175022				JSR	X7,LSPCH				
007024	004767	175016				JSR	X7,LSPCH				IDUMMY CYCLE
007030	004767	175012				JSR	X7,LSPCH				IDUMMY CYCLE,
007034	112700	000057				MOVB	#1/,XB				ITYPE "/"
007040	004767	175002				JSR	X7,LSPCH				
007044	005367	172310				DEC	CTRB				IDECREMENT TAB COUNT,
007050	001346					BNE	TABPA				IBRANCH IF NOT DONE TABBING,
007052	000207					RTS	X7				IDONE, EXIT,

007054	000005	ITYPE LINE OF CHARACTERS ABC	ITEST #
007056	007072	CT91 5	INEXT TEST
007060	104000	CT6	ITYPE "CHARACTER TESTS"
007062	011236	TYPE	
007064	104016	CHRYS	ITYPE LINE
007066	010714	TYPLN3	
007070	104012	A	ICHAIN
		CHAIN	
007072	000006	ITYPE LINE OF CHARACTERS DEF	ITEST #
007074	007104	CT61 6	INEXT TEST
007076	104016	CT7	ITYPE LINE
007100	010717	TYPLN3	
007102	104012	D	ICHAIN
		CHAIN	
007104	000007	ITYPE LINE OF CHARACTERS GHI	ITEST#
007106	007116	CT71 7	INEXT TEST,
007110	104016	CT10	ITYPE LINE
007112	010722	TYPLN3	
007114	104012	G	ICHAIN
		CHAIN	
007116	000010	ITYPE LINE OF CHARACTERS OF JKL	ITEST #
007120	007130	CT101 10	INEXT TEST,
007122	104016	CT11	ITYPE LINE
007124	010725	TYPLN3	
007126	104012	J	ICHAIN
		CHAIN	
007130	000011	ITYPE LINE OF CHARACTERS MNO	ITEST #
007132	007142	CT111 11	INEXT TEST
007134	104016	CT12	ITYPE LINE
007136	010730	TYPLN3	
007140	104012	M	ICHAIN
		CHAIN	
007142	000012	ITYPE LINE OF CHARACTERS POR	ITEST #
007144	007154	CT121 12	INEXT TEST
007146	104016	CT13	ITYPE LINE
007150	010733	TYPLN3	
007152	104012	P	ICHAIN
		CHAIN	
007154	000013	ITYPE LINE OF CHARACTERS STU	ITEST #
007156	007160	CT131 13	INEXT TEST
007160	104016	CT14	
007162	010736	TYPLN3	
007164	104012	S	
		CHAIN	

007166	000014	ITYPE LINE OF CHARACTERS VHX	ITYEST #
007170	007200	CT141 14	INEXT TEST
007172	104016	CT15	ITYPE LINE
007174	010741	TYPLN3	
007176	104012	V	
		CHAIN	ICHAIN
007200	000015	ITYPE LINE OF CHARACTERS YEB	ITYEST #
007202	007212	CT151 15	INEXT TEST
007204	104016	CT16	ITYPE LINE
007206	010744	TYPLN3	
007210	104012	V	
		CHAIN	ICHAIN
007212	000016	ITYPE LINE OF CHARACTERS 123	ITYEST #
007214	007224	CT161 16	INEXT TEST
007216	104016	CT17	ITYPE LINE
007220	010747	TYPLN3	
007222	104012	ONE	
		CHAIN	ICHAIN
007224	000017	ITYPE LINE OF CHARACTERS 456	ITYEST #
007226	007236	CT171 17	INEXT TEST
007230	104016	CT20	ITYPE LINE
007232	010752	TYPLN3	
007234	104012	FOUR	
		CHAIN	ICHAIN
007236	000020	ITYPE LINE OF CHARACTERS 789	ITYEST #
007240	007250	CT201 20	INEXT TEST
007242	104016	CT21	ITYPE LINE
007244	010755	TYPLN3	
007246	104012	SEVEN	
		CHAIN	ICHAIN
007250	000021	ITYPE LINE OF CHARACTERS !"#	ITYEST #
007252	007262	CT211 21	INEXT TEST
007254	104016	CT22	ITYPE LINE
007256	010760	TYPLN3	
007260	104012	C41	
		CHAIN	ICHAIN
007262	000022	ITYPE LINE OF CHARACTERS \$%&	ITYEST #
007264	007274	CT221 22	INEXT TEST
007266	104016	CT23	ITYPE LINE
007270	010763	TYPLN3	
007272	104012	C44	
		CHAIN	ICHAIN
007274	000023	ITYPE LINE OF CHARACTERS '()	ITYEST #
007276	007306	CT231 23	INEXT TEST
007300	104016	CT24	ITYPE LINE
007302	010766	TYPLN3	
007304	104012	C47	
		CHAIN	ICHAIN

