

IDENTIFICATION

Product Code: MAINDEC-08-D2Q D-D

Product Name: Family of 8 ASR33/35
Teletype Tests, Part 2

Date Created: June 4, 1968

Maintainer: Diagnostics Group

(1)

(1)

(1)

1. ABSTRACT

The Family-of-8 ASR33/35 Teletype Tests, Part 2 is the second part of a 2 part package used to test the ASR33 or ASR35 Teletype when attached to a Family-of-8 system.

Part 2 contains nine selectable programs numbered from 0 to 10 (octal). The programs are selected by means of Switch Register (SR).

The available programs are:

PRG0	Printer Test
PRG1	Punch Test
PRG2	Keyboard Test
PRG3	Combined Reader, Printer, Punch Test
PRG4	Printer Exerciser. Prints lines of characters stored in LOC 0021 and 0022. No stalls.
PRG5	Same as PRG4, but stalls between characters.
PRG6	Punch Exerciser. Punches and read checks data blocks of data stored in LOC 0021 and 0022. No stalls.
PRG7	Same as PRG6, but random stalls between characters punched.
PRG10	Punch Exerciser. Punches and read checks blocks of Binary Count pattern. Random stalls between characters punched.

2. REQUIREMENTS

2.1 Equipment

- a. Standard PDP-8/S, PDP-8, or PDP-8/I with
- b. ASR33 or ASR35 Teletype.

2.2 Storage

Locations 0000 through 5173 are used.

2.3 Preliminary Programs

Family-of-8 ASR33/35 Teletype Tests, Part 1. PRG0, PRG1, and PRG2 must have been run successfully.

3. LOADING PROCEDURES

3.1 Method

The Binary Loader is used to load the program.

4. STARTING PROCEDURES (PRG0)

4.1 Control Switch Settings (PRG0)

SR0	Halt at end of routine. Routine number in AC.
SR1	Select routine whose number is set in SR6 through SR11.
SR2	Loop program.
SR6 through SR11	Routine number to be selected.

4.2 Starting Addresses (PRG0)

This program starts at LOC 0200.

4.3 Program and/or Operator Action (PRG0)

- a. Insure Teletype is on-line.
- b. Turn off Teletype reader and punch.
- c. Load address 0200.
- d. Set SR to 0000.
- e. Press START.
- f. Program halts at LOC 0232 to permit setting of options.
- g. Select desired options, if any, in SR. For normal run SR should be 0000. Press

CONTINUE.

- h. Program is executed and halts at program end halt at LOC 0274, unless prevented from ending, by SR options.

NOTE

The resulting printouts during execution of PRG0 must be verified visually by user to determine correct teleprinter operation. Refer to Section 9. Program description.

4.A STARTING PROCEDURES (PRG1)

4.1A Control Switch Settings (PRG1)

SR0	Halt at end of routine. Routine number in AC.
SR1	Select routine whose number is set in SR6 through SR11.
SR2	Loop Program.
SR5=1	Halt on error. Bad character in AC.
SR5=0	Halt at end of data block. Error count in AC.
SR6 through SR11	Routine number to be selected.

4.2A Starting Addresses (PRG1)

This program starts at LOC 0200.

4.3A Program and/or Operator Action (PRG1)

- a. Turn on Teletype punch.
- b. With Teletype off-line, punch a section of blank leader about 6 in. long. Return Teletype to on-line position.
- c. Load leader on reader, leaving very little slack between punch and reader.
- d. Turn on reader.
- e. Load address 0200.
- f. Set SR to 0001.
- g. Press START.
- h. Program halts at LOC 0232 to permit setting of options.
- i. Set desired options, if any, in SR. For normal run, set SR to 0000. Press CONTINUE.
- j. Program is executed and halts at program end halt at LOC 0274, unless prevented from ending by SR options, or if errors occur.

4.B STARTING PROCEDURES (PRG2)

4.1B Control Switch Settings (PRG2)

SR0	Halt at end of routine. Routine number in AC.
SR1	Select routine whose number is set in SR6 through SR11.
SR2	Loop Program.
SR6 through SR11	Routine number to be selected.

4.2B Starting Addresses (PRG2)

This program starts at LOC 0200.

4.3B Program and/or Operator Action (PRG2)

- a. Insure Teletype is on-line.
- b. Turn off Teletype reader and punch.
- c. Load address 0200.
- d. Set SR to 0002.
- e. Press START.
- f. Program title is printed and program halts at LOC 0232 to permit setting of options.
- g. Set desired options, if any, in SR. For normal run, set SR to 0000. Press CONTINUE.
- h. Follow program instructions.
- i. When last routine is completed, and provided that no SR options prevent it, the program stops at program end halt at LOC 0274.

NOTE

Correct operation of the keyboard is determined by user, by checking that the printed characters match with the characters keyed.

4.C STARTING PROCEDURES (PRG3)

4.1C Control Switch Settings (PRG3)

SR0	Halt at end of routine. Routine number in AC.
SR1	Select routine whose number is set in SR6 through SR11.
SR2	Loop program.
SR5=1	Halt on error. Bad character in AC.
SR5=0	Halt at end of data block if errors occurred. Error count in AC.
SR6 through SR11	Routine number to be selected.

4.2C Starting Addresses (PRG3)

This program starts at LOC 0200.

4.3C Program and/or Operator Action (PRG3)

- a. Turn on Teletype punch.
- b. With Teletype off-line, punch a section of blank leader about 6 in. long. Return Teletype to on-line position.
- c. Load leader on reader, leaving very little slack between punch and reader.
- d. Turn on reader.
- e. Load address 0200.
- f. Set SR to 0003.
- g. Press START.
- h. Program halts at LOC 0232 to permit setting of options.
- i. Set desired options, if any, in SR. For normal run, set SR to 0000. Press CONTINUE.
- j. Program is executed and halts at program end halt at LOC 0274, unless prevented from ending, by SR options, or if errors occur.

4.D STARTING PROCEDURES (PRG4 and PRG5)

4.1D Control Switch Settings (PRG4 and PRG5)

None

4.2D Starting Addresses (PRG4 and PRG5)

Both programs are started at LOC 0200.

4.3D Program and/or Operator Action (PRG4 and PRG5)

- a. Insure Teletype is on-line.
- b. Turn off Teletype reader and punch.
- c. Deposit in LOC 0021 and 0022 the 8-bit codes for characters to be printed.
- d. For PRG5, deposit in LOC 0023, the desired stall count in 2's complement form. A count of -1 gives a 1 ms stall, etc.
- e. Load address 0200.
- f. Set SR to 0004, or 0005.
- g. Press START.
- h. The program runs continuously, printing lines with characters stored in LOC 0021 and 0021.

4.E STARTING PROCEDURES (PRG6, PRG7, and PRG10)

4.1E Control Switch Settings (PRG6, PRG7, and PRG10)

SR5=1 Halt on error. Bad character in AC.

SR5=0 Halt at end of data block if errors occurred. Error count in AC.

4.2E Starting Addresses (PRG6, PRG7, and PRG10)

These programs start at LOC 0200.

4.3E Program and/or Operator Action (PRG6, PRG7, and PRG10)

- a. Turn on Teletype punch.
- b. With Teletype off-line, punch a section of blank leader about 6 in. long. Return Teletype to on-line position.
- c. Load leader on reader, leaving very little slack between punch and reader.
- d. Turn on reader.
- e. For PRG6 and PRG7, deposit in LOC 0021 and 0022 the 8-bit codes for characters to be punched.
- f. Load address 0200.
- g. Set SR to 0006, 0007, or 0010.
- h. Press START.
- i. The program runs continuously, unless errors occur.

5. OPERATING PROCEDURE

5.1 Program and/or Operator Action

5.1.1 Normal Halts

- LOC 0232 SR SET halt. Occurs to permit setting of desired options. Press CONTINUE. (PRG0, PRG1, PRG2, PRG3).
- LOC 0274 Program end halt. Occurs if no "loop program" option is set. Set desired options and press CONTINUE. If no options are set, this halt reoccurs. (PRG0, PRG1, PRG2, PRG3).
- LOC 0320 Routine end halt. Occurs at end of routine if SR0 = 1. To proceed, press CONTINUE. (PRG0, PRG1, PRG2, PRG3).

6. ERRORS

6.1 Error Halts and Description

- LOC 0177 Incorrect program number selected. Set SR to correct program number and press CONTINUE. (All programs).
- LOC 0255 Nonexistent routine selected. Set correct routine number in SR6 through SR11 and press CONTINUE. (PRG0, PRG1, PRG2, PRG3).
- LOC 1137 Sync error halt. Sync reader subroutine has not found sync character within 145 characters. Position tape in reader so that sync character (rubout) is within 145 characters from read station, and press CONTINUE. (PRG1, PRG3, PRG6, PRG7 and PRG10).
- LOC 1160 Unexpected Interrupt. A non-Teletype device has caused interrupt. Turn off device, and press CONTINUE. (PRG1, PRG3, PRG6, PRG7, and PRG10).
- LOC 1343 Read Check error A. Bad character in AC. Press CONTINUE. (SR5 must be on).
- LOC 1346 Read check error B. Follow up halt. Correct character in AC. To proceed, press CONTINUE. (PRG1, PRG3, PRG6, PRG7, PRG10).
- LOC 1356 Block errors halt. Number of errors in AC. To proceed press CONTINUE. (SR5 must be off). (PRG1, PRG3, PRG6, PRG7, PRG10).

7. RESTRICTIONS

7.1 Starting Restrictions

All programs must be started at LOC 0200.

7.2 Operating Restrictions

PRG0 and PRG1 must precede execution of PRG3. PRG0 must precede execution of PRG2.

8. MISCELLANEOUS

8.1 Execution Time

PRG0 execution time: 15 minutes

PRG1 execution time: 19 minutes

PRG2 execution time: User dependent

PRG3 execution time: 37 minutes

PRG4 through PRG10 are continuous running programs.

9. PROGRAM DESCRIPTIONS

The Family-of-8 ASR33/35 Teletype Tests Part 2, consists of 9 programs numbered from 0 to 10 (octal).

9.1 PRG0 - Printer Test

This program contains 31 routines numbered from 0 to 36 (octal).

RTN0 Carriage return test. Checks ability of carriage return to print position 1 from all other print positions. No printing should occur in any print position other than position 1.

RTN1 Right margin test. This test shows when the right margin is not correctly adjusted. The test prints 14 groups of ---- I followed by characters - I -. A correctly adjusted margin will give the following printout:

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

The I's are printed to facilitate counting print positions.

RTN2 Space Test. The test prints / in alternate positions of the line. After a double carriage return it scapes to the blank positions and prints a left slant slash. A double carriage return is issued after printing each left slant slash.

RTN3 Line Feed Test. The test prints a left slant slash followed by a line feed, followed by a 250 ms delay until 72 slashes have been printed. The result should appear to be a left slanted line from position 1 to 72. Vertical spacing variations should be apparent if adjustment is required.

RTN4 Types line of characters ABC.

RTN5 Types line of characters DEF.

RTN6	Types line of characters	GHI.
RTN7	Types line of characters	JKL.
RTN10	Types line of characters	MNO.
RTN11	Types line of characters	PQR.
RTN12	Types line of characters	STU.
RTN13	Types line of characters	VWX.
RTN14	Types line of characters	YZO.
RTN15	Types line of characters	123
RTN16	Types line of characters	456
RTN17	Types line of characters	789
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	-./
RTN25	Types line of characters	: ; <
RTN26	Types line of characters	= > ?
RTN27	Types line of characters	@ [\
RTN30	Types line of characters	[↑ ←
RTN31	Types line of all characters	.
RTN32	Types line of all characters. Fixed delay between characters in a line. Delay is determined at random.	
RTN33	Types six lines of ASR33 WORST CASE PATTERN.	
RTN34	Types six lines of ASR33 WORST CASE PATTERN. Fixed delay between characters in a line. Delay is determined at random. The ASR33 WORST CASE PATTERN consists of characters ← W/W←	
RTN35	Types six lines of ASR35 WORST CASE PATTERN.	
RTN36	Types six lines of ASR35 WORST CASE PATTERN. Fixed delay between character in a line. Delay is determined at random. The ASR35 WORST CASE PATTERN consists of characters [? C ? [

9.2 PRG1 - Punch Test

This program contains 15 routines numbered from 0 to 16 (octal). The test sequence used by the routines is:

- a. Set up data block
- b. Punch leader
- c. Punch sync character (Rubout)
- d. Punch data block
- e. Sync the reader
- f. Read data block
- g. Punch trailer
- h. Wait for reader to complete reading of data block before going to next routine.

RTN0	Punch and read check block of all 0s.
RTN1	Punch and read check block of channel 1.
RTN2	Punch and read check block of channel 2.
RTN3	Punch and read check block of channel 3.
RTN4	Punch and read check block of channel 4.
RTN5	Punch and read check block of channel 5.
RTN6	Punch and read check block of channel 6.
RTN7	Punch and read check block of channel 7.
RTN10	Punch and read check block of channel 8.
RTN11	Punch and read check block of sliding 1 pattern.
RTN12	Punch and read check block of sliding 0 pattern.
RTN13	Punch and read check block of 1s and 0s pattern.
RTN14	Same as RTN13, but random delay between characters punched.
RTN15	Punch and read check block of binary count pattern.
RTN16	Same as RTN15, but random delay between characters punched.

9.3 PRG2 - Keyboard Test

This program contains 3 routines numbered from 0 to 2.

