

IDENTIFICATION

PRODUCT CODE: MAINDEC-08-DIKLA-C-D

PRODUCT NAME: KL8-JA & KL8-KA/KB/KC/KD
LOOP BACK TEST

PRODUCT DATE: APRIL 1977

MAINTAINER: DIAGNOSTIC GROUP

AUTHORS: BRUCE HANSEN, R. MOORE, DON RICE

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

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

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

COPYRIGHT (C) 1973 & 1977, DIGITAL EQUIPMENT CORP., MAYNARD, MASS.

TABLE OF CONTENTS

1.0	ABSTRACT
2.0	REQUIREMENTS
2.1	HARDWARE
2.2	STORAGE
2.3	PREREQUISITE SOFTWARE
3.0	LOADING PROCEDURE
3.1	METHOD
4.0	STANDARD TEST PROCEDURE
4.1	CONNECTIONS FOR TESTING
4.1.1	EIA LOOP BACK CONNECTIONS
4.1.2	20MA LOOP BACK CONNECTIONS
4.2	RUN CONTROL/DATA TEST
4.2.1	TELETYPE INTERROGATION
4.3	RESTARTING THE PROGRAM
4.4	BAUD RATE TIMING TEST
5.0	OPERATING PROCEDURES
5.1	STARTING ADDRESSES
5.2	SWITCH REGISTER CONTROL
6.0	PROGRAM AND/OR OPERATOR ACTION
6.1	NORMAL HALTS
7.0	ERRORS
7.1	CONTROL/DATA TEST ERRORS
7.1.1	CONTROL/DATA TEST ERROR RECOVERY
7.2	BAUD RATE TIMING TEST ERRORS
7.2.1	BAUD RATE TIMING TEST ERROR RECOVERY
8.0	PROGRAM DESCRIPTION
8.1	CONTROL/DATA TEST
8.2	BAUD RATE TIMING TEST
9.0	APT-8 INTERFACE DEFINITION.
9.1	DESCRIPTION
9.2	SETUP
9.3	APT-8 INTERFACES
9.3.1	TIMING
9.3.2	ERRORS
9.4	LOADING PRECAUTIONS
9.5	MULTIPLE OPTIONS CONTROLLER
10.0	CONSOLE TERMINAL PACKAGE
	ABSTRACT
10.2	INITIALIZATION
10.3	CONTROL G
10.4	CONTROL S
10.5	CONTROL Q
10.6	CONTROL C
10.7	ERROR REPORTING
10.8	END OF PASS
11.0	LISTING

1.0 ABSTRACT

KL8-JA & KL8-KA/KB/KC/KD LOOP BACK TEST IS A PROGRAM TO CHECK OUT THE TERMINAL CONTROL/DATA INTERFACE OPTION (M8655). THE BOARD IS TESTED IN LOOP BACK MODE BY CONNECTING EITHER EIA OUTPUT TO EIA INPUT OR CONNECTING 20 MA CURRENT OUTPUT TO 20 MA CURRENT INPUT.

2.0 REQUIREMENTS

2.1 HARDWARE

PDP-8,8I,8L WITH A DW8E-P OR DW8E-N BUS CONVERTER
PDP-8E,F OR M
KL8-JA TERMINAL CONTROL/DATA INTERFACE (M8655 9 BAUDS RATES) OR A
KL8-KA (SAME AS THE KL8-JA) OR A
KL8-KB TERMINAL CONTROL/DATA INTERFACE (M8655-YA 1050 BAUD) OR A
KL8-KC TERMINAL CONTROL/DATA INTERFACE (M8655-YB 66.7 BAUD) OR A
KL8-KD TERMINAL CONTROL/DATA INTERFACE (M8655-YC 56.8 BAUD)

2.2 STORAGE

THE PROGRAM OCCUPIES MEMORY LOCATIONS 0000 TO 5400

2.3 PREREQUISITE SOFTWARE

THE SYSTEM MUST BE CAPABLE OF RUNNING ALL BASIC PROCESSOR DIAGNOSTICS.

3.0 LOADING PROCEDURE

3.1 METHOD

THE PROGRAM IS LOADED USING THE STANDARD BINARY LOADER TECHNIQUE, AND THE PROGRAM MUST RESIDE IN FIELD 0.

4.0 STANDARD TEST PROCEDURE
.....

4.1 CONNECTIONS FOR TESTING
.....

4.1.1 EIA LOOP BACK CONNECTIONS
.....

CONNECT PIN F TO PIN J AND PIN E TO PIN M ON THE BERG CONNECTOR

4.1.2 20MA LOOP BACK CONNECTIONS
.....

CONNECT PIN E TO PIN H, PIN K TO PIN KK, AND PIN S TO PIN AA ON THE BERG CONNECTOR

////////////////////////////////////
/ WARNING: 20MA LOOP CONNECTIONS CAN ONLY BE CONNECTED /
/ THIS WAY FOR TESTING IN LOOP BACK MODE. DO NOT ATTEMPT /
/ TO CONNECT 2 M9655'S TOGETHER AT ANY TIME WITH 20MA LOOPS /
////////////////////////////////////

4.2 RUN CONTROL/DATA TEST
.....

- A. DO EITHER STEP 4.1.1 OR 4.1.2 FOR EIA OR 20MA LOOP BACK CONNECTIONS
- B. THE PROGRAM WHEN FIRST LOADED IS INITIALIZED FOR 110 BAUD, 2 STOP BITS, 8 DATA BITS, RECEIVE IOT OF 03 AND A TRANSMIT IOT OF 04. IF THIS IS THE CONFIGURATION DESIRED GO TO PARAGRAPH 4.3 (RESTARTING THE PROGRAM) OTHERWISE GO TO STEP C
- C. THE PROGRAM CAN BE INITIALIZED EITHER OF TWO WAYS:
 1. BY WAY OF THE SWITCH REGISTER OR
 2. BY WAY OF AN OPTIONAL TELETYPE WITH DEVICE CODE OF 03 AND 04. THESE DEVICE CODES CANNOT BE CHANGED.
- D. SET SWITCH REGISTER TO 0200 AND PRESS "LOAD ADDRESS".
- E. SET SR11=0 FOR INITIALIZING THE PROGRAM WITH THE SR OR SET SR11=1 FOR INITIALIZING THE PROGRAM WITH THE TELETYPE AND PRESS "CLEAR" AND THEN "CONTINUE".
- F. IF SR11=0 GO TO G. IF SR11=1 GO TO 4.2.1 FOR TELETYPE INTERROGATION.

- G. THE PROGRAM WILL HALT AT LOCATION 4110 WITH 0207 IN THE AC.
- H. SET SR 0-5 TO THE RECEIVE IOT AND SR 6-11 TO THE TRANSMIT IOT AND PRESS "CONTINUE".
- I. THE PROGRAM WILL HALT AT LOCATION 4110 WITH 0212 IN THE AC.
- J. SETUP THE SWITCH REGISTER FOR THE FOLLOWING CONDITION AND PRESS "CONTINUE".

NOTE: A 1 FOLLOWING A JUMPER OR SWITCH MEANS THAT THE JUMPER IS INSERTED OR A SWITCH IS IN THE ON POSITION.

SR0=1 IF PARITY JUMPER IS INSTALLED NP=1

SR1=1 IF STATUS ENABLE JUMPER IS INSTALLED SWD=1

SR2=1 IF FILLER CHARACTER JUMPER INSTALLED FIL=1

SR5	SR6	SR7	SR8	BAUD RATE	ROCKER SWITCHES	JUMPERS	VARIATION
----	----	----	----	----	-----	-----	-----
0	0	0	0	110 BAUD	B1=0 B2=0 B3=0	W2=1 W5=0	KL8-JA & KA
0	0	0	1	150 BAUD	B1=0 B2=0 B3=1	W2=1 W5=0	KL8-JA & KA
0	0	1	0	300 BAUD	B1=0 B2=1 B3=0	W2=1 W5=0	KL8-JA & KA
0	0	1	1	600 BAUD	B1=0 B2=1 B3=1	W2=1 W5=0	KL8-JA & KA
0	1	0	0	1200 BAUD	B1=1 B2=0 B3=0	W2=1 W5=0	KL8-JA & KA
0	1	0	1	2400 BAUD	B1=1 B2=0 B3=1	W2=1 W5=0	KL8-JA & KA
0	1	1	0	4800 BAUD	B1=1 B2=1 B3=0	W2=1 W5=0	KL8-JA & KA
0	1	1	1	9600 BAUD	B1=1 B2=1 B3=1	W2=1 W5=0	KL8-JA & KA
* 1	0	0	0	19.2K BAUD	B1=1 B2=1 B3=1	W2=0 W5=1	KL8-JA & KA
1	0	0	1	56.8 BAUD	B1=0 B2=0 B3=0	W2=1 W5=0	KL8-KD (M8655-YC)
1	0	1	0	66.7 BAUD	B1=0 B2=0 B3=0	W2=1 W5=0	KL8-KC (M8655-YB)
1	0	1	1	1050 BAUD	B1=1 B2=0 B3=0	W2=1 W5=0	KL8-KB (M8655-YA)

*19.2 KIL0 BAUD IS ONLY OBTAINABLE WITH A SPECIAL UART.

SR9=1 IF TWO STOP BITS-JUMPER NOT INSTALLED SB=0

SR10	SR11	#	DATA BITS/CHARACTER	JUMPERS
----	----	----	-----	-----
0	0	5	DATA BITS/CHARACTER	NB1=1 NB2=1
0	1	6	DATA BITS/CHARACTER	NB1=0 NB2=1
1	0	7	DATA BITS/CHARACTER	NB1=1 NB2=0
1	1	8	DATA BITS/CHARACTER	NB1=0 NB2=0

- K. THE PROGRAM WILL HALT AT LOCATION 4110 WITH 0247 IN THE AC.
- L. SET SWITCH REGISTER TO 0000 IF PROCESSOR IS ONE OF THE PDP-8E FAMILY OTHERWISE SET IT TO 0002 AND PRESS "CONTINUE".
- M. SETTING THE SWITCH REGISTER TO 0200 WHILE RUNNING WILL HALT THE PROCESSOR AT THE COMPLETION OF A PROGRAM PASS AT LOCATION 2327
- N. THE PROGRAM WILL NOW RUN UNTIL AN ERROR IS ENCOUNTERED OR THE PROGRAM IS STOPPED BY THE OPERATOR OR SR4=1.

4.2.1 TELETYPE INTERROGATION

NOTE: THIS SECTION OF PROGRAM WAS ENTERED FROM STEPS
D,E AND F OF PARAGRAPH 4.2.

- A. THE PROGRAM WILL TYPE RECEIVE IOT?
THE OPERATOR TYPES IN THE DEVICE CODE OF THE RECEIVER(2 NUMBERS)
- B. THE PROGRAM WILL TYPE TRANSMIT IOT?
THE OPERATOR TYPES IN THE DEVICE CODE OF THE TRANSMITTER(2 NUMBERS)
- C. THE PROGRAM WILL TYPE PARITY(Y OR N)?
IF NP JUMPER IS INSTALLED TYPE Y IF IT ISN'T TYPE N.
THE PROGRAM WILL THEN TYPE NP=1? IF ANSWER WAS YES, AND NP=0?
IF ANSWER WAS NO. NP=THE PARITY JUMPER 1=INSTALLED 0=NOT INSTALLED.
THE PROGRAM WILL THEN TYPE EVEN PARITY EVN=0? ODD PARITY EVN=1?
EVN= ODD OR EVEN PARITY JUMPER, 1= JUMPER INSTALLED 0= NOT INSTALLED.
- D. THE PROGRAM WILL THEN TYPE STATUS ENABLED(Y OR N)?
IF SWD JUMPER IS INSTALLED TYPE Y IF NOT TYPE N.
THE PROGRAM WILL THEN TYPE SWD=1? IF ANSWER WAS YES, AND SWD=0? IF
ANSWER WAS NO. SWD=STATUS WORD ENABLE JUMPER, 1=JUMPER
INSTALLED, 0=JUMPER NOT INSTALLED.
- E. THE PROGRAM WILL THEN TYPE FILLER CHARACTERS(Y OR N)?
IF FIL JUMPER IS INSTALLED TYPE Y IF NOT TYPE N.
THE PROGRAM WILL THEN TYPE FIL=1? IF ANSWER WAS YES, AND
FIL=0? IF ANSWER WAS NO. FIL=FILLER CHARACTER JUMPER,
1= JUMPER INSTALLED AND 0= JUMPER NOT INSTALLED.
- F. THE PROGRAM WILL NOW TYPE OUT THE FOLLOWING MESSAGE
BAUD RATE(00-13)? 00=110 01=150 02=300 03=600 04=1200 05=2400
06=4800 07=9600 10=19,200 11=56.8 12=66.7 13=1050
THE OPERATOR WILL NOW TYPE IN TWO NUMBERS AND THE PROGRAM
WILL RESPOND WITH THE FOLLOWING MESSAGE:

XXXX BAUD = B1=Y? B2=Y? B3=Y W2=Z W5=Z

(XXXX IS THE BAUD RATE FROM 56.8 TO 19,200 BAUD
Y=0 OR 1 0=SWITCH IN OFF POSITION 1= SWITCH IN ON POSITION
Z=0 OR 1 0=JUMPER NOT INSTALLED 1=JUMPER INSTALLED).
- G. THE PROGRAM WILL NOW TYPE TWO STOP BITS(Y OR N)?
IF SB JUMPER IS NOT INSTALLED TYPE Y IF IT IS TYPE N.
THE PROGRAM WILL THEN TYPE SB=0? IF ANSWER WAS YES,
AND SB=1 IF ANSWER WAS NO. SB=STOP BIT JUMPER,
1=JUMPER INSTALLED 0=JUMPER NOT INSTALLED
- H. THE PROGRAM WILL THEN TYPE DATA BITS/CHARACTER? 0=5 1=6 2=7 3=8
THE OPERATOR WILL NOW TYPE IN ONE NUMBER AND THE PROGRAM
WILL RESPOND WITH THE FOLLOWING MESSAGE:
X DATA BITS-NB1=Y?NB2=Y?
X=THE NUMBER OF DATA BITS SELECTED 5,6,7 OR 8
Y=0 OR 1 0=JUMPER NOT INSTALLED 1=JUMPER INSTALLED

- I. THE PROGRAM WILL HALT AT LOCATION 4110 WITH 0247 IN THE AC.
- J. SET SWITCH REGISTER TO 0000 IF PROCESSOR IS ONE OF THE PDP-8E FAMILY OTHERWISE SET IT TO 0002 AND PRESS "CONTINUE".
- K. SETTING THE SWITCH REGISTER TO 0200 WHILE RUNNING WILL HALT THE PROCESSOR AT THE COMPLETION OF A PROGRAM PASS AT 4110 WITH 2147 IN THE AC.
- L. THE PROGRAM WILL NOW RUN UNTIL AN ERROR IS ENCOUNTERED OR THE PROGRAM IS STOPPED BY OPERATOR OR SR4=1.

4.3

RESTARTING THE PROGRAM

- A. SET SR TO 0201 AND PRESS LOAD ADDRESS
- B. SET SR TO ALL ZEROES IF PROCESSOR IS ONE OF THE PDP-8E FAMILY OTHERWISE SET SR TO 0002 AND PRESS "CLEAR" AND THEN "CONTINUE"
- C. SETTING SR4 TO A ONE WILL HALT THE PROGRAM AFTER ONE COMPLETE PROGRAM PASS AT LOCATION 2327
- D. THE PROGRAM WILL NOW RUN UNTIL AN ERROR IS ENCOUNTERED OR THE PROGRAM IS HALTED BY THE OPERATOR OR SR4=1.

4.4

RUN BAUD RATE TIMING TEST

- A. DO STEP A-K OF PARAGRAPH 4.2 IF NOT ALREADY DONE
- B. THIS TEST IS A 30 SECOND STOP WATCH TIMING TEST
- C. SET SR TO 0202 AND PRESS "LOAD ADDRESS" THEN "CLEAR".
- D. CHECK STOP WATCH AND PRESS "CONTINUE".
- E. THE PROGRAM SHOULD HALT IN APPROXIMATELY 30 SECONDS AT LOCATION 4110 WITH 2330 IN THE AC (SEE NOTE FOR EXCEPTION) IF THE BAUD RATE WAS SETUP CORRECTLY.

NOTE: THE PROGRAM WILL HALT IN APPROXIMATELY 28 SECONDS FOR THE FOLLOWING CONDITIONS:

5 DATA BITS, 2 STOP BITS, AND NO PARITY

5.0 OPERATING PROCEDURES

5.1 STARTING ADDRESSES

200 WITH SR11=0 - INITIALIZE THE PROGRAM BY THE SWITCH REGISTER
 200 WITH SR11=1 - INITIALIZE THE PROGRAM BY THE TELETYPE
 201 RESTART ADDRESS=NO INITIALIZATION NEEDED
 202 BAUD RATE TIMING TEST

5.2 SWITCH REGISTER CONTROL

SR	STATE	ACTION
0	1	DO NOT HALT ON ERROR
1	1	LOOP ON ERROR OR ON A CONSTANT DATA PATTERN
2	1	LOOP ON TEST SEQUENCE
4	1	HALT PROGRAM AFTER A COMPLETE PROGRAM PASS
10	1	PROCESSOR NOT OF THE PDP-8E FAMILY
11	0	INITIALIZE THE PROGRAM WITH SR (STARTING ADDRESS 200 ONLY)
11	1	INITIALIZE THE PROGRAM WITH TELETYPE (STARTING ADDRESS 200 ONLY)

6.0 PROGRAM AND/OR OPERATOR ACTION

6.1 NORMAL HALTS

ALL NORMAL HALTS ARE AT 4110 WITH THE ADDRESSES INDICATED BELOW IN THE AC.

0207 INITIALIZATION OF PROGRAM HALT - SET DEVICE CODES IN THE SR.

0212 INITIALIZATION OF PROGRAM HALT - SETUP THE FOLLOWING CONDITIONS
 OF JUMPERS AND SWITCHES IN THE SWITCH REGISTER-PARITY
 STATUS ENABLE, FILLER CHARACTERS, BAUD RATE, NUMBER OF
 STOP BITS, AND NUMBER OF DATA BITS/CHARACTER

0247 SETUP THE SR OPTIONS FOR RUNNING THE PROGRAM

2147 END OF CONTROL/DATA TEST - SR4=1

2330 END OF BAUD RATE TIMING TEST HALT

7.0 ERRORS

7.1 CONTROL/DATA TEST ERRORS

ALL ERRORS DETECTED BY THE PROGRAM WILL RESULT IN AN ERROR HALT.
REFER TO THE PROGRAM LISTING FOR THE CAUSE OF THE ERROR.

7.1.1 CONTROL/DATA TEST ERROR RECOVERY

SET SWITCH REGISTER 0,1 AND 2 TO A 1 AND PRESS "CONTINUE".
THERE MAY BE 1 OR 2 MORE ERROR HALTS, IF THE ERROR WAS A DATA
ERROR, THE PROGRAM IS NOW IN A SCOPE LOOP.

7.2 BAUD RATE TIMING TEST ERRORS

THE OPERATOR MUST DETECT ANY ERRORS IN THE BAUD RATE TIMING TEST.
ONCE STARTED THE PROGRAM SHOULD HALT IN APPROXIMATELY 30 SECONDS EXCEPT
WHEN THE MODULE IS SET UP FOR 5 DATA BITS, 2 STOP BITS AND
NO PARITY. THE PROGRAM WILL THEN HALT IN APPROXIMATELY 28 SECONDS.
ANY DEVIATIONS OF MORE THAN A 1/2 SECOND IS AN ERROR.

7.2.1 BAUD RATE TIMING TEST ERROR RECOVERY

AFTER CHECKING THE MODULE TO BE SET UP CORRECTLY, RESTART
THE TEST BY SETTING SR2=1 AND PRESSING "CONTINUE".

IF ERROR STILL EXISTS GO TO PARAGRAPH 4.4 AND DO EACH AND
EVERY STEP AGAIN.

IF ERROR STILL EXISTS CHECK THE BAUD RATE WITH A SCOPE.

8.0 PROGRAM DESCRIPTION

8.1 CONTROL/DATA TEST

THE FIRST TEST (CLRRD) ISSUES A CAF INSTRUCTION TO GENERATE AN INITIALIZE PULSE. THE PROGRAM THEN CHECKS THAT THE TRANSMIT AND RECEIVE FLAGS ARE NOT STUCK ON AND THAT KSF,TSF, AND SPI DON'T SKIP. THE PROGRAM ALSO CHECKS THAT INTERRUPT REQUEST LINE IS NOT PULLED LOW. ALL ERRORS WILL RESULT IN AN ERROR HALT AT LOCATION 2210. THE CONTENTS OF THE AC WILL CONTAIN THE ADDRESS WHERE THE ERROR WAS DETECTED.

THE NEXT TEST (SCXMIT) CHECKS THAT THE TRANSMIT FLAG CAN BE SET AND CLEARED BY TFL,TSF AND TCF. THE RECEIVE FLAG IS ALSO CHECKED TO BE 0. KCF,TFL,TCF,KSF ARE CHECKED NOT TO SKIP. TSF IS CHECKED TO SKIP AND NOT TO SKIP. ALL ERRORS WILL RESULT IN AN ERROR HALT AT LOCATION 2210 WITH THE AC EQUAL TO THE PC WHERE THE ERROR WAS DETECTED.

THE NEXT TEST (CAFXMT) CHECKS THAT THE TRANSMIT FLAG CAN BE CLEARED BY CAF AND THAT THE RECEIVE FLAG IS STILL 0. ALL ERRORS WILL RESULT IN AN ERROR HALT AT LOCATION 2210 WITH THE AC EQUAL TO THE PC WHERE THE ERROR WAS DETECTED.

THE NEXT TEST (INTXMT) USES THE TRANSMIT FLAG TO CHECK THAT INTERRUPT ENABLE CAN BE SET AND CLEARED AND THAT THE PROGRAM CAN INTERRUPT. INTERRUPT ENABLE IS SET AND CLEARED BY DATA BIT 11 AND THE KIE COMMAND. SPI IS CHECKED TO SKIP AND NOT TO SKIP AND THE PROGRAM ALSO CHECKS THE MODULE TO INTERRUPT AND NOT TO INTERRUPT. AT THE END OF THE TEST THE RECEIVE FLAG IS CHECKED TO BE A 0. ALL ERRORS WILL RESULT IN AN ERROR HALT AT LOCATION 2210 WITH THE CONTENTS OF THE AC EQUAL TO THE PC WHERE THE ERROR WAS DETECTED.

THE NEXT TEST (CAFINT) CHECKS THAT CAF WILL SET INTERRUPT ENABLE BY USING THE TRANSMIT FLAG TO SKIP AND INTERRUPT ON. ALL ERRORS WILL RESULT IN AN ERROR HALT AT LOCATION 2210 WITH THE AC EQUAL TO THE PC WHERE THE ERROR WAS DETECTED.

THE NEXT TEST (ACNSKP) CHECKS THE EFFECT OF THE IOT ON THE AC AND ALSO CHECKS THAT THE IOT'S DO NOT SKIP. TPC AND TLS ARE NOT TESTED. AN ERROR HALT AT LOCATION 2231 INDICATES THAT AN IOT SKIPPED THAT SHOULDN'T, THE AC CONTAINS THE PC WHERE THE ERROR WAS DETECTED. AN ERROR HALT AT LOCATION 2246 INDICATES THAT THE IOT AFFECTED THE CONTENTS OF THE AC. THE CONTENTS OF THE AC EQUALS THE PC WHERE THE ERROR WAS DETECTED. PRESSING CONTINUE WILL RESULT IN AN ERROR HALT AT LOCATION 4110 WITH THE AC EQUAL TO THE BITS THAT WERE EFFECTED BY THE IOT.

THE NEXT TEST (STFLGS) CHECKS THAT THE TRANSMIT FLAG CAN BE SET BY TPC AND THAT SOMETIME AFTER THE TRANSMIT FLAG IS SET THE RECEIVE FLAG WILL GET SET BY DATA AVAILABLE. THE PROGRAM CHECKS THAT FLAGS CAN CAUSE AN INTERRUPT AND NOT TO INTERRUPT BY SETTING AND CLEARING INTERRUPT ENABLE. THE PROGRAM CHECKS THAT TCF AND KCC WILL CLEAR THE FLAGS. ALL ERRORS WILL RESULT IN AN ERROR HALT AT LOCATION 2211 WITH THE AC EQUAL TO THE PC WHERE THE ERROR WAS DETECTED. WHEN LOOPING ON THE ERROR, THE PROGRAM WILL DELAY APPROXIMATELY 200MS AT THE BEGINNING OF EACH LOOP TO ALLOW TIME FOR THE FLAGS TO SETTLE.

THE NEXT TEST (XMTREC) CHECKS THAT A TPC COMMAND WILL SET THE TRANSMIT FLAG AND THAT A TLS COMMAND WILL CLEAR THE FLAG AND THEN RESET IT. THE TEST ALSO CHECKS THAT THE RECEIVE FLAG WILL GET SET FROM THE RESULT OF A TPC AND TLS COMMAND AND THAT THE RECEIVE FLAG CAN BE CLEARED BY A KRB AND KCC OR KCF COMMAND. ALL ERRORS WILL RESULT IN AN ERROR HALT AT LOCATION 2211 WITH THE AC EQUAL TO THE PC WHERE THE ERROR WAS DETECTED. IF SCOPE LOOPING, THERE WILL BE A 200MS DELAY AT THE BEGINNING OF EACH LOOP TO ALLOW THE FLAGS TO SETTLE.

THE NEXT 7 TESTS (SDTST1 TO 7) ARE SIMPLE DATA TESTS. THE PROGRAM TRANSMITS ONE WORD AND THEN WAITS IN A LOOP FOR THE TRANSMIT FLAG OR RECEIVE FLAG TO SET. WHEN THE TRANSMIT FLAG IS SET THE PROGRAM CLEARS IT AND THEN WAITS FOR THE RECEIVE FLAG. WHEN THE RECEIVE FLAG GETS SET, THE PROGRAM COMPARES THE WORD TRANSMITTED WITH THE WORD RECEIVED AND IF THEY DON'T COMPARE THE PROGRAM HALTS AT LOCATION 1366 WITH THE WORD TRANSMITTED IN THE AC. PRESSING "CONTINUE" WILL RESULT WITH AN ERROR HALT AT LOCATION 1371 WITH THE AC EQUAL TO THE WORD READ. ALL OTHER ERRORS WILL RESULT WITH A HALT AT LOCATION 2211 WITH THE AC CONTAINING THE PC WHERE THE ERROR WAS DETECTED

THE NEXT TEST (FDATEAT) IS A FASTER DATA TEST USING RANDOM DATA. THE PROGRAM TRANSMITS THE FIRST WORD AND THEN WAITS IN A LOOP FOR THE TRANSMIT OR RECEIVE FLAG TO SET. WHEN THE TRANSMIT FLAG GETS SET A NEW WORD IS THEN GENERATED AND TRANSMITTED. THE PROGRAM THEN WAITS IN THE LOOP AGAIN FOR THE RECEIVE FLAG TO SET AND THEN DATA IS COMPARED WITH THE FIRST WORD TRANSMITTED. THE DIFFERENCE BETWEEN THIS TEST AND SDTST IS THAT THE PROGRAM IS TRANSMITTING 1 WORD AHEAD OF WHAT IT IS READING. IF AN ERROR OCCURS THE PROGRAM WILL HALT AT LOCATION 1451 WITH THE AC EQUAL TO THE WORD EXPECTED. PRESS "CONTINUE" AND THE PROGRAM WILL HALT AT LOCATION 1454 WITH THE WORD RECEIVED IN THE AC. PRESS "CONTINUE" AGAIN AND THE PROGRAM WILL HALT AT LOCATION 1457 WITH THE AC EQUAL TO THE NEW WORD TRANSMITTED. THIS WORD MAY BE THE SAME AS THE WORD EXPECTED DEPENDING WHERE THE ERROR WAS DETECTED. WHEN SCOPE LOOPING ON THIS ERROR, THE FIRST AND THIRD ERROR HALT WORDS WILL BE THE WORDS USED TO TRANSMIT. WHEN AN ERROR IS ENCOUNTERED DURING THIS SCOPE LOOP, THE PROGRAM DELAYS 200MS TO ALLOW FLAGS TO SETTLE BEFORE TRANSMITTING AGAIN. THERE ARE NO ERROR HALTS IN THE SCOPE LOOP. ALL OTHER ERRORS WILL RESULT WITH AN ERROR HALT AT LOCATION 2211 WITH THE AC CONTAINING THE PC WHERE THE ERROR WAS DETECTED.

THE NEXT TEST (CHARLG) CHECKS THAT THE OPERATOR HAS SELECTED THE CORRECT NUMBER OF DATA BITS. THE PROGRAM TRANSMITS A 377 AND THEN TAKES THE 1'S COMPLEMENT OF THE NUMBER OF DATA BITS THE OPERATOR HAD SET UP THE PROGRAM WITH AND COMPARES IT TO THE WORD READ. IF THE AC EQUALED ZERO AFTER THE COMPARISON, THE NUMBER OF DATA BITS WERE SELECTED CORRECTLY, OTHERWISE, THE PROGRAM WILL HALT AT LOCATION 1632 WITH THE AC CONTAINING THE BITS THAT WEREN'T SUPPOSED TO BE SELECTED. PRESS "CONTINUE" AND THE PROGRAM WILL HALT AT LOCATION 1635 WITH THE AC EQUAL TO THE BITS THE OPERATOR HAD INITIALIZED THE PROGRAM WITH. ALL OTHER ERRORS WILL RESULT WITH AN ERROR HALT AT LOCATION 2211 WITH THE AC CONTAINING THE PC WHERE THE ERROR WAS DETECTED.

THE NEXT TEST (FILERT) IS A FILLER CHARACTER TEST AND WILL ONLY BE DONE IF THE OPERATOR HAS INITIALIZED THE PROGRAM FOR FILLER CHARACTERS. THE PROGRAM TRANSMITS A LINE FEED AND CHECKS THAT 4 RECEIVE FLAGS GET SET BEFORE THE TRANSMIT FLAG AND THAT THE 5TH RECEIVE FLAG GETS SET AFTER THE TRANSMIT FLAG. THE DATA RECEIVED SHOULD BE 1 WORD OF LINE FEED AND 4 WORDS OF FILLER CHARACTERS. IF THE WORD EXPECTED DOESN'T EQUAL THE WORD RECEIVED THE PROGRAM WILL HALT AT LOCATION 1726 WITH THE AC CONTAINING THE WORD EXPECTED. PRESS "CONTINUE" AND THE PROGRAM WILL HALT AT LOCATION 1731 WITH THE AC CONTAINING THE WORD RECEIVED. SCOPE LOOPING ON THIS ERROR WILL RESULT IN A 200MS DELAY AT THE BEGINNING OF EACH ERROR TO ALLOW TIME FOR THE FLAGS TO SETTLE. ALL OTHER ERRORS WILL RESULT IN AN ERROR HALT AT LOCATION 2211 WITH THE AC EQUAL TO THE PC WHERE ERROR WAS DETECTED.