007306	000024			ITYPE LINE OF CHARACTERS 00,					
007310	007320			CT241 24				ITEST #	
007312	104016			CT25				INEXT TEST	
007314	010771			TYPLN3				ITYPE LINE	
007316	104012			C52					
				CHAIN				ICHAIN	
007320	000025			ITYPE LINE OF CHARACTERS 01,					
007322	007332			CT251 25				ITEST #	
007324	104016			CT26				INEXT TEST	
007326	010774			TYPLN3				ITYPE LINE	
007330	104012			C55					
				CHAIN				ICHAIN	
007332	000026			ITYPE LINE OF CHARACTERS 11C					
007334	007344			CT261 26				ITEST #	
007336	104016			CT27				INEXT TEST	
007340	010777			TYPLN3				ITYPE LINE	
007342	104012			C72					
				CHAIN				ICHAIN	
007344	000027			ITYPE LINE OF CHARACTERS 0>?					
007346	007356			CT271 27				ITEST #	
007350	104016			CT30				INEXT TEST	
007352	011002			TYPLN3				ITYPE LINE	
007354	104012			C75					
				CHAIN				ICHAIN,	
007356	000030			ITYPE LINE OF CHARACTERS 0C\					
007360	007370			CT301 30				ITEST #	
007362	104016			CT31				INEXT TEST	
007364	011005			TYPLN3				ITYPE LINE	
007366	104012			C100					
				CHAIN				ICHAIN	
007370	000031			ITYPE LINE OF CHARACTERS 3 AND LEFT ARROW					
007372	007402			CT311 31				ITEST #	
007374	104016			CT32				INEXT TEST	
007376	011010			TYPLN3				ITYPE LINE	
007400	104012			C135					
				CHAIN				ICHAIN	
007402	000032			ITYPE 2 LINES OF ALL CHARACTERS, FIRST LINE FULL SPEED, SECOND LINE WITH STALLS.					
007404	007440			CT321 32				ITEST #	
007406	004767	174754		CT33				INEXT TEST,	
007412	042767	040000	171626	JSR X7,FBALL				IFILL BUFFER WITH ALL CHARACTERS,	
007420	004767	174564		BIC #BIT14,PRGID				ICLEAR STALL BIT IN PRGID	
007424	052767	040000	171614	JSR X7,TYPLN				ITYPE LINE,	
007432	004767	174552		BIS #BIT14,PRGID				ISET STALL BIT IN PRGID	
007436	104012			JSR X7,TYPLN				ITYPE LINE,	
				CHAIN				ICHAIN,	

				I TYPE 12 LINES OF ASR33 WORST CASE PATTERN, ALTERNATE LINES WITH STALLS.		
007440	000033			CT331	J3	I TEST #
007442	007522				CT34	I NEXT TEST
007444	104013				CK33	I 33?
007446	104012				CHAIN	INO, BYPASS TEST.
007450	104000				TYPE	I TYPE "WORST CASE PATTERN TEST"
007452	011262				WCPTST	
007454	004767	174746			JSR	I PATTERN TO BUFFER.
007460	012767	000000	171670		MOV	I SET COUNT TO 0
007466	042767	040000	171552	CT33A1	BIC	I CLEAR STALL BIT IN PRGID.
007474	004767	174510			JSR	I TYPE LINE
007500	052767	040000	171540		BIS	I SET STALL BIT IN PRGID.
007506	004767	174476			JSR	I TYPE LINE.
007512	005367	171640			DEC	IDONE 0 TIMES?
007516	001363				BNE	I BRANCH IF NOT 0 TIMES YET.
007520	104012				CHAIN	IDONE, CHAIN.
				I TYPE 12 LINES OF ASR33 WORST CASE PATTERN, ALTERNATE LINES WITH STALLS.		
007522	000034			CT341	J4	I TEST #
007524	177777				-1	I LAST TEST.
007526	104014				CK33	I 33?
007530	104012				CHAIN	INO, BYPASS TEST.
007532	104000				TYPE	I TYPE "WORST CASE PATTERN TEST"
007534	011262				WCPTST	
007536	004767	174724			JSR	I PATTERN TO BUFFER.
007542	012767	000000	171600		MOV	I SET COUNT TO 0.
007550	042767	040000	171470	CT34A1	BIC	I CLEAR STALL BIT IN PRGID.
007556	004767	174420			JSR	I TYPE LINE.
007562	052767	040000	171450		BIS	I SET STALL BIT IN PRGID.
007570	004767	174414			JSR	I TYPE LINE
007574	005367	171550			DEC	IDONE 0 TIMES?
007600	001363				BNE	I BRANCH IF NOT 0 TIMES YET.
007602	104012				CHAIN	IDONE, CHAIN.

```

)PRG2=KEYBOARD TEST
007604 012767 007630 171420 PRG21 MOV #ETB,KSTART
007612 052767 000200 171420 B1S #BIT7,PRCID
007620 104000 TYPE
007622 011316 KMSG1
007624 000167 171742 JMP SRSET
)TEST THAT PRESSING KEY SETS DONE FLAG.
007630 000000 ET01 0 )TEST 0
007632 007734 ET1 )NEXT TEST.
007634 012767 000005 171514 MOV #5,CTRA
007642 104000 ET0A1 STRDRV
007644 007700 ET00
007646 104000 TYPE )TYPE "PRESS A KEY WITHIN 10 SECS,"
007650 011334 KMSG2
007652 052777 000100 171326 B1S #BIT6,OTKS )ENABLE KYBD INTERRUPT.
007660 005067 170112 CLR PSH
007664 104400 DELAY )WAIT 10 SECONDS
007666 023420 10000,
007670 104000 TYPE )TYPE "NO KEYBOARD REQUEST,"
007672 011536 KMSG6
007674 104010 EHALT )HALT.
007676 000411 BR ET0CA
007700 105777 171302 ET001 TSTB OTKS )TEST FOR DONE BIT ON
007704 100403 BHI ET0C )BRANCH IF DONE BIT SET,
007706 104000 TYPE )DONE BIT NOT SET, TYPE IF FALSE KEY-
007710 011564 KMSG7 )BOARD OR READER INTERRUPT,
007712 104010 EHALT )HALT
007714 012716 007722 ET0C1 MOV #ET0CA,OX6
007720 000002 RTI )EXIT INTERRUPT.
007722 104011 ET0CA1 SRESET
007724 005367 171426 DEC CTRA )DONE 5 TIMES?
007730 001344 BNE ET0A )BRANCH IF NOT DONE.
007732 104012 CHAIN )CHAIN
)ECHO TEST, KEYED CHARACTER IS TYPED, RUBOUT ENDS ROUTINE.
007734 000001 ET11 1 )TEST 0
007736 010014 ET2 )NEXT TEST.
007740 104000 TYPE )TYPE TITLE AND INSTRUCTIONS.
007742 011374 KMSG3
007744 105777 171236 ET1A1 TSTB OTKS )WAIT FOR DONE FLAG
007750 100375 BPL ,=4
007752 117767 171232 171364 MOVB OTKB,CBUP )MOVE KYBD CHAR TO CBUP.
007760 116777 171360 171226 MOVB CBUP,OTPB )ECHO CHAR READ.
007766 105777 171220 TSTB OTPS )WAIT FOR PRINTER DONE.
007772 100375 BPL ,=4
007774 042767 000200 171342 BIC #BIT7,CBUP )CLEAR BIT 7 FROM CBUP.
010002 122767 000177 171334 CMPB #177,CBUP )COMPARE CBUP TO RUBOUT (177)
010010 001355 BNE ET1A )BRANCH IF NOT RUBOUT (177)
010012 104012 CHAIN )CHAIN