RTN0	Checks that KSF command skips when flag = 1. Test is done 1000 times.
RTN1	Echo Test. Any characters read from keyboard are typed. Correct operation verification is done visually by user. Reading a rubout character ends the test.
RTN2	Octal equivalence test. The octal equivalent of any characters keyed is typed. Reading a rubout ends the test.

9.4 PRG3 - Combined Reader, Printer, Punch Test

This program contains 27 routines numbered from 0 to 32 (octal). All routines use the following test sequence:

- a. Fill core block with data to be punched/printed.
- b. Punch leader.
- c. Punch sync character.
- d. Punch data block (no delay between characters).
- e. Sync the reader.
- f. Read/Check data block (Random delay between characters).
- g. Punch data block (Random delay between characters).
- h. Read data block (no delay between characters).
- i. Punch trailer.
- j. Wait for reader to complete reading data block.
- k. End of test sequence.

RTN0	Punch/Print and read check block of ABC
RTN1	Punch/Print and read check block of DEF
RTN2	Punch/Print and read check block of GHI
RTN3	Punch/Print and read check block of JKL
RTN4	Punch/Print and read check block of MNO
RTN5	Punch/Print and read check block of PQR
RTN6	Punch/Print and read check block of STU
RTN7	Punch/Print and read check block of VWX
RTN10	Punch/Print and read check block of YZ0
RTN11	Punch/Print and read check block of 123
RTN12	Punch/Print and read check block of 456
RTN13	Punch/Print and read check block of 789
RTN14	Punch/Print and read check block of ! "#
RTN15	Punch/Print and read check block of \$ % &
RTN16	Punch/Print and read check block of ' () .
RTN17	Punch/Print and read check block of * + ,
RTN20	Punch/Print and read check block of - . /
RTN21	Punch/Print and read check block of : ; <
RTN22	Punch/Print and read check block of = > ?
RTN23	Punch/Print and read check block of @ [\

RTN24	Punch/Print and read check block of] ↑ ←
RTN25	Punch/Print and read check block of all printable characters.
RTN26	Punch/Print and read check block of ASR33 Printer worst case pattern (▲ ← W/)
RTN27	Punch/Print and read check block of ASR33 Printer worst case pattern with interspersed blanks.
RTN30	Punch/Print and read check block of ASR35 Printer worst case pattern. (▼ [? C)
RTN31	Punch/Print and read check block of ASR35 Printer worst case pattern with interspersed blanks.
RTN32	Punch/Print and read check blocks of space, rubout (1s and 0s).

9.5 PRG4 - Printer Exerciser

Prints lines with data stored in LOC 0021 and 0022, no stalls.

9.6 PRG5 - Printer Exerciser

Prints lines with data stored in LOC 0021 and 0022. Fixed delay between characters.

Delay is determined at random.

9.7 PRG6, PRG7, and PRG10 Punch Exerciser

PRG6 punches and read checks data blocks with data stored in LOC 0021 and 0022. No stalls.

PRG7 is the same as PRG6, but random stalls between characters punched.

PRG10 punches and read checks blocks of Binary Count pattern. Random stalls between characters.

The three exercisers use the following sequence:

- a. Set up data block
- b. Punch leader
- c. Punch sync character (rubout)
- d. Punch data block
- e. Sync the reader
- f. Read data block
- g. Punch data block
- h. Back to step f.

B. J. Jolley

/FAMILY-CP-B ASR33/33 TELETYPE TESTS = PART 2

/PRG0-PRINTER TEST

/PRG1-PUNCH TEST

/PRG2-KEYBOARD TEST

/PRG3-COMBINED READER, PRINTER, PUNCH TEST.

/PRG4-PRINTER EXERCISER, PRINTS LINES OF CHARACTERS STORED IN LOC 0021 AND 0022.

/ NO STALLS, PRINTS LINES OF CHARACTERS STORED IN LOC 0021 AND 0022. STALLS

/PRG5-PRINTER EXERCISER, PRINTS LINES OF CHARACTERS STORED IN LOC 0021 AND 0022. STALLS

/ BETWEEN CHARACTERS,

/PRG6-PUNCH EXERCISER, PUNCHES AND READ CHECKS DATA BLOCKS OF DATA STORED IN LOC 0021

/ AND 0022, NO STALLS

/PRG7- SAME AS PRG6, BUT RANDOM STALLS BETWEEN CHARACTERS PUNCHED.

/PRG8-PUNCH EXERCISER, PUNCHES AND READ CHECKS BLOCKS OF BINARY COUNT PATTERN,

/ RANDOM STALLS

/

/STARTING ADDRESS:0200

/

/SR OPTIONS

/

/SR0-MALT AT END OF ROUTINE, ROUTINE NUMBER IN AC.

/SR1-SELECT ROUTINE WHERE NUMBER IS SET IN SR6 TO 11.

/SR2-LOOP PROGRAM

/SR3=1-MALT ON ERROR, BAD CHARACTER IN AC,

/SR4=0-MALT AT END OF DATA BLOCK, ERROR COUNT IN AC

/SR5 TO SR11 = ROUTINE NUMBER TO BE SELECTED,

/

/ASR33/3D TELETYPE TESTS - PART 2

```

0000
0000
0001 5001
0002 0002
0003 0003
0004 0004
0005 5402
0006 0000
0007 0020
0008 0000
0009 0000
0010 0000
0011 0000
0012 0000
0013 0000
0014 0000
0015 0000
0016 0017
0017 7770
0018 0161
0019 0000
0020 7444
0021 7764
0022 0000
0023 0000
0024 0000
0025 0000
0026 0017
0027 7770
0028 0161
0029 0000
0030 7444
0031 7764
0032 0000
0033 0000
0034 0000
0035 0000
0036 0000
0037 0077
0038 0000
0039 0000
0040 0000
0041 0000
0042 0000
0043 0335
0044 0257
0045 0315
0046 0322
0047 0400
0048 0444
0049 1246
0050 0350
0051 0360
0052 0520
0053 0542
0054 1530
0055 0621
0056 1510
0057 0637
0058 0661
0059 0677
0060 0710

```

```

*0
0000
JMP 1
2
3
JMP 1 2
0
*20
KSTART, 0
PTEMP, 0
PTEMP1, 0
DELAYM, 0
DELAYS, 0
PRGNUM, 0
PRGMSK, 17
PRGLIM, -10
PSW, PRGTAB
CPIU, 0
KPB, -334
KPB5, -14
RTNNO, 0
CURTST, 2
NXTST, 0
TSTMSK, 77
MSCTR, 0
MILCTR, 0
MILI, 0
ULYMS, ULYMS
CHAIN, CHAINN
SHLT, SHALT
SETCTR, STCTR
RANNU, RANGEN
XTYPST, TYPSTG
URDY, RRDY
OLYCNT, OLUNT
UCKLF, CKLF
UPUNCH, PUNCH
MOVE, MOVVE
UTYPE, TYPE
USTBF, STBF
UTPLNS, TYPLNS
UFBF3, FBF3
UFBALL, FBALL
JFBTMP, FBTMP
JFK334, FW334

```

0065	0735	UFW335,	FWS36
0066	0751	UFW335,	FWS35
0067	1000	UFW354,	FWS34
0070	1010	UFW355,	FWS36
0071	1034	UFW355,	FWS35
0072	1067	UPLTLR,	PLTLR
0073	1102	UPSYNG,	PSYNG
0074	1100	URSYNG,	RSYNG
0075	1400	UDLMSK,	ULMSH
0076	1417	UUCNTP,	ULCUNT
0077	1161	UUT,	UUT
0100	1217	UPBLK,	PBLK
0101	1225	UPBLKR,	PBLKR
0102	1270	URUBLK,	KUBLK
0103	1277	URBLKR,	KUBLKR
0104	1445	UNTST,	NTST
0105	1500	UCNTST,	UNTST
0106	1600	UASCCN,	ASCCN
0107	1461	USTST,	STST
0110	1052	CHECK,	CHK
0111	0600	INPATI,	INITPT
0112	0607	GETPT,	GETPTT
0113	0000	TEMP,	0
0114	0000	TEMP1,	0
0115	0000	TEMPU,	0
0116	0000	UTEMP,	0
0117	0000	UTEMP1,	0
0120	0000	UTEMP2,	0
0121	0000	CTRA,	0
0122	0000	CIRB,	0
0123	0100	SRPMSK,	LR0
0124	0000	ERRCR,	0
0125	0000	ERRCTR,	0
0126	0277	DLYMSK,	277
0127	0000	PFLAG,	0
0130	0000	BLACNI,	0
0131	0215	CR,	215
0132	0214	LF,	214
0133	7401	MREOUT,	-377
0134	0000	KBUSY,	0
0135	0000	LINK,	0
0136	0000	AC,	0
0137	0240	SPACE,	240
0140	0257	C257,	257
0141	0334	C334,	334

M1, -1
M2, -2
M16, -16
M44, -44
M110, -110
M111, -111

TEM0, 0
TEM1, 0
FLAG, 0
K77, //
M40, -40
C100, 100
C240, 240
SKIPMA, SMA
SKIPPA, SPA
PRGTAB, PRG0
PRG1
PRG2
PRG3
PRG4
PRG5
PRG6
PRG7
PRG10

SETLOG=JMS I SETCTR
MOVE=JMS I UMOVE
DELAY=JMS I DLYIMS

0142 7777
0143 7776
0144 7762
0145 7734
0146 7670
0147 7667
0150 0000
0151 0000
0152 0000
0153 0077
0154 7740
0155 0100
0156 0240
0157 7500
0160 7510
0161 2400
0162 3060
0163 3440
0164 3537
0165 4076
0166 4104
0167 4111
0170 4131
0171 4151
4446
4450
4445

```

0177 0177 /INCORRECT PROGRAM NUMBER HALT.
7402 7402 /HEAD SR
7604 7604
0020 0020
0201 0201
0202 0202
0203 0203
0204 0204
0205 0205
0206 0206
0207 0207
0210 0210
0211 0211
0212 0212
0213 0213
0214 0214
0215 0215
0216 0216
0217 0217
0220 0220
0221 0221
0222 0222
0223 0223
0224 0224
0225 0225
0226 0226
0227 0227
0230 0230
0231 0231
0232 0232
0233 0233
0234 0234
0235 0235
0236 0236
0237 0237
0240 0240
0241 0241
0242 0242
0243 0243
0244 0244
0245 0245
0246 0246
0247 0247
0250 0250
0251 0251
0252 0252
0253 0253
0254 0254
0255 0255
0256 0256

HLT 0177
LAS 7402
AND PRGMSK
TAU PRGLIM
SMA SEA /VALID PROGRAM NUMBER?
JMP 177 /NO, GO TO LOC 177
LAS 7604
AND PRGMSK
UCA PRGNUM /SAVE PROGRAM NUMBER
TAU PRGNUM /DEVELOP PROGRAM
TAU PSW /START ADDRESS AND
UCA TEMP
TAU I TEMP
UCA PRGADR
CLA CLL CMA RAR /DETERMINE CPU ID,
SPA CLA /IS IT PDP8/8I?
JMP *3 /NO, IT IS A PDP8/S.
TAU KP8 /YES, IT IS PDP8/8I
SKP
TAU KP8S
UCA MILL
JMS I UMOVE
P
1
72
JMP I,*1 /GO TO SELECTED PROGRAM,
PRGADR, 0
SRSET, HLT CLA
GETRDY, CLA KSTART /GET ADDRESS OF 1ST ROUTINE
UCA NXTST /STORE AT NXTST
JMS FORWD /READ SR
LAS
HAL
SMA
JMP I CURTST /ROUTINE SELECT?
LAS /NO, START WITH CURRENT ROUTINE.
AND TSTMSK /YES, READ SR
CLA /GET ROUTINE NUMBER,
TAU RTNNO /2'S COMPLEMENT IT.
SNA CLA /ADD CURRENT ROUTINE NUMBER.
JMP I CURTST /IS IT THIS ROUTINE?
TAU NXTST /YES, GO DO IT.
IAC /NO. IS THIS THE LAST ROUTINE?
SEA CLA
JMP GETRDY+3
HLT
JMP GETRDY

/INCORRECT ROUTINE NUMBER
/YES, INCORRECT ROUTINE NUMBER