THE LAST TEST (STENAB) IS A STATUS ENABLE TEST AND WILL ONLY BE EXECUTED IF THE OPERATOR HAD SET THE STATUS ENABLE BIT TO A ONE WHEN HE INITIALIZED THE PROGRAM. THIS TEST WILL CHECK THAT THE ERROR BIT AND THE OVERRUN BIT CAN BE SET AND CLEARED IN THE STATUS REGISTER. THE TEST WILL CHECK THAT STATUS ENABLE F/F CAN BE SET AND CLEARED. THE RECEIVE BUFFER WILL BE CHECKED TO CONTAIN THE CORRECT WORD. THREE WORDS (1,2+3) WILL BE TRANSMITTED AND THEN THE RECEIVE BUFFER WILL BE CHECKED. IF THERE WAS AN ERROR DURING COMPARISON OF DATA THE PROGRAM WILL HALT AT LOCATION 2117 WITH THE AC EQUAL TO THE PC WHERE THE ERROR WAS DETECTED. PRESS "CONTINUE" AND THE PROGRAM WILL HALT AT LOCATION 2122 WITH THE WORD EXPECTED IN THE AC. PRESS "CONTINUE" AGAIN AND THE PROGRAM WILL HALT AT LOCATION 2125 WITH THE WORD RECEIVED IN THE AC. ALL OTHER ERRORS WILL RESULT IN AN ERROR HALT AT LOCATION 2211 WITH THE PC WHERE THE ERROR WAS DETECTED IN THE AC. SCOPE LOOPING ON THIS ERROR CAUSES THE PROGRAM TO DELAY 200MS BEFORE STARTING TEST OVER TO ALLOW FLAGS TIME TO SETTLE.

END OF TEST-START TEST OVER AT CLRBRD IF SR4=0
OTHERWISE HALT AT LOCATION 4110 WITH 2147 IN THE AC.

8,2

BAUD RATE TIMING TEST

BAUD RATE TIMING TEST IS A STOP WATCH TIMING TEST. ONCE THIS TEST (BAUDTM) HAS BEEN STARTED, THE PROGRAM TURNS THE INTERRUPT ON AND TRANSMITS A CALCULATED NUMBER OF CHARACTERS (DETERMINED FROM THE BAUD RATE, PARITY, NUMBER OF DATA BITS/CHARACTER AND NUMBER OF STOP BITS). THE PROGRAM SHOULD HALT AT LOCATION 4110 WITH 2330 IN THE AC IN 30 SECONDS. EXCEPTIONS TO THIS ARE: ANY BAUD RATE, NO PARITY, 5 DATA BITS/CHARACTER AND 2 STOP BITS. THE PROGRAM IN THIS CASE SHOULD HALT IN 28 SECONDS.

9.0 APT-8 INTERFACES

9.1 DESCRIPTION

TWO INTERFACES HAVE BEEN PROVIDED WHICH WILL ALLOW THIS DIAGNOSTIC TO RUN UNDER THE STANDARD APT-8 SYSTEM. THESE INTERFACES ARE:

1. TIMING INTERFACE
2. ERROR INTERFACE

EACH WILL BE EXPLAINED IN MORE DETAIL.

9.2 SETUP

IN ORDER TO RUN UNDER APT-8, ADDRESSES 20 AND 22 MUST BE ESTABLISHED PRIOR TO RUNNING THE PROGRAM UNDER APT-8 CONTROL. THE FOLLOWING INFORMATION MUST BE INDICATED:

1. THE SYSTEM IS RUNNING UNDER APT-8 TEST SYSTEM.
2. DEVICE CODES TO BE USED.
3. PARITY JUMPER IS INSTALLED.
4. STATUS ENABLE JUMPER INSTALLED.
5. FILLER CHARACTER JUMPER INSTALLED
6. BAUD RATE.
7. TWO STOP BITS JUMPER NOT INSTALLED.
8. DATA BITS/CHARACTER.

ADDRESS 20 (PSEUDO-SWITCH REGISTER)

THIS ADDRESS WILL CONTAIN THE DEVICE CODES TO USE FOR TRANSMIT AND RECEIVE IN THE FOLLOWING FORMAT:

0304 INDICATES A TRANSMIT IOT OF 03 AND A RECEIVE IOT OF 04.

ADDRESS 21 (HARDWARE CONFIGURATION WORD 1)

THE FOLLOWING IS THE BIT DEFINITION FOR HARDWARE CONFIGURATION WORD 2.

NOTE: A 1 FOLLOWING A JUMPER OR SWITCH INDICATES THAT THE JUMPER IS INSERTED OR THE SWITCH IS IN THE ON POSITION

BIT1=1 IF PARITY JUMPER IS INSTALLED NP=1

BIT7-BIT11 AMOUNT OF MEMORY AVAILABLE IN SYSTEM IN 1K SEGMENTS

ADDRESS 22 (HARDWARE CONFIGURATION WORD 2)

THE FOLLOWING IS THE BIT DEFINITION FOR ADDRESS 22.

BIT0=1 DIAGNOSTIC IS RUNNING ON APT-8

BIT1=1 MULTIPLE OPTION CONTROLLER ENABLED.

BIT2=1 IF STATUS ENABLE JUMPER IS INSTALLED SWD=1

BIT3=1 IF FILLER CHARACTER JUMPER INSTALLED FIL=1

BIT5	BIT6	BIT7	BIT8	BAUD RATE	ROCKER SWITCHES	JUMPERS	VARIATION
0	0	0	0	110	BAUD B1=0 B2=0 B3=0	W2=1 W5=0	KL8-JA & KA
0	0	0	1	150	BAUD B1=0 B2=0 B3=1	W2=1 W5=0	KL8-JA & KA
0	0	1	0	300	BAUD B1=0 B2=1 B3=0	W2=1 W5=0	KL8-JA & KA
0	0	1	1	600	BAUD B1=0 B2=1 B3=1	W2=1 W5=0	KL8-JA & KA
0	1	0	0	1200	BAUD B1=1 B2=0 B3=0	W2=1 W5=0	KL8-JA & KA
0	1	0	1	2400	BAUD B1=1 B2=0 B3=1	W2=1 W5=0	KL8-JA & KA
0	1	1	0	4800	BAUD B1=1 B2=1 B3=0	W2=1 W5=0	KL8-JA & KA
0	1	1	1	9600	BAUD B1=1 B2=1 B3=1	W2=1 W5=0	KL8-JA & KA
* 1	0	0	0	19.2K	BAUD B1=1 B2=1 B3=1	W2=0 W5=1	KL8-JA & KA
1	0	0	1	56.8	BAUD B1=0 B2=0 B3=0	W2=1 W5=0	KL8-KD (M8655-YC)
1	0	1	0	66.7	BAUD B1=0 B2=0 B3=0	W2=1 W5=0	KL8-KC (M8655-YB)
1	0	1	1	1050	BAUD B1=1 B2=0 B3=0	W2=1 W5=0	KL8-KB (M8655-YA)

*19.2 KILO BAUD IS ONLY OBTAINABLE WITH A SPECIAL UART.

BIT9=1 IF TWO STOP BITS-JUMPER NOT INSTALLED SB=0

BIT10	BIT11	# DATA BITS/CHARACTER	JUMPERS
0	0	5	DATA BITS/CHARACTER NB1=1 NB2=1
0	1	6	DATA BITS/CHARACTER NB1=0 NB2=1
1	0	7	DATA BITS/CHARACTER NB1=1 NB2=0
1	1	8	DATA BITS/CHARACTER NB1=0 NB2=0

9.3 APT-8 INTERFACES.

9.3.1 TIMING

APT-8 IS NOTIFIED OF PROGRAM RUN BETWEEN .2 SECONDS AND 2.0 SECONDS. THIS WILL ALLOW THE DIAGNOSTIC TO RUN UNDER THE MUCH SLOWER MOS MEMORY WITHOUT CAUSING APT-8 TO GIVE A TIMEOUT ERROR.

9.3.2 ERRORS

ONLY THE ERROR PC IS REPORTED TO APT-8. THE TYPE OF ERROR CAN BE DETERMINED FROM THE CORRESPONDING ADDRESS IN THE PROGRAM LISTING. THERE IS A POSSIBILITY THAT A TIMEOUT ERROR MAY OCCUR. THIS IS CAUSED BY THE ERROR "HUNG DEVICE". THE PROGRAM WILL HAVE TO BE RERUN IN DUMP MODE IF THIS SHOULD HAPPEN.

9.4 LOADING PRECAUTIONS

THIS PROGRAM SHOULD BE LOADED IN SCRIPT MODE INDICATING TO APT-8 THAT CORE SUMCHECKS ARE TO BE IGNORED.

9.5 MULTIPLE OPTION CONTROLLER

THE DIAGNOSTIC HAS BEEN PROVIDED WITH THE MEANS TO FUNCTION ON THE MULTIPLE OPTION CONTROLLER. TO ENABLE THIS FEATURE HARDWARE CONFIGURATION WORD 2 BIT 1 MUST BE SET TO A ONE. WITH THE MULTIPLE OPTION CONTROLLER ENABLED THE FOLLOWING PSEUDO-SWITCH REGISTER BITS ARE DEFINED:

SR0=0	LIGHT THE FAIL LAMP CORRESPONDING WITH FAILING DEVICE.
SR0=1	GO TO APT PROM ON ERROR
SR6=0	TEST 16 OPTIONS
SR6=1	TEST 8 OPTIONS

NOTE: IT SHOULD BE NOTED THAT ALL OPTIONS MUST BE SET TO DEVICE CODES 03 04. ALL OTHERS CAUSE ERRORS.

10.0 CONSOLE TERMINAL PACKAGE
.....

10.1 ABSTRACT
.....

A CONSOLE TERMINAL PACKAGE HAS BEEN PROVIDED TO ALLOW THIS DIAGNOSTIC TO RUN ON THE CL/8 SYSTEM. THIS ALLOWS THE USER TO CONVERSE TO THE DIAGNOSTIC THROUGH A SOFTWARE CONTROLLED SWITCH REGISTER.

10.2 INITIALIZATION
.....

THE CONSOLE TERMINAL PACKAGE IS INITIALIZED BY MEANS OF ADDRESS 22. TO INDICATE THAT THE CONSOLE TERMINAL PACKAGE IS TO BE USED, BIT 3 MUST BE SET TO A ONE. AT THIS TIME ALL ERROR HALTS AND SWITCH REGISTER FUNCTIONS ARE PASSED TO THE CONSOLE PACKAGE.

10.3 CONTROL G
.....

THIS IS THE CONTROL CHARACTER TO OPEN THE PSEUDO SWITCH REGISTER. WHEN CONTROL G IS TYPED THE PROGRAM IS INTERRUPTED AND SR=XXXX IS TYPED. XXXX IS THE PRESENT CONTENTS OF THE PSEUDO SWITCH REGISTER. THE OPERATOR CAN NOW CHANGE THE SETTING BY ENTERING A NEW SET OF NUMBERS, OR NOT CHANGE IT BY TYPING IN A TERMINATING CHARACTER. WHEN THE PROGRAM RECOGNIZES A CONTROL G IT WILL TYPE AN UP ARROW THEN A G TO SIGNAL THE OPERATOR IT IS RESPONDING TO A CONTROL G.

EXAMPLE:

```
TYPE CONTROL G
^G
SR=XXXX          /PRESENT CONTENTS OF
                  /PSEUDO SWITCH REGISTER.
```

TERMINATING CHARACTERS ARE CARRIAGE RETURN OR LINE FEED. EACH WILL CAUSE A RETURN TO THE PROGRAM AT THE POINT AT WHICH IT WAS INTERRUPTED.

10.3 CONTROL S
.....

THIS SPECIAL CHARACTER STOPS SENDING OUTPUT THE TO TERMINAL DEVICE. IT WILL WAIT FOR A CONTROL Q FOR RESUMPTION OF THE DIAGNOSTIC. THIS CONTROL CHARACTER IS NOT ECHOED.

10.4 CONTROL Q
.....

THIS CHARACTER CAUSES RESUMPTION OF TERMINAL OUTPUT. IT IS USED IN CONJUNCTION WITH CONTROL S. THE CHARACTER IS NOT ECHOED.

10.5 CONTROL C
.....

THIS CHARACTER IS USED TO RETURN BACK TO AN
OPERATING SYSTEM KEYBOARD MONITOR. THE OPERATING SYSTEM
SELECTED FOR THE PDP-8 IS THE OS/8 SYSTEM WITH ITS
BOOTSTRAP IN LOCATION 07600. THE CONTROL CHARACTER IS
ECHOED WHEN RECOGNIZED AND IS REPRESENTED BY AN UPARROW
AND A C.

10.6 ERROR REPORTING.

ALL ERRORS ARE REPORTED TO THE TERMINAL. THE PC OF THE ERROR
IS REPORTED AND THE SWITCH REGISTER CONTENTS INDICATED.
TO CONTINUE FROM THIS POINT TYPE A C,RET. OR CHANGE
THE SWITCH REGISTER ACCORDINGLY.

10.7 END OF PASS

END OF PASS WILL BE INDICATED BY THE FOLLOWING:

DIKLAC END OF PASS XXXX.

XXXX IS THE OCTAL NUMBER FOR THE PASS JUST COMPLETED.

11.0 LISTING



3

4



5

6



```
1 /KL8-JA & KL8-KA/KB/KC/KD LOOP BACK TEST.
2 /MAINDEC=08-DIKLA=C-L
3 /
4 /COPYRIGHT (C) 1973, 1976, 1977 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS., 01754
5 /
6 /PROGRAMMER: REV A ORIGINAL RELEASE - BRUCE HANSEN
7 / REV B KL8-K MODIFICATIONS - R. MOORE
8 / REV C APT-8 INTERFACES - DON RICE
9 /
10 /THE IOT'S ARE USED ON THE APT8A TEST SYSTEM
11 /WHEN MULTIPLE DEVICES ARE TO BE TESTED,
12 /
13 6370 APTIO0=6370 /LOAD THE SELECT COUNTER FROM AC BITS
14 /8-11 AND CLEAR THE AC. THIS IS USED
15 /TO INITIALIZE TO DEVICE ZERO
16 6371 APTIO1=6371 /INCREMENT THE SELECT COUNTER
17 6372 APTIO2=6372 /SET THE FAIL FLAG DESCRIBED BY THE
18 /SELECT COUNTER
19 6373 APTIO3=6373 /CLEAR SELECT COUNTERS AND ALL FLAGS
20 6374 APTIO4=6374 /SKIP IF FAILURE FLAG DESCRIBED
21 /BY THE SELECT COUNTER IS SET.
22 6375 APTIO5=6375 /READ SELECT COUNTERS INTO AC BITS 8-11
23 /
24 /
25 /
26 /RECEIVE IOTS FOR KL8-JA,KA,KB,KC & KD
27 /
28 6007 CAF=6007 /CLEAR ALL FLAGS
29 6030 KCF=6030 /CLEAR RECEIVE FLAG,DON'T SET READER RUN F/F
30 6031 KSF=6031 /SKIP ON RECEIVE FLAG
31 6032 KCC=6032 /CLEAR AC AND RECEIVE FLAG AND SET READER RUN
32 6034 KRS=6034 /READ RECEIVE BUFFER AND STATUS
33 6035 KIE=6035 /AC 11=1 SET INTERRUPT ENABLE
34 /AC 11 = 0 CLEAR INTERRUPT ENABLE F/F
35 6035 KSE=KIE /AC10=1 SET STATUS WORD ENABLE
36 /AC10=0 CLEAR STATUS WORD ENABLE
37 6036 KRB=6036 /CLEAR AC AND RECIEVE FLAG,SET READER RUN AND
38 /READ RECEIVE BUFFER AND STATUS
39 /
40 /
41 /TRANSMIT IOTS FOR KL8-JA,KA,KB,KC & KD
42 /
43 6040 TFL=6040 /SET THE TRANSMIT FLAG
44 6041 TSF=6041 /SKIP ON TRANSMIT FLAG
45 6042 TCF=6042 /CLEAR THE TRANSMIT FLAG
46 6044 TPC=6044 /LOAD TRANSMIT BUFFER AND TRANSMIT
47 6045 SPI=6045 /SKIP IF TRANSMIT OR RECEIVE FLAG IS SET AND
48 /INTERRUPT ENABLE FLIP/FLOP IS SET
49 6046 TLS=6046 /LOAD TRANSMIT BUFFER,TRANSMIT AND CLEAR TRANSMIT FLAG
50 7402 HALT=7402
51 /
52 /
53 /SWITCH REGISTER SETTINGS
54 /SR0=1 DON'T HALT ON ERROR
55 /SP1=1 LOOP ON ERROR OR DATA PATTERN
```

```
56 /SR2=1 LOOP ON TEST SEQUENCE
57 /SR10=1 PROCESSOR NOT A PDP-8E
58 /SR11=0 STARTING ADDRESS 200 ONLY-SETUP PROGRAM LIMITS
59 / BY WAY OF THE SWITCH REGISTER
60 /SR11=1 STARTING ADDRESS 200 ONLY-SETUP PROGRAM LIMITS
61 / BY WAY OF THE TELETYPE(DEVICE CODES OF 03604)
62 /
63 7002 BSW=7002
64 /
65 /
66 /
67 /
68 /
```

```
69
70
71
72      0000      *0
73      0000 0303      303      /REVISION C
74      0001 5001      JMP      1
75      0002 0002      2
76      0003 0003      3
77
78      0011      *11
79      0011 0000      A11,      0      /FORUSE ON APT-8
80      0020      *20
81      /THE FOLLOWING 4 LOCATIONS ARE RESERVED FOR USE BY THE APT 8 TEST SYSTEM.
82      /
83      0020 0000      A20,      0
84      0021 0000      A21,      0
85      0022 0000      A22,      0
86      0023 0000      A23,      0
87      /
88      0024      *24
89      /
90      0024 0000      SAVPNT, 0
91      0025 0002      K2,      2
92      0026 0115      K115,    115
93      0027 0117      K117,    117
94      0030 0033      K33,      33
95      0031 0037      K37,      37
96      0032 0077      C77,      77
97      0033 0177      C177,    177
98      0034 0377      K377,    377
99      0035 0304      DEVCOD, 0304
100     0036 0007      SAVRTS, 0007
101     0037 0004      BITNO,    0004
102     0040 0377      DATBIT, 0377
103     0041 0000      RAUDNO, 0000
104     0042 0000      XMTDAT, 0
105     0043 0000      XMTDT1, 0
106     0044 0000      EPRFLG, 0
107     0045 0000      RECDAT, 0
108     0046 0000      LOOPPC, 0
109     0047 0000      NDELAY, 0
110     0050 0000      CNT1,      0
111     0051 0000      CNT2,      0
112     0052 0000      TSTCNT, 0
113     0053 0000      SAV2,      0
114     0054 7000      M1000,    -1000
115     0055 0252      K0252,    0252
116     0056 0125      K0125,    0125
117     0057 0212      K212,      212
118
119
120
121     0060 4460      LOAD=JMS I .
122     0060 2163      XLOAD
123     0061 4461      DELAY=JMS I .
124     0061 2620      XDELAY
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
```

/SUBROUTINE CALLS

```
124     4462      STLPCC=JMS I .
125     0062 4032      XPCRET
126     4463      LOOP=JMS I .
127     0063 4042      XSR2
128     4464      EHLTLP=JMS I .
129     0064 2200      HLTLOP
130     4465      SW1ONE=JMS I .
131     0065 4050      NOTBE
132     4466      TSFSKP=JMS I .
133     0066 2671      WATTSF
134     4467      KSFSKP=JMS I .
135     0067 2707      WATKSF
136     4470      LISN=JMS I .
137     0070 3314      XLISN
138     4471      MESSAGE=JMS I .
139     0071 3400      MESAGX
140     4472      ONEOCT=JMS I .
141     0072 3201      ONEOCK
142     4473      TWOOCT=JMS I .
143     0073 3211      TWOOCK
144     4474      FOROCT=JMS J .
145     0074 3464      FOROCK
146     4475      PRNT1=JMS I .
147     0075 3513      XPRNT1
148     4476      PRNT2=JMS I .
149     0076 3224      XPRNT2
150     4477      PRNT4=JMS I .
151     0077 3500      XPRNT4
152     4500      SPACE2=JMS J .
153     0100 3306      SPACX2
154     4501      TYPE=JMS I .
155     0101 3235      XTYPE
156     4502      CRLF=JMS I .
157     0102 3244      XCRLF
158     4503      MIOT=JMS I .
159     0103 3442      XMIOT
160     4504      XOR=JMS I .
161     0104 3261      XORS
162     4505      YESRNO=JMS I .
163     0105 3273      YESRNX
164     4506      RANDOM=JMS I .
165     0106 3537      XRAND
166     4507      SAVGEN=JMS I .
167     0107 3555      XSAVGN
168     4510      RESGEN=JMS I .
169     0110 3564      XRESGN
170     4511      BSWAP=JMS I .
171     0111 3522      XBSW
172     4512      AERROR=JMS I .
173     0112 3664      XAERRO
174     4513      APT8=JMS I .
175     0113 3600      XAPT8
176     4514      TICK=JMS I .
177     0114 4013      XTICK
178     4515      CHEK22=JMS I .
```

/APT ERROR REPORTER.

/TEST FOR APT.

```
179 0115 4113 XCHK22
180 0116 4516 NERRDR=JMS I
181 0116 3737 XNERRR
182 0117 4517 LAS=JMS I
183 0117 4063 XLAS
184 0120 4520 HLT=JMS I
185 0120 4072 XHALT
186
187 0121 0000 CLKCNT, 0
188 0122 0000 COUNT, 0
189 0123 4000 K4000, 4000
190 0124 0000 TEMP, 0
```

/TEMP STORAGE FOR APT-8

```
191 0200 *200
192
193 0200 4513 BGNINT, APT8 /TEST FOR APT OR
194 /BEGIN INTERROGATION FOR SETUP
195 0201 5250 NOINTR, JMP START /GO TO START OF TEST NO INTERROGATION REQUIRED
196 0202 5777 JMP BAUDTM /BAUD RATE TIMING TEST (THE PROGRAM SHOULD HALT IN 30 SECONDS)
197 0203 4517 LAS /LOOK AT SR11 FOR DESIRED TYPE OF INTERROGATION
198 0204 7012 RTR /PUT BIT 11 INTO ACO
199 0205 7710 SPA CLA /IF AC11=1 USE TELETYPE FOR INPUT, OTHERWISE USE THE SR
200 0206 5776 JMP TYINTR /GO TO TELETYPE FOR INTERROGATION
201 0207 4520 HLT /SET BITS 0-5 TO THE RECEIVE IOT AND BITS 6-11 TO TRANSMIT IOT
202 0210 4517 LAS /GET DEVICE CODE FROM SWITCH REGISTER
203 0211 3035 DCA DEVCOD /SAVE IT FOR IOT MODIFICATION
204 0212 4520 HLT /SET PARITY-STATUS-FILLER-BAUD RATE-STOP BITS-AND # OF DATA BITS IN SR
205 0213 4517 LAS /GET THE SR AND CALCULATE THE RESULTS
206 0214 3036 DCA SAVBTS /SAVE THEM
207 0215 1036 TAD SAVBTS /SETUP THE NUMBER OF DATA BITS
208 0216 0375 AND (3
209 0217 3037 DCA BITNO
210 0220 1374 TAD (TAD K37
211 0221 1037 TAD BITNO
212 0222 3223 DCA +1
213 0223 4520 HLT/TAD K37+(X)
214 0224 3040 DCA DATBIT /THIS NUMBER=37,77,177,377 FOR 5,6,7 OR 8 DATA BITS
215 0225 1036 TAD SAVBTS /SET UP LENGTH OF CHARACTER FROM # OF STOP BITS
216 0226 0373 AND (4 /IF BIT 9=1 2 STOP BITS, IF 0 ONLY 1 STOP BIT
217 0227 7640 SZA CLA
218 0230 2037 ISZ BITNO /ADD 1 MORE TO CHARACTER LENGTH
219 0231 1036 TAD SAVBTS /DOES IT HAVE PARITY
220 0232 7710 SPA CLA
221 0233 2037 ISZ BITNO /YES BUMP THE CHARACTER LENGTH BY 1
222 0234 1036 TAD SAVBTS /SET UP FOR BAUD RATE
223 0235 7012 RTR
224 0236 7010 RAR
225 0237 0372 AND (17
226 0240 3041 DCA BAUDNO /SAVE THE BAUD NUMBER POINTER
227 0241 1041 TAD BAUDNO /IS THE BAUD NUMBER WITHIN LIMITS
228 0242 1371 TAD (-13
229 0243 7740 SNA SZA CLA
230 0244 5212 JMP SETUP-3 /NO, BAUD NUMBER OUT OF BOUNDS GO BACK TO STATUS SETUP
231 0245 4515 CHEK22 /TEST FOR ACTIVE CONSOLE.
232 0246 4770 JMS XC8PSW /ASK SWITCH REGISTER QUESTION.
233 0247 4520 HLT /SET SR TO DESIRED SWITCH SETTINGS
234 0250 1035 TAD DEVCOD /GET THE DEVICE CODE
235 0251 7012 RTR /PUT THE RECEIVE DEVICE CODE IN BITS 3-8
236 0252 7010 RAR
237 0253 4503 MIOT /GO MODIFY THE IOTS
238 0254 5241 RECPNT /RECEIVE IOT TABLE POINTER
239 0255 1035 TAD DEVCOD /GET THE DEVICE CODES
240 0256 7005 RTL /PUT THE TRANSMIT DEVICE CODE IN BITS 3-8
241 0257 7004 RAL
242 0260 4503 MIOT /GO MODIFY THEM
243 0261 5352 XMTIOT /POINTER TO TRANSMIT IOT TABLE
244 0262 1367 TAD (JMP I 2 /SET UP INTERRUPT RETURN LOCATIONS
245 0263 3001 DCA 1
```

```
246 0264 1366 TAD (INTRET
247 0265 3002 DCA 2
248 0266 5271 JMP CLRBRD /GO START TEST
249
250 0267 7240 INTRET, CLA CMA
251 0270 5400 JMP I 0
252
253
254
255
256
257
258
259 0271 4460 CLRBRD, LOAD /LOAD TIMING VALUE FOR APT IF REQUIRED.
260 0272 4462 STLPPC /SET LOOPING PC FOR TEST AND ERROR LOOPING
261 0273 4465 SW10NE /CHECK SR10 TO SEE IF PROCESSOR A PDP-8E
262 0274 0545 ACM5KP /PROCESSOR NOT A PDP-8E
263 0275 6007 CAF /CLFAR THE BOARD-CHECK THE SKIP IOT'S NOT TO SKIP
264 0276 6031 KSF0, KSF /SKIP ON RECEIVE FLAG
265 0277 7410 SKP
266 0300 4464 EHLTLP /ERROR,RECEIVE FLAG SET OR KSF SKIPPED
267 0301 6041 TSF0, TSF /SKIP ON TRANSMIT FLAG
268 0302 7410 SKP
269 0303 4464 EHLTLP /ERROR,TRANSMIT FLAG SET OR TSF SKIPPED
270 0304 6045 SPI0, SPI /SKIP IF XMIT/RECEIVE FLAG SET WITH INT ENB.
271 0305 7410 SKP
272 0306 4464 EHLTLP /SPI SKIPPED OR XMIT/RECEIVE FLAG SET WITH INT ENB
273 0307 6001 ION /CHECK THAT INT REQ IS NOT PULLED BY INT. ENB.
274 0310 7000 NOP /AND TRANSMIT/RECEIVE FLAG BEING SET
275 0311 6002 IOF
276 0312 7710 SPA CLA
277 0313 4464 EHLTLP /INT REQ LINE PULLED LOW OR TRANSMIT/RECEIVE FLAG SET
278 0314 4516 NERROR
279 0315 0273 CLRBRD+2
280 0316 4463 LOOP /LOOP IF SR2=1
281
282
283
284 0317 4462 SCXMIT, STLPPC /CHECK THAT THE TRANSMIT FLAG CAN BE SET AND CLEARED BY TFL,TSF AND TCF
285 0320 6030 KCF0, KCF /SET LOOPING PC FOR TEST AND ERROR LOOPING
286 0321 7410 SKP /CLEAR RECEIVE FLAG FOR SCOPE LOOPING
287 0322 4464 EHLTLP /SAFETY SKIP IN CASE KCF SKIPPED
288 0323 6040 TFL0, TFL /KCF SKIPPED
289 0324 7410 SKP /SET THE TRANSMIT FLAG
290 0325 4464 EHLTLP /SAFETY SKIP TO CHECK TFL NOT TO SKIP
291 0326 6041 TSF1, TSF /ERROR,TFL SKIPPED
292 0327 4464 EHLTLP /SKIP IF TRANSMIT FLAG=1
293 0330 6042 TCF0, TCF /TRANSMIT FLAG NOT SET OR TFL FAILED
294 0331 7410 SKP /CLEAR THE TRANSMIT FLAG
295 0332 4464 EHLTLP /SAFETY SKIP TO CHECK TCF NOT TO SKIP
296 0333 6041 TSF2, TSF /ERROR, TCF SKIPPED
297 0334 7410 SKP /SKIP ON TRANSMIT FLAG
298 0335 4464 EHLTLP /ERROR,TCF FAILED TO CLEAR TRANSMIT FLAG
299 0336 6031 KSF1, KSF /CHECK TO SEE IF RECEIVE FLAG GOT SET
300 0337 7610 SKP CLA
```

```
301 0340 4464 EHLTLP /RECEIVE FLAG SET BY SETTING TRANSMIT FLAG
302 0341 4516 NERROR
303 0342 0320 SCXMIT+1
304 0343 4463 LOOP /LOOP IF SR2=1
305
306
307
308 0344 4462 CAFXMT, STLPPC /CHECK THAT TRANSMIT FLAG CAN BE CLEARED BY CAF
309 0345 6040 TFL1, TFL /SET LOOPING PC FOR TEST AND ERROR LOOPING
310 0346 6041 TSF3, TSF /SET THE TRANSMIT FLAG
311 0347 4464 EHLTLP /SKIP ON TRANSMIT FLAG
312 0350 6007 CAF /ERROR,TRANSMIT FLAG FAILED TO SET
313 0351 6041 TSF4, TSF /CLEAR ALL FLAGS
314 0352 7410 SKP /SKIP ON TRANSMIT FLAG
315 0353 4464 EHLTLP /OK FLAG NOT SET
316 0354 6031 KSF2, KSF /ERROR,CAF FAILED TO CLEAR TRANSMIT FLAG
317 0355 7610 SKP CLA /CHECK TO SEE IF RECEIVE FLAG GOT SET
318 0356 4464 EHLTLP /NO, IT DIDN'T
319 0357 4516 NERROR /ERROR RECEIVE FLAG GOT SET
320 0360 0345 CAFXMT+1
321 0361 4463 LOOP /LOOP IF SR2=1
322 0362 5765 JMP INTXMT
323
324 0365 0400 /
325 0366 0267
326 0367 5402
327 0370 6703
328 0371 7765
329 0372 0017
330 0373 0004
331 0374 1031
332 0375 0003
333 0376 3000
334 0377 2264 PAGE
335 0400 /
336
337
338
339
340
341 0400 4462 INTXMT, STLPPC /SET LOOPING PC FOR TEST AND ERROR LOOPING
342 0401 6030 KCF1, KCF /CLEAR RECEIVE FLAG FOR SCOPE LOOPING
343 0402 7300 CLA CLL
344 0403 6035 KIE0, KIE /CLEAR INTERRUPT ENABLE
345 0404 7410 SKP /SAFETY SKIP TO CHECK KIE NOT TO SKIP
346 0405 4464 EHLTLP /ERROR, KIE SKIPPED
347 0406 6040 TFL2, TFL /SET THE TRANSMIT FLAG
348 0407 6041 TSF5, TSF /SKIP ON TRANSMIT FLAG
349 0410 4464 EHLTLP /TRANSMIT FLAG FAILED TO SET
350 0411 6045 SPI1, SPI /SKIP IF INTERRUPT ENABLE AND XMIT FLAG SET.
351 0412 7410 SKP /OK, INTERRUPT ENABLE NOT SET
352 0413 4464 EHLTLP /ERROR,INT ENB. SET,FAILED TO BE CLEARED BY KIE
353 0414 6001 ION /CHECK THAT INTERRUPT REQUEST IS NOT PULLED
354 0415 7000 NOP /INTERRUPT HERE IF SET
```



```

355 0416 6002      IOF
356 0417 7710      SPA      CLA
357 0420 4464      EHLTLP
358 0421 7001      IAC
359 0422 6035      KIE1,    KIE
360 0423 7610      SKP      CLA
361 0424 4464      EHLTLP
362 0425 6041      TSF6,    TSF
363 0426 4464      EHLTLP
364 0427 6045      SPI2,    SPI
365 0430 4464      EHLTLP
366 0431 6001      ION
367 0432 7000      NOP
368 0433 6002      IOF
369 0434 7700      SMA      CLA
370 0435 4464      EHLTLP
371 0436 6042      TCF1,    TCF
372 0437 6041      TSF7,    TSF
373 0440 7410      SKP
374 0441 4464      EHLTLP
375 0442 6045      SPI3,    SPI
376 0443 7410      SKP
377 0444 4464      EHLTLP
378 0445 6001      ION
379 0446 7000      NOP
380 0447 6002      IOF
381 0450 7710      SPA      CLA
382 0451 4464      EHLTLP
383 0452 6040      TFL3,    TFL
384 0453 6041      TSF8,    TSF
385 0454 4464      EHLTLP
386 0455 6045      SPI4,    SPI
387 0456 4464      EHLTLP
388 0457 6035      KIE2,    KIE
389 0460 6045      SPI5,    SPI
390 0461 7410      SKP
391 0462 4464      EHLTLP
392 0463 6001      ION
393 0464 7000      NOP
394 0465 6002      IOF
395 0466 7710      SPA      CLA
396 0467 4464      EHLTLP
397 0470 6042      TCF2,    TCF
398 0471 6031      KSF3,    KSF
399 0472 7610      SKP      CLA
400 0473 4464      EHLTLP
401 0474 4516      NERROR
402 0475 0401      INTXMT+1
403 0476 4463      LOOP
404 0477 5300      JMP      CAFINT
405
406
407
408
409