```

IOCTAL EQUIVALENT TEST, THE OCTAL EQUIVALENT OF ANY CHARACTER KEYED
 IS PRINTED, RUBOUT ENDS ROUTINE;

010014 000002
 010016 177777
 010020 104001
 010022 011476
 010024 011407
 010026 177777
 010030 009067 171310
 010034 109777 171146
 010040 100379
 010042 117767 171142 171274
 010050 004567 173622
 010054 001344
 010056 011530
 010060 104000
 010062 011526
 010064 042767 000200 171252
 010072 022767 000177 171244
 010100 001355
 010102 104012

ET2I

2
 -1
 TYPES
 KMSG4
 KMSG3A
 -1
 CLR CRBUF
 TSTB 0TKS
 BPL ,=4
 MOVB 0TKB,CRBUF
 JBR XS,ACNV4
 CRBUF
 OCTEQV
 TYPE
 KMSG5
 BIC #BIT7,CRBUF
 CMP #177,CRBUF
 BNE ET2A
 CHAIN

ET2A1

I TEST #
 I LAST TEST
 I TYPE TITLE AND INSTRUCTIONS.

I WAIT FOR DONE FLAG.

I CHARACTER TO CRBUF
 I CONVERT CHAR IN CRBUF TO
 I PRINTABLE OCTAL

I TYPE OCTAL EQUIVALENT

I CLEAR BIT 7 FROM CRBUF
 I TEST FOR RUBOUT CHARACTER,
 I BRANCH IF NOT RUBOUT (177).
 I CHAIN.

```

;PRG3=PRINTER EXERCISER, KEYBOARD CONTROLLED,
;TYPES LINES WITH ANY 5 CHARACTERS, STALLS OR FULL SPEED.
010104 004767 174174 PRG3I JSR X7,STBF ISET UP BUFFER,
010110 104000 TYPE ITYPE TITLE
010112 011612 P7MG1
010114 052767 040000 171124 HTAI BIS #BIT14,PRGID ISET STALL BIT IN PRGID.
010122 012767 177600 173072 MOV #177600,STLMSK ISET STALL MASK
010130 012703 011673 MOV #BLOCK1,X3
010134 104000 TYPE ITYPE "TYPE IN DATA".
010136 011640 P7MG2
010140 012767 000000 171210 MOV #6,CTRA ICHAR COUNT TO CTRA.
010146 004767 000060 HTBI JSR X7,GKBCR IGET AND STORE KYBD CHARACTER,
010152 005367 171200 DEC CTRA IGET 6 CHARACTERS?
010156 001373 BNE HTB IBRANCH IF NOT 6 CHARS YET.
010160 042767 000200 171156 BIC #BIT7,CRBUF
010166 122767 000177 171150 CMPB #177,CRBUF ICHECK 6TH CHAR FOR RUBOUT.
010174 001003 BNE HTC IBRANCH IF NOT A RUBOUT.
010176 042767 040000 171042 BIC #BIT14,PRGID IRUBOUT. CLEAR STALL BIT IN PRGID.
010204 004567 173000 HTCI JSR X5,MOVE IFILL 92 CHAR LINE.
010210 011673 BLOCK1
010212 011700 BLOCK1+5
010214 000103 67,
010216 004767 173766 HTDI JSR X7,TYPLN ITYPE LINE,
010222 005767 167342 TPT SR ICHANGE DATA? (SR15=1).
010226 100732 B I HTA IYES. GO CHANGE DATA
010230 000772 B HTD INO CONTINUE WITH SAME DATA.
010232 105777 170750 GKBCRI TSTB #TKS IWAIT FOR DONE FLAG.
010236 100375 FPL I=4
010240 117767 170744 171070 IOVB #TKB,CRBUF ICHARACTER TO CRBUF.
010246 116723 171072 IOVB CRBUF,(3)+ ICHARACTER TO LINE BUFFER.
010252 116700 171066 MOVB CRBUF,X0
010256 004767 173564 JSR X7,LSPCH IECHO CHARACTER.
010262 000207 RTS X7