```

0257	4515	CHAINN, JMS SHALT	/HALT? (SR0) GO CHECK,
0260	7604	LAS	/HEAD SR
0261	7006	RTL	
0262	7630	SZL CLA	/ROUTINE SELECT?(SR1)
0263	5233	JMP GETRUY	/YES,
0264	1036	IAU NXTST	
0265	7001	IAU	
0266	7640	SEA CLA	/LAST ROUTINE?
0267	5236	JMP GETRUY+3	/NO, SET UP TO DO NEXT ROUTINE
0270	7604	LAS	
0271	7006	RTL	
0272	7710	SPA CLA	/LOOP PROGRAM? (SR2)
0273	5233	JMP GETRUY	/YES, GO REPEAT PROGRAM,
0274	7402	HLT	/PROGRAM END HALT
0275	5257	JMP CHAINN	/GO CHECK FOR OPTIONS AGAIN.
0276	0000	FORWD, 0	
0277	7300	CLA CLL	
0300	1436	TAU I NXTST	/GET AND STORE NEXT ROUTINE
0301	3034	UCA RTNNO	/NUMBER,
0302	2036	ISE NXTST	
0303	1036	TAU NXTST	/SET CURRENT
0304	3115	UCA TEMP	/ROUTINE NUMBER
0305	2036	ISE NXTST	
0306	1036	IAU NXTST	/SET CURRENT
0307	3030	UCA CURTST	/ROUTINE ADDRESS,
0310	1515	TAU I TEMP	/SET NEXT
0311	3036	UCA NXTST	/ROUTINE ADDRESS,
0312	5676	JMP I FORWD	/EXIT,
0313	0000	SHALT, 0	
0314	7604	LAS	/HEAD SR,
0315	7700	SMA CLA	/HALT? (SR0)
0316	5715	JMP I SHALT	/NO, EXIT
0317	1034	IAU RTNNO	/GET CURRENT RTN NUMBER
0320	7402	HLT	/UNCONDITIONAL HALT,
0321	5715	JMP I SHALT	/EXIT,

```

0322 0000 STCTR, 0
0323 7200 CLA I STCTR
0324 1722 UCA TEMP /GET CTR ADDRESS
0325 3115 ISZ STCTR /STORE AT TEMP.
0326 2322 TAU I STCTR /GET COUNT AND STORE
0327 1722 UCA I TEMP /PER C(TEMP)
0330 3515 ISZ STCTR /EXIT.
0331 2322 JMP I STCTR
0332 5722
0333 0000
0334 7300 CLA CLL /GET MILLISECOND COUNT
0335 1025 TAU DELAY /STORE AT MSCTR
0336 3040 UCA MSCTR
0337 5740 JMP I ,+1
0340 0341 *+1
0341 1042 TAU MILL /GET IMS CONSTANT
0342 3041 UCA MILCTR /STORE IN MILCTR
0343 2041 ISZ MILCTR /DELAYED 1 MILLISECOND?
0344 5345 JMP ,+1 /NO.
0345 2040 ISZ MSCTR /YES, DONE DELAYING?
0346 5337 JMP ,+7 /NO, GO DELAY ANOTHER MILSEC.
0347 5735 JMP I DLYMS /EXIT.
0350 0000
0351 4447 JMS I RANDNO /GENERATE RANDOM NUMBER
0352 0126 AND DLYMSK /MASK OUT UNDESIRED BITS
0353 7450 SNA DLYCNT+1 /RESULT ZERO?
0354 5551 JMP IAC /YES, GET ANOTHER NUMBER
0355 7041 UCA DELAYM /NO, 2'S COMPLEMENT IT
0356 3025 JMP I DLYCNT /STORE AT DELAY
0357 5750
0360 0000
0361 7200 CLA I CRLF
0362 1760 UCA CRCTR
0363 3575 ISZ CRLF
0364 2560 JMS I XTYPST
0365 4450 ,+4
0366 0572 ISZ CRCTR
0367 2375 JMP ,+5
0370 5360 JMP I CRLF
0371 5760
0372 0015
0373 0012
0374 0001
0375 0000 CRCTR, 0

```

*. 17/*1
/RANDOM NUMBER GENERATOR SUBROUTINE
RANGEN, 0

0400	0400
0401	6000
0402	7200
0403	1242
0404	1227
0405	7640
0406	5215
0407	1231
0408	0406
0409	3227
0410	1230
0411	7104
0412	7430
0413	7001
0414	3230
0415	1230
0416	1627
0417	3627
0420	1243
0421	7010
0422	1627
0423	2227
0424	3243
0425	1243
0426	5600
0427	0442
0430	6543
0431	0432
0432	6543
0433	3210
0434	0765
0435	5432
0436	2107
0437	7654
0440	4321
0441	1076
0442	7536
0443	0000

CLA
 TAU RANTND
 TAU RANDEX
 SEA CLA
 JMP RANTAD
 TAU RANTBL
 UCA RANDEX
 TAU RANCON
 CLL HAL
 SEL
 TAU

RANTAU,
 UCA RANCON
 TAU RANCON
 TAU I RANDEX
 UCA I RANDEX
 TAU RANSAY
 KAR
 TAU I RANDEX
 ISE RANDEX
 UCA HANSAY
 TAU RANSAY
 JMP I RANGEN

RANDEX, RANTND
 RANCON, 6543
 RANTBL, 1*1
 6543
 3210
 0765
 5432
 2107
 7654
 4321
 1076

RANTND, 1*1
 RANSAY, 0


```

0444 0000 /TYPE CHARACTER STRING SUBROUTINE
0445 7200 TYPSTG, 0
0446 1644 CLA
0447 3150 TAU I TYPSTG /GET AND STORE
0450 3152 UCA TEMQ /INITIAL ADDRESS
0451 2244 UCA FLAG /CLEAR FLAG.
0452 1950 ISZ TYPSTG /SET UP EXIT
0453 7012 TAU I TEMQ /PICK UP DATA
0454 7012 KTK
0455 7012 KTK
0456 4260 JMS TSC2 /GO TYPE 1ST CHARACTER
0457 1950 TAU I TEMQ /PICK UP DATE
0460 4260 JMS TSC2 /GO TYPE 2ND CHARACTER
0461 2150 ISZ TEMQ /EVEN STRING ADDRESS
0462 5252 JMP TSC1 /GO BACK FOR MORE
0463 0000 0
0464 0150 ANU K77 /MASK OFF 6 BITS
0465 3151 UCA TEMR /SAVE CHARACTER
0466 1152 TAU FLAG /TEST "SPECIAL" FLAG,
0467 7640 SZA CLA
0470 5300 JMP TYPSP /SET TYPE SPECIAL
0471 1151 TAU TEMR /NO, REGULAR CHARACTER
0472 7450 SNA
0473 5276 JMP ,+3 /#EH0?
0474 4317 JMS PRINT /YES, SET FLAG.
0475 5665 JMP I TSC2 /NO, PRINT IT.
0476 2152 ISZ FLAG /RETURN,
0477 5665 JMP I TSC2 /SET "SPECIAL" FLAG,
0500 3152 UCA FLAG /EXIT
0501 1151 TAU TEMR /CLEAR FLAG,
0502 7041 CIA /TEST FOR 0,
0503 7450 SNA TYPAT /0:TYPE "0"
0504 5274 IAC /TEST FOR 01
0505 7001 SNA CLA
0506 7650 JMP I TYPSTG /YES, EXIT CODE.
0507 5644 TAU SKIPMA /ALTER INSTRUCTION
0510 1157 UCA SWITCH /TO BE "SMA"
0511 3321 TAU TEMR /TYPE CHAR
0512 1151 JMS PRINT
0513 4317 TAU SKIPPA /ALTER INSTRUCTION
0514 1160 UCA SWITCH /TO BE "SPA"
0515 3321 JMP I TSC2 /RETURN
0516 5665 0
0517 0000 TAU M40 /COMPARE WITH 40
0520 1154 SPA /OR SMA FOR SPECIAL CODES,
0521 7510 TAU C100
0522 1152 TAU C240
0523 1150 JMS I UPUNCH /GO PRINT CHARACTER
0524 4454 JMP I PRINT /RETURN
0525 5717

```

0526	0000				
0527	2127	ISZ PFLAG	/SET PFLAG		
0530	6246	ILS	/PUNCH/PRINT		
0531	7200	CLA			
0532	1127	TAU PFLAG	/GET C(PFLAG)		
0533	7650	SNA CLA	/FLAG RESET?		
0534	5337	JMP ,+3	/YES		
0535	6041	TSF	/NO, FLAG UP?		
0536	5332	JMP ,+4	/NO,		
0537	6042	TCF	/YES, CLEAR PRINTER FLAG.		
0540	3127	UCA PFLAG	/CLEAR PFLAG		
0541	5726	JMP I PUNCH	/EXIT,		
0542	0000				
0543	7200	CLA			
0544	1742	TAU I MOVVE	/GET AND STORE		
0545	3364	UCA FADUR	/"FROM" ADDRESS		
0546	2342	ISZ MOVVE			
0547	1742	TAU I MOVVE	/GET AND STORE		
0550	3362	UCA TADUR	/"TO" ADDRESS		
0551	2342	ISZ MOVVE			
0552	1742	TAU I MOVVE	/GET AND STORE		
0553	3360	UCA MCTR	/"MOVE" COUNT,		
0554	2342	ISZ MOVVE	/SET UP EXIT,		
0555	1764	TAU I FADUR	/GET "FROM" WORD		
0556	3762	UCA I TADUR	/STORE AT "TO" LOCATION		
0557	2364	ISZ FADUR	/+1 TO FADUR		
0560	2362	ISZ TADUR	/+1 TO TADUR		
0561	2360	ISZ MCTR	/DONE MOVING?		
0562	5352	JMP MOVEA	/NO, REPEAT		
0563	5742	JMP I MOVVE	/YES, EXIT,		
0564	0000				
0565	0000	FADUR,			
0566	0000	TADUR,			
		MCTR,			

```

0600 * 17/+1
0601 /INITIALIZE BINARY PATTERN SUBROUTINE
0602 INITPT, 0
0603 CLA
0604 DCA PT0 /SET PT0=0
0605 JMP I INITPT /EXIT,
0606 0
0607 PT0,
0608 PT1,
0609 PTMSK, 37/
0610 /SUBROUTINE TO SET AC TO NEXT BINARY PATTERN CHARACTER
0611 GETPTT, 0
0612 CLA
0613 IAU PT0 /GET PT0
0614 UCA PT1 /STORE AT PT1
0615 IAU PT1 /GET PT1
0616 IAC /+1 TO AC
0617 AND PTMSK /LIMIT TO 8 BITS
0618 UCA PT0 /STORE AT PT0
0619 IAU PT1 /GET PT1
0620 JMP I GETPTT /EXIT

0621 /SET BUFFER AREA SUBROUTINE
0622 STBF, 0
0623 JMS I UMOVE /MOVE CRLF TO BLOCKA
0624 CR
0625 BLUCKA
0626 -2
0627 JMS I UMOVE /MOVE CRLF TO BLOCKB
0628 CR
0629 BLUCKB
0630 -2
0631 JMS I UMOVE /MOVE CRLF TO BLOCKC
0632 CR
0633 BLUCKC
0634 -2
0635 JMP I STBF /EXIT,
0636 5621

```

0637	0000		
0640	7200	CLA	
0641	1637	LAD I FBF3	/FILL 144 CHARACTER BUFFER
0642	3245	UCA ++5	/WITH 3 CHARACTERS WHOSE
0643	2237	ISE FBF3	/ADDRESS IS SPECIFIED
0644	4455	JMS I UMOVE	/AT CALL+1
0645	0000	0	
0646	4177	BLUCK1	
0647	7775	-3	
0650	4455	JMS I UMOVE	
0651	4177	BLUCK1	
0652	4202	BLUCK1+3	
0653	7675	-100	
0654	4455	JMS I UMOVE	
0655	4177	BLUCK1	
0656	4311	BLUCK2	
0657	7670	-110	
0660	5637	JMP I FBF3	/EXIT
0661	0000	0	
0662	4455	JMS I UMOVE	/FILL 144 CHARACTER BUFFER
0663	1710	A	/WITH ALL PRINTABLE ASCII
0664	4177	BLUCK1	/CHARACTERS,
0665	7701	-77	
0666	4455	JMS I UMOVE	
0667	1710	A	
0670	4276	BLUCK1+77	
0671	7767	-11	
0672	4455	JMS I UMOVE	
0673	4177	BLUCK1	
0674	4311	BLUCK2	
0675	7670	-110	
0676	5661	JMP I FBALL	/EXIT
0677	0000	0	
0700	4455	JMS I UMOVE	/FILL 144 CHARACTER BUFFER
0701	0021	PTEMP	/WITH DATA IN PTEMP
0702	4177	BLUCK1	/AND PTEMP1,
0703	7770	-2	
0704	4455	JMS I UMOVE	
0705	4177	BLUCK1	
0706	4201	BLUCK1+2	
0707	7672	-106	
0710	4455	JMS I UMOVE	
0711	4177	BLUCK1	
0712	4311	BLUCK2	
0713	7670	-110	
0714	5677	JMP I FBTMP	/EXIT,

0715	0000		
0716	445D	JMS I UMOVE	/MOVE 4 CHARACTER ASR33 PRINTER
0717	1644	ASSWPD	/WORST CASE PATTERN TO
0720	4177	BLUCK1	/BLUCK1
0721	7774	-4	
0722	445D	JMS I UMOVE	/FILL BLOCK1 WITH PATTERN
0723	4177	BLUCK1	
0724	4205	BLUCK1+4	
0725	7674	-104	
0726	445D	JMS I UMOVE	/FILL BLOCK2 WITH PATTERN
0727	4177	BLUCK1	
0730	4311	BLUCK2	
0731	7670	-110	
0732	571D	JMP I FW334	/EXIT
0733	0000		
0734	445D	JMS I UMOVE	/MOVE 6 CHARACTER AKS33 PRINTER
0735	1650	ASSWPD	/WORST CASE PATTERN TO
0736	4177	BLUCK1	/BLUCK1
0737	7772	-0	
0740	445D	JMS I UMOVE	/FILL BLOCKS WITH PATTERN
0741	4177	BLUCK1	
0742	420D	BLUCK1+6	
0743	7676	-102	
0744	445D	MOVE	/FILL BLOCK2 WITH PATTERN.
0745	4177	BLUCK1	
0746	4311	BLUCK2	
0747	7670	-110	
0750	5735	JMP I FW336	/EXIT
0751	0000		
0752	445D	JMS I UMOVE	/MOVE 8 CHARACTER ASR33 PRINTER
0753	1656	ASSWPS	/WORST CASE PATTERN WITH
0754	4177	BLUCK1	/INTERSPERSED BLANKS TO BLOCK1.
0755	7770	-10	
0756	445D	JMS I UMOVE	/FILL BLOCK1 WITH PATTERN
0757	4177	BLUCK1	
0760	4207	BLUCK1+10	
0761	7700	-100	
0762	445D	JMS I UMOVE	/FILL BLOCK2 WITH PATTERN
0763	4177	BLUCK1	
0764	4311	BLUCK2	
0765	7670	-110	
0766	5751	JMP I FW335	/EXIT