```

/TURN THE INTERRUPT OFF
/PROGRAM INTERRUPTED-CHECK INT. ENB.
/SET INT. ENB. WITH DATA BIT 11 AND KIE
/ERROR, KIE SKIPPED
/SKIP ON TRANSMIT FLAG
/ERROR, TRANSMIT FLAG GOT CLEARED
/SKIP ON INT ENB AND TRANSMIT FLAG
/SPI FAILED TO SKIP OR INT ENB NOT SET
/CHECK THAT INTERRUPT REQUEST IS PULLED
/SHOULD INTERRUPT HERE
/TURN IT OFF
/DID IT INTERRUPT?
/FAILED TO INTERRUPT-CHECK XMIT AND INT ENB
/CLEAR THE TRANSMIT FLAG
/CHECK TO SEE IF IT CLEARED
/IT FAILED TO CLEAR
/SKIP ON INT ENB AND TRANSMIT FLAG
/TRANSMIT FLAG IS GONE IT SHOULDN'T SKIP
/CHECK THAT IT DOESN'T INTERRUPT
/
/
/PROGRAM INTERRUPTED WITHOUT TRANSMIT FLAG
/SET THE FLAG AGAIN
/SKIP ON THE TRANSMIT FLAG
/FLAG FAILED TO SET
/SKIP ON XMIT AND INT. ENB.
/FAILED TO SKIP ON INT ENB AND XMIT FLAG
/CLEAR INTERRUPT ENABLE WITH KIE AND DATA BIT 11
/SKIP IF INT ENB=1 WITH XMIT FLAG
/KIE FAILED TO CLEAR INTERRUPT ENABLE
/CHECK THAT THE PROGRAM DOESN'T INTERRUPT
/
/
/PROGRAM INTERRUPTED WITHOUT INT ENB
/CLEAR TRANSMIT FLAG
/CHECK TO SEE IF RECEIVE FLAG GOT SET
/RECEIVE FLAG SET BY ABOVE CODE
/LOOP ON TEST IF SR2=1

/CHECK THAT CAF WILL SET INTERRUPT ENABLE USING THE TRANSMIT
/FLAG TO SKIP AND INTERRUPT ON.

```

410
411 0500 4462      CAFINT, STLPPC
412 0501 6030      KCF2,    KCF
413 0502 6035      KIE3,    KIE
414 0503 6040      TFL4,    TFL
415 0504 6041      TSF9,    TSF
416 0505 4464      EHLTLP
417 0506 6045      SPI6,    SPI
418 0507 7410      SKP
419 0510 4464      EHLTLP
420 0511 6007      CAF
421 0512 6041      TSF10,   TSF
422 0513 7410      SKP
423 0514 4464      EHLTLP
424 0515 6045      SPI7,    SPI
425 0516 7410      SKP
426 0517 4464      EHLTLP
427 0520 6040      TFL5,    TFL
428 0521 6041      TSF11,   TSF
429 0522 4464      EHLTLP
430 0523 6045      SPI8,    SPI
431 0524 4464      EHLTLP
432 0525 6001      ION
433 0526 7000      NOP
434 0527 6002      IOF
435 0530 7700      SMA      CLA
436 0531 4464      EHLTLP
437 0532 6042      TCF3,    TCF
438 0533 6041      TSF12,   TSF
439 0534 7610      SKP      CLA
440 0535 4464      EHLTLP
441 0536 6035      KIE4,    KIE
442 0537 6031      KSF4,    KSF
443 0540 7610      SKP      CLA
444 0541 4464      EHLTLP
445 0542 4516      NERROR
446 0543 0501      CAFINT+1
447 0544 4463      LOOP
448
449
450
451
452 0545 4462      ACNSKP, STLPPC
453 0546 4465      SW1ONE
454 0547 0577      KSF5=1
455 0550 7240      CLA      CMA
456 0551 6030      KCF
457 0552 7410      SKP
458 0553 4777      JMS      HLTLTP1
459 0554 7040      CMA
460 0555 7440      SZA
461 0556 4776      JMS      HLTLTP2
462 0557 7240      CLA      CMA
463 0560 5775      JMP      KSFS
464

```

/SET THE LOOPING PC FOR TEST AND ERROR LOOPING.
/CLEAR RECEIVE FLAG
/CLEAR INTERRUPT ENABLE
/SET THE TRANSMIT FLAG
/SKIP ON THE TRANSMIT FLAG
/FLAG FAILED TO SET
/SKIP ON TRANSMIT FLAG AND INT ENB.
/SPI SKIPPED OR INT ENB IS SET
/CLEAR TRANSMIT FLAG AND SET INT ENB.
/SKIP IF TRANSMIT FLAG = 1
/CAF FAILED TO CLEAR XMIT FLAG
/SKIP ON TRANSMIT FLAG AND INT ENB.
/SPI SKIPPED WITHOUT TRANSMIT FLAG
/SET THE TRANSMIT FLAG
/SKIP IF XMIT FLAG IS SET
/TRANSMIT FLAG FAILED TO SET
/SKIP ON INTERRUPT ENABLE AND TRANSMIT FLAG
/CAF FAILED TO SET INTERRUPT ENABLE
/CHECK THAT THE PROGRAM WILL INTERRUPT
/GO AND INTERRUPT
/TURN THE INTERRUPT OFF IF IT DIDN'T
/PROGRAM FAILED TO INTERRUPT WITH XMIT AND INT ENB
/CLEAR THE TRANSMIT FLAG
/SKIP IF TRANSMIT FLAG IS SET
/IS FLAG SET
/FLAG FAILED TO CLEAR
/CLEAR INTERRUPT ENABLE
/CHECK TO SEE IF RECEIVE FLAG IS SET
/RECEIVE FLAG GOT SET DURING TEST
/LOOP ON TEST IF SR2=1

/THE FOLLOWING TEST CHECKS THE EFFECT OF THE IOT ON THE AC
/AND ALSO CHECKS THAT THE IOTS DON'T SKIP, TPC AND TLS ARE NOT TESTED.
/SET THE LOOPING PC FOR TEST AND ERROR LOOPING
/CHECK TO SEE IF PROCESSOR A PDP8E
/NOT A PDP-8E GO TO NEXT SUBTEST
/SET AC TO ALL ONE'S
/CLEAR THE RECEIVE FLAG
/ERROR, KCF SKIPPED
/SET THE AC BACK TO ZEROES
/ERROR, KCF CHANGED THE AC
/SET THE AC BACK TO 1'S

```
465 /
466 0575 0600
467 0576 2241
468 0577 2217
      0600
469 /
470 0600 6031 KSF5, KSF /SKIP ON RECEIVE FLAG
471 0601 7410 SKP
472 0602 4777' JMS HLTL1 /ERROR,RECEIVE FLAG SHOULD NOT BE SET
473 0603 7040 CMA /SET THE AC BACK TO ZEROES
474 0604 7440 SZA
475 0605 4776' JMS HLTL2 /ERROR,KSF CHANGED THE AC
476 0606 7240 CLA /SET THE AC TO ALL 1'S
477 0607 6032 KCC0, KCC /CLEAR AC AND RECEIVE FLAG AND SET READER RUN
478 0610 7410 SKP
479 0611 4777' JMS HLTL1 /ERROR, KCC SKIPPED
480 0612 7440 SZA
481 0613 4776' JMS HLTL2 /ERROR,KCC FAILED TO CLEAR AC
482 0614 1375 TAD (-4 /SET AC TO ALL 1'S EXCEPT BITS 10 AND 11
483 0615 6035 KIES, KIE /CLEAR INTERRUPT AND STATUS ENABLE
484 0616 7410 SKP
485 0617 4777' JMS HLTL1 /ERROR, KIE SKIPPED
486 0620 1374 TAD (3 /ADD 3 TO AC AND THEN COMPLEMENT IT
487 0621 7040 CMA
488 0622 7440 SZA
489 0623 4776' JMS HLTL2 /ERROR, KIE CHANGED THE AC
490 0624 7240 CLA /SET THE AC = TO ALL 1'S
491 0625 6034 KRS0, KRS /READ RECEIVE BUFFER STATIC AND STATUS
492 0626 7410 SKP
493 0627 4777' JMS HLTL1 /ERROR, KRS SKIPPED
494 0630 7040 CMA /SET THE AC BACK TO ZEROES
495 0631 7440 SZA
496 0632 4776' JMS HLTL2 /ERROR,KRS CHANGED THE AC
497 0633 1177 TAD (7400 /SET AC BITS 0-3
498 0634 6036 KRB0, KRB /CLEAR AC AND RECEIVE FLAG AND READ RECEIVE BUF
499 0635 7410 SKP
500 0636 4777' JMS HLTL1 /ERROR, KRB SKIPPED
501 0637 0177 AND (7400
502 0640 7440 SZA
503 0641 4776' JMS HLTL2 /KRB FAILED TO CLEAR THE AC
504 0642 4465 SW10NE /CHECK TO SEE IF PDP-8E
505 0643 0653 TCF4=1 /PROCESSOR NOT A PDP-8E GO DO NEXT SUBTEST
506 0644 7240 CLA CMA /SET AC EQUAL TO ALL ONES
507 0645 6040 TFL6, TFL /SET THE TRANSMIT FLAG
508 0646 7410 SKP
509 0647 4777' JMS HLTL1 /ERROR, TFL SKIPPED
510 0650 7040 CMA /SET THE AC BACK TO 0
511 0651 7440 SZA
512 0652 4776' JMS HLTL2 /TFL CHANGED THE AC
513 0653 7240 CLA CMA /SET THE AC TO 1'S
514 0654 6042 TCF4, TCF /CLEAR THE TRANSMIT FLAG
515 0655 7410 SKP
516 0656 4777' JMS HLTL1 /ERROR,TCF SKIPPED
517 0657 7040 CMA /SET THE AC BACK TO 0
518 0660 7440 SZA
```

```
519 0661 4776' JMS HLTL2 /TCF CHANGED THE AC
520 0662 7240 CLA CMA /SET THE AC TO ALL 1'S
521 0663 6041 TSF13, TSF /SKIP ON TRANSMIT FLAG
522 0664 7410 SKP
523 0665 4777' JMS HLTL1 /TRANSMIT FLAG IS SET
524 0666 7040 CMA /SET THE AC BACK TO 0
525 0667 7440 SZA
526 0670 4776' JMS HLTL2 /ERROR,TSF CHANGED THE AC
527 0671 7240 CLA CMA /SET THE AC TO ALL 1'S
528 0672 6045 SPI9, SPI /SKIP IF XMT/REC + INT ENB =1
529 0673 7410 SKP
530 0674 4777' JMS HLTL1 /ERROR,SPI SKIPPED OR XMT/REC AND INT ENB =1
531 0675 7040 CMA /SET THE AC BACK TO ZERO
532 0676 7440 SZA
533 0677 4776' JMS HLTL2 /ERROR,SPI CHANGED THE AC
534 0700 4516 NERROR
535 0701 0546 ACNSKP+1
536 0702 4463 LOOP /LOOP ON TEST IF SR2=1
537
```

```
538
539
540 /START OF LOOP BACK TEST
541 /CHECK THAT THE TRANSMIT FLAG CAN BE SET BY TPC AND THAT DATA
542 /AVAILABLE WILL SET THE RECEIVE FLAG. CHECK THAT THE FLAGS CAN
543 /BE CLEARED BY TCF AND KCC. CHECK THAT THE FLAGS CAN CAUSE AN
544 /INTERRUPT BY MANIPULATING INTERRUPT ENABLE.
545
546 0703 4462 STFLGS, STLPPC /STORE LOOPING PC FOR TEST AND SCOPE LOOPING
547 0704 1176 TAD I=3720 /SET UP A DELAY OF 200MS TO ALLOW FLAGS TO SETTLE
548 0705 3047 DCA NDELAY
549 0706 4461 DELAY
550 0707 6035 KIE6, KIE /WAIT FOR 200MS
551 0710 7610 SKP CLA /CLEAR INTERRUPT ENABLE
552 0711 4464 KCC1, KCC /SAFETY SKIP TO CHECK KIE NOT TO SKIP
553 0712 6032 SKP CLA /ERROR,KIE SKIPPED
554 0713 7610 SKP CLA /CLEAR AC AND RECEIVE FLAG AND SET READER RUN
555 0714 4464 TCF5, TCF /SAFETY SKIP TO CHECK KCC NOT TO SKIP
556 0715 6042 SKP CLA /ERROR,KCC SKIPPED
557 0716 7610 EHLTLP /CLEAR TRANSMIT FLAG
558 0717 4464 KSF6, KSF /SAFETY SKIP TO CHECK TCF NOT TO SKIP
559 0720 6031 SKP CLA /ERROR,TCF SKIPPED
560 0721 7610 SKP CLA /CHECK THE RECEIVE FLAG TO BE ZERO
561 0722 4464 TSF14, TSF /ERROR,RECEIVE FLAG =1 OR KSF SKIPPED
562 0723 6041 SKP CLA /SKIP IF TRANSMIT FLAG =1
563 0724 7610 EHLTLP /ERROR,TRANSMIT FLAG=1 OR TSF SKIPPED
564 0725 4464 TPC0, TPC /LOAD TRANSMIT BUFFER AND TRANSMIT
565 0726 6044 SKP CLA /SAFETY SKIP TO CHECK TPC NOT TO SKIP
566 0727 7610 EHLTLP /ERROR,TPC SKIPPED
567 0730 4464 TSFSKP /WAIT FOR A SECOND FOR TRANSMIT FLAG TO SET
568 0731 4466 EHLTLP /ERROR,TPC FAILED TO SET XMIT FLAG OR TSF FAILED
569 0732 4464 KSF7, KSF /CHECK THE RECEIVE FLAG TO STILL BE A 0
570 0733 6031 SKP CLA
571 0734 7610 EHLTLP /RECEIVE FLAG GOT SET TO SOON
572 0735 4464 SPI10, SPI /SKIP IF XMIT/RECEIVE FLAG=1 AND INT ENB SET
573 0736 6045 SKP CLA
574 0737 7610 EHLTLP /ERROR,SPI SKIPPED OR INTERRUPT ENABLE SET
575 0740 4464 ION /CHECK THE PROGRAM NOT TO INTERRUPT
576 0741 6001 NOP
577 0742 7000 IOF
578 0743 6002 SPA CLA
579 0744 7710 EHLTLP /ERROR,INT ENB SET OR INT REQ PULLED LOW
580 0745 4464 CLA CLL IAC /SET INTERRUPT ENABLE TO A 1
581 0746 7301 KIE7, KIE /AC11=1 AND KIE SET INTERRUPT ENABLE
582 0747 6035 SPI11, SPI /SKIP IF XMIT/RECEIVE FLAG=1 WITH INT ENABLE
583 0750 6045 EHLTLP /INTERRUPT ENABLE FAILED TO SET OR KIE FAILED
584 0751 4464 ION /CHECK THE PROGRAM TO INTERRUPT
585 0752 6001 NOP /IT SHOULD INTERRUPT HERE
586 0753 7000 IOF /TURN IT OFF
587 0754 6002 SMA CLA
588 0755 7700 EHLTLP /ERROR PROGRAM FAILED TO INTERRUPT WITH XMIT AND INT ENABLE
589 0756 4464 KIE8, KIE /SET INT ENB=0 WITH AC11=0 AND KIE COMMAND
590 0757 6035 SPI12, SPI /CHECK THAT INT ENB CLEARED BY KIE
591 0760 6045 SKP CLA
592 0761 7610 EHLTLP /ERROR,INT ENB FAILED TO CLEAR OR SPI SKIPPED
593 0762 4464
```

```
593 0763 6001 ION /CHECK THE PROGRAM NOT TO INTERRUPT
594 0764 7000 NOP
595 0765 6002 IOF
596 0766 7710 SPA CLA
597 0767 4464 EHLTLP /ERROR PROGRAM INTERRUPTED WITHOUT INT ENABLE
598 0770 5773 JMP TSF16
599
600 0773 1000 /
601 0774 0003
602 0775 7774
603 0776 2241
604 0777 2217
605 1000 PAGE
606 1000 6041 TSF16, TSF /CHECK THE TRANSMIT FLAG TO STILL = 1
607 1001 4464 EHLTLP /SOMETHING CLEARED THE TRANSMIT FLAG
608 1002 6042 TCF6, TCF /CLEAR THE TRANSMIT FLAG
609 1003 6041 TSF17, TSF /SKIP IF TRANSMIT FLAG =1
610 1004 7610 SKP CLA
611 1005 4464 EHLTLP /ERROR,TCF FAILED TO CLEAR XMIT FLAG
612 1006 4467 KFSKP /WAIT FOR ABOUT A SECOND FOR RECEIVE FLAG TO SET
613 1007 4464 EHLTLP /ERROR,RECEIVE FLAG=0 OR DATA AVAILABLE FAILED TO SET RECEIVE FLAG
614 1010 6034 KRS1, KRS /CHECK THAT KRS DOESN'T CLEAR RECEIVE FLAG
615 1011 7610 SKP CLA /SAFETY SKIP TO CHECK KRS NOT TO SKIP
616 1012 4464 EHLTLP /ERROR,KRS SKIPPED
617 1013 6031 KSF9, KSF /SKIP ON RECEIVE FLAG
618 1014 4464 EHLTLP /KRS CLEARED RECEIVE FLAG
619 1015 6045 SPI13, SPI /SKIP IF XMIT/RECEIVE FLAG AND INT ENABLE=1
620 1016 7610 SKP CLA
621 1017 4464 EHLTLP /ERROR SPI SKIPPED OR INT ENABLE=1
622 1020 6001 ION /CHECK THE PROGRAM NOT TO INTERRUPT
623 1021 7000 NOP
624 1022 6002 IOF
625 1023 7710 SPA CLA
626 1024 4464 EHLTLP /PROGRAM INTERRUPT WITHOUT INTERRUPT ENABLE
627 1025 7301 KIE9, KIE /SET INT ENB F/F=1
628 1026 6035 SPI14, SPI /SKIP IF RECEIVE AND INT ENB=1
629 1027 6045 EHLTLP /ERROR,SPI FAILED OR RECEIVE/INT ENB NOT= TO A 1
630 1030 4464 ION /CHECK THE PROGRAM TO INTERRUPT
631 1031 6001 NOP
632 1032 7000 IOF
633 1033 6002 SMA CLA
634 1034 7700 EHLTLP
635 1035 4464 KCC2, KCC /ERROR,FAILED TO INTERRUPT WITH INT ENB AND RECEIVE FLAG = A 1
636 1036 6032 KSF10, KSF /CLEAR AC AND RECEIVE FLAG AND SET READER RUN
637 1037 6031 SKP CLA /SKIP IF RECEIVE FLAG =1
638 1040 7610 EHLTLP /ERROR,KCC FAILED TO CLEAR RECEIVE FLAG
639 1041 4464 SPI15, SPI /SKIP IF INT ENB AND RECEIVE FLAG =1
640 1042 6045 SKP CLA
641 1043 7610 EHLTLP /ERROR,SPI SKIPPED WITHOUT RECEIVE FLAG = 1
642 1044 4464 ION /CHECK THE PROGRAM NOT TO INTERRUPT
643 1045 6001 NOP
644 1046 7000 IOF
645 1047 6002 SPA CLA
646 1050 7710
```

```
647 1051 4464 EHLTLP /INTERRUPTED WITHOUT RECEIVE FLAG SET
648 1052 6035 KIE10, KIE /CLEAR INTERRUPT ENABLE
649 1053 7300 CLA CLL
650 1054 4516 NERROR
651 1055 0707 STFLGS+4
652 1056 4463 LOOP /LOOP ON TEST IF SR2=1
653 /
654 /THE FOLLOWING TEST CHECKS THAT A TPC COMMAND WILL SET THE
655 /TRANSMIT FLAG AND THAT A TLS WILL CLEAR THE FLAG AND THEN RESET
656 /IT. CHECK THAT THE RECEIVE FLAG WILL GET SET FROM A TPC AND TLS
657 /COMMAND AND THAT IT CAN BE CLEARED BY A KRB AND KCC OR KCF COMMAND
658
659 1057 4462 XMTREC, STLPPC /STORE LOOPING PC FOR TEST AND ERROR LOOPING
660 1060 1176 TAD [-3720 /DELAY 200MS FOR SCOPE LOOPING TO LET
661 1061 3047 DCA NDELAY /FLAGS SETTLE
662 1062 4461 DELAY /GO DELAY 200MS
663 1063 6032 KCC3, KCC /CLEAR AC AND RECEIVE FLAG
664 1064 6042 TCF7, TCF /CLEAR THE TRANSMIT FLAG
665 1065 6031 KSF11, KSF /CHECK THE RECEIVE FLAG TO BE 0
666 1066 7610 SKP CLA
667 1067 4464 EHLTLP /RECEIVE FLAG STILL = 1 AFTER A KCC COMMAND
668 1070 6041 TSF18, TSF /SKIP IF TRANSMIT FLAG = 1
669 1071 7610 SKP CLA
670 1072 4464 EHLTLP /TRANSMIT FLAG STILL A 1 AFTER A TCF COMMAND
671 1073 6044 TPC1, TPC /LOAD TRANSMIT BUFFER AND TRANSMIT
672 1074 7200 CLA
673 1075 4466 TSFSKP /WAIT FOR THE FLAG TO SET
674 1076 4464 EHLTLP /ERROR XMIT FLAG FAILED TO SET BY TPC
675 1077 6031 KSF12, KSF /CHECK THE RECEIVE FLAG TO STILL BE 0
676 1100 7610 SKP CLA
677 1101 4464 EHLTLP /RECEIVE FLAG SET TO SOON
678 1102 6046 TLS0, TLS /LOAD TRANSMIT BUFFER AND TRANSMIT AND CLEAR FLAG
679 1103 7200 CLA
680 1104 7610 SKP CLA
681 1105 4464 EHLTLP /SAFETY SKIP TO CHECK TLS NOT TO SKIP
682 1106 6041 TSF20, TSF /ERROR, TLS SKIPPED
683 1107 7610 SKP CLA /SKIP IF TRANSMIT FLAG = 1
684 1110 4464 EHLTLP /ERROR, TLS FAILED TO CLEAR TRANSMIT FLAG
685 1111 4467 KSF8, KSF /WAIT FOR RECEIVE FLAG TO SET FROM FIRST XMIT
686 1112 4464 EHLTLP /ERROR, REC FLAG FAILED TO SET FROM FIRST XMIT
687 1113 6036 KRB1, KRB /CLEAR AC AND RECEIVE FLAG AND READ RECEIVE BUFF
688 1114 7610 SKP CLA /SAFETY SKIP TO CHECK KRB NOT TO SKIP
689 1115 4464 EHLTLP /ERROR, KRB SKIPPED
690 1116 6031 KSF14, KSF /SKIP ON RECEIVE FLAG
691 1117 7610 SKP CLA
692 1120 4464 EHLTLP /ERROR, KRB FAILED TO CLEAR RECEIVE FLAG
693 1121 4466 TSFSKP /WAIT FOR TRANSMIT FLAG TO SET FROM 2ND XMIT
694 1122 4464 EHLTLP /TRANSMIT FLAG FAILED TO SET FROM TLS COMMAND
695 1123 6042 TCF8, TCF /CLEAR THE TRANSMIT FLAG
696 1124 6041 TSF22, TSF /SKIP IF TRANSMIT FLAG SET
697 1125 7610 SKP CLA
698 1126 4464 EHLTLP /ERROR, TCF FAILED TO CLEAR FLAG
699 1127 4467 KSF8, KSF /WAIT FOR RECEIVE FLAG TO SET FROM TLS COMMAND
700 1130 4464 EHLTLP /ERROR, RECEIVE FLAG FAILED TO SET FROM 2ND TRANSMIT
701 1131 4517 LAS /CHECK TO SEE IF PDP-8E
```

```
702 1132 7010 RAR
703 1133 7012 PTR
704 1134 7700 SMA CLA
705 1135 5343 JMP +6 /PDP8E
706 1136 6032 KCC4, KCC /CLEAR THE AC AND RECEIVE FLAG
707 1137 6031 KSF16, KSF /SKIP IF RECEIVE FLAG = 1
708 1140 7610 SKP CLA
709 1141 4464 EHLTLP /ERROR, KCC FAILED TO CLEAR RECEIVE FLAG
710 1142 5347 JMP +5
711 1143 6030 KCF4, KCF /CLEAR THE RECEIVE FLAG
712 1144 6031 KSF17, KSF /SKIP IF RECEIVE FLAG SET
713 1145 7610 SKP CLA
714 1146 4464 EHLTLP /ERROR, KCF FAILED TO CLEAR FLAG
715 1147 4516 NERROR
716 1150 1063 XMTREC+4
717 1151 4463 LOOP /LOOP IF SR2=1
718 /
719 /START OF DATA TEST-TRANSMIT 1 WORD AND THEN WAIT FOR THE
720 /RECEIVE FLAG TO SET
721 /
722 /DATA TEST 1 - TRANSMIT A ROTATING BIT AND CHECK THAT IT CAME BACK
723
724 1152 4462 SDTST1, STLPPC /STORE LOOPING PC FOR TEST LOOPING
725 1153 3042 DCA XMTDAT
726 1154 4460 LOAD /LOAD TIMING FOR APT IF REQUIRED.
727 1155 7120 CLL CML /SET LINK
728 1156 1042 TAD XMTDAT
729 1157 7004 RAL /AND SHIFT LEFT ONE
730 1160 3042 DCA XMTDAT /SET TRANSMIT WORD
731 1161 4777 JMS SLWDAT /GO TRANSMIT AND RECEIVE 1 WORD
732 1162 4516 NERROR
733 1163 1156 +5
734 1164 4463 LOOP /LOOP IF SR2=1
735 1165 5766 JMP I +1
736 1166 1200 SDTST2
737 /
738 /
739 1177 1333 PAGE
740 1200 /
741 /
742 /DATA TEST 2 - TRANSMIT ALL ONES AND CHECK THAT 1'S COME BACK
743 1200 4462 SDTST2, STLPPC /STORE LOOPING PC FOR TEST LOOPING
744 1201 7240 CLA CMA
745 1202 0040 AND DATBIT /MASK OUT FOR THE NUMBER OF DATA BITS
746 1203 3042 DCA XMTDAT /SAVE THE WORD FOR TRANSMITTING
747 1204 4333 JMS SLWDAT /GO TRANSMIT AND RECEIVE ONE WORD
748 1205 4516 NERROR
749 1206 1204 +2
750 1207 4463 LOOP /LOOP ON TEST IF SR2=1
751 /
752 /DATA TEST 3 - TRANSMIT ONES AND ZEROES
753 /
754 1210 4462 SDTST3, STLPPC /STORE LOOPING PC FOR TEST LOOPING
755 1211 3042 DCA XMTDAT /SAVE WORD
```

```
756 1212 7120 CLL CML /SET LINK
757 1213 1042 TAD XMTDAT
758 1214 7004 RAL /AND SHIFT LEFT
759 1215 3042 DCA XMTDAT /SET TRANSMIT WORD
760 1216 4333 JMS SLWDAT /GO TRANSMIT AND RECEIVE ONE WORD
761 1217 1042 TAD XMTDAT /COMPLEMENT DATA WORD
762 1220 7040 CMA
763 1221 0040 AND DATBIT
764 1222 3042 DCA XMTDAT /SET TRANSMIT WORD
765 1223 4333 JMS SLWDAT /GO TRANSMIT AND CHECK IT
766 1224 1042 TAD XMTDAT
767 1225 7040 CMA
768 1226 0040 AND DATBIT
769 1227 3042 DCA XMTDAT
770 1230 4516 NERROR
771 1231 1213 SDTST3+3
772 1232 4463 LOOP /LOOP ON TEST IF SR 2=1
773
774 /DATA TEST 4 - TRANSMIT AND RECEIVE A BINARY COUNT PATTERN
775
776 1233 4462 SDTST4, STLPPC /STORE LOOPING PC FOR TEST LOOPING
777 1234 4460 LOAD
778 1235 1040 TAD DATBIT /SET UP WORD COUNTER FROM THE # OF DATA BITS
779 1236 7040 CMA
780 1237 3052 DCA TSTCNT /SAVE IT
781 1240 3042 DCA XMTDAT /CLEAR THE TRANSMIT WORD
782 1241 4333 JMS SLWDAT /GO TRANSMIT AND RECEIVE ONE WORD
783 1242 2042 ISZ XMTDAT /INCREMENT THE TRANSMIT WORD
784 1243 1042 TAD XMTDAT
785 1244 4575 JMS I [FILCHK /GO CHECK FILLER CHAR FOR LF IF SELECTED
786 1245 5242 JMP .-3 /FIL IS SELECTED AND ITS A LF GO GET NEW WORD
787 1246 4516 NERROR
788 1247 1241 .-6 /NO GO DO NEXT WORD
789 1250 4463 LOOP /LOOP ON TEST IF SR2=1
790
791 /DATA TEST 5 - TRANSMIT A COMPLEMENTING BINARY COUNT PATTERN
792
793 1251 4462 SDTST5, STLPPC /STORE LOOPING PC FOR TEST LOOPING
794 1252 4460 LOAD /TIMING VALUE FOR APT.
795 1253 1040 TAD DATBIT /SET UP WC FROM NUMBER OF DATA BITS
796 1254 7040 CMA
797 1255 3052 DCA TSTCNT /SAVE IT
798 1256 3042 DCA XMTDAT /CLEAR THE TRANSMIT WORD
799 1257 4333 JMS SLWDAT /GO TRANSMIT AND RECEIVE ONE WORD
800 1260 1042 TAD XMTDAT /SET THE TRANSMIT WORD TO ITS COMPLEMENT
801 1261 7040 CMA
802 1262 0040 AND DATBIT
803 1263 3042 DCA XMTDAT /SAVE THE NEW WORD
804 1264 1042 TAD XMTDAT
805 1265 4575 JMS I [FILCHK
806 1266 7410 SKP
807 1267 4333 JMS SLWDAT /GO TRANSMIT AND RECEIVE ONE WORD
808 1270 1042 TAD XMTDAT /RESET THE WORD BACK AND ADD ONE
809 1271 7041 CIA
810 1272 0040 AND DATBIT
```

```
811 1273 3042 DCA XMTDAT
812 1274 1042 TAD XMTDAT
813 1275 4575 JMS I [FILCHK
814 1276 5257 JMP SDTST5+6
815 1277 4516 NERROR
816 1300 1256 SDTST5+5
817 1301 4463 LOOP /LOOP ON TEST IF SR2=1
818
819 /DATA TEST 6 - TRANSMIT A COMPLEMENTING ONE'S AND ZEROES PATTERN
820
821 1302 4462 SDTST6, STLPPC /STORE THE LOOPING PC FOR TEST LOOPING
822 1303 1055 TAD K0252
823 1304 0040 AND DATBIT /MASK OUT FOR NUMBER OF BITS
824 1305 3042 DCA XMTDAT /SAVE IT
825 1306 4333 JMS SLWDAT /GO TRANSMIT AND CHECK THE WORD
826 1307 1056 TAD K0125 /GET THE COMPLEMENTING WORD
827 1310 0040 AND DATBIT /MASK OUT FOR NUMBER OF BITS
828 1311 3042 DCA XMTDAT /SAVE IT
829 1312 4333 JMS SLWDAT /GO TRANSMIT AND CHECK IT
830 1313 4516 NERROR
831 1314 1303 SDTST6+1 /DO AGAIN
832 1315 4463 LOOP /LOOP ON TEST IF SR2=1
833
834 /DATA TEST 7 - RANDOM DATA TEST
835
836 1316 4462 SDTST7, STLPPC /STORE LOOPING PC FOR TEST LOOPING
837 1317 4460 LOAD /TIMING FOR APT
838 1320 1054 TAD M1000
839 1321 3052 DCA TSTCNT
840 1322 4506 RANDOM
841 1323 1042 TAD XMTDAT
842 1324 4575 JMS I [FILCHK /CHECK FOR FILLER CHARACTERS
843 1325 5320 JMP .-5
844 1326 4333 JMS SLWDAT /GO TRANSMIT THE RANDOM NUMBER AND CHECK IT
845 1327 4516 NERROR
846 1330 1322 .-6
847 1331 4463 LOOP /LOOP ON TEST IF SR2=1
848 1332 5574 JMP I [FDDATAT
849
850 /SLOW DATA TEST ROUTINE
851
852 1333 0000 SLWDAT, 0
853 1334 7326 CLA CLL CML RTL /SET STATUS WORD ENABLE
854 1335 6035 KSE0, KSE
855 1336 7200 CLA
856 1337 1042 TAD XMTDAT /GET WORD TO BE TRANSMITTED
857 1340 6046 SLWTLs, Tls /LOAD AND TRANSMIT AND CLEAR
858 1341 6031 KSF24, KSF /SKIP IF THE RECEIVE FLAG=1
859 1342 7410 SKP
860 1343 5350 JMP SLWREC /GO GET THE RECEIVE WORD
861 1344 6041 TSF32, TSF /SKIP IF TRANSMIT FLAG SET
862 1345 5341 JMP .-4
863 1346 6042 TCF14, TCF /CLEAR THE TRANSMIT FLAG
864 1347 5341 JMP SLWTLs+1 /GO WAIT FOR THE RECEIVE FLAG
865
```

```

866 1350 7240 SLWREC, CLA CMA
867 1351 6036 KRB11, KRB /READ RECEIVE BUFFER AND CLEAR RECEIVE FLAG
868 1352 3045 DCA RECDAT /SAVE THE WORD RECEIVED
869 1353 1045 TAD RECDAT /COMPARE THE WORD WITH THE WORD TRANSMITTED
870 1354 7041 CIA
871 1355 1042 TAD XMTDAT
872 1356 7650 SNA CLA
873 1357 5372 JMP SLOLAS /GO GET THE NEXT WORD
874 1360 4512 AERROR /TEST FOR APT ERROR CONDITION
875 1361 5372 JMP SLOLAS /ACTIVE CONSOLE
876 1362 4517 LAS
877 1363 7700 SMA CLA /HALT ON ERROR SWITCH SET?
878 1364 5372 JMP SLOLAS /NO, GO TEST FOR LOOP CONDITION
879 1365 1042 TAD XMTDAT
880 1366 7402 HALT /GOOD DATA WORD IN AC-WORD TRANSMITTED
881 1367 7200 CLA
882 1370 1045 TAD RECDAT
883 1371 7402 HALT /AC=BAD DATA WORD-WORD RECEIVED
884 1372 4517 SLOLAS, LAS /CHECK SR1 FOR LOOP ON ERROR
885 1373 7004 RAL /PUT BIT 1 IN AC0
886 1374 7710 SPA CLA /LOOP?
887 1375 5337 JMP SLWTL5-1 /YES,GO TRANSMIT AND RECEIVE SAME WORD
888 1376 5733 SLOLST, JMP I SLWDAT /RETURN FOR THE NEXT WORD
889
890
891 1400 PAGE
892
893 /FASTER DATA TEST - TRANSMIT-TRANSMIT-RECEIVE-TRANSMIT-RECEIVE-ETC
894
895 1400 4462 FDATAT, STLPPC /STORE LOOPING PC FOR TEST LOOPING
896 1401 4460 LOAD /TIMING FOR APT-8
897 1402 1176 TAD [-3720
898 1403 3047 DCA NDELAY
899 1404 1054 TAD /SETUP A DELAY OF 200 MS FOR SCOPE LOOPING
900 1405 3052 DCA TSTCNT /SETUP A TEST LOOP OF 1000 WORDS
901 1406 3044 DCA ERRFLG /CLEAR THE PROGRAM ERROR FLAG
902 1407 4461 DELAY /DELAY 200MS
903 1410 6036 KRB2, KRB /ISSUE A KRB TO CLEAR ANY RECEIVE FLAG SET
904 1411 7200 CLA
905 1412 6042 TCF9, TCF /CLEAR THE TRANSMIT FLAG IF SET
906 1413 4506 RANDOM /GENERATE A RANDOM NUMBER
907 1414 1042 TAD XMTDAT
908 1415 4575 JMS I [FILCHK /CHECK TO SEE IF FILL CHAR OPTION SELECTED
909 1416 5211 JMP ,=5 /GO GET NEW WORD,IT WAS A LF AND FILL WAS SELECTED
910 1417 1042 TAD XMTDAT /GET THE WORD TO TRANSMIT
911 1420 6046 XMIT, TLS /TRANSMIT THE WORD
912 1421 6041 TSF23, TSF /WAIT FOR THE FLAG
913 1422 5221 JMP ,=1
914 1423 5230 JMP ,=5 /GO TRANSMIT ANOTHER CHARACTER.
915 1424 6046 TLS10, TLS
916 1425 6031 KSF18, KSF /WAIT READER FLAG
917 1426 5225 JMP ,=1
918 1427 5236 JMP RECVE
919 1430 4771 RANDM1 /GO GENERATE ANOTHER WORD (XMTDT1)
920 1431 1043 TAD XMTDT1