```

010264 104005
 010266 004767 000036
 010272 000775

```

|
|
|PRG4=PUNCH CLOCK ADJUSTMENT ROUTINE.
|OUTPUTS CHARACTER SET IN LEFT HALF OF SR, AND
|STALLS FOR NUMBER OF MILLISECONDS SET IN RIGHT HALF OF SR.
PRG4I  CHALT                    IHALT TO SET SR,
|TAI   JSR      X7,C1112        IGO OUTPUT CHARACTER SET IN LEFT
|BR     ITA                    IHALF OF SR AND STALL PER SR RIGHT.
    
```

010274 104005
 010276 004767 000020
 010302 017700 170702
 010306 000005
 010310 000025
 010312 000025
 010314 000025
 010316 000025
 010320 000766

```

|
|
|PRG5=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 RB.
PRG5I  CHALT                    IHALT TO SET SR,
|TAI   JSR      X7,C1112M       IGO OUTPUT CHARACTER FROM SR LEFT AND
|MOV   0TKB,X0                  ISTALL PER SR RIGHT, (TKB) TO RB,
|RESET RESET                    I"FIX" (TKB) IN DATA LIGHTS.
|RESET RESET
|RESET RESET
|RESET RESET
|BR     JTA                      IREPEAT.
    
```

010322 052777 000004 170662
 010330 116767 107234 000022
 010336 005767 000016
 010342 001002
 010344 005267 000010
 010350 116777 107215 170636
 010356 104400
 010360 000000
 010362 000207

```

|
|
|C1112M1 BIS      #4,0TPS        ISET MAINTENANCE MODE (PUNCH).
|C1112I MOVB     SR,XTY          ISTALL COUNT TO XTY.
|TST     XTY                    IDISREGARD 0 DELAY.
|BNE     C1112A
|INC     XTY
|C1112A1 MOVB     SR+1,0TPB      ILOAD PUNCH BUFFER.
|DELAY   DELAY                  IDELAY (APPROXIMATELY) THE NUMBER OF
|XTYI    OPEN                      INSECS; SPECIFIED AT SR RIGHT
|RTS     X7                       IEXIT
    
```

IPRG6=MAINTENANCE MODE SINGLE CHARACTER DATA TEST,
 WITH MAINTENANCE MODE SET, OUTPUTS ONTO PUNCH BUFFER AND BACK ONTO
 READER BUFFER THE CHARACTER SET IN SR LEFT, THE CHARACTER IN THE
 READER BUFFER IS COMPARED TO THE CHARACTER IN SR LEFT, IF THE 2 CHARACTERS
 DISAGREE THE PROGRAM HALTS, THE DATA LIGHTS WILL THEN CONTAIN

010364	104005		
010366	052777	000004	170610
010374	105777	170612	
010400	100375		
010402	116767	107163	170735
010410	116777	170731	170576
010416	105777	170564	
010422	100375		
010424	117767	170560	170712
010432	104024		
010434	000754		

LEFT HALF: THE EXPECTED CHARACTER (SR LEFT),
 RIGHT HALF: THE CHARACTER IN THE READER BUFFER,
 PRG6: CHALT ;HALT TO SET SR,
 KTA: BIS #4,OTPS ;SET MAINTENANCE MODE,
 KTB: TSTB #TPS ;WAIT FOR READY,
 BPL #4
 MOVB SR+1,CRBUF+1 ;S/B CHAR TO CRBUF+1,
 MOVB CRBUF+1,OTPB ;OUTPUT CHARACTER,
 TSTB #TKS ;WAIT FOR READER DONE FLAG,
 BPL #4
 MOVB #TKB,CRBUF ;CHAR READ TO CRBUF,
 DATCHK ;GO CHECK AGAINST S/B CHAR,
 BR KTA ;REPEAT,

IPRG7=MAINTENANCE MODE SPECIAL BINARY COUNT PATTERN DATA TEST,
 PERFORMS SAME OPERATION AS PRG13, EXCEPT THAT SPECIAL BINARY COUNT
 PATTERN IS USED,

010436	004767	173034	
010442	012767	177600	172552
010450	052767	040000	170570
010456	052777	000004	170526
010464	032767	000400	167076
010472	001001		
010474	104002		
010476	105777	170510	
010502	100375		
010504	004767	173072	
010510	110167	170631	
010514	110177	170474	
010520	105777	170462	
010524	100375		
010526	117767	170456	170610
010534	104004		
010536	000747		

PRG7: JSR #7,INBIN ;INITIALIZE BINARY COUNT
 MOV #177600,STLMSK ;SET STALL LIMIT
 BIS #BIT14,PRG10 ;ALLOW STALLS
 LTA: BIS #4,OTPS ;SET MAINTENANCE MODE,
 BIT #BIT0,SR ;CHECK STALL SWITCH
 BNE LTB ;BRANCH IF NO STALL WANTED
 STALL ;STALL
 LTB: TSTB #TPS ;WAIT FOR READY,
 BPL #4
 JSR #7,GTBINP ;GET BIN CHARACTER,
 MOVB #1,CRBUF+1 ;MOVE TO S/B CHAR,
 MOVB #1,OTPB ;OUTPUT BIN CHARACTER,
 TSTB #TKS ;WAIT FOR READER DONE,
 BPL #4
 MOVB #TKB,CRBUF ;CHAR IN READ BUFFER TO CRBUF,
 DATCHK ;GO CHECK AGAINST S/B CHAR,
 BR LTA ;CONTINUE,