1000	*. 17/+1		
0000	FW354, 0		
1001 445>	JMS I UMOVE	/MOVE 4 CHARACTER ASR35 PRINTER	
1002 1666	AS2WPA	/WORST CASE PATTERN TO BLOCK1.	
1003 4177	BLOCK1		
1004 7774	-4		
1005 445>	JMS I UMOVE	/FILL BLOCK1 WITH PATTERN	
1006 4177	BLOCK1		
1007 4203	BLOCK1+4		
1010 7674	-104		
1011 445>	JMS I UMOVE	/FILL BLOCK2 WITH PATTERN	
1012 4177	BLOCK1		
1013 4311	BLOCK2		
1014 7670	-110		
1015 5600	JMP I FW354	/EXIT	
1016 0000	0		
1017 445>	JMS I UMOVE	/MOVE 6 CHARACTER ASR35 PRINTER	
1020 1672	AS2WPA	/WORST CASE PATTERN TO BLOCK1	
1021 4177	BLOCK1		
1022 7772	-6		
1023 445>	JMS I UMOVE	/FILL BLOCK1 WITH PATTERN	
1024 4177	BLOCK1		
1025 420>	BLOCK1+6		
1026 7676	-102		
1027 445>	JMS I UMOVE	/FILL BLOCK2 WITH PATTERN	
1030 4177	BLOCK1		
1031 4311	BLOCK2		
1032 7670	-110		
1033 5616	JMP I FW356	/EXIT	
1034 0000	0		
1035 445>	JMS I UMOVE	/MOVE 8 CHARACTER ASR35 PRINTER	
1036 1700	AS2WPS	/WORST CASE PATTERN TO BLOCK1	
1037 4177	BLOCK1		
1040 7770	-10		
1041 445>	JMS I UMOVE	/FILL BLOCK1 WITH PATTERN	
1042 4177	BLOCK1		
1043 4207	BLOCK1+10		
1044 7700	-100		
1045 445>	JMS I UMOVE	/FILL BLOCK2 WITH PATTERN	
1046 4177	BLOCK1		
1047 4311	BLOCK2		
1050 7670	-110		
1051 5634	JMP I FW355	/EXIT	

```

1052 0000 /SUBROUTINE TO COMPARE C(AC) TO C(CALL*1)
1053 3266 CHCK,
1054 1652 DCA WCHK /STORE AC AT WCHK
1055 7041 TAU I CHCK /SET COMPARE DATA
1056 1266 CIA WCHK /2'S COMPLEMENT IT
1057 2252 TAU WCHK /ADD C(WCHK)
1060 7640 ISZ CHCK /SET UP FOR UNEQUAL EXIT
1061 5264 SZA CLA /EQUAL?
1062 2252 JMP ,+3 /NO,
1063 5652 ISZ CHCK /YES, SET UP FOR EQUAL EXIT.
1064 1266 JMP I CHCK /EQUAL EXIT
1065 5652 TAU WCHK /RESTORE AC
1066 0000 JMP I CHCK /UNEQUAL EXIT.
WCHK,
/PUNCH 70 (CODE 376) CHARACTERS SUBROUTINE
PLTLR, 0 JMS I SETCTR /SET P70CTR TO -70
P70CTR -106
TAU LOCDE /GET 376 CODE
JMS I UPUNCH /GO PUNCH IT
ISZ P70CTR /ALL CHARACTERS PUNCHED?
JMP ,+3 /NO, REPEAT,
JMP I PLTLR /YES, EXIT.
P70CTR, 0
LOCDE, 376
/PUNCH SYNC CHARACTER SUBROUTINE (RUBOUT)
PSYNC, 0 CLA CMA /SET AC TO 7777
JMS I UPUNCH /PUNCH A RUBOUT
JMP I PSYNC /EXIT.
/SYNC READER SUBROUTINE
RSYNC, 0 JMS I SETCTR /SET RSCTR TO -145
RSCTR -241
JMS I URROY /WAIT FOR READER NOT BUSY
CLA CMA /HEADER NOT BUSY,
UCA RBUSY /SET READER BUSY INDICATOR
JMS I SETCTR /SET READER INTERRUPT
VCTR /SERVICE RETURN ADDRESS,
KSSERV /ENABLE INTERRUPT
LUN JMP I RSYNC /EXIT
RSCTR, 0
1102 0000
1103 7240
1104 4454
1105 5702
1106 0000
1107 4440
1110 1122
1111 7557
1112 4451
1113 7240
1114 3134
1115 4440
1116 1157
1117 1125
1120 6001
1121 5700
1122 0000

```

1123	6030				
1124	1135	RSSERV,	KRB	/READ	
1125	7040		TAU MRBOUT	/ADD MINUS RUBOUT	
1126	5335		SEA CLA	/IS IT A RUBOUT?	
1127	3134		JMP ,+7 /NO,		
1130	7300		DCA RBUSY	/YES, CLEAR READER BUSY,	
1131	1135		CLA CLL		
1132	7004		TAU LINK		
1133	1136		KAL	/RESTORE LINK	
1134	5400		TAU AC	/RESTORE AC	
1135	2322		JMP I 0	/RETURN	
1136	5477		ISE RSCTR	/145 CHARACTER READ?	
1137	7602		JMP I UOUT	/NO,	
1140	4446		HLT CLA	/YES, NO SYNC,	
1141	1122		JMS I SETCTR	/SET RSCTR TO -145	
1142	7557		RSCTR		
1143	5477		-221		
			JMP I UOUT	/RETURN	
1144	3136	INTSVC,	DCA AC	/SAVE AC	
1145	7010		KAK		
1146	3135		DCA LINK	/SAVE LINK	
1147	6041		TSP	/PUNCH/PRINTER?	
1150	5354		JMP ,+4	/NO,	
1151	6042		TCF	/YES, CLEAR FLAG,	
1152	3127		DCA PFLAG	/CLEAR PFLAG	
1153	5361		JMP OUT	/RETURN	
1154	6031		KSP	/HEADER/KYBD?	
1155	5360		JMP ,+3	/NO ERROR,	
1156	5757	VCTR,	JMP I ,+1	/GO SERVICE READER	
1157	0000		0		
1160	7402	OUT,	HLT	/UNEXPECTED INTERRUPT	
1161	7300		CLA CLL		
1162	1135		TAU LINK		
1163	7004		KAL	/RESTORE LINK	
1164	1136		TAU AC	/RESTORE AC,	
1165	6001		IUN	/ENABLE INTERRUPT	
1166	5400		JMP I 0	/RETURN	


```

1200 * 17/*1
0000 PSTUP, 0
1201 SETLOC
4446 PADUR /PUNCH SETUP
1202 1245 /SET DATA ADDR
1203 4175
1204 4455 /SET BLOCK LENGTH
1205 0130
1206 1244
1207 7777
1210 5600 /EXIT

/ POCR, 0
1211 0000 /PUNCH DATA CHAR SUB,
1212 7200
1213 1645 /GET DATA
1214 2245 /UPDATE PADUR,
1215 4454 /GO PUNCH/PRINT DATA
1216 5611 /EXIT

/ PBLK, 0
1217 0000 /PUNCH DATA BLOCK FULL SPEED
1218 4200
1219 4211 /GO PUNCH CHARACTER
1220 2244 /ALL CHARS PUNCHED?
1221 5221 /NO, REPEAT
1222 5617 /YES, EXIT

/ PBLKR, 0
1225 0000 /PUNCH DATA BLOCK RANDOM STALLS,
1226 4200 /GO DO SET UP
1227 4476 /FILL DELAY BLOCK
1230 4446 /UBLK ADDRESS TO DAP
1231 1243
1232 4575 /GET DELAY WORD
1233 1643
1234 3025
1235 2245 /TO DELAYM
1236 4445 /UPUATE DAP,
1237 4211 /DELAY,
1240 2244 /GO PUNCH CHARACTER
1241 5235 /ALL CHARS PUNCHED?
1242 5625 /NO, REPEAT
1243 0000 /YES, EXIT,
1244 0000
1245 0000
DAP, 0
PCTR, 0
PADUR, 0

```

```

1246 0000
1247 7200
1250 1134
1251 7640
1252 5250
1253 5646
1254 0000
1255 4246
1256 2134
1257 4446
1260 1312
1261 4175
1262 4455
1263 0130
1264 1313
1265 7777
1266 3125
1267 5654
1270 0000
1271 4254
1272 4446
1273 1157
1274 1321
1275 6001
1276 5670
1277 0000
1300 4254
1301 4446
1302 1157
1303 1314
1304 4446
1305 1311
1306 4575
1307 6001
1310 5677
1311 0000
1312 0000
1313 0000
1314 7200
1315 1711
1316 3024
1317 2511
1320 4475
1321 1712
1322 3526
1323 2512
1324 6036
1325 4510
1326 0000
1327 5531
1330 5347

RRDY, 0
CLA
TAU RBUSY
SEA CLA
JMP :=2
JMP I RRUY
RSTUP, 0
JMS RRUY
ISE RBUSY
SETLOC
RAADR
BLUCKA
MOVE
BLKCNT
RBCTR
-1
UCA ERRCTR
JMP I RSTUP
RBLK, 0
JMS RSTUP
SETLOC
VCTR
RUSRV
IUN
JMP I RUBLK
RDBLK, 0
JMS RSTUP
SETLOC
VCTR
RUSRV
SETLOC
JAK
JBLK
IUN
JMP I RUBLKR
DAK, 0
RAADR, 0
RBCTR, 0
/READER
RUSRV, CLA
SERVICE ROUTINES
TAU I DAK
UCA DELAYS
ISE DAK
JMS I UULMSR
TAU I RAADR
UCA SB
ISE RAADR
KKB
JMS I CHECK
Sb, 0
JMP ERROR
JMP RUUNE

/WAIT FOR HDR NOT BUSY SUB,
/FETCH RBUSY,
/READER BUSY?
/YES, TRY AGAIN
/NO,EXIT
/WAIT FOR RUR NOT BUSY
/SET RBUSY INDICATOR
/SET DATA ADDR
/SET DATA BLOCK LENGTH
/CLEAR ERROR COUNTER
/EXIT,
/HEAD DATA BLOCK, FULL SPEED
/GO DO SETUP
/SET READER SERVICE
/ADDRESS,
/ENABLE INT,
/READ DATA BLOCK,RANDOM STALLS
/GO DO SETUP,
/SET READER SERVICE
/ADDRESS,
/SET DELAY BLOCK ADDRESS,
/ENABLE INT,
/EXIT
/MOVE DELAY WORD TO
/DELAYS,
/UPDATE DAK
/STALL,
/GET EXPECTED CHARACTER
/STORE AT SB
/UPDATE RADUR
/READ CHARACTER
/GO CHECK IT,
/ERROR
/GOOD,

```

1331	ERROR,	UCA ERRCR	/STORE BAD CHARACTER
1332		ISZ ERRCTR	/INCREMENT ERROR COUNTER
1333		JMP .+3	
1334		CLA CMA	/OFLOW, 7777 TO AC
1335		UCA ERRCTR	/RESTORE TO 7777.
1336		LAS	/READ SR
1337		LANU SR5MSK	
1340		SNA CLA	/HALT ON ERROR?(SR5)
1341		JMP RUDDONE	/NO,
1342		TAU ERRCR	/YES, GET BAD CHARACTER
1343		HLT	/ERROR HALT, BAD CHAR IN AC
1344		CLA	
1345		TAU SB	/GOOD CHAR IN AC
1346		HLT RBCTR	/ALL DONE?
1347		ISZ RUDDONE	/NO, TO MAINLINE
1350		JMP I UUUT	/YES,
1351		CLA	/GET C(ERRCTR)
1352		TAU ERRCTR	/ANY ERRORS?
1353		SNA CLA	/NO,
1354		JMP .+3	/YES,
1355		TAU ERRCTR	/NUMBER OF ERRORS IN AC,
1356		HLT	
1357		CLA CLL	/CLEAR RBUSY INDICATOR
1360		UCA RBUSY	
1361		TAU LINK	/RESTORE LINK
1362		RAL	/TO MAINLINE
1363		TAU AC	
1364		JMP I 0	

```

1400      * , 17/+1
1400      DLMSR, 0
1401      CLA CLL
1402      TAU DELAYS
1403      DCA RCTRA
1404      JMP I ,+1
1405      ,+1
1406      TAU MILLI
1407      DCA RCTRB
1410      ISZ RCTRB
1411      JMP ,+1
1412      ISZ RCTRA
1413      JMP ,+7
1414      JMP I DLMSR
1415      RCTRA, 0
1416      RCTRB, 0

1417      DLCNTP, 0
1420      MOVE
1421      BLKCNT
1422      UCTR
1423      -1
1424      SETLOC
1425      UADDR
1426      UBLK
1427      JMS I RANDNO
1430      AND DLYMSK
1431      SNA
1432      JMP GNRND
1433      CIA
1434      DCA I DADDR
1435      ISZ DADDR
1436      ISZ DCTR
1437      JMP GNRND
1440      JMP I DLCNTP
1441      DADDR, 0
1442      UCTR, 0

1400      /GET AND STORE MSEC
1401      /DELAY COUNT
1402      /GET AND STORE
1403      /IMS CONSTANT
1404      /DELAYED 1 MS?
1405      /NO,
1406      /NO,
1407      /YES, DONE DELAYING?
1408      /YES, EXIT
1409      /SUB TO FILL DELAY BLOCK
1410      /SET DELAY BLOCK LENGTH
1411      /UBLK ADDR TO UADDR
1412      /GET RANDOM NUMBER,
1413      /REMOVE EXCESS BITS
1414      /ZERO?
1415      /YES, GET ANOTHER NUMBER
1416      /NO, 2'S COMPLEMENT IT
1417      /STORE IT IN DELAY BLOCK
1418      /UPDATA DELAY BLOCK ADDR,
1419      /BLOCK FULL?
1420      /NO, REPEAT,
1421      /YES, EXIT,