```

```

921 1432 4575 JMS I [FILCHK /CHECK TO SEE IF FILL=1 AND THAT IT WAS A LF
922 1433 5226 JMP ,=5 /YES IT WAS,GO GENERATE A NEW WORD
923 1434 1043 TAD XMTDT1 /GET THE WORD AND PRINT IT
924 1435 5224 JMP TLS10 /
925 1436 7240 RECVE, CLA CMA
926 1437 6036 KRB3, KRB /READ THE BUFFER AND CLEAR THE FLAGS
927 1440 3045 DCA RECDAT /SAVE THE WORD
928 1441 1045 TAD RECDAT /GET THE WORD AND COMPARE IT TO THE WORD
929 1442 7041 CIA /TRANSMITTED
930 1443 1042 TAD XMTDAT
931 1444 7650 SNA CLA /ARE THEY EQUAL?
932 1445 5265 JMP UPDATE /YES,GO CHECK LOOP SWITCH
933 1446 4512 AERROR /REPORT TO APT?
934 1447 5263 JMP UPDATE-2 /ACTIVE CONSOLE
935 1450 4517 LAS
936 1451 7700 SMA CLA /HALT ON ERROR CONDITION
937 1452 5263 JMP UPDATE-2 /NO, CHECK FOR LOOP ON ERROR
938 1453 1042 TAD XMTDAT /GET THE FIRST WORD TRANSMITTED
939 1454 7402 HALT /AC=THE 1ST WORD TRANSMITTED
940 1455 7200 CLA
941 1456 1045 TAD RECDAT /
942 1457 7402 HALT /AC=WORD RECEIVED
943 1460 7200 CLA
944 1461 1043 TAD XMTDT1
945 1462 7402 HALT /AC=2ND WORD TRANSMITTED IF PROGRAM GOT THAT FAR
946 1463 7240 CLA CMA
947 1464 3044 DCA ERRFLG /SET ERROR FLAG FOR SCOPE LOOPING
948 1465 4517 UPDATE, LAS /IS SR1=1
949 1466 7004 RAL
950 1467 7710 SPA CLA
951 1470 5310 JMP ERRLOP /LOOP ON DATA PATTERN
952 1471 1043 TAD XMTDT1 /PUT SECOND WORD IN FIRST WORD FOR COMPARISON
953 1472 3042 DCA XMTDAT /OF NEXT READ
954 1473 4516 NERROR
955 1474 1476 ,+2
956 1475 5302 JMP END /END OF TEST
957 1476 1044 TAD ERRFLG /CHECK THE ERROR FLAG FOR RETURN POINTER
958 1477 7640 SZA CLA /ERROR GO START TEST OVER
959 1500 5206 JMP FDATAT+6 /GO TRANSMIT NEXT CHARACTER AND WAIT FOR RECEIVE
960 1501 5221 JMP XMIT+1
961 1502 4467 END, KSFSKP
962 1503 4464 EHLTLP /LAST FLAG FAILED TO SET
963 1504 6036 KRB4, KRB /CLEAR THE FLAG
964 1505 6042 TCF10, TCF /CLEAR THE TRANSMIT FLAG
965 1506 4463 LOOP /LOOP ON TEST IF SR1=1
966 1507 5573 JMP I [CHARLG
967
968
969 1510 4461 ERRLOP, DELAY /DELAY 200MS TO ALLOW FLAGS TO SETTLE
970 1511 6036 KRB5, KRB /CLEAR RECEIVE FLAG IF SET
971 1512 6042 TCF11, TCF /CLEAR TRANSMIT FLAG IF SET
972 1513 7200 CLA
973 1514 1042 TAD XMTDAT /GET THE FIRST WORD TO TRANSMIT
974 1515 6046 TLS1, TLS /LOAD AND TRANSMIT IT
975 1516 6041 TSF24, TSF

```

```
976 1517 5316 JMP .-1 /WAIT FOR THE FIRST TRANSMIT FLAG
977 1520 7200 CLA /
978 1521 1043 TAD XMTDT1 /GET THE SECOND WORD TO TRANSMIT
979 1522 6046 TLS2, TLS /LOAD AND TRANSMIT IT
980 1523 6031 FDTLOP, KSF /WAIT FOR THE FIRST RECEIVE FLAG
981 1524 5323 JMP .-1 /
982 1525 7240 CLA CMA /
983 1526 6036 KRB6, KRB /READ THE FIRST WORD
984 1527 3045 DCA RECDAT /SAVE THE FIRST WORD RECEIVED
985 1530 1045 TAD RECDAT /COMPARE IT TO THE FIRST WORD TRANSMITTED
986 1531 7041 CIA /
987 1532 1042 TAD XMTDAT /
988 1533 7640 SZA CLA /
989 1534 5364 JMP XSR1LP /ERROR GO CHECK SR1
990 1535 6041 TSF25, TSF /
991 1536 5335 JMP .-1 /WAIT FOR 2ND TRANSMIT FLAG
992 1537 1042 TAD XMTDAT /GET THE FIRST WORD AGAIN
993 1540 6046 TLS3, TLS /LOAD AND TRANSMIT IT
994 1541 6031 KSF19, KSF /
995 1542 5341 JMP .-1 /WAIT FOR SECOND RECEIVE FLAG
996 1543 7240 CLA CMA /
997 1544 6036 KRB7, KRB /READ THE SECOND WORD
998 1545 3045 DCA RECDAT /IS IT EQUAL TO SECOND TRANSMIT
999 1546 1045 TAD RECDAT /
1000 1547 7041 CIA /
1001 1550 1043 TAD XMTDT1 /
1002 1551 7640 SZA CLA /
1003 1552 5364 JMP XSR1LP /ERROR,GO CHECK SR1=1
1004 1553 6041 TSF26, TSF /
1005 1554 5353 JMP .-1 /WAIT FOR THE TRANSMIT FLAG
1006 1555 1043 TAD XMTDT1 /GET 2ND WORD AND TRANSMIT IT
1007 1556 6046 TLS4, TLS /LOAD AND TRANSMIT
1008 1557 4517 LAS /CHECK SR1=1 TO LOOP ON TRANSMIT RECEIVE
1009 1560 7004 RAL /
1010 1561 7710 SPA CLA /
1011 1562 5323 JMP FDTLOP /
1012 1563 5206 JMP FDATA+6 /
1013 /
1014 1564 4517 XSR1LP, LAS /
1015 1565 7004 RAL /
1016 1566 7710 SPA CLA /
1017 1567 5310 JMP ERRLOP /
1018 1570 5271 JMP UPDATE+4 /
1019 /
1020 4771 RANDM1=JMS I . /
1021 1571 4123 / XRAND1 /
1022 / /
1023 / /
1024 1600 PAGE /
1025 /
1026 /
1027 /
1028 /
1029 /
1030 /
```

/THE FOLLOWING TEST CHECKS THAT THE NUMBER OF DATA BITS WERE
/SETUP CORRECTLY. TRANSMIT 377 AND TAKE THE 1'S COMPLEMENT
/OF THE DATA BIT MASK WORD AND CHECK THAT THE AC CAME BACK
/AS ZERES.

```
1031 1600 4462 CHARLG, STLPPC /STORE THE LOOPING PC FOR ERPOP AND TEST LOOPING
1032 1601 4460 LOAD /TIMING FOR APT.
1033 1602 1040 TAD DATBIT /SETUP 1'S COMPLEMENT OF SELECTED DATA
1034 1603 7040 CMA /BIT CHARACTER LENGTH
1035 1604 3042 DCA XMTDAT /SAVE IT FOR COMPARISON
1036 1605 1034 TAD K377 /
1037 1606 6046 TLS5, TLS /TRANSMIT 8 BITS OF ONES
1038 1607 4466 TSF5SKP /WAIT FOR THE TRANSMIT FLAG TO SET
1039 1610 4464 EHLTLP /TRANSMIT FLAG FAILED TO SET
1040 1611 6042 TCF12, TCF /CLEAR THE FLAG
1041 1612 4467 KSF5SKP /WAIT FOR THE RECEIVE FLAG TO SET
1042 1613 4464 EHLTLP /ERROR,RECEIVE FLAG FAILED TO SET
1043 1614 7240 CLA CMA /
1044 1615 6036 KRB8, KRB /READ THE WORD AND SAVE IT
1045 1616 3045 DCA RECDAT /
1046 1617 1045 TAD RECDAT /
1047 1620 0042 AND XMTDAT /
1048 1621 7450 SNA /
1049 1622 5242 JMP FILERT-3 /TEST FOR LOOP ON TEST
1050 1623 3043 DCA XMTDT1 /SAVE THE BITS THAT WEREN'T SUPPOSED TO BE SELECTED
1051 1624 4512 AERROR /
1052 1625 5236 JMP CHRLAS /CONSOLE ACTIVE
1053 1626 4517 LAS /
1054 1627 7700 SNA CLA /HALT ON ERROR
1055 1630 5236 JMP CHRLAS /NO, TEST FOR LOOP ON TEST CONDITON
1056 1631 1043 TAD XMTDT1 /
1057 1632 7402 HALT /AC=BITS THAT WEREN'T SUPPOSED TO BE SELECTED
1058 1633 7200 CLA /
1059 1634 1040 TAD DATBIT /
1060 1635 7402 HALT /AC=DATA BITS THAT OPERATOR HAD TOLD THE PROGRAM
1061 1636 4517 CHRLAS, LAS /THAT WERE SELECTED
```

```
1062 1637 7004 RAL
1063 1640 7710 SPA CLA
1064 1641 5204 JMP CHARLG+4
1065 1642 4516 NERROR
1066 1643 1604 CHARLG+4
1067 1644 4463 LOOP
1068
1069
1070
1071
1072
1073
1074
1075
1076 1645 1036 FILERT, TAD SAVBTS
1077 1646 7006 RTL
1078 1647 7700 SMA CLA
1079 1650 5777 JMP STENAB
1080 1651 4462 STLPPC
1081 1652 3047 DCA NDELAY
1082 1653 4461 DELAY
1083 1654 6036 KRB9, KRB
1084 1655 7200 CLA
1085 1656 6042 TCF13, TCF
1086 1657 1376 TAD [-4
1087 1660 3043 DCA XMTDT1
1088 1661 1057 TAD K212
1089 1662 0040 AND DATBIT
1090 1663 3042 DCA XMTDAT
1091 1664 1042 TAD XMTDAT
1092 1665 6046 TLS6, TLS
1093 1666 7200 CLA
1094 1667 1172 FILLOP, TAD [-100
1095 1670 3051 DCA CNT2
1096 1671 3050 DCA CNT1
1097 1672 6041 TSF28, TSF
1098 1673 7610 SKP CLA
1099 1674 4464 EHLTLP
1100 1675 6031 KSF21, KSF
1101 1676 7610 SKP CLA
1102 1677 5305 JMP .+6
1103 1700 2050 ISZ CNT1
1104 1701 5272 JMP .-7
1105 1702 2051 ISZ CNT2
1106 1703 5272 JMP .-11
1107 1704 4464 EHLTLP
1108 1705 1043 TAD XMTDT1
1109 1706 1375 TAD (-4
1110 1707 7640 SZA CLA
1111 1710 3042 DCA XMTDAT
1112 1711 6036 KRB10, KRB
1113 1712 3045 DCA RECDAT
1114 1713 1045 TAD RECDAT
1115 1714 7041 CIA
1116 1715 1042 TAD XMTDAT

/LOOP ON ERROR
/LOOP ON TEST IF SR2=1

/FILLER CHARACTER TEST-DO THIS TEST IF OPERATOR HAS SELECTED
/THE FILLER CHARACTER OPTION. THE PROGRAM TRANSMITS A LINE
/FEED AND CHECKS THAT 5 RECEIVE FLAGS COME BACK. THE DATA RECEIVED
/SHOULD BE 1 WORD OF LINE FEED AND 4 WORDS OF FILLER CHARACTERS.
/THE PROGRAM ALSO CHECKS THAT THE TRANSMIT FLAG DOES NOT GET SET
/UNTIL ALL THE RECEIVE FLAGS ARE IN.

/CHECK TO SEE IF FILLER CHARACTERS SELECTED

/WAS IT SELECTED BY THE OPERATOR
/NO,GO CHECK FOR STATUS ENABLE
/STORE THE LOOPING PC
/SETUP PROGRAM DELAY OF 409MS.
/DELAY TO ALLOW FLAGS TO SETTLE
/CLEAR THE RECEIVE FLAG IF SET

/CLEAR THE TRANSMIT FLAG IF SET
/SETUP A COUNTER TO RECEIVE FOUR FLAGS
/SAVE IT
/GET LINE FEED
/MASK OUT TO WORD LENGTH
/SAVE IT FOR COMPARISON OF FIRST WORD
/GET THE LINE FEED CHARACTER
/LOAD AND TRANSMIT IT

/LOOP TO WAIT FOR THE RECEIVE FLAG AND CHECK XMIT

/SKIP IF TRANSMIT FLAG=1

/ERROR,TRANSMIT FLAG SET-SHOULD GET 4 RECEIVE FLAGS FIRST
/SKIP IF RECEIVE FLAG SET

/ERROR,RECEIVE FLAG NOT SET OR MISSING SOME
/SETUP TO COMPARE FOR EITHER A L,F. OR FILLER
/WAS IT THE FIRST CHARACTER

/NO,THEN CLEAR COMPARE WORD FOR FILLER CHAR
/READ THE WORD AND CLEAR THE FLAG
/SAVE IT
/COMPARE THE WORD RECEIVED WITH WORD EXPECTED
```

```
1117 1716 7650 SNA CLA
1118 1717 5337 JMP CNTREC
1119 1720 4512 AERROR
1120 1721 5344 JMP CNTREC+5
1121 1722 4517 LAS
1122 1723 7700 SMA CLA
1123 1724 5332 JMP CNTREC-5
1124 1725 1042 TAD XMTDAT
1125 1726 7402 HALT
1126 1727 7200 CLA
1127 1730 1045 TAD RECDAT
1128 1731 7402 HALT
1129 1732 4517 LAS
1130 1733 7004 RAL
1131 1734 7710 SPA CLA
1132 1735 5251 JMP FILERT+4
1133 1736 5774 JMP FILEXT
1134 1737 2043 CNTREC, ISZ XMTDT1
1135 1740 5267 JMP FILLOP
1136 1741 1172 TAD [-100
1137 1742 3051 DCA CNT2
1138 1743 3050 DCA CNT1
1139 1744 6031 KSF22, KSF
1140 1745 7610 SKP CLA
1141 1746 4464 EHLTLP
1142
1143 1747 6041 TSF29, TSF
1144 1750 7610 SKP CLA
1145 1751 5357 JMP .+6
1146 1752 2050 ISZ CNT1
1147 1753 5344 JMP .-7
1148 1754 2051 ISZ CNT2
1149 1755 5344 JMP .-11
1150 1756 4464 EHLTLP
1151 1757 6032 KCC5, KCC
1152 1760 1172 TAD [-100
1153 1761 3051 DCA CNT2
1154 1762 6031 KSF23, KSF
1155 1763 7610 SKP CLA
1156 1764 5372 JMP .+6
1157 1765 2050 ISZ CNT1
1158 1766 5362 JMP .-4
1159 1767 2051 ISZ CNT2
1160 1770 5362 JMP .-6
1161 1771 4464 EHLTLP
1162 1772 5773 JMP KCC6
1163
1164 1773 2000
1165 1774 2003
1166 1775 0004
1167 1776 7774
1168 1777 2005
1169 2000 PAGE
1170 2000 6032 KCC6, KCC

/WORD COMPARES,GO BUMP RECEIVE COUNTER
/ACTIVE CONSOLE
/HAIT ON ERROR?
/NO, CHECK FOR LOOP ON ERROR
/PRESS "CONTINUE" FOR EXPECTED CHARACTER
/AC=WORD EXPECTED X12=LF OR 0000=FILE CHAR.

/GET THE WORD RECEIVED
/AC=WORD RECEIVED SHOULD BE 212 OR 12 OR 000
/CHECK SR1 TO LOOP ON ERROR

/LOOP ON THE ERROR
/EXIT THE TEST
/BUMP THE RECEIVE COUNTER
/GO GET THE NEXT RECEIVE FLAG

/CHECK THAT THE RECEIVE FLAG DOESN'T GET SET BEFORE XMIT

/LAST RECEIVE FLAG SHOULDN'T GET SET UNTIL SOME
/TIME AFTER THE TRANSMIT FLAG
/WAIT FOR THE TRANSMIT FLAG TO GET SET

/TRANSMIT FLAG FAILED TO SET AFTER 5 RECEIVE FLAGS
/CLEAR THE FLAG

/LAST RECEIVE FLAG FAILED TO SET

/CLEAR THE RECEIVE FLAG
```



```
1171 2001 4516 NERROR
1172 2002 1657 FILL0P=10 /GO TRANSMIT ANOTHER FILLER CHARACTER
1173 2003 4463 FILEXT, LOOP /LOOP ON TEST IF SR2=1
1174 2004 5205 JMP STENAB
1175
1176
1177
1178 /THE FOLLOWING TEST WILL BE EXECUTED ONLY IF THE OPERATER HAS SET
1179 /THE STATUS ENABLE BIT TO A ONE WHEN HE INITIALIZED THE PROGRAM,
1180 /THIS TEST WILL CHECK THAT THE ERROR BIT AND THE OVERRUN BIT CAN
1181 /BE SET AND CLEARED IN THE STATUS REGISTER, THE TEST WILL ALSO
1182 /CHECK THAT STATUS ENABLE F/F CAN BE SET AND CLEARED, THE RECEIVE
1183 /BUFFER WILL ALSO BE CHECKED TO CONTAIN THE CORRECT WORD, THREE
1184 /WORDS WILL BE TRANSMITTED AND THEN THE STATUS AND THE RECEIVE BUFFER
1185 /WILL BE CHECKED.
1186 2005 1036 STENAB, TAD SAVBTS /CHECK TO SEE IF STATUS ENABLE WAS SELECTED
1187 2006 7004 RAL
1188 2007 7700 SMA CLA /WAS IT SELECTED BY THE OPERATOR
1189 2010 5333 JMP SR4HLT /NO, GO CHECK END OF PROGRAM HALT
1190 2011 4462 STLPPC /STORE THE LOOPING PC FOR TEST AND SCOPE LOOPING
1191 2012 3047 DCA NDELAY /SETUP A DELAY OF 409MS
1192 2013 4461 DELAY /DELAY TO ALLOW FLAGS TO SETTLE FOR SCOPE LOOPING
1193 2014 6036 KRB12, KRB /CLEAR THE RECEIVE FLAG IF SET
1194 2015 6042 TCF15, TCF /CLEAR THE TRANSMIT FLAG IF SET
1195 2016 7346 CLA CLL CMA RTL /SETUP A COUNT OF 3 TO TRANSMIT 3 TIMES
1196 2017 3043 DCA XMTDT1 /SAVE IT
1197 2020 7001 IAC
1198 2021 3042 DCA XMTDAT /SET THE FIRST WORD TO BE TRANSMITTED=1
1199 2022 7326 CLA CLL CML RTL /SET AC BIT 10 TO A 1
1200 2023 6035 KSE1, KSE /AND TRY TO SET STATUS ENABLE TO A 1
1201 2024 7200 CLA
1202 2025 1042 TAD XMTDAT /GET THE WORD (1,2 OR 3)
1203 2026 6046 TLS9, TLS /TRANSMIT IT
1204 2027 7200 CLA
1205 2030 4466 TSFSKP /WAIT FOR THE TRANSMIT FLAG
1206 2031 4464 EHLTLP /ERROR, TRANSMIT FLAG FAILED TO SET
1207 2032 2042 ISZ XMTDAT /BUMP THE WORD TO A 2 THEN 3
1208 2033 2043 ISZ XMTDT1 /TRANSMITTED 3 WORDS YET
1209 2034 5225 JMP /NO, GO TRANSMIT NEXT WORD
1210 2035 6042 TCF16, TCF /CLEAR THE TRANSMIT FLAG
1211 2036 6031 KSF25, KSF /SKIP ON THE RECEIVE FLAG
1212 2037 4464 EHLTLP /ERROR, RECEIVE FLAG FAILED TO SET AFTER 3 XMTS
1213 2040 6034 KRS2, KRS /DO A STATIC READ OF STATUS AND RECEIVE BUFFER
1214 2041 3045 DCA RECDAT /AND SAVE IT
1215 2042 1377 TAD (4402 /GET EXPECTED WORD (ERROR-OVERRUN-DATA OF 2)
1216 2043 3042 DCA XMTDAT /AND SAVE IT FOR COMPARISON
1217 2044 4301 JMS STERR /GO CHECK THE WORDS FOR ERRORS
1218 2045 6035 KSE2, KSE /CLEAR STATUS ENABLE AND CHECK BUFFER FOR A 2
1219 2046 6034 KRS3, KRS /DO A STATIC READ OF THE RECEIVE BUFFER
1220 2047 3045 DCA RECDAT /SAVE THE WORD
1221 2050 7326 CLA CLL CML RTL /SETUP FOR WORD EXPECTED
1222 2051 3042 DCA XMTDAT /SAVE IT FOR COMPARISON
1223 2052 4301 JMS STERR /GO CHECK FOR ERRORS
1224 2053 7326 CLA CLL CML RTL /SET AC BIT 10 TO A 1
1225 2054 6035 KSE3, KSE /RESET STATUS ENABLE
```

```
1226 2055 7200 CLA
1227 2056 1377 TAD (4402
1228 2057 3042 DCA XMTDAT /RESET EXPECTED DATA TO (ERROR-OVERRUN-DATA OF 2)
1229 2060 6036 KRB13, KRB /DO A DYNAMIC READ OF STATUS AND RECEIVE BUFFER
1230 2061 3045 DCA RECDAT /SAVE THE WORD READ FOR COMPARISON
1231 2062 4301 JMS STERR /GO CHECK FOR ERROR CONDITIONS
1232 2063 6031 KSF26, KSF /CHECK THAT RECEIVE FLAG = 0
1233 2064 7610 SKP CLA
1234 2065 4464 EHLTLP /ERROR, RECEIVE FLAG SHOULD NOT BE SET YET
1235 2066 4467 KSFSKP /GO WAIT FOR THE LAST TRANSMITTED CHARACTER
1236 2067 4464 EHLTLP /THE THIRD TRANSMIT DID NOT SET RECEIVE FLAG
1237 2070 6036 KRB14, KRB /READ THE CHARACTER
1238 2071 3045 DCA RECDAT /SAVE IT FOR COMPARISON
1239 2072 1376 TAD (3
1240 2073 3042 DCA XMTDAT /SETUP FOR WORD EXPECTED
1241 2074 4301 JMS STERR /GO CHECK FOR ERRORS
1242 2075 4516 NERROR
1243 2076 2012 STENAB+5 /NO GO DO TEST AGAIN
1244 2077 4463 LOOP /LOOP ON TEST IF SR2=1
1245 2100 5333 JMP SR4HLT /GO CHECK END OF PROGRAM HALT
1246
1247 2101 0000 STERR, 0 /CHECK TO SEE IF ERROR EXIST IN STATUS REGISTER AND RECEIVE BUFFER
1248 2102 1045 TAD RECDAT /GET THE WORD RECEIVED AND COMPARE IT WITH
1249 2103 7041 CIA /THE WORD EXPECTED
1250 2104 1042 TAD XMTDAT /
1251 2105 7650 SMA CLA /ARE THEY EQUAL?
1252 2106 5701 JMP I STERR /YES, CONTINUE TESTING
1253 2107 7340 CLL CLA CMA /AC=-1
1254 2110 1301 TAD STERR /ESATBLISH ERROR PC
1255 2111 4512 AERROR
1256 2112 5326 JMP STLAS /ACTIVE CONSOLE SELECTED
1257 2113 4517 LAS
1258 2114 7700 SMA CLA /HALT ON ERROR
1259 2115 5326 JMP STLAS /NO, GO TEST FOR LOOP ON ERROR
1260 2116 1010 TAD 10 /GET BACK ERROR PC
1261 2117 7402 HALT /AC=PC WERE ERROR WAS DETECTED AT
1262 2120 7200 CLA
1263 2121 1042 TAD XMTDAT /GET THE WORD EXPECTED
1264 2122 7402 HALT /AC=WORD EXPECTED
1265 2123 7200 CLA
1266 2124 1045 TAD RECDAT /GET THE WORD RECEIVED
1267 2125 7402 HALT /AC=WORD RECEIVED
1268 2126 4517 STLAS, LAS /LOOP ON ERROR?
1269 2127 7004 RAL
1270 2130 7710 SPA CLA
1271 2131 5211 JMP STENAB+4 /YES GO LOOP
1272 2132 5701 JMP I STERR /NO, GO GET NEXT ERROR
1273
1274
1275
1276 /HALT AT END OF PROGRAM IF SWITCH REGISTER 4 EQUALS A ONE
1277 2133 1022 SR4HLT, TAD 22
1278 2134 7710 SPA CLA /RUNNING ON APT?
1279 2135 5351 JMP BAUDST /YES
1280 2136 4515 CHEK22
```

```
1281 2137 4775' JMS XC8PAS /PRINT END OF PASS IF ON CONSOLE
1282 2140 7000 NOP
1283 2141 4517 LAS
1284 2142 0374 AND (200
1285 2143 7650 SNA CLA
1286 2144 5351 JMP BAUDST /SET UP BUAD TABLE.
1287 2145 4515 CHEK22
1288 2146 4773' JMS XC8PSW
1289 2147 4520 HLT /END OF THE PROGRAM SR4=1
1290 2150 5772' JMP CLRBRD /PRESS CONTINUE TO GO ON
1291
1292 2151 1022 BAUDST, TAD 22
1293 2152 7004 RAL
1294 2153 7710 SPA CLA /SPECIAL TESTING BEING DONE
1295 2154 4771' JMS NEWDEV /YES UPDATE DEVICE NUMBER
1296 2155 1524 TAD I TEMP
1297 2156 3121 DCA CLKCNT /GET FIRST CLOCK COUNT.
1298 2157 7240 STA
1299 2160 1124 TAD TEMP /SET UP FOR AUTO INDEX.
1300 2161 3011 DCA A11
1301 2162 5772' JMP CLRBRD /EXIT AND CONTINUE PROGRAM
1302
1303 /
1304 /ROUTINE TO LOAD VALUE FOR APT TIMING.
1305 /
1306 2163 0000 XLOAD, 0
1307 2164 1411 TAD I A11 /GET TIMING VALUE.
1308 2165 7450 SNA /TEST FOR A ZERO.
1309 2166 5763 JMP I XLOAD /THERE IS ONE, USE SAME VALUE.
1310 2167 3122 DCA COUNT /SETUP NEW VALUE.
1311 2170 5763 JMP I XLOAD /EXIT.
1312
1313 2171 4000
1314 2172 0271
1315 2173 6703
1316 2174 0200
1317 2175 7241
1318 2176 0003
1319 2177 4402
1320 2200 PAGE
1321
1322 2200 0000 HLTLOP, 0 /INHIBIT ERROR HALT IF SR0=1 AND LOOP ON ERROR IF SR1=1
1323 2201 7240 CLA CMA
1324 2202 1200 TAD HLTLOP
1325 2203 4512 AERROR /GET THE FAILING PC WHERE THE ERROR WAS DETECTED
1326 2204 7410 SKP /APT ERROR HANDLER.
1327 2205 4517 LAS /ACTIVE CONSOLE
1328 2206 7700 SNA CLA /HALT ON ERROR?
1329 2207 5212 JMP ,+3
1330 2210 1010 TAD 10
1331 2211 7402 HALT /GET BACK ERROR PC
1332 2212 4517 LAS /AC=FAILING PC WHERE ERROR WAS DETECTED
1333 2213 7004 RAL /LOOP ON THE ERROR?
1334 2214 7710 SPA CLA
```

```
1335 2215 5446 JMP I LOOPPC
1336 2216 5600 JMP I HLTLOP
1337
1338 2217 0000 HLTLP1, 0 /ROUTINE USED IN ACNSKP TEST TO SAVE AC FOR FALSE SKIPPING
1339 2220 3240 DCA SAVAC /SAVE THE AC FOR NON LOOPING PURPOSES
1340 2221 7240 CLA CMA
1341 2222 1217 TAD HLTLP1 /GET THE FAILING PC WHERE ERROR WAS DETECTED
1342 2223 4512 AERROR /REPORT ERROR TO APT-8 SYSTEM
1343 2224 5232 JMP ,+6 /ACTIVE CONSOLE
1344 2225 4517 LAS /HALT ON ERROR?
1345 2226 7700 SNA CLA
1346 2227 5232 JMP ,+3
1347 2230 1010 TAD 10
1348 2231 7402 HALT /GET BACK ERROR PC
1349 2232 4517 LAS /AC=FAILING PC WHERE ERROR WAS DETECTED
1350 2233 7004 RAL /LOOP ON THE ERROR?
1351 2234 7710 SPA CLA
1352 2235 5446 JMP I LOOPPC /YES, LOOP ON THE ERROR
1353 2236 1240 TAD SAVAC /NO, RESET THE AC AND CONTINUE
1354 2237 5617 JMP I HLTLP1 /RETURN AND CHECK THE EFFECT OF THE IOT ON AC
1355
1356 2240 0000 SAVAC, 0
1357
1358 2241 0000 HLTLP2, 0 /THIS ROUTINE USED ONLY WHEN IOT EFFECTS CONTENTS OF AC
1359 2242 3240 DCA SAVAC /SAVE THE AC FOR ERROR INDICATION
1360 2243 7240 CLA CMA /GET THE FAILING PC WHERE ERROR WAS DETECTED
1361 2244 1241 TAD HLTLP2 /
1362 2245 4512 AERROR /REPORT ERROR TO APT.
1363 2246 5254 JMP ,+6 /ACTIVE CONSOLE
1364 2247 4517 LAS /HALT ON ERROR?
1365 2250 7700 SNA CLA
1366 2251 5254 JMP ,+3
1367 2252 1010 TAD 10
1368 2253 7402 HALT /GET BACK ERROR PC
1369 2254 7200 CLA /AC=FAILING PC WHERE ERROR WAS DETECTED
1370 /PRESS "CONT" TO GET CONTENTS OF AC AFTER EXECUTION
/AND COMPARISON OF THE IOT AND AC
1371 2255 1240 TAD SAVAC
1372 2256 7402 HALT /AC=BITS THAT WERE EFFECTED AFTER EXECUTION OF IOT
1373 2257 4517 LAS /LOOP ON THE ERROR?
1374 2260 7004 RAL
1375 2261 7710 SPA CLA
1376 2262 5446 JMP I LOOPPC /YES, GO LOOP ON THE ERROR
1377 2263 5641 JMP I HLTLP2 /RETURN AND CONTINUE THE TEST
1378
```

```
1379
1380
1381 /BAUD RATE TIMING TEST-LOAD AND START 202 AND WATCH YOUR WATCH
1382 /FOR 30 SECONDS, THE PROCESSOR SHOULD HALT IN 30 SECONDS, IF IT
1383 /DOESN'T CHECK THE BAUD RATE WITH A SCOPE OR CHECK THE BAUD RATE
1384 /SWITCHES
1385 2264 1377 BAUDTM, TAD (JMP I 2
1386 2265 3001 DCA 1
1387 2266 1376 TAD (RETINT
1388 2267 3002 DCA 2
1389 2270 1041 TAD BAUDNO /GET THE BAUD RATE
1390 2271 1375 TAD (BAUDTB /GET THE ADDRESS OF THE BAUD RATE TABLE
1391 2272 3333 DCA BDPNTR /SAVE THE POINTER TO THE BAUD RATE TABLE
1392 2273 1037 TAD BITNO /GET THE CHARACTER LENGTH
1393 2274 7104 CLL RAL /MULTIPLY IT BY 2
1394 2275 1733 TAD I BDPNTR /ADD IN BAUD RATE ADDRESS
1395 2276 3333 DCA BDPNTR /ADDRESS OF BAUD RATE CONSTANTS ARE READY TO BE SETUP
1396 2277 4462 STLPPC /STORE THE LOOPING PC FOR TEST LOOPING
1397 2300 1333 TAD BDPNTR /GET POINTER ADDRESS TO THE CONSTANTS
1398 2301 3052 DCA TSTCNT /SAVE IT IN TEST COUNT
1399 2302 1452 TAD I TSTCNT /GET THE FIRST CONSTANT
1400 2303 3050 DCA CNT1
1401 2304 2052 ISZ TSTCNT /BUMP THE ADDRESS BY 1 FOR NEXT CONSTANT
1402 2305 1452 TAD I TSTCNT
1403 2306 3051 DCA CNT2
1404 2307 6046 TST7, TLS /LOAD AND TRANSMIT FIRST CHARACTER-FLAG
1405 2310 6041 TSF30, TSF /COMES UP ALMOST IMMEDIATELY
1406 2311 5310 JMP *-1
1407 2312 5315 JMP *+3
1408 2313 6036 INTON, KRB /CLEAR THE RECEIVE FLAG
1409 2314 7610 SKP CLA
1410 2315 6046 TST8, TLS /LOAD AND TRANSMIT NEXT CHARACTER AND CLEAR FLAG
1411 2316 6001 ION /TURN THE INTERRUPT ON
1412 2317 6031 RETINT, KSF /SKIP IF RECEIVE FLAG SET
1413 2320 7610 SKP CLA /FLAG NOT SET, CHECK TRANSMIT FLAG
1414 2321 5313 JMP INTON /RECEIVE FLAG SET, GO CLEAR IT
1415 2322 6041 TSF31, TSF /SKIP IF TRANSMIT FLAG SET
1416 2323 5317 JMP INTON+4 /NOT SET YET GO WAIT FOR A FLAG
1417 2324 2050 ISZ CNT1 /BUMP THE FIRST COUNTER
1418 2325 5315 JMP INTON+2 /GO TRANSMIT ANOTHER CHARACTER
1419 2326 2051 ISZ CNT2 /FIRST COUNTER OVERFLOWED
1420 2327 5315 JMP INTON+2 /GO DO ANOTHER 4095 INTERRUPTS
1421 2330 4520 HLT /TRANSMITTED FOR 30 SECONDS????
1422 2331 4463 LOOP /LOOP ON TEST IF SR2=1
1423 2332 5330 JMP *-2 /END OF THE TEST
1424 2333 0000 BDPNTR, 0
1425
1426 /POINTERS TO BAUD RATE TABLE
1427 2334 2400 BAUDTB, BR110
1428 2335 2414 BR150
1429 2336 2430 BR300
1430 2337 2444 BR600
1431 2340 2460 BR1200
1432 2341 2474 BR2400
1433 2342 2510 BR4800
```

```
1434 2343 2524 BR9600
1435 2344 2540 B19200
1436 2345 2554 BR568
1437 2346 2570 BR667
1438 2347 2604 BR1050
1439
1440 /POINTERS TO BAUD RATE TYPEOUTS
1441 2350 4473 MESTAB, MESS6A
1442 2351 4520 MESS6B
1443 2352 4545 MESS6C
1444 2353 4572 MESS6D
1445 2354 4617 MESS6E
1446 2355 4645 MESS6F
1447 2356 4673 MESS6G
1448 2357 4721 MESS6H
1449 2360 4747 MESS6I
1450 2361 4776 MESS6J
1451 2362 5024 MESS6K
1452 2363 5052 MESS6L
1453
1454 /POINTERS TO DATA BIT TYPEOUTS
1455 2364 5145 MESTB1, MES10A
1456 2365 5164 MES10B
1457 2366 5203 MES10C
1458 2367 5222 MES10D
1459
1460
1461 2375 2334
1462 2376 2317
1463 2377 5402
1464 2400 PAGE
1465
1466 /BAUD RATE CONSTANTS FOR 110 BAUD
1467 2400 7051 BR110, -727 /7 BITS AT 15.71 CHAR/SEC=471 CHAR/30 SEC
1468 2401 7777 -1
1469 2402 7143 -635 /8 BITS AT 13.75 CHAR/SEC=413 CHAR/30 SEC
1470 2403 7777 -1
1471 2404 7221 -557 /9 BITS AT 12.22 CHAR/SEC=367 CHAR/30 SEC
1472 2405 7777 -1
1473 2406 7266 -512 /10 BITS AT 11 CHAR/SEC=330 CHAR/30 SEC
1474 2407 7777 -1
1475 2410 7324 -454 /11 BITS AT 10 CHAR/SEC=300 CHAR/30 SEC
1476 2411 7777 -1
1477 2412 7355 -423 /12 BITS AT 9.17 CHAR/SEC=275 CHAR/30 SEC
1478 2413 7777 -1
1479
1480 /BAUD RATE CONSTANTS FOR 150 BAUD
1481
1482 2414 6575 BR150, -1203 /7 BITS AT 21.43 CHAR/SEC=643 CHAR/30 SEC
1483 2415 7777 -1
1484 2416 6715 -1063 /8 BITS AT 18.75 CHAR/SEC=563 CHAR/30 SEC
1485 2417 7777 -1
1486 2420 7014 -764 /9 BITS AT 16.67 CHAR/SEC=500 CHAR/30 SEC
1487 2421 7777 -1
```