IPRG10 ROLE UP TEST
 THE FUNCTION OF THIS TEST IS TO TEST THE ROLL-UP CAPABILITY
 OF THE VT36
 TO DO THIS A LINE OF A CHARACTER AND IT'S COMPLEMENT FOLLOWED
 BY A LINE OF THE COMPLEMENT AND THE CHARACTER IS TRANSMITTED
 THIS SCHEME IS CONTINUED UNTIL SWITCH 15 IS RAISED
 THE CHARACTER SHOULD NOT BE CHANGED UNTIL THE SCREEN HAS BEEN
 COMPLETELY FILLED

010540	012767	177736	000126
010546	016767	000122	000122
010554	012767	177670	000110
010562	005167	000110	

PRG10: MOV #42,TCHAR ;INIT TEMP CHAR
 RENIT: MOV TCHAR,CHAR ;COMPLEMENT OF "I"
 PRG10C: MOV #72,,CNT ;72 CHAR/LINE
 PRG10D: COM CHAR ;

010566	016700	000104		PRG10A1	MOV	CHAR,X0	ILOAD "!"
010572	004767	173250			JSR	X7,LSPCH	IPUNCH "!"
010576	005167	000074			COM	CHAR	ICOMPLEMENT TO "!"
010602	016700	000070			MOV	CHAR,X0	ILOAD "!"
010606	004767	173234			JSR	X7,LSPCH	IPUNCH "!"
010612	005167	000060			COM	CHAR	I "!"
010616	062767	000002	000046		ADD	#2,CNT	IEND OF LINE?
010624	001360				BNE	PRG10A	INO
010626	012700	000015			MOV	#15,X0	ICR
010632	004767	173210			JSR	X7,LSPCH	
010636	012700	000012			MOV	#12,X0	ILF
010642	004767	173200			JSR	X7,LSPCH	
010646	005767	166710			TST	SR	INEXT CHAR
010652	100340				BPL	PRG10C	INO
010654	005367	000014			DEC	TCHAR	IYES CHANGE TCHAR
010660	022767	177677	000006		CMP	#177677,TCHAR	ICCHAR STRING COMPLETE
010666	001724				BEO	PRG10	
010670	000726				BR	RENIT	
010672	000000			CNTI	0		
010674	177736			TCHARI	-42		
010676	000041			CHARI	41		

010700	047	A33WP61	,BYTE	047,137,127,057,127,137
010701	137			
010702	127			
010703	057			
010704	127			
010705	137			
010706	047	A35WP61	,BYTE	047,133,077,103,079,133
010707	133			
010710	077			
010711	103			
010712	077			
010713	133			
010714	101	AI	,BYTE	101,102,103
010715	102			
010716	103			
010717	104	DI	,BYTE	104,105,106
010720	105			
010721	106			
010722	107	GI	,BYTE	107,110,111
010723	110			
010724	111			
010725	112	JI	,BYTE	112,113,114
010726	113			
010727	114			
010730	115	MI	,BYTE	115,116,117
010731	116			
010732	117			
010733	120	PI	,BYTE	120,121,122
010734	121			
010735	122			
010736	123	SI	,BYTE	123,124,125
010737	124			
010740	125			
010741	126	VI	,BYTE	126,127,130
010742	127			
010743	130			
010744	131	VI	,BYTE	131,132,060
010745	132			
010746	060			
010747	061	ONEI	,BYTE	061,062,063
010750	062			
010751	063			
010752	064	FOURI	,BYTE	064,065,066
010753	065			
010754	066			
010755	067	SEVENI	,BYTE	067,070,071
010756	070			
010757	071			
010760	041	C41I	,BYTE	041,042,043
010761	042			
010762	043			
010763	044	C44I	,BYTE	044,045,046
010764	045			
010765	046			

PALX11	V003	20-JUL-71	16102	PAGE 37-1
010766	047		C471	,BYTE 047,050,051
010767	050			
010770	051			
010771	052		C521	,BYTE 052,053,054
010772	053			
010773	054			
010774	055		C551	,BYTE 055,056,057
010775	056			
010776	057			
010777	072		C721	,BYTE 072,073,074
011000	073			
011001	074			
011002	075		C751	,BYTE 075,076,077
011003	076			
011004	077			
011005	100		C1001	,BYTE 100,133,134
011006	133			
011007	134			
011010	135		C1351	,BYTE 135,136,137
011011	136			
011012	137			
011013	377		C3771	,BYTE 377,000,377
011014	020			
011015	377			
011016	045		T0TST1	,ASCII 'X0SPACE YESX0'
011017	043			
011020	123			
011021	120			
011022	121			
011023	103			
011024	105			
011025	040			
011026	124			
011027	105			
011030	123			
011031	124			
011032	045			
011033	043			
011034	040		T0MRK1	,ASCII ' /0'
011035	040			
011036	040			
011037	040			
011040	040			
011041	040			
011042	040			
011043	040			
011044	057			
011045	100			
011046	040		T0MRK11	,ASCII ' /0'
011047	040			
011050	040			
011051	040			
011052	040			

PALX11 V003 26-JUL-71

16102 PAGE 37-2

011053 040
011054 040
011055 057
011056 100

011057 045
011060 100

011061 055
011062 055
011063 055
011064 055
011065 111
011066 100

011067 055
011070 111
011071 055
011072 120

011073 055
011074 055
011075 055
011076 055
011077 111
011080 055
011081 111
011082 100

011083 134
011084 040
011085 100

011086 045
011087 043
011110 103
011111 101
011112 122
011113 122
011114 111
011115 101
011116 107
011117 105
011120 040
011121 122
011122 105
011123 124
011124 125
011125 122
011126 110
011127 040
011130 124
011131 105
011132 123
011133 124