```

```

1443 0000 /PUNCH TEST NORMAL TEST SEQUENCE ROUTINE
1444 4446 NTST, SETLOC /CLEAR RBUSY
1445 0134 RBUSY
1446 0000
1447 1643 TAU I NTST /SELECT PUNCH MODE
1450 3253 DCA NTSTA /PUNCH LEADER
1451 4472 JMS I UPLTLR /PUNCH SYNC CHARACTER
1452 4473 JMS I UPSYNC
1453 0000
1454 4474 JMS I URSYNC /SYNC READER
1455 4502 JMS I URDBLK /READ DATA BLOCK
1456 4472 JMS I UPLTLR /PUNCH TRAILER
1457 4451 JMS I URDRDY /WAIT FOR RDR NOT BUSY
1460 5444 JMP I CHAIN /CHAIN

1461 0000 /PUNCH TESTS SPECIAL TEST SEQUENCE ROUTINE.
1462 4446 NTSTA, SETLOC /CLEAR RBUSY
1463 0134 RBUSY
1464 0000
1465 1661 TAU I STST /SELECT PUNCH MODE
1466 3273 DCA STSTA
1467 1275 TAU STSTA
1470 3276 DCA STSTC
1471 4472 JMS I UPLTLR /PUNCH LEADER
1472 4473 JMS I UPSYNC /PUNCH SYNC CHARACTER
1473 0000
1474 4474 JMS I URSYNC /PUNCH DATA BLOCK
1475 4502 JMS I URDBLK /SYNC READER
1476 0000 JMS I URDRDY /READ DATA BLOCK
1477 5275 JMP STSTB /GO READ AGAIN

1500 0000 /COMBINED TEST NORMAL TEST SEQUENCE
1501 4446 CNTST, SETLOC /CLEAR RBUSY
1502 0134 RBUSY
1503 0000
1504 4472 JMS I UPLTLR /PUNCH LEADER
1505 4473 JMS I UPSYNC /PUNCH SYNC CHARACTER
1506 4500 JMS I UPBLK /PUNCH DATA BLOCK (NO STALLS)
1507 4474 JMS I URSYNC /SYNC READER
1510 4503 JMS I URBLKR /READ DATA BLOCK (STALLS)
1511 4501 JMS I UPBLKR /PUNCH DATA BLOCK (STALLS)
1512 4502 JMS I URDBLK /READ DATA BLOCK (NO STALLS)
1513 4472 JMS I UPLTLR /PUNCH TRAILER
1514 4451 JMS I URDRDY /WAIT FOR READER NOT BUSY
1515 5444 JMP I CHAIN /CHAIN

```

```

1516 0000
1517 7200
1520 3023
1521 1716
1522 3322
1523 2316
1524 4461
1525 0000
1526 4456
1527 5716
1530 0000
1531 4446
1532 1951
1533 7664
1534 4446
1535 1250
1536 4172
1537 1029
1540 7640
1541 4443
1542 1750
1543 4454
1544 2350
1545 2351
1546 5337
1547 5730
1550 0000
1551 0000

```

```

/TYPE LINE OF 3 CHARACTERS (NO DELAY)
TYPLN3, 0

```

```

CLA DELAYM /CLEAR DELAYM
TAU I TYPLN3 /SET AND STORE
UCA :*3 /ADDRESS OF DATA
ISZ TYPLN3
JMS I UFBF3 /GO FILL BUFFER WITH 3 CHARACTERS
0

```

```

JMS I UTYPE /GO TYPE LINE
JMP I TYPLN3 /EXIT
/TYPE LINE OF ASCII PRINTABLE CHARACTERS

```

```

TYPE, 0
JMS I SETCTR /SET TCTR TO =76
TCTH
-114

```

```

JMS I SETCTR /SET FETCH TO ADDRESS
FETCH /OF BLOCKA,

```

```

BLOCKA
TYPEA, TAU DELAYM /GET C(DELAYM)
SEA CLA /0?
JMS I ULYIMS /NO, SO DELAY.
TAU I FETCH /YES, SET CHARACTER
JMS I UPUNCH /GO PRINT CHARACTER
ISZ FETCH /SET UP FOR NEXT CHARACTER
ISZ TCTR /DONE?
JMP TYPEA /NO, REPEAT
JMP I TYPE /YES, EXIT,

```

```

FETCH, 0
TCTR, 0

```

1600
 0000
 1601 1600
 1602 3236
 1603 2200
 1604 1600
 1605 3237
 1606 2200
 1607 1241
 1610 0630
 1611 7112
 1612 7012
 1613 7012
 1614 4225
 1615 2237
 1616 1241
 1617 7040
 1620 0630
 1621 4225
 1622 5600
 1623 0000
 1624 3240
 1625 1240
 1626 7006
 1627 7004
 1630 0242
 1631 1240
 1632 0242
 1633 1245
 1634 3637
 1635 5625
 1636 0000
 1637 0000
 1640 0000
 1641 7100
 1642 0707
 1643 6060

*, 17/+1
 ASCCN: 0
 TAU I ASCCN
 UCA WASC
 ISE ASCCN
 TAU I ASCCN
 UCA SASC
 ISE ASCCN
 TAU K7700
 ANU I WASC
 RTR CLL
 RTR
 RTR
 JMS CNV
 ISE SASC
 TAU K7700
 UCA I WASC
 ANU I WASC
 JMS CNV
 JMP I ASCCN
 0
 DCA ASCT
 TAU ASCT
 RTL
 KAL
 ANU K0707
 TAU ASCT
 ANU K0707
 TAU K6060
 UCA I SASC
 JMP I CNV
 0
 0
 0
 7/00
 K0707, 0/07
 K6060, 0000

 WASC,
 SASC,
 ASCT,
 K7700,
 K0707,
 K6060,

CNV,

1644	0247	A35WP4,	0247	/"
1645	0337		0337	/LEFT ARROW
1646	0327		0327	/"
1647	0257		0257	/"
1650	0247	A35WP6,	0247	/"
1651	0337		0337	/LEFT ARROW
1652	0327		0327	/"
1653	0257		0257	/"
1654	0327		0327	/"
1655	0337		0337	/LEFT ARROW
1656	0247	A33WPS,	0247	/"
1657	0240		0240	/SPACE
1660	0337		0337	/LEFT ARROW
1661	0240		0240	/SPACE
1662	0327		0327	/"
1663	0240		0240	/SPACE
1664	0257		0257	/"
1665	0240		0240	/SPACE
1666	0247	A35WP4,	0247	/
1667	0335		0335	/"
1670	0277		0277	/"
1671	0305		0305	/"
1672	0247	A35WP6,	0247	/"
1673	0335		0335	/"
1674	0277		0277	/"
1675	0305		0305	/"
1676	0277		0277	/"
1677	0335		0335	/"
1700	0247	A35WPS,	0247	/"
1701	0240		0240	/SPACE
1702	0335		0335	/"
1703	0240		0240	/SPACE
1704	0277		0277	/"
1705	0240		0240	/SPACE
1706	0305		0305	/"
1707	0240		0240	/SPACE

1710	0301	A,	0301	/"
1711	0302		0302	/"
1712	0303		0303	/"
1713	0304	D,	0304	/"
1714	0305		0305	/"
1715	0306		0306	/"
1716	0307	G,	0307	/"
1717	0310		0310	/"
1720	0311		0311	/"
1721	0312	J,	0312	/"
1722	0313		0313	/"
1723	0314		0314	/"
1724	0315	M,	0315	/"
1725	0316		0316	/"
1726	0317		0317	/"

1727	0320	P,	520
1730	0321		521
1731	0322		522
1732	0323	S,	523
1733	0324		524
1734	0325		525
1735	0326	V,	526
1736	0327		527
1737	0330		530
1740	0331	Y,	531
1741	0332		532
1742	0260		260
1743	0261	ONE,	261
1744	0262		262
1745	0263		263
1746	0264	FOUR,	264
1747	0265		265
1750	0266		266
1751	0267	SEVEN,	267
1752	0270		270
1753	0271		271
1754	0241	C241,	241
1755	0242		242
1756	0243		243
1757	0244	C244,	244
1760	0245		245
1761	0246		246
1762	0247	C247,	247
1763	0250		250
1764	0251		251
1765	0252	C252,	252
1766	0253		253
1767	0254		254
1770	0255	C255,	255
1771	0256		256
1772	0257		257
1773	0272	C272,	272
1774	0273		273
1775	0274		274
1776	0275	C275,	275
1777	0276		276
2000	0277		277
2001	0300	C300,	300
2002	0333		333
2003	0334		334
2004	0335	C335,	335
2005	0336		336
2006	0337		337

/SLIDING 1 PATTERN

2007	0001	0001
2010	0002	0002
2011	0004	0004
2012	0010	0010
2013	0020	0020
2014	0040	0040
2015	0100	0100
2016	0200	0200
2017	0100	0100
2020	0040	0040
2021	0020	0020
2022	0010	0010
2023	0004	0004
2024	0002	0002
2025	0376	0376
2026	0375	0375
2027	0375	0375
2030	0367	0367
2031	0357	0357
2032	0337	0337
2033	0277	0277
2034	0177	0177
2035	0277	0277
2036	0337	0337
2037	0357	0357
2040	0367	0367
2041	0375	0375
2042	0375	0375

SLID1:

/SLIDING 0 PATTERN

2043	4005	4005
2044	2240	2240
2045	2405	2405
2046	2324	2324
2047	0001	0001
2050	4022	4022
2051	1107	1107
2052	1024	1024
2053	4015	4015
2054	0122	0122
2055	0711	0711
2056	1640	1640
2057	2405	2405
2060	2324	2324
2061	0001	0001

SLID0:

CRTST:	4005	/SPC,C
	2240	/R,SPC
	2405	/T,E
	2324	/S,T
	0001	/END CODE
RMTST:	4022	/SPC,R
	1107	/I,G
	1024	/H,T
	4015	/SPC,M
	0122	/A,R
	0711	/G,I
	1640	/N,SPC
	2405	/T,E
	2324	/S,T
	0001	/END CODE

2062	4025	SPTST,	4025	/SPC,S
2063	2001		2001	/P,A
2064	0305		0305	/C,E
2065	4024		4024	/SPC,T
2066	0525		0525	/E,S
2067	2400		2400	/T
2070	0100		0100	/END CODE
2071	4014	LFTST,	4014	/SPC,L
2072	0640		0640	/P,SPC
2073	2405		2405	/T,E
2074	2324		2324	/S,T
2075	0001		0001	/END CODE
2076	4005	CHRTST,	4005	/SPC,C
2077	1001		1001	/H,A
2100	2201		2201	/R,A
2101	0324		0324	/C,T
2102	0522		0522	/E,R
2103	4024		4024	/SPC,T
2104	0525		0525	/E,S
2105	2425		2425	/T,S
2106	0001		0001	/END CODE
2107	4027	WCPTST,	4027	/SPC,W
2110	1722		1722	/O,R
2111	2324		2324	/O,R
2112	4005		4005	/SPC,C
2113	0125		0125	/A,S
2114	0540		0540	/E,SPC
2115	2001		2001	/P,A
2116	2424		2424	/T,T
2117	0522		0522	/E,R
2120	1640		1640	/N,SPC
2121	2405		2405	/T,E
2122	2324		2324	/S,T
2123	0015		0015	/CR
2124	0012		0012	/LF
2125	0001		0001	/END CODE

2126 0010
2127 0012
2130 4001
2131 2322
2132 6365
2133 5765
2134 6540
2135 1331
2136 0204
2137 4024
2140 0525
2141 2400
2142 1500
2143 1200
2144 0100

KMSG1: 0010 /CR
0012 /LF
4001
2322 /SP,A
6365 /S,R
5765 /S,S
6540 /I,S
1331 /S,SP
0204 /K,Y
4024 /B,O
0525 /SP,T
2400 /E,S
1500 /T
1200 /CH
0100 /LF
/END CODE

/KMSG2, TYPE: PRESS A KEY

2145 0010
2146 0012
2147 4020
2150 2200
2151 2325
2152 4001
2153 4015
2154 0531
2155 5600
2156 1500
2157 1200
2160 0100

KMSG2: 0010 /CR
0012 /LF
4020 /SP,P
2200 /R,E
2325 /S,S
4001 /SP,A
4015 /SP,K
0531 /E,Y
5600 /,
1500 /CH
1200 /LF
0100 /END CODE