1488 2422 7076 -702
1489 2423 7777 -1
1490 2424 7147 -631
1491 2425 7777 -1
1492 2426 7211 -567
1493 2427 7777 -1
1494
1495
1496
1497 2430 5372 BR300, -2406
1498 2431 7777 -1
1499 2432 5633 -2145
1500 2433 7777 -1
1501 2434 6030 -1750
1502 2435 7777 -1
1503 2436 6174 -1604
1504 2437 7777 -1
1505 2440 6316 -1462
1506 2441 7777 -1
1507 2442 6422 -1356
1508 2443 7777 -1
1509
1510
1511
1512 2444 2765 BR600, -5013
1513 2445 7777 -1
1514 2446 3466 -4312
1515 2447 7777 -1
1516 2450 4060 -3720
1517 2451 7777 -1
1518 2452 4370 -3410
1519 2453 7777 -1
1520 2454 4633 -3145
1521 2455 7777 -1
1522 2456 5044 -2734
1523 2457 7777 -1
1524
1525
1526
1527 2460 5750 BR1200, -2030
1528 2461 7776 -2
1529 2462 7153 -625
1530 2463 7776 -2
1531 2464 0140 -7640
1532 2465 7777 -1
1533 2466 0760 -7020
1534 2467 7777 -1
1535 2470 1467 -6311
1536 2471 7777 -1
1537 2472 2110 -5670
1538 2473 7777 -1
1539
1540
1541
1542 2474 3720 BR2400, -4060

/10 BITS AT 15 CHAR/SEC=450 CHAR/30 SEC
/11 BITS AT 13.64 CHAR/SEC=409 CHAR/30 SEC
/12 BITS AT 12.50 CHAR/SEC=375 CHAR/30 SEC

/BAUD RATE CONSTANTS FOR 300 BAUD
/7 BITS AT 42.86 CHAR/SEC=1286 CHAR/30 SEC
/8 BITS AT 37.50 CHAR/SEC=1125 CHAR/30 SEC
/9 BITS AT 33.33 CHAR/SEC=1000 CHAR/30 SEC
/10 BITS AT 30.00 CHAR/SEC=900 CHAR/30 SEC
/11 BITS AT 27.27 CHAR/SEC=818 CHAR/30 SEC
/12 BITS AT 25.00 CHAR/SEC=750 CHAR/30 SEC

/BAUD RATE CONSTANTS FOR 600 BAUD
/7 BITS AT 85.71 CHAR/SEC=2571 CHAR/30 SEC
/8 BITS AT 75.00 CHAR/SEC=2250 CHAR/30 SEC
/9 BITS AT 66.67 CHAR/SEC=2000 CHAR/30 SEC
/10 BITS AT 60.00 CHAR/SEC=1800 CHAR/30 SEC
/11 BITS AT 54.55 CHAR/SEC=1637 CHAR/30 SEC
/12 BITS AT 50.00 CHAR/SEC=1500 CHAR/30 SEC

/BAUD RATE CONSTANTS FOR 1200 BAUD
/7 BITS AT 171.43 CHAR/SEC=5143 CHAR/30 SEC
/8 BITS AT 150 CHAR/SEC=4500 CHAR/30 SEC
/9 BITS AT 133.33 CHAR/SEC=4000 CHAR/30 SEC
/10 BITS AT 120 CHAR/SEC=3600 CHAR/30 SEC
/11 BITS AT 109.09 CHAR/SEC=3273 CHAR/30 SEC
/12 BITS AT 100 CHAR/SEC=3000 CHAR/30 SEC

/BAUD RATE CONSTANTS FOR 2400 BAUD
/7 BITS AT 342.86 CHAR/SEC=10,286 CHAR/30 SEC

1543 2475 7775 -3
1544 2476 6326 -1452
1545 2477 7775 -3
1546 2500 0277 -7501
1547 2501 7776 -2
1548 2502 1737 -6041
1549 2503 7776 -2
1550 2504 3156 -4622
1551 2505 7776 -2
1552 2506 4217 -3561
1553 2507 7776 -2
1554
1555
1556
1557 2510 7640 BR4800, -140
1558 2511 7772 -6
1559 2512 4654 -3124
1560 2513 7773 -5
1561 2514 0575 -7203
1562 2515 7774 -4
1563 2516 3675 -4103
1564 2517 7774 -4
1565 2520 6332 -1446
1566 2521 7774 -4
1567 2522 0436 -7342
1568 2523 7775 -3
1569
1570
1571
1572 2524 7477 BR9600, -301
1573 2525 7765 -13
1574 2526 1530 -6250
1575 2527 7767 -11
1576 2530 1371 -6407
1577 2531 7770 -10
1578 2532 7571 -207
1579 2533 7770 -10
1580 2534 4664 -3114
1581 2535 7771 -7
1582 2536 1073 -6705
1583 2537 7772 -6
1584
1585
1586
1587 2540 7176 B19200, -602
1588 2541 7753 -25
1589 2542 3257 -4521
1590 2543 7756 -22
1591 2544 2761 -5017
1592 2545 7760 -20
1593 2546 7362 -416
1594 2547 7761 -17
1595 2550 1550 -6230
1596 2551 7763 -15
1597 2552 2165 -5613

/8 BITS AT 300 CHAR/SEC=9000 CHAR/30 SEC
/9 BITS AT 266.67 CHAR/SEC=8000 CHAR/30 SEC
/10 BITS AT 240 CHAR/SEC=7200 CHAR/30 SEC
/11 BITS AT 218.18 CHAR/SEC=6545 CHAR/30 SEC
/12 BITS AT 200 CHAR/SEC=6000 CHAR/30 SEC

/BAUD RATE CONSTANTS FOR 4800 BAUD
/7 BITS AT 685.71 CHAR/SEC=20,571 CHAR/30 SEC
/8 BITS AT 600 CHAR/SEC=18,000 CHAR/30 SEC
/9 BITS AT 533.33 CHAR/SEC=16,000 CHAR/30 SEC
/10 BITS AT 480 CHAR/SEC=14,400 CHAR/30 SEC
/11 BITS AT 436.36 CHAR/SEC=13,091 CHAR/30 SEC
/12 BITS AT 400 CHAR/SEC=12000 CHAR/30 SEC

/BAUD RATE CONSTANTS FOR 9600 BAUD
/7 BITS AT 1371.43 CHAR/SEC=41,143 CHAR/30 SEC
/8 BITS AT 1200 CHAR/SEC=36,000 CHAR/30 SEC
/9 BITS AT 1066.67 CHAR/SEC=32000 CHAR/30 SEC
/10 BITS AT 960 CHAR/SEC=28,800 CHAR/30 SEC
/11 BITS AT 872.73 CHAR/SEC=26,182 CHAR/30 SEC
/12 BITS AT 800 CHAR/SEC=24,000 CHAR/30 SEC