CRLF1 ,ASCII 'X0'

RM33A1 ,ASCII '----|0'

RM33B1 ,ASCII '=|=0'

RM37A1 ,ASCII '----|=|0'

SPTSTC1 ,ASCII '\ 0'

CRTST1 ,ASCII 'XOCARRIAGE RETURN TESTX00'

PALX11 V003 26-JUL-71

16102 PAGE 37-3

011134 045
011135 043
011136 100

011137 045
011140 043
011141 122
011142 111
011143 107
011144 110
011145 124
011146 040
011147 115
011150 101
011151 122
011152 107
011153 111
011154 110
011155 040
011156 124
011157 105
011160 123
011161 124
011162 045
011163 043
011164 100

RMTST: ,ASCII 'XRIGHT MARGIN TESTX00'

011165 045
011166 043
011167 103
011170 125
011171 122
011172 123
011173 117
011174 122
011175 040
011176 122
011177 111
011200 107
011201 110
011202 124
011203 040
011204 124
011205 125
011206 123
011207 124
011210 045
011211 043
011212 100

SPTST: ,ASCII 'XCURSOR RIGHT TESTX00'

011213 045
011214 043
011215 114
011216 111
011217 110

LFTST: ,ASCII 'XOLINE FEED TESTX00'

51

011220	105
011221	040
011222	106
011223	105
011224	105
011225	104
011226	040
011227	124
011230	105
011231	123
011232	124
011233	045
011234	043
011235	100

011236	045
011237	043
011240	103
011241	110
011242	101
011243	122
011244	101
011245	103
011246	124
011247	105
011250	122
011251	040
011252	124
011253	105
011254	123
011255	124
011256	123
011257	045
011260	043
011261	100

011262	045
011263	043
011264	127
011265	117
011266	122
011267	123
011270	124
011271	040
011272	103
011273	101
011274	123
011275	105
011276	040
011277	120
011300	101
011301	124
011302	124
011303	105
011304	122

CHRTST1 ,ASCII 'NOCHARACTER TESTSNOO'

WCPTST1 ,ASCII 'WOWRST CASE PATTERN TESTSNOO'

011305 116
011306 040
011307 124
011310 105
011311 123
011312 124
011313 045
011314 043
011315 100

011316 045
011317 043
011320 113
011321 131
011322 102
011323 124
011324 040
011325 124
011326 105
011327 123
011330 124
011331 045
011332 043
011333 100

011334 045
011335 120
011336 122
011337 105
011340 123
011341 123
011342 040
011343 101
011344 040
011345 113
011346 105
011347 131
011350 040
011351 127
011352 111
011353 124
011354 110
011355 111
011356 110
011357 040
011360 061
011361 060
011362 040
011363 123
011364 105
011365 103
011366 117
011367 116
011370 104
011371 123

KMSG11 ,ASCII 'XOKYDD TESTXDD'

KMSG21 ,ASCII 'XPRESS A KEY WITHIN 10 SECONDS,0'

011372 056
011373 100

011374 045
011375 043
011376 105
011377 103
011400 110
011401 117
011402 040
011403 124
011404 105
011405 123
011406 124

KMSG31 ,ASCII 'X#ECHO TEST'

011407 045
011410 103
011411 110
011412 101
011413 122
011414 101
011415 103
011416 124
011417 105
011420 122
011421 040
011422 113
011423 105
011424 131
011425 105
011426 104
011427 040
011430 127
011431 111
011432 114
011433 114
011434 040
011435 102
011436 105
011437 040
011440 124
011441 131
011442 120
011443 105
011444 104
011445 056

KMSG3A1 ,ASCII 'X#CHARACTER KEYED WILL BE TYPED.'

011446 045
011447 122
011450 125
011451 102
011452 117
011453 125
011454 124
011455 040

,ASCII 'X#ROUTINE ENDS ROUTINE,X#0'

PALX11

V003

20-JUL-71

16102

PAGE 37-7

011456 105
011457 116
011460 104
011461 123
011462 040
011463 122
011464 117
011465 125
011466 124
011467 111
011470 110
011471 105
011472 056
011473 045
011474 043
011475 100

011476 045
011477 043
011500 117
011501 103
011502 124
011503 101
011504 114
011505 040
011506 105
011507 121
011510 125
011511 111
011512 120
011513 101
011514 114
011515 105
011516 116
011517 124
011520 040
011521 124
011522 105
011523 123
011524 124
011525 100

011526 045
011527 040

011530 040
011531 040
011532 040
011533 040
011534 045
011535 100

011536 045
011537 116
011540 117

KMSG41 ,ASCII 'X#OCTAL EQUIVALENT TEST#'

KMSG51 ,ASCII 'X '

OCTEQVI ,ASCII ' X#'

KMSG61 ,ASCII 'XNO KEYBOARD REQUEST,0'

PALX11

V003

20-JUL-71

16102

PAGE 37-8

011541 040
011542 113
011543 105
011544 131
011545 102
011546 117
011547 101
011550 122
011551 104
011552 040
011553 122
011554 105
011555 121
011556 125
011557 105
011560 123
011561 124
011562 056
011563 100

011564 045
011565 106
011566 101
011567 114
011570 123
011571 105
011572 040
011573 113
011574 131
011575 102
011576 104
011577 040
011000 111
011001 110
011002 124
011003 105
011004 122
011005 122
011006 125
011007 120
011010 124
011011 100