2161	0010	KMSG3,	0010	/CR
2162	0012		0012	/LF
2163	4000		4000	/SP,E
2164	0310		0310	/C,H
2165	1740		1740	/U,SP
2166	2400		2400	/T,E
2167	2524		2524	/S,T
2170	0010	KMSG3A,	0010	/CR
2171	0012		0012	/LF
2172	4000		4000	/SP,C
2173	1001		1001	/H,A
2174	2201		2201	/R,A
2175	0324		0324	/C,T
2176	0522		0522	/E,R
2177	2340		2340	/S,P
2200	1500		1500	/K,E
2201	3100		3100	/Y,E
2202	0440		0440	/U,SP
2203	2711		2711	/W,I
2204	1414		1414	/L,L
2205	4002		4002	/SP,B
2206	0540		0540	/E,SP
2207	2431		2431	/T,Y
2210	2000		2000	/P,E
2211	0456		0456	/D,Y
2212	0010		0010	/CR
2213	0012		0012	/LF
2214	4022		4022	/SP,R
2215	2502		2502	/U,B
2216	1720		1720	/O,U
2217	2440		2440	/T,SP
2220	0510		0510	/E,N
2221	0420		0420	/D,S
2222	4022		4022	/SP,R
2223	1720		1720	/O,U
2224	2411		2411	/T,I
2225	1600		1600	/N,E
2226	5000		5000	/.
2227	1500		1500	/CR
2230	1200		1200	/LF
2231	1500		1500	/CR
2232	1200		1200	/LF
2233	0100		0100	/END CODE

2234 0012
 2235 0012
 2236 0012
 2237 0012
 2240 4017
 2241 0524
 2242 0114
 2243 4005
 2244 2125
 2245 1126
 2246 0114
 2247 0516
 2250 2440
 2251 2405
 2252 2324
 2253 0015
 2254 0001

 2255 0015
 2256 0012
 2257 4040
 2260 4040
 2261 0001

KMSG4: 0015
 0012
 0015
 0012
 4017
 0524
 0114
 4005
 2125
 1126
 0114
 0516
 2440
 2405
 2324
 0015
 0001

 KMSG5: 0015
 0012
 OCTEOV: 4040
 0001

/CR
 /LF
 /CR
 /LF
 /SP,0
 /C,T
 /A,L
 /SP,E
 /Q,U
 /I,V
 /A,L
 /E,N
 /T,SR
 /T,E
 /S,T
 /CR
 /END CODE

 /CR
 /LF

 /END CODE

```

2400 * , 17/+1
2401 PRGO,
2402 JMS I USTBF /SET UP BUFFER AREA
2403 JMS I SETCTR /SET KSTART TO INITIAL
2404 KSTART /ROUTINE ADDRESS
2405 POTS0
2406 JMP I ,+1 /GO START PROGRAM
2407 SRSET
2408 /CARRIAGE RETURN TEST
2409 POTS0, 0
2410 POTS1
2411 JMS I UCRLF /CRLF TWICE
2412 -2
2413 JMS I XTYPST /PRINT TEST TITLE
2414 CRTST /CRLF TWICE
2415 JMS I UCRLF
2416 -2
2417 TAU C334 /GET "\ " CODE
2418 JMS I UPUNCH /PRINT IT
2419 TAU M111
2420 UCA UTEMP /-75 TO UTEMP
2421 USE UTEMP /ALL DONE?
2422 SKP /NO
2423 JMP I CHAIN /YES, CHAIN
2424 CRTSTB,
2425 TAU UTEMP /UTEMP TO UTEMP1
2426 UCA UTEMP1 /SET "SPACE" CODE
2427 TAU SPACE /PRINT IT
2428 JMS I UPUNCH /SPACED NO, OF TIMES IN UTEMP1?
2429 USE UTEMP1 /NO, SO SPACE AGAIN
2430 JMP ,=3 /YES, SET "CR" CODE,
2431 TAU CR /PRINT IT,
2432 JMS I UPUNCH /JUMMY CYCLE,
2433 UCA C257 /SET "\ " CODE
2434 JMS I UPUNCH /PRINT IT
2435 TAU CRTSTA /GO TO CRTSTA
2436 JMP CRTSTA
2437
2440

```

```

/RIGHT MARGIN TEST
P0T51, 1
P0T52 /CRLF TWICE
JMS I UCRLF /PRINT TEST TITLE
-2 /CRLF TWICE
JMS I XTYPST /PRINT TEST TITLE
RMTST /CRLF TWICE
JMS I UCRLF /PRINT TEST TITLE
-2 /CRLF TWICE
TAD M16 /-14 TO UTEMP
UCA UTEMP /PRINT TEST TITLE
RMTSTA, JMS I XTYPST /PRINT TEST TITLE
,+2
JMP ,+5
P0P0P
P0P0P
1100 /DONE 14 TIMES?
0100 /NO, SO DO IT AGAIN
JMP RMTSTA /YES, PRINT TEST TITLE
,+2
JMP ,+4
P0P11
P0P00
0100 /CHAIN
JMP I CHAIN /CHAIN

```

```

2441 0001
2442 2473
2443 4453
2444 7776
2445 4450
2446 2050
2447 4453
2450 7776
2451 1144
2452 3116
2453 4450
2454 2456
2455 5262
2456 5555
2457 5555
2460 1100
2461 0100
2462 2116
2463 5253
2464 4450
2465 2467
2466 5272
2467 5511
2470 5500
2471 0100
2472 5444

```



```

2473 0002 /SPACE TEST
2474 2541 POTS2, 2
2475 4453 POTS3
2476 4453 JMS I UCRLF /CRLF TWICE
2477 7776 JMS I XTYPST /PRINT TEST TITLE
2500 2062 SPTST
2501 4453 JMS I UCRLF /CRLF TWICE
2502 7776
2503 1145 TAU M44
2504 3116 UCA UTEMP /-36 TO UTEMP
2505 4450 JMS I XTYPST /PRINT \, SPACE
2506 2510
2507 5312 JMP ,*3
2510 3440
2511 0001
2512 2116 ISE UTEMP
2513 5305 JMP SPTSTA
2514 1145 TAU M44
2515 3116 UCA UTEMP
2516 1142 TAU M1
2517 3117 UCA UTEMP1
2520 1117 TAU UTEMP1
2521 3120 UCA UTEMP2
2522 1131 TAU CR
2523 4454 JMS I UPUNCH
2524 4454 JMS I UPUNCH
2525 1137 TAU SPACE
2526 4454 JMS I UPUNCH
2527 2120 ISE UTEMP2
2530 5325 JMP ,*3
2531 1140 TAU C257
2532 4454 JMS I UPUNCH
2533 2116 ISE UTEMP
2534 7410 SKP I CHAIN
2535 5444 TAU M2
2536 1145 TAU UTEMP1
2537 1117 JMP SPTSTB
2540 5317

```

/LINE FEED TEST

2541 0003
 2542 2600
 2543 4453
 2544 7776
 2545 4450
 2546 2071
 2547 4453
 2550 7776
 2551 1146
 2552 3116
 2553 1141
 2554 4454
 2555 1132
 2556 4454
 2557 2116
 2560 7410
 2561 5444
 2562 4452
 2563 4443
 2564 5353

POTS3, 3

POTS4
 JMS I UCRLF /CRLF TWICE
 -2
 JMS I XTYPST /PRINT TEST TITLE
 LFTST
 JMS I UCRLF /CRLF TWICE
 -2
 TAU M11W
 DCA UTEMP /=72 TO UTEMP
 TAU C334 /GET "\n" CODE
 JMS I UPUNCH /PRINT IT
 TAU LF UPUNCH /GET "LF" CODE
 JMS I UPUNCH /PRINT IT
 USE UTEMP /DONE?
 SKP /NO,
 JMP I CHAIN /YES, CHAIN
 JMS I DLYCNT /GENERATE RANDOM DELAY COUNT
 JMS I DLY1MS /GO DELAY,
 JMP LFTSTA /GO TO LFTSTA

LFTSTA:

```

2600 2600 2600 2600
2601 2601 2601 2601
2602 2602 2602 2602
2603 2603 2603 2603
2604 2604 2604 2604
2605 2605 2605 2605
2606 2606 2606 2606
2607 2607 2607 2607
2610 2610 2610 2610
2611 2611 2611 2611
2612 2612 2612 2612
2613 2613 2613 2613
2614 2614 2614 2614
2615 2615 2615 2615
2616 2616 2616 2616
2617 2617 2617 2617
2620 2620 2620 2620
2621 2621 2621 2621
2622 2622 2622 2622
2623 2623 2623 2623
2624 2624 2624 2624
2625 2625 2625 2625
2626 2626 2626 2626
2627 2627 2627 2627
2630 2630 2630 2630
2631 2631 2631 2631
2632 2632 2632 2632
2633 2633 2633 2633
2634 2634 2634 2634
2635 2635 2635 2635
2636 2636 2636 2636
2637 2637 2637 2637
2640 2640 2640 2640
2641 2641 2641 2641
2642 2642 2642 2642
2643 2643 2643 2643
2644 2644 2644 2644
2645 2645 2645 2645
2646 2646 2646 2646
2647 2647 2647 2647
2650 2650 2650 2650

*, 177+1 4
PØTS4, 4
/TYPE LINE OF CHARACTERS ABC
JMS I UCRLF /CRLF TWICE
2
JMS I XTYPST /PRINT TITLE
CHRTST
JMS I UCRLF /CRLF TWICE
2
JMS I UTPLN3 /PRINT LINE
A
JMP I CHAIN
PØTS5, 5
/TYPE LINE OF CHARACTERS DEF
JMS I UTPLN3
U
JMP I CHAIN
PØTS6, 6
/TYPE LINE OF CHARACTERS GHI
JMS I UTPLN3
V
JMP I CHAIN
PØTS7, 7
/TYPE LINE OF CHARACTERS JKL
JMS I UTPLN3
W
JMP I CHAIN
PØTS10, 10
/TYPE LINE OF CHARACTERS MNO
JMS I UTPLN3
X
JMP I CHAIN
PØTS11, 11
/TYPE LINE OF CHARACTERS POR
JMS I UTPLN3
Y
JMP I CHAIN
PØTS12, 12
/TYPE LINE OF CHARACTERS STU
JMS I UTPLN3
Z
JMP I CHAIN

```

2651	0013	P0TS13, 13
2652	2650	P0TS14
		/TYPE LINE OF CHARACTERS VMX
		JMS I UTPLN3
		V
2653	4460	JMP I CHAIN
2654	1735	
2655	5444	P0TS14, 14
2656	0014	P0TS15
2657	2665	/TYPE LINE OF CHARACTERS YZ0
		JMS I UTPLN3
		Y
2660	4460	JMP I CHAIN
2661	1740	
2662	5444	P0TS15, 15
2663	0015	P0TS16
2664	2670	/TYPE LINE OF CHARACTERS 123
		JMS I UTPLN3
		ONE
2665	4460	JMP I CHAIN
2666	1745	
2667	5444	P0TS16, 16
2670	0016	P0TS17
2671	2675	/TYPE LINE OF CHARACTERS 456
		JMS I UTPLN3
		FOUR
2672	4460	JMP I CHAIN
2673	1746	
2674	5444	P0TS17, 17
2675	0017	P0TS20
2676	2702	/TYPE LINE OF CHARACTERS 789
		JMS I UTPLN3
		SEVEN
2677	4460	JMP I CHAIN
2700	1751	
2701	5444	P0TS20, 20
2702	0020	P0TS21
2703	2707	/TYPE LINE OF CHARACTERS !"#
		JMS I UTPLN3
		C241
2704	4460	JMP I CHAIN
2705	1754	
2706	5444	P0TS21, 21
2707	0021	P0TS22
2710	2714	/TYPE LINE OF CHARACTERS \$%&
		JMS I UTPLN3
		C244
2711	4460	JMP I CHAIN
2712	1757	
2713	5444	P0TS22, 22
2714	0022	P0TS25
2715	2721	/TYPE LINE OF CHARACTERS 1()
		JMS I UTPLN3
		C247
2716	4460	JMP I CHAIN
2717	1762	
2720	5444	

```

2721 0025 P0TS23, 23
2722 2726 /TYPE LINE OF CHARACTERS **,
      JMS I UTPLN3
      G2/2
      JMP I CHAIN
2723 4460 P0TS24, 24
2724 1765 /TYPE LINE OF CHARACTERS -, (
2725 5444 JMS I UTPLN3
2726 0024 G2/2
2727 2735 JMP I CHAIN
      P0TS25
2730 4460 /TYPE LINE OF CHARACTERS :K
2731 1770 JMS I UTPLN3
2732 5444 G2/2
2733 0025 JMP I CHAIN
2734 2740 P0TS26
      JMS I UTPLN3
      G2/2
      JMP I CHAIN
2735 4460 P0TS27
2736 1775 /TYPE LINE OF CHARACTERS =>?
2737 5444 JMS I UTPLN3
2740 0026 G2/2
2741 2745 JMP I CHAIN
      P0TS28
2742 4460 /TYPE LINE OF CHARACTERS @ L\
2743 1776 JMS I UTPLN3
2744 5444 G3/0
2745 0027 JMP I CHAIN
2746 2752 P0TS29
      JMS I UTPLN3
      G3/0
      JMP I CHAIN
2747 4460 P0TS30
2750 2001 /TYPE LINE OF CHARACTERS J+ AND LEFT ARROW
2751 5444 JMS I UTPLN3
2752 0030 G3/0
2753 2757 JMP I CHAIN
      P0TS31
2754 4460 /TYPE LINE OF ALL CHARACTERS
2755 2004 JMS I UPBALL /FILL BUFFER WITH ALL CHARS.
2756 5444 UCA DELAYM /% TO DELAYM.
2757 0031 JMS I UTTYPE /TYPE LINE
2760 2765 JMP I CHAIN /CHAIN
      P0TS32
2761 4462 /TYPE LINE OF ALL CHARACTERS, FIXED DELAY BETWEEN CHARACTERS
2762 3025 JMS I UPBALL /FILL BUFFER WITH ALL CHARS
2763 4456 JMS I DLYCNT /GENERATE DELAY COUNT
2764 5444 JMS I UTTYPE /TYPE LINE
2765 0032 JMP I CHAIN /CHAIN
2766 3000 P0TS33
      JMS I UPBALL
      JMS I DLYCNT
      JMS I UTTYPE
      JMP I CHAIN
2767 4462 /TYPE LINE OF ALL CHARACTERS, FIXED DELAY BETWEEN CHARACTERS
2770 4452 JMS I UPBALL /FILL BUFFER WITH ALL CHARS
2771 4456 JMS I DLYCNT /GENERATE DELAY COUNT
2772 5444 JMS I UTTYPE /TYPE LINE
      JMP I CHAIN /CHAIN