/BAUD RATE CONSTANTS FOR 19.2 KILO BAUD
/7 BITS AT 2742.86 CHAR/SEC=82,286 CHAR/30 SEC
/8 BITS AT 2400 CHAR/SEC=72,000 CHAR/30 SEC
/9 BITS AT 2133.33 CHAR/SEC=64,000 CHAR/30 SEC
/10 BITS AT 1920 CHAR/SEC=57,600 CHAR/30 SEC
/11 BITS AT 1745.45 CHAR/SEC=52,364 CHAR/30 SEC
/12 BITS AT 1600 CHAR/SEC=48,000 CHAR/30 SEC

```
1598      2553 7764      -14
1599
1600
1601
1602      /BAUD RATE CONSTANTS FOR 56.8 BAUD
1603      2554 7415      BR568, -363      /7 BITS AT 8.11 CHAR/SEC =243 CHAR/30 SEC
1604      2555 7777      -1
1605      2556 7453      -325      /8 BITS AT 7.10 CHAR/SEC =213 CHAR/30 SEC
1606      2557 7777      -1
1607      2560 7503      -275      /9 BITS AT 6.31 CHAR/SEC =189 CHAR/30 SEC
1608      2561 7777      -1
1609      2562 7526      -252      /10 BITS AT 5.68 CHAR/SEC =170 CHAR/30 SEC
1610      2563 7777      -1
1611      2564 7545      -233      /11 BITS AT 5.16 CHAR/SEC =155 CHAR/30 SEC
1612      2565 7777      -1
1613      2566 7562      -216      /12 BITS AT 4.73 CHAR/SEC =142 CHAR/30 SEC
1614      2567 7777      -1
1615
1616      /BAUD RATE CONSTANTS FOR 66.7 BAUD
1617
1618      2570 7342      BR667, -436      /7 BITS AT 9.53 CHAR/SEC =286/30 SEC
1619      2571 7777      -1
1620      2572 7406      -372      /8 BITS AT 8.35 CHAR/SEC =250/30 SEC
1621      2573 7777      -1
1622      2574 7442      -336      /9 BITS AT 7.41 CHAR/SEC =222/30 SEC
1623      2575 7777      -1
1624      2576 7470      -310      /10 BITS AT 6.67 CHAR/SEC =200/30 SEC
1625      2577 7777      -1
1626      2600 7512      -266      /11 BITS AT 6.06 CHAR/SEC =182/30 SEC
1627      2601 7777      -1
1628      2602 7531      -247      /12 BITS AT 5.56 CHAR/SEC =167/30 SEC
1629      2603 7777      -1
1630
1631      /BAUD RATE CONSTANTS FOR 1050 BAUD
1632
1633      2604 7154      BR1050, -624      /7 BITS AT 150 CHAR/SEC =4500/30 SEC
1634      2605 7776      -2
1635      2606 0236      -7542      /8 BITS AT 131.25 CHAR/SEC =3938/30 SEC
1636      2607 7777      -1
1637      2610 1124      -6654      /9 BITS AT 116.66 CHAR/SEC =3500/30 SEC
1638      2611 7777      -1
1639      2612 1662      -6116      /10 BITS AT 105 CHAR/SEC =3150/30 SEC
1640      2613 7777      -1
1641      2614 2320      -5460      /11 BITS AT 95.45 CHAR/SEC =2864/30 SEC
1642      2615 7777      -1
1643      2616 2677      -5101      /12 BITS AT 87.5 CHAR/SEC =2625/30 SEC
1644      2617 7777      -1
1645
```

```
1646      2620 0000      XDELAY, 0
1647      2621 7300      CLA      CLL
1648      2622 1047      TAD      NDELAY
1649      2623 3243      DCA      DELAYN
1650      2624 1244      DELLOP, TAD      CON100
1651      2625 3245      DCA      US100
1652      2626 2245      ISZ      US100
1653      2627 5226      JMP      *-1
1654      2630 7200      CLA
1655      2631 7200      CLA
1656      2632 7200      CLA
1657      2633 7200      CLA
1658      2634 2243      ISZ      DELAYN
1659      2635 7610      SKP      CLA
1660      2636 5620      JMP I XDELAY
1661      2637 0620      AND I XDELAY
1662      2640 0620      AND I XDELAY
1663      2641 0220      AND XDELAY
1664      2642 5224      JMP      DELLOP
1665
1666      2643 0000      DELAYN, 0
1667      2644 7754      CON100, -24
1668      2645 0000      US100, 0
1669
1670
1671
1672      /IF FILLER CHARACTER OPTION IS SELECTED-DO NOT TRANSMIT A L.F.
1673      /FILLER CHARACTERS WILL BE CHECKED LATER.
1674      2646 0000      FILCHK, 0
1675      2647 3270      DCA      CHKFIL
1676      2650 1036      TAD      SAVBTS
1677      2651 7006      RTL
1678      2652 7700      SMA      CLA
1679      2653 5263      JMP      *-10
1680      2654 1270      TAD      CHKFIL
1681      2655 1377      TAD      (-12
1682      2656 7450      SNA
1683      2657 5265      JMP      *-6
1684      2660 1376      TAD      (-200
1685      2661 7650      SNA      CLA
1686      2662 5265      JMP      *-3
1687      2663 2246      ISZ      FILCHK
1688      2664 5646      JMP I FILCHK
1689      2665 2052      ISZ      TSTCNT
1690      2666 5646      JMP I FILCHK
1691      2667 5646      JMP I FILCHK
1692
1693      2670 0000      CHKFIL, 0
1694
1695      2671 0000      WATTSF, 0
1696      2672 1172      TAD      [-100
1697      2673 3051      DCA      CNT2
1698      2674 3050      DCA      CNT1
1699      2675 6041      TSF15, TSF
1700      2676 7610      SKP      CLA

/WAS FILLER CHARACTER OPTION SELECTED
/BY THE OPERATOR

/NO,EXIT AND TRANSMIT THE CHARACTER
/CHECK TO SEE IF THE WORD TO BE TRANSMITTED IS A
/LINE FEED
/WAS IT A 12

/WAS IT A 212

/YES IT WAS A LINE FEED
/WORD IS OK,GO TRANSMIT IT

/BUMP TEST COUNTER TO GET RID OF WORD
/GO GET ANOTHER WORD
/GO DO TEST OVER IF LAST WORD IS A LINE FEED

/ROUTINE TO WAIT FOR TRANSMIT FLAG.
/IF IT DOESN'T SET IN A SECOND OR SO
/TIMEOUT AND HALT
```

```
1701 2677 5305 JMP ,+6
1702 2700 2050 ISZ CNT1
1703 2701 5275 JMP ,+4
1704 2702 2051 ISZ CNT2
1705 2703 5275 JMP ,+6
1706 2704 5671 JMP I WATTISF /TSF FAILED TO SKIP
1707 2705 2271 ISZ WATTISF
1708 2706 5671 JMP I WATTISF /OK,TSF SKIPPED RETURN
1709
1710 2707 0000 WATKSF, 0 /ROUTINE TO WAIT FOR THE RECEIVE FLAG
1711 2710 1172 TAD [-100 /IF IT DOESN'T SET IN A SECOND OR SO
1712 2711 3051 DCA CNT2 /TIMEOUT AND HALT.
1713 2712 3050 DCA CNT1
1714 2713 6031 KSF8, KSF
1715 2714 7610 SKP CLA
1716 2715 5323 JMP ,+6
1717 2716 2050 ISZ CNT1
1718 2717 5313 JMP ,+4
1719 2720 2051 ISZ CNT2
1720 2721 5313 JMP ,+6
1721 2722 5707 JMP I WATKSF /ERROR,KSF FAILED TO SKIP
1722 2723 2307 ISZ WATKSF /
1723 2724 5707 JMP I WATKSF /OK,RECEIVE FLAG SKIPPED
1724
1725
1726 2776 7600
1727 2777 7766
1728 3000 PAGE
1729
1730 3000 4471 TYINTR, MESSAGE /INTERAGATION SETUP FOR THE TELETYPE
1731 3001 4203 MESS1 /TYPE RECEIVE IOT?
1732 3002 4473 TWOOC2
1733 3003 5200 JMP ,+3 /GET RECEIVE DEVICE CODE
1734 3004 4511 BSWAP /INPUT ERROR
1735 3005 3035 DCA DEVCOD /SWAP IT AROUND TO BITS 0-5
1736 3006 4471 MESSAGE /SAVE THE RECEIVE DEVICE CODE
1737 3007 4213 MESS2 /TYPE TRANSMIT IOT
1738 3010 4473 TWOOC2
1739 3011 5206 JMP ,+3 /GET TRANSMIT IOT
1740 3012 1035 TAD DEVCOD /INPUT ERROR
1741 3013 3035 DCA DEVCOD /ADD TRANSMIT IOT TO RECEIVE IOT
1742 3014 4471 MESSAGE /SAVE THE IOTS
1743 3015 4224 MESS3 /TYPE PARITY(Y OR N)?
1744 3016 4505 YESRNO /WAIT FOR A YES OR NO
1745 3017 5214 JMP ,+3 /NOT A Y OR N
1746 3020 7610 SKP CLA /SET NO PARITY BIT
1747 3021 7330 CLA CLL CML RAR /SET THE PARITY BIT TO A 1
1748 3022 3036 DCA SAVBTS /SAVE THE PARITY BIT IN STATUS WORD
1749 3023 4471 MESSAGE
1750 3024 4236 MESS3A
1751 3025 1036 TAD SAVBTS
1752 3026 7710 SPA CLA
1753 3027 7001 IAC
1754 3030 4475 PRNT1
```

```
1755 3031 4471 MESSAGE /TYPE STATUS REGISTER(Y OR N)
1756 3032 4241 MESS4
1757 3033 4505 YESRNO /WAIT FOR A YES OR NO
1758 3034 5231 JMP ,+3 /NOT A YES OR NO
1759 3035 7610 SKP CLA /NO STATUS REGISTER SELECTED
1760 3036 7332 CLA CLL CML RTR /STATUS REGISTER IS SELECTED
1761 3037 1036 TAD SAVBTS /ADD STATUS REGISTER BIT TO WORD
1762 3040 3036 DCA SAVBTS /AND SAVE IT
1763 3041 4471 MESSAGE
1764 3042 4302 MESS4A
1765 3043 1036 TAD SAVBTS
1766 3044 7004 RAL
1767 3045 7710 SPA CLA
1768 3046 7001 IAC
1769 3047 4475 PRNT1
1770 3050 4471 MESSAGE /TYPE FILLER CHARACTERS(Y OR N)?
1771 3051 4306 MESS5
1772 3052 4505 YESRNO /WAIT FOR A YES OR NO
1773 3053 5250 JMP ,+3 /NOT A YES OR NO
1774 3054 7610 SKP CLA /NO FILLER CHARACTERS
1775 3055 1377 TAD (1000 /YES,FILLER CHARACTERS
1776 3056 1036 TAD SAVBTS /ADD THE FILLER CHARACTER BIT TO STATUS WORD
1777 3057 3036 DCA SAVBTS /
1778 3060 4471 MESSAGE
1779 3061 4326 MESS5A
1780 3062 1036 TAD SAVBTS
1781 3063 7006 RTL
1782 3064 7710 SPA CLA
1783 3065 7001 IAC
1784 3066 4475 PRNT1
1785 3067 4471 MESSAGE /TYPE BAUD RATE(00-13)?
1786 3070 4400 MESS6
1787 3071 4471 MESSAGE
1788 3072 4440 CMES66
1789 3073 4473 TWOOC2 /INPUT A NUMBER FROM 00-13
1790 3074 5267 JMP ,+5 /INPUT ERROR
1791 3075 3364 DCA SAVIT /SAVE THE NUMBER TYPED BY OPERATOR
1792 3076 1364 TAD SAVIT /HAS THE NUMBER WITHIN BAUD RATE LIMITS
1793 3077 1376 TAD (-13
1794 3100 7740 SWA SZA CLA
1795 3101 5267 JMP ,+12 /NOT WITHIN LIMITS GO TYPE MESSAGE OVER
1796 3102 1364 TAD SAVIT /GET THE NUMBER AND PUT IN SAVBTS
1797 3103 7106 RAL
1798 3104 7004 RAL
1799 3105 1036 TAD SAVBTS
1800 3106 3036 DCA SAVBTS /PUT NUMBER IN BITS 5 6 7 AND 8
1801 3107 1036 TAD SAVBTS /ADD IT TO THE STATUS WORD
1802 3110 7012 RTR /AND SAVE IT
1803 3111 7010 RAR
1804 3112 0375 AND (17
1805 3113 1374 TAD (MESTAB
1806 3114 3320 DCA CHGMES
1807 3115 1720 TAD I CHGMES
1808 3116 3320 DCA CHGMES
1809 3117 4471 MESSAGE
```

```
1810 3120 4473 CHGMES, MESS6A
1811 3121 4471 MESSAGE /TYPE TWO STOP BITS?
1812 3122 5100 MESS7
1813 3123 4505 YESRND /WAIT FOR A Y OR N
1814 3124 5321 JMP ,=3 /INPUT ERROR
1815 3125 7610 SKP CLA /ONLY 1 STOP BIT
1816 3126 1373 TAD (4 /2 STOP BITS
1817 3127 1036 TAD SAVBTS /ADD THE NUMBER OF STOP BITS
1818 3130 3036 DCA SAVBTS /TO THE STATUS WORD
1819 3131 4471 MESSAGE
1820 3132 5115 MESS7A
1821 3133 1036 TAD SAVBTS
1822 3134 0373 AND (4
1823 3135 7650 SNA CLA
1824 3136 7001 IAC
1825 3137 4475 PRNT1
1826 3140 4471 MESSAGE /TYPE # OF DATA BITS (0,1,2,3)
1827 3141 5120 MESS10
1828 3142 4472 ONEOCT
1829 3143 5340 JMP ,=3 /WAIT FOR A NUMBER
1830 3144 3320 DCA CHGMES /NOT A NUMBER
1831 3145 1372 TAD (=3 /SAVE THE NUMBER
1832 3146 1320 TAD CHGMES /IS IT GREATER THAN 3
1833 3147 7740 SNA SZA CLA
1834 3150 5340 JMP ,=10 /YES,TRY AGAIN
1835 3151 1320 TAD CHGMES /
1836 3152 1036 TAD SAVBTS /ADD IT TO STATUS WORD
1837 3153 3036 DCA SAVBTS /AND SAVE IT
1838 3154 1371 TAD (MESTB1
1839 3155 1320 TAD CHGMES
1840 3156 3362 DCA ,+4
1841 3157 1762 TAD I ,+3
1842 3160 3362 DCA ,+2
1843 3161 4471 MESSAGE
1844 3162 5145 MES10A
1845 3163 5770 JMP SETUP /GO AND CALCULATE IT
1846
1847 3164 0000 SAVIT, 0
1848 3170 0215
1849 3171 2364
1850 3172 7775
1851 3173 0004
1852 3174 2350
1853 3175 0017
1854 3176 7765
1855 3177 1000
1856 3200 PAGE
1857
1858
1859
1860
1861
1862 3200 PAGE
1863
```

```
1864 3200 0000 FILLER, 0 /SET TO NUMBER OF FILLERS REQUIRED
1865
1866 /INPUT ONE OCTAL NUMBER TO AC 9 THRU 11
1867 /GOOD RETURN IS JMS+2
1868
1869 3201 0000 ONEOCK, 0 /CALL BY "ONEOCT"
1870 3202 4470 LISN
1871 3203 0001 1
1872 3204 3207 ,+3
1873 3205 0000 0
1874 3206 3210 ,+2
1875 3207 2201 ISZ ONEOCK
1876 3210 5601 JMP I ONEOCK
1877
1878 /INPUT TWO OCTAL NUMBERS TO AC 6 THRU 11
1879 /GOOD RETURN IS JMS+2
1880
1881 3211 0000 TWOOCK, 0 /CALL BY "TWOOCK"
1882 3212 4201 JMS ONEOCK
1883 3213 5611 JMP I TWOOCK
1884 3214 7104 CLL RAL
1885 3215 7006 RTL
1886 3216 3224 DCA XPRNT2
1887 3217 4201 JMS ONEOCK
1888 3220 5611 JMP I TWOOCK
1889 3221 1224 TAD XPRNT2
1890 3222 2211 ISZ TWOOCK
1891 3223 5611 JMP I TWOOCK
1892
1893 /PRINT THE TWO OCTAL NUMBERS IN THE AC 6 THRU 11
1894
1895 3224 0000 XPRNT2, 0 /CALL BY "PRNT2"
1896 3225 3211 DCA TWOOCK
1897 3226 1211 TAD TWOOCK
1898 3227 7012 RTR
1899 3230 7010 RAR
1900 3231 4475 PRNT1
1901 3232 1211 TAD TWOOCK
1902 3233 4475 PRNT1
1903 3234 5624 JMP I XPRNT2
1904
1905 /TYPE THE ASCII CHARACTER IN THE AC
1906
1907 3235 0000 XTYPE, 0 /CALL BY "TYPE"
1908 3236 6046 TLS
1909 3237 7200 CLA
1910 3240 6041 TSF
1911 3241 5240 JMP ,=1
1912 3242 6042 TCF
1913 3243 5635 JMP I XTYPE
1914
1915
1916
1917 /TYPE A CR AND LF WITH NUMBER OF FILLERS
1918 /AS DETERMINED BY LOCATION "FILLER"
```

```
1919
1920 3244 0000 XCRLF, 0 /CALL BY "CRLF"
1921 3245 7200 CLA
1922 3246 1260 TAD K215
1923 3247 4501 TYPE
1924 3250 1200 TAD FILLER
1925 3251 7040 CMA
1926 3252 3261 DCA XORS
1927 3253 1057 TAD K212
1928 3254 4501 TYPE
1929 3255 2261 ISZ XORS
1930 3256 5254 JMP .-2
1931 3257 5644 JMP I XCRLF
1932 3260 0215 K215, 0215
1933
1934 /PERFORM THE XOR OF THE AC AND THE CALL+1
1935 /RETURN TO CALL+2
1936
1937 3261 0000 XORS, 0 /CALL BY "XOR"
1938 3262 3273 DCA YESRNX
1939 3263 1273 TAD YESRNX
1940 3264 0661 AND I XORS /IN BRIEF, TAD THE TWO
1941 3265 7041 CIA /NUMBERS THEN SUBTRACT
1942 3266 7104 CLL PAL /THE CARRIES TO PRODUCE
1943 3267 1273 TAD YESRNX /A HALF ADD (XOR)
1944 3270 1661 TAD I XORS
1945 3271 2261 ISZ XORS
1946 3272 5661 JMP I XORS
1947
1948 /LOOK FOR "Y" OR "N" INPUT
1949
1950 3273 0000 YESRNX, 0 /CALL BY "YESRNX"
1951 3274 4470 LISN /INPUT ONE CHARACTER IF AC=0
1952 3275 7447 .-1
1953 3276 3303 .+5 /RETURN TO CALL+3 IF "Y"
1954 3277 7462 .-N
1955 3300 3304 .+4 /RETURN TO CALL+2 IF "N"
1956 3301 0000 0
1957 3302 3305 .+3 /RETURN TO CALL+1 IF NEITHER
1958 3303 2273 ISZ YESRNX
1959 3304 2273 ISZ YESRNX
1960 3305 5673 JMP I YESRNX
1961
1962 /PRINT 2 SPACES
1963
1964 3306 0000 SPACX2, 0 /CALL BY "SPACE2"
1965 3307 4471 MESSAGE
1966 3310 3312 .+2
1967 3311 5706 JMP I SPACX2
1968 3312 4040 4040
1969 3313 0010 K10, 0010 /USED BY LISN
1970
1971
1972 /COMPARE INPUT TO LIST FOLLOWING CALL
1973 /INPUT ONE CHARACTER IF AC=0
```

```
1974 /USE LAST INPUT IF AC NON ZERO
1975
1976 3314 0000 XLISN, 0 /CALL BY "LISN"
1977 3315 7640 SZA CLA
1978 3316 5340 JMP LISN1 /USE LAST INPUT SINCE AC NOT ZERO
1979 3317 6031 KSF
1980 3320 5317 JMP .-1
1981 3321 6036 KRB
1982 3322 0372 AND K177
1983 3323 1373 TAD K200
1984 3324 3706 DCA I LISNT1
1985 3325 1706 TAD I LISNT1
1986 3326 1374 TAD M212
1987 3327 7450 SNA /IS IT A LF?
1988 3330 5334 JMP .+4 /YES
1989 3331 1375 TAD M3
1990 3332 7640 SZA CLA /IS IT A CR?
1991 3333 5336 JMP .+3 /NO
1992 3334 4502 CRLF
1993 3335 5340 JMP LISN1
1994 3336 1706 TAD I LISNT1
1995 3337 4501 TYPE /PRINT THE CHARACTER
1996 3340 1714 LISN1, TAD I XLISN /GET COMPARE VALUE
1997 3341 2314 ISZ XLISN
1998 3342 7450 SNA /EXIT?
1999 3343 5351 JMP LISN3 /YES
2000 3344 7500 SNA
2001 3345 5361 JMP LISNUM /LOOK FOR OCTAL NUMBER
2002 3346 1706 TAD I LISNT1 /COMPARE
2003 3347 7640 SZA CLA /EQUAL?
2004 3350 5356 JMP LISN2 /NO
2005 3351 3571 LISN3, DCA I (XPRNT1
2006 3352 1714 TAD I XLISN
2007 3353 3314 DCA XLISN
2008 3354 1571 TAD I (XPRNT1
2009 3355 5714 JMP I XLISN /AC IS ZERO UNLESS OCTAL NUMBER
2010 3356 7200 LISN2, CLA
2011 3357 2314 ISZ XLISN
2012 3360 5340 JMP LISN1
2013 3361 7200 LISNUM, CLA /LOOK FOR OCTAL NUMBER
2014 3362 1706 TAD I LISNT1
2015 3363 1367 TAD M270
2016 3364 7500 SNA /IS IT LESS THAN 8?
2017 3365 5356 JMP LISN2 /NO, SO NOT AN OCTAL NUMBER
2018 3366 1313 TAD K10
2019 3367 7510 M270, SZA
2020 3370 5356 JMP LISN2 /IS IT GREATER THAN ZERO?
2021 3371 5351 JMP LISN3 /NO, SO NOT A NUMBER
2022 3372 0177 K177, 0177
2023 3373 0200 K200, 0200
2024 3374 7566 M212, 7566
2025 3375 7775 M3, 7775
2026 0170 LISNT1=ISPCX2
2027 3400 PAGE
2028
```



```
2029
2030
2031      /PRINT PACKED ASCII TEXT TERMINATED BY
2032      /SIX-BIT 00
2033
2034      3400 0000      MESAGX, 0      /CALL BY "MESSAGE"
2035      3401 7200      CLA
2036      3402 1600      TAD I      MESAGX
2037      3403 3264      DCA      FOROCK
2038      3404 2200      ISZ      MESAGX      /SET UP RETURN
2039      3405 1664      TAD I FOROCK
2040      3406 7012      RTR
2041      3407 7012      RTR
2042      3410 7012      RTR
2043      3411 4216      JMS      MESAGF
2044      3412 1664      TAD I FOROCK
2045      3413 4216      JMS      MESAGF
2046      3414 2264      ISZ      FOROCK
2047      3415 5205      JMP      .-10
2048      3416 0000      MESAGF, 0
2049      3417 0235      AND      K77
2050      3420 7450      SNA
2051      3421 5600      JMP I      MESAGX      /TERMINATOR (00)?
2052      3422 1236      TAD      M43      /YES
2053      3423 7450      SNA
2054      3424 5233      JMP      .+7      /CRLF?
2055      3425 1237      TAD      K3      /YES
2056      3426 7510      SPA
2057      3427 1240      TAD      K100      /200 OR 300
2058      3430 1241      TAD      K240      /300
2059      3431 4501      TYPE
2060      3432 5616      JMP I      MESAGF      /200
2061      3433 4502      CRLF
2062      3434 5616      JMP I      MESAGF
2063      3435 0077      K77,      0077
2064      3436 7735      M43,      7735
2065      3437 0003      K3,      0003
2066      3440 0100      K100,     0100
2067      3441 0240      K240,     0240
2068
2069      /MODIFY DEVICE CODE FOR A LIST OF IOT ADDRESSES
2070      3442 0000      XMIOT, 0      /CALL BY "MIOT"
2071      3443 0262      AND      K770
2072      3444 3200      DCA      MESAGX
2073      3445 1642      TAD I      XMIOT
2074      3446 2242      ISZ      XMIOT
2075      3447 3264      DCA      FOROCK
2076      3450 1664      TAD I      FOROCK      /GET NEXT ADDRESS
2077      3451 7450      SNA
2078      3452 5642      JMP I      XMIOT      /END OF LIST? (ZERO)
2079      3453 3300      DCA      XPRNT4      /YES
2080      3454 1700      TAD I      XPRNT4
2081      3455 0263      AND      K7007      /GET IOT
2082      3456 1200      TAD      MESAGX      /REMOVE OLD DEVICIE CODE
2083      3457 3700      DCA I      XPRNT4      /ADD NEW DEVICE CODE
                                          /PUT BACK IOT
```

```
2084      3460 2264      ISZ      FOROCK
2085      3461 5250      JMP      .-11
2086
2087
2088
2089      3462 0770      K770,      0770
2090      3463 7007      K7007,     7007
2091
2092      /INPUT 4 OCTAL NUMBERS TO AC
2093      /GOOD RETURN IS CALL+2
2094
2095      3464 0000      FOROCK, 0      /CALL BY "FOROCT"
2096      3465 4473      TWOOCT
2097      3466 5664      JMP I      FOROCK
2098      3467 7106      CLL RTL
2099      3470 7006      RTL
2100      3471 7006      RTL
2101      3472 3300      DCA      XPRNT4
2102      3473 4473      TWOOCT
2103      3474 5664      JMP I      FOROCK
2104      3475 1300      TAD      XPRNT4
2105      3476 2264      ISZ      FOROCK
2106      3477 5664      JMP I      FOROCK
2107
```

```
2108      /PRINT FOUR OCTAL NUMBERS IN AC 11 THRU 0 FOLLOWED
2109      /BY TWO SPACES
2110
2111      3500 0000      XPRNT4, 0      /CALL BY "PRNT4"
2112      3501 3264      DCA      FOROCK
2113      3502 1264      TAD      FOROCK
2114      3503 7012      RTR
2115      3504 7012      RTR
2116      3505 7012      RTR
2117      3506 4476      PRNT2
2118      3507 1264      TAD      FOROCK
2119      3510 4476      PRNT2
2120      3511 4500      SPACE2
2121      3512 5700      JMP I      XPRNT4
2122
2123      /PRINT THE OCTAL NUMBER IN AC 9 THRU 11
2124      XPRNT1, 0      /CALL BY "PRNT1"
2125      3514 0320      AND      K7
2126      3515 1321      TAD      K260
2127      3516 4501      TYPE
2128      3517 5713      JMP I      XPRNT1
2129      3520 0007      K7,      0007
2130      3521 0260      K260,   0260
2131      /
2132      /SWAP BITES IN THE AC, PRESERVE THE LINK
2133      /
2134      3522 0000      XBSW, 0      /CALL BY "BSWAP"
2135      3523 3337      DCA XRAND
2136      3524 7012      RTR
2137      3525 7012      RTR
2138      3526 7012      RTR
2139      3527 1337      TAD      XPRAND
2140      3530 0336      AND      K7700
2141      3531 1337      TAD      XRAND
2142      3532 7006      RTL
2143      3533 7006      RTL
2144      3534 7006      RTL
2145      3535 5722      JMP I XBSW
2146      3536 7700      K7700, 7700
2147
2148
2149      /GENERATE RANDOM NUMBER
2150      /EXIT WITH NUMBER IN AC
2151
2152      3537 0000      XPRAND, 0      /CALL BY "RANDOM"
2153      3540 7301      CLA CLL IAC
2154      3541 1373      TAD      RAN1
2155      3542 1374      TAD      RAN2
2156      3543 7106      CLL RTL
2157      3544 3373      DCA      RAN1
2158      3545 1374      TAD      RAN2
2159      3546 7012      RTR
2160      3547 1373      TAD      RAN1
2161      3550 3374      DCA      RAN2
2162      3551 1374      TAD      RAN2
```

```
2163      3552 0040      AND      DATBIT      /MASK CHAR FOR I/O
2164      3553 3042      DCA      XMTDAT      /STORE
2165      3554 5737      JMP I      XRAND
2166
2167      /SAVE RANDOM
2168      /GENERATOR PRIMES
2169
2170      3555 0000      XSAVGN, 0      /CALL BY "SAVGEN"
2171      3556 7200      CLA
2172      3557 1373      TAD      RAN1
2173      3560 3375      DCA      SAV1
2174      3561 1374      TAD      RAN2
2175      3562 3053      DCA      SAV2
2176      3563 5755      JMP I      XSAVGN
2177
2178      /RESTORE RANDOM
2179      /GENERATOR PRIMES
2180
2181      3564 0000      XRESGN, 0      /CALL BY "RESGEN"
2182      3565 7200      CLA
2183      3566 1375      TAD      SAV1
2184      3567 3373      DCA      RAN1
2185      3570 1053      TAD      SAV2
2186      3571 3374      DCA      RAN2
2187      3572 5764      JMP I      XRESGN
2188      3573 1234      RAN1, 1234
2189      3574 5670      RAN2, 5670
2190      3575 0000      SAV1, 0
2191
2192      /
2193      3600      /PAGE
2194      /
2195      /ROUTINE TO DETERMINE IF ON APT-8 TEST SYSTEM.
2196      /IF ON APT-8 NO HALTS WILL BE RECOGNIZED AND ALL PERTINENT INFORMATION
2197      /WILL BE CONTAINED IN ADDRESSES 20 AND 22.
2198      /
2199      3600 0000      XAPT8, 0
2200      3601 1022      TAD      22
2201      3602 7104      CLL RAL
2202      3603 7520      SNA SNL
2203      3604 5210      JMP      ONAPT-3      /ON APT8 OR MULTIPLE OPTION TESTER
2204      3605 7710      SPA CLA      /NO, GO CHECK FOR CONSOLE.
2205      3606 5216      JMP      ONAPT+3      /SKIP IF NOT ON TESTER
2206      3607 5213      JMP      ONAPT
2207
2208      3610 4515      CHEK22
2209      3611 5777      JMP      SETUP-7
2210      3612 5776      JMP      BGNINT+3
2211      3613 1020      ONAPT, TAD      20
2212      3614 7440      SZA
2213      3615 3035      DCA      DEVCOD
2214      3616 1022      TAD      22
2215
2216
2217      3617 7104      CLL RAL
```

```
2218 3620 0375 AND (3000
2219 3621 3010 DCA 10 /SAVE THIS SECTION
2220 3622 1021 TAD 21 /GET PARITY BIT
2221 3623 0374 AND (2000 /ISOLATE PARITY BIT
2222 3624 7104 CLL PAL
2223 3625 1010 TAD 10
2224 3626 3010 DCA 10 /STORE PARITY,STATUS,AND
2225 /FILLER CHARACTER INFORMATION,
2226 3627 1022 TAD 22
2227 3630 0033 AND C177 /ISOLATE BAUD RATE AND DATA BITS.
2228 3631 1010 TAD 10 /GET BACK OTHER,
2229 3632 3036 DCA SAVBTS /STORE IT FOR THE PROGRAM USE,
2230 3633 1022 TAD 22 /SET ADDRESS 22 TO INDICATE APT,
2231 3634 0373 AND (6000 /ISOLATE APT FEATURE SECTION OF WORD
2232 3635 3022 DCA 22
2233 3636 1036 TAD SAVBTS
2234 3637 7010 RAR
2235 3640 7012 RTR
2236 3641 0372 AND (17 /ISOLATE BAUD RATE
2237 3642 3024 DCA SAVPNT
2238 3643 1024 TAD SAVPNT /GET BAUD RATE POINTER,
2239 3644 1371 TAD (TABLE1
2240 3645 3124 DCA TEMP /TEMP STORAGE FOR TIMING
2241 3646 1524 TAD I TEMP /GET POINTER,
2242 3647 3264 DCA XAERRO /STORE FOR FUTURE USE,
2243 3650 1264 TAD XAERRO /GET POINTER
2244 3651 3124 DCA TEMP /AND STORE FOR USE BY PROGRAM,
2245 3652 7240 STA /SET AC TO =1 FOR AUTO INDEX,
2246 3653 1264 TAD XAERRO /GET BACK POINTER
2247 3654 3011 DCA A11 /STORE VALUE
2248 3655 1664 TAD I XAERRO /GET VALUE
2249 3656 3121 DCA CLKCNT /STORE FOR APT,
2250 3657 1022 TAD 22
2251 3660 7004 RAL
2252 3661 7710 SPA CLA /SPECIAL FEATURES ENABLED?
2253 3662 6370 APTIO0 /YES, INITIALIZE TO DEVICE ZERO
2254 3663 5770 JMP SETUP /NOW RUN PROGRAM SKIPPING
2255 /INITIAL INTERROGATION,
2256 /
2257 /
2258 /ROUTINE TO REPORT ERROR TO APT=8 IF REQUIRED,
2259 /ONLY THE ERROR PC WILL BE ESTABLISHED,
2260 /
2261 3664 0000 XAERRO, 0
2262 3665 7440 SZA /WAS ERROR IN THE AC, SKP IF NO
2263 3666 5271 JMP .+3
2264 3667 7240 STA /AC=-1
2265 3670 1264 TAD XAERRO /ESTABLISH PC
2266 3671 3010 DCA 10 /SAVE IT
2267 3672 1022 TAD 22 /GET HCW2
2268 3673 7004 RAL
2269 3674 7520 SMA SNL /SKP IF ON APT OR MULTIPLE
2270 3675 5317 JMP EXIT /WAS NEITHER EXIT ROUTINE,
2271 3676 6002 IOF /DISABLE INTERRUPT
2272 3677 7700 SMA CLA /MULTIPLE OPTION SELECTED?
```

```
2273 3700 5311 JMP APTCHK /NO, NOW GO TEST FOR APT
2274 3701 6372 APTIO2 /SET FAIL FLAG
2275 3702 7420 SNL /SKP IF ON APT
2276 3703 5307 JMP .+4
2277 3704 4517 LAS /GET SWITCHES
2278 3705 7710 SPA CLA /SKIP IF NOT GOING TO PROM
2279 3706 5313 JMP GOTOUV /GO TO PROM
2280 3707 4767 JMS NEWDEV /GET THE NEXT DEVICE
2281 3710 5766 JMP CLRBRD /AND START OVER
2282 /
2283 3711 7420 APTCHK, SNL /SKP IF ON APT
2284 3712 5317 JMP EXIT /NOT ON APT
2285 /
2286 3713 1010 GOTOUV, TAD 10 /GET BACK ERROR PC
2287 3714 6272 CIF 70 /SET FIELD TO UV PROM (7)
2288 3715 5736 JMP I K6520 /GO TO PROM
2289 3716 5316 JMP . /SOMETHING WENT WRONG
2290 /
2291 3717 7300 EXIT, CLL CLA
2292 3720 4515 CHEK22 /TEST FOR ACTIVE CONSOLE
2293 3721 7410 SKP /CONSOLE ACTIVE,
2294 3722 5334 JMP .+12
2295 3723 4517 LAS
2296 3724 7710 SPA CLA /HALT ON ERROR SELECTED?
2297 3725 5334 JMP .+7 /NO, TAKE EXIT,
2298 3726 4471 MESSAGE
2299 3727 4200 MPC /PRINT PC=
2300 3730 1010 TAD 10 /GET BACK ERROR PC
2301 3731 4477 PRNT4 /PRINT IT
2302 3732 4765 JMS XC8PSW /ASK SR QUESTION
2303 3733 5664 JMP I XAERRO
2304 3734 2264 ISZ XAERRO
2305 3735 5664 JMP I XAERRO
2306 3736 6520 K6520, 6520
2307 /
2308 /ROUTINE TO LOOP ON NO ERROR FOR THE DESIRED NUMBER OF TIMES.
2309 /THIS ALSO CONTAINS THE APT TIMING SECTION,
2310 /
2311 3737 0000 XNERRO, 0
2312 3740 1737 TAD I XNERRO /GET TEST TO LOOP ON
2313 3741 3356 DCA RETURN /STORE FOR RETURN,
2314 3742 2337 ISZ XNERRO /UPDATE RETURN,
2315 3743 4514 TICK /APT TIMING
2316 3744 4515 CHEK22 /CONSOLE ACTIVE
2317 3745 7410 SKP /YES, CHECK FOR KEYBOARD FLAG,
2318 3746 5353 JMP .+5
2319 3747 6031 KSF /KEYBOARD FLAG SET?
2320 3750 5353 JMP .+3 /NO
2321 3751 6036 KRB /YES, READ THE CHARACTER
2322 3752 4764 JMS XC8CNT /CHECK FOR CONTROL CHARACTER
2323 3753 2052 ISZ TSTCNT
2324 3754 5756 JMP I RETURN
2325 3755 5737 JMP I XNERRO /CONTINUE TESTING,
2326 /
2327 3756 0000 RETURN, 0
```

```
2328 /
2329 /
2330 3764 6600
2331 3765 6703
2332 3766 0271
2333 3767 4000
2334 3770 0215
2335 3771 5500
2336 3772 0017
2337 3773 6000
2338 3774 2000
2339 3775 3000
2340 3776 0203
2341 3777 0206
      4000
2342 PAGE
2343 /THIS ROUTINE WILL UPDATE THE DEVICE NUMBER TO TEST IF
2344 /THE PROGRAM IS BEING RUN FOR MULTIPLE TESTING.
2345 /
2346 NEWDEV, 0
2347 CAF
2348 TAD 20
2349 BSW
2350 SPA CLA /MOVE BIT 6 INTO BIT 0
2351 APTIO1 /TEST ALL 16 DEVICES?
2352 APTIO1 /NO, ONLY EVERY OTHER ONE
2353 APTIO4 /IS ERROR INDICATOR SET ON THIS ONE
2354 SKP /NO, JUST EXIT ROUTINE
2355 JMP NEWDEV+1
2356 JMP I NEWDEV /EXIT
2357 /
2358 /
2359 /ROUTINE TO NOTIFY APT THAT THE PROGRAM IS RUNNING.
2360 /
2361 XTICK, 0
2362 TAD 22
2363 AND K4000
2364 SNA CLA /ON APT??
2365 JMP I XTICK /NO, EXIT.
2366 ISZ CLKCNT /READY.
2367 JMP I XTICK /NO, EXIT.
2368 TAD COUNT /INIT COUNTER
2369 DCA CLKCNT
2370 IOF
2371 CDF
2372 CIF 70
2373 JMS I K6500
2374 JMP I XTICK
2375 /
2376 4031 6500 K6500, 6500
2377 /
2378 /SETS UP LOOP COUNTER AND LOOP TEST POINTER.
2379 /
2380 XPCRET, 0
2381 CLA CMA
```

```
2382 4034 1232 TAD XPCRET
2383 4035 3046 DCA LOOPPC
2384 4036 1241 TAD M10
2385 4037 3052 DCA TSTCNT
2386 4040 5632 JMP I XPCRET
2387 /
2388 4041 7770 M10, -10
2389 /
2390 /LOOP ON TEST IF SR2=1
2391 /
2392 /
2393 4042 0000 XSR2, 0
2394 4043 4517 LAS
2395 4044 7006 RTL
2396 4045 7710 SPA CLA
2397 4046 5446 JMP I LOOPPC
2398 4047 5642 JMP I XSR2
2399 /
2400 4050 0000 NOT8E, 0 /IF SR10=1 PROCESSOR NOT PDP-8E FAMILY
2401 4051 4517 LAS
2402 4052 7012 RTR
2403 4053 7010 RAR
2404 4054 7710 SPA CLA
2405 4055 5260 JMP ,+3
2406 4056 2250 ISZ NOT8E
2407 4057 5650 JMP I NOT8E
2408 4060 1650 TAD I NOT8E
2409 4061 3250 DCA NOT8E
2410 4062 5650 JMP I NOT8E
2411 /
2412 /
2413 /ROUTINE TO PERFORM A LAS.
2414 /
2415 /
2416 4063 0000 XLAS, 0
2417 4064 7300 CLL CLA
2418 4065 1021 TAD 21 /GET HCW1
2419 4066 7710 SPA CLA /USE HARDWARE SWITCHES?
2420 4067 7614 TAD 7614 /LAS AND A SKP
2421 4070 1020 TAD 20 /GET PSR
2422 4071 5663 JMP I XLAS
2423 /
2424 /
2425 /PROGRAMED HALT INSTRUCTION. IF ON APT THIS ROUTINE IS A NOP.
2426 /
2427 /
2428 4072 0000 XHALT, 0
2429 4073 3263 DCA XLAS /STORE THE AC IF ANY.
2430 4074 1022 TAD 22
2431 4075 0123 AND K4000
2432 4076 7650 SNA CLA
2433 4077 5301 JMP XHLT /PERFORM HALT FUNCTION
2434 4100 5672 JMP I XHALT
2435 /
2436 4101 4515 XHLT, CHEK22 /TEST FOR ACTIVE CONSOLE
```

```
2437 4102 5672 JMP I XHALT /IGNORE HALT INSTRUCTION.
2438 4103 1263 TAD XLAS
2439 4104 7440 SZA
2440 4105 5310 JMP ,+3
2441 4106 7340 CLL CLA CMA /SET AC TO -1
2442 4107 1272 TAD XHALT /GET ERROR PC
2443 4110 7402 7402
2444 4111 7300 CLL CLA /CLEAR OUT ANYTHING IN AC.
2445 4112 5672 JMP I XHALT
2446
2447 /
2448 /
2449 /THIS ROUTINE TESTS FOR ACTIVE CONSOLE PACKAGE, IF CONSOLE
2450 /ACTIVE INSTRUCTION FOLLOWING CALL IS PERFORMED.
2451 4113 0000 XCHK22, 0
2452 4114 7300 CLL CLA
2453 4115 1022 TAD 22
2454 4116 0322 AND K400
2455 4117 7650 SNA CLA /CONSOLE ACTIVE?
2456 4120 2313 ISZ XCHK22 /NO, UPDATE RETURN
2457 4121 5713 JMP I XCHK22
2458
2459 4122 0400 K400, 400
2460 /
2461 /RANDOM1- GETS A RANDOM NUMBER (PUT IN XMTDAT) AND PUTS IT IN ITS
2462 /PROPER LOCATION WHILE SAVING THE CONTENTS OF XMTDAT
2463 /
2464 4123 0000 XRAND1, 0
2465 4124 7300 CLA CLL
2466 4125 1042 TAD XMTDAT /GET FIRST DATA WORD
2467 4126 3335 DCA TEMSAV /AND SAVE IT
2468 4127 4506 RANDOM /GET THE SECOND DATA WORD
2469 4130 1042 TAD XMTDAT /TRANSFER IT TO ITS PROPER LOCATION
2470 4131 3043 DCA XMTDT1
2471 4132 1335 TAD TEMSAV /GET THE FIRST DATA WORD OUT OF ITS
2472 4133 3042 DCA XMTDAT /TEMP STORAGE AND RESTORE IT TO ITS PROPER LOCATION
2473 4134 5723 JMP I XRAND1
2474
2475 4135 0000 TEMSAV, 0
2476 /
2477 PAGE
2478 /
2479 /
2480 /MESSAGES
2481 4200 4320 MPC, TEXT "#PC= "
2482 4201 0375
2483 4202 4000
2484 4203 4322 MFSS1, TEXT "#RECEIVE IOT? "
2485 4204 0503
2486 4205 0511
2487 4206 2605
2488 4207 4011
2489 4210 1724
2490 4211 7740
2491 4212 4000
```

```
2483 4213 4324 MESS2, TEXT "#TRANSMIT IOT? "
2484 4214 2201
2485 4215 1623
2486 4216 1511
2487 4217 2440
2488 4220 1117
2489 4221 2477
2490 4222 4040
2491 4223 0000
2492 4224 4320 MESS3, TEXT "#PARITY(Y OR N)? "
2493 4225 0122
2494 4226 1124
2495 4227 3150
2496 4230 3140
2497 4231 1722
2498 4232 4016
2499 4233 5177
2500 4234 4040
2501 4235 0000
2502 4236 4040 MESS3A, TEXT " NP="
2503 4237 1620
2504 4240 7500
2505 4241 7743 MESS4, TEXT "?*EVEN PARITY EVN=0? ODD PARITY EVN=1?*STATUS ENABLED(Y OR N)? "
2506 4242 0526
2507 4243 0516
2508 4244 4020
2509 4245 0122
2510 4246 1124
2511 4247 3140
2512 4250 0526
2513 4251 1675
2514 4252 6077
2515 4253 4040
2516 4254 1704
2517 4255 0440
2518 4256 2001
2519 4257 2211
2520 4260 2431
2521 4261 4005
2522 4262 2616
2523 4263 7561
2524 4264 7743
2525 4265 2324
2526 4266 0124
2527 4267 2523
2528 4270 4005
2529 4271 1601
2530 4272 0214
2531 4273 0504
2532 4274 5031
2533 4275 4017
2534 4276 2240
2535 4277 1651
2536 4300 7740
2537 4301 4000
```