011012 045
011013 043
011014 104
011015 111
011016 123
011017 120
011020 114
011021 101
011022 131
011023 040
011024 105
011025 130

KMSG71 ,ASCII 'XFALSE KYBD INTERRUPT0'

P7MG11 ,ASCII 'XDISPLAY EXERCISERX00'

27

011026 105
011027 122
011030 103
011031 111
011032 123
011033 105
011034 122
011035 045
011036 043
011037 100

011040 045
011041 043
011042 124
011043 131
011044 120
011045 105
011046 040
011047 111
011050 110
011051 040
011052 104
011053 101
011054 124
011055 101
011056 040
011057 072
011060 100

011061 125
011062 040
011063 100

011064 040
011065 040
011066 040
011067 040
011070 040

000001

P7MG21 ,ASCII 'X#TYPE IN DATA 10'

BKSUI ,ASCII 'U 0'

DECVLI ,ASCII ' ' ' '

DENDI ,END

22
75

A 010714
 A1ST 003724
 A33WP6 010700
 A35WP6 010700
 ACNV 003734
 ACNV4 003676
 ACNV6 003650
 ACNV8 003670
 ACNVC 003716
 ACNVM 003750
 ACNVX 003732
 ADYENP 004176
 ARDA 002446
 ARDB 002472
 AREAD 002434
 AT0 004540
 AT0A 004556
 AT0B 004562
 AT0E 004566
 AT1 004572
 AT1A 004610
 AT1B 004614
 AT1E 004620
 AT2 004624
 AT24 005040
 AT24A 005056
 AT24B 005100
 AT24C 005120
 AT24E1 005074
 AT24E2 005116
 AT25 005124
 AT25A 005142
 AT25B 005164
 AT25E 005162
 AT26 005170
 AT26A 005200
 AT26B 005222
 AT26C 005242
 AT26E1 005216
 AT26E2 005240
 AT27 005246
 AT27A 005256
 AT27B 005300
 AT27E 005276
 AT2A 004642
 AT2B 004646
 AT2E 004652
 AT3 004656
 AT30 005304
 AT30A 005314
 AT30B 005324
 AT30E 005322

AT31 005330
 AT31A 005340
 AT31B 005362
 AT31E 005360
 AT32 005366
 AT32A 005376
 AT32B 005424
 AT32E 005422
 AT33 005430
 AT33A 005440
 AT33B 005464
 AT33E 005462
 AT34 005470
 AT34A 005504
 AT34B 005526
 AT34C 005532
 AT34E 005524
 AT35 005536
 AT35A 005552
 AT35B 005574
 AT35E 005604
 AT36 005612
 AT36A 005626
 AT36B 005662
 AT36C 005664
 AT37 005674
 AT37A 005704
 AT37B 005732
 AT37C 005742
 AT37D 005756
 AT37E1 005730
 AT37E2 005762
 AT3A 004674
 AT3B 004700
 AT3E 004704
 AT4 004710
 AT40 005770
 AT40A 006004
 AT40B 006042
 AT40C 006044
 AT41 006054
 AT41A 006074
 AT41B 006120
 AT41C 006132
 AT42 006142
 AT42A 006156
 AT42B 006204
 AT42E 006202
 AT4A 004726
 AT4B 004750
 AT4C 004770
 AT4E1 004744

AT4E2 004766
 AT5 004774
 AT5A 005012
 AT5B 005034
 AT5E 005032
 BCHECK 003250
 BDCNV 004070
 BDCNVA 004110
 BELL 000007
 BIT0 002000
 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 011661
 BLK2 012016
 BLK0B 012014
 BLKCC 012126
 BLOCK1 011673
 BLOCK2 012009
 BLOCKA 011671
 BLOCKB 012003
 BLOCKC 012119
 BMOVA 004020
 BMOVE 004010
 BRCTR 002742
 BREAD 002644
 BREADA 002722
 BREADB 002712
 BREADC 002732
 BSYNC 003310
 BYPHAN 000400
 C100 011009
 C1112 010330
 C1112A 010350
 C1112M 010322
 C135 011010
 C377 011013
 C41 010760
 C44 010763
 C47 010766

C52 010771
 C59 010774
 C92 010777
 C95 011002
 CC 177776
 CHAIN 104012
 CHAINN 001672
 CHALT 104005
 CHAR 010676
 CHK33 002250
 CHK33B 002270
 CHK35 002272
 CHKASR 104022
 CMLT 001366
 CMNA 001726
 CMNAA 001754
 CMNB 001762
 CMR1 001346
 CMR2 001350
 CMR3 001352
 CMRTST 011236
 CK33 104013
 CK35 104014
 CKASR 002310
 CLEAN 002122
 CNT 010672
 CNVCTR 004170
 CPRDY 004032
 CPRDYA 004042
 CRBUF 001344
 CRLF 011057
 CRTA 001554
 CRTST 011100
 C90 006242
 C90A 006266
 C90B 006276
 C90C 006304
 C91 006344
 C910 007116
 C911 007130
 C912 007142
 C913 007154
 C914 007166
 C915 007200
 C916 007212
 C917 007224
 C91A 006370
 C92 006410
 C920 007236
 C921 007250
 C922 007262
 C923 007274