```

3000
 3003
 3001
 3002
 3003
 3004
 3005
 3006
 3007
 3010
 3011
 3012
 3013
 3014
 3015
 3016
 3017
 3020
 3021
 3022
 3023
 3024
 3025
 3026
 3027
 3030
 3031
 3032
 3033
 3034
 3035
 3036
 3037
 3040
 3041
 3042
 3043
 3044
 3045
 3046
 3047
 3050
 3051
 3052
 3053
 3054
 3055
 3056
 3057

```

* 17/*1
P0TS33, 33
P0TS34
/TYPE 6 LINES OF ASR33 WORST CASE PATTERN, NO DELAY,
JMS I UCRLF /CRLF TWICE
-2
JMS I XTYPST /PRINT TITLE
WCPTST
JMS I UFW336 /PATTERN TO BUFFER
DCA DELAYM /0 TO DELAYM
JMS I SETCTR /-6 TO CTRA
CTRA
-6
JMS I UTYPE /TYPE LINE
ISE CTRA /ALL LINES TYPED?
JMP ,=2 /NO, REPEAT
JMP I CHAIN /YES, CHAIN,
P0TS34, 34
P0TS35
/TYPE 6 LINES OF ASR33 WORST CASE PATTERN, FIXED DELAY BETWEEN CHARACTERS
JMS I UFW336 /PATTERN TO BUFFER
JMS I SETCTR /-6 TO CTRA
CTRA
-6
JMS I DLYCNT /GENERATE DELAY COUNT
JMS I UTYPE /TYPE LINE
ISE CTRA /ALL LINES TYPED?
JMP ,=3 /NO, REPEAT
JMP I CHAIN /YES, CHAIN
P0TS35, 35
P0TS36
/TYPE 6 LINES OF ASR35 WORST CASE PATTERN, NO DELAY
JMS I UFW356 /PATTERN TO BUFFER
DCA DELAYM /0 TO DELAYM
JMS I SETCTR /-6 TO CTRA
CTRA
-6
JMS I UTYPE /TYPE LINE
ISE CTRA /ALL LINES TYPED?
JMP ,=2 /NO, REPEAT,
JMP I CHAIN /YES, CHAIN
P0TS36, 36
/TYPE 6 LINES OF ASR35 WORST CASE PATTERN, FIXED DELAY BETWEEN CHARACTERS
JMS I UFW356 /PATTERN TO BUFFER
JMS I SETCTR /-6 TO CTRA
CTRA
-6
JMS I DLYCNT /GENERATE DELAY COUNT
JMS I UTYPE /TYPE LINE
ISE CTRA /ALL LINES TYPED?
JMP ,=3 /NO, REPEAT
JMP I CHAIN /YES, CHAIN
  
```

```

3060 4446
3061 0002
3062 1144
3063 4446
3064 0130
3065 7400
3066 4446
3067 0020
3070 3073
3071 5672
3072 0232

3073 0000
3074 3106
3075 4446
3076 4175
3077 0000
3100 4455
3101 4175
3102 4176
3103 7401
3104 4504
3105 4500

3106 0001
3107 3121
3110 4446
3111 4175
3112 0001
3113 4455
3114 4175
3115 4176
3116 7401
3117 4504
3120 4500

3121 0002
3122 3200
3123 4446
3124 4175
3125 0002
3126 4455
3127 4175
3130 4176
3131 7401
3132 4504
3133 4500

/PROGRAM 1, ASR33/35 PUNCH FUNCTION TEST
/
PRG1, JMS I SETCTR /SET INTERRUPT SERVICE ADDRESS
      2 /TO INTSVC
      INTSVC
      SETLOC /SET DATA BLOCK
      BLKCNT /LENGTH TO
      -400 /-256
      JMS I SETCTR /SET KSTART TO INITIAL
      KSTART /SET ROUTINE ADDRESS.
      PIT0 /GO START PROGRAM
      JMP I ,+1
      SRSET

/ROUTINE 0,
/PUNCH AND READ CHECK BLOCK OF ALL 0'S
PIT0, 0
      PIT1
      SETLOC /0 TO BLOCK A
      BLOCKA /FILL BUFFER
      0
      MOVE
      BLOCKA
      BLOCKA+1
      -3/7
      JMS I UNTST /GO TO NORMAL TEST,
      JMS I UPBLK /USE THIS CALL

/ROUTINE 1
/PUNCH AND READ CHECK BLOCK OF CHANNEL 1 PUNCHES,
PIT1, 1
      PIT2
      SETLOC /1 TO BLOCKA
      BLOCKA
      1
      MOVE /FILL BUFFER
      BLOCKA
      BLOCKA+1
      -3/7
      JMS I UNTST /GO TO NORMAL TEST
      JMS I UPBLK /USE THIS CALL

/ROUTINE 2
/PUNCH AND READ CHECK BLOCK OF CHANNEL 2 PUNCHES
PIT2, 2
      PIT3
      SETLOC /2 TO BLOCKA
      BLOCKA
      2
      MOVE /FILL BUFFER
      BLOCKA
      BLOCKA+1
      -3/7
      JMS I UNTST /GO TO NORMAL TEST
      JMS I UPBLK /USE THIS CALL

```

3200

```

* 17/+1
/ROUTINE 3
/PUNCH AND READ CHECK BLOCK OF CHANNEL 3 PUNCHES
PITS, 3
  PIT4
  SETLOC /4 TO BLOCK A
  BLOCKA 4
  MOVE /FILL BUFFER
  BLOCKA
  BLOCKA+1
  -3/7
  JMS I UNTST /GO TO NORMAL TEST
  JMS I UPBLK /USE THIS CALL
/ROUTINE 4
/PUNCH AND READ CHECK BLOCK OF CHANNEL 4 PUNCHES
PIT4, 4
  PIT5
  SETLOC /10 TO BLOCKA
  BLOCKA 10
  MOVE /FILL BUFFER
  BLOCKA
  BLOCKA+1
  -3/7
  JMS I UNTST /60 TO NORMAL TEST
  JMS I UPBLK /USE THIS CALL

```

```

3200 0000
3201 3213
3202 4446
3203 4175
3204 0004
3205 4452
3206 4175
3207 4176
3210 7401
3211 4504
3212 4500

```

```

3213 0004
3214 3220
3215 4446
3216 4175
3217 0010
3220 4452
3221 4175
3222 4176
3223 7401
3224 4504
3225 4500

```

```

/ROUTINE 5
/PUNCH AND READ CHECK BLOCK OF CHANNEL 5 PUNCHES
PIT5, 5
  PIT6
  SETLOC /10 TO BLOCKA
  BLOCKA 20
  MOVE /FILL BUFFER
  BLOCKA
  BLOCKA+1
  -3/7
  JMS I UNTST /GO TO NORMAL TEST
  JMS I UPBLK /USE THIS CALL

```

```

3226 0002
3227 3241
3230 4446
3231 4175
3232 0020
3233 4452
3234 4175
3235 4176
3236 7401
3237 4504
3240 4500

```



```

3241 0006 /ROUTINE 6
3242 3254 /PUNCH AND READ CHECK BLOCK OF CHANNEL 6 PUNCHES
3243 4446 PIT6,
3244 4175 PIT7
3245 0040 SETLOC /40 TO BLOCKA
3246 4455 BLOCKA
3247 4175 MOVE /FILL BUFFER
3250 4175 BLOCKA
3251 7401 BLOCKA+1
3252 4504 -3/7
3253 4500 JMS I UNTST /GO TO NORMAL TEST
JMS I UPBLK /USE THIS CALL

3254 0007 /ROUTINE 7
3255 3267 /PUNCH AND READ CHECK BLOCK OF CHANNEL 7 PUNCHES
3256 4446 PIT7,
3257 4175 PIT10
3260 0100 SETLOC /100 TO BLOCK A
3261 4455 BLOCKA
3262 4175 MOVE /FILL BUFFER
3263 4176 BLOCKA
3264 7401 BLOCKA+1
3265 4504 -3/7
3266 4500 JMS I UNTST /GO TO NORMAL TEST
JMS I UPBLK /USE THIS CALL

3267 0010 /ROUTINE 10
3270 3302 PIT10,
3271 4446 PIT11
3272 4175 SETLOC /200 TO BLOCK A
3273 0200 BLOCKA
3274 4455 MOVE /FILL BUFFER
3275 4175 BLOCKA
3276 4176 BLOCKA+1
3277 7401 -3/7
3300 4504 JMS I UNTST /GO TO NORMAL TEST
3301 4500 JMS I UPBLK /USE THIS CALL

3302 0011 /ROUTINE 11
3303 3316 PIT11,
3304 4455 PIT12
3305 2007 MOVE /FILL BUFFER WITH
3306 4175 SLIJI /SLIDING 1 PATTERN
3307 7762 BLOCKA
3310 4455 -10
3311 4175 MOVE
3312 4213 BLOCKA
3313 7410 BLOCKA+16
3314 4504 -302
3315 4500 JMS I UNTST /GO TO NORMAL TEST
JMS I UPBLK /USE THIS CALL

```

/ROUTINE 12
/PUNCH AND READ CHECK BLOCK OF SLIDING 0 PATTERN.

PTI12, 12
PTI13
MOVE /FILL BUFFER WITH
SLID0 /SLIDING 0 PATTERN
BLOCKA
-16
MOVE
BLOCKA
BLOCKA+16
-362
JMS I UNTST /GO TO NORMAL TEST
JMS I UPBLK /USE THIS CALL

/ROUTINE 13
/PUNCH AND READ CHECK BLOCK OF ONES AND ZEROES,

PTI13, 13
PTI14
SETLOC /377 TO BLOCK A
BLOCKA
377 /0 TO BLOCKA+1
SETLOC
BLOCKA+1
0
MOVE /FILL BUFFER
BLOCKA
BLOCKA+2
-376
JMS I UNTST /GO TO NORMAL TEST
JMS I UPBLK /USE THIS CALL

/ROUTINE 14
/PUNCH AND READ CHECK BLOCK OF ONES AND ZEROES, RANDOM
/STALLS BETWEEN CHARACTERS PUNCHED,

PTI14, 14
PTI15
SETLOC /377 TO BLOCKA
BLOCKA
377 /0 TO BLOCKA+1
SETLOC
BLOCKA+1
0
MOVE /FILL BUFFER
BLOCKA
BLOCKA+2
-376
JMS I UNTST /GO TO NORMAL TEST
JMS I UPBLK /USE THIS CALL

3316 0012
3317 3332
3320 4452
3321 2022
3322 4172
3323 7762
3324 4452
3325 4172
3326 4212
3327 7416
3330 4504
3331 4500

3332 0012
3333 3350
3334 4446
3335 4172
3336 0377
3337 4446
3340 4176
3341 0000
3342 4452
3343 4172
3344 4177
3345 7402
3346 4504
3347 4500

3350 0014
3351 3400
3352 4446
3353 4172
3354 0377
3355 4446
3356 4176
3357 0000
3360 4452
3361 4172
3362 4177
3363 7402
3364 4504
3365 4501

```

3400 0010
3401 0010
3402 4440
3403 0110
3404 4170
3405 4440
3406 0121
3407 7400
3410 4011
3411 4012
3412 0010
3413 2110
3414 2121
3415 0011
3416 4004
3417 4000

3420 0010
3421 7777
3422 4440
3423 0110
3424 4170
3425 4440
3426 0121
3427 7400
3430 4011
3431 4012
3432 0010
3433 2110
3434 2121
3435 0011
3436 4004
3437 4001

* 17/+1
/ROUTINE 10
/PUNCH AND READ CHECK BLOCK OF BINARY COUNT PATTERN
P1115, 10
P1116
SETLOC /BLOCK A ADDR TO TEMP0
TEMPU
BLUCKA /-206 TO CTRA
SETLOC
CTRA
-400
JMS I INPATT /INITIALIZE B, PATTERN
JMS I GETPT /FILL BUFFER WITH
DCA I TEMP0 /BINARY COUNT PATTERN
ISE TEMP0
ISE CTRA
JMP P1115A
JMS I UNIST /GO TO NORMAL TEST
JMS I UPBLK /USE THIS CALL.

/ROUTINE 10
/PUNCH AND READ CHECK BLOCK OF BINARY COUNT PATTERN
/RANDOM STALLS BETWEEN CHARACTERS PUNCHED.
P1116, 10
P1117
SETLOC /BLOCK A ADDR TO TEMP0
TEMPU
BLUCKA /-206 TO CTRA
SETLOC
CTRA
-400
JMS I INPATT /INITIALIZE B, PATTERN
JMS I GETPT /FILL BUFFER WITH
DCA I TEMP0 /BINARY COUNT PATTERN
ISE TEMP0
ISE CTRA
JMP P1116A
JMS I UNIST /GO TO NORMAL TEST
JMS I UPBLKR /USE THIS CALL