2487 4302 4040 MESS4A, TEXT " SWD="

4303 2327

4304 0475

4305 0000

2488 4306 7743 MESS5, TEXT "?FILLER CHARACTERS(Y OR N)? "

4307 0611

4310 1414

4311 0522

4312 4003

4313 1001

4314 2201

4315 0324

4316 0522

4317 2350

4320 3140

4321 1722

4322 4016

4323 5177

4324 4040

4325 0000

2489 4326 4040 MESS5A, TEXT " FIL="

4327 0611

4330 1475

4331 0000

2490 4400 PAGE

2491

2492 4400 MESS6, TEXT "?BAUD RATE(00-13)? 00=110 01=150 02=300 03=600 04=1200 05=2400"

4401 0201

4402 2504

4403 4022

4404 0124

4405 0550

4406 6060

4407 5561

4410 6351

4411 7740

4412 6060

4413 7561

4414 6160

4415 4060

4416 6175

4417 6165

4420 6040

4421 6062

4422 7563

4423 6060

4424 4060

4425 6375

4426 6660

4427 6040

4430 6064

4431 7561

4432 6260

4433 6040

4434 6065

4435 7562

4436 6460

4437 6000

2493 4440 4360 CMESS6, TEXT "#06=4800 07=9600 10=19,200 11=56.8 12=66.7 13=1050 "

4441 6675

4442 6470

4443 6060

4444 4060

4445 6775

4446 7166

4447 6060

4450 4061

4451 6075

4452 6171

4453 5462

4454 6060

4455 4061

4456 6175

4457 6566

4460 5670

4461 4061

4462 6275

4463 6666

4464 5667

4465 4061

4466 6375

4467 6160

4470 6560

4471 4040

4472 0000

2494

2495 4473 4361 MESS6A, TEXT "#110 BAUD = B1=0? B2=0? B3=0? W2=1? W5=0?"

4474 6160

4475 4002

4476 0125

4477 0440

4500 5540

4501 0261

4502 7560

4503 7740

4504 0262

4505 7560

4506 7740

4507 0263

4510 7560

4511 7740

4512 2762

4513 7561

4514 7740

4515 2765

4516 7560

4517 7700

2496 4520 4361 MESS6B, TEXT "#150 BAUD = B1=0? B2=0? B3=1? W2=1? W5=0?"

4521 6560

4522 4002

4523 0125
4524 0440
4525 5540
4526 0261
4527 7560
4530 7740
4531 0262
4532 7560
4533 7740
4534 0263
4535 7561
4536 7740
4537 2762
4540 7561
4541 7740
4542 2765
4543 7560
4544 7700
2497 4545 4363 MESS6C, TEXT "#300 BAUD - B1=0? B2=1? B3=0? W2=1? W5=0?"
4546 6060
4547 4002
4550 0125
4551 0440
4552 5540
4553 0261
4554 7560
4555 7740
4556 0262
4557 7561
4560 7740
4561 0263
4562 7560
4563 7740
4564 2762
4565 7561
4566 7740
4567 2765
4570 7560
4571 7700
2498 4572 4366 MESS6D, TEXT "#600 BAUD - B1=0? B2=1? B3=1? W2=1? W5=0?"
4573 6060
4574 4002
4575 0125
4576 0440
4577 5540
4600 0261
4601 7560
4602 7740
4603 0262
4604 7561
4605 7740
4606 0263
4607 7561
4610 7740
4611 2762

4612 7561
4613 7740
4614 2765
4615 7560
2499 4616 7700 MESS6E, TEXT "#1200 BAUD - B1=1? B2=0? B3=0? W2=1? W5=0?"
4617 4361
4620 6260
4621 6040
4622 0201
4623 2504
4624 4055
4625 4002
4626 6175
4627 6177
4630 4002
4631 6275
4632 6077
4633 4002
4634 6375
4635 6077
4636 4027
4637 6275
4640 6177
4641 4027
4642 6575
4643 6077
2500 4644 0000 MESS6F, TEXT "#2400 BAUD - B1=1? B2=0? B3=1? W2=1? W5=0?"
4645 4362
4646 6460
4647 6040
4650 0201
4651 2504
4652 4055
4653 4002
4654 6175
4655 6177
4656 4002
4657 6275
4660 6077
4661 4002
4662 6375
4663 6177
4664 4027
4665 6275
4666 6177
4667 4027
4670 6575
4671 6077
2501 4672 0000 MESS6G, TEXT "#4800 BAUD - B1=1? B2=1? B3=0? W2=1? W5=0?"
4673 4364
4674 7060
4675 6040
4676 0201
4677 2504
4700 4055

4701 4002
4702 6175
4703 6177
4704 4002
4705 6275
4706 6177
4707 4002
4710 6375
4711 6077
4712 4027
4713 6275
4714 6177
4715 4027
4716 6575
4717 6077
4720 0000
2502 4721 4371 MESS6H, TEXT "#9600 BAUD - B1=1? B2=1? B3=1? W2=1? W5=0?"
4722 6660
4723 6040
4724 0201
4725 2504
4726 4055
4727 4002
4730 6175
4731 6177
4732 4002
4733 6275
4734 6177
4735 4002
4736 6375
4737 6177
4740 4027
4741 6275
4742 6177
4743 4027
4744 6575
4745 6077
4746 0000
2503
2504 4747 4361 MESS6I, TEXT "#19,200 BAUD - B1=1? B2=1? B3=1? W2=0? W5=1?"
4750 7154
4751 6260
4752 6040
4753 0201
4754 2504
4755 4055
4756 4002
4757 6175
4760 6177
4761 4002
4762 6275
4763 6177
4764 4002
4765 6375
4766 6177

4767 4027
4770 6275
4771 6077
4772 4027
4773 6575
4774 6177
4775 0000
2505 4776 4365 MESS6J, TEXT "#56.8 BAUD - B1=0? B2=0? B3=0? W2=1? W5=0?"
4777 6656
5000 7040
5001 0201
5002 2504
5003 4055
5004 4002
5005 6175
5006 6077
5007 4002
5010 6275
5011 6077
5012 4002
5013 6375
5014 6077
5015 4027
5016 6275
5017 6177
5020 4027
5021 6575
5022 6077
5023 0000
2506 5024 4366 MESS6K, TEXT "#66.7 BAUD - B1=0? B2=0? B3=0? W2=1? W5=0?"
5025 6656
5026 6740
5027 0201
5030 2504
5031 4055
5032 4002
5033 6175
5034 6077
5035 4002
5036 6275
5037 6077
5040 4002
5041 6375
5042 6077
5043 4027
5044 6275
5045 6177
5046 4027
5047 6575
5050 6077
5051 0000
2507 5052 4361 MESS6L, TEXT "#1050 BAUD - B1=1? B2=0? B3=0? W2=1? W5=0?"
5053 6065
5054 6040
5055 0201

5056 2504
5057 4055
5060 4002
5061 6175
5062 6177
5063 4002
5064 6275
5065 6077
5066 4002
5067 6375
5070 6077
5071 4027
5072 6275
5073 6177
5074 4027
5075 6575
5076 6077
5077 0000
2508 5100 4324 MESS7, TEXT "#TWO STOP BITS(Y OR N)? "
5101 2717
5102 4023
5103 2417
5104 2040
5105 0211
5106 2423
5107 5031
5110 4017
5111 2240
5112 1651
5113 7740
5114 4000
2509 5115 4040 MESS7A, TEXT " SB="
5116 2302
5117 7500
2510 5120 7743 MESS10, TEXT "?#DATA BITS/CHARACTER? 0=5 1=6 2=7 3=8 "
5121 0401
5122 2401
5123 4002
5124 1124
5125 2357
5126 0310
5127 0122
5130 0103
5131 2405
5132 2277
5133 4060
5134 7565
5135 4061
5136 7566
5137 4062
5140 7567
5141 4063
5142 7570
5143 4040
5144 0000

2511
2512 5145 4365 MESS10A, TEXT "#5 DATA BITS - NB1=1? NB2=1?#"
5146 4004
5147 0124
5150 0140
5151 0211
5152 2423
5153 4055
5154 4016
5155 0261
5156 7561
5157 7740
5160 1602
5161 6275
5162 6177
5163 4300
2513 5164 4366 MESS10B, TEXT "#6 DATA BITS - NB1=0? NB2=1?#"
5165 4004
5166 0124
5167 0140
5170 0211
5171 2423
5172 4055
5173 4016
5174 0261
5175 7560
5176 7740
5177 1602
5200 6275
5201 6177
5202 4300
2514 5203 4367 MESS10C, TEXT "#7 DATA BITS - NB1=1? NB2=0?#"
5204 4004
5205 0124
5206 0140
5207 0211
5210 2423
5211 4055
5212 4016
5213 0261
5214 7561
5215 7740
5216 1602
5217 6275
5220 6077
5221 4300
2515 5222 4370 MESS10D, TEXT "#8 DATA BITS - NB1=0? NB2=0?#"
5223 4004
5224 0124
5225 0140
5226 0211
5227 2423
5230 4055
5231 4016
5232 0261

	5233	7560	
	5234	7740	
	5235	1602	
	5236	6275	
	5237	6077	
	5240	4300	
2516	5241	0320	RECPNT, KCF0
2517	5242	0401	KCF1
2518	5243	0501	KCF2
2519	5244	0551	KCF3
2520	5245	1143	KCF4
2521	5246	0276	KSF0
2522	5247	0336	KSF1
2523	5250	0354	KSF2
2524	5251	0471	KSF3
2525	5252	0537	KSF4
2526	5253	0600	KSF5
2527	5254	0720	KSF6
2528	5255	0733	KSF7
2529	5256	2713	KSF8
2530	5257	1013	KSF9
2531	5260	1037	KSF10
2532	5261	1065	KSF11
2533	5262	1077	KSF12
2534	5263	1116	KSF14
2535	5264	1137	KSF16
2536	5265	1144	KSF17
2537	5266	1425	KSF18
2538	5267	1523	FDTLOP
2539	5270	1541	KSF19
2540	5271	1675	KSF21
2541	5272	1744	KSF22
2542	5273	1762	KSF23
2543	5274	1341	KSF24
2544	5275	2036	KSF25
2545	5276	2063	KSF26
2546	5277	0607	KCC0
2547	5300	0712	KCC1
2548	5301	1036	KCC2
2549	5302	1063	KCC3
2550	5303	1136	KCC4
2551	5304	1757	KCC5
2552	5305	2000	KCC6
2553	5306	0625	KRS0
2554	5307	1010	KRS1
2555	5310	2040	KRS2
2556	5311	2046	KRS3
2557	5312	1335	KSE0
2558	5313	2023	KSE1
2559	5314	2045	KSE2
2560	5315	2054	KSE3
2561	5316	0403	KIE0
2562	5317	0422	KIE1
2563	5320	0457	KIE2
2564	5321	0502	KIE3

2565	5322	0536	KIE4
2566	5323	0615	KIE5
2567	5324	0707	KIE6
2568	5325	0747	KIE7
2569	5326	0757	KIE8
2570	5327	1026	KIE9
2571	5330	1052	KIE10
2572	5331	0634	KRB0
2573	5332	1113	KRB1
2574	5333	1410	KRB2
2575	5334	1437	KRB3
2576	5335	1504	KRB4
2577	5336	1511	KRB5
2578	5337	1526	KRB6
2579	5340	1544	KRB7
2580	5341	1615	KRB8
2581	5342	1654	KRB9
2582	5343	1711	KRB10
2583	5344	2313	INTON
2584	5345	1351	KRB11
2585	5346	2014	KRB12
2586	5347	2060	KRB13
2587	5350	2070	KRB14
2588		/	
2589		/	
2590	5351	0000	0000
2591			
2592			
2593	5352	0323	XMTIOT, TFL0
2594	5353	0345	TFL1
2595	5354	0406	TFL2
2596	5355	0452	TFL3
2597	5356	0503	TFL4
2598	5357	0520	TFL5
2599	5360	0645	TFL6
2600	5361	0301	TSF0
2601	5362	0326	TSF1
2602	5363	0333	TSF2
2603	5364	0346	TSF3
2604	5365	0351	TSF4
2605	5366	0407	TSF5
2606	5367	0425	TSF6
2607	5370	0437	TSF7
2608	5371	0453	TSF8
2609	5372	0504	TSF9
2610	5373	0512	TSF10
2611	5374	0521	TSF11
2612	5375	0533	TSF12
2613	5376	0663	TSF13
2614	5377	0723	TSF14
2615	5400	2675	TSF15
2616	5401	1000	TSF16
2617	5402	1003	TSF17
2618	5403	1070	TSF18
2619	5404	1106	TSF20

2620	5405	1124	TSF22
2621	5406	1421	TSF23
2622	5407	1516	TSF24
2623	5410	1535	TSF25
2624	5411	1553	TSF26
2625	5412	1672	TSF28
2626	5413	1747	TSF29
2627	5414	2310	TSF30
2628	5415	2322	TSF31
2629	5416	1344	TSF32
2630	5417	0330	TCF0
2631	5420	0436	TCF1
2632	5421	0470	TCF2
2633	5422	0532	TCF3
2634	5423	0654	TCF4
2635	5424	0715	TCF5
2636	5425	1002	TCF6
2637	5426	1064	TCF7
2638	5427	1123	TCF8
2639	5430	1412	TCF9
2640	5431	1505	TCF10
2641	5432	1512	TCF11
2642	5433	1611	TCF12
2643	5434	1656	TCF13
2644	5435	1346	TCF14
2645	5436	2015	TCF15
2646	5437	2035	TCF16
2647	5440	0726	TPC0
2648	5441	1073	TPC1
2649	5442	0304	SPI0
2650	5443	0411	SPI1
2651	5444	0427	SPI2
2652	5445	0442	SPI3
2653	5446	0455	SPI4
2654	5447	0460	SPI5
2655	5450	0506	SPI6
2656	5451	0515	SPI7
2657	5452	0523	SPI8
2658	5453	0672	SPI9
2659	5454	0736	SPI10
2660	5455	0750	SPI11
2661	5456	0760	SPI12
2662	5457	1015	SPI13
2663	5460	1027	SPI14
2664	5461	1042	SPI15
2665	5462	1102	TL50
2666	5463	1420	XMIT
2667	5464	1515	TL51
2668	5465	1522	TL52
2669	5466	1540	TL53
2670	5467	1556	TL54
2671	5470	1606	TL55
2672	5471	1665	TL56
2673	5472	2307	TL57
2674	5473	2315	TL58

2675	5474	2026	TL59
2676	5475	1424	TL510
2677	5476	1340	SLWTL5
2678			/
2679			/
2680	5477	0000	0000
2681			/BAUD RATE TABLES FOR USE ONLY UNDER APT-8 TEST SYSTEM.
2682			/
2683	5500	5514	TABLE1, B110
2684	5501	5523	B150
2685	5502	5532	B300
2686	5503	5541	B600
2687	5504	5550	B1200
2688	5505	5557	B2400
2689	5506	5566	B4800
2690	5507	5575	B9600
2691	5510	5604	B19K2
2692	5511	5613	B56P8
2693	5512	5622	B66P7
2694	5513	5631	B1050
2695			/
2696			/ACTUAL VALUES TO USE FOR THE APT TIMING SECTION.
2697			/
2698	5514	7677	B110, -101
2699	5515	7750	-30
2700	5516	7765	-13
2701	5517	7771	-7
2702	5520	7765	-13
2703	5521	7765	-13
2704	5522	7770	-10
2705			/150 BAUD
2706			/
2707	5523	7677	B150, -101
2708	5524	7750	-30
2709	5525	7760	-20
2710	5526	7765	-13
2711	5527	7761	-17
2712	5530	7761	-17
2713	5531	7770	-10
2714			/300 BAUD
2715			/
2716	5532	7677	B300, -101
2717	5533	7750	-30
2718	5534	7740	-40
2719	5535	7752	-26
2720	5536	7744	-34
2721	5537	7744	-34
2722	5540	7770	-10
2723			/
2724			/600 BAUD
2725	5541	7647	B600, -131
2726	5542	0000	0
2727	5543	7700	-100
2728	5544	7724	-54
2729	5545	7700	-100

```
2730 5546 7700 -100
2731 5547 0000 0
2732 /
2733 /1200 BAUD
2734 5550 7647 B1200, -131
2735 5551 0000 0
2736 5552 7600 -200
2737 5553 7650 -130
2738 5554 7600 -200
2739 5555 7600 -200
2740 5556 0000 0
2741 /2400 BAUD
2742 /
2743 5557 7247 B2400, -531
2744 5560 0000 0
2745 5561 0000 0
2746 5562 0000 0
2747 5563 7400 -400
2748 5564 7400 -400
2749 5565 0000 0
2750 /
2751 /4800 BAUD
2752 5566 7247 B4800, -531
2753 5567 0000 0
2754 5570 0000 0
2755 5571 0000 0
2756 5572 7400 -400
2757 5573 7400 -400
2758 5574 0000 0
2759 /
2760 /9600 BAUD
2761 5575 6247 B9600, -1531
2762 5576 0000 0
2763 5577 0000 0
2764 5600 0000 0
2765 5601 0000 0
2766 5602 0000 0
2767 5603 0000 0
2768 /
2769 /19.2 KILO BAUD.
2770 5604 6247 B19K2, -1531
2771 5605 0000 0
2772 5606 0000 0
2773 5607 0000 0
2774 5610 0000 0
2775 5611 0000 0
2776 5612 0000 0
2777 /
2778 /56.8 BAUD.
2779 5613 7740 R56P8, -40
2780 5614 7763 -15
2781 5615 7772 -6
2782 5616 7775 -3
2783 5617 7772 -6
2784 5620 7772 -6
```

```
2785 5621 7773 -5
2786 /
2787 /66.7 BAUD
2788 5622 7740 R66P7, -40
2789 5623 7763 -15
2790 5624 7772 -6
2791 5625 7775 -3
2792 5626 7772 -6
2793 5627 7772 -6
2794 5630 7773 -5
2795 /
2796 /1050 BUAD
2797 5631 7647 B1050, -131
2798 5632 0000 0
2799 5633 7600 -200
2800 5634 7650 -130
2801 5635 7600 -200
2802 5636 7600 -200
2803 5637 0000 0
2804 /
2805 /STANDARD CONSOLE PACKAGE.
2806 /THE FOLLOWING PARAMETERS MUST BE DEFINED PRIOR TO
2807 /ASSEMBLY.
2808 /THSF LD THIS DEFINES THE FIELD THE ROUTINE
2809 / WILL RESIDE IN
2810 /CALFLD THIS DEFINES CALLING FIELD.
2811 /ORIGIN STARTING ADDRESS OF ROUTINES
2812 /STRST DEFINES RESTART TEST FOR LINE FEED DETECTION.
2813 /
2814 /
2815 0000 THSF LD=0 /FIELD 0
2816 0000 CALFLD=00 /CALLING FIELD
2817 6600 ORIGIN=6600 /START AT ADDRESS 6600
2818 0000 CONFLD=THSF LD*10
2819 0000 PRGFLD=CALFLD*10
2820 0271 STRST=CLRB RD
2821
2822 0170 3306
2823 0171 3513
2824 0172 7700
2825 0173 1600
2826 0174 1400
2827 0175 2646
2828 0176 4060
2829 0177 7400
2830 0000 FIELD THSF LD
```