CT24	007306	ERROR	104003	KSTART	001232	PP1	003524
CT25	007320	ETB	007630	KTA	010366	PP1P	003532
CT26	007332	ETBA	007642	KTB	010374	RCMSK	003244
CT27	007344	ETBB	007700	LFTST	011213	RCNT	001342
CT2A	006426	ETBC	007714	LPRGSW	002000	RENIT	010346
CT2B	006446	ETBCA	007722	LSPCH	004046	RIND	003520
CT2C	006454	ET1	007734	LTA	010456	RH33A	011061
CT2D	006476	ET1A	007744	LTB	010476	RH33B	011067
CT2E	006552	ET2	010014	M	010730	RH37A	011073
CT3	006562	ET2A	010034	MACHER	000004	RMB	006404
CT3B	007356	FBALL	004366	MANUAL	100000	RMTST	011137
CT31	007370	FBF3	004322	NITMSW	004000	RNCNT	003246
CT32	007402	FBF3A	004332	NOP	000240	RNGEN	002572
CT33	007440	FBF3B	004340	NPRTSW	020000	RP1	002640
CT33A	007466	FORWD	002010	NTRCSW	010000	RP2	002642
CT34	007522	FORWDA	002050	NXTST	001240	RSTPC	002430
CT34A	007550	FORWDB	002056	OCTEQV	011530	RSTPSW	002432
CT3A	006606	FOUR	010752	ONE	010747	RSTREG	104021
CT3B	006636	FW336	004426	OPEN	000000	RSTRG	002374
CT4	006642	FW356	004466	P	010733	RYNNO	001236
CT4A	006672	G	010722	P7MG1	011612	S	010736
CT4B	006704	GETRDY	001574	P7MG2	011640	SAVREG	104020
CT5	007054	GKBCR	010232	PIND	003526	SAVRG	002334
CT6	007072	GOTST	002074	POPSP	005726	SCOPSW	040000
CT7	007104	GUTSTA	002114	POPSP2	022620	SCOPTX	001244
CTRA	001356	GRCNT	003224	PRG0	004526	SEVEN	010755
CTRB	001360	GTBIN	003534	PRG1	006210	SHALT	001472
CTRC	001362	GTBINP	003602	PRG1B	010540	SHLTA	001506
CTRD	001364	GTROYA	001606	PRG1BA	010566	SPOOT	001204
CURTST	001234	GTROYB	001612	PRG1BC	010554	SPCNT	006752
D	010717	GTROYC	001630	PRG1BD	010562	SPTST	011165
DATCHK	104004	GTROYD	001656	PRG2	007604	SPTSTC	011103
DATHLT	104017	HLTSH	100000	PRG3	010104	SR	177570
DECVAL	011664	HTA	010114	PRG4	010264	SRESEY	104011
DELAY	104400	HTB	010146	PRG5	010274	SRSET	001572
DEND	011671	HTC	010204	PRG6	010364	SRSETY	002554
DIGIT	004172	HTD	010216	PRG7	010436	SRTSW	001000
DLCNT	003162	ICTR	001242	PRGEN0	002012	STAL	003164
OLY	003120	INBIN	003476	PRG10	001246	STALA	003216
OLYA	003140	INCPRG	001550	PRGNUM	001230	STALAA	003176
OLYB	003144	INCRYN	001666	PRGTAB	001290	STALB	003220
OTCHK	001412	ITA	010266	PRTY0	000000	STALL	104002
UTCHKA	001430	J	010725	PRTY1	000040	START	001510
UTHLT	001422	JTA	010276	PRTY2	000100	STBF	004304
EHALT	104010	KMSG1	011316	PRTY3	000140	SPLMSK	003222
EHLT	001400	KMSG2	011334	PRTY4	000200	SPLSPV	002524
EMTA	002172	KMSG3	011374	PRTY5	000240	SPLSRV	002474
EMTINT	002146	KMSG3A	011407	PRTY6	000300	SPPCHV	104007
EMTAD	001272	KMSG4	011476	PRTY7	000340	SPPA	002542
ERCYR	001354	KMSG5	011526	PSW	177776	SPPRA	002512
ERR	001432	KMSG6	011536	PT0	003522	STRDRV	104006
ERRA	001452	KMSG7	011564	PT0P	003530	SUBTEN	004130

SUBTNA	004134	TYPSA	003112
SUBINB	004150	TYPSB	003114
SVRPC	002370	V	010741
SVRPSH	002372	WCPTST	011262
SYCTRA	003472	XTY	010360
SYCTRB	003474	Y	010744
SYNCA	003370		
SYNCB	003404		
SYNCC	003436		
SYNCD	003442		
SYNCE	003464		
TADP	006754		
TADPA	006766		
TADPB	006776		
TADPC	007014		
TBCNT	006750		
TBMHK	011034		
TBMHK1	011046		
TBTST	011016		
TCHAR	010674		
TCTH	004242		
TENPWR	004174		
TKB	001210		
TKLVL	001220		
TKS	001206		
TKVIR	001216		
TPB	001214		
TPBM	006724		
TPBMA	006734		
TPLJA	004266		
TPLVL	001224		
TPS	001212		
TPVIR	001222		
TRPA	002236		
TRPINT	002212		
TRPTAB	001340		
TTYIYP	001226		
TYP	002744		
TYPA	002754		
TYPC	002772		
TYPJ	003020		
TYPJAT	003064		
TYPE	104020		
TYPES	104001		
TYPF	003036		
TYPL	003050		
TYPLJ	004244		
TYPLA	004216		
TYPLB	004222		
TYPLN	004210		
TYPLN3	104016		
TYPS	003000		

PALX11 V003 20-JUL-71

10102 PAGE 37-18

ERRORS DETECTED: 0

RUN-TIME: 24 SECONDS

5K CORE USED

49