```

```

3440 4446 /PROGRAM 2, KEYBOARD TEST
3441 0020 /SET KSTART TO INITIAL
3442 3447 /ROUTINE ADDRESS
3443 4450 /PRINT
3444 2126
3445 5646
3446 0232

0000
3447 0000
3450 3472 /P2T1
3451 4446 /SETLOC
3452 0121 /KCC
3453 0030 /-1/P0
3454 6032 /CLEAR AC AND FLAG
3455 4450 /MSG1
3456 2145 /MSG2
3457 6031 /READY?
3460 5257 /WAIT
3461 6031 /HEADY, SKIP ON FLAG
3462 5266 /NO SKIP, ERROR
3463 2121 /ALL DONE?
3464 5261 /NO, REPEAT
3465 5444 /YES, CHAIN
3466 7602 /KSF FAILURE
3467 6031 /SCOPE LOOP
3470 5267 /SKIPS ON FLAG
3471 5267 /CONTINUOUSLY

P2E0,
0001
3472 0001 /ROUTINE 1,
3473 3511 /ECHO TEST CHARACTER RECEIVED FROM KEYBOARD IS TYPED, THE
3474 6032 /CHARACTER TYPED SHOULD MATCH CHARACTER KEYED, RUBOUT CHARACTER
3475 4450 /ENDS ROUTINE.
3476 2161
3477 6031
3500 5277
3501 6036
3502 6046
3503 6041
3504 5305
3505 1136
3506 7446
3507 5277
3510 5444

P2T1,
0001
3472 0001 /P2T2
3473 3511 /KCC
3474 6032 /MSG1
3475 4450 /MSG2
3476 2161 /READY?
3477 6031 /WAIT
3500 5277 /READ CHARACTER
3501 6036 /PRINT IT
3502 6046 /PRINTER READY?
3503 6041 /NO, WAIT
3504 5305 /IS IT RUBOUT?
3505 1136 /NO
3506 7446 /YES, CHAIN
3507 5277
3510 5444

```

```

/ROUTINE 2,
/OCTAL EQUIVALENT TEST, THE OCTAL EQUIVALENT OF ANY
/CHARACTER KEYPED IS PRINTED, RUBOUT ENDS ROUTINE.
P2T2,
  2
  7777
  KCC
  JMS I XTYPST
  KMSG4
  JMS I XTYPST
  KMSG3A
  KSF
  JMP ,=1
  KRB
  UCA P2T2M
  JMS I UASCCN
  P2T2M
  UCTEQV
  JMS I XTYPST
  KMSG2
  TAU P2T2M
  TAU MRBOUT
  SEA CLA
  JMP P2T2A
  JMP I CHAIN
  0
P2T2M,
  0

```

```

3511 0002
3512 7777
3513 6032
3514 4450
3515 2234
3516 4450
3517 2170
3520 6031
3521 5320
3522 6036
3523 3336
3524 4506
3525 3536
3526 2257
3527 4450
3530 2255
3531 1336
3532 1135
3533 7640
3534 5320
3535 5444
3536 0000

```

/PROGRAM 3, COMBINED READER, PRINTER, PUNCH TEST.
PRG3, SETLOC /SET INTERRUPT SERVICE
2 /ADDRESS TO INTSVC

INTSVC
SETLOC /SET DATA BLOCK LENGTH
BLKCNT /TO =150
-226

JMS I USTBF /SET UP BUFFER AREA
SETLOC /SET KSTART TO INITIAL
KSTART /ROUTINE ADDRESS

PJT0 JMP I ,*1 /START PROGRAM
SRSET

PJT0, 0
PJT1 JMS I UFBF3 /DATA: ABC
A

PJT1, 1
JMS I UCNTST

PJT2, 2
PJT2 JMS I UFBF3 /DATA: DEF
U

PJT2, 4
JMS I UCNTST

PJT3, 6
PJT3 JMS I UFBF3 /DATA: GHI
G

PJT3, 8
JMS I UCNTST

PJT4, 10
PJT4 JMS I UFBF3 /DATA: JKL
J

PJT4, 12
JMS I UCNTST

PJT5, 14
PJT5 JMS I UFBF3 /DATA: MNO
M

PJT5, 16
JMS I UCNTST

PJT6, 18
PJT6 JMS I UFBF3 /DATA: PQR
P

PJT6, 20
JMS I UCNTST

PJT7, 22
PJT7 JMS I UFBF3 /DATA: STU
S

PJT7, 24
JMS I UCNTST

3537 4446
3540 0002
3541 1144
3542 4446
3543 0130
3544 7552
3545 4457
3546 4446
3547 0020
3550 3559
3551 5752
3552 0232
3553 0000
3554 3560
3555 4461
3556 1710
3557 4500
3560 0001
3561 3560
3562 4461
3563 1710
3564 4500
3565 0002
3566 3572
3567 4461
3570 1710
3571 4500
3572 0003
3573 3577
3574 4461
3575 1721
3576 4500
3577 0004
3600 3604
3601 4461
3602 1724
3603 4500
3604 0000
3605 3611
3606 4461
3607 1727
3610 4500
3611 0000
3612 3610
3613 4461
3614 1732
3615 4500

3616 0007
 3617 3623
 3620 4461
 3621 1732
 3622 4502
 3623 0010
 3624 3630
 3625 4461
 3626 1740
 3627 4502
 3630 0011
 3631 3632
 3632 4461
 3633 1743
 3634 4502
 3635 0012
 3636 3642
 3637 4461
 3640 1746
 3641 4502
 3642 0013
 3643 3647
 3644 4461
 3645 1751
 3646 4502
 3647 0014
 3650 3654
 3651 4461
 3652 1754
 3653 4502
 3654 0012
 3655 3661
 3656 4461
 3657 1757
 3660 4502
 3661 0016
 3662 3660
 3663 4461
 3664 1762
 3665 4502
 3666 0017
 3667 4000
 3670 4461
 3671 1762
 3672 4502

P3T7, / P3T10
 JMS I UFBF3 /DATA: VMX
 V JMS I UCNTST
 P3T10, 10
 P3T11, / P3T11
 JMS I UFBF3 /DATA: YZ0
 Y JMS I UCNTST
 P3T11, 11
 P3T12, / P3T12
 JMS I UFBF3 /DATA: 123
 ONE JMS I UCNTST
 P3T12, 12
 P3T13, / P3T13
 JMS I UFBF3 /DATA: 456
 FOUR JMS I UCNTST
 P3T13, 13
 P3T14, / P3T14
 JMS I UFBF3 /DATA: 789
 SEVEN JMS I UCNTST
 P3T14, 14
 P3T15, / P3T15
 JMS I UFBF3 /DATA: !"#
 C241 JMS I UCNTST
 P3T15, 15
 P3T16, / P3T16
 JMS I UFBF3 /DATA: \$%8
 C244 JMS I UCNTST
 P3T16, 16
 P3T17, / P3T17
 JMS I UFBF3 /DATA: ()
 C247 JMS I UCNTST
 P3T17, 17
 P3T20, / P3T20
 JMS I UFBF3 /DATA: **,
 C252 JMS I UCNTST

4000
 4001
 4002
 4003
 4004
 4005
 4006
 4007
 4010
 4011
 4012
 4013
 4014
 4015
 4016
 4017
 4020
 4021
 4022
 4023
 4024
 4025
 4026
 4027
 4030
 4031
 4032
 4033
 4034
 4035
 4036
 4037
 4040
 4041
 4042
 4043
 4044
 4045
 4046
 4047
 4050
 4051
 4052
 4053
 4054

*. 17/+1
 P3T20, 20
 P3T21, 21
 P3T22, 22
 P3T23, 23
 P3T24, 24
 P3T25, 25
 P3T26, 26
 P3T27, 27
 P3T30, 30
 P3T31, 31

JMS I UFBF3 /DATA: =,/
 C222
 JMS I UCNTST
 JMS I UFBF3 /DATA: !!K
 C272
 JMS I UCNTST
 P3T23 C272 /DATA: ==?
 JMS I UFBF3 /DATA: @L\
 C500
 JMS I UFBF3 /DATA: J+ AND LEFT ARROW
 C555
 JMS I UFBALL /DATA: ALL PRINTABLE ASCII
 C600
 JMS I UFBF3 /DATA: ASR33 PRINTER WORST CASE
 /PATTERN
 C650
 JMS I UFBF3 /DATA: ASR33 PRINTER WORST CASE
 /PATTERN WITH INTERSPERSED BLANKS
 C700
 JMS I UFBF3 /DATA: ASR35 PRINTER WORST CASE
 /PATTERN
 C750
 JMS I UFBF3 /DATA: ASR33 PRINTER WORST CASE
 /PATTERN WITH INTERSPERSED BLANKS

4055 0032
 4056 7777
 4057 4446
 4060 4177
 4061 0377
 4062 4446
 4063 4200
 4064 0000
 4065 4455
 4066 4177
 4067 4201
 4070 7672
 4071 4455
 4072 4177
 4073 4511
 4074 7670
 4075 4505

/DATA: ONE'S AND ZEROES

P3T32: 52
 SETLOC
 BLOCK1
 577
 SETLOC
 BLOCK1+1
 0
 MOVE
 BLOCK1
 BLOCK1+2
 -106
 MOVE
 BLOCK1
 BLOCK2
 -110
 JMS I UCNST

4076 7200
 4077 3025
 4100 4457
 4101 4465
 4102 4456
 4103 5302

/PROGRAM 4: PRINT LINES WITH DATA IN PTEMP AND PTEMPI, NO DELAY,
 PRG4,

CLA DELAY /0 TO DELAY
 JMS I USTBF /FILL BUFFER WITH DATA
 JMS I UFTMP /TYPE LINE
 JMP ,=-1 /REPEAT

4104 4465
 4105 4457
 4106 4452
 4107 4456
 4110 5306

/PROGRAM 5: PRINT LINES WITH DATA IN PTEMP AND PTEMPI, FIXED RANDOM DELAY
 PRG5,

JMS I UFTMP /FILL BUFFER WITH DATA,
 JMS I USTBF /GENERATE DELAY COUNT,
 JMS I DLYCNT /TYPE LINE
 JMS I UTYPE /REPEAT
 JMP ,=2

4111 4446
 4112 0002
 4113 1144
 4114 4446
 4115 0130
 4116 7400
 4117 4455
 4120 0021
 4121 4175
 4122 7776
 4123 4455
 4124 4175
 4125 4177
 4126 7400
 4127 4507
 4130 4500

/PROGRAM 6: PUNCH AND READ CHECK DATA BLOCKS
 /WITH DATA IN PTEMP AND PTEMPI, NO DELAY
 PRG6,
 SETLOC
 2
 INTSVC
 SETLOC
 BLKCNT
 -400
 MOVE
 PTEMP
 BLOCKA
 -2
 MOVE
 BLOCKA
 BLOCKA+2
 -376
 JMS I USTST /GO TO SPECIAL TEST SEQUENCE
 JMS I URBLK /CLSE THIS CALL

/PROGRAM 7, PUNCH AND READ CHECK DATA BLOCKS WITH DATA
 /IN PTEMP AND PTEMP1, RANDOM STALLS BETWEEN CHARS PUNCHED
 PRG7, SETLOC /SET INTERRUPT SERVICE
 2 /ADDRESS TO INTSVC

4131 4446
 4132 0002
 4133 1144
 4134 4446
 4135 0130
 4136 7400
 4137 4452
 4140 0021
 4141 4172
 4142 7776
 4143 4455
 4144 4172
 4145 4177
 4146 7402
 4147 4507
 4150 4501

INTSVC /SET BLOCK LENGTH TO
 SETLOC /-226
 BLKCNT
 -400 /FILL BUFFER WITH DATA IN
 MOVE /PTEMP AND PTEMP1
 PTEMP
 BLOCKA
 -2
 MOVE
 BLOCKA
 BLOCKA+2
 -576
 JMS I USTST /GO TO SPECIAL TEST SEQUENCE
 JMS I UPBLKR /USE THIS CALL,

/PROGRAM 10, PUNCH AND READ CHECK BLOCKS OF BINARY
 /COUNT PATTERN, RANDOM STALLS BETWEEN CHARACTERS PUNCHED
 PRG10, SETLOC /SET INTERRUPT SERVICE
 2 /ADDRESS TO INTSVC

4151 4446
 4152 0002
 4153 1144
 4154 4446
 4155 0130
 4156 7400
 4157 4446
 4160 0112
 4161 4172
 4162 4446
 4163 0121
 4164 7400
 4165 4511
 4166 4512
 4167 3512
 4170 2112
 4171 2121
 4172 5366
 4173 4507
 4174 4501

INTSVC /SET BLOCK LENGTH TO
 SETLOC /-226
 BLKCNT
 -400
 SETLOC
 TEMPU
 BLOCKA
 SETLOC
 UTRA
 -400
 JMS I INPATT /FILL BUFFER WITH BINARY
 JMS I GETPT /COUNT PATTERN
 UCA I TEMPU
 ISZ TEMPU
 ISZ CTRA
 JMP PRG10A
 JMS I USTST /GO TO SPECIAL TEST SEQUENCE
 JMS I UPBLKR /USE THIS CALL

PRG10A,

4175 0212	BLOCKA, 212	/CR
4176 0212	212	/LF
4177 0000	BLOCK1, 0	
4307 0212	*BLOCK1+110	/CR
4310 0212	BLOCKB, 212	/LF
4311 0000	BLOCK2, 0	
4421 0212	*BLOCK2+110	/CR
4422 0212	BLOCKC, 212	/LF
4575 0000	*BLOCKA+400	
5175	DBLK, 0	
	*DBLK+400	

3

THERE ARE NO ERRORS