[illegible][illegible]

```
2831      6600      *ORIGIN
2832      /CONTROL CHARACTER DECODE ROUTINE.
2833      /ENTER WITH CHARACTER IN THE AC.
2834      XC8CNT, 0
2835      6601      3360      DCA      C8CHAR      /STORE CHARACTER.
2836      6602      6214      RDF          /GET CALLING DATA FIELD.
2837      6603      1257      TAD      KCDFIF      /SET UP RETURN FIELD.
2838      6604      3241      DCA      FLDRTN      /STORE IT
2839      6605      3252      DCA      INDEXA      /SET DISPLACEMENT TO 0
2840      6606      1254      TAD      XTABLA
2841      6607      3253      DCA      GETDAT      /SET UP POINTER TABLE.
2842      6610      6201      CDF      CONFLD
2843      6611      1653      REDOA, TAD I      GETDAT      /GET CONTROL CHARACTER
2844      6612      7450      SNA          /END OF TABLE.
2845      6613      5222      JMP      DONEA      /YES.
2846      6614      1360      TAD      C8CHAR
2847      6615      7650      SNA CLA          /GET A MATCH?
2848      6616      5243      JMP      GOITA      /YES.
2849      6617      2252      ISZ          /NO, UPDATE DISPLACEMENT
2850      6620      2253      ISZ      GETDAT      /UPDATE POINTER.
2851      6621      5211      JMP      REDOA      /GET NEXT COMPARE.
2852      6622      1360      DONEA, TAD      C8CHAR      /GET BACK CHARACTER.
2853      6623      1777      TAD      C8M260
2854      6624      7700      SNA CLA
2855      6625      5234      JMP      ,+7      /NO, THEN JUST PRINT IT.
2856      6626      1360      TAD      C8CHAR
2857      6627      4776      JMS      TSTCRL      /GO TEST FOR CRLF.
2858      6630      5236      JMP      FLDRTN=3      /CRLF FOUND.
2859      6631      1775      TAD      ARROW
2860      6632      4774      JMS      XC8TYP      /PRINT "
2861      6633      1256      TAD      C80100      /SET UP CHARACTER
2862      6634      1360      TAD      C8CHAR
2863      6635      4774      JMS      XC8TYP      /PRINT IT.
2864      6636      1260      TAD      C8K277
2865      6637      4774      JMS      XC8TYP      /PRINT ?
2866      6640      4773      JMS      XC8CRL      /AND <CRLF>
2867      6641      6203      FLDRTN, CIF CDF PRGFLD      /RETURN FIELD.
2868      6642      5600      JMP I      XC8CNT      /EXIT BACK TO CALL.
2869      /
2870      6643      1255      GOITA, TAD      XTABL8      /GET CONTROL CHARACTER ROUTINES.
2871      6644      1252      TAD      INDEXA      /ADD IN OFFSET
2872      6645      3251      DCA      GOTOA      /STORE THIS
2873      6646      1651      TAD I      GOTOA      /GET ACTUAL POINTER.
2874      6647      3251      DCA      GOTOA
2875      6650      5651      JMP I      GOTOA      /PERFORM FUNCTION.
2876      /
2877      /CONSTANTS USED BY THIS ROUTINE.
2878      /
2879      6651      0000      GOTOA, 0
2880      6652      0000      INDEXA, 0
2881      6653      0000      GETDAT, 0
2882      6654      7254      XTABLA, TABLA
2883      6655      7261      XTABL8, TABL8
2884      /
2885      /
```

```
2886      6656      0100      C80100, 100
2887      6657      6203      KCDFIF, CDF CIF
2888      6660      0277      C8K277, 277
2889      /
2890      /ABORT PRINT OUT ROUTINE.
2891      /ALL TERMINAL OUTPUT IS STOPPED UNTIL A "Q" IS INPUT
2892      /ALL OTHER CONTROL FUNCTIONS CAN BE PERFORMED.
2893      /
2894      6661      1302      CNTRL8, TAD      INMODE
2895      6662      7640      SZA CLA          /HAS ROUTINE BEEN ENTERED BEFORE.
2896      6663      5272      JMP      ,+7
2897      6664      1200      TAD      XC8CNT
2898      6665      3301      DCA      C8RTRN      /STORE RETURN ADDRESS.
2899      6666      7240      STA
2900      6667      3302      DCA      INMODE      /SET UP FOR A REENTRY.
2901      6670      4772      JMS      XC8TTY      /GO BACK AND WAIT FOR NEXT CHARACTER.
2902      6671      4200      JMS      XC8CNT      /DECODE FOR CONTROL FUNCTION.
2903      6672      5270      JMP      ,+2      /NO CONTROL FUNTION, GO BACK AND WAIT.
2904      /
2905      /CONTROL Q FUNCTION.
2906      /CLEARS INMODE POINTER, AND SETS UP PROPER RETURN
2907      /TO CALLING FIELD.
2908      /
2909      6673      1302      CNTRL8, TAD      INMODE
2910      6674      7650      SNA CLA          /HAS CONTOL S BEEN TYPED.?
2911      6675      5241      JMP      FLDRTN      /NO.
2912      6676      3302      DCA      INMODE      /CLEAR REENTRY POINTER
2913      6677      6203      ACTFLD, CIF CDF PRGFLD
2914      6700      5701      JMP I      C8RTRN      /AND TAKE A RETURN
2915      /
2916      6701      0000      C8RTRN, 0000      /GETS SETUP TO RETURN ADDRESS.
2917      6702      0000      INMODE, 0000
2918      /
2919      /LOADS PSEUDO-SWITCH REGISTER WITH NEW VALUE.
2920      /
2921      6703      0000      XC8PSW, 0
2922      6704      7300      CLA CLL
2923      6705      6214      RDF          /GET CLING DATA FIELD
2924      6706      1257      TAD      KCDFIF
2925      6707      3347      DCA      PSWRTN      /SET UP RETURNING FIELD
2926      6710      6201      CDF      CONFLD      /SET DATA FIELD TO FIELD OF CONSOLE.
2927      6711      7200      CLA
2928      6712      3357      DCA      CHRTWP
2929      6713      4773      JMS      XC8CRL      /<CRLF>
2930      6714      4771      JMS      XC8PNT
2931      6715      7265      MESA
2932      6716      6201      CDF      PRGFLD      /PRINT "SR="
2933      6717      1770      TAD I      (20      /SET UP FIELD OF PSW.
2934      6720      6201      CDF      CONFLD      /GET PSEUDO SWITCH REGISTER.
2935      6721      3767      DCA      C8STOR      /RESTORE FELDS.
2936      6722      1767      TAD      C8STOR
2937      6723      4766      JMS      XC8OCT      /PRINT THE CURRENT VALUE.
2938      6724      1356      TAD      C8M4
2939      6725      3765      DCA      TMCNT
2940      6726      1764      TAD      C8K240      /SET UP TO INPUT FOUR CHARACTERS.
```

```
2941 6727 4774 JMS XC8TYP /PRINT A SPACE.
2942 6730 4772 JMS XC8TTY /GET A CHARACTER.
2943 6731 4763 JMS TSTCHA /CHECK FOR A VALID CHARACTER.
2944 6732 5304 JMP XC8PSW+1 /ERROR RETURN.
2945 6733 3360 DCA C8CHAR
2946 6734 1357 TAD CHRTMP
2947 6735 7106 CLL RTL
2948 6736 7004 RAL /MOVE OVER THREE BITS
2949 6737 1360 TAD C8CHAR
2950 6740 3357 DCA CHRTMP /AND STORE IT.
2951 6741 2765 ISZ TPCNT /DONE YET.
2952 6742 5330 JMP =12 /NO, GET NEXT VALUE.
2953 6743 4773 JMS XC8CRL
2954 6744 6201 TSTRN, CDF PRGFLD
2955 6745 1357 TAD CHRTMP /GET NEW SWITCH VALUE
2956 6746 3770 DCA I (20 /AND STORE IT IN PROGRAM FIELD
2957 6747 0000 PSWRTN, 0000
2958 6750 2354 ISZ LFFLAG /WAS A LINE FEED FOUND.
2959 6751 5703 JMP I XC8PSW
2960 6752 6203 CIF CDF PRGFLD /RETURN TO PROGRAM FIELD.
2961 6753 5755 JMP I RSTART /GO RESTART TESTING.
2962
2963 6754 0000 LFFLAG, 0
2964 6755 0271 RSTART, STRIST /STARTING TEST OF DIAGNOSTIC.
2965 6756 7774 C8M4, -4
2966 6757 0000 CHRTMP, 0
2967 6760 0000 C8CHAR, 0
2968
2969 6763 7022
2970 6764 7065
2971 6765 7066
2972 6766 7215
2973 6767 7075
2974 6770 0020
2975 6771 7076
2976 6772 7135
2977 6773 7146
2978 6774 7126
2979 6775 7011
2980 6776 7200
2981 6777 7072
2982 7000 PAGE
2983 /
2984 /CONTROL C FUNCTION.
2985 /ROUTINE CAUSES RETURN TO MONITOR IF ON AN
2986 /OPERATING SYSTEM OR TO THE LOADER IF NOT.
2987 7000 1211 CNTRLC, TAD ARROW
2988 7001 4326 JMS XC8TYP /PRINT AN ARROW
2989 7002 1210 TAD C
2990 7003 4326 JMS XC8TYP /AND A "C"
2991 7004 4346 JMS XC8CRL /NOW A <CRLF>
2992 7005 6203 CIF CDF /LOADER MUST BE IN FIELD 0
2993 7006 5607 JMP I K7600
2994 /
```

```
2995 /
2996 7007 7600 K7600, 7600
2997 7010 0303 C, "C
2998 7011 0336 ARROW, "
2999 7012 0307 G, "G
3000
3001 /CONTROL G FUNCTION
3002 /PRINT "SP=XXXX" WHERE XXXX IS THE CURRENT VALUE OF
3003 /THE PSEUDO-SWITCH REGISTER, IT THE WAITS FOR A NEW SETTING.
3004 /
3005 7013 1211 CNTRLG, TAD ARROW
3006 7014 4326 JMS XC8TYP /PRINT "
3007 7015 1212 TAD G
3008 7016 4326 JMS XC8TYP /AND A "G
3009 7017 4346 JMS XC8CRL /AND FINALLY A CRLF
3010 7020 4777 XCNTLG, JMS XC8PSW /GO GET NEW SWITCH VALUE.
3011 7021 5776 JMP FLDRTN /EXIT ROUTINE.
3012
3013 /THIS ROUTINE WILL VERIFY CHARACTER IS A VALID OCTAL DIGIT
3014 /OR A LINE FEED OR CARRIAGE RETURN.
3015 /
3016 7022 0000 TSTCHA, 0
3017 7023 3274 DCA C8TEMP
3018 7024 1274 TAD C8TEMP
3019 7025 1375 TAD (-203 /TEST FOR CONTROL C
3020 7026 7650 SNA CLA /WAS IT?
3021 7027 5200 JMP CNTRLC /YES.
3022 7030 1274 TAD C8TEMP
3023 7031 4774 JMS TSTCRL /TEST FOR <CRLF>
3024 7032 5251 JMP TSTEXT /FOUND ONE
3025 7033 1274 TAD C8TEMP
3026 7034 4326 JMS XC8TYP
3027 7035 1274 TAD C8TEMP
3028 7036 1272 TAD C8M260
3029 7037 7710 SPA CLA /WAS GREATER THAN 0
3030 7040 5261 JMP ERREXT /NO, TAKE ERROR EXIT.
3031 7041 1274 TAD C8TEMP
3032 7042 1273 TAD C8M270
3033 7043 7700 SNA CLA /WAS 7 OR LESS
3034 7044 5261 JMP ERREXT /NO TAKE ERROR EXIT.
3035 7045 1274 TAD C8TEMP
3036 7046 0271 AND C8K7
3037 7047 2222 ISZ TSTCHA /UPDATE RETURN FOR VALID CHARACTER.
3038 7050 5622 JMP I TSTCHA
3039 7051 4346 TSTEXT, JMS XC8CRL
3040 7052 7307 CLL CLA IAC RTL /+4 TO AC
3041 7053 1266 TAD TPCNT
3042 7054 7640 SZA CLA /CHANGE SWITCH REGISTER.
3043 7055 5773 JMP TSTRN /GET BACK TO CALL.
3044 7056 1275 TAD C8STOR
3045 7057 3772 DCA CHRTMP
3046 7060 5773 JMP TSTRN
3047 7061 1771 ERREXT, TAD C8K277
3048 7062 4326 JMS XC8TYP
3049 7063 4346 JMS XC8CRL /AND A <CRLF>
```

```
3050 7064 5622 JMP I TSTCHA /PRINT A "?" AND EXIT.
3051 7065 0240 C8K240, 240
3052 7066 0000 TPCNT, 0
3053 7067 7566 C8M212, -212
3054 7070 7775 C8M3, -3
3055 7071 0007 C8K7, 7
3056 7072 7520 C8M260, -260
3057 7073 7510 C8M270, -270
3058 7074 0000 C8TEMP, 0
3059 7075 0000 C8STOR, 0
3060
3061 /
3062 /MESSAGE TYPE ROUTINE. MESSAGE TO PRINT IS
3063 /CALL +1
3064 /
3065 7076 0000 XC8PNT, 0
3066 7077 7300 CLL CLA
3067 7100 1676 TAD I XC8PNT /GET POINTER TO MESSAGE.
3068 7101 3275 DCA C8STOR
3069 7102 2276 ISZ XC8PNT /UPDATE RETURN.
3070 7103 1675 C8DO1, TAD I C8STOR /GET A SET OF CHARACTERS.
3071 7104 7012 RTR
3072 7105 7012 RTR
3073 7106 7012 RTR
3074 7107 4314 JMS C8DCOD
3075 7110 1675 TAD I C8STOR
3076 7111 4314 JMS C8DCOD
3077 7112 2275 ISZ C8STOR
3078 7113 5303 JMP C8DO1
3079 /
3080 /
3081 /
3082 7114 0000 C8DCOD, 0
3083 7115 0356 AND C8K77
3084 7116 7450 SNA
3085 7117 5676 JMP I XC8PNT /END OF TEXT.
3086 7120 1357 TAD C8M40 /YES.
3087 7121 7510 SPA
3088 7122 1360 TAD C8K100
3089 7123 1265 TAD C8K240
3090 7124 4326 JMS XC8TYP /ESTABLISHES CHARACTER TO PRINT.
3091 7125 5714 JMP I C8DCOD /NOW PRINT IT.
3092 /
3093 /
3094 /TYPE ROUTINE.
3095 /
3096 7126 0000 XC8TYP, 0
3097 7127 6046 TLS
3098 7130 6041 TSF
3099 7131 5330 JMP , -1
3100 7132 6042 TCF
3101 7133 7200 CLA
3102 7134 5726 JMP I XC8TYP
3103 /
3104 /ROUTINE INPUTS A CHARACTER AND EXITS WITH IT IN THE AC
```

```
3105 /
3106 7135 0000 XC8TTY, 0
3107 7136 6031 KSF
3108 7137 5336 JMP , -1
3109 7140 6036 KRB
3110 7141 0344 AND C8K177
3111 7142 1345 TAD C8K200
3112 7143 5735 JMP I XC8TTY
3113 /
3114 7144 0177 C8K177, 177
3115 7145 0200 C8K200, 200
3116 /
3117 /
3118 7146 0000 XC8CRL, 0
3119 7147 1355 TAD C8K215
3120 7150 4326 JMS XC8TYP
3121 7151 1354 TAD C8K212
3122 7152 4326 JMS XC8TYP
3123 7153 5746 JMP I XC8CRL
3124 /
3125 7154 0212 C8K212, 212
3126 7155 0215 C8K215, 215
3127 /
3128 7156 0077 C8K77, 77
3129 7157 7740 C8M40, -40
3130 7160 0100 C8K100, 100
3131 /
3132 7171 6660
3133 7172 6757
3134 7173 6744
3135 7174 7200
3136 7175 7575
3137 7176 6641
3138 7177 6703
3139 /
3140 7200 0000 PAGE
3141 7201 1777 /
3142 7202 7440 TSTCRL, 0
3143 7203 5207 TAD C8M212
3144 7204 7240 SZA /IS IT A LINE FEED?
3145 7205 3776 JMP , +4 /NO
3146 7206 5600 STA
3147 7207 1775 DCA LFFLAG
3148 7210 7640 JMP I TSTCRL /EXIT TO PROPER POINT
3149 7211 2200 TAD C8M3
3150 7212 3776 SZA CLA /IS IT A CARRIAGE RETURN?
3151 7213 5600 ISZ TSTCRL /NO, UPDATE RETURN
3152 7214 5600 DCA LFFLAG /CLEAR LINEFEED FLAG
3153 /
3154 /
3155 /OCTAL TYPE ROUTINE.
3156 /
3157 7215 0000 XC8OCT, 0
3158 7216 3237 DCA C8OCT /STORE CHARACTER TO BE PRINTED.
```



```
3159 7217 1774' TAD C8M4
3160 7220 3773' DCA TMPCNT /SET UP FOR NUMBER OF SHIFTS,
3161 7221 1237 TAD C8OCT /GET VALUE
3162 7222 7004 RAL
3163 7223 7004 RAL
3164 7224 7006 RTL /MOVE INTO BITS 9-11
3165 7225 3237 DCA C8OCT /DON'T CLEAR LINK,
3166 7226 1237 TAD C8OCT
3167 7227 0772' AND C8K7
3168 7230 1240 TAD C8K260
3169 7231 4771' JMS XC8TYP /NOW TYP DIGIT,
3170 7232 1237 TAD C8OCT /GET BACK CHARACTER
3171 7233 2773' ISZ TMPCNT /DONE ALL FOUR YET.
3172 7234 5223 JMP *-11 /NO, GOBACK AND DO NEXT.
3173 7235 7300 CLL CLA
3174 7236 5615 JMP I XC8OCT /EXIT ROUTINE.
3175
3176 7237 0000 C8OCT, 0
3177 7240 0260 C8K260, 260
3178 7241 0000 XC8PAS, 0
3179 7242 6201 CDF CONFLD
3180 7243 2253 ISZ PASSES
3181 7244 4770' JMS XC8PNT /PRINT END OF PASS MESSAGE
3182 7245 7267 MESB
3183 7246 1253 TAD PASSES /PRINT NUMBER OF PASSES
3184 7247 4215 JMS XC8OCT
3185 7250 4767' JMS XC8CRL
3186 7251 6203 CIF CDF CALFLD
3187 7252 5641 JMP I XC8PAS /EXIT
3188
3189 7253 0000 PASSES, 0
3190
3191 7254 7575 TABLA, 100-"C
3192 7255 7571 100-"G
3193 7256 7557 100-"Q
3194 7257 7555 100-"S
3195 7260 0000 0000 /END OF TABLE
3196
3197 7261 7000 TABLB, CNTPLC
3198 7262 7013 CNTRLG
3199 7263 6673 CNTRLQ
3200 7264 6661 CNTRL5
3201
3202 7265 2322 MESA, TEXT "SR="
3203 7266 7500
7267 0516 MESB, TEXT "END OF PASS "
7270 0440
7271 1706
7272 4020
7273 0123
7274 2340
7275 0000
3204
3205 7367 7146
3206 7370 7076
```

```
3207 7371 7126
3208 7372 7071
3209 7373 7066
3210 7374 6756
3211 7375 7070
3212 7376 6754
3213 7377 7067
```

0000
0100

0200
0300

0400
0500

0600
0700

1000
1100

1200
1300

1400
1500

1600
1700

2000
2100

2200
2300

2400
2500

2600
2700

3000
3100

3200
3300

3400
3500

3600
3700

4000
4100

4200
4300

4400
4500

4600
4700

5000
5100

5200
5300

5400
5500

5600
5700

6000
6100

6200
6300

6400
6500

6600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
6700 11111111 11111111 11111111 11111111 11111111 11111111 10011111 11111111

7000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
7100 11111111 11111111 11111111 11111111 11111111 11111111 10000000 01111111

7200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111100
7300 00000000 00000000 00000000 00000000 00000000 00000000 00000001 11111111

7400
7500

7600
7700

A11	0011	C77	0032	DELLOP	2624	K4000	0123
A20	0020	C80100	6656	DEVCOO	0035	K6500	4031
A21	0021	C8CHAR	6760	DONEA	6622	K6520	3736
A22	0022	C8DCOD	7114	EHLTLP	4464	K7	3520
A23	0023	C8DO1	7103	END	1502	K7007	3463
ACNSKP	0545	C8K100	7160	ERREXT	7061	K7600	7007
ACTFLD	6677	C8K177	7144	ERRFLG	0044	K77	3435
AERROR	4512	C8K200	7145	ERRLOP	1510	K770	3462
APT8	4513	C8K212	7154	EXIT	3717	K7700	3536
APTCHK	3711	C8K215	7155	FDATAT	1400	KCC	6032
APTI00	6370	C8K240	7065	FDTLOP	1523	KCC0	0607
APTI01	6371	C8K260	7240	FILCHK	2646	KCC1	0712
APTI02	6372	C8K277	6660	FILERT	1645	KCC2	1036
APTI03	6373	C8K7	7071	FILEXT	2003	KCC3	1063
APTI04	6374	C8K77	7156	FILLER	3200	KCC4	1136
APTI05	6375	C8M212	7067	FILLOP	1667	KCC5	1757
ARROW	7011	C8M260	7072	FLDRTN	6641	KCC6	2000
B1050	5631	C8M270	7073	FOROCK	3464	KCDFIF	6657
B110	5514	C8M3	7070	FOROCT	4474	KCF	6030
B1200	5550	C8M4	6756	G	7012	KCF0	0320
B150	5523	C8M40	7157	GETDAT	6653	KCF1	0401
B19200	2540	C8OCT	7237	G0ITA	6643	KCF2	0501
B19K2	5604	C8RTRN	6701	G0TOA	6651	KCF3	0551
B2400	5557	C8STOR	7075	G0TOUV	3713	KCF4	1143
B300	5532	C8TEMP	7074	HALT	7402	KIE	6035
B4800	5566	CAF	6007	HLT	4520	KIE0	0403
B56P8	5613	CAFINT	0500	HLTLOP	2200	KIE1	0422
B600	5541	CAFXTM	0344	HLTLP1	2217	KIE10	1052
B66P7	5622	CALFLD	0000	HLTLP2	2241	KIE2	0457
B9600	5575	CHARLG	1600	INDEXA	6652	KIE3	0502
BAUDNO	0041	CHEK22	4515	INMODE	6702	KIE4	0536
BAUDST	2151	CHGMES	3120	INTON	2313	KIE5	0615
BAUDTB	2334	CHKFIL	2670	INTRET	0267	KIE6	0707
BAUDTM	2264	CHRLAS	1636	INTXMT	0400	KIE7	0747
BDPNTR	2333	CHRTNP	6757	K0125	0056	KIE8	0757
BGNINT	0200	CLKCNT	0121	K0252	0055	KIE9	1026
BITNO	0037	CLRRBD	0271	K10	3313	KRB	6036
BR1050	2604	CMESS6	4440	K100	3440	KRB0	0634
BR110	2400	CNT1	0050	K115	0026	KRB1	1113
BR1200	2460	CNT2	0051	K117	0027	KRB10	1711
BR150	2414	CNTREC	1737	K177	3372	KRB11	1351
BR2400	2474	CNTRLG	7000	K2	0025	KRB12	2014
BR300	2430	CNTRLG	7013	K200	3373	KRB13	2060
BR4800	2510	CNTRLQ	6673	K212	0057	KRB14	2070
BR568	2554	CNTRLS	6661	K215	3260	KRB2	1410
BR600	2444	CON100	2644	K240	3441	KRB3	1437
BR667	2570	CONFLD	0000	K260	3521	KRB4	1504
BR9600	2524	COUNT	0122	K3	3437	KRB5	1511
BSW	7002	CRLF	4502	K33	0030	KRB6	1526
BSWAP	4511	DATBIT	0040	K37	0031	KRB7	1544
C	7010	DFLAY	4461	K377	0034	KRB8	1615
C177	0033	DELYN	2643	K400	4122	KRB9	1654

KRS	6034	M43	3436	PSWRTN	6747	SPI9	0672
KRS0	0625	MES10A	5145	RAN1	3573	SR4HLT	2133
KRS1	1010	MES10B	5164	RAN2	3574	START	0250
KRS2	2040	MES10C	5203	RANDM1	4771	STENAB	2005
KRS3	2046	MES10D	5222	RANDOM	4506	STERR	2101
KSE	6035	MESA	7265	RECDAT	0045	STFLGS	0703
KSE0	1335	MESAGE	4471	RECEVE	1436	STLAS	2126
KSE1	2023	MESAGF	3416	RECPNT	5241	STLPPC	4462
KSE2	2045	MESAGX	3400	REDOA	6611	STRTST	0271
KSE3	2054	MESB	7267	PESGEN	4510	SW1ONE	4465
KSF	6031	MESS1	4203	RETINT	2317	TABLA	7254
KSF0	0276	MESS10	5120	RETURN	3756	TABLB	7261
KSF1	0336	MESS2	4213	RSTART	6755	TABLC	5500
KSF10	1037	MESS3	4224	SAV1	3575	TCF	6042
KSF11	1065	MESS3A	4236	SAV2	0053	TCF0	0330
KSF12	1077	MESS4	4241	SAVAC	2240	TCF1	0436
KSF14	1116	MESS4A	4302	SAVBTS	0036	TCF10	1505
KSF16	1137	MESS5	4306	SAVGEN	4507	TCF11	1512
KSF17	1144	MESS5A	4326	SAVIT	3164	TCF12	1611
KSF18	1425	MESS6	4400	SAVPNT	0024	TCF13	1656
KSF19	1541	MESS6A	4473	SCXMIT	0317	TCF14	1346
KSF2	0354	MESS6B	4520	SDTST1	1152	TCF15	2015
KSF21	1675	MESS6C	4545	SDTST2	1200	TCF16	2035
KSF22	1744	MESS6D	4572	SDTST3	1210	TCF2	0470
KSF23	1762	MESS6E	4617	SDTST4	1233	TCF3	0532
KSF24	1341	MESS6F	4645	SDTST5	1251	TCF4	0654
KSF25	2036	MESS6G	4673	SDTST6	1302	TCF5	0715
KSF26	2063	MESS6H	4721	SDTST7	1316	TCF6	1002
KSF3	0471	MESS6I	4747	SETUP	0215	TCF7	1064
KSF4	0537	MESS6J	4776	SLOLAS	1372	TCF8	1123
KSF5	0600	MESS6K	5024	SLOLST	1376	TCF9	1412
KSF6	0720	MESS6L	5052	SLWDAT	1333	TEMP	0124
KSF7	0733	MESS7	5100	SLWREC	1350	TEMSAV	4135
KSF8	2713	MESS7A	5115	SLWTL5	1340	TFL	6040
KSF9	1013	MESTAB	2350	SPACE2	4500	TFL0	0323
KSF8KP	4467	MESTB1	2364	SPACX2	3306	TFL1	0345
LAS	4517	MIOT	4503	SPI	6045	TFL2	0406
LFFLAG	6754	MPC	4200	SPI0	0304	TFL3	0452
LISN	4470	NDELAY	0047	SPI1	0411	TFL4	0503
LISN1	3340	NERROR	4516	SPI10	0736	TFL5	0520
LISN2	3356	NEWDEV	4000	SPI11	0750	TFL6	0645
LISN3	3351	NOINTR	0201	SPI12	0760	THSFLD	0000
LISNT1	0170	NOTBE	4050	SPI13	1015	TICK	4514
LISNUM	3361	ONAPT	3613	SPI14	1027	TL5	6046
LOAD	4460	ONEOCK	3201	SPI15	1042	TL50	1102
LOOP	4463	ONEOCT	4472	SPI2	0427	TL81	1515
LOOPPP	0046	ORIGIN	6600	SPI3	0442	TL810	1424
M10	4041	PASSES	7253	SPI4	0455	TL82	1522
M1000	0054	PRGFLD	0000	SPI5	0460	TL83	1540
M212	3374	PRNT1	4475	SPI6	0506	TL84	1556
M270	3367	PRNT2	4476	SPI7	0515	TL85	1606
M3	3375	PRNT4	4477	SPI8	0523	TL86	1665

TL87	2307	XAERRO	3664
TL88	2315	XAPT8	3600
TL89	2026	XBSW	3522
TMPCNT	7066	XC8CNT	6600
TPC	6044	XC8CRL	7146
TPC0	0726	XC8OCT	7215
TPC1	1073	XC8PAS	7241
TSF	6041	XC8PNT	7076
TSF0	0301	XC8PSW	6703
TSF1	0326	XC8TTY	7135
TSF10	0512	XC8TYP	7126
TSF11	0521	XCHK22	4113
TSF12	0533	XCNLTG	7020
TSF13	0663	XCRLF	3244
TSF14	0723	XDELAY	2620
TSF15	2675	XHALT	4072
TSF16	1000	XHLT	4101
TSF17	1003	XLAS	4063
TSF18	1070	XLISN	3314
TSF2	0333	XLOAD	2163
TSF20	1106	XMIOT	3442
TSF22	1124	XMIT	1420
TSF23	1421	XMTDAT	0042
TSF24	1516	XMTDT1	0043
TSF25	1535	XMTIOT	5352
TSF26	1553	XMTPEC	1057
TSF28	1672	XNERRO	3737
TSF29	1747	XOR	4504
TSF3	0346	XORS	3261
TSF30	2310	XPCRET	4032
TSF31	2322	XPRNT1	3513
TSF32	1344	XPRNT2	3224
TSF4	0351	XPRNT4	3500
TSF5	0407	XRAND	3537
TSF6	0425	XRAND1	4123
TSF7	0437	XRESGN	3564
TSF8	0453	XSAVGN	3555
TSF9	0504	XSR1LP	1564
TSFSKP	4466	XSR2	4042
TSCHA	7022	XTABLA	6654
TSCHT	0052	XTABLB	6655
TSICRL	7200	XTICK	4013
TSTEXT	7051	XTYPE	3235
TSRTN	6744	YESRNO	4505
TWOOCK	3211	YESRNX	3273
TWOOCKT	4473		
TVINTR	3000		
TYPE	4501		
UPDATE	1465		
US100	2645		
WATKSF	2707		
WATTSF	2671		

ERRORS DETECTED: 0

LINKS GENERATED: 81

RUN-TIME: 7 SECONDS

3K CORE USED

SEQ 0099

SEQ 0100

SEQ 0101

SEQ 0102

[illegible]

LOAD	120#	259	726	777	794	837	896	1032									
LOOP	126#	280	304	321	403	447	536	652	717	734	750	772	789	817	860	0104	
	832	847	965	1067	1173	1244	1422										
LOOPPP	108#	1335	1352	1376	2383	2397											
N10	2384	2388#															
N1000	114#	838	899														
N212	1986	2024#															
N270	2015	2019#															
N3	1989	2025#															
N43	2052	2064#															
MES10A	1455	1844	2512#														
MES10B	1456	2513#															
MES10C	1457	2514#															
MES10D	1458	2515#															
MESA	2931	3202#															
MESSAGE	138#	1730	1736	1742	1749	1755	1763	1770	1778	1785	1787	1809	1811	1819			
	1826	1843	1965	2298													
MSGAGF	2043	2045	2048#	2060	2062												
MSGAGX	139	2034#	2036	2038	2051	2072	2082										
MESB	3182	3203#															
MESS1	1731	2482#															
MESS10	1827	2510#															
MESS2	1737	2483#															
MESS3	1743	2484#															
MESS3A	1750	2485#															
MESS4	1756	2486#															
MESS4A	1764	2487#															
MESS5	1771	2488#															
MESS5A	1779	2489#															
MESS6	1786	2492#															
MESS6A	1441	1810	2495#														
MESS6B	1442	2496#															
MESS6C	1443	2497#															
MESS6D	1444	2498#															
MESS6E	1445	2499#															
MESS6F	1446	2500#															
MESS6G	1447	2501#															
MESS6H	1448	2502#															
MESS6I	1449	2504#															
MESS6J	1450	2505#															
MESS6K	1451	2506#															
MESS6L	1452	2507#															
MESS7	1812	2508#															
MESS7A	1820	2509#															
MESTAB	1441#	1805															

SEQ 0105

SEQ 0106

TFL5	427#	2598															
TFL6	507#	2599															
THSFLO	2815#	2818	2830														
TICK	176#	2315															
TL5	48#	678	857	911	915	974	979	993	1007	1037	1092	1203	1404	1410			
TL50	1908	3097															
TL51	678#	2665															
TL510	974#	2667															
TL52	915#	924	2676														
TL53	979#	2668															
TL54	993#	2669															
TL55	1007#	2670															
TL56	1037#	2671															
TL57	1092#	2672															
TL58	1404#	2673															
TL59	1410#	2674															
TL59	1203#	2675															
TMPCNT	2939	2951	3041	3052#	3160	3171											
TPC	45#	564	671														
TPCO	564#	2647															
TPC1	671#	2648															
TSF	43#	267	291	296	310	313	348	362	372	384	415	421	428	438			
	521	561	606	609	668	682	696	861	912	975	990	1004	1097	1143			
	1405	1415	1699	1910	3098												
TSF0	267#	2600															
TSF1	291#	2601															
TSF10	421#	2610															
TSF11	428#	2611															
TSF12	438#	2612															
TSF13	521#	2613															
TSF14	561#	2614															
TSF15	1699#	2615															
TSF16	598	606#	2616														
TSF17	609#	2617															
TSF18	668#	2618															
TSF2	296#	2602															
TSF20	682#	2619															
TSF22	696#	2620															
TSF23	912#	2621															
TSF24	975#	2622															
TSF25	990#	2623															
TSF26	1004#	2624															
TSF28	1097#	2625															
TSF29	1143#	2626															
TSF3	310#	2603															
TSF30	1405#	2627															
TSF31	1415#	2628															
TSF32	861#	2629															
TSF4	313#	2604															
TSF5	348#	2605															
TSF6	362#	2606															
TSF7	372#	2607															
TSF8	384#	2608															

SEQ 0107

TSF9	415#	2609															
TSFSKP	132#	567	673	693	1038	1205											
TSTCHA	2943	3016#	3037	3038	3050												
TSTCNT	112#	780	797	839	900	1398	1399	1401	1402	1689	2323	2385					
TSTCRL	2857	3023	3140#	3146	3149	3151	3152										
TSTEXT	3024	3039#															
TSTRTN	2954#	3043	3046														
TWOOCK	143	1881#	1883	1888	1890	1891	1896	1897	1901								
TWOOCK	142#	1732	1738	1789	2096	2102											
TYINTR	200	1730#															
TYPE	154#	1923	1928	1995	2059	2127											
UPDATE	932	934	937	948#	1018												
US100	1651	1652	1668#														
WATKSF	135	1710#	1721	1722	1723												
WATTSF	133	1695#	1706	1707	1708												
XAERRO	173	2242	2243	2246	2248	2261#	2265	2303	2304	2305							
XAPTR	175	2199#															
XBSW	171	2134#	2145														
XC8CNT	2322	2834#	2868	2897	2902												
XC8CRL	2866	2929	2953	2991	3009	3039	3049	3118#	3123	3185							
XC8OCT	2937	3157#	3174	3184													
XC8PAS	1281	3178#	3187														
XC8PNT	2930	3065#	3067	3069	3085	3181											
XC8PSW	232	1288	2302	2921#	2944	2959	3010										
XC8TTY	2901	2942	3106#	3112													
XC8TYP	2860	2863	2865	2941	2988	2990	3006	3008	3026	3048	3090	3096#	3102	3120			
	3122	3169															
XCHK22	179	2451#	2456	2457													
XCNTLG	3010#																
XCRIF	157	1920#	1931														
XDELAY	123	1646#	1660	1661	1662	1663											
XHALT	185	2428#	2434	2437	2442	2445											
XHLT	2433	2436#															
XLAS	183	2416#	2422	2429	2438												
XLISN	137	1976#	1996	1997	2006	2007	2009	2011									
XLOAD	121	1306#	1309	1311													
XMIOT	159	2070#	2073	2074													
XMIT	911#	960	2666														
XMTDAT	104#	725	728	730	746	755	757	759	761	764	766	769	781	783			
	784	798	800	803	804	808	811	812	824	828	841	856	871	879			
	907	910	930	938	953	973	987	992	1035	1047	1090	1091	1111	1116			
	1124	1198	1202	1207	1216	1222	1228	1240	1250	1263	1264	2466	2469	2472			
XMTDT1	105#	920	923	944	952	978	1001	1006	1050	1056	1087	1108	1134	1196			
	1208	2470															
XMTIOT	243	2593#															
XMTREC	659#	716															
XNERRO	181	2311#	2312	2314	2325												
XOR	160#																
XORS	161	1926	1929	1937#	1940	1944	1945	1946									
XPCRET	125	2380#	2382	2386													
XPRNT1	147	2005	2008	2124#	2128												
XPRNT2	149	1886	1889	1895#	1903												
XPRNT4	151	2079	2080	2083	2101	2104	2111#	2121									

SEQ 0108

SEQ 0109

SEQ 0110

.V0003	208	332#	486	601#	1239	1318#											
.V0004	216	330#	1109	1166#	1816	1822	1851#										
.V0017	225	329#	1804	1853#	2236	2336#											
.V0020	2933	2956	2974#														
.V0200	1284	1316#															
.V0203	2210	2340#															
.V0206	2209	2341#															
.V0215	1845	1848#	2254	2334#													
.V0267	246	325#															
.V0271	1290	1301	1314#	2281	2332#												
.V0400	322	324#															
.V0600	463	466#															
.V1000	598	600#	1775	1855#													
.V1031	210	331#															
.V1333	731	739#															
.V1400	848	2826#															
.V1600	966	2825#															
.V2000	1162	1164#	2221	2338#													
.V2003	1133	1165#															
.V2005	1079	1168#															
.V2217	458	468#	472	479	485	493	500	509	516	523	530	604#					
.V2241	461	467#	475	481	489	496	503	512	519	526	533	603#					
.V2264	196	334#															
.V2317	1387	1462#															
.V2334	1390	1461#															
.V2350	1805	1852#															
.V2364	1838	1849#															
.V2646	785	805	813	842	908	921	2827#										
.V3000	200	333#	2218	2339#													
.V3306	2026	2822#															
.V3513	2005	2008	2823#														
.V4000	1295	1313#	2280	2333#													
.V4060	546	660	897	2828#													
.V4402	1215	1227	1319#														
.V5402	244	326#	1385	1463#													
.V5500	2739	2335#															
.V6000	2231	2337#															
.V6600	2322	2330#															
.V6641	3011	3137#															
.V6660	3047	3132#															
.V6703	232	327#	1288	1315#	2302	2331#	3010	3138#									
.V6744	3043	3046	3134#														
.V6754	3145	3150	3212#														
.V6756	3159	3210#															
.V6757	3045	3133#															
.V7011	2859	2979#															
.V7022	2943	2969#															
.V7065	2940	2970#															
.V7066	2939	2951	2971#	3160	3171	3209#											
.V7067	3141	3213#															
.V7070	3147	3211#															
.V7071	3167	3208#															
.V7072	2853	2981#															

SEQ 0111

.V7075	2935	2936	2973#														
.V7076	2930	2975#	3181	3206#													
.V7126	2860	2863	2865	2941	2978#	3169	3207#										
.V7135	2901	2942	2976#														
.V7146	2866	2929	2953	2977#	3185	3205#											
.V7200	2857	2980#	3023	3135#													
.V7215	2937	2972#															
.V7241	1281	1317#															
.V7400	497	501	2829#														
.V7575	3019	3136#															
.V7600	1684	1726#															
.V7700	1094	1136	1152	1696	1711	2824#											
.V7765	228	328#	1793	1854#													
.V7766	1681	1727#															
.V7774	482	602#	1086	1167#													

SEQ 0112



